A Pricing Incentive Mechanism for Mobile Crowd Sensing in Edge Computing

Xin Chen[1], Zhuo Li[1,2], Lianyong Qi[3](✉), Ying Chen[1], Yuzhe Zhao[1], and Shuang Chen[1]

[1] School of Computer Science, Beijing Information Science and Technology University, Beijing, China
{chenxin,lizhuo,chenying}@bistu.edu.cn, duyadude@163.com, 1980995580@qq.com

[2] Beijing Key Laboratory of Internet Culture and Digital Dissemination Research, Beijing, China

[3] School of Information Science and Engineering, Qufu Normal University, Jining, China
lianyongqi@gmail.com

Abstract. Mobile crowd sensing (MCS) has been recognized as a promising method to acquire massive volume of data. Stimulating the enthusiasm of participants could be challenging at the same time. In this paper, we first propose a three-layer mobile crowd sensing architecture and introduce edge servers into it. The edge servers are used to process raw data and improve response time. Our goal is to maximize social welfare. Specifically, we model the social welfare maximization problem by Markov decision process and study a convex optimization pricing problem in the proposed three-layer architecture. The size of the tasks the edge servers assign is adjustable in this system. Then Lagrange multiplier method is leveraged to solve the problem. We derive the experimental data from real-world dataset and extensive simulations demonstrate the performance of our proposed method.

Keywords: Mobile crowd sensing · Pricing · Social welfare · Incentive mechanism · Convex optimization

Supported by the National Natural Science Foundation of China (Nos. 61872044, 61502040), Beijing Municipal Program for Excellent Teacher Promotion (no. PXM2017_014224.000028), Beijing Municipal Program for Top Talent Cultivation (CIT&TCD201804055), Open Program of Beijing Key Laboratory of Internet Culture and Digital Dissemination Research (ICDDXN001), Qinxin Talent Program of Beijing Information Science and Technology University, Supplementary and Supportive Project for Teachers at Beijing Information Science and Technology University (No. 5111823401) and Key Research and Cultivation Projects at Beijing Information Science and Technology University (No. 5211823411).

Y. Yin et al. (Eds.): MobiCASE 2019, LNICST 290, pp. 184–197, 2019.
https://doi.org/10.1007/978-3-030-28468-8_14

1 Introduction

There are many smart-phones with sensor devices proliferating in our daily life, which promote the prevalence of mobile crowd sensing. Mobile crowd sensing can be considered as a novel method to obtain data, handle and share the data [1]. It can be applied in many scenes, such as location [2,3], environmental monitoring [4] and smart transportation [5,6]. However, the process of obtaining data causes consume of the power, flow. Meanwhile, high quality of sensory data is crucial to the platform. Therefore, we need some incentive mechanisms to stimulate the users [10–16].

The traditional mobile crowd sensing system is two-layer framework [10]. With the rapid development of Internet of Thing (IoT), the platform need response quickly and provide service with high reliability [11]. Considering of the above, we introduce edge servers into traditional mobile crowd sensing system [12]. The flow of a typical three-layer mobile crowd sensing system in edge computing is shown in Fig. 1. It is composed of mobile crowd sensing cloud platform, crowds and edge servers. The edge servers can be deployed with mobile equipments (base stations, wireless routers). The task initiators and crowds could use the system to acquire or provide sensing data. The cloud platform could be regarded as an interface of task initiators and crowds.

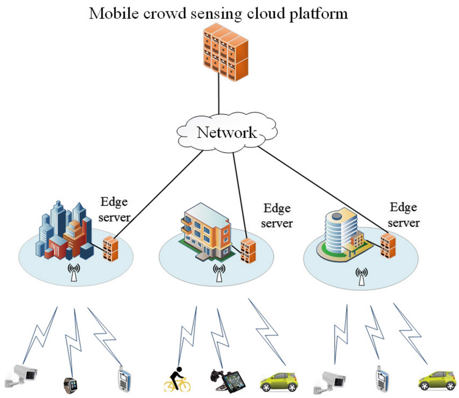

Fig. 1. A mobile crowd sensing system

In the three-layer system, each part of crowds wants to maximize its own utility because of the selfish of users. So we focus on designing an incentive mechanism to stimulate them. Our goal is to maximize the social welfare.

To design an efficient pricing incentive mechanism, there are three challenges we have to address. First, we introduce edge servers in a three-layer structure in mobile crowd sensing system. Second, our goal is to make the social welfare maximization and we must solve the problem in a polynomial time. The third challenge is on how to adjust the demand and supply according to the ability of crowds. Faced with these challenges, we consider a three-layer architecture and tend to boil down the social welfare maximization problem as a Walrasian equilibrium problem. Then convex optimization and Markov decision process(MDP) are used to model and solve the problem. Experiments show that our proposed method is efficient.

