A Proposal of an Incentive-Based Crowdsensing Method Using a Cellular Network and D2D Communication

Tetsushi Matsuda
Mitsubishi Electric Corp./Graduate School of Science and Technology, Shizuoka University
Japan
Matsuda.Tetsushi@dh.MitsubishiElectric.co.jp

Toru Inada
Mitsubishi Electric Corp.
Japan
Inada.Toru@ak.MitsubishiElectric.co.jp

Susumu Ishihara
Shizuoka University
Japan
ishihara.susumu@shizuoka.ac.jp

***Abstract*— The demand for services that provide current sensing data (data) at a specified location is expected to grow in the future. To realize such services with lower sensor installation costs and lower operating costs, we consider using pull-type mobile crowdsensing with reverse auction (pull-type reverse auction) that determines the amount of incentive paid to mobile hosts (MHs) which obtain data with reverse auction. In pull-type reverse auction, more cellular traffic between a server and MHs can increase the cost of service operation. In this paper, we propose a new method for pull-type reverse auction that reduces cellular traffic between a server and MHs by utilizing Device to Device (D2D) communication. Through simulation, we confirmed that the proposed method can reduce cellular traffic compared to the existing method for pull-type reverse auction while generally selecting a successful bidder which obtains data at the same location for the same amount of reward within the same amount of time as in the existing method.**

***Keywords*— D2D Communication, Mobile Crowdsensing, MCS, Reverse Auction, Incentive**

I. Introduction

The demand for services that provide current sensing data (data) at a specified location is expected to grow in the future. Examples of such services are as follows. 1) A driver obtains information on traffic status ahead when the driver is in a traffic jam. 2) A sight-seer queries how crowded his or her destination is. The cost for installing, operating and maintaining sensors that are necessary to collect data is a problem in providing such services. Mobile crowdsensing (MCS) [1] is a method which can collect sensing data without pre-installing sensors in the field. An MCS system consists of mobile hosts (MHs) with sensors such as smartphones and cars, which are willing to help a service obtain data, and a server operated by a service provider, which collects data with the help of the MHs. Let us assume that (i) communication between the server and the MHs is through a cellular network, (ii) data request sources are not limited to an MH, and (iii) there is no constraint on the relationship between the location of a data requester and a Point of Interest (POI) of the data request. In this paper, we propose a method for incentive-based MCS that reduces cellular traffic between the server and the MHs by utilizing Device to Device (D2D) communication.

Incentive-based MCS [2] is a method that makes data more likely to be obtained by motivating MHs to obtain the data in MCS. A goal in incentive-based MCS is for a server to obtain data of higher quality with less incentive. Reverse auction (RevAuc) [2][3][4][5], in which a bidder requesting the least reward wins a bid, is a solution to achieve the goal. In MCS using RevAuc, the server acts as an auctioneer which determines a successful bidder and sends bid invitations to MHs. An MH sends a bid that includes information on the quality of data which it plans to obtain and the reward that it requests to the server. The server requests the MH that tendered a bid with the highest evaluation score among bids sent by MHs to obtain data. The evaluation score of a bid is calculated based on the reward requested by the bid and the quality of data proposed by the bid.

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 an MH when it receives a query for bid invitations from the MH. Another is push-type, in which the server sends a bid invitation to MHs spontaneously [1]. In this paper, we assume the use of pull-type RevAuc. It is easier to build a client-server type system with pull-type RevAuc than with push-type RevAuc because an action is started by the MHs in pull-type RevAuc.

Reduction of communication traffic between a server and MHs is a problem when realizing a service that collects current sensing data at a specified location by pull-type RevAuc. The reason is as follows. All MHs need to query bid invitations and receive bid invitations periodically to quantify three characteristics, that is, Timeliness of data acquisition, Quality of data and Cost for data acquisition. Even though the data size of these messages and bid messages is small, this traffic can markedly increase the cost of cellular communication because all MHs transmit these messages periodically. Thus, reduction in traffic for a query for bid invitation, bid invitation and bid is a problem in pull-type RevAuc.

