Stochastic Contextual Bandits with Long Horizon Rewards

Yuzhen Qin[1], Yingcong Li[1], Fabio Pasqualetti[1], Maryam Fazel[2], Samet Oymak[1,3]

[1] University of California, Riverside
[2] University of Washington
[3] University of Michigan
{yuzhenq, yli692}@ucr.edu, fabiopas@engr.ucr.edu, mfazel@uw.edu, oymak@umich.edu

Abstract

The growing interest in complex decision-making and language modeling problems highlights the importance of sample-efficient learning over very long horizons. This work takes a step in this direction by investigating contextual linear bandits where the current reward depends on at most s prior actions and contexts (not necessarily consecutive), up to a time horizon of h. In order to avoid polynomial dependence on h, we propose new algorithms that leverage sparsity to discover the dependence pattern and arm parameters jointly. We consider both the data-poor ($T < h$) and data-rich ($T \geq h$) regimes, and derive respective regret upper bounds $\tilde{O}(d\sqrt{sT} + \min\{q, T\})$ and $\tilde{O}(\sqrt{sdT})$, with sparsity s, feature dimension d, total time horizon T, and q that is adaptive to the reward dependence pattern. Complementing upper bounds, we also show that learning over a single trajectory brings inherent challenges: While the dependence pattern and arm parameters form a rank-1 matrix, circulant matrices are not isometric over rank-1 manifolds and sample complexity indeed benefits from the sparse reward dependence structure. Our results necessitate a new analysis to address long-range temporal dependencies across data and avoid polynomial dependence on the reward horizon h. Specifically, we utilize connections to the restricted isometry property of circulant matrices formed by dependent sub-Gaussian vectors and establish new guarantees that are also of independent interest.

Introduction

Multi-armed bandits (MAB) serve as a prototypical model to study exploration-exploitation trade-off in sequential decision-making (e.g., see Bubeck et al. (2012)). The agent needs to repeatedly make decisions by interacting with an unknown environment, aiming to maximize the cumulative reward. As a generalization of MAB, the contextual bandits allow the agent to take actions based on contextual information (Langford and Zhang 2007). Extensive studies have been conducted on contextual bandits due to its wide applications such as clinical trials, recommendation, and advertising (e.g., see Woodroofe (1979); Chu et al. (2011); Li et al. (2017, 2010); Qin et al. (2022a,b)).

Most existing work on contextual bandits assume that each reward only depends on a single action and the associated context. This action can be the one just taken (instantaneous reward) or the one taken a certain number of steps before (delayed rewards). However, in realistic decision-making scenarios, the reward generating process can have a more complex, non-Markovian nature. Multiple prior actions can jointly affect the current reward. For instance, whether a learner will take a course recommended by an online education platform depends not only on that course, but also on what combination of courses they have taken before. Recommending courses in a complicated curriculum to users with diverse backgrounds and past experiences requires accounting for the combined effects of past contexts on the current recommendation. Similarly, the attention mechanism (Vaswani et al. 2017) is finding increasing success in reinforcement learning and NLP applications (Chen et al. 2021; Brown et al. 2020) and it makes predictions by assessing the similarities between current and past contexts (e.g., that correspond to words in a sentence or frames in a video game) and creating a history-weighted adaptive context. In connection to this, the benefit of using a long context history has been well acknowledged in RL and control theory (e.g. frame/state stacking practice (Hessel et al. 2018)).

These observations motivates the following central questions: *Can we provably and efficiently learn from long-horizon rewards? What is the role of reward dependence structure in sample efficiency?* In this work, we thoroughly address these questions for a novel variation of stochastic linear contextual bandits, where the current reward depends on a subset of prior contexts, up to a time horizon of h (see Fig. 1 for an illustration). Specifically, the reward is determined by a *filtered context* that is a linear combination of prior h selected contexts. Moreover, inspired by practical decision making scenarios, we consider sparse interactions where only s ($s \ll h$) of h prior contexts actually contributing to the current reward. Here $s = 1$ corresponds to the special instance of delayed rewards. Crucially, we develop strategies that leverage this sparse dependence structure and establish regret guarantees for long horizon rewards.

Related Work

Composite anonymous rewards. Pike-Burke et al. (2018) considered bandits with composite anonymous rewards, where 1) the reward that the agent receives at each round is the sum of the delayed rewards of an unknown subset of past actions, and 2) individual contributions of past

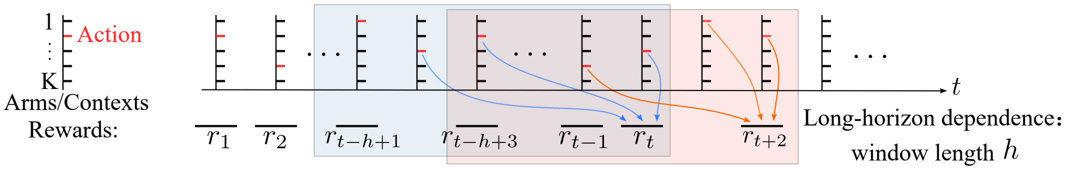

Figure 1: Contextual Bandits with Long-Horizon Rewards. The reward at each round t depends on the contexts associated with latest h actions (h can be very large).

actions to the reward are not discernible. Cesa-Bianchi et al. (2018) generalized this setting to a case where the reward generated by an action is not simply revealed to the agent at a single instant in the future, but rather spreads over multiple rounds. Recent work along this line is also found in Garg and Akash (2019), Zhang et al. (2022), and Wang et al. (2021). In this paper, we consider a contextual setting with, which is different from the above ones and poses new challenges since each arm no longer has a fixed reward distribution.

Delayed rewards. Bandits with delayed rewards are also related to our work. Stochastic linear bandits with random delayed rewards was studied in Vernade et al. (2020). Li et al. (2019) investigated the case where the delay is unknown. Generalized stochastic linear bandits with random delays were studied in Zhou et al. (2019) and Howson et al. (2022). Cella and Cesa-Bianchi (2020) and Lancewicki et al. (2021) studied bandit problems with reward-dependent delays. In Lancewicki et al. (2021) and Thune et al. (2019), delays are allowed to be unrestricted. Arm-dependent delays in stochastic bandits are studied in Gael et al. (2020). Delays are also considered in adversarial bandits (Bistritz et al. 2019; Gyorgy and Joulani 2021; Zimmert and Seldin 2020). Recently, non-stochastic cooperative linear bandits with delays have also been studied (Ito et al. 2020; Cesa-Bianchi et al. 2019). In fact, our setting captures unknown fixed delays and also aggregated and anonymous delayed rewards.

Sparse parameters. In sparse bandits, feature vectors can have large dimension d, but only a small subset, $s \ll d$, of them affect rewards. Early studies on sparse *linear bandits* are found in Carpentier and Munos (2012) and Abbasi-Yadkori et al. (2012). Recent results studied both the data-poor and data-rich regimes, depending on whether the total horizon T is less or larger than d. In the data-rich regime, Lattimore and Szepesvári (2020) proved a regret lower bound $\Omega(\sqrt{sdT})$. In the data-poor regime, Hao et al. (2020) showed a regret lower bound $\Omega(s^{\frac{1}{3}}T^{\frac{2}{3}})$. A recent work used information-directed sampling techniques Hao et al. (2021). Sparse *contextual linear bandits* also receive increasing interests. Kim and Paik (2019) proposed an algorithm that combines Lasso with doubly-robust techniques, and provided an upper bound $O(s\log(dT)\sqrt{T})$. An extended setting wherein each arm has its own parameter was studied in Bastani and Bayati (2020), Wang et al. (2018), where upper bounds $O(s^2\log^2(T))$ and $O(s^2\log(T))$ were shown, respectively. Oh et al. (2021) proposed an exploration-free algorithm and obtained an upper bound $O(s^2\log(d) + s\sqrt{T\log(dT)})$. In Ariu et al. (2022), a thresholded Lasso algorithm is presented, resulting in an upper bound $O(s^2\log(d) + \sqrt{sT})$. In Ren and Zhou (2020), the dynamic batch learning approach was used and a upper bound $O(s\cdot\text{polylog}(d) + \log(T)\sqrt{sT\log(d)})$ was obtained. In comparison, sparsity in our case results from the reward dependence structure. As we will discuss in Sec. , learning the dependence pattern is challenging since the measurements have an inherent circulant structure.

