

Optimal Posted-Price Mechanism in Microtask Crowdsourcing

Zehong Hu, Jie Zhang

Rolls-Royce@NTU Corporate Lab, School of Computer Science and Engineering
 Nanyang Technological University, Singapore
 huze0004@e.ntu.edu.sg

Abstract

Posted-price mechanisms are widely-adopted to decide the price of tasks in popular microtask crowdsourcing. In this paper, we propose a novel posted-price mechanism which not only outperforms existing mechanisms on performance but also avoids their need of a finite price range. The advantages are achieved by converting the pricing problem into a multi-armed bandit problem and designing an optimal algorithm to exploit the unique features of microtask crowdsourcing. We theoretically show the optimality of our algorithm and prove that the performance upper bound can be achieved without the need of a prior price range. We also conduct extensive experiments using real price data to verify the advantages and practicability of our mechanism.

1 Introduction

Crowdsourcing is an economical method to outsource tasks to online workers [Slivkins and Vaughan, 2014]. Depending on the types of tasks, crowdsourcing takes different forms. Microtask crowdsourcing, one of the most widely-adopted forms, targets small tasks such as labeling images and filling up surveys [Gao *et al.*, 2015]. These tasks usually are repetitive and easy for an individual to perform. In microtask crowdsourcing, a requester often needs to accomplish hundreds of tasks within a given budget. The requester thus posts these tasks on microtask crowdsourcing platforms along with the price for each task. Workers, who are willing to perform the tasks, submit their solutions and will be paid with the prescribed price. Microtask crowdsourcing has become increasingly prevalent in many domains, especially in collecting training data for machine learning algorithms [Difallah *et al.*, 2015; Simpson *et al.*, 2015; Wang and Zhou, 2016].

One of the key challenges in microtask crowdsourcing is to determine the price of tasks properly. Overpricing causes inefficient use of the budget, whereas underpricing may lead to an insufficient number of participating workers [Anari *et al.*, 2014]. Thus, various pricing mechanisms have been proposed [Singla and Krause, 2013; Singer and Mittal, 2013], among which the posted-price mechanism is one of the most attractive branches. It is because the posted-price mechanism only requires workers to make reject-or-accept deci-

sions, which greatly facilitates its practical usage. With these binary decisions, the mechanism can learn the worker model online and accordingly adjusts the price to be optimal.

Nevertheless, existing posted-price mechanisms are still inadequate in two aspects. Firstly, they overlook the unique features of microtask crowdsourcing: the number of workers willing to accept the task is unknown and increases monotonically with the increasing price, whereas the number of workers allowed by the limited budget is accurately known and decreases monotonically. These features can be employed to guide the price adjustment of posted-price mechanisms. Secondly, the performance of existing posted-price mechanisms significantly degrades if the number of possible prices is very large. Thus, the requester is required to input a proper range of prices in advance, which causes much inconvenience.

In this paper, we propose a novel posted-price mechanism to exploit the unique features of microtask crowdsourcing. More specifically, we first convert the pricing problem into an equivalent multi-armed bandit (MAB) problem. Then, we develop an algorithm that offers each coming worker the minimum price at which the anticipated number of workers willing to accept the task approximately equals the number of workers allowed by the budget. Due to the monotonicity features of the two numbers mentioned above, our algorithm is proven to be optimal. In addition, these features ensure that our algorithm will never explore overly high prices and thus does not need to set a price range in advance. To empirically validate the advantages of our mechanism over existing ones, we conduct extensive experiments using three popular worker models as well as the real-world price data collected from MTurk, a widely-adopted microtask crowdsourcing platform. Experimental results confirm that our mechanism achieves almost the same performance as the idealized case where the accurate worker model is known in advance (i.e., the optimal price is used from the very beginning). We also carry out robustness tests to ensure the practicability of our mechanism.

2 Related Work

Existing posted-price mechanisms assume that a worker accepts a task if the offered price is higher than the cost. Without prior knowledge about workers' costs, these mechanisms learn the cost distribution online by counting the acceptance frequency. To maximize requesters' revenue under the uncertainty about workers' costs, different price selection al-

gorithms have been employed. Badanidiyuru *et al.* [2012] discretize the range of workers’ costs into a geometric progression, and use a heuristic policy to pick the price from the progression. Singla and Krause [2013] then propose an improved policy which selects the price with the highest upper confidence bound of the expected revenue under the budget constraint. Badanidiyuru *et al.* [2013] and Agrawal and Devanur [2014] propose to employ linear programming (LP) to choose the optimal price. The LP problem is formulated with the upper and lower confidence bounds of the expected revenue and the budget cost of all possible prices. To summarize, when a worker comes, the existing mechanisms estimate the expected revenue for all possible prices using the learned worker model, to choose the optimal price. However, the learned worker model may be inaccurate, causing the inaccuracy of the estimated revenue. Besides, an essential condition of this approach is that the number of possible prices has to be finite. Its accuracy will become even lower for a larger number of possible prices. Thereby, a finite prior price range must be provided. In contrast, in our mechanism, we only compare the estimated number of workers willing to accept the task with the accurately known number of workers allowed by the limited budget. Without the comparison among the possibly inaccurate estimations of expected revenue, the optimal price computed by our mechanism is more accurate. Meanwhile, since our mechanism only cares about the minimum price at which the two numbers mentioned above are equal, there is no need to have a proper price range given in advance.

