What Do My Projects Focus on?

Multi-player Multi-armed Bandits (MP-MAB)

Application scene: Cognitive Radio Network, Internet of Things.

Problem fomulation:

- M players, K arms, T total steps.
- Let $[M] := \{1, ..., M\}$ and $[K] := \{1, ..., K\}$.
- At each step t, each player $j \in [M]$ pulls an arm $\pi^j(t) \in [K]$.
- Each player observes $< r^j(t), \eta^j(t) >$, where
  1. $r^j(t) := X^j(t)[1 - \eta^j(t)]$ is a reward, and $X^j(t) \sim \text{Bernoulli}(\mu_{\pi^j(t)})$;
  2. $\eta^j(t) := \mathbb{1}\left[\exists j' \neq j, j' \in [M] : \pi^j(t) = \pi^{j'}(t)\right]$ is a collision indicator.

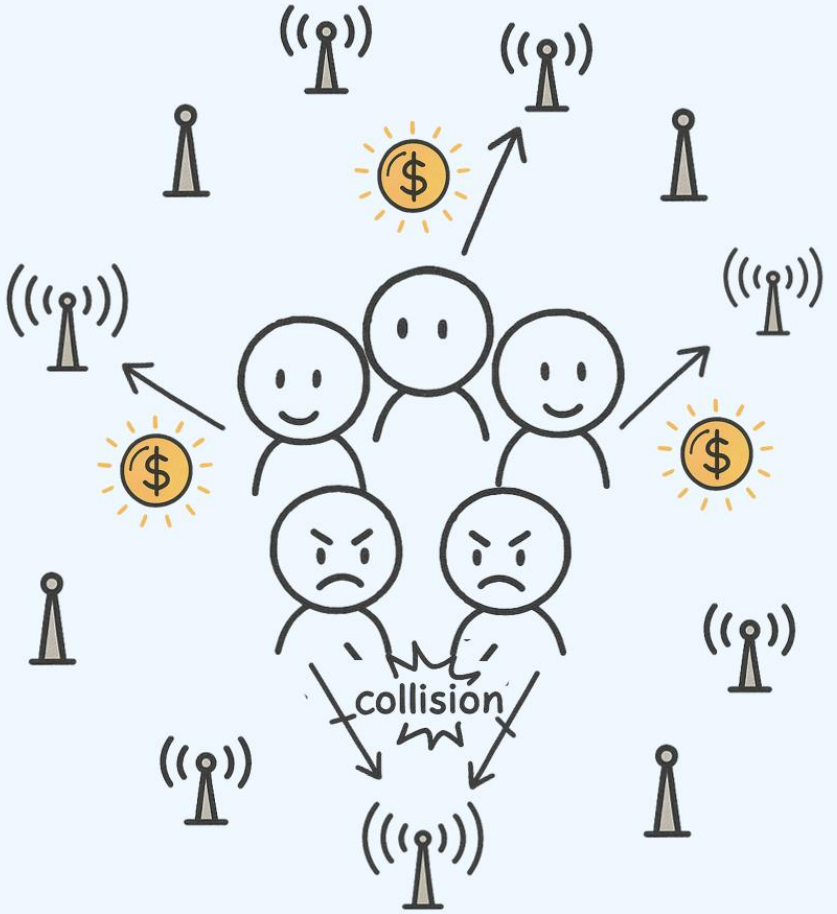
Goal: minimize the regret

$$\mathbb{E}[R(T)] := \sum_{t \leq T} \sum_{k \leq M} \mu_k - \mathbb{E}\left[\sum_{t \leq T} \sum_{j \leq M} r^j(t)\right],$$

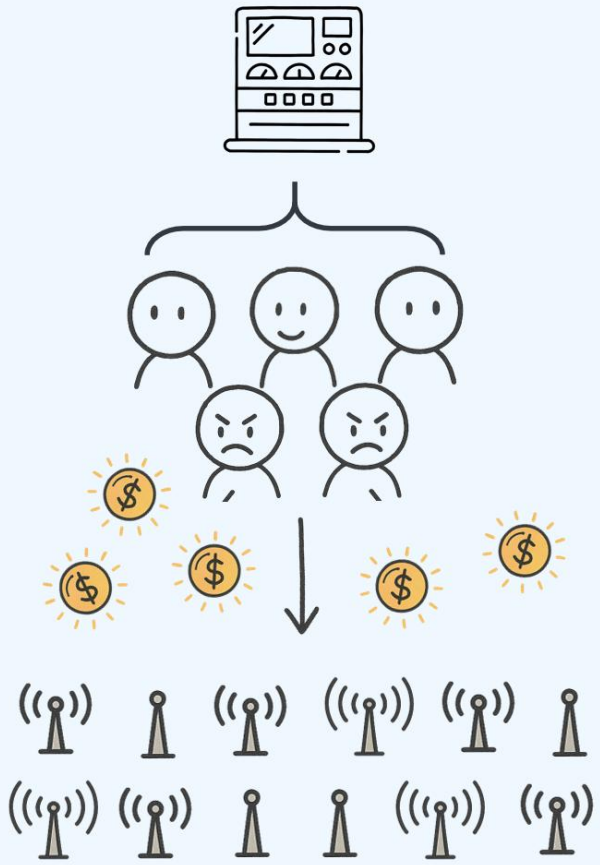
where μ_k is the k-th biggest reward expectation. $\mu_1 \geq \cdots \geq \mu_K$.

Difference with Multi-agent Bandits:

in MP-MAB, players who pull the same arm get zeron reward.

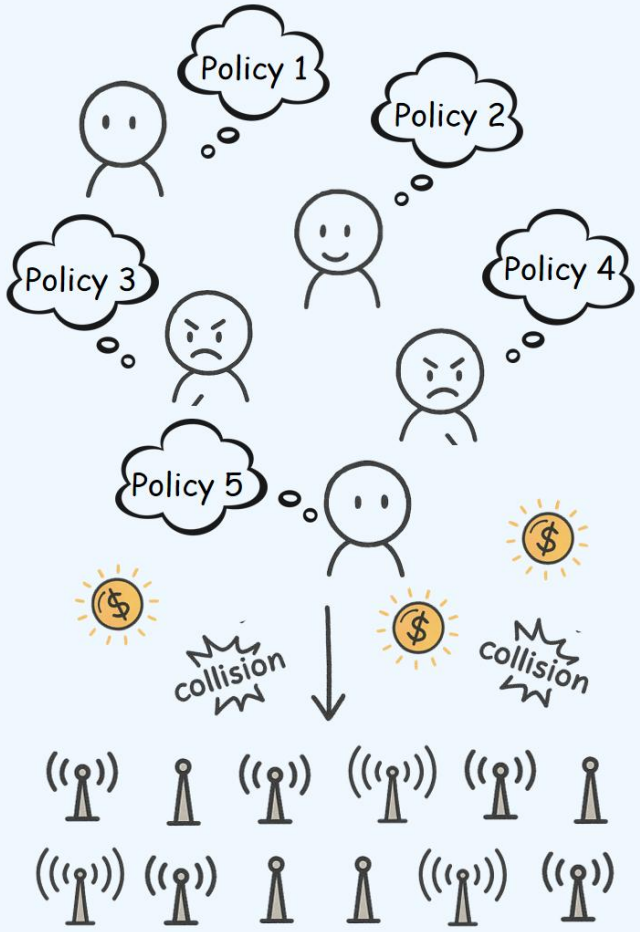


What Do My Projects Focus on?



Centralized MP-MAB

A centra coordinator controls players to pull arms. No collision happen.



Decentralized MP-MAB

There is no centra coordinator. Payers run their own policies and cannot see others' choices, rewards, collisions.

