Keep Your Promise: Mechanism Design against Free-riding and False-reporting in Crowdsourcing

Xiang Zhang, *Student Member, IEEE*, Guoliang Xue, *Fellow, IEEE*, Ruozhou Yu, *Student Member, IEEE*,

Dejun Yang, *Member, IEEE*, and Jian Tang, *Senior Member, IEEE*

***Abstract*—Crowdsourcing is an emerging paradigm where users can have their tasks completed by paying fees, or receive rewards for providing service. A critical problem that arises in current crowdsourcing mechanisms is how to ensure that users pay or receive what they deserve. Free-riding and false-reporting may make the system vulnerable to dishonest users. In this paper, we design schemes to tackle these problems, so that each individual in the system is better off being honest and each provider prefers completing the assigned task. We first design a mechanism EFF which eliminates dishonest behavior with the help from a trusted third party for arbitration. We then design another mechanism DFF which, without the help from any third party, discourages dishonest behavior. We prove that EFF eliminates free-riding and false-reporting, while guaranteeing truthfulness, transaction-wise budget-balance, and computational efficiency. We also prove that DFF is semi-truthful, which discourages dishonest behavior such as free-riding and false-reporting when the rest of the individuals are honest, while guaranteeing transaction-wise budget-balance and computational efficiency. Performance evaluation shows that within our mechanisms, no user could have a utility gain by unilaterally being dishonest.**

***Index Terms*—Crowdsourcing, free-riding, false-reporting, game theory, incentive mechanisms.**

1. Introduction

A. Crowdsourcing

For the past few years, we have witnessed the proliferation of crowdsourcing [16] as it becomes a booming online market for labor and resource redistribution. One example is mobile crowdsourcing [25, 34], which leverages a cloud computing platform for recruiting mobile users to collect data (such as photos, videos, mobile user activities, etc) for applications in various domains such as environmental monitoring, social networking, healthcare, transportation, etc. Several commercial crowdsourcing websites, such as Yelp [1], Yahoo! Answers [2], Amazon Mechanical Turk [3], and UBER [4], provide trading markets where some users (*service requesters*) can have their tasks completed by paying fees, and some other users (*service providers*) can receive rewards for providing service. An important functionality of these websites is to empower platforms not only to offer markets for laboring and resource redistribution, but also to help decide fair prices for all providers and requesters.

In general, a requester would post a task on one of these platforms, and wait till some provider to work on it. To incentivize providers for their service, the requester needs to offer a certain reward. A provider will get reward as compensation for its cost of completing the task.

Zhang, Xue, and Yu are with Arizona State University, Tempe, AZ 85287. Email: {xzhan229, xue, ruozhouy}@asu.edu. Yang is with Colorado School of Mines, Golden, CO 80401. Email: djyang@mines.edu. Tang is with Syracuse University, Syracuse, NY 13244. Email: jtang02@syr.edu. This research was supported in part by ARO grants W911NF-09-1-0467 and W911NF-13-1-0349 and NSF grants 1217611, 1218203, 1420881, 1421685, and 1444059. The information reported here does not reflect the position or the policy of the federal government. This is an extended and enhanced version of the paper [35] that appeared in IEEE GLOBECOM 2014.

B. Auction Theory

Auction is an efficient mechanism for trading markets, with its advantage in discovering prices for buyers and sellers. Auctions involving the interactions among multiple buyers and multiple sellers are called *double auctions* [12]. When determining payments for buyers and sellers, it is always feared that the prices are manipulated to make the free market vulnerable to dishonest individuals. Therefore, several economic properties, such as truthfulness, individual rationality, budget-balance, and computational efficiency, are desired. Here are the definitions of these properties:

- Truthfulness: An auction is truthful if no individual can achieve a higher utility by reporting a value deviating from its true valuation (or cost) regardless of the bids (or asks) from other individuals.
- Individual Rationality: An auction is individually rational if for any buyer or seller, it will not get a negative utility by revealing its true valuation (or cost).
- Budget-balance: An auction is budget-balanced if the auctioneer always makes a non-negative profit.
- Computational Efficiency: An auction is computationally efficient if the whole auction process can be conducted within polynomial time.

Among these properties, truthfulness is the crowning jewel which incentivizes all participants to be honest during the auction [5, 11, 19, 27, 34, 37]. However, this alone cannot guarantee the robustness of the mechanism, since other dishonest behavior, such as free-riding and false-reporting (to be defined later), may make the crowdsourcing system vulnerable.

C. Free-riding and False-reporting in Crowdsourcing

Truthfulness of an auction can prevent individuals benefit from lying about the prices. However, this alone is not adequate.

Free-riding and false-reporting can make the mechanism vulnerable to dishonest users.

If the payment is made *before* the provider starts to work, a provider always has the incentive to take the payment and devote no effort to complete the task, which is known as "*free-riding*" [36]. If the payment is made *after* the provider completes its work, the requester always has the incentive to refuse the payment by lying about the status of this task, which is known as "*false-reporting*" [36]. Thus, extra precautions are necessary. Most recent works [10, 15, 36] are focused on avoiding free-riding and false-reporting by building up reputation systems or grading systems. These mechanisms may discourage such dishonest behavior overall, but *based on the assumption that all individuals are patient and they would stay in the system*. However, in reality, some dishonest individuals may stay in the system for a short period of time, and get huge rewards by being dishonest. In addition, impatient users may leave the platform since they feel unsafe and being cheated.

In this paper, we tackle free-riding and false-reporting by a game-theoretic approach, such that free-riding and false-reporting are eliminated or discouraged.

D. Game Theory Basics

Game theory [23] is a field of study where interactions among different players are involved. Each player wants to maximize its own utility, which depends on not only its own strategies, but also other players' strategies. We need the concept of *dominant strategy* and *Nash Equilibrium* [24] as follows.

- Dominant Strategy: A strategy s dominates all other strategies if the payoff to s is no less than the payoff to any other strategy, regardless of the strategies chosen by other players. s is a *strongly dominant strategy* if its payoff is strictly larger than the payoff to any other strategy.
- Nash Equilibrium (NE): When each player has chosen a strategy and no player can benefit by unilaterally changing its strategy, the current set of strategies constitutes an NE.

Extensive-form game theory [23] is an important branch of game theory. An extensive-form game is a game where sequencing players' strategies and their choices at each decision point is allowed. The corresponding equilibrium, which is called *Sub-game Perfect Equilibrium* (SPE), is a strategy profile where in every sub-game, an NE is reached by the corresponding subset of strategies.

E. Summary of Contributions

In this paper, we design two novel auction-based mechanisms, **EFF** and **DFF**, to avoid free-riding and false-reporting, while incentivizing providers to complete their assigned tasks. These mechanisms are based on *any* existing truthful double auction scheme for winner selection and pricing. Each auction winner is required to submit a *warranty* first, then submit a report on the status of the corresponding task. Based on these reports from providers and requesters, the final payments are determined by the platform.

The main contributions of this paper are the following:

- To the best of our knowledge, we are the first to solve free-riding and false-reporting *in each single round*.
- We design a mechanism **EFF**, which, with the help of arbitrations, eliminates free-riding and false-reporting, and incentivizes providers to complete their assigned tasks. We also prove that **EFF** is truthful, transaction-wise budget-balanced, and computationally efficient.
- We design a mechanism **DFF**, which, without any arbitration, discourages individuals from being dishonest. We prove that **DFF** is semi-truthful, which means that no individual could have a higher utility by lying when others are honest. We also prove that **DFF** is transaction-wise budget-balanced and computationally efficient, while incentivizing providers to finish their assigned tasks.

The remainder of this paper is organized as follows. In Section 2, we briefly review the current literature on crowdsourcing mechanisms, truthful auctions, and mechanisms avoiding free-riding and false-reporting. In Section 3, we describe the system model. We present and analyze **EFF** and **DFF** in Section 4 and Section 5, respectively. We present performance evaluation in Section 6 and draw our conclusions in Section 7.

2. Related Work

Many incentive mechanisms have been proposed for crowdsourcing and mobile sensing [5, 6, 9, 11, 17, 20, 26, 27, 31, 32, 34, 37]. In 2012, Yang *et al.* [34] studied two models: user-centric and platform-centric models. For the user-centric model, an auction-based mechanism is presented, which is computationally efficient, individually rational, profitable, and truthful. For the platform-centric model, the Stackelberg game is formulated and the unique Stackelberg Equilibrium is calculated to maximize the platform utility. In [9], DiPalantino and Vojnovic proposed an all-pay auction, where each provider may choose the job that it wants to work on. In [17], an online question and answer forum was introduced, where each user has a piece of information and can decide when to answer the question. Singla and Krause in [27] proposed BP-UCB, which is an online truthful auction mechanism achieving near-optimal utilities for all providers. A crowdsourcing Bayesian auction is proposed in [5], where the system is aware of the distribution of valuations of all users. With this assumption, two techniques were applied to the new Bayesian optimal auction with high system performance. Zhao *et al.* [37] proposed an online truthful auction for crowdsourcing with a constant approximation ratio with respect to the platform utility. In [11], a truthful auction algorithm of task allocation to optimize the total utility of all smartphone users was proposed by Feng *et al.* for the offline model, and an online model was also introduced which reaches a constant approximation ratio of $\frac{1}{2}$.