In this paper, we propose a new method for pull-type RevAuc that reduces cellular traffic between a server and MHs compared to the existing method for pull-type RevAuc. The proposed method has three features as follows. 1) Reduced cellular traffic by utilizing D2D communication. 2) Selection of a successful bidder so that it can obtain data at the same location for the same amount of reward within the same amount of time as in the existing method. 3) Mitigation of effects of malicious REPs which drop packets intentionally.

In what follows, we describe related studies and the existing method that the proposed method is based on before delineating the new method and confirming its efficacy through simulations.

II. RELATED WORK

Existing studies on RevAuc in MCS [3][4][5] do not mention how to reduce cellular traffic between a server and MHs. We review related work on methods for reducing cellular traffic between a server and MHs in MCS, although notably those related studies do not focus on MCS using RevAuc.

Mota et al. [6] propose D2D Extended Sensing (D2D-ES) and D2D Content Dissemination (D2D-CD), which reduce cellular traffic between a server and MHs with the help of representative MHs that relay messages between a server and MHs through D2D communication in MCS. However, they do not show how to select representative MHs nor how to use D2D-ES and D2D-CD for pull-type RevAuc. Thus, cellular traffic of pull-type RevAuc cannot be reduced by their approach.

Matsumoto et al. [7] propose a method for reducing cellular traffic by uploading only image data that are requested by drivers in a server. Their method handles image data taken and kept by cars spontaneously and it sends metadata about the image data held by cars to the server with the help of D2D communication. They do not mention how to extend their method for pull-type RevAuc.

Pu et al. [8] propose a method with which an MH (requester) requesting data finds MHs that obtain data only through D2D communication without using a server. Their method cannot obtain data at a location far away from the current location of the data requester. Thus, again, their method is not suitable for our purpose, which assumes no constraint on POI.

In summary, the reviewed existing studies cannot be applied to reducing cellular traffic of pull-type RevAuc as it is.

III. EXISTING METHOD

In this section, we explain the procedure of pull-type RevAuc that uses only cellular communication (Figure 1). It is considered that this method is equivalent to the method of MCS using RevAuc assumed in existing studies [3][4]. In pull-type RevAuc, a server sends a bid invitation for a currently effective data request to an MH if the server judges that the MH is highly likely to obtain the data requested by the data request.

(1) An MH sends a "Bid Invitation Query message", which includes its location, to a server with cycle T_q.

(2) The server selects data requests for a bid invitation based on the location of the MH included in Bid Invitation Query messages. The server sends a bid invitation to the MH by a "Bid Invitation message", which includes data request (POI, specification of data to be obtained), the maximum reward to be paid and the time when a successful bidder is selected (bid end time). In this paper, we assume that the successful bidder for a data request is selected time T_L after the data request is generated.

(3) The MH determines the bid for a bid invitation based on the information included in the bid invitation. It sends its bid to the server by a "Bid message". The bid includes host ID, expected location for data acquisition, requested reward, and expected time when the data will be obtained. Note that an evaluation function is defined to calculate a bid's evaluation score and the server knows this.

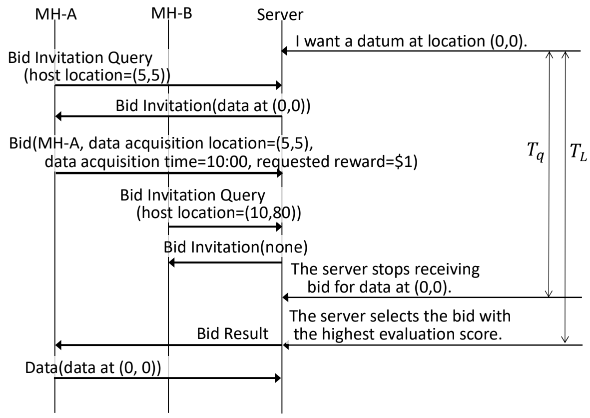

Figure 1 Example sequence of the existing method

(4) The server selects the successful bidder, which means the MH that sent a bid with the highest evaluation score among the bids received within time T_q after the data request was generated. It sends a "Bid Result message" to the successful bidder to request the MH to obtain data.

