# Nearly Optimal Regret for Stochastic Linear Bandits with Heavy-Tailed Payoffs

**Bo Xue**, **Guanghui Wang**, **Yimu Wang** and **Lijun Zhang** *
National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023, China
{xueb, wanggh, wangym, zhanglj}@lamda.nju.edu.cn

## Abstract

In this paper, we study the problem of stochastic linear bandits with finite action sets. Most of existing work assume the payoffs are bounded or sub-Gaussian, which may be violated in some scenarios such as financial markets. To settle this issue, we analyze the linear bandits with heavy-tailed payoffs, where the payoffs admit finite $1+\epsilon$ moments for some $\epsilon \in (0, 1]$. Through median of means and dynamic truncation, we propose two novel algorithms which enjoy a sublinear regret bound of $\widetilde{O}(d^{\frac{1}{2}}T^{\frac{1}{1+\epsilon}})$, where $d$ is the dimension of contextual information and $T$ is the time horizon. Meanwhile, we provide an $\Omega(d^{\frac{\epsilon}{1+\epsilon}}T^{\frac{1}{1+\epsilon}})$ lower bound, which implies our upper bound matches the lower bound up to polylogarithmic factors in the order of $d$ and $T$ when $\epsilon = 1$. Finally, we conduct numerical experiments to demonstrate the effectiveness of our algorithms and the empirical results strongly support our theoretical guarantees.

## 1 Introduction

Bandit online learning is a powerful framework for modeling various important decision-making scenarios with applications ranging from medical trials to advertisement placement to network routing [Bubeck and Cesa-Bianchi, 2012]. In the basic stochastic multi-arm bandits (MAB) [Robbins, 1952], a learner repeatedly selects one from $K$ arms to play, and then observes a payoff drawn from a fixed but unknown distribution associated with the chosen arm. The learner's goal is to maxmize the cumulative payoffs through the trade-off between exploration and exploitation, i.e., pulling the arms that may potentially give better outcomes and playing the optimal arm in the past [Auer, 2002]. The classic upper confidence bound (UCB) algorithm achieves a regret bound of $O(K \log T)$ over $T$ iterations and $K$ arms, which matches the minimax regret up to a logarithmic factor [Lai and Robbins, 1985].

One fundamental limitation of stochastic MAB is that it ignores the side information (contexts) inherent in the aforementioned real-world applications, such as the user and webpage features in advertisement placement [Abe *et al.*, 2003], which could guide the decision-making process. To address this issue, various algorithms have been developed to exploit the contexts, based on different structures of the payoff functions such as Lipschitz [Kleinberg *et al.*, 2008; Bubeck *et al.*, 2011] or convex [Agarwal *et al.*, 2013; Bubeck *et al.*, 2015]. Among them, the stochastic linear bandits (SLB) has received significant research interests [Auer, 2002; Chu *et al.*, 2011], in which the expected payoff at each round is assumed to be a linear combination of features in the context vector. More precisely, in each round of SLB, the learner first observes feature vector $x_{t,a} \in \mathbb{R}^d$ for each arm $a$. After that, he/she selects an arm $a_t$ and receives payoff $r_{t,a_t}$, such that

$$\mathrm{E}[r_{t,a_t}|x_{t,a_t}] = x_{t,a_t}^\top \theta_* \tag{1}$$

where $\theta_* \in \mathbb{R}^d$ is a vector of unknown parameters. The metric to measure the learner's performance is expected regret, defined as

$$R(T) = \sum_{t=1}^{T} x_{t,a_t^*}^\top \theta_* - \sum_{t=1}^{T} x_{t,a_t}^\top \theta_*$$

where $a_t^* = \mathrm{argmax}_{a\in\{1,2,\ldots,K\}} x_{t,a}^\top \theta_*$ and $a_t$ is the action chosen by the learner at round $t$.

While SLB has been explored extensively [Auer, 2002; Chu *et al.*, 2011; Abbasi-yadkori *et al.*, 2011; Zhang *et al.*, 2016], most of the previous work assume the payoffs are bounded or satisfy the sub-Gaussian property. However, in many real-world scenarios such as financial markets [Cont and Bouchaud, 2000] and neural oscillations [Roberts *et al.*, 2015], the payoffs $r_{t,a}$ fluctuate rapidly and do not exhibit bounded or sub-Gaussian property but satisfy heavy-tailed distributions [Foss *et al.*, 2013], i. e.,

$$\lim_{c\to\infty} \mathbb{P}\left\{r_{t,a} - \mathrm{E}[r_{t,a}] > c\right\} \cdot e^{\lambda c} = \infty, \ \ \forall \lambda > 0.$$

There exists a rich body of work on learning with heavy-tailed distribution [Audibert and Catoni, 2011; Catoni, 2012; Brownlees *et al.*, 2015; Hsu and Sabato, 2016; Zhang and Zhou, 2018; Lu *et al.*, 2019], but limited work contributed to the setting of stochastic linear bandits. Medina and Yang [2016] is the first to investigate this problem, and develop two algorithms achieving $\widetilde{O}(dT^{\frac{2+\epsilon}{2(1+\epsilon)}})$ and $\widetilde{O}(\sqrt{d}T^{\frac{1+2\epsilon}{1+3\epsilon}} + dT^{\frac{1+\epsilon}{1+3\epsilon}})$ regret bounds respectively under the assumption that

*Lijun Zhang is the corresponding author.

the distributions have finite moments of order $1+\epsilon$ for some $\epsilon \in (0, 1]$. Later, Shao *et al.* [2018] improve these bounds to $\widetilde{O}(dT^{\frac{1}{1+\epsilon}})$ by developing two more delicate algorithms. When the variance of payoff is finite (i. e., $\epsilon = 1$), this bound becomes $\widetilde{O}(d\sqrt{T})$, which is nearly optimal in terms of $T$. However, when the number of arms is finite, this upper bound is sub-optimal as there exists an $O(\sqrt{d})$ gap from the lower bound $\Omega(\sqrt{dT})$ [Chu *et al.*, 2011]. Thus, an interesting challenge is to recover the regret of $O(\sqrt{dT})$ under the heavy-tailed setting for linear bandits with finite arms.