Many other truthful auctions also fit for crowdsourcing and mobile sensing. There are two basic truthful double auctions, VCG [7, 14, 28] and McAfee [21], from which most of the current truthful auctions are derived. Along this line, many mechanisms [8, 13, 18, 19, 22, 29, 33] are proposed. While they are not specifically designed for crowdsourcing platforms, they can be applied to the scenario with minor modifications.

These auction schemes are based on the assumption that all providers will complete the assigned tasks, and all requesters are satisfied with the status of the tasks. This may not always happen. Therefore, there are works focusing on how to make the mechanism robust against free-riding and false-reporting [10, 15, 36]. In these papers, free-riding and false-reporting are avoided by building reputation systems or grading systems. Such systems are based on the assumption that *all individuals are patient and will stay in the system for a very long time*. Thus, a user may gain extra benefit by free-riding or false-reporting during a short period of time. Impatient users may feel being cheated and turn to other platforms.

3. System Model

In this section, we present the crowdsourcing model and the corresponding double auction model, and define the utilities for the platform, the providers, and the requesters.

There is a set of m *requesters* $\mathcal{R} = \{R_1, R_2, ..., R_m\}$. For each R_i, it requires service to complete its task T_i, which has a private *valuation* $v_i > 0$ to R_i if T_i is completed. T_i is tagged with a *bid* b_i by R_i, which is the maximum reward that R_i is willing to pay for the completion of T_i. Note that b_i is not necessarily equal to v_i. We define the *bid vector* $\mathbf{b} = (b_1, b_2, ..., b_m)$.

There is a set of n *providers* $\mathcal{P} = \{P_1, P_2, ..., P_n\}$. Each provider P_j has a *cost* $c_j^i > 0$ to complete T_i, where c_j^i is set to $+\infty$ if P_j is unable to complete T_i. P_j would post an *ask* a_j^i for T_i, which is the minimum amount of reward that P_j demands for completing T_i. Note that a_j^i is not necessarily equal to c_j^i. We define the *cost vector* $\mathbf{c}_j = (c_j^1, c_j^2, ..., c_j^m)$, the *ask vector* $\mathbf{a}_j = (a_j^1, a_j^2, ..., a_j^m)$, and the *ask matrix* $\mathbb{A} = (\mathbf{a}_1; \mathbf{a}_2; ...; \mathbf{a}_n)$.

We assume that providers and requesters are non-colluding. By applying a sealed-bid single-round double auction where requesters are buyers and providers are sellers, we get a set $\mathcal{W}$ of *winning requester-provider pairs*, such that $(R_i, P_j) \in \mathcal{W}$ if and only if P_j is assigned to complete T_i. Each provider is assigned to at most one task, while no more than one provider works on each task. We define the function $\delta(\cdot) : \mathcal{P} \to \mathcal{R}$ such that $\delta(j) = i$ if and only if $(R_i, P_j) \in \mathcal{W}$, and $\delta(j) = 0$ if P_j is not assigned to any task. As a result of the auction, R_i will be *charged* β_i, and P_j will *receive* $\alpha_j^{\delta(j)}$. As a technical convention, we define $\beta_i = 0$ if R_i is not a winning requester, and $\alpha_j^i = 0$ if $(R_i, P_j) \notin \mathcal{W}$. We define $\beta = (\beta_1, \beta_2, ..., \beta_m)$, $\alpha_j = (\alpha_j^1, \alpha_j^2, ..., \alpha_j^m)$, and $\alpha = (\alpha_1; \alpha_2; ...; \alpha_n)$. The platform (auctioneer) has a *profit* $\Sigma_{(R_i,P_j)\in\mathcal{W}}(\beta_i - \alpha_j^i)$ from the auction.

The *auction mechanism*, denoted as $\mathbb{M}$, takes the bid vector $\mathbf{b}$ and the ask matrix $\mathbb{A}$ as input. $\mathbb{M}$ outputs the winning requester-provider pair set $\mathcal{W}$ and the payments α and β. Thus, we have $(\mathcal{W}, \beta, \alpha) \leftarrow \mathbb{M}(\mathbf{b}, \mathbb{A})$.

We call an auction mechanism $\mathbb{M}$ *transaction-wise budget-balanced* if $\beta_i \geq \alpha_j^i$, for each $(R_i, P_j) \in \mathcal{W}$. Note that transaction-wise budget-balance implies budget-balance, *i.e.*, $\sum_{(R_i,P_j)\in\mathcal{W}}(\beta_i - \alpha_j^i) \geq 0$. *Throughout this paper, we assume that* $\mathbb{M}$ *is truthful, individually rational, transaction-wise budget-balanced, and computationally efficient.* All mechanisms in [8, 18, 29, 31, 33] satisfy these properties. In particular, TASC [33] is used in our implementation.

For each $(R_i, P_j) \in \mathcal{W}$, the platform collects warranties from both R_i and P_j. We model the post-auction process as an extensive-form game between P_j and R_i. In this game, P_j first decides the effort towards completing T_i; then R_i and P_j evaluate the result and submit independent reports on the status of T_i to the platform. We assume that P_j is capable of completing T_i, hence the status of T_i depends on the willingness of P_j. The reports are either **C**, which is short for **Completed**, or **I**, which is short for **Incomplete**. Both R_i and P_j know the status of T_i. However, the platform is not aware of T_i's status. The platform decides the final payment for each individual according to these reports. If R_i and P_j submit the same report, the platform would believe that both of them are telling the truth. Otherwise, the platform concludes that one of them lies and consults for arbitration if available. We assume that there are no ambiguities on the status of T_i.

The *utility of* R_i is defined as

$$U_i^R = x_i v_i - (w_i^R - \bar{\beta}_i), \tag{3.1}$$

where x_i is the indicator of the status of T_i: $x_i = 1$ if T_i is completed; $x_i = 0$ otherwise. w_i^R is the warranty that R_i pays to the platform, and $\bar{\beta}_i$ is the final payment that R_i receives from the platform after the platform collects the reports.

The *utility of* P_j is defined as

$$U_j^P = (\bar{\alpha}_j^{\delta(j)} - w_j^P) - y_j^{\delta(j)} c_j^{\delta(j)}, \tag{3.2}$$

where $y_j^{\delta(j)}$ is the indicator of P_j's devotion on $T_{\delta(j)}$: $y_j^{\delta(j)} = 1$ if P_j worked on $T_{\delta(j)}$; $y_j^{\delta(j)} = 0$ otherwise. w_j^P is the warranty from P_j, and $\bar{\alpha}_j^{\delta(j)}$ is the final amount that P_j receives from the platform after the platform collects the reports.

The *platform utility* is defined as

$$U = \sum\nolimits_{(R_i,P_j)\in\mathcal{W}}[(w_i^R - \bar{\beta}_i + (w_j^P - \bar{\alpha}_j^{\delta(j)}) - z_i\tau_i], \tag{3.3}$$

where z_i is the indicator of arbitration on T_i: $z_i = 1$ if the platform has consulted for arbitration on T_i; $z_i = 0$ otherwise. τ_i is the corresponding arbitration fee. We assume that τ_i is a value known by all providers, requesters, and the platform. For each $(R_i, P_j) \in \mathcal{W}$, we define

$$U(R_i, P_j) = (w_i^R - \bar{\beta}_i) + (w_j^P - \bar{\alpha}_j^{\delta(j)}) - z_i\tau_i, \tag{3.4}$$

which is the portion of platform utility contributed by (R_i, P_j). Clearly, $U = \sum_{(R_i,P_j)\in\mathcal{W}} U(R_i, P_j)$.

Remark 3.1: The utilities of providers, requesters, and the auctioneer in this paper are different from those in [35]. Another difference between this paper and [35] is that we model the post-auction process into an extensive-form game, in contrast to a simultaneous game. □

The notations of this paper are summarized in Table 1.

TABLE 1
NOTATIONS

Symbol	Meaning
$R_i,\ P_j$	provider (seller), requester (buyer)
$\mathcal{R},\ \mathcal{P}$	the set of providers, requesters
T_i	R_i's task
v_i	valuation of T_i
$\mathbf{b}/b_i$	bid vector/ bid from R_i
$\mathbf{c_j}/c_j^i$	cost vector of P_j/ cost of P_j to finish T_i
$\mathbb{A}/\mathbf{a_j}/a_j^i$	ask matrix/ ask vector of P_j/ ask of P_j to finish T_i
$\mathcal{W}$	set of winning provider-requester pairs
$\delta(\cdot)$	assignment function from providers to requesters
β_i	payment of R_i from the auction
β	payment vector for requesters from the auction
α_j^i	payment of P_j for T_i from the auction
$\alpha_\mathbf{j}$	payment vector of P_j from the auction
α	payment matrix of providers from the auction
$\mathbb{M}$	the auction mechanism
$U_i^R/U_j^P/U$	utility of R_i/ P_j/ the platform
x_i	indicator of T_i's status
$y_j^{\delta(j)}$	indicator of P_j's devotion on $T_{\delta(j)}$
z_i	indicator of arbitration on T_i
τ_i	arbitration fee of T_i
$\bar{\beta}_i$	final payment that R_i receives from the platform
$\bar{\alpha}_j^{\delta(j)}$	final payment that P_j receives from the platform

4. EFF: ELIMINATING FREE-RIDING AND FALSE-REPORTING WITH ARBITRATION

In this section, we present a crowdsourcing mechanism **EFF**, which, with the help from a trusted third party for arbitration, eliminates free-riding and false-reporting and incentivizes each provider to complete the assigned task.

