

Balancing Exploration and Exploitation: Insights from ETC, UCB, and Thompson Sampling

Xinyu Cao *

College of Business, Missouri State University, Missouri, USA

* Corresponding Author Email: xc43s@missouristate.edu

Abstract. The multi-armed bandit problem is a core issue that cannot be ignored in online decision-making. It is widely applied in areas such as advertising recommendation, clinical medical experiments, and financial investment optimization. This paper examines three classic algorithms for multi-armed slot machines: Explore-Then-Commit (ETC), Upper Confidence Bound (UCB), and Thompson Sampling (TS). First, the basic principles and operation mechanisms of each algorithm will be introduced, and the related cumulative regrets will be analyzed. By simulating different environments for different algorithms, the stability related to these algorithms can be verified. This further validates the advantages of the ETC, UCB and TS algorithms in the exploration-exploitation process. The results show that the ETC algorithm is simple and easy to implement, the UCB algorithm has excellent theoretical performance and does not require prior information, and the TS algorithm is more efficient in exploration, making it suitable for dynamic environments. This article provides theoretical support and practical examples for understanding and applying the multi-armed slot machine algorithm.

Keywords: Exploration-exploitation, Algorithm environment, Practical application.

1. Introduction

The multi-arm slot machine problem originated from statistical decision theory. Its name is derived from a classic metaphor: In a casino, there is a row of slot machines, each with a different and unknown winning probability. The player decides which one to pull to maximize long-term earnings. The multi-armed slot machine problem was first proposed by Herbert Robbins, aiming to study exploration and exploitation. The most important aspect of the multi-armed slot machine problem is that the number of draws is limited, and it is impossible to explore the reward distribution of all slot machines. Therefore, we need to make a trade-off between exploration and exploitation. More exploration can lead to better subsequent actions, but it also incurs certain waste. The multi-armed slot machine problem is a classic issue of balancing the relationship between exploration and exploitation [1]. From the theoretical establishment stage in the 1950s until the UCB (Upper Confidence Bound) algorithm was proposed in the 1990s, Bayesian methods such as the commonly used TS (Thompson Sampling) algorithm have emerged since the 2010s. At the same time, more complex scenarios have been extended and applied, and combined with deep learning and reinforcement learning.

Although significant progress has been made in the research on multi-arm slot machines in recent years, as the application has continued to deepen, some limitations of the algorithms and shortcomings in practical applications have been revealed. For instance, the theoretical assumptions are overly idealized. In most MAB models, the reward distribution is fixed, independent, and can be observed immediately without any delay. In real life, the distribution of rewards may change over time, for instance, due to changes in users' interests. There is a correlation between different arms, for instance, the contents of the two advertisements are similar. Regarding the limitations of the algorithms, for instance, conventional algorithms such as UCB and TS perform well in small-scale independent distributions, but their performance deteriorates in large-scale or complex environments.

2. The Explore-Then-Commit Algorithm

The ECT algorithm is a classic method for solving the multi-armed bandit problem and is one of the earliest approaches to exploring and utilizing solutions. The core of the problem lies in how to balance exploration and exploitation in an uncertain environment to maximize long-term benefits. Before selecting the optimal arm, this algorithm conducts m (exploration times for each arm) $\times k$ (the number of arms) rounds of exploration. The formula is as follows:

$$A_t = \arg \max_i \mu^i(mk) \text{ if } t > mk, \quad (1)$$

The larger the “ m ” is, the more exploration is conducted and the more accurate the decision becomes [2]. This can increase the probability of selecting the optimal arm. However, as the utilization period becomes shorter, the overall profit will decrease. The smaller the “ m ” is, the fewer exploration attempts there are, and one can enter the exploitation stage earlier, obtaining returns more quickly. However, with insufficient data, the risk of choosing the wrong arm increases.

One of the major advantages of the ETC algorithm is its simplicity and ease of use. It can be easily operated by people and is suitable for completing simple algorithm tasks. In a static structure, a clear upper bound for the regret can also be obtained. However, the limitation of this algorithm is that it lacks dynamic adaptability and is not suitable for complex environments.

