Incentive Mechanisms for Crowdblocking Rumors in Mobile Social Networks

Yaguang Lin, Zhipeng Cai, *Senior Member, IEEE*, Xiaoming Wang, and Fei Hao

***Abstract*—Mobile social networks (MSNs) have become an indispensable way for people to access information, express emotions, and communicate with each other. However, the advent and extensive use of MSNs has also created fertile soil for the breeding and rapid spread of rumors. Therefore, blocking the spread of rumors in MSNs has always been a hot topic in this field. With the idea of crowdsourcing, we propose a novel rumor control framework, called *Crowdblocking*, in which users can implement control schemes in a collaborative and distributed way, so that the rumors can be controlled more effectively. With the proposed framework, the main problem that arises is how to motivate more users to actively participate in rumor blocking activities. To this end, we design two effective incentive mechanisms in this paper. First, we propose an incentive mechanism based on the Stackelberg game for homogeneous control tasks. We theoretically analyze the Stackelberg equilibrium to maximize the utility of the network manager and users involved in blocking rumor tasks, and ensure that no user can improve its own utility by unilaterally changing the current strategy. Second, for heterogeneous control tasks, we design a real-time reverse auction incentive mechanism, which allows users to have more autonomy and freely customize their own plans to participate in control tasks. Also, we prove that the mechanism possesses the desired properties of task timeliness, computational efficiency, user rationality, manager profitability, and price truthfulness. Finally, we validate the efficiency of the proposed mechanisms through extensive simulation experiments on the real datasets.**

***Index Terms*—Mobile social networks, rumor blocking, incentive, auction, Stackelberg game.**

I. Introduction

THANKS to the rapid proliferation of mobile intelligent devices and the quick development of instant messaging services, Mobile Social Networks (MSNs) have become the most important way for people to access information, express emotions and communicate with each other [1], [2]. Many popular mobile social networking platforms, such as Twitter, Facebook, WeChat and Sina Weibo, demonstrate the unprecedented power of MSNs in information dissemination and sharing [3], [4]. Compared to the traditional way of information interaction, MSNs provide certain benefits such as low transmission delay, high real-time performance and no space limitation, etc. In recent years, MSNs have attracted increasing attention in academia and industry due to their potential value [5], [6].

The development of MSNs provides a continuous impetus for the flow of information. Because of its ubiquitous and accessible characteristics, MSNs not only improve the scope of information sharing, but also speed up the information dissemination [7], [8]. However, it may also become a fertile breeding ground for generating and disseminating malicious rumors or misinformation [9]. When people use MSNs to enjoy convenient and high-speed information services, some unauthenticated information or rumors will flow into it and may spread widely. This poses a serious threat to normal social network activities, such as harming the interests of others, disturbing public order and threatening social stability [10], [11]. Next, let's introduce a motivating scenario. In September 2018, a powerful hurricane "Hurricane Florence" caused severe damage in the Carolinas of America. However, there were rumors on MSNs that "*the Brunswick nuclear power plant is in danger due to nearby flooding*", "*the environmental protection agency does not have the budget to respond to natural disasters*", etc [12]. These have led to serious panic among the public and made the rescue even more difficult.

Rumors spreading in MSNs have been a critical threat to our society. Therefore, how to effectively block the spread of rumors has become a research hotspot. In the recent years, many researchers have devoted themselves to this problem and proposed a variety of control methods [9], [13]–[17]. These methods can be divided into two categories. The first is to identify or infer the source nodes of rumor diffusion based on the network topology and the information flow records, and block or isolate the source nodes to prevent them from further spreading rumors to the network [10], [18], [19]. Although this type of method can uncover the source of rumors and implement corresponding control measures, it ignores other users who are being infected by rumors, leading to the emergence of new sources. Thus, rumors cannot be completely eradicated from the network. The second method is to inject correct information into the network to fight rumors, so as to block the spread of rumors [13]–[15]. For example, some information can be released by authoritative departments or managers. Then this authoritative

Manuscript received March 17, 2019; revised June 6, 2019; accepted July 16, 2019. Date of publication July 24, 2019; date of current version September 17, 2019. This work was supported in part by the National Natural Science Foundation of China under Grants 61872228, 61877037, 61702317, 61601273, and 61602289, in part by the National Science Foundation under Grant 1704287, in part by the Fundamental Research Funds for the Central Universities of China under Grants 2017TS046 and GK201803081, in part by the Scholarship of China Scholarship Council under Grant 201806870045, and in part by the Natural Science Basic Research Plan in Shaanxi Province of China under Grant 2019JM-379. The review of this paper was coordinated by Prof. L. Guo. *(Corresponding author: Xiaoming Wang.)*

Y. Lin, X. Wang, and F. Hao are with the Key Laboratory of Modern Teaching Technology, Ministry of Education, Xi'an 710062, China, and also with the School of Computer Science, Shaanxi Normal University, Xi'an 710119, China (e-mail: light@snnu.edu.cn; wangxm@snnu.edu.cn; fhao@snnu.edu.cn).

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

Digital Object Identifier 10.1109/TVT.2019.2930667

information can be spread to the whole network to maximize its influence, so that the users who may be misled will no longer believe the rumors that come to them. Meanwhile, there are two ways to spread authoritative information: continuance and pulse [11], [20]. The continuance method is to continuously send authoritative information to users, while the pulse method is to disseminate authoritative information to users at intervals. In recent years, inspired by the optimal control theory, some optimal control strategies for blocking rumors based on continuance and pulse forms have been proposed [15], [21], [22]. That is, calculating the optimal control intensity (sending different quantities or different contents of authoritative information against rumors) at each time or period under cost constraints, so as to achieve the optimal control effect.

However, some new problems arise in the implementation of the control methods mentioned above, especially in a wide-area network scenario. It is not difficult to find that most of the current control measures are implemented by managers in a centralized way. The direct result of this is that the managers take on all the workloads of control tasks. The workloads are so heavy that the managers can't finish their work. Following that, based on the concept of maximizing influence by selecting *Top-k* seed nodes in social networks [23], some scholars proposed an improved scheme to assist managers to perform control tasks by selecting some specific users (e.g., users with a high degree of sociality) [13], [24], [25]. Although this relieves the pressure of executing users to some extent, it still does not fundamentally solve the problem of overload. At the same time, a method was proposed for all users to perform tasks in turn, that is, to select different users as executors of tasks at different times [26], [27]. The advantage of this method is to further balance the workloads among users. However, in the network with diverse users and complex structure, the efficiency and quality of tasks are not high due to the difference of social attributes among users. In addition, some users are reluctant to perform control tasks because of selfishness or resource consumption [28], [29].

Different from the existing control methods, we propose a novel control framework inspired by the idea of crowdsourcing [30]–[32], called *Crowdblocking*, that is, users can participate in the activities of control rumors more flexibly and autonomously, and all users can cooperate to implement the whole control scheme. Furthermore, on the basis of the Crowdblocking rumor control framework, if more capable users in the network can be motivated to spontaneously participate in the Crowdblocking rumors and forward the authoritative information, it will be faster to block the spread of rumor information.

Toward this end, we further design incentive mechanisms to encourage more users in MSNs to take an active part in blocking the spread of rumors, so as to achieve distributed and cooperative implementation of all control tasks among more users, and to control rumors faster and more efficiently. Considering the attributes of control tasks, we establish homogeneous control task model and heterogeneous control task model. Then, based on the Stackelberg game theory [33] and the reverse auction theory [34], we establish efficient incentive mechanisms for the above models, respectively. The multi-fold contributions of this paper are summarized as follows:

- We put forward a new Crowdblocking framework for control rumors in MSNs by using crowd intelligence. Besides, we propose incentive mechanisms to encourage more users to actively participate in rumor blocking activities. More importantly, to the best of our knowledge, this paper pioneer the concept of Crowdblocking and incentive mechanisms for control rumors in MSNs.
- For homogeneous control tasks, we design an incentive mechanism based on the Stackelberg game, called HTIM. We solve the Stackelberg equilibrium, and theoretically analyze the optimal strategies of users which can maximize their utilities when other users' strategies are fixed, and ensure that no user can improve its utility by unilaterally changing the strategy. In addition, we propose a method for calculating the optimal budget of the manager to maximize its utility.
- For heterogeneous control tasks, we propose an incentive mechanism based on the real-time online reverse auction called RTA to assign tasks to active participants at the earliest. Meanwhile, we theoretically prove that RTA possesses the desired properties of task timeliness, computational efficiency, user rationality, manager profitability, and price truthfulness.
- We evaluate the performance of the proposed incentive mechanisms on the real dataset. The effect of the HTIM is very close to that of the theoretical optimal strategy, which can create maximum utility for the manager and users. In addition, compared to existing mechanisms, the proposed RTA can speed up the task allocation by 20%.

The remainder of the paper is organized as follows. In Section II, some previous works are reviewed. We present the system model and problem formulation in Section III. Then, in Section IV and V, we propose HTIM and RTA for homogeneous task model and heterogeneous task model, respectively. After evaluating the performance our proposed models and mechanisms in Section VI. We conclude this paper in Section VII.

II. Related Work

Rumor blocking has become a research hotspot in the field of MSNs in the recent years [25]. Scholars have realized the urgency of blocking and controlling rumors, and proposed various control schemes from the perspective of network managers [10], [11], [13]–[21].

One of the representative methods is to find the source of rumors and block it. In [18], Luo *et al.* studied the problem of exploring the source of rumors. According to the cascade relationship of rumor propagation, they established a computational model to estimate the probabilities of users who are most likely to be the source of rumors. Then the problem is transformed into a mixed integer quadratic programming problem with quadratic constraints. The optimal solution of the above problem is obtained by using standard optimization toolboxes. In addition, considering the constraints of algorithm complexity, they also proposed an improved algorithm based on the heuristic optimization to estimate the sources (users) of rumors more quickly. Jiang *et al.* [10] proposed that identifying the sources of rumors and isolating them in time play a vital role in reducing the harm caused by rumors in social networks. They used a back-propagation strategy to identify suspects who spread rumors, and established a microscopic rumor spreading model to calculate the maximum likelihood for each suspect. Their work can more accurately identify the sources of rumors. However, adopting the method of blocking some users to control the spread of rumors will inevitably affect the experience of some users and the quality of service of the network, and to some extent disturb the normal communication order of the network. Wang *et al.* [14] realized this realistic problem and proposed a model of dynamic rumor influence minimization with the user experience. They regarded user experience as a constraint of network services and set a threshold of maximum blocking time for each user. Once the maximum blocking time was exceeded, the normal communication activities of the blocked user were restored immediately.

