# Multi-Armed Bandit Problem with Distributed Players


## ABSTRACT

TBD


## 1. INTRODUCTION

- Yahoo!'s front page *Today* module is powered by *explore-exploit multi-armed bandit technology*. To handle the huge traffic load, Yahoo! operates several Colos (or data-centers) distributed accros the US ((depicted in Figure 1))
- Consider a setup where a group of Colos are coomunicating to perform joint explore-exploite Web experiments over a set of users, to determine a certain best outcome (e.g., optimal Web page layout).
- There is a range of strategies for sharing data over a broadcast network. The two extremes are (a) no communication - each Colo conducts separate experiments; and (b) full communication - every user interaction is shared to all Colos
- Reducing the distributed explore-exploit Web experiments setup to a multi-armed bandit framework is done here, by replacing stories/articles, user page views, user clicks, and colos, with arms, arms pulls, rewards, and players, respectively.
- The idea of multi-armed bandit (MAB) with distributed players was already considered in [3]. However, in that setup the players were not exchanging any information, and a simultaneous arm pull by two or more agents resulted in a collision which was translated to a reward loss or some arbitrary reward sharing between the colliding arms.

## 2. PROBLEM SETUP

A stochastic multi armed bandit game with distributed players is characterized by the number of arms $K$, total budget of $n$ pulls, and $K$ probability distributions $\nu_1, \ldots, \nu_K$. In contrast to the conventional setup where only one player (forecaster) is involved, here, $M$ players are participating in the game. The players are assumed to be homogeneous in the sense that they have the same utility distribution over the arms. For $t = 1, \ldots, n$ rounds, one of the player (determined by round-robin scheduling) selects an arm $I_t$ in the set of arms $\{1, \ldots, K\}$ and observes a reward drawn from $\nu_{I_t}$ independently from past events (actions and observations). We further assume that the players are sharing their

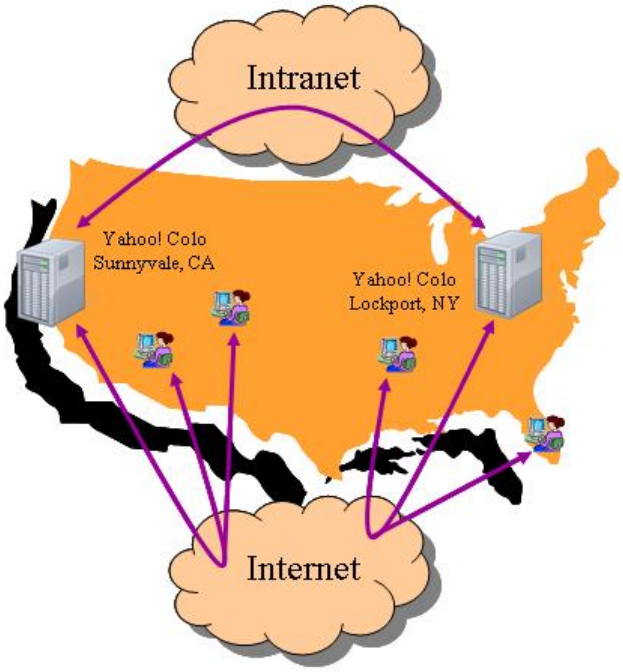


**Figure 1: A group of Colos are communicating via Intranet to perform joint MAB/Explore-Exploit experiments over the Internet.**

information via a delayless and error free backhaul network where a unit cost is charged for every broadcast.

