Mathematical Modeling of Crowdsourcing Systems: Incentive Mechanism and Rating System Design

Hong Xie,    John C.S. Lui
Department of Computer Science & Engineering, The Chinese University of Hong Kong, Hong Kong, {hxie,cslui}@cse.cuhk.edu.hk

Wenjie Jiang
Google Inc, USA, wenjie@google.com

Abstract—Crowdsourcing systems like Yahoo!Answers, Amazon Mechanical Turk, and Google Helpouts, etc., have seen an increasing prevalence in the past few years. The participation of users, high quality solutions, and a fair rating system are critical to the revenue of a crowdsourcing system. In this paper, we design a class of simple but effective incentive mechanisms to attract users participating, and providing high quality solutions. Our incentive mechanism consists of a task bundling scheme and a rating system, and pay workers according to solution ratings from requesters. We also propose a probabilistic model to capture various human factors like biases in rating, and we quantify its impact on the incentive mechanism, which is shown to be highly robust. We develop a model to characterize the design space of a class of commonly used rating systems – threshold based rating systems. We quantify the impact of such rating systems and the bundling scheme on the incentive mechanism.

Keywords—crowdsourcing; incentive mechanism; bundling; rating system;

I. INTRODUCTION

Over the past few years, we have witnessed an increasing importance and prevalence of crowdsourcing services [1]. Many well-known Internet companies and services are using crowdsourcing services, e.g., Wikipedia [2], Yahoo!Answers [3], Amazon Mechanical Turk [4], Google Helpouts [5], etc. By eliciting collective intelligence of the crowd, crowdsourcing can benefit companies in production [6], product design [7], etc, or to reduce overhead and improve profit [1]. In addition to business applications, crowdsourcing systems like Amazon Mechanical Turk, have become a novel platform to conduct experimentation for social science [8], where a large amount of human behavioral data can be collected efficiently, which in turn may improve the studies of social science. Furthermore, crowdsourcing has evolved as an efficient paradigm to shed new light on many challenging problems like anomaly detection [9].

Typically, a crowdsourcing system operates with three basic components: Users, tasks and rewards. Users are classified into requesters and workers. A user can be a requester or a worker, and in some cases, a user can be a requester/worker at the same time. Requesters outsource tasks to a crowdsourcing system and at the same time, associate each task with certain rewards, which will be granted to the workers who solve the task. Workers, on the other hand, work on the assigned tasks and then receive the reward. The reward can be in form of money [4], entertainment [10] or altruism [3], etc.

The participation of users (requesters and workers), high quality solutions, and a fair rating system are critical to the revenue of a crowdsourcing system. There are at least two challenges in achieving this objective. The first one is how to set the proper reward so to attract workers and encourage requesters to participate. Obviously, the higher the reward, the system will attract more workers. However, this also implies higher overhead for requesters and will discourage some requesters to shift to another crowdsourcing system. The second challenges is how to design a fair rating system to evaluate workers’ contribution, and pay workers according to their contributions so they provide high quality solutions. This is challenging, because many human factors like bias can influence requesters in evaluating solutions, which may lead to biased solution evaluation that impairs the incentive of workers. In this paper, we address the above challenges. Our contributions are:

• We design a class of simple but effective incentive mechanisms, which consist of a task bundling scheme and a rating system, and pay workers according to solution ratings from requesters. We derive the minimum amount of rewards to guarantee workers providing high-quality solutions.

• We show that our proposed family of incentive mechanisms is highly robust against various human factors like bias in rating, via proposing a probabilistic model to capture such human factors, as well as quantifying their impact on the incentive mechanism.

• We develop a model to characterize the design space of a class of commonly users rating systems – threshold based rating system. We quantify the impact of such rating systems, and the bundling scheme on reducing requesters’ overhead.

This is the outline of the paper. In §II, we present the foundation of game-theoretic model, as well as our incentive mechanism. In §III we present a probabilistic model to capture human factors in rating. In §IV, we present a model to characterize the design space of rating systems. Related work is given in §V and §VI concludes.

