An Optimization and Auction-Based Incentive Mechanism to Maximize Social Welfare for Mobile Crowdsourcing

Yingjie Wang, *Member, IEEE*, Zhipeng Cai, *Senior Member, IEEE*, Zhi-Hui Zhan, *Senior Member, IEEE*, Yue-Jiao Gong, *Member, IEEE*, and Xiangrong Tong

***Abstract*— Mobile crowdsourcing is an emerging crowdsourcing paradigm, which generates large-scale sensing tasks and sensing data. One of the major issues in mobile crowdsourcing is how to maximize social welfare through selecting appropriate sensing tasks for crowd workers and selecting appropriate workers for sensing tasks such that it can improve the effectiveness and efficiency of mobile crowdsourcing. This paper proposes an incentive mechanism to maximize social welfare for mobile crowdsourcing and, respectively, investigates worker-centric task selection and platform-centric worker selection. This paper applies an optimization algorithm in task selection for mobile crowdsourcing systems. A discrete particle swarm optimization (DPSO) algorithm for worker-centric task selection is designed to maximize the utilities of workers. In addition, a platform-centric worker selection method, which integrates multiattribute auction and two-stage auction, is proposed to maximize the utility of the platform. The performance of the proposed incentive mechanism is evaluated through experiments. The experimental results show that the proposed incentive mechanism can improve the efficiency and truthfulness of mobile crowdsourcing effectively.**

***Index Terms*— Incentive mechanism, mobile crowdsourcing, social welfare, task selection, worker selection.**

I. Introduction

WITH the rapid development of mobile crowd sensing networks (MCSNs), mobile crowdsourcing (also known as spatiotemporal crowdsourcing) has become a hot research topic in MCSNs. In the traditional crowdsourcing paradigm, such as the Amazon Mechanical Turk and oDesk, the requester first submits a task to the platform and, then, specifies the payment for crowd workers per task, as well as the data quality that workers should provide. Mobile crowdsourcing is an extension of the traditional crowdsourcing systems where the participants perform their sensing tasks through sensing information such as images, sounds, locations, and mobility [1]. Compared with traditional crowdsourcing systems, mobile crowdsourcing has the following advantages: broader coverage, higher scalability, and lower deployment cost [2]. A number of mobile crowdsourcing applications have emerged in the online-to-offline (O2O) field, such as disaster monitoring, traffic management, public security, logistics management, and social media [3], [4]. In mobile crowdsourcing marketplace, crowd workers are paid to perform tasks using their mobile devices. For example, crowd workers are recruited to check product price in supermarkets, or to sense surrounding environment, including traffic information, remaining parking lots, and so on [5], [6]. In MCSNs, how to inspire workers to participate in sensing tasks is an important research content for improving the efficiency of MCSNs. Therefore, the research of incentive mechanism is a research hotspot in MCSNs [7].

A mobile crowdsourcing system consists of three components: requesters, platform, and crowd workers [8]. When requesters issue sensing tasks, the platform will assign the tasks to appropriate workers to maximize the utility of workers. In the next step, the workers who are interested in the assigned task will submit the bidding profile to the platform. Based on the auction algorithms, the appropriate workers are selected by the platform, which can maximize the utility of the platform. The above two steps compose the incentive mechanism of one mobile crowdsourcing, which can maximize social welfare. Therefore, how to select tasks and select workers are very important for the success of a mobile crowdsourcing system [9].

Task selection and worker selection are the core research problems in traditional crowdsourcing systems. In recent years, researchers have proposed many task assignment algorithms to optimize the efficiency of the systems. Generally, the proposed algorithms can be grouped into two categories: 1) task assignment algorithms based on binary image matching model [10], for which the typical applications include

Manuscript received July 9, 2018; revised October 21, 2018 and January 19, 2019; accepted March 18, 2019. Date of publication April 11, 2019; date of current version June 10, 2019. This work was supported in part by the National Natural Science Foundation of China under Grant 61502410, Grant 61822602, Grant 61772207, Grant 61572418, Grant 61602399, Grant 61702439, and Grant 61773331, in part by the China Postdoctoral Science Foundation under Grant 2017M622691, in part by the National Science Foundation (NSF) under Grant 1704287, Grant 1252292, and Grant 1741277, and in part by the Natural Science Foundation of Shandong Province under Grant ZR2014FQ026 and Grant ZR2016FM42.
(Corresponding authors: Zhipeng Cai; Zhi-Hui Zhan.)

Y. Wang and X. Tong are with the School of Computer and Control Engineering, Yantai University, Yantai 264005, China (e-mail: towangyingjie@163.com).

Z. Cai is with the Department of Computer Science, Georgia State University, Atlanta, GA 30303 USA (e-mail: zcai@gsu.edu).

Z.-H. Zhan and Y.-J. Gong are with the School of Computer Science and Engineering, South China University of Technology, Guangzhou 510006, China, and also with the Guangdong Provincial Key Laboratory of Computational Intelligence and Cyberspace Information, South China University of Technology, Guangzhou 510006, China (e-mail: zhanapollo@163.com).
Digital Object Identifier 10.1109/TCSS.2019.2907059

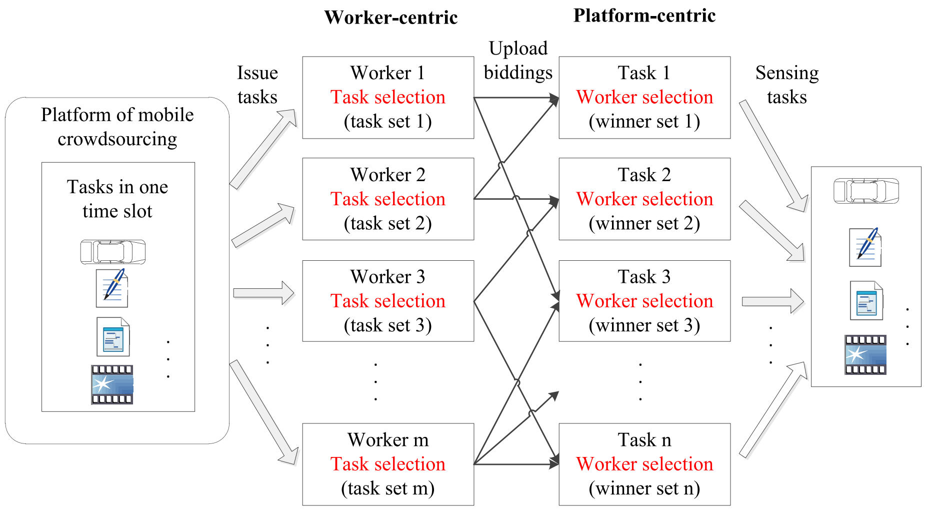

Fig. 1. Process of selecting tasks and selecting workers.

real-time car services, such as DiDi taxi and 2) task assignment algorithms based on planning model [11], where the typical applications include logistics delivery service, such as Baidu take-out. However, the static task assignment algorithms for traditional crowdsourcing are inappropriate for mobile crowdsourcing, since they cannot satisfy the real-time property of mobile crowdsourcing. Thus, dynamic task assignment methods have become the research focus in mobile crowdsourcing. In order to maximize social welfare, selecting appropriate workers to sense tasks was also researched by scholars. Auction algorithms were widely applied in MCSNs when designing the worker selection method. However, how to dynamically select workers in complex spatiotemporal mobile crowdsourcing should be further researched by scholars [12]. In addition, very few works utilized an optimization algorithm to analyze the process of task assignment for mobile crowdsourcing systems.

In this paper, we investigate a new task assignment mechanism for mobile crowdsourcing in order to improve the performance of mobile crowdsourcing, as well as to inspire workers to participate in tasks. The incentive mechanism composed of two phases, i.e., the worker-centric task selection and the platform-centric worker selection.

The aim of the *worker-centric* task selection is to maximize worker's utility while the central objective is to assign suitable tasks to a worker in order to maximize his/her utility in one-time slot [13].

For the *platform-centric* worker selection, the objective is to select suitable workers to participate in a sensing task through an auction in order to maximize the utility of a mobile crowdsourcing system [14].

Requesters, mobile crowdsourcing platform, and workers constitute the mobile crowdsourcing system. The role of the platform includes the information preprocessing module, the incentive module, and the data processing and feedback module. The worker-centric task selection and the platform-centric worker selection submodules in the incentive module play the key roles in our mobile crowdsourcing system. In the process of selecting tasks, first, the system selects the task sets for workers. Then, interested workers upload their biddings to the platform for participating in the task. In the end, the platform selects the most suitable worker for each task [15]. To accomplish the above-mentioned procedures of incentive mechanism, we develop a new incentive mechanism to maximize the social welfare of MCSNs. The main contributions of our paper are summarized as follows.

1) In order to adapt to the online MCSNs, and provide more accurate tasks, we apply an optimization algorithm when selecting tasks for crowd workers in mobile crowdsourcing systems and propose a discrete particle swarm optimization (DPSO)-based algorithm that maximizes the utility of workers [16]. Based on the Gaussian white noise, the DPSO is improved to increase the diversity, which can avoid generating premature convergence.
2) In order to select more appropriate workers to participate in tasks, a novel platform-centric worker selection method by combining multiattribute auction and two-stage auction is proposed. The platform-centric worker selection method takes worker's data quality, trust degree, position similarity, and privacy sensitivity into considerations in the process of auction.
3) The extensive experiments are carried out to evaluate and compare the performance of our incentive mechanism with other methods. The experimental results verify the advantages of our incentive mechanism.

The process of selecting tasks and selecting workers during one-time slot is shown in Fig. 1. As shown in Fig. 1, our proposed incentive mechanism includes two steps, i.e., task selection step and worker selection step. In one-time slot, the tasks are issued by the platform for workers. In the first step (worker-centric task selection), based on the issued tasks and a worker's interests, the platform will recommend tasks for the worker that can benefit the worker. For the selected tasks, the worker will upload biddings in order to participate in the interested tasks. In the second step (platform-centric worker selection), based on the biddings submitted by workers, the platform will select satisfied crowd workers for the task that can benefit the task. The worker-centric task selection and platform-centric worker selection are the two steps of the incentive mechanism for a mobile crowdsourcing system. To assure the bottom-line benefit of workers, each worker announces a reserve price to sell his/her service. The platform then selects the set of winners and pays them the payments that are no lower than the workers' reserve prices. Through the proposed incentive mechanism, social welfare could be maximized.

The rest of this paper is organized as follows. Section II introduces the related works. Section III describes the design of the proposed incentive mechanism, which includes worker-centric task selection and platform-centric worker selection. Section IV gives the comparison experiments, along with analysis and discussion of the experimental results. Finally, Section V draws the conclusion.

II. Related Works

In recent years, many incentive mechanisms were studied for MCSNs. Researchers studied the task assignment methods through considering many factors, such as real-time property [17], data quality [18], multiagent environment [19], social relationship influence [20], trust degree [21], [22], and privacy aware [23]–[25]. In this section, we discuss and analyze the related works for task assignment mechanisms and auction algorithms.

A. Task Assignment Mechanism

The typical task assignment mechanisms include the task assignment based on matching model and the task assignment based on the planning model. In task assignment based on matching model, Kazemi and Shahabi [26] proposed the method of task assignment query based on binary image matching in static offline scenario. In this method, the crowdsourcing tasks and the participants are regarded as two nonintersecting point sets in a binary image. The method can solve the task assignment problem of mobile crowdsourcing in the static offline scenario. However, the authors failed to consider that a different worker may have different sensing qualities for different tasks. Therefore, Kazemi *et al.* [27] further proposed a quality-constraint task assignment model in order to guarantee the sensing quality. Considering the conflicting requirements between different tasks, She *et al.* [28] proposed a conflict-aware task assignment method, which can maximize the utility of a global match. For the application-aware factor, Zheng *et al.* [29] studied the task assignment problem based on the quality of service (QoS), which can significantly improve the efficiency of task assignment. In addition, Cheng *et al.* [30] proposed a reliable diversity-based task assignment method by not only considering worker's reliability but also incorporating worker's spatiotemporal diversity information. The aforementioned related works were proposed based on the static offline scenario. However, for dynamic MCSNs, the above-mentioned task assignment methods cannot adapt to the online scenario. Therefore, Tong *et al.* [6] proposed the mobile microtask assignment method based on the online bilateral weighted binary image matching model. Once a new task appears, the platform will assign the task to the appropriate workers immediately, as well as once a new worker appears, the platform will also assign appropriate tasks for this worker immediately. Considering a dynamic online scenario, She *et al.* [31] further proposed the conflict-aware task assignment method. Different from a static offline scenario, spatiotemporal conflict will affect the result of task assignment in a dynamic scenario. Random-threshold-based algorithm [32] randomly selects tasks that their utilities are not less than the threshold and add them into the task set. However, this algorithm performs unsteadily because of the big difference in utilities when selecting different thresholds.

