Students, Exams and Bandits

EE6106 Online Learning and Optimization - Course Project

Prabhat Reddy (22M2156)
Rahul Choudhary (200070065)

April 29, 2023

Indian Institute of Technology Bombay

Outline

Motivation

What is the best strategy a student can follow when writing an exam to get maximum possible marks?

Can we obtain an answer to this question by thinking of exam writing as a multi-armed bandit problem?

Problem setting

- Arms $1, 2, \ldots, K$ and fixed rewards $r_1, r_2, \ldots, r_K$.
- Without loss of generality $r_1 \geq r_2 \geq \ldots \geq r_K$
- Time Budget T
- Arm i consumes a stochastic time t_i. $\mathbb{E}(t_i) = T_i$.
- For all positive integers t and $i, j \in [K]$ such that $i < j$, we have

$$\mathbb{P}(t_i \leq t) \leq \mathbb{P}(t_j \leq t)$$

  which tells us that we expect questions with higher marks to take more time to solve.
- Note that the above assumption gives us the ordering $T_1 \geq T_2 \geq \cdots \geq T_K$.

Paper with 2 questions

Consider policy π_1 which pulls arm 1 before arm 2, and policy π_2 which does the opposite. Note that

$$\begin{aligned}\mathbb{E}_{\pi_1}(\text{Reward}) &= \mathbb{E}(r_1\mathbb{I}\{t_1 \leq T\} + r_2\mathbb{I}\{t_1 + t_2 \leq T\}) \\ &= r_1\mathbb{E}(\mathbb{I}\{t_1 \leq T\}) + r_2\mathbb{E}(\mathbb{I}\{t_1 + t_2 \leq T\}) \\ &= r_1\mathbb{P}(t_1 \leq T) + r_2\mathbb{P}(t_1 + t_2 \leq T)\end{aligned}$$

Similarly, we get $\mathbb{E}_{\pi_2}(\text{Reward}) = r_2\mathbb{P}(t_2 \leq T) + r_1\mathbb{P}(t_1 + t_2 \leq T)$.

Paper with 2 questions

The difference between rewards from both the policies is given as follows.

$$\mathbb{E}_{\pi_1}(\mathrm{R}) - \mathbb{E}_{\pi_2}(\mathrm{R}) = r_1 \mathbb{P}(t_1 \leq T) - r_2 \mathbb{P}(t_2 \leq T) - (r_1 - r_2)\mathbb{P}(t_1 + t_2 \leq T)$$

We now consider how this quantity varies in the following situations:

1. All questions are expected to be solved within time T
2. At least one question takes too much time to solve

All questions can be solved within time T

In this case, we have $\mathbb{P}(t_1 \leq T) \approx \mathbb{P}(t_2 \leq T)$. We obtain the following inequality using the above expression.

$$\begin{aligned}\mathbb{E}_{\pi_1}(\mathrm{R}) - \mathbb{E}_{\pi_2}(\mathrm{R}) \approx & (r_1 - r_2)\mathbb{P}(t_1 \leq T) - (r_1 - r_2)\mathbb{P}(t_1 + t_2 \leq T) \\ = & (r_1 - r_2)(\mathbb{P}(t_1 \leq T) - \mathbb{P}(t_1 + t_2 \leq T)) \\ \geq & 0\end{aligned}$$

This tells us that it's better to solve questions with higher marks in this case.

At least one question takes too much time to solve

In this case, we have $\mathbb{P}(t_1 \leq T) \approx \mathbb{P}(t_1 + t_2 \leq T)$. We obtain the following inequality using the above expression.

$$\begin{aligned}\mathbb{E}_{\pi_1}(\mathrm{R}) - \mathbb{E}_{\pi_2}(\mathrm{R}) \approx & r_2(\mathbb{P}(t_1 + t_2 \leq T) - \mathbb{P}(t_2 \leq T)) \\ \leq & 0\end{aligned}$$

This tells us that in this case, it's better to solve the easier questions first in order to get more marks.