To the best of our knowledge, this is the first work which investigates heavy-tailed SLB with finite arms and our contributions are highlighted as follows:

- We propose two novel algorithms to address the heavy-tailed issue in stochastic linear bandits with finite arms. One is developed based on median of means, and the other adopts the truncation technique. Furthermore, we establish an $\widetilde{O}(d^{\frac{1}{2}}T^{\frac{1}{1+\epsilon}})$ regret bound for both algorithms.
- We provide an $\Omega(d^{\frac{\epsilon}{1+\epsilon}}T^{\frac{1}{1+\epsilon}})$ lower bound for heavy-tailed SLB problem, which matches our upper bound in terms of the dependence on $T$. It also implies the dependence on $d$ in our upper bound is optimal up to a logarithmic term when $\epsilon = 1$.
- We conduct numerical experiments to demonstrate the performance of our algorithms. Through comparisons with existing work, our proposed algorithms exhibit improvements on heavy-tailed bandit problem.

## 2 Related Work

In this section, we briefly review the related work on bandit learning. The $p$-norm of vector $x \in \mathbb{R}^d$ is $\|x\|_p = (|x_1|^p + \ldots + |x_d|^p)^{1/p}$ and the $\ell_2$-norm is denoted as $\|\cdot\|$.

### 2.1 Bandit Learning with Bounded/Sub-Gaussian Payoffs

The celebrated work of Lai and Robbins [1985] derived a lower bound of $\Omega(K \log T)$ for stochastic MAB, and proposed an algorithm which achieves the lower bound asymptotically by making use of the upper confidence bound (UCB) policies. Auer [2002] studied the problem of stochastic linear bandits, and developed a basic algorithm named LinRel to solve this problem. However, he failed to provide a sublinear regret for LinRel since the analysis of the algorithm requires all observed payoffs so far to be independent random variables, which may be violated. To resolve this problem, he turned LinRel to be a subroutine which assumes independence among the payoffs, and then constructed a master algorithm named SupLinRel to ensure the independence. Theoretical analysis demonstrates that SupLinRel enjoys an $\widetilde{O}(\sqrt{dT})$ regret bound, assuming the number of arms is finite. Chu *et al.* [2011] modified LinRel and SupLinRel slightly to BaseLinUCB and SupLinUCB, which enjoy similar regret bound but less computational cost and easier theoretical analysis. They also provided an $\Omega(\sqrt{dT})$ lower bound for SLB. Dani *et al.* [2008] considered the setting where the arm set is infinite, and proposed an algorithm named ConfidenceBall$_2$ which enjoys a regret bound of $\widetilde{O}(d\sqrt{T})$. Later, Abbasi-yadkori *et al.* [2011] provided a new analysis of ConfidenceBall$_2$, and improved the worst case bound by a logarithmic factor.

The main diffculty in bandit problem is the trade-off between exploitation and exploration. Most of the existing work take advantage of UCB to settle this issue and adopt the tool of ridge regression to estimate $\theta_*$ [Auer, 2002; Chu *et al.*, 2011]. The least square estimator of Chu *et al.* [2011] is

$$\hat{\theta}_t = \operatorname*{argmin}_{\theta \in \mathbb{R}^d} \|V_t\theta - Y_t\|^2 + \|\theta\|^2 \tag{2}$$

where $V_t = [x_{\tau,a_\tau}]_{\tau \in \Psi_t} \in \mathbb{R}^{|\Psi_t| \times d}$ is a matrix of the historical contexts, $Y_t = [r_{\tau,a_\tau}]_{\tau \in \Psi_t} \in \mathbb{R}^{|\Psi_t| \times 1}$ is the historical payoff vector and $\Psi_t \subseteq \{1, 2, \ldots, t-1\}$ is a filtered index set. The confidence interval for arm $a$ at round $t$ is

$$\left[x_{t,a}^\top \hat{\theta}_t - w_{t,a},\ x_{t,a}^\top \hat{\theta}_t + w_{t,a}\right] \tag{3}$$

where $w_{t,a} = (\alpha_t + 1)\sqrt{x_{t,a}^\top A_t x_{t,a}}, A_t = I_d + V_t^\top V_t$ and $\alpha_t = O(\sqrt{\ln(TK)})$. If $w_{t,a}$ is small for all $a \in \{1, 2, \ldots, K\}$, which means the estimations for coming payoffs are accurate enough, the arm with highest upper confidence bound is played to execute exploitation. Otherwise, if there exists an arm $a$ with $w_{t,a}$ large enough, arm $a$ is played to explore more information.

### 2.2 Bandit Learning with Heavy-tailed Payoffs

The classic paper of Bubeck *et al.* [2013] studied stochastic MAB with heavy-tailed payoffs, and proposed a UCB-type algorithm which enjoys a logarithmic regret bound, under the assumption that the $1+\epsilon$ moment of the payoffs is bounded for some $\epsilon \in (0, 1]$. They also constructed a matching lower bound. Medina and Yang [2016] extended the analysis to SLB, and developed two algorithms enjoying $\widetilde{O}(dT^{\frac{2+\epsilon}{2(1+\epsilon)}})$ and $\widetilde{O}(\sqrt{d}T^{\frac{1+2\epsilon}{1+3\epsilon}} + dT^{\frac{1+\epsilon}{1+3\epsilon}})$ regret bounds respectively. In a subsequent work, Shao *et al.* [2018] constructed an $\Omega(dT^{\frac{1}{1+\epsilon}})$ lower bound for SLB with heavy-tailed payoffs, assuming the arm set is infinite, and developed algorithms with matching upper bounds in terms of $T$.

An intuitive explanation for heavy-tailed distribution is that extreme values are presented with high probability. One strategy tackling the heavy-tailed problem is median of means [Hsu and Sabato, 2016], whose basic idea is to divide all samples drawn from the distribution into several groups, calculate the mean of each group and take the median of these means. Another strategy is truncation following the line of research stemmed from Audibert and Catoni [2011], whose basic idea is to truncate the extreme values. Most of the existing work for heavy-tailed bandits develop algorithms based on median of means and truncation [Bubeck *et al.*, 2013; Medina and Yang, 2016; Shao *et al.*, 2018].

