Best Arm Identification with Limited Precision Sampling

P. N. Karthik

Institute of Data Science
National University of Singapore
March 23, 2023

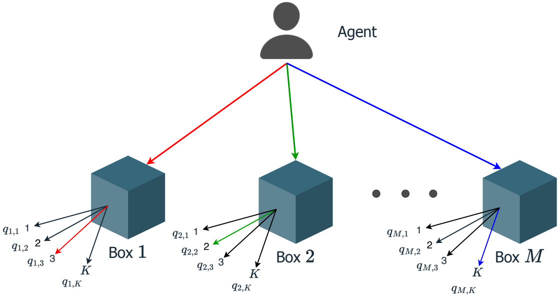
Joint Work With

Kota Srinivas Reddy
IIT Chennai

Nikhil Karamchandani
IIT Mumbai

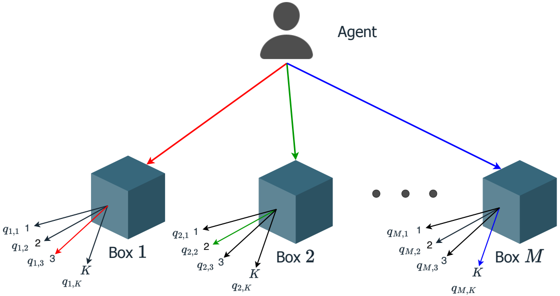
Jayakrishnan Nair
IIT Mumbai

Problem Setup



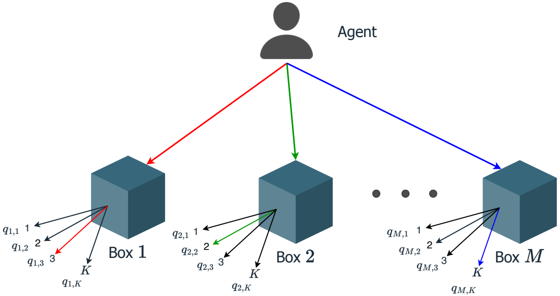
- M boxes, K arms per box

Problem Setup



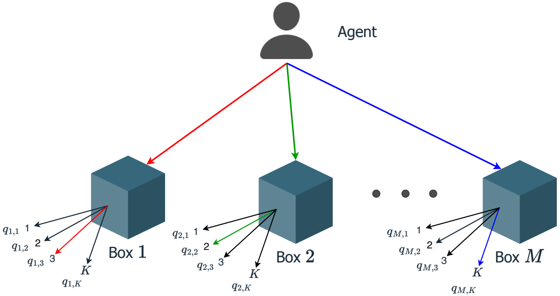
- M boxes, K arms per box
- Agent can only pick boxes

Problem Setup



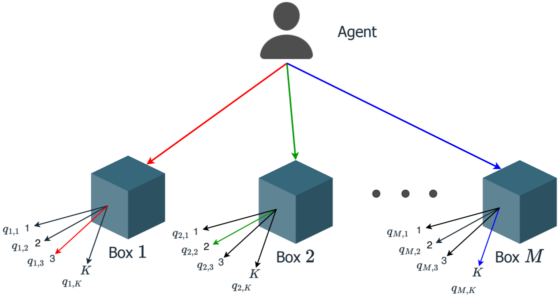
- M boxes, K arms per box
- Agent can only pick boxes
- $P(\text{arm} = k | \text{box} = m) = q_{m,k}$

Problem Setup



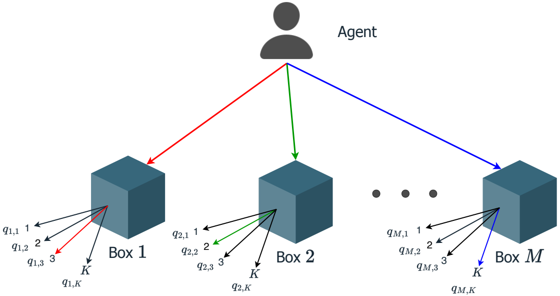
- M boxes, K arms per box
- Agent can only pick boxes
- $P(\text{arm} = k | \text{box} = m) = q_{m,k}$
- Instance: $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_K)$

Problem Setup



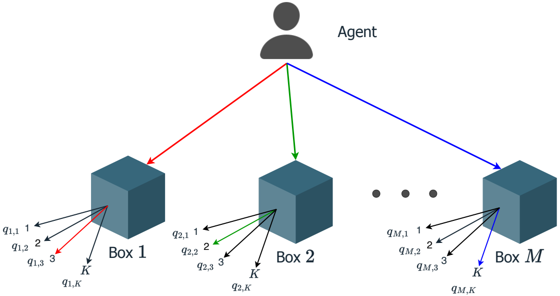
- M boxes, K arms per box
- Agent can only pick boxes
- $P(\text{arm} = k | \text{box} = m) = q_{m,k}$
- Instance: $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_K)$
- $\nu_k = \mathcal{N}(\mu_k, 1), \; k \in [K]$

Problem Setup



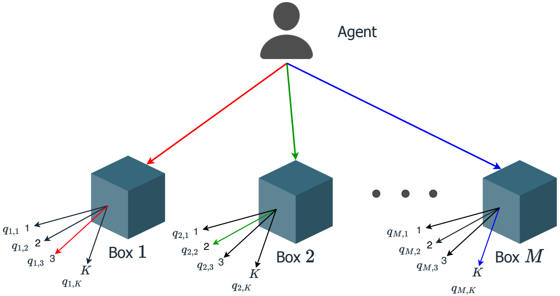
- M boxes, K arms per box
- Agent can only pick boxes
- $P(\text{arm} = k | \text{box} = m) = q_{m,k}$
- Instance: $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_K)$
- $\nu_k = \mathcal{N}(\mu_k, 1), \; k \in [K]$
- Unknowns:

Problem Setup



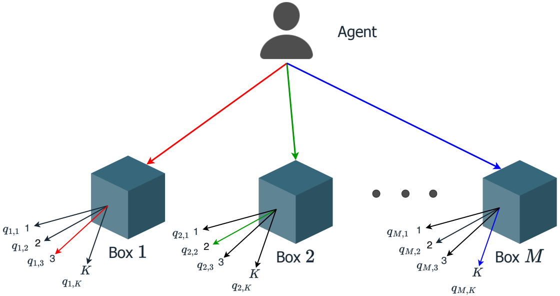
- M boxes, K arms per box
- Agent can only pick boxes
- $P(\text{arm} = k | \text{box} = m) = q_{m,k}$
- Instance: $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_K)$
- $\nu_k = \mathcal{N}(\mu_k, 1), \; k \in [K]$
- Unknowns:
  - $\boldsymbol{q} = \{q_{m,k}\}_{m,k}$

Problem Setup



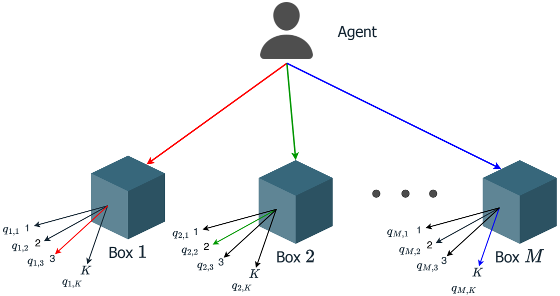
- M boxes, K arms per box
- Agent can only pick boxes
- $P(\text{arm} = k|\text{box} = m) = q_{m,k}$
- Instance: $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_K)$
- $\nu_k = \mathcal{N}(\mu_k, 1), \;\; k \in [K]$
- Unknowns:
  - $\boldsymbol{q} = \{q_{m,k}\}_{m,k}$
  - $\boldsymbol{\mu} = \{\mu_k\}_k$

Problem Setup



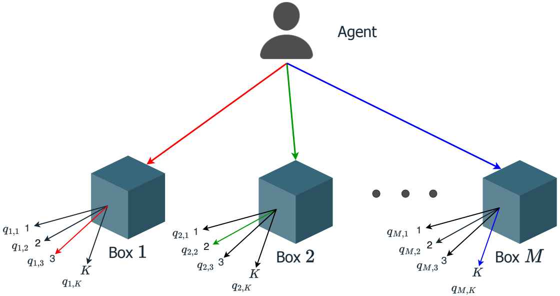
- M boxes, K arms per box
- Agent can only pick boxes
- $P(\text{arm} = k | \text{box} = m) = q_{m,k}$
- Instance: $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_K)$
- $\nu_k = \mathcal{N}(\mu_k, 1), \; k \in [K]$
- Unknowns:
  - $\boldsymbol{q} = \{q_{m,k}\}_{m,k}$
  - $\boldsymbol{\mu} = \{\mu_k\}_k$
- Best arm: $k^* = \arg\max_k \mu_k$

Problem Setup



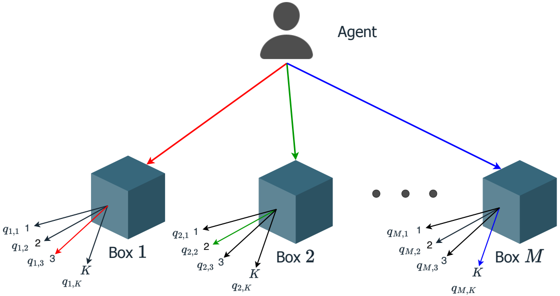
- M boxes, K arms per box
- Agent can only pick boxes
- $P(\text{arm} = k | \text{box} = m) = q_{m,k}$
- Instance: $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_K)$
- $\nu_k = \mathcal{N}(\mu_k, 1), \ k \in [K]$
- Unknowns:
  - $\boldsymbol{q} = \{q_{m,k}\}_{m,k}$
  - $\boldsymbol{\mu} = \{\mu_k\}_k$
- Best arm: $k^* = \arg\max_k \mu_k$
- Goal: Fixed-confidence BAI

Problem Setup



- M boxes, K arms per box
- Agent can only pick boxes
- $P(\text{arm} = k | \text{box} = m) = q_{m,k}$
- Instance: $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_K)$
- $\nu_k = \mathcal{N}(\mu_k, 1), \;\; k \in [K]$
- Unknowns:
  - $\boldsymbol{q} = \{q_{m,k}\}_{m,k}$
  - $\boldsymbol{\mu} = \{\mu_k\}_k$
- Best arm: $k^* = \arg\max_k \mu_k$
- Goal: Fixed-confidence BAI

Problem Setup

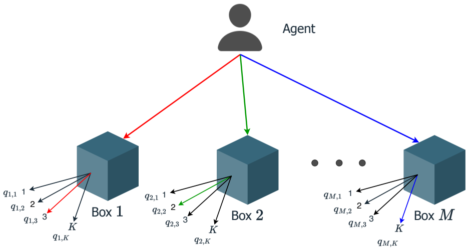


- M boxes, K arms per box
- Agent can only pick boxes
- $P(\text{arm} = k | \text{box} = m) = q_{m,k}$
- Instance: $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_K)$
- $\nu_k = \mathcal{N}(\mu_k, 1), \ k \in [K]$
- Unknowns:
  - $\boldsymbol{q} = \{q_{m,k}\}_{m,k}$
  - $\boldsymbol{\mu} = \{\mu_k\}_k$
- Best arm: $k^* = \arg\max_k \mu_k$
- Goal: Fixed-confidence BAI