Exam with K questions

Carrying on the intuition from the previous analysis, we consider the following idea to create a good algorithm:

- If all questions can be solved within the total time, choose to answer the question with highest number of marks.
- If any question takes more time to solve than the stipulated time, choose to answer the question with least number of marks.

However, note that we don't know the expected time to solve each question. We can't develop an algorithm using the above idea without that knowledge.

Exam with K questions

For desigining the algorithm, we assume a linear model that relates the expected time and known rewards as follows.

$$T_i = ar_i$$

The parameter a is estimated using previously observed rewards and times.

$$\hat{a} = \frac{\sum_{i \in \mathcal{H}} t_i}{\sum_{i \in \mathcal{H}} r_i}$$

where $\mathcal{H}$ is the set of arms previously pulled.

Exam with K questions

We mathematically approximate the previously discussed ideas as follows. Let $\hat{T} = \sum_{i \in [K] \setminus \mathcal{H}} \hat{ar}_i$ denote the **sum of estimated expected times** to solve all remaining questions, and let $T_R = T - \sum_{i \in \mathcal{H}} t_i$ denote the **remaining time available**.

- If $\hat{T} \leq T_R$, choose to answer the question with highest number of marks.
- If $\hat{T} > T_R$, choose to answer the question with least number of marks.

Algorithm for exam with K questions

Algorithm

Initialize T, $S_r \leftarrow 0$, $S_t \leftarrow 0$.

Pull arm k with least reward r_k and obtain time consumed $\widetilde{t}_k$.

$T \leftarrow T - \widetilde{t}_k$

Initialize $a \leftarrow \frac{\widetilde{t}_k}{r_k}$.

Until all questions are solved:

- Calculate $\hat{T} = \sum a r_i$ (summation over all remaining arms)
- If $\hat{T} \leq T$: Play arm with highest reward and obtain time $\widetilde{t}$
- Else: Play arm with least reward and obtain time $\widetilde{t}$
- $a \leftarrow \frac{\text{Sum of all times obtained till now}}{\text{Sum of all rewards obtained till now}}$
- $T \leftarrow T - \widetilde{t}$

Table of marks

Algorithm	**Poisson Parameter (λ)**				
	$b-0.1$	b	$b+0.5$	$b+1$	$b+2$
Our Algorithm	**100.00**	89.15	71.30	**59.95**	**42.75**
Least marks first	92.00	87.00	69.50	59.00	41.50
Highest marks first	**100.00**	**96.70**	**72.80**	58.30	41.60
Random	97.20	88.20	72.60	58.35	41.95

Table 1: Total marks obtained by algorithms under different time delay distributions. Note that $b = \frac{T}{\sum r_i}$ is the ratio of available time budget and sum of all rewards.