Another mainstream approach to effectively control the spread of rumors in MSNs is that sending positive information corresponding to rumors. Wang *et al.* [15] explored the impact of user mobility on information diffusion, and proposed a computational model to describe the process of rumor propagation. Then, a double pulse control strategy is proposed to dispel rumors by sending two kinds of positive information regularly in the network. Besides, considering cost constraints, an optimal control method for epidemic information diffusion in social networks is proposed [21]. They modeled real-time control measures as time-varying dynamic functions, and solved the optimal distribution of dynamic functions through dynamic programming, thus minimizing the control cost in the network. He *et al.* [11] paid attention to the heterogeneity of users in social network, and designed a control method which combines the real-time control with pulse control to suppress the spread of rumors. They also calculated the implementation intensity of control strategies for different victims to ensure cost effectiveness. In summary, all of the above works have designed effective control measures, providing a feasible solution for rumor control in MSNs. However, none of these efforts addressed the issue of how to implement control strategies.

Inspired by the problem of how to select *Top-k* seed nodes to maximize the impact of information in social networks, Tong *et al.* [13] explored how to select k seed users under a given budget to maximize the number of unaffected users by disseminating positive information. They proposed a sampling method based on the reverse tuple to select seed users to perform tasks, and proved that the proposed method has low time complexity. In order to solve the problem of user selection in rumor blocking, Zhu *et al.* [24] studied how to use the least cost to select the least positive seed user, so that the effect of positive seed can reach the preset threshold. At the same time, considering the disturbance of rumor information to seeds and the time-effectiveness of information, they proposed a scalable greedy algorithm to determine the positive seed nodes. Both methods can find the users set of task execution under a certain budget, but the premise is that all users are willing to participate in the task execution. So, they only focus on the selection of users, not the design of incentives.

With the increasing scale of MSNs, the types of users in the network become more and more complex. Scholars have proposed some incentive mechanisms to encourage users to participate in the task execution independently [30], [35], [36]. These incentives are mostly common in the field of crowdsensing and so on. Referring to the auction theory in economics, Gao *et al.* [35] proposed a Lyapunov-based Vickrey Clarke groves auction mechanism to motivate users to perform tasks, which greatly improves the proportion of user participation. Similarly, an incentive mechanism called LSB is proposed to motivate users to participate in the crowdsourcing tasks [30]. The authors designed a reverse auction mechanism, proved the submodule property of its objective function, and proposed a greedy strategy to select the user who performs the task. However, the LSB mechanism has been proved to be inconsistent with the truthfulness property.

In addition, participating in sensing tasks may expose users' privacy information, in order to motivate privacy-sensitive users to perform sensing tasks. Koh *et al.* [37] proposed an incentive mechanism based on the Stackelberg game, which can meet the privacy requirements of users, and increase the range of sensors and the diversity of data. Then the unique solution of Stackelberg equilibrium is derived and the stability of the mechanism is proved. Nevertheless, aforementioned mechanisms are all offline, and cannot be applied to real-time task allocation environment. Lin *et al.* [38] designed an online incentive mechanism called SOS, which can make assignment decision immediately when a user or task arrives, and proved that it can overcome the Sybil attack in the network. However, they assume that only one user can be assigned to tasks in a period of time. When multiple tasks or users come at the same time, the mechanism allocates tasks slowly.

The design of incentive mechanisms has also been studied in many other fields, such as cooperative communication, spectrum trading, network routing, etc [39], [40]. These incentive mechanisms provide a theoretical basis on how to make users actively participate in the execution of tasks, and the proposed methods are worthy of our reference. However, because of the different objectives and different nature of the scenarios, the proposed incentive mechanism cannot be directly applied to the scenario of blocking rumor spreading in MSNs.

For all this, we will design new incentive mechanisms to encourage users to actively participate the task of controlling the spread of rumors. Considering different types of tasks, we will propose incentive mechanisms for homogeneous tasks and heterogeneous tasks, respectively.

III. System Models and Problem Formulation

Firstly, we introduce the Crowdblocking rumor control framework. Secondly, considering the attributes of control tasks, we establish the homogeneous control task model and the heterogeneous task model. In the homogeneous task model, the manager publishes the same control tasks. For example, the manager wants to refute rumors by sending the same correct information to users. On the contrary, in the heterogeneous task model, the content of control tasks released by the manager is different. For

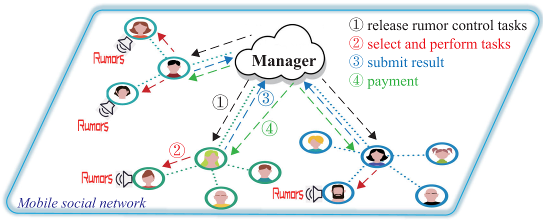


Fig. 1. The diagram of Crowdblocking framework.

instance, the manager needs to send specific control information to prevent or correct users according to the degree of rumor infection. In addition, in the homogeneous task model, users receive the same payment for each task because all tasks are the same, whereas, in the heterogeneous task model, the payment of each task is different as the content and value of tasks are different.

Therefore, the above two models are different but complementary. The design of incentive mechanism in the homogeneous task model can provide task assignment scheme for homogeneous networks or scenarios with the same control measures. On the contrary, the design of incentive mechanism in heterogeneous task models can be applied to heterogeneous networks and scenarios where more flexible control measures are adopted. In addition, the two models proposed can also be applied to information maximization scenarios with minor modifications.

A. Crowdblocking Framework

We consider the Crowdblocking rumor control framework to have a manager M and n network users $S = \{1, 2, \ldots, n-1, n\}$, where $n \geq 2$. As shown in Fig. 1, all users can communicate with the manager in the cloud. Firstly, the manager broadcasts some tasks for rumor control to uninfected neighbor users according to the rumor blocking plan (step ①). After considering the factors of cost and energy consumption, etc., the users receiving broadcast messages may select some tasks from M to make their own execution plans and carry out the corresponding tasks (step ②). After users complete the selected tasks, the execution results are submitted to the manager (step ③). Then, the manager checks the results of users and pays them a certain amount of rewards (we call payments) to compensate for their loss in energy consumption and so on (step ④), and the task assignment of this round is completed. It is worth noting that the manager can identify whether the users in the network are infected by rumors or not, and it will not assign tasks to users who have been infected by rumors. In fact, some effective methods for identifying the nodes (users) infected by rumors through the analysis of user activity records, have been proposed in recent years [10], [11], [15], [18]. Note that although the method of identifying infected nodes is beyond the scope of this work, the proposed Crowdblocking framework is compatible with any existing method.

In addition, we assume that the manager and users in the network are selfish and rational. The goal of the manager is to assign tasks to network users in a timely manner with minimal cost. For network users, the goal is to earn benefits to themselves by participating in the control tasks. Undoubtedly, users will not actively participate in control tasks without guaranteed profits. Therefore, while completing the control tasks, it must be guaranteed that a user will be paid no less than its cost, otherwise the user will refuse to participate in the control tasks. Similarly, in each round of task assignment, the manager must ensure that the payment allocated to all users is less than the budget, that is to say, its own income is positive.

B. Homogeneous Task Model

In the homogeneous task model, the manager publishes some homogeneous control tasks and announces that the total budget for the tasks is R, where $R > 0$. If the user i is not infected with rumors and it decides to participate in the execution of tasks, we use n_i to represent the number of tasks that user i decides to perform, where n_i is a positive integer, i.e., $n_i \in N^+$. Undoubtedly, a user needs to spend a certain amount of cost to execute tasks. Here, we define the total cost of user i to execute n_i tasks as the following non-linear function:

$$c_i = \alpha + \gamma_i n_i + \kappa n_i^2. \tag{1}$$

Where, $\alpha > 0$ is the cost of receiving messages from the manager. As all tasks are the same in the homogeneous task model, each user only needs to receive messages once. $\gamma_i > 0$ is the unit cost caused by channel occupancy, energy consumption and other factors when user i forwards a message. κn_i^2 is used to represent the additional cost to users due to bad phenomena such as message retransmissions caused by signal interference or poor channel quality, where $\kappa > 0$. Note that, we do not impose a strong assumption on the c_i since any other form of non-linear function could also be used here to derive the solution. Further, we assume that $R \gg c_i$, and when users finish their selected tasks, the manager will divide R proportionally according to the number of tasks each user performs as its payment. If the user i performs n_i tasks, the payment of user i is $p_i = \frac{R}{\sum_{j\in S} n_j} n_i$, and the utility of user i is:

$$u_i = \frac{R}{\sum_{j\in S} n_j} n_i - \left(\alpha + \gamma_i n_i + \kappa n_i^2\right). \tag{2}$$

It is worth noting that each user in the network is selfish, and users make decisions before participating in the tasks to ensure $u_i > 0$. Then, if $\frac{R}{\sum_{j\in S} n_j} \leq \frac{\alpha+\gamma_i n_i+\kappa n_i^2}{n_i}$ (i.e., the payment for each task is less than the cost of each task performed by the user i), user i will not participate in the n_i-th task. For the manager with the budget R, the more control tasks completed by users, the greater the benefits created for it. Here, we represent the number of tasks completed by all users as m_0, then $m_0 = \sum_{j\in S} n_j$, and we define the utility of the manager as follows:

$$U = f(m_0) - R, \tag{3}$$

where f is the manager's valuation function of the number of tasks completed by all users. We assume that f is a strictly convex function and increases monotonously with the increase of m_0. It is worth noting that this assumption is feasible and realistic, and has been accepted and adopted in the previous works [30], [35], [37], [40].

Under the above model, the problems and challenges we faced in designing incentive mechanism are:

Problem 1: $\max\ u_i\,(n_i)$, that is, how to determine the strategy of user i so as to maximize its utility u_i when the budget R and other users' strategies are fixed.

Problem 2: $\max\ U\,(R)$, that is, how to determine the budget R to maximize the manager's utility U.