Online Convex Optimization (OCO). Another line of research related to ours is OCO with memory where the losses depend on the past decisions taken from a convex set (Anava et al. 2015; Shi et al. 2020; Kumar et al. 2022).

Contributions

The contributions of this paper are summarized as follows:

1. We introduce a new contextual bandit model, motivated by realistic scenarios where rewards have a long-range and sparse dependence on prior actions and contexts. The problem of identifying the reward parameter and sparse delay pattern admits a special low-rank and sparse structure.

2. We propose two sample-efficient algorithms for the data-poor and data-rich regimes by leveraging sparsity prior. For the former, we prove a regret upper bound $O\big(d\sqrt{sT\log(dT)} + \min\{q,T\}\big)$ that is adaptive to the reward dependence pattern described by q; for the latter, we obtain a regret upper bound $O(\sqrt{sdT\log(dT)})$. Note that neither of the bounds has polynominal dependence on the horizon h, enabling efficient learning across long horizons; and both are optimal in T (up to logarithmic factors).

3. We make technical contributions to address temporal dependencies within data that has a block-Toeplitz/circulant matrix form. First, the seminal work by Krahmer et al. (2014) on Restricted Isometry Property (RIP) of circulant matrices assume context vectors have i.i.d. entries. We generalize their result to milder concentrability conditions that allow dependencies. Second, we establish results that highlight the challenges of low-rank estimation unique to circulant measurements. In line with theory, numerical experiments demonstrate that our sparsity-based approach indeed outperforms low-rank ones.

Problem Setting

Notation. Given $\boldsymbol{x} = [\boldsymbol{x}_1^\top, \dots, \boldsymbol{x}_h^\top]^\top \in \mathbb{R}^{dh}$ with each block $\boldsymbol{x}_i \in \mathbb{R}^d$, denote $\|\boldsymbol{x}\|_{2,1}^{(d)} := \sum_{i=1}^h \|\boldsymbol{x}_i\|_2$; for $\boldsymbol{A} \in \mathbb{R}^{m\times n}$, $\|\boldsymbol{A}\|_{\text{op}} := \sup_{\|\boldsymbol{x}\|_2 \le 1} \|\boldsymbol{A}\boldsymbol{x}\|_2$ denotes its operator norm. Let $[n] = \{1, 2, \dots, n\}$ for any integer n. For any $\mathcal{S} \subset [n]$, $\boldsymbol{x}_{\mathcal{S}}$ denotes the sub-vector of $\boldsymbol{x}$ with entries indexed by $\mathcal{S}$. Let $\langle \cdot, \cdot \rangle$ be the inner product; for $\boldsymbol{A}$ and $\boldsymbol{B} \in \mathbb{R}^{m\times n}$, $\langle \boldsymbol{A}, \boldsymbol{B} \rangle = \text{Tr}(\boldsymbol{A}^\top \boldsymbol{B})$. Let $\otimes$ be the Kronecker product. Given $\boldsymbol{A} \in \mathbb{R}^{m \times p}$, it is said to satisfy RIP if there is $\delta \in (0,1)$ such that $(1-\delta)\|\boldsymbol{x}\|_2^2 \leq \|\boldsymbol{A}\boldsymbol{x}\|_2^2 \leq (1+\delta)\|\boldsymbol{x}\|_2^2$ holds for all $\boldsymbol{x} \in \mathbb{R}^p$; the smallest δ satisfying this inequality is called the RIP constant (see Appendix for more details).

Stochastic Linear Contextual Bandits

In this paper, we study a stochastic linear contextual bandit problem with rewards that depend on past actions and contexts (see Fig. 1 for an illustration). Let K be the number of arms, and then the action set is $[K]$. At each round t, the agent observes K context vectors, $\{\boldsymbol{x}_{t,a} \in \mathbb{R}^d : a \in [K]\}$, each associated with an arm and drawn i.i.d. from an unknown distribution ν. It then selects an action $a_t \in [K]$ and receives a reward generated by

$$r_t = \langle \boldsymbol{y}_{t,a_t}, \boldsymbol{\theta} \rangle + \varepsilon_t, \tag{1}$$

$$\text{where} \qquad \boldsymbol{y}_{t,a_t} = \sum\nolimits_{i=0}^{h-1} w_i \boldsymbol{x}_{t-i,a_{t-i}}, \text{ and } t \in [T].$$

Here $\boldsymbol{\theta} \in \mathbb{R}^d$ is the coefficient vector, $\varepsilon_t \in \mathbb{R}$ is additive noise that is zero-mean 1-sub-Gaussian. Particularly, the vector $\boldsymbol{y}_{t,a_t}$ is the *filtered context*, determined by the weight vector $\boldsymbol{w} := [w_0, w_1, \ldots, w_{h-1}]^\top \in \mathbb{R}^h$ that describes how rewards depend on the past and current selected contexts (where $w_i \geq 0$). The range of the dependence h can be very large, indicating that a reward can have a long-range contextual dependence. Assume $\boldsymbol{x}_{j,a_j} = 0$ for $j \leq 0$ since a_j's in this case correspond to nonexistent actions.

In this paper, we consider sparse contextual dependence, that is, the weight vector $\boldsymbol{w}$ is *s-sparse* (i.e., $\|\boldsymbol{w}\|_0 \leq s$) with $s \ll h$. This is particularly relevant to many realistic situations since often only a small number of past "events" matter. As we mentioned before, this setting captures: a) bandits with unknown delays ($\boldsymbol{w}$ has only one non-zero entry and it is 1-valued), and b) bandits with aggregated and anonymous rewards (all the non-zero entries of $\boldsymbol{w}$ are 1-valued).

The coefficient vector $\boldsymbol{\theta} \in \mathbb{R}^d$ and the weight vector $\boldsymbol{w} \in \mathbb{R}^h$ are unknown. Without loss of generality, we assume that $\|\boldsymbol{\theta}\|_2 \leq 1$ and $\|\boldsymbol{w}\|_1 \leq 1$. We also made the mild boundedness assumption that $\boldsymbol{x}_{t,a}$ satisfies $\|\boldsymbol{x}_{t,a}\|_\infty \leq 1$ for all $t \in [T]$ and $a \in [K]$.

The agent's objective is to maximize the cumulative reward over the course of T rounds, or equivalently, to minimize the pseudo-regret defined as

$$R_T = \left[\sum_{t=1}^{T} \sum_{i=0}^{h-1} w_i \left(\langle \boldsymbol{x}_{t,a_t^*}, \boldsymbol{\theta} \rangle - \langle \boldsymbol{x}_{t,a_t}, \boldsymbol{\theta} \rangle \right) \right], \tag{2}$$

where $a_t^* = \arg\max_{a \in [K]} \langle \boldsymbol{x}_{t,a}, \boldsymbol{\theta} \rangle$ defines the optimal action at round t.

