Article

Communication Method Using Cellular and D2D Communication for Reverse Auction-Based Mobile Crowdsensing †

Tetsushi Matsuda 1, *, Toru Inada 1 and Susumu Ishihara 2,3, *

1 Information Technology R&D Center, Mitsubishi Electric Corporation, Kamakura 247-8501, Japan
2 College of Engineering, Academic Institute, Shizuoka University, Hamamatsu 432-8011, Japan
3 Graduate School of Science and Technology, Shizuoka University, Hamamatsu 432-8011, Japan
* Correspondence: matsuda.tetsushi@dh.mitsubishielectric.co.jp (T.M.); ishihara.susumu@shizuoka.ac.jp (S.I.)
† This paper is an extended version of our paper published in the 2020 IEEE International Conference on Pervasive Computing and Communications Workshops, Austin, TX, USA, 23–27 March 2020.

Abstract: The demand for services that provide current sensing data (data) obtained at a specified location is expected to grow in the future. To realize such services with lower sensor installation and operation costs, we consider the use of a pull-type mobile crowdsensing with reverse auction (pull-type RevAuc) that determines the size of the incentive paid to mobile hosts that provide data. In a pull-type RevAuc, more cellular traffic between a server and mobile hosts can increase the service operation cost. In this paper, we propose a new communication method for the pull-type RevAuc that reduces cellular traffic between a server and mobile hosts by using device-to-device communication. Through simulations, we confirmed that the proposed method can reduce cellular traffic compared to the existing method for the pull-type RevAuc while selecting a successful bidder that can provide data at almost the same location for almost the same reward within almost the same amount of time as in the existing method.

Keywords: D2D communication; mobile crowdsensing; MCS; reverse auction; incentive

1. Introduction

The demand for services that provide current sensing data (data) at a specified location is expected to grow in the future [1]. Examples of such services are (1) a driver obtains information on the traffic ahead of him or her during a traffic jam or (2) a sightseer asks a data service about the crowding at his or her destination. The cost of installing, operating, and maintaining sensors to obtain such sensing data is a problem when providing such services. Mobile crowdsensing (MCS) [2,3] is a method that can collect sensing data without the pre-installation of sensors in the field. An MCS system consists of mobile hosts with sensors, such as smartphones and cars, that are willing and able to help a service obtain data and a server operated by a service provider, which collects data with the help of the mobile hosts.

The cost of cellular traffic can be a problem in MCS. Let us assume an MCS system that provides a user with current data about the point of interest (POI) specified by the user. To make the MCS system applicable to various services providing current sensing data, it is desirable to impose no restriction on the location of a POI. Therefore, we assume that a server communicates with mobile hosts through a cellular network because a cellular network allows mobile hosts to communicate with the server almost everywhere. It is also desirable to have many mobile hosts participate in MCS so that the system can provide sensing data for many locations. However, the cellular traffic between the server and mobile hosts increases as the number of mobile hosts increases. Thus, an increase in cellular traffic can increase the operation cost of the MCS system. In this paper, we propose a new communication method for MCS that reduces cellular traffic between the server and the mobile hosts by exploiting device-to-device (D2D) [4] communication, which enables...
neighboring mobile hosts to communicate with each other through a free wireless link, such as an IEEE 802.11p wireless network. In the proposed method, only a limited number of mobile hosts directly communicate and exchange messages with the server via a cellular network and forward the messages to other mobile hosts through D2D communication. Thus, cellular traffic can be reduced. We identify design challenges arising from the use of D2D communication and propose solutions for them.

To successfully obtain data with MCS, it is necessary to motivate mobile hosts to help. Incentive-based MCS [5,6] is a method that increases data availability by motivating mobile MCS hosts to obtain the data. The design goal of the incentive-based MCS is that the server can obtain higher-quality data by offering fewer incentives. The reverse auction (RevAuc) [7], in which the bidder requesting the smallest reward wins the bid, is a solution to achieve the goal (Figure 1). An example of a reward is money paid to a successful bidder. In MCS using RevAuc, the server acts as an auctioneer that sends bid invitations and determines a successful bidder. Each mobile host sends a bid to the server. The bid includes information on the quality of data that the mobile host plans to obtain and transmit and the reward that it requests. In this paper, we assume that the quality of data depends on factors such as the distance between the POI and the place where the data is obtained, the time when the data is obtained, and the resolution of the data. The evaluation score of each bid is calculated based on the reward requested by the bidder and the quality of data proposed by the bidding mobile host. The server asks the mobile host that has tendered the bid with the highest evaluation score among bids sent by mobile hosts to obtain and transmit the data.

There are two methods for a server to send a bid invitation in MCS using RevAuc. One is pull-type, in which the server sends a bid invitation to a mobile host when it receives a query for bid invitations from the mobile host. Another is push-type, in which the server sends a bid invitation spontaneously (Figure 2) [2]. In this paper, we use the pull-type RevAuc. In a RevAuc, the server needs to know the participating mobile hosts and their locations by some method. It is easier to build a system with a pull-type RevAuc than with a push-type RevAuc because actions for the RevAuc are started by mobile hosts, and the server can acquire information on the participating mobile hosts and their locations from mobile hosts in a pull-type RevAuc procedure while the server needs to track the position of each participating mobile host with some method such as periodic reports of location by mobile hosts in a push-type RevAuc.
The volume of communication traffic between a server and mobile hosts is a problem when realizing a service that collects current sensing data at a specified location by a pull-type RevAuc. The reasons are related to the three characteristics required during data acquisition by a pull-type RevAuc:

1. **Timeliness:** Obtaining data within a specified timeframe.
2. **Quality:** Obtaining higher-quality data.
3. **Cost:** Obtaining lower-cost data.

To satisfy the above characteristics, the server needs to receive bids from all mobile hosts that can obtain data within the pre-determined amount of time $T_g$ and select a bid with the highest evaluation score. To participate in a RevAuc, a mobile host needs to query the server for a bid invitation within a period shorter than $T_g$ and send a bid if it is willing to provide data. Even though the amount of data in each message for a query for a bid invitation, the actual bid invitation, and the bid is small, the traffic related to a RevAuc can significantly increase the cost of cellular communication in the system because all mobile hosts transmit all these messages periodically. Thus, a reduction in traffic for bid-related messages to lower the cost of cellular communication in system operation is a need for a pull-type RevAuc.

In this paper, we propose a new communication method for a pull-type RevAuc that reduces the cellular traffic between a server and mobile hosts compared to the existing method. The proposed method has the following three features.

1. **Reduced cellular traffic by using D2D communication.**

   The proposed method reduces cellular traffic compared to the existing method by sending a bid invitation and collecting bids through D2D communication with the help of so-called representative mobile hosts (REPs). Among mobile hosts that can communicate with a mobile host (referred to as MH-A) by one-hop D2D communication (one-hop neighbors of MH-A), one mobile host volunteers to serve as a REP autonomously.

2. **Selection of a successful bidder that can provide data at almost the same location for almost the same size of reward within the same amount of time as in the existing method.**

   In the proposed method, bid invitations are forwarded by REPs. If a REP for a mobile host does not receive a bid invitation from a server, the mobile host cannot receive the bid invitation. However, in the existing method without D2D communication, the mobile host may receive the bid invitation from the server and may provide the requested data for a POI. In this case, the RevAuc result can be different between the proposed method and the existing method without D2D communication. If a server selects suitable REPs to which it sends a bid invitation, a mobile host that can provide the requested data in the existing method without D2D communication will be able to receive the bid invitation and join the RevAuc. In the proposed method using D2D communication, a server should select

---

**Figure 2.** Sequence of pull-type and push-type RevAuc.
REPs to which it will send a bid invitation for a data request so that a set of mobile hosts that receive the bid invitation via REPs includes the set of mobile hosts that receive bid invitations in the existing method to obtain almost the same result as in the existing method.

