Optimizing the ε Parameter in ε-Greedy Strategy for Multi-Armed Bandits

Haotong Jiang

School of Computing and Artificial Intelligence, Southwest Jiaotong University, Sichuan, 611756, China

Abstract. With the widespread adoption of the internet, online advertising has grown exponentially. To enhance ad recommendation efficiency, various Multi-Armed Bandit (MAB) algorithms have been deployed. Among these, the Thompson ε-Greedy algorithm integrates the ε-Greedy policy with Thompson Sampling. To optimize the algorithm, specifically, to reduce cumulative regret and improve arm selection accuracy, this paper analyzes the parameter ε in the ε-Greedy framework. This paper argues that fixing ε wastes environmental information learned over time. As the number of rounds increases, environmental understanding deepens, and ε should decay with both the rounds and the selection count of the current best arm T_(t,arm_max), since a higher selection count implies greater confidence in its optimality. Two primary decay modes are considered: linear and nonlinear decay. The study analyzes both modes and optimizes their parameters using genetic algorithms. Results demonstrate that after introducing parameter T_(t,arm_max), nonlinear ε-decay achieves lower cumulative regret under optimal parameter settings, whereas linear decay shows no such improvement.

1 Introduction

In contemporary society, the internet has become highly advanced, resulting in a growing number of individuals seeking required information online. Consequently, companies necessitate an online advertising recommendation system capable of precisely targeting customer preferences to pursue higher click-through rates (CTR).In order to optimize ad placements and increase user engagement, Multi-armed Bandit algorithms are essential [1]. Businesses frequently have to choose how to divide their cash between various marketing strategies or ads in order to boost user engagement, clicks, or conversions. Depending on observed performance, multi-armed Bandit algorithms assist advertisers in dynamically allocating resources to different alternatives (ads) [2]. In a Multi-armed Bandit problem, players can select arms or actions as they are called from now on N times, where N is the control horizon, and players receive a one-step income while selecting an arm or an action [3]. The players' goal is to maximize the mathematical expectation of total income or minimize the cumulative regret, which is the gap between the optimal return and the return after the player's choice. But achieving this goal requires striking a careful balance between

jht20031018@my.swjtu.edu.cn

exploiting known possibilities for maximum gain and investigating unknown ones to learn about their benefits [4]. This is the fundamental conundrum in decision-making: The trade-off between exploration and exploitation [4]. Some classic algorithms are used to solve this problem, such as the epsilon-Greedy algorithm and the Thompson Sampling (TS) algorithm. In the epsilon-Greedy algorithm, one drawback, though, is that because ε is static, it cannot exhibit adaptive behavior; as it gains more environmental knowledge, the algorithm does not inherently become more exploitative [5].To address the shortcomings of the traditional epsilon-Greedy algorithm, decreasing ε with the increase in the number of selections is a good method because it is essential for successfully striking a balance between exploration and exploitation, particularly in settings with a wide variety of possibilities or states [6]. This paper argues that the more times an arm is considered the best, the more credible the message is, so ε should also decrease based on the increase in the selection counts of the current best action or arm. This paper will compare three strategies for handling ε: fixed ε values, linearly decreasing ε values, and non-linearly decreasing ε values in the ε-Greedy algorithm. In addition, some of the parameters used when processing ε values will also be studied to the extent to which they affect the algorithm.

2 Related Work

2.1 Thompson Sampling Algorithm

The Thompson sampling is a more principled technique that will produce more balanced results in null conditions and uses a Bayesian approach, using a probability distribution to represent the reward distribution for each arm [2, 7]. Commonly used probability distributions include the Beta distribution, the Gaussian distribution, etc. First, through the Bayesian update mechanism, the posterior distribution is updated based on historical observations. Then, according to the probability distribution of each arm, the arm from which the sample is sampled and the arm that receives the maximum value are obtained. Next, select the arm and observe the reward, and adjust the parameters of the probability distribution of the arm according to the reward.

