

# Rotting Infinitely Many-Armed Bandits

Jung-hun Kim<sup>1</sup>, Milan Vojnović<sup>2</sup>, Se-Young Yun<sup>1</sup>

<sup>1</sup>KAIST, <sup>2</sup>London School of Economics

# Motivation

---

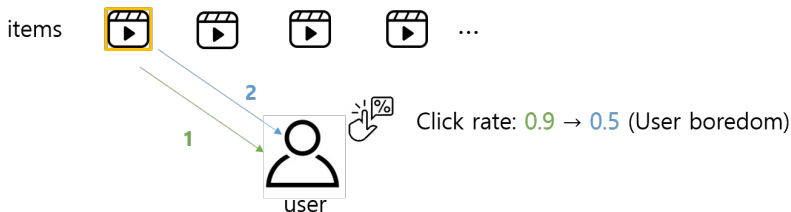
## Infinitely many-armed bandits with rotting rewards

A variant of multi-armed bandits with **infinitely many actions** where the mean reward for a selected action is **decreasing**.

# Motivation

## Infinitely many-armed bandits with rotting rewards

A variant of multi-armed bandits with **infinitely many actions** where the mean reward for a selected action is **decreasing**.



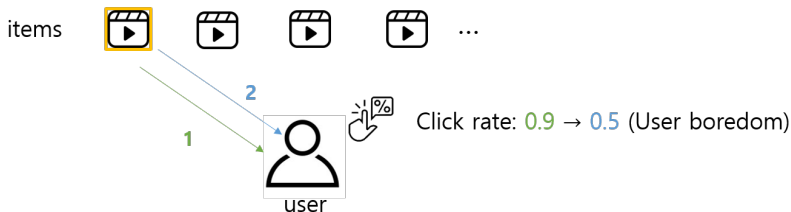
## Applications

- Decreasing click-through rate from **user boredom** in recommender systems

# Motivation

## Infinitely many-armed bandits with rotting rewards

A variant of multi-armed bandits with **infinitely many actions** where the mean reward for a selected action is **decreasing**.



## Applications

- Decreasing click-through rate from **user boredom** in recommender systems
- Decreasing medicine efficacy from **drug tolerance** in clinical trials

# Related Work

---

## Rotting bandits with finite arms

- This problem deals with finite arms with rotting rewards. Seznec et al. 2019; Seznec et al. 2020 proposed algorithms achieving  $\tilde{O}(\sqrt{KT})$  regret bound which is the same as the case of stationary MAB.

# Related Work

---

## Rotting bandits with finite arms

- This problem deals with finite arms with rotting rewards. Seznec et al. 2019; Seznec et al. 2020 proposed algorithms achieving  $\tilde{O}(\sqrt{KT})$  regret bound which is the same as the case of stationary MAB.

## Infinitely many-armed bandits with stationary rewards

- This problem deals with infinite arms with stationary rewards. Berry et al. 1997; Wang et al. 2008; Bonald et al. 2013; Bayati et al. 2020 proposed algorithms that achieve at most  $\tilde{O}(\sqrt{T})$  in the case of uniformly generated mean rewards.

# Related Work

---

## Rotting bandits with finite arms

- This problem deals with finite arms with rotting rewards. Seznec et al. 2019; Seznec et al. 2020 proposed algorithms achieving  $\tilde{O}(\sqrt{KT})$  regret bound which is the same as the case of stationary MAB.

## Infinitely many-armed bandits with stationary rewards

- This problem deals with infinite arms with stationary rewards. Berry et al. 1997; Wang et al. 2008; Bonald et al. 2013; Bayati et al. 2020 proposed algorithms that achieve at most  $\tilde{O}(\sqrt{T})$  in the case of uniformly generated mean rewards.

In our work, we consider **rotting bandits** with **infinitely many arms**.

# Problem Statement

---

## Rotting rewards

- At time  $t$ , an agent selects an arm  $a_t$  and receives a reward as

$$r_t = \mu_t(a_t) + \eta_t.$$



# Problem Statement

---

## Rotting rewards

- At time  $t$ , an agent selects an arm  $a_t$  and receives a reward as

$$r_t = \mu_t(a_t) + \eta_t.$$

- The mean reward of the selected arm  $a_t$  decreases as

$$\mu_{t+1}(a_t) = \mu_t(a_t) - \varrho_t, \text{ where } 0 \leq \varrho_t \leq \varrho = o(1).$$

# Problem Statement

---

## Rotting rewards

- At time  $t$ , an agent selects an arm  $a_t$  and receives a reward as

$$r_t = \mu_t(a_t) + \eta_t.$$

- The mean reward of the selected arm  $a_t$  decreases as

$$\mu_{t+1}(a_t) = \mu_t(a_t) - \varrho_t, \text{ where } 0 \leq \varrho_t \leq \varrho = o(1).$$

## Infinitely many arms

- There exist infinitely many actions with which an agent deals at each time.
- Each mean reward is generated from the uniform distribution  $[0, 1]$ .