(3) Mitigation of the effects of misbehaving REPs that fail to fulfill the role of REP.

In MCS, a mobile host may fail to fulfill the role of REP after volunteering to serve as a REP due to some reasons, such as temporary disconnection from the cellular network, failure of the mobile host, shortage of power supply, and accident. If a misbehaving REP does not forward a bid invitation to its one-hop neighbors or the bids of its one-hop neighbors to a server (i.e., causing packet loss), a server may not receive a bid that achieves a high evaluation score, and the RevAuc can fail. In the proposed method, multiple REPs volunteer among the one-hop neighbors of a mobile host, and the server sends the same bid invitation to those REPs. Thus, the mobile host can tender a bid if there is at least one non-misbehaving REP among the available REPs, which decreases the probability of failure of the RevAuc due to misbehaving REPs.

The main contributions of this paper are the proposal of the new communication method for the pull-type RevAuc and the confirmation of the effectiveness of the proposed method through simulation. The remainder of this paper is organized as follows. Section 2 describes related studies. Section 3 explains the existing method on which the proposed method is based. Section 4 explains the proposed method. Section 5 explains evaluation methods and results and discusses the results, and Section 6 provides conclusions.

2. Related Work

We review related work on MCS that uses RevAuc, the target of this research, and related work on methods for reducing cellular traffic between a server and mobile hosts in MCS. First, we review related work on MCS that uses RevAuc.

Lee et al. [8] proposed a method for a server to determine the amount of incentive given to a successful bidder, which they named reverse auction-based dynamic pricing virtual participatory credit. The method provides relatively more incentive to mobile hosts that continue to participate in a RevAuc even if they are not selected as successful bidders so that the number of participating hosts increases beyond a certain threshold value. They did not mention how the server selects mobile hosts that receive a bid invitation for a data request nor how the server communicates with the mobile hosts.

Zhang et al. [9] proposed an algorithm for a server to determine whether a mobile host is a successful bidder or not on receiving a bid from the mobile host after receiving bids from a certain number of mobile hosts to solve a problem; in other words, it takes a long time to select successful bidders if a server determines the successful bidders after receiving bids from all mobile hosts. They did not mention how the server selects which mobile hosts to send a bid invitation for a data request nor how the server communicates with the mobile hosts.

Liu et al. [10] proposed an incentive mechanism based on reverse auction for location-aware sensing, which determines successful bidders so as to increase the number of participating mobile hosts and decrease the ratio of mobile hosts that do not provide data to the server after being selected as successful bidders. They assume that a server sends bid invitations for all data requests to all mobile hosts, and all mobile hosts tender bids for some bid invitations. They also assume that the server communicates with mobile hosts directly through some network, such as a cellular network.

Yang et al. [11] proposed an algorithm for a server to select successful bidders in a RevAuc-based MCS. They assume that a server sends bid invitations for all data requests to all mobile hosts, and all mobile hosts tender bids for some bid invitations. They also assume that the server communicates with mobile hosts directly through some network, such as a cellular network.

The above studies related to MCS that uses a RevAuc assume that a server sends bid invitations to mobile hosts and receives bids from mobile hosts through some network,
such as a cellular network. They do not mention how to reduce traffic between the server and mobile hosts.

Next, we review related work on methods for reducing cellular traffic between a server and mobile hosts in MCS, although those related studies do not focus on MCS using RevAuc.

Mota et al. [12] proposed D2D extended sensing (D2D-ES) and D2D content dissemination (D2D-CD), which reduce cellular traffic between a server and mobile hosts using D2D communication in MCS. In D2D-ES, REPs receive sensing data from other mobile hosts, and they send sensing data to the server. D2D-ES assumes that REPs that send sensing data to the server are determined in advance. Therefore, cellular traffic used for the mobile hosts to send sensing data to the server can be reduced. In D2D-CD, REPs receive some information from the server, and they send the information to other mobile hosts through D2D communication. D2D-CD assumes that REPs that receive information from the server are determined in advance. Therefore, cellular traffic used for the server to send information to the mobile hosts can be reduced. Even though they do not mention RevAuc, cellular traffic in a RevAuc can be reduced by applying D2D-CD to transmissions of bid invitations from the server to mobile hosts and by applying D2D-ES to transmissions of bids from mobile hosts to the server. However, they do not mention how to apply their method to pull-type RevAuc.

Matsumoto et al. [13] proposed a method for reducing cellular traffic by uploading only image data that are requested by drivers to a data server. In their target system, cars upload image data that they take with onboard cameras to the server, and the server provides the image data of a location requested by a driver. They propose a method in which the server can find cars that hold necessary image data efficiently and ask one of the cars to upload the necessary image data by having cars exchange metadata about an image (e.g., the location where and time when the image was acquired) through D2D communication and by having the cars report to the server the metadata about themselves and the cars next to them. Their method handles image data taken and retained by cars spontaneously, and cars send metadata about their image data to the server with the help of D2D communication. They do not mention how to apply their approach to pull-type RevAuc.

Pu et al. [14] proposed an MCS method in which a mobile host requesting data sends information on a data request and payable reward to a mobile host (MH-A) that it communicates with via D2D communication while moving around. The requester receives information on the quality of data that MH-A can provide and determines whether to ask MH-A to send data. They propose an algorithm for a requester to determine which mobile hosts it will ask to obtain and transmit data. The method proposed in [14] has two constraints. One is that it can only deal with data requests by mobile hosts. Another is that current data at a location far away from the current location of the data requester cannot be obtained because the data requester tries to find mobile hosts only via D2D communication. These constraints impose a restriction on the location of a POI. Thus, their method is applicable only to some of the services providing current sensing data.

Wang et al. [15] proposed a crowdsensing algorithm in which mobile hosts obtain data at POIs and aggregate data received from other mobile hosts through D2D communication while relaying data to an access point, which acts as a server collecting data. The incentive given to a mobile host is determined by the value of the contribution made by the mobile host for obtaining data. Their algorithm determines which mobile hosts should obtain data and how obtained data should be aggregated and relayed to the access point through D2D communication under the condition that the movement of mobile hosts can be predicted. They did not mention how a server can send information to mobile hosts with less cellular traffic nor how to apply their method to pull-type RevAuc.

Yang et al. [16] proposed a crowdsensing algorithm that collects data obtained by mobile hosts spontaneously (i.e., without prior knowledge of a POI) to a server through D2D communication with incentives. A mobile host can relay data received from another mobile host via D2D communication. They propose a lightweight joint sensing rate control
and dynamic routing algorithm in a D2D network, OptMPSS. They also proposed an algorithm for determining incentives for mobile hosts, Backpressure Meets Taxes. They did not mention how to carry out pull-type RevAuc with their method.

Wang et al. [17] proposed an algorithm to determine which mobile hosts are to participate in MCS to minimize the cellular communication cost under the condition that some mobile hosts pay based on the amount of cellular traffic that they use, that is, pay as you go (PAYG), and other mobile hosts pay a constant fee regardless of the amount of cellular traffic, that is, pay monthly (PAYM). They try to minimize the cellular communication cost by uploading data to a server via PAYM hosts as much as possible, while PAYG hosts relay data to PAYM hosts through D2D communication. They did not mention how to carry out pull-type RevAuc with their method. Additionally, their method can be used only when PAYM hosts exist in a system.