2.2 ε-Greedy Strategy

The ε-greedy strategy does not involve complicated parameter tuning or optimization procedures, and it is easy to implement and computationally efficient [8]. In the ε-greedy strategy, when an arm is chosen, there is an ε probability of randomly picking an arm each time, and there is a 1-ε probability of choosing the arm with the largest current average reward. In the vanilla version of the ε-greedy algorithm, ϵ is constant, but this is not necessary [9]. Although it is not required, ϵ is constant in the vanilla form of the ϵ-greedy algorithm. Making ε iteration dependent (linearly or exponentially decreasing) makes sense [9]. In other words, $\varepsilon(t) = \varepsilon_0 - rt$ or $\varepsilon(t) = \varepsilon_0 e^{-rt}$,where ε_0 is the initial value, and r is the decay rate.

2.3 Thompson ε-Greedy Algorithm

This algorithm combines the ε-Greedy algorithm and the Thompson Sampling algorithm, and adopts the mechanism of the Thompson sampling algorithm when obtaining rewards. On the one hand, the rewards are extracted from the probability distribution corresponding to each arm. On the other hand, different parameters apply depending on the distribution, such as mean and variance for a Gaussian distribution or α and β for a beta distribution [10].

The idea of the ε-Greedy algorithm is used in the selection of arms; in other words, there is an ε probability that the arm with the largest current average will be selected. This algorithm is more computationally efficient than TS because it performs exploration with a probability of ϵ, which reduces exploration, particularly when sampling from the posterior distribution is computationally costly [11].

3 Methodology

3.1 Data Processing

The methods in this article will be tested on a dataset related to advertisement recommendations, featuring ten ads and ten thousand users. A user's click on an ad is recorded as 1, while no click is recorded as 0. During testing, the ten ads are treated as ten arms, and the recorded 1 or 0 serves as the reward. Due to the degree of uncertainty involved in Thompson sampling, this paper repeats the experiment 50 times for each set of data, averaging the above results to obtain the final data.

3.2 Two Decay Modes of ε

In the epsilon-greedy strategy, the value of ε will greatly affect the efficiency and results of the algorithm. In order to find the best arm efficiently, the value of ε should be adjusted based on the information learned from the environment. As players select more arms, they will become more aware of the distribution of rewards in each arm of the bandit, and the uncertainty of the next choice should be reduced, meaning the value of ε should be reduced. As ε decreases, it can decrease linearly or non-linearly during the process of reduction. When having chosen arms t rounds, the formula for the two different ways is shown below. Formula (1) represents the linear decay of ε, and formula (2) represents the nonlinear decay of ε.

$$\varepsilon(t) = \varepsilon_0 e^{-rt} (r > 0) \tag{1}$$

Where ε_0 is the initial value when ε decreases nonlinearly, and r is an unknown parameter.

$$\varepsilon(t) = \varepsilon_1 (1 - \frac{t}{n}) \tag{2}$$

Where ε_1 is the initial value when ε decreases linearly, n is the total rounds.

When t rounds selections are made, the player can also know how many times (T_{t,arm_max}) the current best arm has been selected before. As T_{t,arm_max} increases, the more likely it is that the arm will be the best arm, and the probability of a random arm selection next time should be reduced. Therefore, the above formula can be added with the parameter T_{t,arm_max} to further improve the ε. The improved formula is shown below. Formula (3) represents the new linear decay of ε and Formula (4) represents the new nonlinear decay of ε.

$$\varepsilon(t, T_{t,arm\_max}) = \varepsilon_0 \frac{e^{-rt}}{(T_{t,arm\_max}+1)^{\alpha}} \tag{3}$$

where ε_0 is the initial value when ε decreases nonlinearly, α and r are unknown parameters.

$$\varepsilon\left(t, T_{t,arm\_max}\right) = \varepsilon_1(1 - \beta\frac{t}{n} - (1-\beta)\frac{T_{t,arm\_max}}{n}) \quad (4)$$

where ε_1 is the initial value when ε decreases linearly, β is an unknown parameter, and n is the total rounds.

3.3 The Best Parameters for ε

In this paper, the Thompson ε-Greedy algorithm is used to test the influence of ε before and after improvement on the results, as well as the effect of different values of the parameters in the formulas. When the ε are linearly decreasing, they contain unknown parameters such as β and ε_1; while when the ε are nonlinearly decreasing, it includes unknown parameters such as α, r and ε_0. To investigate the influence of these parameters on the results, this paper adopted the control variable method, which involves altering the value of one parameter while keeping the other parameters constant. Eventually, this paper uses a genetic algorithm to find the optimal solution for these parameters and compares it with the results before optimization.