C. Heterogeneous Task Model

In heterogeneous task model, the manager publishes m different control tasks $\Gamma = \{\tau_1, \tau_2, \ldots, \tau_{m-1}, \tau_m\}$ within time step $t \in [1, \cdots, T]$. Because of the heterogeneity of tasks, the costs for completing different tasks are different. Moreover, due to different network structures, the cost for performing the same task varies with different users. We use c_i^j to represent the cost for user i to complete task τ_j. In addition, by implementing control tasks, users are no longer misled by rumors. Therefore, for managers, performing each task can create a certain value. We define the value created by completing the task τ_j as v_{τ_j}. Besides, each user has its own working time $[t_i, t'_i]$, in which $t_i, t'_i \in [1, \cdots, T]$ and $t_i < t'_i$. Users can only perform tasks during their working time, and when $t \notin [t_i, t'_i]$, the cost of user i performing any task is $+\infty$. Users can select tasks they want to perform during their working time independently and form their own tasks set. Assuming that user i selects a set of tasks Γ_i, where $\Gamma_i \subseteq \Gamma$.

Then, the user i will submit a bidding information $B_i = (t_i, t'_i, \Gamma_i, b_i)$ to the manager, where b_i is the bid for the user i to perform the tasks set Γ_i, and $b_i \geq \sum_{\tau_j \in \Gamma_i} c_i^j$. We call B_i the *time-task-bid*, to express its willingness and requirements to participate in the control tasks. After receiving users' time-task-bid pair, the manager will select some users as the executors to form a set of winners $\bar{S}$, where $\bar{S} \subseteq S$. If $i \in \bar{S}$, the manager will pay the user i a certain amount of payment p_i after all the selected tasks have been performed, where $p_i \geq b_i$. By this stage, the utility $\tilde{u}_i$ of user i by participating in these tasks is as follows:

$$\tilde{u}_i = \begin{cases} p_i - \sum_{\tau_j \in \Gamma_i} c_i^j, & \text{if } i \in \bar{S}; \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

After all winners complete their tasks, the utility that the manager will gain is shown as follows:

$$\tilde{U} = \sum_{\tau_j \in \cup_{i \in \bar{S}} \Gamma_i} v_{\tau_j} - \sum_{i \in \bar{S}} p_i \tag{5}$$

Another problem and challenge we face are:

Problem 3: How to design an effective incentive mechanism to motivate more users to participate in heterogeneous control tasks more autonomously, while satisfying the following conditions:

- *Task timeliness:* the mechanism can assign tasks to users at the earliest time.
- *Computational efficiency:* the mechanism can determine winners and calculate payment in polynomial time.
- *User rationality:* the mechanism can guarantee that no user's utility is negative.
- *Manager profitability:* the mechanism can ensure that the utility of the manager is non-negative.
- *Price truthfulness:* None of the users can get more payment by raising the bid that is different from its cost.

TABLE I
NOTATIONS IN HOMOGENEOUS TASK MODEL

Symbols	Description
R, n	budget of the manager and number of users
$m_0, \bar{S}$	number of all tasks performed and set of winners
S, S_{-i}	set of all users and set of all users excluding user i
n_i, n_i^*	strategy of user i and optimal strategy of user i
N	set of all users' strategies
N_{-i}	set of all all users' strategies excluding user i
α	cost of users receiving a message
γ_i	cost of user i sending a message
κ	additional cost weight for users to perform tasks
c_i, p_i	total cost of user i and payment of user i
u_i, U	utility of user i and utility of the manager

IV. Incentive Mechanism for Homogeneous Tasks

From the homogeneous task model we established in Sub section III-B, there is a obvious conflict of interest between the manager and users (that is, users want the budget as much as possible, while the manager wants the budget as little as possible). From the perspective of Game Theory [33], we build the strategic interaction between the manager and users as a non-cooperative game. Each user's strategy is to determine the number of tasks n_i, it has to perform so as to maximize its utility when the other users' strategies are fixed. The manager's strategy is to determine the budget R to maximize its own utility. Since the Stackelberg game can solve the conflict of interests between users and manager, and find out their optimal strategy [33], [40], we model the Homogeneous Tasks Incentive Mechanism as a Stackelberg game, called HTIM. Both users and manager are players in HTIM, and the manager is the leader and the users are the followers, where both users and manager only seek to maximize their own utilities. We use S to represent the set of all users in the network, and S_{-i} to represent the set of all users excluding user i, we have $S = S_{-i} \cup i$. Then, we use N and N_{-i} to denote the set of all users' strategies and the set of all users' strategies excluding user i, respectively. So, we also have $N = N_{-i} \cup n_i$. For clarity, we list the frequently used notations in homogeneous task model and their respective descriptions in Table I.

A. Theoretical Optimal Strategies of Users

First, we calculate the optimal strategy n_i^* to maximize the utility of user i in case of a fixed budget R and a strategies set N_{-i}, where $n_i^* \geq 0$ and $n_i^* \in N$. We calculate the first and second derivatives of the utility function Eq. (2), respectively:

$$\frac{\partial u_i}{\partial n_i} = \frac{R \sum_{j \in S_{-i}} n_j}{\left(\sum_{j \in S} n_j\right)^2} - \gamma_i - 2\kappa n_i. \tag{6}$$

$$\frac{\partial^2 u_i}{\partial {n_i}^2} = -\frac{2R \sum_{j \in S_{-i}} n_j}{\left(\sum_{j \in S} n_j\right)^3} - 2\kappa. \tag{7}$$

For each user $j \in S$, we have $n_j \geq 0$. Besides, we have $R > 0$ and $\kappa > 0$. So, the second-order derivative of u_i is always negative. Thus, the utility function u_i is strictly convex, which means that if the optimal strategy n_i^* of user i exists, then n_i^* is unique. Next, by setting Eq. (6) to 0, we have:

$$\frac{R}{\sum_{j\in S} n_j} - \frac{Rn_i}{\left(\sum_{j\in S} n_j\right)^2} - \gamma_i - 2\kappa n_i = 0,\ i \in S. \quad (8)$$

If user i participates in the tasks, then i is winner and $n_i > 0$, otherwise $n_i = 0$. We define the set of winners as $\bar{S}$, and $\bar{S} = \{j \in S\,|n_j > 0\}$. Let k_0 be the number of winners, and $k_0 = \left|\bar{S}\right|$. Then, considering that $\sum_{j\in S} n_j = \sum_{q\in\bar{S}} n_j$, we have:

$$\frac{R}{\sum_{q\in\bar{S}} n_q} - \frac{Rn_p}{\left(\sum_{q\in\bar{S}} n_q\right)^2} - \gamma_p - 2\kappa n_p = 0,\ p \in \bar{S}. \quad (9)$$

For above equation, we sum all the elements in $\bar{S}$ and get:

$$\frac{k_0 R}{\sum\limits_{q\in\bar{S}} n_q} - \frac{R}{\sum_{q\in\bar{S}} n_q} - \sum_{p\in\bar{S}} \gamma_p - 2\kappa \sum_{q\in\bar{S}} n_q = 0. \quad (10)$$

Further, we solve the $\sum_{q\in\bar{S}} n_q$. By discarding the meaningless negative value, we can get the following result:

$$\sum_{q\in\bar{S}} n_q = \frac{\sqrt{\left(\sum_{p\in\bar{S}} \gamma_p\right)^2 + 8R\kappa\left(k_0 - 1\right)} - \sum_{p\in\bar{S}} \gamma_p}{4\kappa}. \quad (11)$$

Let $\theta = \sum_{q\in\bar{S}} n_q$, and bring θ into Eq. (9), we have:

$$n_p^* = \frac{\theta\left(R - \gamma_p\theta\right)}{R + 2\kappa\theta^2},\ p \in \bar{S}. \quad (12)$$

Hence, the optimal strategy of user i can be obtained as:

$$n_i^* = \begin{cases} \frac{\theta(R-\gamma_i\theta)}{R+2\kappa\theta^2}, & \text{if } i \in \bar{S}, \\ 0 & \text{otherwise.} \end{cases} \quad (13)$$

It is worth noting that the calculated n_i^* is a positive number, not a positive integer, that is, $n_i^* \in N$ and $n_i^* \notin N^+$.

B. Nash Equilibrium

In this subsection, we explore whether a stable state can be achieved in HTIM when all users adopt the optimal strategies, so that no user can improve its utility by unilaterally changing the current strategy in this stable state. As we all know, this state is the Nash equilibrium in the non-cooperative game [41]. Next, we define Nash equilibrium for HTIM.

Definition 1: All users' strategies $(n_1^{ne}, n_2^{ne}, \ldots, n_{n-1}^{ne}, n_n^{ne})$ in HTIM are the Nash equilibrium state. Then for any user i, the strategy n_i^{ne} satisfies:

$$u_i\left(n_i^{ne}, N_{-i}^{ne}\right) \geq u_i\left(n_i, N_{-i}^{ne}\right).$$

Given a budget R, if the Nash equilibrium exists and is unique in HTIM, we can determine the unique optimal strategy for each user to maximize its utility when other users' strategies are fixed. Next, we analyze the existence and uniqueness of Nash equilibrium in HTIM. We have:

Theorem 1: For a given budget R, if the set of strategies $N^* = \left(n_1^*, n_2^*, \ldots, n_{n-1}^*, n_n^*\right)$ for all users is the unique Nash equilibrium of HTIM, then it satisfies the following conditions:

i) let $\bar{S} = \{i \in S\,|n_i^* > 0\}$, then $\left|\bar{S}\right| \geq 2$;

ii) $n_i^* = \{ \begin{smallmatrix} \frac{\theta(R-\gamma_i\theta)}{R+2\kappa\theta^2} & \text{if } i\in\bar{S} \\ 0 & \text{otherwise;} \end{smallmatrix}$

iii) if $\gamma_p \leq \max_{j\in\bar{S}} \{\gamma_j\}$, then $p \in \bar{S}$;

iv) The cost of user i to perform a homogeneous task is $c_i^1 = \alpha + \gamma_i + \kappa$. We rank the costs of n users in a non-decreasing sequence $\gamma_1, \gamma_2, \ldots, \gamma_{n-1}, \gamma_n$. Let h be the largest positive integer less than n and satisfy $\gamma_h < \frac{R}{\theta} < \gamma_{h+1}$, then $\bar{S} = \{1, 2, \ldots, h-1, h\}$.

We prove the condition (i) as follows.