Note that there are many other pricing mechanisms proposed in the literature of crowdsourcing. These mechanisms consider a different architecture or scenario. For example, procurement auction determines the price based on workers’ truthful bids about their costs [Singer and Mittal, 2013; Zhang *et al.*, 2014; Chandra *et al.*, 2015]. In microtask crowdsourcing, due to the large number of workers, this architecture will significantly increase the communication burden and easily be affected by workers’ bounded rationality [Rivas, 2015]. In [Badanidiyuru *et al.*, 2012] and [Singla and Krause, 2013], posted-price mechanisms also show competitive and even better performance than procurement auctions. Therefore, we focus on the posted-price mechanism in this paper. Another popular direction of crowdsourcing studies is to determine payment according to the quality of work [Yin and Chen, 2015; Liu and Chen, 2016; Radanovic *et al.*, 2016]. Research in this direction tries to exert more efforts from workers, while our aim is to recruit more workers. These two aspects are in fact complementary.

3 Modeling of Microtask Crowdsourcing

In microtask crowdsourcing, a requester posts tasks along with the price for each task. Workers can accept or reject the task. If they accept and finish one task, they will be paid the offered price. In this section, we formulate the requester and worker models, and study the unique features, optimal price and performance metric of microtask crowdsourcing.

3.1 Requester and Worker Models

The requester gets revenue from the completed tasks. In this paper, we assume that each task has unit value. Thus, the

requester wishes to maximize the number of completed tasks within the given budget B . Assume there are N workers in the market and workers will finish the task if accepting it.

The worker model depicts how workers decide to accept or reject the task. Here, we give three typical worker models which will be utilized in our experiments:

- Private Cost Model [Singla and Krause, 2013] assumes that worker w_i accepts a task only if the offered price p_i is not lower than the cost c_i for performing the task.
- Discrete Choice Model [Gao and Parameswaran, 2014] describes human’s preference for tasks with higher utility. It assumes worker w_i accepts price p_i with the probability $\exp[U(p_i)]/\{\exp[U(p_i)] + M_i\}$, where M_i denotes the effects of other tasks. $U(p_i) = \alpha_i p_i + \beta_i$ denotes the utility of performing the offered task.
- Reference Payment Model [Yin and Chen, 2015] depicts human’s habit to maintain a reference payment level r_i . It assumes worker w_i accepts price p_i with the probability $\{1 + \exp[-\alpha_i - \beta_i(p_i - r_i)]\}^{-1}$, where α_i and β_i denote worker i ’s interest and activeness, respectively.

In this paper, we assume the stochastic arrival of workers. This setting is widely-adopted in the research on microtask crowdsourcing [Singla and Krause, 2013]. Under this assumption, workers arrive one at a time, and the decision parameters of different workers (e.g. the cost c_i) are the same or *i.i.d.* sampled from a same distribution. Thus, the input-output relationship of the worker models mentioned above can be described by a probability function $F(p)$. It denotes the probability at which the coming worker accepts the offered price p . Since prices in practice must be positive, we require the support of $F(p)$ to be $(0, +\infty)$ (i.e. $F(p) = 0$ for $p \leq 0$). Note that the stochastic arrival assumption may be violated in real crowdsourcing markets, and the number of workers N is also always changing. Therefore, in our experiments, we evaluate the robustness of our mechanism in more practical settings where these factors are all considered.

3.2 Unique Features, Optimal Price and Metric

If we choose a price p for all tasks, the requester’s expected revenue will be $U(p) = \min\{N \cdot F(p), B/p\}$. The first item denotes the expected number of workers accepting the price, and the second item represents the budget constraint. Then, we can summarize the features of microtask crowdsourcing:

- Higher prices attract more workers — i.e. the acceptance probability function $F(p)$ is monotonically increasing;
- Higher prices allow fewer workers to be recruited — i.e. the budget constraint B/p is monotonically decreasing;
- The optimal price $p^* = \arg \max_p U(p)$ is the point where the budget equals the costs to recruit all workers willing to accept the price, i.e. $N \cdot F(p^*) = B/p^*$.

Besides, the available prices in practice should be discrete (e.g. 1, 2, ... cents), and the optimal price p^* may be unavailable. Thus, we write the possible prices as an increasing sequence $(p_1 < p_2 < \dots)$, and denote the available optimal price with the optimal subscript $k^* = \arg \max_k U(p_k)$.

We also introduce a performance metric for posted-price mechanisms, termed regret. It denotes the expected revenue

gap between mechanism M and taking the optimal price p_{k^*} :

$$R(N, B) = U(p_{k^*}, B, N) - U(M, B, N) \quad (1)$$

Maximizing requester's revenue is thus equivalent to minimizing the regret. Note that the stochastic combination of different prices may generate higher expected revenue than the optimal price p^* [Badanidiyuru *et al.*, 2013]. Nevertheless, finding the optimal combination requires much more accurate $F(p)$ than identifying the single optimal price. Thus, the mechanisms targeting the optimal combination need to explore sub-optimal prices more, and their performance turns out to be worse than ours in experimental evaluation.

4 Indirect Design of Posted-Price Mechanism

When selecting prices, we face a dilemma between exploring possible better prices and exploiting the current best price. A general idea to resolve this dilemma is to compute and select the current optimal price while considering the uncertainties of the learned worker model. However, it is not easy to properly measure the uncertainties. In addition, budget constraint is another factor that needs attention. The requester accumulates revenue by recruiting workers, and the recruitment stops when the budget is exhausted. The stopping point is in fact not fixed due to the change of selected prices, making it difficult to decide the current optimal price. Thus, designing a mechanism that can also exploit unique features of microtask crowdsourcing becomes even more challenging.

We adopt an indirect approach to design our mechanism. More specifically, instead of directly solving the pricing problem of microtask crowdsourcing, we define an equivalent multi-armed bandit (MAB) problem at first. It not only has the same optimal solution as microtask crowdsourcing but also inherits all the unique features. The only difference is that this equivalent problem has a fixed stopping point which simplifies computing the current optimal. For this equivalent problem, we then develop an algorithm which optimally exploits the unique features of microtask crowdsourcing and use it as the core of our mechanism. For clarity, all the theoretical analysis, including the optimality of our MAB algorithm, the linkage between the two problems on regret and the regret under the infinite price range, will be provided in Section 5.

