Multi-armed Bandits with Compensation

NeurIPS (NIPS) 2018

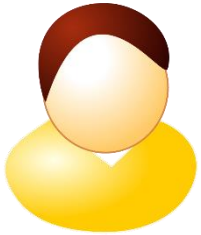
Reviewed by Tae Hyeon, osi lab

Contents

Introduction

Reviewed by Tae Hyeon, osi lab

Introduction: M.A.B.



Player

Model Description

Assume it's stochastic MAB model, then, feedbacks from a single arm follow an associated distribution, which is unknown to the player.

Purpose

maximize the sum of rewards during the game by choosing a proper arm to pull in each time slot, and the decision can depend on all available information, i.e., past chosen arms and feedbacks

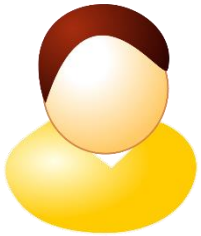
T. L. Lai and H. Robbins. Asymptotically efficient adaptive allocation rules. Advances in applied mathematics, 6(1):4–22, 1985.

Introduction: M.A.B.

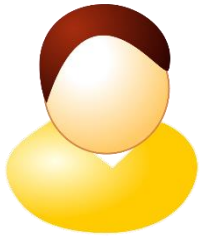
Real world Problem

Controller: Long-term
player: Short-term

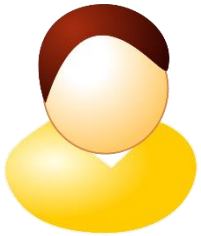
Controller



Player



Player



Player

Objective

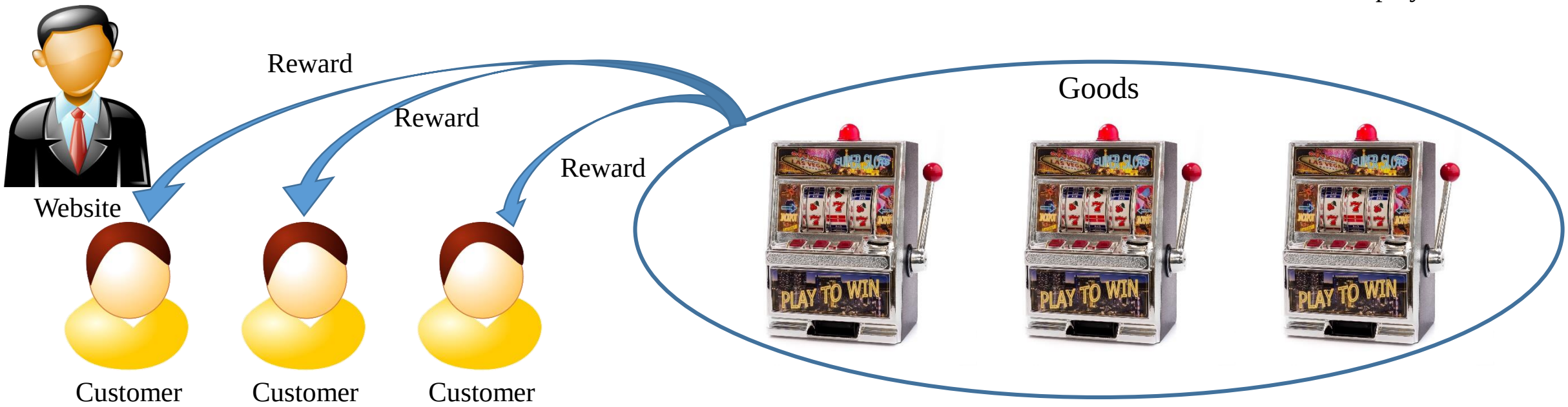
To seek an incentivizing policy, so as to minimize regret while not giving away too much compensation

In many real-world applications, arms are not pulled by the controller concerning long-term performance. Instead, actions are taken by short-term players interested in optimizing their instantaneous reward. In this case, an important means is to provide monetary compensation to players, so that they act as if they are pulling the arms on behalf of the controller, to jointly minimize regret

Introduction: Known-compensation multi-armed bandit(KCMAB)

Example: website (e-commerce)

Controller: Long-term
player: Short-term



Issue

When a consumer chooses to purchase a certain good, he receives the reward of that good. The website similarly collects the same reward as a recognition of the recommendation. To maximize the total reward during the game, in the respect of the website, he needs to devise a scheme to influence the choices of customers (=short-term consumers). To achieve this goal in practice is that the website offers customized discounts for certain goods to consumers.