(5) An MH judges that it is not the successful bidder if it does not receive a Bid Result message from the server by a bid end time included in a bid invitation. When receiving a Bid Result message, an MH obtains data according to the information in the Bid Result message and it sends a "Data message" that includes data request ID and data to the server.

When receiving a Bid Invitation Query message from an MH, a server selects data requests included in bid invitations as follows. The server includes currently effective data requests for which the MH is likely to obtain data in bid invitations sent to the MH. A data request is currently effective if time less than T_q has elapsed since its generation. For example, the server judges whether an MH is likely to obtain data based on the distance between the POI and the current location of the MH.

Herein, we compare the existing method with the proposed method by simulation. In the simulation model, we assumed that an MH obtains a requested data in its planned trajectory and a server determines if an MH is likely to obtain data for a data request as follows. Based on the location of an MH included in a Bid Invitation Query message, the server updates a database of the latest location of MHs (host location DB). Then, 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 MHs in circle C is greater than or equal to configuration parameter θ_n (Figure 2). If the calculated value of R is larger than configuration parameter $R_{\max}$, then R is set to $R_{\max}$. The server judges that an MH is likely to obtain data for the data request if the location of the MH recorded in the host location DB is in circle C. Taking planned trajectory of MHs into consideration in the model is for further study.

IV. PROPOSED METHOD

A. RevAuc procedure using D2D communication

Compared to the existing method, the proposed method reduces cellular traffic between the server and MHs by utilizing

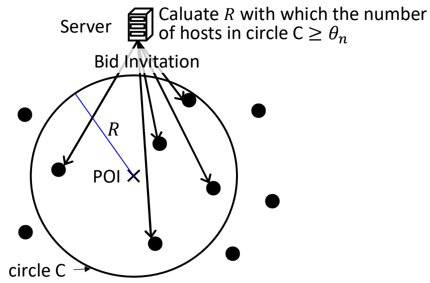

Figure 2 How to select MHs that can obtain data for a data request (existing method, for $\theta_n = 5$)

D2D communication. Among MHs which can communicate with an MH (MH-A) by one-hop D2D communication (one-hop neighbors of MH-A), an MH stands as a representative MH (REP) if no MH has stood as a REP among one-hop neighbors of MH-A for configuration parameter time T'_q. A REP sends a Bid Invitation Query message when it stands as REP. Then, the REP forwards a bid invitation received from the server to its one-hop neighbors and receives bids from its one-hop neighbors and then sends the bid with the highest evaluation score to the server (Figure 3 (a)).

B. Design challenges in obtaining data at the same location for the same amount of reward as in the existing method

Design challenges arise when REPs are used to execute 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 of potential bidders may not be able to tender their bids if no REP forwards bid invitations to those potential bidders (Figure 3 (b)). Our goal is to obtain data at the same location for 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) with less traffic between the server and MHs than in the existing method. 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 MHs $S_p(q)$ so that $S_p(q) \supseteq S_e(q)$ is satisfied, where $S_e(q)$ is expressed as a set of MHs that receive a bid invitation for data request q in the existing method (Figure 3 (a)). $S_e(q)$ is a set of MHs that exist in circle C shown in Figure 2. So that $S_p(q) \supseteq S_e(q)$ is satisfied, it is necessary to (i) select the receivers that receive a bid invitation for data request q appropriately and (ii) mitigate the effects of malicious REPs that drop packets intentionally. (i) can be decomposed into two sub-challenges, (i-a) appropriate selection of REPs and (i-b) selection of REPs that the server sends a bid invitation for data request q to. In the following, these three design challenges are explained in detail.

