Fast UCB-type algorithms for stochastic bandits with heavy and super heavy symmetric noise

Yuriy Dorn
MSU Institute for Artificial Intelligence
Moscow Institute of Physics and Technology
Moscow, Russia
dornyv@my.msu.ru

Aleksandr Katrutsa
Skoltech, AIRI
Moscow, Russia
amkatrutsa@gmail.com

Ilgam Latypov
MSU Institute for Artificial Intelligence
Moscow Institute of Physics and Technology
Moscow, Russia
i.latypov@iai.msu.ru

Andrey Pudovikov
MSU Institute for Artificial Intelligence
Moscow, Russia
a.pudovikov@iai.msu.ru

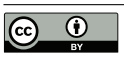
ABSTRACT

This paper considers stochastic multi-armed bandit problems (MAB) and presents a novel framework for constructing UCB-type algorithms. The main ingredient of UCB-type algorithms is the estimate of the confidence bound typically derived from statistical assumptions. On the opposite, our approach derives the confidence bounds from the convergence rate of the base convex optimization method, which helps to solve auxiliary optimization problems in every round. To show the relations between the convergence of the optimization method and the novel UCB-type algorithm, we derive the regret bounds corresponding to the convergence rates of the selected optimization method. To illustrate the proposed framework, we introduce a new algorithm, Clipped-SGD-UCB, for the MAB with heavy-tailed reward distribution, where Clipped-SGD is used as a base convex optimization method since its convergence for the heavy-tail inexact oracle is known. We show theoretically and empirically that in the case of symmetric noise in the reward distribution, one can achieve an $O(\log T\sqrt{KT\log T})$ regret bound instead of $O\left(T^{\frac{1}{1+\alpha}}K^{\frac{\alpha}{1+\alpha}}\right)$. These bounds correspond to the cases where the reward distribution satisfies $\mathbb{E}_{X\in\mathcal{D}}[|X|^{1+\alpha}] \le \sigma^{1+\alpha}$ ($\alpha \in (0,1]$), i.e. perform better than it is assumed by the general lower bound for bandits with heavy-tails.

KEYWORDS

multi-armed bandits; UCB algorithm; heavy tails

ACM Reference Format:
Yuriy Dorn, Aleksandr Katrutsa, Ilgam Latypov, and Andrey Pudovikov. 2025. Fast UCB-type algorithms for stochastic bandits with heavy and super heavy symmetric noise. In *Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Detroit, Michigan, USA, May 19 – 23, 2025*, IFAAMAS, 9 pages.



Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Y. Vorobeychik, S. Das, A. Nowé (eds.), May 19 – 23, 2025, Detroit, Michigan, USA. © 2025 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org).

1 INTRODUCTION

In this work, we consider the stochastic multi-armed bandit problem (MAB) with a heavy-tailed reward distribution introduced by [4]. This problem is a special case of the classical MAB problem introduced by [18]. The problem is formulated as follows: an agent sequentially chooses one of the K actions (arms) in every round with a total number of rounds equal to T. For each arm $i = 1, \dots, K$ there is a corresponding unknown probability distribution of reward $\mathcal{D}_i$ with a finite mean μ_i and finite $(1+\alpha)$-moment, where $\alpha \in (0, 1]$, i.e. more formally there exists fixed $\sigma_i > 0$ such that $\mathbb{E}_{X\in\mathcal{D}_i}[|X|^{1+\alpha}] \le \sigma_i^{1+\alpha}$. In every round t, the agent selects an arm A_t, and then the reward is sampled independently from $\mathcal{D}_{A_t}$. The agent minimizes the regret accumulated throughout T rounds

$$R_T = T\max_{1\le i\le K}\mu_i - \sum_{t=1}^{T}\mathbb{E}[\mu_{A_t}]. \tag{1}$$

It is well-known [4] that a regret lower bound is $\Omega\left(T^{\frac{1}{1+\alpha}}K^{\frac{\alpha}{1+\alpha}}\right)$.

In [4], the general template for constructing UCB-type algorithms named Robust UCB is proposed. It requires a tractable and robust mean estimation procedure and can be described as follows:

- For each arm i find mean estimator $\hat{\mu}_i^{k_i}$ based on k_i samples of reward, such that $|\hat{\mu}_i^{k_i} - \mu_i| \le r(k_i, \delta)$ holds with probability at least $1-\delta$ and the confidence radius $r(k_i, \delta)$ decreases with increasing number of samples k_i.
- For each arm i construct upper confidence bound (UCB) from the mean estimator $\hat{\mu}_i^{k_i}$ and the confidence radius $r(k_i, \delta)$

$$UCB_i(k_i, \delta) = \hat{\mu}_i^{k_i} + r(k_i, \delta), \tag{2}$$

  which is used as a high probability upper bound on mean reward.
- Play arm A_t such that $A_t = \arg\max_{1\le i\le K} UCB_i(k_i, \delta)$, observe reward and update $UCB_{A_t}(k_{A_t}, \delta)$.

In the vanilla UCB introduced by [2], the mean estimator $\hat{\mu}_i^{k_i}$ is computed via empirical mean $\hat{\mu}_i^{k_i} = \frac{1}{k_i}\sum_{t=1}^{T} X_t\mathbb{I}_{\{A_t=i\}}$ and confidence interval $r(k_i, \delta) = \sqrt{\frac{2v\log 1/\delta}{k_i}}$. In Robust UCB, the empirical mean can be replaced by a truncated empirical mean, median of means, or Catoni's M-estimator to construct $\hat{\mu}_i^{k_i}$ and confidence

radius $r(k_i, \delta) = v^{1/\alpha}\left(\frac{c \log 1/\delta}{k_i}\right)^{\frac{\alpha}{1+\alpha}}$ with an appropriate choice of parameters v and c.

However, the standard estimators used in the Robust UCB template, like the truncated mean, require intensive computations and reduce its practical relevance. Recently proposed algorithms in [1, 7] have the same drawback. To address this issue, we introduce a novel approach to constructing UCB-type algorithms for the MAB problem based on stochastic optimization algorithms.

A resulting algorithm has the same per-round complexity as a stochastic algorithm used as a base and a regret rate as a standard UCB-type algorithm. UCB estimation in every round can be done independently for each arm, and one needs to update only a single element of a vector with UCB values. Since the single UCB value changes at each round, sorting these values can be done quickly.

To demonstrate the efficiency of our algorithm, we compare it with the following SOTA algorithms in numerical experiments: Clipped-INF-med-SMD [10], HTINF [9], Adaptive Robust UCB algorithm [7] and classical Robust UCB [4].

The contributions of the presented study are the following.

- We propose a novel framework for constructing UCB-type algorithms from the convergence bound of a base convex optimization method.
- We establish relations between the convergence rate of the base optimization method and the corresponding UCB-type algorithm in terms of regret.
- We illustrate the introduced framework with the heavy-tailed MAB problem, where Clipped-SGD is a base optimizer.
- The extensive numerical experiments confirm the performance of our approach compared to alternatives.

1.1 Related works.

The Robust UCB algorithm from [4] is probably the first relevant UCB-type algorithm for bandits with heavy tails. Its version with a median of means estimator is very close to our approach and demonstrates similar per-iteration convergence. However, the crucial drawback of the Robust UCB algorithm is its high computational costs. Our approach is less costly than Robust UCB.

