Exploration and Fairness in Infinite Armed Bandit Problems

Akshay Channesh and **Venkateshwar Ragavan**
University of Illinois Chicago
{achann4,vragv2}@uic.edu

Abstract

1 Introduction

Patil et al. (2021)

Wang et al. (2008) study the setting of the infinite armed bandit

2 Related Work and Preliminary Concepts

2.1 Infinite Multi-Armed-Bandit

The infinite multi armed bandit problem consists of an infinite number of arms each with a quality parameter μ_i and every time they are pulled by an agent, they give a reward of 0 or 1 which is sampled from a Bernoulli distribution with the quality parameter as the mean. The agent that pulls the arm does not know the quality parameter of the arms before hand.

Every agent has a fixed number of rounds to play. At every round the agent is allowed to pull an arm that it has already pulled before or a new arm from the set of infinite arms. As each agent goes on pulling the arms, it goes on collecting rewards while simultaneously learning the qualities of each of the arms it has pulled.

2.2 Regret

Regret (cumulative regret) is defined as the expected sum of all the differences between the actual reward and the best reward that was possible. In expectation this is the same as the difference between the quality parameter of the best arm and the arm pulled at any given round.

$R(T) = E[\sum_{t=1}^{T} \mu_{i_t^*} - \mu_{i_t}]$

In the infinite arm setting there are two kinds of regret that can be defined, one with respect to the best arm that is within the exploration set at any given round, and the other with respect best arm that could have been hypothetically pulled whether or not it is in the exploration set. Thus we define two notions of regret.

Definition 1 (Seen Regret) *Seen Regret is the regret with respect to the best arm within the exploration set of the agent.*

Definition 2 (Unseen Regret) *Unseen Regret is the regret with respect to any best arm whether or not it is within the exploration set of the agent.*

It has been shown that both UCB and Thompson Sampling achieve a regret that is logarithmic in the number of rounds. One can look at Auer (2003) or Chatterjee et al. (2017) for more theoretical results.

2.3 Exploration Methods

There are two well known exploration methods and various variations of them. The first one being Upper Confidence Bound and the other being Thompson Sampling. One is a deterministic approach while the latter is a randomized algorithm.

2.3.1 UCB

Auer (2003) first introduced the concept of using confidence bounds within the setting of Multi Armed Bandits. We have used the same in our agents. Here the agent maintains a count of how many times an arm has been pulled (n_i) and how many times the arm has given a reward (s_i). Then when the agent has to chose an arm to pull it picks the arm i at round t by maximizing the following expression. $i_t = \arg\max(\frac{s_i}{n_i} + \sqrt{\frac{8 \log t}{n_i}})$

2.3.2 Thompson Sampling

Thompson Sampling is a technique where an agent maintains a Beta distribution over each arm and takes samples from those distributions $\theta_i \sim Beta(s_i+1, n_i-s_i+1)$. It then picks the arm with the best sample at each round $i_t \in \arg\max \theta_i$.

Kaufmann et al. (2012) showed that Thompson Sampling asymptotically achieves the same regret as UCB. However other empirical work such as

Chatterjee et al. (2017) and Chapelle and Li (2011) have shown empirically that it is performs much better than UCB in many cases.

2.4 Fairness

Patil et al. (2021) have introduced the notion of α-Fairness in the context of Multi-Armed-Bandits wherein every known arm is guaranteed to be picked a minimum number of times (r_i). And any deviations from this fairness guarantee happen less than α percent of the times. They introduce a Fair Learn meta algorithm that adds fairness constraint to any known exploration algorithm. At each round it first defines a set of arms which haven't yet been picked as many times as guaranteed. $A_t = i|r_i \cdot (t-1) - N_i \geq \alpha$ It then picks the arm that has been treated the most unfairly so far $i_t = \arg\max(r_i \cdot (t-1) - N_i)$. If all the known arms have been treated fairly so far then it uses the exploration algorithm to pick an arm.

3 Proposed Method

3.1 Curiosity

Pathak et al. (2017) introduce curiosity driven exploration in the context of Q-Learning whereby they incentivize the agent to explore more by associating a small utility with every new state that the agent explores. They employ this method in the case of agents working through sparse reward environments.

3.2 Surplus

4 Experiments and Discussion

We evaluate eight agents in three environments. We record the regret accumulated by each agent as it runs and graph it. We present the regret vs. time graphs and show the relative ordering of each of the agents.

Among the eight agents the first two are trivial ones whose regret increases linearly with time. Every other agent is more intelligent and achieves sub-linear regret. We plot the regret vs. time graph of all agents except the first two since they are very trivial.

We run each agent for 20k rounds. Each environment takes about ~10 minutes to run the simulation for all eight agents.

