Computer Networks 102 (2016) 157–171

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

An incentive mechanism with privacy protection in mobile crowdsourcing systems[☆]

Yingjie Wang[a], Zhipeng Cai[b,c,*], Guisheng Yin[b], Yang Gao[d], Xiangrong Tong[a], Guanying Wu[e]

[a] School of Computer and Control Engineering, Yantai University, Yantai, 264005, China
[b] College of Computer Science and Technology, Harbin Engineering University, Harbin, 150001, China
[c] Department of Computer Science, Georgia State University, Atlanta, GA 30303, USA
[d] School of Mathematics and Information Science, Yantai University, Yantai, 264005, China
[e] China Japan Friendship Hospital, Beijing, 100029, China

ARTICLE INFO

Article history:
Received 16 January 2015
Revised 13 March 2016
Accepted 28 March 2016
Available online 1 April 2016

Keywords:
Crowdsourcing
Incentive mechanism
Privacy
Auction
Reputation

ABSTRACT

In order to improve the efficiency and utility of mobile crowdsourcing systems, this paper proposes an incentive mechanism with privacy protection in mobile crowdsourcing systems. Combining the advantages of offline incentive mechanisms and online incentive mechanisms, this paper proposes an incentive mechanism that selects the worker candidates statically, and then dynamically selects winners after bidding. The proposed incentive mechanism includes two algorithms which are an improved two-stage auction algorithm (ITA) and a truthful online reputation updating algorithm (TORU). Through simulations, we verify the efficiency and effectiveness of the proposed incentive mechanism, which can solve the free-riding problem and improve the efficiency and utility of mobile crowdsourcing systems effectively.

1. Introduction

The market of smartphones has proliferated rapidly in the recent years and continues to expand. Mobile crowdsourcing refers to crowdsourcing activities on smartphones or other mobile devices. Thanks to the improved, technological smartphone features, including reliable GPS, high resolution cameras, and continuously advanced software, mobile phone users can work on crowdsourcing tasks with ease [1,2]. Nowadays, these tasks involve more than simple site descriptions. Mobile crowdsourcing can be used to collect data either passively or actively. Users who have smartphones equipped with GPS can be located to create movement profiles [3,4]. In active crowdsourcing, smartphone users upload data including restaurant photos, accurate addresses and businesses (geocoding) or information about menus [5,6]. Meanwhile, mobile crowdsourcing can help with disaster rescue by coordinating rescuers in real time and documenting damage situations. Data gathered via mobile crowdsourcing is up-to-date and accurate, and a large amount of data can be delivered very quickly [7].

A mobile crowdsourcing system is a new form of commercial crowdsourcing system. In traditional commercial crowdsourcing systems, such as Amazon Mechanical Turk [8], an employer submits a task to the crowdsourcing platform and defines how much the workers will be paid per task and how the workers have to provide proof of a completed task. Random workers from the crowd choose to work on the task and submit the required proof to the crowdsourcing platform. The work proof is forwarded to the employer, who pays the worker if the task is completed successfully. However, in mobile crowdsourcing, it is common that workers are coming and bidding for a specific task sequentially, and the decision on accepting or denying a worker's bidding must be made by the platform instantly upon the user's arrival. Therefore, compared with the traditional commercial crowdsourcing systems, mobile crowdsourcing systems need higher real-time performance. In addition, in order to obtain better benefit and effectiveness, there is a bidding process for workers in mobile crowdsourcing systems.

Realizing the great potential of the mobile phone sensing, many researchers have developed numerous applications and systems, such as Sensorly [9] for making cellular/WiFi network coverage maps, VTrack [10] for providing traffic information. However, the existing mobile crowdsourcing systems face a serious practical

☆ Fully documented templates are available in the elsarticle package on CTAN.

* Corresponding author.
E-mail addresses: towangyingjie@hotmail.com (Y. Wang), zcai@gsu.edu (Z. Cai), yinguisheng@hrbeu.edu.cn (G. Yin), togaoyang@163.com (Y. Gao), txr@ytu.edu.cn (X. Tong), wuguanying2008@163.com (G. Wu).

http://dx.doi.org/10.1016/j.comnet.2016.03.016
1389-1286/

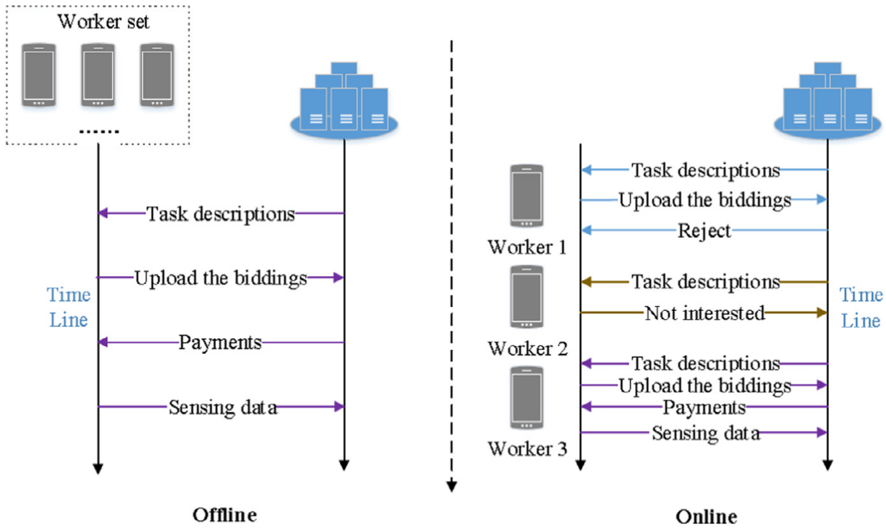


Fig. 1. Traditional offline settings and online settings for incentive mechanisms.

challenge: providing appropriate incentives for workers to participate and well-perform in tasks. More concretely, a requester needs to establish sufficient rewards to attract workers' contributions when workers do not solve tasks solely for altruistic motivations. For these reasons, designing an effective incentive mechanism to encourage workers' contributions is crucial to maintain the performance of crowdsourcing systems.

Therefore, how to maximize the social welfare is one of the most popular interests for mobile crowdsourcing systems. The establishment of incentive mechanisms becomes the focus in the research of optimizing mobile crowdsourcing systems. Traditional incentive mechanisms include two types which are offline settings and online settings. The mobile nature of these distributed computation and sensing powers further complicates the incentive mechanism design [11]. In a mobile crowdsoucing system, a task is described and posted by a requester together with the associated reward budget. If a worker is interested in a task, he will upload his bidding, *i.e.*, the solution (sensing time and sensing cost), to this requester [12]. According to this bidding, the requester can determine to accept this worker or reject this worker. The pioneer works mainly refer to the offline incentive mechanisms. The offline incentive mechanisms determine the winners in auction after all the participators upload their bidding [13]. These offline schemes assume that all the users present from the very beginning of one round of task distribution for bidding and cannot accept new bidding afterwards (shown in the left part of Fig. 1). In other words, the offline incentive mechanisms all fail in a more practical yet dynamic setting of mobile sensing. In order to resolve the problems of offline incentive mechanisms, Zhang et al. [14] proposed an online incentive mechanism, which is shown in the right part of Fig. 1. However, the online incentive mechanism fails to select the set of candidates from the workers' reputation database, resulting in inefficiency in the process of auction. Therefore, we combine offline settings and online settings to design incentive mechanism shown by Fig. 2. In the proposed incentive mechanism, the platform determines the worker set that can be assigned the given task based on the offline incentive mechanism; then once the selected workers arrive, there are transactions between platform and workers based on the online incentive mechanism. In addition, we add the privacy protection for the participant workers. Therefore, the proposed incentive mechanism can overcome the disadvantages of offline incentive mechanism and online incentive mechanism, protect workers' privacy, and improve the efficiency of mobile crowdsourcing system.

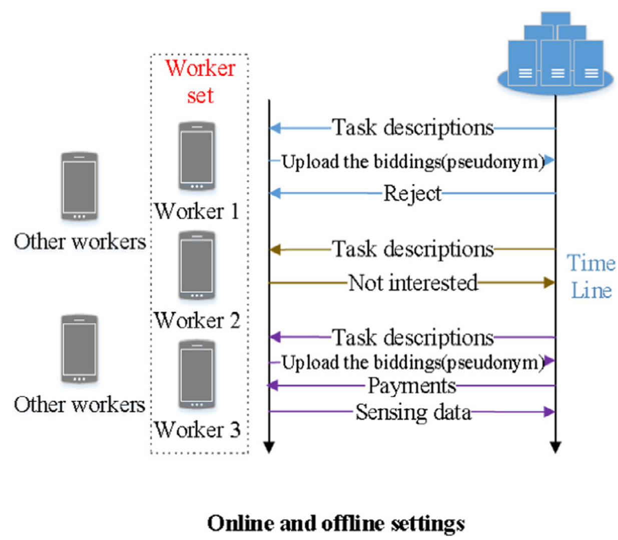


Fig. 2. The processing procedure for the proposed incentive mechanism.

In the process of an auction between workers and a platform, how to develop an auction algorithm is very important for improving the efficiency of mobile crowdsourcing systems [15]. In addition, there exists a lot of free-riding phenomena in complex networks such as social networks, computer networks and so on. Unfortunately, networks cannot automatically adjust the selections of nodes for trust strategies. The individuals in a network have the nature of selfishness, thus an individual prefers to select the strategies that can increase its benefit. The free-riders prefer to select a unreliable strategy as their first choice, resulting in the decrease of network benefit [16].

Therefore, how to establish an effective incentive mechanism to inspire the selection of reliable strategies is very important for complex networks. Mobile crowdsourcing systems are developed upon mobile social networks, and they are full of complexities, thus mobile crowdsourcing systems have the features of complex networks. Such features include a heavy tail in the degree distribution, a high clustering coefficient, assortativity or disassortativity among vertices, community structure, and hierarchical structure. Therefore, for mobile crowdsourcing systems, the establishment of an incentive mechanism is also an important research focus.

With the increase of network scale and number of mobile users, mobile crowdsourcing systems become more and more complex. Thus, an incentive mechanism needs to be established to adjust the benefit equilibrium between workers and a platform in mobile crowdsourcing systems, in order to solve the free-riding problem and promote the mobile crowdsourcing systems to develop steady [17,18].

