applied sciences

Article

A Reputation-Based Collaborative User Recruitment Algorithm in Edge-Aided Mobile Crowdsensing

Yang Liu [1], Yong Li [1,*], Wei Cheng [1], Weiguang Wang [1] and Junhua Yang [2]

[1] School of Electronics and Information, Northwestern Polytechnical University, Xi'an 710129, China; hnliuyang90@mail.nwpu.edu.cn (Y.L.)
[2] School of Electronic Engineering, Xi'an University of Posts and Telecommunications, Xi'an 710121, China
* Correspondence: ruikel@nwpu.edu.cn

Abstract: Mobile CrowdSensing (MCS) has become a convenient method for many Internet of Things (IoT) applications in urban scenarios due to the full utilization of the mobility of people and the powerful capabilities of their intelligent devices. Nowadays, edge computing has been introduced into MCS to reduce the time delays and computational complexity in cloud platforms. To improve task completion and coverage rates, how to design a reasonable user recruitment algorithm to find suitable users and take full advantage of edge nodes has raised huge challenges for Mobile CrowdSensing. In this study, we propose a Reputation-based Collaborative User Recruitment algorithm (RCUR) under a certain budget in an edge-aided Mobile CrowdSensing system. We first introduce edge computing into MCS and build an edge-aided MCS system in urban scenarios. Moreover, we analyze the influence of user reputation on user recruitment. Then we establish a user reputation module to deduce the user reputation equation by combining the user's past reputation score with an instantaneous reputation score. Finally, we utilize the sensing ability of edge nodes and design a collaborative sensing method. We use the greedy method to help choose the appropriate users for the tasks. Simulation results compared with the other three algorithms prove that our RCUR approach can significantly achieve better performance in task completion rate and task coverage rate.

Keywords: mobile crowdsensing; edge computing; user reputation; collaborative sensing; user recruitment algorithm

check for updates

Citation: Liu, Y.; Li, Y.; Cheng, W.; Wang, W.; Yang, J. A Reputation-Based Collaborative User Recruitment Algorithm in Edge-Aided Mobile Crowdsensing. *Appl. Sci.* **2023**, *13*, 6040. https://doi.org/10.3390/app13106040

Academic Editors: Juan A. Gómez-Pulido and Antonio Fernández-Caballero

Received: 14 March 2023
Revised: 2 May 2023
Accepted: 13 May 2023
Published: 14 May 2023

1. Introduction

With the rapid development of sensing, computing, and communicating technologies, intelligent devices equipped with multifarious sensors, for example, mobile phones, iPads, and wearable devices have already played a crucial role in human life, which makes the people who take these intelligent devices to become powerful sensing nodes [1–3]. The inherent mobility nature of people has empowered and inspired the people to take part in ubiquitous sensing, and the rich sensing capabilities of sensor-enhanced devices make pervasive computing possible, which stimulates the emergence and promotes the development of an appealing paradigm named Mobile CrowdSensing (MCS) [4]. MCS enables and inspires a vast number of people to sense and contribute data; therefore, it has become a convenient method for many Internet of Things (IoT) applications, such as smart cities [5–7], environmental monitoring [8–12], smart transportation [13,14] and intelligent medicine [15,16], which improves work efficiency and quality of our life.

Compared with traditional sensing technology, the core idea of MCS is to utilize people's mobility and recruit people with intelligent devices to meet the requirements and complete sensing tasks. There is no need to deploy massive static nodes in the sensing areas, which can largely reduce installation costs [17]. Moreover, the widespread popularity of intelligent devices can provide strong guarantees for spatial coverage and task completion quality, even when facing emergencies and unpredictable tasks. Meanwhile, the outstanding computing capabilities of mobile devices can conduct the computation for the cloud platform, which has been widely used in many real-time scenarios, such as autonomous driving [18–20]. However, the inadequate resources of mobile devices are unable to execute the complex computation of large amounts of real-time information, which would lead to the latency of data and information propagation [21]. To address the issues, some researchers introduced Edge Computing into the MCS system, whose basic idea is to add edge nodes, such as smart lampposts [22,23] and base stations [24], with sufficient energy, excellent storage, and computing capabilities at the edge of the network. So sensing data are computed and processed by the closest edge nodes instead of the cloud platform, which is promising to effectively reduce the time delays and computational complexity in cloud platforms. However, how to take full advantage of the edge nodes more reasonably has aroused great interest from researchers [25,26].

Although MCS is an appealing sensing paradigm and shows great potential, it has raised plenty of challenging issues, for example, user recruitment algorithms [27,28], incentive mechanisms [29,30], task allocations [31–33], and data fusion [34,35]. As one of the most vital challenges in MCS, the user recruitment algorithm refers to the problem of how to recruit appropriate participants from extensive potential users to meet the various requirements and complete the tasks under some constraints [36]. More and more researchers have devoted plenty of effort to finding optimal methods to solve this problem by comprehensively considering various factors. In [37], Wu D et al. jointly considered the effects of users' willingness, reputation, and activity, then proposed a user characteristics aware participant selection (UCPS) mechanism to improve the task completion rate and ensure the quality of sensing data in different regions. The results showed that the mechanism could improve the task completion rate and data quality. However, they ignored the situation of potential users in the task area. Taking into account the requirements of both quantity and quality of users' participation, Li Q et al. introduced the reputation evaluation model and proposed a crowdsensing task selection algorithm to encourage participants to make the greatest contributions [38]. Experimental results indicated that the method could effectively ensure the quantity and the quality of users' participation. However, the complex computation would lead to time delays and information propagation. To guarantee the coverage quality of the tasks, from the perspective of the tasks, authors in [39] focused on the task attributes, and divided the whole sensing region by a weighted Voronoi diagram, then proposed a novel willingness-aware user recruitment strategy (WAUR) for MCS. Simulations revealed that the approach could significantly improve the performance compared with other algorithms. While these studies above integrated and interweaved many different variables, however, they ignored the most important part of MCS—the uncertainty of mobile users. The users' random mobility and uneven distribution could not enable the cloud platform to have enough users to perform the tasks, but also would bring about a shortage of users for the tasks when facing emergencies and unpredictable tasks, which leads to an unnecessary accuracy loss. More seriously, some users may upload fake data for compensation which would critically decrease the quality of tasks. Furthermore, although numerous mobile devices can replace traditional sensors, the instability and uncertainty of these mobile users may affect the completion and quality of tasks. Selecting reliable users and completing tasks more effectively has become a huge challenge in MCS.

Hence, to avoid the fraud of users and ensure the task coverage quality and improve the task completion rate, by jointly considering the user reputation and edge computing, we present a reputation-based collaborative user recruitment algorithm (RCUR) for edge-aided Mobile Crowdsensing with the goal of improving the completion rate and maximizing the spatial coverage of tasks under temporal and budget constraints. The main contributions of our work are summarized as follows:

(1) Existing MCS systems in urban scenarios suffer from high delay and limited storage space, which negatively affect task completion rate and spatial coverage. To address these issues, we introduce edge computing into MCS and develop an edge-aided MCS system in urban scenarios. Our system adds an edge layer that efficiently handles the

computing pressure and storage needs while leveraging the powerful sensing abilities of edge nodes. Moreover, the user recruitment problem is formulated with spatial and budget constraints to improve completion rates and maximize spatial coverage.

(2) Prior methods for user selection in MCS systems fail to consider user reliability and credibility adequately. To address this gap, we establish a user reputation module to evaluate the user's past reputation score based on previous performance. Then we jointly consider the distance and coverage ability of the user to calculate the instantaneous reputation score, which is combined with the user's past reputation score to generate an overall reputation equation. Our cloud platform selects the proper users according to their reputation scores to complete the tasks efficiently and qualitatively.

(3) In order to achieve a better task completion rate and maximum spatial coverage, we propose a reputation-based collaborative user recruitment algorithm (RCUR) in an edge-aided Mobile CrowdSensing system under certain budgets. Our innovative approach leverages the user reputation method to identify suitable participants who can guarantee the coverage quality of the tasks. Additionally, we use collaborative sensing to select the edge nodes to finish the tasks and ensure coverage when lacking potential users. Moreover, we design a greedy method to recruit the optimal participant for each task. Our simulations and experimental results verify that our proposed RCUR algorithm is outstanding compared with other methods in terms of task completion rate and spatial coverage.

The rest of the paper is organized as follows. In Section 2, we review and summarize related work. Section 3 presents our system model and problem formulation. In Section 4, we propose a reputation-based collaborative user recruitment algorithm (RCUR) under budget constraints. Simulation results compared with other methods are shown in Section 5. Finally, we conclude our work in Section 6.

2. Related Work

Recently, user recruitment algorithms have attracted the attention of many scholars, who have proposed extensive promising studies in MCS systems. Most of them are devoted to finding the optimal solutions to recruit the most appropriate participant to complete the tasks. Guo B et al. [40] focused on data diversity and proposed a unified visual crowdsensing framework called UtiPay based on the microscopic and macroscopic visual task types to improve the quality of group sensing data. In [41], to complete the task with the minimum cost, Zhou et al. defined the "t-sweep k-coverage" crowdsensing tasks and designed a user recruitment algorithm based on the theory of greedy strategy. After simulation experiments on real data sets, the superiority and effectiveness of this method were proved. Wang et al. [42] fully exploited the content information together with the context information to model workers' preferences on tasks accurately. Then they proposed a novel personalized task-oriented worker recruitment mechanism for mobile crowdsensing based on a careful characterization of workers' preferences.

One of the most important challenges in MCS is the coverage quality for some applications. Especially in urban scenarios, coverage has become one of the criteria to measure the completion quality of tasks. In [43], Zhang et al. proposed a novel user recruitment that only selected a subset of mobile participants to achieve the maximum spatial coverage with a constrained budget. The results revealed that the algorithm achieved a near-optimal solution compared with the brute-force search results. However, this method would spend too much running time and cost a lot for the platform. Alagha A et al. [44] jointly considered the parameters, such as user localization, mobility traces, and reputation, to design a two-steps novel Stable Data-based Recruitment System (SDRS) for localization tasks. Meanwhile, they took into account the range-free sensors and the mobility of workers to improve the coverage quality of tasks. The results indicated that the proposed method could attain a speedy localization with better quality compared with other benchmarks. Yucel F et al. [45] tried to tackle the issues of finding task assignments that could make the optimal tradeoff between coverage-aware preferences of service requesters and profit-based preferences of workers under the budget constraints in MCS. The simulations showed that the proposed algorithms mostly provided substantially better task assignments in terms of user happiness and coverage quality.