For heavy-tailed SLB algorithms adopting median of means, it is common to play the chosen arm multiple times and get $r$ payoffs $\{r_{t,a_t}^j\}_{j=1}^r$ at each round. Different "means"

is considered in existing work [Medina and Yang, 2016; Shao *et al.*, 2018]. The algorithm MoM [Medina and Yang, 2016] takes the median of $\{r^j_{t,a_t}\}_{j=1}^r$ to conduct least square estimation by one time and the subsequent algorithm MENU [Shao *et al.*, 2018] adopts the median of means of least square estimations. More precisely, for $j = 1, 2, \ldots, r$, the $j$-th estimator in MENU is

$$\tilde{\theta}^j_t = \underset{\theta\in\mathbb{R}^d}{\operatorname{argmin}} \|\widetilde{V}_t\theta - \widetilde{Y}^j_t\|^2 + \|\theta\|^2$$

where $\widetilde{V}_t = [x_{\tau,a_\tau}]_{\tau=1}^{t-1} \in \mathbb{R}^{(t-1)\times d}$ and $\widetilde{Y}^j_t = [r^j_\tau]_{\tau=1}^{t-1} \in \mathbb{R}^{(t-1)\times 1}$. After that, the median of means of least square estimations is

$$m_j = \text{median of } \left\{\|\tilde{\theta}^j_t - \tilde{\theta}^s_t\|_{\widetilde{A}_t} : s = 1, \ldots, r\right\}$$

where $\|x\|_{\widetilde{A}_t} = \sqrt{x^\top \widetilde{A}_t x}$ for $x \in \mathbb{R}^d$ and $\widetilde{A}_t = I_d + \widetilde{V}_t^\top \widetilde{V}_t$. Then MENU selects the estimator

$$\tilde{\theta}^{k_*}_t \text{ where } k_* = \underset{j\in\{1,2,\ldots,r\}}{\operatorname{argmin}} \{m_j\} \tag{4}$$

to predict the payoffs for all arms.

For heavy-tailed SLB algorithms adopting truncation, the essential difference between existing work is the term chosen to be truncated. The algorithm based on Confidence Region with Truncation (CRT) [Medina and Yang, 2016] conducts truncation on payoffs $|r_{t,a_t}|$ such that $\tilde{r}_{t,a_t} = r_{t,a_t}\mathbb{I}_{|r_{t,a_t}|\le\eta_t}$ for $\eta_t = t^{\frac{1}{2(1+\epsilon)}}$ and obtains the least square estimator through truncated payoffs $\tilde{r}_{t,a_t}$. An improved algorithm TOFU [Shao *et al.*, 2018] truncates the term $|u^i_\tau r_{\tau,a_\tau}|$. More precisely, let $[u^1, \ldots, u^d] = \widetilde{A}_t^{-1/2}\widetilde{V}_t^\top$ and $u^i = [u^i_1, u^i_2, \ldots, u^i_{t-1}]$ for $i = 1, 2, \ldots, d$. The truncation is operated as

$$\bar{Y}^i_t = [r_{1,a_1}\mathbb{I}_{|u^i_1 r_{1,a_1}|\le b_t}, \ldots, r_{t-1,a_{t-1}}\mathbb{I}_{|u^i_{t-1} r_{t-1,a_{t-1}}|\le b_t}]$$

where $b_t = O(t^{\frac{1-\epsilon}{2(1+\epsilon)}})$ and $\mathbb{I}_{\{\cdot\}}$ is the indicator function. Then the estimator of TOFU is

$$\tilde{\theta}'_t = \widetilde{A}_t^{-1/2}[u^1 \cdot \bar{Y}^1_t, \ldots, u^d \cdot \bar{Y}^d_t] \tag{5}$$

such that $u^i \cdot \bar{Y}^i_t = \sum_{\tau=1}^{t-1} u^i_\tau r_{\tau,a_\tau}\mathbb{I}_{|u^i_\tau r_{\tau,a_\tau}|\le b_t}$ for $i = 1, 2, \ldots, d$.

# 3 Algorithms

In this section, we demonstrate two novel bandit algorithms based on median of means and truncation respectively and illustrate their theoretical guarantees. Without loss of generality, we assume feature vectors and target coefficients are contained in the unit ball, that is

$$\|x_{t,a}\| \le 1, \quad \|\theta_*\| \le 1.$$

Following the work of Chu *et al.* [2011], each of our two original algorithms is divided into basic and master algorithms. The main role of basic algorithms is providing confidence intervals via filtered historical informations, and master algorithm is responsible for ensuring the payoffs' independence.

## 3.1 Basic Algorithms

In the conventional setting where the stochastic payoffs are distributed in $[0, 1]$, Chu *et al.* [2011] ultilized the Azuma-Hoeffing's inequality to get the narrow confidence interval (3). Here, we consider the heavy-tailed setting, i. e., for some $\epsilon \in (0, 1]$, there exists a constant $v > 0$, such that

$$\mathrm{E}\left[|r_{t,a_t} - \mathrm{E}[r_{t,a_t}]|^{1+\epsilon}\right] \le v. \tag{6}$$

Note that in this case, Azuma-Hoeffing's inequality is unapplicable as the bounded assumption is violated. The estimator (2) and confidence interval (3) are not suitable for heavy-tailed setting. Therefore, the challenge is how to establish a robust estimator associated with proper confidence intervals.

The existing work estimate the payoffs for all arms with a single estimator at each round [Auer, 2002; Chu *et al.*, 2011; Medina and Yang, 2016; Shao *et al.*, 2018], while the expeted payoff $\mathrm{E}[r_{t,a}]$ depends not only on $\theta_*$ but also on the contexts $x_{t,a}$. Thus an intuitive conjecture is that it's better to take estimators adaptive to arms' contexts, and the following example confirms such conjecture.

**Example 1.** *We assume $\theta_* = [0.5, 0.5]$, the contextual information is $x_{t,1} = [1, 0]$ for arm 1 and $x_{t,2} = [0, 1]$ for arm 2. If we have two estimator $\hat{\theta}^1_t = [0.5, 0]$ and $\hat{\theta}^2_t = [0, 0.5]$, it's obvious that $\hat{\theta}^1_t$ is a better estimator for $x_{t,1}$ as $x^\top_{t,1}\hat{\theta}^1_t = x^\top_{t,1}\theta_*$ and $\hat{\theta}^2_t$ is better for $x_{t,2}$.*

The above example encourages us to design estimators adaptive to contexts.

**Median of Means**

We first present the basic algorithm through median of means (BMM) to get confidence intervals for coming payoffs. The complete procedure is provided in Algorithm 1.