For the task assignment based on planning model, shortest distance path query and shortest time path query are the two main task assignment methods. Luo *et al.* [33] proposed the most frequent path method to search for the most frequent path and perform shortest time path query for task assignment. Su [34] developed a crowd-based route recommendation system (CrowdPlanner) through applying best path query. In addition, Demiryurek *et al.* [35] utilized route planning method to assign multitask for a worker. Route planning method aims to maximize a worker's utility through planning the route and the order of sensing tasks. The aforementioned planning-based task assignment mechanism is designed for only one worker. For route planning of multiworker, She *et al.* [28] studied the problem of recommending routes for multiple workers. Deng *et al.* [36] also studied the multiworker task assignment problem and proposed two heuristic frameworks to solve the problem of task assignment for multiple workers.

In mobile crowdsourcing, few works applied optimization algorithms in task assignment. However, MCSN is an extension of traditional wireless sensor network (WSN). In WSNs, optimization algorithms have been used to optimize the task assignment. Zhu and Gao [37] proposed a nested optimization based on Genetic Algorithm (GA) for energy-efficient task assignment in multihop clusters. However, GA-based task assignment is not appropriate for large-scale computational problem. The PSO was improved by scholars to applied in WSNs, which has faster optimum speed and can effectively optimize the system parameters [38]. However, PSO can easily run into local optimum, i.e., the premature convergence problem, as well as the Ant Colony algorithm [39], [40]. According to the PSO algorithm, Higashi and Iba [41] studied the PSO algorithm with the ideas of Gaussian mutation, which can obtain a result superior to GA. The individuals were selected with the predetermined probability and their positions were determined with the probability under the Gaussian distribution.

B. Auction Algorithm

The main auction algorithms applied in mobile crowdsourcing systems include reverse auction (RA), combinatorial auction (CA), multiattributive auction (MAA), all-pay auction (AA), double auction (DA), and Vickrey–Clarke–Groves auction (VCG). Lee and Hoh [42] applied RA into the incentive mechanisms of MCSNs, which can guarantee the minimum payment cost and a high participation rate. Based on DA and the distributed property, Duan *et al.* [43] considered both the time factor and the cost factor when designing the auction algorithm, which allows it to adapt to

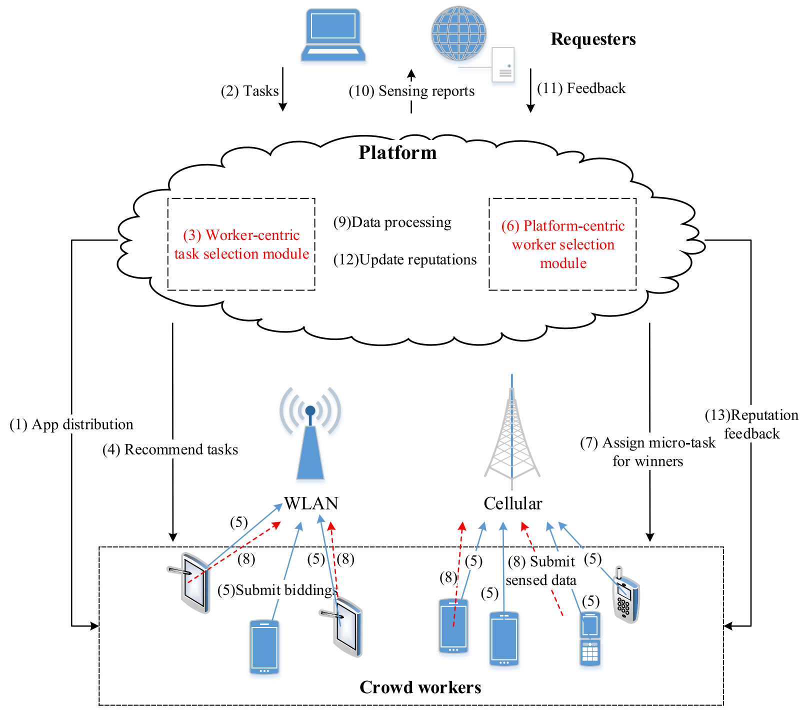

Fig. 2. System model of this mobile crowdsourcing.

the heterogeneity of MCSNs. Feng *et al.* [44] adopted CA to inspire the participants. Participants can bid multiple tasks according to their location and sensing range. The winners will be determined by the platform based on the overall biddings. Krontiris and Albers [45] considered both the participation rate and the data quality based on MAA. In order to increase the bidding price, participants will improve their data quality according to the auction feedback results. Luo *et al.* [46] proposed AA to inspire workers to participate in the sensing tasks. AA specifies that the platform will only pay the reward to the bidder who provides the greatest contribution but not all the bidders. Yang *et al.* [47] adopted DA and k-anonymity privacy protection to inspire users to participate in tasks. In crowd sensing, participants have different location privacy sensitivity levels. Thus, DA can inspire the participants with low location privacy sensitivity to participate with k-anonymity privacy protection for protecting the location information of the participants with high location privacy sensitivity [48]. VCG includes allocation rule and updating rule. Gao *et al.* [49] improved VCG through adding updating rule, which can update allocation rule based on user's trust degree. Our previous work in [50] proposed improved two-stage auction (ITA) based on two-stage auction to complete real-time auction process, which can inspire workers to participate in tasks efficiently in mobile crowdsourcing.

C. Summary

According to the discussions for the aforementioned methods, most of the methods failed to comprehensively consider both worker utility and platform. The multiattribute of workers (includes historic behaviors and social attribute) and the real-time arriving property of individuals were not considered in most of these methods. Therefore, how to design the task selection mechanisms with better adaptation and effectiveness is the research focus of this paper. By considering the worker-centric and platform-centric aspects, we study the DPSO-based task selection and the auction-based worker selection problems. Moreover, few works have applied optimization algorithms in mobile crowdsourcing systems to optimize task selection. Therefore, we propose a DPSO-based task selection method by considering Gaussian white noise to maximize worker utility [41], [51]. In addition, by combining multiattribute auction and the two-stage auction, the proposed auction-based task selection mechanism can maximize the utility of the platform.

III. Proposed Incentive Mechanism

Because of the dynamic environment of MCSNs, static task selection methods cannot adapt well to complex MCSNs. Therefore, online task selection mechanisms need to be studied for mobile crowdsourcing systems [52]. Similar to the work in [50], the proposed task selection mechanism needs to satisfy the following four properties, i.e., *computational efficiency*, *individual rationality*, *profitability*, and *truthfulness*.

A. System Model

In this section, the system model for MCSNs is given first. In our mobile crowdsourcing system, we divide the timeline into multiple time slots. In each time slot, the incentive mechanism is carried out. The system model is shown in Fig. 2. From the system model, it can be seen that the platform dynamically recommends tasks for workers in the worker-centric task selection module. In each time slot, we recommend tasks

TABLE I
DESCRIPTIONS FOR NOTATIONS IN OUR SYSTEM MODEL

Notation	*Description*
w_i	The ith crowd worker.
φ_j	The jth sensing task.
$Win(\varphi_j)$	The winner set of crowd workers for φ_j.
u_{ij}	The utility of w_i through sensing φ_j.
p_{ij}	The payment made by the platform to w_i that performed φ_j.
c_{ij}	The cost to w_i for performing φ_j.
t_{ij}	The sensing time that w_i spent performing φ_j.
T_j	The total sensing time that φ_j requested.
B_j	The budget for φ_j.
$V_j(W)$	The total income of the platform with respect to φ_j.
$P_j(W)$	The total payments to the workers for task φ_j.
v_{ij}	The contribution that w_i brings for the platform through sensing φ_j.

for workers based on the optimization algorithm. From the whole timeline, the tasks are recommended dynamically. In the platform-centric worker selection module, the bidders are selected as time goes on. Therefore, the proposed incentive mechanism is adaptive to the real-time dynamics of mobile crowdsourcing. The details of the algorithms are shown in Sections III-B and III-C. The corresponding descriptions for notations in this section are given by Table I.

Let w_i indicate the ith crowd worker. The parameter φ_j expresses the jth sensing task in the MCSN. If worker w_i is selected to perform task φ_j, the utility of w_i is computed by the following equation:

$$u_{ij} = \begin{cases} p_{ij} - c_{ij}, & \text{if} \quad w_i \in Win(\varphi_j) \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

where $Win(\varphi_j)$ indicates the winner set, i.e., the workers who can perform φ_j. In (1), p_{ij} is the payment made by the platform to w_i that performed φ_j, and c_{ij} is the cost of w_i for performing φ_j, such as electricity cost, communication cost, and so on. Let τ_i indicate the cost of w_i in unit time, thus $c_{ij} = \tau_i \times t_{ij}$, where $0 < \tau_i < 1$. In our model, p_{ij} is calculated using the following equation:

$$p_{ij} = \frac{t_{ij}}{T_j} \times B_j \tag{2}$$

where t_{ij} denotes the sensing time that w_i spent performing φ_j, T_j is the total sensing time that φ_j requested, and B_j represents the budget of φ_j. In our system model, we assume $(B_j/T_j) \geq 1$. The utility of the platform $\bar{u_j}$ with respect to φ_j is given by the following equation:

$$\bar{u_j} = V_j(W) - P_j(W) \tag{3}$$

where $V_j(W) = \sum_{w_i \in Win(\varphi_j)} v_{ij}$ indicates the total income of platform with respect to φ_j, and v_{ij} expresses the contribution that w_i brings for platform through sensing φ_j. $P_j(W) = \sum_{w_i \in Win(\varphi_j)} p_{ij}$ indicates the total payments to the workers who performed φ_j.

In the system model, the step of worker-centric task selection is executed first. The outputs of Algorithm 1 are the task sets of workers during one-time slot. Then, based on the outputs, workers submit the biddings to the platform in the step of platform-centric worker selection (Algorithm 2) for their interested tasks.

Algorithm 1 G-DPSO-Based Worker-Centric Task Selection Algorithm

Input:
 m, B_i^w, c_1, c_2, ω
Output:
 $\widetilde{P}_{gbest}$

```
procedure G-DPSO-based worker-centric task selection
for each particle x_ij do
  Initialize velocity ~Vel_i and position ~X_i for particle x_ij;
  Evaluate particle x_ij and set ~P_pbesti = ~X_i;
end for
~P_gbest = max{~P_pbesti};
while not stop do
  for j = 1 to m do
    Update the velocity and position of particle x_ij, and it
    should satisfy sum_{j=1}^{m} c_ij x_ij <= B_i^w;
    if f(X_i) > f(~P_pbesti) then
      ~P_pbesti = X_i;
    end if
    if f(~P_pbesti) > f(~P_gbest) then
      f(~P_gbest) = f(~P_pbesti);
    end if
  end for
end while
print ~P_gbest;
end procedure
```

B. Worker-Centric Task Selection Method

How to select appropriate tasks for a crowd worker to maximize utility is the target of the worker-centric task selection method [53], [54]. Because of the real-time property, offline methods cannot adapt to the online environment of MCSNs. In addition, DPSO-based algorithm can optimize the dynamic utility better through comparing with other task selection methods. Therefore, we improve the DPSO algorithm to optimize worker utility. We also give the descriptions for notations appear in this section, which are shown in Table II.