With the continuous development and maturity of edge computing technology, to further improve the task completion rate and reduce the load of the cloud platform, many researchers have introduced edge computing into MCS. Ma L et al. [46] proposed two privacy-preserving reputation management schemes for edge computing enhanced MCS to preserve privacy and deal with malicious participants simultaneously. The experiments demonstrated that both schemes achieved high-cost efficiency. In [47], Zhang Y et al. studied dynamic user recruitment under the scenarios for different sensing tasks. For the long-duration task and short-duration tasks, they presented two different user recruitment algorithms with edge-aided MobileCrowdsensing. The experiments showed the proposed user recruitment could achieve better validity and reliability. To solve the problem that user recruitment failed to effectively address the task requirements or the relevant maximization and diversification, Xiong J et al. [48] designed a task-oriented user selection incentive mechanism (TRIM) in an effort toward a task-centered design framework in edge-aided MCS. The simulation results indicated that TRIM could achieve feasible and efficient user selection.

However, the previous works have presented outstanding simulation results, and there are several shortcomings related to user recruitment in mobile crowdsensing (MCS) systems. Many studies focus on finding optimal solutions to recruit appropriate participants to complete tasks but do not consider the suitability of users' characteristics or the potential of edge computing resources. Users' historical performance usually indicates enthusiasm and reliability for the tasks, which will directly affect the completion and coverage quality. Some studies aim to improve coverage quality in urban scenarios, but their methods are time-consuming or costly. Others introduce edge computing but do not fully utilize its sensing and communicating capabilities. They ignore the abundant resources such as sensing, and communicating capabilities, which can also be used as a powerful static sensing node. Hence, the edge-aided MCS has significant potential space to improve the flexibility and efficiency of the MCS system. To address all aforementioned issues, we propose an edge-aided MCS system in urban scenarios with a user reputation module to evaluate historical performance and a reputation-based collaborative user recruitment algorithm (RCUR) that efficiently improves task completion rate and achieves optimal coverage under budget constraints, which is promising in addressing the identified shortcomings and utilizing the potential of edge computing resources.

3. System Model and Problem Formulation

3.1. System Model

The MCS system generally is comprised of mobile users U and the cloud platform P. However, in a dynamically changing environment, especially an urban environment, except for a large number of mobile people, there are still quite a few pre-deployed static nodes with powerful sensing, collecting, computing, and communicating capabilities, such as smart lampposts, signal base stations, smart camera and so on, which also can collect the sensing information. Hence, we not only utilize the mobile users to gather relevant information and complete the tasks; furthermore, we take advantage of these edge nodes as a supplement to build a new CrowdSensing system. As shown in Figure 1, our system consists of a three-layer hierarchical network.

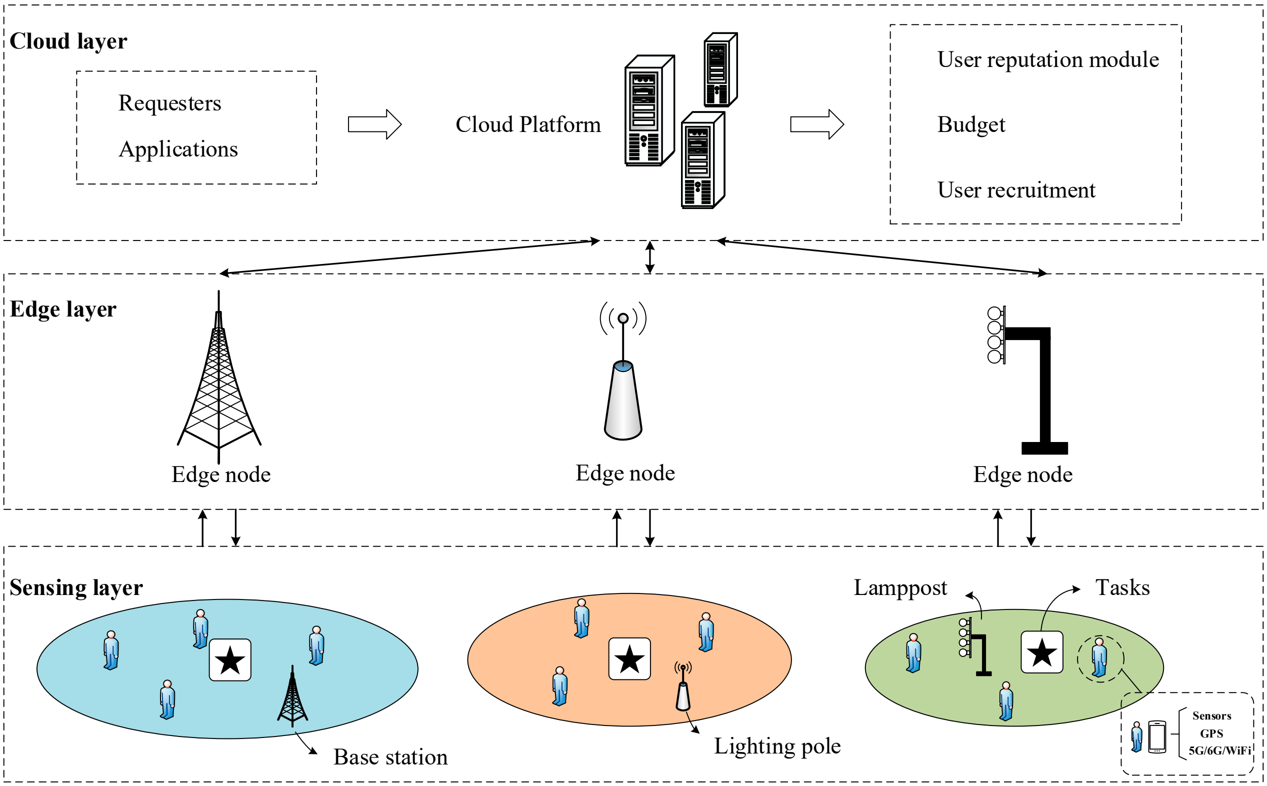


Figure 1. Edge-aided MCS system.

(i) **Cloud layer**. The Cloud layer is responsible for the publication of the tasks, and it provides computing and analyzing resources for collecting data. As a centralized controller, the cloud platform gets the tasks from some applications or requesters, then publishes these tasks to the users. Meanwhile, the cloud platform will enable interactions with the other two layers and manage the user's personal information, including historical information, locations, and reputations. In our system, the cloud platform will establish a user reputation module to select the proper users based on their historical performance to complete the tasks efficiently and qualitatively. Additionally, the cloud layer is also a data processing center that will analyze and summarize these sensed data, then package it to the applications and requesters.

(ii) **Edge layer**. As depicted in Figure 1, the edge layer is located between the cloud platform and the mobile sensing end devices. Due to powerful computing and communicating capabilities, the pre-deployed static nodes would be the perfect edge nodes in the CrowdSensing system, which can support and enhance the performance of the mobile end devices. In the urban environment, edge nodes can be replaced by smart static nodes such as smart lampposts and base stations. Heterogeneous mobile devices would be connected to the cloud platform through the edge layer closer to the users in 5G/6G or WiFi, which can release the computing pressure on the cloud, reduce the delay and increase the storage space of information.

(iii) **Sensing layer**. As the most important and basic layer of the CrowdSensing system, mobile users who take the intelligent terminals with many embedded sensors are the key component in the sensing layer and play a fundamental role in sensing and collecting various data at a certain time in specific locations. We assume that all users volunteer to participate in tasks and that sensed data are accurate. The random mobility of mobile users greatly enhances the flexibility of ubiquitous sensing, and the static nodes can ensure the coverage of the sensing network. The collaboration and interaction between mobile and static sensors could promote the performance and quality of sensing and data processing.

3.2. Problem Formulation

In our system, the cloud platform P gets the tasks θ_{total}, including some requirements from some applications or the requesters, such as traffic monitoring, noise conditions, and air conditions. Then the cloud platform P slices the massive tasks into several sub-tasks $\theta = (\theta_1, \theta_2, \ldots, \theta_n)$ on the basis of the time limitations of the sensing tasks, then publishes these sub-tasks. Usually, the detailed information about the tasks θ can be expressed by a multi-element tuple $F_\theta = \left(\theta_j^{typ}, l_{\theta_j}, t_{\theta_j}, R_{\theta_j}, R\omega_{\theta_j}\right)$, which indicates the type, locations, time limitations, coverage radius, and task reward. The time limitations of tasks t_{θ_j} are usually divided into some timeslots to be performed. Based on the detailed information about the tasks θ, mobile users $U = \{u_1, u_2, \ldots, u_U\}$ will decide whether to take part in these tasks and choose what kinds of tasks to participate in. The relevant details of the mobile users $U = \{u_1, u_2, \ldots, u_U\}$ can be described as $F_u = \left(l_{u_i}, R_{u_i}, \sigma_{u_i}, C_{u_i}^{\theta_j}\right)$, where l_{u_i} is the real-time location, R_{u_i} is the sensing radius, σ_{u_i} is the reputation scores, and $C_{u_i}^{\theta_j}$ is the cost for task θ_j. When users U accomplish a task, the reputation module will evaluate the performance of the users and update their reputation score. Simultaneously, in the urban environment, there are some smart static nodes S_n, such as smart lampposts and base stations, which can replace the edge nodes. The smart static nodes generally perform other work or just stay asleep to reduce the energy cost. In special circumstances, these static nodes can wake up rapidly and start to work according to the requirements. S_n can be represented as $S_n = \left(l_{S_n}, R_{S_n}, C_{S_n}^{\theta_j}\right)$, where l_{S_n} denotes the location of the static node. R_{S_n} denotes the sensing radius of the static node and $C_{S_n}^{\theta_j}$ represents the sensing cost of the static node for task θ_j.