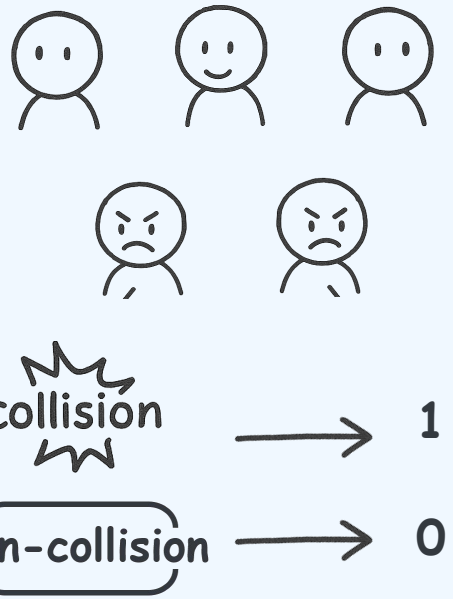
Any uniformly good policy, centralized or not, satisfies (Anantharam et al., 1987):

$$\mathbb{E}[R(T)] \geq \sum_{k>M} \frac{\mu_M - \mu_k}{\mathrm{kl}(\mu_k, \mu_M)} \log T\,.$$

In decentralized setting, when players run their own policies and do not communicate on exploration results,

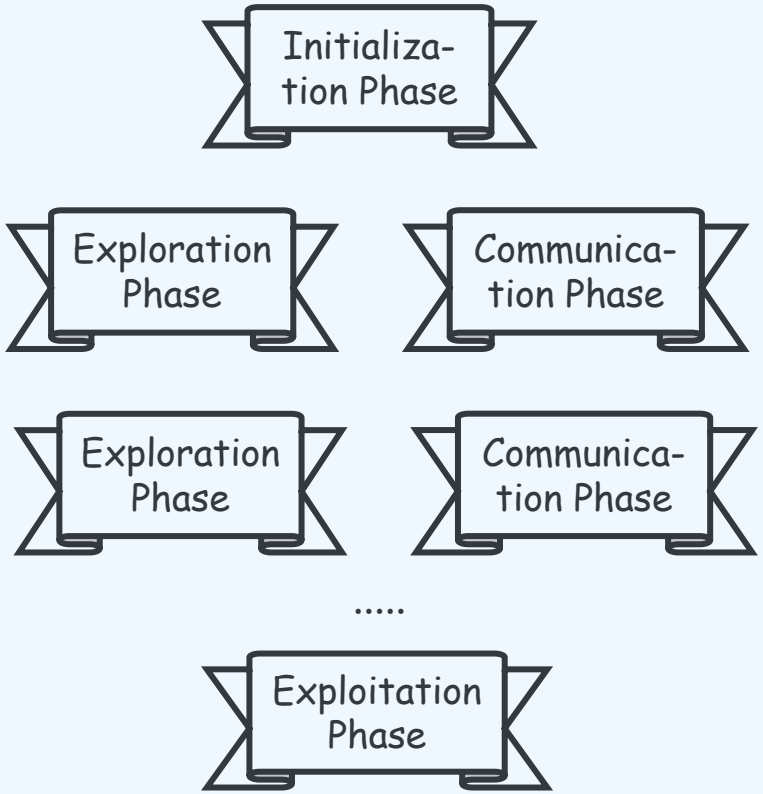
The regret upper bound is often scaled by a multiplicative factor of *M*.

What Do My Projects Focus on?



Design some special phases where players delibrately collide to pass binary information to others.

Players can utilize others' exploration results!



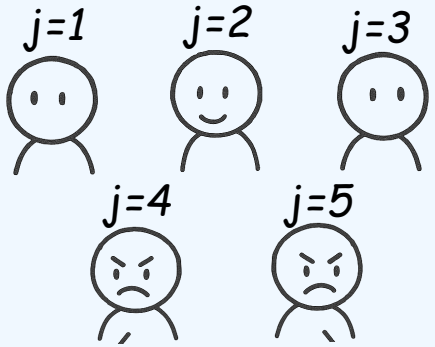
Result:

- Aviod the multiplication of M in the regret of exploration.
- The regret of "communication" is constant.

The problem is solved !

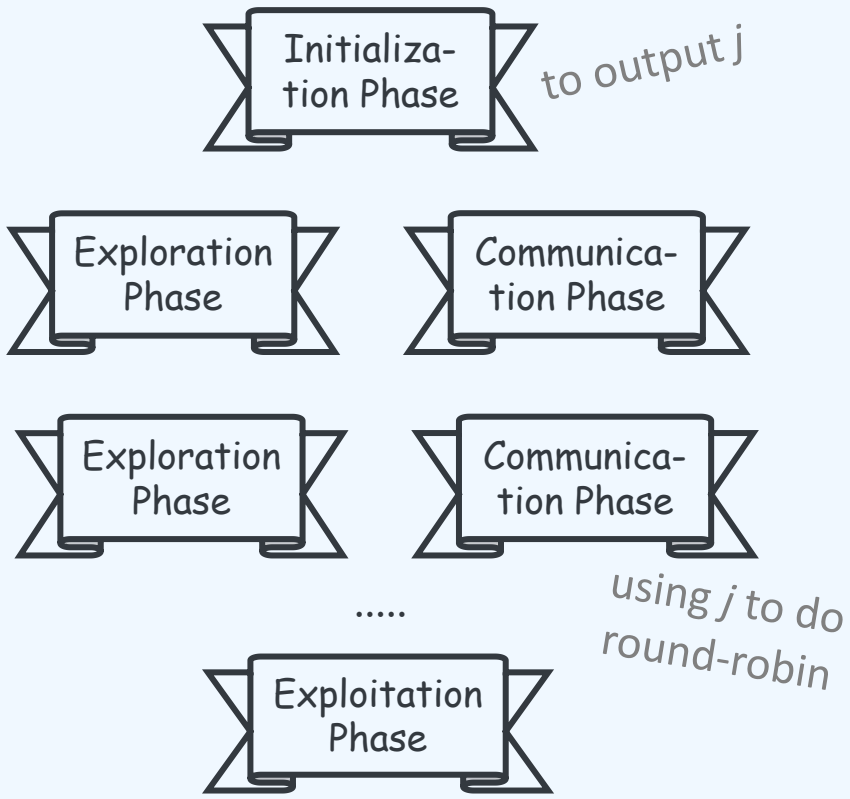
What might happen if players wrongly send/receive *1* or *0*?

What Do My Projects Focus on?



Design some special phases where players delibrately collide to pass binary information to others.

Players can utilize others' exploration results!



Result:

- Aviod the multiplication of M in the regret of exploration.
- The regret of "communication" is constant.

The problem is solved !

What might happen if players wrongly send/receive *1* or *0*?

wrong *1* or *0* in Init Phase:

wrong *j*

↓

failed communication (since using round-robin)

↓

frequent collisions

wrong *1* or *0* in Com Phase:

wrong communication (want to pass "1", but only pass "0")

↓

wrong upate of rewards

e.g.

1 to 0000 0000 0000 0001 to

1000 0000 0000 0001 to 32,769

What Do My Projects Focus on?

I focus on asynchrony in decentralized MP-MAB, motivated by the asynchronous nature of real-world networks.

1. The feedback received by players is asynchronous:

- **Setting:** After pulling arms, players do not observe feedback immediately
- **Challenge:**

  Feedback arrives at different times, breaking the synchronization of updates. However, some updates are critical. Missing these updates leads to frequent collisions.

2. Players themselves are asynchronous:

- **Setting:** They may join or leave the system at different times.
- **Challenge:**

  (1) Players do not know when others enter the system, so it is difficult to leverage the exploration results of others.

  (2) Players do not know when others leave the system. When a player who is exploiting her optimal arm leaves the system, the left arms that are still exploited by players may become sub-optimal.