According to the aforementioned reasons, this paper proposes an incentive mechanism with privacy protection in mobile crowdsourcing systems. The contributions of this paper are summarized as follows:

1. Combining the advantages of offline incentive mechanisms and online incentive mechanisms, we propose an incentive mechanism that selects the worker candidates statically, and then dynamically selects winners after bidding. Under this incentive mechanism framework, a privacy protection is proposed in order to protect the privacy of workers.
2. We design an improved two-stage auction algorithm (ITA) to determine the winners in real-time for the platform, which can overcome the unfairness problem and encourage workers to arrive in time.
3. We propose a truthful online reputation updating algorithm (TORU) to update workers' reputations, which can solve the free-riding problem in mobile crowdsourcing systems.
4. In order to verify the effectiveness of the proposed incentive mechanism, we compare the proposed incentive mechanism with some typical algorithms through simulations. The results show the advantages and improvements of our algorithms.

The rest of the paper is organized as follows. Section 2 presents the related works. Section 3 introduces the proposed incentive mechanism including ITA and TORU, and analyzes the properties of the proposed incentive mechanism. Section 4 illustrates the simulations, along with the parameter settings, followed by the result analysis and discussions. Finally, Section 5 concludes this paper.

2. Related work

In mobile crowdsourcing systems, selfishness and privacy protection problems have gained extensive attentions. Thus, how to establish effective incentive mechanisms is a challenging research focus in mobile crowdsourcing systems. Scholars have spent a lot of efforts on the selfishness and privacy protection problems.

2.1. On the aspect of incentive mechanisms

Offline settings and online settings are the two typical schemes. Yang et al. [13] proposed an offline incentive mechanism using a Stackelberg game, where the platform is the leader while the users are the followers. Two system models are considered: the platform-centric model where the platform provides a reward shared by the participating users, and the user-centric model where the users have more control over the payment they will receive. The scheme of an offline incentive mechanism is shown in the left part of Fig. 1. However, the offline incentive mechanisms assume that all the users will stay from the very beginning of one round of task distribution for bidding and cannot accept new biddings afterwards. In other words, the offline incentive mechanisms all fail in a more practical yet dynamic setting of mobile sensing [19,20].

Zhang et al. [14] proposed an online incentive mechanism. Two online incentive mechanisms based on online reverse auction are provided: threshold-based auction (TBA) and truthful online incentive mechanism (TOIM). However, the online incentive mechanisms fail to select the set of candidates from the workers' reputation database, resulting in inefficiency in the process of an auction.

2.2. On the aspect of auction algorithms

The two-stage auction algorithm is applied widely [21]. The process of a two-stage auction algorithm indicates that the first batch of users is rejected and used as the sample which enables making an informed decision on whether to accept the rest of the users. However, this method fails to guarantee the consumer sovereignty, since the first batch of users has no chance to win the auction no matter how low the cost is. This method automatically rejects the first batch of users, so that it encourages users to arrive late. In other words, the users who arrive early have no incentive to report their biddings, which may hinder the users' competition or even result in task starvation [22]. In addition, Sodagari et al. [23] proposed the on cost-sharing mechanisms in cognitive radio networks. They casted the issues to submodular class of games and showed how a link can be established between the truthful auctioning mechanism and the cost-sharing algorithm. However, this approach failed to consider the real-time auction, thus it is not suitable for the online network environment.

In [13], an auction-based incentive mechanism was proposed. The authors utilized the announced total reward R (budget) and user i's sensing plan t_i (user i's willingness on how long he wants to participate in the sensing task) to design a novel auction based on submodular function. In [24], the authors designed incentive compatible some mechanisms that maximize a requester's objective under a budget. It is known as *budget feasiblity* where the mechanism must be designed so that the sum of its payments does not exceed the budget. Therefore, budget and sensing plan are two important factors for designing an effective auction algorithm in crowdsourcing systems, such as Amazon Mechanical Turk, a successful crowdsourcing system that performs under a budget. However, in the above mechanisms, the auction thresholds are static and cannot be changed dynamically, which induces the unfair problem.

2.3. On the aspect of incentive and punishment strategies

In order to solve the free-riding problem in mobile crowdsourcing systems, a lot of incentive and punishment strategies were proposed. Many of the incentive mechanisms on crowdsourcing web sites rely on monetary rewards in the form of micro-payments [25–27]. The platform pays workers in the form of cash upon the completion of a task. However, the current pricing schemes fail to solve the social dilemma existing between the workers and platform. Zhang et al. [12] proposed a novel class of incentive protocols based on social norms which integrates reputation mechanisms with the existing pricing schemes currently implemented on the crowdsourcing web sites. However, some potential trustful workers may be isolated because of the unique threshold in this incentive mechanism, so that generates unfair problem.

In addition, according to privacy protection, researchers have highlighted security and privacy challenges. The typical proposal PEPSI [28] enable anonymous data collection from mobile users. PEPSI's extension [29], considered scenarios where external entities query specific users sensing data and proposed a scheme to hide which user matches a query. Cristofaro et al. [30] proposed a privacy-enhanced participatory sensing infrastructure on the basis of PEPSI. In addition, Li et al. [31] proposed a privacy-aware incentive scheme with a trustable third party (TTP), which can protect users' privacy. Wang et al. [32] proposed PEALS framework to support privacy-aware mobile crowdsourcing, which can help potential contributors assess the privacy and security risks they face

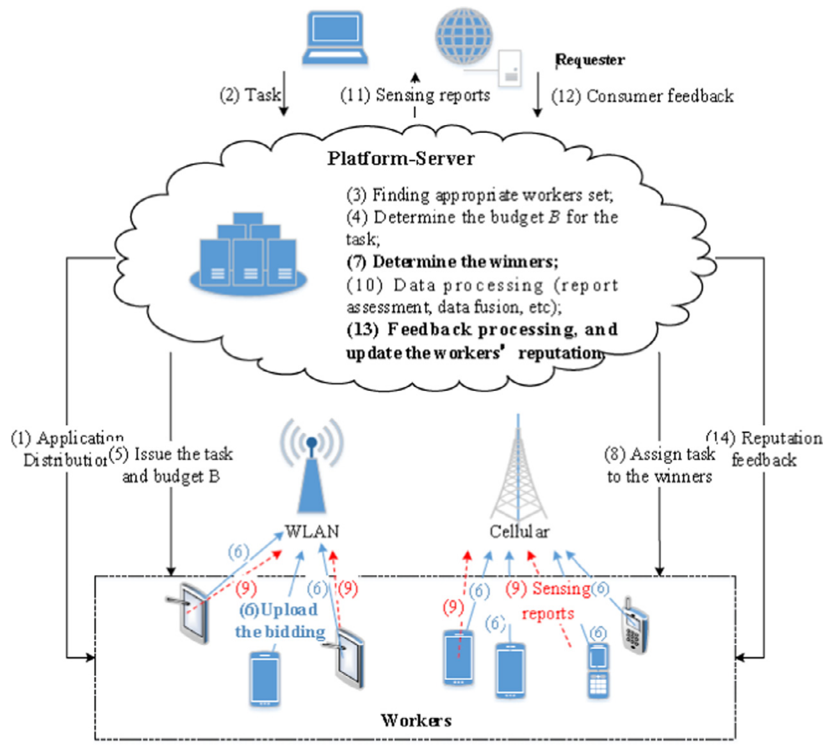


Fig. 3. The structure of a mobile crowdsourcing system.

with a mobile crowdsourcing system and make an informed decision about contributing. However, most of the proposals considered the condition of offline mechanisms, so they fail to consider the real-time property according to the online mechanisms.

Targeting on the above problems, this paper proposes an incentive mechanism with privacy protection in mobile crowdsourcing systems. Combining the advantages of offline incentive mechanisms and online incentive mechanisms, this paper proposes an incentive mechanism that selects the worker candidates statically, and then dynamically selects winners after bidding. Under this incentive mechanism framework, a privacy protection protocol for mobile crowdsourcing systems is designed. In addition, we design an improved two-stage auction algorithm to determine the winners for the platform, which can overcome the unfairness problem, and encourage workers to arrive in time. A truthful online reputation updating algorithm (TORU) is proposed to update workers' reputations, which can solve the free-riding problem in mobile crowdsourcing systems.

3. The proposed incentive mechanism

Similar to the work in [14], our objective is to design an online incentive mechanism with the following four properties:

1. *Computational efficiency*. An online mechanism is computationally efficient if it has a polynomial time complexity.
2. *Individual rationality*. A user will get nonnegative utility upon completing a sensing task.
3. *Profitability*. The platform will get nonnegative utility at the end of a sensing task.
4. *Truthfulness*. A mechanism is truthful, or incentive compatible, if a bidder cannot improve her utility by submitting a bidding price deviating from her true value in spite of others' bidding prices.

The mobile nature of these distributed computation and sensing powers further complicates the incentive mechanism design. In brief, it is common in practical mobile sensing that users are coming and bidding for a specific task sequentially, and the decision on accepting or denying a worker's bidding must be made by the platform instantly upon the worker's arrival. Nevertheless, pioneer works on incentive mechanism are static and offline, in which the concurrent presence of numerous smartphone candidates is required. These offline schemes assume that all the workers will stay from the very beginning of one round of task distribution for bidding and cannot accept new biddings afterwards. In other words, the offline mechanisms all fail in a more practical yet dynamic setting of mobile phone sensing.

In this paper, we combine offline incentive mechanism and online incentive mechanism to propose an incentive mechanism that selects the worker candidates statically, and then dynamically selects winners after bidding. Assuming the set of worker candidates is $C = (1, 2, \ldots, n)$, where n expresses the total number of worker candidates. Therefore, *zero arrival-departure* is considered in this paper because of the dynamic auction process. There are three cases in one transaction between the platform and a worker:

1. The platform issues the task and budget B to worker i who uploads bidding b_i and sensing plan t_i. The platform rejects worker i because of discontent.
2. The platform issues the task and budget B to worker i who is not interested in the task.
3. The platform issues the task and budget B to worker i who uploads bidding b_i and sensing plan t_i. The platform accepts worker i and gives the payment to worker i, then worker i uploads the sensing report.

Because the platform may reject the workers, it will inspire workers to upload appropriate values of bidding and sensing time. Therefore, in order to be the winner in the auction and obtain the reward, a worker will be inspired to perform well.

The structure of this mobile crowdsourcing system is shown in Fig. 3. In this paper, we divide the crowdsourcing process into 14 steps. Steps (6), (7) and (13) are the focus of this paper.