To adopt median of means in bandit learning, we play the chosen arm $r$ times and obtain $r$ sequences of payoffs. After that, BMM executes least square estimation for each sequence of payoffs and gets $r$ estimators (Step 1-3). For $j = 1, 2, \ldots, r$,

$$\hat{\theta}^j_t = \underset{\theta\in\mathbb{R}^d}{\operatorname{argmin}} \|V_t\theta - Y^j_t\|^2 + \|\theta\|^2 \tag{7}$$

where $V_t = [x_{\tau,a_\tau}]_{\tau\in\Psi_t} \in \mathbb{R}^{|\Psi_t|\times d}$ is a matrix of the historical contexts, $Y^j_t = [r^j_{\tau,a_\tau}]_{\tau\in\Psi_t}$ is the historical payoff vector and $\Psi_t \subseteq \{1, 2, \ldots, t-1\}$ is an index set filtered by the master algorithm. Then, BMM selects an adaptive estimator $\hat{\theta}_{t,a}$ for each arm $a$ by taking the estimated payoffs as "means" (Step 6). More specifically, the estimator for arm $a$ at current round is $\hat{\theta}_{t,a} \in \{\hat{\theta}^j_t\}_{j=1}^r$ such that

$$x^\top_{t,a}\hat{\theta}_{t,a} = \text{ median of } \{x^\top_{t,a}\hat{\theta}^j_t\}_{j=1}^r. \tag{8}$$

By ulilizing median of means, BMM constructs a reliable confidence interval for the expected payoff (Step 6-7), which is

$$\left[x^\top_{t,a}\hat{\theta}_{t,a} - w_{t,a},\ x^\top_{t,a}\hat{\theta}_{t,a} + w_{t,a}\right] \tag{9}$$

where $w_{t,a} = (\alpha_t + 1)\sqrt{x^\top_{t,a}A_t x_{t,a}}$, $A_t = I_d + V_t^\top V_t$ and $\alpha_t = O(t^{\frac{1-\epsilon}{2(1+\epsilon)}})$.

**Algorithm 1** Basic algorithm through Median of Means (BMM)

**Input:** $\alpha_t \in \mathbb{R}_+, r \in \mathrm{N}, \Psi_t \subseteq \{1, 2, \ldots, t-1\}$
**Output:** $\hat{r}_{t,a},\ w_{t,a},\ a = 1, 2, \ldots, K$

$A_t \leftarrow I_d + \sum_{\tau \in \Psi_t} x_{\tau,a_\tau} x_{\tau,a_\tau}^\top$
$b_t^j \leftarrow \sum_{\tau \in \Psi_t} r_{\tau,a_\tau}^j x_{\tau,a_\tau},\ r_{\tau,a_\tau}^j$ is the $j$-th payoff of playing the arm $a_\tau$ in round $\tau$, $j = 1, 2, \ldots, r$
$\hat{\theta}_t^j \leftarrow A_t^{-1} b_t^j,\ j = 1, 2, \ldots, r$
Observe $K$ arm features, $x_{t,1}, x_{t,2}, \ldots, x_{t,K} \in \mathbb{R}^d$
**for** $a = 1, 2, \ldots, K$ **do**
    $\hat{r}_{t,a} \leftarrow x_{t,a}^\top \hat{\theta}_{t,a}$, where $x_{t,a}^\top \hat{\theta}_{t,a}$ is the median of $\{x_{t,a}^\top \hat{\theta}_t^j\}_{j=1}^r$
    $w_{t,a} \leftarrow (\alpha_t + 1)\sqrt{x_{t,a}^\top A_t^{-1} x_{t,a}}$
**end for**

When compared with existing algorithms, the main difference lies in how to combine median of means with least square estimation. As we introduced in related work, MoM of Medina and Yang [2016] and MENU of Shao *et al.* [2018] take payoffs and the distance between different estimators as "means" respectively, while BMM takes estimated payoffs (8) as "means" and predicts coming payoffs with estimators adaptive to contexts. The theoretical guarantee for our estimators is displayed as follows. The payoffs' independence for filtered set $\Psi_t$ is ensured by the master algorithm SupBMM and we will present it later.

**Proposition 1.** *For fixed feature vectors $x_{\tau,a_\tau}$ with $\tau \in \Psi_t$ in BMM, the payoffs $\{r_{\tau,a_\tau}^j\}_{\tau \in \Psi_t}, j = 1, 2, \ldots, r$ are independent random variables which satisfy* (1) *and* (6)*. Then, if $\alpha_t = (12v)^{\frac{1}{1+\epsilon}} t^{\frac{1-\epsilon}{2(1+\epsilon)}}$ and $r = \lceil 8 \ln \frac{2KT \ln T}{\delta} \rceil$, with probability at least $1 - \delta/T$, for any $a \in \{1, 2, \ldots, K\}$, we have*

$$|\hat{r}_{t,a} - x_{t,a}^\top \theta_*| \le (\alpha_t + 1)\sqrt{x_{t,a}^\top A_t^{-1} x_{t,a}}.$$

**Remark.** The confidence interval of BMM depends on the $1+\epsilon$ central moment of the payoff distribution, which is constructed at the cost of $r$ times to play the chosen arm. When the payoffs admit a finite variance, i.e., $\epsilon = 1$, our algorithm utilizes tighter confidence intervals with $\alpha_t = \sqrt{12v}$, in contrast, Chu *et al.* [2011] constructed confidence intervals with $\alpha_t = O(\sqrt{\ln(TK)})$. The detailed proof can be found in the full paper [Xue *et al.*, 2020].

**Truncation**

In this section, we develop the basic algorithm through truncation (BTC) to get confidence intervals for coming payoffs. The complete procedure is provided in Algorithm 2.