Remark 1. The definition of the regret here is slightly different from simply summing up $\langle \boldsymbol{y}_{t,a_t^*} - \boldsymbol{y}_{t,a_t}, \boldsymbol{\theta} \rangle$. In fact, the two regret definitions are essentially the same. The reason is that taking an action, say a_t, gives the agent a total reward $\sum_{i=0}^{h-1} w_i \langle \boldsymbol{x}_{t,a_t}, \boldsymbol{\theta} \rangle$ that spreads over the next h rounds. Thus, the agent can make decisions without knowing $\boldsymbol{w}$ if $\boldsymbol{\theta}$ is known as a prior: a greedy strategy seeking to maximize the instantaneous reward at each round actually maximizes the cumulative reward in the long run. See Appendix F.1 in Qin et al. (2023) (the full version) for further discussion . △

Remark 2. Although the only knowledge of $\boldsymbol{\theta}$ seems sufficient for our decision-making purpose, learning $\boldsymbol{\theta}$ is actually challenging. This is because each reward can come in a composite manner, possibly consisting of the contributions from the latest h actions. Learning $\boldsymbol{\theta}$ requires to sort out the reward dependence structure. △

Discussion of Challenges

Next, we discuss some technical challenges inherent in our problem. For this purpose, we denote $\boldsymbol{\xi}_t = \boldsymbol{x}_{t,a_t}$ as the chosen context for each t to make notation concise.

Circulant design matrices in low-rank matrix recovery. Let $\boldsymbol{Z}_t = [\boldsymbol{\xi}_t, \boldsymbol{\xi}_{t-1}, \ldots, \boldsymbol{\xi}_{t-h+1}] \in \mathbb{R}^{d \times h}$, and then (1) can be rewritten as

$$r_t = \langle \boldsymbol{Z}_t, \boldsymbol{\theta}\boldsymbol{w}^\top \rangle + \varepsilon_t. \tag{3}$$

At first glance, it seems that the problem reduces to reconstructing the rank-1 and sparse matrix $\boldsymbol{\theta}\boldsymbol{w}^\top \in \mathbb{R}^{d \times h}$, and classic techniques for low-rank (and sparse) matrix recovery can be applied (e.g., Richard et al. (2012), Oymak et al. (2015) Davenport and Romberg (2016), and Wainwright (2019, Chap. 10)). However, we find this is *not* true due to the *Toeplitz/circulant structure* of the design matrices Z_t. The following lemma shows that circulant matrices, even if its first row has i.i.d. entries, do not obey RIP for rank-1 matrices with exponentially high probability (see Appendix C in Qin et al. (2023) for more details).

Lemma 1. Let $\mathbf{C} \in \mathbb{R}^{n \times p}$ be a subsampled circulant matrix whose first row has i.i.d. Gaussian entries (normalized properly) and $n \leq p$. For any $\delta \leq 1$, there exists a constant $c < 1$ such that with probability at least $1 - c^p$, $\mathbf{C}$ does not obey RIP over rank-1 matrices in $\mathbb{R}^{p_1 \times p_2}$ with $p = p_1 p_2$.

We further provide numerical experiments in Fig. 2 to show that circulant measurements are indeed problematic while dealing with low-rank matrix recovery.

These findings indicate that tackling our problem via low-rank matrix estimation may not work. Therefore, we resort to another technique– *sparsity estimation*– by leveraging the sparse structure in the reward dependence pattern. First, let $\boldsymbol{a}_t = \{a_t, a_{t-1}, \ldots, a_1\}$ denote the sequence of past actions, $\boldsymbol{z}_{\boldsymbol{a}_t} = [\boldsymbol{\xi}_t^\top, \boldsymbol{\xi}_{t-1}^\top, \ldots, \boldsymbol{\xi}_{t-h+1}^\top]^\top$, and $\boldsymbol{\phi} = \boldsymbol{w} \otimes \boldsymbol{\theta} \in \mathbb{R}^{dh}$. Then, (1) can be rewritten as

$$r_t = \langle \boldsymbol{z}_{\boldsymbol{a}_t}, \boldsymbol{\phi} \rangle + \varepsilon_t. \tag{4}$$

Since $\boldsymbol{w}$ is s-sparse, reconstructing $\boldsymbol{\theta}$ and $\boldsymbol{w}$ becomes to estimate the s-block-sparse vector $\boldsymbol{\phi}$.

Denote $\boldsymbol{r}_t = [r_1, \ldots, r_t]^\top$ and $\boldsymbol{\varepsilon}_t = [\varepsilon_1, \ldots, \varepsilon_t]^\top$, and it follows from (4) that

$$\boldsymbol{r}_t = \begin{bmatrix} \boldsymbol{\xi}_1^\top & 0 & \cdots & 0 \\ \boldsymbol{\xi}_2^\top & \boldsymbol{\xi}_1^\top & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{\xi}_h^\top & \boldsymbol{\xi}_{h-1}^\top & \cdots & \boldsymbol{\xi}_1^\top \\ \boldsymbol{\xi}_{h+1}^\top & \boldsymbol{\xi}_h^\top & \cdots & \boldsymbol{\xi}_2^\top \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{\xi}_t^\top & \boldsymbol{\xi}_{t-1}^\top & \cdots & \boldsymbol{\xi}_{t-h+1}^\top \end{bmatrix} \boldsymbol{\phi} + \boldsymbol{\varepsilon}_t := \Xi_t \boldsymbol{\phi} + \boldsymbol{\varepsilon}_t. \tag{5}$$

One can observe that the design matrix Ξ_t above also has a *Toeplitz/circulant* structure. Learning the block-sparse $\boldsymbol{\phi}$ using this special form of design matrices has some other challenges, which we discuss below.

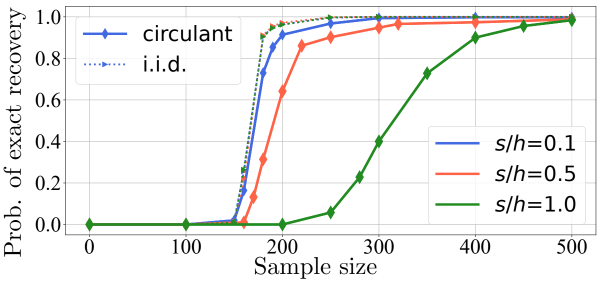

Figure 2: Probability of exact recovery of the matrix $\Phi = \boldsymbol{\theta}\boldsymbol{w}^\top \in \mathbb{R}^{d\times h}$ from the noiseless measurement $\boldsymbol{y} = \langle \boldsymbol{Z}, \Phi\rangle$ using low-rank recovery ($d = 10$ and $h = 100$). We compare two types of design matrices: 1) $\boldsymbol{Z}$ with i.i.d. entries, and 2) circulant $\boldsymbol{Z}$ generated by an i.i.d. vector. Different instances, where $\boldsymbol{w}$ has different sparsity (quantified by s), are considered. The experiment shows that: 1) the same amount of data is needed for all instances if $\boldsymbol{Z}$ has i.i.d. entries, but 2) the number of samples needed varies significantly when it comes to circulant $\boldsymbol{Z}$. These observations indicate that for circulant measurements, the amount of data needed to recover a low-rank matrix may not simply depend on the rank, which is substantially different from i.i.d. measurements.

Circulant matrices with dependent entries. Estimating $\boldsymbol{\phi}$ is a sparse regression problem. RIP and related restricted eigenvalue condition (REC) are widely used for such problems (Candes and Tao 2007; Bickel et al. 2009). Earlier studies show that sub-sampled circulant matrices whose first row is i.i.d. sub-Gaussian satisfy RIP for s-sparse vectors if there are at least $\tilde{\Omega}(s\log^2(s))$ samples (Krahmer et al. 2014). In our case, the circulant matrix is generated by random vectors with *dependent* entries (i.e., entries in each $\boldsymbol{\xi}_t$ may be dependent). The new challenge is: how many samples are needed for such circulant measurements to satisfy RIP/REC?

Technical Result

We first present a technical result on RIP that paves the way for the analysis of our bandit problem, which is also of independent interest (see Qin et al. (2023) for the proof).