In this model, the platform announces a set $\Phi = (\varphi_1, \varphi_2, \ldots, \varphi_m)$ of tasks for the workers to select. According to the selected task, worker i has a *contribution value* $v_i > 0$ to the platform, and also has an *associated cost* c_i, which is private and other workers do not know it. Worker i's *bidding* is represented by b_i, where b_i is the reserved price of the service worker i wants to sell. The *sensing plan* of worker i is represented by t_i, which is the number of the time units during which worker i can provide the sensing service. Based on the special task, worker i first submits b_i and t_i to the platform. Upon receiving the bidding and sensing plans from all the workers, the platform selects a *subset of workers as winners W* and determines the *payment* p_i for each winning worker i. Therefore, the *utility of worker i* based on the submitted sensing plan is shown by Eq. (1):

$$u_i = \begin{cases} p_i - c_i, & if\ i \in W \\ 0, & otherwise \end{cases} \tag{1}$$

According to the reality, we define $t_i \geq 0$, where $t_i = 0$ represents that worker i will not participate in this task. In this paper, we define $c_i = \tau \times t_i$, where τ is the unit cost of workers, and $0 < \tau < 1$. In addition, p_i is determined by t_i and the budget B, and B represents *the budget for a specific task* determined by the platform.

In Eq. (1), c_i is determined by the submitted t_i and τ, however, c_i will evolve with the change of sensing time t_i at the end of sensing task. Therefore, the *real utility of worker i* at the end of sensing task is derived by Eq. (2):

$$u_i{}' = \begin{cases} p_i - c_i{}', & if\ i \in W \\ 0, & otherwise \end{cases} \tag{2}$$

where $c_i{}'$ represents the real associated cost that derived by $c_i{}' = \tau \times t_i{}'$, and $t_i{}'$ indicates the real sensing time of work i. According to the above analysis, p_i is derived by Eq. (3):

$$p_i = \frac{t_i}{T} \times B \tag{3}$$

where T indicates the *maximal sensing time* for this task, and we define $\frac{B}{T} \geq 1$. The *utility of platform* is defined by Eq. (4):

$$\bar{u} = \lambda \times log\left(1 + \frac{V(W)}{\lambda}\right) - P(W) \tag{4}$$

where $V(W) = \sum_{i \in W} v_i$ is the *total benefit of the platform*, and $P(W) = \sum_{i \in W} p_i$ is the *total payments for the workers*. In addition, v_i is the *specific contribution value* that worker i brings to the platform, and v_i is evaluated through the sensing time submitted by worker i. The log term in Eq. (4) captures the platform's marginal diminishing return on the selected workers, which conforms to the usual economic assumption [14,33]. λ is a system parameter that can control the gradient of the diminishing return, and $\lambda > 1$.

A worker will be isolated by the platform and is forbidden to interact with requesters and participate in any task if his reputation is low. In this case, the social norm does not require the worker to do anything. On the contrary, the platform activates the worker by allowing him to participate in tasks if his reputation is high, and the social norm requires the worker to devote a high level of efforts in his transactions.

3.1. The design of privacy protection for mobile crowdsourcing systems

In the process of sensing, some workers may want other worker's bidding, sensing data and updated reputation to adjust their strategies in order to get more benefit. Even worse, there may be malicious attackers in the network. Once they obtain others' private data, they attack the workers whose privacy leaked. In order to protect the privacy of workers who participates in the task, we design a privacy protection for mobile crowdsourcing systems. In this paper, we divide privacy protection into three stages: *uploading bidding* stage, *uploading sensing data* stage and *updating reputation* stage.

In this privacy protection, we give the system initialization and cryptographic schemes. i) Considering the computation efficiency reasons, we apply time-lapse cryptography (TLC) service in our privacy protection mechanism. TLC service not only has hiding property, but also blinding property [34]. At a given time T the service publishes a public key so that anyone can use it, even anonymously. Senders encrypt their messages with this public key whose private key is not known to anyone, until a predefined and specific future time $T + N$, at which point the private key is constructed and published. At or after that time, anyone can decrypt the ciphertext using this private key. It will prevent workers from revealing committed subtask selections, and prevent platform from discarding received commitments, revealing committed subtask selections. Platform publishes a public key of a non-malleable encryption scheme, and sends the corresponding private key only when each stage ends. ii) In order to guarantee the security, we apply Blinded Nyberg-Rueppel signature scheme [18,35]. Applying this scheme, signers do not need to verify the authenticity of them, and signee can obtain their information from all signers. The specific process is shown as follows. The signer randomly selects $k \in_R Z_q$ and sends $r = g^k (\mathrm{mod} p)$ to signee; The signee randomly chooses $t_1, t_2, t_3 \in_R Z_q$, computes $R = Mr^{t_1} g^{t_2} y^{t_3} (\mathrm{mod} p)$ and $r' = (R + t_3) t_1^{-1} (\mathrm{mod} q)$. Then, he sends r' to the signer; The signer computes $s = r'x + k (\mathrm{mod} q)$ and sends s to the signee; The signee computes $S = st_1 + t_2 \pi (\mathrm{mod} q)$, and the pair (R, S) is the signature for M. Finally, others check whether $M = g^{-S} y^R R (\mathrm{mod} p)$ to verify the correctness.

1. *Uploading bidding*
   We assume that worker i is selected by the platform to participate in a task, and worker i is interested in this task. First of all, we define *TID* as the task identifier. We use *TPK* to represent the time-lapse encryption key. In the stage of uploading bidding, worker i generates a pseudonym ps_i randomly. In order to upload his bidding without exposing his information to others, worker i encrypts his bidding, sensing time and pseudonym as $Eb_i = (b_i | t_i | ps_i)_{K_{ppub}}$ by platform's public key K_{ppub}. Then worker i makes a commitment $Cb_i = E_{TPK}(Eb_i | rb_i | TID)$, where rb_i is the bit string generated randomly as the proof of correctness. Finally, worker i signs the commitment and sends his bidding request $Rb_i = (i, Cb_i, sign_i(Cb_i | TID))$ to the platform. Once the platform receives worker i's bidding request Rb_i, it will check this bidding request. If Rb_i passes the check, the platform returns a signed receipt $Pb_i = sign_p[i | Cb_i | TID]$ to worker i. Otherwise, the platform discards Rb_i.
   When receiving Rb_i, the platform computes the auction result of worker i and the random bit string rb_i by applying the platform decryption key. Then the platform determines whether worker i is a winner and the corresponding payment for worker i. If the platform rejects worker i's auction, p_i is set to 0. The platform will encrypt the information as $Ew_i = (p_i | i)_{K_{ipub}}$ by worker i's public key K_{ipub}. A commitment $Cw_i = E_{TPK}(Ew_i | rw_i | TID)$ is made by the platform, where rw_i is also the bit string generated randomly as the proof of correctness. Finally, the platform signs the commitment and sends the decision $Pw_i = (Cw_i, sign_p(Cw_i | TID))$ to worker i. Once worker i receives the decision from the platform, he will return a signed receipt $Ww_i{}' = sign_i[Pw_i | TID]$ to the platform and begins his sensing work.
2. *Uploading sensing data*

In the stage of uploading sensing data, in order to upload his sensing data without exposing his information to others, worker i sends his encrypted sensing data as $Es_i = (i|s_i)_{K_{ppub}}$ by using the platform's public key K_{ppub}, where s_i expresses the sensing data of worker i. Then worker i makes a commitment $Cs_i = E_{TPK}(Es_i|rs_i|TID)$, where rs_i is also the bit string generated randomly as the proof of correctness. Finally, worker i signs the commitment and sends his sensing data $Rs_i = (i, Cs_i, sign_i(Cs_i|TID))$ to the platform. Once the platform receives worker i's sensing data, it returns a signed receipt $Ps_i{}' = sign_p[i|Cs_i|TID]$ to worker i.

3. *Updating reputation*
In the stage of updating reputation, the platform returns the updated reputation to worker i through encrypting the information $Er_i = (r_i{}^{t+1}|i)_{K_{ipub}}$ by worker i's public key K_{ipub}. $r_i{}^{t+1}$ indicates the updated reputation of worker i. The platform makes a commitment $Cr_i = E_{TPK}(Er_i|rr_i|TID)$, where rr_i is also the bit string generated randomly as the proof of correctness. Finally, the platform signs the commitment and sends worker i's reputation $Pr_i = (Cr_i, sign_p(Cr_i|TID))$ to worker i. Once worker i receives the updated reputation from the platform, he will return a signed receipt $Wp_i{}' = sign_i[Pr_i|TID]$ to the platform. The reputation updating algorithm with incentive and punishment will be introduced in Subsection 3.4.

In this privacy protection mechanism, other workers cannot get any information about one's bidding, sensing data and updating reputation. The privacy protection is designed under PKI infrastructure which improves the security and time sensitivity further for private data. Therefore, this privacy protection mechanism is privacy-preserving for workers.

3.2. Improved two-stage auction algorithm (ITA)

The improved two-stage auction algorithm corresponds to Step (6) and Step (7) shown in Fig. 3. In Step (6) and Step (7), we design an improved two-stage auction algorithm to determine the winners in real-time for the platform. We divide the auction stage into two stages. The first stage is the sample collection stage that establishes the base for the next stage. Different from the previous solutions, the first batch of workers also have chance to win the auction in order to solve the unfairness problem. This design can encourage workers to arrive in time. The second stage is the competition stage that adjusts the bidding threshold in each transaction dynamically based on the result from the sample collection stage. The specific process is shown as follows:

1. The platform announces budget B for a specific task and the maximal sensing time T, based on which by combining the historic experience, the platform determines threshold κ for the bidding.
2. The platform determines *marginal budget* B' for stage 1 based on T and B. Let $P' = \sum_{j\in\Gamma} p_j$ be the payment sum in stage 1, where Γ represents the *winner set* in stage 1. If $\frac{b_j}{t_j} \le \kappa$, the platform accepts worker j, otherwise, the platform rejects worker j. The platform repeats this process in stage 1 until $P' > B'$. In addition, Eq. (5) expresses the value of B' based on the *multiple-stage sampling-accepting* process [22] which determines the sample size dynamically:

$$B' = \left\lfloor \frac{B}{2^{\lfloor \ln T \rfloor}} \right\rfloor \tag{5}$$

In addition, we define T' be the limited time in stage 1. Accordingly, the calculation method of the limited time in stage 1, T', is derived by Eq. (6):

$$T' = \left\lfloor \frac{T}{2^{\lfloor \ln T \rfloor}} \right\rfloor \tag{6}$$

3. After stage 1, the auction enters stage 2. We obtain the total benefit of the platform $V(M)$ in stage 1, where M represents the set of winning workers during stage 1. In stage 2, with new arriving workers, we adjust the bidding threshold each time based on the marginal density. For new arriving worker i, v_i is shown by Eq. (7):

$$v_i = V\left(M \bigcup \{i\}\right) - V(M) \tag{7}$$

where v_i represents the marginal utility of platform, as well as the specific contribution value that worker i brings to the platform.
Marginal utility is defined as: the marginal utility of a good or service is the gain from an increase, or loss from a decrease, in the consumption of that good or service. Economists sometimes speak of a law of diminishing marginal utility, meaning that the first unit of consumption of a good or service yields more utility than the second and subsequent units, with a continuing reduction for greater amounts. The marginal decision rule states that a good or service should be consumed at a quantity at which the marginal utility is equal to the marginal cost.
In this paper, we utilize marginal utility to determine whether accept the worker. Therefore, the density of marginal utility for worker i is $\frac{v_i}{p_i}$, which can reflect the increasing density or diminishing density. If $\frac{v_i}{p_i} \ge \frac{V(W)}{P(W)}$, and $b_i \le p_i$, the platform accepts worker i, otherwise, it rejects worker i. Once a new worker arrives, the platform computes its marginal density each time, and compares its marginal density with the previous workers' global result. This process is repeated until $\sum_{i\in W} p_i > B$.