Hui et al. [18] proposed a utility-based data computing scheme to provide sensing services in Internet of Things. In their scheme, a roadside unit (RSU) collects sensing data from vehicles. An RSU can collect sensing data from vehicles directly or request a roadside buffer (RSB) to collect sensing data from vehicles. Utilities for an RSU and RSBs are calculated based on the power and time cost needed for collecting sensing data. The interactions between the RSU and RSBs are modeled by a bargaining game, and both the RSU and RSBs try to maximize their utilities. In the bargaining game, RSBs tender proposals to the RSU. Then the RSU decides which RSBs or vehicles to obtain sensing data from. Vehicles do not participate in the bargaining game even though the bargaining game is similar to RevAuc. Even though their scheme can contribute to reducing cellular traffic between the RSU and vehicles through the maximization of utilities, the authors of [18] did not mention how to apply their scheme to our scenario, where vehicles participate in RevAuc.

Hui et al. [19] proposed a block-chain-based collaborative crowdsensing (BCC) scheme in an autonomous vehicular network. In their scheme, an edge computing device (ECD) on roadside collects sensing data from vehicles. Their scheme comprises (1) transaction architecture to protect the privacy of vehicles and ensure the security and nonrepudiation of a transaction, (2) an incentive mechanism to encourage the participation of vehicles, and (3) an algorithm to optimally select vehicles that execute tasks collaboratively with the target of minimizing a task execution cost (TED). Their scheme does not make any assumptions about the communication network between ECDs and vehicles. Even though their scheme may be applicable to reducing the cellular traffic between ECDs and vehicles with an appropriate design of the communication network and TED [19], it does not mention how to apply their scheme to our scenario, where vehicles participate in RevAuc.

Felici-Castell et al. [20] proposed a smart live video adaptive streaming technique over a multi-hop ad hoc network for road traffic monitoring. In [20], a camera node transmits video data to a traffic monitoring server via a multi-hop D2D communication network. Using the multi-hop D2D network, their method can eliminate cellular traffic between the camera node and the traffic monitoring server while sharing video data between the camera node and the traffic monitoring server. The authors of [20] did not mention how to apply their method to pull-type RevAuc.

Lopez-Ballester et al. [21] proposed a method enabling a sensor node to compute psycho-acoustic parameters in real time using convolutional neural networks. Their method allows wireless acoustic sensor nodes to compute psycho-acoustic parameters without sending raw audio streams over a wireless acoustic sensor network. Their method can be used to reduce traffic in the wireless acoustic sensor network. The authors of [21] did not mention how to apply their method to a pull-type RevAuc.

In summary, existing studies on MCS have described how to reduce cellular traffic between a server and mobile hosts using D2D communication. However, as far as we know, no existing study can be applied to reducing the cellular traffic of pull-type RevAuc as it is currently employed.
3. Existing Method

In this section, we first explain the procedure of the pull-type RevAuc that uses only cellular communication (Figure 3), which we call the existing method in this paper. This method is based on the method of MCS using RevAuc described in previous studies [8,9]. We assume that sensing data service users can send a data request for any location (POI) to a server at any time. We make two assumptions to reduce cellular traffic in RevAuc for the existing method. We assume that the server sends a bid invitation for a currently effective data request to a mobile host if the server determines that the mobile host is highly likely to provide the requested data in pull-type RevAuc. This assumption is for reducing cellular traffic for useless Bid Invitation messages. We assume that a common bid evaluation function is defined so that the server and mobile hosts calculate the evaluation score of a bid based on the location of a mobile host, the time when it obtains data, and the reward offered in the bid. Each mobile host calculates the evaluation score of the bid that it can tender. If the evaluation score of a bid calculated by a mobile host is more than zero, the mobile host tenders the bid so that cellular traffic for useless Bids messages can be reduced.

(1) Each mobile host sends a Bid Invitations Query message, which includes the location of the mobile host, to a server periodically with a cycle \( T_q \). This message serves to trigger a bid invitation from the server.

(2) The server selects data requests that it has received recently and matches the Bid Invitations Query message based on the location of the mobile host included in the query message. The server sends multiple bid invitations in a single Bid Invitations message to the mobile host. Each bid invitation in the message includes the data request (data request identifier (ID), POI, and specification of data to be obtained (e.g., the type of sensor to be used)), the maximum reward to be paid, and the time when the successful bidder will be determined (auction end time). In this study, we assume that the server selects a successful bidder for a data request when time \( T_L \) has passed after the data request is received by the server. \( T_L \) is longer than the time \( T_q \), which is the duration in which the request is effective.

(3) The mobile host calculates the reward it will offer for the bid invitations in the Bid Invitations message. It sends bids to the server by a Bids message. Each bid in the message includes the data request ID, bidding mobile host’s ID, expected location and time for data acquisition by the mobile host, specification of the data to be obtained, and the reward (i.e., bid price) it offers to the server for the data. We assume that the mobile hosts and the server use the same bid evaluation function.

(4) For each data request, the server selects a successful bidder mobile host that sent the bid with the highest evaluation score among mobile hosts that have sent a bid within time \( T_q \) after the data request was received. Then, the server sends a Bid Result message, which asks the mobile host to obtain and transmit the data, to the successful bidder mobile host. The Bid Result message includes the content of the bid, as described in step (3), that was sent with the Bids message from the successful bidder.

(5) If a mobile host that has sent a bid to the server has not received a Bid Result message from the server by the auction end time included in the Bid Invitation, it knows that it was not selected as the successful bidder. If a mobile host receives a Bid Result message, it sends a Data message to the server after it obtains the requested data. The Data message includes the data request ID and requested data. After receiving a Data message, the server processes the received data to fulfill the original data request.

When receiving a Bid Invitations Query message from a mobile host, the server selects data requests that will be included in bid invitations sent with a Bid Invitations message as follows. The server selects currently effective requests for data that are likely to be obtained by the mobile host based on the location of the mobile host. One possible way of selecting such requests is by comparing the distance between the POI and the current location of the mobile host with a threshold distance.
In Section 5, we compare the existing method with the proposed method by simulation. In the simulation model, we assume that a mobile host obtains requested data on its planned trajectory, and a server predicts whether a mobile host obtains the requested data as follows. The server uses the location of a mobile host included in a Bid Invitations Query message to update a database of the latest locations of mobile hosts. Then, for each data request, the server calculates the minimum value of \( R \), the radius of circle \( C \) with the POI of the data request as the center, so that the expected number of hosts in circle \( C \) can be greater than or equal to a configuration parameter \( \theta \) (Figure 4). If the calculated value of \( R \) is larger than the configuration parameter \( R_{\text{max}} \), then \( R \) is set to \( R_{\text{max}} \). The server determines that a mobile host is likely to provide data for the data request if the location of the mobile host recorded in the host location database is within circle \( C \). The current model does not take the planned trajectory of mobile hosts into consideration. The extension of the model to consider the planned trajectory of mobile hosts is for further study.

Note that a bid invitation and a bid are protected by a message authentication method, such as an electronic signature, to assure sender authentication and prevent message modification. We assume that some identity information of a mobile host is used by the server for message authentication. This can lead to privacy issues. Privacy issues are for further study.
4. Proposed Method

In this section, we propose a pull-type RevAuc method that uses D2D communication to reduce cellular traffic between a server and mobile hosts compared with the existing method.