Theorem 1. Let $\boldsymbol{\xi}_1, \dots, \boldsymbol{\xi}_n \in \mathbb{R}^d$ be independent sub-Gaussian isotropic random vectors. Assume that each $\boldsymbol{\xi}_i$ satisfies the Hanson-Wright inequality (HWI)

$$\begin{aligned} &\Pr\left[|\boldsymbol{\xi}_i^\top \boldsymbol{A}\boldsymbol{\xi}_i - \mathbb{E}(\boldsymbol{\xi}_i^\top \boldsymbol{A}\boldsymbol{\xi}_i)| \geq \eta\right] \\ &\leq 2\exp\Big(-\frac{1}{c}\min\Big\{\frac{\eta^2}{k^4\|\boldsymbol{A}\|_F^2}, \frac{\eta}{k^2\|\boldsymbol{A}\|_{\text{op}}}\Big\}\Big), \quad \forall \eta > 0, \end{aligned} \tag{6}$$

for any positive semi-definite matrix $\boldsymbol{A} \in \mathbb{R}^{d\times d}$, where k is a constant, and c is an absolute constant. Let $\boldsymbol{\Xi} \in \mathbb{R}^{m\times nd}$ be a matrix formed by sub-sampling *any* m rows of the block-circulant matrix given by

$$\boldsymbol{C} = \begin{bmatrix} \boldsymbol{\xi}_n^\top & \boldsymbol{\xi}_{n-1}^\top & \cdots & \boldsymbol{\xi}_1^\top \\ \boldsymbol{\xi}_1^\top & \boldsymbol{\xi}_n^\top & \cdots & \boldsymbol{\xi}_2^\top \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{\xi}_{n-1}^\top & \boldsymbol{\xi}_{n-2}^\top & \cdots & \boldsymbol{\xi}_n^\top \end{bmatrix}.$$

Then, for all s-sparse vectors, the restricted isometry constant of Ξ, denoted by δ_s, satisfies $\delta_s \leq \delta$ if $m \geq c_1\delta^{-2}s\log^2(s)\log^2(nd)$ for some constant c_1.

Remark 3. Although we just need to reconstruct a block-sparse vector in the bandit problem, this theorem applies to *general sparse* vectors. The assumption (6) holds for many random vectors, e.g., sub-Gaussian vectors with independent entries and random vectors that obey convex concentration property (Adamczak 2015).

Algorithms and Main Results

Next, we present some algorithms for the bandit problem described in (1), taking into account data-poor and data-rich regimes, accompanied with their regret bounds. First, we make the following assumption.

Assumption 1. We assume that the distribution ν is such that for all t: (a) the context vectors, $\boldsymbol{x}_{t,a}, a \in [K]$, are i.i.d., (b) $\mathbb{E}_{\boldsymbol{x}} = \mathbb{E}[\frac{1}{K}\sum_{a=1}^K \boldsymbol{x}_{t,a}\boldsymbol{x}_{t,a}^\top]$ satisfies $\lambda_{\min}(E_{\boldsymbol{x}}) \geq \sigma^2$ for some σ, and (c) $\boldsymbol{x}_{t,a}$ satisfies HWI given by (6).

Remark 4. If we let ξ_t be a random vector deterministically chosen from the set $\{x_{a,t}, a \in [K]\}$, Assumption 1 ensures that ξ_t also satisfies the Hanson-Wright inequality (see Appendix D for the proof). We will use this property with Theorem 1 to analyze our bandit algorithms. △

Data-Poor Regime

First, we consider the situation where the dimension of the weight vector $\boldsymbol{w}$ is larger than the number of rounds (i.e., $T < h$). In this data-poor regime, one can observe from (5) that it is impossible to reconstruct $\boldsymbol{\phi}$. Fortunately, it is not necessary to completely learn $\boldsymbol{\phi}$ to guide the decision-making; instead, a good estimate of $\boldsymbol{\theta}$ is sufficient (see the definition of regret in (2) for the reason). Thus, we propose the following approach to *partially* and *gradually* learn $\boldsymbol{\phi}$ such that $\boldsymbol{\theta}$ can be estimated and exploited at an early stage.

Approach 1. Recall that $\boldsymbol{\phi} = \boldsymbol{w} \otimes \boldsymbol{\theta}$ with $\boldsymbol{w} \in \mathbb{R}^h$. For any integer $k \leq h$, let $\boldsymbol{w}_{[k]} = [w_1, \dots, w_k]^\top$ and $\boldsymbol{\phi}_{\mathcal{K}} = \boldsymbol{w}_{[k]} \otimes \boldsymbol{\theta} \in \mathbb{R}^{kd}$. Then, if one has learned $\hat{\boldsymbol{\phi}}_{\mathcal{K}}$ as an estimate of $\phi_{\mathcal{K}}$, $\boldsymbol{\theta}$ can be estimated in the following steps:

(a1) Transform vector $\hat{\boldsymbol{\phi}}_{\mathcal{K}}$ into a matrix $\hat{\boldsymbol{\Phi}}_{\mathcal{K}} \in \mathbb{R}^{d\times k}$ (the ith column of $\hat{\Phi}_{\mathcal{K}}$ is the ith block of $\hat{\boldsymbol{\phi}}_{\mathcal{K}}$ with size d.

(a2) Let $\hat{\boldsymbol{\theta}}$ be the left singular vector of $\hat{\Phi}_{\mathcal{K}}$ that is associated with the largest singular value[1]. △

With this approach, we use a doubling trick to design our algorithm (see Algorithm 1 and Fig. 3). We select a constant L satisfying $s \leq L < h$ and define a sequence of sets $\mathcal{S}_1, \mathcal{S}_2, \dots, \mathcal{S}_m$ with growing number of elements, where $\mathcal{S}_i = [2^{i-1}Ld]$. Then, we aim to estimate

$$\boldsymbol{\phi}_{\mathcal{S}_1}, \boldsymbol{\phi}_{\mathcal{S}_2}, \boldsymbol{\phi}_{\mathcal{S}_3}, \dots, \boldsymbol{\phi}_{\mathcal{S}_m},$$

in *sequential epochs*, where $\boldsymbol{\phi}_{\mathcal{S}_i} \in \mathbb{R}^{2^{i-1}Ld}$ contains the first $2^{i-1}L$ blocks of $\boldsymbol{\phi}$ and m is the largest integer such that $2^{m-1}L \leq h$. The main idea is to learn a small portion of $\boldsymbol{\phi}$

[1] We point out that what we estimate in this step is not exactly $\boldsymbol{\theta}$, but rather its direction $\boldsymbol{\theta}/\|\boldsymbol{\theta}\|$. As it turns out later, a good estimate of the direction ensures a small angle between $\hat{\boldsymbol{\theta}}$ and $\boldsymbol{\theta}$, and is thus sufficient for good decision-making.

Algorithm 1: Doubling Lasso

```
Input: parameter $L$, the doubling sequence $\{T_i\}$ with $T_i = 4(2^i - 1)L$ [see Fig. 3 (a)], $\hat{\boldsymbol{\theta}}_0 = 0$.
for $t = 1 : T$ do
    Observe context vectors $\{\boldsymbol{x}_{t,a} : a \in [K]\}$.
    Take the greedy action $a_t \in \sup_{a\in[K]}\langle \boldsymbol{x}_{t,a}, \hat{\boldsymbol{\theta}}_{i-1}\rangle$, and receive a reward $r_t$.
    if $t = T_i$ then
        # end of the ith epoch, estimate a new $\hat{\theta}$
        Calculate $\hat{\phi}_{2^{i-1}L}$ according to the Lasso (11).
        Let $\hat{\boldsymbol{\theta}}_i$ be the singular vector of $\hat{\Phi}_{\mathcal{S}_i}$ associated with the largest singular value.
    end if
end for
```

when there is little data; as more data is collected, we learn a progressively larger portion.