The key: determining the optimal box weight(s)

Preliminaries

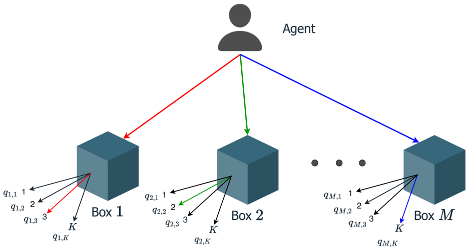
The key: determining the optimal box weight(s)



- Agent sees the pulled arm and its reward at each time

Preliminaries

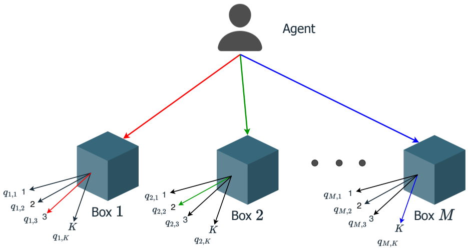
The key: determining the optimal box weight(s)



- Agent sees the pulled arm and its reward at each time
- SE-type or LUCB-type algorithms cannot be applied verbatim

PRELIMINARIES

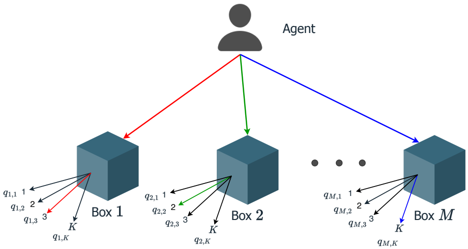
The key: determining the optimal box weight(s)



- Agent sees the pulled arm and its reward at each time
- SE-type or LUCB-type algorithms cannot be applied verbatim
- To pull a certain arm k:

Preliminaries

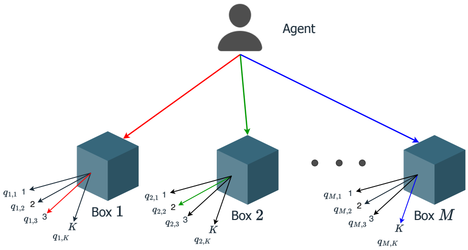
The key: determining the optimal box weight(s)



- Agent sees the pulled arm and its reward at each time
- SE-type or LUCB-type algorithms cannot be applied verbatim
- To pull a certain arm k:
  - If $\boldsymbol{q} = \{q_{m,k}\}_{m,k}$ is known, then choose box $m^* \in \arg\max_m q_{m,k}$

Preliminaries

The key: determining the optimal box weight(s)



- Agent sees the pulled arm and its reward at each time
- SE-type or LUCB-type algorithms cannot be applied verbatim
- To pull a certain arm k:
  - If $\boldsymbol{q} = \{q_{m,k}\}_{m,k}$ is known, then choose box $m^* \in \arg\max_m q_{m,k}$
  - If $\boldsymbol{q} = \{q_{m,k}\}_{m,k}$ is unknown, then?
    If $\{q_{m,k}\}_k = \{q_{m'k}\}_k$ for all m, m', then every box weight is optimal

Outline

Asymptotic Analysis

Converse

Fix a problem instance $\boldsymbol{q}_0 = \{q_{m,k}^0\}_{m,k}$, $\boldsymbol{\mu}_0 = \{\mu_k^0\}_k$

Converse

Fix a problem instance $\boldsymbol{q}_0 = \{q^0_{m,k}\}_{m,k}$, $\boldsymbol{\mu}_0 = \{\mu^0_k\}_k$

Theorem

$$\liminf_{\delta \downarrow 0} \inf_{\pi \ \delta\text{-}PC} \frac{\mathbb{E}[\tau_\pi]}{\log(1/\delta)} \geq \frac{1}{T^*(\boldsymbol{q}_0, \boldsymbol{\mu}_0)},$$

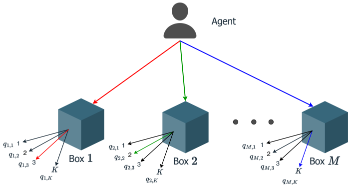
where $T^*(\boldsymbol{q}_0, \boldsymbol{\mu}_0)$ is given by

$$T^*(\boldsymbol{q}_0, \boldsymbol{\mu}_0) = \sup_{\mathbf{w} \in \Sigma_M} \inf_{\boldsymbol{\lambda} \in ALT(\boldsymbol{\mu}_0)} \sum_{m=1}^{M} \sum_{k=1}^{K} w_m \, q^0_{m,k} \, \frac{(\mu^0_k - \lambda_k)^2}{2}.$$

The supremum is over $\Sigma_M = \{\mathbf{w} = (w_1, \ldots, w_M) : \; w_m \geq 0 \;\; \forall m, \;\; \sum_{m=1}^{M} w_m = 1\}$.

- From transportation Lemma 1 of Kaufmann et al. [2016],

$$\pi\ \delta\text{-PC} \implies \inf_{\boldsymbol{\lambda} \in \mathsf{ALT}(\boldsymbol{\mu}_0)} \sum_{k=1}^{K} \underbrace{\mathbb{E}[N_k(\tau_\pi)]}_{\#\text{ pulls of arm } k} \frac{(\mu_k^0 - \lambda_k)^2}{2} \geq D_{\mathsf{KL}}(\mathsf{Ber}(\delta) \| \mathsf{Ber}(1-\delta)).$$



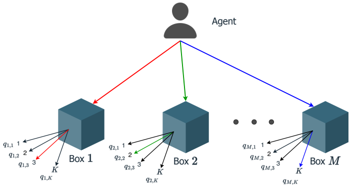
$$\boxed{T^*(\boldsymbol{q}_0, \boldsymbol{\mu}_0) = \sup_{\mathbf{w} \in \Sigma_M} \inf_{\boldsymbol{\lambda} \in \mathsf{ALT}(\boldsymbol{\mu}_0)} \sum_{m=1}^{M} \sum_{k=1}^{K} w_m\, q_{m,k}^0 \frac{(\mu_k^0 - \lambda_k)^2}{2}}$$