A. *Description of* EFF

In **EFF**, a trustworthy third party is available, who can provide arbitration on the status of T_i with an arbitration fee $\tau_i > 0$.

The first part of **EFF** applies auction $\mathbb{M}$, where $\mathcal{W}$, β, and α are decided. After the auction, the platform collects a warranty $w_i^R = \beta_i + \tau_i$ from R_i and a warranty $w_j^P = \theta\alpha_j^i + \tau_i$ from P_j for each $(R_i, P_j) \in \mathcal{W}$, where $\theta > 0$ is a system parameter. We use $\theta\alpha_j^i$ to help incentivize P_j to complete T_i. If P_j has been assigned to work on T_i but does not complete T_i, it will receive a penalty of $\theta\alpha_j^i$.

After R_i and P_j submitted their warranties, an extensive-form game is applied. P_j decides whether or not to work on T_i. If P_j chooses to devote effort on T_i, T_i would be completed. Otherwise, T_i is incomplete with no effort devoted by P_j. Then R_i and P_j submit their independent reports on the status of T_i. Gathering the reports from R_i and P_j, the platform decides the final payments $\bar{\beta}_i$ and $\bar{\alpha}_j^i$ for R_i and P_j, respectively. The formal description of **EFF** is presented as Algorithm 1.

Algorithm 1: EFF

1 $(\mathcal{W}, \beta, \alpha) \leftarrow \mathbb{M}(\mathbf{b}, \mathbb{A})$;
2 $\forall (R_i, P_j) \in \mathcal{W}, w_i^R \leftarrow \tau_i + \beta_i, w_j^P \leftarrow \tau_i + \theta\alpha_j^i$;
3 R_i submits w_i^R and P_j submits w_j^P to the platform;
4 $\bar{\beta}_i \leftarrow w_i^R$; $\bar{\alpha}_j^i \leftarrow w_j^p$;
5 P_j decides whether or not to work on T_i, and devotes the corresponding effort;
6 R_i and P_j submit independent reports on the status of T_i to the platform;
7 **if** *Reports are different* **then**
8   The platform consults arbitration for the status of T_i, and pays τ_i to the arbitrator;
9   **if** *R_i submits a false report* **then**
10         $\bar{\beta}_i \leftarrow \bar{\beta}_i - \tau_i$;
11   **else**
12         $\bar{\alpha}_j^i \leftarrow \bar{\alpha}_j^i - \tau_i$;
13   **end**
    // The arbitration fee is taken from the liar
14 **else**
15   The platform adopts the reports from R_i and P_j;
16 **end**
17 **if** *The platform concludes that T_i is completed* **then**
18     $\bar{\beta}_i \leftarrow \bar{\beta}_i - \beta_i$; $\bar{\alpha}_j^i \leftarrow \bar{\alpha}_j^i + \alpha_j^i$;
    // The auction payment is made
19 **else**
20     $\bar{\alpha}_j^i \leftarrow \bar{\alpha}_j^i - \theta\alpha_j^i$;
    // $\theta\alpha_j^i$ is taken from P_j for not completing T_i
21 **end**
22 The platform returns $\bar{\beta}_i$ to R_i and $\bar{\alpha}_j^i$ to P_j, respectively.

With the payments $\bar{\beta}_i$ and $\bar{\alpha}_j^i$ from Algorithm 1, utilities of R_i and P_j can be computed based on equations (3.1) and (3.2), respectively. These utility values are shown in Table 2. The first row indicates P_j's effort on T_i and the second row lists the reports from P_j. The first column lists the reports from R_i. For each of the 2-tuples, the first element is the utility of R_i, and the second element is the utility of P_j.

We pick two entries in Table 2 and explain how the corresponding utilities are derived. First, we explain the entry where T_i is completed and both R_i and P_j submit **C** (marked in red in Table 2). In Line 2 of Algorithm 1, we have $w_i^R = \tau_i + \beta_i$ and $w_j^P = \tau_i + \theta\alpha_j^i$. In Line 4, we have $\bar{\beta}_i = \tau_i + \beta_i$ and $\bar{\alpha}_j^i = \tau_i + \theta\alpha_j^i$. Because both R_i and P_j submit **C**, no arbitration is consulted, and the platform concludes that T_i is completed. Thus, $\bar{\beta}_i = (\tau_i + \beta_i) - \beta_i = \tau_i$. Since P_j has devoted effort on T_i and T_i is completed, we have $y_j^i = 1$ and $x_i = 1$. In Line 18, we have $\bar{\alpha}_j^i = (\tau_i + \theta\alpha_j^i) + \alpha_j^i$. Based on equations (3.1) and (3.2), R_i's utility is $v_i - \beta_i$ and P_j's utility is $\alpha_j^i - c_j^i$. Next, we explain the entry where T_i is incomplete, R_i submits **I**, and P_j submits **C** (marked in blue in Table 2). In Line 2, we have $w_i^R = \tau_i + \beta_i$ and $w_j^P = \tau_i + \theta\alpha_j^i$. In Line 4, we have $\bar{\beta}_i = \tau_i + \beta_i$ and $\bar{\alpha}_j^i = \tau_i + \theta\alpha_j^i$. Because R_i and P_j submit different reports, an arbitration is consulted and the arbitration result indicates that P_j submits the dishonest report. In Line 12, we have $\bar{\alpha}_j^i = (\tau_i + \theta\alpha_j^i) - \tau_i = \theta\alpha_j^i$. Because T_i is incomplete, we have $y_j^i = 0$, $x_i = 0$. In Line 20, we have

TABLE 2
UTILITIES OF R_i AND P_j IN **EFF** FOR $(R_i, P_j) \in \mathcal{W}$, THE ENTRY FOR THE UNIQUE SPE IS MARKED BY ✓.

R_i \ P_j	T_i completed		T_i not completed	
	C	**I**	**C**	**I**
C	$(v_i - \beta_i, \alpha_j^i - c_j^i)$✓	$(v_i - \beta_i, \alpha_j^i - c_j^i - \tau_i)$	$(-\beta_j, \alpha_j^i)$	$(-\tau_i, -\theta\alpha_j^i)$
I	$(v_i - \beta_i - \tau_i, \alpha_j^i - c_j^i)$	$(v_i, -\theta\alpha_j^i - c_j^i)$	$(0, -\theta\alpha_j^i - \tau_i)$	$(0, -\theta\alpha_j^i)$

$\bar{\alpha}_j^i = \theta\alpha_j^i - \theta\alpha_j^i = 0$. Based on equations (3.1) and (3.2), R_i's utility is 0 and P_j's utility is $-\theta\alpha_j^i - \tau_i$.

According to Table 2, *there is a unique SPE*. If P_j completes T_i, P_j would report **C** regardless of what R_i reports. Since P_j reports **C**, R_i prefers reporting **C** to reporting **I**. Hence, if P_j completes T_i, both R_i and P_j would report **C**. For similar reason, if P_j does not complete T_i, both R_i and P_j would report **I**. Comparing the utilities of P_j when completing T_i (which is $\alpha_j^i - c_j^i > 0$) and not completing T_i (which is $-\theta\alpha_j^i < 0$), P_j prefers completing T_i. *Thus, the unique SPE is reached when P_j completes T_i, and both R_i and P_j submit* **C**.

B. Analysis of **EFF**

In this subsection, we prove several properties of **EFF**.

Theorem 1: **EFF** eliminates free-riding and false-reporting, while guaranteeing truthfulness, transaction-wise budget-balance, and computational efficiency. □

To prove this theorem, we need to prove lemmas 4.1-4.3. Recall that auction $\mathbb{M}$ is individually rational, transaction-wise budget-balanced, computationally efficient, and truthful.

Lemma 4.1: **EFF** eliminates free-riding and false-reporting, while guaranteeing truthfulness. □

Proof: Suppose that (R_i, P_j) is in $\mathcal{W}$ when R_i bids v_i and P_j bids $\mathbf{c}_j$. The unique SPE is achieved when P_j completes T_i and both R_i and P_j submit **C**. According to Table 2 and the individual rationality of $\mathbb{M}$, R_i's utility is $v_i - \beta_i \geq 0$ and P_j's utility is $\alpha_j^i - c_j^i \geq 0$. When R_i or P_j changes its bid during the auction, due to the truthfulness of $\mathbb{M}$, the corresponding payment of R_i could not be lower than β_i and the corresponding payment of P_j could not be higher than α_j^i. Thus, neither R_i nor P_j has incentive to change its bid. When R_i submits **I** for false-reporting, P_j still reports **C** since it is P_j's dominant strategy when T_i is completed. R_i's corresponding utility would be $v_i - \beta_i - \tau_i < v_i - \beta_i$. When P_j chooses not to complete T_i but submits **C** for free-riding, R_i still reports **I** since it is R_i's dominant strategy when T_i is incomplete. P_j's corresponding utility would be $-\theta\alpha_j^i - \tau_i < 0 \leq \alpha_j^i - c_j^i$. Thus, neither R_i nor P_j could benefit from being dishonest.