Proof: Let's first assume that $\left|\bar{S}\right| = 0$, and no user currently involves in the control tasks. So, any user i in the network can unilaterally change its strategy from $n_i = 0$ to $n_i > 0$ to participate in the control tasks and gain utility. This is contrary to the definition of the Nash equilibrium, so $\left|\bar{S}\right| \neq 0$.

Then, we assume that if $\left|\bar{S}\right| = 1$, user i will involve in the control tasks, and we consider it in the following two Cases:

Case 1: If $n_i > 1$, the utility that the user i gets from the manager is $u_i = R - \alpha - \gamma_i n_i - \kappa n_i^2$. At this point, the user can gain more utility by changing the strategy from n_i to $n_i - 1$, which is contrary to the definition of the Nash equilibrium.

Case 2: If $n_i = 1$, because $R \gg c_i$, any other user j except i in the network can unilaterally change its strategy from $n_j = 0$ to $n_j > 0$ to participate in control tasks and gain more utility. It still contradicts the definition of the Nash equilibrium.

To sum up, we prove that $\left|\bar{N}\right| \geq 2$. ■

Then, we prove the condition (ii) as follows.

Proof: From Eq. (13), we can see that if $i \in \bar{S}$, i participates in the control tasks. At this time, the optimal strategy taken by user i is $n_i^* = \frac{\theta(R-\gamma_i\theta)}{R+2\kappa\theta^2}$, which can ensure that i can get the greatest utility when other users' strategies are fixed. In addition, when $i \notin \bar{S}$, then i is not involved in the control task, there is no doubt that the strategy of i is $n_i = 0$.

Next, we prove the condition (iii) as follows.

Proof: Undoubtedly, if $i \in \bar{S}$, then $n_i > 0$. From (ii) we know that if $n_i > 0$, there is $R - \gamma_i\theta > 0$. So, we have:

$$\gamma_i < \frac{R}{\theta},\ \ i \in \bar{S}. \quad (14)$$

Further, we can draw the following conclusion:

$$\max_{i\in\bar{S}} \gamma_i < \frac{R}{\theta}. \quad (15)$$

Assuming that $\gamma_p \leq \max_{j\in\bar{S}} \{\gamma_j\}$, and $p \notin \bar{S}$. From (ii) it is known that the strategy of p is $n_p = 0$. At this point, we take the strategy $n_p = 0$ and substitute it in Eq. (6) and get the following:

$$\frac{R}{\sum_{j\in S} n_j} - \gamma_p = \frac{R}{\theta} - \gamma_p > \max_{i\in\bar{S}} \gamma_i - \gamma_p \geq 0. \quad (16)$$

Then, we can conclude that if $\gamma_p \leq \max_{j\in\bar{S}} \{\gamma_j\}$ and $p \notin \bar{S}$, user p can increase its utility to $u_p > 0$ by unilaterally changing the strategy to $n_p > 0$, which is contrary to the Nash equilibrium. Therefore, the condition (iii) is proved. ■

Finally, we prove the condition (iv) as follows.

Algorithm 1: HTIM.

Input: $R, S, \{\gamma_i\}$
Output: $\{n_i\}, \{p_i\}$
1 Set $\bar{S} \leftarrow \emptyset, \{n_i\} \leftarrow \{0\}, \{p_i\} \leftarrow \{0\}$;
2 Sort $i \in S$ by the order $\gamma_1 \leq \gamma_2 \leq \cdots \leq \gamma_{k-1} \leq \gamma_k$;
3 $\bar{S} \leftarrow \bar{S} \cup \{1, 2\}$; $j \leftarrow 3$;
4 **while** $j \leq k$ *and* $\gamma_j < \frac{4\kappa R}{\sqrt{\left(\sum_{q=1}^{j} \gamma_q\right)^2 + 8R\kappa|\bar{S}|} - \sum_{q=1}^{j} \gamma_q}$ **do**
5 $\quad \bar{S} \leftarrow \bar{S} \cup \{j\}$; $j++$;
6 **end**
7 $\theta \leftarrow \frac{\sqrt{\left(\sum_{p \in \bar{S}} \gamma_p\right)^2 + 8R\kappa\left(|\bar{S}|-1\right)} - \sum_{p \in \bar{S}} \gamma_p}{4\kappa}$;
8 **foreach** $i \in S$ **do**
9 $\quad$ **if** $i \in \bar{S}$ **then** $n_i \leftarrow \left\lfloor \frac{\theta(R - \gamma_i \theta)}{R + 2\kappa\theta^2} \right\rfloor$;
10 $\quad$ **else** $n_i \leftarrow 0$;
11 **end**
12 $n_0 \leftarrow \sum_{i \in \bar{S}} n_i$;
13 **foreach** $i \in S$ **do**
14 $\quad$ **if** $n_i > 0$ **then** $p_i \leftarrow n_i \frac{R}{n_0}$;
15 $\quad$ **else** $p_i \leftarrow 0$;
16 **end**

Proof: The n users are sorted in a cost non-decreasing order $\gamma_1, \gamma_2, \ldots, \gamma_{n-1}, \gamma_n$. Because h is the largest positive integer that satisfies $\gamma_h < \frac{R}{\theta}$, it is known $h \in \bar{S}$ from Eq. (14). From (iii) we can get that if p is the largest positive integer that satisfies $\gamma_p \leq \max_{j \in \bar{S}} \{\gamma_j\}$, then $p \in \bar{S}$ and $\bar{S} = \{1, 2, \ldots, p-1, p\}$. So, we have $p \leq h$. Assume that $p < h$, then there is $\gamma_{p+1} < \frac{R}{\theta}$ and $p+1 \notin \bar{S}$. Similarly, we take the strategy $n_{p+1} = 0$ of user $p+1$ into Eq. (6) and get:

$$\frac{R}{\sum_{j \in S} n_j} - \gamma_{p+1} = \frac{R}{\theta} - \gamma_{p+1} > 0. \tag{17}$$

The above equation shows that user $p+1$ can gain more utility by increasing the number of tasks performed, which is contrary to the Nash equilibrium. So, we have $p = h$, and $\bar{S} = \{1, 2, \ldots, h-1, h\}$. ∎

C. Incentive Mechanism

In the above subsections, we calculate the optimal strategy set $N^* = \left(n_1^*, n_2^*, \ldots, n_{n-1}^*, n_n^*\right)$ for all users with a given budget and prove that the strategy set N^* is the unique Nash equilibrium for HTIM.

Next, we design the detailed steps of HTIM, as shown in Alg. 1. It is worth noting that the theoretical optimal strategy n_i^* is a decimal value. Here, we have to consider an important scenario constraint that the number of tasks n_i that user i chooses to perform must be an integer value. In order to ensure user utility $u_i \geq 0$, the n_i^* is rounded down as the number tasks n_i of user i in HTIM, that is, $n_i = \lfloor n_i^* \rfloor$.

In Alg. 1, we initialize the winners set $\bar{S}$, the winners' task number set $\{n_i\}$ and the winners' payment set $\{p_i\}$ (line 1). Then we sort all the users belonging to S in a non-decreasing order in accordance with the cost γ_i, and add the first two users to the set $\bar{S}$ as winners (lines 2-3). Next, we add other eligible users (i.e., users get more payments than their costs for participating in tasks) as winners to the set $\bar{S}$ (lines 4-6). We calculate the number of tasks performed by each user (lines 8-11). Finally, we calculate the payment for all users according to the number of tasks they performed (lines 13-16).

The time complexity of Alg. 1 is analyzed as follows. Sorting all users (lines 4-6) takes $O\left(n\log_2 n\right)$, The time required to determine the winners (lines 4-6), calculate the number of tasks for all users and calculate the payment for all users is equal to $O\left(n\right)$. Therefore, the overall time complexity is dominated by sorting, and is $O\left(n\log_2 n\right)$.

D. Optimal Strategy of the Manager

In HTIM, both users and manager are players, in which the manager is the leader and the users are the followers. In the case of any budget R, all users' strategies have the unique Nash equilibrium state. So, the manager can determine the value of R to maximize its own utility. We bring the optimal strategy calculated by Eq. (12) into Eq. (3) and conclude that:

$$U = f\left(\sum_{i \in S} n_i^*\right) - R, \tag{18}$$

where

$$n_i^* = \begin{cases} \frac{\theta(R - \gamma_i \theta)}{R + 2\kappa\theta^2}, & \text{if } i \in \bar{S}. \\ 0 & \text{otherwise.} \end{cases}$$

Theorem 2: There is a unique optimal strategy R^*, which constitutes HTIM's unique Stackelberg Equilibrium (R^*, N^*), where N^* is the set of all users' optimal strategies calculated by Eq. (12) when the budget is R^*. And, under the condition of (R^*, N^*), the manager's utility is maximized.

Proof: As can be seen from Eq. (3), the manager's utility function $U(n_i)$ is a strictly concave function related to n_i under a fixed R. From the Subsection IV-A, we can conclude that HTIM can determine the unique optimal strategies set N^* for all users at any value of R. Therefore, we can conclude that the manager's utility function $U(R, N^*)$ is still strictly convex at any R. There must be the unique R^*, so that the manager's utility can be maximized under the condition of (R^*, N^*), and the unique R^* can be calculated by using either bisection or Newton's method [42]. ∎

V. Incentive Mechanism for Heterogeneous Tasks

The auction-based mechanism is considered to be an effective solution to motivate users to participate in the execution of tasks. In this section, we propose a Real-Time reverse Auction mechanism, called RTA with the aim of motivating users to participate in heterogeneous tasks. We consider the manager to be the auctioneer and the users as bidders. Table II lists the frequently used notations and descriptions in heterogeneous task model while the meanings of symbols $n, S, \bar{S}, p_i$ are the same as that in Section IV.

A. Objectives of RTA Mechanism

The RTA must satisfy task timeliness, computational efficiency, user rationality, manager profitability, and price truthfulness. Task timeliness is used to ensure that RTA can assign

TABLE II
NOTATIONS IN HETEROGENEOUS TASK MODEL

Symbols	Description
t, T	current time step and maximum time step
τ_j, Γ, m	task, set of tasks and number of tasks
$\Gamma_i, [t_i t'_i]$	set of tasks selected by user i and active period of user i
c_i^j	cost of user i performing task τ_j
b_i	bid of user i for executing selected tasks set
B_i	time-task-bid pair submitted by user i to the manager
v_{τ_j}, v_i	value of the task τ_j and marginal value of user i
$\tilde{u}_i, \tilde{U}$	utility of user i and utility of manager
$S^t, \bar{S}^t$	set of active users and set of winners in time step t
S^t_{-i}	set of active users in time step t excluding user i

Algorithm 2: RTA Mechanism.