II. MODEL

We present the system model and the design of our incentive mechanism. We then analyze the incentive mechanism via the game-theoretic technique, showing its high effectiveness.
A. System Model

Consider a crowdsourcing system which categorizes tasks into \( K \) types. For example, in “Yahoo! Answers”, it contains 25 types of tasks ranging from “Health” to “Travel” [11]. Users of a crowdsourcing system are classified into requesters and workers. A user can be a requester, or a worker, or in some cases, a requester/worker at the same time. Requesters outsource tasks to a crowdsourcing system and at the same time, associate each type \( k \) task with a reward of \( r_k \), \( k \in \{ 1, \ldots, K \} \). The reward \( r_k \) will be granted to the workers who make contributions to the corresponding task. For a type \( k \) task, a requester also pays \( T_k \) to the crowdsourcing system as service charge. We focus on one task type in our analysis, and it can be generally applied to all task types. We thus drop the subscript.

A task is assigned to only one worker. We capture the scenario that a task requires many workers, in which the task can be divided into many copies and each copy requires one worker. Some service allows requesters to pick workers, such as Google Helpouts, while others practice the other way, such as [12]. We emphasize that our model support both cases. A worker can exert \( L \geq 2 \) levels of effort \( L = \{ 1, \ldots, L \} \) in solving a task, which results in \( L \) levels of contribution \( C_L = \{ C_1, \ldots, C_L \} \). We assume that \( C_L > C_{L-1} > \ldots > C_1 \), where \( C_1 > C_j \) represents that contribution \( C_i \) is higher than \( C_j \). For the ease of presentation, we use \( \{ C_1, \ldots, C_L \} \) to denote the action set for workers. When a worker acts with \( C_i \), it implies the worker exerts the \( i \)-th level of effort to solve the task. The cost in making a \( C_j \) contribution to a task is denoted as \( c_j \), where \( c_L > c_{L-1} > \ldots > c_1 = 0 \). Here, we use \( c_1 = 0 \) to model the the “free-riding” scenario from workers. For a task, if a worker exerts \( C_j \) to provide a solution, then it brings a benefit of \( V_j \) to a requester, where \( V_L > V_{L-1} > \ldots > V_1 = 0 \). Again, \( V_j = 0 \) models free-riding because \( c_1 = 0 \). We require \( V_L > r+T \), which induces incentives for requesters to participate. If \( V_\kappa < r+T, \forall \kappa < L \), means that level \( \kappa \) contribution is not incentive-compatible. The objective of this work is show how to incentivize workers to exert \( C_L \), the highest possible contribution.

B. Incentive Mechanism Design

We present a class of incentive mechanisms to address the above problem, which consists of two key components: a bundling scheme, and a rating system. Tasks are completed via transactions under a task bundling scheme, which can be precisely described as follows. When posting a task, a requester submits its reward \( r \) and service charge \( T \) to the administrator. The administrator bundles \( n \geq 1 \) tasks of similar type. Once a task is solved, a worker submits its solution to the administrator. After all tasks within a bundle are solved, the administrator delivers them to the corresponding requesters. Requesters provide feedbacks on their solutions to the administrator in the form of solution rating. In particular, requesters rate solutions such that a rating \( i \) indicates that the solution was solved with the \( C_i \) level of contribution. Note that solutions are independent, and a requester can only express ratings of solution to her task. Finally, when all feedback ratings for a bundle are collected, the crowdsourcing administrator divides the total reward, which is \( nr \), to all workers engaged in that bundle. Specifically, the worker who receives the highest rating takes all the reward. When there is a tie, the crowdsourcing administrator divides the total \( nr \) evenly among the tie. We call this reward scheme as “winner takes all scheme”.

Remark. We call the above bundling scheme the \( n \)-bundling scheme. One can observe that the administrator will not withhold the reward. A requester under our bundling scheme will not benefit by intentionally providing false ratings (since the reward was given to the administrator when she submits the task, and the reward will not be returned to the requester). The challenge is how to incentivize participating workers make high quality contribution.