(1) Appropriate selection of REPs (Design challenge (i-a))

REPs need to be selected so that $\bigcup_{i \in S_R} G_i = S_T$ is satisfied, 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 MHs. If $\bigcup_{i \in S_R} G_i \neq S_T$, the server cannot send a bid invitation to some MHs and there is a possibility that $S_p(q) \supseteq S_e(q)$ is not satisfied. In addition, it is necessary to select a REP among an MH and its one-hop neighbors within time T_q so that the server can send a bid invitation within T_q as in the existing method.

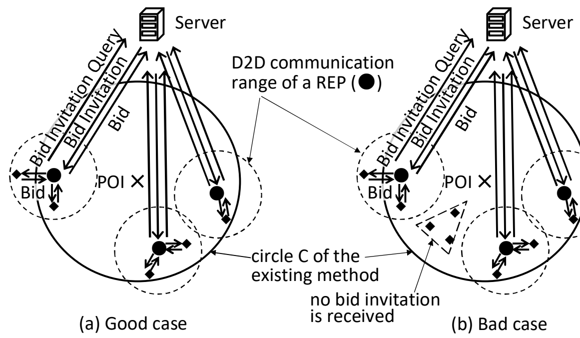

Figure 3 RevAuc procedure using D2D communication

(2) Selection of REPs that the server sends a bid invitation for a data request to (Design challenge (1-b))

When selecting a set of REPs R_q that the server sends a bid invitation for a data request q to, the server needs to try to ensure that $S_p(q) = \bigcup_{i \in R_q} G_i \supseteq S_e(q)$ is satisfied while minimizing $|R_q|$ to reduce cellular traffic between the server and MHs. For that purpose, the server needs to select R_q based on the locations of all MHs (i.e. REPs and their one-hop neighbors) as in the case of the existing method. 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 location. However, if a REP sends the location of each one-hop neighbor, cellular traffic increases as the number of one-hop neighbors increases. To reduce cellular traffic, it will be useful to send the locations of a REP and its one-hop neighbors by a fixed sized message regardless of the number of one-hop neighbors. For that purpose, the locations of a REP and its one-hop neighbors need to be approximated in some way. Additionally, the method with which the server selects R_q needs to be able to accommodate approximated locations of the REP and its one-hop neighbors.

(3) Mitigating the effects of malicious REPs that drop packets intentionally (Design challenge (ii))

In the presence of malicious actions by REPs seeking to gain an advantage in RevAuc, RevAuc does not work correctly. Specifically, a malicious REP may not send a bid invitation to its on-hop neighbors, or the malicious REP could ignore bids from its one-hop neighbors and send its own bid to the server.

C. Outline of the proposed method

(1) Appropriate selection of REPs

Each MH stands as a REP if neither one-hop neighbors nor itself has been a REP within configuration parameter time T'_q that is shorter than T_q. Thus, a REP is selected among an MH and its one-hop neighbors at an interval shorter than or equal to T_q. To handle changes in one-hop neighbors due to the movement of MHs, a REP finishes its role of REP after sending a Bid message and a new REP is selected.

Later, we will introduce another type of REP to mitigate the effects of malicious REPs that drop packets intentionally and we

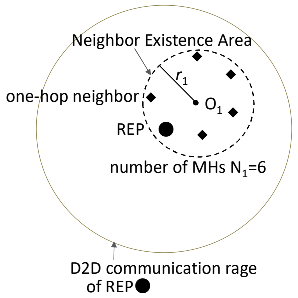

Figure 4 Example of Neighbor Existence Area

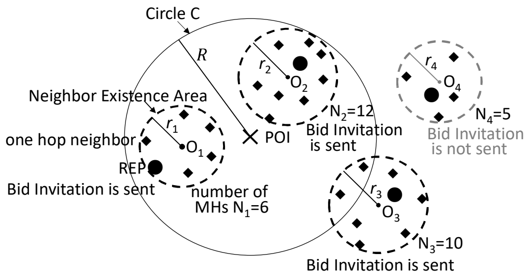

Figure 5 Example of selection of REPs that the server sends a bid invitation for a data request to (for $\theta_n = 17$)