Input: $T, S, \Gamma, \{B_i\}$
Output: $\{\bar{S}^t\}, \{p_i\}$
1 Set $\bar{S} \leftarrow \emptyset, \bar{S}^t \leftarrow \emptyset, S^t \leftarrow \emptyset, t \leftarrow 1, A \leftarrow \Gamma$;
2 Winner Selection $(A, t, S, T, \{B_i\})$;
3 **foreach** $i \in \bar{S}$ **do**
4 $\quad \{p_i\} \leftarrow \{0\}, t \leftarrow t_i$;
5 $\quad$ Winner Pricing($t, S^t, \bar{S}, \{\bar{S}^t\}, \{B_i\}$);
6 **end**

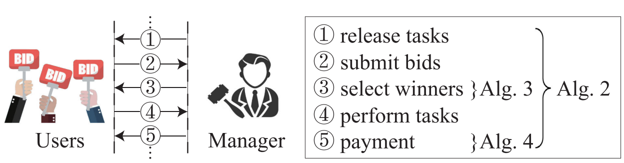


Fig. 2. The schematic diagram of RTA.

tasks to users in a timely manner. Computational efficiency is used to ensure that RTA can be executed in polynomial time. User rationality is used to ensure that each user can benefit from participating in tasks. Similarly, manager profitability is used to ensure that the manager's utility is non-negative. In addition, price truthfulness is used to ensure the fairness and robustness of RTA. We use the following theorem to ensure that RTA possesses the property of price truthfulness.

Theorem 3: [43] If RTA ensures price truthfulness, the following two conditions must be satisfied.

- The winner selection rule must be monotonous, that is, if user i wins the auction by $B_i = (t_i, t'_i, \Gamma_i, b_i)$, user i can also win it by submitting a lower bid $\hat{b}_i$ or a broader working time $\hat{t}_i, \hat{t}'_i$, where $\hat{b}_i \leq b_i$ and $\hat{t}_i \leq t_i, \hat{t}'_i \geq t'_i$;
- The payment p_i to each winner must be the critical value, and if the user i changes the bid to $\hat{b}_i > p_i$, the user i will not win the auction.

B. Design of RTA Mechanism

We design the implementation steps of RTA, as shown in Alg. 2. It consists of two parts. The first part is the winner selection (line 2). The second part is the calculation of each winner's payment (lines 3-6). In addition, we show the functions and relationships of the Algorithms in Fig. 2.

The detailed steps of winner selection are shown in Alg. 3. In each time step, we first calculate the active users set S^t at

Algorithm 3: Winner Selection.

Input: $A, t, S, T, \{B_i\}$
Output: $\{\bar{S}^t\}, \bar{S}$
1 Set $\bar{S} \leftarrow \emptyset, \bar{S}^t \leftarrow \emptyset, S^t \leftarrow \emptyset, t \leftarrow 1, A \leftarrow \Gamma$;
2 **while** $A \neq \emptyset$ *and* $t \leq T$ **do**
3 $\quad$ **foreach** $i \in S$ **do**
4 $\quad\quad$ **if** $t_i \leq t$ and $t'_i \geq t$ **then** $S^t \leftarrow S^t \cup \{i\}$;
5 $\quad$ **end**
6 $\quad$ **if** $S^t \backslash \bar{S} \neq \emptyset$ **then**
7 $\quad\quad i \leftarrow \arg\max_{j \in S^t \backslash \bar{S}} \left(v_j\left(\bar{S}\right) - b_j\right)$;
8 $\quad\quad$ **while** $v_i\left(\bar{S}\right) \geq b_i$ *and* $S^t \backslash \bar{S} \neq \emptyset$ **do**
9 $\quad\quad\quad \bar{S} \leftarrow \bar{S} \cup \{i\},\ \bar{S}^t \leftarrow \bar{S}^t \cup \{i\},\ A \leftarrow A \backslash \Gamma_i$;
10 $\quad\quad\quad$ **if** $S^t \backslash \bar{S} \neq \emptyset$ **then**
$\quad\quad\quad\quad i \leftarrow \arg\max_{j \in S^t \backslash \bar{S}} \left(v_j\left(\bar{S}\right) - b_j\right)$;
11 $\quad\quad$ **end**
12 $\quad$ **end**
13 $\quad t \leftarrow t + 1$;
14 **end**

Algorithm 4: Winner Pricing.

Input: $t, S^t, \bar{S}, \{\bar{S}^t\}, \{B_i\}$
Output: $\{p_i\}$
1 **while** $t \leq t'_i$ **do**
2 $\quad S^t_{-i} = S^t \backslash \{i\},\ C^t \leftarrow \bigcup_{\forall k < t} \bar{S}^k$;
3 $\quad$ **if** $S^t_{-i} \backslash C^t \neq \emptyset$ *and* $\max_{j \in S^t_{-i} \backslash C^t} \left(v_j\left(C^t\right) - b_j\right) > 0$ **then**
4 $\quad\quad$ **repeat**
5 $\quad\quad\quad i_j \leftarrow \arg\max_{j \in S^t_{-i} \backslash C^t} \left(v_j\left(C^t\right) - b_j\right)$;
6 $\quad\quad\quad p_i \leftarrow$ $\max\{p_i, \min\{v_i(C^t) - (v_{i_j}(C^t) - b_{i_j}), v_i(C^t)\}\}$;
7 $\quad\quad\quad C^t \leftarrow C^t \cup \{i_j\}$;
8 $\quad\quad$ **until** $v_{i_j} \leq b_{i_j}$ *or* $S^t_{-i} \backslash C^t = \emptyset$;
9 $\quad$ **end**
10 $\quad p_i \leftarrow \max\{p_i, v_i(C^t)\}$;
11 $\quad t \leftarrow t + 1$;
12 **end**

t time step (lines 3-5), and then we assign tasks to users in a greedy manner according to the marginal value (we will show the definition of marginal value in Theorem 5) created by users and the bid of users (lines 6-12).

We show detailed steps of winner pricing in Alg. 4. For each user i, at each time step t in its working time, we first calculate S^t_{-i}, which is the set of active users in time step t excluding user i (line 2). Then, we compute the maximum bid of user i such that i can be selected instead of other users in S^t_{-i} in time step t. Finally, we use the maximum bid of the user i during the working time as the critical value of its payment (lines 3-10).

C. Theoretical Analysis of RTA Mechanism

In this subsection, we will theoretically prove that RTA has the properties of task timeliness, computational efficiency, user rationality, manager profitability, and price truthfulness. As shown in the following theorems:

Theorem 4: RTA achieves computational efficiency with a time complexity of $O(nm^3)$.

Proof: In winner selection stage, the worst case is that only one task is assigned in each while-loop (lines 2-14 of Alg. 3). Calculating the active users set S^t takes $O(n)$ time (lines 3-5 of Alg. 3). It takes $O(nm)$ to compute the user with the largest marginal value. Because the number of tasks is m, each winner gets assigned at least one task, and the maximum number of winners is m. Thus, the time complexity of winner selection stage is $O(nm^2)$. When calculating the payments for each user i, the worst case is to re-select the winners from all users excluding user i (this process is similar to Alg. 3), and it takes $O(nm^2)$ time. Since the maximum number of winners is m, the for-loop (lines 3-6 of Alg. 2) takes $O(nm^3)$ time. To sum up, the time complexity of RTA is dominated by this for-loop, and is $O(nm^3)$. ∎

Theorem 5: RTA is rational for all users.

Proof: From the line 8 of Alg. 3, if i can become winner, there must be $v_i(\bar{S}) \geq b_i$, where $v_i(\bar{S})$ is the new added utility created by user i for the manager on the basis of the selected winners set $\bar{S}$, that is, $v_i(\bar{S}) = \sum_{\tau_j \in \Gamma_i \setminus \cup_{k \in \bar{S}} \Gamma_k} v_{\tau_j}$. We call $v_i(\bar{S})$ the*marginal value* created by the user i to the manager. From line 6 of Alg. 4, we can get $p_i \geq \min\{v_i - (v_{i_j} - b_{i_j}), v_i\}$, where user i_j is the $j\text{-}th$ winner after removing user i at $t \in [t_i, t'_i]$. At this point, we consider two Cases:

Case 1: If i_i exists, then we have $p_i \geq \min\{v_i - (v_{i_i} - b_{i_i}), v_i\}$. Because the order of winners is determined by the Greedy method, we have $v_i - b_i \geq v_{i_i} - b_{i_i}$. So, $b_i \leq v_i - (v_{i_i} - b_{i_i})$. Then we can get $b_i \leq \min\{v_i - (v_{i_i} - b_{i_i}), v_i\} \leq p_i$.

Case 2: If i_i does not exist, it means that the user who replaces user i at the $i\text{-}th$ position does not exist when user i is removed, that is, the condition of line 3 of Alg. 4 is not satisfied. According to line 10 of Alg. 4, we have $b_i \leq v_i = p_i$. ∎

Therefore, RTA has been proved to be rational for users.

Theorem 6: RTA can ensure manager profitability.

Proof: As can be seen from Alg. 3, each user $i \in \bar{S}$ satisfies $b_i \leq v_i$. Therefore, the utility of the manager is $\tilde{U} = \sum_{i \in \bar{S}} (v_i - p_i)$. Next, we prove that $p_i \leq v_i$ is satisfied for any user $i \in \bar{S}$.