C. Formulating the n-player Game

With the proposed incentive mechanism (winner takes all scheme), we apply the game-theoretic technique to derive the desired amount of reward such that workers are guaranteed to make high quality contributions.

Consider an \( n \)-bundling scheme, we formulate an \( n \)-player game to capture the strategic behavior of workers who participate in the same task bundle. Specifically, players of this game are \( n \) workers engaged in a bundle and we denote them as \( w_1, \ldots, w_n \). The action set for a player is \( \{ C_1, \ldots, C_L \} \). We use the notation \( s_j \) to represent the strategic action of worker \( w_j \). Let \( s_j = [s_k]_{K \neq j} \) denote a vector of strategic actions for all players except \( w_j \). We use the notation \( u_j(s_j, s_{-j} | r) \) to denote the utility for player \( w_j \) under strategy profile \( (s_j, s_{-j}) \), which is defined as the reward minus cost. Formally, the utility of player \( w_j \) can be expressed as:

\[
u_j(s_j, s_{-j} | r) = R_j(s_j, s_{-j} | r) - c_{s_j}, \quad \text{if } s_j = C_{s_j}, \tag{1}\]

where \( R_j(s_j, s_{-j} | r) \) is the reward under strategy profile \( (s_j, s_{-j}) \). We express \( R_j(s_j, s_{-j} | r) \) under the winner takes all scheme as

\[
R_j(s_j, s_{-j} | r) = \begin{cases} 
\frac{nr}{\sum_{\kappa = 1}^{r} 1} , & \text{if } s_j = \max_{\kappa} s_{\kappa} \\
0 , & \text{otherwise}.
\end{cases}
\tag{2}
\]

Our objective is to guarantee each player in the above game plays \( C_L \). One sufficient condition to achieve this: the strategy profile \( (C_L, \ldots, C_L) \) is a unique “Nash Equilibrium”.

Definition II.1. The desired Nash equilibrium \( (C_L, \ldots, C_L) \) is a strategy profile that all workers solve the task at \( C_L \) level.

We present a formal way to show the uniqueness of the desired Nash equilibrium \( (C_L, \ldots, C_L) \) in the following lemma. We present strictly dominated strategy, which a player never plays.

Definition II.2. A strategy \( s_i \in C_{s_i} \) is a strictly dominated strategy for player \( i \) if there exists some \( s_i^* \in C_{s_i} \) such that \( u_i(s_i^*, s_{-i} | r) > u_i(s_i, s_{-i} | r) \), for all \( s_{-i} \in C_{s_{-i}} \).

Lemma II.1 (Uniqueness [13]). Consider a pure Nash equilibrium \( (s_1^*, \ldots, s_n^*) \) for the \( n \)-player game. If iterated elimination of strictly dominated strategies eliminates all but the strategies \( (s_1^*, \ldots, s_n^*) \), then it is a unique Nash equilibrium.

In the following, we derive the desired amount of reward to guarantee the above sufficient condition. We show that for the 1-bundling scheme, it is impossible to achieve this
Lemma II.2 (1-bundling scheme). Under the 1-bundling scheme, workers will always exert $C_1$ contribution, or “free-riding”, no matter what reward we set.

Remark. The above lemma states that there is no way to incentivize the single worker to contribute at a higher level. In the following, we show how to eliminate this undesirable result by bundling more than one task.

2-bundling scheme. We show that under this bundling scheme, we can incentivize high quality contributions via setting a proper reward. To illustrate, consider an example with three levels of contribution $L = 3$. We express the utility matrix for this example in Table II, in which one can observe that the strategy profile $(C_3, C_3)$ is a unique Nash Equilibrium if and only if $r > c_3 - c_1 = c_3$. We generalize this positive result for $L$ levels of contribution in the following lemma. To present the lemma, we need the following definition.

Definition II.3. We define the “critical value” $\zeta$ as the minimum amount of reward to incentivize high quality contribution.

