**Asympt. Upper bound**

MK

**More realistic dynamic model: first logarithmic regret algorithm**

Observation: coll. indicator \( \eta_k(t) \in \{0,1\} \) → bit sent from one player to another

Communication possible between players

**Algorithm 1: SIC-MMAB**

**Initialization phase:** estimate \( M \) and player rank \( j \) in time \( c(K \log T) \) for \( p = 1, ..., \infty \) do

**Exploration phase:** explore each active arm \( 2^p \) times without collision

**Communication phase:** players exchange statistics of arms using collisions

**Eliminate suboptimal arms**

**Attribute optimal arms to players (who enter exploitation phase)**

**End**

**Exploitation phase:** pull attributed arm until \( T \)

**Communication protocol**

When player \( i \) sends stats of arm \( k \) to player \( j \):

- Pull arm \( j \) to send 1 bit (collision)
- Pull arm \( i \) to send 0 bit (no collision)
- Send quantized empirical mean in binary

For \( 2^p \) exploration rounds, communicate in \( p \rightarrow \) sublogarithmic comm. regret

**Regret of SIC-MMAB**

\[
 \mathbb{E}[R_T] \leq \sum_{k>M} \log(T) \frac{MK \log(T)}{M^2 \mu_k} + MK \log(T) \frac{MK \log(T)}{M^2 \mu_k} + MK \log(T) \frac{MK \log(T)}{M^2 \mu_k}
\]

**Contradiction with the lower bound**

Centralized lower bound

\[
 \sum_{k>M} \log(T) \frac{MK \log(T)}{M^2 \mu_k}
\]

Decentralized lower bound

\[
 M \sum_{k>M} \log(T) \frac{MK \log(T)}{M^2 \mu_k}
\]

SIC-MMAB: static with collision sensing

Toward a realistic model

Unrealistic communication protocols → loophole in current model
Which assumption did go wrong?

- Collision sensing: No! still send a bit in \( \log\frac{T}{\Delta} \) rounds without it
- Synchronisation: Yes! Communication possible because players know when to talk to each other

Synchronisation: unrealistic assumption leading to hacking algorithms
Let’s remove it → dynamic model

**Algorithm 2: DYN-MMAB**

**Exploration phase:** pull \( k \sim \mathcal{U}(M) \)

Update occupied arms and queue of arms to exploit

if \( r_j(t) > 0 \) and \( k = \text{arm to exploit} \) then

[Enter exploitation phase]

end

**Exploitation phase:** pull exploited arm until \( T \)

**How does it work?**

Do not estimate \( \mu_k \) but \( \gamma \mu_k \) where \( \gamma \gg \mathcal{P} \) (no collision)

Still distinguish optimal from suboptimal arms

What if another player exploits \( k \)?

- either \( r_j \) → 0 quickly and arm \( k \) becomes suboptimal
- or too many \( \gamma \)s in a row \( \implies k \) is occupied

**Regret of DYN-MMAB**

\[
 \mathbb{E}[R_T] \leq \frac{MK \log(T)}{M^2 \mu_k} + \frac{MK \log(T)}{M^2 \mu_k} + \frac{MK \log(T)}{M^2 \mu_k}
\]

Summary

- Synchronisation → communication through collisions in MMAB
- Contradicts previous lower bounds
- Synchronisation is unrealistic and a loophole in the model
- More realistic dynamic model: first logarithmic regret algorithm

**Multiplier bandits model**

| Motivation | cognitive radio networks |
| Framework | \( K \) arms, \( M \leq K \) players. At each \( t \in T \), player \( j \) pulls \( \pi(t) \) |
| Reward | \( \pi(t) := \mathbb{X}_{\pi(t)}(1 - \delta_{\pi(t)}) \) |
| Decentralized | no information exchange between players |
| Observations | \( \mathbb{X}(t) \in \{0,1\} \text{ i.i.d. with } \mathbb{E}[\mathbb{X}(t)] = \mu_k \) |
| Synchro models | Static: any player plays for \( t = 1, \ldots, T \) Dynamic: player \( j \) plays for \( t = 1 + \tau_j, \ldots, T \) |
| Goal | with \( \mathbb{M}(t) \) set of players at time \( t \), minimize regret |
| \( R_T := \sum_{t=1}^{T} \sum_{j=1}^{M} \mu_k(t) - \mathbb{X}_j(t) \sum_{j=1}^{M} \pi(t) \) | |
| Notations | \( \mu_1 \geq \cdots \geq \mu_M \) and \( \Delta_M := \min_{k=1,\ldots,M} (\mu_k - \mu_{k+1}) \) |

**State of the art bounds**

<table>
<thead>
<tr>
<th>Setting</th>
<th>Prior knowledge</th>
<th>Asympt. Upper bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>Centralized Multiplayer</td>
<td>( M )</td>
<td>( \sum_{k&gt;M} \log(T) \frac{MK \log(T)}{M^2 \mu_k} ) [3]</td>
</tr>
<tr>
<td>Decentralized, Col. Sensing</td>
<td>( \mu_k^{-1} \Delta_{k(M+1)} )</td>
<td>( \sum_{k&gt;M} \log(T) \frac{MK \log(T)}{M^2 \mu_k} ) [4]</td>
</tr>
<tr>
<td>Decentralized, Coll. Sensing</td>
<td>( \mu_k^{-1} \Delta_{k(M+1)} )</td>
<td>( \sum_{k&gt;M} \log(T) \frac{MK \log(T)}{M^2 \mu_k} ) [4]</td>
</tr>
<tr>
<td>Dec., Coll. Sensing, Dynamic</td>
<td>( \Delta_{k(M+1)} )</td>
<td>( \sum_{k&gt;M} \log(T) \frac{MK \log(T)}{M^2 \mu_k} ) [4]</td>
</tr>
<tr>
<td>Dec., No Sensing, Dynamic</td>
<td>( \Delta_{k(M+1)} )</td>
<td>( \sum_{k&gt;M} \log(T) \frac{MK \log(T)}{M^2 \mu_k} ) [4]</td>
</tr>
</tbody>
</table>

Our results in red.

**References**


