Multi-Agent Learning: From Theory to Practice

Report of "The Non Stochastic Multi-Armed Bandit Problem"

Carlos Alberto Santillán Moreno, M.Sc. student

July 15th 2023

1 Summary

The paper [5] tackles the *adversarial* multi-armed bandit problem. In this context, an agent is presented with K different arms, but their rewards are not drawn from fixed distributions (for a thorough analysis in the stochastic case, see Auer et al. [3]). Instead, each arm has been assigned a sequence of rewards, i.e., one reward for each round t. The reward of the i-th arm at round t is denoted as $x_i(t)$. We aim to minimize the *weak regret* over the time horizon T, which is the difference between $G_{\max}(T)$, the cumulative reward of the globally best single action, and $G_A(T)$, the cumulative reward of algorithm A. The adversarial setting was originally presented in Auer et al. [4]. The main algorithm proposed in the paper is **Exp3** (see Algorithm 1).

The idea behind **Exp3**, and all of its variants, is to update the probability $p_i(t)$ of pulling arm i if the arm is promising. We start from a uniform distribution since all weights are equal to 1, therefore all arms are equally likely to be chosen. At each round t we pull an arm according to the probabilities $p_1(t), \ldots, p_K(t)$ and get a reward $x_{i_t}(t)$. We introduce the estimated reward $\hat{x}(t)$, which is zero for all arms that were not pulled at round t, and $x_j(t)/p_j(t)$ for the pulled arm. The factor $1/p_j(t)$ compensates for the probability of taking that action. This guarantees that $\mathbb{E}[\hat{x}_j(t)|i_1, \ldots, i_{t-1}] = x_j(t)$, i.e., expectations are equal to the actual reward for each action. The last step is updating the weights by an exponential factor $\gamma \cdot \hat{x}_j(t)/K$, which will make the algorithm more likely to choose arms with higher rewards (i.e., the more promising ones). By assigning different probabilities to each arm we try to balance exploration and exploitation, so it is still possible to pull less promising arms, albeit with a lower probability. It can be proven that

Algorithm 1 Exp3

Parameters: Real $\gamma \in (0,1]$
Initialization: $w_i(1) = 1$ for $i = 1, \ldots, K$

1: **for** $t = 1, 2, \ldots$ **do**
2: Set $p_i(t) = (1-\gamma)\frac{w_i(t)}{\sum_{j=1}^{K} w_j(t)} + \frac{\gamma}{K}$ for $i = 1, \ldots, K$
3: Draw i_t randomly according to the probabilities $p_1(t), \ldots, p_K(t)$
4: Receive reward $x_{i_t}(t) \in [0, 1]$
5: **for** $j = 1, \ldots, K$ **do**

$$\hat{x}_j(t) = \begin{cases} x_j(t)/p_j(t) & \text{if } j = i_t \\ 0 & \text{otherwise} \end{cases}$$

$$w_j(t+1) = w_j(t) \exp(\gamma \hat{x}_j(t)/K)$$

6: **end for**
7: **end for**

by a proper choice of the parameter γ the *expected* weak regret of **Exp3** is $O(\sqrt{gK \ln K})$, where $g \geq G_{\max}$[1].

Exp3 can be further modified to work without knowledge of T, provide more guarantees on the regret bound, or both. The paper presents **Exp3.1**, which is designed to work without knowing T. **Exp3.1** "guesses" g and calls **Exp3**. When the *estimated* cumulative reward of an arm is larger than g, **Exp3.1** restarts **Exp3** with an updated g (each restart corresponds to an epoch). **Exp3.1** has a regret of $O(\sqrt{KT \ln K})$, which in general can be worse than **Exp3**.

A problem of **Exp3** is that its expected regret has a large variance, because the estimated rewards $\hat{x}_i(t)$ can have a variance close to $1/p_i(t)$, which can lead to a regret variance as large as $T^{3/4}$. **Exp3.P** addresses this by using confidence bounds on the estimated cumulative reward, which leads to a *weak regret* (not its expectation) of $O(\sqrt{KT \ln(KT/\delta)})$with probability at least $1 - \delta$. Using a similar approach to **Exp3.1** we can obtain **Exp3.P.1**, which does not require knowledge of T, and also keeps the variance under control.