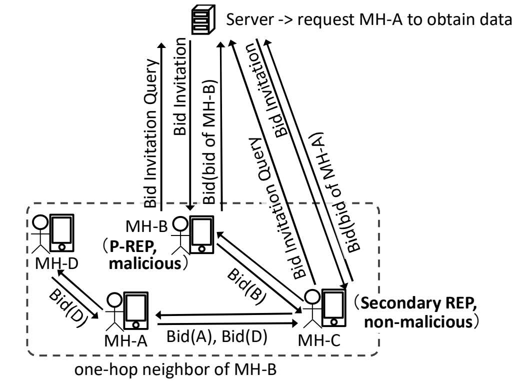

Figure 6 S-REP sending a bid invitation and receiving bids

need to distinguish them from the REPs described here. We denote a REP explained here as a primary representative MH (P-REP) hereafter and we term a REP that will be introduced later a secondary representative MH (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.

(2) Selection of REPs that the server sends a bid invitation for a data request to

The server estimates approximate locations of MHs and selects a subset of REPs that it sends a bid invitation for data request q to in a similar way to the case with the existing method using the "Neighbor Existence Area" (NEA). The server receives an NEA from each REP with a Bid Invitation Query message. An NEA approximates information on locations of the REP sending the NEA and its one-hop neighbors with constant data size. The NEA is the smallest circle (Center O, Radius r) that covers the area where the REP and its one-hop neighbors exist and the number of MHs N (Figure 4). The server calculates the minimum value of R, the radius of a circle C with POI of data request q as the center so that the expected number of MHs in circle C can be greater than or equal to configuration parameter θ_n (Figure 5). MHs in the NEA are considered to be in circle C only when circle C covers the NEA. The server judges that a P-REP or one-hop neighbors of the P-REP is 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 5). Thus $S_p(q) \supseteq S_e(q)$ will be satisfied.

(3) Mitigating the effects of malicious REPs that drop packets intentionally

We mitigate the effects of malicious REPs by making multiple MHs among one-hop neighbors of an MH and the MH work as REPs and configuring the server to send the same bid invitation to those REPs. For this purpose, one or more secondary representative MHs (S-REP) stand as a REP triggered by the end of a selection of a P-REP. When the selection of a P-REP ends, the P-REP broadcasts a notification message and the one-hop neighbors of the P-REP know the end of the selection. S-REPs are selected from one-hop neighbors of the P-REP. The server sends the same bid invitation to the S-REPs as it sent to the P-REP corresponding to the S-REPs. An S-REP sends a bid invitation to the one-hop neighbors of the P-REP and receives bids from them through D2D communication (Figure 6). After sending a bid message to the server, an S-REP finishes its role of S-REP. If at least one of P-REP and S-REPs is non-malicious, each MH can receive a bid invitation and send its bid to the server. Thus, effects of malicious REPs can be mitigated.

D. Procedure of the proposed method

In the following, we explain the key details of RevAuc procedure executed by a REP (Figure 7). We assume that an evaluation function is defined to calculate an evaluation score of a bid and both the server and MHs know this.

(1) Some MHs stand as a P-REP or an S-REP. REP sends a "Bid Invitation Query message" that includes information on whether the sender is a P-REP or an S-REP, query ID and the NEA.

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

(3) When receiving a Bid Invitation message, a REP sends a "D2D Bid Invitation message" including the bid invitations through two-hop D2D broadcast. Two-hop D2D broadcast is used to allow one-hop neighbors of the P-REP to receive the D2D Bid Invitation message from an 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.