ITA improves traditional two-stage auction by solving the unfairness problem for earlier arriving workers and improve the efficiency of auction, which is summarized in Algorithm 1.

Table 1
An asymmetric game.

	X	Y
X	a, b	c, c
Y	c, c	a, b

3.3. Game theory based analysis for trust in mobile crowdsourcing systems

In order to establish the reputation updating algorithm, we analyze the behavior of workers in mobile crowdsourcing systems firstly. In this subsection, we utilize the asymmetric game to analyze the relationship between workers' sensing reports and the platform.

Game theory is a study of strategic decision making. Specifically, it is the study of mathematical models of conflict and cooperation between intelligent rational decision-makers [36].

Asymmetric games are games where there are not identical strategy sets for both players. For instance, the ultimatum game and similarly the dictator game have different strategies for each player. It is possible, however, for a game to have identical strategies for both players, yet be asymmetric. For example, the game showed in Table 1 is asymmetric despite having identical strategy sets for both players (player X and player Y), where a, b and c express the payoffs.

Algorithm 1 Improved two-stage auction algorithm (ITA)

Input:
n, B, T, κ
Output:
Winners
Stage 1:
$B' = \lfloor \frac{B}{2^{\lfloor \ln T \rfloor}} \rfloor$;
$T' = \lfloor \frac{T}{2^{\lfloor \ln T \rfloor}} \rfloor$;
$j = 1$; $P' = 0$; $V(W) = 0$;
while $j \le n$ and $P' \le B'$ **do**
  $b_j = b_j$; $t_j = t_j$;
  if $\frac{b_j}{t_j} \le \kappa$ **then**
    $\Gamma \leftarrow \Gamma \bigcup \{j\}$;
    $p_j \leftarrow \frac{t_j}{T} \times B$;
    $P' \leftarrow P' + p_j$;
    $V(W) \leftarrow V(W) + v_j$;
  end if
  $j \leftarrow j + 1$;
end while
Stage 2:
$P(W) \leftarrow P'$;
$i = 1$;
while $i \le n$ and $P(W) \le B$ **do**
  $b_i = b_i$; $t_i = t_i$;
  $v_i = V(W \bigcup \{i\}) - V(W)$; $p_i = \frac{t_i}{T} \times B$;
  if $\frac{v_i}{p_i} \ge \frac{V(W)}{P(W)}$ **then**
    $\Gamma \leftarrow \Gamma \bigcup \{i\}$;
    $P(W) \leftarrow P(W) + p_i$;
    $V(W) \leftarrow V(W) + v_i$;
  end if
  $i \leftarrow i + 1$;
end while

The interaction between a worker and a platform in a task, which is defined as a *transaction*, can be modeled as an asymmetric gift-giving game. From the aspect of the platform, it makes the payments for workers' sensing work according to workers' bidding and sensing plans. However, the platform gains different payoffs based on the different results about sensing reports. If a worker provides a truthful sensing report, the platform will gain a high payoff. If a worker is selfish, *i.e.*, the worker provides an distrustful report, the platform will gain a low payoff.

From the aspect of workers, because a worker receives the payment in advance, he can strategically choose his action, *i.e.*, determines the level of effort devoted to this task. The quality of worker's sensing report affects not only his own payoff, but also that of the platform. For simplicity, we define q_i to be the *quality type of worker i's sensing report*, which is chosen from a binary set $Q = \{Trust, Distrust\}$.

Trust indicates the quality of the sensing report is high, *i.e.*, represents the level of confidence about the reliability and correctness of the reported sensing data. *Distrust* indicates the quality of the sensing report is low. In order to quantize the quality of a sensing report, we define q as the specific quality of the sensing report, where $q \in [0, 1]$. Because of the selfishness of workers, they may select to expend less cost and time for the task, so that they can obtain more benefit. In this paper, we define α as the *threshold for quality of sensing report*, so the quality types of worker i's sensing report are determined by Eq. (8). The payoff matrix of one transaction is illustrated in Table 2, which is specified as follows:

Table 2
Payoff matrix of one transaction.

	Trust	*Distrust*
Select	$v_i - p_i,\ p_i - c_i'$	$-p_i,\ p_i$
No select	/, 0	/, 0

$$q_i = \begin{cases} Trust, & if\ q \ge \alpha \\ Distrust, & otherwise \end{cases} \tag{8}$$

In the condition that worker i is selected by the platform, if $q_i = Trust$, worker i spends c_i' for solving this task, and he can obtain payment p_i from the platform, thus the payoff of worker i is $p_i - c_i'$. This part of task is solved by worker i and the platform receives a benefit of v_i, thus the payoff of the platform is $v_i - p_i$. If $q_i = Distrust$, worker i free-rides through taking the payment and consuming a low cost, which is approximated by 0 here, and the platform receives no benefit. Thus, in this condition, the payoff of worker i is p_i, and the payoff of the platform is -p_i. This part of task is not solved and remains open for future workers.

In order to find evolutionary stable strategy, we give the definition of Evolutionary Game Theory (EGT). EGT is the application of game theory to evolving populations of lifeforms in biology. EGT differs from classical game theory by focusing more on the dynamics of strategy change as influenced not solely by the quality of the various competing strategies, but by the effect of the frequency with which those various competing strategies are found in the population [37,38]. In EGT, *Evolutionary Stable Strategy (ESS)* is a strategy which, if adopted by a population in a given environment, cannot be invaded by any alternative strategy that is initially rare. That is to say, once the ESS is fixed, alternative strategies will be prevented by natural selection from invading successfully. An ESS is an equilibrium refinement of the Nash equilibrium [39].

Therefore, we need to find a stable strategy based on EGT, *i.e.*, ESS in EGT under this dynamic network environment [40]. In mobile crowdsourcing system, the strategies will change dynamically. With the evolution of mobile crowdsourcing system, there exists an ESS. Therefore, how to find the ESS is an important research content when we establish an effective incentive mechanism. Based on EGT, we explore the ESS in this paper.

Now we consider ESS, *i.e.*, to find the Nash equilibrium strategy [41]. ESS is the functional balance with genuine stability and predictive capability in EGT. When we analyze the stability of the whole mobile crowdsourcing system based on EGT, we must first identify some parameters. In this paper, we first assume the total number of the workers is fixed, x is defined as the proportion of the truthful workers in a mobile crowdsourcing system, and y is defined as the proportion of the distrustful workers in a mobile crowdsourcing system, $x + y = 1$. In addition, U_1 is defined as the expected payoff who adopts the *Trust* strategy, U_2 is defined as the expected payoff who adopts the *Distrust* strategy, and U is defined as the average excepted payoff of all the workers. According to Table 1, we can get $U_1 = x \cdot (p - c')$, $U_2 = y \cdot p$ and $U = x \cdot U_1 + y \cdot U_2$, where p and c' are fixed payment and real cost in a mobile crowdsourcing system.

Therefore, based on the EGT, we can get the evolution models according to *Trust* strategy and *Distrust* strategy respectively, which are shown by Eqs. (9) and (10):

$$\frac{dx}{dt} = \frac{\frac{x \cdot U_1}{U} - x}{\Delta t} \tag{9}$$

$$\frac{dy}{dt} = \frac{\frac{y \cdot U_2}{U} - y}{\Delta t} \tag{10}$$

where t represents evolution generation, which means the times of transactions in this paper, and Δt means the updated time step, which is set as 1 in this paper. Eqs. (9) and (10) represent the evolutionary game model. They can reflect the dynamic changing of different types of individuals in a system, as well as deduce the final convergence status of this system. Therefore, we can utilize Eqs. (9) and (10), *i.e.*, evolutionary game model, to discuss the dynamic changing of different types of individuals and find the ESS.

In order to get ESS, *i.e.*, find the Nash equilibrium point, it should satisfy $F(x) = \frac{dx}{dt} = 0$ and $F(x') < 0$. Through computing $F(x) = \frac{dx}{dt} = 0$ and $F(x') < 0$, we can obtain that $x = 0$ is the Nash equilibrium point, *i.e., Distrust* strategy becomes the ESS.

Therefore, from the aspect of workers, *Distrust* strategy is a Nash equilibrium strategy, which results in the free-riding phenomenon. In the zero arrival-departure model, the payment is made before the task is done, and a worker always has the incentive to take the payment and devote no efforts to accomplish the task, which is known as free-riding [42]. The definition of free-riding is that the inherent feature of an entity is to maximize its utility while minimizing the utilities of other entities, which leads to under-provision of goods or services, or when it leads to overuse or degradation of a common property resource. This feature makes selfish behavior to dominate the evolution direction of the whole system, which incurs the free-riding problem. It seriously influences the overall balance and reduces the efficiency of a system [43–45].

3.4. Truthful online reputation updating algorithm (TORU)

In order to solve free-riding problem in mobile crowdsourcing systems, we propose a truthful online reputation updating algorithm (TORU) to control the gaming process by effective means to make *Trust* strategy to be the preferred strategy for workers. Therefore, we can ensure a system's overall income to be optimal. In this paper, an effective incentive strategy when updating workers' reputations is established. We design an updating method for a worker's reputation according to the quality of its sensing report in one transaction. We define r_i^t to be the reputation of worker i before sensing this task, and r_i^{t+1} to be the reputation of worker i after completing this transaction. Assume $R = \{0, 1, 2, ..., r_{max}\}$ is a reputation set, where r_{max} represents the maximal reputation. A high reputation relates to a worker's good social status, which reflects his good behavior on completing tasks in the past. The reputation of each worker is maintained by the platform. It is updated depending on the report of the requester about the outcome of the transaction.