Study [1] proposed an optimal algorithm that matches the lower bound exactly in the first-order term. The proposed index concentrates faster compared to well-known truncated or trimmed empirical mean estimators for the mean of heavy-tailed distributions. However, this index is computationally demanding. The authors of [12] propose an optimal algorithm if it is known that the reward distribution has the p-th moment, but the bounding constant is unknown. In addition, indices for all arms are recomputed in every round, making these algorithms slow. Adaptive Robust UCB algorithm [7] matches the lower bound and requires no additional knowledge of the reward's distribution parameters. However, the estimate used there is computationally demanding.

In recent years, another idea, usually referred to as "best-of-two-worlds" was proposed. The name implies that the proposed algorithms achieve lower bounds in stochastic and adversarial MAB settings. This idea assumes the application of techniques for adversarial bandit [3] to stochastic MAB problems with a heavy-tailed distribution of rewards, see [6, 9, 13, 20, 22]. Powered by recent advances in online convex optimization (OCO) (see [8, 16]), this approach leads to optimal algorithms in both adversarial and stochastic settings with instance-independent regret bound $O\left(T^{\frac{1}{1+\alpha}} K^{\frac{\alpha}{1+\alpha}}\right)$.

Despite optimality in regret rate, the "best-of-the-two-worlds" framework has high per-iteration complexity. To adjust the target distribution, this framework uses online mirror descent (OMD), which can be computationally intensive. Since OMD performs operations on the entire vector in every round, it slows down in high dimensions. Moreover, for some prox functions, OMD updates require solving an optimization problem in every round (see [21]). Thus, this issue of the "best-of-the-two-worlds" framework restricts its usefulness for real-time risk-sensitive and highly loaded systems.

There are alternative problem statements for the multi-armed bandits with heavy-tailed rewards. Study [14] considers Lipschitz bandits and establishes the corresponding lower bounds. The authors of [15] consider linear bandits, and optimal algorithms for this setting (up to a logarithmic factor) were introduced in [19]. These problem statements are out of the scope of our work.

2 FROM STOCHASTIC OPTIMIZATION METHODS TO UCB-TYPE ALGORITHMS.

This section presents the general framework for converting a stochastic optimization method with the known convergence rate to the UCB-type algorithm. The proposed framework is motivated by the key observation that the typical mean estimators in UCB-type algorithms are solutions to auxiliary optimization problems. For example, the empirical mean estimator maximizes the likelihood function for the standard normal noise model. Therefore, we can equip every arm with the auxiliary stochastic optimization problems that satisfy the following requirements:

1) the solution of the i-th optimization problem at the round t coincides with the estimate of the i-th arm reward at the same round:
$$\hat{\mu}_i = \arg\min_x f_i(x) \tag{3}$$
2) a noise distribution induced by the inexact oracle used in solving stochastic optimization problems coincides with the noise model in MAB.

In this work, we consider a stochastic bandit problem with a reward distribution that satisfies the following assumption.

Assumption 1. *The random reward X_i^t for any arm $i = 1, \ldots, K$ and at any round $t = 1, \ldots, T$ is computed as $X_i^t = \mu_i + \xi_i^t$, where the noise ξ_i^t with the probability density function ρ_i^t satisfies the following conditions:*

- *it is i.i.d. random variable: $\rho_i^t(u) = \rho_i(u)$ for any $u \in \mathbb{R}$,*
- *it is symmetric: $\rho(u) = \rho(-u)$ for any $u \in \mathbb{R}$,*
- *it has heavy tail: there are $\sigma > 0$ and $\alpha > 0$, such that $\mathbb{E}[|\xi_i^t|^\alpha] \le \sigma^\alpha$.*

To complete the presentation of our framework, we have to provide the approach to constructing the confidence radius $r(k, \delta)$ used in (2). This confidence radius is closely related to the convergence of the selected basic stochastic optimization method. Formally, we define the first-order and zero-order $g(k, \delta)$-bounded optimization methods below (see Definitions 2 and 3) and derive the proper form of the confidence radius from their convergence rates.

Definition 2. A first-order stochastic optimization method

$$x_{k+1} = \mathcal{A}\left(x_0, \mathcal{G}_N(x_0), \ldots, x_k, \mathcal{G}_N(x_k)\right),$$

where $\mathcal{G}_N(x_i)$ is the aggregated N samples $f'_{\xi_1}(x_i), \ldots, f'_{\xi_N}(x_i)$ generated by inexact first-order oracle, is referred to as a $g(k,\delta)$-bounded for solving minimization problem $\min_x f(x)$ if for any $k \in \mathbb{N}$ and $\delta > 0$ inequality

$$f(x_k) - f(x^*) \leq g(k,\delta)$$

holds with a probability of at least $1-\delta$.

Definition 3. A zero-order stochastic optimization method

$$x_{k+1} = \mathcal{B}\left(x_0, \mathcal{H}_N(x_0), \ldots, x_k, \mathcal{H}_N(x_k)\right),$$

where $\mathcal{H}_N$ is the aggregated N samples $f(x_i) + \xi_1, \ldots, f(x_i) + \xi_N$ generated by the stochastic zero-order oracle, is referred to as a $g(k,\delta)$-bounded zero-order algorithm for solving minimization problem $\min_x f(x)$ if for any $k \in \mathbb{N}$ and $\delta > 0$ inequality

$$f(x_k) - f(x^*) \leq g(k,\delta)$$

holds with probability at least $1-\delta$.

The value of N used to aggregate the stochastic gradient samples in Definition 2 corresponds to the bias and variance of the used gradient estimator. Further, this parameter plays an important role in the convergence analysis of a particular optimization method. The same role this parameter plays in Definition 3, where the bias and variance of the objective function can be estimated.

The known convergence rates $g(k,\delta)$ can be used to estimate the confidence radius for the mean estimator. Below, we provide the First-order UCB (FO-UCB) and Zero-order UCB (ZO-UCB) algorithms constructed based on the $g(k,\delta)$-bounded optimization methods of the corresponding orders.

2.1 First-order UCB algorithms.

The ingredients for constructing FO-UCB method are the following.

(1) Objective function for the i-th arm is $f_i(x) = \frac{1}{2}(x-\mu_i)^2$. Other functions are feasible if the condition (3) holds.
(2) The corresponding inexact first-order oracle $f'_i(x,\xi) = x - \mu_i - \xi$, where ξ is a random noise with distribution aligned with the noise distribution for rewards, and the result of the aggregation $\mathcal{G}_N(x)$ for the pre-defined parameter N.
(3) A basic first-order optimization method $\mathcal{A}$, which is $g(k,\delta)$-bounded.

In particular, assume that the aforementioned ingredients are prepared, then we can compose the following UCB-type algorithm. The mean estimator in every round is computed through the step of the selected basic optimizer, and the corresponding confidence radius is estimated as $\sqrt{g(k,\delta)}$. Then, the UCB index similar to (2) is computed and is used to select the next arm to play. The summary of this scheme is presented in Algorithm 1.

Since our FO-UCB heavily relies on the selected $g(k,\delta)$-bounded first-order optimization method, it is natural to expect that the convergence of the FO-UCB is closely related to the convergence of the selected optimizer. Indeed, we present the convergence of our FO-UCB with respect to the regret metric (1) in Theorem 4.

