International Journal of Advanced Trends in Engineering and Technology (IJATET)
Impact Factor: 5.665, ISSN (Online): 2456 - 4664
(www.dvpublication.com) Volume 2, Issue 2, 2017

NEAR-OPTIMAL TASK ALLOCATION FOR PIGGYBACK CROWD SENSING UNDER AODV ROUTING PROTOCOL

S. Nithya* & Dr. C. Deepa**

* Department of Computer Science, Sri Ramakrishna College of Arts and Science (Formerly SNR Sons College), Coimbatore, Tamilnadu

** Associate Professor, Department of Information Technology, Sri Ramakrishna College of Arts and Science (Formerly SNR Sons College), Coimbatore, Tamilnadu

Cite This Article: S. Nithya & Dr. C. Deepa, "Near-Optimal Task Allocation for Piggyback Crowd Sensing under AODV Routing Protocol", International Journal of Advanced Trends in Engineering and Technology, Volume 2, Issue 2, Page Number 159-163, 2017.

Abstract:

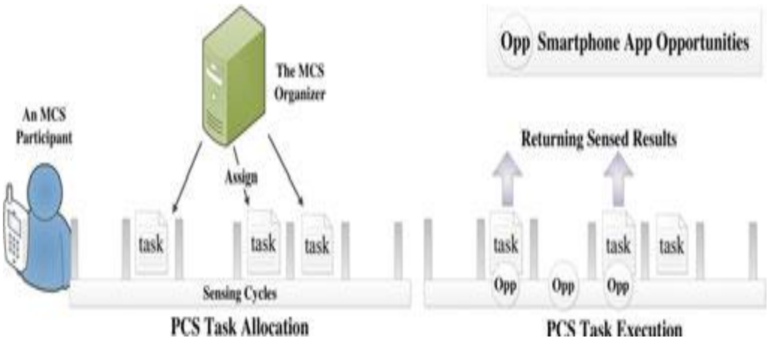
Mobile crowd sensing is a technique where a large group of individuals having mobile devices capable of sensing and computing .Various solutions have been proposed to reduce energy consumption of individual mobile device, ranging from adapting sensing frequency to inferring part of the data rather than sensing and uploading all data. Mobile crowd sensing had a novel spatial-temporal coverage metric, k-depth coverage, problems. This metric considers both the fraction of subareas covered by sensor readings and the number of sensor readings collected in each covered subarea. Then iCrowd, a generic MCS task allocation framework operating with the energy-efficient Piggyback Crowd sensing task model, is proposed to optimize the MCS task allocation with different incentives and k-depth coverage objectives/ constraints. iCrowd first predicts the call and mobility of mobile users based on their historical records, then it selects a set of users in each sensing cycle for sensing task participation, so that the resulting solution achieves two dual optimal MCS data collection like near-maximal k-depth coverage without exceeding a given incentive budget and near-minimal incentive payments. The Ad-hoc On-demand Distance Vector (AODV) routing protocol is a routing protocol used for dynamic wireless networks where nodes can enter and leave the network at will. To find a route to a particular destination node, the source node broadcasts a RREQ to its immediate neighbors. If one of these neighbors has a route to the destination, then it replies back with a RREP. Otherwise the neighbors in turn rebroadcast the request. This continues until the RREQ hits the final destination or a node with a route to the destination. At that point a chain of RREP messages is sent back and the original source node finally has a route to the destination.

Key Words: Mobile Crowd Sensing (MCS), MCS Task Allocation & Incentives.