There is a threshold θ for a worker's reputation. If $r_i^t \geq \theta$, the platform assigns this task to worker i. If $r_i^t < \theta$, the platform does not assign this task to worker i. Therefore, the update for the reputation of worker i after this transaction is derived by Eq. (11):

$$r_i^{t+1} = \begin{cases} \min(r_i^t + 1, r_{max}), & if\ q_i = Trust\ and\ r_i^t \geq \theta \\ \theta - 1, & if\ q_i = Distrust\ and\ r_i^t > \theta \\ 0, & if\ q_i = Distrust\ and\ r_i^t = \theta \\ r_i^t, & if\ r_i^t < \theta \end{cases} \tag{11}$$

In this reputation updating process, if the quality of its sensing report is high, *i.e.*, the sensing result is trustable, and $r_i^t \geq \theta$, the updated reputation adds 1 in the case of $r_i^t + 1 \leq r_{max}$, otherwise, the updated reputation should be r_{max}. When the quality of its sensing report is low, *i.e.*, the sensing result is distrustful, we have two cases: if $r_i^t > \theta$, the reputation of worker i is punished by the platform, which means $r_i^{t+1} = \theta - 1$; if $r_i^t = \theta$, the reputation of worker i is also punished by the platform, which means $r_i^{t+1} = 0$. Otherwise, if $r_i^t < \theta$, the platform does not assign this task to worker *i, i.e.*, worker i cannot participate in this task, so the reputation remains r_i^t.

However, the above method has a problem. In the above method, there is only one threshold for a worker's reputation, which is θ. Under this condition, once a worker selects *Distrust* strategy once, he will never be selected by the platform forever, *i.e.*, be isolated all the time, which is unfair for the potential trustable workers. To solve this problem, we improve the above method to establish a more fair incentive mechanism.

We define a set $\Theta = \{\theta_0, \theta_1, \theta_2, ..., \theta_m\}$ as the reputation threshold set for different tasks, where θ_0 represents the smallest reputation threshold in the system, *i.e.*, $\theta_1, \theta_2, ..., \theta_m \geq \theta_0$. There exists a mapping $\sigma : \Theta - \theta_0 \to \Phi$, which means that each task has a corresponding reputation threshold, *i.e.*, $\theta_1 \to \varphi_1, \theta_2 \to \varphi_2, ..., \theta_m \to \varphi_m$. Therefore, Eq. (11) is improved by Eq. (12):

$$r_i^{t+1} = \begin{cases} \min(r_i^t + 1, r_{max}), & if\ q_i = Trust\ and\ r_i^t \geq \theta_k \\ \theta_k - 1, & if\ q_i = Distrust,\ r_i^t > \theta_k\ and\ \theta_k > \theta_0 \\ \theta_0, & if\ q_i = Distrust,\ r_i^t > \theta_k\ and\ \theta_k = \theta_0 \\ \theta_0, & if\ q_i = Distrust,\ r_i^t = \theta_k\ and\ \theta_k > \theta_0 \\ \theta_0 - 1, & if\ q_i = Distrust,\ r_i^t = \theta_k\ and\ \theta_k = \theta_0 \\ r_i^t, & if\ r_i^t < \theta_k \end{cases} \tag{12}$$

In the improved reputation updating process, we define a reputation threshold set to solve the problem such that only on a reputation threshold will incur the unfairness problem for the potential truthful workers. According to the task φ_k, the corresponding reputation threshold should be θ_k. In the case of $q_i = Distrust$, $r_i^t > \theta_k$ and $\theta_k = \theta_0$, and the case of $q_i = Distrust$, $r_i^t = \theta_k$ and $\theta_k > \theta_0$, we set $r_i^{t+1} = \theta_0$. This design not only punishes the distrustful workers, but also gives a chance to them in order to incentive them to select good behavior in future. In this case, if the workers behave credibly next time, their reputations will be increased, otherwise, they will be rejected by the platform in future. Therefore, in order to obtain the maximal benefit in a long term, workers must select truthful behavior to be their preferred strategy, which makes *Trust* strategy become the ESS.

In addition, the repetition attack may occur in process of crowdsourcing. *Repetition attack* is defined that an attacker needn't decrypt a packet, he can simply re-send sensing data just as is at a later time. This may pollute the data sent by the original worker and cause a decrease in its reputation. According to this problem, we utilize time sensitivity to identify repetition attackers. In the stage of sensing data processing for worker i, platform will compare the data with previous data. If there exists identical data in previous sensing data, platform will set $q_i = Distrust$. The design can recognize repetition attackers in crowdsourcing systems, and punish them on reputations.

In order to solve the free-riding problem about the truthfulness of the worker's sensing report, the proposed reputation updating algorithm inspires the workers to select *Trust* strategy. TORU is shown in Algorithm 2.

3.5. Properties of the proposed incentive mechanism

Lemma 1. *The proposed incentive mechanism is computationally efficient.*

Proof. Let the number of the workers be at most n. So in ITA, the while-loop is of $O(n)$ time complexity at most. In GTRU, the time complexity is also $O(n)$. Thus, the time complexity of the proposed incentive mechanism is $O(n)$, *i.e.*, the proposed incentive mechanism can be computed in polynomial time. □

Lemma 2. *The proposed incentive mechanism is individually rational.*

Proof. If worker i is a winner, he will receive payment $p_i = \frac{t_i}{T} \times B$ paid by the platform. We have set $\frac{B}{T} \geq 1$ in Eq. (2), and cost

Algorithm 2 The game theory based reputation updating algorithm (GTRU)

Input:
r_i^t, θ_k, θ_0, q, α, r_{max}
Output:
r_i^{t+1}
if $q \geq \alpha$ **then**
  $q_i = Trust$;
  if $r_i^t \geq \theta_k$ **then**
    $r_i^{t+1} \leftarrow \min(r_i^t + 1, r_{max})$;
  end if
end if
if $q < \alpha$ **then**
  $q_i = Distrust$;
  if $r_i^t > \theta_k$ **then**
    if $\theta_k > \theta_0$ **then**
      $r_i^{t+1} \leftarrow \theta_k - 1$;
    end if
    if $\theta_k = \theta_0$ **then**
      $r_i^{t+1} \leftarrow \theta_0$;
    end if
  end if
  if $r_i^t = \theta_k$ **then**
    if $\theta_k > \theta_0$ **then**
      $r_i^{t+1} \leftarrow \theta_0$;
    end if
    if $\theta_k = \theta_0$ **then**
      $r_i^{t+1} \leftarrow \theta_0 - 1$;
    end if
  end if
end if

$c_i = \tau \times t_i$, where $0 < \tau < 1$. Thus, the benefit of worker i is $p_i - c_i = (\frac{B}{T} - \tau) \times t_i > 0$. Therefore, this incentive mechanism is individually rational. □

Lemma 3. *The proposed incentive mechanism is profitable.*

Proof. We have $V(W) - P(W) > 0$. This is because that the workers will select *Trust* strategy to be their optimal strategy with this incentive mechanism, and the platform will have nonnegative utility. Thus, at the end of the algorithm, the total utility of the platform is nonnegative. □

Lemma 4. *The proposed incentive mechanism is truthful.*

Proof. In TORU, we establish the incentive and punishment strategy to inspire the workers to select *Trust* strategy to be their optimal strategy. Once they behave unreliably, they will be punished and will not be selected next time. Thus, in order to obtain more utility, the workers should select a truthful strategy. So this incentive mechanism is truthful. □

4. Numerical simulations

We conduct two groups of simulations to evaluate the proposed incentive mechanism. First of all, we verify the efficiency of ITA through comparing it with a general auction algorithm and a two-stage auction algorithm. Then, the effectiveness of TORU is verified.

All the experiments were run on Windows XP operating system with Intel Core (TM) Duo 2.66 GHz CPU, 12 GB memory and Matlab 7.0 simulation platform. They are the event-based simulations for our experiments. Each measurement is averaged over 50 instances.

Table 3
The settings of parameters for ITA.

	First group	*Second group*	*Third group*
n	40	60	80
B	50	100	200
T	25	50	100
κ	2	2	2

4.1. The efficiency of ITA

In order to verify the efficiency of ITA, we simulate three tasks with different budgets and required total sensing times. The budgets of the three tasks are set to be 50, 100 and 200 respectively. Accordingly, the required total sensing times are set to be 25, 50 and 100 respectively. The numbers of worker candidates are 40, 60 and 80 respectively. In these experiments, we set $\kappa = 2$ based on the expertise. In order to specialize the settings of parameters, we set the corresponding parameters in Table 3.

In these simulations, we compare the improved two-stage auction algorithm with a general auction algorithm and a two-stage auction algorithm. The general auction algorithm has a defined threshold. Once the bidding from a worker exceeds this threshold, the platform will reject this worker, otherwise, it will accept this worker. We select the auction algorithm in [46] as the general auction algorithm to compare in the comparison experiments. The two-stage auction algorithm, in [21], rejects the first batch of workers which is used as the sample. In order to compare the efficiencies of the algorithms better, we compute the total payments for the workers: $P(W)$ in these simulations. The x-coordinate indicates the transaction time sequence, and y-coordinate shows the values of $P(W)$. Under different budgets, the algorithm has higher efficiency which can reach the budget value faster. Therefore, through the comparison on the values of $P(W)$, we can deduce the efficiencies that the task be completed under different algorithms.

Fig. 4 shows the experimental result when the budget is 50. We can see that with the improved two-stage auction algorithm, the system completes the task at 20 rounds. However, with general auction algorithm, the system completes the task at 30 rounds, and the two-stage auction algorithm needs 35 rounds. From the experimental results, it can be seen that the improved two-stage auction algorithm can complete a task the fastest, and the two-stage auction algorithm performs the worst, which completes a task the slowest. This is because that the traditional two-stage auction algorithm discards some early arriving workers, so that they have to take more time to complete a task.

Fig. 5 shows the experimental result when the budget is 100. From the experimental result, we can see that the improved two-stage auction algorithm also has the best performance, the system completes the task at 25 rounds. Before 27 rounds, general auction algorithm performs better than two-stage auction algorithm. However, after 27 rounds, two-stage auction algorithm has better performance compared with general auction algorithm. The cause is that the traditional two-stage auction algorithm discards some early arriving workers in the initial several rounds, after that, it performs better than general auction algorithm.