Algorithm 1 FO-UCB

Require: Basic first-order $g(k,\delta)$-bounded optimization method $\mathcal{A}$, number of arms K, period T, initial estimates $x_1^0 = \cdots = x_K^0 = x^0$, parameter δ, aggregation rule $\mathcal{R}$, number of samples for aggregation N.

1: Run $\mathcal{A}$ for each arm $i = 1, \ldots, K$ independently to compute $x_i^1 = \mathcal{A}(x_0, f'(x_0, \xi))$.
2: For each arm $i = 1, \ldots, K$ set $k_i = 1$ and initialize $UCB_i(k_i, \delta) = x_i^1 + \sqrt{g(1,\delta)}$.
3: **for** $t = 1, \ldots, T$ **do**
4: Choose arm $i_t = \arg\max_{1 \leq i \leq K} UCB_i(k_i, \delta)$.
5: Play i_t arm N times, observe N rewards $\mu_{i_t} + [\xi_{i_t}^t]_1, \ldots, \mu_{i_t} + [\xi_{i_t}^t]_N$, where $[\xi_{i_t}^t]_j$ is the j-th sample of the noise.
6: Compute $x_{i_t}^{k_{i_t}+1} = \mathcal{A}(x_{i_t}^1, \mathcal{G}_N(x_{i_t}^1), \ldots, x_{i_t}^{k_{i_t}}, \mathcal{G}_N(x_{i_t}^{k_{i_t}}))$, where $\mathcal{G}_N(x_{i_t}^{k_{i_t}}) = \mathcal{R}(f'(x_{i_t}^{k_{i_t}}, [\xi_{i_t}^t]_1), \ldots, f'(x_{i_t}^{k_{i_t}}, [\xi_{i_t}^t]_N))$, where $f'(x_{i_t}^{k_{i_t}}, [\xi_{i_t}^t]_j) = x_{i_t}^{k_{i_t}} - \mu_{i_t} - [\xi_{i_t}^t]_j$.
7: Update UCB index of the played arm and preserve others:

$$UCB_i(k_i+1,\delta) = \begin{cases} UCB_i(k_i,\delta), & i \neq i_t, \\ x_{i_t}^{k_{i_t}} + \sqrt{2g(k_{i_t},\delta)}, & i = i_t. \end{cases}$$

8: Increase iteration counter for the played arm: $k_{i_t} := k_{i_t} + 1$.
9: **end for**

Theorem 4 (Convergence of FO-UCB). *The regret of the* FO-UCB *with $g(k,\delta)$-bounded first-order algorithm for the MAB problem with K arms, functions $f_i(x) = \frac{1}{2}(x-\mu_i)^2$, period T, $\delta = \frac{1}{T^2}$ satisfies*

$$R_T \leq \sum_{i:\Delta_i > 0} \Delta_i \left(g^{-1}\left(\left\lceil \frac{\Delta_i^2}{8} \right\rceil, \frac{1}{T^2} \right) + 2 \right), \tag{4}$$

where $\Delta_i = \mu_{i^*} - \mu_i$, $i^* = \arg\max_{1 \leq i \leq K} \mu_i$, and g^{-1} is such a function that $g(g^{-1}(k,\delta),\delta) = k$.

Proof. For proof, we follow the standard approach (see [11]). Denote by $\Delta_i = \mu_{i^*} - \mu_i$, where $i^* = \arg\max_{1 \leq i \leq K} \mu_i$. Then, regret can be computed as

$$R_T = \sum_{i=1}^{K} \Delta_i \mathbb{E}[n_i(T)],$$

where $n_i(t)$ is the number of rounds before round t when arm i was chosen.

Let G_i be a "good" event defined by

$$G_i = \left\{ \mu_{i^*} < \min_{1 \leq t \leq T} UCB(i^*, n_{i^*}(t), \delta) \right\} \cap \left\{ UCB(i, u_i, \delta) < \mu_{i^*} \right\},$$

where the constant u_i will be chosen later.

We show that if G_i holds, then $n_i(T) \leq u_i$. We assume that this is not true and $n_i(T) > u_i$. Then, there exists a round $t \leq T$ such that $n_i(t-1) = u_i$ and $A_t = i$. Then

$$UCB(i, n_i(t-1), \delta) = x_i^{u_i} + \sqrt{2g(u_i,\delta)} < \mu_{i^*} < UCB(i^*, n_{i^*}(t-1), \delta).$$

Hence, $A_t = \arg\max_{1 \leq j \leq K} UCB(j, n_j(t-1), \delta) \neq i$ and we obtain a contradiction.

Next, we bound the probability of the complementary event:

$$\hat{G}_i = \left\{\mu_{i^*} > \min_{1\le t\le T} UCB(i^*, n_{i^*}(t), \delta)\right\} \cup \left\{x_i^{u_i} + \sqrt{2g(u_i,\delta)} > \mu_{i^*}\right\}.$$

We can then determine the probability of the first term using a union bound:

$$\begin{aligned}
&\mathbb{P}\left[\mu_{i^*} > \min_{1\le t\le T} UCB(i^*, n_{i^*}(t), \delta)\right] \\
&= \mathbb{P}\left[\cup_{s\le T}\left\{\mu_{i^*} > UCB(i^*, n_{i^*}(s), \delta)\right\}\right] \\
&\le \sum_{s\le T} \mathbb{P}\left[\mu_{i^*} > UCB(i^*, n_{i^*}(s), \delta)\right] \le \delta T.
\end{aligned}$$

To bound the probability of the second term, we use the following scheme:

$$\begin{aligned}
&\mathbb{P}\left[x_i^{u_i} + \sqrt{2g(u_i,\delta)} > \mu_{i^*}\right] \\
&= \mathbb{P}\left[x_i^{u_i} - \mu_i + \sqrt{2g(u_i,\delta)} > \mu_{i^*} - \mu_i\right] \\
&= \mathbb{P}\left[x_i^{u_i} - \mu_i > \Delta_i - \sqrt{2g(u_i,\delta)}\right] \\
&= \mathbb{P}\left[x_i^{u_i} - \mu_i > \Delta_i - \sqrt{2g(u_i,\delta)} | \quad |x_i^{u_i} - \mu_i| > \sqrt{2g(u_i,\delta)}\right] \cdot \\
&\mathbb{P}\left[|x_i^{u_i} - \mu_i| > \sqrt{2g(u_i,\delta)}\right] + \\
&\mathbb{P}\left[x_i^{u_i} - \mu_i > \Delta_i - \sqrt{2g(u_i,\delta)} | \quad |x_i^{u_i} - \mu_i| \le \sqrt{2g(u_i,\delta)}\right] \cdot \\
&\mathbb{P}\left[|x_i^{u_i} - \mu_i| \le \sqrt{2g(u_i,\delta)}\right] \\
&(\text{choose } u_i\text{: } \sqrt{2g(u_i,\delta)} = \Delta_i - \sqrt{2g(u_i,\delta)}) \\
&\le \mathbb{I}\cdot\delta + 0\cdot(1+\delta) = \delta.
\end{aligned}$$

Hence

$$\mathbb{E}[n_i(T)] = \mathbb{E}[n_i(T)\mathbb{I}_{G_i}] + \mathbb{E}[n_i(T)\mathbb{I}_{\hat{G}_i}] \le u_i + \delta T(T+1).$$

Assuming $\delta = \frac{1}{T(T+1)}$ and taking $u_i = g^{-1}\left(\frac{\Delta_i^2}{8}, \delta\right)$, where g^{-1} is such that $g\left(g^{-1}(x,\delta),\delta\right) = x$, we get

$$\mathbb{E}[n_i(T)] \le g^{-1}\left(\frac{\Delta_i^2}{8}, \frac{1}{T(T+1)}\right) + 1.$$

Now, we can proceed with estimating regret.

$$R_T = \sum_{i:\Delta_i>0} \Delta_i \mathbb{E}[n_i(T)] \le \sum_{i:\Delta_i>0} \Delta_i \left(g^{-1}\left(\frac{\Delta_i^2}{8}, \frac{1}{T(T+1)}\right) + 1\right).$$

□

2.2 Zero-order UCB algorithms.

Similar to the first-order UCB, the following ingredients are needed to construct a ZO-UCB.