# Problem Statement

---

## Rotting rewards

- At time  $t$ , an agent selects an arm  $a_t$  and receives a reward as

$$r_t = \mu_t(a_t) + \eta_t.$$

- The mean reward of the selected arm  $a_t$  decreases as

$$\mu_{t+1}(a_t) = \mu_t(a_t) - \varrho_t, \text{ where } 0 \leq \varrho_t \leq \varrho = o(1).$$

## Infinitely many arms

- There exist infinitely many actions with which an agent deals at each time.
- Each mean reward is generated from the uniform distribution  $[0, 1]$ .

## Regret

$$\mathbb{E}[R(T)] = \mathbb{E} \left[ \sum_{t=1}^T (1 - \mu_t(a_t)) \right].$$

# Contributions

---

- First, we show a regret lower bound

$$\mathbb{E}[R(T)] = \Omega\left(\max\left\{\varrho^{1/3}T, \sqrt{T}\right\}\right).$$

# Contributions

---

- First, we show a regret lower bound

$$\mathbb{E}[R(T)] = \Omega\left(\max\left\{\varrho^{1/3}T, \sqrt{T}\right\}\right).$$

- Knowing the maximum rotating rate  $\varrho$ , we propose an algorithm achieving

$$\mathbb{E}[R(T)] = \tilde{O}\left(\max\left\{\varrho^{1/3}T, \sqrt{T}\right\}\right).$$

# Contributions

---

- First, we show a regret lower bound

$$\mathbb{E}[R(T)] = \Omega \left( \max \left\{ \varrho^{1/3} T, \sqrt{T} \right\} \right).$$

- Knowing the maximum rotating rate  $\varrho$ , we propose an algorithm achieving

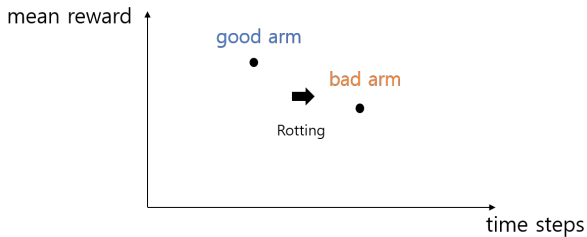
$$\mathbb{E}[R(T)] = \tilde{O} \left( \max \left\{ \varrho^{1/3} T, \sqrt{T} \right\} \right).$$

- Without knowing  $\varrho$ , we propose another algorithm achieving

$$\mathbb{E}[R(T)] = \tilde{O} \left( \max \left\{ \varrho^{1/3} T, T^{3/4} \right\} \right).$$

# Challenge

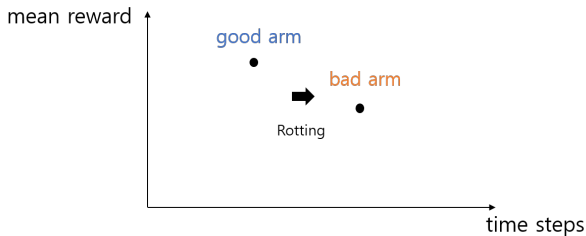
---



- Due to rotting, an initially good arm can become a bad arm by pulling the arm several times.

# Challenge

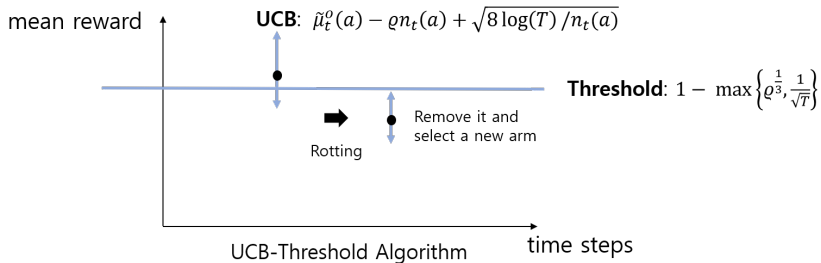
---



- Due to rotting, an initially good arm can become a bad arm by pulling the arm several times.
- Therefore, even though we found a good arm at some point, it is necessary to continue exploring a new good arm over a time horizon  $T$ .