Fig. 6 shows the experimental result when the budget is 200. The experimental result also indicates that the improved two-stage auction algorithm can obtain the best result. However, if the required total sensing time is long enough, two-stage auction algorithm performs better then general auction algorithm. In addition, from Fig. 6, it can be seen that the improved two-stage auction algorithm and the traditional two-stage auction algorithm have the similar experimental results under this condition. This is because that the discarded earlier transactions have little influence to the

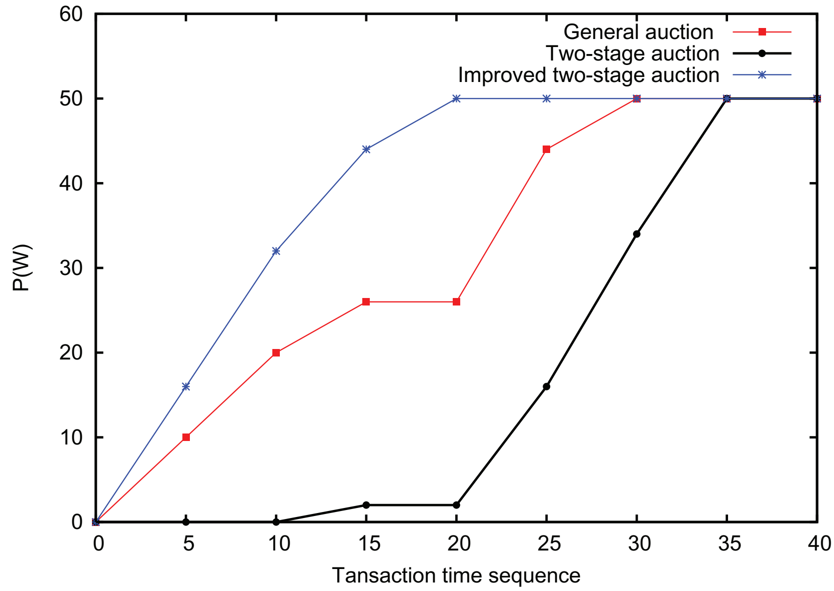


Fig. 4. The comparison of auction efficiencies when the task budget is 50.

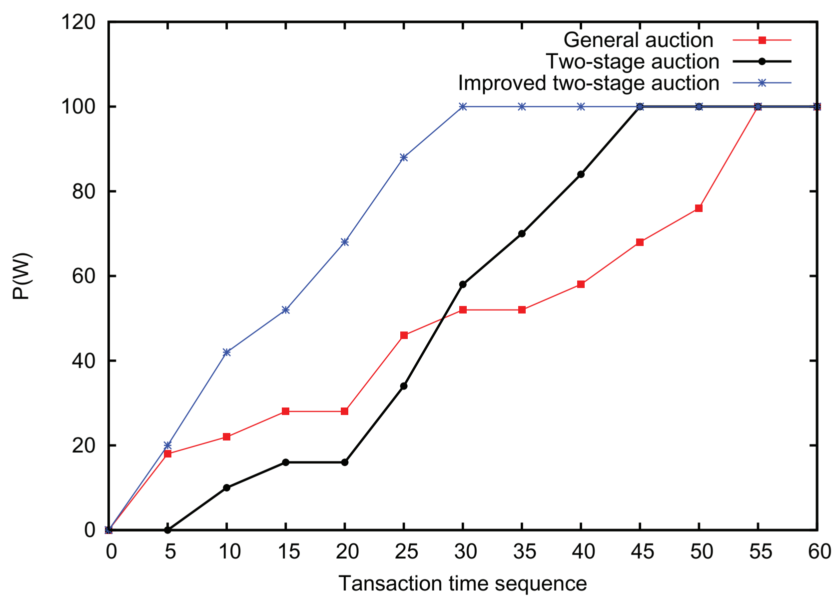


Fig. 5. The comparison of auction efficiencies when the task budget is 100.

experimental result if the budget and required total sensing time are adequate.

In order to compare the efficiencies better, we design an experiment to show the transaction times with different auction algorithms and budgets which is shown by Fig. 7. In Fig. 7, the actual time consumptions of the three algorithms under different conditions are shown, which corresponds to 50, 100 and 200 required total sensing time respectively. We can see that the improved two-stage auction algorithm can always complete a task the fastest compared with other two auction algorithms. The platform will complete a task faster through applying the improved two-stage auction algorithm.

In order to measure the sensibility of budget B, we compare the transaction times under different budgets, 50, 100 and 200 respectively. The comparison result is shown by Fig. 8. From the comparison result, we can see that with the increase of budget, the transaction time increases steadily. Therefore, the proposed algorithm ITA has good stability according to the different budgets B.

From these experiments, we can see that ITA has good convergence and stability. Compared with general auction and two-stage auction, ITA shows better performances and efficiencies under different budgets and sensing times.

4.2. The effectiveness of TORU

We simulate the free-riding phenomenon firstly based on the payoff game matrix. Then, according to the free-riding problem in mobile crowdsourcing systems, we verify the effectiveness of TORU.

According to the payoff matrix shown in Table 2, we set $p = 5$ and $c' = 2$. We select different values of x in order to determine the boundary value of x (the initial proportion of truthful individuals). Thus, we can find boundary value of x that can generate the free-riding phenomenon. We select $x = 0.9$, $x = 0.8$, $x = 0.7$,

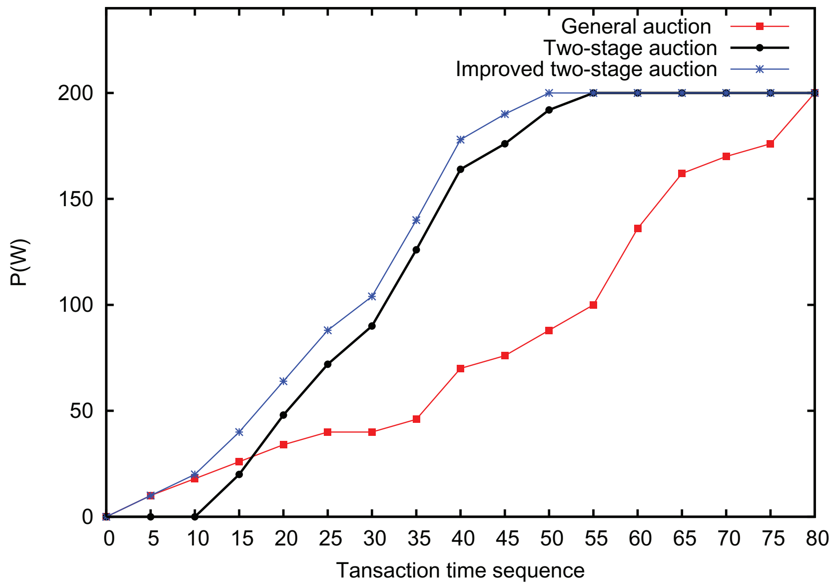


Fig. 6. The comparison of auction efficiencies when the task budget is 200.

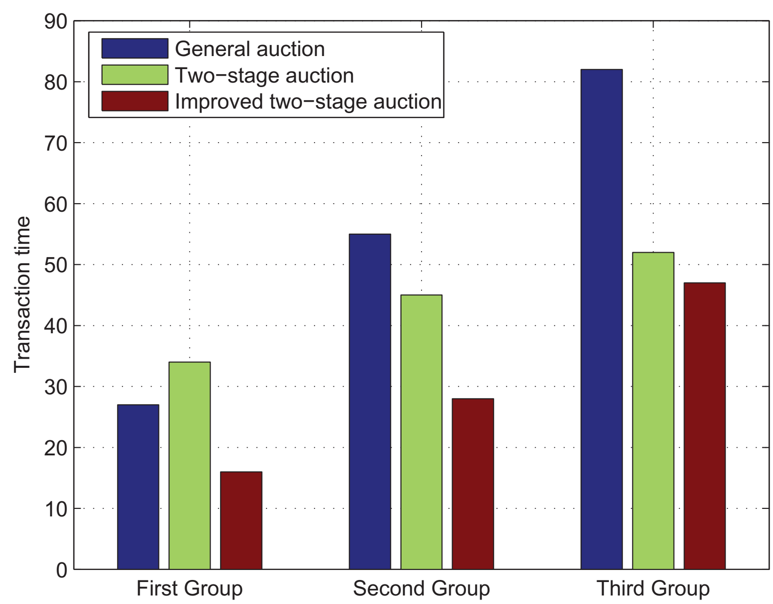


Fig. 7. The comparative results of time consumptions.

Table 4
The settings of parameters for TORU.

θ_0	6
θ_k	9
r_{max}	10
p	5
c'	2

$x = 0.6$ and $x = 0.5$ respectively to find boundary value of x. In order to specialize the settings of parameters, we set the corresponding parameters in Table 4.

Fig. 9 shows the experimental result. It indicates that when x decreases to 0.6, a mobile crowdsourcing system generates the free-riding phenomenon. It means that if there is not an incentive mechanism, the workers will select *Distrust* strategy to be their optimal strategy because of the natural selfishness, resulting in the free-riding problem. Thus, it is very important to establish an effective incentive mechanism to solve the free-riding problem in mobile crowdsourcing systems.

In order to verify the effectiveness of TORU, we compare TORU with the traditional social norm based reputation updating algorithm in [12]. In this simulation, we set the number of initial trust individuals to be 100. The comparison results are shown in Fig. 10. Because the total number of workers is fixed and there is no other workers enter the system, there will be distrustful workers that be obsolete by platform during the evolution process. This will lead to the decrease of the total number of workers in mobile crowdsourcing system. Therefore, the truthful individuals should decrease in the whole crowdsourcing system because of the decrease of the total number of workers.

From Fig. 10, we can see that TORU performs better than the traditional social norm based reputation updating algorithm. Compared with the traditional social norm based reputation updating algorithm, TORU can inspire workers to select a truthful strategy.

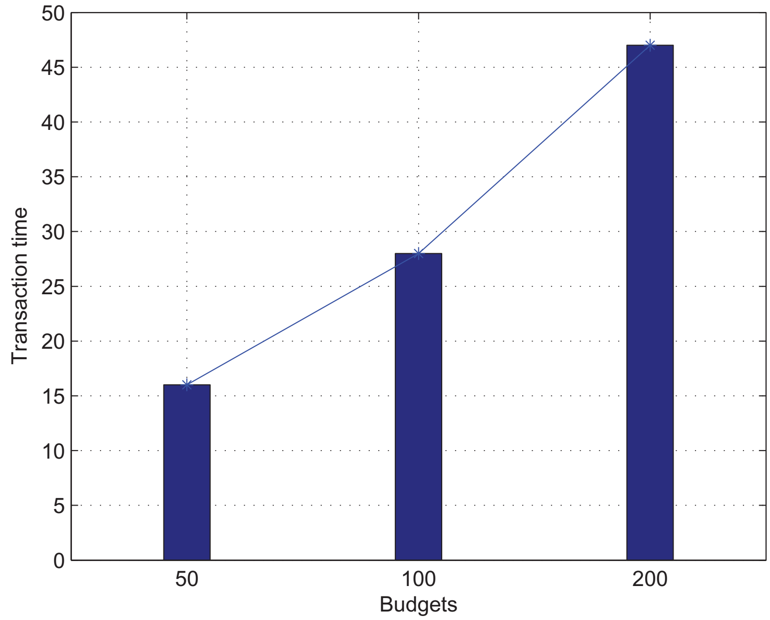


Fig. 8. The comparative results of the transaction times under different budgets.