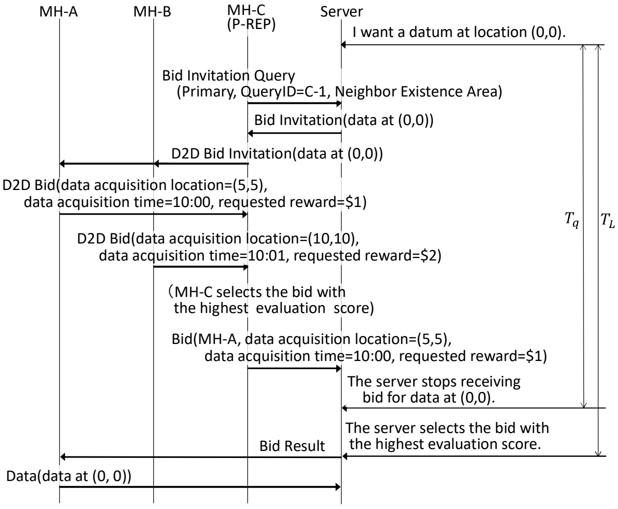

Figure 7 Example of RevAuc procedure executed by a P-REP

(4) When receiving a D2D Bid Invitation message, an MH determines whether to tender a bid for a bid invitation or not. If the MH tenders a bid, it sends a "D2D Bid message" including the bid to the REP through two-hop D2D broadcast.

(5) A REP waits for a configuration parameter time T_w after sending a D2D Bid Invitation message. Then, the REP selects a bid with the highest evaluation score among its own bid and bids received in D2D Bid messages while waiting. The REP sends the selected bid to the server by a Bid message.

(6) The server selects the successful bidder, which means the MH that sent a bid with the highest evaluation score among bids received within time T_q after the data request was generated. It sends a "Bid Result message" to the successful bidder to request the MH to obtain data.

(7) When receiving a Bid Result message, an MH obtains data and sends the data by a "Data message".

V. PERFORMANCE EVALUATION

A. Performance metrics

We confirmed the reduction in cellular traffic by comparing the total number of Bid Invitation Query messages, Bid Invitation messages, and Bid messages in the existing method with the number of those messages in the proposed method. We compared the amount of cellular traffic in terms of the number of messages because messages sizes are almost identical across both methods and each message can be sent in one 128 bytes radio interface packet that is generally a unit for measuring cellular traffic. We also evaluated differences in evaluation scores of successful bidders' bids between the proposed method and the existing method. We conducted ten simulations with different random number seeds for 1000 seconds with 250, 500, and 750 MHs.

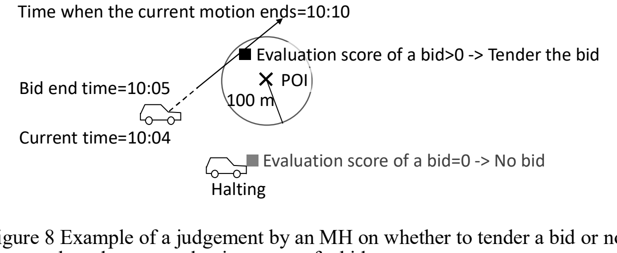

Figure 8 Example of a judgement by an MH on whether to tender a bid or not based on an evaluation score of a bid

We confirmed the mitigation of the effects of malicious REPs that drop packets intentionally by evaluating the differences in evaluation score of successful bidders' bids between three cases, that is, no malicious MH, 5% malicious MHs, and 30% malicious MHs, with one S-REP and with no S-REP for each P-REP. Simulations were conducted ten times with different random number seeds for 1000 seconds with 250 and 500 MHs.

B. Evaluation Environment

D2D communication was modeled using a unit disc model with 100 m propagation range without packet loss. MHs moved in a 1000 m by 1000 m square area by random waypoint (RWP)[9]. The period for which an MH stops was chosen to be between 0 second and 5.0 seconds by a uniform random number. Taking an automotive example, MH velocity was chosen to be between 6.0 m/s and 15.0 m/s by a uniform random number. The destination of an MH was selected by a uniform random number. The density of MHs in a cell varies from cell to cell even in this scenario [9]. A data request was generated every 5 seconds and POI was selected by a uniform random number.