3. Epsilon-Greedy Algorithm

The ϵ -Greedy strategy maintains a certain exploration probability throughout the process, and performs stably in many scenarios, thereby reducing the impact of excessive exploration on the utilization situation [2]. At each time t , a random exploration arm is selected with a probability of ϵ , or the arm with the highest current mean is chosen with a probability of $1 - \epsilon$.

The drawback of ϵ -Greedy is that the exploration is a random choice, and it does not prioritize the exploration of the arms with higher uncertainty. In the long run, it cannot be compared with more advanced algorithms.

4. Upper Confidence Bound Algorithm

In the early algorithms, exploration was random and episodic, and there was no intelligent way to determine which arms might be better [3]. The core idea of the UCB algorithm is: at each step, the arm with the largest upper confidence bound is selected to actively explore the arms with high uncertainty, thereby achieving an adaptive balance between exploration and exploitation.

$$UCB_i(t) = \hat{\mu}_i(t) + \sqrt{\frac{2 \log t}{n_i(t)}} \quad (2)$$

This formula calculates the UCB value for each arm at time t . The UCB value of each arm represents the possible maximum reward. The arm with the highest current

UCB value is selected, which corresponds to the optimal reward [4]. By studying the random scenario Monte Carlo problem, a new algorithm was proposed. Utilizing the representation ability of deep neural networks, the confidence upper bound of the reward was constructed based on the random features of the neural network.

The UCB algorithm allocates exploration resources more rationally and efficiently by averaging the rewards and adding the upper confidence bound [5]. It has a logarithmic regret accumulation rate. It is suitable for online systems with limited resources, such as advertising systems. Using the UCB-RS method to enhance the confidence upper bound through the recommendation system, aiming to solve the product recommendation problem in online advertising [6].

However, the drawback of UCB is that its overly strong performance in the initial stage, as well as the overly wide or narrow confidence intervals, can all affect the selection process. The distribution of rewards is fixed. If the environment changes, there may be errors in the data.

5. KL-UCB Algorithm

KL-UCB is an improvement of the UCB algorithm. Its core idea is to use Kullback-Leibler to construct a confidence upper bound, thereby enabling more precise exploration [7].

$$\text{KL-UCB}_i(t) = \sup\{q \in [\hat{\mu}_i(t), 1] : n_i(t) \cdot D_{\text{KL}}(\hat{\mu}_i(t) || q) \leq \log t + c \log \log t\}, \quad (3)$$

In terms of the exploration steps, KL-UCB is consistent with UCB, but compared to the UCB algorithm, its upper bound is more accurate, approaching the theoretical optimum, and performs more stably when the differences between different reward expectations are small [8]. However, the calculation of KL-UCB is more complex and is not easily extendable to the LinUCB and Contextual Bandit scenarios.

6. Thompson Sampling Algorithm

The TS algorithm was first introduced in clinical trials to minimize errors and the risks to patients caused by treatment [9]. Its purpose was to enable dynamic selection of more effective methods when different therapies were being tested. Thompson proposed a strategy based on Bayesian inference, maintaining a probability distribution of success rates for each treatment option, and randomly sampling to select the optimal one.

The core idea of the TS algorithm is not like UCB, which always selects the arm with the highest average reward at present, but rather makes a probabilistic choice of the optimal arm. In each round, the choice of which arm to select is determined through sampling.

$$\theta_i \sim \text{Beta}(\alpha_i, \beta_i), \quad (4)$$

$$A_t = \arg \max_i \theta_i. \quad (5)$$

Bayesian modeling is flexible and can select the appropriate posterior distribution based on the type of rewards [9]. This is one of the advantages of the TS algorithm. It is relatively easy to implement, especially in the Bernoulli scenario, and each update of the sampling process is very simple and efficient [10]. However, the limitations of the TS algorithm lie in its overly complex theory, which lacked theoretical support in the early stages. Only in recent years have there been significant breakthroughs.