Let U_i represent the total utility of w_i in one-time slot, which is obtained by the following equation:

$$U_i = \sum_{j=1}^{m} u_{ij}. \tag{4}$$

Therefore, the objective function is $max\left\{U_i\right\}$ with the constraint condition $C_i \leq B_i^w$, where C_i represents the total cost of w_i, and $C_i = \sum_{j=1}^{m} c_{ij}$. B_i^w is the budget of w_i that includes the electricity, communication energy.

In order to explain the worker-centric task selection clearly, we can treat the process of selecting tasks as a 0-1 knapsack problem. The DPSO algorithm is utilized to solve the problem of worker-centric task selection. In this paper, we divide time into multiple time slots, x_{ij}^k indicates whether φ_j is selected by w_i in the kth iteration. The value of x_{ij}^k is determined by

TABLE II
DESCRIPTIONS FOR NOTATIONS IN THE WORKER-CENTRIC TASK SELECTION MECHANISM

Notation	*Description*
U_i	The total utility of w_i in one time slot.
C_i	The total cost of w_i.
B_i^w	The budget of w_i.
p_{ij}	The payment made by the platform to w_i that performed φ_j.
X_i^k	The set of particles, *i.e.*, the task set assigned to w_i in the kth iteration.
x_{ij}^k	Whether φ_j is assigned to w_i in the kth iteration.
Vel_i^k	The velocity set of w_i in the kth iteration.
v_{ij}^k	The contribution that w_i brings to φ_j in the kth iteration.
vel_{ij}^k	The change of position value of w_i for φ_j in the kth iteration.
P_{pbesti}^{k-1}	The personal best value of X_i^{k-1} in the $(k-1)$th iteration.
P_{gbest}^{k-1}	The global best value of X_i^{k-1} in the $(k-1)$th iteration.
ω	The parameter that can adjust the local and global search capability.
c_1, c_2	The learning factors.
r_1, r_2	The random values.
$\widetilde{Vel}_i^{k-1}$, $\widetilde{P}_{pbesti}^{k-1}$, $\widetilde{X}_i^{k-1}$, $\widetilde{P}_{gbest}^{k-1}$	The corresponding operators after updating the dimensionality.
δ_{ip}^{k-1} and δ_{ig}^{k-1}	The Gaussian disturbed parameters for the personal best value and the global best value.

the following equation, where Φ_i indicates the task set of w_i. In DPSO algorithm, x_{ij}^k represents a particle in its kth iteration, thus X_i^k indicates the positions of particles, i.e., a feasible solution in its kth iteration

$$x_{ij}^k = \begin{cases} 1, & \text{if } \varphi_j \overset{\text{assign}}{\to} w_i \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

Therefore, the objective function is transformed into the following equation:

$$\max\left\{\sum_{j=1}^{m} u_{ij} x_{ij}^k\right\}. \tag{6}$$

The corresponding constraint condition is expressed by the following equation:

$$\sum_{j=1}^{m} c_{ij} x_{ij}^k \le B_i^w. \tag{7}$$

In DPSO algorithm, the position of particle indicates a feasible solution, thus the fitness function of particle is given by the following equation:

$$f(X_i^k) = \sum_{j=1}^{m} v_{ij}^k x_{ij}^k \tag{8}$$

where X_i^k represents the set of particles, i.e., $X_i^k = (x_{i1}^k, x_{i2}^k, \cdots, x_{ij}^k, \cdots, x_{\text{im}}^k)$. The parameter v_{ij}^k means the contribution that w_i brings to φ_j in the kth iteration. Therefore, optimizing $f(X_i^k)$ through applying DPSO algorithm can optimally assign tasks for w_i. A particle flies in the solution space, and adjusts its velocity and position based on its experience and its partners. Therefore, Vel_i represents the velocity set of w_i, and $Vel_i^k = (vel_{i1}^k, vel_{i2}^k, \cdots, vel_{ij}^k, \cdots, vel_{\text{im}}^k)$, where vel_{ij}^k indicates the change of position value for φ_j in the kth iteration. According to the value of vel_{ij}^k, we give the calculation method, which is shown by the following equation:

$$vel_{ij}^k = \begin{cases} 0, & \text{if } x_{ij}^k = x_{ij}^{k-1} \\ x_{ij}^k, & \text{otherwise} \end{cases} \tag{9}$$

where $vel_{ij}^k = 0$ indicates that the position of particle remains unchanged in the kth iteration for φ_j, and $vel_{ij}^k = x_{ij}^k$ means its position will change in the kth iteration. Therefore, one particle will update its velocity and position based on the following equations, where P_{pbesti}^{k-1} denotes the personal best value of X_i^{k-1}, P_{gbest}^{k-1} is the global best value, and ω is the inertia weight

$$\begin{aligned} Vel_i^k &= \omega \cdot Vel_i^{k-1} + c_1 \cdot r_1 \cdot \left(P_{\text{pbesti}}^{k-1} - X_i^{k-1}\right) \\ &\quad + c_2 \cdot r_2 \cdot \left(P_{gbest}^{k-1} - X_i^{k-1}\right) \end{aligned} \tag{10}$$

$$X_i^k = X_i^{k-1} + Vel_i^k. \tag{11}$$

The parameter ω can adjust the local and global search capability, and $0 < \omega < 1$. Parameters c_1 and c_2 are the learning factors (acceleration factors) that express the trust levels for individual cognition and social cognition, respectively. In our task selection mechanism, the values of c_1 and c_2 are determined by a mobile crowdsourcing system that cannot be changed with the change of k. The aforementioned three parameters are constants and need to satisfy the following constraint: $c_1 + c_2 > 0$ and $c_1 + c_2 - 2 \cdot \omega < 2$, i.e., a triangular convergence region that express the relationships among ω, c_1 and c_2. The parameters r_1 and r_2 are two random values, and $r_1, r_2 \in [0, 1]$. Because of the dynamics of MCSNs, new tasks are continually added and the task dimensionality changes dynamically. In this paper, we use $\widetilde{Vel}_i^{k-1}$, $\widetilde{P}_{\text{pbesti}}^{k-1}$, $\widetilde{X}_i^{k-1}$, $\widetilde{P}_{gbest}^{k-1}$ to represent the corresponding operators after updating the dimensionality. We set the initial value of the added dimensionality to be 0, thus each particle updates its

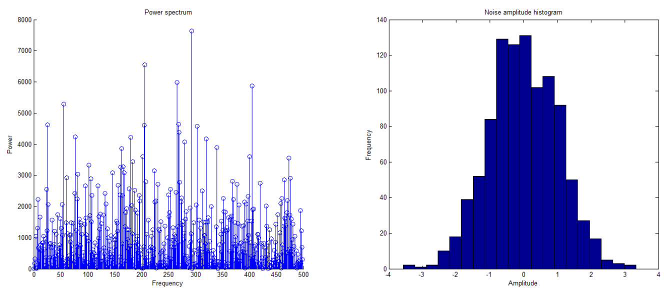

Fig. 3. Distribution of Gaussian white noise.

velocity and position by the following equations:

$$Vel_i^k = \omega \cdot \widetilde{Vel}_i^{k-1} + c_1 \cdot r_1 \cdot \left(\widetilde{P}_{\text{pbesti}}^{k-1} - \widetilde{X}_i^{k-1}\right) + c_2 \cdot r_2 \cdot \left(\widetilde{P}_{gbest}^{k-1} - \widetilde{X}_i^{k-1}\right) \tag{12}$$

$$X_i^k = \widetilde{X}_i^{k-1} + Vel_i^k. \tag{13}$$

Through analyzing the DPSO algorithm, it can be seen that DPSO algorithm can easily run into local optimum, i.e., the premature convergence problem [55], [56]. How to avoid the premature convergence problem is an important research problem. In order to resolve this problem, we utilize Gaussian white noise to disturb the extremum, which can enhance the population diversity and avoid running into local optimum. The distribution of Gaussian white noise is shown in Fig. 3. From Fig. 3, it can be seen that the added white noise follow the normal distribution. Utilizing the Gaussian white noise to perturb the evolution of population, which can help the DPSO algorithm jump out of the local optimum and increase the diversity of population. The velocity updating function is improved by the following equation, where δ_{ip}^{k-1} and δ_{ig}^{k-1} indicate the Gaussian disturbed parameters for the personal best value and the global best value, respectively. In this algorithm, δ_{ip}^{k-1} and δ_{ig}^{k-1} are generated by the platform for each iteration. The distributions of δ_{ip}^{k-1} and δ_{ig}^{k-1} follow the normal distribution as shown in Fig. 3

$$Vel_i^k = \omega \cdot \widetilde{Vel}_i^{k-1} + c_1 \cdot r_1 \cdot \left(\delta_{ip}^{k-1} \cdot \widetilde{P}_{\text{pbesti}}^{k-1} - \widetilde{X}_i^{k-1}\right) + c_2 \cdot r_2 \cdot \left(\delta_{ig}^{k-1} \cdot \widetilde{P}_{gbest}^{k-1} - \widetilde{X}_i^{k-1}\right). \tag{14}$$

We utilize Gaussian white noise to improve Gaussian white noise-based-DPSO (G-DPSO), i.e., the G-DPSO is proposed to optimize the worker-centric task selection in mobile crowdsourcing. The proposed G-DPSO-based worker-centric task selection algorithm for w_i is summarized in Algorithm 1. In this paper, we divide the timeline into many time slots. In the process of G-DPSO-based worker-centric task selection algorithm, we optimize the task set for one-time slot. From the whole timeline, the task is dynamically selected as time goes on.

From Algorithm 1, the time complexity of the G-DPSO-based worker-centric task selection algorithm is $O(m \times n)$ in the while-loop (lines 7–17), where n indicates the maximal number of iterations, and m means the number of tasks. Since the value of each particle's position can only be 0 or 1, the whole solution space is finite. For the specific problem of mobile crowdsourcing task selection, there are a large number of infeasible solutions in the solution space, so the feasible solution space corresponding to the algorithm is also finite. Assuming that the maximum number of iterations of the algorithm is infinite, the particle's position always has the ability to change and new solutions may be constantly explored in the better solution domain. Therefore, for the finite feasible solution space, after the algorithm goes through an infinite number of iterations, the particle's position must be able to traverse all feasible solutions. The current optimal value is always saved to the next generation population until the end of the algorithm. Thus, the G-DPSO algorithm has global convergence in theory. However, in practice, since it is impossible to make the algorithm carry out infinite iteration calculations, the algorithm may still fall into local optimum in the actual optimization calculation. In the proposed G-DPSO algorithm, we add Gaussian white noise to perturb the velocity operator in the optimization process. Therefore, it can increase the diversity of population and avoid it falling into the local optimum, which is conducive to search for the global optimum and reduce the probability that the algorithm falls into the local optimum.

C. Platform-Centric Worker Selection Method

After selecting sensing tasks for workers that can maximize their utilities, workers will bid the interested tasks through submitting their biddings to the platform. Therefore, how to select appropriate crowd workers to sense the task is another important problem in our mechanism. In order to maximize the utility of platform, an effective platform-centric worker selection method is necessary. By combining multiattribute auction and two-stage auction, the platform-centric worker selection method is obtained. *Multiattribute auction* indicates that buyers not only consider the biddings of bidders but also consider their data quality, trust degree, location, and other attributes in the process of auction. In this paper, we combine multiattribute auction with the ITA [50] to resolve the problem of unfairness for the first arrived workers and to further improve the effectiveness of auction.