My Projects

Multi-player Multi-armed Bandits with Delayed Feedback

Accepted by IJCAI-25

Jingqi Fan, Zilong Wang, Shuai Li*, Linghe Kong

Proof is mainly completed in my sophomore year; Experiments are mainly completed in my junior year.

Problem formulation:

- At each step s, each player $j \in [M]$ pulls an arm $\pi^j(t) \in [K]$.
- The environment generates $X^j(s) \sim \text{Bernoulli}(\mu_{\pi^j(s)})$ and $r^j(s) := X^j(s)[1 - \eta^j(s)]$.
- The environment also generates $d^j(s) \sim D_{\pi^j(s)}$, where $D_{\pi^j(s)}$ is an unknow distribution.
- Then, at step $s + d^j(s) - 1$, player j receives the feedback $[r^j(s), \eta^j(s), s]$.

Regret Definition:

$$\mathbb{E}[R(T)] := \sum_{s \le T} \sum_{k \le M} \mu_k - \mathbb{E}\left[\sum_{s \le T} \sum_{j \le M} r^j(s)\right].$$

Assumption:

- $D_k = D_{k'} = D, \forall k \in [K]$. D is sub-Gaussian. σ_d^2 denotes the sub-Gaussian parameter and $\mathbb{E}[d]$ denotes the expectation. Note that σ_d^2 and $\mathbb{E}[d]$ are unknown.
- Each player is aware of her own rank j.

My Projects

Multi-player Multi-armed Bandits with Delayed Feedback

Accepted by IJCAI-25

Jingqi Fan, Zilong Wang, Shuai Li*, Linghe Kong

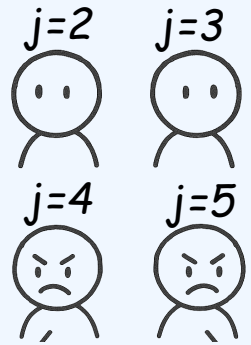
Proof is mainly completed in my sophomore year; Experiments are mainly completed in my junior year.

Method:

- Players randomly initialize $\mathcal{M}^j(1) = \{3, 2, 6, 1, 4\}$ which is a list with $|\mathcal{M}^j(1)| = M$.
- The leader initializes $\mathcal{K} = [K] = \{1, \ldots, K\}$

- Explore arms in $\mathcal{M}^j(1)$ and $\mathcal{K} \setminus \mathcal{M}^j(1)$.
- Add arms with the first M-th higgest reward means into $\mathcal{M}^j(2)$.
- Remove arm k from $\mathcal{K}$ if $\mathrm{UCB}_k^j(t) \le \mathrm{LCB}_\ell^j(t), \forall \ell \in \mathcal{K} \setminus \{k\}$.

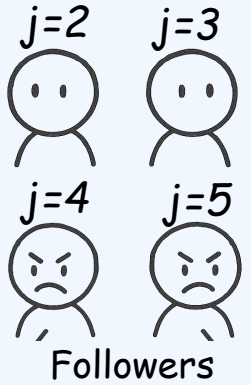
Exploration Phase 1



- Explore arms in $\mathcal{M}^j(1)$

- When $t = 1 \cdot KM \log T$, a Com phase starts.
- Compare $\mathcal{M}^j(1) = \{3, 2, 6, 1, 4\}$ with $\mathcal{M}^j(2) = \{3, 2, 6, 7, 4\}$.
- Send $i_{k^-} = 4$ by pulling arm 1 for M steps.
- Send $k^+ = 7$ by pulling arm 7 for K steps.
- Send $\texttt{End} = \texttt{False}$ by pulling arms in $\mathcal{M}^j(1)$ round-robinly for M steps.

Communication Phase 1



- Receive a collision from the 4-th arm by pulling arms in $\mathcal{M}^j(1)$ round-robinly.
- Receive a collision from arm 7 by pulling arms in $[K]$ round-robinly.
- Receive non-collision (indicating $\texttt{End} = \texttt{False}$) when pulling arms in $\mathcal{M}^j(1)$ round-robinly.

My Projects

Multi-player Multi-armed Bandits with Delayed Feedback

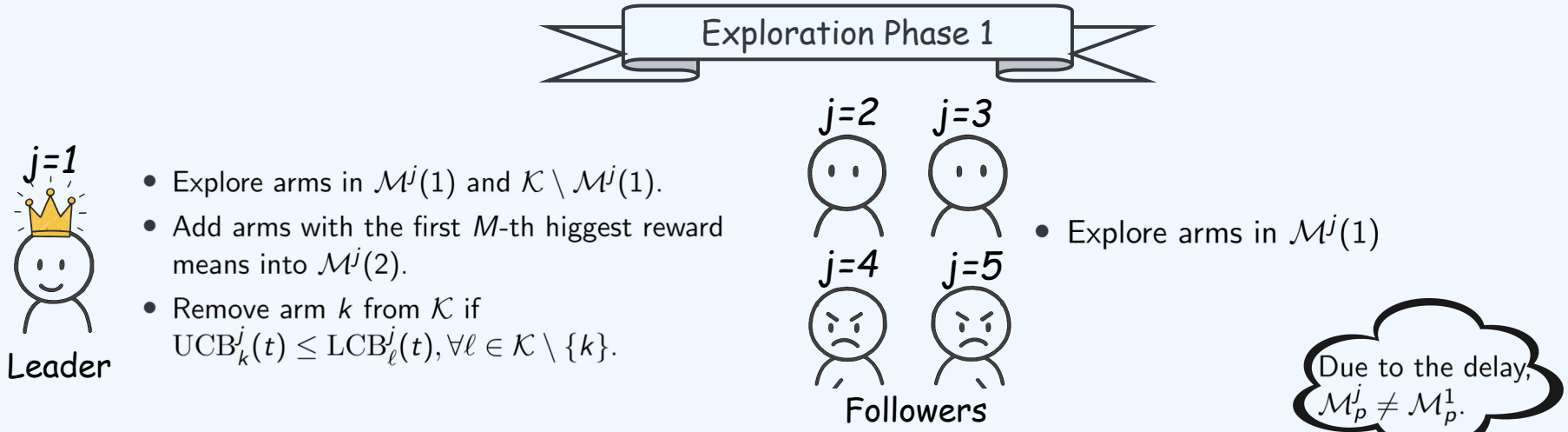
Accepted by IJCAI-25

Jingqi Fan, Zilong Wang, Shuai Li*, Linghe Kong

Proof is mainly completed in my sophomore year; Experiments are mainly completed in my junior year.

Method:

- Players randomly initialize $\mathcal{M}^j(1) = \{3, 2, 6, 1, 4\}$ which is a list with $|\mathcal{M}^j(1)| = M$.
- The leader initializes $\mathcal{K} = [K] = \{1, \ldots, K\}$



- Explore arms in $\mathcal{M}^j(1)$ and $\mathcal{K} \setminus \mathcal{M}^j(1)$.
- Add arms with the first M-th higgest reward means into $\mathcal{M}^j(2)$.
- Remove arm k from $\mathcal{K}$ if $\mathrm{UCB}_k^j(t) \leq \mathrm{LCB}_\ell^j(t), \forall \ell \in \mathcal{K} \setminus \{k\}$.

- Explore arms in $\mathcal{M}^j(1)$

- When $t = 1 \cdot KM \log T$, a Com phase starts.
- Compare $\mathcal{M}^j(1) = \{3, 2, 6, 1, 4\}$ with $\mathcal{M}^j(2) = \{3, 2, 6, 7, 4\}$.
- Send $i_{k^-} = 4$ by pulling arm 1 for M steps.
- Send $k^+ = 7$ by pulling arm 7 for K steps.
- Send $\texttt{End} = \texttt{False}$ by pulling arms in $\mathcal{M}^j(1)$ round-robinly for M steps.