3.4 Evaluation Metrics

In MAB questions, the evaluation metric is usually the cumulative regret value. In the same round, the smaller the cumulative regret value, the better the performance of the related algorithm. Because the experiment was repeated for each set of data during the test, the average cumulative regret value was used as the evaluation index in this paper. The Formula (5) is:

$$R(n) = \frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{n}(\mu^* - \mu_{t,i}) \quad (5)$$

where T is the number of replicates of the experiment, n is the total rounds of each experiment, $\mu^*$is the expected value of the reward for the best arm, $\mu_{t,i}$ is the expected value of the reward for the chosen arm in the round i of the experiment t, R is the average cumulative regret.

At the end of the algorithm, the probability that the chosen arm is the best arm is also one of the evaluation indicators. That is:

$$\eta = \frac{t_{true}}{T} \quad (6)$$

where η is the accuracy, T is the number of replicates of the experiment, t_{true} is the number of times the chosen arm is the best arm.

4 Results and Analysis

4.1 Impact of Parameters on Linear ε-Decay

This paper compares and analyzes the performance of the algorithm when parameter β is set to 0.2, 0.5 and 0.8, and parameter ε_1 is independently assigned values of 0.2, 0.5 and 0.8. By

applying the evaluation metric defined in Formula (6) to these parameter combinations, the results demonstrate that the algorithm achieves near-perfect accuracy (consistently 100% in most cases, with occasional instances of 98%) in selecting the optimal arm across these configurations. So when ε decreases linearly, the accuracy remains nearly 100% as long as ε_1 and β are within reasonable ranges, exhibiting little correlation with parameter values.

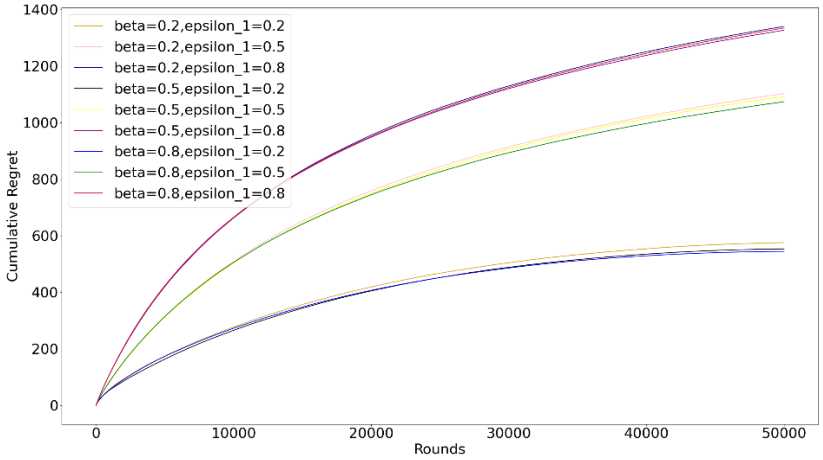

Figure 1 Curves of regret variation with rounds for different ε_1 values and β values (Picture credit: Original)

Figure 1 shows the cumulative regret curves for different values of ε_1 when $\beta = 0.2, \beta = 0.5, \beta = 0.8$, respectively. From the figures, it can be observed that for a fixed β, the cumulative regret increases as ε_1 grows. Additionally, the five curves exhibit minimal differences in each figure, suggesting that β has a limited impact on cumulative regret. However, when ε_1 remains constant but β varies, the cumulative regret shows little divergence. This indicates that the choice of the initial ε value hardly affects the algorithm's performance.

4.2 Impact of Parameters on Nonlinear ε-Decay

This paper compares and analyzes the performance of the algorithm when parameter r is set to 0.001, 0.0005, 0.0001 and 0.005, and parameter α is independently assigned values of 0.05, 0.1, 0.5, 1 and 2. When evaluating the algorithm using the accuracy of selecting the optimal arm as the evaluation metric (as defined in Formula (6)), it is observed that most parameter configurations demonstrate suboptimal performance, with only a limited subset of parameter values enabling the algorithm to approach near-perfect accuracy (98–100%). Meanwhile, the accuracy exhibits significant variations as r and α change, indicating a strong correlation between the accuracy and their values.