In a traditional two-stage auction, the bidding process is provided into two stages. The workers arriving during the first stage, the platform will reject them to participate in the task. However, their bidding information will be collected to compute and generate the auction threshold for the second stage of auction. Therefore, it is unfair for the early arriving workers that fail to inspire workers to participate in tasks further. ITA improves the traditional two-stage auction that allows the workers arriving during the first stage to participate in the auction. After collecting the bidding information in the first stage, the auction threshold is computed dynamically in the second stage, which can inspire workers to participate in tasks actively and solve the unfair problem. However, this method failed to consider the social attribute of workers, e.g., trust degree and privacy sensibility, which cannot inspire workers to behave truthfully, and cannot inspire the workers with high privacy sensibility to participate in the task actively. In our platform-centric worker selection method, considering the social attributes of workers, we combine multiattribute auction and ITA to propose the multi-attribute and ITA (M-ITA)

TABLE III
DESCRIPTIONS FOR NOTATIONS IN THE PLATFORM-CENTRIC WORKER SELECTION METHOD

Notation	*Description*
q_{im}	The average data quality of w_i.
tr_{im}	The average trust degree of w_i.
$d_{i(m+1)}$	w_i's distance to the correct sensing location.
$pr_{i(m+1)}$	The privacy sensitivity that w_i requested.
q_i^j	The specific data quality of w_i for his jth sensing task.
β_j	The time decay factor in the jth sensing task.
tr_i^j	The specific trust degree of w_i for his jth sensing task.
$(x_i^{m+1}, y_i^{m+1}, z_i^{m+1})$	w_i's position coordinates.
$(\overline{x}_i^{m+1}, \overline{y}_i^{m+1}, \overline{z}_i^{m+1})$	The correct sensing position.
Pr_{m+1}	The privacy sensitivity bound of the $(m+1)$th task.
$cs_{i(m+1)}$	The comprehensive score of w_i for his $(m+1)$th task.
ω_1, ω_2 and ω_3	The weights for q_{im}, tr_{im} and $d_{i(m+1)}$ respectively.
κ_{m+1}	The threshold of biddings for the $(m+1)$th sensing task of w_i.
CS_{m+1}	The threshold of multi-attribute score for the $(m+1)$th sensing task of w_i.
Pr_{m+1}	The privacy sensitivity threshold of multi-attribute score for the $(m+1)$th sensing task of w_i.

algorithm to improve the efficiency of auction. Similarly, the descriptions for notations appear in this section is given in Table III.

The attributes in this platform-centric worker selection include data quality, trust degree, position similarity, and privacy sensitivity. According to the new sensing task, i.e., the $(m+1)$th sensing task of w_i that he/she wants to participate in, the following attributes need to be computed: $(q_{\text{im}}, tr_{\text{im}}, d_{i(m+1)}, pr_{i(m+1)})$. The attribute q_{im} denotes the average data quality of w_i, and tr_{im} denotes the average trust degree of w_i. The values of q_{im} and tr_{im} are computed based on the historical information of w_i. The attribute $d_{i(m+1)}$ denotes the position similarity, i.e., w_is distance to the correct sensing location, and $pr_{i(m+1)}$ denotes the privacy sensitivity that w_i requested. The average data quality q_{im} is computed by the following equation:

$$q_{\text{im}} = \begin{cases} \dfrac{\sum_{j=1}^{m} q_i^j \cdot \beta_j}{\sum_{j=1}^{m} \beta_j}, & \text{if } \quad m \neq 0 \\ 0, & \text{otherwise} \end{cases} \tag{15}$$

where q_i^j indicates the specific data quality of w_i for his/her jth sensing task, and β_j is the time decay factor in the jth sensing task, which is computed based on the Ebbinghaus Forgetting Curve from psychology. The curve indicates that the influence of historical information will weaken gradually with the passage of time, thus, it will decay to 0 in the end. The parameter β_j is given by the following equation based on the Ebbinghaus Forgetting Curve:

$$\beta_j = \begin{cases} 1, & \text{if } \quad j = m \\ e^{-\frac{1}{j}}, & \text{else if } \quad 1 \leq j < m. \end{cases} \tag{16}$$

Another important factor is the average trust degree tr_{im}. Its calculation is similar to the average data quality, which is given by the following equation:

$$tr_{\text{im}} = \begin{cases} \dfrac{\sum_{j=1}^{m} tr_i^j \cdot \beta_j}{\sum_{j=1}^{m} \beta_j}, & \text{if } \quad m \neq 0 \\ 0, & \text{otherwise} \end{cases} \tag{17}$$

where tr_i^j expresses the specific trust degree of w_i for his/her jth sensing task. In addition, the position similarity $d_{i(m+1)}$ also influence the auction result. Let $(x_i^{m+1}, y_i^{m+1}, z_i^{m+1})$ indicate w_is position coordinates, and $(\overline{x}_i^{m+1}, \overline{y}_i^{m+1}, \overline{z}_i^{m+1})$ indicates the correct sensing position. When the crowd worker stays in the correct sensing position, $d_{i(m+1)} = 0$. Otherwise, $d_{i(m+1)}$ is computed using cosine similarity as shown in the following equation:

$$d_{i(m+1)} = \frac{x_i^{m+1} \cdot \overline{x}_i^{m+1} + y_i^{m+1} \cdot \overline{y}_i^{m+1} + z_i^{m+1} \cdot \overline{z}_i^{m+1}}{\sqrt{x_i^{m+1^2} + y_i^{m+1^2} + z_i^{m+1^2}} \cdot \sqrt{\overline{x}_i^{m+1^2} + \overline{y}_i^{m+1^2} + \overline{z}_i^{m+1^2}}}. \tag{18}$$

In our mechanism, q_{im} and tr_{im} are computed by the platform. The workers should announce their privacy sensitivity $pr_{i(m+1)}$ when submitting their biddings to the platform. When w_i receives tasks, he/she will submit his/her biddings that include $b_{i(m+1)}$, $t_{i(m+1)}$ and $pr_{i(m+1)}$ to the platform if he/she is interested in the task. For the new task, there exists a privacy sensitivity bound Pr_{m+1}, and $pr_{i(m+1)}$ should be no bigger than Pr_{m+1}. Therefore, before computing the biddings of w_i, the platform will calculate the comprehensive score $cs_{i(m+1)}$ of w_i based on its multiple attributes. The calculation method of $cs_{i(m+1)}$ is given by the following equation:

$$cs_{i(m+1)} = \omega_1 \cdot q_{\text{im}} + \omega_2 \cdot tr_{\text{im}} + \omega_3 \cdot d_{i(m+1)} \tag{19}$$

where ω_1, ω_2, and ω_3 represent the corresponding weights, and $\omega_1 + \omega_2 + \omega_3 = 1$.

In this paper, we combine the multiattribute auction with ITA [50] to maximize the utility of the platform. The process of auction is divided into two stages. In a normal two-stage auction, the first arrived workers will be rejected. The system then determines the threshold of bidding based on the biddings submitted by the first arrived workers, which will generate the unfairness problem for the first arrived workers.

In the ITA algorithm, the first arrived workers also have a chance to be the winner in the process of auction, thus eliminating the unfairness problem. Therefore, ITA can inspire

Algorithm 2 M-ITA-Based Platform-Centric Worker Selection Algorithm

Input:
$n,\ B_{m+1},\ T_{m+1},\ \kappa_{m+1},\ CS_{m+1},\ Pr_{m+1}$

Output:
$Win(\varphi_{m+1})$

1: Stage 1:
2: $B'_{m+1} = \lfloor \frac{B_{m+1}}{2^{\lfloor \ln T_{m+1} \rfloor}} \rfloor$;
3: $T'_{m+1} = \lfloor \frac{T_{m+1}}{2^{\lfloor \ln T_{m+1} \rfloor}} \rfloor$;
4: $i = 1$; $P_{m+1}(W) = 0$; $V_{m+1}(W) = 0$;
5: **while** $P_{m+1}(W) \leq B'_{m+1}$ **do**
6: $b_{i(m+1)} = b_{i(m+1)}$; $t_{i(m+1)} = t_{i(m+1)}$; $cs_{i(m+1)} = cs_{i(m+1)}$; $pr_{i(m+1)} = pr_{i(m+1)}$;
7: **if** $cs_{i(m+1)} \geq CS_{m+1}$, $\frac{b_{i(m+1)}}{t_{i(m+1)}} \leq \kappa_{m+1}$ and $pr_{i(m+1)} \leq Pr_{m+1}$ **then**
8: $\Gamma \leftarrow \Gamma \bigcup \{i\}$;
9: $p_{i(m+1)} \leftarrow \frac{t_{i(m+1)}}{T_{m+1}} \times B_{m+1}$;
10: $P_{m+1}(W) \leftarrow P_{m+1}(W) + p_{i(m+1)}$;
11: $V_{m+1}(W) \leftarrow V_{m+1}(W) + v_{i(m+1)}$;
12: **end if**
13: $i \leftarrow i + 1$;
14: Update the value of CS_{m+1};
15: **end while**
16: Stage 2:
17: $j = i$;
18: **while** $P_{m+1}(W) \leq B_{m+1}$ **do**
19: $b_{j(m+1)} = b_{j(m+1)}$; $t_{j(m+1)} = t_{j(m+1)}$; $cs_{j(m+1)} = cs_{j(m+1)}$; $pr_{j(m+1)} = pr_{j(m+1)}$;
20: $v_{j(m+1)} = V_{m+1}(W \bigcup \{j\}) - V_{m+1}(W)$; $p_{j(m+1)} = \frac{t_{j(m+1)}}{T_{m+1}} \times B_{m+1}$;
21: **if** $\frac{v_{j(m+1)}}{p_{j(m+1)}} \geq \frac{V_{m+1}(W)}{P_{m+1}(W)}$, $cs_{j(m+1)} \geq CS_{m+1}$ and $pr_{j(m+1)} \leq Pr_{m+1}$ **then**
22: $\Gamma \leftarrow \Gamma \bigcup \{j\}$;
23: $P_{m+1}(W) \leftarrow P_{m+1}(W) + p_{j(m+1)}$;
24: $V_{m+1}(W) \leftarrow V_{m+1}(W) + v_{j(m+1)}$;
25: Update the values of $\frac{V_{m+1}(W)}{P_{m+1}(W)}$, CS_{m+1};
26: **end if**
27: $j \leftarrow j + 1$;
28: **end while**

workers to participate in tasks on time, thus improving the efficiency of auction. For the second stage of ITA, the bidding threshold will be adjusted dynamically. In this paper, we further improve the ITA algorithm and propose the M-ITA algorithm by combining the multiattribute auction and ITA. The M-ITA algorithm is shown in Algorithm 2 with respect to the $(m+1)$th sensing task of w_i. In Algorithm 2, κ_{m+1} indicates the threshold of biddings for the $(m+1)$th sensing task of w_i, which is determined by the historical experience of mobile crowdsourcing. CS_{m+1} represents the threshold of multiattribute score for the new sensing task, and Pr_{m+1} is the privacy sensitivity threshold of the new sensing task.

In Algorithm 2, the first stage of the auction is shown from lines 1 to 15. Different from the traditional two-stage auction algorithm, workers who arrive in the first stage can also be accepted by the platform based on their multiple attributes. In the first stage, the initial thresholds κ_{m+1} and CS_{m+1} are given by platform. Based on the initial thresholds, the first arrived workers can participate in the auction. The bidding profiles in the first stage will be collected and computed by platform. At the end of the first stage, the value of CS_{m+1} is updated by the following equation. The parameter N_1 indicates the total number of workers in the first stage

$$CS_{m+1} = \frac{\sum_{i=1}^{N_1} cs_{i(m+1)}}{N_1}. \tag{20}$$

The second stage of auction is shown from lines 16 to 28 in Algorithm 2. In this stage, the threshold of biddings is $(V_{m+1}(W)/P_{m+1}(W))$ and it will dynamically change with the arriving of workers. In addition, the threshold of multiattribute score, CS_{m+1}, dynamically changes in this stage of auction. The value of $(V_{m+1}(W)/P_{m+1}(W))$ is updated by (21). In this stage of auction, we also give the calculation method of CS_{m+1}, which is updated by (22)

$$\frac{V_{m+1}(W)}{P_{m+1}(W)}_{(j+1)} = \frac{1}{2}\Big(\frac{V_{m+1}(W)}{P_{m+1}(W)}_{(j)} + \frac{v_{(j+1)(m+1)}}{p_{(j+1)(m+1)}}\Big) \tag{21}$$

where $(V_{m+1}(W)/P_{m+1}(W))_{(j+1)}$ means the threshold of biddings for the $(j+1)$th worker, and $(V_{m+1}(W)/P_{m+1}(W))_{(j)}$ is the threshold of biddings for the jth worker

$$CS_{m+1}^{(j+1)} = \frac{1}{2}\big(CS_{m+1}^{(j)} + cs_{j(m+1)}\big) \tag{22}$$

where $CS_{m+1}^{(j)}$ and $CS_{m+1}^{(j+1)}$ mean the threshold of multiattribute score for the jth and $(j+1)$th workers, respectively.