4.1 Equivalent MAB Problem

We first define the equivalent MAB problem using the notations explained in microtask crowdsourcing. Suppose that we need to repeatedly select a price $p_i \in \{p_1, \dots, p_K\}$ for N times. After selecting the price p_i in round n , we can observe a stochastic signal X_n which follows the Bernoulli distribution. The mean of X_n equals to $F(p_i)$. Meanwhile, the reward for choosing p_i is $\tilde{U}_i = \min\{F_i, C_i\}$, where $F_i = F(p_i)$ and $C_i = B/(N \cdot p_i)$. The rewards are accumulated in background, so we can only know the rewards after finishing all the N selections. In fact, this new problem just equally divides the expected revenue in microtask crowdsourcing into N rounds, i.e. $\tilde{U}_i = U(p_i)/N$. This conversion helps us bypass the difficulty caused by the changing stopping point when trying to exploit the unique features of microtask crowdsourcing. It is also easy to conclude that this equivalent

problem inherits all the unique features of microtask crowdsourcing and reaches the optimal at the optimal price p_{k^*} .

Based on the three unique features, we introduce the following two kinds of possible optimal prices (POPs):

- POP-1 is the price p_k that satisfies $C_k > F_k \geq C_{k+1}$;
- POP-2 is the price p_k that satisfies $F_k \geq C_k > F_{k-1}$.

Here, $k \in \{1, \dots, K\}$. For completeness, we add the conventions that $p_0 = 0$ and $p_{K+1} = +\infty$. The rationale behind introducing POPs lies in the following theorem:

Theorem 1. *There is always one and only one POP-1 or POP-2. The optimal price p_{k^*} must be POP-1 or POP-2.*

Proof of Existence: Due to the monotonicity of $F(p)$ and B/p , $F_i - C_i$ is monotonically increasing when i changes from 1 to K . Meanwhile, the conventions, $p_0 = 0$ and $p_{K+1} = +\infty$, lead to $F_0 - C_0 < 0$ and $F_{K+1} - C_{K+1} > 0$, respectively. Therefore, there must exist k satisfying $F_k - C_k < 0$ and $F_{k+1} - C_{k+1} \geq 0$. In this case, if $F_k \geq C_{k+1}$, then p_k is POP-1; otherwise, p_{k+1} is POP-2. \square

Proof of Uniqueness: Let p_k be POP-1. Then, for $\forall i < k$, $F_i \leq F_k < C_k \leq C_{i+1} < C_i$. For $\forall i > k$, $F_i \geq F_{i-1} \geq F_k \geq C_{k+1} \geq C_i$. Thus, $p_{i \neq k}$ cannot be POP-1 or POP-2. Besides, let p_k be POP-2. Then, for $\forall i < k$, $F_i \leq F_{k-1} < C_k \leq C_{i+1} < C_i$. For $\forall i > k$, $F_i \geq F_k \geq C_k \geq C_{i-1} > C_i$. Thus, $p_{i \neq k}$ also cannot be POP-1 or POP-2. \square

Proof of Optimality: Let p_k be POP-1 or POP-2. From the uniqueness proof, we can have $F_i \leq \min\{C_k, F_k\}$ for $\forall i < k$ and $C_i \leq \min\{C_k, F_k\}$ for $\forall i > k$. Therefore, $\tilde{U}_k \geq \tilde{U}_{i \neq k}$, and p_k equals to the optimal price p_{k^*} . \square

4.2 Optimal MAB Algorithm

Algorithm 1: Optimal MAB Algorithm (OA-MAB)

Input: $n, S = \{p_1, \dots, p_K\}, \mu_i(n)$ for $i = 1 \dots K$

- 1 Search for the minimum POP in S based on $\mu_i(n)$;
- 2 **Output** the current optimal price $p_{k(n)}$ at which:

$$k(n) = \begin{cases} \hat{k} & \text{if } p_{\hat{k}} \text{ is POP-1,} \\ \hat{k} & \text{if } p_{\hat{k}} \text{ is POP-2 and } \frac{l_{\hat{k}}-1}{2} \in \mathbb{N} \\ \hat{k} & \text{if } p_{\hat{k}} \text{ is POP-2, } \frac{l_{\hat{k}}-1}{2} \notin \mathbb{N} \text{ and } b_{\hat{k}-1} < C_{\hat{k}} \\ \hat{k}-1 & \text{if } p_{\hat{k}} \text{ is POP-2, } \frac{l_{\hat{k}}-1}{2} \notin \mathbb{N} \text{ and } b_{\hat{k}-1} \geq C_{\hat{k}} \end{cases}$$

where \hat{k} is the subscript of the minimum POP.