In general, the cloud platform has a certain budget B, and the tasks will have spatial and temporal requirements. Specifically, t_{θ_j} represents a certain range of the sensing time and R_{θ_j} is denoted as the certain range of the sensing radius for the tasks. Meanwhile, there is a reward $R\omega_{\theta_j}$ for the user or static node who accomplishes the task and $\sum R\omega_{\theta_j} \leq B$. The cloud platform P evaluates the users' reputation scores and other information to recruit the most appropriate user to execute the task. Aiming at maximizing the spatial and temporal coverage for the sensing tasks, there would be two kinds of approaches to achieve the goal. If the cloud platform P can not find the users to complete the task efficiently and qualitatively, then it will choose a static node that meets the requirements to finish the task. Let $A_{\theta_j}^{\text{cov}}$ denotes the expected sensing coverage area of the task θ_j. $A_{u_i,\theta_j}^{\text{cov}}$ and $A_{S_{n_i},\theta_j}^{\text{cov}}$ are the coverage area which the user u_i sense and the static node S_{n_i}sense, respectively, for task θ_j. In certain time limitations t_{θ_j},$A_{u_i,\theta_j}^{\text{cov}} + A_{S_{n_i},\theta_j}^{\text{cov}}$ denotes the sensed area by the user and static node for the task θ_j. Therefore, the task spatial coverage percentage $\Re_{\theta_j}$for task $\theta_j$can be calculated by:

$$\Re_{\theta_j} = \frac{A_{u_i,\theta_j}^{\text{cov}} + A_{S_{n_i},\theta_j}^{\text{cov}}}{A_{\theta_j}^{\text{cov}}}. \tag{1}$$

As a result, in order to achieve the goal of maximizing the sensing spatial coverage area of the tasks, the problem of user recruitment can be expressed as:

$$Maximize \sum \Re_{\theta_j} \tag{2}$$

$$s.t.$$

$$(1) C_{u_i}^{\theta_j} + C_{S_n}^{\theta_j} \leq R\omega_{\theta_j} \tag{3}$$

$$(2) t_{u_i} \in t_{\theta_j} \tag{4}$$

$$(3) t_{S_{n_i}} \in t_{\theta_j} \tag{5}$$

where t_{u_i} and $t_{S_{n_i}}$, respectively, denote the time that the user $u_i$and static node S_n start to conduct the sensing work for task θ_j. Specifically, with the purpose of maximizing the sensing spatial coverage area of the tasks, how to recruit the optimal user and utilize collaborative sensing reasonably to complete the tasks will be a great challenge.

4. Reputation-Based Collaborative User Recruitment Algorithm

In this section, we try to analyze the problem of sensing spatial and temporal coverage for tasks of MCS to ensure quality in dynamically changing environments, especially urban scenarios. The quality of the information sensed by users is always the most important parameter to the sensing tasks, which affects excellent quality and provides meaningful guarantees for task completion. Therefore, how to guarantee the task completion rate and meanwhile improve the spatial and temporal coverage rate has raised a new challenge. First, a user reputation module is established to calculate users' reputation scores based on their historical performance and select the proper users according to their instantaneous reputation scores to complete the tasks efficiently and qualitatively. Then we analyze other recruitment indicators and propose collaborative sensing user recruitment in detail. Finally, we design a reputation-based collaborative user recruitment algorithm to guarantee the task completion rate and maximize the sensing spatial coverage for tasks in MCS. Table 1 contains the key symbols and interpretations used in the algorithm.

Table 1. Symbols and interpretations.

Symbols	Description
P	Cloud Platform
U	Users
θ	Tasks
S_n	Static nodes
$\sigma_{u_i}(t)$	User reputation
B	Total budget
$\Re_{\theta_j}$	Task spatial coverage percentage
D	Distance factor
C	Cost for task
f	Fitness function
$\alpha, \beta, \omega, \lambda$ η, μ, φ	Balancing coefficients

4.1. User Reputation

Intuitively, reputation is a key indicator for tasks to track users' historical performance and assess the accuracy of these sensing data [49]. In MCS, because of the users' random mobility, users can easily choose to participate in different tasks anywhere and at any time. However, when the cloud platform P recruits users for tasks, past reputation scores only represent the users' previous performance and credibility. There are other parameters that will have an influence on recruiting the appropriate users to complete the tasks, for instance, distance, time limitation, and so on. Therefore, the user reputation score should be comprehensively calculated based on past performance and current characteristics.

In our system, the cloud platform P publishes $n$sensing tasks $\theta = (\theta_1, \theta_2, \ldots, \theta_n)$. The detailed information can be described as $F_\theta = \left(\theta_j^{typ}, l_{\theta_j}, t_{\theta_j}, R_{\theta_j}, R\omega_{\theta_j}\right)$. Then the cloud platform P collects the information of all users and establishes a user reputation module to evaluate the historical performance of the users and calculate their reputation scores, which aims at selecting the proper users and promoting the whole performance on sensing and computing for tasks. With increased participation time, users would have massive records of their historical performance. However, as the former records will diminish values over time, these outdated data will be worthless and no longer reflect the real situation of users at present. Consequently, we should pay more attention to the recent records.

Therefore, we introduce and improve Equation (6) [37] to estimate the users' reputation scores based on their past behavior. We give more weight to the last five user records and deduce the user's u_i past reputation scores $\sigma_{u_i}(t^-)$ before time t^- as follows:

$$\sigma_{u_i}(t^-) = \sum_{N_{Complete}=1}^{\min(N_{Complete}^{Max},5)} \left(\frac{1}{2}\right) \sigma_{u_i,score}^{N_{Complete}}, \quad (6)$$

where $N_{Complete}$ denotes the number of tasks completed by user u_i and $\sigma_{u_i,score}^{N_{Complete}}$ is the score the task θ calculates after user u_i accomplishes it. We comprehensively consider the spatial and temporal coverage for the tasks, so $\sigma_{u_i,score}^{N_{Complete}}$ is defined by

$$\sigma_{u_i,score}^{N_{Complete}} = \alpha \frac{A_{u_i,\theta_j}^{\text{cov}}}{A_{\theta_j}^{\text{cov}}} + (1-\alpha)\frac{T_{u_i,\theta_j}}{T_{\theta_j}}, \quad (7)$$

where α denotes the weight coefficients ranging from 0 to 1. $A_{\theta_j}^{\text{cov}}$ represents the expected sensing coverage area of the task θ_j and $A_{u_i,\theta_j}^{\text{cov}}$ is the coverage area that the user u_i senses for the task θ_j. T_{u_i,θ_j} denotes the working time of user u_i for the task θ_j and T_{θ_j} is the sensing duration of the task θ_j. A larger value of $\sigma_{u_i,score}^{N_{Complete}}$ signifies the excellent performance of the user for the task, which also shows higher reliability and credibility.

As we know, when the cloud platform P recruits users for task θ_j, the user reputation module first calculates all the participants' past reputation scores. Although past reputation scores play an important role in users' reliability and credibility, the cloud platform will also consider real-time factors, for instance, distance and time limitations. We assume that $\sigma_{u_i}^{inst}(t)$ denotes the user's instantaneous reputation score, which is limited by some restrictions, such as the locations and sensing radius. So we can attain the instantaneous reputation score $\sigma_{u_i}^{inst}(t)$ of user u_i for task θ_j as follow:

$$\sigma_{u_i}^{inst}(t) = \beta D_i + (1-\beta)\frac{R_{u_i}}{R_{\theta_j}}, \quad (8)$$

where β is the weight coefficients ranging from 0 to 1. R_{u_i} is the sensing radius of user u_i, and R_{θ_j} represents the coverage radius of task θ_j. So R_{u_i}/R_{θ_j} indicates the coverage ability of user u_i for task θ_j. D_i is the distance factor, which can be represented by

$$D_i = 1 - \left(D_{u_i,\theta_j} \middle/ D_{\max}\right), \quad (9)$$

where D_{u_i,θ_j} represents the distance between user u_i and task θ_j, and $D_{\max}$ equals to $R_{u_i} + R_{\theta_j}$, which is the maximum distance the users can provide the coverage for the task. Through Equation (8), we know that the instantaneous reputation score $\sigma_{u_i}^{inst}(t)$ affects whether a user would complete the task with high coverage and quality.

In addition, we consider all the factors to deduce the user u_i reputation score $\sigma_{u_i}(t)$ equation at time t as follows:

$$\sigma_{u_i}(t) = \omega\sigma_{u_i}(t^-) + (1-\omega)\sigma_{u_i}^{inst}(t), \quad (10)$$

where ω represents the balancing parameter ranging from 0 to 1.

4.2. Cost for Tasks

Generally speaking, in MCS, for any sensing tasks, there will be different constraints. For example, air quality sensing needs sensing data in different periods at the same locations, and autonomous driving will arrange the route in advance based on real-time traffic conditions. In order to meet the various requirements of the tasks, we assume that each task in each timeslot will recruit one user to sense and collect data, which can achieve the goal of more effective sensing and maximum coverage.

No matter what kinds of nodes, the mobile users or the edge nodes, the participants who are recruited by the cloud platform P execute the whole sensing process and complete the tasks, which would have a cost, such as time and energy loss. Actually, when the users or edge nodes perform the tasks, the longer the execution time is, the more energy will cost. If the tasks need longer sensing time, the users or edge nodes will spend more time and energy executing the tasks. Although participants are volunteers, they will ask for a reward to cover their costs. In our system model, we assume that when each participant acquires the information about task θ_j, there will be a cost representing the loss which is proportional to the time they will spend performing the tasks. $C_{u_i}^{\theta_j}$ is the cost of the user u_i for task θ_j and $C_{S_n}^{\theta_j}$ represents the sensing cost of the edge node S_n for task θ_j. For every task, it will have a reward for the participants to encourage their contributions when they complete the tasks and collect these expected data. $R\omega_{\theta_j}$ is described as the reward of the task θ_j.

In order to keep the budget balanced for each task, the cloud platform P will recruit participants whose costs are below the reward of the task. Therefore, we can attain the relationship between the costs of the participants and the tasks as follow:

$$C_{u_i}^{\theta_j} \le R\omega_{\theta_j}, \tag{11}$$

$$C_{S_n}^{\theta_j} \le R\omega_{\theta_j}. \tag{12}$$

4.3. Reputation-Based User Recruitment

In our MCS system, we assume that in urban scenarios, there are plentiful mobile users with intelligent devices randomly distributed in the city. They move at a constant speed v to roam around the task locations. After the cloud platform P publishes the tasks θ_{total} from some applications or the requesters, these mobile users U volunteer to take part in these tasks, and they will choose what kinds of tasks to participate in according to the information of the tasks. Hence, for each task θ_j, the cloud platform P will select the users who are close to the task in time and space and whose costs are less than the reward as the candidates. Then it will form a subset of users S_{Can} for the task θ_j and choose the most appropriate user to complete the task.