- Receive a collision from the 4-th arm by pulling arms in $\mathcal{M}^j(1)$ round-robinly.
- Receive a collision from arm 7 by pulling arms in $[K]$ round-robinly.
- Receive non-collision (indicating $\texttt{End} = \texttt{False}$) when pulling arms in $\mathcal{M}^j(1)$ round-robinly.

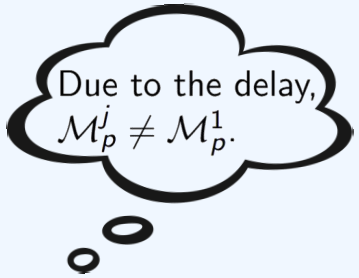
My Projects

Multi-player Multi-armed Bandits with Delayed Feedback

Accepted by IJCAI-25

Jingqi Fan, Zilong Wang, Shuai Li*, Linghe Kong

Proof is mainly completed in my sophomore year; Experiments are mainly completed in my junior year.



Problem here:

- After each Com phase, the latest $\mathcal{M}^j(p)$ becomes inconsistent. Players may repeatedly collide with others until $\mathcal{M}^j(p)$ is updated.
- Inconsistent $\mathcal{M}^j(p)$ leads to incorrect information being propagated during the Com phase. Such errors are irrecoverable for followers.

Method:

When a player j receive a feedback at time t, she update the estimation of $\mathbb{E}[d]$ and σ_d^2 with

$$\hat{\mu}_d^j(t) := \frac{\sum_{s<t}\left(d^j(s)\mathbb{1}\{s+d^j(s)<t\}\right)}{\sum_{s<t}\mathbb{1}\{s+d^j(s)<t\}},$$

$$(\hat{\sigma}_d^2)^j(t) := \frac{\sum_{s<t}\left([d^j(s)-\hat{\mu}_d^j(t)]\mathbb{1}\{s+d^j(s)<t\}\right)^2}{\sum_{s<t}\mathbb{1}\{s+d^j(s)<t\}}.$$

Thus, each player j aims to find $q_j \in \mathbb{N}$ such that

$$q_j = \arg\min_q \left\{ q \mid t > \hat{\mu}_d^j(t) + (p-q)KM\log(T)\sqrt{2(\hat{\sigma}_d^2)^j(t)\log\left((M-1)(K+2M)(T)\right)} \right\}.$$

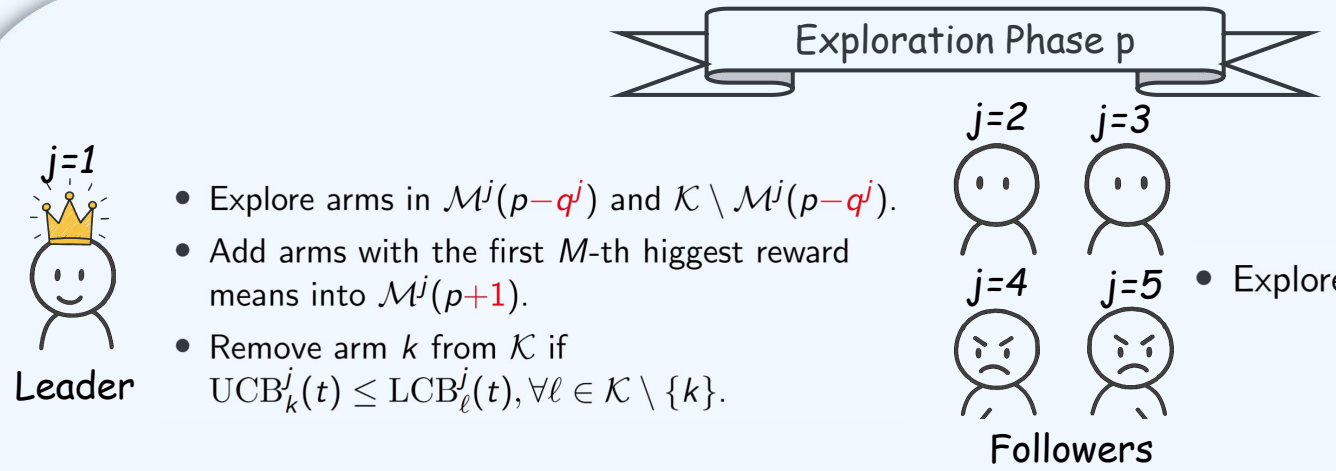
My Projects

Multi-player Multi-armed Bandits with Delayed Feedback

Accepted by IJCAI-25

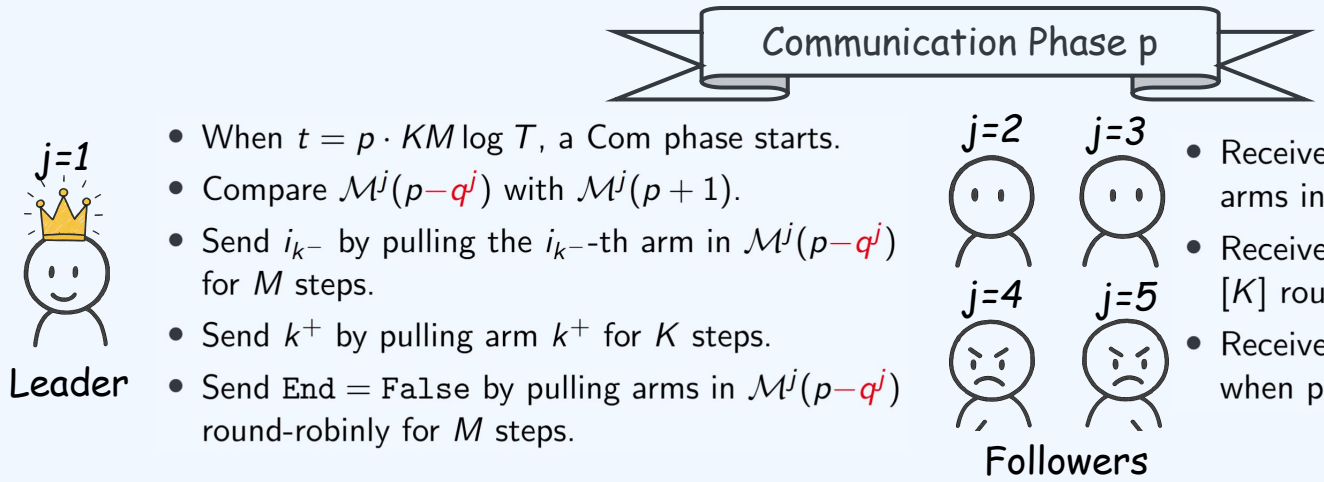
Jingqi Fan, Zilong Wang, Shuai Li*, Linghe Kong

Proof is mainly completed in my sophomore year; Experiments are mainly completed in my junior year.



- Explore arms in $\mathcal{M}^j(p-q^j)$ and $\mathcal{K} \setminus \mathcal{M}^j(p-q^j)$.
- Add arms with the first M-th higgest reward means into $\mathcal{M}^j(p+1)$.
- Remove arm k from $\mathcal{K}$ if $\mathrm{UCB}_k^j(t) \le \mathrm{LCB}_\ell^j(t), \forall \ell \in \mathcal{K} \setminus \{k\}$.

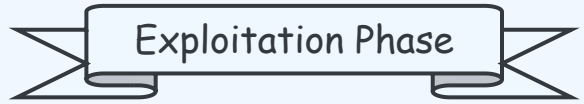
- Explore arms in $\mathcal{M}^j(p-q^j)$



- When $t = p \cdot KM \log T$, a Com phase starts.
- Compare $\mathcal{M}^j(p-q^j)$ with $\mathcal{M}^j(p+1)$.
- Send i_{k^-} by pulling the i_{k^-}-th arm in $\mathcal{M}^j(p-q^j)$ for M steps.
- Send k^+ by pulling arm k^+ for K steps.
- Send $\texttt{End} = \texttt{False}$ by pulling arms in $\mathcal{M}^j(p-q^j)$ round-robinly for M steps.