Firstly, it can be concluded from line 6 of Alg. 4, that is:

$$p_i = \max\left\{\max_{1 \leq j \leq k}\left(v_{i(j)} - \left(v_{i_j} - b_{i_j}\right)\right), v_{i(k+1)}\right\} \tag{19}$$

where i_j is the $j\text{-}th$ winner after removing user i in $t \in [t_i, t'_i]$, and k is the number of winners. $v_{i(j)}$ represents the marginal value of user i at $j\text{-}th$ position. Then we consider the following two cases:

Case 1: If i_j exists, we assume that after removing user i, the $r-th$ winner in $t \in [t_i, t'_i]$ satisfies:

$$r = \arg\max_{j \in S^t_{-i} \setminus C^t} \min\{v_{i(j)} - (v_{i_j} - b_{i_j}), v_{i(j)}\} \tag{20}$$

Further, we have:

$$p_i = \min\{v_{i(r)} - (v_{i_r} - b_{i_r}), v_{i(r)}\} \tag{21}$$

From the lines 4-8 of Alg. 4, we have $v_{i_r} > b_{i_r}$. Because the order of winners is determined by the Greedy method, we also have $v_{i(r)} \leq v_i$. Moreover, we can conclude that $p_i = v_{i(r)} - (v_{i_r} - b_{i_r}) < v_{i(r)} \leq v_i$

Case 2: If i_j does not exist, it means that the user who replaces user i at the $j\text{-}th$ position does not exist when user i is removed, that is, the condition of line 3 of Alg. 4 is not satisfied. According to line 10 of Alg. 4, we have $v_i = p_i$.

Thus, the manager profitability of RTA is guaranteed. ∎

Theorem 7: RTA is truthful for all users.

Proof: We show that any user i can't make more utility by submitting a false working time $[\hat{t}_i, \hat{t}'_i]$ or a false bid $\hat{b}_i$.

We first demonstrate that the winner selection rule of RTA is monotonic. If a false working time window $[\hat{t}_i, \hat{t}'_i]$ submitted by the user i is narrower than the true working time window $[t_i, t'_i]$, the user i may lose the chance to be selected as the winner. At the same time, the marginal value generated by submitting $[\hat{t}_i, \hat{t}'_i]$ will not be greater than that generated by submitting $[t_i, t'_i]$. If the $[\hat{t}_i, \hat{t}'_i]$ submitted by the user i is wider than $[t_i, t'_i]$, when user i is selected to perform the task in $t \in [\hat{t}_i, \hat{t}'_i] \setminus [t_i, t'_i]$, then the cost is $c_i = \infty$ and the payment is $p_i = 0$. Then, we can conclude that by submitting a false working time $[\hat{t}_i, \hat{t}'_i]$, the user i can neither be selected as a winner in advance nor increase its payment. In addition, as we can see from Alg. 3 that in the same time step t, the value of $v_j\left(\bar{S}\right) - b_j$ for each user j will increase monotonously with a decreased bid b_j. Therefore, the winner selection rule of RTA is monotonic.

Next, we show that the payment to each winner is the critical value. User i is selected as the winner by submitting bid b_i in time step t_i, and the payment is p_i. If user i submits a false bid $\hat{b}_i$ and $\hat{b}_i > p_i$, then during the period of $t \in [t_i, t'_i)$, user i will not be selected as winner. This is because the winner selection rule of RTA is monotonic and $v_i - \hat{b}_i < v_i - b_i$. From Eq. (19), there is always a user who satisfies $v_i - \hat{b}_i < v_{i_j} - b_{i_j}$ in $t \in [t_i, t'_i]$. Conversely, if $\hat{b}_i \leq p_i$, the user i will be a winner in $t \in [t_i, t'_i]$. Hence, the payment p_i is a critical value for user i to be winner.

To sum up, we can conclude that RTA can ensure truthfulness for all users. ∎

Theorem 8: RTA is timely for all tasks.

Proof: From Theorem 7, we conclude that the winner selection rule of RTA is monotonic. In addition, from line 8 of Alg. 3, we can know that in each time step t, as long as the candidate user meets the criteria $v_i \geq b_i$, it can become a winner, and the selected tasks can be executed in time at t. That is to say, RTA guarantees task timeliness to the greatest extent on the basis of platform profitability. ∎

VI. Performance Evaluation

In this section, we conduct extensive simulations to validate the performance of our proposed models and mechanisms. Firstly, for homogeneous task model, we compare the number of winners, the number of tasks executed and the user utility while using the optimal strategy and HTIM, respectively, and we compare the approximation between the optimal strategy and HTIM mechanism. Secondly, for heterogeneous task model, we analyze the speed of assigning tasks, the utility of user and the utility of manager in RTA. In addition, we compare RTA with LSB mechanism [30] and SMS mechanism [38] to illustrate the advantages of RTA. Finally, we validate the adaptability and effectiveness of the HTIM and RTA in large-scale network scenarios.

TABLE III
THE DEFAULT VALUES OF BASIC PARAMETERS

Parameter	Description	Value
b	user cost parameter in HTIM	100
σ	user cost parameter in RTA	0.5
R	total budget	1000000
κ	additional cost of user in HTIM	0.1
k_{max}	maximum number of selected tasks	5
$[v_{min}, v_{max}]$	value of task	[10, 20]
$[t_{min}, t_{max}]$	working time window of user	[0, 300]
$[t_{short}, t_{long}]$	working time length of user	[10, 20]

A. Experimental Scenario Setting

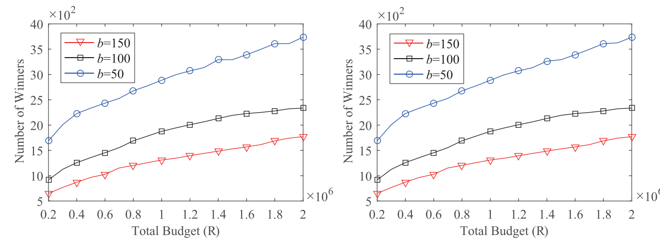
In order to get more objective and realistic experimental results, we carry out experiments on the real Twitter dataset. This dataset has the attributes of nodes and the relationships among nodes in Twitter network [44]. In the original Twitter dataset, we selected a connected subset of 10,000 users and their relationships to conduct our experiments. It is worth noting that the above users are not deliberately selected, hence the experimental results are without loss of generosity.

In the homogeneous task model, we consider that the cost for sending messages is related to the degree of a user on the social network graph (i.e. the number of neighbors in an undirected graph), and the higher the degree of users is, the lower the corresponding cost will be. So, let's assume that $\gamma_i = b/\ln(1+\lambda_i)+\eta$, where λ_i is the degree of user i on the social network graph, the term $\ln(1+\lambda_i)$ is used to smooth the degree differences among users. And b is a constant that determines the magnitude of the γ_i, and $\eta \in [0,1]$ is a random value to add a perturbation to the γ_i. Note that we do not impose a strong assumption on the γ_i, and any other forms of functions that can reflect the above relationship could also be used here to derive similar results.

In addition, in the heterogeneous task model, we have the following settings: 1) The values of tasks are distributed evenly in $[v_{min}, v_{max}]$. 2) The working time of users is distributed evenly in $[t_{min}, t_{max}]$ and the lengths of users' working time are distributed evenly in $[t_{short}, t_{long}]$. 3) The number of tasks selected by users is evenly distributed in $[1, k_{max}]$. 4) The cost of user i for executing the selected tasks set is $c_i = \left(1-\sigma\log_{(1+\lambda_{max})}(1+\lambda_i)\right)\sum_{\tau_j\in\Gamma_i} v_{\tau_j}$, in which v_{τ_j} is the value of task τ_j, λ_i is the degree of user i on the social network graph, and λ_{max} is the maximum degree of users. The term $\log_{(1+\lambda_{max})}(1+\lambda_i)$ is used to smooth the user's degree and get a normalized result, and $\sigma \in [0,1]$ is a constant used to determine the magnitude of the costs of all users. Similarly, the form of c_i we present here is not imposed by a strong assumption either. Besides, we set the default values of basic parameters as mentioned in Table III.

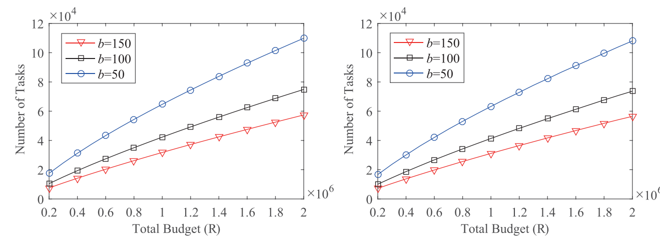
B. Simulation Results of Homogeneous Tasks

Number of winners: Fig. 3 shows the relationships between the number of winners and the manager's budget or parameter b under the optimal strategy and HTIM. As we can see, with the increase in the budget, more and more users will become winners. In addition, as the value of b increases, the cost of users performing tasks increases, resulting in fewer winners. Therefore, we can conclude that the larger the total budget or the lower the cost of users performing tasks, the more likely users will become winners. It is worth pointing out that the uneven trend of the number of winners is due to the nonuniform distribution of users' degree in the Twitter dataset. More importantly, by comparing Fig. 3(a) with Fig. 3(b), we can see that the number of winners determined by the optimal strategy and HTIM is very close in different scenarios. It is proved that the proposed HTIM and the optimal strategy have a high approximation.



(a) Optimal strategy (b) HTIM mechanism

Fig. 3. The number of winners vs. the total budget.

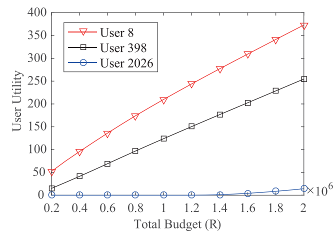


(a) Optimal strategy (b) HTIM mechanism

Fig. 4. The number of tasks performed vs. the total budget.

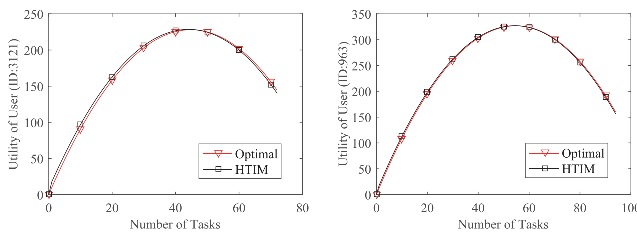
Number of tasks performed: Fig. 4 shows the relationships between the number of tasks performed and the manager's budget or parameter b under the optimal strategy and HTIM. We can see that the number of tasks performed increases with an increase in the budget or a decrease in the value of b. Therefore, if the manager's budget is bigger or each user's cost is lower, the more users are motivated to perform more tasks. By comparing Fig. 4(a) with Fig. 4(b), it can be found that the number of tasks in HTIM is slightly lower than that in the optimal strategy. The reason is that HTIM adopts rounded down method on the basis of the optimal strategy solution and abandons some tasks.