For heavy-tailed SLB algorithms adopting truncation, the key point is how to combine the least square estimation with truncation. The existing least square estimator (2) without truncation does not take use of current epoch's contexts $x_{t,a}$, while Example 1 encourages us to consider adaptive estimator. The estimated payoff of Chu *et al.* [2011] is a linear combination of historical payoffs, i. e.,

$$x_{t,a}^\top A_t^{-1} V_t^\top Y_t = \sum_{\tau \in \Psi_t} \beta_\tau r_{\tau,a_\tau}$$

**Algorithm 2** Basic algorithm through Truncation (BTC)

**Input:** $\alpha_t \in \mathbb{R}_+,\ \Psi_t \subseteq \{1, 2, \ldots, t-1\}$
**Output:** $\hat{r}_{t,a},\ w_{t,a},\ a = 1, 2, \ldots, K$

1: $A_t \leftarrow I_d + \sum_{\tau \in \Psi_t} x_{\tau,a_\tau} x_{\tau,a_\tau}^\top$
2: $V_t \leftarrow [x_{\tau,a_\tau}]_{\tau \in \Psi_t}$
3: Observe $K$ arm features, $x_{t,1}, x_{t,2}, \ldots, x_{t,K} \in \mathbb{R}^d$
4: **for** $a = 1, 2, \ldots, K$ **do**
5: $\quad [\beta_{\tau_1}, \beta_{\tau_2}, \ldots, \beta_{\tau_{|\Psi_t|}}] \leftarrow x_{t,a}^\top A_t^{-1} V_t^\top$
6: $\quad h_{t,a} \leftarrow \|x_{t,a}^\top A_t^{-1} V_t^\top\|_{1+\epsilon}$
7: $\quad \widehat{Y}_{t,a} \leftarrow [\hat{r}_{\tau,a_\tau}]_{\tau \in \Psi_t}$ where $\hat{r}_{\tau,a_\tau} = r_{\tau,a_\tau} \mathbb{I}_{|\beta_\tau r_{\tau,a_\tau}| \le h_{t,a}}$
8: $\quad \hat{\theta}_{t,a} \leftarrow A_t^{-1} V_t^\top \widehat{Y}_{t,a}$
9: $\quad \hat{r}_{t,a} \leftarrow x_{t,a}^\top \hat{\theta}_{t,a}$
10: $\quad w_{t,a} \leftarrow (\alpha_t + 1)\sqrt{x_{t,a}^\top A_t^{-1} x_{t,a}}$
11: **end for**

where $x_{t,a}^\top A_t^{-1} V_t^\top = [\beta_\tau]_{\tau \in \Psi_t} \in \mathbb{R}^{1 \times |\Psi_t|}$ depending on contexts. For the sake of designing an estimator adaptive to contexts, BTC truncates term $\beta_\tau r_{\tau,a_\tau}$ (Step 7) and obtains estimated payoff,

$$x_{t,a}^\top A_t^{-1} V_t^\top \widehat{Y}_t = \sum_{\tau \in \Psi_t} \beta_\tau r_{\tau,a_\tau} \mathbb{I}_{|\beta_\tau r_{\tau,a_\tau}| \le h_{t,a}} \tag{10}$$

where $\widehat{Y}_t = \left[r_{\tau,a_\tau} \mathbb{I}_{|\beta_\tau r_{\tau,a_\tau}| \le h_{t,a}}\right]_{\tau \in \Psi_t}$ and $h_{t,a}$ is the truncation criterion. We set $h_{t,a} = \|x_{t,a}^\top A_t^{-1} V_t^\top\|_{1+\epsilon}$, and the confidence interval for the adaptive estimator $\hat{\theta}_{t,a} = A_t^{-1} V_t^\top \widehat{Y}_t$ is

$$\left[x_{t,a}^\top \hat{\theta}_{t,a} - w_{t,a},\ x_{t,a}^\top \hat{\theta}_{t,a} + w_{t,a}\right] \tag{11}$$

where $w_{t,a} = (\alpha_t + 1)\sqrt{x_{t,a}^\top A_t^{-1} x_{t,a}}$ and $\alpha_t = O(\ln(TK) t^{\frac{1-\epsilon}{2(1+\epsilon)}})$ (Step 9-10).

When compared with existing work, the main difference lies in the term chosen to be truncated. CRT of Medina and Yang [2016] truncates the payoff $r_{t,a_t}$ and TOFU of Shao *et al.* [2018] truncates the term $u_\tau^i r_{\tau,a_\tau}$ as we mentioned in related work. Since $r_{t,a_t}$ and $u_\tau^i r_{\tau,a_\tau}$ do not depend on current epoch's contexts, the estimators of CRT and TOFU are not adaptive. BTC develops an adaptive estimator $\hat{\theta}_{t,a}$ by performing least square estimation with the truncated term $\beta_\tau r_{\tau,a_\tau}$. Whether the confidence interval (11) is true is the main difficulty in the analysis of estimator $\hat{\theta}_{t,a}$ because truncation results in a bias.

BTC requires that for some $\epsilon \in (0, 1]$, the $1+\epsilon$ raw moment of the payoffs is bounded, i. e., there is a constant $v > 0$, the payoffs admit

$$\mathrm{E}\left[|r_{t,a_t}|^{1+\epsilon}\right] \le v. \tag{12}$$

**Proposition 2.** *For fixed feature vectors $x_{\tau,a_\tau}$ with $\tau \in \Psi_t$ in BTC, the payoffs $\{r_{\tau,a_\tau}\}_{\tau \in \Psi_t}$ are independent random variables which satisfy* (1) *and* (12)*. If $\alpha_t = \left(\frac{2}{3} \ln \frac{2TK \ln T}{\delta} + \sqrt{2 \ln \frac{2TK \ln T}{\delta} v} + v\right) t^{\frac{1-\epsilon}{2(1+\epsilon)}}$, then with*

*probability at least* $1-\delta/T,\ \forall a \in \{1,2,\ldots,K\}$*, we have*

$$|\hat{r}_{t,a} - x_{t,a}^\top \theta_*| \leq (\alpha_t + 1)\sqrt{x_{t,a}^\top A_t^{-1} x_{t,a}}.$$

**Remark.** The above proposition indicates that the confidence interval (11) provided by BTC is true wih high probability. BTC is less expensive when compared with TOFU of Shao *et al.* [2018] as $A_t^{-1}V_t^\top$ can be computed online by the Sherman-Morrison formula [Golub and Van Loan, 1996] while $\widetilde{A}_t^{-1/2}$ of TOFU can not. As we mentioned in related work, TOFU has to store both historical contextual matrix $\widetilde{V}_t$ and historical payoffs $\{r_{\tau,a_\tau}\}_{\tau=1}^{t-1}$, while BTC only needs to store historical payoffs.

### 3.2 Master Algorithm

Here we display the master algorithm to settle the independence issue and establish its theoretical gurantees. The master algorithm is adapted from SupLinUCB [Chu *et al.*, 2011] and the complete procedure is summarized in Algorithm 3.