- Receive a collision from the i_{k^-}-th arm by pulling arms in $\mathcal{M}^j(p-q^j)$ round-robinly.
- Receive a collision from arm k by pulling arms in $[K]$ round-robinly.
- Receive non-collision (indicating $\texttt{End} = \texttt{False}$) when pulling arms in $\mathcal{M}^j(p-q^j)$ round-robinly.

.....



Each player j pulls the j-th arm in $\mathcal{M}^j(p_{\max})$ until T.

My Projects

Multi-player Multi-armed Bandits with Delayed Feedback

Accepted by IJCAI-25

Jingqi Fan, Zilong Wang, Shuai Li*, Linghe Kong

Proof is mainly completed in my sophomore year; Experiments are mainly completed in my junior year.

Theoretical Results

Upper Bound

Let $\Delta_{k,\ell} := \mu_k - \mu_\ell$ and $\Delta := \min_{k \le M} \mu_k - \mu_{k+1}$. In decentralized setting, given any K, M and a quantile $\theta \in (0,1)$, the regret of the algorithm satisfies

$$\mathbb{E}[R(T)] \le \sum_{k>M} \frac{323 \log(T)}{\theta \Delta_{M,k}} + \frac{M}{K-M} \sum_{k>M} \Delta_{1,k} d_1 + \frac{15}{\theta} d_2 + d_3 + C\,,$$

Lower Bound

For any sub-optimal gap set $S_\Delta = \{\Delta_{M,k} \mid \Delta_{M,k} = \mu_{(M)} - \mu_{(k)} \in [0,1]\}$ of cardinality $K - M$ and a quantile $\theta \in (0,1)$, there exists an instance with an order on S_Δ and a sub-Gaussian delay distribution such that

$$\mathbb{E}[R(T)] \ge \sum_{k>M} \frac{(1-o(1)) \log(T)}{2\theta \Delta_{M,k}} + \frac{M}{2K} \sum_{k>M} \Delta_{M,k} d_4 - \frac{2}{\theta}\,,$$

where

$$d_1 = 2\mathbb{E}[d] + \sigma_d \sqrt{3 \log(K)}\,, \quad d_2 = \mathbb{E}[d] + \sigma_d \sqrt{2 \log(\frac{1}{1-\theta})}\,,$$

$$d_3 = \frac{656\sqrt{2}\sigma_d^2}{\theta K^2 M^2} + 3\sqrt{6}\sigma_d\,, \quad C = \sum_{k>M} \frac{195}{\theta \Delta_{M,k}} + \frac{4M}{\Delta^2}\,.$$

My Projects

Multi-player Multi-armed Bandits with Delayed Feedback

Accepted by IJCAI-25

Jingqi Fan, Zilong Wang, Shuai Li*, Linghe Kong

Proof is mainly completed in my sophomore year; Experiments are mainly completed in my junior year.

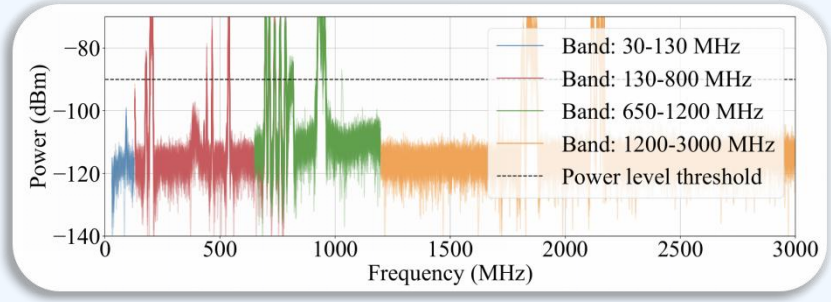
Experimental Results

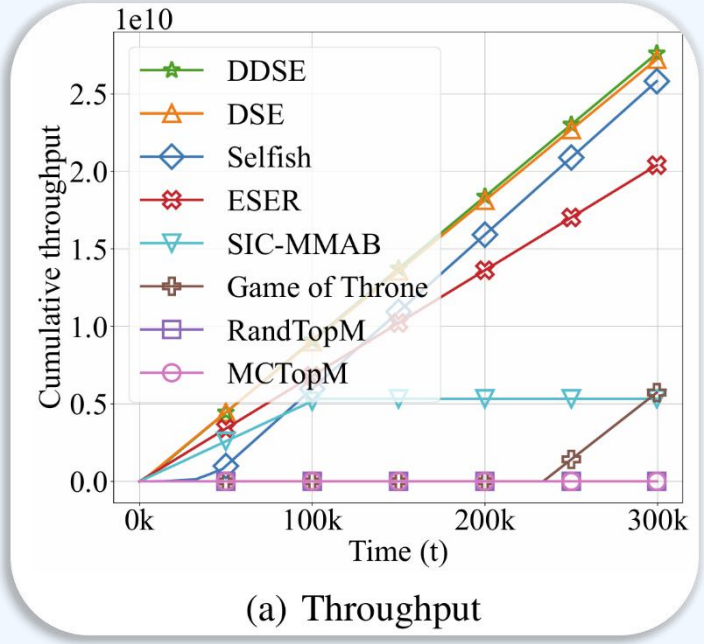
Real-world Experiments

Use spectrum data collected in Finland by a 5G-Xcast project to simulate a real-world environment.

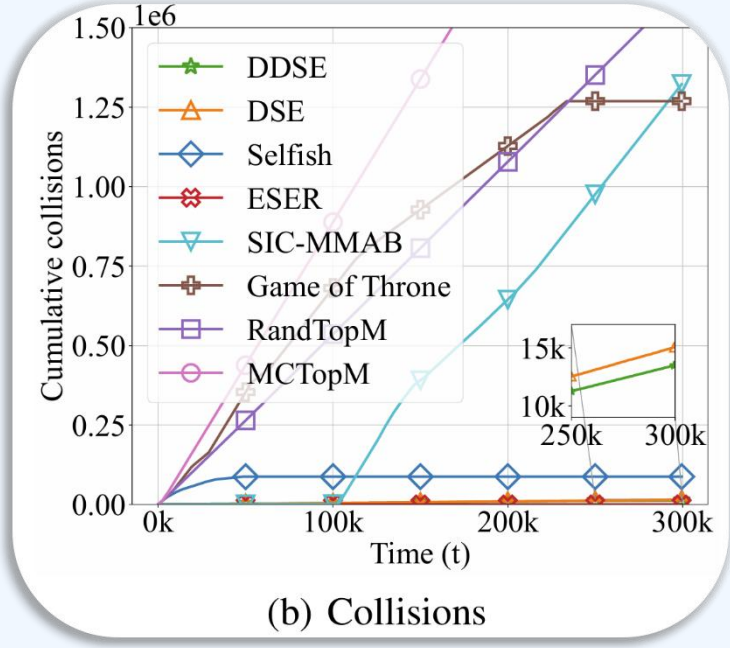
Test the cumulative throughtput and collisions.
Our algorithm: DDSE and DSE.

some of the data





(a) Throughput



(b) Collisions

My Projects

Decentralized Asynchronous Multi-player Bandits

Under review of NeurIPS-25

Jingqi Fan, Shuai Li, Siwei Wang*

Both the theoretical proof and experiments were completed during my junior year, as part of my internship at MSR Asia.