A scenario related to an MH's behavior in RevAuc is as follows. The amount of reward requested by each MH was set at a fixed value determined for each MH by a uniform random number at the start of a simulation. The maximum amount of reward paid to an MH by a server was larger than the reward requested by any MH. If a current motion state (stop or move) ends before the time when a successful bidder is determined (bid end time), an MH does not tender a bid. An MH creates a bid under the condition that it obtains a requested datum at a location nearest to POI in its current trajectory. If the evaluation score of the created bid is greater than 0, the MH tenders the bid (Figure 8).

The evaluation function E was defined as the average of (1) an evaluation function E_Q that evaluates the distance between the POI and the location where an MH will obtain data, (2) an evaluation function E_T that evaluates the time between a bid end time and the time when the MH will obtain data (data acquisition time), and (3) an evaluation function E_C that evaluates a reward requested by the MH.

The parameter values used here were $T_q = 12.45$ s, $T_L = 12.95$ s, $\theta_n = 20$, $R_{\max} = 150$ m, $T'_q = 5$ s, $T_s = 0.5$ s, $M = 2$, $T_b = 1$ s, and $T_w = 1.6$ s.

C. Reduction in the cellular traffic compared to the existing method

We explain how the reduction of cellular traffic compared to the existing method varies and how the difference in the evaluation score of a successful bidder's bid between the proposed method and the existing method varies according to the number of MHs in the simulation results. We show the ratio of cellular traffic in the proposed method to cellular traffic in the existing method and the 95% confidence interval (almost

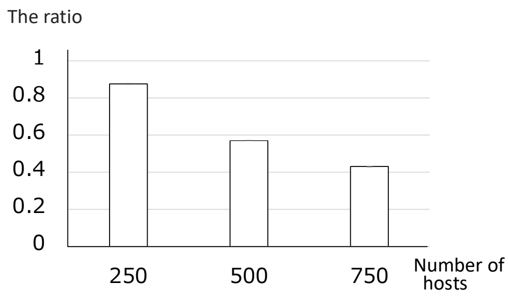

Figure 9 Ratio of cellular traffic in the proposed method to cellular traffic in the existing method

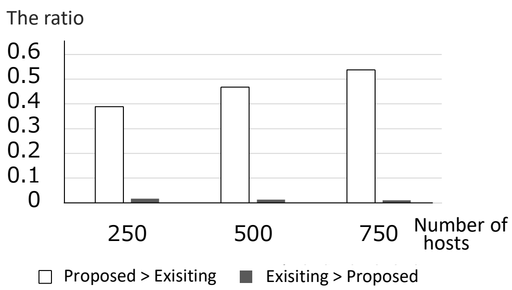

Figure 10 Ratio of the number of different results between the existing method and the proposed method

invisible) in Figure 9. Hereafter, we refer to the number of data requests for which an evaluation score of a successful bidder's bid is different between two simulation scenarios by "the number of different results". In Figure 10, we show the ratio of the number of different results between the existing method and the proposed method to 1970 data requests. In Figure 10 white bars show the ratio of data requests with the evaluation score of the proposed method being higher and gray bars show the ratio of data requests with the evaluation score of the existing method being higher.

As shown in Figure 9, the ratios of cellular traffic in the proposed method to cellular traffic in the existing method are 0.875 with 250 MHs, 0.570 with 500 MHs, and 0.430 with 750 MHs. These results show that as the number of MHs increases, the relative amount of cellular traffic decreases. This trend is reasonable because the number of one-hop neighbors of an MH is expected to increase as the number of MHs increases. From Figure 9, we confirmed that the proposed method can reduce cellular traffic compared to the existing method.