Another variant is **Exp4**, we rely on "expert" advice, experts are defined as a probability distribution over the actions at each round, so we choose the most promising expert rather than the arm itself. The last variant, **Exp3.S**,

[1] Assuming rewards in the range $[0, 1]$ we can set $g = T$, which requires knowledge of T

generalizes the regret against any sequence of actions $(j_1, \ldots, j_T)$, not only the globally best single action. However, the bound on the expected regret depends on the "hardness" of the sequence, where the hardness is the number of times one has to change the arm to follow the sequence plus 1.

2 Technical Evaluation

We provide a short version of the proof of the upper bound of the expected regret of **Exp3**.

Theorem 2.1 *For any $K > 0$ and for any $\gamma \in (0, 1]$*

$$G_{\max} - \mathbb{E}[G_{\mathbf{Exp3}}] \leq (e-1)\gamma G_{\max} + \frac{K \ln K}{\gamma}$$

holds for any assignment of rewards and for any $T > 0$

Let $W_t = w_1(t) + \cdots + w_K(t)$. We bound the ratio W_{t+1}/W_t for all sequences $i_1, \ldots, i_T$ drawn by **Exp3**.

$$\begin{aligned}
\frac{W_{t+1}}{W_t} &= \sum_{i=1}^{K} \frac{w_i(t+1)}{W_t} \\
&= \sum_{i=1}^{K} \frac{p_i(t) - \frac{\gamma}{K}}{1-\gamma} \exp\left(\frac{\gamma}{K}\hat{x}_i(t)\right) \\
&\leq 1 + \frac{\frac{\gamma}{K}}{1-\gamma} x_{i_t}(t) + \frac{(e-2)(\frac{\gamma}{K})^2}{1-\gamma} \sum_{i=1}^{K} \hat{x}_i(t).
\end{aligned}$$

By taking the logarithm of the ratio and remembering some well-known inequalities we obtain

$$\ln \frac{W_{t+1}}{W_t} \leq \frac{\frac{\gamma}{K}}{1-\gamma} x_{i_t}(t) + \frac{(e-2)(\frac{\gamma}{K})^2}{1-\gamma} \sum_{i=1}^{K} \hat{x}_i(t)$$

Summing over t we get

$$\ln \frac{W_{T+1}}{W_1} \leq \frac{\frac{\gamma}{K}}{1-\gamma} G_{\mathbf{Exp3}} + \frac{(e-2)(\frac{\gamma}{K})^2}{1-\gamma} \sum_{t=1}^{T} \sum_{i=1}^{K} \hat{x}_i(t)$$

We know that for any action j

$$\ln \frac{W_{T+1}}{W_1} \geq \ln \frac{w_j(T+1)}{W_1} = \frac{\gamma}{K} \sum_{t=1}^{T} \hat{x}_j(t) - \ln K$$

By subtracting the last two expressions we obtain

$$G_{\mathbf{Exp3}} \geq (1-\gamma) \sum_{t=1}^{T} \hat{x}(t) - \frac{K \ln K}{\gamma} - (e-2) \frac{\gamma}{K} \sum_{t=1}^{T} \sum_{i=1}^{K} \hat{x}_i(t)$$