At round $t$, the algorithm screens the candidate arms through $S$ stages until an arm is chosen. The algorithm chooses an arm either when the expected payoff is close to the optimal one or when the confidence interval's width is large. More precisely, we consider three situations at each stage. If the estimation payoffs of all arms are accurate enough, which means the confidence level is up to $1/\sqrt{T}$ (Step 8), we do not need to do exploration and choose the arm maxmizing the upper confidence bound. Otherwise, we notice that the width of confidence interval at stage $s$ is supposed to be $2^{-s}$. If $w_{t,a_t}^s > 2^{-s}$ for some $a_t \in \hat{A}_s$ (Step 11), we play it to take more exploration on this arm. The last situation is that we can not decide which arm to choose at current stage (Step 13), and only those arms which are sufficiently close to the optimal arm are filtered to the next stage (Step 14). The master algorithms taking BMM and BTC as subroutines are called SupBMM and SupBTC, respectively.

We notice that the updation of $\Psi_t^s$ is associated with the historical trails in set $\bigcup_{\sigma<s}\Psi_t^\sigma$ and $w_{t,a}^s$, while $w_{t,a}^s$ only depends on contexts $x_{t,a}$ and $x_{\tau,a_\tau}$ with $\tau \in \Psi_t^s$. Thus the payoffs $r_{\tau,a_\tau}$ are independent random variables for any fixed sequence of $x_{\tau,a_\tau}$ with $\tau \in \Psi_t^s$, which satisfies the assumptions of Propositions 1 and 2. We can deduce the following two regret bounds from Propositions 1 and 2.

**Theorem 1.** *Assume all payoffs admit* (1) *and* (6)*. Let* $r = \left\lceil 8\ln\frac{2KT\ln T}{\delta}\right\rceil$*, for any* $\delta \in (0,1)$*, with probability at least* $1-\delta$*, the regret of SupBMM satisfies*

$$R(T) \leq 120\left(1+(12v)^{\frac{1}{1+\epsilon}}\right)\ln T\sqrt{2d\ln\frac{2KT\ln T}{\delta}}T^{\frac{1}{1+\epsilon}} + 5\sqrt{2T\ln\frac{2KT\ln T}{\delta}}.$$

**Remark.** We achieve a regret bound of order $\widetilde{O}(d^{\frac{1}{2}}T^{\frac{1}{1+\epsilon}})$. For $\epsilon = 1$, it reduces to an $\widetilde{O}(\sqrt{dT})$ bound, which implies that we get the same order as bounded payoffs assumption in terms of both $d$ and $T$ [Chu *et al.*, 2011]. We point out that for any random variable $X$,

$$\mathrm{E}[|X-\mathrm{E}X|^{\epsilon_1}] \leq (\mathrm{E}[|X-\mathrm{E}X|^{\epsilon_2}])^{\frac{\epsilon_1}{\epsilon_2}}$$

**Algorithm 3** Master Algorithm (SupBMM and SupBTC)

**Input:** $T \in \mathbb{N}$

1: $S \leftarrow \lfloor \ln T \rfloor$
2: $\Psi_1^s \leftarrow \varnothing$ for all $s \in \{1,2,\ldots,S\}$
3: **for** $t = 1,2,\ldots,T$ **do**
4: $\quad s \leftarrow 1, \hat{A}_1 \leftarrow \{1,2,\ldots,K\}$
5: $\quad$ **repeat**
6: $\quad\quad$ **SupBMM:** Use BMM with $\Psi_t^s$ to calculate the width $w_{t,a}^s$ and upper confidence bound $\hat{r}_{t,a}^s + w_{t,a}^s$ for every $a \in \hat{A}_s$
7: $\quad\quad$ **SupBTC:** Use BTC with $\Psi_t^s$ to calculate the width $w_{t,a}^s$ and upper confidence bound $\hat{r}_{t,a}^s + w_{t,a}^s$ for every $a \in \hat{A}_s$
8: $\quad\quad$ **if** $w_{t,a}^s \leq 1/\sqrt{T}\ \ \forall a \in \hat{A}_s$ **then**
9: $\quad\quad\quad$ Choose $a_t = \mathrm{argmax}_{a\in\hat{A}_s}(\hat{r}_{t,a}^s + w_{t,a}^s)$
10: $\quad\quad\quad$ Keep the same index sets at all levels: $\Psi_{t+1}^{s'} \leftarrow \Psi_t^{s'}\ \forall s' \in \{1,2,\ldots,S\}$
11: $\quad\quad$ **else if** $w_{t,a_t}^s > 2^{-s}$ for some $a_t \in \hat{A}_s$ **then**
12: $\quad\quad\quad$ Choose this arm $a_t$ and update the index sets at all levels:

$$\Psi_{t+1}^{s'} \leftarrow \begin{cases} \Psi_t^{s'} \cup \{t\} & \text{if } s' = s \\ \Psi_t^{s'} & \text{otherwise} \end{cases}$$

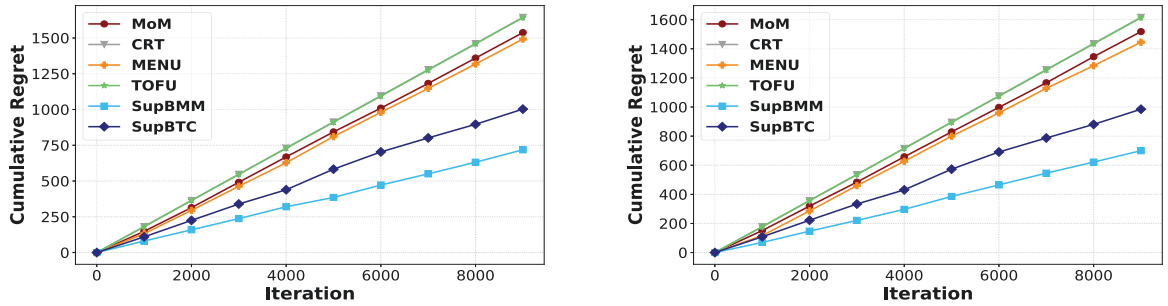
13: $\quad\quad$ **else** $w_{t,a}^s \leq 2^{-s}\ \forall a \in \hat{A}_s$
14: $\quad\quad\quad \hat{A}_{s+1} \leftarrow \{a \in \hat{A}_s | \hat{r}_{t,a}^s + w_{t,a}^s \geq \max_{a'\in\hat{A}_s}(\hat{r}_{t,a'}^s + w_{t,a'}^s) - 2^{1-s}\}$
15: $\quad\quad\quad s \leftarrow s+1$
16: $\quad\quad$ **end if**
17: $\quad$ **until** an arm $a_t$ is found.
18: $\quad$ **SupBMM:** Play $a_t$ $r$ times and observe payoffs $r_{t,a_t}^1, r_{t,a_t}^2, \ldots, r_{t,a_t}^r$
19: $\quad$ **SupBTC:** Play $a_t$ and observe payoff $r_{t,a_t}$
20: **end for**

where $\epsilon_1, \epsilon_2 \in \mathbb{R}_+$ and $\epsilon_1 \leq \epsilon_2$. Therefore, our upper bound $\widetilde{O}(\sqrt{dT})$ also holds for the payoffs with finite higher order ($\epsilon > 1$) central moments, and matches the lower bound of bounded payoffs up to some polylogarithmic factors.

