Regret Guarantees for a UCB-based Algorithm for Volatile Combinatorial Bandits

Extended Abstract

Abhishek Kumar
International Institute of Information Technology, Hyderabad
Hyderabad, India
kumar.abhishek@research.iiit.ac.in

Andra Siva Sai Teja
Indian Institute of Technology Hyderabad
Hyderabad, India
ai22mtech11001@iith.ac.in

Ganesh Ghalme
Indian Institute of Technology Hyderabad
Hyderabad, India
ganeshghalme@ai.iith.ac.in

Sujit Gujar
International Institute of Information Technology, Hyderabad
Hyderabad, India
sujit.gujar@iiit.ac.in

Y. Narahari
Indian Institute of Science, Bengaluru
Bengaluru, India
narahari@iisc.ac.in

ABSTRACT

We study the combinatorial multi-armed bandit (MAB) problem with an additional constraint that an arbitrary subset of arms is unavailable at any given time instant. We refer to this setting as a volatile combinatorial MAB setting. The bandit algorithm must pull a subset of arms from the set of available arms to minimize the regret. Under some mild smoothness conditions, we show that the proposed CV-UCB algorithm—a straightforward extension of well-known C-UCB algorithm—achieves a $O(\log(T))$ instance-dependent regret guarantee under a semi-bandit feedback setting. We further show that under some mild restrictions on the range of reward functions, CV-UCB incurs $O(\sqrt{T\log(T)})$ regret, which we call *weak* instance-independent regret. We further show that the instance-independent regret of $O(\sqrt[3]{T^2\log(T)})$ for CV-UCB algorithm, completing the hierarchy of regret guarantees obtained by gradually relaxing the dependence on the instance parameters.

KEYWORDS

Regret Analysis, Multi-armed Bandits

ACM Reference Format:
Abhishek Kumar, Andra Siva Sai Teja, Ganesh Ghalme, Sujit Gujar, and Y. Narahari. 2025. Regret Guarantees for a UCB-based Algorithm for Volatile Combinatorial Bandits: Extended Abstract. In *Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Detroit, Michigan, USA, May 19 – 23, 2025*, IFAAMAS, 3 pages.

1 INTRODUCTION

We study a combination of two well studied extensions of classical stochastic MABs; *sleeping/ volatile bandits* [2, 4, 11] and *combinatorial bandits* [7–9]. In the volatile bandits setting, a subset of arms is unavailable at any given time instant. This variant, sometimes also known as mortal bandits [3], models many real-world scenarios such as crowdsourcing [13], online advertising [3, 10], and network routing [2, 11], where, an algorithm is restricted to select from only the available set of choices.

Studying these two variants together poses interesting technical questions. First, as the regret notion is modified to accommodate the volatile nature of the arms, it is unclear priori that the algorithms that achieve optimal regret guarantee for the *non-volatile* case will produce optimal regret guarantees for the volatile case. Second, in contrast to the non-volatile bandit's case, the optimal super-arm may change at each time instant. Hence, it is unclear whether the regret guarantees for learning in the non-volatile case translate to the volatile case in the general reward case, i.e., whether it will prohibitively elongate the learning process. In this paper, we address both of the above questions. A longer version of the paper with detailed proofs and simulation analysis is available online [1].

1.1 Model and Assumptions

We consider a volatile combinatorial bandits problem with $[k] := \{1, 2, \cdots, k\}$ denoting the set of *base* arms and $\boldsymbol{\mu} \in [0,1]^k$, the vector of unknown mean qualities of the base arms. Similar to the classical stochastic MAB problem, each base arm i corresponds to an unknown distribution $\mathcal{D}_i \in \Delta([0,1])$ with mean $\mu_i \in [0,1]$ over its quality. At each time instant t, base arms belonging to a subset $A_t \subseteq [k]$ become available. We consider that A_t is an arbitrary non-empty subset. An algorithm can pull any non-empty subset $S_t \subseteq A_t$ of arms and receive a reward $R(S_t, \boldsymbol{\mu})$. The reward depends on the selected subset S_t and the mean qualities of the arms, $\boldsymbol{\mu}$. We write $R_S := R(S, \boldsymbol{\mu})$. Furthermore, note that the reward depends only on the qualities of the arms pulled by the algorithm, i.e, S_t. We remark here that the classical stochastic bandits setting is a special case of our setting with $A_t = [k], |S_t| = 1$ and $R_t = X_{S_t,t}$ for all t.