User utility: We randomly select three users (IDs are 8, 398 and 2026) to explore the relationship between the utility of user and the manager's budget under the optimal strategy and HTIM, as shown in Fig. 5. The relationship of the selected three user costs is $c_8 < c_{398} < c_{2026}$. With the increase of the budget R, the utilities of three users keep increasing. In addition, we can also detect that the user's utility in HTIM is slightly greater than that in the optimal strategy, because a small part of tasks is abandoned in HTIM mechanism, resulting in fewer total number of tasks to be performed, so the payment for a single task will increase, and the user's utility will increase accordingly.



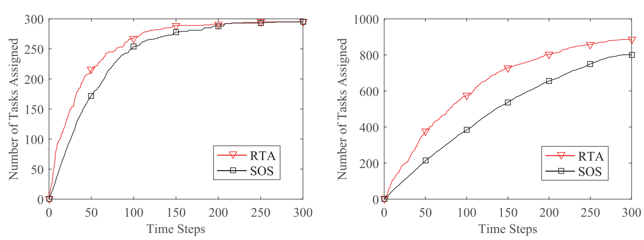
(a) Optimal strategy (b) HTIM mechanism

Fig. 5. The user utility vs. the total budget.



(a) User ID: 3121 (b) User ID: 963

Fig. 6. The utility of user vs. the number of tasks.



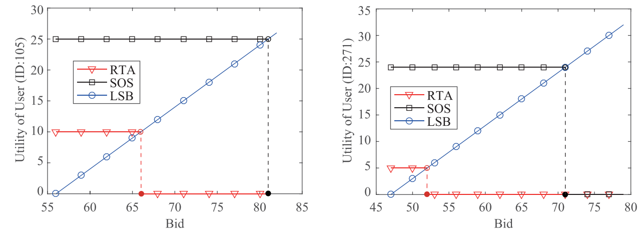
(a) 300 tasks (b) 1000 tasks

Fig. 7. The number of tasks assigned vs. time steps.

User strategy: We randomly select two users (IDs are 3121 and 963) to analyze the relationship between user utility and the number of tasks performed with a fixed budget and fixed other users' strategies. Users 3121 and 963 are both winners. From Fig. 6, we can see that with the increase of the number of tasks performed, the utility of the corresponding user increases first and then decreases. We can then verify that when the budget and other users' strategies are fixed, the user can get the greatest utility by adopting the optimal strategy (45 and 57, respectively). Changing the strategy, whether increasing or reducing the number of tasks, will lead to a decrease in the utility. In addition, the results in the HTIM are very close to those in the optimal strategy, which proves that the optimal strategy we proposed can accurately calculate the optimal strategy for each user. The proposed HTIM can make the user's utility very close to the theoretical optimal value under constraints.

C. Simulation Results of Heterogeneous Tasks

Speed of assigning tasks: Fig. 7 shows the trend of the number of tasks assigned over time steps in RTA and SOS. As can be seen, the number of tasks assigned in both mechanisms increases over time steps. Obviously, the number of tasks assigned in RTA is growing faster than that in SOS. Therefore, we can conclude



(a) User ID: 105 (b) User ID: 271

Fig. 8. The bid of user vs. the utility of user.

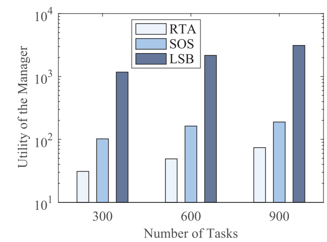
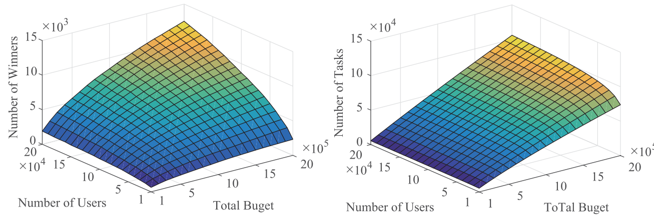


Fig. 9. The utility of the manager vs. the number of tasks.

that RTA can assign tasks to executing users more quickly. In addition, we can also detect that the advantage of RTA is more obvious in scenarios with more tasks. Because RTA allocates as many tasks as possible to active users in each time step, while SOS only chooses one user to allocate tasks in each time step.

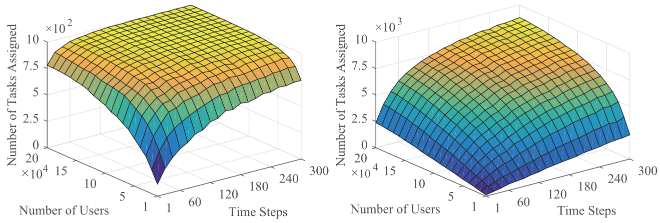
Utility of user: We select two users (IDs are 105 and 271) to analyze the changes of utilities obtained by submitting different bids. The costs of user 105 and user 271 are 56 and 47, and the value created by performing all tasks is 82 and 79, respectively. From Fig. 8, we can see that with the increase of bids, the utilities of users in LSB increase linearly, which means that users can make more utility by submitting a bid higher than the cost. However, in RTA and LSB, when the user increases the bid, the utility will remain unchanged until it is greater than a certain critical value. Then, the user will lose the chance to become a winner, and the utility is 0. Thus, the critical values of user 105 in RTA and SOS are 66 and 81. When the bid is below the critical value, user will become the winner. RTA and SOS regard 66 and 81 as the payment to user 105, and the corresponding utilities are 10 and 25, respectively. A similar conclusion can be drawn for user 271. Therefore, RTA and SOS are truthful.

Utility of the manager: Fig. 9 shows the relationship between the utility of the manager and the number of tasks. As the number of tasks increases, the utilities of the manager are increasing in all three mechanisms with LSB having the greatest utility. This is because the payments to users in LSB are determined by the bids of users, and LSB can create maximum utility for the manager if users' bids equal to their costs. In RTA and SOS, the payments are given according to the calculated critical values, and the critical value will not be lower than the bid, so the utilities of the manager in RTA and SOS are less than that in LSB. Thus, RTA and SOS can guarantee the price truthfulness at the cost of a certain amount of manager's utility.



(a) Number of winners (b) Number of tasks

Fig. 10. The performance of HTIM large-scale scenarios.



(a) 1000 tasks (b) 10000 tasks

Fig. 11. The performance of RTA in large scale scenarios.

D. Simulation Results on Large-Scale Scenarios

In order to evaluate the performance of the proposed HTIM and RTA mechanisms in large-scale scenarios, we built a synthetic dataset as the large-scale network scenarios used in the experiments. In the synthetic dataset, the number of users is different while the other parameters are consistent with the settings in the Subsection VI-A. When the number of network users varies from 10,000 to 200,000, we first investigate the number of winners and the number of tasks completed in HTIM mechanism, and then we simulate the number of tasks assigned by RTA mechanism. The simulation results are shown in Figs. 10 and 11.

Performance of HTIM: Fig. 10(a) shows the relationships between the number of winners with the number of users and the budget under HTIM. We can see that as the number of users increases, more and more users will be selected as winners. Also, an increase in the budget can lead to an increase in the number of winners. In addition, comparing Fig. 10(a) with Fig. 3(b), we can find that they show the same trend when the number of users is fixed. Fig. 10(b) shows the relationships between the number of tasks with the number of users and the budget under HTIM. Clearly, with the increase of the number of users or the budget, the number of tasks performed increases. In addition, we can still observe that the trends in the number of tasks as budget changes in Fig. 10(b) and Fig. 4(b) are consistent. The above results and conclusions show that our proposed HTIM can be fully applicable to large-scale scenarios with satisfactory and stable performance.

Performance of RTA: Fig. 11 shows the changing trend of the number of tasks assigned with the number of users and time steps when the total number of tasks is 1,000 and 10,000, respectively. Similar to Fig. 7, the number of tasks successfully assigned to executing users in both scenarios increases over time steps. In addition, as the number of users increases, more and more users participate in the task execution by submitting more bids, so the task allocation and execution speed will be faster. Importantly, we can still conclude that RTA has robust performance in large-scale scenarios and can still assign tasks to users faster.

VII. Conclusion

In this paper, we propose two effective incentive mechanisms to encourage users to actively participate in the task of blocking the spread of rumor information in MSNs. For homogeneous control tasks, we focus on user fairness and propose an incentive mechanism based on the Stackelberg game. For heterogeneous tasks, we focus on user autonomy and timeliness of task allocation, and propose an incentive mechanism based on the real-time online reverse auction. This paper can provide theoretical basis and feasible technical approaches for the user motivation or the task allocation in rumor blocking scenarios. The proposed methods can motivate more users to participate in the execution of tasks independently, so the advantage of these mechanisms will be more prominent in large-scale networks. In addition, the proposed incentive mechanisms can be extended to other scenarios, such as advertising, cooperative communication, maximizing influence, etc. In future works, we plan on exploring other factors such as the users' interests on the impact of incentive measures.

References

[1] T. Qiu, B. Chen, A. K. Sangaiah, J. Ma, and R. Huang, "A survey of mobile social networks: Applications, social characteristics, and challenges," *IEEE Syst. J.*, vol. 12, no. 4, pp. 3932–3947, Dec. 2018.

[2] Z. He, Z. Cai, and J. Yu, "Latent-data privacy preserving with customized data utility for social network data," *IEEE Trans. Veh. Technol.*, vol. 67, no. 1, pp. 665–673, Jan. 2018.

[3] Y. Zhang *et al.*, "Mobile social big data: Wechat moments dataset, network applications, and opportunities," *IEEE Netw.*, vol. 32, no. 3, pp. 146–153, May/Jun. 2018.

[4] Z. Cai, Z. He, X. Guan, and Y. Li, "Collective data-sanitization for preventing sensitive information inference attacks in social networks," *IEEE Trans. Dependable Secure Comput.*, vol. 15, no. 4, pp. 577–590, Jul./Aug. 2018.

[5] F. Hao, G. Min, M. Lin, C. Luo, and L. T. Yang, "Mobifuzzytrust: An efficient fuzzy trust inference mechanism in mobile social networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 11, pp. 2944–2955, Nov. 2014.