At the end of the $n$ rounds, the players should unanimously select an arm, denoted $J_n$, which is evaluated in terms of the difference between the mean reword of the optimal arm and the mean reward of $J_n$. In particular, let $\mu_1, \ldots, \mu_K$ be the respective means of $\nu_1, \ldots, \nu_K$. Let $\mu^* = \max_{k \in \{1,\ldots,K\}} \mu_k$. The regret of the decision is $r_n \triangleq \mu^* - \mu_{J_n}$. We also assume binary rewards $[0, 1]$, and that there is a unique optimal arm denoted by $k^*$ (i.e., $\mu_{k^*} = \mu^*$). For arm $k \neq k^*$ we define the gap $\Delta_k = \mu^* - \mu_k$, and the minimum gap $\Delta^* = \min_{\substack{k=1,\ldots,K \\ k \neq k^*}} \Delta_k$. We use the notation $(k)$ to denote the $k$th best arm (with random tie break), hence, $\Delta^* = \Delta_{(1)} \leq \Delta_{(2)} \leq \Delta_{(K)}$. Following [2] we denote by $e_n$ the error probability $e_n = Pr\left(J_n \neq i^*\right)$ and use it as our merit throughout this work since it behaves similarly to the average regret $\mathbb{E}\left(r_n\right)$.

For each arm $i$ and all time $t \geq 1$, we denote by $T_i^m(t)$ the number of times arm $i$ was pulled from round 1 to $t$ by player $m$, and by $X_{i,1}^m, \ldots, X_{i,T_i^m(t)}^m$ the sequence of associated rewards. The average reward of arm $i$ at player $m$ after $s$ pulls is denoted $\hat{X}_{i,s}^m \triangleq \frac{1}{s}\sum_{j=1}^{s} X_{i,j}^m$ while the total average reward of arm $i$ after $s$ pulls is denoted by $\hat{X}_{i,s} \triangleq \frac{1}{M}\sum_{m=1}^{M} \hat{X}_{i,s}^m$. (((Eshcar: These definitions are valid only for algorithms in which different players execute the same number of pulls for a given arm.)))

# 3. BACKGROUND AND PREVIOUS RESULTS

## 3.1 Successive Elimination Algorithm

Successive Elimination Algorithm:

- Let $A_1 = \{1, \ldots, K\}$, $n_0 = 0$, $n_k \triangleq \lceil \frac{1}{\overline{\log K}} \frac{n-K}{K+1-k} \rceil$ for $k = 1, \ldots, K-1$ where $\overline{\log K} \triangleq \frac{1}{2} + \sum_{i=2}^{K} \frac{1}{i}$
- For each phase $k = 1, \ldots, K-1$ do the following:
  - Pull each surviving arm $i \in A_k$, $n_k - n_{k-1}$ times
  - Eliminate the "worst" arm (with random tie brakes) $A_{k+1} = A_k \setminus \text{argmin}_{i \in A_k} \left\{ \hat{X}_{i,n_k} \right\}$
- Choose the last arm as the "best arm" $J_n = \{A_K\}$

The probability of error of the SE algorithm is upper bounded in [2].

**Theorem 1** *The probability of error of SE satisfies*

$$e_n \leq \frac{K(K-1)}{2} \exp\left(-\frac{n-K}{H_2 \log K}\right) . \quad (1)$$

## 3.2 Fagin's Algorithm

Assume you have $M$ urns with balls of $K$ colors, and the goal is to find the color with minimum total amount of balls in all urns with least number of "data transfers" between urns. A generalized version of this problem was studied in [1].

Fagin's algorithm (simplified version)

- Sort the number of balls of each color in each urn (ascending order)
- Collect candidates: starting with the color with least balls, broadcast the index of the next color in each urn until you see at least one color in all urns
- Collect information: for each index that has been collected, broadcast the number of balls in each urn
- Choose the color with minimum aggregated number of balls

Fagin's algorithm is optimal with high probability, in terms of number of index (candidates) broadcasts and information broadcasts, in the worst case if the balls in each urn are selected independently.