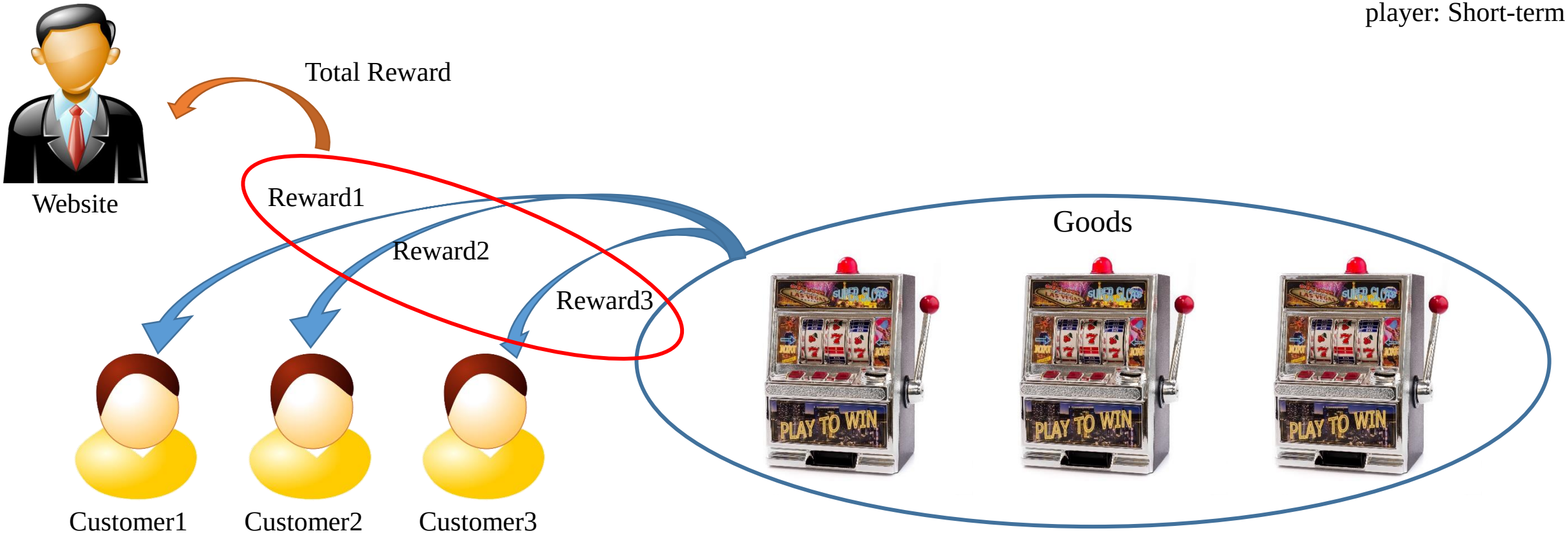
Goal

To find an optimal compensation policy to minimize his regret, while not spending too much additional payment.

Introduction: Known-compensation multi-armed bandit(KCMAB)

Example: website (e-commerce)

Controller: Long-term
player: Short-term



Introduction: Known-compensation multi-armed bandit(KCMAB)

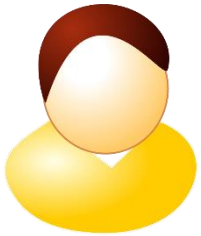
Example: website (e-commerce)

Controller: Long-term
player: Short-term

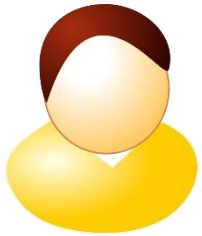
Introduction: Known-compensation multi-armed bandit(KCMAB)

Controller: Long-term
player: Short-term

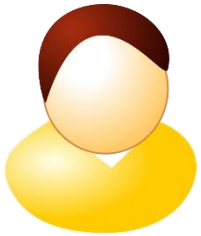
Website



Customer



Customer



Customer

Goods

Key Challenge

Trade-off between regret and compensation

‘compensation value depends on the random history’

Related Works

Reviewed by Tae Hyeon, osi lab

Related Works

(Basic) Incentivized Learning

Model contains a prior distribution for each arm's mean reward at the beginning. As times goes on, observations from each arm update the posterior distributions, and subsequent decisions are made based on posterior distributions. The objective is to optimize the total discounted rewards.

Preview

You can regard this problem as contextual bandit problem. But, key difference from contextual bandits, where the context is often an exogenous random variable and the controller focuses on identifying the best arm under given contexts, in our case, the context is given by the controller and itself influenced by player actions. Also, the controller needs to pay for obtaining a desired context.