4.1. RevAuc Procedure Using D2D Communication

Compared to the existing method, the proposed method reduces cellular traffic between the server and mobile hosts by utilizing D2D communication. Among mobile hosts that can communicate with a mobile host MH-A by one-hop D2D communication (one-hop neighbors of MH-A), a mobile host volunteers to serve as a REP autonomously if no mobile host has volunteered to serve as a REP among one-hop neighbors of MH-A for the configuration parameter time \( T_q \). A REP sends a Bid Invitations Query message to the server when it volunteers to serve as REP. When it receives bid invitations from the server, it forwards the bid invitations to its one-hop neighbors. Then, it receives bids from its one-hop neighbors. Finally, it evaluates the bids and forwards the bid with the highest evaluation score for each bid invitation to the server (Figure 5a). The REP finishes its role as REP after forwarding the bid to the server. After that, another mobile host will volunteer to serve as a new REP to accommodate the movement of mobile hosts.

![Figure 5. RevAuc procedure using D2D communication.](image)

4.2. Design Challenges in Obtaining Data at Almost the Same Location for Almost the Same Reward as in the Existing Method

Some design challenges arise when REPs are used to execute a RevAuc. In the existing method, all potential bidders can tender their bids to the server. When REPs forward bid invitations and bids between the server and their one-hop neighbors, some potential bidders may not be able to tender their bids if they have not received bid invitations from REPs due to insufficient connectivity to REPs and/or the misbehavior of one or more REPs (Figure 5b).

Our goal is to obtain data associated with the same location with the same amount of reward within the same amount of time as in the existing method (i.e., to obtain the same result as in the existing method) without increasing traffic between the server and mobile hosts compared with the existing method. Let \( S_p(q) \) denote a set of mobile hosts that receive a bid invitation for the data request \( q \) in the proposed method and let \( S_e(q) \) denote a set of mobile hosts that receive a bid invitation for the data request \( q \) in the existing method. \( S_e(q) \) is expressed as a set of mobile hosts that exist within circle \( C \) shown in Figure 4. To obtain the same result as in the existing method, a bid invitation for data request \( q \) needs to be delivered to a set of mobile hosts \( S_p(q) \) so that \( S_p(q) \supseteq S_e(q) \) is satisfied (Figures 5a and 6).
To satisfy $S_p(q) \supseteq S_v(q)$, it is necessary to (i) select suitable REPs to deliver a bid invitation for data request $q$ to $S_v(q)$ and (ii) mitigate the effects of the behavior of misbehaving REPs that fail to fulfill the role of REP; (i) can be decomposed into two sub-challenges, that is, (i-a) stimulating mobile hosts at suitable positions to volunteer to serve as REPs and (i-b) selection of REPs at the server to send a bid invitation to. If the set of mobile hosts that can communicate with REPs via D2D communication does not include all mobile hosts, some mobile hosts cannot receive a bid invitation from a server through REP. This means the server may not be able to send a bid invitation to a prospective mobile host. “At suitable positions” in (i-a) means that the set of mobile hosts that can communicate with REPs via D2D communication includes all mobile hosts. If (i-a) is satisfied, the server can select REPs to send a bid invitation to, that is (i-b). In the following discussion, these three challenges, (i-a), (i-b), and (ii), are explained in detail.

(i-a) Stimulating mobile hosts at suitable positions to volunteer to serve as REPs

The system needs REPs that satisfy $\bigcup_{i \in S_R} G_i = S_T$, where $S_R$ is a set of REPs, $G_i$ is a set of REP $i$ and its one-hop neighbors, and $S_T$ is a set of all mobile hosts (Figure 7). If $\bigcup_{i \in S_R} G_i \neq S_T$ (Figure 8), the server cannot send a bid invitation to some mobile hosts and there is a possibility that $S_p(q) \supseteq S_v(q)$ is not satisfied. In addition, it is necessary that a mobile host volunteers to serve as a REP for its one-hop neighbors within time $T'_q$ and that the server sends a bid invitation to mobile hosts via the REP so that the bid invitation can be delivered to all mobile hosts in $S_i(q)$ within the same amount of time as in the existing method.

(i-b) Selection of REPs to which the server sends a bid invitation for a data request

When the server selects a set of REPs $R_q$ to which it sends a bid invitation for a data request $q$, it needs to try to ensure that $S_p(q) = \bigcup_{i \in R_q} G_i \supseteq S_i(q)$ can be satisfied while minimizing $|R_q|$ to reduce cellular traffic between the server and mobile hosts. For this reason, the server needs to select $R_q$ based on the locations of all mobile hosts (i.e., REPs and their one-hop neighbors), as in the existing method, to satisfy $S_p(q) \supseteq S_i(q)$. In the proposed method, a REP communicates with the server through a cellular network on behalf of its one-hop neighbors. Thus, a REP needs to send the locations of its one-hop neighbors in addition to its own location. However, if a REP sends the locations of all its one-hop neighbors, cellular traffic increases as the number of one-hop neighbors increases. To reduce cellular traffic, it is useful to put the mobile host location data into a small fixed-size packet by abstracting the location of each mobile host in some way. Additionally, the method with which the server selects $R_q$ needs to be able to accommodate abstracted locations of the REP and its one-hop neighbors.
Mitigating the effects of misbehaving REPs that fail to fulfill the role of REP

In the presence of misbehaving REPs, RevAuc does not work correctly. Specifically, a misbehaving REP may fail to forward bid invitations to its one-hop neighbors or it may fail to forward bids from its one-hop neighbors and itself to the server.

4.3. Basic Strategy to Solve Design Challenges

(i-a) Stimulating mobile hosts at suitable positions to volunteer to serve as REPs

Each mobile host volunteers to serve as a REP if neither its one-hop neighbors nor itself has been a REP within time $T'_{b}$ that is shorter than $T_{q}$. $T'_{b}$ is a configuration parameter set at a value less than or equal to $T_{q}$. Thus, one mobile host volunteers to serve as a REP for its one-hop neighbors at an interval shorter than or equal to $T_{q}$. Then, the REP sends a Bid Invitations Query message to the server. To deal with changes in one-hop neighbors due to their movement, a REP finishes its role as REP after sending a Bids message to the server, and another mobile host volunteers to serve as a new REP.

Later, we will introduce another type of REP to mitigate the effects of misbehaving REPs that fail to fulfill the role of REP, and we need to distinguish them from the normal REPs described here. We denote a normal REP as a primary REP (P-REP) hereafter and the REP that will be introduced later as a secondary REP (S-REP). However, in the following, we just use the term REP when there is no need to distinguish between P-REP and S-REP.

(i-b) Selection of REPs to which the server sends a bid invitation for a data request

The server estimates abstracted locations of mobile hosts and selects a subset of REPs to which it sends a bid invitation for a data request in a similar way to the case with the existing method using a neighbor existence area (NEA). An NEA abstracts information on the locations of a REP and its one-hop neighbors with a constant data size. An NEA consists of a small circle (center O, radius $r$) that covers the area where the REP and its one-hop neighbors exist and the number of mobile hosts $N$ (Figure 9). The server receives NEA information from each REP with a Bid Invitations Query message. For each data

Figure 7. Example of a case where $\bigcup_{i \in S_p} G_i = S_T$.

Figure 8. Example of a case where $\bigcup_{i \in S_p} G_i \neq S_T$. 
request, the server calculates the minimum value of $R$, the radius of the circle $C$ centered on the POI for data request $q$, so that the expected number of mobile hosts in circle $C$ can be greater than or equal to the configuration parameter $\theta_n$ (Figure 10). Mobile hosts in the NEA are considered to be in circle $C$ only when circle $C$ covers the NEA. The server determines that a P-REP or one-hop neighbors of the P-REP are likely to obtain data for data request $q$ if the NEA reported by the P-REP overlaps with circle $C$. Then, the server sends a bid invitation for data request $q$ to the P-REP (Figure 10). Thus, $S_p(q) \supseteq S_e(q)$ will be satisfied.