This proposed M-ITA algorithm not only can resolve the unfairness problem for the first arrived workers but can also improve the truthfulness and efficiency of MCSNs further by considering the multiple attributes of workers.

In order to maximize the final social welfare in one-time slot, we utilize Walrasian equilibrium in economics to change the value of payment p_{ij} dynamically. Walrasian equilibrium is the traditional concept of economic equilibrium, appropriate for the analysis of commodity markets with flexible prices and many traders, and serving as the benchmark of efficiency in economic analysis. The social welfare J, the total utility of mobile crowdsourcing system, could be expressed by the following equation:

$$J = \sum_{i=1}^{m}\sum_{j=1}^{n} u_{ij} + \sum_{j=1}^{n} \overline{u}_j. \tag{23}$$

Therefore, maximizing social welfare is the objective function of the incentive mechanism, which is shown in (24). The equal objective function of (24) is shown in (25), i.e., we can optimize the social welfare by (25)

$$\max J = \sum_{i=1}^{m}\sum_{j=1}^{n}(p_{ij} - c_{ij}) + \sum_{j=1}^{n}\sum_{i=1}^{m}(v_{ij} - p_{ij}) \tag{24}$$

$$\min -J = \sum_{i=1}^{m}\sum_{j=1}^{n}(c_{ij} - p_{ij}) - \sum_{j=1}^{n}\sum_{i=1}^{m}(v_{ij} - p_{ij}). \tag{25}$$

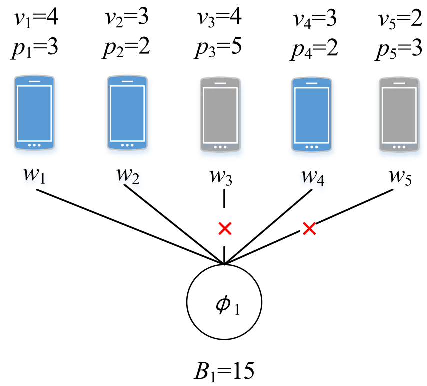

Fig. 4. Example showing the profit of the proposed incentive mechanism.

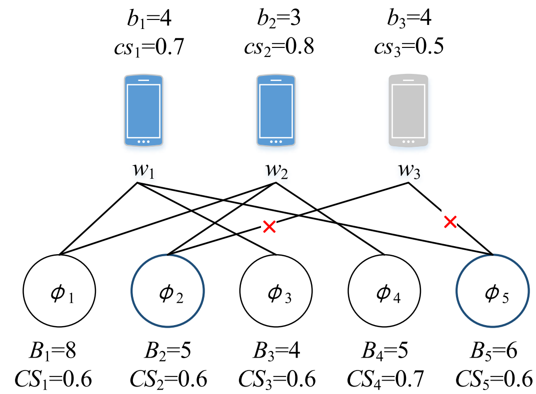

Fig. 5. Example showing the truthfulness of the proposed incentive mechanism.

In (25), the values of c_{ij} and υ_{ij} could not be changed by platform. Hence, dynamically change the value of p_{ij} to achieve the global optimum and maximize social welfare. According to the above-mentioned analysis, the proposed incentive mechanism can dynamically recommend tasks and select winners in the process of optimization and auction, which can maximize social welfare.

D. Properties of the Proposed Incentive Mechanism

In this paper, we study the incentive mechanism in order to maximize social welfare of MCSNs. An effective incentive mechanism needs to have four properties: computational efficiency, individual rationality, profitability, and truthfulness.

Lemma 1: The proposed incentive mechanism is computationally efficient.

Proof: The number of tasks is m. The maximum number of iterations is n. Based on Algorithm 1, the time complexity of the G-DPSO-based worker-centric task selection algorithm is $O(m \times n)$ in the while-loop (lines 7–17). In the M-ITA-based platform-centric worker selection mechanism shown in Algorithm 2, assume that n indicates the total number of bidders, thus the time complexity is at most $O(n)$ in the while-loop (lines 5–15 and 18–28). According to the above-mentioned analysis, it can be seen that the proposed incentive mechanism is computationally efficient.

Lemma 2: The proposed incentive mechanisms are individually rational.

Proof: If the platform determines that w_i is a winner for φ_j, w_i will receive payment $p_{ij} = (t_{ij}/T_j) \times B_j$ from the platform. In (2), the budget B_j and required total sensing time T_j satisfy the constraint: $(B_j/T_j) \geq 1$. From (2), we can obtain $c_{ij} = \tau_i \times t_{ij}$, where $0 < \tau_i < 1$. Therefore, the utility of w_i is $p_{ij} - c_{ij} = ((B_j/T_j) - \tau_i) \times t_{ij} > 0$. According to the above-mentioned analysis, we can infer that the proposed incentive mechanism is individually rational.

Lemma 3: The proposed incentive mechanism is profitable.

Proof: According to (3), $\bar{u_j} = V_j(W) - P_j(W)$, where $V_j(W) = \sum_{w_i \in Win(\varphi_j)} \upsilon_{ij}$ and $P_j(W) = \sum_{w_i \in Win(\varphi_j)} p_{ij}$. For the M-ITA algorithm, an instance to verify this property of the proposed incentive mechanism is shown in Fig. 4.

In Fig. 4, the mobile phones represent crowd workers, and the circles represent crowd tasks. Assume the initial value of $(V_1(W)/P_1(W))$ is 1. When w_1 participates in the auction, $(\upsilon_1/p_1) = (4/3)$, which satisfies $(\upsilon_1/p_1) > (V_1(W)/P_1(W))$, thus, w_1 is selected by the platform. In this stage, $(V_1(W)/P_1(W)) = (4/3)$. When w_2 participates it the auction, $(\upsilon_2/p_2) = (3/2)$, which satisfies $(\upsilon_2/p_2) > (V_1(W)/P_1(W))$, thus, w_2 is selected by the platform. Therefore, $(V_1(W)/P_1(W)) = (4+3/3+2) = (7/5)$. For w_3, $(\upsilon_3/p_3) = (4/5)$ and $(\upsilon_3/p_3) < (V_1(W)/P_1(W))$, thus, w_3 cannot be selected by the platform. For w_4, $(\upsilon_4/p_4) = (3/2)$, which satisfies $(\upsilon_4/p_4) > (V_1(W)/P_1(W)) = (4/3)$, thus, w_4 is selected by the platform. After this bidding, $(V_1(W)/P_1(W)) = (4+3+3/3+2+2) = (10/7)$. When w_5 participates in the auction, $(\upsilon_5/p_5) = (2/3)$ and $(\upsilon_5/p_5) < (V_1(W)/P_1(W))$, thus, w_5 cannot be selected by the platform. In this auction, w_1, w_2, and w_4 are selected to be the winners. According to this task, the platform utility is $\bar{u_1} = V_1(W) - P_1(W) = 10-7 = 3 > 0$. Therefore, the platform has nonnegative utility. According to the above-mentioned analysis, we can infer that the proposed incentive mechanism is profitable.

Lemma 4: The proposed incentive mechanism is truthful.

Proof: In the M-ITA-based task selection mechanism shown in Algorithm 2, multiple attributes are considered when w_i participates in the auction for φ_j. When participating the auction, the value of $cs_{i(m+1)}$ is calculated by platform, i.e., $cs_{i(m+1)} = \omega_1 \cdot q_{\text{im}} + \omega_2 \cdot tr_{\text{im}} + \omega_3 \cdot d_{i(m+1)}$. Therefore, if w_i wants to be the winner of auction, $cs_{i(m+1)}$ satisfies the condition $cs_{i(m+1)} \geq CS_{m+1}$, i.e., w_i should keep the truthful behaviors in order to get more opportunities to participate in subsequent tasks. We use an example shown in Fig. 5 to verify the proposed incentive mechanism is truthful.

In Fig. 5, the mobile phones represent crowd workers, and the circles represent crowd tasks. In this scenario, we assume that the three crowd workers act in the target area, thus, $d_1 = d_2 = d_3 = 0$. Therefore, the values of cs_i can express the truthfulness of crowd workers. A line indicates that a crowd worker bids for a crowd task. For φ_1, $cs_1 > CS_1$ and $b_1 < B_1$, thus, w_1 is selected. Then, w_2 participates in the auction, $cs_2 > CS_1$ and $b_1 + b_2 < B_1$, thus, w_2 is selected. For φ_2, $cs_2 > CS_2$ and $b_2 < B_2$, thus, w_2 is selected. When w_3 participates in the auction, $cs_3 < CS$, so w_3 cannot be selected in this auction. Similarly, w_3 cannot be selected by φ_5 because $cs_3 < CS_5$. From the instance, it can be inferred that the crowd workers with low trust degree cannot participate

TABLE IV
SETTINGS OF EXPERIMENTS

	Populations	m	B_i^w
First group	60	6	3000
Second group	80	8	4000
Third group	100	10	5000

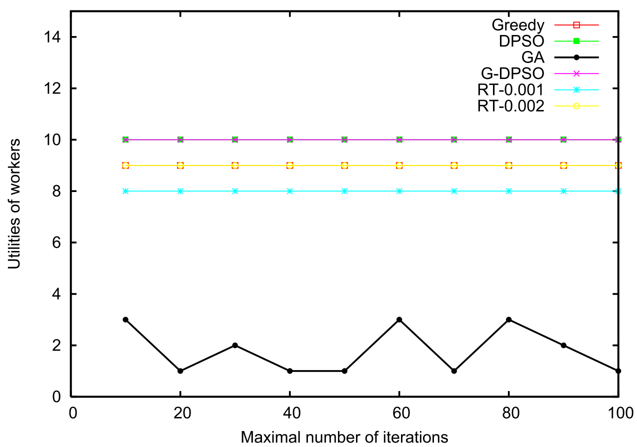

Fig. 6. Comparison results of utilities for the first group of settings.

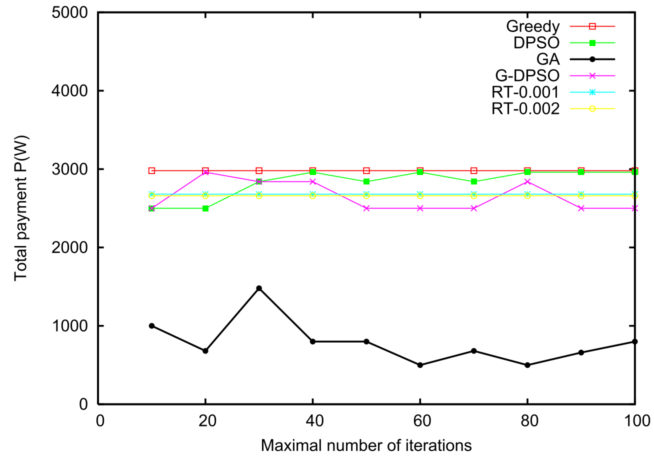

Fig. 7. Comparison results of $P(W)$ for the first group of settings.

in the crowd workers, which can guarantee the truthfulness of the proposed incentive mechanism. According to the above-mentioned analysis, we can infer that the proposed incentive mechanism is truthful.

IV. NUMERICAL SIMULATIONS

In this section, comparison experiments are conducted for evaluating the performances of the proposed incentive mechanism. For the worker-centric task selection method, we compared the proposed task selection algorithm G-DPSO with the Greedy algorithm, DPSO algorithm, GA, and random threshold algorithm (RT). In our comparison experiments, the thresholds are set as 0.001 and 0.002 for RT. For the platform-centric worker selection method, we evaluate the performance of M-ITA by comparing with the ITA algorithm [50], the auction algorithm in [57] as the general auction (RA), and the traditional two-stage auction algorithm [58]. All the comparison experiments were performed on Windows 10 operating system with Intel Core (TM) i7-5500U CPU, 8-GB Memory, and on MATLAB 7.0. In our simulations, we conduct the event-based simulations, and each evaluation is averaged over 100 instances.