Suppose that R_i loses the auction by bidding $b_i = v_i$. Its utility is 0. Suppose that R_i changes its bid. If R_i still loses the auction, its utility remains to be 0. If R_i wins the auction, its new payment in the auction β_i is no smaller than v_i since $\mathbb{M}$ is truthful. If T_i is completed, according to Table 2, R_i's utility is no more than $v_i - \beta_i \leq v_i - b_i = 0$ since P_j would always report **C**. If T_i is incomplete, R_i could not have a positive utility. Thus, R_i could not benefit from being dishonest.

Suppose that P_j loses the auction by asking $\mathbf{a}_j = \mathbf{c}_j$. Its utility is 0. Suppose that P_j changes its ask. If P_j still loses the auction, its utility remains to be 0. If P_j wins the auction and is assigned to complete T_i, P_j's new payment in the auction α_j^i is no larger than c_j^i since $\mathbb{M}$ is truthful. If P_j completes T_i, according to Table 2, its utility is no larger than $\alpha_j^i - c_j^i \leq a_j^i - c_j^i = 0$. If P_j does not complete T_i, its utility is no more than $-\theta\alpha_j^i \leq 0$ since R_i would report **I**. Thus, P_j could not benefit from being dishonest.

To sum up all cases, **EFF** eliminates free-riding and false-reporting, while guaranteeing truthfulness. ■

Remark 4.1: By Lemma 4.1, dishonest behaviors are eliminated in **EFF**. According to the unique SPE in the extensive-form game in **EFF**, for each $(R_i, P_j) \in \mathcal{W}$, P_j would finish T_i, which indicates that **EFF** incentivizes each winning provider to complete its assigned task. □

Lemma 4.2: **EFF** is transaction-wise budget-balanced. □

Proof: Note that by the transaction-wise budget-balance of $\mathbb{M}$, $\beta_i - \alpha_j^i \geq 0$ for each $(R_i, P_j) \in \mathcal{W}$. We prove this lemma by the following case analysis:

- If R_i and P_j both submit **C**, the platform believes that T_i is completed without consulting for arbitration. Thus, we have $\bar{\beta}_i = \tau_i$ and $\bar{\alpha}_j^i = \tau_i + (1+\theta)\alpha_j^i$. By equation (3.4), we have $U(R_i, P_j) = \beta_i - \alpha_j^i \geq 0$.
- If R_i and P_j both submit **I**, the platform believes that T_i is incomplete without consulting for arbitration. Thus, we have $\bar{\beta}_i = \tau_i + \beta_i$ and $\bar{\alpha}_j^i = \tau_i$. By equation (3.4), we have $U(R_i, P_j) = \theta\alpha_j^i \geq 0$.
- If R_i submits **C** and P_j submits **I** when T_i is completed, the platform consults for arbitration, and the result shows that P_j lies in the report. Thus, we have $\bar{\beta}_i = \tau_i + \beta_i$ and $\bar{\alpha}_j^i = (1+\theta)\alpha_j^i$. By equation (3.4), we have $U(R_i, P_j) = \beta_i - \alpha_j^i \geq 0$.
- If R_i submits **C** and P_j submits **I** when T_i is incomplete, the platform consults for arbitration, and the result shows that R_i lies in the report. Thus, we have $\bar{\beta}_i = \beta_i$ and $\bar{\alpha}_j^i = \tau_i$. By equation (3.4), we have $U(R_i, P_j) = \theta\alpha_j^i \geq 0$.
- If R_i submits **I** and P_j submits **C** when T_i is completed, the platform consults for arbitration, and the result shows that R_i lies in the report. Thus, we have $\bar{\beta}_i = 0$ and $\bar{\alpha}_j^i = \tau_i + (1+\theta)\alpha_j^i$. By equation (3.4), we have $U(R_i, P_j) = \beta_i - \alpha_j^i \geq 0$.
- If R_i submits **I** and P_j submits **C** when T_i is incomplete, the platform consults for arbitration, and the result shows that P_j lies in the report. Thus, we have $\bar{\beta}_i = \tau_i + \beta_i$ and $\bar{\alpha}_j^i = 0$. By equation (3.4), we have $U(R_i, P_j) = \theta\alpha_j^i \geq 0$.

Thus, **EFF** is transaction-wise budget-balanced. ■

Lemma 4.3: **EFF** is computationally efficient. □

Proof: Since $\mathbb{M}$ is computationally efficient, and report submission, arbitration, and pricing all take constant time, **EFF** is computationally efficient. ■

Lemmas 4.1-4.3 complete the proof for Theorem 1.

Remark 4.2: **EFF** is not individually rational. To incentivize each winning provider to complete its assigned task, we impose a penalty $\theta\alpha_j^i$ in Algorithm 1, Line 20. If P_j fails to complete T_i but behaves honestly, its utility is $-\theta\alpha_j^i < 0$, which violates individual rationality. However, this incentivizes P_j to complete the task. This part differs from that in [35], where no such penalty is imposed and individual rationality is achieved. □

5. DFF: Discouraging Free-riding and False-reporting without Arbitration

EFF successfully eliminates free-riding and false-reporting with the help from a trusted third party for arbitration. However, it is not true that an arbitration is always available. In this section, we present another mechanism **DFF**, which, without any arbitration, discourages free-riding and false-reporting, while still guaranteeing several economic properties.

A. *Description of* DFF

Same as **EFF**, the first part of **DFF** applies the auction $\mathbb{M}$. For each $(R_i, P_j) \in \mathcal{W}$, **DFF** collects a warranty $w_i^R = (1+\zeta)\beta_i$ from R_i and a warranty $w_j^P = (\zeta+\eta)\alpha_j^i$ from P_j, where $\zeta > 0$ and $\eta > 0$ are system parameters. In **DFF**, we use ζ to help discourage dishonest behavior, and η to help incentivize providers to complete their assigned tasks.

After the auction, R_i and P_j pay their warranties to the platform. We model the post-auction part of **DFF** as another extensive-form game. P_j first decides whether or not to work on T_i, and devotes the corresponding effort. Then both R_i and P_j submit their independent reports on the status of T_i. With the reports from R_i and P_j, the platform decides the final payments $\bar{\beta}_i$ and $\bar{\alpha}_j^i$ for each of the following cases:

- When R_i and P_j both submit **C**, we have $\bar{\beta}_i = \zeta\beta_i$ and $\bar{\alpha}_j^i = (\zeta+\eta)\alpha_j^i + \alpha_j^i$. This means that R_i pays β_i to the platform and P_j receives α_j^i from the platform.
- When R_i and P_j both submit **I**, we have $\bar{\beta}_i = (1+\zeta)\beta_i$ and $\bar{\alpha}_j^i = \zeta\alpha_j^i$. This means that R_i pays nothing to the platform and P_j pays $\eta\alpha_j^i$ to the platform.
- When R_i submits **C** and P_j submits **I**, we have $\bar{\beta}_i = \zeta\beta_i$ and $\bar{\alpha}_j^i = \zeta\alpha_j^i$. This means that R_i pays β_i to the platform and P_j pays $\eta\alpha_j^i$ to the platform.
- When R_i submits **I** and P_j submits **C**, we have $\bar{\beta}_i = 0$ and $\bar{\alpha}_j^i = 0$. This means that R_i pays $(1+\zeta)\beta_i$ to the platform and P_j pays $(\zeta+\eta)\alpha_j^i$ to the platform.

The formal description of **DFF** is presented as Algorithm 2.

Algorithm 2: DFF

1 $(\mathcal{W}, \beta, \alpha) \leftarrow \mathbb{M}(\mathbf{b}, \mathbb{A})$;
2 $\forall (R_i, P_j) \in \mathcal{W}, w_i^R \leftarrow (1+\zeta)\beta_i, w_j^P \leftarrow (\zeta+\eta)\alpha_j^i$;
3 R_i submits w_i^R and P_j submits w_j^P to the platform;
4 $\bar{\beta}_i \leftarrow w_i^R$; $\bar{\alpha}_j^i \leftarrow w_j^p$;
5 P_j decides whether or not to work on T_i, and devotes the corresponding effort;
6 R_i and P_j submit independent reports on the status of T_i to the platform;
7 **if** *(Both R_i and P_j submit* **C***)* **then**
8 | $\bar{\beta}_i \leftarrow \bar{\beta}_i - \beta_i$; $\bar{\alpha}_j^i \leftarrow \bar{\alpha}_j^i + \alpha_j^i$;
9 **else if** *(Both R_i and P_j submit* **I***)* **then**
10 | $\bar{\alpha}_j^i \leftarrow \bar{\alpha}_j^i - \eta\alpha_j^i$;
11 **else if** *(R_i submits* **C** *and P_j submits* **I***)* **then**
12 | $\bar{\beta}_i \leftarrow \bar{\beta}_i - \beta_i$; $\bar{\alpha}_j^i \leftarrow \bar{\alpha}_j^i - \eta\alpha_j^i$;
13 **else**
14 | $\bar{\beta}_i \leftarrow \bar{\beta}_i - (1+\zeta)\beta_i$; $\bar{\alpha}_j^i \leftarrow \bar{\alpha}_j^i - (\zeta+\eta)\alpha_j^i$;
15 **end**
16 The platform returns $\bar{\beta}_i$ to R_i and $\bar{\alpha}_j^i$ to P_j, respectively.