The contributions of our paper are listed as follows:

(1) First, we first propose a three-layer mobile crowd sensing platform and introduce edge servers into the platform to make the platform response quickly;
(2) Second, the dynamic of the crowds is considered in this paper. Then we use Walrasian Equilibrium to describe the problem and model by convex optimization and Markov decision process;
(3) Finally, the performance of our proposed algorithms are evaluated through Matlab. The performance of our proposed algorithms is 32.4% better than the existing method SWMA algorithm [17] and 39.3% better than the existing method NWSA [18]. We also compare the overpayment radio and our proposed algorithms is most closest cost than [17,18].

The organization of this paper is as follows. We review the related work in Sect. 2. In Sect. 3, we present the model and the problem formation. The algorithms of pricing for mobile crowd sensing are presented in Sect. 4. Section 5 conducts simulations to evaluate the performance of our proposed algorithms. We conclude the simulation results in Sect. 6.

2 Related Work

We review the related works from three aspects: incentive of mobile crowd sensing, pricing on mobile crowd sensing and incentive of edge computing in this section.

2.1 Mobile Crowd Sensing Applications

The mobile crowd sensing could be applied in transportation, environmental monitoring, healthcare and social network. Tse et al. [5] analyzed the relationship between traffic jam and weather conditions in Beijing through Sina Weibo using social networks. Kalejaiye et al. [6] developed a mobile application for developing areas to predict bus arrival time. Matarazzo et al. [7] used the moving smartphones to monitor bridge vibrations and evaluated bridge avoiding unexpected rehabilitation. Xu et al. developed a NoiseSense system to house a rel-time urban mapping service [8]. Wang et al. [9] leveraged the influenced propagation on the social network to recruit workers.

2.2 Incentive of Mobile Crowd Sensing

The incentive mechanisms of mobile crowd sensing solve the problem that stimulating the enthusiasm of users' participation. Sun *et al.* [14] designed an online incentive mechanism and solved the social welfare maximization problem. It was based on heterogeneous belief values for joint social states and realtime throughput. However, it doesn't consider the optimality of the proposed auction. Jin *et al.* [13] guaranteed near-optimal social welfare based on reverse auction. But this paper doesn't consider the demand of the platform and can't adjust the supply according to the required. Peng *et al.* [15] considered the effort levels of participants to bridge the gap between sensing data quality and reward. However, the aforementioned incentive mechanisms don't consider the uncertainty of the mobile crowd sensing. Gao *et al.* [16] ensured a high probability of success to perform tasks using reverse-auction-based incentive mechanism.

2.3 Pricing on Mobile Crowd Sensing

A proper price of sensing data makes users willing to submit high quality sensing data [19]. Zheng *et al.* [19] presented the architecture of mobile crowd-sensed data market and introduced in-depth study into online data pricing. The method is leveraged to aggregate raw data and determine the trading pricing of sensing data. Duan *et al.* [17] introduced Walrasian Equilibrium as a comprehensive metric to price and solved social welfare maximization problem by dual decomposition. Like this, He *et al.* [20] solved the same problem but leveraged reverse flow network. The aforementioned works don't consider the data quality while pricing. The data quality is took into consideration in [13,21]. Han *et al.* [21] treated the pricing problem as non-submodular optimization problem and then converted it into submodular problem by Poisson binomial distributions.

2.4 Incentive of Edge Computing

The incentive mechanisms of edge computing are based on game theory mostly. Yang *et al.* [22] designed a distributed manner to solve the multi-user computation offloading problem in a multi-channel environment. Liu *et al.* [23] modeled the edge server owners' interaction and solved simulating computation offloading problem based on stackellberg game. Yu *et al.* [24] proposed Wi-Fi monetization model and used stackellberg game to analyse the factors affecting the venue owners. The above works don't solve the computation offloading problem effectively. Zhou *et al.* [25] combined deep learning and edge computing. They leveraged edge computing to process raw data and used reservation pricing auction to recruit participants.

Unlike the aforementioned studies, we first propose a three-layer mobile crowd sensing architecture and add edge servers into the system in this paper. Then we transfer the social welfare maximization problem as convex optimization and solve it by lagrangian multiplier method.

3 System Model and Problem Formulation

In this section, we first present variables to be used in the article. Each edge server plays a game with the crowds to decide which crowds to perform the task.

The crowds are dynamically moving because they may move to another spot while performing the tasks. So we suppose they can accomplish the tasks with a certain probability. The social welfare maximization problem is conducted in this section. Then we use MDP and convex optimization to model and solve it.