- From transportation Lemma 1 of Kaufmann et al. [2016],

$$\pi\ \delta\text{-PC} \implies \inf_{\boldsymbol{\lambda}\in\mathsf{ALT}(\boldsymbol{\mu}_0)} \sum_{k=1}^{K} \underbrace{\mathbb{E}[N_k(\tau_\pi)]}_{\#\text{ pulls of arm } k} \frac{(\mu_k^0-\lambda_k)^2}{2} \geq D_{\mathsf{KL}}(\mathsf{Ber}(\delta)\|\mathsf{Ber}(1-\delta)).$$

- For each $k\in[K]$,

$$\mathbb{E}[N_k(\tau_\pi)] = \sum_{m=1}^{M} q_{m,k}^0 \underbrace{\mathbb{E}[N(\tau_\pi,m)]}_{\#\text{ selections of box } m}.$$



$$\boxed{T^*(\boldsymbol{q}_0,\boldsymbol{\mu}_0) = \sup_{\mathbf{w}\in\Sigma_M} \inf_{\boldsymbol{\lambda}\in\mathsf{ALT}(\boldsymbol{\mu}_0)} \sum_{m=1}^{M}\sum_{k=1}^{K} w_m\, q_{m,k}^0 \frac{(\mu_k^0-\lambda_k)^2}{2}}$$

Non-Uniqueness of Optimal Box Weights

Example (1)

$\{q^0_{m,k}\}_k$ independent of m, i.e., $\{q^0_{m,k}\}_k = \{q^0_{m',k}\}_k = \{\alpha_k\}_k$ for all m, m'. In this case,

$$\sum_{m=1}^{M} w_m \, q^0_{m,k} = \sum_{m=1}^{M} w_m \, \alpha_k = \alpha_k \quad \forall k \in [K], \; \mathbf{w} \in \Sigma_M.$$

Example (2)

$M = 2$, $K = 4$, $\boldsymbol{\mu}_0 = \{0.5, 0.4, 0.3, 0.3\}$, $\boldsymbol{q}_0 = \begin{pmatrix} 0.3 & 0.3 & 0.3 & 0.1 \\ 0.3 & 0.3 & 0.1 & 0.3 \end{pmatrix}$

For the above examples, every $\mathbf{w} \in \Sigma_M$ is optimal

Achievability: Preliminaries

- Set of optimal box weights under $(\boldsymbol{q}, \boldsymbol{\mu})$:

$$\mathcal{W}^\star(\boldsymbol{q}, \boldsymbol{\mu}) = \arg\sup_{w \in \Sigma_M} \inf_{\boldsymbol{\lambda} \in \textsc{Alt}(\boldsymbol{\mu})} \sum_{m=1}^{M} \sum_{k=1}^{K} w_m \, q_{m,k} \, \frac{(\mu_k - \lambda_k)^2}{2}$$

Achievability: Preliminaries

- Set of optimal box weights under $(\boldsymbol{q}, \boldsymbol{\mu})$:

$$\mathcal{W}^\star(\boldsymbol{q}, \boldsymbol{\mu}) = \arg \sup_{w \in \Sigma_M} \inf_{\boldsymbol{\lambda} \in \textsc{Alt}(\boldsymbol{\mu})} \sum_{m=1}^{M} \sum_{k=1}^{K} w_m \, q_{m,k} \, \frac{(\mu_k - \lambda_k)^2}{2}$$

- $(\boldsymbol{q}, \boldsymbol{\mu}) \mapsto \mathcal{W}^\star(\boldsymbol{q}, \boldsymbol{\mu})$ is compact-valued and upper hemicontinuous

Achievability: Preliminaries

- Set of optimal box weights under $(\boldsymbol{q}, \boldsymbol{\mu})$:

$$\mathcal{W}^\star(\boldsymbol{q}, \boldsymbol{\mu}) = \arg \sup_{w \in \Sigma_M} \inf_{\boldsymbol{\lambda} \in \textsc{Alt}(\boldsymbol{\mu})} \sum_{m=1}^{M} \sum_{k=1}^{K} w_m \, q_{m,k} \, \frac{(\mu_k - \lambda_k)^2}{2}$$

- $(\boldsymbol{q}, \boldsymbol{\mu}) \mapsto \mathcal{W}^\star(\boldsymbol{q}, \boldsymbol{\mu})$ is compact-valued and upper hemicontinuous
- $\mathcal{W}^\star(\boldsymbol{q}, \boldsymbol{\mu})$ is convex for each $(\boldsymbol{q}, \boldsymbol{\mu})$

Achievability: Modified D-Tracking (1)

- Parameter estimates at time t:

$$\hat{q}_{m,k}(t) = \frac{\text{\# times box } m \text{ selected and arm } k \text{ pulled}}{N(t,m)}, \quad \hat{\mu}_k(t) = \frac{1}{N_k(t)} \sum_{s=1}^{t} \mathbf{1}_{\{A_s=k\}} X_s$$

Achievability: Modified D-Tracking (1)

- Parameter estimates at time t:

$$\hat{q}_{m,k}(t) = \frac{\text{\# times box } m \text{ selected and arm } k \text{ pulled}}{N(t,m)}, \qquad \hat{\mu}_k(t) = \frac{1}{N_k(t)} \sum_{s=1}^{t} \mathbf{1}_{\{A_s=k\}} X_s$$

