Repeated Overlapping Coalition Game Model for Mobile Crowd Sensing Mechanism

Sungwook Kim[1]*
[1] Department of Computer Science, Sogang University,
35 Baekbeom-ro (Sinsu-dong), Mapo-gu, Seoul, 121-742, South Korea
* Corresponding author: Sungwook Kim
[e-mail:swkim01@sogang.ac.kr]

Received February 6, 2017; revised March 24, 2017; revised April 6, 2017; accepted April 27, 2017; published July 31, 2017

Abstract

With the fast increasing popularity of mobile services, ubiquitous mobile devices with enhanced sensing capabilities collect and share local information towards a common goal. The recent Mobile Crowd Sensing (MCS) paradigm enables a broad range of mobile applications and undoubtedly revolutionizes many sectors of our life. A critical challenge for the MCS paradigm is to induce mobile devices to be workers providing sensing services. In this study, we examine the problem of sensing task assignment to maximize the overall performance in MCS system while ensuring reciprocal advantages among mobile devices. Based on the overlapping coalition game model, we propose a novel workload determination scheme for each individual device. The proposed scheme can effectively decompose the complex optimization problem and obtains an effective solution using the interactive learning process. Finally, we have conducted extensive simulations, and the results demonstrate that the proposed scheme achieves a fair tradeoff solution between the MCS performance and the profit of individual devices.

Keywords: Mobile crowd sensing; Task assignment problem; Overlapping coalition game; Reciprocal advantage; Interactive learning process.

https://doi.org/10.3837/tiis.2017.07.006 **ISSN : 1976-7277**

1. Introduction

With the development of Internet of Things (IoT) and embedded technology, the remote intelligent monitoring system will be applied in more comprehensive scopes. Therefore, ubiquity of internet-connected portable devices is enabling a new class of applications to perform sensing tasks in the real world. Among mobile devices, smartphones have evolved as key electronic devices for communications, computing, and entertainment, and have become an important part of people's daily lives. Most of current mobile phones are equipped with a rich set of embedded sensors, which can also be connected to a mobile phone via its Bluetooth interface. These sensors can enable attractive sensing applications in various domains such as environmental monitoring, social network, healthcare, transportation, and safety [1]-[4].

Mobile Crowd Sensing (MCS) refers to the technology that uses mobile devices, i.e., smartphones, to collect and analyze the information of people and surrounding environments [2]. Based on this information, we can analyze statistical characteristics of group behaviors, reveal hidden information of social activity patterns, and finally provide useful information and services to end users. By involving anyone in the process of sensing, MCS greatly extends the service of IoT and builds a new generation of intelligent networks that interconnect things-things, things-people and people-people. Therefore, to provide a new way of perceiving the world, MCS has a wide range of potential applications [2].

To effectively operate the MCS system, we collect sensing data from multiple smartphones to maximize the utility of sensed information; different smartphones are intrinsically different in terms of the quality of their embedded sensors. Generally, it is desirable to obtain more sensing data from high quality sensors. However, performing a sensing task consumes precious resources on smartphones, such as energy, computing, and cellular bandwidth. Therefore, one of the central problems for the MCS system is the allocation of sensing-workload among smartphones in order to support various applications. It has a great impact on the overall MCS system performance [5]-[6].

In the early research of MCS system, smartphones are assumed as volunteers for the MCS system. However, in practice, smartphone users are always selfish individuals that will not contribute their resources without getting paid. Therefore, the traditional volunteer models may not be suitable for the real-world MCS operation [6]. This situation can be seen a game theory problem. Game theory is a decision-making process between independent decision-making players as they attempt to reach a joint decision that is acceptable to all participants. In general, players consistently pursue their own objectives and try to maximize the expected value of their own payoffs, which is measured in some utility scale. In the traditional game theory, a solution concept is a rule that defines what it means for a decision vector to be acceptable to all players in the light of the conflict/cooperation environment [7].

Coalitional games are very influential game models for the multi-agent system research community due to their ability to capture the cooperative behaviors among agents [8]. With respect to this activity, there are many approaches that try to generate the optimal coalition structure which maximizes the payoffs of all agents. However, these approaches are all based on the assumption that each agent must belong to one coalition. This means that even if an agent has excessive resources to participate other coalitions, it is still not allowed to take advantages of the remaining resources [8]. In 2010, Chalkiadakis et al. introduced the new concept of Overlapping Coalition Formation (OCF) game [9]. In OCF games, individual agents can cooperate with each other to form coalitions, and the coalitions can be overlapping.

Typically, smartphones consume their own resources to accomplish the sensing task. Therefore, the MCS system should purchase sensing services from smartphones to compensate their resource consumptions. Without compensation of a plenty of smartphones, the MCS system is not able to collect enough information to accomplish its task. In this study, we focus on the multi-workload allocation problem among smartphones. To attract sufficient participations, paying price influences the willingness of smartphones to serve. However, it is not easy to characterize behaviors of smartphone users who may decide their actions to serve. In particular, as the number of smartphones can be huge, it is difficult to apply conventional optimization methods. They pose a heavy computation burden and implementation overheads.

Motivated by the above discussion, we propose a new MCS control scheme based on the repeated OCF game model. To develop a novel multi-workload allocation algorithm, we decomposes the complex optimization problem into several sub-problems, and each sub-problem is solved through an interactive procedure imitating the negotiation process. It is practical and suitable for real implementation. Based on the iterative feedback mechanism, the proposed scheme dynamically adjusts the price for each sensing task, and smartphone users individually respond to such price setting in order to optimize their payoffs. Under the real-world MCS environments, the central server and smartphone users are mutually dependent on each other to maximize their profits while flexibly adapting the current system situations.

The major contributions of our proposed scheme are: i) the adjustable dynamics considering the current MCS system environments, ii) the ability to maximize the total system performance by incorporating the OCF game methodology, iii) the ability to achieve the socially balanced outcome while ensuring individual rationality, and iv) practical approach to effectively reach a desirable solution. Especially, the important novelties of our proposed scheme are obtained from the key principles of practical game approach. To the best of our knowledge, relatively little research has been done on this issue over the years.

The rest of this paper is organized as follows. In Section II, we review the related work. In Section III, we familiarize the reader with the basics of OCF game model, and explain in detail the developed MCS scheme based on the feedback based iterative OCF game. We present experimental results in Section IV and compare the performance to other existing schemes [5],[6],[10]. Finally, we give our conclusion and future work in Section V.

2. Related Work

Over the years, a lot of state-of-the-art research work on the MCS system operation has been conducted. Some MCS schemes [11]-[12] were developed based on the Stackelberg game model. The scheme in [11] proposed two types of incentive mechanisms for the MCS system in the perspectives of the agent platform and mobile users, respectively. First, the platform centric mechanism was assumed that the agent platform had the absolute control over the total payment to users who can only adjust their strategies to comply. Second, the user-centric incentive mechanism utilized an auction-based algorithm and owned benefits such as truthfulness [11]. The scheme in [12] classified the MCS system into two classes: data acquisition and distributed computing. The data acquisition served the purpose of collecting data for building up a database, and the distributed computing utilized computation power to solve problem that could be expensive for a single device. In addition, the contract theory was applied in the distributed computing scenario where the complete information and incomplete information settings were considered [12]. Even though the Stackelberg game based MCS schemes have merits, they needed the platform to know the information of users in advance, which was too strong in the practical system.