(1) The objective function $f_i(x) = |x - \mu_i|$ is assigned to the i-th arm. Other functions that satisfy the requirement (3) are also feasible.
(2) The corresponding zero-order inexact oracle $|x - \mu_i - \xi|$, where ξ is a random noise whose distribution consistent with the reward noise model and result of aggregation $\mathcal{H}_N(x)$ for a pre-defined value of N.
(3) A basic zero-order optimization method $\mathcal{B}$ which is $g(k,\delta)$-bounded

Then, one can construct the following UCB-type algorithm from these ingredients.

Algorithm 2 ZO-UCB

Require: Basic zero-order $g(k,\delta)$-bounded optimization method $\mathcal{B}$, number of arms K, period T, initial estimates $x_1^0 = \cdots = x_K^0 = x^0$, parameter δ, aggregation rule $\mathcal{R}$ and the number of samples for aggregation N.

1: Run $\mathcal{B}$ for each arm $i = 1,\ldots,K$ independently to compute $x_i^1 = \mathcal{B}(x_0, f_i(x_0,\xi))$.
2: For each arm $i = 1,\ldots,K$ set $k_i = 1$ and initialize $UCB_i(k_i,\delta) = x_i^1 + g(1,\delta)$.
3: **for** $t = 1,\ldots,T$ **do**
4: Choose arm $i_t = \arg\max_{1\le i\le K} UCB_i(k_i,\delta)$.
5: Play i_t arm N times, observe N rewards $\mu_{i_t} + \xi_1, \ldots, \mu_{i_t} + \xi_N$, where ξ_j is the j-th sample of the noise.
6: Compute $x_{i_t}^{k_{i_t}+1} = \mathcal{B}(x_{i_t}^1, \mathcal{H}_N(x_{i_t}^1), \ldots, x_{i_t}^{k_{i_t}}, \mathcal{H}_N(x_{i_t}^{k_{i_t}}))$, where $\mathcal{H}_N(x_{i_t}^{k_{i_t}}) = \mathcal{R}(x_{i_t}^{k_{i_t}} - \mu_{i_t} - \xi_1, \ldots, x_{i_t}^{k_{i_t}} - \mu_{i_t} - \xi_n)$.
7: Update UCB index of the played arm and preserve others:

$$UCB_i(k_i+1,\delta) = \begin{cases} UCB_i(k_i,\delta), & i \ne i_t, \\ x_{i_t}^{k_{i_t}} + g(k_{i_t},\delta), & i = i_t. \end{cases}$$

8: Increase iteration counter for the played arm: $k_{i_t} := k_{i_t} + 1$.
9: **end for**

Simple example. The simplest example of the suggested ZO-UCB is the Robust UCB algorithm from [4]. In particular, one can consider the basic zero-order method $\mathcal{B}$ as an identity operator and use the Median of Means (MoM) aggregation rule $\mathcal{R}$. In this case, ZO-UCB is reduced to the Robust UCB algorithm, which could be further improved with non-trivial zero-order method $\mathcal{B}$.

Now, we discuss the convergence properties of the introduced ZO-UCB algorithm. Similar to Theorem 4, we present the convergence of a Zero-Order UCB algorithm in terms of the convergence of a basic zero-order method $\mathcal{B}$ in Theorem 5.

Theorem 5 (Convergence of ZO-UCB). *The regret of the* ZO-UCB *with $g(k,\delta)$-bounded first-order algorithm for the MAB problem with K arms, functions $f_i(x) = |x - \mu_i|$, period T, $\delta = \frac{1}{T^2}$ satisfies*

$$R_T \le \sum_{i=1}^{K} \Delta_i \left(g^{-1}\left(\left\lceil\frac{\Delta_i}{2}\right\rceil, \frac{1}{T^2}\right) + 2\right),$$

where notations are the same as in Theorem 4.

The proof of this theorem follows a similar scheme as the proof of Theorem 4 from Section 2.1. The complete proof is presented in supplementary materials [5].

Remark 6. *Note that Theorems 4 and 5 do not mean that* ZO-UCB *achieves better regret bound compared to* FO-UCB *since bounding functions $g(k,\delta)$ for first-order and zero-order algorithms are different and therefore they result in different convergence of algorithms.*

Thus, we have obtained the general convergence rates for UCB-type algorithms constructed based on the $g(k,\delta)$-bounded first-order and zero-order basic optimization methods. In the next section, we provide the example of constructing the particular instance of FO-UCB algorithm based on the particular $g(k,\delta)$-bounded first-order optimization method. The resulting FO-UCB algorithm demonstrates faster convergence in both runtime and the number of rounds compared to alternatives.

3 CLIPPED-SGD-UCB: EXAMPLE OF THE FIRST-ORDER UCB ALGORITHM.

As we can see from Theorem 4, to make a good first-order UCB-type algorithm, one needs a $g(k,\delta)$-bounded first-order algorithm with the known convergence rate and feasible stochastic first-order oracle. The feasibility here means that the noise model in the oracle is consistent with the noise model in the MAB problem. Since we focus on the heavy-tail MAB problem, we consider the basic optimization method, which is efficient for the stochastic oracle with a heavy-tailed noise distribution. We find that clipped-SGD method [17] satisfies these requirements and can be used to construct the FO-UCB algorithm for the heavy-tailed MAB problem.

3.1 Clipped-SGD method to minimize convex function with heavy-tailed noise in stochastic gradient.

In this section, we briefly describe the clipped-SGD method [17] with the median of means gradient estimator and highlight that it is appropriate for constructing the first-order UCB-type algorithm.

The clipped-SGD method generates the sequence approaching the optimal point according to the following update rule:

$$x^{k+1} = x^k - \gamma_k \texttt{clip}(f'_{\Xi^k}(x^k), \lambda_k), \tag{5}$$

where $f'_{\Xi^k}(x^k)$ is an estimator satisfying Assumption 7 and sampled independently from previous iterations. Also, the function $\texttt{clip}(\mathbf{g}, \lambda_k) = \min\{1, \lambda_k/\|\mathbf{g}\|_2\}\mathbf{g}$ for the predefined sequence $\{\lambda_k\}_{k=1}^{\infty}$ with $\lambda_k > 0$, and learning rate $\gamma_k > 0$. Particular values of λ_k and γ_k are specified in Section 4.