Then the reputation module will evaluate the performance of the users in S_{Can} and update their past reputation scores $\sigma_{u_i}(t^-)$. The higher value $\sigma_{u_i}(t^-)$ means that the users u_i have outstanding performance in the previous tasks with higher quality, which leads to a higher possibility of being recruited for the next task. Through Equation (6), we focus on the performance of the last five tasks the user participates in to attain the past reputation score. Additionally, we consider other real-time factors, spatial and temporal constraints. We calculate the user instantaneous reputation score $\sigma_{u_i}^{inst}(t)$, which is related to locations and sensing radius. Finally, the cloud platform can attain the user u_i reputation score $\sigma_{u_i}(t)$ for the task θ_j at time t.

In order to maximize profits, we propose a Reputation-Based User Recruitment (RBUR). The cloud platform P prefers to recruit users who can complete the tasks efficiently and achieve the maximum coverage qualitatively but require less reward for their costs. Therefore, we establish a fitness function that is based on the user reputation and cost for the task, which is given as follows:

$$f_u = \lambda \sigma_{u_i}(t) + (1-\lambda)\left(1 - \frac{C_{u_i}^{\theta_j}}{R\omega_{\theta_j}}\right), \tag{13}$$

where λ represents the balancing parameter ranging from 0 to 1, and $C_{u_i}^{\theta_j} \leq R\omega_{\theta_j}$. The fitness function f_u can provide strong guarantees by adopting the user reputation method with the goal of improving the task completion rate and achieving higher coverage quality, meanwhile maintaining the proper tradeoff between the cost to the user and the reward of the task.

4.4. Collaborative Sensing Method

Random movement by the vast number of mobile users in the city brings great opportunity for MCS, but the significance of user mobility is two-fold. It makes ubiquitous sensing possible, but it may lead to the uneven distribution of users in some regions. For instance, because of the random movement of users, some sensing areas may have insufficient or even very few users, which would prevent the cloud platform P from recruiting a suitable user to finish the task. In this case, pre-deployed static nodes with powerful sensing, collecting, computing, and communicating capabilities in the city can supplement and collect the sensing information for the tasks.

In our system, there are some pre-deployed smart static nodes S_n that generally perform other work or just stay asleep. When the cloud platform P is unable to find a suitable user to execute the task, these static nodes can be the best substitute to sense the expected data and achieve optimal coverage collaboratively. As these static nodes S_n usually have more powerful sensing ability and radius than mobile users, we comprehensively consider three factors: location, sensing radius, and cost. Hence, we deduce the fitness function for the task to recruit the optimal static node, which is described as follows:

$$f_{S_n} = \eta D_{S_n} + \mu \frac{R_{S_n}}{R_{\theta_j}} + \varphi \left(1 - \frac{C_{S_n}^{\theta_j}}{R\omega_{\theta_j}} \right), \tag{14}$$

where D_{S_n} is the distance factor for the static node S_n similar to D_i. R_{S_n}/R_{θ_j} represents the coverage ability for S_n. η, μ and φ are the balancing coefficients and $\eta + \mu + \varphi = 1$. In addition, $C_{S_n}^{\theta_j} \leq R\omega_{\theta_j}$.

4.5. Process of Reputation-Based Collaborative User Recruitment Algorithm

To help the cloud platform find the most suitable participant to accomplish the task efficiently and qualitatively, we propose a reputation-based collaborative user recruitment algorithm (RCUR) by introducing the user reputation method and collaborative sensing with the purpose of achieving a higher task completion rate and bigger coverage. It is a task-centered algorithm to improve the benefits of tasks and cloud platforms. Then we apply the greedy algorithm to help us solve the problem. The specific process of the RCUR algorithm is described as follows and shown in Algorithm 1.

After the cloud platform P publishes the tasks $\theta = (\theta_1, \theta_2, \ldots, \theta_j)$ from some applications or the requesters, these mobile users U volunteer to take part in these tasks, and they will choose what kinds of tasks to participate in according to the information provided on the tasks. According to Equation (2), in order to achieve the goal of maximizing the sensing spatial coverage area of the tasks, the problem of user recruitment can be expressed as $Maximize \sum \Re_{\theta_j}$. However, it is difficult to find the optimal user set for the whole task. Hence, the greedy algorithm is used to solve this issue. We divide the whole problem into several sub-problems and find the optimal solution for each sub-problem. In our system, for each task θ_j, the cloud platform P will select the users who are close to the task in time and space and whose costs are less than the reward as the candidates. It will form a subset of users $S_{Can} = (u_1, u_2, \ldots, u_x)$ for the task θ_j. Then the user reputation module will evaluate the performance of the users in S_{Can} and update their past reputation scores $\sigma_{u_i}(t^-)$. Then P calculates the user's instantaneous reputation score $\sigma_{u_i}^{inst}(t)$. Finally, P can attain the user u_i reputation score $\sigma_{u_i}(t)$ for the task θ_j at time t.

Algorithm 1: A reputation-based collaborative User Recruitment algorithm (RCUR) in edge-aided MCS

Input: input U, θ
Output:
for u in U **do**
  $D_{u_i,\theta_j} \leq D_{\max}, t_{u_i} \in t_{\theta_j}$
  form the candidate set $S_{Can} = (u_1, u_2, \ldots, u_x)$
end
for u in S_{Can} **do**
  Update their past reputation scores $\sigma_{u_i}(t^-)$
  Calculate the user instantaneous reputation score $\sigma_{u_i}^{inst}(t)$
  Get the user u_i reputation score $\sigma_{u_i}(t)$
  Calculate f_u of $u \in S_{Can}$
  Select $u_i \in S_{Can}$ who has max f_u
end
if $S_{Can} = \varnothing$
  Calculate f_{S_n} of S_n
  Select S_n that Maximize f_{S_n}
end

Next, for each task θ_j, we establish a fitness function f_u that is based on the user reputation and cost for the task to recruit the most suitable user to meet the requirements of the task. However, in some areas with insufficient or very few users, we further utilize the static nodes to supplement and replace the users to collect the sensing information for the tasks. After that, we comprehensively consider the locations, sensing radius, and cost, then deduce the fitness function f_{S_n} for the task to recruit the optimal static node.

In different conditions, our proposed RCUR algorithm presents two fitness functions f_u and f_{S_n}, then selects the most suitable participant based on the highest value of the fitness function to accomplish the tasks. For all tasks, the greedy algorithm is adopted to help the cloud platform select the most suitable user or edge node for each task. When all tasks time end, the user or the edge node for each task will form the solution set for all tasks.

5. Simulation and Numerical Results

In this section, extensive simulations are performed to verify our proposed reputation-based user recruitment algorithm (RBUR) and reputation-based collaborative user recruitment algorithm (RCUR), then compare the performance with the other three benchmark methods. We apply the RCUR algorithm to the custom simulator, which is called CrowdSenSim [50], to help us conduct the experiments in realistic urban scenarios. In our simulation, we select a rectangular area of 10 km × 10 km in the real city of Ottawa as the experimental space similar to Figure 2, and there are three important components which are users, tasks, and edge nodes. The mobile users who take the intelligent devices are randomly distributed and wander along the streets in the city at a speed of $1m/s$ in this area. The number of mobile users ranges from 5000 to 25,000, and the relevant details $F_u = \left(l_{u_i}, R_{u_i}, \sigma_{u_i}, C_{u_i}^{\theta_j}\right)$ of the mobile users U will be set up randomly before the experiments start. In the beginning, the users' reputation scores are the same. As a user completes tasks, the value of their reputation score will change based on the quality of the completed tasks. Meanwhile, there are some edge nodes pre-deployed along the street in this area, and we set the key parameters according to [22]. Then the cloud platform will publish a certain number of tasks at different locations with different budgets in this area, which varies $[50, 250]$ and divide each task into 10 timeslots, in which the cloud platform will select the best suitable user or edge node based on the fitness function to complete the tasks. We list the detailed experimental settings in Table 2.