At each epoch i, the following greedy action is repeatedly taken for $2^{i+1}L$ times (*doubling trick*, see Fig. 3 (a)):

$$a_t \in \arg\max_{a\in[K]}\langle \boldsymbol{x}_{t,a}, \hat{\boldsymbol{\theta}}_{i-1}\rangle, \qquad i \geq 1, \tag{7}$$

where $\hat{\boldsymbol{\theta}}_{i-1}$ is the estimate of $\boldsymbol{\theta}$ at the $(i-1)$th epoch ($\hat{\boldsymbol{\theta}}_0 = 0$). If there are more than one greedy actions, the agent uniformly randomly picks one. Then, we collect $2^{i+1}L$ data points generated by (5). However, we only use half of them to learn $\hat{\boldsymbol{\theta}}_i$. Specifically, dividing the data into four $2^{i-1}L$-dimensional chucks, we use the second and the fourth chucks (see Fig. 3 (a)). From (5), the rewards in these two chucks are respectively generated by

$$\boldsymbol{r}'[i] = \boldsymbol{\Xi}'[i]\boldsymbol{\phi} + \boldsymbol{\varepsilon}'[i], \quad \boldsymbol{r}''[i] = \boldsymbol{\Xi}''[i]\boldsymbol{\phi} + \boldsymbol{\varepsilon}''[i], \tag{8}$$

where $\boldsymbol{r}'[i], \boldsymbol{r}''[i] \in \mathbb{R}^{2^{i-1}L}$, $\boldsymbol{\Xi}'[i], \boldsymbol{\Xi}''[i] \in \mathbb{R}^{2^{i-1}L\times dh}$, and $\boldsymbol{\varepsilon}'[i], \boldsymbol{\varepsilon}''[i] \in \mathbb{R}^{2^{i-1}L}$ are the corresponding reward vectors, context matrices, and noise vectors.

Rewrite $\boldsymbol{\phi} = [\boldsymbol{\phi}_{\mathcal{S}_i}^\top, \boldsymbol{\phi}_{\bar{\mathcal{S}}_i}^\top]^\top$, where $\boldsymbol{\phi}_{\mathcal{S}_i} \in \mathbb{R}^{2^{i-1}Ld}$ is what we want to learn. Then, one can rewrite (8) into

$$\begin{bmatrix}\boldsymbol{r}'[i]\\ \boldsymbol{r}''[i]\end{bmatrix} = \begin{bmatrix}\boldsymbol{P}_i' & \boldsymbol{Q}_i'\\ \boldsymbol{P}_i'' & \boldsymbol{Q}_i''\end{bmatrix}\begin{bmatrix}\boldsymbol{\phi}_{\mathcal{S}_i}\\ \boldsymbol{\phi}_{\bar{\mathcal{S}}_i}\end{bmatrix} + \begin{bmatrix}\boldsymbol{\varepsilon}'[i]\\ \boldsymbol{\varepsilon}''[i]\end{bmatrix}, \tag{9}$$

where $\boldsymbol{r}'[i] \in \mathbb{R}^{2^{i-1}L}$, and $\boldsymbol{\Xi}'[i] = [\boldsymbol{P}_i', \boldsymbol{Q}_i']$ and $\boldsymbol{\Xi}''[i] = [\boldsymbol{P}_i'', \boldsymbol{Q}_i'']$ (see Fig. 3 (b) for an illustration). To learn $\boldsymbol{\phi}_{\mathcal{S}_i}$, let $\bar{\boldsymbol{r}}[i] = \boldsymbol{r}''[i] - \boldsymbol{r}'[i]$, $\bar{\boldsymbol{P}}_i = \boldsymbol{P}_i'' - \boldsymbol{P}_i'$, $\bar{\boldsymbol{Q}}_i = \boldsymbol{Q}_i'' - \boldsymbol{Q}_i'$, and $\bar{\varepsilon}[i] = \varepsilon''[i] - \varepsilon'[i]$, and then we have

$$\bar{\boldsymbol{r}}[i] = \bar{\boldsymbol{P}}_i\boldsymbol{\phi}_{\mathcal{S}_i} + \bar{\boldsymbol{Q}}_i\boldsymbol{\phi}_{\bar{\mathcal{S}}_i} + \bar{\varepsilon}[i] := \bar{\boldsymbol{P}}_i\boldsymbol{\phi}_{\mathcal{S}_i} + \boldsymbol{\epsilon}[i], \tag{10}$$

where the $\bar{\boldsymbol{Q}}_i\boldsymbol{\phi}_{\bar{\mathcal{S}}_i} + \bar{\varepsilon}[i]$ is taken as the new noise $\boldsymbol{\epsilon}[i]$.

Then, $\boldsymbol{\phi}_{\mathcal{S}_i}$ (which is at most s-block-sparse since $\boldsymbol{\phi}$ is) is estimated by solving the block-sparsity-recovery Lasso:

$$\hat{\boldsymbol{\phi}}_{\mathcal{S}_i} = \underset{\tilde{\boldsymbol{\phi}}\in\mathbb{R}^{2^{i-1}Ld}}{\arg\min}\left(\frac{1}{2^iL}\left\|\bar{\boldsymbol{P}}_i\tilde{\boldsymbol{\phi}} - \bar{\boldsymbol{r}}[i]\right\|_2^2 + \lambda_i\|\tilde{\boldsymbol{\phi}}\|_{2,1}^{(d)}\right), \tag{11}$$

where the regularization parameter is selected as

$$\lambda_i = cd\sqrt{\frac{2\log(2^idL/\gamma)}{2^{i-1}L}} \tag{12}$$

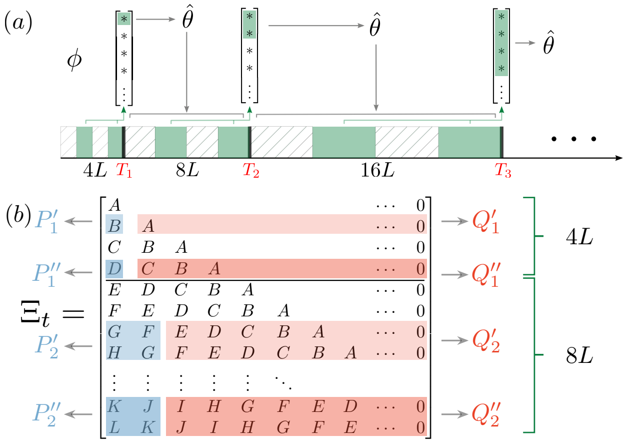

Figure 3: Illustration of Doubling Lasso. (a) In each epoch i, we play the greedy action in (7) for $2^{i+1}L$ rounds (T_i's are the epochs' ends). Then, we use the second and fourth quarters of the collected data (green areas) to learn the first $2^{i-1}L$ block of $\boldsymbol{\phi}$ and estimate $\boldsymbol{\theta}$ subsequently (see Approach 1). The learned $\hat{\boldsymbol{\theta}}$ is used for decision-making in the next epoch that has double length. (b) Illustration of how matrices in (9) are defined. Here, the matrix $\boldsymbol{\Xi}_t$ defined in (5) is represented in a form of $(L \times Ld)$-dimensional block matrices.

for some $c > 0$. Subsequently, we use Approach 1 to estimate $\boldsymbol{\theta}$. The algorithm is presented in Algorithm 1. The following theorem provides a regret upper bound for it.

Theorem 2. Consider the stochastic contextual linear bandit model with long-horizon rewards described in (1). In Algorithm 1, choose $L = csd\log^2(sd)\log^2(hd)$, where $c > 0$ is a constant. When $T < h$, the regret satisfies

$$R_T = O\Big(d\sqrt{sT\log(dT)} + \min\{q(\boldsymbol{w}), T\}\Big), \tag{13}$$

where $q(\boldsymbol{w})$ is a function of the weight vector $\boldsymbol{w}$ that describes how the weights in $\boldsymbol{w}$ are distributed. Specifically, $q(\boldsymbol{w}) := h^{\alpha(\mu)}$, where $\mu \in (0,1)$ and $\alpha(\mu) = \inf_{\alpha\in[0,1]}\|\boldsymbol{w}_{q(\boldsymbol{w})}\|_2 \geq \mu$ with $\boldsymbol{w}_{q(\boldsymbol{w})} := \{w_1, w_2, \ldots, w_{\lceil q(\boldsymbol{w})\rceil}\}$ and $1/\mu = \Theta(1)$.