4.1 The Agents

The eight agents that we focus on are as follows.

1. **Random Agent** picks an unseen arm half the times and a random seen arm otherwise.

2. **Always New Arm Agent** always picks an unseen arm.

3. **Naive UCB Agent** picks a new unseen arm with $p = 1\%$ and every other time it picks among the known arms using UCB.

4. **Naive Thompson Sampling Agent** picks a new unseen arm with $p = 1\%$ and every other time it picks among the known arms using Thompson Sampling.

5. **Fair UCB Agent** picks a new unseen arm with $p = 1\%$ and every other time it picks among the known arms using UCB while also ensuring α-Fairness among known arms.

6. **Fair TS Agent** picks a new unseen arm with $p = 1\%$ and every other time it picks among the known arms using Thompson Sampling while also ensuring α-Fairness among known arms.

7. **Surplus Curiosity UCB Agent** picks a new unseen arm with a probability based on the surplus, and every other time it picks among known arms using UCB.

8. **Surplus Curiosity TS Agent** picks a new unseen arm with a probability based on the surplus, and every other time it picks among known arms using Thompson Sampling.

4.2 Environments

The three environments that we evaluate the agents in are as follows.

1. **Uniform Environment** is one in which every new arm has a quality μ which is sampled from a uniform distribution between 0 and 1.

2. **Increasingly Better Environment** is one in which arms with bad quality are available in the begining and progressively better and better ones become available. In this environment every new arm has a quality μ which is sampled from a uniform distribution between 0 and T, where T starts from 0 and increases slowly till 1 by the time we reach the last round.

3. **Progressively Worse Environment** is the reverse of the previous environment. We start with good arms and later sampled arms go on getting worse and worse. Here T starts from 1 and decreases slowly till 0 by the time we reach the last round.

4.3 Results

4.3.1 Uniform Environment

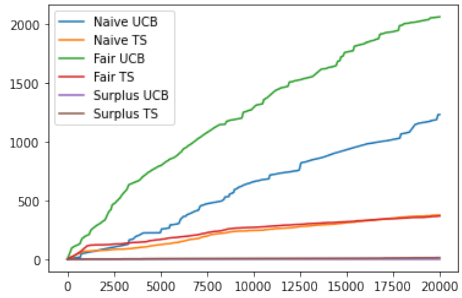

Figure 1: Uniform Environment. Linear-Linear Graph

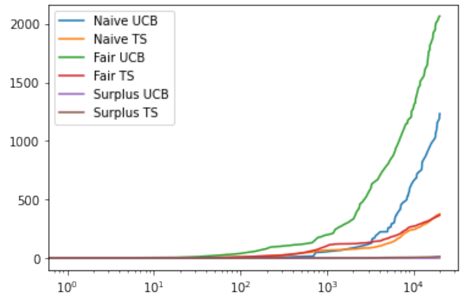

Figure 2: Uniform Environment. Log-Linear Graph

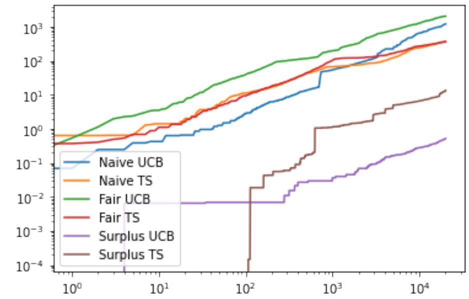

Figure 3: Uniform Environment. Log-Log Graph

Figures 1 to 3 show the same data associated with the Uniform environment in three different scales. Only the relative ordering of each of the agents is important for us and not the precise numerical value of the regret associated with each arm. Thus we show the data in three different scales to highlight the ordering. The Y axis is the regret and the X axis is the number of rounds.