Based on the final payments $\bar{\beta}_i$ and $\bar{\alpha}_j^i$ from Algorithm 2, utilities of R_i and P_j can be computed based on equations (3.1) and (3.2), respectively. These utility values are shown in Table 3. Notations in Table 3 represent similar meanings as those in Table 2.

We pick two entries and explain how the corresponding utilities are derived. First, we explain the entry where T_i is completed and both R_i and P_j submit **C** (marked in red in Table 3). In Line 2 of Algorithm 2, we have $w_i^R = (1+\zeta)\beta_i$ and $w_j^P = (\zeta+\eta)\alpha_j^i$. In Line 4, we have $\bar{\beta}_i = (1+\zeta)\beta_i$ and $\bar{\alpha}_j^i = (\zeta+\eta)\alpha_j^i$. Because both R_i and P_j submit **C**, in Line 8, we have $\bar{\beta}_i = (1+\zeta)\beta_i - \beta_i = \zeta\beta_i$ and $\bar{\alpha}_j^i = (\zeta+\eta)\alpha_j^i + \alpha_j^i$. Since P_j has devoted effort on T_i and T_i is completed, we have $y_j^i = 1$ and $x_i = 1$. Based on equations (3.1) and (3.2), R_i's utility is $v_i - \beta_i$ and P_j's utility is $\alpha_j^i - c_j^i$. Next, we explain the entry where T_i is incomplete, R_i submits **I**, and P_j submits **C** (marked in blue in Table 3). In Line 2, we have $w_i^R = (1+\zeta)\beta_i$ and $w_j^P = (\zeta+\eta)\alpha_j^i$. In Line 4, we have $\bar{\beta}_i = (1+\zeta)\beta_i$ and $\bar{\alpha}_j^i = (\zeta+\eta)\alpha_j^i$. Because R_i submits **I** and P_j submits **C**, in Line 14, we have $\bar{\beta}_i = (1+\zeta)\beta_i - (1+\zeta)\beta_i = 0$ and $\bar{\alpha}_j^i = (\zeta+\eta)\alpha_j^i - (\zeta+\eta)\alpha_j^i = 0$. Because T_i is incomplete, we have $y_j^i = 0$ and $x_i = 0$. Based on equations (3.1) and (3.2), R_i's utility is $-\beta_i - \zeta\beta_i$ and P_j's utility is $-\zeta\alpha_j^i - \eta\alpha_j^i$.

B. *Analysis of* DFF

In this subsection, we introduce the definition of semi-truthfulness and prove several properties of **DFF**.

Definition 5.1: A mechanism is *semi-truthful* if each individual has no positive increment in utility when it unilaterally behaves dishonestly while others are being honest. □

Comparing the definitions of semi-truthfulness and truthfulness, semi-truthfulness requires a stronger requirement, assuming that the other individuals are honest, while truthfulness has no such assumption.

Theorem 2: **DFF** is semi-truthful, transaction-wise budget-balanced, and computationally efficient. □

TABLE 3
UTILITIES OF R_i AND P_j IN **DFF** FOR EACH $(R_i, P_j) \in \mathcal{W}$, THE ENTRY FOR EACH EQUILIBRIUM IS MARKED BY ✓.

R_i \ P_j	T_i completed		T_i not completed	
	C	**I**	**C**	**I**
C	$(v_i - \beta_i, \alpha^i_j - c^i_j)$✓	$(v_i - \beta_i, -c^i_j - \eta\alpha^i_j)$	$(-\beta_i, \alpha^i_j)$	$(-\beta_i, -\eta\alpha^i_j)$
I	$(v_i - \beta_i - \zeta\beta_i, -c^i_j - \zeta\alpha^i_j - \eta\alpha^i_j)$	$(v_i, -c^i_j - \eta\alpha^i_j)$	$(-\beta_i - \zeta\beta_i, -\zeta\alpha^i_j - \eta\alpha^i_j)$✓	$(0, -\eta\alpha^i_j)$

To prove Theorem 2, we need the following three lemmas.

Lemma 5.1: **DFF** is semi-truthful. □

Proof: Suppose that (R_i, P_j) is in $\mathcal{W}$ when R_i bids v_i and P_j bids $\mathbf{c}_j$. If P_j completes T_i and no individual lies, R_i's utility is $v_i - \beta_i \geq 0$ and P_j's utility is $\alpha^i_j - c^i_j \geq 0$. Suppose that R_i bids a value deviating from v_i, it will not pay a lower payment due to the truthfulness of $\mathbb{M}$. If R_i lies by reporting **I** instead of **C**, and P_j does not lie and reports **C**, R_i's utility would be $v_i - (1+\zeta)\beta'_i \leq v_i - (1+\zeta)\beta_i \leq v_i - \beta_i$, where β'_i is the corresponding payment after R_i changes its bid. Thus, R_i has no incentive to lie when T_i is completed and P_j is honest. For P_j, by deviating its ask from $\mathbf{c}_j$, it will not receive a higher payment. If P_j lies by reporting **I**, and R_i does not lie and reports **C**, P_j's corresponding utility would be $-\eta{\alpha'}^i_j - c^i_j \leq {\alpha'}^i_j - c^i_j \leq \alpha^i_j - c^i_j$, where ${\alpha'}^i_j$ is the corresponding payment of P_j after P_j changes its bid. Thus, P_j has no incentive to lie when T_i is completed and R_i is honest. If P_j chooses not to complete T_i and reports **I** honestly, its utility would be $-\eta{\alpha'}^i_j < 0$. If P_j reports **C** dishonestly, since R_i is honest and reports **I**, P_j's utility would be $-(\eta + \zeta){\alpha'}^i_j < 0$. Thus, P_j would complete T_i. Therefore, if R_i or P_j knows that the other one is honest, it would be honest.

Suppose that R_i loses the auction by bidding $b_i = v_i$. Its original utility is 0. Suppose that R_i changes its bid. If it still loses the auction, its utility remains to be 0. If it wins the auction and P_j is assigned to T_i, R_i's payment in the auction β_i is no smaller than v_i since $\mathbb{M}$ is truthful. If T_i is completed, according to Table 2, R_i's utility is no larger than $v_i - \beta_i \leq v_i - b_i = 0$ since P_j is honest and reports **C**. If T_i is incomplete, R_i could not have a positive utility. Thus, R_i could not benefit from unilaterally being dishonest.

Suppose that P_j loses the auction by asking $\mathbf{a}_j = \mathbf{c}_j$. Its original utility is 0. Suppose that P_j changes its ask. If it still loses the auction, its utility remains to be 0. If it wins the auction and is assigned to T_i, P_j's payment in the auction α^i_j is no larger than c^i_j since $\mathbb{M}$ is truthful. If P_j completes T_i, according to Table 2, its utility is no larger than $\alpha^i_j - c^i_j \leq a^i_j - c^i_j = 0$. If P_j does not complete T_i, it utility is no more than $-\theta\alpha^i_j \leq 0$ since R_i is honest and reports **I**. Thus, P_j could not benefit from unilaterally being dishonest.

To sum up all cases, **DFF** is semi-truthful ■

Remark 5.1: With semi-truthfulness, we know that there is one equilibrium left in correspondence with each status of T_i. If P_j completes T_i, both R_i and P_j would report **C**. If P_j does not complete T_i, both R_i and R_j would report **I**. Comparing these two equilibria, P_j has a utility of $\alpha^i_j - c^i_j > 0$ in the first equilibrium, and a utility of $-\eta\alpha^i_j < 0$ in the second one. Thus, *P_j prefers to complete T_i, which makes the first equilibrium the unique SPE and encourages all winning providers to complete their assigned tasks, while discouraging free-riding and false-reporting.* Note that in [35], the simultaneous game computes two equilibria with no preference of P_j to complete T_i or not. However, by applying the extensive-form game instead of the simultaneous game, we have one unique SPE which incentivizes all winning providers to complete the tasks and discourages dishonest behaviors. □

Lemma 5.2: **DFF** is transaction-wise budget-balanced. □

Proof: Since $\mathbb{M}$ is transaction-wise budget-balanced, we have $\beta_i \geq \alpha^i_j$ for each $(R_i, P_j) \in \mathcal{W}$. If both R_i and P_j submit **C**, the platform utility generated from this winning pair is $U(R_i, P_j) = \beta_i - \alpha^i_j \geq 0$. If R_i submits **I** and P_j submits **C**, $U(R_i, P_j) = (1+\zeta)\beta_i + (\zeta + \eta)\alpha^i_j \geq 0$. If R_i submits **C** and P_j submits **I**, $U(R_i, P_j) = \beta_i + \eta\alpha^i_j \geq 0$. If both of them submit **I**, $U(R_i, P_j) = \eta\alpha^i_j \geq 0$. Thus, **DFF** is transaction-wise budget-balanced. ■

Lemma 5.3: **DFF** is computationally efficient. □

The proof is the same as that of Lemma 4.3 since the two mechanisms have the same time complexity.