[6] Y. Lin, X. Wang, F. Hao, L. Wang, L. Zhang, and R. Zhao, "An on-demand coverage based self-deployment algorithm for big data perception in mobile sensing networks," *Future Gener. Comput. Syst.*, vol. 82, pp. 220–234, 2018.

[7] Z. Lu, Y. Wen, W. Zhang, Q. Zheng, and G. Cao, "Towards information diffusion in mobile social networks," *IEEE Trans. Mobile Comput.*, vol. 15, no. 5, pp. 1292–1304, May 2016.

[8] X. Zheng, Z. Cai, J. Yu, C. Wang, and Y. Li, "Follow but no track: Privacy preserved profile publishing in cyber-physical social systems," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 1868–1878, Dec. 2017.

[9] A. Zubiaga, A. Aker, K. Bontcheva, M. Liakata, and R. Procter, "Detection and resolution of rumours in social media: A survey," *ACM Comput. Surv.*, vol. 51, no. 2, 2018, Art. no. 32.

[10] J. Jiang, S. Wen, S. Yu, Y. Xiang, and W. Zhou, "Rumor source identification in social networks with time-varying topology," *IEEE Trans. Dependable Secure Comput.*, vol. 15, no. 1, pp. 166–179, Jan./Feb. 2018.

[11] Z. He, Z. Cai, J. Yu, X. Wang, Y. Sun, and Y. Li, "Cost-efficient strategies for restraining rumor spreading in mobile social networks," *IEEE Trans. Veh. Technol.*, vol. 66, no. 3, pp. 2789–2800, Mar. 2017.

[12] "Hurricane florence rumor control," Oct. 2018. [Online]. Available: https://www.fema.gov/florence-rumors

[13] G. Tong *et al.*, "An efficient randomized algorithm for rumor blocking in online social networks," *IEEE Trans. Netw. Sci. Eng.*, to be published, doi: 10.1109/TNSE.2017.2783190.

[14] B. Wang, G. Chen, L. Fu, L. Song, and X. Wang, "Drimux: Dynamic rumor influence minimization with user experience in social networks," *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 10, pp. 2168–2181, Oct. 2017.

[15] X. Wang, Y. Lin, Y. Zhao, L. Zhang, J. Liang, and Z. Cai, "A novel approach for inhibiting misinformation propagation in human mobile opportunistic networks," *Peer-to-Peer Netw. Appl.*, vol. 10, no. 2, pp. 377–394, 2017.

[16] X. Wang, L. Zhang, Y. Lin, Y. Zhao, and X. Hu, "Computational models and optimal control strategies for emotion contagion in the human population in emergencies," *Knowl. Based Syst.*, vol. 109, pp. 35–47, 2016.

[17] Z. He, Z. Cai, and X. Wang, "Modeling propagation dynamics and developing optimized countermeasures for rumor spreading in online social networks," in *Proc. IEEE 35th Int. Conf. Distrib. Comput. Syst.*, 2015, pp. 205–214.

[18] W. Luo, W. P. Tay, and M. Leng, "How to identify an infection source with limited observations," *IEEE J. Sel. Topic Signal Process.*, vol. 8, no. 4, pp. 586–597, Aug. 2014.

[19] F. Ji and W. P. Tay, "An algorithmic framework for estimating rumor sources with different start times," *IEEE Trans. Signal Process.*, vol. 65, no. 10, pp. 2517–2530, Mar. 2017.

[20] L. Huo and C. Ma, "Dynamical analysis of rumor spreading model with impulse vaccination and time delay," *Physica A*, vol. 471, pp. 653–665, 2017.

[21] P. Chen, S. Cheng, and K. Chen, "Optimal control of epidemic information dissemination over networks," *IEEE Trans. Cybern.*, vol. 44, no. 12, pp. 2316–2328, Dec. 2014.

[22] C. Nowzari, V. M. Preciado, and G. J. Pappas, "Optimal resource allocation for control of networked epidemic models," *IEEE Trans. Control Netw. Syst.*, vol. 4, no. 2, pp. 159–169, Jun. 2017.

[23] R. Bian, Y. S. Koh, G. Dobbie, and A. Divoli, "Identifying top-k nodes in social networks: A survey," *ACM Comput. Surv.*, vol. 52, no. 1, 2019, Art. no. 22.

[24] Y. Zhu, D. Li, and Z. Zhang, "Minimum cost seed set for competitive social influence," in *Proc. IEEE INFOCOM*, 2016, pp. 1–9.

[25] G. Tong, W. Wu, and D.-Z. Du, "Distributed rumor blocking with multiple positive cascades," *IEEE Trans. Comput. Social Syst.*, vol. 5, no. 2, pp. 468–480, Jun. 2018.

[26] K. Kandhway and J. Kuri, "How to run a campaign: Optimal control of sis and sir information epidemics," *Appl. Math. Comput.*, vol. 231, pp. 79–92, 2014.

[27] G. Tong, W. Wu, S. Tang, and D.-Z. Du, "Adaptive influence maximization in dynamic social networks," *IEEE/ACM Trans. Netw.*, vol. 25, no. 1, pp. 112–125, Feb. 2017.

[28] P. Hui, K. Xu, V. O. Li, J. Crowcroft, V. Latora, and P. Lio, "Selfishness, altruism and message spreading in mobile social networks," in *Proc. IEEE INFOCOM*, 2009, pp. 1–6.

[29] 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, 2015.

[30] D. Yang, G. Xue, X. Fang, and J. Tang, "Incentive mechanisms for crowdsensing: Crowdsourcing with smartphones," *IEEE/ACM Trans. Netw.*, vol. 24, no. 3, pp. 1732–1744, Jun. 2016.

[31] J. L. Z. Cai, M. Yan, and Y. Li, "Using crowdsourced data in location-based social networks to explore influence maximization," in *Proc. IEEE INFOCOM*, 2016, pp. 1–9.

[32] Y. Wang, Z. Cai, Z.-H. Zhan, Y.-J. Gong, and X. Tong, "An optimization and auction-based incentive mechanism to maximize social welfare for mobile crowdsourcing," *IEEE Trans. Comput. Social Syst.*, vol. 6, no. 3, pp. 414–429, Jun. 2019.

[33] R. B. Myerson, *Game Theory*. Cambridge, MA, USA: Harvard University Press, 2013.

[34] V. Krishna, *Auction Theory*. New York, NY, USA: Academic, 2009.

[35] L. Gao, F. Hou, and J. Huang, "Providing long-term participation incentive in participatory sensing," in *Proc. IEEE INFOCOM*, 2015, pp. 2803–2811.

[36] 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.

[37] J. Y. Koh, G. W. Peters, D. Leong, I. Nevat, and W.-C. Wong, "Privacy-aware incentive mechanism for mobile crowd sensing," in *Proc. IEEE Int. Conf. Commun.*, 2017, pp. 1–6.

[38] J. Lin, M. Li, D. Yang, and G. Xue, "Sybil-proof online incentive mechanisms for crowdsensing," in *Proc. IEEE INFOCOM*, 2018, pp. 2438–2446.

[39] T. Ning, Y. Liu, Z. Yang, and H. Wu, "Incentive mechanisms for data dissemination in autonomous mobile social networks," *IEEE Trans. Mobile Comput.*, vol. 16, no. 11, pp. 3084–3099, Nov. 2017.

[40] L. Shi, L. Zhao, G. Zheng, Z. Han, and Y. Ye, "Incentive design for cache-enabled D2D underlaid cellular networks using stackelberg game," *IEEE Trans. Veh. Technol.*, vol. 68, no. 1, pp. 765–779, Jan. 2019.

[41] E. Maskin, "Nash equilibrium and welfare optimality," *Rev. Econ. Studies*, vol. 66, no. 1, pp. 23–38, 1999.

[42] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

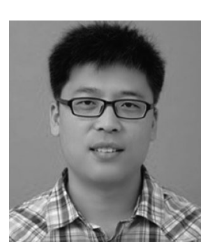
[43] Y. Singer, "Budget feasible mechanisms," in *Proc. IEEE Symp. Found. Comput. Sci.*, 2010, pp. 765–774.

[44] R. Zafarani and H. Liu, "Social computing data repository at ASU," 2009. [Online]. Available: http://socialcomputing.asu.edu

Yaguang Lin received the B.S. degree in computer science and technology from Xi'an Technological University, Xi'an, China, in 2013. He is working toward the Ph.D. degree with the School of Computer Science, Shaanxi Normal University, Xi'an, China. He's also a Visiting Ph.D. Student with the Department of Computer Science, Georgia State University, Atlanta, GA, USA, from 2018. His research interests include information diffusion, rumor blocking, and mobile social networks.

Zhipeng Cai received the B.S. degree from the Beijing Institute of Technology, Beijing, China, and the M.S. and Ph.D. degrees in computing science from the University of Alberta, Edmonton, AB, Canada. He is currently an Associate Professor with the Department of Computer Science, Georgia State University, Atlanta, GA, USA. His research interests include networking, privacy, and big data. He was the recipient of NSF CAREER Award. He is an Editor/Guest Editor for the IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, the IEEE NETWORKING LETTERS, Algorithmica, Theoretical Computer Science, the *Journal of Combinatorial Optimization*, and the IEEE/ACM TRANSACTIONS ON COMPUTATIONAL BIOLOGY AND BIOINFORMATICS.

Xiaoming Wang received the Ph.D. degree in computer software and theory from Northwest University, Xi'an, China, in 2005. He was a Senior Visiting Scholar with the Department of Computer Science, Georgia State University, USA, from 2007 to 2008. He is currently a Professor and the Dean with the School of Computer Science, Shaanxi Normal University, Xi'an, China. He has served as a reviewer/editor for many journals. He authored or coauthored three books and more than 100 papers. His main research interests include information diffusion, mobile social networks, social computing, and ubiquitous computing.



Fei Hao received the Ph.D. degree from Soonchunhyang University, Asan, South Korea, in 2016. He is currently an Associate Professor with the School of Computer Science, Shaanxi Normal University, Xi'an, China. His research interests include social computing, ubiquitous computing, and mobile cloud computing. He was the recipient of five best paper awards from KISM 2012, GreenCom 2013, MUE 2015, UCAWSN 2015, and CUTE 2016. He is a recipient of the IEEE Outstanding Leadership Award at CPSCom 2013 and the IEEE Outstanding Service Awards at SmartData 2017 and DSS 2018.