4.3.2 Increasingly Better Environment

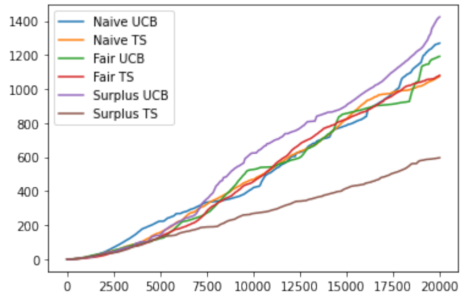

Figure 4: Increasingly Better Environment. Linear-Linear Graph

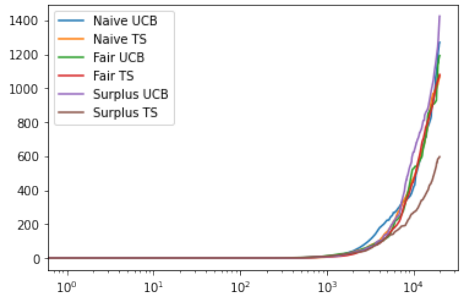

Figure 5: Increasingly Better Environment. Log-Linear Graph

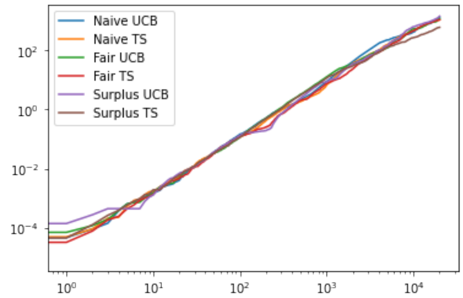

Figure 6: Increasingly Better Environment. Log-Log Graph

Similarly Figures 4 to 6 show regret vs. time graphs associated with the Increasingly Better Environment.

4.3.3 Progressively Worse Environment

Figures 7 to 9 show the regret vs. time graphs in the Progressively Worse Environment.

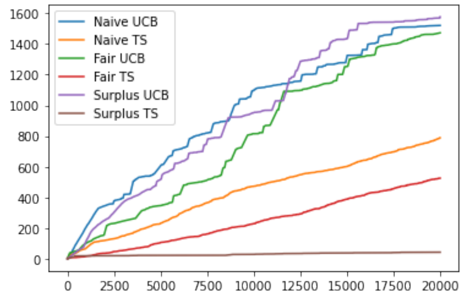

Figure 7: Progressively Worse Environment. Linear-Linear Graph

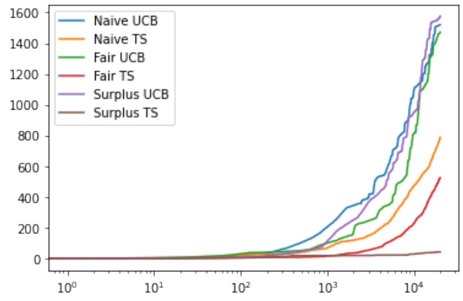

Figure 8: Progressively Worse Environment. Log-Linear Graph

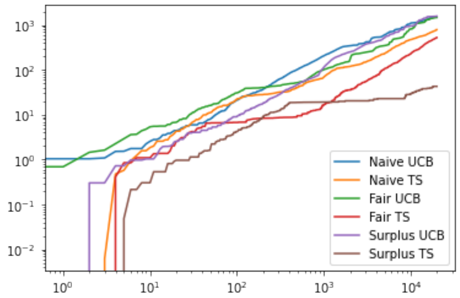

Figure 9: Progressively Worse Environment. Log-Log Graph

4.4 Discussion

We observe the following trends from these graphs.

1. Agents using Thompson Sampling perform better than those using UCB. Chatterjee et al. (2017) make the claim that Thompson Sampling adapts better to the environment and performs better. We observe this in our experiments as well.

2. There is a cost to ensuring fairness. Algorithms that incorporate fairness fare worse than their naive counterparts, although not by much.

3. Algorithms using the surplus weighted curiosity perform better than those that do not. Of these the Surplus Curiosity Thompson Sampling agent performs the best.

5 Conclusion

References

Peter Auer. 2003. Using confidence bounds for exploitation-exploration trade-offs. *J. Mach. Learn. Res.*, 3(null):397–422, mar.

Olivier Chapelle and Lihong Li. 2011. An empirical evaluation of thompson sampling. In *Proceedings of the 24th International Conference on Neural Information Processing Systems*, NIPS'11, page 2249–2257, Red Hook, NY, USA. Curran Associates Inc.

Aritra Chatterjee, Ganesh Ghalme, Shweta Jain, Rohit Vaish, and Y. Narahari. 2017. Analysis of thompson sampling for stochastic sleeping bandits. In *UAI*.

Emilie Kaufmann, Nathaniel Korda, and Rémi Munos. 2012. Thompson sampling: An asymptotically optimal finite-time analysis. In Nader H. Bshouty, Gilles Stoltz, Nicolas Vayatis, and Thomas Zeugmann, editors, *Algorithmic Learning Theory*, pages 199–213, Berlin, Heidelberg. Springer Berlin Heidelberg.

Deepak Pathak, Pulkit Agrawal, Alexei A. Efros, and Trevor Darrell. 2017. Curiosity-driven exploration by self-supervised prediction. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 2778–2787. JMLR.org.

Vishakha Patil, Ganesh Ghalme, Vineet Nair, and Y. Narahari. 2021. Achieving fairness in the stochastic multi-armed bandit problem. *Journal of Machine Learning Research*, 22(174):1–31.

Yizao Wang, Jean-yves Audibert, and Rémi Munos. 2008. Algorithms for infinitely many-armed bandits. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Advances in Neural Information Processing Systems*, volume 21. Curran Associates, Inc.