Lemma II.3 (2-bundling). Consider a 2-bundling scheme and $L$ levels of contribution. The strategy profile $(C_L, C_L)$ is a unique Nash Equilibrium if and only if $r > \zeta = c_L$.

Proof: Please refer to [14] for derivation.

Remark. Since $(C_L, C_L)$ is a unique Nash equilibrium, workers will make high quality contribution $C_L$ provided that they do not collude. If collusion is allowed, the best strategy for them is $(C_1, C_1)$ so that everyone freerides. One way to eliminate this undesirable result is by bundling more tasks so to guarantee that at least one worker will not collude. However, we prove that increasing the bundle size does not increase the cost for requesters, in the following theorem.

Theorem II.1. Consider a $n$-bundling scheme with $n \geq 2$ and $L$ levels of contribution. The strategy profile $(C_L, \ldots, C_L)$ is a unique Nash Equilibrium if and only if $r > \zeta = c_L$.

Proof: This proof is similar to that of Lemma II.3. All results in this section assume that requesters can perfectly correct ratings to indicate workers’ contribution level. In the next section, we analyze how human factors like preferences or biases may influence the design of the incentive system.

III. HUMAN FACTORS IN SOLUTION RATING

We present a model to capture various human factors in solution rating, and quantify their impact on the incentive mechanism as well. Specifically, one can incentivize workers to make high quality contribution $C_L$ by setting a proper reward even in the presence of human factors in rating. Besides, the complexity in computing the desired reward is $\Theta(nL^2)$.

A. Model for Human Factors in Ratings

We like to point out that the above model assumes an idealistic setting: requesters can perfectly express the correct rating to indicate workers’ contribution level. Let us now extend the above model to a realistic scenario by accommodating various important human factors in rating such as bias, preference, leniency, etc [15], [16]. Specifically, it may happen that a requester is unsatisfied with a high quality solution, or satisfied with a low quality solution. These result in a low rating on a high quality contribution, or a high rating on a low quality contribution. We call such ratings “erroneous ratings”.

We present a probabilistic model to capture the above human factors in solution ratings. To illustrate, consider a worker who exerted $C_L$ or a high quality contribution. Due to human factors, a worker may receive a rating ranging from 1 to $L$. We use the notation $\alpha_{i,j} \in [0, 1]$ to represent the probability that this worker receives a rating $j$, i.e., the requester evaluates his solution as a $C_j$ contribution. Mathematically, we have $\alpha_{i,j} = \Pr[\text{evaluated as } C_j \text{ contribution} | \text{C_L contribution}]$, where $\sum_{j=1}^{L} \alpha_{i,j} = 1$. Then $\{\alpha_{i,1}, \ldots, \alpha_{i,L}\}$ forms a probability distribution. The mean and variance of this distribution measures how erroneous a requester’s rating can be. For example a mean of $L$ and variance of 0 implies that the requesters are perfectly accurate in rating. Smaller variance implies a higher confidence of the result. Similarly, we define $\alpha_{i,j}$ as $\Pr[\text{evaluated as } C_j \text{ contribution} | \text{C_i contribution}]$. When $\alpha_{i,j} = 1$ for $i \in \{1, \ldots, L\}$, this implies requesters have perfect evaluation on the contribution by workers. Otherwise, $\alpha_{i,j}$ can represent different degrees of variability in evaluation. Note that all $\alpha_{i,j}, \forall i, j$ composes a matrix, which presents the probability of all possible rating outcome. We use $\alpha$ to represent this matrix, or formally

$$\alpha = \begin{bmatrix} \alpha_{1,1} & \cdots & \alpha_{1,L} \\ \vdots & \ddots & \vdots \\ \alpha_{L,1} & \cdots & \alpha_{L,L} \end{bmatrix}.$$
We state two natural properties that a variability matrix \( \alpha \) should satisfy. First, consider a \( C_L \) solution. Intuitively, when contribution \( C_i \) is higher than \( C_j \) (or \( C_i > C_j \)), the probability that a requester misjudges this solution as \( C_i \) should be larger than misjudging it as a \( C_j \) one, or formally \( \alpha_{L,i} > \alpha_{L,j} \). Generalizing this statement, we obtain the first natural property, namely, row diagonally dominated and row singly peaked, which we define as follows.