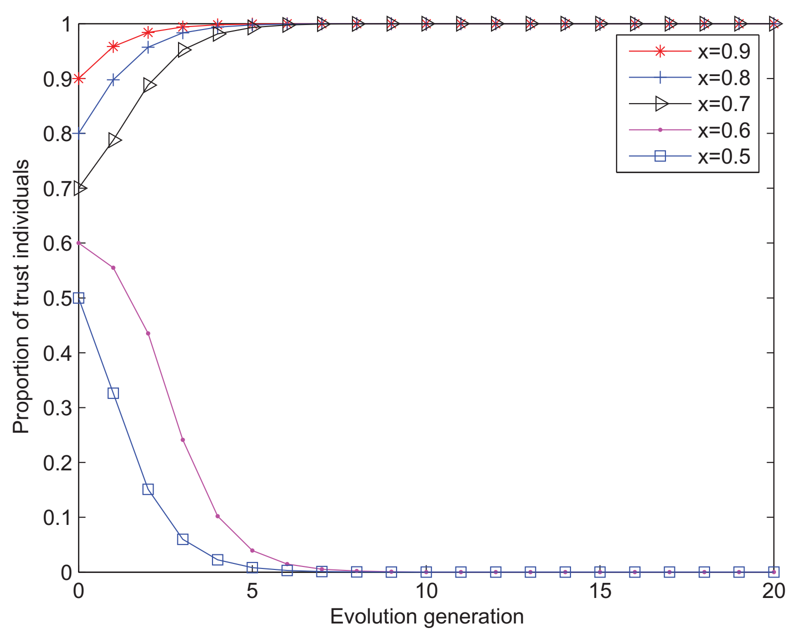


Fig. 9. The illumination of the free-riding problem for different *x*'s.

Although the traditional social norm based reputation updating algorithm can punish distrustful behavior well, once a worker behaves unreliably, he will be discarded by the system, and will not have any opportunity to participate in a task. However, TORU balances the incentive and punishment to inspire workers and punish workers. It gives an opportunity to the worker, who behaves unreliably in one transaction, to inspire him to select *Trust* strategy in the future transactions.

From the experimental result, we can see that TORU has a good convergence. The number of trustful workers can get to stable status after 10 transactions. In this experiment, the confidence level is set to be 0.95. According to the number of trustful individuals, the confidence interval of TORU is [77, 90], and the confidence interval of traditional social norm is [15, 81]. Therefore, compared with the traditional social norm, TORU has a better performance and efficiency.

In order to evaluate the influences of different system's reputation thresholds θ_0 on the effectiveness of TORU, we compare different thresholds θ_0 to evaluate the effectiveness of TORU. In this simulation, the reputation threshold θ_k of task φ_k is set to be 9. We obtain the different results when $\theta_0 = 6$, $\theta_0 = 7$, $\theta_0 = 8$, and $\theta_0 = 9$, which are shown in Fig. 11. From Fig. 11, we can see that when $\theta_0 = 6$, $\theta_0 = 7$, and $\theta_0 = 8$, TORU can inspire workers to select a truthful strategy, and the system tends to be trustful and stable. However, when $\theta_0 = 9$, the system tends to be distrustful, this is because that in this simulation, we set $\theta_k = 9$, so that $\theta_0 = \theta_k$. In

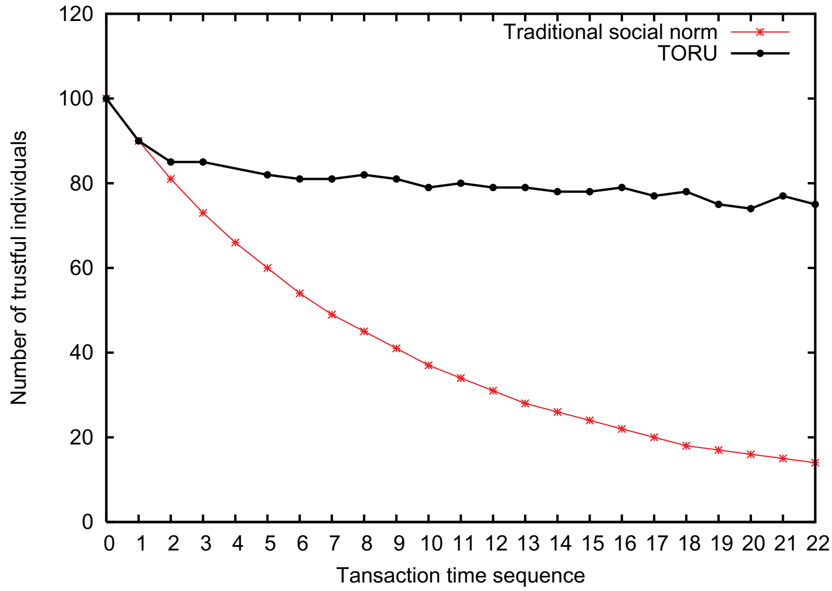


Fig. 10. The effectiveness of TORU.

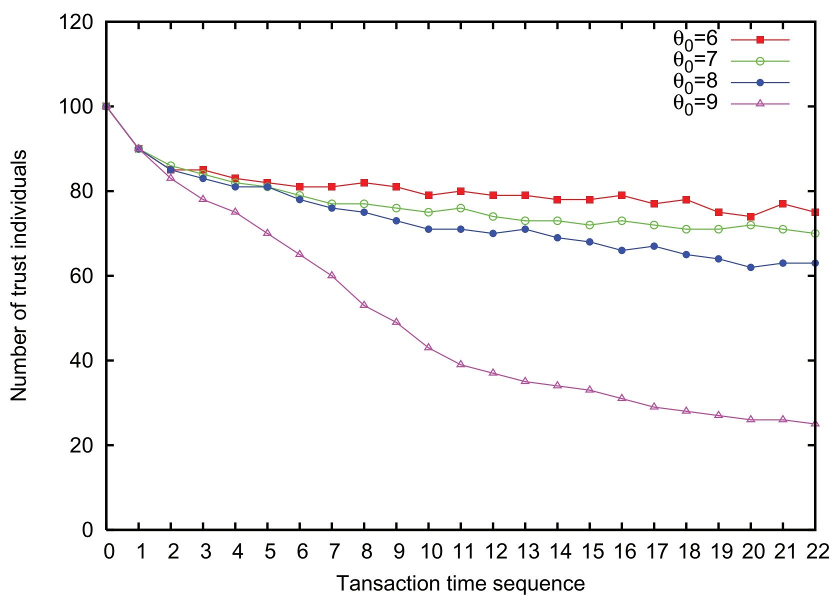


Fig. 11. The influences of different system's thresholds θ_0 on the effectiveness of TORU.

this condition, TORU is transformed to the traditional social norm based reputation updating algorithm, so the system tends to be distrustful in the end.

5. Conclusion

The market of smartphones has proliferated rapidly in the recent years and continues to expand. In order to improve the efficiency and utility of mobile crowdsourcing systems, this paper proposes an incentive mechanism with privacy protection in mobile crowdsourcing systems. Combining the advantages of offline incentive mechanisms and online incentive mechanisms, this paper proposes an incentive mechanism that selects the worker candidates statically, and then dynamically selects winners after bidding. An improved two-stage auction algorithm is proposed in order to determine the winners in real-time and overcome the unfairness problem. According to the free-riding problem, this paper proposes a truthful online reputation updating algorithm (TORU) to update workers' reputations effectively. Through simulations, we verify the efficiency and effectiveness of the proposed incentive mechanism, which can solve the free-riding problem and improve the efficiency and utility of mobile crowdsourcing systems effectively.

As future works, we will focus on the malicious fluctuation behavior of workers, which indicate that workers accumulate reputations in some transactions then behave unreliably in later transactions. We will further investigate how to solve the malicious fluctuation problem.

Acknowledgments

This work is supported by the National Natural Science Foundation of China under Grants No. 61502410, No. 61572418, No. 61403329, No. 61272186 and No. 61472095, the Natural Science Foundation of Shandong Province under Grants No. ZR2013FQ020, ZR2014FQ026 and ZR2013FQ023.

References

[1] G. Chatzimilioudis, A. Konstantinidis, C. Laoudias, D. Zeinalipour-Yazti, Crowdsourcing with smartphones, IEEE Internet Comput. 16 (2012) 36–44, doi:10.1109/MIC.2012.70.

[2] J. Li, Z. Cai, M. Yan, Y. Li, Using crowdsourced data in location-based social networks to explore influence maximization, San Francisco, CA, USA, in: The 35rd Annual IEEE International Conference on Computer Communications, 2016.

[3] L. Zhang, Z. Cai, J. Lu, X. Wang, Mobility-aware routing in delay tolerant networks, Pers. Ubiquit. Comput. 19 (2015) 1111–1123, doi:10.1007/s00779-015-0880-x.

[4] X. Wang, Y. Lin, S. Zhang, Z. Cai, A social activity and physical contact based routing algorithm in mobile opportunistic networks for emergency response of sudden disasters, Enterp. Inf. Syst. 3 (2015) 1–30, doi:10.1080/17517575.2015.1067840.

[5] I. Koutsopoulos, Optimal incentive-driven design of participatory sensing systems, in: The 32nd IEEE Conference on Computing Communications. Turin, Italy, 2013, pp. 1402–1410, doi:10.1109/INFCOM.2013.6566934.

[6] T. Das, P. Mohan, V. Padmanabhan, R. Ramjee, A. Sharma, Prism: platform for remote sensing using smartphones, in: The 8th Annual International Conference on Mobile Systems, Applications and Services, San Francisco, CA, USA, 2010, pp. 63–76, doi:10.1145/1814433.1814442.

[7] P.A. Frangoudis, G.C. Polyzos, Reputation-based crowdsourced wi-fi topology discovery, Comput. Netw. 79 (2015) 1–16, doi:10.1016/j.comnet.2014.12.012.

[8] A. Kittur, E.H. Chi, B. Suh, Crowdsourcing user studies with mechanical turk, in: The 26th Annual SIGCHI Conference on Human Factors in Computing Systems, Florence, Italy, 2008, pp. 453–456, doi:10.1145/1357054.1357127.

[9] Sensorly, Sensorly http://www.sensorly.com.

[10] A. Thiagarajan, L. Ravindranath, K. LaCurts, S. Madden, H. Balakrishnan, S. Toledo, J. Eriksson, Vtrack: accurate, energy-aware road traffic delay estimation using mobile phones, in: The 7th ACM Conference on Embedded Networked Sensor Systems, Berkeley, CA, USA, 2009, pp. 85–98, doi:10.1145/1644038.1644048.

[11] C. Ho, S. Jabbari, J. Vaughan, Adaptive task assignment for crowdsourced classification, in: The 30th International Conference on Machine Learning, Atlanta, GA, USA, 2013, pp. 534–542.