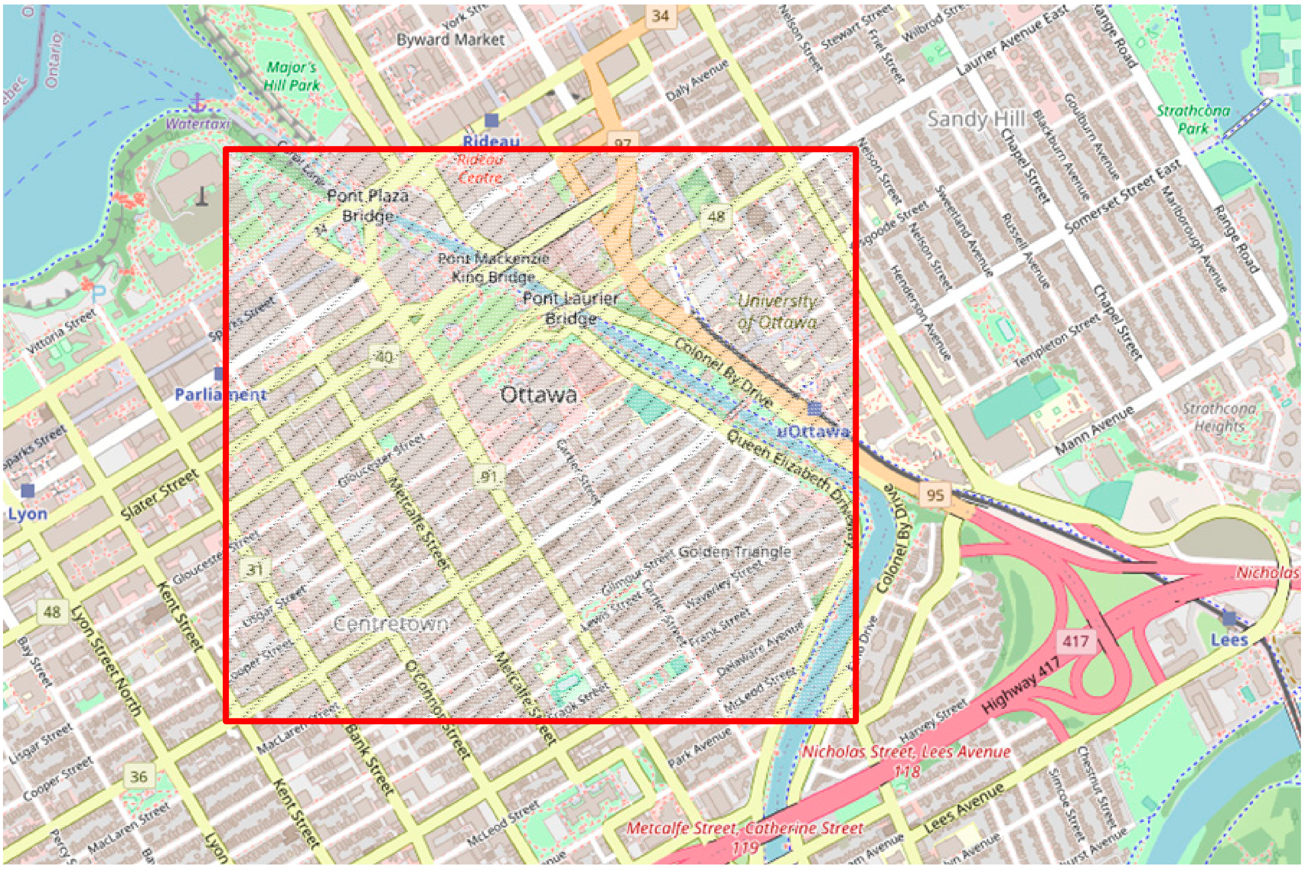


Figure 2. Sensing area.

Table 2. Experimental settings.

Parameters		Value
Number of users		[5000, 25,000]
Number of tasks		[50, 250]
Coverage radius of task		[20, 50]
Number of edge nodes		[30, 70]
Budget		[100, 200]
Initial user reputation		[0.2, 1.0]
Cost		[0, 2.0]
Sensing radius of user		[5, 25]
Sensing radius of edge nodes		[20, 30]
Evaluation period		8:00 AM–20:00 PM
Task duration		30 min
Timeslot duration		3 min
Balancing coefficient	$\alpha, \beta, \omega, \lambda$	0.6, 0.6, 0.5, 0.7
	η, μ, φ	0.3, 0.4, 0.3

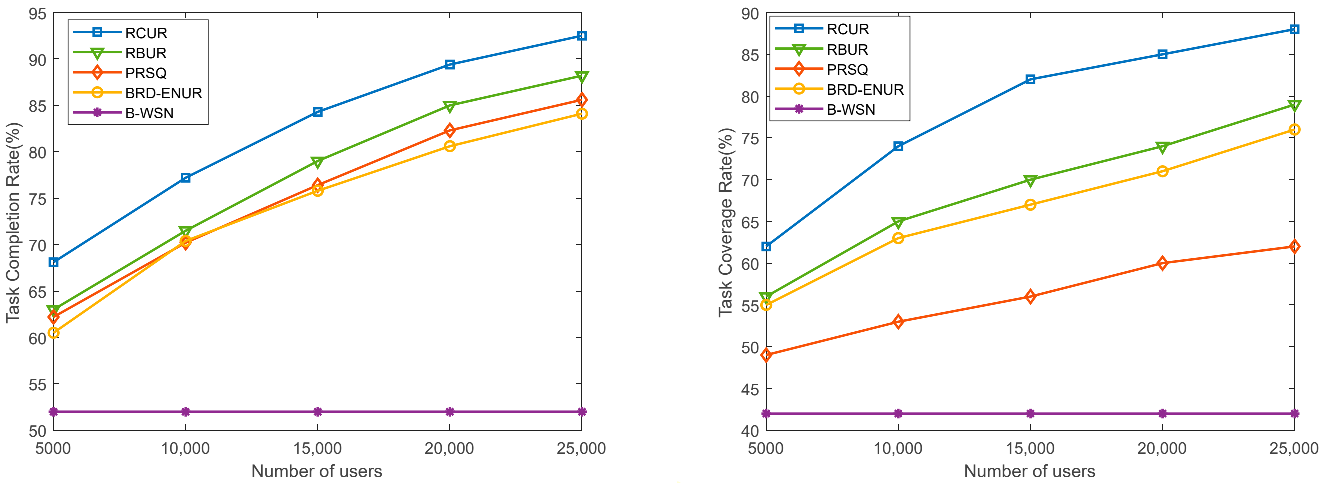
In our study, we choose three benchmark algorithms, including a participant recruitment method aiming at service quality (PRSQ) [51], a budget re-distribution algorithm for edge nodes user recruitment (BRD-ENUR) [47], and a basic wireless sensor network (WSN) sensing method (B-WSN). Then we compare them with our proposed RBUR and RCUR algorithms to verify their effectiveness and efficiency. The PRSQ algorithm constructed a quality of service model based on the accumulated reputation and willingness of participants, then selected the most suitable participants to ensure the QoS. The BRD-ENUR introduced edge computing into MCS. Then it estimated the quality of sensing data according to the participant's reputation and proposed the algorithm to recruit users statically. B-WSN was the fundamental sensing algorithm and chose the static nodes to complete the tasks. The differences between these five algorithms are listed in Table 3. Finally, we perform the experiments over 100 runs and obtain the average values to compare these five algorithms in some metrics, such as the task completion rate, task coverage rate, etc. As we know, the tasks will be divided into 10 timeslots. We assume that when more than half of the timeslots of one task are finished by the participants, we can define the task as completed. So we can calculate the overall task completion rate based on the completion of the task's timeslots. Moreover, according to Equation (1), the coverage rate of one task can be defined as the value of the sensed coverage area by user or edge node divided by the expected sensing coverage area of the task in certain time limitations.

Table 3. Differences between these four algorithms.

Algorithm	Sensing Participants	Key Factor	Advantage
RBUR	Mobile users	Reputation	Ensure sensing task quality on a limited budget
RCUR	Mobile users, edge nodes	Reputation	Mobile users and edge nodes collaboratively sense the whole task
PRSQ	Mobile users	Reputation Willingness	Ensure sensing task quality
BRD-ENUR	Mobile users	Reputation	Introduce edge nodes to reduce the time delays and computational complexity
B-WSN	edge nodes	Distance	Edge nodes have bigger coverage

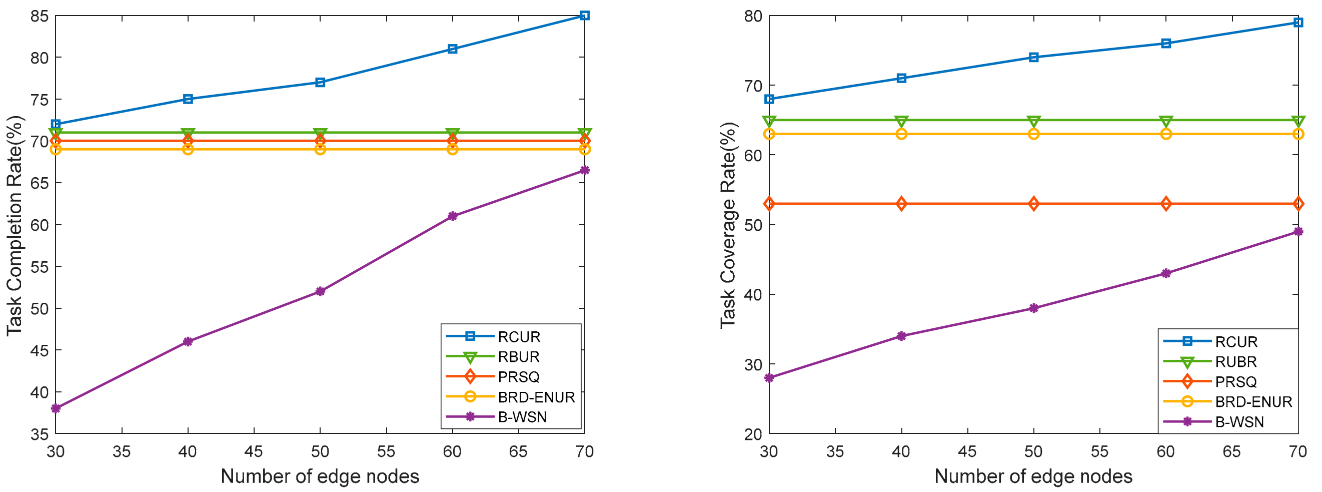
In Figure 3, we evaluate the performance of these five algorithms under the various number of users when we set the tasks = 100, edge nodes = 50, and budget = 150. From Figure 3a, we compare the task completion rate and find that our proposed RCUR is more outstanding than the other three benchmark algorithms. The RBUR is closer but better than the PRSQ and BRD-ENUR. In contrast, B-WSN maintains a relatively low task completion rate. Figure 3b demonstrates that our RCUR also can achieve a higher task coverage rate than other algorithms. By taking into account multiple factors, the RBUR has a better result than the PRSQ and BRD-ENUR. Since B-WSN depends on the sensing and cover ability of the edge nodes, which are less numerous than mobile users, it only selects edge nodes to sense these data and complete tasks. As the number of mobile users increases, MCS algorithms will have a larger user pool to find the most suitable user to complete each task. The more mobile users that participate, the higher possibility the tasks can be completed. Even more, a more covered area can be attained. As a result, when the number of users ranges from 5000 to 25,000, the performance of three algorithms in MCS shows an upward trend both in the task completion rate and the task coverage rate. However, our proposed RCUR approach not only recruits the appropriate mobile user to finish tasks but also utilizes the edge nodes as a supplement when there are no available users in some sensing areas. Hence, the performance of RCUR is better than others.