The scheme in [13] proposed an online incentive mechanism for the scenario where workers arrived one by one, which was in contrast to some mechanisms assuming all of workers reported their profiles to the agent platform in advance. In this scheme, the MCS mechanism was modeled as an online auction process, where mobile users submitted their private information to the platform over time and a subset of users were selected before a specified deadline. To shorten the crowd response time, the scheme in [14] recruited workers in advance and held idle for a small amount of expense called retainer. The reserved workers would respond quickly when tasks were assigned. Based on the retainer model, the scheme in [14] proposed a combinatorial allocation and pricing mechanism for crowdsourcing tasks with time constraints. The schemes in [13]-[14] were also developed to obtain an efficient solution for the MCS system operations. However, they did not consider the reliability of the submitted data. Therefore, they provided lower efficiency of funding utilization. The volunteer's dilemma is an N-person public good game in which a public good is produced if and only if at least one player volunteers to pay a cost. The basic model of volunteer's dilemma can be applied to many cases in the social sciences; it also can be applied to the design of MCS control schemes [7],[15]-[16].

The *Load Balanced Mobile Crowd Sensing* (*LBMCS*) scheme in [5] considered two important objectives, i) load balancing, and ii) sensing data utility maximization. However, there is an intrinsic tradeoff between load balance and utility maximization. To strike a good balance between these conflicting objectives, the *LBMCS* scheme was designed as a Nash bargaining game model. According to the bargaining process, this scheme can achieve a fair tradeoff between workload balance and data utility maximization in a distributed manner [5].

The *Energy Aware Mobile Crowd Sensing* (*EAMCS*) scheme in [10] was a new participant sampling behavior model to quantify and explicitly build up the relationship between their remaining energy level and the willingness for participation. In particular, this scheme introduced a new concept of 'QoI satisfaction ratio' to quantify the degree of how collected sensory data can satisfy multi-dimensional QoI requirements of tasks in terms of data granularity and quantity. Based on this mechanism, the *EAMCS* scheme calculated the rejection probability that represented the chance of a participant to reject the sensing task if the recommended number of data samples from the server cloud exceeded its sensing capabilities [10].

The *Overlapping Coalition Game based Collaborative Sensing* (*OCGCS*) scheme in [6] was developed as an incentive mechanism in which the users can get satisfying rewards from the platform by efficiently allocating their resources to achieve a relatively high social welfare. Specifically, to solve the resource allocation problem in the incentive mechanism, this scheme considered a cooperative game model with overlapping coalitions, in which the smartphone users can self-organize into the overlapping coalitions for different sensing tasks. Finally, the *OCGCS* scheme proposed a distributed algorithm that converges to a stable outcome in which no user had the motivation to change its current resource allocation so as to increase its individual payoff [6].

All the earlier work in [5]-[6],[10] has attracted a lot of attention and introduced unique challenges to efficiently handle the MCS system. However, there are several disadvantages. First, these existing schemes rely on the impractical assumption for real MCS operations. Control algorithms based on the inapplicable presumption can cause potential erroneous decisions. Second, these schemes cause the extra control overhead, which exhausts the system resources and needs intractable computation. Third, these schemes operate the MCS system by some fixed system parameters. Under dynamic system environments, it is an inappropriate approach to operate real world network MCS systems. In this study, we demonstrate that our proposed scheme significantly outperform these schemes in [5]-[6],[10] through extensive simulation and the analysis is given in Section IV.

3. Overlapping Coalition based MCS Algorithm

In this section, we present our OCF game model, which employs an interactive feedback approach. And then, we explain in detail about the proposed MCS algorithm through the OCF concept. Finally, the proposed scheme is described strategically in the eight-step procedures.

A. Overlapping Coalition Formation Game Model

To model cooperative games with overlapping coalitions, it is assumed that players possess a certain amount of resources which they can distribute among the coalitions they join. If overlapping coalitions are allowed, players selectively participate some coalitions, and players' contribution to a coalition is given by the fraction of their resources that they allocate to it. In the traditional non-overlapping coalition formation game, a coalition is a subset of players, and a game is defined by its characteristic function $v : 2^N \rightarrow \mathbb{R}$ with player set $\boldsymbol{N} = \{1, \ldots, n\}$, representing the maximum total payoff that a coalition can get [9].

In the OCF game, a coalition k is given by a vector $\boldsymbol{r}^k = \left(r_1^k, \ldots, r_n^k\right)$, where r_i^k is the fraction of agent i's resources contributed to the coalition k; $r_i^k = 0$ means that the player i is not a member of the coalition k. The *support* of coalition k, denoted by $\mathrm{supp}(\boldsymbol{r}^k)$, is given by $\mathrm{supp}\left(\boldsymbol{r}^k\right) = \left\{i \in N \mid r_i^k \neq 0\right\}$. The OCF game is given by a characteristic function $v(k)$: $\left[0, r_{i,1\leq i\leq n}^k\right]^n \rightarrow \mathfrak{R}$, where $v(0^n) = 0$ and $\mathfrak{R}$ represents the set of real numbers. Function $v(k)$ is *monotone*, and maps each contribution r_i^k in $\boldsymbol{r}^k$ to the corresponding payoff [9].

In this study, we develop a new OCF game model ($\mathbb{G}$) for MCS system. To model strategic MCS situations involving interactive process, we assume that players seek to choose their strategy based on the reciprocal relationship. In our game model, each coalition represents individual sensing task in the MCS system.

Definition 1. Our OCF game model constitutes a 8-tuple $\mathbb{G} = (\boldsymbol{N} \cup \{0\}, m, \boldsymbol{r}^{(\cdot)}, v(\cdot), \boldsymbol{\mathcal{T}}, \mathbb{A}, \boldsymbol{S}_{i,1\le i\le n}, \boldsymbol{U}_{i,1\le i\le n})$, where

(i) $\boldsymbol{N} \cup \{0\}$ is a set of game players; $\boldsymbol{N} = \{1, \dots, n\}$ is the set of smartphone users and 0 represents the central server of MCS system,

(ii) m, i.e., $1 \le k \le m$, is the number of sensing tasks in our MCS system,

(iii) $\boldsymbol{r}^{(\cdot)}$ is a vector to represent the contributed resources from players in $\boldsymbol{N}$,

(iv) $v(\cdot)$ represents a satisfaction level of each task. It is a *monotone* function to evaluate each task payoff. Therefore, v function satisfies $v(\boldsymbol{r}^k) \ge v(\boldsymbol{r}'^k)$ for any $\boldsymbol{r}^k$, $\boldsymbol{r}'^k$ such that $r_i^k \ge r_i'^k$ for all $i \in \boldsymbol{N}$.