1. Introduction:

Mobile Crowd sensing (MCS) has become an efficient way to sense and collect environment data of urban area in real-time (e.g., air quality, temperature or noise level). Instead of deploying static and expensive sensor network in urban area, MCS leverages the sensors embedded in mobile phones and the mobility of mobile users to sense their surroundings, and utilizes the existing communication infrastructure (e.g., 4G, Wi-Fi etc.) to collect data from mobile phones scattered in the urban area. By collecting sensor readings from mobile users, a "big picture" of the environment in the target area can be obtained using MCS without significant cost.

iCrowd - a near-optimal task allocation framework for mobile crowdsensing, which can improve the efficiency of environment data collection with less cost. Here we first discuss the motivations and background of our MCS research, then we formulate a new MCS research problem with a unified set of research assumptions and objectives. We elaborate the technical challenges of the proposed research and finally we summarize our technical contributions.



In MCS, there are two main players: MCS organizer who is the person or organization coordinating the sensing task, and MCS participants who are the mobile users involved in the sensing task. An MCS task usually requires the organizer to recruit participants, to allocate sensing tasks to selected participants, and to collect sensor readings from these participants' mobile devices that well represent the characteristics of the target sensing region [4], often with budget constraints on participant incentives.

Specifically, the MCS organizer needs to specify the target sensing area, which often consists of a set of subareas, and further specify the sensing duration (e.g., 10 days), which is usually divided into equal-length sensing cycles (e.g., each cycle lasts for an hour). Once the settings of subareas and sensing cycles are determined, the MCS application usually needs to collect a number of sensor readings from each subarea of the target region in each sensing cycle. Taking a one-week urban air quality monitoring MCS task as an example, the MCS organizer first divides the whole area into 1 km2 grid cells, then splits the oneweek MCS study time into a sequence of one-hour sensing cycles, and further requests at least one MCS participant in each grid to upload the air quality sensor reading in each sensing cycle. In this case, however, the cost of the whole MCS task, including the energy consumption caused by the MCS application on each participant's mobile device and the overall incentives cost to recruit participants, could be quite high. In order to lower the cost of MCS, the mechanism to reduce the energy consumption and control the overall incentive cost, while ensuring the spatial-temporal coverage of collected environment sensor readings, is thus needed. Next we introduce the background of our research from following thee aspects:

Energy-efficient piggyback crowdsensing (PCS). So far, various solutions have been proposed to reduce energy consumption of individual mobile device, ranging from adapting sensing frequency to inferring part of the data rather than sensing and uploading all data. One of the effective solutions is Piggyback Crowdsensing, which reduces energy consumption by leveraging smartphone opportunities to perform sensing tasks and return sensor readings. For example, uploading sensing data in parallel with a 3G call can reduce about 75 percent of energy consumption in data transfer compared to the 3G-based solution.

Spatial-temporal coverage of MCS tasks. The typical approach for measuring the spatial-temporal coverage is to use the fraction of subareas being covered by at least one sensor reading in each sensing cycle. AnMCS application may need to collect sensor readings to achieve either full spatial-temporal coverage or partial spatial-temporal coverage. Usually, the use of full spatial-temporal coverage is to ensure the collected sensor readings representing each subarea in each sensing cycle, while the use of partial coverage aims to collect data that could represent a certain fraction (e.g., 80 percent) of subareas in each cycle.

Incentives, budget and task allocation. In order to recruit participants for MCS, each selected participant is typically offered a certain amount of money as incentives and thus the MCS organizer needs to prepare a budget equal to the total incentives paid to all participants in each MCS task. Once the spatial-temporal coverage and total budget are determined, the MCS organizer needs to select participants with the goal minimizing the total budget while ensuring the spatial temporal coverage, or maximizing the spatial-temporal coverage with a fixed budget.

In order to achieve either of above goals, given users who are willing to participate the MCS task, an MCS organizer needs to allocate sensing tasks to users, where the organizer first selects participants for MCS, and then decides in which sensing cycles each participant should perform the MCS task. Only the participants selected for MCS will be paid with incentives.

With all aforementioned coverage, energy, and incentives issues, we are motivated to study the problem of optimizing MCS tasks, subject to various spatial-temporal coverage and incentive cost objectives/constraints.