Marks vs Time plot with $\lambda = b - 0.1$

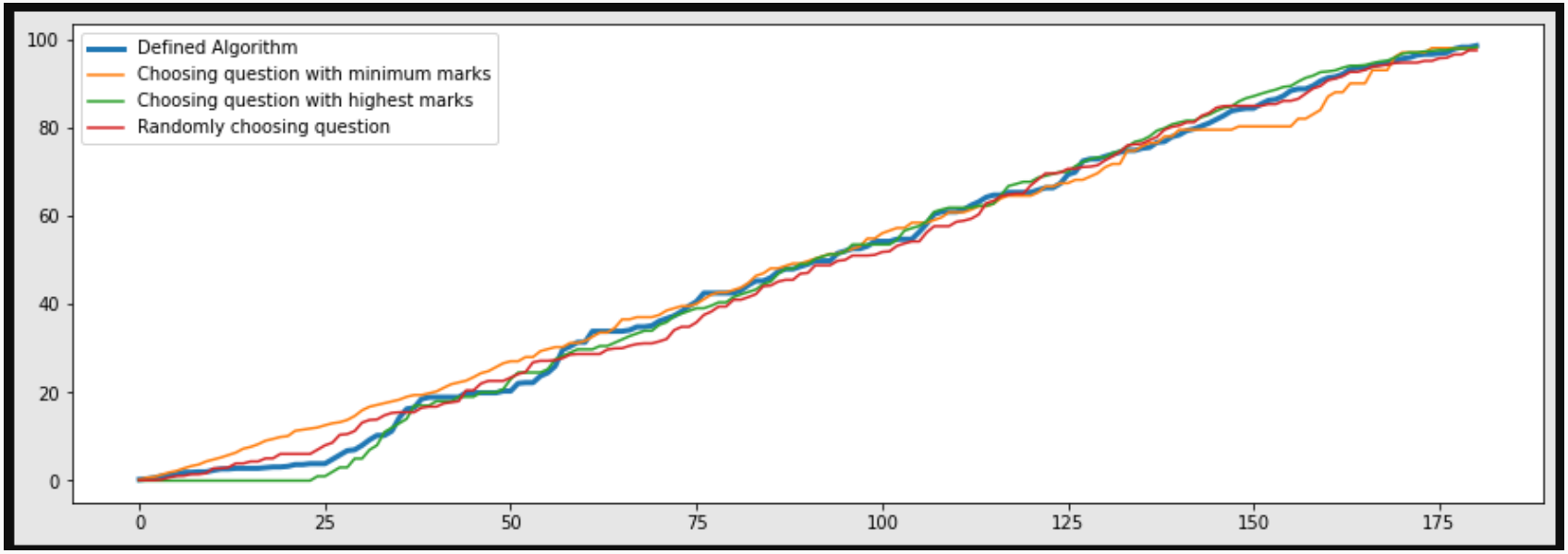

Figure 1: This figure shows how different algorithms obtain rewards over time, when delays are Poisson distributed with parameter $\lambda = b - 0.1$. Averaged over 20 runs.

