**ROTTING BANDITS ARE NOT HARDER THAN STOCHASTIC ONES**

J. SEZNEC, A. LOCATELLI, A. CARPENTIER, A. LAZARIC, M. VALKO
julien.seznec@lelivrescolaire.fr

### Setup

**BANDITS:**
- At each round $t$:
  - SELECT an action $i$
  - RECEIVE noisy reward

**ROTTING HYPOTHESIS:**
- Each time we pull an arm, reward decay
- Maximum decay between two pulls: $L$

**GOAL:** Maximize cumulative reward $\sum_{t=0}^{T-1} \mu(t)$

**APPLICATIONS:** education, economics …

### What’s new?

**FEWA algorithm**
- Minimax optimal $T^{2/3} \rightarrow T^{1/2}$
- Solves an open problem
- First problem dependent guarantee - recovers bandits
- Adaptive to $L$ - does not need it as an input

**Prior work**
- Heidari et al. (2016)
  - Optimal oracle - knows $\mu_i$
    - Select the arm with largest next reward
  - Define regret against this optimal oracle
    $$ R_T(z) = \sum_{i=1}^{N} \mu_i(z) - \sum_{i=1}^{N} \sum_{t=0}^{T-1} \mu_i(z) $$
- No noise - but unknown $\mu_i$
  - Select the arm with largest last reward
  - Is minimax optimal.

- Levine et al. (2017)
  - Noisy reward sliding window average:
  - Select the arm with largest average of the $h$ last reward sample
  - Optimize $h$ for the bias-variance trade-off
    $$ \mathbb{E} \left[ R_T(\pi_{SWA}) \right] = O \left( L^{1/3} \sigma^{2/3} K^{1/3} T^{2/3} \right) $$

### Algorithm

**FEWA:** Filtering on Expanding Window Average

**Inputs:** $K$, $\sigma$, $\alpha$
1. for $t \leftarrow K + 1, K + 2, \ldots$ do
2. $\delta_t \leftarrow \frac{1}{\sigma^2}$
3. $h_t \leftarrow 1$
4. $K_{t+1} \leftarrow K$
5. do
6. $K_{h+1} \leftarrow \{ i \in K_h | \tilde{p}_i(x) \geq \max_{j \in K} \tilde{p}_j(x) - 2\epsilon(h, \delta_t) \}$
7. $h_t \leftarrow h_t + 1$
8. while $h_t \leq \min_{i \in K_h} N_{i,t}$
9. SELECT: $\{ i \in K_h | h > N_{i,t} \}$
10. end for

**How does it work?**
- **Method:** Filter the set of arms
- **Based on:** Expanding size of arms history (newest samples first)
- **Statistical tool:** New way of using Hoeffding bound to both
  - Select relevant data history
  - Select arm maximizing exploration/exploitation tradeoff

**Lemma**
- w.p. $1 - \delta_t$, $\forall h \leq N_{i,t}$, $\tilde{p}_i(x) \geq \max_{j \in \mathcal{X}} \mu_j(x) - 4\epsilon(h, \delta_t)$

**Guarantees**

**WORST-CASE BOUND**
$$ \mathbb{E} \left[ R_T(\pi_T) \right] \leq C\sigma \sqrt{K} \log(KT) + KL $$

**PROBLEM-DEPENDENT BOUND**
$$ \mathbb{E} \left[ R_T(\pi_T) \right] \leq \sum_{i \in \mathcal{X}} \mathcal{O} \left( \log(KT) \Delta_i,h_{i-1}^{-1} \right) $$

$\Delta_i,h_T$ = Difference between the worst reward pulled by the optimal policy and the average of the $h$ first overpulls of arm $i$.

$\hat{h}_{i,T}$ = High-probability upper bound on the number of overpulls for FEWA.

### Computational complexity

FEWA has an $O(t)$ time and space complexity:

1. Perform $\log(t)$ filters: $h_t \leftarrow 2h$
2. Keep $\log(t)$ statistics:

### Experiments

**2 arms - Minimal single drop experiment**
- 1 constant arm
- 9 arms with abrupt decay at 1000 pulls
- Geometric sequence of decays: $0.002 \rightarrow 20$

**10 arms - Adapting to multiple decays**
- 1 constant arm
- 9 arms with abrupt decay at 1000 pulls
- Geometric sequence of decays: $0.002 \rightarrow 20$

**Competing against UCB1**
1. Filtering policy $< \text{UCB index policy}$
2. More possible events $\rightarrow$ Looser CB