2. Related Works:

Crowd Recruiter operates on top of energy-efficient Piggyback Crowdsensing (PCS) task model and minimizes incentive payments by selecting a small number of participants while still satisfying probabilistic coverage constraint. In order to achieve the objective when piggybacking crowdsensing tasks with phone calls, CrowdRecruiter first predicts the call and coverage probability of each mobile user based on historical records. It then efficiently computes the joint coverage probability of multiple users as a combined set and selects the near-minimal set of participants, which meets coverage ratio requirement in each sensing cycle of the PCS task. We evaluated CrowdRecruiter extensively using a large-scale realworld dataset and the results show that the proposed solution significantly outperforms three baseline algorithms by selecting 10.0% - 73.5% fewer participants on average under the same probabilistic coverage constraint.

Nowadays, there is an increasing demand to provide real-time environmental information such as air quality, noise level, traffic condition, etc. to citizens in urban areas for various purposes. The proliferation of sensor-equipped smartphones and the mobility of people are making Mobile Crowdsensing (MCS) an effective way to sense and collect information at a low deployment cost. In MCS, instead of deploying static sensors in urban areas, people with mobile devices play the role of mobile sensors to sense the information of their surroundings and the cellular network is used to transfer data for MCS applications. Since the sensing coverage in MCS relies on the uncontrollable mobility of people, it is thus imperative to take people's mobility pattern into account in order to ensure that the collected sensing data well represent the characteristics of the target sensing region. For many MCS applications, such as environment monitoring, full coverage is not always required. It is often sufficient to ensure a high ratio of spatial coverage in a specified time frame and get an idea of the situations in most places that people frequently visit.

Selecting Participants for the Whole MCS Task:

Studies in this line of research usually assume each participant is paid with a fixed amount of incentives; then a group of participants are selected to perform the MCS task in all sensing cycles. Reddy et al.

first study the research issue of participant selection in participatory sensing, and then propose a coverage-based participant search framework to select a predefined number of participants to maximize the spatial coverage. Singla and Krause propose a novel adaptive participant selection mechanism for maximizing spatial coverage under total incentive constraint in community sensing with respect to privacy. Cardone et al. develop a Mobile Crowdsensing platform, where a simple participant selection mechanism is proposed to maximize the spatial coverage of crowdsensing with a predefined number of participants.

Selecting participants for each sensing cycle—Studies in this line of research usually assume each participant is paidwith a varied amount of incentives with respect to the number of sensing cycles when the participant performed the MCS task; then for each sensing cycle a subset of participants are selected for the MCS task. In, the authors introduce the notion of virtual sensors which intend to collaboratively infer sensing values of each subarea that is not covered by any participant in each sensing cycle, and they propose spatial and temporal coverage quality metrics and leverage the virtual sensor approach in order to reduce the number of participants required in each sensing cycle, while still meeting the coverage quality constraint. Most recently, Hachem et al. proposes a cycle assignment framework for participatory sensing, where the framework predicts mobile users' future locations in next timeslot (or sensing cycle) based on their current location and recent trajectory. From the prediction they select a minimal number of mobile users expecting to cover a certain percentage of the target area in the next time slot.

Previous Protocols:

Selecting participants for each sensing cycle - Studies in this line of research usually assume each participant is paidwith a varied amount of incentives with respect to the number of sensing cycles when the participant performed the MCS task; then for each sensing cycle a subset of participants are selected for the MCS task. In, the authors introduce the notion of virtual sensors which intend to collaboratively infer sensing values of each subarea that is not covered by any participant in each sensing cycle, and they propose spatial and temporal coverage quality metrics and leverage the virtual sensor approach in order to reduce the number of participants required in each sensing cycle, while still meeting the coverage quality constraint. Most recently, Hachem et al. proposes a cycle assignment framework for participatory sensing, where the framework predicts mobile users' future locations in next timeslot (or sensing cycle) based on their current location and recent trajectory. From the prediction they select a minimal number of mobile users expecting to cover a certain percentage of the target area in the next time slot.