A. Experiments for Worker-Centric Task Selection

For the G-DPSO-based worker-centric task selection algorithm, we conduct three groups of comparison experiments by comparing with the Greedy-based task selection algorithm, the G-DPSO-based task selection algorithm, the GA-based task selection algorithm, and the RT-based task selection algorithm. The experimental settings are shown in Table IV, where m indicates the total number of tasks.

According to the first group of settings, the comparison results for utilities are shown in Fig. 6. Compared with the Greedy-based, GA-based, and RT-based task selection algorithms, DPSO and G-DPSO algorithms can maximize the utility of the worker effectively. In Fig. 6, the x-coordinate indicates the maximal number of iterations, and y-coordinate means the utility of the worker. We also design the comparison experiments according to the cost of the worker, which is shown in Fig. 7. In Fig. 7, the x-coordinate indicates the maximal number of iterations, and y-coordinate means the total payment $P(W)$. It also can be seen that GA performs worst through compared with other algorithms. The reason is that GA is easy to be trapped into local optimum, which cannot find the global optimum. In addition, GA-based task selection method has the slowest search speed. As a GA-based task selection method cannot recommend tasks for workers effectively, $P(W)$ is the least compared with the other three algorithms. Greedy-based algorithm has the most $P(W)$ compared with the other four algorithms. RT-based task selection algorithm has lower $P(W)$ compared with Greedy-based and DPSO-based algorithms but less utility than other optimization algorithms except GA-based method. Both the DPSO-based and G-DPSO-based task selection algorithms can maximize the utility of the worker (”10” in this experiments), however, G-DPSO-based algorithm has less $P(W)$ compared with the DPSO-based algorithm. Therefore, G-DPSO-based has the best performance in this group of experiments.

According to the second group of settings, the corresponding comparison results for utilities and $P(W)$ are shown in Figs. 8 and 9, respectively. In these experimental settings, we can see that the performance of G-DPSO-based algorithm has a significant advantage over the other methods. GA-based task selection algorithm also has the worst performance compared with other algorithms because of the premature problem of GA. In addition, it has the slowest search speed.

According to the third group of settings, the comparison results for utilities and $P(W)$ are shown in Figs. 10 and 11, respectively. Based on the experimental results, we can see that the G-DPSO-based task selection algorithm also performs the best among the five algorithms. GA-based task selection algorithm also has the slowest search speed. In addition, it performs worst compared with other methods.

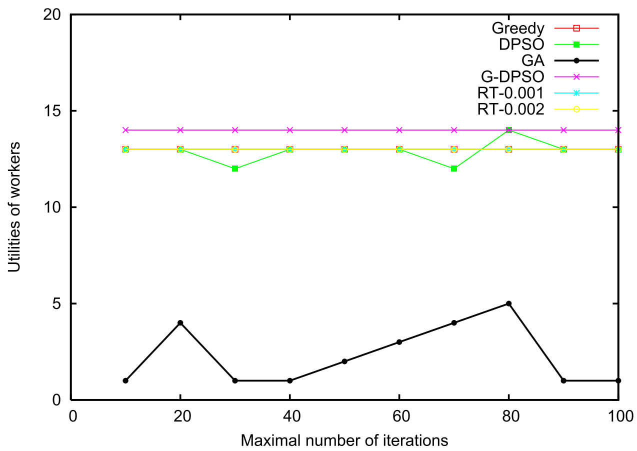

Fig. 8. Comparison results of utilities for the second group of settings.

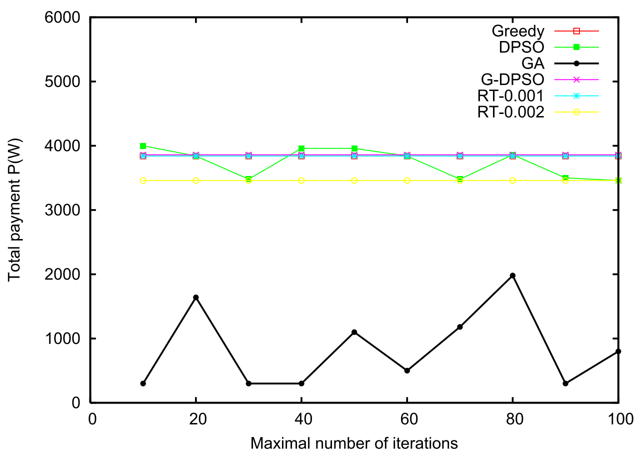

Fig. 9. Comparison results of costs for the second group of settings.

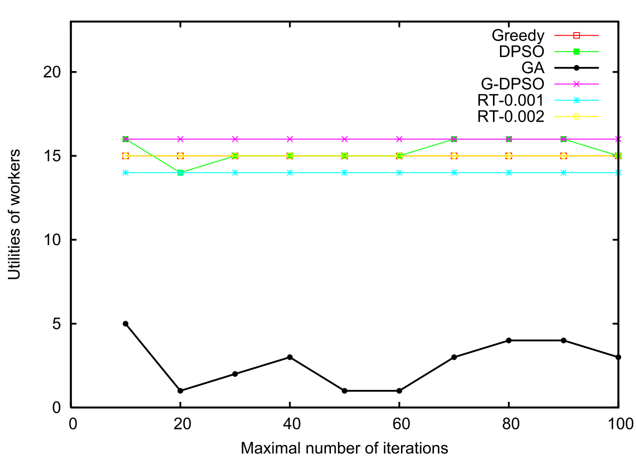

Fig. 10. Comparison results of utilities for the third group of settings.

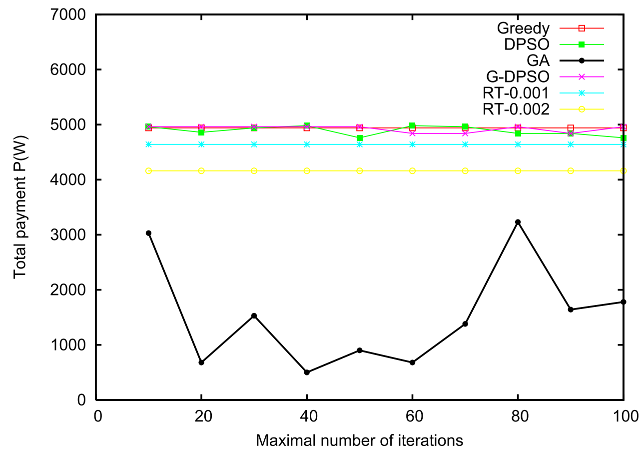

Fig. 11. Comparison results of costs for the third group of settings.

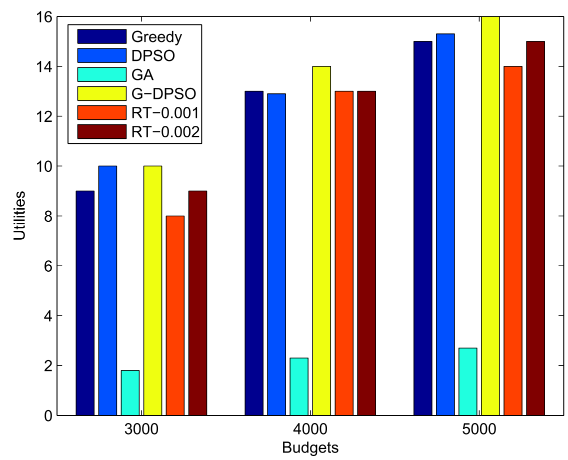

Fig. 12. Comparison results of utilities.

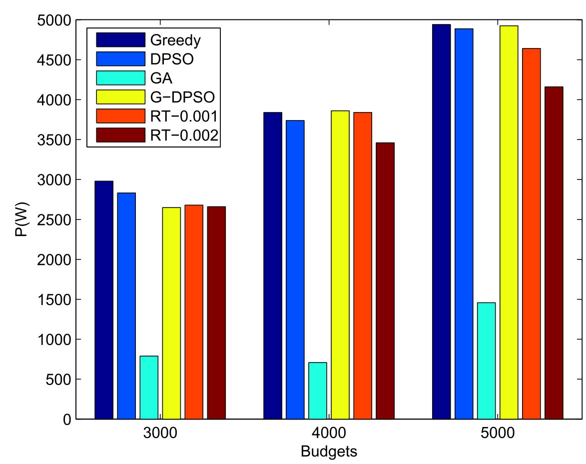

Fig. 13. Comparison results of $P(W)$.

In addition, we also compare the average utilities under different budgets, the comparison results are shown in Fig. 12. From Fig. 12, it can be inferred that G-DPSO-based worker-centric task selection algorithm can maximize the utility of the worker effectively, thus it has the best performance. The comparison results of average $P(W)$ under different budgets are shown in Fig. 13. The G-DPSO-based worker-centric task selection algorithm, Greedy-based task selection algorithm, DPSO-based task selection algorithm, and RT-based task selection algorithm have similar $P(W)$, and they can maximize the utility of the worker under acceptable $P(W)$. From the experimental results, it can be inferred that the proposed

TABLE V
SETTINGS FOR PLATFORM-CENTRIC WORKER SELECTION

	B_m	T_m	n	κ_m	CS_m	Pr_m
First group	50	25	40	2	0.7	5
Second group	100	50	60	2	0.7	5
Third group	200	100	80	2	0.7	5

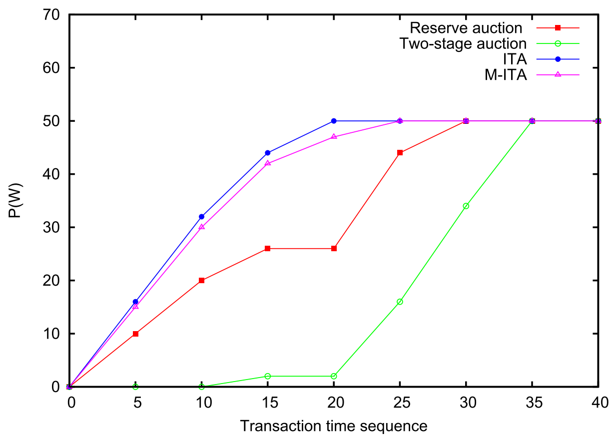

Fig. 14. Comparison results of efficiencies for the first group.

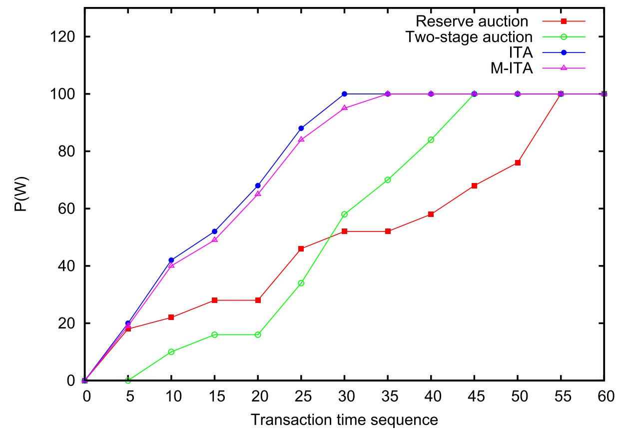

Fig. 15. Comparison results of efficiencies for the second group.

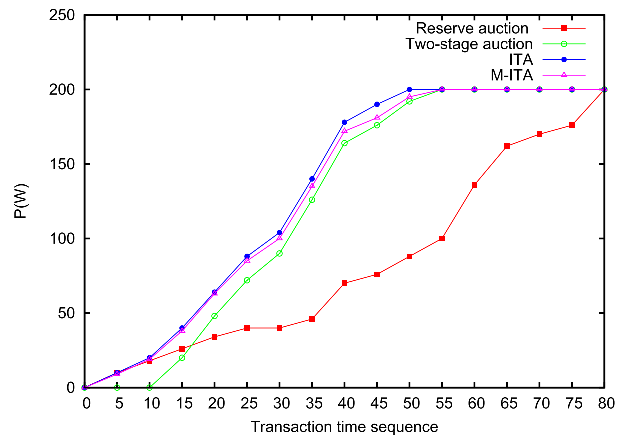

Fig. 16. Comparison results of efficiencies for the third group.