For a given reward function $R(.)$, the problem reduces to one of finding a reward-maximizing subset of arms. This problem, even when the qualities of the base arms are known, is known to be NP-hard [14]. However, many important settings, such as sub-modular reward functions, admit polynomial time approximation schemes that provide an attractive approximation guarantee. We assume existence of a (γ, β)-approximation oracle denoted by (γ, β)-Oracle,

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).

which, given an availability set A and a quality vector $\boldsymbol{\mu}$, outputs a set S such that $R(S,\boldsymbol{\mu}) \geq \gamma \cdot R(S',\boldsymbol{\mu})$, for all $S' \in 2^A$ with the probability of at least β, with $\gamma, \beta \in (0,1]$. This can be also written as $S := \textsc{Oracle}(\boldsymbol{\mu}, A)$. This computation oracle separates the learning task from the offline computation task and is extensively used in the literature [5, 6, 9]. For the semi-bandit feedback to work effectively, we assume below smoothness properties.

Property 1. ***Monotonicity**: Let $\boldsymbol{\mu}, \boldsymbol{\mu}' \in [0,1]^k$ be two vectors such that $\boldsymbol{\mu}'_i \geq \boldsymbol{\mu}_i$ for all $i \in [k]$, then, for any $S \subseteq [k]$, $R(S,\boldsymbol{\mu}') \geq R(S,\boldsymbol{\mu})$.*

Property 2. ***Lipschitz Continuity**: There exists a real valued constant $C \geq 1$ such that for all $S \subseteq [k]$, we have $|R(S,\boldsymbol{\mu}) - R(S,\boldsymbol{\mu}')| \leq C \cdot \max_{i\in S} |\boldsymbol{\mu}_i - \boldsymbol{\mu}'_i|$.*

Property 3. ***Bounded Smoothness**: There exists a strictly increasing function f such that for any $S \subseteq [k]$, $|R(S,\boldsymbol{\mu}) - R(S,\boldsymbol{\mu}')| \leq f(\Lambda)$ whenever $\max_{i\in S} |\boldsymbol{\mu}_i - \boldsymbol{\mu}'_i| \leq \Lambda$.*

To evaluate the performance of an algorithm with limited availability of arms, we extend the notion of regret appropriately and call it a *volatile bandit regret* given by

$$\mathcal{R}_{\textsc{Alg}}(T) := \max_{(A_t)_{t=1}^T} \mathbb{E}_{\textsc{Alg}}\Big[\sum_{t=1}^{T}(R(S_t^\star,\boldsymbol{\mu}) - R(S_t,\boldsymbol{\mu}))\Big]. \tag{1}$$

Here, $S_t^\star \in \arg\max_{S\subseteq A_t} R(S,\boldsymbol{\mu})$. Note that when $A_t = [k]$ for all t, we recover the setting of [6]. Next, we define the regret in the presence of (γ,β)-oracle. Let, B_t be the event that an oracle returns an γ-approximate solution at time t i.e. $B_t = \{R_{S_t} \geq \gamma \cdot R_{S_t^\star}\}$. Note that $\mathbb{P}(B_t) \geq \beta$. The expected volatile bandit regret of $\textsc{Alg}$ with Oracle access is given by,

$$\mathcal{R}_{\textsc{Alg}}(T) = \max_{(A_t)_{t=1}^T} \mathbb{E}_{\textsc{Alg}}\Big[\sum_{t=1}^{T}(\gamma \cdot \beta \cdot R_{S_t^\star} - R_{S_t})\Big]. \tag{2}$$