[12] Y. Zhang, M.V.D. Schaar, Reputation-based incentive protocols in crowdsourcing applications, in: The 31st IEEE Conference on Computer Communications, Orlando, FL, USA, 2012, pp. 2140–2148, doi:10.1109/INFCOM.2012.6195597.

[13] D. Yang, G. Xue, X. Fang, J. Tang, Crowdsourcing to smartphones: incentive mechanism design for mobile phone sensing, in: The 18th Annual International Conference on Mobile Computing and Networking, Istanbul, Turkey, 2013, pp. 173–184, doi:10.1145/2348543.2348567.

[14] X. Zhang, Z. Yang, Z. Zhou, H. Cai, L. Chen, X. Li, Free market of crowdsourcing: incentive mechanism design for mobile sensing, IEEE Trans. Parall. Distr. Syst. 25 (2014) 3190–3200, doi:10.1109/TPDS.2013.2297112.

[15] N. Chen, N. Gravin, P. Lu, On the approximability of budget feasible mechanisms, in: The 22nd Annual ACM-SIAM Symposium on Discrete Algorithms, San Francisco, CA, USA, 2011, pp. 685–699.

[16] Y. Wang, G. Yin, Z. Cai, Y. Dong, H. Dong, A trust-based probabilistic recommendation model for social networks, J. Netw. Comput. Appl. 55 (2015) 59–67, doi:10.1016/j.jnca.2015.04.007.

[17] T. Liu, Y. Zhu, Social welfare maximization in participatory smartphone, Comput. Netw. 73 (2014) 195–209, doi:10.1016/j.comnet.2014.08.014.

[18] J. Sun, General privacy-preserving verifiable incentive mechanism for crowdsourcing markets, Comput. Scie. Game Theory (2013) 1–8.

[19] L. Duan, T. Kubo, K. Sugiyama, J. Huang, T. Hasegawa, J. Walrand, Incentive mechanisms for smartphone collaboration in data acquisition and distributed computing, in: The 31st IEEE Conference on Computer Communications, Orlando, FL, USA, 2012, pp. 1701–1709, doi:10.1109/INFCOM.2012.6195541.

[20] L. Jaimes, I. Vergara-Laurens, M. Labrador, A location-based incentive mechanism for participatory sensing systems with budget constraints, in: The 10th IEEE International Conference on Pervasive Computing and Communications, Lugano, Switzerland, 2012, pp. 103–108, doi:10.1109/PerCom.2012.6199855.

[21] M. Bateni, M. Hajiaghayi, M. Zadimoghaddam, Submodular secretary problem and extensions, ACM Tran. Algorithms 9 (2013) 39–52, doi:10.1145/2500121.

[22] D. Zhao, X. Li, H. Ma, How to crowdsource tasks truthfully without sacrificing utility: online incentive mechanims with budget constraint, in: The 33rd IEEE Conference on Computer Communications, Toronto, ON, Canada, 2014, pp. 1213–1221, doi:10.1109/INFOCOM.2014.6848053.

[23] S. Sodagari, S. Bilen, On cost-sharing mechanisms in cognitive radio networks, Eur. Trans. Telecommun. 22 (2011) 515–521, doi:10.1002/ett.1501.

[24] Y. Singer, M. Mittal, Pricing mechanisms for crowdsourcing markets, in: The 22nd International Conference on World Wide Web, Rio de Janeiro, Brazil, 2013, pp. 1157–1166.

[25] W. Mason, D. Watts, Financial incentives and the performance of crowds, ACM SIGKDD Explor. Newslett. 11 (2009) 100–108, doi:10.1145/1809400.1809422.

[26] D. Dipalantino, M. Vojnocic, Crowdsourcing and all-pay auctions, in: The 10th ACM conference on Electronic commerce, Stanford, CA, USA, 2009, pp. 119–128, doi:10.1145/1566374.1566392.

[27] S. Jain, Y. Chen, D. Parkes, Designing incentives for online question and answer forum, in: The 10th ACM Conference on Electronic Commerce, Stanford, CA, USA, 2009, pp. 129–138, doi:10.1145/1566374.1566393.

[28] E.D. Cristofaro, C. Soriente, Pepsi: privacy-enhanced participatory sensing infrastructure, in: The 4th ACM Conference on Wireless Network Security, Hamberg, Germany, 2011, pp. 23–28, doi:10.1145/1998412.1998418.

[29] E.D. Cristofaro, C. Soriente, Extended capabilities for a privacy-enhanced participatory sensing infrastructure (pepsi), IEEE Trans. Inf. Foren. Secur. 8 (2013a) 2021–2033, doi:10.1109/TIFS.2013.2287092.

[30] E. Cristofaro, C. Soriente, Participatory privacy: enabling privacy in participatory sensing, IEEE Netw. 27 (2013b) 32–36, doi:10.1109/MNET.2013.6423189.

[31] Q. Li, G. Gao, Providing privacy-aware incentives for mobile sensing, in: The 11th IEEE International Conference on Pervasive Computing and Communications, San Diego, CA, USA, 2013, pp. 76–84, doi:10.1109/PerCom.2013.6526717.

[32] Y. Wang, Y. Huang, C. Louis, Respecting user privacy in mobile crowdsourcing, Science 2 (2013) 50–64.

[33] J. Pratt, Risk aversion in the small and in the large, Econometrica: J. Econo. Soc. 32 (2012) 122–136, doi:10.1007/978-94-015-7957-5_3.

[34] M.O. Rabin, C. Thorpe, Time-laspe cryptography, ResearchGate (2007) 1–16.

[35] Z. Huang, Y. Wang, K. Chen, Generalization and improvement of nyberg-rueppel message recovery blind signatures, J. China Ins. C. 26 (2005) 131–135.1000-436X(2005)12-0131-05

[36] J. Luo, M. Fan, Research on trust model based on game theory in mobile ad-hoc networks, J. Comput. Res. Dev. 45 (2008) 1704–1710.

[37] W. Li, W. Tian, H. Huang, Evolutionary games theory and agent self-organizing dynamics, J. Comput. Res. Dev. 43 (2005) 46–50.

[38] W. Liu, X. Wang, Study on evolutionary games based on pso-neural networks, Syst. Eng. Electron. 29 (2007) 1282–1284.

[39] P. Taylor, L. Jonker, Evolutionarily stable strategies and game dynamics, Math. Biosci. 40 (1978) 145–156, doi:10.1016/0025-5564(78)90077-9.

[40] Z. He, Z. Cai, X. Wang, Modeling propagation dynamics and developing optimized countermeasures for rumor spreading in online social networks, in: The 35rd Annual IEEE International Conference on Computer Communications, Columbus, OH, USA, 2015, pp. 205–214, doi:10.1109/ICDCS.2015.29.

[41] M. Poulakis, S. Vassaki, A. Panagooulos, P. Constantinou, QoS-driven spectrum sharing game in cognitive radio systems: a variational inequality approach, Trans. Emerg. Telecommun. Technol. (2014), doi:10.1002/ett.2823.

[42] X. Zhang, G. Xue, R. Yu, D. Yang, J. Tang, You better be honest: discouraging free-riding and false-reporting in mobile crowdsourcing, in: Global Communications Conference (GLOBECOM), Austin, TX, 2014, pp. 4971–4976, doi:10.1109/GLOCOM.2014.7037593.

[43] G. Yin, Y. Wang, Y. Dong, H. Dong, Wrightfisher multi-strategy trust evolution model with white noise for internetware, Expert Syst. Appl. 40 (2013) 7367–7380, doi:10.1016/j.eswa.2013.07.026.

[44] L. Zhang, X. Wang, J. Lu, M. Ren, Z. Duan, Z. Cai, A novel contact prediction-based routing scheme for DTNS, Trans. Emerg. Telecommun. Technol. (2014a), doi:10.1002/ett.2889.

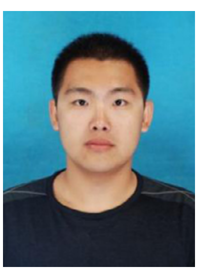
[45] L. Zhang, Z. Cai, J. Lu, X. Wang, Spacial mobility prediction based routing scheme in delay/disruption-tolerant networks, in: International Conference on Identification, Information and Knowledge in the Internet of Things, Beijing, China, 2014b.

[46] C.Z. Zheng, High bids and broke winners, J. Econ. Theory 100 (2001) 129–171, doi:10.1006/jeth.2000.2715.

Yingjie Wang was born in 1986, China. She received the Ph.D. degree in computer science and technology from Harbin Engineering University. She visited Georgia State University from 2013/09 to 2014/09 as a visiting scholar. Dr. Wang is currently a assistant professor in the School of Computer and Control Engineering at Yantai University. Her research interests are trust computing, mobile social networks and Internetware. She has published 12 papers in well known journals and conferences in her research field. In addition, she has presided one National Natural Science Foundation of China projects, and joined three National Natural Science Foundation of China projects and one Natural Science Foundation of Heilongjiang Province project.

Zhipeng Cai received his PhD and M.S. degree in Department of Computing Science at University of Alberta, and B.S. degree from Department of Computer Science and Engineering at Beijing Institute of Technology. Dr. Cai is currently an Assistant Professor in the Department of Computer Science at Georgia State University. Prior to joining GSU, Dr. Cai was a research faculty in the School of Electrical and Computer Engineering at Georgia Institute of Technology. Dr. Cai's research areas focus on networking and big data. Dr. Cai is the recipient of an NSF CAREER Award.

Guisheng Yin was born in 1964, China. He received the Ph.D. degree in automatic control from Harbin Engineering University, where he is a Full Professor and Doctoral tutor, the Dean of College of Computer Science and Technology and the Dean of School of Software Engineering. He ever worked in Tokyo University before he joined the current university. His research interests are trustworthy software, information security, Internetware and so on. He has published more than 70 papers in well known journals and conferences. In addition, he has presided four national projects and five provincial projects.



Yang Gao was born in 1985, China. He received the Master degree in computer science and technology from Northeast Forest University. He is currently a teacher of Yantai University. His research interests are computer science, social networks and wireless sensor networks. He has published some papers in well known journals and international conferences.

Xiangrong Tong was born in 1975, China. He received the Ph.D. degree in computer science and technology from Beijing Jiaotong University. He is a Full Professor of Yantai University. His research interests are computer science, intelligent information processing and social networks. He has published more than 30 papers in well known journals and conferences. In addition, he has presided and joined three national projects and three provincial projects.

Guanying Wu was born in 1979, China. He received Master degree in Peking University Health Science Center. He is an associateáchiefáphysician in China-Japan Friendship Hospital. His research interests are machine learning and medical information system. He has published more than 10 papers in well known journals and international conferences.