Assumption 7. *There exists $N \in \mathbb{N}$, aggregation rule $\mathcal{R}$ and (possibly dependent on T) constants $b \geq 0$, $\sigma \geq 0$ such that for an $x \in \mathbb{R}$ i.i.d. samples $f'_{\xi_1}(x), \ldots, f'_{\xi_N}(x)$ from the oracle $\mathcal{G}(x)$ satisfy the following relations:*

$$\left|\mathbb{E}\left[f'_{\Xi}(x)\right] - f'(x)\right| \leq b \quad \mathbb{E}\left[\left|f'_{\Xi}(x) - \mathbb{E}\left[f'_{\Xi}(x)\right]\right|^2\right] \leq \sigma^2,$$

where $f'_{\Xi}(x) = \mathcal{R}(f'_{\xi_1}(x), \ldots, f'_{\xi_N}(x))$ and expectations are taken w.r.t. $f'_{\xi_1}(x), \ldots, f'_{\xi_N}(x)$.

Also, we select the aggregation rule $\mathcal{R}$ equal to the smoothed median of means, see Definition 8.

Definition 8. Let ζ be a random element in $\mathbb{R}$ and let $\theta > 0$ be an arbitrary number. For any positive integers m and n, the smoothed median of means $\texttt{SMoM}_{m,n}(\zeta, \theta)$ is defined as follows:

$$\texttt{SMoM}_{m,n}(\zeta, \theta) = \texttt{Med}\left(v_1, \ldots, v_{2m+1}\right), \tag{6}$$

where, for each $j \in \{0, \ldots, 2m\}$,

$$v_j = \texttt{Mean}(\zeta_{jn+1}, \ldots, \zeta_{(j+1)n}) + \theta\,\eta_{j+1},$$

$\zeta_1, \ldots, \zeta_{(2m+1)n}$ are i.i.d. samples of ζ, and $\eta_1, \ldots, \eta_{2m+1} \sim \mathcal{N}(0,1)$ are independent standard Gaussian random variables.

According to [17], the smoothed median of means aggregation rule satisfies Assumption 7 for samples $f'_{\xi_1}(x), \ldots, f'_{\xi_N}(x)$, where $N = (2m+1)n$ for some pre-defined integers m and n. We also need the following assumption for technical reasons.

Assumption 9. *There exists a set $Q \subseteq \mathbb{R}$ and constant $L > 0$ such that for all $x, y \in Q$*

$$\|f'(x) - f'(y)\| \leq L\|x - y\|, \quad \|f'(x)\|^2 \leq 2L\left(f(x) - f_*\right),$$

where $f_* = \inf_{x \in Q} f(x) > -\infty$.

Now we are ready to present the particular case of a theorem from [17] to show that clipped-SGD can be considered as an example of the first-order $g(k,\delta)$-bounding algorithm to solve auxiliary optimization problems assigned to the arms.

Theorem 10. *Consider the unconstrained minimization problem, where objective function $f(x) = \frac{1}{2}(x-\mu)^2$ is 1-strongly convex and satisfies Assumption 9. The first-order inexact oracle $\mathcal{G}$ gives an unbiased gradient estimate. Also, we assume that the aggregation rule $\mathcal{R}$ satisfies Assumption 7. Then, there exists $C > 0$ such that the* clipped-SGD *method with $\gamma_k \equiv \gamma = \min\left(\frac{1}{400L\ln\frac{4(K+1)}{\delta}}, \frac{\ln((K+1)R^2)}{K+1}\right)$ and clipping hyperparameter $\lambda_k = \frac{\exp(-\gamma(1+k/2))R}{120\gamma\ln\frac{4(K+1)}{\delta}}$ provides the iterates such that after $k = 1, \ldots, K$ iterations the following bound holds with probability at least $1-\delta$*

$$f(x_k) - f^* \leq C\frac{\ln\frac{4(K+1)}{\delta}\ln^2((K+1)R^2)}{k+1}$$

where K is sufficiently large and $R \geq \|x_0 - x^*\|_2$.

The proof of this theorem is presented in supplementary materials [5].

Corollary 11. *Let Assumptions 7 and 9 hold on $Q = B_{2R}(x^*)$, where $R \geq \|x^0 - x^*\|$. Suppose that $f'_{\Xi^k}(x^k)$ satisfies Assumption 7, and $\gamma = \min\left(\frac{1}{400L\ln\frac{4(K+1)}{\delta}}, \frac{\ln((K+1)R^2)}{K+1}\right)$ $\lambda_k \equiv \frac{\exp(-\gamma(1+k/2))R}{120\gamma\ln\frac{4(K+1)}{\delta}}$. Then* clipped-SGD *is $C\frac{\ln\frac{4(K+1)}{\delta}\ln^2((K+1)R^2)}{k+1}$-bounding first-order algorithm.*

Now we are ready to present the resulting first-order UCB-type algorithm based on the briefly introduced clipped-SGD method.

3.2 Clipped-SGD-UCB algorithm and its convergence.

The combination of the general framework presented in Algorithm 1 with a particular basic first-order optimization method (5) leads to the clipped-SGD-UCB algorithm. This algorithm focuses on the rewards that satisfy Assumption 1. Since we use the smoothed median of means (SMoM) estimator of the gradient, the single estimate of the reward is not enough. According to Definition 8, we need $N = (2m+1)n$ reward estimates to construct samples $f'_{\xi_1}, \ldots, f'_{\xi_N}$ and generate the resulting gradient estimate. The resulting clipped-SGD-UCB algorithm is presented in Algorithm 3.

Algorithm 3 Clipped-SGD-UCB

Require: Number of arms K, period T, two positive integers m and n such that the number of rewards samples $N = (2m+1)n$, initial estimates $x_1^0 = \cdots = x_K^0 = x^0$, clipping regime $\{\lambda_t\}_{t=1}^{\infty}$, learning rate schedule $\{\gamma_t\}_{t=1}^{\infty}$, parameter δ.

1: Run Clipped-SGD-UCB for each arm $i = 1, \ldots, K$ independently for N times and compute $f'_{\Xi^1}(x_i^0)$ and $x_i^1 = x_i^0 - \gamma_0 \texttt{clip}(f'_{\Xi^1}(x_i^0), \lambda_0)$.
2: For each arm i ($i = 1, \ldots, K$) set $n_i(K) = 1$ and compute $UCB(i, n_i(K), \delta) = x_i^1 + \sqrt{g(1, \delta)}$.
3: **for** $t = 1, \ldots, T$ **do**
4: Choose arm $i_t = \arg\max_{1 \le i \le K} UCB(i, n_i, \delta)$.
5: Play i_t arm N times, observe rewards and compute $\nabla_{\Xi_{i_t}^{n_{i_t}}} f_{i_t}(x_{i_t}^{n_{i_t}})$ with smooth median of means aggregation rule, see Definition 8.
6: Compute $x_{i_t}^{n_{i_t}+1} = x_{i_t}^{n_{i_t}} - \gamma_{n_{i_t}} \texttt{clip}(\nabla_{\Xi_{i_t}^{n_{i_t}}} f_i(x_{i_t}^{n_{i_t}}), \lambda_{n_{i_t}})$.
7: Set $n_{i_t}(t+1) = n_{i_t}(t) + 1$ (increase counter by one).
8: Set

$$UCB(i, n_i(t+1), \delta) = \begin{cases} UCB(i, n_i(t), \delta), & i \neq i_t, \\ x_{i_t}^{n_{i_t}} + \sqrt{2g(n_{i_t}(t+1), \delta)}, & \text{otherwise.} \end{cases}$$

9: **end for**

Theorem 12 (Convergence of Clipped-SGD-UCB). *The regret of the* Clipped-SGD-UCB *for multi-armed bandit problem with*