In Figure 4, when we set tasks = 100, users = 10,000, and budget = 150, we tried to discuss how the task completion rate and task coverage rate change when the number of edge nodes increases. Figure 4 illustrates that there is a huge difference between our proposed RBUR, RCUR, and the other three algorithms. Although BRD-ENUR introduces edge nodes into the MCS system, it only takes these nodes as edge platforms which can reduce storage and computing load in the cloud platform. RCUR and B-WSN make full use of the enhanced sensing ability of these edge nodes to sense the data and cover the sensing areas. Therefore, when there are more and more edge nodes in the sensing areas, each task will have more options to find the optional solution, which can give rise to a great improvement in the task completion rate and task coverage rate. From Figure 4, we can see that when the edge nodes are equal to 30, RUCR is slightly better than PRSQ and BRD-ENUR both in the task completion rate and task coverage rate. The RBUR remains unchanged, while the B-WSN stays at a low value. With the number of edge nodes growing, Figure 4a indicates that RCUR is improved from 72% to 85%, and B-WSN has a great improvement from 38% to 66% in task completion rate. The reason is that while our RCUR takes the edge nodes as a collaborative sensing method, it mainly recruits mobile users to finish the tasks. Consequently, it does not attain much improvement over the B-WSN. In Figure 4b, the task coverage rates of RCUR and B-WSN also present an increasing trend, and our RCUR always outperforms other algorithms.



(**a**) (**b**)

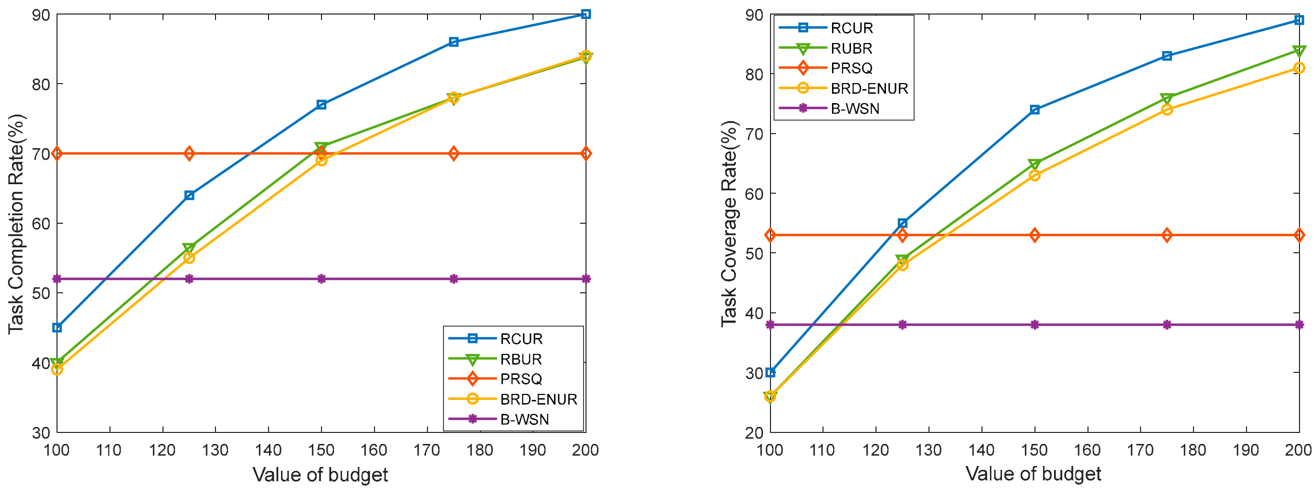
Figure 3. Performances under various numbers of users. (**a**) Task Completion Rate; (**b**) Task Coverage Rate.



(**a**) (**b**)

Figure 4. Performances under various numbers of edge nodes. (**a**) Task Completion Rate; (**b**) Task Coverage Rate.

When we set tasks = 100, users = 10,000, and edge nodes = 50, it can be seen from Figure 5 that the task completion rate and task coverage rate increase substantially with the total budget growing. Since PRSQ and B-WSN do not consider the budget, as the value of the total budget changes, the performances of the task completion rate and task coverage rate of PRSQ and B-WSN remain constant. While our RBUR, RUCR, and BRD-ENUR take the budget as an important consideration, the three curves remain in an upward state with the increasing value of the budget, and RBUR is very close to BRD-ENUR. When the budget is equal to 100, we can find that RUCR and BRD-ENUR only achieve 45% and 39% task completion rates in Figure 5a, meanwhile 30% and 26% task coverage rates in Figure 5b. With the increasing budget, the performances have a rapid growth when budget = 200. The reason is that the limited budget will reduce user enthusiasm, and the task rewards may not cover the user costs. When the total budget expands, more users will be more willing to take part in the tasks, so the platform will have a greater possibility to select the suitable user for each task. Compared with the BRD-ENUR, under the same condition, our RUCR effectively combines the capabilities of mobile users and edge nodes so that this collaborative sensing method can provide strong guarantees for task completion and task coverage. Therefore, our RUCR is more excellent than the BRD-ENUR approach, no matter the value of the budget.

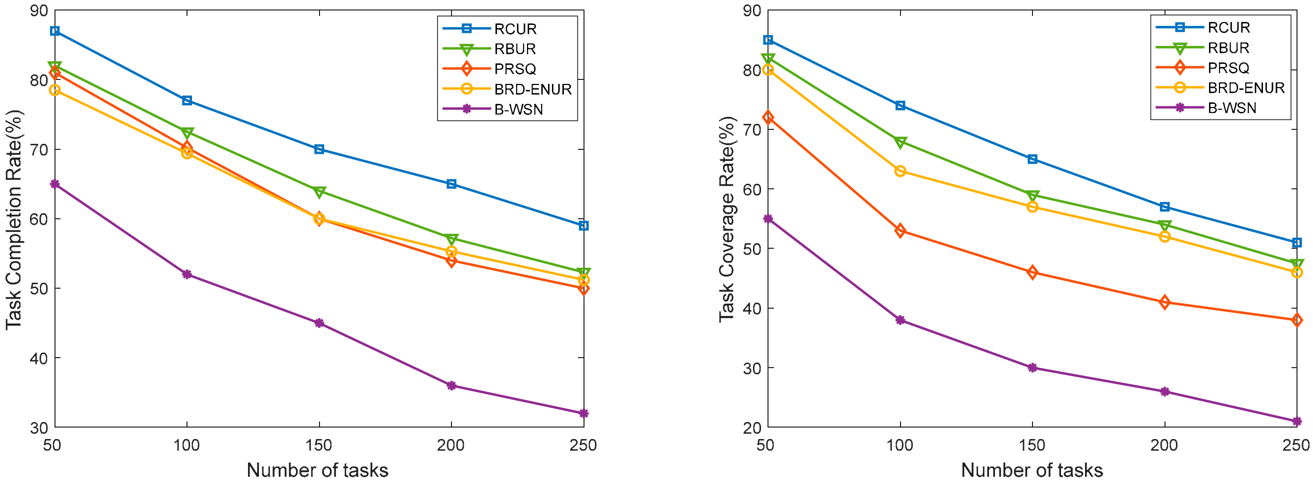


(**a**) (**b**)

Figure 5. Performances under various values of budget. (**a**) Task Completion Rate; (**b**) Task Coverage Rate.

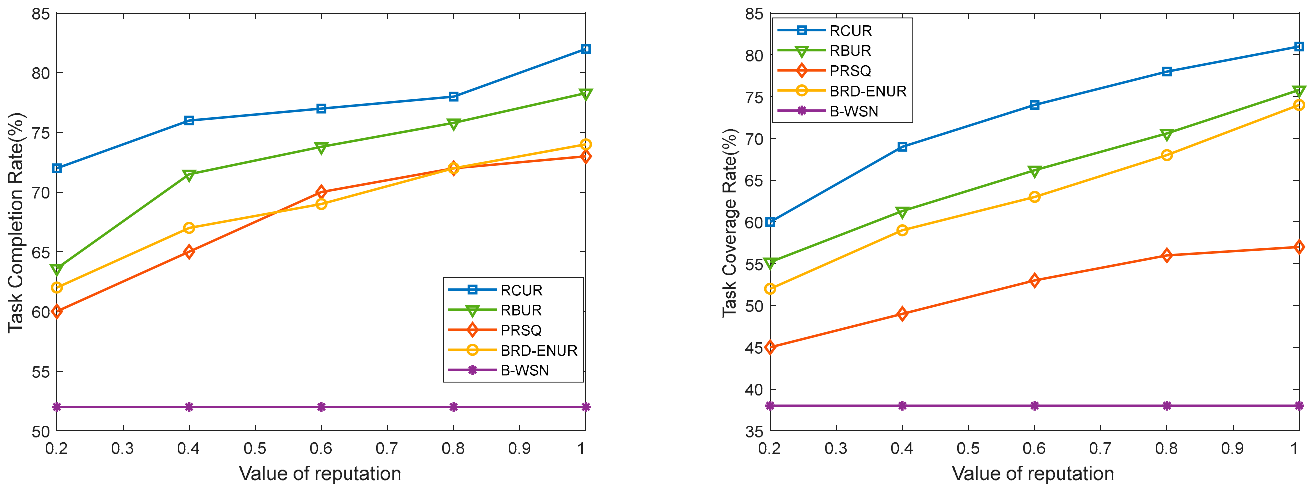
In Figure 6, we verify the relationship between the performance of the task completion rate and task coverage rate with the varying number of tasks. Through the comparison with the other three algorithms, our RCUR displays a better performance, and all of the algorithms show a downward trend with the number of tasks increasing. When the number of tasks is equal to 50, it is obvious that our RCUR significantly outperforms the other three benchmark algorithms. However, when the number of users and edge nodes remains unchanged, more tasks mean that there are not sufficient users or edge nodes to be candidates, which cannot guarantee the task completion rate. Furthermore, the platform may not find the most suitable participant to complete the task and ensure coverage. From Figure 6a, it can be seen that when the tasks increase to 250, our RCUR decreases from 87% to 59% in terms of the task completion rate but is still far superior to other algorithms. The results in Figure 6b are just as we expected that our RCUR takes a huge advantage in the performance of task coverage rate.

In Figure 7, we compare the task completion rate and task coverage rate of our RCUR approach with the other three algorithms under the different values of the initial user reputation. As we know, We except for the B-WSN algorithm, the other algorithms regard the user reputation as a fundamental consideration and recruit the participant based on the user reputation. However, our RBUR and RCUR not only evaluate the user's past reputation scores according to historical performance. Furthermore, it also takes account of the instantaneous reputation score and combines them to attain the user reputation. Figure 7a indicates that the performance of these algorithms has a slow growth with the initial reputation increasing. However, our RBUR and RCUR occupy the leading position in terms of task completion rate. When the value of a user's initial reputation increases, it means that there will be more users who can meet the requirement of the tasks and can be potential participants in finishing the task. In Figure 7b, we can observe that the increasing value of the initial reputation will have a more obvious influence in terms of the task coverage rate. This is because our RBUR and RCUR approach take the cover ability of the user as a vital indicator of the instantaneous reputation so that the platform will relatively select the user who may have the better cover ability to complete the task.