Lemmas 5.1 − 5.3 complete the proof for Theorem 2.

Remark 5.2: Truthfulness is not guaranteed by **DFF**. For instance, when T_i is incomplete and R_i reports **C**, P_j may benefit by reporting **C** dishonestly instead of reporting **I**. Individual rationality is another economic property that **DFF** does not guarantee. Because when T_i is incomplete and P_j lies, R_i's utility is negative even if R_i reports honestly. □

6. PERFORMANCE EVALUATION

To evaluate the performance of **EFF** and **DFF**, we implemented both mechanisms, and carried out extensive testing on various cases. In both **EFF** and **DFF**, we used TASC [33] as the auction mechanism $\mathbb{M}$. We chose Maximum Matching Algorithm [30] as the bipartite matching algorithm used in TASC. The tests were run on a PC with a 3.5 GHz CPU, and 16 GB memory.

Remark 6.1: TASC [33] is a double auction based incentive mechanism proposed for cooperative communication, where the relay nodes offer relay services for rewards. TASC is truthful, individually rational, transaction-wise budget-balanced, and computationally efficient. TASC consists of two major steps. First, it applies a bid-independent bipartite matching algorithm to compute an assignment from the buyers to the sellers. Then it applies the McAfee [21] double auction to determine final payments for all buyers and sellers. □

Performance Metrics: We first studied the impact of the size of users on the running time as a demonstration of computational efficiency, and compare the running time of **EFF** and **DFF** with that of TASC. We next studied the impact of the size of users and user behavior on the platform utility. Finally, we studied the utilities of providers and requesters.

A. Simulation Setup

For **EFF**, we set $\tau_i = 10$ for all T_i and $\theta = 0.1$. For **DFF**, we set $\zeta = 0.6$ and $\eta = 0.1$. Note that this setting is for simplicity only. For properties of τ_i, θ, ζ, and η, see Section 4 and Section 5, respectively. Valuations, costs, bids, and asks are uniformly distributed over $[0, 20]$. All results in Figs. 1-8 are averaged over $50,000$ runs for each configuration.

To study the behavior of the running time as a function of the number of users, we first fixed $m = 100$ and let n increase from 10 to 1000; then we fixed $n = 100$ and let m increase from 10 to 1000. The corresponding results are reported in Fig. 1 and Fig. 2, respectively.

To study the behavior of the platform utility as a function of the number of users, we first fixed $m = 100$ and let n increase from 10 to 1000; then we fixed $n = 100$ and let m increase from 10 to 1000. We chose the scenario where all providers complete their tasks and all individuals are honest (we will present the impact of dishonest users with incomplete tasks on the platform utilities later). In this case, according to Algorithm 1, Algorithm 2, and equation (3.3), the platform utilities in **EFF** and **DFF** are the same. Thus, we only show the platform utilities in **EFF**, and the corresponding results are reported in Fig. 3.

To study the platform utility as a function of the percentage of completed tasks in **DFF**, we fixed $m = 100$ and $n = 100$, and let the percentage of completed tasks vary from 0% to 100% with an increment of 5%. To study the impact of the number of honest users (who submit honest reports) on platform utility, we first set all requesters to be honest and monitor different percentages of honest providers with different system parameter values of η and ζ in Fig. 4 and Fig. 5, respectively. Then we set all providers to be honest, and monitor different percentages of honest requesters in Fig. 6.

To study the platform utility as a function of the percentage of completed tasks in **EFF**, we fixed $m = 100$ and $n = 100$, and let the percentage of completed tasks vary from 0% to 100% with an increment of 5%. The corresponding results are presented in Fig. 7 and Fig. 8.

To study the utilities of requesters and providers, we set $m = n = 100$, and ran both **EFF** and **DFF**. The corresponding results are reported in Figs. 9-12, where R_{87} is a randomly picked requester and P_{31} is the corresponding provider, with $v_{87} = 17$, $c_{31}^{87} = 4$. R_{87} is matched up with P_{31} by the Maximum Matching Algorithm.

B. Simulation Results

Running Time

We study the impact of the size of users on the running time as a demonstration of computational efficiency. Since **EFF** and **DFF** have the same running time, we only show the running time of **EFF**. From Fig. 1, we observe that with the increase of m, the running time increases quadratically when $m < n$, then grows linearly when $m > n$. This is because the first part of TASC is a bipartite matching algorithm [30], which takes $O(\min\{m, n\} \times |E|)$ time, where $|E|$ is the number of edges in the graph and is bounded by $O(mn)$. Let $l = \min\{m, n\}$. The total time complexity is $O(l|E| + l \log l)$. Thus, in Fig. 1, when $m < n = 100$, the running time is $O(m^2)$; when $m > n = 100$, the running time is $O(m)$. Fig. 2 can be interpreted by a similar approach. Comparing Fig. 1 and Fig. 2, we observe that the impact from m and n are almost the same. Another observation is that the running time of **EFF** is almost the same as that of TASC, which implies that applying **EFF** or **DFF** does not introduce a high computation cost to $\mathbb{M}$.

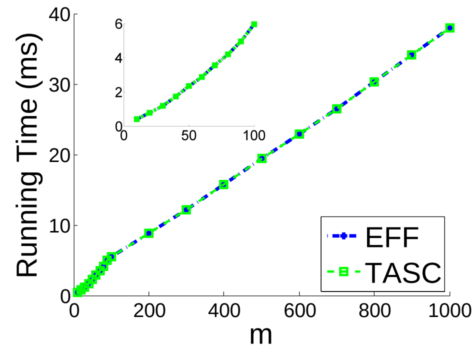


Fig. 1. Running time of **EFF** and TASC with $n = 100$

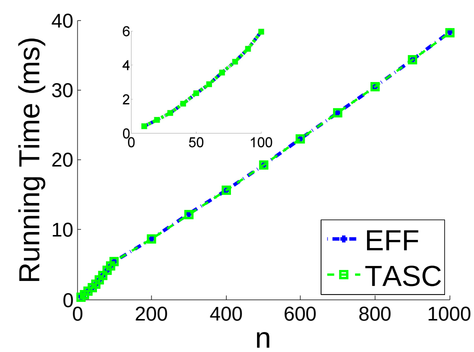
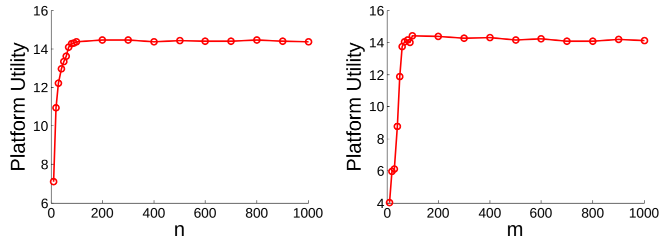


Fig. 2. Running time of **EFF** and TASC with $m = 100$

Platform Utility

We study the platform utility of **EFF** as a function of the number of providers and requesters in Fig. 3. We observe that the platform utility is non-negative, which is guaranteed by the transaction-wise budget-balance of **EFF**. The platform utility increases at first, then stays steady when $m > 100$. This is because the platform utility depends on not only the final payments, but also $|\mathcal{W}|$, which is bounded by $\min\{m, n\}$. Therefore, when $m < n$, $|\mathcal{W}|$ increases and the platform utility increases. When $|\mathcal{W}|$ reaches $\min\{m, n\}$, the platform utility stays at a steady level. Since in this scenario, the platform utilities in **EFF** and **DFF** are the same. we only show the platform utilities in **EFF**.



(a) (b)

Fig. 3. Platform Utility in **EFF**: (a) $m = 100$, (b) $n = 100$.

We study the impact of user behavior on the platform utility in **DFF**. The results are presented in Fig. 4, Fig. 5, and Fig. 6, respectively. In Fig. 4 and Fig. 5, we observe that when all requesters are honest in **DFF**, with the increasing percentage of completed tasks, the platform utility decreases. This is because with more tasks completed, the platform collects less penalties from providers for incomplete tasks. However, in Fig. 5 where we set $\eta = 0.3$ and $\zeta = 4$, with the increasing percentage of completed tasks, the platform utility increases. This is because with more completed tasks, the platform collects more penalties from the situation where providers are lying on their reports. Comparing Fig. 4 and Fig. 5, the different trends are due to different system parameter values. Since it is not the major concern of this paper on how to set the values for τ_i, η, and ζ, we do not provide theoretical analysis on the impact of τ_i, η and ζ on the platform utility. In Fig. 4, Fig. 5, and Fig. 5, we can observe that with more honest users, the platform utility decreases. This is because when R_i and P_j both behaves honestly for each $(R_i, P_j) \in \mathcal{W}$, the platform would not collect penalties from them for being dishonest.