3.1 System Overview

Each task needs to be performed in serval spots called Area of Interest (AoI). We divide all task areas into several interest spots $Z = \{z_1, z_2, ..., z_l\}$. There are many interest spots in the AoI. The task set is $A = \{A_1, A_2, ..., A_m\}$, where A_i is a quintuple $A_i = \{Z_i, t_i^b, t_i^f, t_i, N_i\}$. Each task is interested in several spots. There are n crowds $U = \{u_1, u_2, ..., u_n\}$ to perform tasks. Suppose each edge server can receive and perform several tasks at the same time. The edge servers x can be defined as $G = \{G_1, G_2, ..., G_M\}$ where $G_x = \{l_x, g_x^b, g_x^f, c_x, M_x, M_X\}$. We consider that each user can perform a task at one time. The sensing time to perform the task can be divided into many time slots. p_{ij} is the unit price of task a_i in spot z_j. t_{ij} is the time performing task a_i in z_j. We consider the unit price p_{ij} of completing task a_i in different interest spots is different because of the complexity and cost of performing tasks.

$Z_i = \sum_{a_i \in z_j} z_j$ is the locations of task i requests. t_i^b and t_i^f are the earliest beginning time and the latest finishing time respectively. t_i is the sensing time of the task i required and N_i is the number of crowds the sensing task needs. l_x is the location of edge server x. g_x^b, g_x^f is the beginning time and the finish time of the edge server x correspondingly. c_x is the cost of the edge server x calculating from its crowds. M_x is the number of tasks currently being completed and M_X maximum number of performing tasks at the same time.

In this system, each part of the system wants to maximize their own utility and every member works toward this goal in each layer game. We formulate it as a social welfare maximization problem.

3.2 System Model

In this section, the edge servers and the crowds paly a game and decide which crowds to perform the task. Because the crowds are mobile, we can not accurately know the location of crowd. MDP is a common method to deal with continuous optimization in discrete-time. The basic idea of MDP is to choose the appropriate decision-making behavior to maximize the expected return value in the current state.

The MDP consists of a quintet $M = (D, S, A, P_{sa}, R)$, where

D: is the decision points. $D = \{0, 1, 2, ..., N\}$ where N represents that the time all sensing tasks completed.

S: is the states set, $s \in S$, s_i is the state of step i. $S = \boldsymbol{G} \times \boldsymbol{L} \times \boldsymbol{T} \times V = \{G_1, G_2, ..., G_m, L_1, L_2, ..., L_m, T_1, T_2, ..., T_m, V\}$, where $\boldsymbol{G} = \{G_1, G_2, ..., G_m\}$ is an m-dimensional vector represented currently-executing task. $G_i \in \{0, 1\}, i = 1, 2, ..., m$. $G_i = 1$ indicates that the crowds is performing the task and $G_i = 0$

means that the task isn't performed. $\boldsymbol{L} = \{L_1, L_2, ..., L_m\}$ is an m-dimensional vector represented the location of the tasks. $\boldsymbol{T} = \{T_1, T_2, ..., T_m\}$ is an m-dimensional vector represented the sensing time of the tasks and V is the moving rate of the crowds. In a time, the state of crowd is $s \in S$, the current task is $t \in T$, the movement rate is $v \in V$.

$\boldsymbol{a}$: is the set of actions, a_i is the action of step i. The crowds can perform many different tasks, so $\boldsymbol{a} = (a_1, a_2, ..., a_m)$, where $a_i \in \{0, 1\}, i = 1, 2, ..., m$, $a_i = 1$ indicates that the crowds is performing the task and $a_i = 0$ means that the task hasn't been performed.

P_{sa}: is the probability of state transition. P_{sa} is the probability distribution of the other states in the current state $a \in S$ after performing action a. For example, when the crowd takes action a at state s, the probability transferring to s' can be expressed as $p(s'|s, a)$. The current state is $\boldsymbol{s} = [a_1, a_2, ..., a_m, l_1, l_2, ..., l_m, t_1, t_2, ..., t_m, v]$. We choose action $\boldsymbol{a}$ then the transition probability of next state $\boldsymbol{s'} = [a'_1, a'_2, ..., a'_m, l'_1, l'_2, ..., l'_m, t'_1, t'_2, ..., t'_m, v']$ is

$$P(s'|s, a) = \begin{cases} P[v'|v] \prod P[l'_i, t'_i|l_i, t_i], \text{ if } g' = a \\ \\ 0, \text{ else} \end{cases} \tag{1}$$

where $P[v'|v]$ is the transition probability of moving rate. $P[l'_i, t'_i|l_i, t_i]$ is the union transition probability of the sensing time and the location of task i.