Paper

1. Incentivized Learning
- P. Frazier, D. Kempe, J. Kleinberg, and R. Kleinberg. Incentivizing exploration. In Fifteenth ACM Conference on Economics and Computation, pages 5–22, 2014.

2. Bayesian Incentivized Learning model
- Y. Mansour, A. Slivkins, and V. Syrgkanis. Bayesian incentive-compatible bandit exploration. In Proceedings of the Sixteenth ACM Conference on Economics and Computation, pages 565–582. ACM, 2015.
- Y. Mansour, A. Slivkins, V. Syrgkanis, and Z. S. Wu. Bayesian exploration: Incentivizing exploration in bayesian games. arXiv preprint arXiv:1602.07570, 2016.

KC MAB

Reviewed by Tae Hyeon, osi lab

KC MAB – Model and Notation

Controller

N slot machines
{1, 2, … , N}

Without loss of generality, $1 \geq \mu_1 \geq \mu_2 \geq \ldots \mu_N \geq 0$, $\Delta_i = \mu_1 - \mu_i$, these are the mean of each arm. Each arm i has a reward distribution denoted by D_i with support [0,1].

After the player pulls arm a(t), the player and the controller each receive a reward X(t) from the distribution $D_{a(t)}$, denoted by $X_{a(t)}(t) \sim D_{a(t)}$, which is an independent random variable every time arm a(t) is pulled.

Paid compensation : $c(t) = c_{a(t)}(t)$ and assume that it can depend on all the previous information.

Each Player is assumed to choose an arm greedily to maximize his total expected income based on past observations.
Here income for pulling arm i equals to $\hat{\mu}_i(t) + c_i(t)$, where $\hat{\mu}_i(t) \triangleq {}^{M_i(t)}/_{N_i(t)}$ is the empirical mean of arm i, with $N_i(t) = \sum_{\tau<t} I[a(\tau) = i]$

P. Frazier, D. Kempe, J. Kleinberg, and R. Kleinberg. Incentivizing exploration. In Fifteenth ACM Conference on Economics and Computation, pages 5–22, 2014.

KC MAB – Model and Notation

Controller

N slot machines
{1, 2, … , N}

$\mathrm{Com_i(T)} = \mathrm{E}[\sum_{\tau=1}^{\mathrm{T}} \mathrm{I}[\mathrm{a}(\tau) = \mathrm{i}]\mathrm{c}(\tau)]$ to denote the expected compensation padif for arm I

$\mathrm{Com(T)} = \sum_{\mathrm{i}} \mathrm{Com_i(T)}$ the expected total compensation

Regret: $Reg(T) = T \max_i \mu_i - Rew(T) = T\mu_1 - Rew(T)$

Reg(T) has a lower bound of $\Omega(\sum_{i=2}^{N} \frac{\Delta_i \log T}{KL(D_i, D_1)})$

Objective: to minimize the compensation while keeping the regret upper bounded by $O(\sum_{i=2}^{N} \frac{\Delta_i \log T}{KL(D_i, D_1)})$

P. Frazier, D. Kempe, J. Kleinberg, and R. Kleinberg. Incentivizing exploration. In Fifteenth ACM Conference on Economics and Computation, pages 5–22, 2014.

KC MAB – Algorithm

Compensation lower bound

Fact 1. *If the long-term controller wants the short-term player to choose arm i in time slot t, then the minimum compensation he needs to pay on pulling arm i is* $c_i(t) = \max_j \hat{\mu}_j(t) - \hat{\mu}_i(t)$.

Theorem 1. *In KCMAB, if an algorithm guarantees an $o(T^\alpha)$ regret upper bound for any fixed T and any $\alpha > 0$, then there exist examples of Bernoulli Bandits, i.e., arms having reward 0 or 1 every time, such that the algorithm must pay $\Omega\left(\sum_{i=2}^{N} \frac{\Delta_i \log T}{KL(D_i, D_1)}\right)$ for compensation in these examples.*

Similar with Lai and robins assumption

Proof Sketch: Suppose an algorithm achieves an $o(T^\alpha)$ regret upper bound for any $\alpha > 0$. We know that it must pull a sub-optimal arm i for $\Omega\left(\frac{\log T}{KL(D_i, D_1)}\right)$ times almost surely [12]. Now denote $t_i(k)$ be the time slot (a random variable) that we choose arm i for the k-th time. We see that one needs to pay $\mathbb{E}[\max_j \hat{\mu}_j(t_i(k)) - \hat{\mu}_i(t_i(k))] \geq \mathbb{E}[\hat{\mu}_1(t_i(k))] - \mathbb{E}[\hat{\mu}_i(t_i(k))]$ for compensation in that time slot. By definition of $t_i(k)$ and the fact that all rewards are independent with each other, we always have $\mathbb{E}[\hat{\mu}_i(t_i(k))] = \mu_i$.