As shown in Figure 10, the ratios of the number of different results between the existing method and the proposed method are 0.406 with 250 MHs, 0.481 with 500 MHs, and 0.547 with 750 MHs. These results show that as the number of MHs increases, so does the ratio. In addition, the ratio of data requests with the evaluation score of the proposed method being higher is much larger than the ratio of the reverse case. The former ratio increases as the number of MHs increases. On the other hand, the ratio of data requests with the evaluation score of the existing method being higher decreases as the number of MHs increases.

D. *Mitigating the effects of malicious REPs that drop packets intentionally*

For the case of 5% malicious MHs and 250 MHs, the number of different results is 16 for no S-REP case and 6 for one S-REP case. For 5% malicious MHs and 500 MHs, the number of different results is 6 for no S-REP case and 2 for one S-REP case. For 30% malicious MHs and 250 MHs, the number of different results is 90 for no S-REP case and 43 for one S-REP case. For 30% malicious MHs and 500 MHs, the number of different results is 46 for no S-REP case and 21 for one S-REP case. In all cases, the number of different results is smaller for one S-REP case compared to no S-REP case. In addition, the evaluation scores of successful bids in the case with malicious MHs were lower than the evaluation scores of successful bids in the case without malicious MHs when evaluation scores were different. Thus, these results confirm that S-REPs can mitigate the effects of malicious MHs that drop packets intentionally.

VI. CONCLUSION

In this paper, we proposed a reverse auction-based mobile crowdsensing method that can reduce the amount of cellular traffic between a server and MHs. The proposed method reduces cellular traffic by making representative MHs transmit bid invitations received from the server to their one-hop neighbors and relay bids from their one-hop neighbors to the server through D2D communication. The representative MHs are selected through D2D communication in a distributed manner. Generally, with the proposed method, a successful bidder obtains data at the same location for the same amount of reward within the same amount of time as in the existing method. In addition, the proposed method can mitigate the effects of malicious MHs that drop packets intentionally. We confirmed the effectiveness of the proposed method through simulation.

REFERENCES

[1] J. Wang, et al., "Task Allocation in Mobile Crowd Sensing: State of the Art and Future Opportunities", IEEE Internet of Things journal, Vol. 5, Issue 5, Oct. 2018.

[2] X. Zhang, et al., "Incentives for Mobile Crowd Sensing: A Survey", IEEE Communications Surveys & Tutorials, Vol. 18, Issue 1, first quarter 2016, pp.54-67.

[3] J. Lee et al., "Dynamic pricing incentive for participatory sensing", Pervasive and Mobile Computing, No.6, 2010, pp.693-708.

[4] X. Zhang, et al., "Free Market of Crowdsourcing: Incentive Mechanism Design for Mobile Sensing", IEEE Trans. on Parallel and Distributed Systems, Vol.25, No.12, Dec. 2014, pp.3190-3200.

[5] Y. Liu, et al., "Reverse Auction Based Incentive Mechanism for Location-Aware Sensing in Mobile Crowd Sensing", IEEE International Conference on Communications, 2018.

[6] V. Mota, et al., "Towards scalable mobile crowdsesing through device-to-device communication", Journal of Network and Computer Applications, No.122, 2018, pp.99-106.

[7] Matsumoto, et al., "Low Server-Load Onboard Camera Image Delivery Scheme Based on a Cellular Network and Cooperation of Neighboring Vehicles through Vehicle-to-vehicle Communication", IPSJ Journal, Vol.56, No.11, Nov. 2016, pp.2106-2116.

[8] L. Pu, et al., "Crowdlet: Optimal Worker Recruitment for Self-Organized Mobile Crowdsourcing", IEEE INFOCOM, 2016.

[9] J. Broch, et al., "A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols", Proc. ACM/IEEE Int. Sym. on Mobile Computing and Networking (MobiCom'98) , 1998, pp.85-97