- K *arms*
- *period* T
- $\gamma = \min\left(\frac{1}{400L \ln \frac{4(K+1)}{\delta}}, \frac{\ln((K+1)R^2)}{K+1}\right)$
- $\lambda_k = \frac{\exp(-\gamma(1+k/2))R}{120\gamma \ln \frac{4(K+1)}{\delta}}$
- *symmetric distribution of rewards*

satisfies:

$$R_T \le 4\log((T+1)TR^2)\sqrt{C\log(4T(T+1)^2)TK} + \sum_i \Delta_i$$

$$R_T \le \sum_{i:\Delta_i>0} \left[\Delta_i + \frac{8C\log(4T(T+1^2))\log^2((T+1)TR^2)}{\Delta_i}\right]$$

Proof. From Theorem 4, we get

$$R_T \le \sum_{i:\Delta_i>0} \Delta_i \left(g^{-1}\left(\frac{\Delta_i^2}{8}, \frac{1}{T(T+1)}\right) + 1\right).$$

In case $g(k, \delta) = \frac{C\log(4(T+1)/\delta)\log^2((T+1)R^2)}{k}$ we get $g^{-1}(x, \delta) = \frac{C\log(4(T+1)/\delta)\log^2(T+1)R^2}{x}$, and

$$R_T \le \sum_{i:\Delta_i>0} \left[\Delta_i + \frac{8C\log(4T(T+1^2))\log^2((T+1)TR^2)}{\Delta_i}\right].$$

We get instance-dependent bound. Now let $\Delta > 0$ be some fixed value. Then we can bound regret in the following way:

$$R_T = \sum_{i:\Delta_i>0} \Delta_i \mathbb{E}[n_i(T)] = \sum_{i:\Delta_i<\Delta} \Delta_i \mathbb{E}[n_i(T)] + \sum_{i:\Delta_i\ge\Delta} \Delta_i \mathbb{E}[n_i(T)]$$
$$\le T\Delta + \frac{8C\log(4T(T+1)^2)\log^2((T+1)TR^2)K\log T}{\Delta} + \sum_i \Delta_i.$$

If $\Delta = \frac{\log((T+1)TR^2)\sqrt{8C\log(4T(T+1)^2)K}}{\sqrt{T}}$ then the following bound holds

$$R_T \le 4\log((T+1)TR^2)\sqrt{C\log(4T(T+1)^2)TK} + \sum_i \Delta_i.$$

□

Remark 13. *If the algorithm uses a batch of samples to perform a single step with the batch size b, the regret will increase in b times, but the number of iterations will be* $\frac{T}{b}$. *Thus adaptive part of the bound will not change, and only the fixed part will increase, i.e. each arm will require at least* $2b$ *samples instead of* 2.

4 NUMERICAL EXPERIMENTS.

We demonstrate the superior performance of the proposed algorithm in the following environments. The main feature of the test environment is the structure of the noise that models the uncertainty of the observed rewards. In the simulations of the multi-armed bandit, one can obtain a reward estimate $r_i = \mu_i + \xi$ corresponding to the i-th arm, where μ_i is the ground-truth reward and ξ is the noise, whose distribution is the key feature of the testing environments. We focus on super-heavy and heavy tail noise distributions. The additional feature of the environment is the number of arms and the distribution of the corresponding ground-truth rewards. The closer these rewards are, the more challenging the MAB problem is. To better illustrate the algorithms' performance, we adjust the particular instances of such environments for the considered noise structure and provide details in the corresponding sections.

We consider as baselines the Robust UCB [4] that uses the median of means to estimate rewards (`RUCB-Median`), the Adaptively Perturbed Exploration with a p-Robust Estimator (`APE`) [13], HT-INF [9] and Clipped-INF-med-SMD [10]. In our experiments, we found that `APE` works better with overestimated values of the noise moment p. Therefore, we consider $p = 1.25 + \alpha$ and $p = 2$.

We compare these baselines with algorithms derived from Algorithm 3 with different aggregation rules. In particular, SGD-UCB corresponds to the values $m = 0, n = 1$ and uses the single sample to estimate the gradient similar to vanilla SGD. SGD-UCB-Median corresponds to the values $m = 1, n = 1$, takes three samples, and uses their median to estimate the gradient. SGD-UCB-SMoM corresponds to the values $m = 1, n = 2$ and uses SMoM (6) as a gradient estimate. The parameters m and n are used in Algorithm 3 to generate batch size b and construct gradient estimate according to Definition 8. The source code for reproducing our results is in the GitHub repository https://github.com/IAIOnline/fast_SGD_UCB.

Initialization of reward estimates. To initialize the reward estimate for every arm we use the following procedure. Every arm is pulled p times (p is an odd number), and the median of the obtained rewards is used as initialization x_i^1 in Algorithm 3. In this setup, we skip line 1 in pseudocode presented in Algorithm 3. From our

experience, we recommend using $p = 1$ for Gaussian rewards noise and $p = 3$ for heavy and super-heavy tail rewards nose.

4.1 Super-heavy tail MAB.

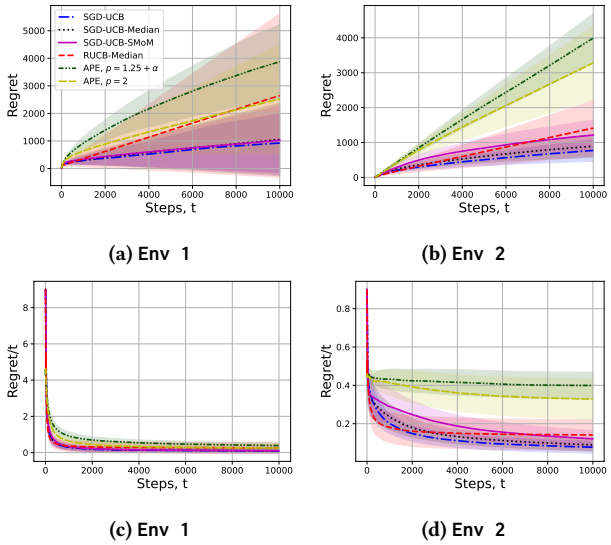
This section considers the super-heavy tail distributions of the noise used in the rewards uncertainty simulation. A distribution has a super-heavy tail if the expectation of the corresponding random variable does not exist. We test Cauchy distributions with the CDF $p_C(x) = \frac{1}{\pi\gamma[1+(\frac{x}{\gamma})^2]}$, where $\gamma = 1$, Fréchet distribution with the CDF $p_F(x) = e^{-x^{-\beta}}$, where $\beta = 1$, the mixture of Cauchy ($\gamma = 1$) and exponential distributions with the CDF $p_{CE}(x) = 0.7 \cdot p_C(x) + 0.3 \cdot e^{-(x+1)}\mathbb{I}\{x \geq -1\}$ and the mixture of Cauchy ($\gamma = 1$) and Pareto distributions with the CDF $p_{CP} = 0.7 \cdot p_C(x) + 0.3 \cdot \frac{3}{(x+1.5)^4}\mathbb{I}\{x \geq -1.5\}$. Note that the latter two mixtures of distributions represent the asymmetric distributions. Although we do not consider the asymmetric noise above, we demonstrate the performance of the proposed framework for such noise distributions empirically.