(a) (b)

Figure 6. Performances under various numbers of tasks. (**a**) Task Completion Rate; (**b**) Task Coverage Rate.



(a) (b)

Figure 7. Performances under various values of reputation. (**a**) Task Completion Rate; (**b**) Task Coverage Rate.

Figure 8 illustrates that the values of the user sensing radius have a significant impact on the task completion rate and task coverage rate. The user sensing radius represents the user's cover ability, which means that the user is able to provide a guarantee for task coverage. From Figure 8a, we can clearly see that when the value of the user sensing radius becomes bigger, mobile users have more powerful sensing and cover ability, so the four algorithms in MCS have an improvement in terms of the task completion rate. Furthermore, similar to Figure 7, the results have proven that user sensing radius is a contributing factor to the task coverage rate in Figure 8b. When the user sensing radius is equal to five, the limited cover ability cannot satisfy the task's constraints, so all algorithms attain a relatively low task coverage rate. With the user sensing radius increasing, in some cases, the tasks may be completely covered by the mobile users. Meanwhile, due to the collaborative sensing by the mobile users and edge nodes, our RCUR can achieve a better performance compared with other algorithms.

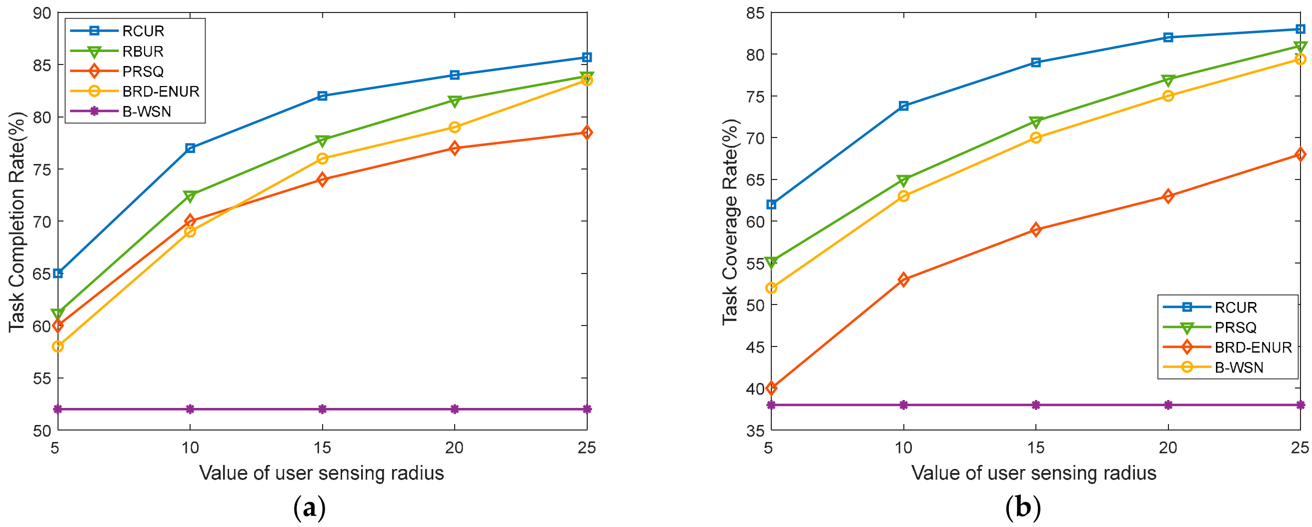


Figure 8. Performances under various values of user sensing radius. (**a**) Task Completion Rate; (**b**) Task Coverage Rate.

6. Conclusions and Future Work

The emergence of the Internet of Things and the widespread popularity of intelligent mobile devices enable Mobile CrowdSensing (MCS) to be a convenient platform for many applications. Meanwhile, the introduction of edge computing has also accelerated the development of MCS. However, how to make MCS more efficient and utilize edge computing technology has raised huge challenges. Hence, designing a reasonable user recruitment algorithm to find suitable users and take full advantage of edge nodes in order to improve the task completion rate and coverage rate has become increasingly important and urgent. In this study, we propose a reputation-based collaborative user recruitment algorithm (RCUR) under a certain budget in an edge-aided Mobile CrowdSensing system. We first introduce edge computing into MCS and build an edge-aided MCS system in urban scenarios. Moreover, we analyze the influence of user reputation on user recruitment. Then we establish a user reputation module to evaluate the user's past reputation score based on the previous performance, which indicates the users' reliability and credibility. Then we jointly consider the distance and coverage ability of the user to calculate the instantaneous reputation score. Thus, we deduce the user reputation equation by combining the user's past reputation score with the instantaneous reputation score. Finally, we utilize the sensing ability of edge nodes and design a collaborative sensing method. We use the greedy method to help choose the appropriate users for the tasks. Simulation results compared with the other three algorithms prove that our RCUR approach can significantly achieve better performance in task completion rate and task coverage rate.

A dynamic user recruitment algorithm is one of our future works. Generally speaking, user recruitment always recruits one participant to accomplish a certain task. However, when the tasks require more users to sense these data, how to design an optimal algorithm to select a set of users has been an important problem in MCS. Meanwhile, how to dynamically determine the number of required users for the tasks according to the user's characteristics, such as cover ability, reputation, and cost, is also a tough problem. In the future, we will concentrate on this dynamic user recruitment in MCS.

Author Contributions: Conceptualization, Y.L. (Yang Liu), Y.L. (Yong Li) and W.C.; methodology, Y.L. (Yang Liu) and W.C.; software, Y.L. (Yang Liu), W.W. and J.Y.; validation, Y.L. (Yang Liu) and W.W.; formal analysis, Y.L. (Yang Liu) and W.C.; investigation, Y.L. (Yang Liu), W.W. and J.Y.; resources, Y.L. (Yang Liu) and W.W.; data curation, Y.L. (Yang Liu) and J.Y.; writing—original draft preparation, Y.L. (Yang Liu); writing—review and editing, Y.L. (Yang Liu), Y.L. (Yong Li) and W.C.; supervision, Y.L. (Yang Liu), Y.L. (Yong Li) and W.C.; project administration, Y.L. (Yang Liu) and W.C.; funding acquisition, Y.L. (Yong Li) and W.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Fundamental Research Funds for the Central Universities (3102017zy026).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Jiang, W.; Chen, P.; Zhang, W.; Sun, Y.; Junpeng, C.; Wen, Q. User Recruitment Algorithm for Maximizing Quality under Limited Budget in Mobile Crowdsensing. *Discret. Dyn. Nat. Soc.* **2022**, *2022*, 4804231. [CrossRef]
2. Weerapanpisit, P.; Trilles, S.; Huerta, J.; Painho, M. A Decentralized Location-Based Reputation Management System in the IoT Using Blockchain. *IEEE Internet Things J.* **2022**, *9*, 15100–15115. [CrossRef]
3. Granell, C.; Kamilaris, A.; Kotsev, A.; Ostermann, F.O.; Trilles, S. Internet of things. In *Manual of Digital Earth*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 387–423.
4. Guo, B.; Liu, Y.; Liu, S.; Yu, Z.; Zhou, X. CrowdHMT: Crowd Intelligence with the Deep Fusion of Human, Machine, and IoT. *IEEE Internet Things J.* **2022**, *9*, 24822–24842. [CrossRef]
5. Alvear, O.; Calafate, C.T.; Cano, J.-C.; Manzoni, P. Crowdsensing in smart cities: Overview, platforms, and environment sensing issues. *Sensors* **2018**, *18*, 460. [CrossRef] [PubMed]
6. Jezdović, I.; Popović, S.; Radenković, M.; Labus, A.; Bogdanović, Z. A crowdsensing platform for real-time monitoring and analysis of noise pollution in smart cities. *Sustain. Comput. Inform. Syst.* **2021**, *31*, 100588. [CrossRef]
7. Trilles, S.; Calia, A.; Belmonte, Ó.; Torres-Sospedra, J.; Montoliu, R.; Huerta, J. Deployment of an open sensorized platform in a smart city context. *Future Gener. Comput. Syst.* **2017**, *76*, 221–233. [CrossRef]
8. Chen, J.; Yang, J. Maximizing coverage quality with budget constrained in mobile crowd-sensing network for environmental monitoring applications. *Sensors* **2019**, *19*, 2399. [CrossRef]
9. Yu, Z.; Ma, H.; Guo, B.; Yang, Z. Crowdsensing 2.0. *Commun. ACM* **2021**, *64*, 76–80. [CrossRef]
10. Tseng, L.; Yao, X.; Otoum, S.; Aloqaily, M.; Jararweh, Y. Blockchain-based database in an IoT environment: Challenges, opportunities, and analysis. *Clust. Comput.* **2020**, *23*, 2151–2165. [CrossRef]
11. Castaño, F.; Haber, R.E.; Mohammed, W.M.; Nejman, M.; Villalonga, A.; Lastra, J.L. Quality Monitoring of Complex Manufacturing Systems on the Basis of Model Driven Approach. 2020. Available online: https://trepo.tuni.fi/handle/10024/127663 (accessed on 25 October 2020).
12. Forestiero, A.; Papuzzo, G. Recommendation platform in Internet of Things leveraging on a self-organizing multiagent approach. *Neural Comput. Appl.* **2022**, *34*, 16049–16060. [CrossRef]
13. Zhao, P.; Li, C.; Fu, Y.; Hui, Y.; Zhang, Y.; Cheng, N. Blockchain-enabled conditional decentralized vehicular crowdsensing system. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 18937–18950. [CrossRef]
14. Theodoridis, E.; Mylonas, G.; Chatzigiannakis, I. Developing an iot smart city framework. In Proceedings of the IISA 2013, Piraeus, Greece, 10–12 July 2013; pp. 1–6.
15. Mankodiya, H.; Palkhiwala, P.; Gupta, R.; Jadav, N.K.; Tanwar, S.; Neagu, B.-C.; Grigoras, G.; Alqahtani, F.; Shehata, A.M. A Real-Time Crowdsensing Framework for Potential COVID-19 Carrier Detection Using Wearable Sensors. *Mathematics* **2022**, *10*, 2927. [CrossRef]
16. Trilles, S.; Torres-Sospedra, J.; Belmonte, Ó.; Zarazaga-Soria, F.J.; González-Pérez, A.; Huerta, J. Development of an open sensorized platform in a smart agriculture context: A vineyard support system for monitoring mildew disease. *Sustain. Comput. Inform. Syst.* **2020**, *28*, 100309. [CrossRef]
17. Capponi, A.; Fiandrino, C.; Kantarci, B.; Foschini, L.; Kliazovich, D.; Bouvry, P. A survey on mobile crowdsensing systems: Challenges, solutions, and opportunities. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 2419–2465. [CrossRef]
18. Chen, L.; He, X.; Zhao, X.; Li, H.; Huang, Y.; Zhou, B.; Chen, W.; Li, Y.; Wen, C.; Wang, C. GoComfort: Comfortable Navigation for Autonomous Vehicles Leveraging High-Precision Road Damage Crowdsensing. *IEEE Trans. Mob. Comput.* **2022**, 1–18. [CrossRef]
19. Beruvides, G.; Juanes, C.; Castaño, F.; Haber, R.E. A self-learning strategy for artificial cognitive control systems. In Proceedings of the 2015 IEEE 13th International Conference on Industrial Informatics (INDIN), Cambridge, UK, 22–24 July 2015; pp. 1180–1185.
20. He, Y.; Wang, D.; Huang, F.; Zhang, R.; Gu, X.; Pan, J. A V2I and V2V Collaboration Framework to Support Emergency Communications in ABS-Aided Internet of Vehicles. *IEEE Trans. Green Commun. Netw.* **2023**, 1. [CrossRef]