**Theorem 2.** *Assume all payoffs admit* (1) *and* (12)*. For any* $\delta \in (0,1)$*, with probability at least* $1-\delta$*, the regret of SupBTC satisfies*

$$R(T) \leq 2\sqrt{T} + 40\ln T\sqrt{dT} + 40\left(\frac{2}{3}\ln\frac{2TK\ln T}{\delta} + \sqrt{2\ln\frac{2TK\ln T}{\delta}}v + v\right)\ln T\sqrt{d}T^{\frac{1}{1+\epsilon}}.$$

**Remark.** The above theorem assumes a finite $1+\epsilon$ raw moment of payoffs and achieves a regret bound of the same order $\widetilde{O}(d^{\frac{1}{2}}T^{\frac{1}{1+\epsilon}})$ as Theorem 1, while Theorem 1 depends on the $1+\epsilon$ central moments of payoffs. Shao *et al.* [2018] proposed the algorithms achieving the regret bound $\widetilde{O}(dT^{\frac{1}{1+\epsilon}})$, so our algorithms have a better dependence on $d$ for finite-armed SLB with heavy-tailed payoffs.




(a) Student's $t$-Noise (b) Pareto Noise

Figure 1: Comparison of our algorithms versus the MoM, CRT, MENU and TOFU.

## 4 Lower Bound

In this section, we give the lower bound for finite-armed SLB with heavy-tailed payoffs.

**Theorem 3.** *For any algorithm $\mathcal{A}$ with $T \geq K \geq 4$ and $T \geq (2d)^{\frac{1+\epsilon}{2\epsilon}}$, let $\gamma = (K/(T+2K))^{\frac{1}{1+\epsilon}}$, there exists a sequence of feature vectors $\{x_{t,a}\}_{t=1}^{T}$ for $a = 1, 2, \ldots, T$ and a coefficient vector $\theta_*$ such that the payoff for each arm is in $\{0, 1/\gamma\}$ with mean $x_{t,a}^\top \theta_*$. If $d \geq K$, we have*

$$\mathrm{E}\left[R(T)\right] \geq \frac{1}{32}(d-1)^{\frac{\epsilon}{1+\epsilon}} T^{\frac{1}{1+\epsilon}} = O\left(d^{\frac{\epsilon}{1+\epsilon}} T^{\frac{1}{1+\epsilon}}\right).$$

**Remark.** The above theorem essentially establishes an $\Omega(d^{\frac{\epsilon}{1+\epsilon}} T^{\frac{1}{1+\epsilon}})$ lower bound associated with $d$ and $T$ for SLB under the heavy-tailed setting, which matches the upper bounds of Theorems 1 and 2 in the sense of the polynomial order on $T$. To the best of our knowledge, this is the first lower bound for finite-armed SLB with heavy-tailed payoffs. The detailed proof can be found in the full paper [Xue *et al.*, 2020].

## 5 Experiments

In this section, we conduct experiments to evaluate the proposed algorithms. All algorithms' parameters are set to $\epsilon = 1$ and $\delta = 0.01$. We adopt MoM and CRT of Medina and Yang [2016], MENU and TOFU of Shao *et al.* [2018] as baselines for comparison.

Let the feature dimension $d = 10$, the number of arms $K = 20$ and $\theta_* = \mathbf{1}/\sqrt{d} \in \mathbb{R}^d$, where $\mathbf{1}$ is an all-1 vector so that $\|\theta_*\| = 1$. Each element of the vector $x_{t,a}$ is sampled from the uniform distribution of $[0, 1]$, and then the vector is normalized to a unit vector ($\|x_{t,a}\| = 1$). According to the linear bandit model, the observed payoff is

$$r_{t,a} = x_{t,a}^\top \theta_* + \eta_t$$

where $\eta_t$ is generated from the following two noises.

(i) Student's $t$-Noise: The probability density function of this noise is $\eta_t \sim \frac{\Gamma(2)}{\sqrt{3\pi}\Gamma(1.5)}\left(1 + \frac{x^2}{3}\right)^{-2}$ for $x \in \mathbb{R}$ and $\Gamma(\cdot)$ is the Gamma function. Thus, the bounds of the second central moment and second raw moment of payoff are 3 and 4, respectively.

(ii) Pareto Noise: The probability density function of this noise is $\eta_t \sim \frac{s x_m^s}{x^{s+1}} \mathbb{I}_{x \geq x_m}$ for $x \in \mathbb{R}$ and we set the shape $s = 3$ and the scale $x_m = 0.01$. The bounds of the second central moment and second raw moment of payoff are 1 and 2.

The main difference between the above two heavy-tailed noises is that Student's $t$-distribution is symmetric while Pareto distribution is not.

We run 10 independent repetitions for each algorithm and display the average cumulative regret with time evolution. Fig. 1(a) compares our algorithms against algorithms of Medina and Yang [2016] and Shao *et al.* [2018] under Student's $t$-noises. Fig. 1(b) presents the cumulative regrets under Pareto noises. Our algorithms outperform MoM, CRT, MENU and TOFU with the interference of symmetric or asymmetric noises, which verifies the effectiveness of our algorithms on the heavy-tailed bandit problem. SupBMM achieves the smallest regret which is expected,since compared to SupBTC it has a more favorable logarithmic factor in regret bound.

## 6 Conclusion and Future Work

In this paper, we develop two novel algorithms to settle the heavy-tailed issue in linear contextual bandit with finite arms. Our algorithms only require the existence of bounded $1+\epsilon$ moment of payoffs, and achieve $\widetilde{O}(d^{\frac{1}{2}} T^{\frac{1}{1+\epsilon}})$ regret bound which is tighter than that of Shao *et al.* [2018] by an $O(\sqrt{d})$ factor for finite action sets. Furthermore, we provide a lower bound on the order of $\Omega(d^{\frac{\epsilon}{1+\epsilon}} T^{\frac{1}{1+\epsilon}})$. Finally, our proposed algorithms have been evaluated based on numerical experiments and the empirical results demonstrate the effectiveness in addressing heavy-tailed problem.