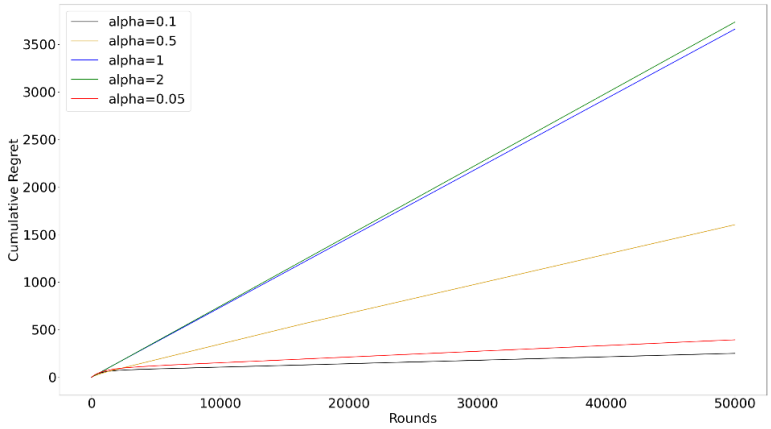

Figure 2 Curves of regret variation with rounds for different α values ($\varepsilon_0 = 0.67$,r $= 0.001$) (Picture credit: Original)

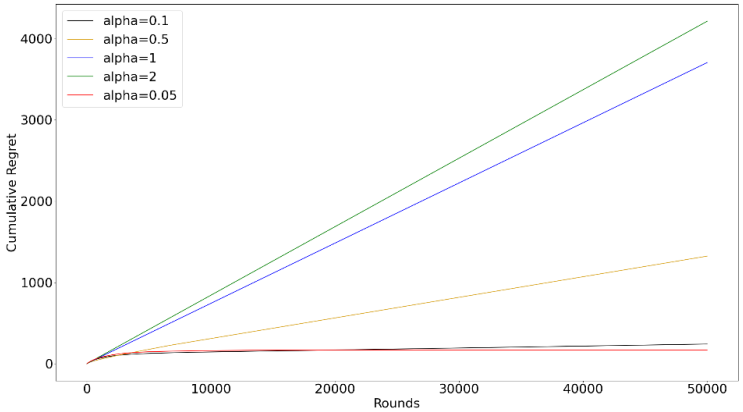

Figure 3 Curves of regret variation with rounds for different α values ($\varepsilon_0 = 0.67$,r $= 0.0005$) (Picture credit: Original)

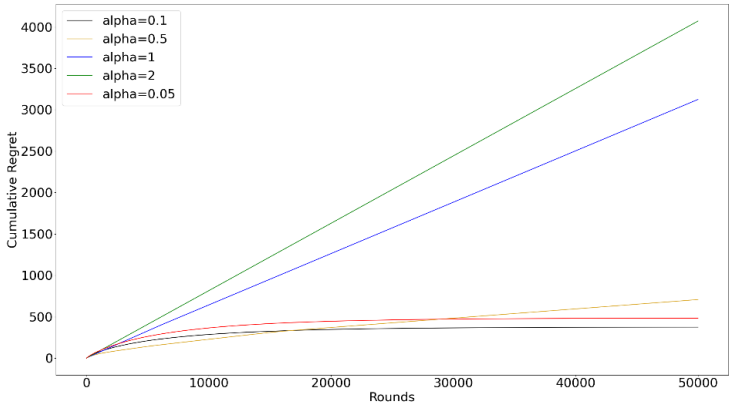

Figure 4 Curves of regret variation with rounds for different α values ($\varepsilon_0 = 0.67$,r $= 0.0001$) (Picture credit: Original)

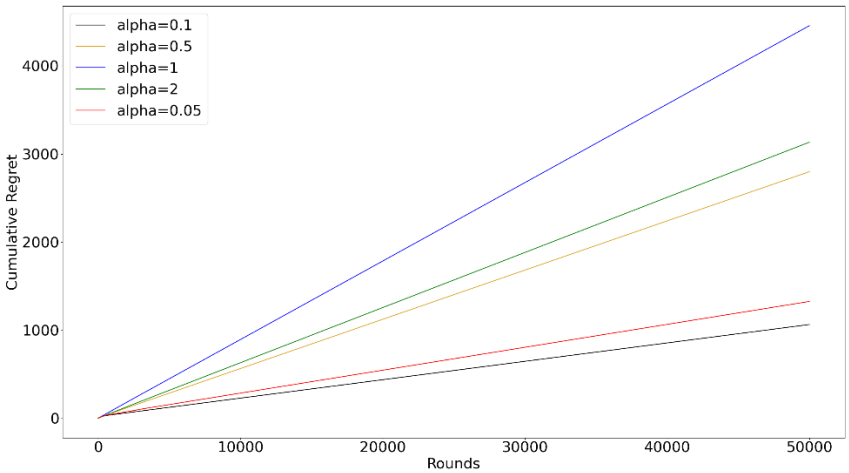

Figure 5 Curves of regret variation with rounds for different α values ($\varepsilon_0 = 0.67$,r $= 0.005$) (Picture credit: Original)