- $f(t) = \frac{\sqrt{t}}{\sqrt{M}}$, $\quad N(0,m) = 0$ for all m

Achievability: Modified D-Tracking (1)

- Parameter estimates at time t:

$$\hat{q}_{m,k}(t) = \frac{\text{\# times box } m \text{ selected and arm } k \text{ pulled}}{N(t,m)}, \qquad \hat{\mu}_k(t) = \frac{1}{N_k(t)} \sum_{s=1}^{t} \mathbf{1}_{\{A_s = k\}} X_s$$

- $f(t) = \frac{\sqrt{t}}{\sqrt{M}}$, $\quad N(0,m) = 0$ for all m
- $\underbrace{i_0 = 0, \quad i_{t+1} = (i_t \bmod M) + \mathbf{1}_{\{\min_{m \in [M]} N(t,m) < f(t)\}} \quad \text{for all } t \geq 0}_{\text{round-robin box selection counter}}$

Achievability: Modified D-Tracking (1)

- Parameter estimates at time t:

$$\hat{q}_{m,k}(t) = \frac{\text{\# times box } m \text{ selected and arm } k \text{ pulled}}{N(t,m)}, \qquad \hat{\mu}_k(t) = \frac{1}{N_k(t)} \sum_{s=1}^{t} \mathbf{1}_{\{A_s = k\}} X_s$$

- $f(t) = \frac{\sqrt{t}}{\sqrt{M}}$, $\quad N(0,m) = 0$ for all m
- $\underbrace{i_0 = 0, \quad i_{t+1} = (i_t \bmod M) + \mathbf{1}_{\{\min_{m \in [M]} N(t,m) < f(t)\}} \quad \text{for all } t \geq 0}_{\text{round-robin box selection counter}}$
- Let $\{\mathbf{w}(t) : t \geq 1\}$ be such that $\mathbf{w}(t+1) \in \mathcal{W}^\star(\hat{\boldsymbol{q}}(t), \hat{\boldsymbol{\mu}}(t))$ for all t

Achievability: Modified D-Tracking (2)

- The modified D-Tracking rule:

$$B_{t+1} = \begin{cases} i_t, & \min_{m \in [M]} N(t,m) < f(t), \\ b_t, & \text{otherwise}, \end{cases}$$

where $\{b_t : t \geq 1\}$ is specified by

$$b_t = \arg \min_{m \in \text{supp}(\sum_{s=1}^{t} w(s))} N(t,m) - \sum_{s=1}^{t} w_m(s).$$

Achievability: Modified D-Tracking (2)

- The modified D-Tracking rule:

$$B_{t+1} = \begin{cases} i_t, & \min_{m \in [M]} N(t,m) < f(t), \\ b_t, & \text{otherwise}, \end{cases}$$

where $\{b_t : t \geq 1\}$ is specified by

$$b_t = \arg \min_{m \in \mathrm{supp}(\sum_{s=1}^{t} w(s))} N(t,m) - \sum_{s=1}^{t} w_m(s).$$

- Inspired from Jedra and Proutiere [2020]

Tracking the Optimal Set

- Define $d_\infty(\mathbf{x}, \mathbf{y}) = \max_i |x_i - y_i|, \quad d_\infty(\mathbf{x}, C) = \min_{y \in C} d_\infty(x, y)$

Tracking the Optimal Set

- Define $d_\infty(\mathbf{x}, \mathbf{y}) = \max_i |x_i - y_i|, \quad d_\infty(\mathbf{x}, C) = \min_{y \in C} d_\infty(x, y)$

Tracking the Optimal Set

- Define $d_\infty(\mathbf{x}, \mathbf{y}) = \max_i |x_i - y_i|$, $\quad d_\infty(\mathbf{x}, C) = \min_{y \in C} d_\infty(x, y)$

Lemma

Under the modified D-tracking rule,

$$\lim_{t \to \infty} d_\infty\left((N(t,m)/t)_{m \in [M]},\ \mathcal{W}^\star(\boldsymbol{q}_0, \boldsymbol{\mu}_0)\right) = 0 \quad a.s..$$

Tracking the Optimal Set

- Define $d_\infty(\mathbf{x}, \mathbf{y}) = \max_i |x_i - y_i|, \quad d_\infty(\mathbf{x}, C) = \min_{y \in C} d_\infty(x, y)$

Lemma

Under the modified D-tracking rule,

$$\lim_{t\to\infty} d_\infty\left((N(t,m)/t)_{m\in[M]},\, \mathcal{W}^\star(\boldsymbol{q}_0, \boldsymbol{\mu}_0)\right) = 0 \quad a.s..$$

- Inspired by Degenne and Koolen [2019], the key idea in the proof is to track the behaviour of $\bar{\mathbf{w}}(t) = \frac{1}{t}\sum_{s=1}^{t} \mathbf{w}(s) \in \mathcal{W}^\star(\hat{\boldsymbol{q}}(t-1), \hat{\boldsymbol{\mu}}(t-1))$

Tracking the Optimal Set

- Define $d_\infty(\mathbf{x},\mathbf{y}) = \max_i |x_i - y_i|, \quad d_\infty(\mathbf{x}, C) = \min_{y\in C} d_\infty(x,y)$

Lemma

Under the modified D-tracking rule,

$$\lim_{t\to\infty} d_\infty\big((N(t,m)/t)_{m\in[M]},\ \mathcal{W}^\star(\boldsymbol{q}_0,\boldsymbol{\mu}_0)\big) = 0 \quad a.s..$$

- When $\mathcal{W}^\star(\boldsymbol{q}_0,\boldsymbol{\mu}_0) = \{\mathbf{w}^\star\}$, we recover the classical tracking result

$$\frac{N(t,m)}{t} \xrightarrow{t\to\infty} w_m^\star \quad \forall m, \ \text{a.s.}.$$

