

## INTRODUCTION

### Model Setting and Notations

- $K$ : Number of arms.
- $\delta$ : Tolerance level of wrong identification
- $\nu = \{\nu_a\}_{a=1}^K$ : Reward distribution of arm  $a$
- $\mu_a$ : Mean reward of arm  $a \in [K]$   
 $\Delta_{i,j} = |\mu_i - \mu_j|$ , WLOG,  $\mu_1 \geq \mu_2 \geq \dots \geq \mu_K$
- $\mu_0$ : Known threshold for comparison
- $\mathcal{S}^{\text{pos}} = \{\nu : \mu_1 > \mu_0\}$ ,  $\mathcal{S}^{\text{neg}} = \{\nu : \mu_1 < \mu_0\}$ , For  $\Delta > 0$ ,  
 $\mathcal{S}_{\Delta}^{\text{pos}} = \{\nu : \mu_1 - \mu_0 \geq \Delta\}$ ,  $\mathcal{S}_{\Delta}^{\text{neg}} = \{\nu : \mu_0 - \mu_1 \geq \Delta\}$ .
- $A_t$ : Action in round  $t$
- $X_t$ : Observed reward in round  $t$
- $H(t) = \{(A_s, X_s)\}_{s=1}^t$ : History collected **up to time  $t$** .

#### Known Information

- $K, \mu_0, \delta$

#### Unknown Information

- $\nu_a, \mu_a$ , for all  $a \in [K]$

### Dynamics and Model Uncertainty

At each round  $t = 1, 2, \dots$ , the Decision Maker

- 1 Pull an arm  $A_t \in [K]$ ,  $A_t$  is  $\sigma(H(t-1))$ -measurable,
- 2 Receive the outcome  $X_t \sim \nu_{A_t}$ ,
- 3 Update the history  $H(t) = H(t-1) \cup \{(A_t, X_t)\}$

The agent stops at the end of time step  $\tau$ ,

$\tau$  is a **stopping time** with respect to the filtration  $\{\sigma(H(t))\}_{t=1}^{\infty}$

Upon stopping, the agent **outputs arm  $\hat{a} \in [K] \cup \{\text{None}\}$**

### Definition of PAC Requirement

$i^*(\nu)$ :

- $i^*(\nu) = \{a : \mu_a \geq \mu_0\}$ , for  $\nu \in \mathcal{S}^{\text{pos}}$
- $i^*(\nu) = \{\text{None}\}$ , for  $\nu \in \mathcal{S}^{\text{neg}}$

$\delta$ -PAC:

- A pulling strategy is  $\delta$ -PAC, if for any  $\delta \in (0, 1)$ ,  
 $\nu \in \mathcal{S}^{\text{pos}} \cup \mathcal{S}^{\text{neg}}$ , it satisfies  $\Pr_{\nu}(\tau < +\infty, \hat{a} \in i^*(\nu)) > 1 - \delta$ .

$(\Delta, \delta)$ -PAC:

- A pulling strategy is  $(\Delta, \delta)$ -PAC, if it is  $\delta$ -PAC, and for any  $\Delta, \delta > 0$ , we have  $\sup_{\nu \in \mathcal{S}_{\Delta}^{\text{pos}} \cup \mathcal{S}_{\Delta}^{\text{neg}}} \mathbb{E}_{\nu} \tau < +\infty$ .

### Objective: Minimize $\mathbb{E}\tau$

The agent aims to design a  $(\Delta, \delta)$ -PAC pulling strategy  $(\pi, \tau, \hat{a})$  that **minimizes the *sampling complexity*  $\mathbb{E}[\tau]$** .

### Lit Review

Algorithm	Bound	Opt in pos	Opt in Neg
S-TaS	$\lim_{\delta \rightarrow 0} \frac{\mathbb{E}\tau}{\log \frac{1}{\delta}} = \begin{cases} H & \text{pos} \\ H_1^{\text{neg}} & \text{neg} \end{cases}$	✓	✓
HDoC	$\mathbb{E}\tau \leq \begin{cases} O(H \log \frac{K}{\delta} + H_1 \log \log \frac{1}{\delta} + \frac{K}{\epsilon^2}) & \text{pos} \\ O(H_1^{\text{neg}} \log \frac{K}{\delta} + \frac{K}{\epsilon^2}) & \text{neg} \end{cases}$	×	×
APGAI	$\mathbb{E}\tau \leq \begin{cases} O(H_0(\log \frac{K}{\delta})) & \text{pos} \\ O(H_1^{\text{neg}}(\log \frac{K}{\delta})) & \text{neg} \end{cases}$	×	✓
SEE (*)	$\mathbb{E}\tau \leq \begin{cases} O(H \log \frac{1}{\delta}) + O(H_1^{\text{pos}} \log \frac{K}{\Delta_{0,1}}) & \text{pos} \\ O(H_1^{\text{neg}}(\log \frac{1}{\delta} + \log H_1^{\text{neg}})) & \text{neg} \end{cases}$	✓	✓
Lower Bound (*)	$\mathbb{E}\tau \geq \begin{cases} O(H \log \frac{1}{\delta} + \frac{1}{m} H_1^{\text{low}} - \frac{1}{\Delta_{i,m+1}^2}) & \text{pos} \\ \Omega(H_1^{\text{neg}} \log \frac{1}{\delta}) & \text{neg} \end{cases}$	NA	NA