Notational Setup: For each base arm $i \in [k]$, let $N_{i,t}$ denote the number of time instances arm i is pulled till t and let $\hat{\mu}_{i,t}$ be the average reward obtained from these pulls. Further, let

$$\overline{\mu}_{i,t} := \hat{\mu}_{i,t} + \sqrt{3\log(t)/2N_{i,t}}. \tag{3}$$

We call $\overline{\mu}_{i,t}$ as the UCB estimate of arm i at time t and let $\Delta_S := \gamma \cdot \textsc{opt}_A - R_S$ be the regret incurred by pulling super-arm S. Here, $\textsc{opt}_A := R_{S^\star} = \max_{S\subseteq A} R_S$ denotes the optimal reward when the set of available arms is A. A super-arm $S \subseteq A$ is *bad* (sub-optimal), if $\Delta_S > 0$. For a given $A \subseteq [k]$, we define the set of bad super-arms as $S_B(A) = \{S \subseteq A | \Delta_S > 0\}$. Further, for a given $A \subseteq [k]$, define $\Delta_{\min}(A) = \gamma \cdot \textsc{opt}_A - \max_{S\in S_B(A)} R_S$ and $\Delta_{\max}(A) = \gamma \cdot \textsc{opt}_A - \min_{S\in S_B(A)} R_S$. Note that, for any availability set A, we have $\Delta_{\max}(A) \geq \Delta_{\min}(A) > 0$. The strict inequality follows from the definition of $S_B(A)$. Next, define $\Delta_{\max} = \max_{A\subseteq[k]} \Delta_{\max}(A)$ and $\Delta_{\min} = \min_{A\subseteq[k]} \Delta_{\min}(A)$.

2 CV-UCB ALGORITHM AND RESULTS

CV-UCB Algorithm: At each time t, CV-UCB receives the set of available arms A_t. If there is a base arm in A_t which is not pulled previously, an algorithm pulls all the available arms. For each time instance where all available arms are pulled atleast once, CV-UCB obtains $S_t = \textsc{Oracle}(\overline{\boldsymbol{\mu}}_t, A_t)$. The algorithm then pulls a super-arm S_t and obtains rewards $R(S_t,\boldsymbol{\mu})$ and an individual base arm rewards (semi-bandit feedback) $X_{i,t}$ for each $i \in S_t$.

Theorem 2.1. *The (instance-dependent) expected regret incurred by CV-UCB when the reward function satisfies Lipschitz condition (Properties 1 and 2) is given by*

$$\mathcal{R}_{\textit{CV-UCB}}(T) \leq 2\beta kC(\zeta(3)(1 + \sqrt{3\log(T)/2}) + 3\sigma C\log(T)/\Delta_{\min})$$

Here, ζ is the Reimann zeta function and $\sigma = \Delta_{\max}/\Delta_{\min}$.

We remark here that for smaller values of $\Delta_{\min}$, the regret bound of Thm. 2.1 becomes vacuous. We now prove the *weak* instance-dependent regret bound (see Theorem 2.2) and instance-independent regret bound (Thm. 2.3) of the CV-UCB algorithm.

Theorem 2.2. *The (weak instance-dependent) expected regret of CV-UCB when the reward function satisfies Lipschitz condition (Properties 1 and 2) is given by*

$$\mathcal{R}_{\textit{CV-UCB}}(T) \leq 4C\sqrt{6k\sigma T\log(T)} + 2kC\zeta(3).$$

Here, $\zeta(.)$ is a Reimann zeta function and $\sigma = \Delta_{\max}/\Delta_{\min}$.

Observe that the regret guarantee increases from $O(\log(T))$ to $O(T\log(T))$ in the weak instance-dependent setting, where the dependence shifts from the minimum reward gap $\Delta_{\min}$ to the reward ratio σ. In our next result, we further mitigate the dependence on σ to achieve an instance-independent regret guarantee.

Theorem 2.3. $\lambda = (1 + \sqrt{3\log(T)/2})$. *The instance-independent volatile bandit regret of CV-UCB when the reward function satisfies Lipschitz condition (Properties 1 and 2) is given by*

$$\mathcal{R}_{\textit{CV-UCB}}(T) \leq C(1+\lambda) \cdot \sqrt[3]{6kT^2\log(T)} + 2k\lambda C\zeta(3).$$

Next, we consider the bounded smoothness condition.