**Definition III.1 (Row/Column Diagonally Dominated).** We say \( \alpha \) is a row/column diagonally dominated matrix, if each row/column has a maximum entry at its diagonal entry.

**Definition III.2 (Row/Column Singly Peaked).** We say \( \alpha \) is row/column singly peaked, if the entry of each row/column strictly increasing/decreasing prior/after the diagonal entry.

Secondly, the probability that a requester misjudges a \( C_i \) solution as a \( C_L \) one should be larger than misjudging it as a \( C_j \) solution as a \( C_L \) one. Generalizing this statement, we obtain another property: column diagonally dominated and column singly peaked. We summarize these two properties as follows.

**Proposition III.1.** The variability matrix \( \alpha \) has the properties: row/column diagonally dominated, and singly peaked.

Consider an example of three levels of contribution \( L = 3 \), one possible instance of variability matrix \( \alpha \) could be

\[
\alpha = \begin{bmatrix}
0.719 & 0.216 & 0.065 \\
0.188 & 0.625 & 0.187 \\
0.065 & 0.216 & 0.719
\end{bmatrix}.
\]

One can easily check that the above example satisfies the properties specified in proposition III.1. It is important to note that we add no symmetric constraints on the variability matrix \( \alpha \). In other words \( \alpha \) can be an asymmetric matrix.

**B. Deriving the Critical Value under Human Factors**

We now explore the impact of human factors, i.e., variability matrix \( \alpha \) on our proposed incentive mechanism. Specifically, we prove the existence and uniqueness of the desired Nash equilibrium \( (C_L, \ldots, C_L) \). And we quantify the impact of the variability matrix \( \alpha \) on the critical value. We state these results in the following theorem.

**Theorem III.1.** Consider a \( n \)-bundling scheme with \( n \geq 2 \), \( L \) levels of contribution, and a variability matrix \( \alpha \). The strategy profile \( (C_L, \ldots, C_L) \) is a unique Nash equilibrium iff

\[
r > \mathcal{L} = \max_k \left\{ \frac{c_L - c_k}{1 - \sum_{l=1}^{L} \sum_{k=1}^{n-1} \alpha_{L,k} \alpha_k, j \alpha_{L,l}^{\ell-1}} \right\}.
\]

**Proof:** Please refer to [14] for derivation.

**Remark.** It implies that one can incentivize workers to make high quality contribution \( C_L \) by setting a proper reward even in the presence of erroneous ratings. Besides, the complexity in computing the desired reward is \( \Theta(nL^2) \).

We show some illustrating numerical examples for the critical value in Table III, where we consider three levels of contribution \( L = 3 \) and a variability matrix \( \alpha \) specified in Equation (3). One can observe that when there is no erroneous ratings (or no variability), as we vary the bundle size from two to four, the critical value remains at \( c_3 \). While in the presence of erroneous ratings (or variability), the critical value decreases from \( 1.256c_3 \) to \( 1.106c_3 \). This implies that requesters need to pay more, due to variability. Also, as we increase bundle size, we decrease the critical value.

<table>
<thead>
<tr>
<th>( n )</th>
<th>( \mathcal{L} ) (no variability)</th>
<th>( \mathcal{L} ) (variability)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>( c_3 )</td>
<td>1.256c_3</td>
</tr>
<tr>
<td>3</td>
<td>( c_3 )</td>
<td>1.106c_3</td>
</tr>
<tr>
<td>4</td>
<td>( c_3 )</td>
<td>( c_3 )</td>
</tr>
</tbody>
</table>

Our model thus far considers a rating system with a small number of contribution level \( L \). When \( L \) is large, it may be difficult for a requester to express a rating accurately, i.e., the time or cognitive cost will be high [17]. How to design a proper rating system to address this challenge? How different designs of rating system may influence the incentive mechanism?