To simulate multi-armed bandit, we use three environments: 10 arms and the ground-truth reward of the i-th arm $\mu_i = i, i = 0, \ldots, 9$, 10 arms and the ground-truth reward of the i-th arm $\mu_i = i/10, i = 0, \ldots, 9$, and 100 arms and the ground-truth reward of the i-th arm $\mu_i = i/50, i = 0, \ldots, 99$. We refer to these environments as `Env1`, `Env2` and `Env3`, respectively. Due to space limitations, only the first two environments and some distributions are presented in the article. Figures with all three environments for all considered distributions are presented in the supplementary materials [5].

Convergence comparison. To compare the convergence of the considered algorithms, we test the three environments mentioned above. Due to the space limitation, we only provide plots corresponding to the Cauchy distribution ($\gamma = 1$) of the reward noise ξ. The similar plots corresponding to the super-heavy tail distributions are presented in supplementary materials [5]. We use the hyperparameters of the algorithms, which give the best convergence. Figure 1 shows that the proposed algorithms outperform `RUCB-Median` and `APE` algorithms in `Env1` and `Env2`. HTINF and Clipped-INF-med-SMD show worse performance; therefore, for clarity, we exclude them from Figure 1. We show in the next paragraph (see Table 1) that our algorithms are significantly faster in terms of runtime since the per iteration costs are much smaller.

Runtime comparison. In addition to the convergence comparison presented in Figure 1, we also provide the runtime comparison for the considered algorithms. This experiment is essential for highlighting the difference in the costs per step in the discussed algorithms. We assign to every algorithm the budget for total pulls of arms equal to 10^4. We track the mean regret R_T/T and measure the runtime to achieve the target values of this metric. We test the target metrics R_T/T equal 0.1 and 0.05. If an algorithm does not achieve the target mean regret within the assigned budget, we consider such a run as a fail. We run 100 trials for every algorithm and show in Table 1 the 90% percentile of their running time. HTINF and Clipped-INF-med-SMD do not reach the target mean regret within the assigned budget and we exclude them from Table 1.

Runtime vs # arms. We compare how performance of the algorithms depends on the number of arms. We consider scenarios



(a) Env 1 (b) Env 2

(c) Env 1 (d) Env 2

Figure 1: The convergence of the regret metric (the first row) and the mean regret metric (the second row) for the considered algorithms with Cauchy distribution ($\gamma = 1$) of a reward noise. `APE` uses $\alpha = 0$. We report the averaged values over 120 trials and standard deviation area via shaded regions. Our algorithms show faster convergence compared to competitors.

Table 1: Runtime comparison of the considered algorithms to achieve the given values of R_T/T for the `Env1` with Cauchy distribution ($\gamma = 1$) as a reward noise. Our algorithms more often reach the target mean regret within the assigned budget and are significantly faster than `RUCB-Median` and `APE`. We highlight the best values for runtime, and # fails with bold.

Algorithms	Runtime for $R_T/T = 0.1$, ms.	# fails	Runtime for $R_T/T = 0.05$, ms	# fails
SGD-UCB-SMoM	**11.5**	**8**	**19.3**	**17**
SGD-UCB-Median	15.5	13	27.5	18
SGD-UCB	23.1	12	45.9	**17**
RUCB-Median	55.5	24	113.9	25
APE, $p = 2$	265.8	21	432.7	99

with $K \in \{10, 10^2, 10^3, 5 \cdot 10^3\}$ arms, and run every algorithm for $T = 15 \cdot 10^3$ steps. The reported timings are measured after the single run and shown in Figure 2. They demonstrate that the proposed algorithms are consistently faster than the baselines over the considered range of arms. This result aligns with the algorithm construction, where only an update for a 1D optimization problem is performed in every round.

4.2 Heavy-tail MAB.

For the heavy-tail MAB problem, we use the similar environments as in the previous section and Fréchet distribution with the CDF $p_F(x) = e^{-x^{-\beta}}$, where $\beta = 1.1$, to model the noise in the reward estimates. Figure 3 shows that our algorithms (`SGD-UCB` and `SGD-UCB-Median`) provide smaller mean regret for the considered number of steps in `Env2` than `RUCB-Median` and `UCB`. `Env1` is especially challenging for the proposed algorithms. The `RUCB-Median`

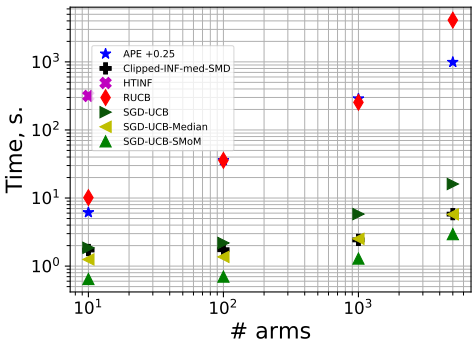
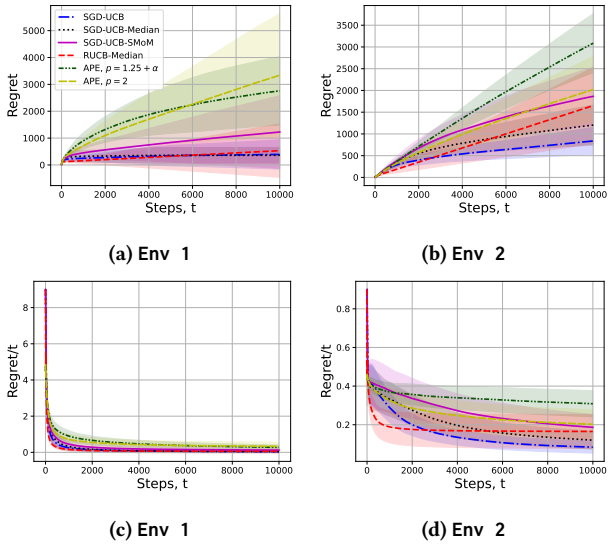


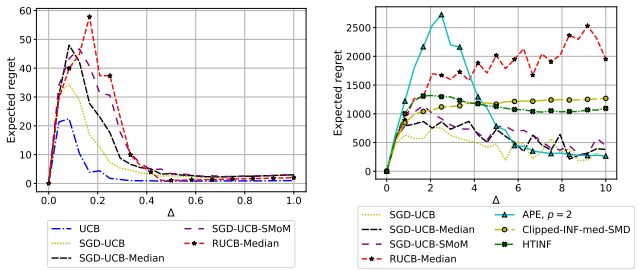
Figure 2: Dependence of runtime on the numbers of arms for $T = 15 \cdot 10^3$ steps. Our algorithms outperform baselines due to efficient gradient estimation and faster optimization steps. `RUCB-Median` and `APE` slow down with larger number of arms due to usage of the rewards history. `Clipped-INF-med-SMD` and `HTINF` solve optimization problems in every round and become slower with a larger number of arms. `HTINF` is shown only for 10 arms since it extremely slows for larger values.



(a) Env 1 (b) Env 2

(c) Env 1 (d) Env 2

Figure 3: The convergence of regret metric (the first row) and the mean regret metric (the second row) for the considered algorithms with Fréchet distribution ($\beta = 1.1$) of a reward noise. `APE` uses $\alpha = 0.1$. We report the averaged values over 120 trials and the corresponding standard deviation area via shaded regions. Our algorithms converge faster in `Env2` and slower than `RUCB-Median` in `Env1`.