(v) $\boldsymbol{\mathcal{T}} = \{\mathfrak{T}^1, \dots, \mathfrak{T}^k, \dots, \mathfrak{T}^m\}$ is a set of each task's thresholds, $\mathfrak{T} > 0$. If $\boldsymbol{r}^k < \mathfrak{T}^k$, $v(\boldsymbol{r}^k) = 0$. This means that players must allocate resource at least the $\mathfrak{T}^k$ amount to complete the task k,

(vi) $\mathbb{A} = \{\mathcal{A}_1, \dots, \mathcal{A}_i, \dots, \mathcal{A}_n\}$ is a set of available resources for players in $\boldsymbol{N}$. $\mathcal{A}_i$ represents the total resource amount of the player $i \in \boldsymbol{N}$. For the sake of simplicity, we assume one type of resource, e.g., sensing capacity, that is needed for all tasks,

(vii) $\boldsymbol{S}_i = \{s_i^1, \dots, s_i^k, \dots, s_i^m\}$ is a nonempty finite set of all pure strategies of the player $i \in \boldsymbol{N} \cup \{0\}$. In particular, if $i = 0$, s_i^k represents the central server's price strategy for the task *k*. Otherwise, if $i \in \boldsymbol{N}$, s_i^k represents the smartphone *i*'s contribution to the task *k*.

(viii) $\boldsymbol{U}_i = \{u_i^1, \dots, u_i^k, \dots, u_i^m\}$ is the utility set of the player $i \in \boldsymbol{N} \cup \{0\}$. Therefore, u_i^k represents the player *i*'s payoff for the task *k*. If $i = 0$, $\boldsymbol{U}_0$ represents a satisfaction level of the central server: $(s_0^1 \times s_0^2 \times \cdots \times s_0^m) \to \boldsymbol{U}_0 = \sum_{k=1}^m v(\boldsymbol{r}^k)$. Otherwise, if $i \in \boldsymbol{N}$, $\boldsymbol{U}_i$ represents a satisfaction level of player *i*, and it is decided according to the set of player *i*'s strategies : $(s_i^1 \times s_i^2 \times \cdots \times s_i^m) \to \boldsymbol{U}_i = \sum_{k=1}^m u_i^k$.

In this study, our OCF game model describes a scenario where players can split their resources to work different m tasks. Each task has its own resource requirement $\mathfrak{T}$ and a utility function $v(\cdot)$. If the $\mathrm{supp}(\boldsymbol{r}^k)$, which is the total sum of contribution resources from players that work on the task *k*, is higher than the $\mathfrak{T}^k$, the task *k* has sufficient resources to be completed, and the central server can obtain the outcome of $v(\boldsymbol{r}^k)$. Otherwise, the payoff from the task *k* is 0. Without loss of generality, we assume that each player chooses independently to work on each individual task while not preventing another players from choosing any tasks as well. For real-world MCS operations, this assumption is generally holds.

B. Interactive Mobile Crowd Sensing Algorithm

In this study, we assume a formal model for the MCS system. We consider a multitask-oriented central server, and n smartphone users randomly spread over the MCS region. Each smartphone is embedded with various sensors, and different sensing-works are assumed to be independent of each other, both temporarily and spatially. The central server publicizes m sensing tasks, and gathers the sensing information from the sensory data contributors, i.e., smartphones, in the MCS region. The total sensing resource of each smartphone is limited to $\mathcal{A}$, and smartphone users can distribute their resources freely among m tasks. Each task is associated with a set of thresholds ($\boldsymbol{\mathcal{T}}$). Therefore, for each task completion, smartphone users need to upload at least the minimum amount $\mathfrak{T}$ from their devices.

To evaluate each task's sensing performance, the central server has its own utility function, which is a function of the participating smartphones and their corresponding contributions. To recruit smartphones for collecting sensory data, a proper payment mechanism should be employed to model the interactions between smartphone users and the central server. By using the reciprocal relationship between benefit and cost, the central server's payoff corresponds to the received outcome minus the incurred cost. In the central server, the utility function for the task k is defined as follows;

$$\boldsymbol{U}_0\left(\boldsymbol{s}_0^{k,1\le k\le m}\right) = \sum_{k=1}^{m} v\left(\boldsymbol{r}^k\right) = \sum_{k=1}^{m}\left(\zeta^k - \mathcal{C}_0\left(\boldsymbol{r}^k\right)\right) \tag{1}$$

$$\text{s.t.}, \boldsymbol{r}^k = \sum_{l=1}^{n} r_l^k \text{ and } \mathcal{C}_0\left(\boldsymbol{r}^k\right) = \mathcal{P}^k \times \boldsymbol{r}^k$$

where $\mathcal{P}^k$ and ζ^k are the *unit_price* for resource and the obtained outcome from the task k completion, respectively. If $\boldsymbol{r}^k < \mathfrak{T}^k$, the ζ^kvalue is 0. According to the equation (1), the $\boldsymbol{U}_0(\cdot)$ function has a nice interpretation: the net gain of central server's utility decreases proportionately by the sensing payment. In contrast to the central server, the net gain of smartphones increases proportionately by the sensing payment. Therefore, the individual utility function for the player $i \in \boldsymbol{N}$ ($\boldsymbol{U}_i(\boldsymbol{S}_i)$) is defined as follows;

$$\boldsymbol{U}_i(\boldsymbol{S}_i) = \sum_{k=1}^{m} u_i^k = \sum_{k=1}^{m}\left(\left\{\frac{\mathcal{P}^k}{1+\exp\left(-\zeta^k \times \frac{s_i^k}{\mathcal{A}_i}\right)}\right\} - \mathcal{C}_i\left(k, s_i^k\right)\right) \tag{2}$$

$$\text{s.t.}, \mathcal{C}_i\left(k, s_i^k\right) = \eta_i^k \times \left(\exp\left(\frac{s_i^k}{\mathcal{A}_i}\right) - 1\right) \text{ and } \sum_{k=1}^{m} s_i^k \le \mathcal{A}_i$$

where $\mathcal{C}_i\left(k, s_i^k\right)$ and η_i^k are the player i's cost function with the strategy s_i^k, and a cost control parameter for the task k, respectively. Traditionally, the exponential function is widely used in literature to model the consuming cost with assigned resource [7].

The goal of each player in $\boldsymbol{N}$ is to maximize all their profits; they select their strategies from a selfish motive. If some tasks need high costs for sensing works, players in $\boldsymbol{N}$ do not contribute their resources for these non-profitable tasks, and these tasks can not be completed. At this time, all players pay a penalty cost ($\mathfrak{F}(\cdot)$), which is a great damage for players in $\boldsymbol{N}$. Therefore, each individual player prefers that all tasks are completed, but also, they prefer that other players contribute their resources to non-profitable tasks. Due to this reason, each player faces the decision of either making a small sacrifice from which all will benefit, or freeriding.

Usually, social dilemmas are situations in which the optimal decision of an individual contrasts with the optimal decision for the group. In game theory, this usually means games in which a dominant strategy leads to a Pareto inefficient equilibrium; the prisoner's dilemma is probably the most famous example [15]. In 1985, M. Diekmann first proposed the volunteer's dilemma game in the social sciences [16]. In this game, each individual player prefers to avoid the cost of volunteering and exploit the benefit of the public good, but someone must volunteer and pay the cost of producing the good; if nobody volunteers, the cost paid is greater than the cost of volunteering [16].