Stopping & Recommendation Rules

- The GLLR statistic between arms $a, b \in [K]$ at time t is

$$Z_{a,b}(t) = \begin{cases} N_a(t)\frac{\left(\hat{\mu}_a(t)-\hat{\mu}_{a,b}(t)\right)^2}{2} + N_b(t)\frac{\left(\hat{\mu}_b(t)-\hat{\mu}_{a,b}(t)\right)^2}{2}, & \hat{\mu}_a(t) \geq \hat{\mu}_b(t), \\ -Z_{b,a}(t), & \text{otherwise,} \end{cases}$$

where $\hat{\mu}_{a,b}(t) = \frac{N_a(t)}{N_a(t)+N_b(t)}\,\hat{\mu}_a(t) + \frac{N_b(t)}{N_a(t)+N_b(t)}\,\hat{\mu}_b(t)$.

Stopping & Recommendation Rules

- The GLLR statistic between arms $a, b \in [K]$ at time t is

$$Z_{a,b}(t) = \begin{cases} N_a(t)\frac{\left(\hat{\mu}_a(t)-\hat{\mu}_{a,b}(t)\right)^2}{2} + N_b(t)\frac{\left(\hat{\mu}_b(t)-\hat{\mu}_{a,b}(t)\right)^2}{2}, & \hat{\mu}_a(t) \geq \hat{\mu}_b(t), \\ -Z_{b,a}(t), & \text{otherwise}, \end{cases}$$

  where $\hat{\mu}_{a,b}(t) = \frac{N_a(t)}{N_a(t)+N_b(t)}\,\hat{\mu}_a(t) + \frac{N_b(t)}{N_a(t)+N_b(t)}\,\hat{\mu}_b(t)$.
- Let $Z(t) = \max_a \min_{b \neq a} Z_{a,b}(t)$

Stopping & Recommendation Rules

- The GLLR statistic between arms $a, b \in [K]$ at time t is

$$Z_{a,b}(t) = \begin{cases} N_a(t)\frac{\left(\hat{\mu}_a(t)-\hat{\mu}_{a,b}(t)\right)^2}{2} + N_b(t)\frac{\left(\hat{\mu}_b(t)-\hat{\mu}_{a,b}(t)\right)^2}{2}, & \hat{\mu}_a(t) \geq \hat{\mu}_b(t), \\ -Z_{b,a}(t), & \text{otherwise}, \end{cases}$$

  where $\hat{\mu}_{a,b}(t) = \frac{N_a(t)}{N_a(t)+N_b(t)}\,\hat{\mu}_a(t) + \frac{N_b(t)}{N_a(t)+N_b(t)}\,\hat{\mu}_b(t)$.
- Let $Z(t) = \max_a \min_{b \neq a} Z_{a,b}(t)$
- Given $\delta \in (0,1)$, let $\beta(t,\delta,\rho) = \log \frac{C\,t^{1+\rho}}{\delta}$, where C is a predetermined constant

Stopping & Recommendation Rules

- The GLLR statistic between arms $a, b \in [K]$ at time t is

$$Z_{a,b}(t) = \begin{cases} N_a(t)\frac{\left(\hat{\mu}_a(t)-\hat{\mu}_{a,b}(t)\right)^2}{2} + N_b(t)\frac{\left(\hat{\mu}_b(t)-\hat{\mu}_{a,b}(t)\right)^2}{2}, & \hat{\mu}_a(t) \geq \hat{\mu}_b(t), \\ -Z_{b,a}(t), & \text{otherwise}, \end{cases}$$

  where $\hat{\mu}_{a,b}(t) = \frac{N_a(t)}{N_a(t)+N_b(t)}\,\hat{\mu}_a(t) + \frac{N_b(t)}{N_a(t)+N_b(t)}\,\hat{\mu}_b(t)$.
- Let $Z(t) = \max_a \min_{b \neq a} Z_{a,b}(t)$
- Given $\delta \in (0,1)$, let $\beta(t,\delta,\rho) = \log \frac{C\,t^{1+\rho}}{\delta}$, where C is a predetermined constant
- Stopping rule: $\tau_{\delta,\rho} = \min\{t \geq 1 : Z(t) \geq \beta(t,\delta,\rho)$ and $\min_{k\in[K]} N_k(t) > 0\}$

Stopping & Recommendation Rules

- The GLLR statistic between arms $a, b \in [K]$ at time t is

$$Z_{a,b}(t) = \begin{cases} N_a(t)\frac{\left(\hat{\mu}_a(t)-\hat{\mu}_{a,b}(t)\right)^2}{2} + N_b(t)\frac{\left(\hat{\mu}_b(t)-\hat{\mu}_{a,b}(t)\right)^2}{2}, & \hat{\mu}_a(t) \geq \hat{\mu}_b(t), \\ -Z_{b,a}(t), & \text{otherwise}, \end{cases}$$

  where $\hat{\mu}_{a,b}(t) = \frac{N_a(t)}{N_a(t)+N_b(t)}\,\hat{\mu}_a(t) + \frac{N_b(t)}{N_a(t)+N_b(t)}\,\hat{\mu}_b(t)$.
- Let $Z(t) = \max_a \min_{b \neq a} Z_{a,b}(t)$
- Given $\delta \in (0,1)$, let $\beta(t,\delta,\rho) = \log \frac{C\,t^{1+\rho}}{\delta}$, where C is a predetermined constant
- Stopping rule: $\tau_{\delta,\rho} = \min\{t \geq 1 : Z(t) \geq \beta(t,\delta,\rho)$ and $\min_{k\in[K]} N_k(t) > 0\}$
- Recommendation rule: $\hat{k} = \arg\max_k \hat{\mu}_k(\tau_{\delta,\rho})$