**IV. Modeling Rating Systems**

We present a model to characterize the design space a class of commonly used rating systems – threshold based rating system, as well as quantify their impact on requesters’ overhead. We find out that in the idealistic scenario without erroneous ratings, a binary rating system, i.e., two rating points indicating satisfied or not, is sufficient to incentivize workers to exert their highest contributions. While in the presence of erroneous ratings due to human factors, a rating system with five rating points would be proper.

**A. Threshold Based Rating Systems**

Many crowdsourcing services adopt threshold based rating systems, where the quality of a solution below a “threshold” receives the lowest rating, which may incur some warnings or punishments, etc, to a worker. We develop a model to characterize the design space of such rating systems. Our objective is to quantify its impact on the requesters’ overhead, as well as illustrate how to model and analyze a rating system.

A threshold based rating system is a triplet \( (L', C_L, \mathcal{R}(-)) \), where \( L' = \{1, \ldots, L'\} \) represents an \( L' \)-level cardinal rating metric such that \( 2 \leq L' \leq L \). And \( C_L = \{C_1, \ldots, C_L\} \) denotes a set of potential contribution levels. The notation \( \mathcal{R}(-) \) represents a rating function which maps any given contribution \( C_l \in C_L \) to a specific rating \( j \in L' \), or mathematically \( \mathcal{R}(-): C_L \to L' \). The rating function \( \mathcal{R}(-) \) maps the highest contribution \( C_l \) to the highest rating \( L' \), and maps the second highest contribution \( C_{l-1} \) to the second highest rating \( L' - 1 \). This process continues until the threshold contribution level \( L' + 1 \) is reached, which is mapped to the lowest rating 1, and all the remaining levels of contribution are mapped to rating 1. We formally express this rating function \( \mathcal{R}(-) \) as

\[
\mathcal{R}(C_k) = \begin{cases}
    k - L + L', & k > L - L' + 1 \\
    L - L' + 1, & k \leq L - L' + 1
\end{cases}
\]

where \( C_{L-L'+1} \) is the threshold contribution, or the minimum requirement on solutions. We show some illustrating examples in Table IV, where we consider four levels of contribution.
L = 4, and we vary the number of rating points L' from 2 to 4. We show the corresponding rating function \( \mathcal{R}(\cdot) \). One can see that when \( L' = 2 \), we have \( \mathcal{R}(C_1) = \mathcal{R}(C_2) = \mathcal{R}(C_3) = 1, \) and \( \mathcal{R}(C_4) = 2 \).

<table>
<thead>
<tr>
<th>( L' )</th>
<th>( C_1 )</th>
<th>( C_2 )</th>
<th>( C_3 )</th>
<th>( C_4 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \mathcal{R}(\cdot) )</td>
<td>4</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>( \mathcal{R}(\cdot) )</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>( \mathcal{R}(\cdot) )</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

One can observe that the rating system introduced in Section II is a special case of \( L' = 2 \). When \( L' = 2 \), we obtain a binary rating system. One can vary the value of \( L' \) to obtain a rating system with different complexity, i.e., the number of rating points. We next quantify the impact of threshold based rating systems on the incentive mechanism.

### B. Derivation of the Critical Value

We seek to quantify the impact of threshold based rating systems on the critical value. We explore the setting without/with erroneous ratings respectively.