In the traditional volunteer's dilemma game, a public good is produced if and only if at least one player volunteers to pay a cost. The basic model of N-person volunteer's dilemma is the following: each of N individuals can choose to volunteer (Volunteer) or not (Ignore). A public good is produced if and only if at least one individual volunteers [15]. Volunteering has a cost $c > 0$. Therefore, the individuals that volunteer have a payoff $1 - c$ and the ones that do not have a payoff 1. If nobody volunteers, the public good is not produced; everybody pays a cost $a > c$ (i.e., payoff $1 - a$). The fitness of the pure strategy Volunteer (W_V) is $W_V = 1 - c$ and the fitness of the pure strategy Ignore (W_I) is $W_{\mathrm{I}} = \left(\gamma^{N-1} \times (1-a)\right) + (1 - \gamma^{N-1})$ where γ is the probability of ignoring (not volunteering). The fitness of the mixed strategy is $W_{mix} = (\gamma \times W_{\mathrm{I}}) + ((1-\gamma) \times W_V)$. The mixed-strategy equilibrium (γ_{eq}) can be found by equating the fitness of the two pure strategies; $\gamma_{\mathrm{eq}} = (c/a)^{1/(N-1)}$. The volunteer's dilemma can be applied to many cases in the social sciences [15].

For effective MCS operations, the proposed scheme applies the concept of volunteer's dilemma to our OCF game model. When the amount of contribution for a specific task is less than its corresponding threshold, players make two decisions; i) whether to contribute their resources or not, ii) if they contribute, how much resource would be contributed. Under dynamically changing MCS environments, these decisions should be made to effectively adapt to the current MCS condition. In order to adaptively implement this decision process, players can learn how to perform well by interacting with other players and dynamically adjust their decisions.

To decide adaptively above two questions, we present a novel dynamic learning approach. While time is ticking away, players obtain their payoffs ($\boldsymbol{U}_i(\cdot)$) as a consequence of their decisions. Based on this information, each player individually decides his strategy selection probability. If the task k has failed to be completed, the central server announces this situation. At this time, the player i in $\boldsymbol{N}$ selects his strategy (s_i^k); s_i^k can be defined as multiple amount levels of contributing resource, i.e., $s_i^k = \{s_i^{k(0)}, \ldots s_i^{k(\alpha)}\}$ where $s_i^{k(0)}$ (*or* $s_i^{k(\alpha)}$) means that the player i does not contribute his resource (*or* contributes α more units of his resource). For the $(t+1)^{\text{th}}$ game round time, the selection probability for the $s_i^{k(d),1 \le d \le \alpha}$ strategy $\left(p^{t+1}\left(s_i^{k(d)}\right)\right)$ is estimated based on the propensity of $s_i^{k(d)}$

$\left(\mathcal{G}_i^{t+1}\left(s_i^{k(d)}\right)\right)$; it is described in the following;

$$p^{t+1}\left(s_i^{k(d)}\right) = e^{\frac{\mathcal{G}_i^{t+1}\left(s_i^{k(d)}\right)}{\psi}} \Bigg/ \sum_{h=1}^{\alpha}\left(e^{\frac{\mathcal{G}_i^{t+1}\left(s_i^{k(h)}\right)}{\psi}}\right) \tag{3}$$

$$\text{s.t.,}\begin{cases} \mathcal{G}_i^{t+1}\left(s_i^{k(d)}\right) = \left\{\left((1-\xi)\times\mathcal{G}_i^{t}\left(s_i^{k(d)}\right)\right) + \left(\frac{\delta^k}{\alpha}\times\left[\frac{\boldsymbol{U}_i^t(\boldsymbol{S}_i)-\boldsymbol{U}_i^{t-1}(\boldsymbol{S}_i)}{\boldsymbol{U}_i^{t-1}(\boldsymbol{S}_i)}\right]\right)\right\}, \\ \qquad \text{if } s_i^{k(d)} \in \boldsymbol{S}_i \text{ is selected at the } t \text{ time and } \delta^k = \sum_{l=1}^{n}\left(\frac{\zeta^k}{\zeta^l}\right) \\ \mathcal{G}_i^{t+1}\left(s_i^{k(f)}\right) = \left\{(1-\xi)\times\mathcal{G}_i^{t}\left(s_i^{k(f)}\right)\right\}, \quad \text{otherwise,} \ \ \text{s.t.,} s_i^{k(f)} \neq s_i^{k(d)} \end{cases}$$

where ψ is a positive Boltzmann cooling parameter and ξ is the forgetting factor; it is essentially required when players face a game when the propensity adaptively changes over time. δ^k is the learning rate for the task k toward maximizing the utility function.

Considering the equations (1)-(3), we can set the maximization problem. From the viewpoint of central server, the main interest is to maximize its total MCS revenue according to the dynamically adjusting $\mathcal{P}^k$ for each task. From the viewpoint of self-interesting individual smartphones, the major goal is to maximize their own payoff by selecting their strategies s_i^k. That is formally formulated like as

$$\max_{\mathcal{P}^k}\{\boldsymbol{U}_0(\boldsymbol{S}_0)|\mathcal{P}^k \in \boldsymbol{S}_0, 1\le k\le m\} \quad \text{and} \quad \max_{s_i^k}\left\{[\boldsymbol{U}_i(\boldsymbol{S}_i)-\mathcal{F}]\middle|s_i^k \in \boldsymbol{S}_i, 1\le i\le n\right\} \tag{4}$$

$$\text{s.t.,}\mathcal{F} = \sum_{k=1}^{m}\mathfrak{F}(k) \ \text{ and } \ \mathcal{C}_i(k,\mathcal{A}_i) \le \mathfrak{F}(k)$$

where $\mathfrak{F}(k)$ is the damage cost for all smartphone users if the task k has not been completed. Since the utilities of players are obtained from multiple tasks, players dynamically select different strategies for different tasks. For each task, smartphones invest their resources, and a central server assigns different *unit_prices*, individually. Therefore, the players' interactive actions among various tasks is important in maximizing the players' own income while improving the sensing performance of the MCS platform.

In our OCF game model, game players are selfish but cooperate with each other to effectively form coalitions to accomplish the tasks. At each round of OCF game operations, players periodically observe the current their payoffs, which are obtained by the coordination of other players' behaviors. Therefore, individual players can periodically observe the behaviors of other players in a roundabout way, and dynamically adjust their strategies. In our scenario, coalitions are tasks that consists of multiple players in $\boldsymbol{N}$. They are willing to participate in tasks while automatically forming coalitions; they can be overlapping, and their contributions are rewarded by the central server. According to the *individual rationality*, they do not invest their resources to a specific task unless their work produces any profits. From the point of view of central server, a cost-loss can be caused if the participation level of players in $\boldsymbol{N}$ is less than some tasks' thresholds. Or a profit-loss can be caused when the participation level of players in $\boldsymbol{N}$ is much higher than some tasks' thresholds. Therefore, to approximate an optimized solution, the central server adaptively adjusts the sensing prices through any possible set of central server's price strategies to encourage players in participating or withdrawing some task works. During the step-by-step iteration, players individually adjust their strategies by using the dynamics of feedback-based repeated process, and attempt to guarantee the *group rationality*. Therefore, under widely diverse MCS situations, the main advantage of our proposed approach is a real-world practicality.