Here we describe the optimal algorithm for the MAB problem, a novel price selection algorithm with the regret proven to match the Lai-Robbins regret lower bound in Section 5.1. To formally describe the algorithm, we need the following notations. Let $N_i(n)$ denote the number of times that price p_i has been selected up to round n . The empirical estimate of the acceptance probability F_i in round n is $\mu_i(n) = \frac{1}{N_i(n)} \sum_{t=1}^n 1\{k(t) = i\} X_t$. Here, $k(t)$ denotes the subscript of the selected price in round t , and we set $\mu_i(n) = 1$ if $N_i(n) = 0$. Besides, we employ the KL-divergence to compute the upper confidence bound of the estimate as:

$$b_i(n, \mu_i, N_i) = \sup\{q \geq \mu_i(n) : N_i(n) \cdot KL[\mu_i(n), q] \leq \log(n) + 3 \log(\log(n))\} \quad (2)$$

with the convention that $b_i(n, \mu_i, 0) = 1$ and $b_i(n, 1, N_i) = 1$. Here, $KL[x, y] = x \log(\frac{x}{y}) + (1-x) \log(\frac{1-x}{1-y})$ denotes the

KL-divergence between two Bernoulli distributions whose means are x and y respectively. We adopt this upper confidence bound formulation because Garivier and Cappé [2011] prove it to outperform other formations in Bernoulli distributions. Furthermore, we define $l_i(n)$ as the number of times that p_i has been identified as not only the minimum possible optimal price (being POP-1 or POP-2) but also POP-2. The pseudo-code of the algorithm is presented in Algorithm 1.

4.3 Posted-Price Mechanism

Our posted-price mechanism is presented in Algorithm 2. The input parameters include the budget B , the number of workers N and the minimum price gap δ_p . Here, δ_p is a parameter of the microtask crowdsourcing platform and denotes the minimum payment unit (e.g. 1 cent). Due to the search of the minimum possible optimal price in our MAB algorithm, the probability of exploring overly large prices exponentially decreases. We call this property as probability decay and will further explain it in Section 5.3. This special property ensures the performance of our MAB algorithm to be unaffected when $K \rightarrow +\infty$. Thus, our mechanism does not need the input of a finite price range, which is required by other existing mechanisms. Furthermore, to strictly ensure the offered price is not higher than the remaining budget B_n , we use B_n to update the available price set S in our mechanism.

Algorithm 2: Optimal Posted-Price Mechanism (OPPM)

```

1 Parameters:  $B, N, \delta_p$ 
2 Initialize: worker counter:  $n = 1$ ; available budget:  $B_n = B$ ;
   the number of recruited workers:  $U = 0$ ;
   the number of available prices:  $K = \lfloor B/\delta_p \rfloor$ ;
   the estimate of  $F$ :  $\mu_i = 1$  for  $i = 1 \dots K$ ;
begin
3   while  $B_n > \delta_p$  and  $n \leq N$  do
4      $K \leftarrow \lfloor B_n/\delta_p \rfloor, S \leftarrow \{\delta_p, 2\delta_p, \dots, K\delta_p\}$ ;
5      $p_k \leftarrow \text{OA-MAB}(n, S, \mu_1, \dots, \mu_K)$ ;
6     Offer price  $p_{k(n)}$  to the coming worker;
7     Observe the decision:  $X_n = 0$  (reject) or 1 (accept);
8     Update:  $U \leftarrow U + X_n; B_{n+1} \leftarrow B_n - p_k X_n$ ;
        $\mu_k \leftarrow (N_k \mu_k + X_n)/(N_k + 1)$ .

```

5 Theoretical Analysis

In this section, we provide extensive theoretical analysis to support our indirect approach to designing the mechanism.

5.1 Optimality of the MAB Algorithm

To prove the optimality of Algorithm 1, we firstly derive the Lai-Robbins regret lower bound of the MAB problem defined in Section 4.1 [Lai and Robbins, 1985]. This classic bound represents the possible best performance that can be achieved by any uniformly good algorithm. We say an algorithm π is uniformly good if its regret is at most $O(\log N)$ for all possible $F(p)$. Besides, we keep $B/N = \text{const}$ when $N \rightarrow +\infty$. This setting ensures the optimal price p_{k^*} to be unchanged and can greatly facilitate the asymptotic analysis.

Theorem 2. *In the MAB problem defined in Section 4.1, any uniformly good algorithm π satisfies:*

$$\liminf_{N \rightarrow \infty} \frac{R^\pi(N)}{\log(N)} = \begin{cases} 0 & p_{k^*} \text{ is POP-1} \\ \frac{C_{k^*} - F_{k^*} - 1}{KL[F_{k^*} - 1, C_{k^*}]} & p_{k^*} \text{ is POP-2} \end{cases} \quad (3)$$

Proof. To prove this theorem, we firstly employ the Theorem 1 in [Graves and Lai, 1997]¹ and can get $\lim_{N \rightarrow \infty} \inf [R^\pi(N)/\log(N)] \geq w(F)$, where $w(F)$ equals the output of the following LP problem:

$$\begin{aligned} \min \quad & \sum_{j=1}^K w_j \cdot [\tilde{U}_{k^*}(F) - \tilde{U}_j(F)] \\ \text{s.t.} \quad & \inf_{\tilde{F} \in Z(F)} \sum_{j \neq k^*} w_j \cdot KL[F_j, \tilde{F}_j] \geq 1 \end{aligned} \quad (4)$$

where $w_j \geq 0$ and \tilde{F} denotes the *bad* distribution that has the same value as F at k^* but provides the largest rewards at another price $\tilde{k} \neq k^*$. Even starting from k^* , any algorithm still needs to explore other prices to distinguish F and \tilde{F} . Otherwise, it may miss the *real* optimal price \tilde{k} if we substitute F with \tilde{F} . All the *bad* distributions form the set $Z(F)$:

$$Z(F) = \{\tilde{F} | \tilde{F}_{k^*} = F_{k^*}, \tilde{U}_{k^*}(\tilde{F}) < \max_k \tilde{U}_k(\tilde{F})\}. \quad (5)$$

Secondly, we solve the LP with the monotonicity of $F(p)$ and $C(p)$ considered. The detailed deduction is similar to those in Theorem 4.1 of [Combes and Proutiere, 2014]. Therefore, we only provide the results here: if p_{k^*} is POP-1, $Z(F) \equiv \emptyset$ and $w(F) = 0$; if p_{k^*} is POP-2, $\inf_{\tilde{F} \in Z(F)} \sum_{j \neq k^*} w_j KL[F_j, \tilde{F}_j] = w_{k^* - 1} \cdot KL[F_{k^* - 1}, C_{k^*}]$ and $w(F) = (C_{k^*} - F_{k^* - 1})/KL[F_{k^* - 1}, C_{k^*}]$. \square