It remains to bound the value $\mathbb{E}[\hat{\mu}_1(t_i(k))]$. Intuitively, when μ_1 is large, $\mathbb{E}[\hat{\mu}_1(t_i(k))]$ cannot be small, since those random variables are with mean μ_1. Indeed, when $\mu_1 > 0.9$ and D_1 is a Bernoulli distribution, one can prove that $\mathbb{E}[\hat{\mu}_1(t_i(k))] \geq \frac{\mu_1}{2} - 2\delta(T)$ with a probabilistic argument, where $\delta(T)$ converges to 0 as T goes to infinity. Thus, for large μ_1 and small μ_2 (so are μ_i for $i \geq 2$), we have that $\mathbb{E}[\hat{\mu}_1(t_i(k))] - \mu_i = \Omega(\mu_1 - \mu_i)$ holds for any i and $k \geq 2$. This means that the compensation we need to pay for pulling arm i once is about $\Theta(\mu_1 - \mu_i) = \Omega(\Delta_i)$. Thus, the total compensation $\Omega\left(\sum_{i=2}^{N} \frac{\Delta_i \log T}{KL(D_i, D_1)}\right)$. □

P. Frazier, D. Kempe, J. Kleinberg, and R. Kleinberg. Incentivizing exploration. In Fifteenth ACM Conference on Economics and Computation, pages 5–22, 2014.

KC MAB – Algorithm: UCB

Upper Confidence Bound

1: **for** $t = 1, 2, \cdots, N$ **do**
2: Choose arm $a(t) = t$.
3: **end for**
4: **for** $t = N + 1, \cdots$ **do**
5: For all arm i, compute $r_i(t) = \sqrt{\frac{2 \log t}{N_i(t)}}$ and $u_i(t) = \hat{\mu}_i(t) + r_i(t)$
6: Choose arm $a(t) = \text{argmax}_i\, u_i(t)$ (with compensation $\max_j \hat{\mu}_j(t) - \hat{\mu}_{a(t)}(t)$)
7: **end for**

Algorithm 1: The UCB algorithm for KCMAB.

Only different point is compensation.

Deterministic policy: UCB1.
Initialization: Play each machine once.
Loop:

- Play machine j that maximizes $\bar{x}_j + \sqrt{\frac{2 \ln n}{n_j}}$, where $\bar{x}_j$ is the average reward obtained from machine j, n_j is the number of times machine j has been played so far, and n is the overall number of plays done so far.

Algorithm 2: The original UCB algorithm

P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time analysis of the multi armed bandit problem. Machine learning, 47(2-3):235–256, 2002.

KC MAB – Algorithm: UCB

Theorem 2. *In Algorithm 1, we have that*

$$Com(T) \leq \sum_{i=2}^{N} \frac{16 \log T}{\Delta_i} + \frac{2N\pi^2}{3}$$

Proof Sketch: First of all, it can be shown that the each sub-optimal arm is pulled for at most $\frac{8}{\Delta_i^2} \log T$ times in Algorithm 1 with high probability. Since in every time slot t the long-term controller choose the arm $a(t) = \text{argmax}_j \hat{\mu}_j(t) + r_j(t)$, we must have $\hat{\mu}_{a(t)}(t) + r_{a(t)}(t) = \max_j(\hat{\mu}_j(t) + r_j(t)) > \max_j \hat{\mu}_j(t)$. This implies that the compensation is at most $r_{a(t)}(t)$. Moreover, if arm $a(t)$ has been pulled the maximum number of times, i.e., $N_{a(t)}(t) = \max_j N_j(t)$, then $r_{a(t)}(t) = \min_j r_j(t)$ (by definition). Thus, $\hat{\mu}_{a(t)}(t) = \max_j \hat{\mu}_j(t)$, which means that the controller does not need to pay any compensation.

Step 1

if arm has been pulled the maximum number of times

Step 2

first inequality: above red box

(a) Jensen inequality

last inequlity is proved already in the previous paper.