The arrival time of crowds follows a random point distribution. We suppose the arrival of crowds obeys the poisson distribution which is shown in Eq. (2). The arrival time is a random sequence of independent exponentially and distribution identically. Because the arrival of crowds is a poisson distribution, the number of crowds in different time is independent. The transition probability of location also obeys the poisson distribution.

$$P_n(k) = \frac{(\lambda t)^n}{n!} e^{-\lambda t} \tag{2}$$

R: $S \times A \to \mathbb{R}$, R is the reward function. If (s, a) transfers to the next state $s\prime$, the reward function is $r(s'|s, a)$. In each state, the value function of task a_i is $V_i(a_i)$ and it is a convex function. The utility of edge server $U_i(a_i)$ is $V_i(a_i)$ minus the payoff paid to crowds, which is defined in Eq. (3).

$$U_i(a_i) = V_i(a_i) - \sum_{j=1}^{l} p_{ij} t_{ij} \tag{3}$$

For the user u_k, the cost function of performing task a_i is $C_{ki}(t_{ki})$ and it is a convex function increased with sensing time. The utility function $U_k(u_k)$ of user u_k is the payoff getting from the edge server minus the cost $C_{ki}(t_{ki})$ of performing tasks which is defined in Eq. (4).

$$U_k(u_k) = \sum_{i=1}^{m} p_{ij} t_{ij} - \sum_{i=1}^{m} C_{ki}(t_{ki}) \tag{4}$$

For the edge servers and crowds, they all want to maximum their utility while the supply from the crowds and the demand from the edge servers are equal. According to exchange market theory of economics, this state reaches Walrasian equilibrium. Walrasian equilibrium means that the total amount of excess demand and excess supply in the entire market must be equal. Then the overall system reaches a Pareto optimal point. Pareto optimal is a kind of ideal state of resource allocation.

3.3 Problem Formulation

Social welfare of the whole system can be defined as Eq. (5)

$$W = \sum_{i=1}^{m} U_i(a_i) + \sum_{j=1}^{n} U_k(u_k) \tag{5}$$

For the edge servers and the crowds, they want to maximize their utilities. Then the problem can be described as a social welfare maximization problem which is defined as follows.

$$\max \quad W \tag{6}$$

$$\text{s.t.} \quad t_{ij} \leq t_i \tag{7}$$

Each task assigned by edge server is t_i, we divide the task t_i into several subtasks, the size of each subtask is less than or equal to corresponding task, which is described in Eq. (7).

4 Pricing Incentive Mechanism for Mobile Crowd Sensing

4.1 Convex Optimization Problem

The social welfare maximization problem proposed in Eqs. (6)–(7) is a convex optimization problem. We transform constrained optimization problem into unconstrained optimization problem using penalty function. The value of edge server $V(a_i)$ and the cost function of performing task a_i is $C_{ki}(t_{ki})$ are convex functions. We can apply Lagrange multiplier method to solve them. First, we introduce Lagrange multiplier method to obtain the augmented matrix where $\lambda_k > 0$. The Lagrange function is defined as Eq. (8)

$$W = \sum_{i=1}^{m} V(a_i) - \sum_{k=1}^{n} \sum_{i=1}^{m} C_{ki}(t_{ki}) + \sum_{j=1}^{l} \sum_{i=1}^{m} \lambda_{ij}(t_i - t_{ij}) \tag{8}$$

Then we define the value function $V(a_i)$ of task a_i as

$$V(a_i) = \omega log(1 + \omega) \tag{9}$$

Different application scenarios have different selection and measurement indicators of t_{ij}. Yang *et al.* [27] used the sensing time submitted by users to evaluate t_{ij}.

In [28], t_{ij} depends on the locations of users through a coverage function. In this paper, t_{ij} is the sensing time the task a_i requests.

The cost of the crowds performing task a_i is

$$C_{ki}(t_{ki}) = b_{ki}t_{ki}^2 + c_{ki}t_{ki} \tag{10}$$

where $b_{ki} > 0$ and $c_{ki} > 0$.

We bring Eqs. (9) and (10) into Eq. (8) then we have

$$W(\mathbf{t},\boldsymbol{\lambda}) = \sum_{i=1}^{m} \omega log(1+\omega) - \sum_{k=1}^{n}\sum_{i=1}^{m}(b_{ki}t_{ki}^2 + c_{ki}t_{ki}) + \sum_{j=1}^{l}\sum_{i=1}^{m}\lambda_{ij}(t_i - t_{ij}) \tag{11}$$

where $\mathbf{t}$ is the vector of sensing time got from the crowds, $\mathbf{t} = (t_{ij})_{z_j \in Z}$. $\mathbf{p}$ is the price vector, $\mathbf{p} = (p_{ij})_{a_i \in A, z_j \in Z}$.