![Figure 9. Example of neighbor existence area.](image)

Next, we explain how the above strategy can make $S_p(q)$ include a mobile host that is not in $S_e(q)$ and how it can also make $S_p(q)$ exclude a mobile host in $S_e(q)$. With this strategy, the server sends a bid invitation for data request $q$ to a P-REP and its one-hop neighbors if the NEA reported by the P-REP overlaps with circle $C$. Because the NEA includes areas where neither the P-REP nor its one-hop neighbors exist, the P-REP and its one-hop neighbors can receive bid invitations for more data requests than in the existing method. Therefore $S_p(q)$ can include a mobile host that is not in $S_e(q)$ (Figure 6). Moreover, a mobile host can be a one-hop neighbor of different REPs. Thus, each mobile host may be counted multiple times in the number of mobile hosts $N$ of different NEAs (e.g., Figure 11). In this case, the radius $R$ of circle $C$ can be estimated to be smaller than its actual value because the number of mobile hosts in circle $C$ is estimated to be greater than the actual value. If the estimated value of $R$ becomes less than its actual value, mobile hosts that actually need to receive a bid invitation for a data request may not receive the bid invitation. Therefore, $S_p(q)$ may not include a mobile host that is in $S_e(q)$.
Existence Area

(i) Mitigating the effects of misbehaving REPs that fail to fulfill the role of REP

We mitigate the effects of misbehaving REPs by making additional mobile hosts among the one-hop neighbors of a P-REP work as REPs and having the server send the same bid invitations to those REPs. For this purpose, one or more S-REPs volunteer to serve as a REP triggered by the appearance of a P-REP. When a mobile host volunteers to serve as a P-REP, the P-REP broadcasts a notification message, and the one-hop neighbors of the P-REP know the P-REP has volunteered. S-REPs volunteer among the one-hop neighbors of the P-REP. The server sends the same bid invitations to S-REPs that it sent to the P-REP corresponding to S-REPs. An S-REP sends a bid invitation to the one-hop neighbors of the P-REP and receives bids from them via D2D communication (Figure 12). After sending a Bids message to the server, an S-REP finishes its role as S-REP because the corresponding P-REP also stops being a P-REP after sending a Bids message to the server. If at least one REP among the P-REP and S-REPs is non-misbehaving, each mobile host can receive a bid invitation and send its bid to the server. Thus, the effects of misbehaving REPs can be mitigated. Note that if a mobile host does not broadcast a notification message when it becomes a P-REP, another mobile host becomes a P-REP, and its one-hop neighbors can receive a bid invitation and send their bids to the server.

Figure 11. Example of mobile hosts that are included in the NEA of different P-REPs.

Figure 12. S-REP sending a bid invitation and receiving bids.

Note that we assume the following to avoid the situation where a REP may tamper with the bidding message or upload the wrong bidding results to increase its profits. We assume that a sensing service provider distributes applications to mobile hosts such as smartphones and those applications execute MH’s processing in the proposed method on mobile hosts. We also assume that the sensing service provider detects unintended changes to the applications by some method such as remote attestation and prevents the modified applications from participating in MCS. Thus, the sensing service provider can prevent the
situation where a REP may tamper with the bidding message or upload the wrong bidding results to increase its profits.

4.4. Procedure of the Proposed Method

In this section, we explain the key details of the RevAuc procedure executed by a REP (Figure 13). We define an evaluation function to calculate bid scores and assume that both the server and mobile hosts know this function.

1. Some mobile hosts volunteer to serve as a P-REP or an S-REP that requests bid invitations from a server. The REP sends a Bid Invitations Query message that includes information on whether the sender is a P-REP or an S-REP, query ID, and the NEA.

2. When the server receives a Bid Invitations Query message from a P-REP, the server selects data requests included in bid invitations sent to the P-REP based on the NEA in the Bid Invitations Query message. Then, the server sends the bid invitations to the P-REP with a Bid Invitations message. After receiving a Bid Invitations Query message from an S-REP, the server identifies a Bid Invitations Query message from a P-REP that corresponds to the S-REP by matching the query ID in the Bid Invitations Query message. Then, the server sends the same bid invitations to the S-REP as the bid invitations it sent to the P-REP.

3. After receiving a Bid Invitations message, a REP sends a D2D Bid Invitations message containing the bid invitations received from the server via two-hop D2D broadcast. Two-hop D2D broadcast is used to allow the one-hop neighbors of the P-REP to receive the D2D Bid Invitations message from the S-REP. Note that the one-hop neighbors of the S-REP are not the same as the one-hop neighbors of the P-REP.

4. After receiving a D2D Bid Invitations message, a mobile host determines whether to tender a bid in response to the invitation. If the mobile host tenders a bid, it sends it with a D2D Bids message to the REP via two-hop D2D broadcast.

5. Each REP waits for a configuration parameter time $T_w$ after sending a D2D Bid Invitations message. Then, for each bid invitation, the REP selects the bid with the highest evaluation score among its own bid and other bids received with D2D Bids messages while waiting for time $T_w$. The REP sends the selected bids to the server in a Bids message.

6. The server selects the successful bidder, which is the host that sent the bid with the highest evaluation score among the bids received within time $T_w$ after the data request was received. It sends a Bid Result message to the successful bidder via the cellular network to ask the mobile host to obtain and transmit the requested data.

7. After receiving a Bid Result message, the successful bidder mobile host obtains the requested data and transmits it to the server in a Data message via the cellular network.

![Figure 13. Example of the RevAuc procedure carried out by a P-REP.](image-url)
Because the actions in step (1) are a little complicated, we explain more details of this step below.

4.4.1. How to Stimulate Mobile Hosts at Suitable Positions to Volunteer to Serve as REPs

In the proposed method, mobile hosts decide whether to volunteer to serve as a REP based on the messages exchanged through D2D communication with its one-hop neighbors. The procedure to stimulate mobile hosts to volunteer to serve as REPs is explained here and in Figure 14.

(1) Each mobile host decides whether it should volunteer to serve as a P-REP at every time step $T_q'$. The start of the cycle is selected randomly by each mobile host. Specifically, a mobile host volunteers to serve as a P-REP and sends a Bid Invitations Query message to the server if it has neither received any D2D Primary Notification message from its one-hop neighbors nor been a P-REP within time $T_q'$.

(2) On becoming a P-REP, the mobile host sends a D2D Primary Notification message through one-hop D2D broadcast to notify its one-hop neighbors of its current status as a P-REP.

(3) For volunteering to serve as an S-REP, a mobile host (MH-X) that receives a D2D Primary Notification message waits for a random wait time between 0 and configuration parameter $T_s$. If the number of D2D Secondary Notification messages received by MH-X while waiting is less than $(M - 1)$, where $M$ is a configuration parameter, MH-X volunteers to serve as an S-REP. When volunteering to serve as an S-REP, the S-REP sends a Bid Invitations Query message to receive the same bid invitations as a corresponding P-REP.

(4) At the same time, the S-REP sends a D2D Secondary Notification message through two-hop D2D broadcast to notify the one-hop neighbors of the P-REP that it is currently an S-REP.

(5) If the number of D2D Secondary Notification messages received by MH-X while waiting is equal to or greater than $(M - 1)$, MH-X does nothing. Thus, at least $(M - 1)$ mobile hosts among the one-hop neighbors of a P-REP volunteer to serve as S-REPs.