Then, let \tilde{R} be the regret of Algorithm 1. The optimality of our algorithm is guaranteed by the following theorem:

Theorem 3. *The regret upper bound of Algorithm 1 equals the Lai-Robbins regret lower bound, namely*

$$\limsup_{N \rightarrow \infty} \tilde{R}/\log(N) = \liminf_{N \rightarrow \infty} R^\pi/\log(N). \quad (6)$$

To prove this theorem, we firstly decompose the rounds where sub-optimal prices are selected into the following sets:

$$\{n \leq N | k(n) \neq k^*\} \subset A_1 \cup A_2 \cup A_3, \quad (7)$$

where $A_1 = \{n | \hat{k}(n) < k^*\}$ and $A_2 = \{n | \hat{k}(n) > k^*\}$ denote the cases where POP is wrongly identified. $A_3 = \{n | \hat{k}(n) = k^*, k(n) = k^* - 1\}$ represents the cases where POP is correct but the price is wrongly selected. Next, based on this decomposition, we compute the regret of Algorithm 1 as:

Theorem 4. $\tilde{R} \leq \mathbb{E}|A_1| + \mathbb{E}|A_2| + (C_{k^*} - F_{k^* - 1}) \cdot \mathbb{E}|A_3|$, where $|\cdot|$ denotes the size of a set.

Proof. Considering the fact that $\tilde{U}_{k^*} \leq C_{k^*}, \tilde{U}_{k^* - 1} = F_{k^* - 1}$ and $0 < \tilde{U}_j \leq \tilde{U}_{k^*} \leq 1$ hold for $\forall j$, we can have

$$R(B, N) \leq \mathbb{E}|A_1| + \mathbb{E}|A_2| + (C_{k^*} - F_{k^* - 1}) \cdot \mathbb{E}|A_3|$$

where, N_j denotes the total times that p_j is selected. \square

Furthermore, we can bound $\mathbb{E}|A_1|, \mathbb{E}|A_2|$ and $\mathbb{E}|A_3|$ with:

Theorem 5. $\mathbb{E}|A_1| < +\infty$.

¹The Condition (2.14) for Theorem 1 in [Graves and Lai 1997] is equivalent to $F(p(k^* - 1)) > 0$, which is satisfied in all cases.

Proof. According to the settings of Algorithm 1, we derive

$$\left. \begin{aligned} Q_j &\stackrel{\text{def}}{=} \{n|\hat{k} = j\} \\ W_j &\stackrel{\text{def}}{=} \{n|\hat{k} = j, k(n) = j\} \end{aligned} \right\} \Rightarrow |W_j| \geq |Q_j|/2 \quad (8)$$

using the fact that p_j is selected at least when $\frac{l_j-1}{2} \in \mathbb{N}$. For $j < k^*$, to achieve $\hat{k} = j$, μ_j should satisfy $\mu_j \geq C_{j+1}$. Meanwhile, $F_j < C_{j+1}$. Thus, $\mu_j > F_j$ holds for all elements in W_j . According to Theorem 4.1 of [Rajeev *et al.*, 1989], $\mathbb{E}|W_j| < \infty$ and $\mathbb{E}|A_1| = \sum_{j < k^*} \mathbb{E}|Q_j| < +\infty$. \square

Theorem 6. $\mathbb{E}|A_2| = O(\log \log(N))$.

Proof. For any $j > k^*$, we can define the following sets:

$$Q_j \stackrel{\text{def}}{=} \{n|p_j \text{ is POP-1}\} \quad W_j \stackrel{\text{def}}{=} \{n|p_j \text{ is POP-2}\} \quad (9)$$

Furthermore, we can decompose the set W_j as:

$$X_j \stackrel{\text{def}}{=} \{n \in W_j | N_{j-1}(n) \geq l_j(n)/4\} \quad (10)$$

$$Y_j \stackrel{\text{def}}{=} \{n \in W_j | N_{j-1}(n) < l_j(n)/4\}$$

where $l_j(n)$ denotes the times that p_j is identified as the minimum POP and POP-2. Similar as Theorem 5, we can prove $\mathbb{E}|Q_j| < +\infty$ and $\mathbb{E}|X_j| < +\infty$. Then, we can derive

$$Z_j \stackrel{\text{def}}{=} \{n \in W_j | b_{j-1}(n) < C_j\} \Rightarrow |Y_j| \leq 4|Z_j| \quad (11)$$

using the fact that p_j is selected at least $l_j(n)/4$ times under the condition that $(l_j - 1)/2 \notin \mathbb{N}$ and $b_{j-1}(n) < C_j$. Since $C_j < F_{j-1}$, according to Theorem 2 of [Garivier and Cappé, 2011], $\mathbb{E}|Z_j| \leq \gamma \log \log(N)$, where γ is a constant. Thus, $\mathbb{E}|A_2| = \sum_{j > k^*} (\mathbb{E}|Q_j| + \mathbb{E}|W_j|) = O(\log \log(N))$. \square

Theorem 7. If p_{k^*} is POP-1, $\mathbb{E}|A_3| < +\infty$. If p_{k^*} is POP-2, $\mathbb{E}|A_3| = O((1 + \epsilon) \log(N)/KL[F_{k^*-1}, C_{k^*}])$, where ϵ can be any positive number [Garivier and Cappé, 2011].