Figures 2, 3, 4, and 5 display the cumulative regret curves for different α when r=0.001, r=0.0005, r=0.0001, r=0.005(set ε_0 to a fixed value), respectively. If, under a specific parameter configuration, the algorithm can gradually identify the optimal arm, the per-round regret will progressively diminish, leading to a reduced slope of cumulative regret R(n) with respect to the number of rounds. However, in the four figures above, the slopes of most curves show no significant decrease, indicating that these parameter settings fail to enable the algorithm to reliably converge to the optimal arm. Therefore, it can be inferred that when ε decreases nonlinearly, the algorithm is highly sensitive to parameter selection. Only when the parameter values are sufficiently small can the algorithm identify the optimal arm with high probability.

4.3 Parameter Optimization

In the case of ε linear decreasing, β represents the weight of the decreasing part of the ε with the rounds, so β should satisfy the inequality: $0 \leq \beta \leq 1$. Regardless of whether parameter T_{t,arm_max} is considered, both Formula (2) and Formula (4) share a common parameter ε_1. To ensure a meaningful comparison between the optimized Formula (4) and Formula (2), ε_1 must be assigned reasonable values that enable the algorithm to identify the optimal arm when substituted into Formula (2). Based on the analysis of results with varying ε_1 in Section 4.1, it was found that ε_1 can plausibly take values within the range of 0.3 to 0.7. This study selects 0.67 as the value of ε_1 within this range. Meanwhile, taking the value of the accumulated regret in the last round as the output of the objective function, and then according to the inequalities above, the optimal value β can be obtained by using the genetic algorithm. It is:$\beta = 0.99$.

When ε decreases nonlinearly, this paper examines the impact of parameters on regret while fixing the initial value as a constant. To ensure that the cumulative regret approximately follows a logarithmic relationship with the number of rounds, a range for parameter values can be inferred based on the results from Figures 4, 5, 6, and 7 :$0 < r \leq 0.001$,$0 < \alpha \leq 0.1$. Similar to the process of ε linear decreasing, setting $\varepsilon_0$to 0.67 and

inequality above is taken as the upper and lower bounds of r and α, then the optimal solutions of them are found using a genetic algorithm. It is:$r = 0.00068$,$\alpha = 0.0358$.

The optimal value of β is a number very close to 1. This implies that when ε decreases linearly, setting the parameter to 1 allows the algorithm to achieve its best performance. Specifically, introducing T_{t,arm_max} cannot further optimize the scenario of linearly decreasing ε.

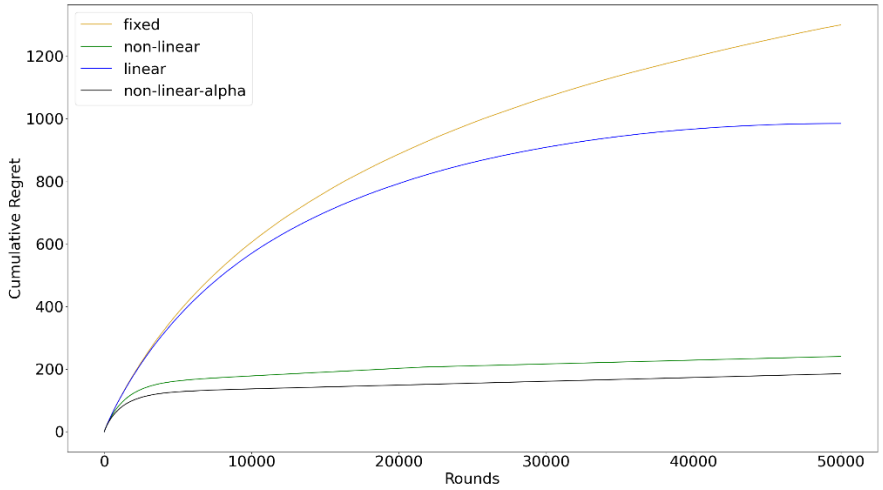

Figure 6 Compare the four cases of ε (Picture credit: Original)

In Figure 6, the cumulative regret of introducing T_{t,arm_max} is lower than not introducing T_{t,arm_max} when ε nonlinear decline. And they are also much smaller than the cumulative regret when ε decreases linearly or ε is fixed.