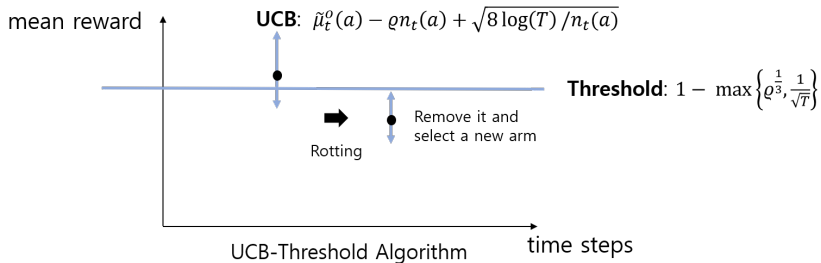


# Algorithms



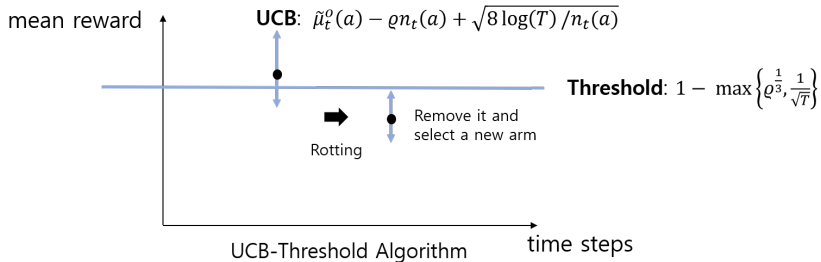
- Our proposed algorithm utilizes **UCB** (upper confidence bound) and **threshold**.

# Algorithms



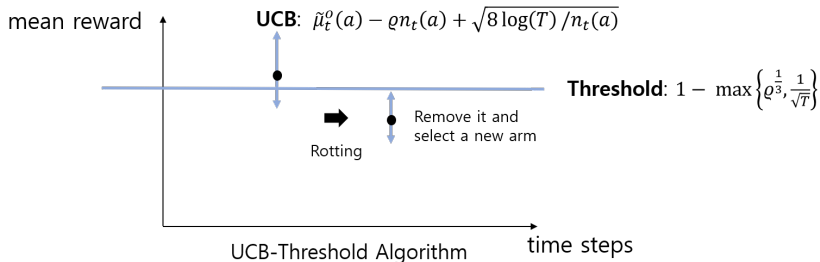
- Our proposed algorithm utilizes **UCB** (upper confidence bound) and **threshold**.
- The algorithm pulls an arm until its UCB value falls below a threshold value, which implies that the arm becomes a bad arm.
- Then, it removes the arm and selects a new arm. Repeat this procedure.

# Algorithms



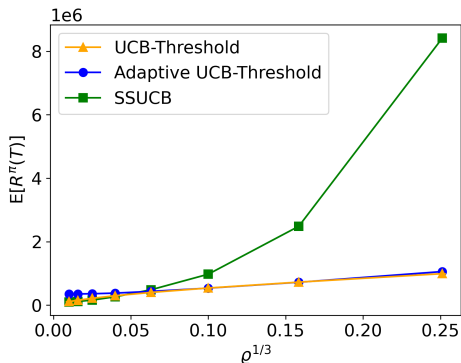
- Our proposed algorithm utilizes **UCB** (upper confidence bound) and **threshold**.
- The algorithm pulls an arm until its UCB value falls below a threshold value, which implies that the arm becomes a bad arm.
- Then, it removes the arm and selects a new arm. Repeat this procedure.
- Information of  $\varrho$  is used to determine the UCB and threshold.

# Algorithms



- Our proposed algorithm utilizes **UCB** (upper confidence bound) and **threshold**.
- The algorithm pulls an arm until its UCB value falls below a threshold value, which implies that the arm becomes a bad arm.
- Then, it removes the arm and selects a new arm. Repeat this procedure.
- Information of  $\varrho$  is used to determine the UCB and threshold.
- Without knowing  $\varrho$ , we propose an adaptive UCB-Threshold algorithm using the Bandit-over-Bandit approach (Cheung et al. 2019).

# Experiment result










- Our algorithms show robust performance by increasing  $\rho$  compared with SSUCB (Bayati et al. 2020), which is known to be near-optimal in a stationary setting.

**Thank you**

**Hall E #1017**

# References

---

-  Bayati, Mohsen et al. (2020). “Unreasonable effectiveness of greedy algorithms in multi-armed bandit with many arms”. In: *Advances in Neural Information Processing Systems 33*, pp. 1713–1723.
-  Berry, Donald A et al. (1997). “Bandit problems with infinitely many arms”. In: *The Annals of Statistics 25.5*, pp. 2103–2116.
-  Bonald, Thomas and Alexandre Proutiere (2013). “Two-target algorithms for infinite-armed bandits with bernoulli rewards”. In: *Advances in Neural Information Processing Systems 26*.
-  Cheung, Wang Chi, David Simchi-Levi, and Ruihao Zhu (2019). “Learning to optimize under non-stationarity”. In: *The 22nd International Conference on Artificial Intelligence and Statistics*. PMLR, pp. 1079–1087.
-  Seznec, Julien et al. (2019). “Rotting bandits are no harder than stochastic ones”. In: *The 22nd International Conference on Artificial Intelligence and Statistics*. PMLR, pp. 2564–2572.
-  Seznec, Julien et al. (2020). “A single algorithm for both restless and rested rotting bandits”. In: *International Conference on Artificial Intelligence and Statistics*. PMLR, pp. 3784–3794.
-  Wang, Yizao, Jean-Yves Audibert, and Rémi Munos (2008). “Algorithms for infinitely many-armed bandits”. In: *Advances in Neural Information Processing Systems 21*.