![Figure 14. Example of the selection of a P-REP and S-REPs (in the case of $M = 2$).](image-url)
4.4.2. How REP Calculates an NEA

To calculate an NEA, a REP obtains information on the location and the number of its one-hop neighbors. Each mobile host broadcasts its location information every configuration parameter time $T_b$ to its one-hop neighbors via D2D communication. The mobile host uses the location information broadcast by its one-hop neighbors to track their number and location.

Next, the REP calculates an NEA based on the location and number of its one-hop neighbors. The REP calculates the center $O$ and the radius $r$ of a circle that covers the locations of itself and its one-hop neighbors and $N$, the number of mobile hosts including itself, and records $(O, r, N)$. The center $O$ is set at the centroid of the locations of the REP and its one-hop neighbors, and $r$ is set at the maximum distance between the center $O$ and the REP or its one-hop neighbors. The REP sends the recorded $(O, r, N)$ values as an NEA in a Bid Invitations Query message.

5. Performance Evaluation Results and Discussion

We evaluated the effectiveness of the proposed method. Specifically, we used simulations to evaluate the reduction in cellular traffic, the mitigation of the effects of misbehaving REPs that fail to fulfill the role of REP, and the effect of the configuration parameter $T_b$ on successful bidders’ bids. Two movement scenarios for mobile hosts, that is, a pedestrian case and vehicle case, were simulated. Mobile hosts move in a 500 m by 500 m square area according to the random waypoint model (RWP) [22] for the pedestrian case and a 1000 m by 1000 m square area by RWP for the vehicle case. The purpose of the vehicle case is to evaluate how the higher velocity of the mobile hosts affects the effectiveness of the proposed method. The amount of reward requested by each mobile host is set at a fixed value determined for each mobile host by a uniform random number at the start of a simulation. Each mobile host tenders a bid under the condition that it fulfills a data request at a location nearest the POI in its planned trajectory. A data request for a randomly selected POI is generated every 5 s. The evaluation score of a bid is calculated based on the distance between the POI and the location where the mobile host obtains the data, the time that it takes for the mobile host to obtain the data, and the reward requested by the mobile host. The event-driven simulation program was written in the C programming language.

5.1. Evaluation Viewpoints

5.1.1. Reduction in Cellular Traffic

We confirmed the reduction in cellular traffic compared to the existing method by comparing the total number of cellular packets used for RevAucs. The cellular packets for Bid Invitations Query messages, Bid Invitations messages, and Bids messages were counted. We compared the amount of cellular traffic in terms of the number of 128-byte radio interface packets because such packets are generally used as units for calculating network usage charges [23–25]. Table 1 shows the size of messages used in our simulation, which assumes that a message is transported in an IPv4/UDP packet and bid invitations and bids are signed by SHA-256 with RSA-2048 (i.e., 256-byte signature).

To determine whether data were obtained in the proposed method at almost the same location for almost the same reward as in the existing method, we also evaluated the differences in the evaluation scores of successful bids. Hereafter, we refer to the number of data requests for which an evaluation score of a successful bid is different between two simulation scenarios by "the number of different successful bids". We measured the ratio of the number of different successful bids between the proposed method and the existing method to a total of 1970 data requests. For example, if the successful bids for a total of three data requests with the proposed methods are A, B, and C, and the ones with the existing method are A, B, and D, then the ratio of the number of different successful bids between the proposed method and the existing method is 1/3 because C and D are different. Simulations were conducted 10 times with different random number seeds for 1000 s with
250, 500, and 750 mobile hosts in an environment without misbehaving REPs. We assumed crowded places such as an urban area in choosing the number of mobile hosts.

Table 1. Size of messages in bytes and number of packets.

<table>
<thead>
<tr>
<th>Message</th>
<th>Existing Method [Byte (# of Packets)]</th>
<th>Proposed Method [Byte (# of Packets)]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bid Invitations Query</td>
<td>40 (1)</td>
<td>52 (1)</td>
</tr>
<tr>
<td>Bid Invitations (No bid invitation)</td>
<td>290 (3)</td>
<td>290 (3)</td>
</tr>
<tr>
<td>Bid Invitations (One bid invitation)</td>
<td>322 (3)</td>
<td>322 (3)</td>
</tr>
<tr>
<td>Bid Invitations (Two bid invitations)</td>
<td>354 (3)</td>
<td>354 (3)</td>
</tr>
<tr>
<td>Bid Invitations (Three bid invitations)</td>
<td>386 (4)</td>
<td>386 (4)</td>
</tr>
<tr>
<td>Bids (One bid)</td>
<td>326 (3)</td>
<td>326 (3)</td>
</tr>
<tr>
<td>Bids (Two bids)</td>
<td>358 (3)</td>
<td>618 (5)</td>
</tr>
<tr>
<td>Bids (Three bids)</td>
<td>390 (4)</td>
<td>910 (8)</td>
</tr>
</tbody>
</table>

5.1.2. Effects of Misbehaving REPs and Mitigation of the Effects with S-REPs

We confirmed the effects of misbehaving REPs that fail to fulfill the role of REP and the mitigation of these effects with S-REPs by evaluating the differences in the evaluation scores of successful bids between one case without a misbehaving mobile host and four cases with some misbehaving mobile hosts, that is, (a), (b), (c), and (d) in Table 2. We measured the number of different successful bids between the cases with and without misbehaving mobile hosts among 1970 data requests. Simulations were conducted 10 times with different random number seeds for 1000 s with 250 and 500 mobile hosts in an environment where mobile hosts were selected by a uniform random number at the beginning of a simulation to drop packets intentionally.

Table 2. Four cases with misbehaving hosts.

<table>
<thead>
<tr>
<th>Misbehaving Hosts</th>
<th>5% Misbehaving Hosts</th>
<th>30% Misbehaving Hosts</th>
</tr>
</thead>
<tbody>
<tr>
<td>No S-REP per P-REP</td>
<td>(a)</td>
<td>(c)</td>
</tr>
<tr>
<td>One S-REP per P-REP</td>
<td>(b)</td>
<td>(d)</td>
</tr>
</tbody>
</table>

5.1.3. Effect of Configuration Parameter $T_b$ on Successful Bids

We evaluated how the configuration parameter $T_b$ affects the successful bids because this parameter can be modified to reduce D2D communication messages. $T_b$ specifies the intervals between a mobile host’s broadcasts of its location information via D2D communication. Because the D2D communication capacity has an upper limit, reducing D2D messages by making $T_b$ larger can be useful. However, a larger $T_b$ can affect a successful bid for the following reason. According to the way an NEA is calculated by REP (described in Section 4.4.2, the location information of one-hop neighbors used for calculating the NEA becomes more outdated with a larger $T_b$. Older location information for one-hop neighbors makes the NEA less accurate because the hosts are mobile. As the server’s selection of REPs to which it sends a bid invitation depends on the NEA, as described in Section 4.3, changes in the NEA can change the set of REPs to which the server sends a bid invitation. Therefore, the evaluation score of a successful bid can change when parameter $T_b$ is increased. We measured the number of successful bids with different evaluation scores between the cases with $T_b = 1$ and $T_b = 3$ among 1970 data requests. We also measured the number of D2D messages for broadcasting mobile host locations and the number of D2D messages for
RevAucs. Simulations were conducted 10 times with different random number seeds for 1000 s with 500 mobile hosts in an environment without misbehaving REP. We chose 1 s as the value of $T_b$ considering the balance between the accuracy of the location information of the mobile hosts and the number of D2D messages. We chose 3 s as the maximum value of $T_b$ so that a mobile host close to a REP is less likely to move out of the D2D communication range of the REP during $T_b$.