Results

Under the box sampling, stopping, and recommendation rules stated before:

Theorem

- $P(\tau_{\delta,\rho} < \infty, \hat{k} \neq k^*) \leq \delta$

Results

Under the box sampling, stopping, and recommendation rules stated before:

Theorem

- $P(\tau_{\delta,\rho} < \infty, \hat{k} \neq k^*) \leq \delta$
- $\tau_{\delta,\rho}$ *satisfies*

$$\tau_{\delta,\rho} \leq \frac{1+\rho}{T^\star(\boldsymbol{q}_0, \boldsymbol{\mu}_0)} \log(1/\delta) + o(\log(1/\delta)) \quad a.s..$$

Hence, $\quad P(\tau_{\delta,\rho} < \infty) = 1.$

Results

Under the box sampling, stopping, and recommendation rules stated before:

Theorem

- $P(\tau_{\delta,\rho} < \infty, \hat{k} \neq k^*) \leq \delta$
- $\tau_{\delta,\rho}$ *satisfies*

$$\tau_{\delta,\rho} \leq \frac{1+\rho}{T^\star(\boldsymbol{q}_0, \boldsymbol{\mu_0})} \log(1/\delta) + o(\log(1/\delta)) \quad a.s..$$

  Hence, $\quad P(\tau_{\delta,\rho} < \infty) = 1.$

- *Asymptotic upper bound on* $\mathbb{E}[\tau_{\delta,\rho}]$*:*

$$\limsup_{\delta \downarrow 0} \frac{\mathbb{E}[\tau_{\delta,\rho}]}{\log(1/\delta)} \leq \frac{1+\rho}{T^*(\boldsymbol{q}_0, \boldsymbol{\mu}_0)}.$$

Non-Asymptotic Analysis:

Preliminaries

- In SE-type algorithms, multiple arms are pulled at each time instant; sub-optimal arms are eliminated on-the-fly

Preliminaries

- In SE-type algorithms, multiple arms are pulled at each time instant; sub-optimal arms are eliminated on-the-fly
- In LUCB-type algorithms, two arms are pulled at each time instant

Preliminaries

- In SE-type algorithms, multiple arms are pulled at each time instant; sub-optimal arms are eliminated on-the-fly
- In LUCB-type algorithms, two arms are pulled at each time instant
- In our setup, the learner cannot pull arms directly

Preliminaries

- In SE-type algorithms, multiple arms are pulled at each time instant; sub-optimal arms are eliminated on-the-fly
- In LUCB-type algorithms, two arms are pulled at each time instant
- In our setup, the learner cannot pull arms directly
- To maximise the chances of pulling a given arm k:

Preliminaries

- In SE-type algorithms, multiple arms are pulled at each time instant; sub-optimal arms are eliminated on-the-fly
- In LUCB-type algorithms, two arms are pulled at each time instant
- In our setup, the learner cannot pull arms directly
- To maximise the chances of pulling a given arm k:
  - When $\boldsymbol{q}_0$ is known: select box $m^* \in \arg\max_m q^0_{m,k}$

Preliminaries

- In SE-type algorithms, multiple arms are pulled at each time instant; sub-optimal arms are eliminated on-the-fly
- In LUCB-type algorithms, two arms are pulled at each time instant
- In our setup, the learner cannot pull arms directly
- To maximise the chances of pulling a given arm k:
  - When $\boldsymbol{q}_0$ is known: select box $m^* \in \arg\max_m q^0_{m,k}$
  - When $\boldsymbol{q}_0$ is unknown: select $m^* \in \arg\max_m \hat{q}_{m,k}(t)$ at time t

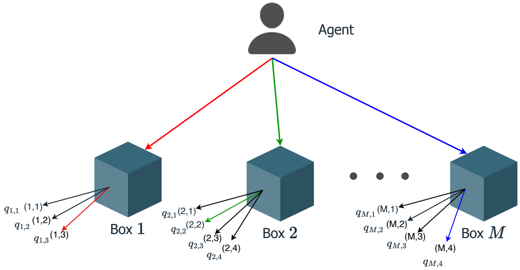
Preliminaries

- In SE-type algorithms, multiple arms are pulled at each time instant; sub-optimal arms are eliminated on-the-fly
- In LUCB-type algorithms, two arms are pulled at each time instant
- In our setup, the learner cannot pull arms directly
- To maximise the chances of pulling a given arm k:
    - When $\boldsymbol{q}_0$ is known: select box $m^* \in \arg\max_m q^0_{m,k}$
    - When $\boldsymbol{q}_0$ is unknown: select $m^* \in \arg\max_m \hat{q}_{m,k}(t)$ at time t
- Open question: Are the above strategies optimal?

Preliminaries

- In SE-type algorithms, multiple arms are pulled at each time instant; sub-optimal arms are eliminated on-the-fly
- In LUCB-type algorithms, two arms are pulled at each time instant
- In our setup, the learner cannot pull arms directly
- To maximise the chances of pulling a given arm k:
  - When $\boldsymbol{q}_0$ is known: select box $m^* \in \arg\max_m q^0_{m,k}$
  - When $\boldsymbol{q}_0$ is unknown: select $m^* \in \arg\max_m \hat{q}_{m,k}(t)$ at time t
- Open question: Are the above strategies optimal?
- Simplified setting: arms partitioned across boxes

Simplified Problem Setup: Partition



Goal: fixed-confidence BAI

- Arms partitioned across boxes
- Arm k of box m indexed as $A_{m,k}$ or simply as (m,k)
- $\mathcal{A}_m$: set of arms in box m
- $\sum_{m=1}^{M} |\mathcal{A}_m| = K$
- Agent knows $\mathcal{A}_1, \ldots, \mathcal{A}_M$
- Unknowns:
  - $\boldsymbol{q}_0 = \{q_{m,k}^0\}_{m,k}$
  - $\boldsymbol{\mu}_0 = \{\mu_{m,k}^0\}_{m,k}$
- Best arm: $(m^*, k^*) = \arg\max_{m,k} \mu_{m,k}^0$