Remark 5. Notice that the following two facts in our algorithm are crucial for our analysis: 1) we use the difference between $\boldsymbol{P}_i''$ and $\boldsymbol{P}_i'$ (i.e., $\bar{\boldsymbol{P}}_i$ in (10)) as the measurement matrix to learn $\boldsymbol{\phi}_{\mathcal{S}_i}$, ensuring that $\bar{\boldsymbol{P}}_i$ has zero mean, and 2) the doubling trick and the choice of data to use ensure that $\boldsymbol{P}_i''$ and $\boldsymbol{P}_i'$ are *non-overlapping* and *independent*, and $\bar{P}_i$'s in different epochs are also *non-overlapping* and *independent* (see Fig. 3 (b)). Our analysis uses Theorem 1 to show each $\bar{\boldsymbol{P}}_i$ in (11) satisfies the restrictive eigenvalue condition for block-sparse vectors (see Theorem E.1 in the Appendix). Then, we derive Theorem F.1 that generalizes Theorem 7.13 in Wainwright (2019) to complete the proof. △

Remark 6. The value of α describes a "mass-like" distribution of the weights in s-sparse vector $\boldsymbol{w}$. A small α means that non-zero entries appear at early positions of $\boldsymbol{w}$, making it easier to learn useful information of $\boldsymbol{\theta}$ at an early stage than the case of a large α. For instance, if $\alpha \leq \frac{1}{2}\log_h T$ (i.e.,

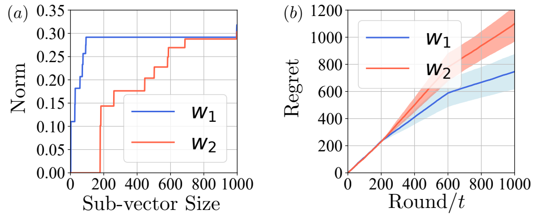

Figure 4: Regret comparison of Algorithm 1 for different instances of $\boldsymbol{w}$. (a) The ℓ_2-norms of the sub-vectors formed by the first k entries of $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$, respectively (k is the x-axis). Notice that the "mass" of $\boldsymbol{w}_1$ is distributed at earlier positions than $\boldsymbol{w}_2$. (b) Experiments show that Algorithm 1 results in a smaller regret for $\boldsymbol{w}_1$ than for $\boldsymbol{w}_2$, as predicted by Theorem 2. (Shaded regions show standard error in 10 trials. Parameters: $h = 1000, T = 999$, and $s = 10$.)

half of the "mass" is located at the first $\sqrt{T}$ positions of $\boldsymbol{w}$), $R_h = \tilde{O}(d\sqrt{sT})$. If $\alpha = 1$, i.e., $\boldsymbol{w} = [0, 0, \ldots, 1]^\top$, then $R_h = O(T)$, which is intuitive since no information can be gathered to help decision making throughout the entire horizon. The upper bound (13) indicates that our algorithm is adaptive to different instances. We conjecture that the dependence on q is optimal; for instance, for delayed bandits, q becomes the delay, which is unavoidable.

Experiments. In Fig. 4, we perform some experiments by considering two different $\boldsymbol{w}$'s, i.e., one with the "mass" distributed at earlier positions and the other at later positions. As predicted by our theory, our algorithm indeed achieves a lower regret in the former case (see Fig. 4 (b)).

Remark 7. Apart from the term $q(\boldsymbol{w}) = h^\alpha$, which is presumably *unavoidable* since it measures the hardness of a problem instance, the upper bound in (13) has no polynomial dependence on h. This means that exploring the sparsity in the reward dependence pattern is indeed beneficial especially when $sd \ll h$. Hao et al. (2020) studied a sparse linear bandit problem in the data-poor regime and obtained an optimal bound, instantiated in our setting, $\tilde{\Theta}((sd)^{\frac{2}{3}} T^{\frac{2}{3}})$. We obtained a distinct bound since we consider a different setting rather than a sparse arm parameter. △

Data-Rich Regime

Now, we consider the situation where there are more rounds than the dimension of the weight vector $\boldsymbol{w}$, i.e., $T \geq h$. In this data-rich regime, we introduce an algorithm outlined in Algorithm 2 (see also Fig. 5 for an illustration).

There are two phases in this algorithm, making it adaptive: 1) in the initial h rounds, we employ the Doubling Lasso (see Algorithm 1); 2) from the $h + 1$ round on, we propose another algorithm. In the second phase, we also use a doubling trick similar to Algorithm 1. The only differences are: 1) the length of epoch i is $2^i h$ instead of $2^{i+1}L$, 2) in each epoch, we estimate the *entire* $\boldsymbol{\phi}$ instead of a portion of it, and 3) the later half of collected data is used.

Same as in Algorithm 1, we collect $2^j h$ data points in each epoch. From (5), the $2^{j-1}h$ rewards in the later half (See Fig. 5) are generated by

$$\tilde{\boldsymbol{r}}[j] = \tilde{\boldsymbol{\Xi}}[j]\boldsymbol{\phi} + \tilde{\boldsymbol{\varepsilon}}[j],$$

where $\tilde{\boldsymbol{r}}[j] \in \mathbb{R}^{2^j h}$, $\tilde{\boldsymbol{\Xi}}[j] \in \mathbb{R}^{2^j h \times dh}$, and $\tilde{\boldsymbol{\varepsilon}}[j] \in \mathbb{R}^{2^j h}$ are the corresponding reward vector, context matrix, and noise vector in the later half of the epoch j, respectively.

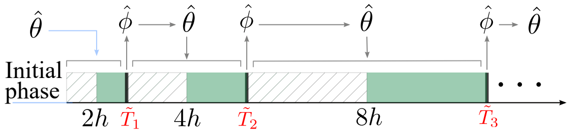

Figure 5: Illustration of AD-Lasso. For the initial phase of h rounds, we use the Doubling Lasso in Algorithm 1. For $t > h$, we also use the doubling trick, but slightly different from Algorithm 1: 1) here $L = h$, and 2) we estimate the entire $\boldsymbol{\phi}$ in each epoch.

Algorithm 2: Adaptive Doubling Lasso (AD-Lasso)

```
Input: $L$ for the initial phase, the doubling sequence $\{\tilde{T}_j\}$ with $\tilde{T}_j = (2^{j+1} - 1)h$ [see Fig. 5]
for $t = 1 : h$ do
    Implement Algorithm 1 with parameter $L$.
end for
Reset $\hat{\boldsymbol{\theta}}_0$ to the latest $\hat{\boldsymbol{\theta}}$.
for $t = h + 1 : T$ do
    Observe contexts vectors $\{\boldsymbol{x}_{t,a} : a \in [K]\}$.
    Take the greedy action $a_t \in \sup_{a \in [K]} \langle \boldsymbol{x}_{t,a}, \hat{\boldsymbol{\theta}}_{j-1} \rangle$, and receive a reward $r_t$.
    if $t = \tilde{T}_j$ then
                                  # end of the jth epoch, estimate a new $\hat{\boldsymbol{\theta}}$
        Calculate $\hat{\boldsymbol{\phi}}[j]$ according to the Lasso (14).
        Let $\hat{\boldsymbol{\theta}}_j$ be the singular vector of $\hat{\Phi}$ associated with the largest singular value.
    end if
end for
```

To learn $\boldsymbol{\phi}$, we calculate the following Lasso program:

$$\hat{\boldsymbol{\phi}}[j] = \mathop{\arg\min}_{\boldsymbol{\phi} \in \mathbb{R}^{hd}} \left(\frac{1}{2^j h} \left\| \tilde{\boldsymbol{\Xi}}[j]\boldsymbol{\phi} - \boldsymbol{r}[j] \right\|_2^2 + \lambda_j \|\boldsymbol{\phi}\|_{2,1}^{(d)} \right), \quad (14)$$

where the regularization parameter is

$$\lambda_j = 2\sqrt{\frac{2d\log(2^j h/\gamma)}{2^{j-1}h}}. \quad (15)$$

Theorem 3. *Consider the stochastic contextual linear bandit model with long-horizon rewards described in (1). In Algoritm 2, let L be the same as in Thoerem 2. When $T \geq h$, the regret has the following upper bound:*

$$R_T = O\Big(d\sqrt{sh\log(dh)} + \min\{q(\boldsymbol{w}), h\} + \sqrt{sdT\log(dT)}\Big), \quad (16)$$

where $q(\boldsymbol{w})$ is defined in Theorem 2.