5.2. Simulation Model

5.2.1. Communication Model

D2D communication was modeled using a unit disc model without packet loss by interference or congestion, with a 50-m propagation range for the pedestrian case and a 100-m propagation range for the vehicle case (Figure 15). The transmission rate of D2D communication was set at 1 Mbps. The area in which mobile hosts moved was 500 m by 500 m for the pedestrian case and 1000 m by 1000 m for the vehicle case. A data request was generated every 5 s, and the POI was selected by a uniform random number. Both the propagation range of D2D communication and the length of a side of the area for the vehicle case is twice as large as that for the pedestrian case. Thus, the average number of mobile hosts in the D2D communication range of a mobile host for the vehicle case is expected to be equal to that for the pedestrian case when the total number of mobile hosts for the vehicle case is equal to the total number of mobile hosts for the pedestrian case.

![Figure 15. Simulation model for vehicle case (communication and mobility).](image)

5.2.2. Mobility Model

Mobile hosts moved by RWP, and they repeatedly stopped for a random period and moved to a randomly selected destination at a randomly selected velocity (Figure 15). The period for which a mobile host stops was selected to be between 0 s and 5.0 s by a uniform random number. The destination of a mobile host was selected by a uniform random number. The density of the mobile hosts within the movement area varied from location to location, even in this scenario [26–28]. We used different velocity ranges for the pedestrian and vehicle cases. For the pedestrian case, the host velocity was between 0.1 and 3.0 m/s, as determined by a uniform random number. For the vehicle case, the host velocity was selected to be between 6.0 and 15.0 m/s by a uniform random number.

5.2.3. RevAuc Behavior Model

The reward requested by a mobile host was a fixed value selected between 1.0 and 10.0 at the simulation start for each mobile host. The maximum amount of reward paid to a mobile host by the server was set at 10.0. Thus, insufficient reward did not prevent a mobile host from obtaining the requested data. Each mobile host determined the value of the bid it tendered. If the current movement state (stopped or moving) ended before the time when a
A successful bidder was determined (auction end time), the mobile host did not tender a bid because it did not know its location at the auction end time. Otherwise, the mobile host calculated the location and the time when it would be closest to the POI after the auction end time in the current movement state and created a bid assuming that it would obtain the requested data at that time. Finally, if the evaluation score of the calculated bid was greater than 0, the mobile host tendered the bid (Figure 16).

![Simulation model for vehicle case](image)

**Figure 16.** Example of decision-making by a mobile host on whether to tender a bid based on an evaluation score of a bid.

### 5.2.4. Evaluation Function Used to Calculate Bid Scores

In both the existing and proposed methods, the server and mobile hosts use an evaluation function to calculate the score of a bid in the RevAuc procedure. The evaluation function $E$ is defined as the average of functions $E_Q$, $E_T$, and $E_C$. The evaluation function $E_Q$ evaluates the distance between the POI and the location where a mobile host will obtain data. The evaluation function $E_T$ evaluates the time between an auction end time and the time when the mobile host will obtain data (data acquisition time). The evaluation function $E_C$ evaluates the reward requested by the mobile host:

$$ E = \begin{cases} (E_Q + E_T + E_C)/3 & (E_Q E_T E_C > 0) \\ 0 & (E_Q E_T E_C = 0) \end{cases} $$

(1)

Functions $E_Q$, $E_T$, and $E_C$ are defined as follows:

$$ E_Q(d) = (1 - d/120) \ (d \leq 100 \text{ m}), \ 0 \ (d > 100 \text{ m}) $$

(2)

where $d$ is the distance between the POI and the location where the mobile host will obtain data. $E_Q(d)$ is designed so that $E_Q(d)$ decreases linearly as $d$ increases while $d \leq 100$ m and $E_Q(d)$ becomes zero if $d > 100$ m, which means that data obtained at a place closer to the POI is more valuable:

$$ E_T(t) = (1 - t/40) \ (t \leq 30 \text{ s}), \ 0 \ (t > 30 \text{ s}) $$

(3)

where $t$ is data acquisition time ($t \geq 0$). $E_T(t)$ is designed so that $E_T(t)$ decreases linearly as $t$ increases while $t \leq 30$ s and $E_T(t)$ becomes zero if $t > 30$ s, which means that data obtained sooner is more valuable:

$$ E_C(c, C_{\text{max}}) = \max[(1 - c/C_{\text{max}}), \ 0], $$

(4)

where $c$ is the reward requested by a mobile host ($c > 0$) and $C_{\text{max}}$ is the maximum reward paid to a mobile host by the server ($C_{\text{max}} > 0$).

Table 3 shows the parameter values used by the existing method and the method proposed here.
5.3. Reduction in Cellular Traffic Compared with the Existing Method

Figure 17 shows the ratio of cellular traffic in the proposed method to the cellular traffic in the existing method for the pedestrian case and the vehicle case (note that the 95% confidence interval is less than 0.0031 and not visible without exaggerating the y-axis of the plots). Figure 18 shows the ratio of the number of different successful bids between the existing method and the proposed method to 1970 data requests for the pedestrian case and the vehicle case. In Figure 18, the white bars show the ratio of the number of different successful bids with the evaluation score in the proposed method being higher to the 1970 data requests, and the gray bars show the ratio of the number of successful bids with the evaluation score in the existing method being higher to the 1970 data requests.

The results in Figure 17 show that, as the number of mobile hosts increases, the relative amount of cellular traffic decreases for both the pedestrian case and the vehicle case. This trend is reasonable because the number of one-hop neighbors of a mobile host is expected to increase as the number of mobile hosts increases. From Figure 17, we confirmed that the proposed method can reduce cellular traffic compared with the existing method.

The results in Figure 18 show that, as the number of mobile hosts increases, so does the ratio for both the pedestrian and vehicle cases. In addition, the ratio of successful bids with the evaluation score in the proposed method being higher is much larger than the ratio of successful bids with the evaluation score in the existing method being higher. The former ratio increases as the number of mobile hosts increases for both the pedestrian and vehicle cases. In contrast, the ratio of data requests with a higher evaluation score for the existing method decreases as the number of mobile hosts increases for both the pedestrian and vehicle cases.

### Table 3. Parameters used by the existing method (left) and the proposed method (left and right).

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>( T_q )</td>
<td>12.45 s</td>
<td>( T'_{q} )</td>
<td>5 s</td>
</tr>
<tr>
<td>( T_L )</td>
<td>12.95 s</td>
<td>( T_s )</td>
<td>0.5 s</td>
</tr>
<tr>
<td>( \theta_h )</td>
<td>20</td>
<td>( M )</td>
<td>2</td>
</tr>
<tr>
<td>( R_{\text{max}} )</td>
<td>150 m</td>
<td>( T_b )</td>
<td>1 s</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( T_w )</td>
<td>1.6 s</td>
</tr>
</tbody>
</table>

![Figure 17](image1.png)

**Figure 17.** Ratio of cellular traffic in the proposed method to cellular traffic in the existing method.

![Figure 18](image2.png)

**Figure 18.** Ratio of the number of different successful bids between the existing method and the proposed method to the 1970 data requests.
The evaluation score of a successful bid in the proposed method can differ from the evaluation score of a successful bid in the existing method because the proposed method can include a mobile host in \( S_P(q) \) that is not included in \( S_e(q) \), and it can also exclude a mobile host from \( S_P(q) \) that is included in \( S_e(q) \), as explained in Section 4.3. From Figure 18, we can say that \( S_P(q) \) includes mobile hosts that are not in \( S_e(q) \) more often than \( S_P(q) \) excludes mobile hosts in \( S_e(q) \) in our simulation scenario. We also note that the ratio of the number of different successful bids is larger for the vehicle case than for the pedestrian case. We think that this result is because the faster movement of vehicle hosts causes REPs to communicate with hosts that are not in \( S_e(q) \) and faster movement of vehicle hosts causes REPs to fail to communicate with hosts in \( S_e(q) \) via D2D communication more often than in the pedestrian case.