C. The Main Steps of Proposed Algorithm

With the development of mobile sensing and mobile Internet techniques, a new MCS paradigm has become popular while enabling a broad range of mobile applications. A critical challenge for the MCS paradigm is to induce smartphone users to be workers providing sensing services. While some control mechanisms for general-purpose crowdsourcing have been proposed, it is still an open issue as to how to incorporate the practical algorithms into the real-world MCS system. In this study, we propose a novel MCS control scheme based on the OCF game model. The proposed approach is a natural extension of traditional OCF game with adopt the concept of volunteer's dilemma. Considering the step-by-step interactive feedback mechanism, the developed algorithm is designed as a repeated OCF game. The proposed algorithm is described by the following major steps, Pseudo code and a Flow diagram.

Step 1: At the initial time, the strategy selection probability $s_i^{k(\cdot)}$ of the player $i \in \boldsymbol{N}$ is equally distributed. This starting guess guarantees that each $s_i^{k(\cdot)}$strategy is selected randomly at the beginning of the game.

Step 2: Control parameters n, m, η, ψ and ξ are given from the simulation scenario (refer to the **Table 1**).

Step 3: During our iterative OCF game process, the central server decides price strategies $\left(\mathcal{P}^{(\cdot)}\right)$ to maximize its payoff according to (1) and (4).

Step 4: Based on the central server's strategies, rational smartphone users individually select their strategies to maximize their own payoff. Using the equation (2) and (4), these decisions are made in an entirely distributed manner.

Step 5: If some tasks have not been completed, the central server dynamically adjust the price strategies again in the same manner in **Step 3**, and all player in $\boldsymbol{N}$ re-select their strategies $\left(s_i^{k(\cdot)}\right)$ based on the equation (3).

Step 6: The propensity of $s_i^{k(\cdot)}$ is dynamically adjusted based on the interactive learning mechanism.

Step 7: During the step-by-step iteration, players individually adjust their strategies by using the dynamics of feedback-based repeated process.

Step 8: Under the real-world MCS environments, the central server and smartphone users are mutually dependent on each other to maximize their profits, and they constantly are self-monitoring the current system conditions; proceeds to Step 3 for the next game iteration.

Init ( )
{ **1:** Control parameter values ($n, m,\ \eta,\ \psi,\ \alpha,\ \boldsymbol{\mathcal{T}},\ \mathbb{A}$ and ξ) are given from the **Table 1** in the Section IV.
2: $s_{i,i\in\boldsymbol{N}}^{k(\cdot)}$ is equally distributed
}

Main_Routine for the Central Server ( )
{ **Start:** *Init* ();
For (; ;) {
3-1: For the current OCF game process, the central server decides $\mathcal{P}^{(\cdot)}$ to maximize its payoff based on the equation (1) and (4).
4-1: Constantly observe the resource contribution from the individual users.
5-1: If $\mathcal{P}^{(\cdot)}$ and users' responses are not changed while completing all tasks,
Then the OCF game process is temporarily stop; Go to step **4-1**.
6-1: If some tasks have not been completed,
Then Go to step **3-1** to adaptively decrease the price $\mathcal{P}^{(\cdot)}$.
Else Go to step **3-1** to adaptively increase the price $\mathcal{P}^{(\cdot)}$. }
}

Main_Routine for Individual Users ( )
{ **Start:** *Init* ();
For (; ;) {
3-2: For the current OCF game process, each users individually select their strategies to maximize their own payoff according to the equation (2) and (4).
4-2: Constantly observe the price strategy ($\mathcal{P}^{(\cdot)}$) from the central server.
5-2: Using (3), the propensity of $s_i^{k(\cdot)}$ is dynamically adjusted.
6-2: If the central server changes the price $\mathcal{P}^{(\cdot)}$,
Then Go to step **3-2** to re-consider the current strategy.
Else Go to step **4-2** for the next game iteration. }
}