Proof. A_3 denotes the case where p_{k^*} is identified as POP-2 in Algorithm 1 and $k^* - 1$ is selected because $b_{k^*-1} \geq C_{k^*}$. If k^* is actually POP-1, similar as Theorem 5, we can prove $\mathbb{E}|A_3| < +\infty$. If k^* is POP-2, we can get $F_{k^*-1} < C_{k^*} \leq b_{k^*-1}$. According to Theorem 2 of [Garivier and Cappé, 2011], $\mathbb{E}|A_3| = O((1 + \epsilon) \log(N)/KL[F_{k^*-1}, C_{k^*}])$. \square

Using Theorems 4~7, we can finally conclude Theorem 3.

5.2 Regret of Our Mechanism

Since the stopping point of posted-price mechanisms is not fixed, the regret of our mechanism has a little difference with the regret of the proposed MAB algorithm. Thus, we here derive the regret upper bound of our mechanism.

Theorem 8. The expected regret of our mechanism satisfies:

$$R(N, B) \leq \mathbb{E}|A_1| + (F_{k^*} - F_{k^*-1}) \mathbb{E}|A_3| \\ [K_r \delta_p - F_{k^*} p_{k^*}] \cdot \mathbb{E}|A_2| / p_{k^*} \quad (12)$$

where $K_r > k^*$ denotes the subscript of the highest price explored by our mechanism. Due to the probability decay explained in Section 5.3, K_r may be far smaller than $\lfloor B/\delta_p \rfloor$.

The proof of Theorem 8 is similar to the proof of Lemma 1 in [Singla and Krause, 2013]. Thus, we omit it here. The core idea is to derive the connection between the expected stopping point and the regret. Furthermore, Theorems 4 and 8 show that the regrets of both the algorithm and the posted-price mechanism are linearly proportional to $\mathbb{E}|A_1|$, $\mathbb{E}|A_2|$ and $\mathbb{E}|A_3|$. This linkage explains the rationale behind our indirect approach, which is to first develop the algorithm with the lowest regret in the equivalent MAB problem and then design our mechanism based on the optimal algorithm.

5.3 The Regret under Infinite Price Range

In our mechanism, the price range is set as the budget B . However, in real markets, B is usually very large, leading to a large number of possible prices. To avoid the low efficiency of exploring a large price space, existing mechanisms all require inputting a prior price range. By contrast, the regret of our mechanism is not affected by the infinitely increasing price range. To demonstrate, we analyze the probability distribution of the largest price explored in our mechanism:

Theorem 9 (Probability Decay). There exists a probability $p_d < 1$ which ensures $Pr(K_r = k^* + n) \leq p_d^{n-1}$.

Proof. If Algorithm 1 outputs K_r , p_{K_r} or p_{K_r+1} must be identified as the minimum POP. Thus, $\mu_k < C_k$ must hold for $k^* < \forall k < K_r$. In this case, we can have

$$Pr(K_r = k^* + n) \leq \prod_{k=k^*+1}^{k^*+n-1} Pr(\mu_k \leq C_k) \quad (13)$$

Meanwhile, since the initial value of μ_i is set as 1 in Algorithm 2, to satisfy $\mu_k < C_k$, p_k must be tried for at least one time. Considering the fact that $F_k > C_k$ and the Chernoff-Hoeffding bound [Auer *et al.*, 2002], we can get

$$Pr(\mu_k \leq C_k) \leq e^{-2N_k(F_k - C_k)^2} \leq e^{-2\Delta^2} \quad (14)$$

where $\Delta = F_{k^*+1} - C_{k^*+1}$. Combining Equations 13 and 14, we can conclude Theorem 9 by setting p_d as $\exp(-2\Delta^2)$. \square

Then, we conduct asymptotic analysis of the regret as:

Theorem 10. When $N \rightarrow \infty$ and $B/N = \text{const}$,

$$R(N, B) \leq \begin{cases} O(\log \log N) & p_{k^*} \text{ is POP-1} \\ O\left(\frac{F_{k^*} - F_{k^*-1}}{KL[F_{k^*-1}, C_{k^*}]} \log N\right) & p_{k^*} \text{ is POP-2} \end{cases}$$

Proof. Considering the probability decay, we can bound the expected value of the right-hand side of Equation 12 using

$$\sum_{n=1}^{\infty} K_r \delta_p Pr(K_r = k^* + n) \leq \sum_{n=1}^{\infty} (k^* + n) \delta_p p_d^n \quad (15)$$

The other items in Equation 12 are not affected by K_r . Furthermore, we can compute the above infinite series as

$$\sum_{i=1}^{\infty} (k^* + n) p_d^n = (k^* + 1)Z(p_d) + Z^2(p_d) < +\infty \quad (16)$$

where $Z(p_d) = p_d/(1 - p_d)$. Thus, the effects of the infinite price range ($B \rightarrow \infty$) is bounded by a finite constant. Hence, using Theorems 5~7, we can conclude Theorem 10. \square

Theorem 10 shows that our mechanism not only outperforms the state-of-the-art mechanism, BP-UCB [Singla and Krause, 2013], but also classic MAB algorithms, such as UCB-1 [Auer *et al.*, 2002] and OSUB [Combes and Proutiere, 2014]. Above all, the performance of our mechanism is almost not affected by the number of possible prices, which is attractive for practical usage but unreachable by existing posted-price mechanisms and MAB algorithms.