Problem formulation:

- Let $1 \leq T^j_{\text{start}} < T^j_{\text{end}} \leq T$. A player is active at step t means that she needs to pull an arm at this step. Let m_t denote the number of active players at step t.
- Each player $j \in [M]$ is only active from T^j_{start} to T^j_{end}.
- Player j is only aware of T, but does not know T^j_{start} and T^j_{end}.
- At each step $t \in [T^j_{\text{start}}, T^j_{\text{end}}]$, player j pulls an arm $\pi^j(t) \in [K]$ and observes $< r^j(t), \eta^j(t) >$.

Regret Definition:

$$\mathbb{E}[R(T)] := \sum_{t \leq T} \sum_{k \leq m_t} \mu_k - \mathbb{E}\left[\sum_{t \leq T} \sum_{j: T^j_{\text{start}} \leq t \leq T^j_{\text{end}}} r^j(t) \right].$$

Assumption:

- There exists a constant m such that for any t, $m_t \leq m \leq K/2$.

My Projects

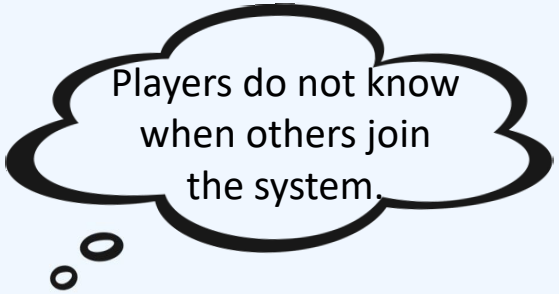
Decentralized Asynchronous Multi-player Bandits

Under review of NeurIPS-25

Jingqi Fan, Shuai Li, Siwei Wang*

Both the theoretical proof and experiments were completed during my junior year, as part of my internship at MSR Asia.

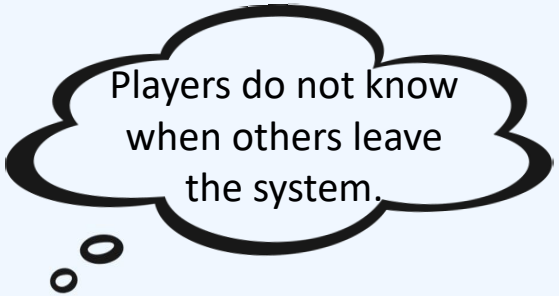
Challenge 1



Previous Init or Com phase does not work. However, I do not want to assume a known *j* here. And the assumption is not reasonable in the setting.

Thus, it is difficult to avoid collisions.

Challenge 2



The optimal arms depend on the number of active players. It can change.

When a player who is exploiting her optimal arm leaves the system, the left arms that are still exploited by players may become sub-optimal.

My Projects

Decentralized Asynchronous Multi-player Bandits

Under review of NeurIPS-25

Jingqi Fan, Shuai Li, Siwei Wang*

Both the theoretical proof and experiments were completed during my junior year, as part of my internship at MSR Asia.

Challenge 1:

difficult to avoid collisions

Solution 1:

- There is no `Init` or `Com` phase; each player independently executes her own policy.
- Player j maintains a set $\mathcal{A}^j$, representing the arms believed to be occupied by other players.
- Player j explores arms in $[K] \setminus \mathcal{A}^j$ uniformly at random.
- If arms in $[K] \setminus \mathcal{A}^j$ frequently result in collisions, player j infers that those arms are likely being exploited by others and adds them to $\mathcal{A}^j$.

Challenge 2:

change of optimal arms

Solution 2:

- Player j always pulls arms in $\mathcal{A}^j$ with a small probability ε.
- If arms in $\mathcal{A}^j$ frequently result in non-collisions, player j infers that those arms are likely being released by others and removes them from $\mathcal{A}^j$.

Player j adaptively changes between an exploration phase and an exploitation phase:

- **Exploration phase:** If there exists an arm k such that $\mathrm{LCB}_k^j \geq \mathrm{UCB}_\ell^j$ for all $\ell \neq k$, $\ell \in [K] \setminus \mathcal{A}^j$, then player j transitions to the exploitation phase and pulls arm k with probability $1 - \varepsilon$.
- **Exploitation phase:** If player j detects that an arm in $\mathcal{A}^j$ has been released, she switches back to the exploration phase.

My Projects

Decentralized Asynchronous Multi-player Bandits

Under review of NeurIPS-25

Jingqi Fan, Shuai Li, Siwei Wang*

Both the theoretical proof and experiments were completed during my junior year, as part of my internship at MSR Asia.

Challenge 1:

difficult to avoid collisions

Solution 1:

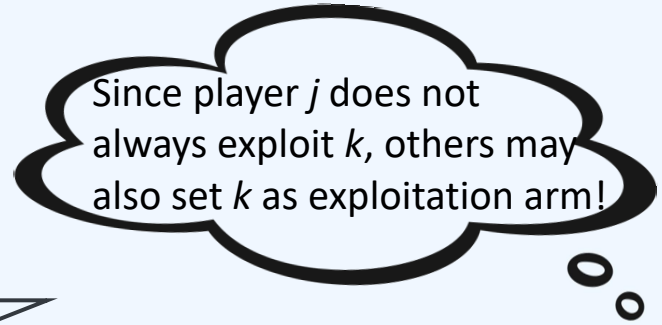
- There is no `Init` or `Com` phase; each player independently executes her own policy.
- Player j maintains a set $\mathcal{A}^j$, representing the arms believed to be occupied by other players.
- Player j explores arms in $[K] \setminus \mathcal{A}^j$ uniformly at random.
- If arms in $[K] \setminus \mathcal{A}^j$ frequently result in collisions, player j infers that those arms are likely being exploited by others and adds them to $\mathcal{A}^j$.

Challenge 2:

change of optimal arms

Solution 2:

- Player j always pulls arms in $\mathcal{A}^j$ with a small probability ε.
- If arms in $\mathcal{A}^j$ frequently result in non-collisions, player j infers that those arms are likely being released by others and removes them from $\mathcal{A}^j$.



Algorithmic Framework

Player j adaptively changes between an exploration phase and an exploitation phase:

- **Exploration phase:** If there exists an arm k such that $\mathrm{LCB}_k^j \geq \mathrm{UCB}_\ell^j$ for all $\ell \neq k$, $\ell \in [K] \setminus \mathcal{A}^j$, then player j transitions to the exploitation phase and pulls arm k with probability $1 - \varepsilon$.
- **Exploitation phase:** If player j detects that an arm in $\mathcal{A}^j$ has been released, she switches back to the exploration phase.

My Projects

Decentralized Asynchronous Multi-player Bandits

Under review of NeurIPS-25

Jingqi Fan, Shuai Li, Siwei Wang*

Both the theoretical proof and experiments were completed during my junior year, as part of my internship at MSR Asia.

DoubleSelection

- **Exploration phase:** player j samples $k \sim \mathrm{Uniform}([K] \setminus \mathcal{A}^j)$.
  - w.p. $1-\varepsilon$: pulls k twice;
  - w.p. ε: pull arm k once, then pull an arm $k' \sim \mathrm{Uniform}(\mathcal{A}^j)$.
- **Exploitation phase:** let $\hat{k}^j$ denote player j's exploitation arm.
  - w.p. $1-\varepsilon$: pulls $\hat{k}^j$ twice;
  - w.p. ε: pulls $\hat{k}^j$ once, then pull an arm $k' \sim \mathrm{Uniform}(\mathcal{A}^j)$.

Therefore, when a player want to enter the exploitation phase, she needs to find an arm k satisfying:

- **Condition 1:** $\eta_{k_1}(t-1) + \eta_{k_2}(t) = 0$, where $k_1 = k_2 = k$;
- **Condition 2:** $\mathrm{LCB}_k^j \geq \mathrm{UCB}_\ell^j$ for all $\ell \neq k$, $\ell \in [K] \setminus \mathcal{A}^j$.