We first explore the setting without erroneous ratings. We extend the \( n \)-player game specified in Section II to accommodate the threshold based rating system. We rewrite the reward function derived in Equation (2), as

\[
R_j(s_j, s_{-j}| r) = \begin{cases} 
\sum_{k=1}^{\kappa} \mathbf{1}(\mathcal{R}(s_j) = \mathcal{R}(s_k)) & \text{if } \mathcal{R}(s_j) = \max_k \mathcal{R}(s_k) \\
0 & \text{otherwise}
\end{cases}
\]

To illustrate, consider a simple example of 2-bundling scheme, three levels of contribution \( L = 3 \), and two rating points \( L' = 2 \). We then have \( R_1(C_3, C_2|r) = 2r \), and \( R_1(C_1, C_2|r) = R_2(C_1, C_2|r) = r \). We can show the corresponding utility in Table V. The strategy profile \((C_3, C_4)\) is a unique Nash equilibrium if and only if \( r > c_3 - c_1 = c_3 \). We generalize this result in the following lemma.

<table>
<thead>
<tr>
<th>( w_2 )</th>
<th>( C_3 )</th>
<th>( C_2 )</th>
<th>( C_1 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( w_1 )</td>
<td>( r - c_3 ), ( r - c_3 )</td>
<td>( 2r - c_3 ), ( -c_3 )</td>
<td>( 2r - c_3 ), ( -c_3 )</td>
</tr>
<tr>
<td>( C_2 )</td>
<td>( -c_3 ), ( 2r - c_3 )</td>
<td>( r - c_3 ), ( -c_3 )</td>
<td>( r - c_3 ), ( -c_3 )</td>
</tr>
<tr>
<td>( C_1 )</td>
<td>( -c_3 ), ( 2r - c_3 )</td>
<td>( r - c_3 ), ( -c_3 )</td>
<td>( r - c_3 ), ( -c_3 )</td>
</tr>
</tbody>
</table>

Lemma IV.1. Consider the setting without erroneous ratings, a \( n \)-task bundling scheme and a threshold based rating system \((L', C_L, \mathcal{R}(\cdot))\). The strategy profile \((C_L, \ldots, C_L)\) is a unique Nash Equilibrium if and only if \( r > L - c_L \).

Proof: This proof is similar to that of Lemma II.1. 

Remark. Under the perfect scenario without erroneous ratings, the critical value is invariant of the number of rating points \( L' \). This implies that the simplest rating system, e.g., \( L' = 2 \), is also an optimal system, where requesters only need to provide binary feedbacks to indicate whether they are satisfied or not with a solution.

We now explore the scenario with erroneous ratings. In the following theorem we prove the existence and uniqueness of the desired Nash Equilibrium \((C_L, \ldots, C_L)\), and we quantify the impact of the variability matrix \( \alpha \) and the threshold based rating system on the critical value.

Theorem IV.1. Consider a \( n \)-bundling scheme, a variability matrix \( \alpha \), and a \((L', C_L, \mathcal{R}(\cdot))\) rating system. The strategy profile \((C_L, \ldots, C_L)\) is a unique Nash equilibrium iff

\[
r > r = \max_k \left\{ (c_L - c_k) / \left[ 1 - \sum_{\ell=1}^{L-L' + 1} \sum_{k=1}^{L} \alpha_{k,\ell} \right] \right\}
\]

Proof: Please refer to [14] for derivation.

Remark. The importance of the above theorem is on the existence and uniqueness of the desired Nash Equilibrium \((C_L, \ldots, C_L)\) under different design of rating systems. In addition, it quantifies the impact of the number of rating points \( L' \) on the critical value. As we shall see later, this result serves as building blocks to explore rating system design tradeoffs.

We show some illustrating numerical examples on the critical value in Table VI, where we examine the impact of number of rating points. In Table VI, we specify the cost for each level of contribution as

\[
c_j = (j - 1)^2 / (L - 1)^2, \quad j = 1, \ldots, L,
\]

and we specify the variability matrix \( \alpha \) as

\[
\alpha_{j,\kappa} = \theta |j - \kappa| / \sum_{k=1}^{L} \theta |j - \kappa|, \quad \forall j, \kappa = 1, \ldots, L.
\]

where \( \theta \in (0, 1) \). Note that this choice of the cost function and variability matrix are only for illustration purpose. Practically, one can infer them from data. The higher the value of \( \theta \), the higher the variability in ratings. The number of contribution levels is \( L = 7 \). And we vary the value of the number of rating points \( L' \) from 2 to 7. One can observe that as we increase the number rating points \( L' \), we decrease the critical value. When we increase the value of \( \theta \), we increase the critical value. Namely, the higher variability in rating, the higher the reward requesters need to pay. It is interesting to observe that five rating points \( L' = 5 \) is actually good enough, since the critical value is very close to \( c_2 \), and further increasing on the number of rating points, decreases the critical very less than 0.5%. This coincides with that five-level rating system is commonly used in many crowdsourcing systems.