Pseudo code. MCS System Control Procedure

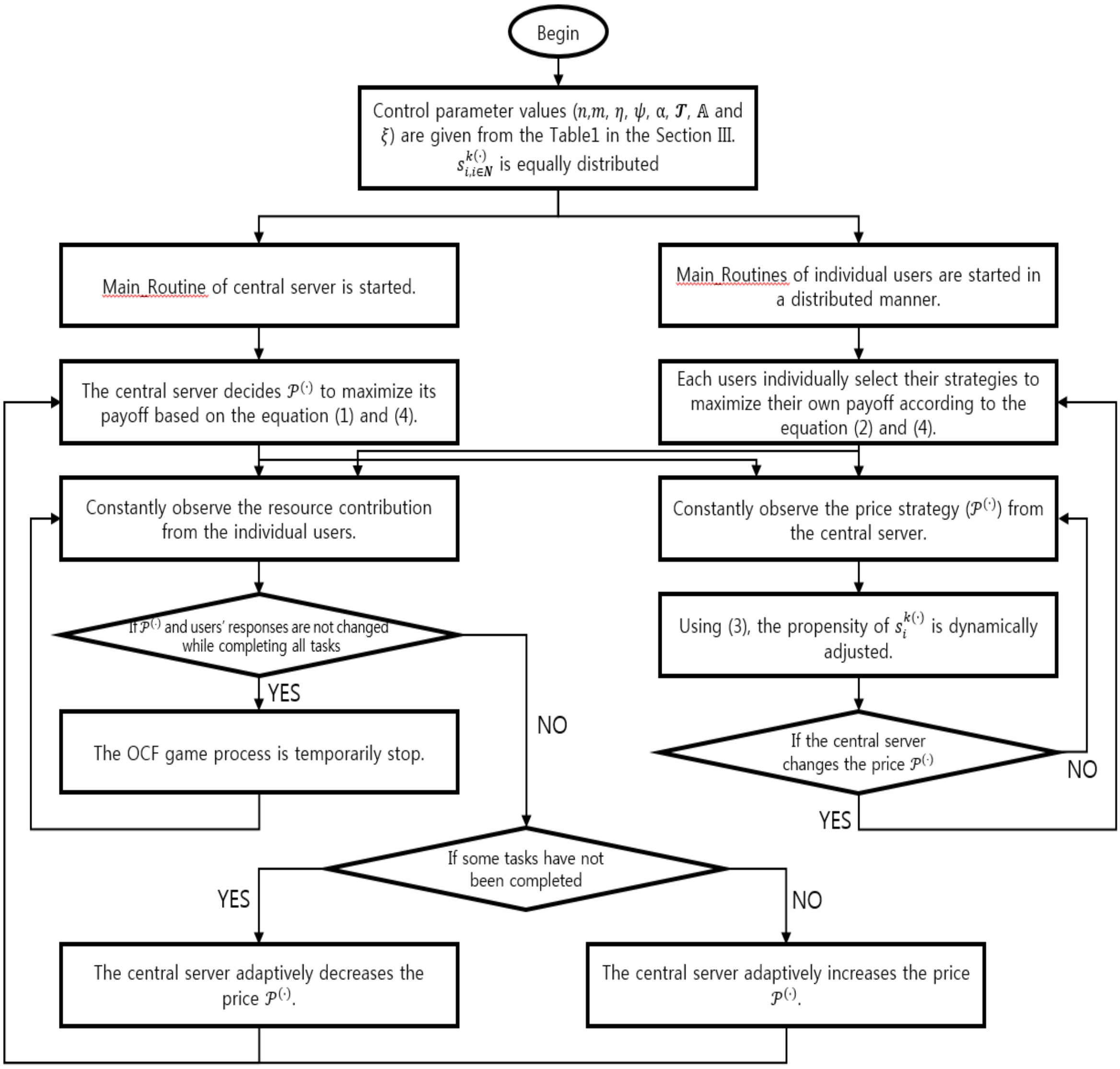


Fig. 1. Flow diagram for the proposed algorithm

4. Performance Evaluation

In this section, we compare the performance of our scheme with other existing schemes [5],[6],[10] and can confirm the performance superiority of the proposed approach by using a simulation model. Our simulation model is a representation of the MCS system environments that includes one central server with multiple sensing tasks, and individual smartphone users. To facilitate the development and implementation of our simulator, Table 1 lists the system control parameters.

Table 1. System parameters used in the simulation experiments

Parameter	Value	Description
n	$1 \leq n \leq 30$	the number of smartphone users (i.e., players in $\boldsymbol{N}$)
m	5	the number of sensing tasks in the central server
η	1	a cost control parameter for each task
ψ	1	a positive Boltzmann cooling parameter
ξ	0.1	a forgetting factor for the effective learning
α	5	the number of more resource units for the non-completed task
$\boldsymbol{\mathcal{T}}$	{5, 10, 15, 20, 25}	a set of each task's thresholds
$\mathbb{A}$	{5, 5, …, 5}	a set of available resources for players in $\boldsymbol{N}$

To ensure the model is sufficiently generic to be valid in a real-world MCS scenario, the assumptions implemented in our simulation model were as follows.

- The simulated system consists of one central server and n number of smartphone users ($1 \leq n \leq 30$) for the MCS system.
- System performance measures are plotted as a function of the offered n numbers.
- All players in $\boldsymbol{N}$ have the same amount of resources.
- The number of tasks (m) in the central server is 5 and the outcome from the task completion is $\zeta^1 = 10$, $\zeta^2 = 30$, $\zeta^3 = 40$, $\zeta^4 = 50$, $\zeta^5 = 100$.
- The set of each task's thresholds ($\boldsymbol{\mathcal{T}}$) is $\{\mathfrak{T}^1 = 5$, $\mathfrak{T}^2 = 10$, $\mathfrak{T}^3 = 15$, $\mathfrak{T}^4 = 20$, $\mathfrak{T}^5 = 25\}$.
- $\boldsymbol{S}_{i,0\leq i\leq n}$ is dynamically decided to maximize their payoffs.
- The cost control parameter (η) of all players in $\boldsymbol{N}$ is the same for each task.
- The amount of unit resource is 1, and multiple unit resources are adaptively distributed for each task.
- System performance measures obtained on the basis of 50 simulation runs.
- For simplicity, we assume the absence of physical obstacles in the experiments.

Performance measures obtained through simulation are normalized payoffs for the central server and smartphone users, task completion probability, and resource utilization under different number of players. In this paper, we compare the performance of the proposed scheme with the existing schemes; the *LBMCS* scheme [5], the *EAMCS* scheme [10] and the *OCGCS* scheme [6]. These existing schemes were also recently developed as effective MCS control algorithms. However, these existing schemes were one-sided protocols and can not adaptively respond the current MCS system conditions. Therefore, they did not provide suitable solutions under different practical constraints.

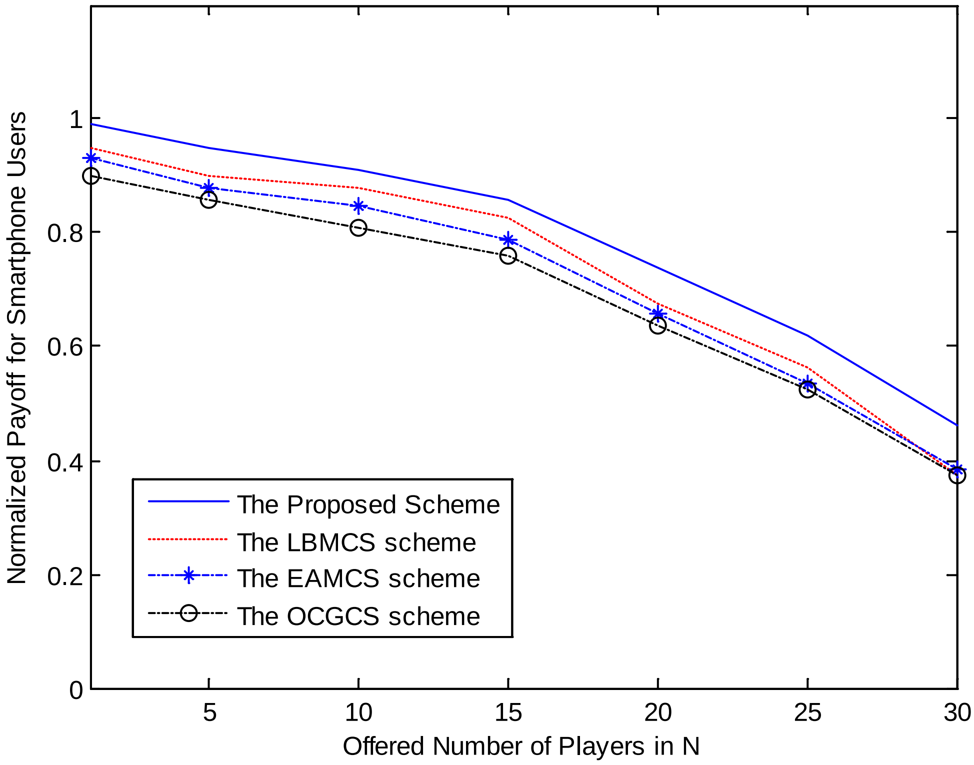


Fig. 2. Normalized Payoffs for Smartphone Users

Fig. 2 shows the performance comparison of each scheme in terms of the smartphone users' normalized payoff on the MCS platform. From the point of view of smartphone users, it is an important performance metric. As the number of users increases, the total amount of available resources also increases, and the competition between users in sensing works is more intense. Therefore, the central server can lower the sensing prices to complete tasks. Due to this reason, the normalized payoff of smartphone users proportionally decreases with increasing the number of users. As shown in **Fig. 2**, we find that all the schemes produce similar performance trends. However, the payoff produced by the proposed scheme is higher than other schemes from few to many smartphone users.

In **Fig. 3**, we depict how the central server's normalized payoff changes over the number of smartphone users. In general, the better central server's payoff gain means that the MCS system can successfully perform the sensing works. We observe that as more and more users are participating in sensing works, the sensing tasks are easily completed with lower sensing prices. This situation can lead to the higher payoff of central server. Due to the inclusion of interactive repeated OCF game approach, the proposed scheme can keep a better central server's performance during the MCS system operations.