# 4. EXPERIMENTS AND ANALYSIS

## 4.1 Simple Distributed Successive Elimination

This section evaluates a simple algorithm for the distributed MAB problem. This algorithm runs one Fagin step on top of a distributed successive elimination variant [2]. Consider $n$ rounds and $M$ players, each players executes as follows:

- runs the SE scheme independently, executing $n/M$ rounds
- broadcasts the index of its winning arm
- collects the former broadcasts to form a common list of candidates (arms)
- broadcasts the rewards and number of pulls (executed by the player) for each arm in the candidates list

Finally, the total average reward (total reward divided by total number of pulls) is calculated for each arm in the candidates list; the best arm is the arm with the highest total average.

### 4.1.1 Error Probability Analysis

For simplicity let us consider a setup with only two players conducting separate SE algorithm with $n/2$ pulls each. Then, they exchange the reward associated with their winning arms. The decision procedure:

- If both players have selected the same arm then it is preannounced as the best arm
- Otherwise, the selected arm is the one with greater reward

More formaly, the decision is given by

$$J = \underset{J^1, J^2}{\text{argmax}} \left\{ \hat{X}_{J^1}^1, \hat{X}_{J^2}^2 \right\} . \quad (2)$$

An error event is when both players are wrong, or one of them is right but the reward of the wrong arm indicated by the second player is higher. Hence,

$$\begin{aligned} e_n &= 2Pr\left(J^2 \neq J^1 = i^*, \hat{X}_{J^1}^1 \leq \hat{X}_{J^2}^2\right) + Pr\left(J^1 = J^2 \neq i^*\right) \\ &= 2Pr\left(\hat{X}_{J^1}^1 \leq \hat{X}_{J^2}^2\right)\left(1 - \widetilde{Pe}\right)\widetilde{Pe} + \widetilde{Pe}^2 \\ &\leq 2e^{-\Delta_{(1)}^2 n}\left(1 - \widetilde{Pe}\right)\widetilde{Pe} + \widetilde{Pe}^2 \\ &= \left(2e^{-\Delta_{(1)}^2 n} + \widetilde{Pe}\right)\widetilde{Pe}, \end{aligned} \quad (3)$$

where (a) $J^1, J^2$ and $\hat{X}_{J^1}^1 \leq \hat{X}_{J^2}^2$ are the indices and average rewards of the winning arms of the two players respectively; (b) $\widetilde{Pe}$ is the error probability of an arbitrary algorithm with $n/2$ pulls; and (c) the third inequality is due to Hoeffding's inequality.

The error expression (3) is valid for any best arm identification algorithms. In the special case where SE is applied we can use the upper bound (1) to rewrite (3) as

$$e_n \leq \frac{3K^2(K-1)^2}{4} \exp\left(-\frac{n-2K}{H_2 \log K}\right) . \quad (4)$$

It is concluded that when two players are involved, the simple distributed SE algorithm achieves the same error exponent (in the number of pulls $n$) as that of the centralized SE algorithm.