Motivation and Problem Support:

Given a set of volunteer mobile users, the target region divided by a set of subareas (e.g, cell towers in our study), and the MCS process consisting of a sequence of equallength sensing cycles (e.g., one cycle per hour), the task allocation problem of iCrowd is to select a number of participants from the volunteer mobile users and to determine in which sensing cycles each selected participant is assigned the PCS task, subject to various optimal MCS data collection goals. With respect to the research objectives introduced in Section 1, we primarily study task allocation problems of the following two goals:

Goal 1: Maximizing k-depth coverage under Budget Constraint.

Goal 2: Minimizing Overall Incentive Payment under k-depth coverage Constraint.

3. Proposed Work:

iCrowd - a near-optimal task allocation framework for mobile crowdsensing, which can improve the efficiency of environment data collection with less cost. Here we first discuss the motivations and background of our MCS research, then we formulate a new MCS research problem with a unified set of research assumptions and objectives. We elaborate the technical challenges of the proposed research and finally we summarize our technical contributions. We propose to study a novel MCS task allocation problem for Piggyback Crowdsensing applications, where we first assume that each MCS participant senses and uploads sensor readings leveraging smartphone opportunities (e.g., placing a 3G call) to reduce the MCS energy consumption.

We can increase the data delivery ratio and reduce the effects of packet loss caused by the node mobility. Specifically, the framework considers the regularity in mobility patterns during the construction of the routing tree and deployment of nodes. It also includes an overhearing mechanism for mobile nodes to further improve the data delivery ratio.

Objective:

k -depth coverage of MCS tasks. While the existing spatialtemporal coverage metrics usually assume that the environment data (e.g., air quality) of a subarea in a sensing cycle could be represented by a single sensor reading, it is reasonable to believe that the each subarea could be better characterized if we could deduce the environment characteristics using multiple sensor readings collected from the same subarea. However, if we increase the number of sensor readings in a subarea above a certain threshold, the accuracy of the deduced value may not increase anymore. Thus we propose a novel spatial-temporal coverage metrics–i.e., k-depth coverage, which could be used as either an objective

Utility-Based User-Cycle Combination Selection Algorithm:

Xn–the set of user-cycle combinations already selected in the nth outer loop, U-the overall set of users, I-total number of sensing cycles, and bo-the incentive payment for bonus.

Output: X0-a new set of user-cycle combinations selected in the current inner loop

1. begin /* initialize */
2. X0 ;; /* getting all users in Xn */
3. Sn getAllUsers(Xn); /* getting all possible user cycle-combinations */
4. C fhu; iij8u 2 U; 0 _ i < Ig;
5. if bo ¼ 0 then /* when bo ¼ 0 (i.e., Fixed Individual Incentive Setting), select a new user (with all cycles) having the maximal utility */
6. u0 argmaxu2UnSn P 0_j<I Utility(hu; jijXn);
7. X0 fhu0; jij0 _ j < Ig;
8. else /* when bo > 0 (i.e., Varying Individual Incentive Setting), select a new usercycle combination having the maximal utility */
9. hu0; i0i argmaxu2CnXn Utility(hu; iijXn);X0 fhu0; i0ig;
10. return X0;

In this way, the inner-loop greedy process continues selecting/ adding a new subset of user-cycle combinations and deciding whether new user-cycle combinations should be added using the constraint-based stopping criterion, until the corresponding constraint-based stopping criterion algorithm decides to stop selecting new combinations.