Marks vs Time plot with $\lambda = b$

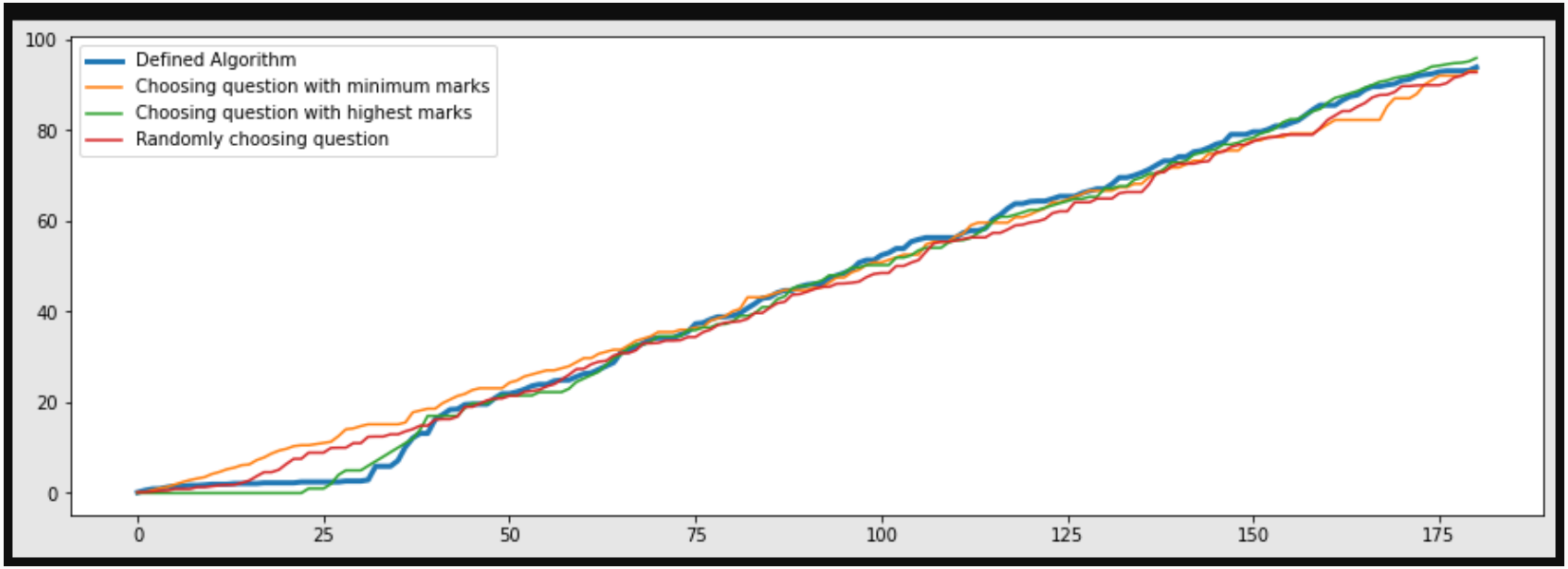

Figure 2: This figure shows how different algorithms obtain rewards over time, when delays are Poisson distributed with parameter $\lambda = b$. Averaged over 20 runs.

Marks vs Time plot with $\lambda = b + 2$

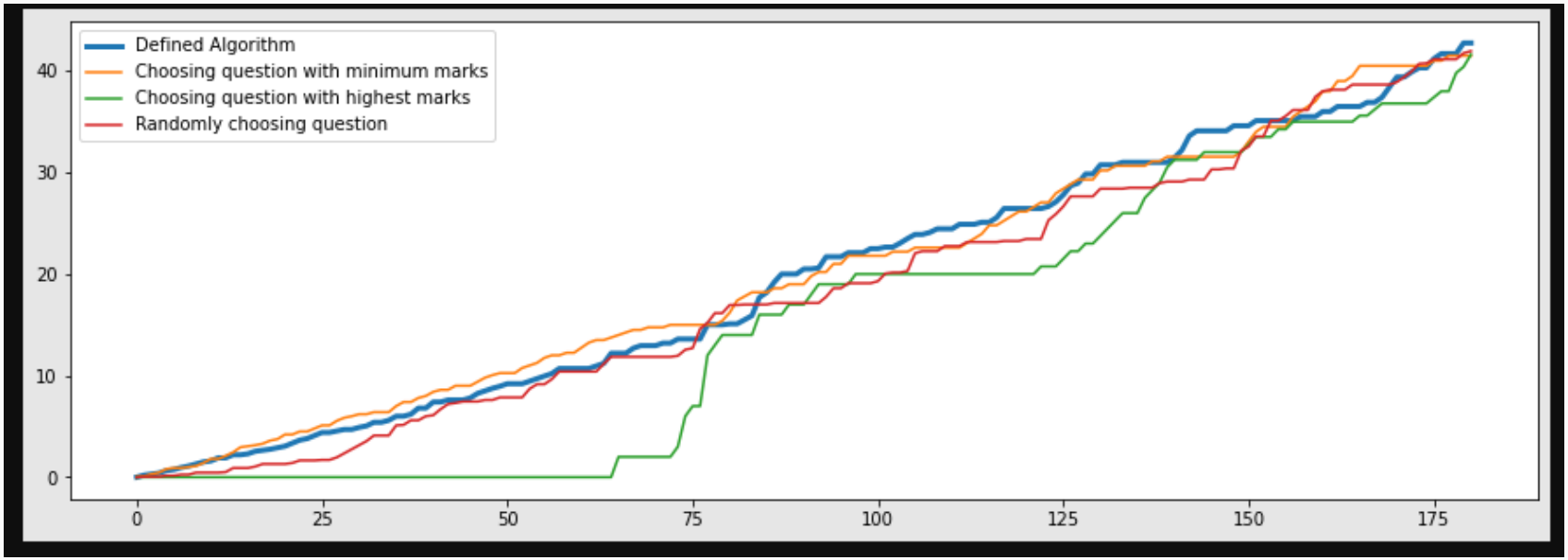

Figure 3: This figure shows how different algorithms obtain rewards over time, when delays are Poisson distributed with parameter $\lambda = b + 2$. Averaged over 20 runs.

Other Related Work

- *Farewell to arms (F2A) (Sharoff et al. 2020)*: Setting with Major arms with stochastic rewards and minor arms that limits delays.
- *Optimal waiting time problem (Lattimore et al., 2014)*: The time spent doesn't affect the rewards in this setting.

Thank you!

Any questions?