We are motivated the method of constructing lagrange function by literature [29], we modify the lagrange function and consider the MDP problem into it as follows:

$$\begin{aligned} W(\mathbf{t},\boldsymbol{\lambda}) = &\sum_{i=1}^{m} \omega log(1+\omega) - \sum_{k=1}^{n}\sum_{i=1}^{m}(b_{ki}t_{ki}^2 + c_{ki}t_{ki}) + \sum_{j=1}^{l}\sum_{i=1}^{m}\lambda_{ij}(t_i - t_{ij}) \\ &+ \sum_{s' \in S} \tau p(s'|s,a)v_k(s')) + \frac{1}{2\sigma}||\Delta||_2^2 \end{aligned} \tag{12}$$

We consider the Lagrangian dual problem of problem (6), which is shown as follows:

$$\min_{\boldsymbol{\lambda} \geq \mathbf{0}} \quad \max_{\sum_{i=1}^{m} t_{ki} \leq \chi_k} \quad W(\mathbf{t},\boldsymbol{\lambda}) \tag{13}$$

$$\text{s.t.} \quad t_{ij} \leq t_i \tag{14}$$

4.2 Walrasian Equilibrium Algorithm

The dual decomposition method mainly aims at the convex optimization problems. It introduces the Lagrange multiplier, absorbs the constraint conditions into the objective function. Then we solve the optimal Lagrange multiplier as the main problem and decompose the optimization problem of the given Lagrange multiplier into several subproblems and solve separately.

In our paper, the dual problem (13)–(14) can be decomposed into main problem and sub-problem. The sub-problem is that given the Lagrange multiplier $\boldsymbol{\lambda}$, how to optimize $\mathbf{t}$ and $\mathbf{p}$ to maximize $W(\mathbf{t},\boldsymbol{\lambda})$. The main problem is that how to optimize the Lagrange multiplier $\boldsymbol{\lambda}$ to minimize $W(\mathbf{t},\mathbf{p},\boldsymbol{\lambda})$. Then the problem can be solved by two layers of circulation.

Sub Problem Algorithm. The sub problem is how to allocate the task to each crowd in each spot to maximize $W(\mathbf{t}, \mathbf{p}, \boldsymbol{\lambda})$ while given the Lagrange multiplier $\boldsymbol{\lambda}$. We can take the partial derivatives to get the optimal task allocation. We take the derivative of the Eqs. (9) and (10) which is shown in Eq. (15). Then we use the greedy iteration to allocate the task. The algorithm is shown in Algorithm 1.

$$\frac{\partial W(\mathbf{t},\boldsymbol{\lambda})}{\partial \mathbf{t}} = \omega log(1+\omega) - 2b_{ki}t_{ki} - c_{ki} - \lambda_k \tag{15}$$

Algorithm 1. The task allocation algorithm between the edges and the crowds

Require: $\boldsymbol{\lambda}$
Ensure: $\boldsymbol{P}, \boldsymbol{T}, \boldsymbol{W}$
1: initialize the task allocation matrix $T_{i,j,k} \leftarrow \overrightarrow{0}$, the price matrix $P_{i,j,k} \leftarrow \overrightarrow{0}$, the social welfare matrix $W_{i,j,k} \leftarrow \overrightarrow{0}$, where $i \in m$, $j \in l$, $k \in n$
2: **for** $i \in m, j \in l, k \in n$ **do**
3: calculate $T^*_{i,j,k}$ and $P^*_{i,j,k}$ according to Eq.(15)
4: **end for**
5: **repeat**
6: **for** each task in each spot **do**
7: calculate $W_{i,j,k}$ according to Eq.(12)
8: **end for**
9: $(i^*, j^*, k^*) \leftarrow arg \max_{(i,j,k)\in T} W_{i,j,k}$
10: allocate the task to user k^*
11: $T \leftarrow T \setminus \{(i^*, j^*, k^*)\}$
12: **until** $T \in \phi$
13: **return** $\boldsymbol{P}, \boldsymbol{T}, \boldsymbol{W}$

Line 1 initializes the parameters used in this algorithm. We initialize the task allocation matrix $T_{i,j,k} \leftarrow \overrightarrow{0}$, the price matrix $P_{i,j,k} \leftarrow \overrightarrow{0}$ and the the social welfare matrix $W_{i,j,k} \leftarrow \overrightarrow{0}$. For all tasks in all spots, we calculate $T^*_{i,j,k}$ and $P^*_{i,j,k}$ according to the derived function Eq. (15) which is shown in line 2–4. Line 5–12 is the process of allocating tasks to appropriate crowds. Line 6–8 calculates the social welfare of tasks in each spot for each user. Then we select the crowd of maximizing social welfare and allocate the task to him. Finally we get the $\boldsymbol{P}, \boldsymbol{T}, \boldsymbol{W}$.

The time complexity of the Algorithm 1 is $O(lmn + ln^2S)$, where S is the average sensing ability of each crowds.