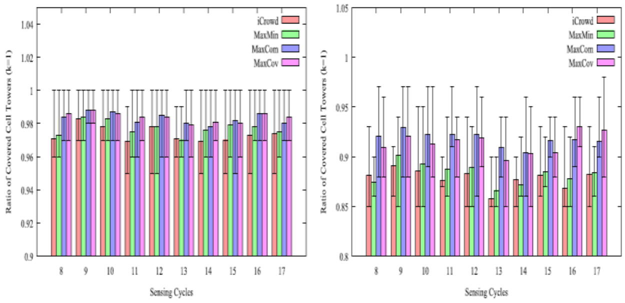
Convergence-based outer-loop stopping criterion –Given the selected set of user-cycle combinations (e.g., Xn in the nth outer-loop iteration), the algorithm decides whether to return the task allocation results or continue for further computation. When bo ¼ 0 – i.e., the incentive to each participant is fixed, the algorithm stops at the first outer-loop iteration and returns X1 directly as the task allocation result. When bo > 0 – i.e., the individual incentive is dependent on the number of participating cycles, the algorithm needs to decide if to return the task allocation result or continue to obtain Xnþ1, with respect to the two MCS data collection goals.

AODV Routing Protocol:

The Ad-hoc On-demand Distance Vector (AODV) routing protocol is a routing protocol used for dynamic wireless networks where nodes can enter and leave the network at will. To find a route to a particular destination node, the source node broadcasts a RREQ to its immediate neighbors. If one of these neighbors has a route to the destination, then it replies back with a RREP. Otherwise the neighbors in turn rebroadcast the request. This continues until the RREQ hits the final destination or a node with a route to the destination. At that point a chain of RREP messages is sent back and the original source node finally has a route to the destination.

We proved that AODV protocol never produces routing loops by proving that a combination of sequence numbers and hop counts is monotonic along a route. This means that there can't be any loop in the routing table. The proof was done completely automatically and our algorithm was able to generate all the predicates needed. The Ad hoc On Demand Distance Vector (AODV) routing algorithm is a routing protocol designed for ad hoc mobile networks. AODV is capable of both unicast and multicast routing. It is an on demand algorithm, meaning that it builds routes between nodes only as desired by source nodes. It maintains these routes as long as they are needed by the sources. Additionally, AODV forms trees which connect multicast group members. The trees are composed of the group members and the nodes needed to connect the members. AODV uses sequence numbers to ensure the freshness of routes. It is loop-free, self-starting, and scales to large numbers of mobile nodes.

AODV builds routes using a route request / route reply query cycle. When a source node desires a route to a destination for which it does not already have a route, it broadcasts a route request (RREQ) packet across the network. Nodes receiving this packet update their information for the source node and set up backwards pointers to the source node in the route tables. In addition to the source node's IP address, current sequence number, and broadcast ID, the RREQ also contains the most recent sequence number for the destination of which the source node is aware. A node receiving the RREQ may send a route reply (RREP) if it is either the destination or if it has a route to the destination with corresponding sequence number greater than or equal to that contained in the RREQ. If this is the case, it unicasts a RREP back to the source. Otherwise, it rebroadcasts the RREQ. Nodes keep track of the RREQ's source IP address and broadcast ID. If they receive a RREQ which they have already processed, they discard the RREQ and do not forward it. As the RREP propagates back to the source, nodes set up forward pointers to the destination. Once the source node receives the RREP, it may begin to forward data packets to the destination. If the source later receives a RREP containing a greater sequence number or contains the same sequence number with a smaller hopcount, it may update its routing information for that destination and begin using the better route. As long as the route remains active, it will continue to be maintained. A route is considered active as long as there are data packets periodically travelling from the source to the destination along that path. Once the source stops sending data packets, the links will time out and eventually be deleted from the intermediate node routing tables. If a link break occurs while the route is active, the node upstream of the break propagates a route error (RERR) message to the source node to inform it of the now unreachable destination(s). After receiving the RERR, if the source node still desires the route, it can reinitiate route discovery.