Remark 8. The first two terms in (16) result from the initial phase ($t \leq h$) when data is poor. Note that they are T-independent even if they are h-dependent; they play a role in the upper bound only when T has the same order of h, i.e., $T = \Theta(h)$. In this case, the upper bound becomes

$$R_T = O\big(d\sqrt{sT\log(dT)} + \min\{q(\boldsymbol{w}), T\}\big).$$

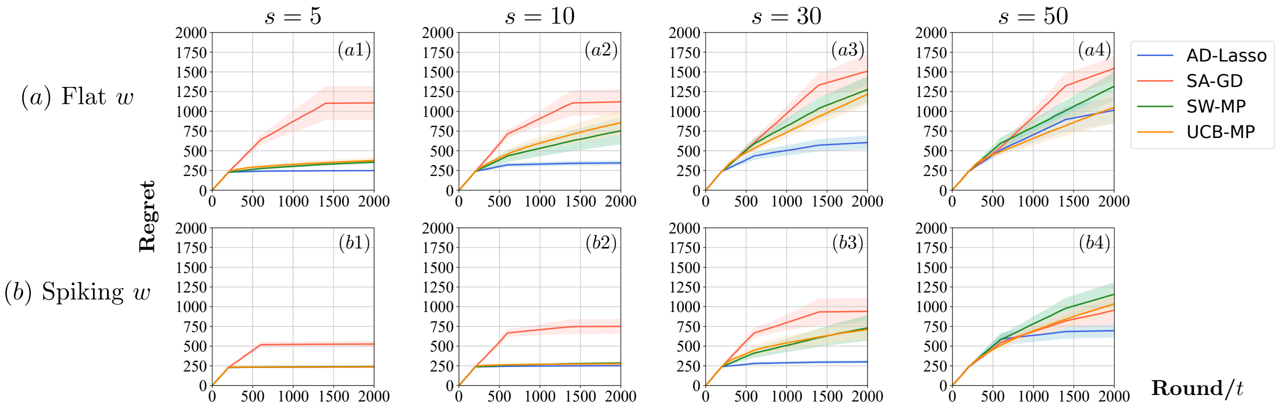

Figure 6: Performance comparison of different algorithms. (a) Flat $\boldsymbol{w}$: the non-zero entries in $\boldsymbol{w}$ are equally spread. (b) Spiking $\boldsymbol{w}$: the majority of the weights concentrates at only 20% of the non-zeros positions in $\boldsymbol{w}$. Different sparsity of $\boldsymbol{w}$ is also considered for both cases. (Universal parameters: $T = 2000, h = 100, d = 5$, and $\|\boldsymbol{w}\|_1 = 1$.)

By contrast, if T is large, specifically, $T \geq \max\{dh, h^{2\alpha(\mu)}/(sd\log(T))\}$, the first two terms are dominated by the last one in (16), and the upper bound reduces to

$$R_T = O\Big(\sqrt{sdT\log(dT)}\Big).$$

Then, our upper bound is *optimal* in d and T (up to logarithmic factors), which follows from the lower bound $\sqrt{dT}$ shown in Chu et al. (2011) for linear contextual bandits.

Discussion on lower bound: Ren and Zhou (2020) obtained a lower bound $\Omega(\sqrt{sT})$ for s-sparse contextual linear bandits. Taking into account the low-rank and sparse nature of our problem, one can show a lower bound of $\Omega(\sqrt{(s+d)T})$ in our case by adapting their proof. Thus, the gap between the our bound in Theorem 3 and this lower bound is at most a factor of $\log(dT)\min\{\sqrt{s}, \sqrt{d}\}$. However, we believe that the actual gap is much smaller. We presume that a tighter lower bound can be constructed since we find that the sampling complexity of low-rank estimation using circulant measurements does not simply depends on the rank (see Sec. for the discussion). △

Experiments. We perform some experiments to compare our algorithm AD-Lasso with the following three:

1. *Sparse-Alternating Gradient Descent* (SA-GD). The core of SA-GD is rank-1 and then sparse matrix estimation. Based on (3), SA-GD alternatively reconstructs $\boldsymbol{\theta}$ and $\boldsymbol{w}$ by gradient descent and projects $\boldsymbol{w}$ to the s-sparse space.
2. *Single-Weight Matching Pursuit* (SW-MP). The core of SW-MP is to locate the *largest* weight in $\boldsymbol{w}$ by testing the correlation between the reward vector and the columns of the context matrix. Then, with this location information, $\boldsymbol{\theta}$ is estimated simply by the least-squares regression, ignoring other weights in $\boldsymbol{w}$.
3. *UCB with Matching Pursuit* (UCB-MP). This algorithm is similar to SW-MP; the difference is that in each epoch we use UCB to update $\hat{\boldsymbol{\theta}}$ and make decisions.

To facilitate fair comparison, we use the same doubling scheme with identical epoch lengths for all the algorithms. The only difference is the method we use to estimate $\boldsymbol{\theta}$ (see Appendix G for more details of these algorithms). Different sparsity and reward dependence structure are considered in the experiments (see the caption in Fig. 6).

Our algorithm outperforms SA-GD significantly when $\boldsymbol{w}$ is highly sparse (see (a1), (a2), (b1), and (b2)). Since SA-GD is primarily reliant on rank-1 factorization, this indicates that, relative to low-rankness, sparsity plays a more dominant role in the estimation quality in line with our theory. Surprisingly, as $\boldsymbol{w}$ becomes less sparse, our algorithm can still outperform SA-GD, even in the regime $sd > d + h$. This supports the difficulty of low-rank matrix estimation with circulant measurements, which is consistent with our discussion in Sec. . Yet, stronger theoretical analysis is desirable to formalize these findings beyond our Lemma 1.

AD-Lasso performs as well as SW-MP and UCB-MP, even when the weights of $\boldsymbol{w}$ are highly concentrated over few entries. When the weights are more spread out, AD-Lasso works much better, indicating that simply exploring and exploiting the largest weight becomes suboptimal.

Concluding Remarks

In this paper, we introduce a novel variation of the stochastic contextual bandits problem, where the reward depends on s prior contexts, up to a time horizon of h. Leveraging the sparsity in the reward dependence pattern, we propose two algorithms that account for both the data-poor and data-rich regimes. We also derive horizon-independent (up to $\log(h)$ terms) regret upper bounds for both algorithms, establishing that their sample efficiency is theoretically guaranteed.

Our work opens up many future potential directions. For instance, the reward can depend on the prior contexts in a nonlinear fashion or sparsity pattern can vary in a data-dependent fashion. In either scenarios learning the reward dependence pattern will be more challenging. Also, beyond bandit problems, it is of interest to explore RL and control scenarios with long-term non-Markovian structures where new strategies will be required.

Acknowledgments

This work was supported in part by the Air Force grant AFOSR-FA9550-20-1-0140, NSF CCF-2046816, NSF CNS-1932254, NSF TRIPODS II-DMS 2023166, NSF CCF-2007036, NSF CCF-2212261, NSF DMS-1839371, and Army Research Office grants ARO-78259-NS-MUR and W911NF2110312.

References

Abbasi-Yadkori, Y.; et al. 2012. Online-to-confidence-set conversions and application to sparse stochastic bandits. In *Artificial Intelligence and Statistics*, 1–9.

Adamczak, R. 2015. A note on the Hanson-Wright inequality for random vectors with dependencies. *Electronic Communications in Probability*, 20: 1–13.