Main Problem Algorithm. We use the subgradient method to optimize Lagrange multiplier $\boldsymbol{\lambda}$ until it converge to $\boldsymbol{\lambda^*}$. In every iteration, the Lagrange multiplier is updated according to Eq. (16)

$$\lambda_{N+1} = [\lambda_N - \mu_\lambda(N)\frac{\partial W(\mathbf{t},\boldsymbol{\lambda})}{\partial \lambda^N}]^+ \tag{16}$$

where $[x]^+ = max\{0, x\}$. In Eq. (16), $\mu_\lambda(N)$ is the iteration steps. When $N \rightarrow \infty$, $\mu_\lambda(N) \rightarrow 0$ to ensure convergence. If the objective function is derivable, $\frac{\partial W(\mathbf{t},\boldsymbol{\lambda})}{\partial \lambda^N}$ is the corresponding gradient value of the objective function in λ_N. Else, $\frac{\partial W(\mathbf{t},\boldsymbol{\lambda})}{\partial \lambda^N}$ is the time gradient value of the objective function in λ_N.

Through the gradual release and transformation of the original optimization problem, the iterative optimization algorithm is finally obtained which is shown in Algorithm 2.

Algorithm 2. The main problem solution algorithm

Require: N^{max}
Ensure: $\boldsymbol{\lambda}, \boldsymbol{\sigma}, \boldsymbol{P^*}, \boldsymbol{T^*}, \boldsymbol{W^*}$

1: set the initialize number of iteration as $N_0 = 0$, set the initialize lagrangian multiplier λ.
2: **while** $N_0 < N^{max}$ **do**
3:     use algorithm 1 to calculate $\boldsymbol{P^*}, \boldsymbol{T^*}$
4:     update $\boldsymbol{\lambda}$ according to the Eq.(16) using the output of algorithm 1
5:     **if** $|\lambda^{N+1} - \lambda^N| > \varepsilon$ **then**
6:         $t_{ij} = t_{ij} + \alpha$
7:     **else**
8:         break
9:     **end if**
10:     $N_0 = N_0 + 1$
11: **end while**
12: $p^*_{ij} = p_{ij}$
13: $t^*_{ij} = t_{ij}$
14: $\lambda^*_{ij} = \lambda_{ij}$
15: $W^*_{ij} = W_{ij}$
16: **return** $\boldsymbol{p^*}, \boldsymbol{t^*}, \boldsymbol{W^*}, \boldsymbol{\lambda^*}$

Line 1 sets the initialize number of iteration as $N_0 = 0$ and the initialize lagrangian multiplier λ. Line 2–11 is the process of getting the finally P^*, T^*, W^*. In line 3 we use Algorithm 1 to calculate $\boldsymbol{P^*}, \boldsymbol{T^*}$. Then we use the output of Algorithm 1 to update λ. If $|\lambda^{N+1} - \lambda^N| > \varepsilon$, we increase the size of task and continue the iterative process. After the iterative process, we get the $\boldsymbol{p^*}, \boldsymbol{t^*}, \boldsymbol{W^*}, \boldsymbol{\lambda^*}$.

In Algorithm 2, line 3 use Algorithm 1, so the time complexity is $O(lmn + ln^2S)$. The time complexity the Algorithm 2 is $O((n+1)^2(lmn + ln^2S)(n+1))$, that is $O(n^4l(m + nS))$.

5 Performance Evaluation

5.1 Simulation Setup

To evaluate the performance of our proposed algorithms, we take simulations on Matlab. We choose the data set from Stanford Large Network Dataset Collection [33]. The dataset contains the user's id, check-in time, latitude, longitude and location. We use the latitude and longitude to simulate the crowds' location. We classify the latitude between 40–41 and longitude between −123–122 of the crowds to edge servers and they can work for the edge servers. Other edge servers are in a similar manner. The size of tasks, the beginning time and ending time are generated randomly.

5.2 Simulation Results

First we generate 10 edge servers and 100 crowds. First we analyse the convergence and optimality, which is shown in Fig. 2. From the figure, we can get that the more iterations, the greater the social welfare. As the number of iterations increases, the social welfare converges when the iteration at around 800. The more accurate of the Lagrange multiplier λ, the greater the social welfare.