Theorem 2.4. *The expected volatile bandits regret incurred by CV-UCB when the reward function satisfies bounded smoothness condition (Properties 1 and 3), is given by*

$$\mathcal{R}_{\textit{CV-UCB}}(T) \leq \big(6\log(T)/(f^{-1}(\Delta_{\min}))^2 + 2\zeta(3)\big)k \cdot \Delta_{\max}.$$

Theorem 2.5. *Let $f(x) = p \cdot x^q$ where $p > 0$ and $q \in (0,1]$ and define $r = \frac{q}{q+2}$. The (instance independent) expected volatile bandits regret incurred by CV-UCB when the reward function satisfies bounded smoothness condition (Properties 1 and 3) is given by*

$$\mathcal{R}_{\textit{CV-UCB}}(T) \leq (k\Delta_{\max} + 1) \cdot 6^r \log(T)^r T^{1-r} p^{1-r} + 2k\zeta(3)\Delta_{\max}.$$

3 FUTURE WORK

Extending regret guarantees of existing MAB algorithms such as Thompson sampling and ε-greedy to CV bandit setting is an interesting future direction. A tight instance-independent regret bound under bounded smoothness setting still remains an open problem along with a finely tuned analysis with availability specific instead of worst-case regret bound. Furthermore, a study of CV bandits where the pulling strategy affects future unavailability of arms such as in rotting bandits [12] is also an interesting future direction.

ACKNOWLEDGMENTS

Ganesh Ghalme gratefully acknowledges support from ANRF under grant CRG/2022/007927 and Sujit Gujar under CRG/2022/004980.

REFERENCES

[1] Kumar Abhishek, Ganesh Ghalme, Sujit Gujar, and Yadati Narahari. 2021. Sleeping Combinatorial Bandits. *CoRR* abs/2106.01624 (2021). arXiv:2106.01624 https://arxiv.org/abs/2106.01624

[2] Zahy Bnaya, Rami Puzis, Roni Stern, and Ariel Felner. 2013. Volatile Multi-Armed Bandits for Guaranteed Targeted Social Crawling. *the proceedings of AAAI 2013* 2, 2.3 (2013), 16–21.

[3] Deepayan Chakrabarti, Ravi Kumar, Filip Radlinski, and Eli Upfal. 2009. Mortal Multi-Armed Bandits. In *the proceedings of NIPS 2009*.

[4] Aritra Chatterjee, Ganesh Ghalme, Shweta Jain, Rohit Vaish, and Y Narahari. 2017. Analysis of Thompson Sampling for Stochastic Sleeping Bandits.. In *the proceedings of UAI 2017*.

[5] Wei Chen, Wei Hu, Fu Li, Jian Li, Yu Liu, and Pinyan Lu. 2016. Combinatorial multi-armed bandit with general reward functions. In *the proceedings of NIPS 2016*.

[6] Wei Chen, Yajun Wang, and Yang Yuan. 2013. Combinatorial multi-armed bandit: General framework and applications. In *the proceedings of ICML 2013*.

[7] Wei Chen, Yajun Wang, Yang Yuan, and Qinshi Wang. 2016. Combinatorial multi-armed bandit and its extension to probabilistically triggered arms. *the proceedings of JMLR 2016* 17, 1 (2016).

[8] Richard Combes, Mohammad Sadegh Talebi Mazraeh Shahi, Alexandre Proutiere, and marc lelarge. 2015. Combinatorial Bandits Revisited. In *the proceedings of NIPS 2015*.

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

[10] Ganesh Ghalme, Swapnil Dhamal, Shweta Jain, Sujit Gujar, and Y. Narahari. 2021. Ballooning multi-armed bandits. *Artificial Intelligence* 296 (2021), 103485. https://doi.org/10.1016/j.artint.2021.103485

[11] Robert Kleinberg, Alexandru Niculescu-Mizil, and Yogeshwer Sharma. 2010. Regret bounds for sleeping experts and bandits. *the proceedings of Machine learning 2010* 80, 2-3 (2010).

[12] Nir Levine, Koby Crammer, and Shie Mannor. 2017. Rotting bandits. In *the proceedings of NIPS 2017*.

[13] Fengjiao Li, Jia Liu, and Bo Ji. 2019. Combinatorial sleeping bandits with fairness constraints. *IEEE Transactions on Network Science and Engineering 2019* (2019).

[14] Laurence A Wolsey and George L Nemhauser. 1999. *Integer and combinatorial optimization*. the proceedings of John Wiley & Sons 1999.