To prove Theorem 1 we show that, for any suboptimal machine j,

$$I\!E[T_j(n)] \leq \frac{8}{\Delta_j^2} \ln n$$

Next, for any sub-optimal arm i, with high probability the compensation that the long-term controller pays for it can be upper bounded by:

$$Com_i(T) \leq \mathbb{E}\left[\sum_{\tau=1}^{N_i(T)} \sqrt{\frac{2 \log T}{\tau}}\right] \leq \mathbb{E}\left[\sqrt{8 N_i(T) \log T}\right] \overset{(a)}{\leq} \sqrt{8 \mathbb{E}[N_i(T)] \log T} \leq \frac{8 \log T}{\Delta_i}$$

Here the inequality (a) holds because $\sqrt{x}$ is concave. As for the optimal arm, when $N_1(t) \geq \sum_{i=2}^{N} \frac{8 \log T}{\Delta_i^2}$, with high probability $N_1(t) = \max_j N_j(t)$. Thus, the controller does not need to pay compensation in time slots with $a(t) = 1$ and $N_1(t) \geq \sum_{i=2}^{N} \frac{8 \log T}{\Delta_i^2}$. Using the same argument, the compensation for arm 1 is upper bounded by $Com_1(T) \leq \sum_{i=2}^{N} \frac{8 \log T}{\Delta_i}$ with high probability. Therefore, the overall compensation upper bound is given by $Com(T) \leq \sum_{i=2}^{N} \frac{16 \log T}{\Delta_i}$ with high probability. □

P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time analysis of the multi armed bandit problem. Machine learning, 47(2-3):235–256, 2002.

KC MAB – Algorithm: Modified ε-greedy

Input: ϵ,
for $t = 1, 2, \cdots, N$ **do**
  Choose arm $a(t) = t$.
end for
$a_e \leftarrow 1$
for $t = N + 1, \cdots$ **do**
  With probability $\min\{1, \frac{\epsilon}{t}\}$, choose arm $a(t) = a_e$ and set $a_e \leftarrow (a_e \mod N) + 1$ (with compensation $\max_j \hat{\mu}_j(t) - \hat{\mu}_{a(t)}(t)$).
  Else, choose the arm $a(t) = \text{argmax}_i \hat{\mu}_i(t)$.
end for

Algorithm 2: The modified ε-greedy algorithm for KCMAB.

Notes

1. They use the round robin method to explore the arms.
   - This guarantees that, given the number of total explorations, each arm will be explored a deterministic number of times.
2. The number of explorations of each single arm is almost the same as classic ε-greedy algorithm in expectation.

Randomized policy: ε_n-GREEDY.
Parameters: $c > 0$ and $0 < d < 1$.
Initialization: Define the sequence $\varepsilon_n \in (0, 1]$, $n = 1, 2, \ldots$, by

$$\varepsilon_n \stackrel{\text{def}}{=} \min\left\{1, \frac{cK}{d^2 n}\right\}$$

Loop: For each $n = 1, 2, \ldots$

- Let i_n be the machine with the highest current average reward.
- With probability $1 - \varepsilon_n$ play i_n and with probability ε_n play a random arm.

P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time analysis of the multi armed bandit problem. Machine learning, 47(2-3):235–256, 2002.

KC MAB – Algorithm: Modified ε-greedy

Theorem 3. *In Algorithm 2, if we have* $\epsilon = \frac{cN}{\Delta_2^2}$, *then*

$$Com(T) \leq \sum_{i=2}^{N} \frac{c\Delta_i \log T}{\Delta_2^2} + \frac{N^2}{2\Delta_2}\sqrt{c \log T}. \quad (1)$$

Proof Sketch: Firstly, our modified ε-greedy algorithm chooses the arm with the largest empirical mean in non-exploration steps. Thus, we only need to consider the exploration steps, i.e., steps during which we choose to explore arms according to round-robin. Let $t_i^{\varepsilon}(k)$ be the time slot that we explore arm i for the k-th time. Then the compensation the controller has to pay in this time slot is $\mathbb{E}[\max_j \hat{\mu}_j(t_i^{\varepsilon}(k)) - \hat{\mu}_i(t_i^{\varepsilon}(k))]$.

Since the rewards are independent of whether we choose to explore, one sees that $\mathbb{E}[\hat{\mu}_i(t_i^{\varepsilon}(k))] = \mu_i$. Thus, we can decompose $\mathbb{E}[\max_j \hat{\mu}_j(t_i^{\varepsilon}(k)) - \hat{\mu}_i(t_i^{\varepsilon}(k))]$ as follows:

$$\begin{aligned} \mathbb{E}[\max_j \hat{\mu}_j(t_i^{\varepsilon}(k)) - \hat{\mu}_i(t_i^{\varepsilon}(k))] &= \mathbb{E}[\max_j(\hat{\mu}_j(t_i^{\varepsilon}(k)) - \mu_i)] \\ &\leq \mathbb{E}[\max_j(\hat{\mu}_j(t_i^{\varepsilon}(k)) - \mu_j)] + \mathbb{E}[\max_j(\mu_j - \mu_i)]. \quad (2) \end{aligned}$$