G-DPSO-based worker-centric task selection algorithm has a faster convergence speed than other optimization algorithms.

According to the above-mentioned experiments, we can infer that the proposed G-DPSO-based worker-centric task selection method can maximize utilities of workers effectively. Through compared with nonoptimization algorithm (RT), optimization algorithms have better performance, which can improve the efficiency of MCSNs effectively. We also can see that the proposed G-DPSO can improve the utilities effectively, and the average values of $P(W)$ stay under acceptable range. Therefore, the proposed G-DPSO can effectively recommend appropriate tasks for workers, which can inspire workers to participate in sensing tasks better.

B. Experiments for Platform-Centric Worker Selection

Here, we perform the comparison experiments for M-ITA algorithm with RA, two-stage auction, and ITA algorithms. The experimental settings are shown in Table V, which comes from [50]. In Table V, n represents the total number of workers.

The experimental results for the three groups of settings are shown in Figs. 14–16, respectively. In Figs. 14–16, the x-coordinate expresses the transaction time, and the y-coordinate shows the value of $P(W)$. The experimental results reflect the efficiencies of different auction algorithms applied in mobile crowdsourcing. From the three groups of experimental results shown in Figs. 14–16, it can be inferred that ITA and M-ITA have the best performances compared with the other two algorithms. In Fig. 14, it can be seen that the two-stage auction performs the worst, it is because the first batch of workers is rejected by the platform. In the first group of comparison experiment, $B_m = 50$ is too limited for the two-stage auction. Therefore, the two-stage auction has the worst performance in this experiment. With the increase of B_m, the two-stage auction has a better performance than the RA. The reason is that when B_m is big enough, the influence of the first arrived workers will decrease. Based on the experimental results, we can also infer that the efficiency of ITA is slightly better than M-ITA. It is because we consider the multiple attributes of workers, i.e., cs_i^m. Some workers will be eliminated if their comprehensive score cannot satisfy the condition $cs_j^m \geq CS_m$. However, by considering multiattribute of workers, the truthfulness of MCSNs can be guaranteed. In addition, this algorithm can inspire workers to submit truthful data to the platform, and can, therefore, improve the truthfulness and efficiency of MCSNs.

In order to verify the truthfulness of the proposed M-ITA, we design the comparison experiments on the truthfulness of mobile crowdsourcing system. When the number of bidders is 60, the comparison result is shown in Fig. 17. The x-coordinate expresses the number of bidders, and the y-coordinate indicates the trust degree of mobile crowdsourcing system. From Fig. 17, it can be seen that the proposed M-ITA has the best performance on truthfulness through compared with other algorithms. The trust degree of mobile crowdsourcing under M-ITA can stabilize around 0.8, but ITA only can stabilize around 0.3.

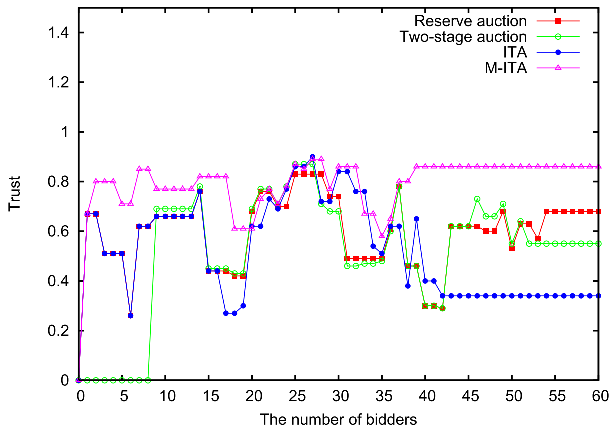

Fig. 17. Trust degrees of mobile crowdsourcing under different auction algorithms, when $n = 60$.

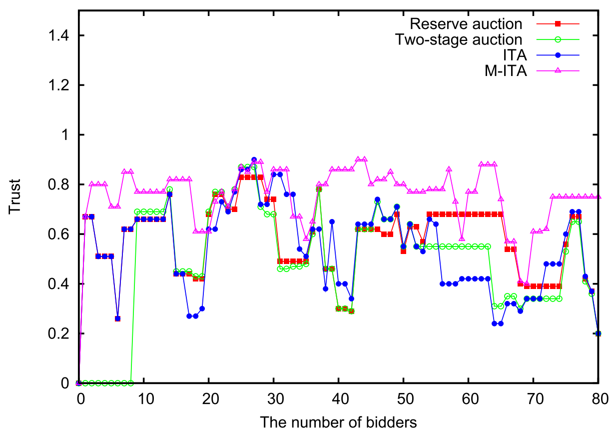

Fig. 18. Trust degrees of mobile crowdsourcing under different auction algorithms, when $n = 80$.

When $n = 80$, the comparison experiment is also conducted in this paper. The corresponding experimental result is shown in Fig. 18, where x-coordinate expresses the number of bidders, and the y-coordinate indicates the trust degree of mobile crowdsourcing system. From Fig. 18, we can see that the proposed M-ITA also has the best performance on truthfulness through compared with other algorithms.

From the above-mentioned experimental results, we can infer that the proposed M-ITA can guarantee the truthfulness and efficiency of mobile crowdsourcing system comprehensively. Therefore, M-ITA has the best performance through compared with the other three auction algorithms. As social welfare of MCSNs is constituted by worker utility and platform, we can infer that our incentive mechanism has the best performance according to the above-mentioned experiments.

V. Conclusion

With the development of mobile crowdsourcing, how to assign appropriate tasks for workers has become the main research focus. This paper proposed an incentive mechanism, i.e., G-DPSO-based worker-centric task selection and M-ITA-based platform-centric worker selection. The proposed G-DPSO-based worker-centric task selection method can maximize worker utility by using the DPSO algorithm with Gaussian white noise perturbation. The proposed M-ITA-based platform-centric worker selection method combines multiattribute auction and the ITA algorithm and is able to maximize the utility of platform effectively. Through comparison experiments, the effectiveness, efficiency, truthfulness, and adaptiveness of the proposed incentive mechanism were verified thoroughly. It was shown convincingly that the proposed incentive mechanism can improve the truthfulness and the efficiency of MCSNs.

In future work, we further consider the social attributes of workers to establish the truthful incentive mechanism for improving the efficiency of mobile crowdsourcing system effectively.

References

[1] X. Zhang *et al.*, "Incentives for mobile crowd sensing: A survey," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 1, pp. 54–67, 1st Quart., 2016.

[2] Y. Wen *et al.*, "Quality-driven auction-based incentive mechanism for mobile crowd sensing," *IEEE Trans. Veh. Technol.*, vol. 64, no. 9, pp. 4203–4214, Sep. 2015.

[3] W. Shi, W.-N. Chen, Y. Lin, T. Gu, S. Kwong, and J. Zhang, "An adaptive estimation of distribution algorithm for multipolicy insurance investment planning," *IEEE Trans. Evol. Comput.*, vol. 23, no. 1, pp. 1–4, Feb. 2018.

[4] T. Yong-Xin, Y. Ye, C. Yu-Rong, C. Lei, and W. Guo-Ren , "Survey on spatiotemporal crowdsourced data management techniques," *J. Softw.*, vol. 28, no. 1, pp. 35–58, 2017.

[5] Y.-H. Zhang, Y.-J. Gong, W.-N. Chen, T.-L. Gu, H.-Q. Yuan, and J. Zhang, "A dual-colony ant algorithm for the receiving and shipping door assignments in cross-docks," *IEEE Trans. Intell. Transp. Syst.*, to be published.

[6] Y. Tong, J. She, B. Ding, L. Wang, and L. Chen, "Online mobile micro-task allocation in spatial crowdsourcing," in *Proc. IEEE Int. Conf. Data Eng.*, May 2016, pp. 49–60.

[7] Z.-G. Chen *et al.*, "Multiobjective cloud workflow scheduling: A multiple populations ant colony system approach," *IEEE Trans. Cybern.*, to be published. doi: 10.1109/TCYB.2018.2832640.2018.

[8] Y. Hu, Y. Wang, Y. Li, and X. Tong, "An incentive mechanism in mobile crowdsourcing based on multi-attribute reverse auctions," *Sensors*, vol. 18, p. 3453, Oct. 2018.

[9] Z.-H. Zhan *et al.*, "Cloudde: A heterogeneous differential evolution algorithm and its distributed cloud version," *IEEE Trans. Parallel Distrib. Syst.*, vol. 28, no. 3, pp. 704–716, Mar. 2017.

[10] Y. Tong, J. She, B. Ding, L. Chen, T. Wo, and K. Xu, "Online minimum matching in real-time spatial data: experiments and analysis," *Proc. VLDB Endowment*, vol. 9, no. 12, pp. 1053–1064, 2016.

[11] Y. Li, M. L. Yiu, and W. Xu, "Oriented online route recommendation for spatial crowdsourcing task workers," in *Proc. Int. Symp. Spatial Temporal Databases*, Aug. 2015, pp. 137–156.

[12] X.-F. Liu, Z.-H. Zhan, J. D. Deng, Y. Li, T. L. Gu, and J. Zhang, "An energy efficient ant colony system for virtual machine placement in cloud computing," *IEEE Trans. Evol. Comput.*, vol. 22, no. 1, pp. 113–128, Feb. 2018.

[13] Y.-H. Jia *et al.*, "Distributed cooperative co-evolution with adaptive computing resource allocation for large scale optimization," *IEEE Trans. Evol. Comput.*, to be published.

[14] X. Zhang, Z. Yang, Y.-J. Gong, Y. Liu, and S. Tang, "SpatialRecruiter: Maximizing sensing coverage in selecting workers for spatial crowdsourcing," *IEEE Trans. Veh. Technol.*, vol. 66, no. 6, pp. 5229–5240, Jun. 2017.

[15] H. To, C. Shahabi, and L. Kazemi, "A server-assigned spatial crowdsourcing framework," *ACM Trans. Spatial Algorithms Syst.*, vol. 1, no. 1, pp. 1–28, 2015.

[16] X.-F. Liu, Z.-H. Zhan, Y. Gao, J. Zhang, S. Kwong, and J. Zhang, "Coevolutionary particle swarm optimization with bottleneck objective learning strategy for many-objective optimization," *IEEE Trans. Evol. Comput.*, to be published. doi: 10.1109/TEVC.2018.2875430.2018.

[17] M. Han, M. Yan, Z. Cai, and Y. Li, "An exploration of broader influence maximization in timeliness networks with opportunistic selection," *J. Netw. Comput. Appl.*, vol. 63, p. 39–49, Mar. 2016.

[18] Y. Kang, X. Miao, K. Liu, L. Chen, and Y. Liu, "Quality-aware online task assignment in mobile crowdsourcing," in *Proc. 12th Int. Conf. Mobile Ad Hoc Sensor Syst.*, Oct. 2015, pp. 127–135.

[19] Z.-H. Zhan, X.-F. Liu, Y.-J. Gong, J. Zhang, H. S.-H. Chung, and Y. Li, "Cloud computing resource scheduling and a survey of its evolutionary approaches," *ACM Comput. Surv.*, vol. 47, no. 4, p. 63, 2015.

[20] J. Li, Z. Cai, M. Yan, and Y. Li, "Using crowdsourced data in location-based social networks to explore influence maximization," in *Proc. 35th Annu. IEEE Int. Conf. Comput. Commun.*, Apr. 2016, pp. 1–9.

[21] Y. Wang, G. Yin, Z. Cai, Y. Dong, and H. Dong, "A trust-based probabilistic recommendation model for social networks," *J. Netw. Comput. Appl.*, vol. 55, pp. 59–67, Sep. 2015.

[22] J. Li, Z. Cai, J. Wang, M. Han, and Y. Li, "Truthful incentive mechanisms for geographical position conflicting mobile crowdsensing systems," *IEEE Trans. Comput. Social Syst.*, vol. 5, no. 2, pp. 324–334, Jun. 2018.