Converse

- WLOG, let $(1,1)$ be the best arm
- Let $\Delta_{m,k} = \mu^0_{1,1} - \mu^0_{m,k}$ for all $(m,k) \neq (1,1)$, and $\Delta_{1,1} = \min_{(m,k)\neq(1,1)} \Delta_{m,k}$

Converse

- WLOG, let $(1,1)$ be the best arm
- Let $\Delta_{m,k} = \mu^0_{1,1} - \mu^0_{m,k}$ for all $(m,k) \neq (1,1)$, and $\Delta_{1,1} = \min_{(m,k)\neq(1,1)} \Delta_{m,k}$

Theorem

Under any δ-PC algorithm,

$$\mathbb{E}[\tau_\pi] \geq \log\left(\frac{1}{2.4\,\delta}\right) \cdot \sum_{m=1}^{M} \max_{k\in\mathcal{A}_m} \frac{1}{q^0_{m,k}\,\Delta^2_{m,k}}.$$

- Technique: change-of-measure arguments of Garivier and Kaufmann [2016]

Achievability: Successive Elimination

Notations:

- $t_{m,k}(n)$: # pulls of arm $A_{m,k}$ up to round n
- $\alpha_\delta(x) = \sqrt{\frac{2\ln(8Kx^2/\delta)}{x}}$
- $\mathrm{UCB}_{m,k}(n) = \hat{\mu}_{m,k}(n) + \alpha_\delta(t_{m,k}(n))$
- $\mathrm{LCB}_{m,k}(n) = \hat{\mu}_{m,k}(n) - \alpha_\delta(t_{m,k}(n))$

Select box until each active arm is pulled n times in round n

---

Algorithm 1 Successive Elimination

Input: $K, M, \delta > 0, \mathcal{A}_m$ for $m \in [M]$
Output: $\hat{a} \in [K]$ (best arm).
Initialization: $S = [K], B = [M], S_m = [a_m], n = 0,$
$\hat{\mu}_{m,k}(n) = 0\ \forall k, m,\ S_m = \mathcal{A}_m\ \forall m, t = 0.$

```
while |S| > 1 do
    n ← n + 1
    For each m ∈ B, select box m until every active arm A_{m,k} in box m is pulled at least n times.
    For every box selection, increment t by 1.
    Update t_{m,k}(n), μ̂_{m,k}(n), UCB_{m,k}(n) and LCB_{m,k}(n) for all the active arms.
    if ∃A_{m',k'} ∈ S such that UCB_{m,k}(n) < LCB_{m',k'}(n) then
        S_m ← S_m \ A_{m,k},   S ← ∪_{m∈[M]} S_m,
        B ← {m : S_m ≠ ∅}.
    end if
    if |S| = 1 then
        â ← a ∈ S,   S ← ∅,   B ← ∅.
    end if
end while
return â.
```

Results

Theorem

Fix $\delta \in (0,1)$. With probability greater than $1-\delta$:

- *The SE algorithm outputs the correct best arm*
- *The SE algorithm stops at time $\leq \sum_{m=1}^{M} U_m$, where U_m is a random variable with*

$$P\left(U_m = \max_{k \in \mathcal{A}_m} O\left(\frac{\ln\left(\frac{K}{\delta \Delta_{m,k}}\right)}{q_{m,k}^0 \, \Delta_{m,k}^2}\right)\right) \geq 1 - \frac{\delta |\mathcal{A}_m|}{K}.$$

Results

Theorem

Fix $\delta \in (0,1)$. With probability greater than $1-\delta$:

- *The SE algorithm outputs the correct best arm*
- *The SE algorithm stops at time $\leq \sum_{m=1}^{M} U_m$, where U_m is a random variable with*

$$P\left(U_m = \max_{k \in \mathcal{A}_m} O\left(\frac{\ln\left(\frac{K}{\delta \Delta_{m,k}}\right)}{q_{m,k}^0 \, \Delta_{m,k}^2}\right)\right) \geq 1 - \frac{\delta |\mathcal{A}_m|}{K}.$$

- Lower bound $= \Omega\left(\sum_{m=1}^{M} \max_{k \in \mathcal{A}_m} \frac{1}{q_{m,k}^0 \, \Delta_{m,k}^2}\right)$ (order-wise matching in problem unknowns)

In Summary

- Problem studied: BAI with limited precision sampling
- Modified D-tracking algorithm to handle non-unique optimal box weights
- Partition setting: SE algorithm that selects each box until each active arm is pulled n times in round n
- Non-partition setting: SE/LUCB-type algorithm design is an open question

Thank You!

Questions? Hit me up!
Email: karthik@nus.edu.sg
Web: https://karthikpn.com

References

Rémy Degenne and Wouter M Koolen. Pure exploration with multiple correct answers. *Advances in Neural Information Processing Systems*, 32, 2019.

Aurélien Garivier and Emilie Kaufmann. Optimal best arm identification with fixed confidence. In *Conference on Learning Theory*, pages 998–1027. PMLR, 2016.

Yassir Jedra and Alexandre Proutiere. Optimal best-arm identification in linear bandits. *Advances in Neural Information Processing Systems*, 33:10007–10017, 2020.

Emilie Kaufmann, Olivier Cappé, and Aurélien Garivier. On the complexity of best-arm identification in multi-armed bandit models. *The Journal of Machine Learning Research*, 17(1):1–42, 2016.