7. Contextual Thompson Sampling (CTS)

CTS is an extension of the classic TS algorithm, used for the context multi-armed bandit problem [11]. In traditional multi-armed bandit problems, the rewards of each arm are fixed and unknown. CTS is more in line with practical applications such as recommendation systems and advertising placements.

$$E[r_{t,a} | x_{t,a}] = x_{t,a}^{\top} \theta_a, \quad (6)$$

The advantage of the CTS algorithm lies in its ability to utilize context and fully leverage user characteristics and other information, resulting in more accurate decisions. However, high-dimensional features are difficult to calculate, the posterior update process has high computational cost, and the modeling complexity is also high [12].

8. Conclusion

This article focuses on the slot machine problem and compares three classic algorithms: ETC, UCB, and TS. Through the analysis of algorithm principles, theoretical analysis, and comparison of advantages and limitations, it can be seen that ETC is simple and easy to implement, but lacks flexibility. UCB achieves adaptive exploration through the use of confidence upper bounds, resulting

in more stable theoretical performance. TS achieved more natural exploration and exploitation balance through Bayesian sampling.

The ETC algorithm is usually applied in clinical experiments and A/B tests, while the UCB algorithm is more commonly used in online advertising and recommendation systems. Google once employed the UCB algorithm to select the advertisement with the highest click-through rate, ensuring the maximization of click-through rate while also allowing for the exploration of new advertisement materials. The TS algorithm is also used in financial investment and industrial control. Among them, workshop scheduling is one of the major challenges in application.

References

- [1] Robbins, Herbert. Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society*, 1952, 58(5): 527–535.
- [2] Shukla, Avi L. Multi-armed bandit study of exploration vs. exploitation. 2022.
- [3] Auer, Peter; Cesa-Bianchi, Nicolò; Fischer, Paul. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 2002, 47(2–3): 235–256.
- [4] Gonçalves, Richard A.; Almeida, Carolina P.; Pozo, Aurora. Upper confidence bound (UCB) algorithms for adaptive operator selection in MOEA/D. In: *International Conference on Evolutionary Multi-Criterion Optimization*. Cham: Springer International Publishing, 2015.
- [5] Nguyen-Thanh, Nhan, et al. Recommendation system-based upper confidence bound for online advertising. *arXivpreprint arXiv:1909.04190*, 2019.
- [6] Jourdan, Marc; Degenne, Rémy. Non-asymptotic analysis of a UCB-based top two algorithm. *Advances in Neural Information Processing Systems*, 2023, 36: 68980–69020.
- [7] Garivier, Aurélien; Cappé, Olivier. The KL-UCB algorithm for bounded stochastic bandits and beyond. In: *Proceedings of the 24th Annual Conference on Learning Theory*. JMLR Workshop and Conference Proceedings, 2011.
- [8] Cappé, Olivier; Garivier, Aurélien; Maillard, Olivier; Munos, Rémi; Stoltz, Gilles. Kullback-Leibler upper confidence bounds for optimal sequential allocation. *The Annals of Statistics*, 2013, 41(3): 1516–1541. Project Euclid
- [9] Thompson, William R. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 1933, 25(3/4): 285–294. Oxford AcademicSciSpace.
- [10] Agrawal, Shipra; Goyal, Navin. Analysis of Thompson Sampling for the multi-armed bandit problem. In: *Proceedings of the 25th Annual Conference on Learning Theory (COLT)*, JMLR Proceedings, 2012, 23: 39.1–39.26.
- [11] Agrawal, Shipra; Goyal, Navin. Thompson Sampling for contextual bandits with linear payoffs. In: *International Conference on Machine Learning (ICML)*, 2013.
- [12] Nasiri, Mohammad Mahdi; Kianfar, Farhad. A GES/TS algorithm for the job shop scheduling. *Computers & Industrial Engineering*, 2012, 62(4): 946–952.