<table>
<thead>
<tr>
<th>( L' )</th>
<th>( 2 )</th>
<th>( 3 )</th>
<th>( 4 )</th>
<th>( 5 )</th>
<th>( 6 )</th>
<th>( 7 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \theta = 0.1 )</td>
<td>1.111</td>
<td>1.010</td>
<td>1.001</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
</tr>
<tr>
<td>( \theta = 0.2 )</td>
<td>1.250</td>
<td>1.042</td>
<td>1.008</td>
<td>1.002</td>
<td>1.001</td>
<td>1.001</td>
</tr>
<tr>
<td>( \theta = 0.3 )</td>
<td>1.429</td>
<td>1.000</td>
<td>1.029</td>
<td>1.010</td>
<td>1.005</td>
<td>1.004</td>
</tr>
</tbody>
</table>

Summary. By using the threshold based rating system, it is possible incentivize high quality contributions under even with
erroneous ratings. We also quantify the impact of number of rating points on the critical value.

V. RELATED WORK

Crowdsourcing has recently drawn a lot of attentions. Many aspects of crowdsourcing are being studied, e.g., applying the concept of crowdsourcing to design new applications and algorithms [9], [10], user behavior study [18], and investigating the performance issues, like quality management [19], incentive design [20], fairness [21], etc. A survey can be found in [22].

A variety of approaches have been proposed to price tasks. One typical approach is mining the price from data [23]. More concretely, building models to infer workers’ benefit and cost in solving tasks. Another approach determines the price automatically or dynamically via designing some efficient algorithms [9], [10], user behavior study [18], and investigating the concept of crowdsourcing to design new applications and algorithms. A survey can be found in [22].

Recently, a few workers investigated incentive mechanism design for crowdsourcing applications. An experiment in an online labor market was conducted to understand the effectiveness of a collection of social and finical incentive schemes [20]. A game-theoretic model of an online question and answer forum was developed in [26], where the authors investigated the impact of various score sharing rules on the incentives for providing solution. A few reputation based incentive protocols were developed in [27], [28], [29]. They induce incentive via maintaining a reputation for each worker to track the contribution history and penalize workers with low reputation. However, they simplified the model by assuming a binary contribution (either high effort or free-riding) set for workers. Our work is different from theirs in a few aspects. First, we allow multi-level contributions. Second, to guarantee workers to provide high quality contributions, their protocol requires a high reward and suffers from high revenue loss due to punishment of workers. However, our incentive mechanism has no revenue loss and require a small amount of reward (close to workers’ cost). Third, their protocols have a strong restriction on the task assignment scheme, i.e., random pick. However, our incentive mechanism are general enough to be deployed on any task assignment algorithms.

VI. CONCLUSION

This paper studies incentive mechanism and rating system design for crowdsourcing applications, as well as techniques to reduce the payment. We design a class of simple but effective incentive mechanisms, which consist of a task bundling scheme and a rating system. We propose a probabilistic model to capture various human factors, e.g., bias, in rating, and quantify its impact on the incentive mechanism, which is shown to be highly robust. We develop a model to characterize the design space of a class of commonly users rating systems – threshold based rating systems. We quantify the impact of such rating systems, and the bundling scheme on reducing requesters’ overhead.

ACKNOWLEDGMENT
The authors would like to thank anonymous reviewers for their helpful reviews. The research of John C.S. Lui is supported in part by the GRF Grant 415112.

REFERENCES