first term　　second term

epsilon greedy probability

Since $\mathbb{E}[|T_i|] = \frac{c}{\Delta_2^2}\log T$, we can bound the first term in (2) as $\frac{N^2\sqrt{c\log T}}{2\Delta_2}$. Summing this with $\sum_{i=2}^{N} \frac{c\Delta_i \log T}{\Delta_2^2}$ above for the second term, we obtain the compensation upper bound in (1). □

second term's bound

The second term in (2) is bounded by $\Delta_i = \mu_1 - \mu_i$. Summing over all these steps and all arms, we obtain the first term $\sum_{i=2}^{N} \frac{c\Delta_i \log T}{\Delta_2^2}$ in our bound (1).

We turn to the first term in (2), i.e., $\mathbb{E}[\max_j(\hat{\mu}_j(t_i^{\varepsilon}(k)) - \mu_j)]$. We see that it is upper bounded by

$$\mathbb{E}[\max_j(\hat{\mu}_j(t_i^{\varepsilon}(k)) - \mu_j)] \leq \mathbb{E}[\max_j(\hat{\mu}_j(t_i^{\varepsilon}(k)) - \mu_j)^+] \leq \sum_j \mathbb{E}[(\hat{\mu}_j(t_i^{\varepsilon}(k)) - \mu_j)^+]$$

because they use the round robin method to explore

where $(*)^+ = \max\{*, 0\}$. When arm i has been explored k times (line 7 in Algorithm 2), we know that all other arms have at least k observations (in the first N time slots, there is one observation for each arm). Hence, $\mathbb{E}[(\hat{\mu}_j(t_i^{\varepsilon}(k)) - \mu_j)^+] = \frac{1}{2}\mathbb{E}[|\hat{\mu}_j(t_i^{\varepsilon}(k)) - \mu_j|] \leq \frac{1}{4\sqrt{k}}$ (the equality is due to the fact that $\mathbb{E}[|x|] = 2\mathbb{E}[x^+]$ if $\mathbb{E}[x] = 0$).

Suppose arm i is been explored in time set $T_i = \{t_i^{\varepsilon}(1), \cdots\}$. Then,

$$\sum_{k \leq |T_i|} \mathbb{E}[\max_j(\hat{\mu}_j(t_i^{\varepsilon}(k)) - \mu_j)^+] \leq \sum_{k \leq |T_i|} \frac{N}{4\sqrt{k}} \leq \frac{N\sqrt{|T_i|}}{2}$$

P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time analysis of the multi armed bandit problem. Machine learning, 47(2-3):235–256, 2002.

KC MAB – Algorithm: Modified Thompson Sampling policy

Init: $\alpha_i = 1, \beta_i = 1$ for each arm i.
for $t = 1, 2, \cdots, N$ **do**
  Choose arm $a(t) = t$ and receive the observation $X(t)$.
  Update$(\alpha_{a(t)}, \beta_{a(t)}, X(t))$
end for
for $t = N+1, N+3, \cdots$ **do**
  For all i sample values $\theta_i(t)$ from Beta distribution $\mathrm{B}(\alpha_i, \beta_i)$;
  Play action $a_1(t) = \mathrm{argmax}_i\, \hat{\mu}_i(t)$, get the observation $X(t)$. Update$(\alpha_{a_1(t)}, \beta_{a_1(t)}, X(t))$
  Play action $a_2(t+1) = \mathrm{argmax}_i\, \theta_i(t)$ (with compensation $\max_j \hat{\mu}_j(t+1) - \hat{\mu}_{a_2(t+1)}(t+1)$), receive the observation $X(t+1)$. Update$(\alpha_{a_2(t+1)}, \beta_{a_2(t+1)}, X(t+1))$
end for

Algorithm 3: The Modified Thompson Sampling Algorithm for KCMAB.

Input: $\alpha_i, \beta_i, X(t)$
Output: updated α_i, β_i
$Y(t) \leftarrow 1$ with probability $X(t)$, 0 with probability $1 - X(t)$
$\alpha_i \leftarrow \alpha_i + Y(t)$; $\beta_i \leftarrow \beta_i + 1 - Y(t)$

Algorithm 4: Procedure Update

Notes

1. Theirs is motivated by the idea of the LUCB algorithm.
2. They divide time into rounds containing two time steps each, and pull not only the arm with largest empirical mean, but also the arm with largest sample value in each round.