We can take the expection on both sides w.r.t. the distribution of the chosen arms, we get

$$\mathbb{E}[G_{\mathbf{Exp3}}] \geq (1-\gamma) \sum_{t=1}^{T} x_j(t) - \frac{K \ln K}{\gamma} - (e-2) \frac{\gamma}{K} \sum_{t=1}^{T} \sum_{i=1}^{K} x_i(t)$$

where we exploited the fact that $\mathbb{E}[\hat{x}_j(t)|i_1, \ldots, i_{t-1}] = x_j(t)$. Considering that we chose the action j arbitrarily, and that

$$\sum_{t=1}^{T} \sum_{i=1}^{K} x_i(t) \leq K G_{\max}$$

we obtain the inequality stated by the theorem. ■

All the bounds presented in the paper are proven following a similar reasoning. In particular, they make use of the following observations

$$\hat{x}_i(t) \leq 1/p_i(t) \leq K/\gamma,$$

$$\sum_{i=1}^{K} p_i(t) \hat{x}_i(t) = p_{i_t}(t) \frac{x_{i_t}(t)}{p_{i_t}(t)} = x_{i_t}(t),$$

$$\sum_{i=1}^{K} p_i(t) \hat{x}_i(t)^2 = p_{i_t}(t) \frac{x_{i_t}(t)}{p_{i_t}(t)} \hat{x}_{i_t}(t) \leq \hat{x}_{i_t}(t) = \sum_{i=1}^{K} \hat{x}_i(t).$$

The paper also presents proofs for the optimality of parameters such as γ, or also α in **Exp3.P**.

The assumptions made are as general as possible for the adversarial setting. For example, we assume that rewards belong to the interval $[0, 1]$. However, this does not limit the generalization of the algorithms presented. We can consider any reward interval $[a, b]$ (with $a < b$) by scaling the rewards accordingly.

$$\frac{x_j(t) - a}{b - a}$$

so we end up with a $[0, 1]$ interval. This obviously requires knowledge of the interval.

To corroborate the findings outlined in the paper, we present the results obtained by a Python implementation[2] of some of the algorithms. We consider binary rewards, 10 arms, and $T = 100000$. The rewards for each arm were assigned at each round with these probabilities [0.30, 0.28, 0.26, 0.24, 0.22, 0.19, 0.17, 0.15, 0.14, 0.10], so the best arm is the first one. We show the performance of **Exp3** with the optimal γ (Figure 1).

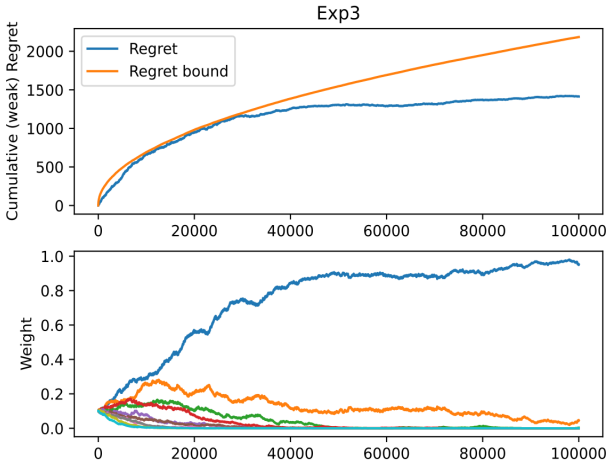

Figure 1: Cumulative regret and weights of **Exp3** over the time horizon

We now show the performance of **Exp3.1** on the same dataset of rewards (Figure 2).

Each epoch is clearly visible, since the weights are reset to 1. As expected, **Exp3.1** has a higher regret than **Exp3**. This is caused by the loss of information about T.

We also show the performance of **Exp3.P** on the same dataset (Figure 3). As we can see, the theoretical bound on the regret is significantly higher compared to **Exp3**. However, in this particular instance, we are well below the theoretical bound, performing only slightly worse than **Exp3**.

These results are of course subject to the randomness of the algorithms, and performance may vary significantly from realization to realization.