My Projects

Decentralized Asynchronous Multi-player Bandits

Under review of NeurIPS-25

Jingqi Fan, Shuai Li, Siwei Wang*

Both the theoretical proof and experiments were completed during my junior year, as part of my internship at MSR Asia.

Let $\mathcal{P}_k^j, \mathcal{Q}_k^j$ denote two queues with fixed length $L_p = 866 \ln T$ and $L_q = 570 \ln T$, respectively.

Let T_o^j, T_r^j denote the numbers of time steps that are required for player j to identify an occupied arm k and a released arm k, respectively.

To solve Challenge 1

- At step t, if $k1 = k_2$ and they are both sampled from $[K] \setminus \mathcal{A}^j$, then player j adds $[\eta_{k_1}(t-1) \cdot \eta_{k_2}(t)]$ into a queue $\mathcal{P}_k^j$.
- If there exists an arm k s.t. $\sum_{i \in \mathcal{P}_k^j} i \geq \lceil 0.85 L_p \rceil$, then player j adds k to $\mathcal{A}^j$.

Lemma 1.
With probability at least $1 - 1/T^2$:
i) If arm k is occupied and remains occupied thereafter, player j will add k to $\mathcal{A}^j(t)$ with $E[T_o^j] \leq 1926 K ln T$ time steps;
ii) If arm k is not occupied and remains not occupied thereafter, player j will not add k to $\mathcal{A}^j(t)$.

To solve Challenge 2

- At step t, if k is sampled from $\mathcal{A}^j$, then player j adds $[1 - \eta_k(t)]$ into a queue $\mathcal{Q}_k^j$.
- If there exists an arm k s.t. $\sum_{i \in \mathcal{Q}_k^j} i \geq \lceil 0.142 L_q \rceil$, then player j removes k from $\mathcal{A}^j$.

Lemma 2.
With probability at least $1 - 1/T^2$:
i) If arm k is released and never occupied again, player j will remove k from $\mathcal{A}^j(t)$ with $E[T_r^j] \leq 1141 m ln T / \varepsilon$ time steps;
ii) If arm k is not released and remains not released thereafter, player j will not remove k from $\mathcal{A}^j(t)$.

My Projects

Decentralized Asynchronous Multi-player Bandits

Under review of NeurIPS-25

Jingqi Fan, Shuai Li, Siwei Wang*

Both the theoretical proof and experiments were completed during my junior year, as part of my internship at MSR Asia.

Theorem 1.

Given K arms and M players, and let $\varepsilon = \min\{\sqrt{\frac{1141m^3\ln(T)}{2T}}, \frac{1}{K}, \frac{1}{10}\}$, the regret of Algorithm 1 is bounded by

$$\mathbb{E}[R(T)] \leq \frac{576emKM\log(T)}{\Delta^2} + 96m^{3/2}M\sqrt{T\ln(T)} + 7704m^2KM\ln(T) + (4emKM)^2 ,$$

where $\Delta := \min_{k\leq m}(\mu_k - \mu_{k+1})$.

$\mathcal{O}(\log T/\Delta^2)$ **arises from Challenge 1:**
Players cannot completely avoid collisions, leading to a regret of $\mathcal{O}(\log T/\Delta^2)$ instead of the standard $\mathcal{O}(\log T/\Delta)$.
$\mathcal{O}(\sqrt{T\log T})$ **incurs from Challenge 2:**
The set of optimal arms may change over time, so players must pull occupied arms with a small probability. This persistent exploration contributes a regret of $\mathcal{O}(\sqrt{T\log T})$.

Corollory 1.

Given K arms and M players, $\varepsilon = \min\{\sqrt{\frac{1141K^3\ln(T)}{16T}}, \frac{1}{K}, \frac{1}{10}\}$, the regret of Algorithm 1 is bounded by

$$R(T) \leq \frac{288eK^2M\log(T)}{\Delta^2} + 34K^{3/2}M\sqrt{T\ln(T)} + 1926K^3M\ln(T) + (3eK^2M)^2 ,$$

My Projects

Decentralized Asynchronous Multi-player Bandits

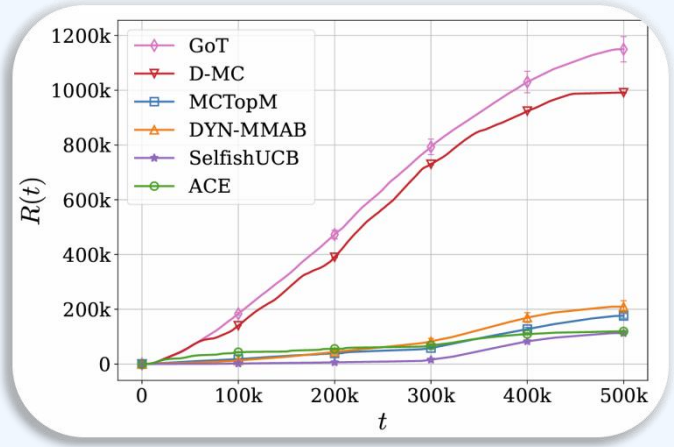
Under review of NeurIPS-25

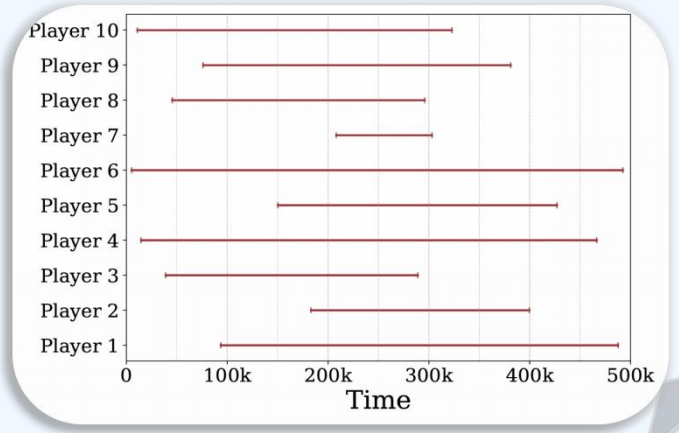
Jingqi Fan, Shuai Li, Siwei Wang*

Both the theoretical proof and experiments were completed during my junior year, as part of my internship at MSR Asia.

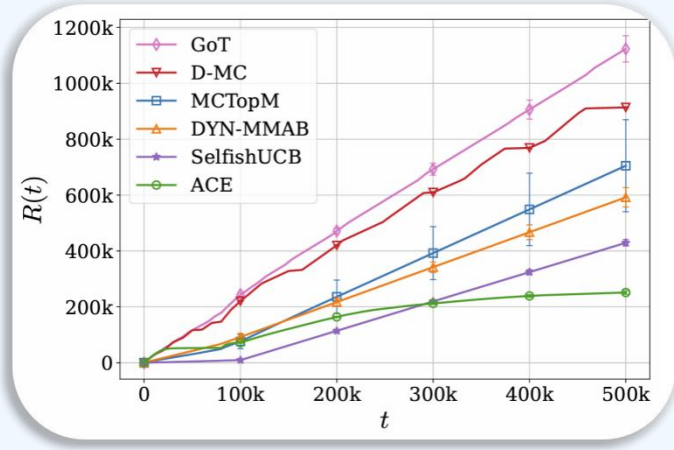
Numerical Experiments

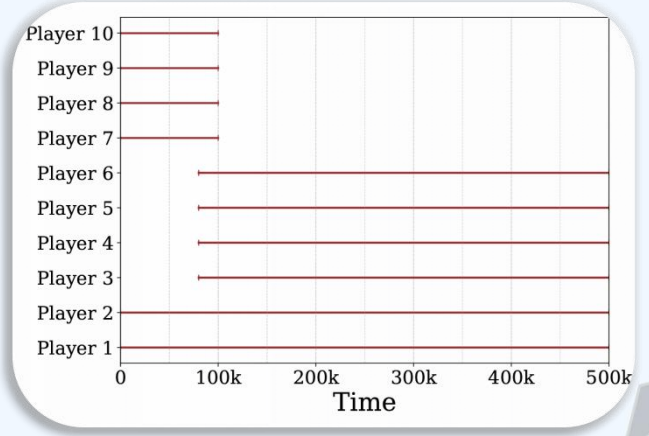
Random asynchronization





Snythetic asynchronization





Our algorithm: ACE

My Projects

Decentralized Asynchronous Multi-player Bandits

Under review of NeurIPS-25

Jingqi Fan, Shuai Li, Siwei Wang*

Both the theoretical proof and experiments were completed during my junior year, as part of my internship at MSR Asia.