Algorithm 1 Thompson Sampling for Bernoulli bandits
For each arm $i = 1, \ldots, N$ set $S_i = 0, F_i = 0$.
foreach $t = 1, 2, \ldots,$ **do**
  For each arm $i = 1, \ldots, N$, sample $\theta_i(t)$ from the Beta$(S_i + 1, F_i + 1)$ distribution.
  Play arm $i(t) := \arg\max_i \theta_i(t)$ and observe reward r_t.
  If $r = 1$, then $S_{i(t)} = S_{i(t)} + 1$, else $F_{i(t)} = F_{i(t)} + 1$.
end

Shipra Agrawal, Navin Goyal. Analysis of Thompson Sampling for the Multi-armed Bandit Problem. JMLR: Workshop and Conference Proceedings vol23 (2012) 39.1~39.26, 2012

KC MAB – Algorithm: Modified Thompson Sampling policy

Theorem 4. *In Algorithm 3, we have*

$$Reg(T) \leq \sum_i \frac{2\Delta_i}{(\Delta_i - \varepsilon)^2} \log T + O\left(\frac{N}{\varepsilon^4}\right) + F_1(\boldsymbol{\mu})$$

for some small $\varepsilon < \Delta_2$ and $F_1(\boldsymbol{\mu})$ does not depend on (T, ε). As for compensation, we have:

$$Com(T) \leq \sum_i \frac{8}{\Delta_i - \varepsilon} \log T + N \log T + O\left(\frac{N}{\varepsilon^4}\right) + F_2(\boldsymbol{\mu})$$

where $F_2(\boldsymbol{\mu})$ does not depend on (T, ε) as well.

Proof Sketch: In round $(t, t+1)$, we assume that we first run the arm with largest empirical mean on time slot t and call t an empirical step. Then we run the arm with largest sample on time slot $t+1$ and call $t+1$ a sample step.

We can bound the number of sample steps during which we pull a sub-optimal arm, using existing results in [1], since all sample steps form an approximation of the classic TS algorithm. Moreover, [11] shows that in sample steps, the optimal arm is pulled for many times (at least t^b at time t with a constant $b \in (0,1)$). Thus, after several steps, the empirical mean of the optimal arm will be accurate enough. Then, if we choose to pull sub-optimal arm i during empirical steps, arm i must have an inaccurate empirical mean. Since the pulling will update its empirical mean, it is harder and harder for the arm's empirical mean to remain inaccurate. As a result, it cannot be pulled for a lot of times during the empirical steps as well.

Next, we discuss how to bound its compensation. It can be shown that with high probability, we always have $|\theta_i(t) - \hat{\mu}_i(t)| \leq r_i(t)$, where $r_i(t) = \sqrt{\frac{2\log t}{N_t(t)}}$ is defined in Algorithm 1. Thus, we can focus on the case that $|\theta_i(t) - \hat{\mu}_i(t)| \leq r_i(t)$ happens for any i and t. Note that we do not need to pay compensation in empirical steps. In sample steps, suppose we pull arm i and the largest empirical mean is in arm $j \neq i$ at the beginning of this round. Then, we need to pay $\max_k \hat{\mu}_k(t+1) - \hat{\mu}_i(t+1)$, which is upper bounded by $\hat{\mu}_j(t) - \hat{\mu}_i(t) + (\hat{\mu}_j(t+1) - \hat{\mu}_j(t))^+ \leq \hat{\mu}_j(t) - \hat{\mu}_i(t) + \frac{1}{N_j(t)}$ (here $\hat{\mu}_i(t+1) = \hat{\mu}_i(t)$). As $\theta_i(t) \geq \theta_j(t)$, we must have $\hat{\mu}_i(t) + r_i(t) \geq \theta_i(t) \geq \theta_j(t) \geq \hat{\mu}_j(t) - r_j(t)$, which implies $\hat{\mu}_j(t) - \hat{\mu}_i(t) \leq r_i(t) + r_j(t)$. Thus, what we need to pay is at most $r_i(t) + r_j(t) + \frac{1}{N_j(t)}$ if $i \neq j$, in which case we can safely assume that we pay $r_j(t) + \frac{1}{N_j(t)}$ during empirical steps, and $r_i(t)$ during sample steps.