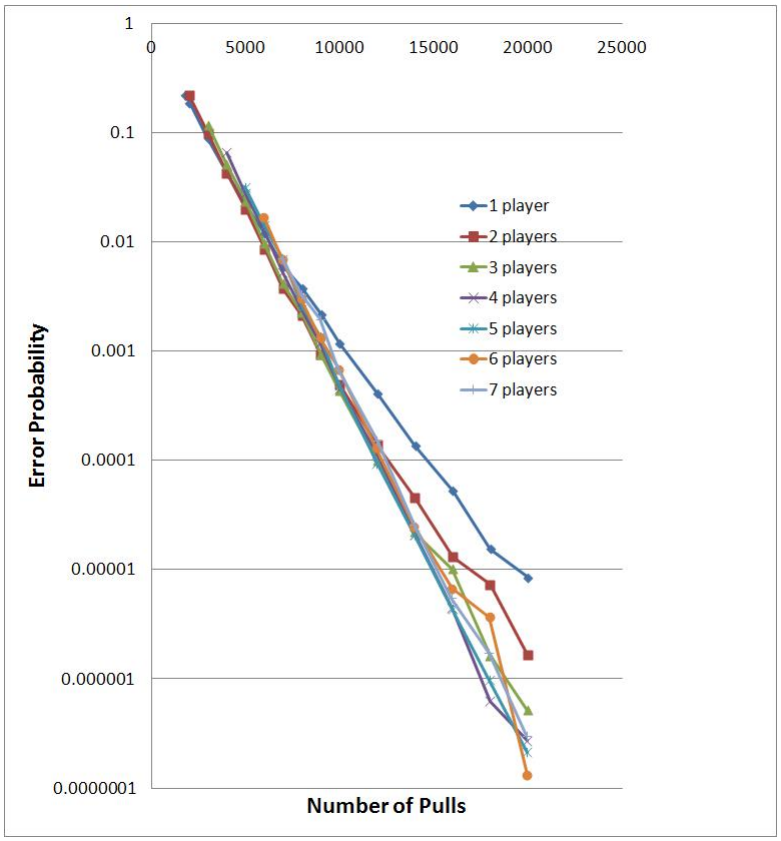


**Figure 2: Simple SE algorithm best arm identification error probability as functions of the number of pulls for several numbers of players.**

#### 4.1.2 *Numerical Evaluation*

We measure the error probability of the algorithm as a function of the total number of pulls, with 1 to 5 players. The instance set up is of a worst case scenario, which includes one "good" arm $P_g = 0.5$ and one group of "bad" arms $P_b = 0.4$ (total of $K = 20$ arms). The results (log scaled) are depicted in Figure 2.

Figure 2 shows that as the number of pulls increases, passing some threshold above the minimal number of pulls, running with more players exhibits lower error probability than running with a single player.

## 4.2 Communication Intensive Successive Elimination

Here we use the SE algorithm (see 3.1) with a slight modification which is dictated by the communication restrictions we apply. In particular, in each of the algorithm phase we eliminate the approximated "worst" arm using a restricted version of Fagin's algorithm (see 3.2. Accordingly, in phase A we restrict the number of index broadcasts not to exceed $L$ transmissions. It is evident that in this case errors may occur and occasionally not the "worst" arm is eliminated.

It is easily verified that the total number of broadcasts (during both phases and assuming $L \leq K$) is upper bounded by

$$N_b \leq \sum_{k=1}^{\max\{K-L+1,K-1\}} 2kLM + \sum_{k=K-L+2}^{K-1} 2k(K-k+1)M \ . \quad (5)$$

Evidently from the way our system is designed, the number of broadcasts is independent of the number of pulls. As a baseline for comparing communication loads a naive system where all pulls are broadcasts to all players is considered. Such a system uses $Mn$ broadcasts.

#### 4.2.1 *Error Probability Analysis*

$L_k^m$: arms to be transmitted by player $m$ at round $k$

$A_k$: surviving arms after round $k$

$B_k$: candidates to be dismissed at the end of round $k$

$C$: $\exists k, \exists j, |\hat{X}_{j,n_k} - \mu_j| > \frac{\Delta_{(K+1-k)}}{\alpha}$ (where $\alpha \geq 2$)

$D$: $\exists k, \forall j \in B_k : \alpha\Delta_j < \Delta_{(K+1-k)}$

$\phi$: $\forall k, \exists m, \forall l \in \{(K), (K-1), \ldots, (K+1-k)\}, \forall j : \alpha\Delta_j \leq \Delta_l, \hat{X}^m_{j,\frac{n_k}{M}} > \hat{X}^m_{l,\frac{n_k}{M}}$

$\neg\phi$: $\exists k, \forall m, \exists l \in \{(K), (K-1), \ldots, (K+1-k)\}, \exists j : \alpha\Delta_j \leq \Delta_l, \hat{X}^m_{j,\frac{n_k}{M}} \leq \hat{X}^m_{l,\frac{n_k}{M}}$