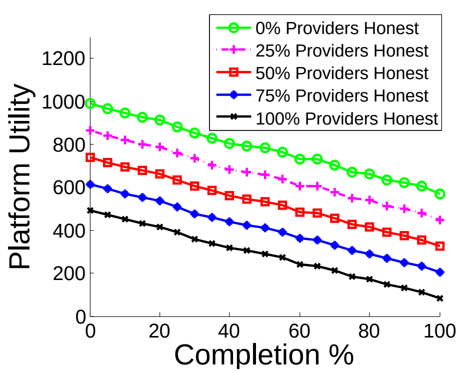


Fig. 4. Platform utilities in **DFF** with honest requesters, $\eta = 0.1$ and $\zeta = 0.6$

Fig. 6 shows the platform utilities of **EFF** as a function of the number of completed tasks, and it can be interpreted by a similar approach as that of Fig. 5. However, it can be proved theoretically that changing system constants η and ζ would not change the increasing trend in Fig. 6. Since it is not the major concern in this paper, we do not show the analysis here.

We study the impact of user behavior on the platform utility in **EFF**, and the results are shown in Fig. 7 and Fig. 8, respectively. We observe that when all requesters are honest in **EFF**, with the increasing percentage of the completed

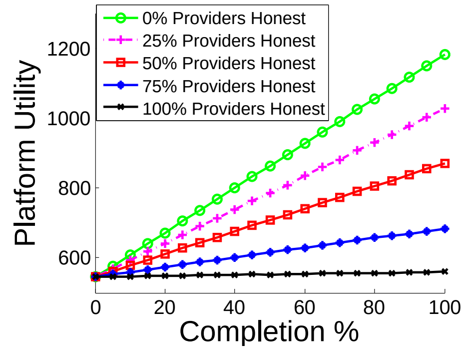


Fig. 5. Platform utilities in **DFF** with honest requesters, $\eta = 0.3$ and $\zeta = 4$

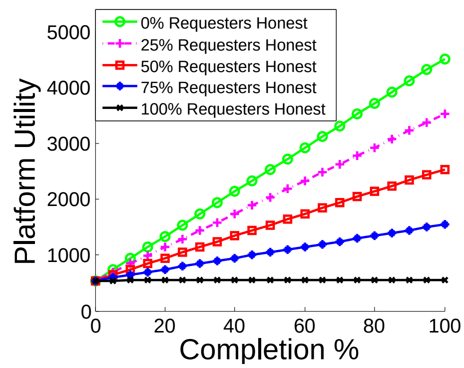


Fig. 6. Platform utilities in **DFF** when all providers are honest

tasks, the platform utility decreases. This is because with more tasks completed, the platform collects less penalties from the providers for incomplete tasks. Another observation is that the number of honest users does not have an impact on the platform utility. This is because in **EFF**, if an individual submits a dishonest report, it pays τ_i to the platform. However, the platform needs to pay τ_i for arbitration. Thus, the number of honest users does not influence the platform utility in **EFF**.

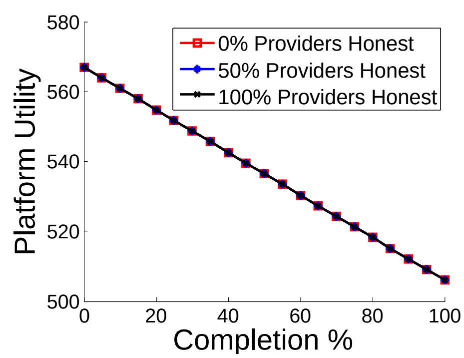


Fig. 7. Platform utilities in **EFF** with honest providers

Individual Utility

We monitor the utilities of R_{87} and P_{31} in **EFF**. Fig. 9 shows the utilities of R_{87} and P_{31}, where P_{31} completed the assigned task in **EFF**. From Fig. 9, we observe that neither R_{87} nor P_{31} can have a utility higher than that it has when bidding $b_{87} = v_{87} = 17$ and asking $a_{31}^{87} = c_{31}^{87} = 4$. From Fig. 9b,

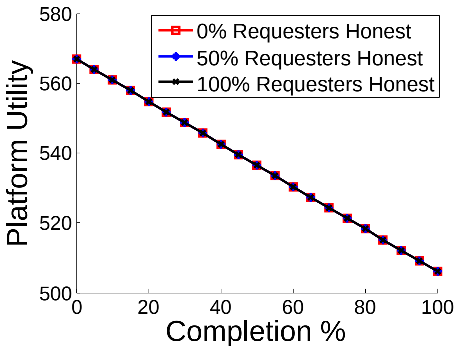


Fig. 8. Platform utilities in **EFF** with honest requesters

we observe that reporting **C** is the dominant strategy for P_{31}. According to Fig. 9a, R_{87} would report **C** when P_{31} reports **C**. Thus, both R_{87} and P_{31} would report **C**.

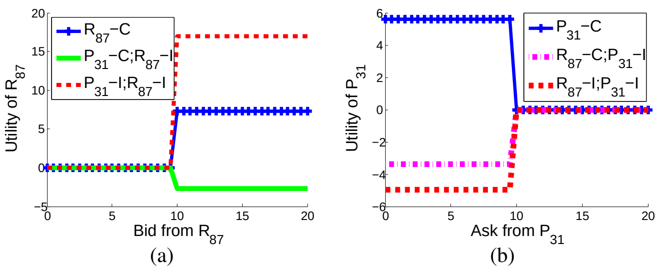


Fig. 9. Completed tasks in **EFF**: (a) Utility of R_{87}, (b) Utility of P_{31}.

Fig. 10 shows the utilities when P_{31} does not complete the task in **EFF**. Again, we observe that no individual can get a positive utility increment by being dishonest using a similar analysis approach. Hence, both R_{87} and P_{31} would report **I**.

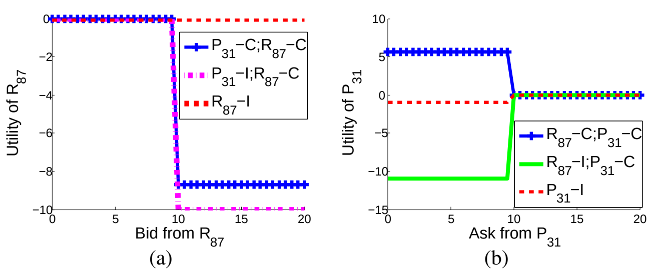


Fig. 10. Incomplete tasks in **EFF**: (a) Utility of R_{87}, (b) Utility of P_{31}.

Comparing the utilities of P_{31} in **EFF** when it completes the task and when it does not complete the task (represented by the blue line in Fig. 9b and the red line in Fig. 10b), P_{31} would choose to complete the task for a higher utility.

Next we monitor the utilities of R_{87} and P_{31} in **DFF**. From Fig. 11, we observe that when P_{31} completes the task, there is no incentive for an individual to be dishonest when the other is honest. Thus, both R_{87} and P_{31} would report **C**.

Fig. 12 shows the semi-truthfulness when P_{31} fails to complete the task in **DFF**. Again, we observe that there is no incentive for an individual to be dishonest when the other is honest. Hence, both R_{87} and P_{31} would report **I**.

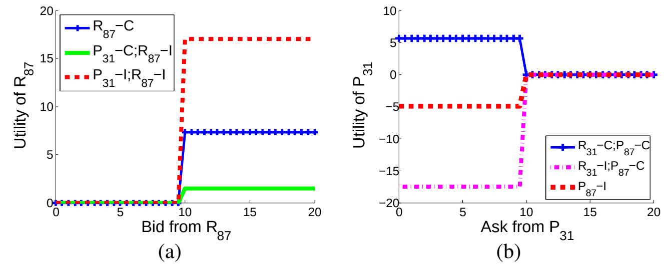


Fig. 11. Completed tasks in **DFF**: (a) Utility of R_{87}, (b) Utility of P_{31}.

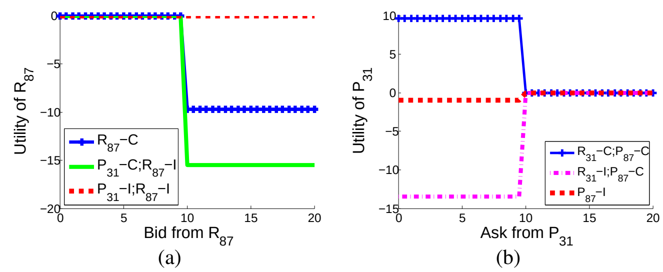


Fig. 12. Incomplete tasks in **DFF**: (a) Utility of R_{87}, (b) Utility of P_{31}.

Comparing the utilities of P_{31} in **DFF** when it completes the task and when it does not complete the task (represented by the blue line in Fig. 11b and the red line in Fig. 12b, respectively), P_{31} would choose to complete the task for a higher utility.