5.4. Effects of Misbehaving REPs and Mitigation of the Effects by S-REPs

Figure 19 shows the number of different successful bids between cases with and without misbehaving mobile hosts for both the pedestrian and vehicle simulation cases. We chose 5% and 30% as the ratio of misbehaving hosts to evaluate how bid results are affected by the ratio of misbehaving hosts and how S-REPs can mitigate the effect of misbehaving hosts for low and high ratios. In all cases, the number of different successful bids is smaller with one S-REP per P-REP compared to the case with no S-REPs for both the pedestrian and vehicle cases. In addition, the evaluation scores of successful bids in cases with some misbehaving mobile hosts were lower than the evaluation scores of successful bids without misbehaving hosts when the evaluation scores were different. These results confirmed that S-REPs can mitigate the effects of misbehaving REPs that fail to fulfill the role of REP.

![Figure 19. Number of different successful bids in cases with misbehaving hosts.](image)

We illustrate the reason why the number of different successful bids with 500 mobile hosts is smaller than that with 250 mobile hosts in Figure 19. According to the method by which P-REPs volunteer, as described in Section 4.4.1, one mobile host can be a one-hop neighbor of multiple P-REPs. If a mobile host becomes a one-hop neighbor of multiple P-REPs, the mobile host can receive D2D Bid Invitations messages, including the same bid invitation from multiple P-REPs. Therefore, the mobile host can receive the same bid invitation from multiple P-REPs. This has the same effect as having additional S-REPs. As the number of mobile hosts increases, each mobile host is more likely to be a one-hop neighbor of multiple P-REPs, and the number of different successful bids becomes smaller.

We note two additional points. (i) The number of different successful bids between cases with and without misbehaving mobile hosts is larger in the vehicle case than in the pedestrian case. (ii) The reduction rate in the number of different successful bids between cases with and without misbehaving mobile hosts is smaller in the pedestrian case.
and 21.16 for the vehicle with $T_b = 1$. This implies that the average number of mobile hosts participating in a D2D RevAuc with a P-REP is larger in the vehicle case than in the pedestrian case. Therefore, more mobile hosts participate in RevAuc through misbehaving P-REPs in the vehicle case than in the pedestrian case. Regarding the reason for (ii), we think that the faster movement of vehicle hosts causes the one-hop neighbors of a P-REP to be unreachable through two-hop D2D communication by S-REPs more often than for pedestrian hosts, and the effectiveness of S-REPs is reduced.

In the proposed method, S-REPs send and receive the same amount of data as P-REPs for Bid Invitations Query messages and Bid Invitations messages. The amount of data for Bids messages of S-REPs is the same as the amount of data for Bids messages of P-REPs in many cases even though the set of mobile hosts that P-REPs communicate with can be different from the set of mobile hosts that S-REPs communicate with. Therefore, the cellular traffic with $m$ S-REPs per P-REP becomes almost $(m + 1)$ times as large as the cellular traffic with no S-REP per P-REP. If we assume that two S-REPs are deployed for each P-REP, the ratio of cellular traffic in the proposed method to cellular traffic in the existing method will be $1.5(= \frac{1+2}{1})$ times as large as that in Figure 17. Namely, the ratios become about 1.5 for 250 MHzs, 0.93 for 500 MHzs, and 0.7 for 750 MHzs in the pedestrian case, and the ratios become about 1.4 for 250 MHzs, 0.92 for 500 MHzs, and 0.7 for 750 MHzs in the vehicle case. On the other hand, in Figure 19, one S-REP per P-REP reduces the number of different successful bids to 6 or less in the 5% misbehaving hosts case, which seems more probable than the 30% misbehaving hosts case. Based on these points, the demerit of the increase in successful bids to 6 or less in the 5% misbehaving hosts case, which seems more probable than the 30% misbehaving hosts case.

5.5. Effect of Configuration Parameter $T_b$ on Successful Bids

Figure 21 shows the number of different successful bids between the cases with $T_b$ (value used in other simulations) and with $T_b = 3$. Figure 21 shows the number of successful bids with the evaluation score of successful bids with $T_b = 3$ being higher ($T_b = 3 > T_b = 1$) and the number of data requests with the evaluation score of successful bids with $T_b = 1$ being higher ($T_b = 1 > T_b = 3$). The results in Figure 21 show that the number of different successful bids between the cases with $T_b = 3$ and $T_b = 1$ is small for
both the pedestrian and vehicle cases. The results also show that the number of different successful bids between the case with $T_b = 3$ and the case with $T_b = 1$ is larger in the vehicle case than in the pedestrian case. Figure 20 shows the average number of D2D messages per host per second. From Figure 20, we can see that the number of transmitted and received D2D messages was reduced with $T_b = 3$. From these results, we confirmed that the number of D2D messages can be reduced by changing $T_b$ from 1 to 3 s with few changes in the successful bids for this simulation scenario.

![Figure 20. Number of D2D messages.](image)

The reason why the number of different successful bids is larger in the vehicle case than in the pedestrian case is related to the NEA accuracy. As explained in Section 5.1.3, host movement reduces the accuracy of NEA information. As the mobile hosts move faster, the error in the information on the location of the mobile hosts used to calculate the NEA increases, which is further increased with a larger $T_b$. This explains why the number of different successful bids in the vehicle case is larger than in the pedestrian case. In Figure 20, the average number of transmitted and received D2D RevAuc messages is a little bit larger with $T_b = 3$ than with $T_b = 1$ in both the pedestrian case and the vehicle case. This indicates that the number of mobile hosts participating in a RevAuc through D2D communication is larger with $T_b = 3$ than with $T_b = 1$. We think that this is the reason why the number of successful bids with higher evaluation scores with $T_b = 3$ is greater than the number of successful bids with higher evaluation scores with $T_b = 1$.

6. Conclusions

With this paper, we proposed a communication method for RevAuc-based MCS that can reduce the amount of cellular traffic between a server and mobile hosts. The proposed method reduces cellular traffic by recruiting REPs to forward bid invitations and bids between the server and their one-hop neighbors via D2D communication. We confirmed the effectiveness of the proposed method through simulation. With the proposed method, a successful bidder can obtain data at almost the same location for almost the same reward within almost the same amount of time as in the existing method while reducing the cellular traffic by more than half in simulation cases with 750 mobile hosts. In addition, the proposed method can mitigate the effects of misbehaving mobile hosts that fail to fulfill the role of REP. We also confirmed that the number of D2D communication messages can be reduced by changing the configuration parameter $T_b$ from 1 to 3 s with few changes in the successful bids.

There are several issues for further study. The first issue is to improve the selection of data requests in bid invitations sent to REPs by considering the predicted trajectory of mobile hosts. The second issue is to perform additional evaluations of the proposed method with other simulation scenarios, such as the case where mobile hosts change their trajectory to obtain data. We hope that the proposed method will enhance the viability of the deployment of services for the collection of sensing data in near real time when D2D communication technologies, such as vehicle to vehicle (V2V), are widely adopted in our society.
Author Contributions: Conceptualization, T.M. and S.I.; methodology, T.M. and S.I.; software, T.M.; validation, T.M. and S.I.; investigation, T.M.; data curation, T.M.; writing—original draft preparation, T.M.; writing—review and editing, S.I. and T.I.; visualization, T.M. and S.I.; supervision, S.I. and T.I.; project administration, T.I.; funding acquisition, T.I. and S.I. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

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