Anava, O.; et al. 2015. Online learning for adversaries with memory: Price of past mistakes. In *Advances in Neural Information Processing Systems*, volume 28.

Ariu, K.; et al. 2022. Thresholded lasso bandit. In *International Conference on Machine Learning*, 878–928.

Bastani, H.; and Bayati, M. 2020. Online decision making with high-dimensional covariates. *Operations Research*, 68(1): 276–294.

Bickel, P. J.; et al. 2009. Simultaneous analysis of Lasso and Dantzig selector. *The Annals of Statistics*, 37(4): 1705–1732.

Bistritz, I.; et al. 2019. Online exp3 learning in adversarial bandits with delayed feedback. In *Advances in Neural Information Processing Systems*, volume 32.

Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, 1877–1901.

Bubeck, S.; et al. 2012. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends® in Machine Learning*, 5(1): 1–122.

Candes, E.; and Tao, T. 2007. The Dantzig selector: Statistical estimation when p is much larger than n. *The Annals of Statistics*, 35(6): 2313–2351.

Carpentier, A.; and Munos, R. 2012. Bandit theory meets compressed sensing for high dimensional stochastic linear bandit. In *Artificial Intelligence and Statistics*, 190–198.

Cella, L.; and Cesa-Bianchi, N. 2020. Stochastic bandits with delay-dependent payoffs. In *International Conference on Artificial Intelligence and Statistics*, 1168–1177.

Cesa-Bianchi, N.; et al. 2018. Nonstochastic bandits with composite anonymous feedback. In *Conference On Learning Theory*, 750–773.

Cesa-Bianchi, N.; et al. 2019. Delay and Cooperation in Nonstochastic Bandits. *Journal of Machine Learning Research*, 20: 1–38.

Chen, L.; Lu, K.; Rajeswaran, A.; Lee, K.; Grover, A.; Laskin, M.; Abbeel, P.; Srinivas, A.; and Mordatch, I. 2021. Decision transformer: Reinforcement learning via sequence modeling. In *Advances in Neural Information Processing Systems*, volume 34, 15084–15097.

Chu, W.; et al. 2011. Contextual bandits with linear payoff functions. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, 208–214.

Davenport, M. A.; and Romberg, J. 2016. An overview of low-rank matrix recovery from incomplete observations. *IEEE Journal of Selected Topics in Signal Processing*, 10(4): 608–622.

Gael, M. A.; et al. 2020. Stochastic bandits with arm-dependent delays. In *International Conference on Machine Learning*, 3348–3356.

Garg, S.; and Akash, A. K. 2019. Stochastic bandits with delayed composite anonymous feedback. *arXiv preprint arXiv:1910.01161*.

Gyorgy, A.; and Joulani, P. 2021. Adapting to delays and data in adversarial multi-armed bandits. In *International Conference on Machine Learning*, 3988–3997.

Hao, B.; et al. 2020. High-dimensional sparse linear bandits. In *Advances in Neural Information Processing Systems*, volume 33, 10753–10763.

Hao, B.; et al. 2021. Information Directed Sampling for Sparse Linear Bandits. In *Advances in Neural Information Processing Systems*, volume 34.

Hessel, M.; Modayil, J.; Van Hasselt, H.; Schaul, T.; Ostrovski, G.; Dabney, W.; Horgan, D.; Piot, B.; Azar, M.; and Silver, D. 2018. Rainbow: Combining improvements in deep reinforcement learning. In *32nd AAAI Conference on Artificial Intelligence*.

Howson, B.; et al. 2022. Delayed Feedback in Generalised Linear Bandits Revisited. *arXiv preprint arXiv:2207.10786*.

Ito, S.; et al. 2020. Delay and cooperation in nonstochastic linear bandits. In *Advances in Neural Information Processing Systems*, volume 33, 4872–4883.

Kim, G.-S.; and Paik, M. C. 2019. Doubly-robust lasso bandit. In *Advances in Neural Information Processing Systems*, volume 32.

Krahmer, F.; et al. 2014. Suprema of chaos processes and the restricted isometry property. *Communications on Pure and Applied Mathematics*, 67(11): 1877–1904.

Kumar, R.; et al. 2022. Online Convex Optimization with Unbounded Memory. *arXiv preprint arXiv:2210.09903*.

Lancewicki, T.; et al. 2021. Stochastic multi-armed bandits with unrestricted delay distributions. In *International Conference on Machine Learning*, 5969–5978.

Langford, J.; and Zhang, T. 2007. The epoch-greedy algorithm for contextual multi-armed bandits. In *Advances in Neural Information Processing Systems*, volume 20, 96–1.

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

Li, B.; et al. 2019. Bandit online learning with unknown delays. In *The 22nd International Conference on Artificial Intelligence and Statistics*, 993–1002.

Li, L.; et al. 2010. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th International Conference on World Wide Web*, 661–670.

Li, L.; et al. 2017. Provably optimal algorithms for generalized linear contextual bandits. In *International Conference on Machine Learning*, 2071–2080.

Oh, M.-h.; et al. 2021. Sparsity-agnostic lasso bandit. In *International Conference on Machine Learning*, 8271–8280.

Oymak, S.; et al. 2015. Simultaneously structured models with application to sparse and low-rank matrices. *IEEE Transactions on Information Theory*, 61(5): 2886–2908.

Pike-Burke, C.; et al. 2018. Bandits with delayed, aggregated anonymous feedback. In *International Conference on Machine Learning*, 4105–4113.

Qin, Y.; Li, Y.; Pasqualetti, F.; Fazel, M.; and Oymak, S. 2023. Stochastic Contextual Bandits with Long Horizon Rewards. *arXiv preprint arXiv:2302.00814*.

Qin, Y.; Menara, T.; Oymak, S.; Ching, S.; and Pasqualetti, F. 2022a. Representation Learning for Context-Dependent Decision-Making. In *2022 American Control Conference (ACC)*, 2130–2135.

Qin, Y.; et al. 2022b. Non-Stationary Representation Learning in Sequential Linear Bandits. *IEEE Open Journal of Control Systems*, 1: 41–56.

Ren, Z.; and Zhou, Z. 2020. Dynamic batch learning in high-dimensional sparse linear contextual bandits. *arXiv preprint arXiv:2008.11918*.

Richard, E.; et al. 2012. Estimation of simultaneously sparse and low rank matrices. *arXiv preprint arXiv:1206.6474*.

Shi, G.; et al. 2020. Online optimization with memory and competitive control. In *Advances in Neural Information Processing Systems*, volume 33, 20636–20647.

Thune, T. S.; et al. 2019. Nonstochastic multiarmed bandits with unrestricted delays. In *Advances in Neural Information Processing Systems*, volume 32.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30.

Vernade, C.; et al. 2020. Linear bandits with stochastic delayed feedback. In *International Conference on Machine Learning*, 9712–9721.

Wainwright, M. J. 2019. *High-Dimensional Statistics: A Non-Asymptotic Viewpoint*. Cambridge University Press.

Wang, S.; et al. 2021. Adaptive Algorithms for Multi-armed Bandit with Composite and Anonymous Feedback. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 10210–10217.

Wang, X.; et al. 2018. Minimax concave penalized multi-armed bandit model with high-dimensional covariates. In *International Conference on Machine Learning*, 5200–5208.

Woodroofe, M. 1979. A one-armed bandit problem with a concomitant variable. *Journal of the American Statistical Association*, 74(368): 799–806.

Zhang, M.; et al. 2022. Gaussian Process Bandits with Aggregated Feedback. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 9074–9081.

Zhou, Z.; et al. 2019. Learning in generalized linear contextual bandits with stochastic delays. In *Advances in Neural Information Processing Systems*, volume 32.

Zimmert, J.; and Seldin, Y. 2020. An optimal algorithm for adversarial bandits with arbitrary delays. In *International Conference on Artificial Intelligence and Statistics*, 3285–3294.