5 Conclusion

When ε decreases linearly and both β and ε_1 are within certain ranges, the cumulative regret increases as ε_1 increases, meanwhile, the algorithm can identify the optimal arm with high accuracy across most value ranges of these two parameters. When ε decreases nonlinearly, the algorithm struggles to identify the optimal arm effectively when α and r take larger values. Therefore, to improve the algorithm's performance, these two parameters should take a smaller value within a reasonable range. The optimization results obtained using a genetic algorithm yield a value of 0.00068 for r and a value of 0.0358 for α. In the Thompson ε-Greedy algorithm, introducing the parameter T_{t,arm_max} does not improve the algorithm's performance when ε decreases linearly. However, for scenarios where ε decreases nonlinearly, incorporating T_{t,arm_max} can lead to smaller cumulative regret, thereby optimizing the algorithm. And also, compared to algorithms where ε takes a fixed value or decreases linearly, the algorithm with nonlinearly decreasing εperforms better. After introducing T_{t,arm_max} , ε may follow various forms of nonlinear decrease. This paper, however, only explores one specific form and does not compare alternative forms, such as logarithmic or exponential decay of T_{t,arm_max}.

References

1. Cao, Q.: 'Exploration vs. Exploitation: Comparative Analysis and Practical Applications of Multi-Armed Bandit Algorithms', Proc. 1st Int. Conf. Engineering Management, Information Technology and Intelligence, 2024, pp.412–416.
2. Liu, Z.: 'Investigation of progress and application related to Multi-Armed Bandit algorithms', Proc. Int. Conf. Machine Learning and Automation, October 2023, pp.155-159.
3. Ershov, M. A., Voroshilov, A. S.: 'UCB Strategy for Gaussian and Bernoulli Multi-armed Bandits', Proc. Int. Conf. Mathematical Optimization Theory and Operations Research, September 2023, pp.67–78.
4. Han, Y.: 'Comparative Evaluation, Challenges, and Diverse Applications of Multi-Armed Bandit Algorithms', Proc. Int. Conf. Highlights in Science, Engineering and Technology, January 2024, pp.94.
5. Song, R.: 'Optimizing decision-making in uncertain environments through analysis of stochastic stationary Multi-Armed Bandit algorithms', Proc. 6th Int. Conf. Computing and Data Science, September 2024, 93-113.
6. Zhao, J.: 'Comparison of Multi-Armed Bandit Algorithms in Advertising Recommendation Systems'. Proc. CONF-MLA 2024 Workshop: Semantic Communication Based Complexity Scalable Image Transmission System for Resource Constrained Devices, November 2024, pp.62-71.
7. Umami, I., Rahmawati, L.: 'Comparing Epsilon Greedy and Thompson Sampling model for Multi-Armed Bandit algorithm on marketing dataset', J. Adv. Data Sci., 2023, 2, (2), pp.14–26.
8. Fu, L.: 'Exploring the efficacy of Multi-Armed Bandit Algorithms in dynamic decision-making', Proc. 2nd Int. Conf. Machine Learning and Automation, November 2024, pp.141-148.
9. Gangan, E.: 'Survey of multi-armed bandit algorithms applied to recommendation systems', Int. J. Open Inf. Technol., 2021, 9, (4), pp. 123–135.
10. Yu, J.: 'Thompson -Greedy Algorithm: An Improvement to the Regret of Thompson Sampling and -Greedy on Multi-Armed Bandit Problems', Proc. Int. Conf. Software Engineering and Machine Learning, August 2023, pp.507-516.
11. Jin, T., Yang, X., Xiao, X., Xu, P.: 'Thompson Sampling with Less Exploration is Fast and Optimal', Proc. 40th Int. Conf. Machine Learning, 2023, pp.15239–15261.