6 Experimental Evaluation

In this section, we empirically compare our mechanism with state-of-the-art mechanisms including BP-UCB [Singla and Krause, 2013], PD-BwK [Badanidiyuru *et al.*, 2013] and

Table 1: Experimental settings

Expt.	Worker Model	Price Range ¹	B/N
#1	Private Cost Model: $c_i \sim U[5, 200]$ ²	[5, 200]	40
#2	Discrete Choice Model: $\alpha_i = 1/15, \beta_i = 0.39, M_i = 2000$	[1, 200]	30
#3	Reference Payment Model: $\alpha_i \in \{0, 1, 3\}, \beta_i \in \{0, 1, 3\}, r_i \in \{20, 60, 120\}$, Evenly Distributed	[1, 200]	70
#4	The data (Fig. 1e) collected using MTurk-Tracker [Difallah <i>et al.</i> , 2015]	[1, 100]	10

¹ The price range is set for the benchmark mechanisms.

² Here, U denotes the uniform distribution.

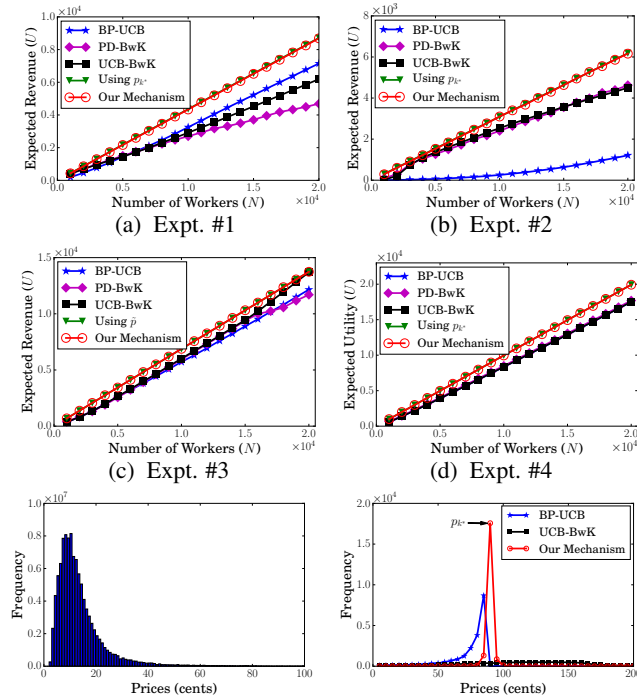


Figure 1: Experiment results on different worker models

UCB-BwK [Agrawal and Devanur, 2014]. The idealized case with the optimal price p_{k^*} known in advance and always selected is also employed for comparison. The testbeds are built based on the three worker models mentioned in Section 3.1 and the real-world price data collected from MTurk. In our experiments, workers come sequentially. The mechanism offers a price for each worker. The worker decides to accept or reject the price according to the worker model. After observing worker’s decision, the mechanism updates the price offered to the next worker. The settings of our experiments are shown in Table 1. The price unit in all experiments is the cent. The expected revenue is estimated with the mean of 100 runs. Besides, we keep the ratio between the budget B and the number of workers N fixed so as to ensure the optimal price p_{k^*} to be unchanged and thus a fair comparison.

The results are shown in Fig. 1(a-d), respectively. We can conclude that our mechanism remarkably outperforms state-of-the-art mechanisms. It even achieves the same performance as the idealized case where p_{k^*} is known and always selected. To explain the reason for the optimal performance

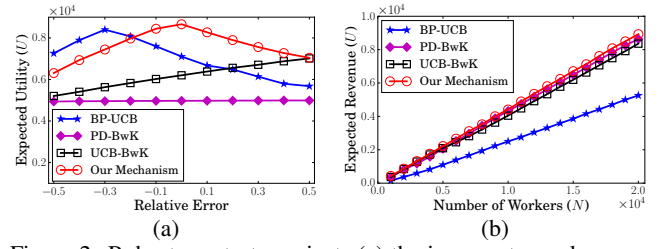


Figure 2: Robustness tests against: (a) the inaccurate worker number; (b) the non-stochastic arrival of workers

of our mechanism, we compare the selected price distributions in Fig. 1f. We can observe an overstay of BP-UCB at the current optimal price, which is caused by an overmuch exploitation on the current estimate of $F(p)$. UCB-BwK, on the other hand, shows an overmuch exploration on the possible better prices. By contrast, our mechanism performs much better in balancing exploitation and exploration. It can not only move quickly to p_{k^*} , but also accurately stop exploration around p_{k^*} . Note that PD-BwK behaves similarly as UCB-BwK and thus is not included for comparison in Fig. 1f.

To further verify the practical usability of our mechanism, we conduct robustness analysis by considering two abnormal cases where the assumptions used for design are violated. Here, we use the private cost model as the testbed, and the settings are the same as Expt. #1. In Fig. 2a, we compare the performance of different mechanisms when the number of workers N is not accurately given. The real number of workers N is 20000, and the relative error is calculated as $(\bar{N} - N)/N$, where \bar{N} is the value offered to the mechanism. The comparison shows that our mechanism has distinct advantages over state-of-the-art mechanisms in this abnormal case. Fig. 2b presents the comparison of different mechanisms in a non-stochastic setting where two groups with the private cost uniformly distributed in $[5, 100)$ and $[100, 200]$ respectively arrive one after another. This setting can explain the phenomenon that working at night may cost workers more than at day. The results show that the performance advantage of our mechanism is kept in this non-stochastic setting.

7 Conclusion and Future Work

In this paper, we propose an optimal posted-price mechanism for microtask crowdsourcing. Compared with existing mechanisms, our mechanism not only has better performance but also requires fewer inputs. To demonstrate the advantages, we firstly prove the optimality of our algorithm that its regret matches the Lai-Robbins regret lower bound. This lower bound applies to any possible algorithms and denotes the best performance that can be achieved. Then, we prove that the regret of our mechanism is not affected by the infinite price range. Besides, the empirical results on various worker models and the real price data collected from MTurk also verify the advantages of our mechanism. For future work, we will improve our mechanism in two aspects. Firstly, when searching for the optimal price, our mechanism needs to try the price one-by-one. Adaptively changing the searching step can enhance efficiency. Secondly, the work quality can be further considered by designing a more practical payment scheme consisting of base salary and bonus-penalty.