[2]The code is available at this repository `https://github.com/Plinkett/MALReport`

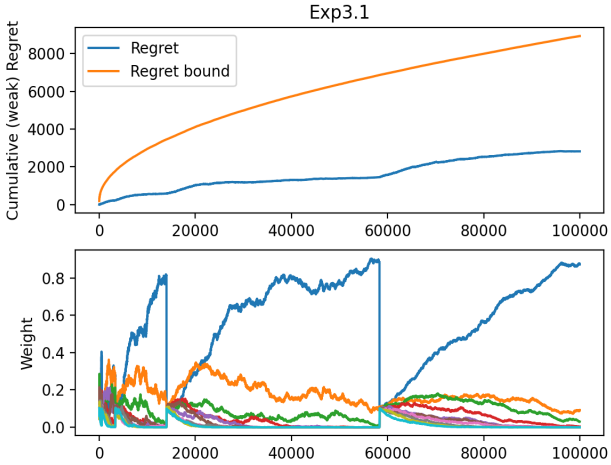

Figure 2: We can observe the weights being reset at each epoch. Also, it appears to converge faster to the optimal arm (weight in blue) for each successive epoch

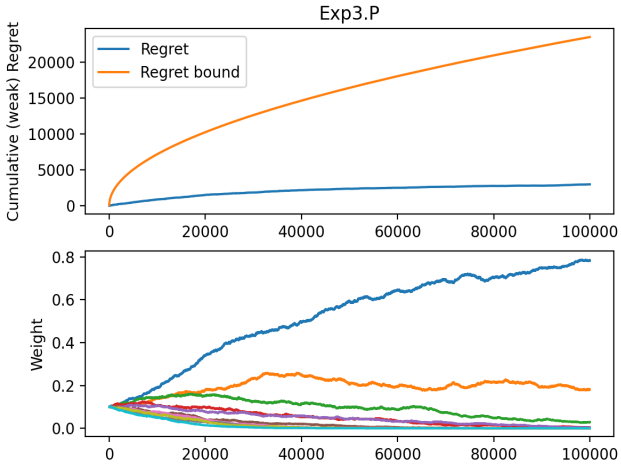

Figure 3: The theoretical upper bound is much higher than those of **Exp3** and **Exp3.1**. However, the actual regret in this realization is approximately 2500

3 Advantages and Drawbacks

The paper starts by presenting the adversarial setting and the necessity of dropping the stochasticity assumption. It proposes an efficient algorithm to solve the problem. However, it also adapts it to work in environments where some variables are unknown, such as the time horizon T (solved with **Exp3.1**).

Additionally, the paper addresses less obvious issues that can impact performance, such as the variance of the expected regret. To mitigate this problem, the authors propose **Exp3.P**. A strong point of this work is the wide selection of **Exp3** variants shown, especially considering the scarcity of literature on the subject at the time of publication.

The paper adequately presents the non-trivial mathematical instruments required for each proof, e.g., well-known inequalities. An example is the use of Jensen's inequality in the proof of the bound for **Exp3.1**, that is, given a random variable X and a convex function φ, then $\varphi(\mathbb{E}[X]) \leq \mathbb{E}[\varphi(X)]$.

Clarity of execution is greatly appreciated, with careful justification of every non-obvious step, given the depth and rigor required for each proof.

The final part on applications to game theory is adequately motivated and coherent with the previous results.

The main weakness of this work is perhaps the abrupt switch to **Exp4** and **Exp3.S**. Both variants are not as natural as **Exp3.1**, **Exp3.P**, and **Exp3.P.1**. They seem to be preliminary work for a more complete analysis, and they lack additional context. For example, there is no variant of **Exp4** that guarantees a regret bound with high probability. Beygelzimer [6], using a similar technique to **Exp3.P**, proposed **Exp4.P**, which addresses this issue.

4 Improvements and Extensions

Over 20 years have passed since the publication of this work, a wide family of algorithms has been developed, and the setting has been further extended.

An open question is whether the degree of exploration can be reduced without compromising performance. For example, in a problem with many suboptimal arms, a significant amount of time may be wasted despite continuously adapting drawing distributions. Recently, Neu [13] proposed "implicit exploration" techniques, which aim to minimize the number of necessary draws for an optimal guarantee.