21. Bellavista, P.; Belli, D.; Chessa, S.; Foschini, L. A social-driven edge computing architecture for mobile crowd sensing management. *IEEE Commun. Mag.* **2019**, *57*, 68–73. [CrossRef]
22. Gehlot, A.; Alshamrani, S.S.; Singh, R.; Rashid, M.; Akram, S.V.; AlGhamdi, A.S.; Albogamy, F.R. Internet of things and long-range-based smart lampposts for illuminating smart cities. *Sustainability* **2021**, *13*, 6398. [CrossRef]
23. Shi, S.; Cui, J.; Jiang, Z.; Yan, Z.; Xing, G.; Niu, J.; Ouyang, Z. VIPS: Real-time perception fusion for infrastructure-assisted autonomous driving. In Proceedings of the 28th Annual International Conference on Mobile Computing And Networking, Sydney, NSW, Australia, 17–21 October 2022; pp. 133–146.
24. Park, G.S.; Kim, W.; Jeong, S.H.; Song, H. Smart base station-assisted partial-flow device-to-device offloading system for video streaming services. *IEEE Trans. Mob. Comput.* **2016**, *16*, 2639–2655. [CrossRef]
25. Zhou, P.; Chen, W.; Ji, S.; Jiang, H.; Yu, L.; Wu, D. Privacy-preserving online task allocation in edge-computing-enabled massive crowdsensing. *IEEE Internet Things J.* **2019**, *6*, 7773–7787. [CrossRef]
26. Xiong, J.; Zhao, M.; Alam Bhuiyan, Z.; Chen, L.; Tian, Y. An AI-enabled three-party game framework for guaranteed data privacy in mobile edge crowdsensing of IoT. *IEEE Trans. Ind. Inform.* **2019**, *17*, 922–933. [CrossRef]
27. Xiao, M.; Gao, G.; Wu, J.; Zhang, S.; Huang, L. Privacy-preserving user recruitment protocol for mobile crowdsensing. *IEEE/ACM Trans. Netw.* **2020**, *28*, 519–532. [CrossRef]
28. Wang, E.; Yang, Y.; Wu, J.; Liu, W.; Wang, X. An efficient prediction-based user recruitment for mobile crowdsensing. *IEEE Trans. Mob. Comput.* **2017**, *17*, 16–28. [CrossRef]
29. Zhang, Y.; Zhang, X. Price learning-based incentive mechanism for mobile crowd sensing. *ACM Trans. Sens. Netw. (TOSN)* **2021**, *17*, 1–24. [CrossRef]
30. Yin, J.; Wei, L.; Sun, H.; Lin, Y.; Zhao, X. An Incentive Mechanism for Mobile Crowd Sensing in Vehicular Ad Hoc Networks. *J. Transp. Technol.* **2021**, *12*, 96–110. [CrossRef]
31. Zhang, J.; Zhang, X. Multi-task allocation in mobile crowd sensing with mobility prediction. *IEEE Trans. Mob. Comput.* **2021**, *22*, 1081–1094. [CrossRef]
32. Xie, Z.; Hu, L.; Huang, Y.; Pang, J. A semiopportunistic task allocation framework for mobile crowdsensing with deep learning. *Wirel. Commun. Mob. Comput.* **2021**, *2021*, 6643229. [CrossRef]
33. Abualigah, L.; Elaziz, M.A.; Khodadadi, N.; Forestiero, A.; Jia, H.; Gandomi, A.H. Aquila optimizer based PSO swarm intelligence for IoT task scheduling application in cloud computing. In *Integrating Meta-Heuristics and Machine Learning for Real-World Optimization Problems*; Springer International Publishing: Cham, Switzerland, 2022; pp. 481–497.
34. Rui, L.; Zhang, Y.; Zhang, P.; Qiu, X. Location-dependent sensing data collection and processing mechanism in vehicular network. *Trans. Emerg. Telecommun. Technol.* **2019**, *30*, e3283. [CrossRef]
35. Liu, Y.; Yu, Z.; Wang, J.; Guo, B.; Su, J.; Liao, J. CrowdManager: An Ontology-Based Interaction and Management Middleware for Heterogeneous Mobile Crowd Sensing. *IEEE Trans. Mob. Comput.* **2022**, 1–18. [CrossRef]
36. Truong, N.B.; Lee, G.M.; Um, T.W.; Mackay, M. Trust evaluation mechanism for user recruitment in mobile crowd-sensing in the Internet of Things. *IEEE Trans. Inf. Forensics Secur.* **2019**, *14*, 2705–2719. [CrossRef]
37. Wu, D.; Li, H.; Wang, R. User characteristic aware participant selection for mobile crowdsensing. *Sensors* **2018**, *18*, 3959. [CrossRef] [PubMed]
38. Li, Q.; Cao, H.; Wang, S.; Zhao, X. A reputation-based multi-user task selection incentive mechanism for crowdsensing. *IEEE Access* **2020**, *8*, 74887–74900. [CrossRef]
39. Liu, Y.; Li, Y.; Cheng, W.; Wang, W.; Yang, J. A willingness-aware user recruitment strategy based on the task attributes in mobile crowdsensing. *Int. J. Distrib. Sens. Netw.* **2022**, *18*, 15501329221123531. [CrossRef]
40. Guo, B.; Chen, H.; Han, Q.; Yu, Z.; Zhang, D.; Wang, Y. Worker-contributed data utility measurement for visual crowdsensing systems. *IEEE Trans. Mob. Comput.* **2016**, *16*, 2379–2391. [CrossRef]
41. Zhou, J.; Yu, Z.Y.; Guo, W.Z.; Guo, L.K.; Zhu, W.P. Participant selection algorithm for t-sweep k-coverage crowd sensing tasks. *Comput. Sci.* **2018**, *45*, 157–164.
42. Wang, Z.; Zhao, J.; Hu, J.; Zhu, T.; Wang, Q.; Ren, J.; Li, C. Towards personalized task-oriented worker recruitment in mobile crowdsensing. *IEEE Trans. Mob. Comput.* **2020**, *20*, 2080–2093. [CrossRef]
43. Zhang, M.; Yang, P.; Tian, C.; Tang, S.; Gao, X.; Wang, B.; Xiao, F. Quality-aware sensing coverage in budget-constrained mobile crowdsensing networks. *IEEE Trans. Veh. Technol.* **2015**, *65*, 7698–7707. [CrossRef]
44. Alagha, A.; Mizouni, R.; Singh, S.; Otrok, H.; Ouali, A. SDRS: A stable data-based recruitment system in IoT crowdsensing for localization tasks. *J. Netw. Comput. Appl.* **2021**, *177*, 102968. [CrossRef]
45. Yucel, F.; Yuksel, M.; Bulut, E. Coverage-aware stable task assignment in opportunistic mobile crowdsensing. *IEEE Trans. Veh. Technol.* **2021**, *70*, 3831–3845. [CrossRef]
46. Ma, L.; Liu, X.; Pei, Q.; Xiang, Y. Privacy-preserving reputation management for edge computing enhanced mobile crowdsensing. *IEEE Trans. Serv. Comput.* **2018**, *12*, 786–799. [CrossRef]
47. Zhang, Y.; Li, P.; Zhang, T.; Liu, J.; Huang, W.; Nie, L. Dynamic User Recruitment in Edge-aided Mobile Crowdsensing. *IEEE Trans. Veh. Technol.* **2023**. [CrossRef]
48. Xiong, J.; Chen, X.; Yang, Q.; Chen., L.; Yao, Z. A task-oriented user selection incentive mechanism in edge-aided mobile crowdsensing. *IEEE Trans. Netw. Sci. Eng.* **2019**, *7*, 2347–2360. [CrossRef]

49. Sun, J.; Pei, Y.; Hou, F.; Ma, S. Reputation-aware incentive mechanism for participatory sensing. *IET Commun.* **2017**, *11*, 1985–1991. [CrossRef]
50. Fiandrino, C.; Capponi, A.; Cacciatore, G.; Kliazovich, D.; Sorger, U.; Bouvry, P.; Kantarci, B.; Granelli, F.; Giordano, S. Crowdsensim: A simulation platform for mobile crowdsensing in realistic urban environments. *IEEE Access* **2017**, *5*, 3490–3503. [CrossRef]
51. Jiang, W.; Chen, J.; Liu, X.; Liu, Y.; Lv, S. Participant recruitment method aiming at service quality in mobile crowd sensing. *Wirel. Commun. Mob. Comput.* **2021**, *2021*, 6621659. [CrossRef]