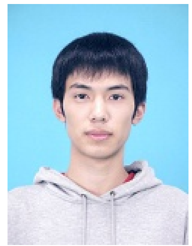
7. Conclusions

In this paper, we proposed two novel mechanisms to tackle free-riding and false-reporting in crowdsourcing. We first presented **EFF** and proved that with arbitration, **EFF** eliminates free-riding and false-reporting, while guaranteeing truthfulness, transaction-wise budget-balance, and computational efficiency. We then presented **DFF** and proved that without arbitration, **DFF** is semi-truthful, while guaranteeing transaction-wise budget-balance and computational efficiency. Another feature of our mechanisms is that providers are incentivized to complete their assigned tasks. We implemented both **EFF** and **DFF**. Extensive numerical results are presented to study the performance of our proposed mechanisms.

References

[1] [Online]. Available: http://www.yelp.com/
[2] [Online]. Available: http://answers.yahoo.com/
[3] [Online]. Available: https://www.mturk.com/mturk/welcome
[4] [Online]. Available: https://www.uber.com/
[5] P. Azar, J. Chen, and S. Micali, "Crowdsourced Bayesian auctions," in *Proc. ACM ITCS '12*, pp. 236–248.
[6] S. Chawla, J. D. Hartline, and B. Sivan, "Optimal crowdsourcing contests," in *Proc. ACM-SIAM SODA '12*, pp. 856–868.
[7] E. H. Clarke, "Multipart pricing of public goods," *Public Choice*, vol. 11, pp. 17–33, 1971.
[8] K. Deshmukh, A. V. Goldberg, J. D. Hartline, and A. R. Karlin, "Truthful and competitive double auctions," in *ESA'02*, pp. 361–373.
[9] D. DiPalantino and M. Vojnovic, "Crowdsourcing and all-pay auctions," in *Proc. ACM EC'09*, pp. 119–128.
[10] M. Feldman, C. Papadimitriou, J. Chuang, and I. Stoica, "Free-riding and whitewashing in peer-to-peer systems," *IEEE JSAC*, vol. 24, 2006.

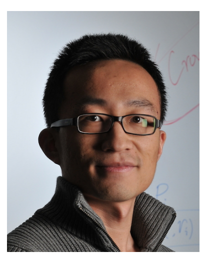
[11] Z. Feng, Y. Zhu, Q. Zhang, H. Zhu, J. Yu, J. Cao, and L. Ni, "Towards truthful mechanisms for mobile crowdsourcing with dynamic smartphones," in *Proc. IEEE ICDCS'14*, pp. 11–20.
[12] D. Fundenberg and J.Tirole, *Game Theory*. MIT, 2010.
[13] A. V. Goldberg, J. D. Hartline, and A. Wright, "Competitive auctions and digital goods," in *Proc. ACM-SIAM SODA'01*, pp. 735–744.
[14] T. Groves, "Incentives in Teams," *Econometrica*, vol. 41, pp. 617–31, 1973.
[15] S. Guo, Y. Gu, B. Jiang, and T. He, "Opportunistic flooding in low-duty-cycle wireless sensor networks with unreliable links," in *Proc. ACM MOBICOM'09*.
[16] J. Howe, *Crowdsourcing: Why the Power of the Crowd Is Driving the Future of Business*. Crown Publishing Group, 2008.
[17] S. Jain, Y. Chen, and D. Parkes, "Designing incentives for online question and answer forums," in *Proc. ACM EC'09*, pp. 129–138.
[18] R. Lavi and C. Swamy, "Truthful and near-optimal mechanism design via linear programming," in *Proc. IEEE FOCS'05*, pp. 595–604.
[19] X.-Y. Li, Z. Sun, and W. Wang, "Cost sharing and strategyproof mechanisms for set cover games," in *Proc. STACS 2005*, pp. 218–230.
[20] W. Mason and D. J. Watts, "Financial incentives and the "performance of crowds"," in *Proc. ACM HCOMP'09*, pp. 77–85.
[21] R. McAfee, "A dominant strategy double auction," *Journal of Economic Theory*, vol. 56, pp. 434 – 450, 1992.
[22] A. Mu'Alem and N. Nisan, "Truthful approximation mechanisms for restricted combinatorial auctions," *Games and Economic Behavior*, vol. 64, no. 2, pp. 612–631, 2008.
[23] R. B. Myerson, *Game Theory: Analysis of Conflict*. Harvard University Press, 1991.
[24] J. F. Nash Jr., "Equilibrium points in n-person games," *PNAS*, vol. 36, pp. 48–49, 1950.
[25] X. Sheng, J. Tang, X. Xiao, and G. Xue, "Sensing as a service: Challenges, solutions and future directions," *IEEE Sensors Journal*, vol. 13, pp. 3733–3741, 2013.
[26] Y. Singer and M. Mittal, "Pricing tasks in online labor markets," in *Human Computation, AAAI Workshops*, vol. WS-11-11, 2011.
[27] A. Singla and A. Krause, "Truthful incentives in crowdsourcing tasks using regret minimization mechanisms," in *Proc. WWW'13*, pp. 1167–1178.
[28] W. Vickrey, "Counterspeculation, auctions, and competitive sealed tenders," *Journal of Finance*, vol. 16, pp. 8–37, 1961.
[29] Q. Wang, K. Ren, and X. Meng, "When cloud meets ebay: Towards effective pricing for cloud computing," in *Proc. IEEE INFOCOM'12*, pp. 936–944.
[30] R. J. Wilson, *Introduction to Graph Theory*. John Wiley & Sons, Inc., 1986.
[31] W. Xu, H. Huang, Y.-e. Sun, F. Li, Y. Zhu, and S. Zhang, "DATA: A double auction based task assignment mechanism in crowdsourcing systems," in *Proc. IEEE CHINACOM'12*, pp. 172–177.
[32] D. Yang, G. Xue, X. Fang, and J. Tang, "Incentive mechanisms for crowdsensing: Crowdsourcing with smartphones," *IEEE/ACM Transactions on Networking*, vol. PP, no. 99, pp. 1–13, 2015.
[33] D. Yang, X. Fang, and G. Xue, "Truthful auction for cooperative communications," in *Proc. ACM MOBIHOC'11*, pp. 9:1–9:10.
[34] D. Yang, G. Xue, X. Fang, and J. Tang, "Crowdsourcing to smartphones: Incentive mechanism design for mobile phone sensing," in *Proc. ACM MOBICOM'12*, pp. 173–184.
[35] X. Zhang, G. Xue, R. Yu, D. Yang, and J. Tang, "You better be honest: Discouraging free-riding and false-reporting in mobile crowdsourcing," in *Proc. IEEE GLOBECOM'14*.
[36] Y. Zhang and M. van der Schaar, "Reputation-based incentive protocols in crowdsourcing applications," in *Proc. IEEE INFOCOM'11*, pp. 2140–2148.
[37] D. Zhao, X.-Y. Li, and H. Ma, "How to crowdsource tasks truthfully without sacrificing utility: Online incentive mechanisms with budget constraint," in *Proc. IEEE INFOCOM'14*, pp. 1213–1221.



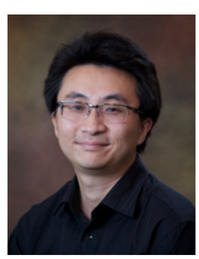
Xiang Zhang (Student Member 2013) received his B.S. degree from University of Science and Technology of China, Hefei, China, in 2012. Currently he is a Ph.D student in the School of Computing, Informatics, and Decision Systems Engineering at Arizona State University. His research interests include network economics and game theory in crowdsourcing and cognitive radio networks.

Guoliang Xue (Member 1996, Senior Member 1999, Fellow, 2011) is a Professor of Computer Science and Engineering at Arizona State University. He received the PhD degree (1991) in computer science from the University of Minnesota, Minneapolis, USA. His research interests include survivability, security, and resource allocation issues in networks. He is an Editor of *IEEE Network* and the Area Editor of *IEEE Transactions on Wireless Communications* for the area of Wireless Networking. He is an IEEE Fellow.

Ruozhou Yu (Student Member 2013) received his B.S. degree from Beijing University of Posts and Telecommunications, Beijing, China, in 2013. He is currently pursuing the Ph.D degree in the School of Computing, Informatics, and Decision Systems Engineering at Arizona State University. His interests include software-defined networking, data center networking, network function virtualization.



Dejun Yang (Student Member 2008, Member 2013) is the Ben L. Fryrear Assistant Professor of Computer Science at Colorado School of Mines. He received his PhD degree from Arizona State University in 2013 and BS degree from Peking University in 2007. His research interests lie in optimization and economic approaches to networks. He received Best Paper Awards at IEEE ICC'2011 and IEEE ICC'2012. He has served as a TPC member for many conferences including IEEE INFOCOM.



Jian Tang (Member 2006, Senior Member 2013) is an associate professor in the Department of Electrical Engineering and Computer Science at Syracuse University. He earned his Ph.D degree in Computer Science from Arizona State University in 2006. His research interests lie in the areas of Cloud Computing, Big Data and Wireless Networking. He received an NSF CAREER award in 2009. He serves as an editor for IEEE Transactions on Vehicular Technology and an editor for IEEE Internet of Things Journal.