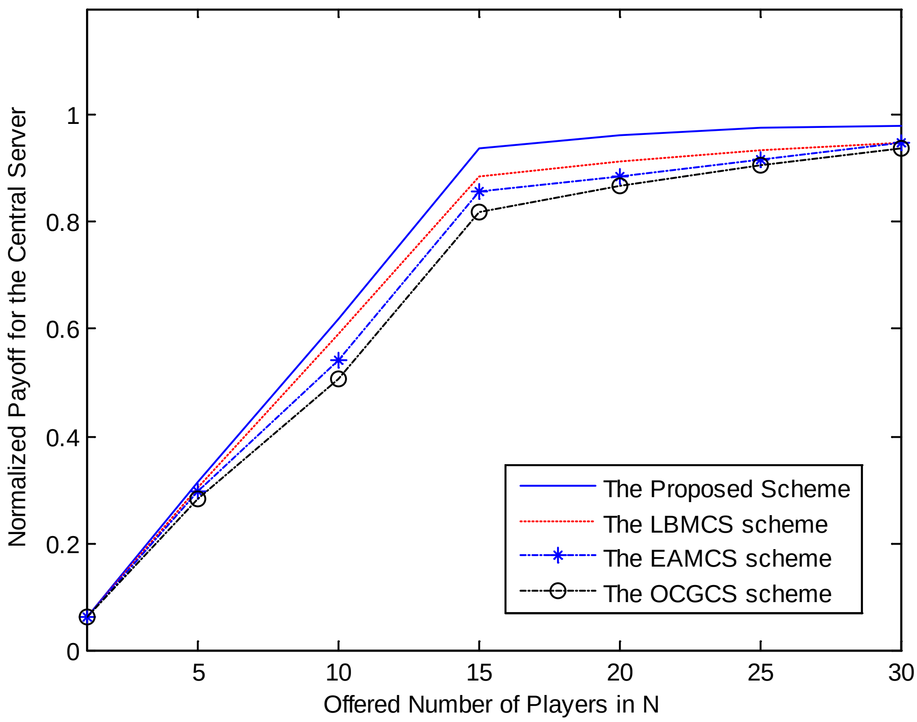


Fig. 3. Normalized Payoffs for the Central Server

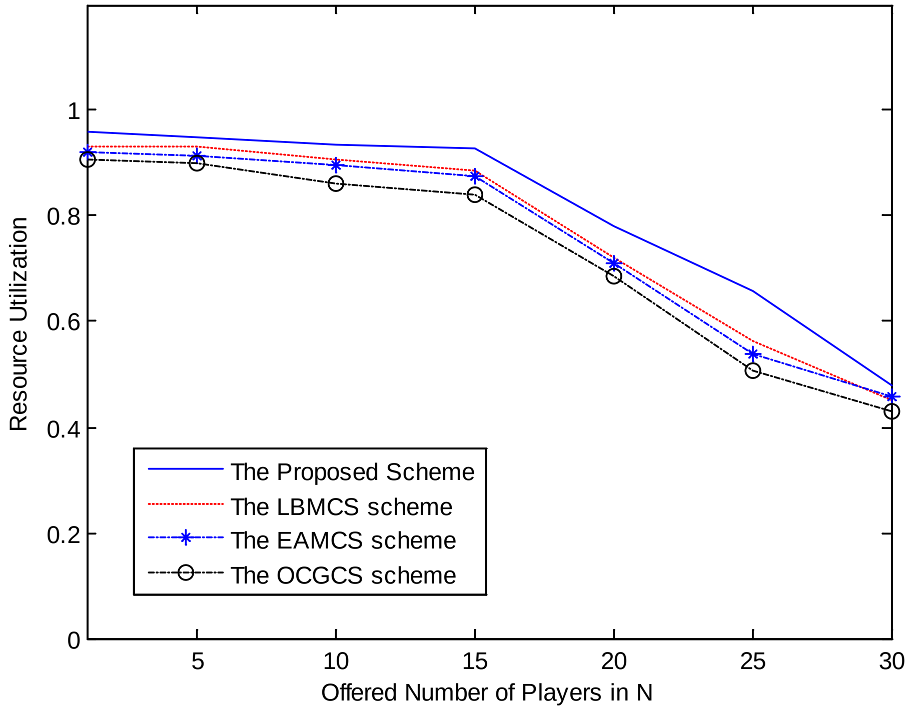


Fig. 4. Resource Utilization

The curves in **Fig. 4** show the sensing resource utilization under different number of smartphone users. In this study, the resource utilization is calculated as the ratio of actually using bandwidth over total available bandwidth. It is observed that when the number of users is low ($n < 15$), the resource utilization is high. However, as the number of users increases, it decreases linearly. This is because with the increasing number of users, the amount of surplus resources also increases. This is intuitively correct. From the simulation results, the main observation is that the proposed scheme can effectively allocate sensing resources for multiple sensing tasks while maintaining a higher resource utilization than other existing schemes.

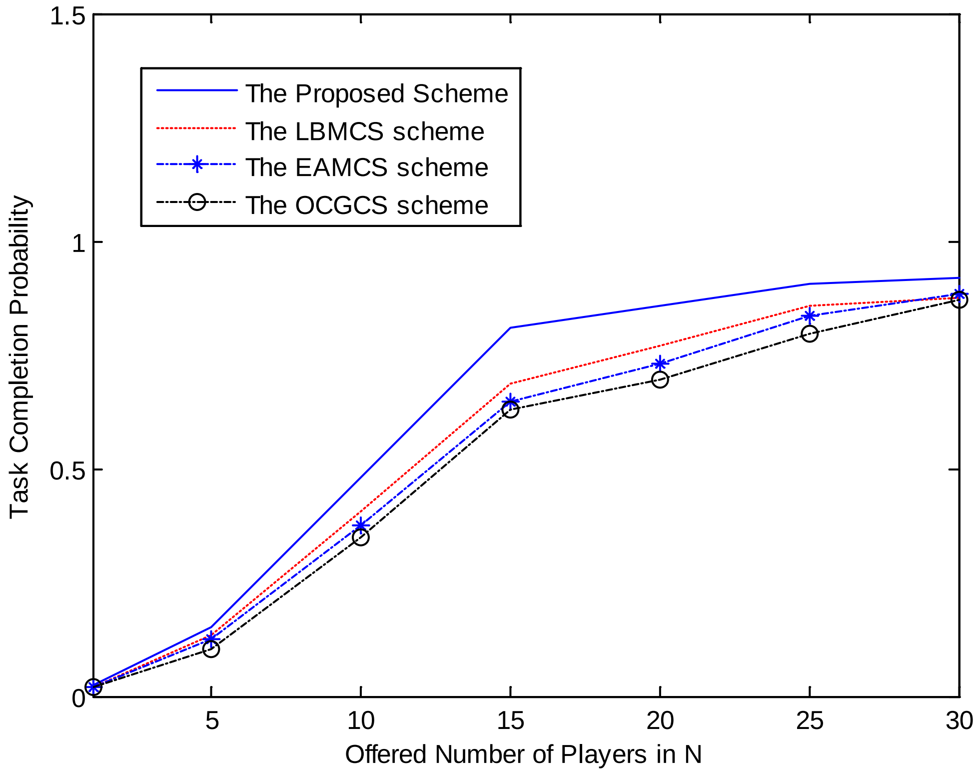


Fig. 5. Task Completion Probability

Fig. 5 indicates the task completion probability in the multiple sensing process. This measure is a key factor to successfully operate the MCS system. The performance trend is similar to the **Fig. 3**. Since the users in our OCF game-based scheme have more opportunities to effectively distribute their resources into sensing tasks, the proposed scheme can maintain the excellent task completion probability than other existing schemes. This feature is a highly desirable property for the multi-task MCS system management. The simulation results shown in **Figs. 2-5** demonstrate the performance comparison of the proposed scheme and other existing schemes [5], [6], [10], and verify that the proposed OCF game-based scheme can provide an attractive MCS system performance.