(a) MERGED 95% (b) MERGED 85%

Max/Min/Average ratio of covered cell towers based on the MERGED region (Please see the result of residential and business regions in Appendix, available in the online supplemental material)

Conclusion:

In this paper, we proposed a unified task allocation framework, iCrowd, for Piggyback Crowdsensing. iCrowd is designed to optimally allocate sensing tasks to PCS participants, subject to different incentive and spatial-temporal coverage constraints/objectives. Specifically, iCrowd could be adopted to either maximize the overall k-depth coverage across all sensing cycles with a fixed budget or to minimize the overall incentive payment while ensuring a predefined k-depth coverage constraint, by selecting a number of participants and determining in which sensing cycles each selected participant is needed for the PCS task participation. The PCS was adopted to reduce energy consumption of individual mobile device, by exploiting call opportunities to perform sensing tasks and upload sensed data. In order to allocate PCS task for either optimal MCS data collection goals, iCrowd first predicts the coverage probability of each mobile user, then performs a near-optimal participant/cycle task allocation search algorithm with low computational complexity. Theoretical analysis proves that iCrowd can achieve near-optimality for both optimal MCS data collection goals, and evaluations with a large-scale real-world dataset show that iCrowd outperformed six baseline approaches. For Goal, 1 it achieved 3-60 percent higher k-depth coverage compared to baseline approaches under the same budget constraint, while for Goal. 2 iCrowd required 10.0-90.5 percent less overall incentive compared to baselines under the same k-depth coverage constraint. We are analyzing to include the encryption and decryption process in this project. This will avoid the node failures and collision occurred within the network. This encryption process secures the data when it is transmitting through the network. Also the future system will be doing the efficient algorithm for the advanced encryption process.

References:

1. D. Zhang, H. Xiong, L. Wang, and G. Chen, “Crowd recruiter: Selecting participants for piggyback crowd sensing under probabilistic coverage constraint,” in Proc. ACM Int. Joint Conf. Pervasive Ubiquitous Comput., 2014, pp. 703–714.
2. H. Xiong, D. Zhang, G. Chen, L. Wang, and V. Gauthier, “Crowd asker: Maximizing coverage quality in piggyback crowd sensing under budget constraint,” in Proc. IEEE Int. Conf. Pervasive Comput. Commun., 2015, pp. 55–62
3. R.K. Ganti, F. Ye, and H. Lei, “Mobile crowd sensing: Current state and future challenges,” IEEE Commun. Mag., vol. 49, no. 11, pp. 32–39, Nov. 2011.
4. M. Musolesi, M. Piraccini, K. Fodor, A. Corradi, and A.T Campbell, “Supporting energy-efficient uploading strategies for continuous sensing applications on mobile phones,” Pervasive Comput., 2010, pp. 355–372.
5. Y. Zheng, F. Liu, and H.-P. Hsieh, “U-air: When urban air quality inference meets big data,” in Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2013, pp. 1436–1444.
6. S. Reddy, D. Estrin, and M. Srivastava,, “Recruitment framework for participatory sensing data collections,” in Proc. 8th Int. Conf. Pervasive, 2010, pp. pp. 138–155
7. G. Cardone, L. Foschini, P. Bellavista, A. Corradi, C. Borcea, M. Talasila, and R. Curtmola, “Fostering participation in smart cities: A geo-social crowd sensing platform,” IEEE Commun. Mag., vol. 51, no. 6, pp. 112–119, Jun. 2013
8. X. sheng, J. Tang, and W. Zhang, “Energy-efficient collaborative sensing with mobile phones,” in Proc. IEEE Conf. Comput. Commun., 2012, pp. 1916–1924.
9. D. Philipp, J. Stachowiak, P. Alt, F. D€urr, and K. Rothermel, “Drops: Model-driven optimization for public sensing systems,” in Proc. IEEE Int. Conf. Pervasive Comput. Commun., 2013, vol. 18, p. 22.
10. A. Singla and A. Krause, “Incentives for privacy tradeoff in community sensing,” in Proc. 1st AAAI Conf. Human Comput. Crowd sourcing, 2013, pp. 165–173.