[23] Z. Chi, Y. Wang, Y. Huang, and X. Tong, "The novel location privacy-preserving CKD for mobile crowdsourcing systems," *IEEE Access*, vol. 6, pp. 5678–5687, 2017.

[24] X. Zheng, Z. Cai, J. Li, and H. Gao, "Location-privacy-aware review publication mechanism for local business service systems," in *Proc. IEEE Conf. Comput. Commun.*, May 2017, pp. 1–9.

[25] Z. Cai, Z. He, X. Guan, and Y. Li, "Collective data-sanitization for preventing sensitive information inference attacks in social networks," *IEEE Trans. Depend. Sec. Comput.*, vol. 15, no. 4, pp. 577–590, Jul./Aug. 2016.

[26] L. Kazemi and C. Shahabi, "GeoCrowd: Enabling query answering with spatial crowdsourcing," in *Proc. 20th Int. Conf. Adv. Geographic Inf. Syst.*, Nov. 2012, pp. 189–198.

[27] L. Kazemi, C. Shahabi, and L. Chen, "GeoTruCrowd: Trustworthy query answering with spatial crowdsourcing," in *Proc. Adv. Geographic Inf. Syst.*, Nov. 2013, pp. 314–323.

[28] J. She, Y. Tong, and L. Chen, "Utility-aware social event-participant planning," in *Proc. Int. Conf. Manage. Data.*, May 2015, pp. 1629–1643.

[29] X. Zheng, Z. Cai, J. Li, and H. Gao, "A study on application-aware scheduling in wireless networks," *IEEE Trans. Mobile Comput.*, vol. 16, no. 7, pp. 1787–1801, Jul. 2017.

[30] P. Cheng *et al.*, "Reliable diversity-based spatial crowdsourcing by moving workers," *Proc. VLDB Endowment*, vol. 8, no. 10, pp. 1022–1033, Jun. 2015.

[31] J. She, Y. Tong, L. Chen, and C. C. Cao, "Conflict-aware event-participant arrangement and its variant for online setting," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 9, pp. 2281–2295, Sep. 2016.

[32] T. Song, Y. Tong, L. Wang, and K. Xu, "Online task assignment for three types of objects under spatial crowdsourcing environment," *J. Softw.*, vol. 28, no. 3, pp. 611–630, 2017.

[33] W. Luo, H. Tan, L. Chen, and L. M. Ni, "Finding time period-based most frequent path in big trajectory data," in *Proc. ACM SIGMOD Int. Conf. Manage. Data* , Jun. 2013, pp. 713–724.

[34] H. Su, K. Zheng, J. Huang, H. Jeung, L. Chen, and X. Zhou, "Crowd-Planner: A crowd-based route recommendation system," in *Proc. IEEE 30th Int. Conf. Data Eng.*, Mar./Apr. 2013, pp. 1144–1155.

[35] D. Deng, C. Shahabi, and U. Demiryurek, "Maximizing the number of worker's self-selected tasks in spatial crowdsourcing," in *Proc. 21st ACM SIGSPATIAL Int. Conf. Adv. Geographic Inf. Syst.*, 2013, pp. 324–333.

[36] D. Deng, C. Shahabi, and L. Zhu, "Task matching and scheduling for multiple workers in spatial crowdsourcing," in *Proc. 23rd SIGSPATIAL Int. Conf. Adv. Geographic Inf. Syst.*, 2015, p. 21.

[37] J.-H. Zhu and H. Gao, "An energy efficient algorithm for task allocation in wireless sensor networks," *J. Softw.*, vol. 18, no. 5, p. 1198, 2007.

[38] Y.-J. Gong *et al.*, "Genetic learning particle swarm optimization," *IEEE Trans. Cybern.*, vol. 46, no. 10, pp. 2277–2290, Oct. 2016.

[39] Q. Yang *et al.*, "Adaptive multimodal continuous ant colony optimization," *IEEE Trans. Evol. Comput.*, vol. 21, no. 2, pp. 191–205, Apr. 2017.

[40] Y.-J. Gong, E. Chen, X. Zhang. L. M. Ni, and J. Zhang, "AntMapper: An ant colony-based map matching approach for trajectory-based applications," *IEEE Trans. Intell Transp. Syst.*, vol. 19, no. 2, pp. 390–401, Feb. 2018.

[41] N. Higashi and H. Iba, "Particle swarm optimization with Gaussian mutation," in *Proc. IEEE Swarm Intell. Symp.*, Apr. 2003, pp. 72–79.

[42] J.-S. Lee and B. Hoh, "Sell your experiences: a market mechanism based incentive for participatory sensing," in *Proc. IEEE Int. Conf. Pervas. Comput. Commun.*, Mar./Apr. 2010, pp. 60–68.

[43] Z. Duan, W. Li, and Z. Cai, "Distributed auctions for task assignment and scheduling in mobile crowdsensing systems," in *Proc. IEEE 37th Int. Conf. Distrib. Comput. Syst.*, Jun. 2017, pp. 635–644.

[44] Z. Feng, Y. Zhu, Q. Zhang, L. M. Ni, and A. V. Vasilakos, "TRAC: Truthful auction for location-aware collaborative sensing in mobile crowdsourcing," in *Proc. IEEE Conf. Comput. Commun.*, Apr./May 2014, pp. 1231–1239.

[45] I. Krontiris and A. Albers, "Monetary incentives in participatory sensing using multi-attributive auctions," *Int. J. Parallel Emergent Distrib. Syst.*, vol. 27, no. 4, pp. 317–336, 2012.

[46] T. Luo, H.-P. Tan, and L. Xia, "Profit-maximizing incentive for participatory sensing," in *Proc. IEEE Conf. Comput. Commun.*, Apr./May 2014, pp. 127–135.

[47] D. Yang, X. Fang, and G. Xue, "Truthful incentive mechanisms for k-anonymity location privacy," in *Proc. IEEE INFOCOM*, Apr. 2013, pp. 2994–3002.

[48] Y. Wang, Z. Cai, X. Tong, Y. Gao, and G. Yin, "Truthful incentive mechanism with location privacy-preserving for mobile crowdsourcing systems," *Comput. Netw.*, vol. 135, pp. 32–43, Apr. 2018.

[49] L. Gao, F. Hou, and J. Huang, "Providing long-term participation incentive in participatory sensing," in *Proc. IEEE Conf. Comput. Commun.*, Apr./May 2015, pp. 2803–2811.

[50] Y. Wang, Z. Cai, G. Yin, Y. Gao, X. Tong, and G. Wu, "An incentive mechanism with privacy protection in mobile crowdsourcing systems," *Comput. Netw.*, vol. 102, pp. 157–171, Jun. 2016.

[51] Z.-H. Zhan, J. Li, J. Cao, J. Zhang, H. S.-H. Chung, and Y.-H. Shi, "Multiple populations for multiple objectives: A coevolutionary technique for solving multiobjective optimization problems," *IEEE Trans. Cybern.*, vol. 43, no. 2, pp. 445–463, Apr. 2013.

[52] X. Zhang, Z. Yang, Y. Liu, J. Li, and Z. Ming, "Toward efficient mechanisms for mobile crowdsensing," *IEEE Trans. Veh. Technol.*, vol. 66, no. 2, pp. 1760–1771, Feb. 2017.

[53] X. Wen *et al.*, "A maximal clique based multiobjective evolutionary algorithm for overlapping community detection," *IEEE Trans. Evol. Comput.*, vol. 21, no. 3, pp. 363–377, Jun. 2017.

[54] B. Zhao, Y. Wang, Y. Li, Y. Gao, and X. Tong, "Task allocation model based on worker friend relationship for mobile crowdsourcing," *Sensors*, vol. 19, p. 921, Feb. 2019.

[55] Y.-J. Gong and Y. Zhou, "Differential evolutionary superpixel segmentation," *IEEE Trans. Image Process.*, vol. 27, no. 3, pp. 1390–1404, Mar. 2018.

[56] Y.-J. Gong, J. Zhang, and Y. Zhou, "Learning multimodal parameters: A bare-bones niching differential evolution approach," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 7, pp. 2944–2959, Jul. 2018.

[57] C. Z. Zheng, "High bids and broke winners," *J. Econ. Theory*, vol. 100, no. 1, pp. 129–171, Sep. 2001.

[58] M. Bateni, M. Hajiaghayi, and M. Zadimoghaddam, "Submodular secretary problem and extensions," *ACM Trans. Algorithms*, vol. 9, no. 4, pp. 39–52, 2013.

Yingjie Wang (M'16) received the Ph.D. degree from the College of Computer Science and Technology, Harbin Engineering University, Harbin, China, in 2015.

From 2013 to 2014, she was a Visiting Scholar with Georgia State University, Atlanta, GA, USA. She is currently an Associate Professor with the School of Computer and Control Engineering, Yantai University, Yantai, China. She holds a post-doctoral position with the South China University of Technology, Guangzhou, China. She has presided one National Natural Science Foundation of China Project, one China Postdoctoral Science Foundation, and joined three National Natural Science Foundation of China projects and one Natural Science Foundation of Shandong Province Project. She has authored or co-authored more than 30 papers in well-known journals and conferences in her research field, which include an ESI high cited paper. Her current research interests include mobile crowdsourcing, privacy protection, and trust computing.

Zhipeng Cai (SM'08) received the B.S. degree from the Department of Computer Science and Engineering, Beijing Institute of Technology, Beijing, China, in 2001, and the M.S. and Ph.D. degrees from the Department of Computing Science, University of Alberta, Edmonton, AB, Canada, in 2004 and 2008, respectively.

He was a Research Faculty with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, USA. He is currently an Associate Professor with the Department of Computer Science, Georgia State University, Atlanta, GA, USA. His current research interests include networking and big data.

Dr. Cai was a recipient of the NSF CAREER Award.

Zhi-Hui Zhan (M'13–SM'18) received the bachelor's and Ph.D. degrees from the Department of Computer Science, Sun Yat-Sen University, Guangzhou, China, in 2007 and 2013, respectively.

He is currently the Changjiang Scholar Young Professor and the Pearl River Scholar Young Professor with the School of Computer Science and Engineering, South China University of Technology, Guangzhou. He has authored or co-authored more than 100 research papers in international journals and conference proceedings. His current research interests include evolutionary computation algorithms, swarm intelligence algorithms, and their applications in real-world problems, and in environments of cloud computing and big data.

Dr. Zhan was a recipient of the China Computer Federation (CCF) Outstanding Dissertation and the IEEE Computational Intelligence Society (CIS) Outstanding Dissertation for his Ph.D. dissertation. He was a recipient of the Wu Wen Jun Artificial Intelligence Excellent Youth from the Chinese Association for Artificial Intelligence in 2017, the Outstanding Youth Science Foundation from the National Natural Science Foundations of China (NSFC) in 2018, and the First Grade Award in Natural Science from Guangdong Province in 2018. He is listed as one of the Most Cited Chinese Researchers in Computer Science. He is currently an Associate Editor of the *Neurocomputing*.

Yue-Jiao Gong (S'10–M'15) received the B.S. and Ph.D. degrees in computer science from Sun Yat-sen University, Guangzhou, China, in 2010 and 2014, respectively.

From 2015 to 2016, she was a Post-Doctoral Research Fellow with the Department of Computer and Information Science, University of Macau, Macau, China. She is currently a Professor with the School of Computer Science and Engineering, South China University of Technology, Guangzhou. She has authored or co-authored more than 70 papers, including more than 20 IEEE Transactions papers, in her research area. Her current research interests include evolutionary computation, swarm intelligence, and their applications to intelligent transportation scheduling.

Xiangrong Tong received the Ph.D. degree from the School of Computer and Information Technology, Beijing Jiaotong University, Beijing, China, in 2010.

He is currently a Full Professor with Yantai University, Yantai, China. He has presided and joined three national projects and three provincial projects. He has authored or co-authored more than 50 papers in well-known journals and conferences. His current research interests include computer science, intelligent information processing, and social networks.