(\*) denotes the result is from this paper

Extra Comments

- S-TaS (Degenne & Koolen 2019): only achieves asymptotic optimality
- HDoC (Kano et.al 2018), APGAI (Jourdan et.al 2023) are **not**  $(\Delta, \delta)$ -PAC
  - Two-armed instance
  - Arm 1 Gaussian,  $\mu_1 > \mu_0$ , Arm 2 Constant,  $\mu_2 = \mu_0$
  - $\text{UCB}_1 < \mu_0$  holds with non-zero prob, then HDoC and APGAI will never stop
- SEE is nearly optimal
  - $\nu \in \mathcal{S}^{\text{neg}}$ , all the arms are below  $\mu_0$
  - Only one arm is above  $\mu_0$
- For SEE, If  $\nu \in \mathcal{S}^{\text{pos}}$ , coefficient of  $\log \frac{1}{\delta}$  is independent of arm number  $K$

## Algorithm

### Sequential Exploration Exploitation (Informal)

- 1: **procedure** SEE(Input: Action set  $[K]$ , threshold  $\mu_0$ , tolerance level  $\delta$ ,  $C > 1$ ).
- 2:   **Tune**  $\{\delta_k, T_k^{\text{et}}, T_k^{\text{ee}}\}_{k=1}^{+\infty}$ .
- 3:   **for** Phase  $k = 1, 2, \dots$  **do**
- 4:     (Exploration) Run algorithm LUCB\_G with tolerance level  $\delta_k$  and previous exploration history.  
Stops until one of the two conditions holds.
  - Total pulling times in all exploration phases is not greater than  $T_k^{\text{ee}}$ .
Take  $\hat{a}_k = \text{Not Complete}$ .
  - LUCB\_G stops and output  $\hat{a}_k \in [K] \cup \{\text{None}\}$ .
- 5:     **if**  $\hat{a}_k \in [K]$  **then**
- 6:       (Exploitation) Keep pulling arm  $\hat{a}_k$  with independent samples  
Stops until one of the two conditions holds.
  - Pulling times of  $\hat{a}_k$  is not smaller than  $T_k^{\text{et}}$ .
  - LCB defined by  $\delta$  is above  $\mu_0$ , output  $\hat{a}_k$  as a qualified arm
- 7:     **else if**  $\hat{a}_k = \text{None}$  and  $\delta_k < \frac{\delta}{3}$  **then**
- 8:       Output the instance is negative
- 9:     **end if**
- 10:   **end for**
- 11:   **return**  $x$
- 12: **end procedure**

### Notation on Complexity

$$H_1^{\text{neg}} = \sum_{a=1}^K \frac{2}{\Delta_{0,a}^2}, H_1^{\text{low}} = \sum_{a:\mu_a < \mu_0} \frac{2}{\Delta_{1,a}^2},$$

$$H_1^{\text{pos}} = \sum_{a=1}^K \frac{2}{\max\{\Delta_{0,a}^2, \Delta_{1,a}^2\}}, H = \frac{2}{\Delta_{0,1}^2}$$

$$H_1 = \sum_{a=2}^K \frac{2}{\Delta_{1,a}^2}, H_0 = \sum_{a:\mu_a \geq \mu_0} \frac{2}{\Delta_{0,a}^2},$$

$$H_1^{\text{BAI}} = \frac{2}{\Delta_{0,1}^2} + \sum_{a=2}^K \frac{2}{\Delta_{1,a}^2}.$$

### Key Ideas Behind the Algorithm

Property of LUCB\_G

- LUCB\_G is an adapted BAI alg
  - Pull arms with highest UCB
  - Return an arm whose LCB is greater than  $\mu_0$
- Take  $\delta' \in (0, 1)$  as input for LUCB\_G.  
Conditioned on event holds with prob  $1 - O(\delta')$ , LUCB\_G
  - return  $\mu_a > \mu_0$  with pulling times  $O(H_1^{\text{pos}}(\log \frac{1}{\delta'} + \log H_1^{\text{pos}}))$
  - return **None**, with pulling times  $O(H_1^{\text{neg}}(\log \frac{1}{\delta'} + \log H_1^{\text{neg}}))$
- Take  $\delta' \in (0, 1)$  as input for LUCB.  
If the event doesn't hold, LUCB might never stop

Conduct Exploration

- Call LUCB\_G for exploration,  
decreasing  $\{\delta_k\}_{k=1}^{+\infty}$  as tolerance level for each phase
- Since it might get stuck in a non-stopping loop,  
**set up maximum pulling tiems  $T_k^{\text{ee}}$**

Conduct Exploitation, if  $\hat{a}_k \in [K]$

- **Keep pulling  $\hat{a}_k$  with new samples**
- Output  $\hat{a}_k$  if its LCB( $\delta$ )  $> \mu_0$

Accept  $\hat{a}_k = \text{None}$

- Only when  $\delta_k = \Theta(\delta)$
- Inspired by Degenne & Koolen 2019, negative instance is similar to Best Arm Identification

### Numeric Settings

Benchmark Algorithms

- Adapted Murphy Sampling (Kaufmann et.al 2018)
- Adapted Track and Stop (Garivier & Kaufmann 2016)
- HDoC, LUCB\_G (Kano et.al 2018)
- lilHDoC (Tsai et.al 2024)

Instance Setting

- All the mean rewards are below  $\mu_0$
- Fraction of Qualified Arms (100%, 50%, 25%, Unique)
- Linear reward vector

## Numeric Experiments

### Numeric Experiments