Acknowledgments

This work was conducted within Rolls-Royce@NTU Corporate Lab with support from the National Research Foundation (NRF) Singapore under the Corp Lab@University Scheme.

References

- [Agrawal and Devanur, 2014] Shipra Agrawal and Nikhil R Devanur. Bandits with concave rewards and convex knapsacks. In *Proc. of ACM EC*, 2014.
- [Anari *et al.*, 2014] Nima Anari, Gagan Goel, and Afshin Nikzad. Mechanism design for crowdsourcing: An optimal $1-1/e$ competitive budget-feasible mechanism for large markets. In *Proc. of FOCS*, 2014.
- [Auer *et al.*, 2002] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256, 2002.
- [Badanidiyuru *et al.*, 2012] Ashwinkumar Badanidiyuru, Robert Kleinberg, and Yaron Singer. Learning on a budget: posted price mechanisms for online procurement. In *Proceedings of ACM EC*, 2012.
- [Badanidiyuru *et al.*, 2013] Ashwinkumar Badanidiyuru, Robert Kleinberg, and Aleksandrs Slivkins. Bandits with knapsacks. In *Proc. of FOCS*, 2013.
- [Chandra *et al.*, 2015] Praphul Chandra, Yadati Narahari, Debmalya Mandal, and Prasenjit Dey. Novel mechanisms for online crowdsourcing with unreliable, strategic agents. In *Proc. of AAAI*, 2015.
- [Combes and Proutiere, 2014] Richard Combes and Alexandre Proutiere. Unimodal bandits: Regret lower bounds and optimal algorithms. In *Proc. of ICML*, 2014.
- [Difallah *et al.*, 2015] Djellel Eddine Difallah, Michele Catasta, Gianluca Demartini, Panagiotis G Ipeirotis, and Philippe Cudré-Mauroux. The dynamics of micro-task crowdsourcing: The case of amazon mturk. In *Proc. of WWW*, 2015.
- [Gao and Parameswaran, 2014] Yihan Gao and Aditya Parameswaran. Finish them!: Pricing algorithms for human computation. *Proceedings of the VLDB Endowment*, 7(14):1965–1976, 2014.
- [Gao *et al.*, 2015] Yang Gao, Yan Chen, and KJ Ray Liu. On cost-effective incentive mechanisms in microtask crowdsourcing. *IEEE Transactions on Computational Intelligence and AI in Games*, 7(1):3–15, 2015.
- [Garivier and Cappé, 2011] Aurélien Garivier and Olivier Cappé. The kl-ucb algorithm for bounded stochastic bandits and beyond. In *Proc. of COLT*, 2011.
- [Graves and Lai, 1997] Todd L Graves and Tze Leung Lai. Asymptotically efficient adaptive choice of control laws in controlled markov chains. *SIAM journal on control and optimization*, 35(3):715–743, 1997.
- [Lai and Robbins, 1985] Tze Leung Lai and Herbert Robbins. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22, 1985.
- [Liu and Chen, 2016] Yang Liu and Yiling Chen. Learning to incentivize: Eliciting effort via output agreement. In *Proc. of IJCAI*, 2016.
- [Radanovic *et al.*, 2016] Goran Radanovic, Boi Faltings, and Radu Jurca. Incentives for effort in crowdsourcing using the peer truth serum. *ACM Transactions on Intelligent Systems and Technology*, 7(4):48, 2016.
- [Rajeev *et al.*, 1989] A Rajeev, Demosthenis Teneketzis, and Venkatachalam Anantharam. Asymptotically efficient adaptive allocation schemes for controlled iid processes: Finite parameter space. *IEEE Transactions on Automatic Control*, 34(3), 1989.
- [Rivas, 2015] Javier Rivas. Mechanism design and bounded rationality: The case of type misreporting. *Mathematical Social Sciences*, 78:6 – 13, 2015.
- [Simpson *et al.*, 2015] Edwin D Simpson, Matteo Venanzi, Steven Reece, Pushmeet Kohli, John Guiver, Stephen J Roberts, and Nicholas R Jennings. Language understanding in the wild: Combining crowdsourcing and machine learning. In *Proc. of WWW*, 2015.
- [Singer and Mittal, 2013] Yaron Singer and Manas Mittal. Pricing mechanisms for crowdsourcing markets. In *Proc. of WWW*, 2013.
- [Singla and Krause, 2013] Adish Singla and Andreas Krause. Truthful incentives in crowdsourcing tasks using regret minimization mechanisms. In *Proc. of WWW*, 2013.
- [Slivkins and Vaughan, 2014] Aleksandrs Slivkins and Jennifer Wortman Vaughan. Online decision making in crowdsourcing markets: Theoretical challenges. *ACM SIGecom Exchanges*, 12(2):4–23, 2014.
- [Wang and Zhou, 2016] Lu Wang and Zhi-Hua Zhou. Learning by crowdsourcing. In *Proc. of IJCAI*, 2016.
- [Yin and Chen, 2015] Ming Yin and Yiling Chen. Bonus or not? learn to reward in crowdsourcing. In *Proc. of IJCAI*, 2015.
- [Zhang *et al.*, 2014] Xinglin Zhang, Zheng Yang, Zimu Zhou, Haibin Cai, Lei Chen, and Xiangyang Li. Free market of crowdsourcing: Incentive mechanism design for mobile sensing. *IEEE transactions on parallel and distributed systems*, 25(12):3190–3200, 2014.