(a) Gaussian MAB (b) Heavy tail MAB

Figure 4: (a) Mean regret for the considered algorithms in Gaussian MAB with two arms with rewards $\{0, \Delta\}$. Our algorithms distinguish arms with close rewards similar to the competitors. (b) Mean regret for the considered algorithms in heavy tail MAB with five arms with rewards $\{0, 0, 0, 0, \Delta\}$. Our algorithms distinguish arms with close rewards even if the noise is generated from the Cauchy distribution ($\gamma = 1$).

and `APE` provide smaller regret values in the considered number of steps. However, the difference between the regret given by the `RUCB-Median`, `APE` and the `SGD-UCB` is not large and the corresponding mean regret values are already almost the same. We do not show regret and mean regret for HTINF and Clipped-INF-med-SMD since they perform much worse than the shown algorithms.

4.3 MAB problem with hardly distinguished arms

To evaluate the robustness of the algorithms, we consider the bandit with arms, whose rewards are hardly distinguishable. In this experiment, we include the `UCB` algorithm [2] to baselines. For Gaussian MAB we consider two arms such that the corresponding rewards are $\{0, \Delta\}$. The values of Δ vary from 0 to 1 with a step size of 0.04. For heavy tail MAB, we consider Cauchy distribution ($\gamma = 1$) with 5 arms such that the corresponding rewards are $\{0, 0, 0, 0, \Delta\}$. The values of Δ vary from 0 to 10 with a step size of 0.4. We run 300 trial simulations for each environment for 2000 steps and average the final regret on trials. The result of the robustness analysis is presented in Figure 4. It shows that `UCB` is optimal for Gaussian MAB, and our algorithms are close to the `RUCB-Median` algorithm in terms of the expected regret for close arms rewards. We exclude HTINF, APE and Clipped-INF-med-SMD from this experiment since they were developed for the heavy tail rewards distribution. In the heavy tail environment, our algorithms show smaller expected regrets than competitors as arms become more distinguishable. We do not plot `UCB` since expected regret grows crucially and suffers readability.

5 CONCLUSION AND FUTURE WORK.

We suggested a new framework to construct UCB-type algorithms for stochastic multi-armed bandits with heavy tails. The main ingredient is to use $g(k, \delta)$-bounded algorithms for optimization problems with inexact oracle. As the main example, we propose Clipped-SGD-UCB algorithm and its particular instances SGD-UCB, SGD-UCB-Median and SGD-UCB-SMoM. The proposed algorithms show convergence even in the case of noise, which has no expectation. Future work includes the construction of the efficient algorithms fine-tuned for the proposed framework. Finding non-trivial zero-order algorithms appropriate to the proposed framework is also a promising research direction.

REFERENCES

[1] Shubhada Agrawal, Sandeep K Juneja, and Wouter M Koolen. 2021. Regret minimization in heavy-tailed bandits. In *Conference on Learning Theory*. PMLR, 26–62.

[2] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. 2002. Finite-time analysis of the multiarmed bandit problem. *Machine learning* 47 (2002), 235–256.

[3] Peter Auer, Nicolo Cesa-Bianchi, Yoav Freund, and Robert E Schapire. 2002. The nonstochastic multiarmed bandit problem. *SIAM journal on computing* 32, 1 (2002), 48–77.

[4] Sébastien Bubeck, Nicolo Cesa-Bianchi, and Gábor Lugosi. 2013. Bandits with heavy tail. *IEEE Transactions on Information Theory* 59, 11 (2013), 7711–7717.

[5] Yuriy Dorn, Aleksandr Katrutsa, Ilgam Latypov, and Andrey Pudovikov. 2024. Fast UCB-type algorithms for stochastic bandits with heavy and super heavy symmetric noise. *arXiv preprint arXiv:2402.07062* (2024).

[6] Yuriy Dorn, Nikita Kornilov, Nikolay Kutuzov, Alexander Nazin, Eduard Gorbunov, and Alexander Gasnikov. 2024. Implicitly normalized forecaster with clipping for linear and non-linear heavy-tailed multi-armed bandits. *Computational Management Science* 21, 1 (2024), 19.

[7] Gianmarco Genalti, Lupo Marsigli, Nicola Gatti, and Alberto Maria Metelli. 2023. Towards Fully Adaptive Regret Minimization in Heavy-Tailed Bandits. *arXiv preprint arXiv:2310.02975* (2023).

[8] Elad Hazan et al. 2016. Introduction to online convex optimization. *Foundations and Trends® in Optimization* 2, 3-4 (2016), 157–325.

[9] Jiatai Huang, Yan Dai, and Longbo Huang. 2022. Adaptive Best-of-Both-Worlds Algorithm for Heavy-Tailed Multi-Armed Bandits. In *International Conference on Machine Learning*. PMLR, 9173–9200.

[10] Nikita Kornilov, Yuriy Dorn, Aleksandr Lobanov, Nikolay Kutuzov, Innokentiy Shibaev, Eduard Gorbunov, Alexander Gasnikov, and Alexander Nazin. 2024. Zeroth-order Median Clipping for Non-Smooth Convex Optimization Problems with Heavy-tailed Symmetric Noise. *arXiv preprint arXiv:2402.02461* (2024).

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

[12] Kyungjae Lee and Sungbin Lim. 2022. Minimax Optimal Bandits for Heavy Tail Rewards. *IEEE Transactions on Neural Networks and Learning Systems* (2022).

[13] Kyungjae Lee, Hongjun Yang, Sungbin Lim, and Songhwai Oh. 2020. Optimal algorithms for stochastic multi-armed bandits with heavy tailed rewards. *Advances in Neural Information Processing Systems* 33 (2020), 8452–8462.

[14] Shiyin Lu, Guanghui Wang, Yao Hu, and Lijun Zhang. 2019. Optimal algorithms for Lipschitz bandits with heavy-tailed rewards. In *International Conference on Machine Learning*. PMLR, 4154–4163.

[15] Andres Munoz Medina and Scott Yang. 2016. No-regret algorithms for heavy-tailed linear bandits. In *International Conference on Machine Learning*. PMLR, 1642–1650.

[16] Francesco Orabona. 2019. A modern introduction to online learning. *arXiv preprint arXiv:1912.13213* (2019).

[17] Nikita Puchkin, Eduard Gorbunov, Nikolay Kutuzov, and Alexander Gasnikov. 2023. Breaking the Heavy-Tailed Noise Barrier in Stochastic Optimization Problems. *arXiv preprint arXiv:2311.04161* (2023).

[18] Herbert Robbins. 1952. Some aspects of the sequential design of experiments. *Bull. Amer. Math. Soc.* 58, 6 (1952), 527–535.

[19] Han Shao, Xiaotian Yu, Irwin King, and Michael R Lyu. 2018. Almost optimal algorithms for linear stochastic bandits with heavy-tailed payoffs. *Advances in Neural Information Processing Systems* 31 (2018).

[20] Jiujia Zhang and Ashok Cutkosky. 2022. Parameter-free regret in high probability with heavy tails. *Advances in Neural Information Processing Systems* 35 (2022), 8000–8012.

[21] Julian Zimmert and Teodor Vanislavov Marinov. 2024. PRODuctive bandits: Importance Weighting No More. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

[22] Julian Zimmert and Yevgeny Seldin. 2019. An optimal algorithm for stochastic and adversarial bandits. In *The 22nd International Conference on Artificial Intelligence and Statistics*. PMLR, 467–475.