5. Summary and Conclusions

Nowadays, the exponential growth of smartphones creates a compelling paradigm of MCS. The MCS paradigm provides us various exciting and profitable applications. However, few studies addressed this issue, which significantly affects the functionality of MCS systems. In this study, we propose a novel MCS control scheme based on the repeated OCF game model. The proposed scheme is specifically for the real-world MCS system, where the smartphone users are paid off based on their contributions. Moreover, we have incorporated the concept of volunteer's dilemma into our scheme to maximize the total system performance. Finally, we evaluate the proposed scheme by using simulation model and present extensive experimental results. Compared with the existing schemes, simulation result shows that the proposed scheme helps smartphone users to effectively select actions to achieve the socially balanced outcome while ensuring individual rationality. Future work will be pursued in the following directions. Theoretical analysis needs to be further developed. In addition, we are currently exploring a unified architecture for collecting and processing sensor data from smartphone sensing devices at a societal scale. Furthermore, the proposed OCF game model can be extended toward for other research areas; control decisions in resource management and scheduling, machine cognition, data mining, machine learning, big data analysis and natural computation, etc.

References

[1] Xiang Sheng, Jian Tang, Xuejie Xiao, and Guoliang Xue, "Leveraging GPS-Less Sensing Scheduling for Green Mobile Crowd Sensing," *IEEE Internet of Things Journal, vol.1, no.4*, pp.328-336, 2014. Article (CrossRef Link).

[2] An Jian, Gui Xiaolin, Yang Jianwei, Sun Yu, and He Xin, "Mobile Crowd Sensing for Internet of Things: A Credible Crowdsourcing Model in Mobile-Sense Service," *IEEE BigMM'2015*, pp.92-99, 2015. Article (CrossRef Link).

[3] Ingook Jang, Dohoo Pyeon, Sunwoo Kim, and Hyunsoo Yoon, "A Survey on Communication Protocols for Wireless Sensor Networks," *JCSE, vol.7, no.4*, pp.231-241, 2013. Article (CrossRef Link).

[4] Osman Salem, Yaning Liu, and Ahmed Mehaoua, "Anomaly Detection in Medical Wireless Sensor Networks," *JCSE, vol.7, no.4, pp.272-284*, 2013. Article (CrossRef Link).

[5] Juan Li, Yanmin Zhu, and Jiadi Yu, "Load balance vs utility maximization in mobile crowd sensing: A distributed approach," *IEEE GLOBECOM'2014*, pp.265-270, 2014. Article (CrossRef Link).

[6] Boya Di, Tianyu Wang, Lingyang Song, and Zhu Han, "Incentive mechanism for collaborative smartphone sensing using overlapping coalition formation games," in *Proc. of IEEE GLOBECOM'2013*, pp.1705-1710, 2013. Article (CrossRef Link).

[7] Sungwook Kim, *Game Theory Applications in Network Design*, IGI Global, 2014. Article (CrossRef Link).

[8] Y. Zhan, J. Wu, C. Wang, and J. Xie, "On the Complexity and Algorithms of Coalition Structure Generation in Overlapping Coalition Formation Games," in *Proc. of IEEE ICTAI'2012*, pp.868-873, 2012. Article (CrossRef Link).

[9] G. Chalkiadakis, E. Elkind, E. Markakis, M. Polukarov, and N. R. Jennings, "Cooperative games with overlapping coalitions," *Journal of Artificial Intelligence Research*, vol.39, no.1, pp.179-216, 2010. Article (CrossRef Link).

[10] Liu, C.H., Zhang, B., Su, X., Ma, J., Wang, W. and Leung, K.K., "Energy-Aware Participant Selection for Smartphone-Enabled Mobile Crowd Sensing," *has been accepted for inclusion in a future issue of IEEE Systems Journal*. Article (CrossRef Link).

[11] D. Yang, G. Xue, X. Fang, and J. Tang, "Crowdsourcing to smartphones: Incentive mechanism design for mobile phone sensing," *ACM Mobicom*'2012, pp.195-205, 2012. Article (CrossRef Link).

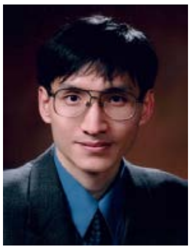
[12] Duan, L., Kubo, T., Sugiyama, K., Huang, J., Hasegawa, T. and Walrand, J., "Incentive mechanisms for smartphone collaboration in data acquisition and distributed computing," in *Proc. of IEEE INFOCOM'2012*, pp.1701-1709, 2012. Article (CrossRef Link).

[13] D. Zhao, X. Li, and H. Ma, *OMG: How Much Should I Pay Bob in Truthful Online Mobile Crowdsourced Sensing?*, [Online]. Available: http://arxiv.org/abs/1306.5677, 2013. Article (CrossRef Link).

[14] M. Bernstein, J. Brandt, R. Miller, and D. Karger, "Crowds in two seconds: Enabling realtime crowd-powered interfaces," in *Proc. of ACM symposium on User interface software and technology'2011*, pp.33-42, 2011. Article (CrossRef Link).

[15] Archetti, M., "Cooperation as a volunteer's dilemma and the strategy of conflict in public goods games," *Journal of evolutionary biology, vol.22, no.11*, pp. 2192-2200, 2009. Article (CrossRef Link).

[16] Diekmann, A., "Volunteer's dilemma," *Journal of Conflict Resolution*, vol.29, no.4, pp.605-610, 1985. Article (CrossRef Link).



Sungwook Kim received the BS, MS degrees in computer science from the Sogang University, Seoul, in 1993 and 1995, respectively. In 2003, he received the PhD degree in computer science from the Syracuse University, Syracuse, New York, supervised by Prof. Pramod K. Varshney. He has held faculty positions at the department of Computer Science of ChoongAng University, Seoul. In 2006, he returned to Sogang University, where he is currently an associate professor of department of Computer Science & Engineering, and is a research director of the Internet Communication Control research laboratory (ICC Lab.). His research interests include resource management, online algorithms, multimedia network management, bandwidth allocation, adaptive QoS control and game theory for wireless network management.