In the future, we will investigate more on closing the gap between upper bound and lower bound with respect to the dimension $d$.

## Acknowledgments

This work was partially supported by NSFC (61976112), NSFC-NRF Joint Research Project (61861146001), and the Collaborative Innovation Center of Novel Software Technology and Industrialization.

## References


[Abbasi-yadkori *et al.*, 2011] Yasin Abbasi-yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems 24*, pages 2312–2320. 2011.

[Abe *et al.*, 2003] Naoki Abe, Alan W Biermann, and Philip M Long. Reinforcement learning with immediate rewards and linear hypotheses. *Algorithmica*, 37(4):263–293, 2003.

[Agarwal *et al.*, 2013] A. Agarwal, D. Foster, D. Hsu, S. Kakade, and A. Rakhlin. Stochastic convex optimization with bandit feedback. *SIAM Journal on Optimization*, 23(1):213–240, 2013.

[Audibert and Catoni, 2011] Jean-Yves Audibert and Olivier Catoni. Robust linear least squares regression. *The Annals of Statistics*, 39(5):2766–2794, 2011.

[Auer, 2002] Peter Auer. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, 3:397–422, 2002.

[Brownlees *et al.*, 2015] Christian Brownlees, Emilien Joly, and Gábor Lugosi. Empirical risk minimization for heavy-tailed losses. *The Annals of Statistics*, 43(6):2507–2536, 2015.

[Bubeck and Cesa-Bianchi, 2012] Sébastien Bubeck and Nicolò Cesa-Bianchi. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends in Machine Learning*, 5(1):1–122, 2012.

[Bubeck *et al.*, 2011] Sébastien Bubeck, Gilles Stoltz, and Jia Yuan Yu. Lipschitz bandits without the lipschitz constant. In *Proceedings of the 22Nd International Conference on Algorithmic Learning Theory*, pages 144–158, 2011.

[Bubeck *et al.*, 2013] S. Bubeck, N. Cesa-Bianchi, and G. Lugosi. Bandits with heavy tail. *IEEE Transactions on Information Theory*, 59(11):7711–7717, 2013.

[Bubeck *et al.*, 2015] Sébastien Bubeck, Ofer Dekel, Tomer Koren, and Yuval Peres. Bandit convex optimization: $\sqrt{T}$ regret in one dimension. In *Proceedings of The 28th Conference on Learning Theory*, pages 266–278, 2015.

[Catoni, 2012] Olivier Catoni. Challenging the empirical mean and empirical variance: A deviation study. *Annales de l'I.H.P. Probabilités et statistiques*, 48(4):1148–1185, 2012.

[Chu *et al.*, 2011] Wei Chu, Lihong Li, Lev Reyzin, and Robert Schapire. Contextual bandits with linear payoff functions. In *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics*, pages 208–214, 2011.

[Cont and Bouchaud, 2000] Rama Cont and Jean-Philipe Bouchaud. Herd behavior and aggregate fluctuations in financial markets. *Macroeconomic Dynamics*, 4(02):170–196, 2000.

[Dani *et al.*, 2008] Varsha Dani, Thomas P. Hayes, and Sham M. Kakade. Stochastic linear optimization under bandit feedback. In *Proceedings of the 21st Annual Conference on Learning*, pages 355–366, 2008.

[Foss *et al.*, 2013] Sergey Foss, Dmitry Korshunov, and Stan Zachary. *An Introduction to Heavy-Tailed and Subexponential Distributions*. Springer, New York, 2013.

[Golub and Van Loan, 1996] Gene H. Golub and Charles F. Van Loan. *Matrix Computations (3rd Ed.)*. Johns Hopkins University Press, USA, 1996.

[Hsu and Sabato, 2016] Daniel Hsu and Sivan Sabato. Loss minimization and parameter estimation with heavy tails. *Journal of Machine Learning Research*, 17(18):1–40, 2016.

[Kleinberg *et al.*, 2008] Robert Kleinberg, Aleksandrs Slivkins, and Eli Upfal. Multi-armed bandits in metric spaces. In *Proceedings of the 40th Annual ACM Symposium on Theory of Computing*, pages 681–690, 2008.

[Lai and Robbins, 1985] T. L. Lai and Herbert Robbins. Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*, 6(1):4–22, 1985.

[Lu *et al.*, 2019] Shiyin Lu, Guanghui Wang, Yao Hu, and Lijun Zhang. Optimal algorithms for Lipschitz bandits with heavy-tailed rewards. In *Proceedings of the 36th International Conference on Machine Learning*, pages 4154–4163, 2019.

[Medina and Yang, 2016] Andres Munoz Medina and Scott Yang. No-regret algorithms for heavy-tailed linear bandits. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning*, pages 1642–1650, 2016.

[Robbins, 1952] Herbert Robbins. Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society*, 58(5):527–535, 1952.

[Roberts *et al.*, 2015] James A Roberts, Tjeerd W Boonstra, and Michael Breakspear. The heavy tail of the human brain. *Current Opinion in Neurobiology*, 31:164–172, 2015.

[Shao *et al.*, 2018] Han Shao, Xiaotian Yu, Irwin King, and Michael R. Lyu. Almost optimal algorithms for linear stochastic bandits with heavy-tailed payoffs. In *Advances in Neural Information Processing Systems 32*, pages 8430–8439, 2018.

[Xue *et al.*, 2020] Bo Xue, Guanghui Wang, Yimu Wang, and Lijun Zhang. Nearly optimal regret for stochastic linear bandits with heavy-tailed payoffs. *arXiv preprint*, abs/2004.13465, 2020.

[Zhang and Zhou, 2018] Lijun Zhang and Zhi-Hua Zhou. $\ell_1$-regression with heavy-tailed distributions. In *Advances in Neural Information Processing Systems 31*, pages 1084–1094, 2018.

[Zhang *et al.*, 2016] Lijun Zhang, Tianbao Yang, Rong Jin, Yichi Xiao, and Zhi-Hua Zhou. Online stochastic linear optimization under one-bit feedback. In *Proceedings of the 33rd International Conference on Machine Learning*, pages 392–401, 2016.