The Twenty-Second ACM Conference on Economics and Computation (EC'21)

Noble Deceit: Optimizing Social Welfare for Myopic Multi-Armed Bandits

Ashwin Maran, Jeremy McMahan, and Nathaniel Sauerberg
University of Wisconsin--Madison

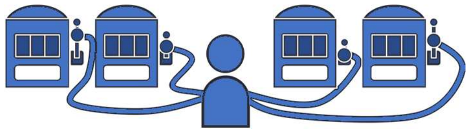
Problem Statement

Problem Statement:

- Multi-armed bandit problem but each arm is pulled by a myopic agent
- An instance of the problem involves arms $a_1, \ldots, a_m$, each having a persistent but a-priori random reward R_i drawn independently from distribution D_i with mean reward μ_i
- **Goal:** maximize expected reward

Game Timeline:

- The rewards are drawn at the start of the game
- The agents begin arriving over time
- Agents receive some information from the mechanism and use it to decide which arm to pull
- They pull an arm, receive their reward, and exit the system
- The mechanism gets to observe the reward received by the agent and can use this observation to decide what to reveal to future agents



Our Approach: use information asymmetry to incentivize agents to explore, extending the results from [1]

[1] Kremer, Ilan, Yishay Mansour, and Motty Perry. "Implementing the "wisdom of the crowd"." Journal of Political Economy 122.5 (2014): 988-1012.

Approach and Key Ideas

Goal: determine the best arm by exploring each arm
Divide into phases: In the k-phase, we explore a_k

What if an agent is recommended a_k?

- **Key Information Asymmetry**: agents don't know if they're exploring or exploiting:
  - Might be first to explore a_k -- incur some **cost**
  - Maybe a_k was explored by a previous agent and found to be better than a_i -- get some **benefit**
- Design k-phase to balance these two factors

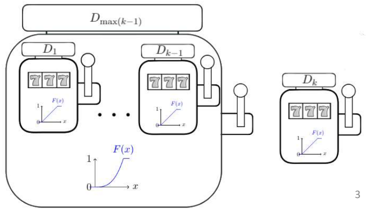
Structure of the k-phase:

- Partition support of $D_{\max\{k-1\}} = \max\{D_1, \ldots, D_{k-1}\}$ into a set of intervals $\{I_t\}$
- Let a_i be the best arm among $\{a_1, \ldots, a_{k-1}\}$
- Each agent t will explore a_k iff $r_i \in I_t$
- Before Exploration: agents exploit a_i
- After Exploration: agents pull the better of a_k and a_i

This works if agents view the value of a_i as a random draw from $D_{\max(k-1)}$, i.e. they can't learn anything about what happened in previous phases

Ensure phases are independent:

- k-phase ends only when it is certain that a_k would have been explored irrespective of realizations of arms
- the length of k-phase depends only on the distributions $D_1, \ldots, D_k$ and not the actual realizations



Our Mechanism (IPM)

1. Recommend a_1 to the first agent
2. For each $k \in \{2, \ldots, m\}$, begin the k-phase:
   a) Partition $D_{\max(k-1)}$ into intervals $I_1, \ldots, I_T$
   b) Find the best arm i among $\{1, \ldots, k-1\}$
   c) Find $t \leq T$ such that $r_i \in I_t$
   d) For an agent j,
      i. If $j < t$, recommend a_i
      ii. If $j = t$, recommend a_k
      iii. If $j > t$, and $r_k \geq r_i$, recommend a_k
      iv. If $j > t$, and $r_k < r_i$, recommend $\arg\max\{r_i, \mu_{k+1}\}$
3. Recommend the best arm after phases end

IPM is IC and always determines the best explorable* arm

$\Longrightarrow$ IPM attains constant regret w.r.t. the optimal offline mechanism whenever all arms are explorable

$\Longrightarrow$ IPM attains constant regret w.r.t. the optimal IC mechanism unconditionally

We also show IPM achieves first-best whenever possible for an IC mechanism to do so

* An arm a_i is *unexplorable* if another arm exceeds its mean reward with certainty. No IC mechanism can explore such arm