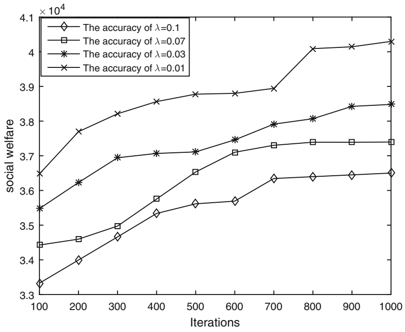

Fig. 2. The convergence of the proposed algorithms

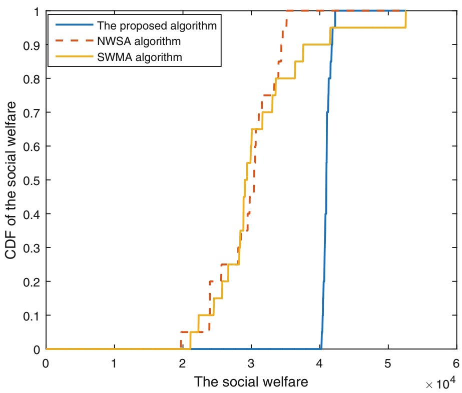

Fig. 3. The CDFs of the social welfare

To evaluate the performance of our proposed method, we take 100 experiments at each scenario. We compare our method with the existing methods Nonoptimal Winner Selection Algorithm(NWSA) [18] and Social Welfare Maximization Algorithm (SWMA) [17]. The cumulative distribution function (CDF) of the social welfare is shown in Fig. 3. The performance of our proposed algorithms are 32.4% better than the existing method SWMA algorithm and 39.3% better than NWSA.

Then we compare the overpayment ratio of our proposed method and the other existing methods. We define the overpayment ratio as $(payoff - cost)/cost$. The payoff is the payment the edge server pays to the crowd and the cost is the crowd performs the task. Figure 4 is the CDFs of the overpayment ratio. The average overpayment ratio of our proposed method is 0.05. The average overpayment ratios of SWMA algorithm and NWSA are 9.65 and 1.83 respectively.

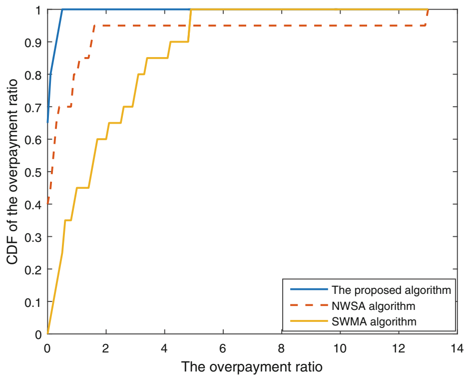

Fig. 4. The CDFs of the overpayment ratio

6 Conclusion

In this work, we first propose a three-layer mobile crowd sensing system structure. The edge servers are introduced to improve the response speed and service with high reliability. Then we conduct a game between the crowds and the edge servers. We build an MDP model and considered the social welfare maximization problem. Then we solve the problem by lagrangian multiplier method. The algorithms are designed to calculate the Lagrange multiplier and the social welfare. We implement them and evaluate the performance by real-world dataset. Our proposed algorithms are better than the existing methods NWSA and SWMA in social welfare and overpayment ratio.

References

1. Ni, J., Zhang, K., Yu, Y., et al.: Providing task allocation and secure deduplication for mobile crowdsensing via fog computing. IEEE Trans. Dependable Secur. Comput. (2018)
2. Jing, Y., Guo, B., Wang, Z., et al.: CrowdTracker: optimized urban moving object tracking using mobile crowd sensing. IEEE Internet Things J. **5**(5), 3452–3463 (2018)
3. Zhou, R., Li, Z., Wu, C.: A truthful online mechanism for location-aware tasks in mobile crowd sensing. IEEE Trans. Mob. Comput. **17**(8), 1737–1749 (2018)
4. Jezdović, I., Nedeljković, N., Živojinović, L., et al.: Smart city: a system for measuring noise pollution. Smart Cities Reg. Dev. (SCRD) J. **2**(1), 79–85 (2018)
5. Tse, R., Zhang, L.F., Lei, P., et al.: Social network based crowd sensing for intelligent transportation and climate applications. Mob. Netw. Appl. **23**(1), 1–7 (2017)
6. Kalejaiye, G.B., Orefice, H.R., Bafutto, M., et al.: Frugal crowd sensing for bus arrival time prediction in developing regions: poster abstract. In: International Conference on Internet-Of-Things Design and Implementation, pp. 355–356. ACM (2017). https://doi.org/10.1145/3054977.3057328
7. Matarazzo, T.J., Santi, P., Pakzad, S.N., et al.: Crowdsensing framework for monitoring bridge vibrations using moving smartphones. Proc. IEEE **106**(4), 577–593 (2018)
8. Xu, Y., Zhu, Y., Qin, Z.: Urban noise mapping with a crowd sensing system. Wirel. Netw. **2018**(3), 1–14 (2018)
9. Wang, J., Wang, F., Wang, Y., et al.: Social-network-assisted worker recruitment in mobile crowd sensing. IEEE Trans. Mob. Comput. **18**(7), 1661–1673 (2018)
10. Zhang, Y., Jiang, C., Song, L., et al.: Incentive mechanism for mobile crowdsourcing using an optimized tournament model. IEEE J. Sel. Areas Commun. **35**(4), 880–892 (2017)
11. Ni, J., Zhang, K., Xi, Q., et al.: Enabling strong privacy preservation and accurate task allocation for mobile crowdsensing. IEEE Trans. Mob. Comput. (2019). https://doi.org/10.1109/TMC.2019.2908638
12. Li, W., Liao, K., He, Q., Xia, Y.: Performance-aware cost-effective resource provisioning for future grid IoT-cloud system. J. Energy Eng. (2019, to appear). https://doi.org/10.1061/(ASCE)EY.1943-7897.0000611
13. Jin, H., Su, L., Chen, D., et al.: Thanos: incentive mechanism with quality awareness for mobile crowd sensing. IEEE Trans. Mob. Comput. (2018)
14. Sun, J.: An incentive scheme based on heterogeneous belief values for crowd sensing in mobile social networks. In: 2013 Global Communications Conference (GLOBECOM), pp. 1717–1722 (2013). https://doi.org/10.1109/GLOCOM.2013.6831321
15. Peng, D., Wu, F., Chen, G.: Data quality guided incentive mechanism design for crowdsensing. IEEE Trans. Mob. Comput. **17**(2), 307–319 (2018)
16. Gao, G., Xiao, M., Wu, J., et al.: Truthful incentive mechanism for nondeterministic crowdsensing with vehicles. IEEE Trans. Mob. Comput. **17**(12), 2982–2997 (2018)
17. Duan, X., Zhao, C., He, S., et al.: Distributed algorithms to compute walrasian equilibrium in mobile crowdsensing. IEEE Trans. Ind. Electron. **64**(5), 4048–4057 (2017)
18. Li, J., Cai, Z., Wang, J., et al.: Truthful incentive mechanisms for geographical position conflicting mobile crowdsensing systems. IEEE Trans. Comput. Soc. Syst. **5**(2), 324–334 (2018)