For an sub-optimal arm i, we have $Com_i(T) \leq \sum_i \frac{4}{\Delta_i - \varepsilon} \log T + O(\frac{1}{\varepsilon^4}) + F_1(\boldsymbol{\mu}) + \log T$ (summing over $r_i(t)$ gives the same result as in the UCB case, and summing over $\frac{1}{N_i(t)}$ is upper bounded by $\log T$). As for arm 1, when $a_1(t) = a_2(t+1) = 1$, we do not need to pay $r_1(t)$ twice. In fact, we only need to pay at most $\frac{1}{N_1(t)}$. Then, the number of time steps that $a_1(t) = a_2(t+1) = 1$ does not happen is upper bounded by $\sum_{i=2}^N \left(\frac{2}{(\Delta_i - \varepsilon)^2} \log T\right) + O\left(\frac{N}{\varepsilon^4}\right) + F_1(\boldsymbol{\mu})$, which is given by regret analysis. Thus, the compensation we need to pay on arm 1 is upper bounded by $\sum_i \frac{4}{\Delta_i - \varepsilon} \log T + O(\frac{1}{\varepsilon^4}) + F_1(\boldsymbol{\mu}) + \log T$. Combining the above, we have the compensation bound $Com(T) \leq \sum_i \frac{8}{\Delta_i - \varepsilon} \log T + N \log T + O(\frac{1}{\varepsilon^4}) + F_2(\boldsymbol{\mu})$. □

Shipra Agrawal, Navin Goyal. Analysis of Thompson Sampling for the Multi-armed Bandit Problem. JMLR: Workshop and Conference Proceedings vol23 (2012) 39.1~39.26, 2012

KC MAB – Experiment

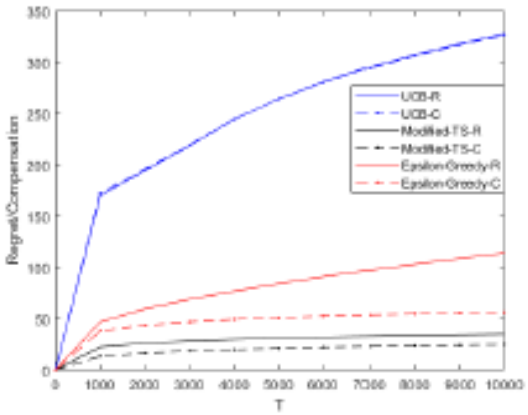


Figure 1: Regret and Compensation of Three policies.

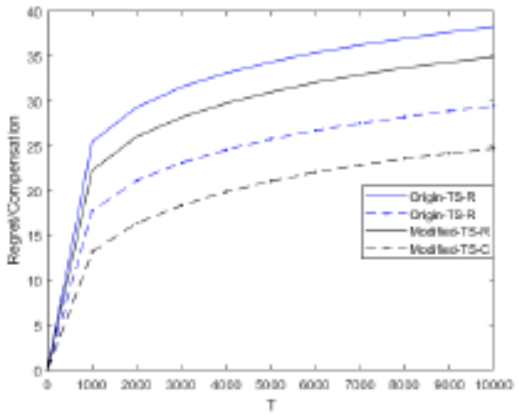


Figure 2: Regret and Compensation of TS and modified-TS.

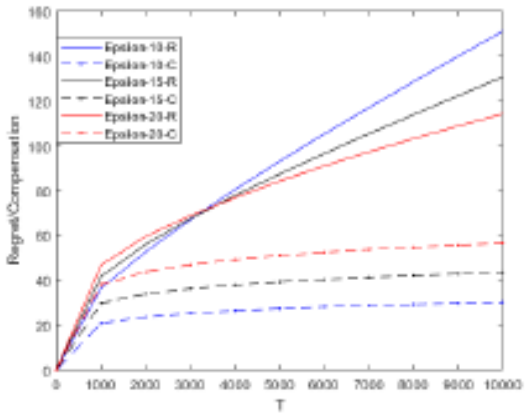


Figure 3: Regret and Compensation of modified ε-greedy.

In experiments, there are a total of nine arms with expected reward vector μ = [0.9, 0.8, 0.7, 0.6, 0.5, 0.4, 0.3, 0.2, 0.1]
They run the game for T=10,000 time steps.

Shipra Agrawal, Navin Goyal. Analysis of Thompson Sampling for the Multi-armed Bandit Problem. JMLR: Workshop and Conference Proceedings vol23 (2012) 39.1~39.26, 2012

Contribution

Contribution

1. Design a proper compensation policy, so as to minimize his regret with minimum compensation. KCMAB is a non-Bayesian and non-discounted extension of the model.
2. This compensation lower bound has the same order as the regret lower bound, which means that one cannot expect a compensation to be much less than its regret, if the regret is already small.
3. All these algorithms have O(logT) regret upper bounds while using compensation upper bounded O(logT), which matches the lower bound (in order).
4. In experiments, modified TS policy behaves better than UCB policy, while the modified epsilon-greedy policy has regret and compensation slightly larger than those under the modified-TS policy.

Future work

Future Work

1. Bayesian Model
2. Budgeted MAB model
3. …