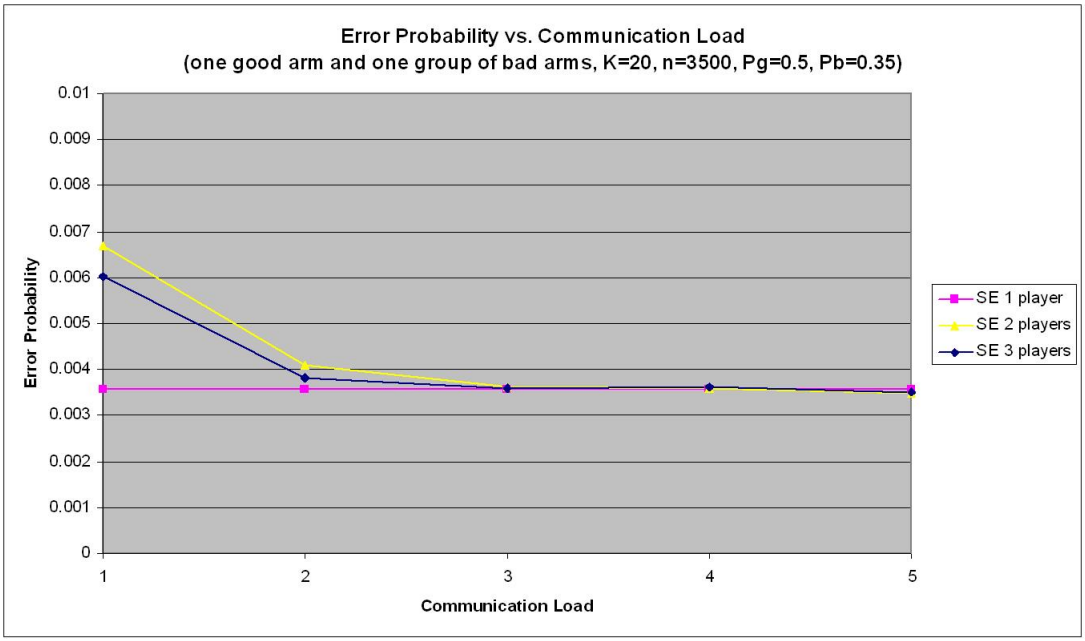


**Figure 3: Communication intensive SE best arm identification error probability as functions of the communication limitation for several numbers of players.**

$$
\begin{aligned}
D &\Rightarrow \neg\phi \\
\mathbb{P}(e_n) &\leq \mathbb{P}(C \cup D) \leq \mathbb{P}(C) + \mathbb{P}(D) \\
&\leq \mathbb{P}(C) + \mathbb{P}(\neg\phi)
\end{aligned}
$$

$$
\begin{aligned}
& C : \exists k, \exists j, |\hat{X}_{j,n_k} - \mu_j| > \frac{\Delta_{(K+1-k)}}{\alpha} \\
\mathbb{P}(C) &\leq 2\sum_{k=1}^{K-1}\sum_{j=1}^{K}\mathbb{P}(\hat{X}_{j,n_k} - \mu_j > \frac{\Delta_{(K+1-k)}}{\alpha}) \\
&\leq 2\sum_{k=1}^{K-1}\sum_{j=1}^{K}\exp -\frac{2n_k\Delta^2_{(K+1-k)}}{\alpha^2} \\
&\leq 2\sum_{k=1}^{K-1}\sum_{j=1}^{K}\exp -\frac{n-K}{\overline{log}(K)H_2}\frac{2}{\alpha^2} \\
&\leq 2K^2\exp -\frac{n-K}{\overline{log}(K)H_2}\frac{2}{\alpha^2}
\end{aligned}
$$