19. Zheng, Z., Peng, Y., Wu, F., et al.: An online pricing mechanism for mobile crowdsensing data markets. In: Proceedings of the 18th ACM International Symposium on Mobile Ad Hoc Networking and Computing, pp. 1–10. ACM (2017). https://doi.org/10.1145/3084041.3084044
20. He, S., Shin, D.H., Zhang, J., et al.: An exchange market approach to mobile crowdsensing: pricing, task allocation, and walrasian equilibrium. IEEE J. Sel. Areas Commun. **35**(4), 921–934 (2017)
21. Han, K., Huang, H., Luo, J.: Quality-aware pricing for mobile crowdsensing. IEEE/ACM Trans. Netw. **26**(4), 1728–1741 (2018)
22. Chen, X., Jiao, L., Li, W., et al.: Efficient multi-user computation offloading for mobile-edge cloud computing. IEEE/ACM Trans. Netw. **2016**(5), 2795–2808 (2016)
23. Liu, Y., Xu, C., Zhan, Y., et al.: Incentive mechanism for computation offloading using edge computing: a Stackelberg game approach. Comput. Netw. **129**(201), 399–409 (2017)
24. Yu, H., Cheung, M.H., Gao, L., et al.: Economics of public Wi-Fi monetization and advertising. In: IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications, pp. 1–9 (2016)
25. Zhou, Z., Liao, H., Gu, B., et al.: Robust mobile crowd sensing: when deep learning meets edge computing. IEEE Netw. **32**(4), 54–60 (2018)
26. Hu, T., Yang, T., Hu, B.: A data quality index based incentive mechanism for smartphone crowdsensing. In: 2016 IEEE/CIC International Conference on Communications in China (ICCC), pp. 1–6. IEEE (2016). https://doi.org/10.1109/ICCChina.2016.7636875
27. Yang, D., Xue, G., Fang, X., et al.: Crowdsourcing to smartphones: incentive mechanism design for mobile phone sensing. In: Proceedings of the 18th Annual International Conference on Mobile Computing and Networking, 2012, pp. 173–184. ACM (2012). https://doi.org/10.1145/2348543.2348567
28. Singla, A., Krause, A.: Incentives for privacy tradeoff in community sensing. In: First AAAI Conference on Human Computation and Crowdsourcing (2013)
29. Esser, E., Zhang, X., Chan, T.F.: A general framework for a class of first order primal-dual algorithms for convex optimization in imaging science. SIAM J. Imaging Sci. **3**(4), 1015–1046 (2010)
30. Ota, K., Dong, M., Gui, J., et al.: QUOIN: incentive mechanisms for crowd sensing networks. IEEE Netw. **32**(2), 114–119 (2018)
31. Wang, Z., Tan, R., Hu, J., et al.: Heterogeneous incentive mechanism for time-sensitive and location-dependent crowdsensing networks with random arrivals. Comput. Netw. **131**(2018), 96–109 (2018)
32. Jiang, C., Gao, L., Duan, L., et al.: Data-centric mobile crowdsensing. IEEE Trans. Mob. Comput. **17**(6), 1275–1288 (2018)
33. Snap Datasets: Stanford Large Network Dataset Collection, June 2014. http://snap.stanford.edu/data