We can also consider a hybrid setting where we have stochastic and adversarial rewards at intervals, which is also referred to as "adversarial corruption." The problem has been studied in Altschuler [2], Kapoor [11], and

Jun[10]. An interesting solution that achieves (almost) optimal regret in both cases is the **SOA** algorithm (Bubeck et al. [7]). However, it does not directly address the corruption problem. Gupta [9] recently proposed **BARBAR**, an algorithm capable of handling adversarial corruption up to a constant C. It achieves regret (with high probability) of $O(KC + \sum_{i \neq i^\star} \frac{\log T}{\Delta_i} \log(\frac{K}{\delta} \log T))$.

The setting can be further enriched by allowing the adversary to adapt to our actions. Formally, this means that the adversarial choice of assigning a reward at round t is a function of our previous actions (i.e., the arms pulled at rounds $t-1, t-2, \dots$). Dekel et al. [8] proposed a new notion of regret for addressing this problem called policy regret.

Lastly, we can also encounter situations where we have information about the other arms' rewards. This is a common occurence in web advertising or sensor networks for example. Graph-based approaches for modelling the relationship between the rewards have been developed by Mannor [12] and Alon et al. [1].

References

[1] Noga Alon, Nicolò Cesa-Bianchi, Claudio Gentile, and Yishay Mansour. From bandits to experts: A tale of domination and independence. In Christopher J. C. Burges, Léon Bottou, Zoubin Ghahramani, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pages 1610–1618, 2013.

[2] Jason M. Altschuler, Victor-Emmanuel Brunel, and Alan Malek. Best arm identification for contaminated bandits. *CoRR*, abs/1802.09514, 2018.

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

[4] Peter Auer, Nicolò Cesa-Bianchi, Yoav Freund, and Robert E. Schapire. Gambling in a rigged casino: The adversarial multi-armed bandit problem. *Electron. Colloquium Comput. Complex.*, TR00-068, 2000.

[5] Peter Auer, Nicolò Cesa-Bianchi, Yoav Freund, and Robert E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM J. Comput.*, 32(1):48–77, 2002.

[6] Alina Beygelzimer, John Langford, Lihong Li, Lev Reyzin, and Robert E. Schapire. An optimal high probability algorithm for the contextual bandit problem. *CoRR*, abs/1002.4058, 2010.

[7] Sébastien Bubeck and Aleksandrs Slivkins. The best of both worlds: Stochastic and adversarial bandits. In Shie Mannor, Nathan Srebro, and Robert C. Williamson, editors, *COLT 2012 - The 25th Annual Conference on Learning Theory, June 25-27, 2012, Edinburgh, Scotland*, volume 23 of *JMLR Proceedings*, pages 42.1–42.23. JMLR.org, 2012.

[8] Ofer Dekel, Ambuj Tewari, and Raman Arora. Online bandit learning against an adaptive adversary: from regret to policy regret. In *Proceedings of the 29th International Conference on Machine Learning, ICML 2012, Edinburgh, Scotland, UK, June 26 - July 1, 2012*. icml.cc / Omnipress, 2012.

[9] Anupam Gupta, Tomer Koren, and Kunal Talwar. Better algorithms for stochastic bandits with adversarial corruptions. *CoRR*, abs/1902.08647, 2019.

[10] Kwang-Sung Jun, Lihong Li, Yuzhe Ma, and Xiaojin (Jerry) Zhu. Adversarial attacks on stochastic bandits. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 3644–3653, 2018.

[11] Sayash Kapoor, Kumar Kshitij Patel, and Purushottam Kar. Corruption-tolerant bandit learning. *Mach. Learn.*, 108(4):687–715, 2019.

[12] Shie Mannor and Ohad Shamir. From bandits to experts: On the value of side-observations. In John Shawe-Taylor, Richard S. Zemel, Peter L. Bartlett, Fernando C. N. Pereira, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24: 25th Annual Conference on Neural Information Processing Systems 2011. Proceedings of a meeting held 12-14 December 2011, Granada, Spain*, pages 684–692, 2011.

[13] Gergely Neu. Explore no more: improved high-probability regret bounds for non-stochastic bandits. *CoRR*, abs/1506.03271, 2015.