Real-world Experiment Setup

Existence of Primary Users (PUs):
When a PU selects a channel, Secondary Users (SUs) are not allowed to access it (interpreted as a collision).

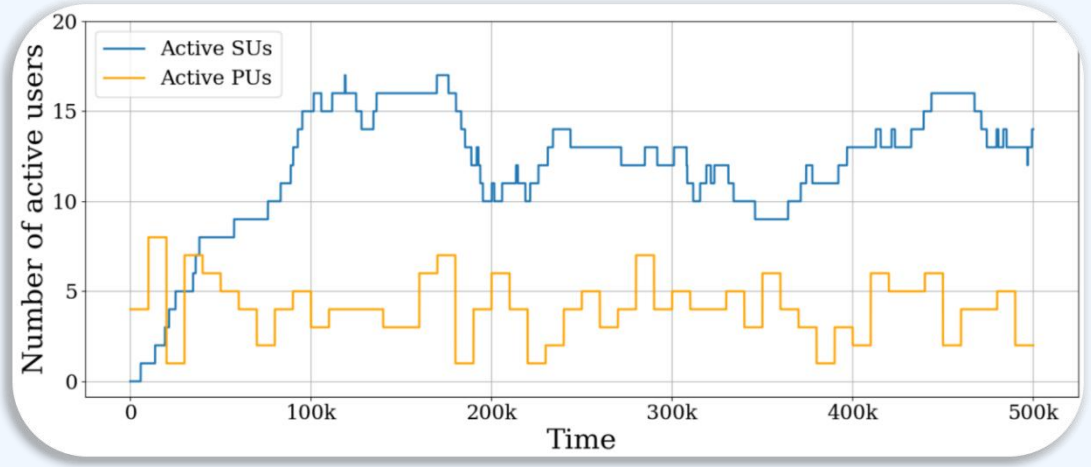
Heterogeneous and perturbed Rewards:

- Each player has different reward distributions.
- Gaussian-Markov mobility model introduces variations in channel quality over time, causing reward fluctuations.

Asynchronous SU Settings:

- Arrival times: Modeled by a Poisson process.
- Stay duration: Exponentially distributed with a lower bound (clipped).

Concurrency limit:
At most 20 SUs can be active at the same time.



My Projects

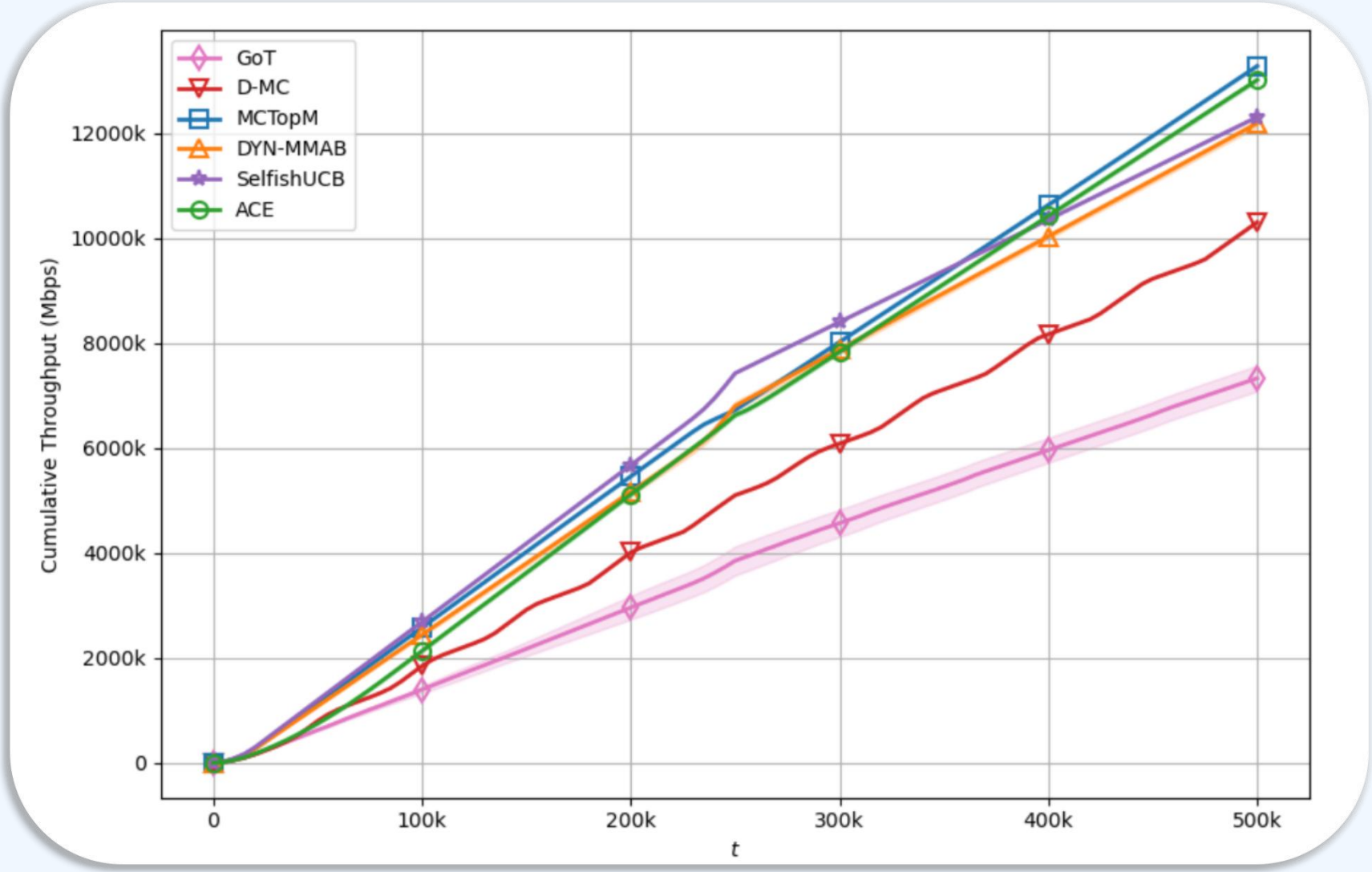
Decentralized Asynchronous Multi-player Bandits

Under review of NeurIPS-25

Jingqi Fan, Shuai Li, Siwei Wang*

Both the theoretical proof and experiments were completed during my junior year, as part of my internship at MSR Asia.

Real-world Experiment Result



Summary

2024 - 2025

The feedback received by players is asynchronous:

Multi-player Multi-armed Bandits with Delayed Feedback

Accepted by IJCAI-25

Jingqi Fan, Zilong Wang, Shuai Li*, Linghe Kong

2025

Players themselves are asynchronous:

Decentralized Asynchronous Multi-player Bandits

Under review of NeurIPS-25

Jingqi Fan, Shuai Li, Siwei Wang*

Asynchrony

asynchrony in decentralized MP-MAB, motivated by the asynchronous nature of real-world networks.