$$
\begin{aligned}
& \neg\phi : \exists k, \forall m, \exists l \in \{(K), (K-1), \ldots, (K+1-k)\}, \\
& \quad \exists j : \alpha\Delta_j \leq \Delta_l, \hat{X}^m_{j,\frac{n_k}{M}} - \hat{X}^m_{l,\frac{n_k}{M}} \leq 0 \\
\mathbb{P}(\neg\phi) &\leq \sum_{k=1}^{K-1}(\sum_{l\in\{(K),\ldots,(K+1-k)\},j:\alpha\Delta_j\leq\Delta_l}\mathbb{P}(\hat{X}^m_{j,\frac{n_k}{M}} - \hat{X}^m_{l,\frac{n_k}{M}} \leq 0))^M \\
&\leq \sum_{k=1}^{K-1}(\sum_{l\in\{(K),\ldots,(K+1-k)\},j:\alpha\Delta_j\leq\Delta_l}\exp(-\frac{n_k}{M}(\Delta_l - \Delta_j)^2))^M \\
&\leq \sum_{k=1}^{K-1}(kK\exp(-\frac{n_k}{M}\Delta^2_{(K+1-k)}(1-\frac{1}{\alpha})^2))^M \\
&\leq \sum_{k=1}^{K-1}(kK\exp(-\frac{1}{M}\frac{n-K}{\overline{log}(K)H_2})(1-\frac{1}{\alpha})^2)^M \\
&\leq \sum_{k=1}^{K-1}(kK)^M\exp(-\frac{n-K}{\overline{log}(K)H_2})(1-\frac{1}{\alpha})^2) \\
&\leq K^{2M+1}\exp(-\frac{n-K}{\overline{log}(K)H_2})(1-\frac{1}{\alpha})^2)
\end{aligned}
$$

$$
\mathbb{P}(e_n) \leq 2K^2\exp -\frac{n-K}{\overline{log}(K)H_2}\frac{2}{\alpha^2} + K^{2M+1}\exp(-\frac{n-K}{\overline{log}(K)H_2})(1-\frac{1}{\alpha})^2)
$$

when $\alpha = \sqrt{2}+1$

$$
\mathbb{P}(e_n) = O(K^{2M+1}\exp(-\frac{0.34n-K}{\overline{log}(K)H_2}))
$$

### 4.2.2 *Numerical Evaluation*

In Figure 3 Monte-Carlo simulations of the error probability of the distributed SE algorithm are plotted as functions of the maximum number of index broadcasts $L$, for several

number of players. The curves are plotted for a worst case setup which includes one "good" arm and a one group of "bad" arms (total of $K = 20$ arms).

A few observations:

- As expected the probability of error reduces with increasing values of $L$ and coincides with the centralized case (one player)
- Fixing $L$, the probability of error reduces with the number of players $M > 1$ (remark: we should prove it analytically)

# 5. CONCLUDING REMARKS

We have experimented with two distributed variants of the SE algorithm. Trying to mimic the centralized setup where only one player is considered we have formulated the communication intensive SE algorithm. Numerical evaluation reveals a general trend in which the probability of error decreases when the number of broadcasts after each SE round increases. Moreover, with unlimited number of broadcasts the probability of error of the distributed setup converges to that of the centralized one.

The second approach includes separate SE algorithm by each player and a final phase where information regarding the winning arms is shared. Surprisingly, this scheme that does not mimic the centralized setup performs better when the number of pulls is above a certain threshold.

# 6. REFERENCES


[1] R. Fagin. Combining fuzzy information: an overview. In *the ACM Special Interest Group on Management of Data (SIGMOD'2002)*, Wisconsin, USA, June 3–6 2002.

[2] J. Y. Audibert, S. Bubeck, and R. Munos. Best arm identification in multi-armed bandits. In *23rd annual conference on learning theory (COLT'2010)*, Jun. 27–29 2010.

[3] K. Liu and Q. Zhao. Distributed learning in multi-armed bandit with multiple players. *IEEE Transactions on Signal Processing*, 58(11):5667–5681, 2010.