2022 16th International Conference on Ubiquitous Information Management and Communication (IMCOM) | 978-1-6654-2678-7/22/$31.00 ©2022 IEEE | DOI: 10.1109/IMCOM53663.2022.9721737

Collection Tree for Wireless Coverage Problem in Mobile Crowdsensing

1st Dejun Kong
Shanghai Jiao Tong University
Shanghai, China
kdjkdjkdj99@sjtu.edu.cn

2nd Xiaofeng Gao
Shanghai Jiao Tong University
Shanghai, China
gao-xf@cs.sjtu.edu.cn

3rd Guihai Chen
Shanghai Jiao Tong University
Shanghai, China
gchen@cs.sjtu.edu.cn

***Abstract*—Mobile devices play an important role in crowdsensing with the help of storage infrastructure. In the event of emergencies, a wireless sensor network can be constructed for people to sense the environment. When devices' mobility is out of control, collection trees are considered to route all the data to the sink. We propose a new wireless coverage problem in mobile crowdsensing when the core network is corrupted, and our aim is constructing collection trees to minimize the average delay of information. We propose two kinds of algorithms to construct the collection trees. We first propose Greedy Collection Tree (GCT) and Dynamic Greedy Collection Tree (DGCT) with two different objective functions, transferring probability and entropy increment. Then we propose Improved Greedy Collection Tree (IGCT) based on the assumption of low population fluidity. The effectiveness of our algorithms is testified in aspects of transmission average delay, delay's standard deviation and entropy evolution.**

***Index Terms*—wireless coverage, mobile crowdsensing, greedy collection tree, sensor networks, communications**

I. Introduction

Mobile crowdsensing platforms are created for monitoring the traffic situation [1] and the surrounding environment [14] as many sensors are embedded in mobile devices. A wireless sensor network can be constructed based on the capacity of wireless communication.

In this paper, we suppose that data are transferred through wireless communication, which means communication between mobile devices and relay nodes. In the case of emergent situations, such as earthquakes or thunder and lightning destructing a base station, it would be hard to restore the function of base station in a short time and therefore cause a core network corruption. At this point, an opportunistic network would replace the role of cellular network since it can be established by making use of users' mobility.

Our paper considers the scenario that the core network is corrupted, but people want instant information of the whole area. There are still two kinds of components in the network, static infrastructures and mobile devices. The infrastructures have large storage spaces while the mobile devices could sense the environment, keep the sensing data and transfer them to the relay nodes. A mobile device would exchange data with an infrastructure when they are close since wireless sensor network provides the capacity of wireless communication in proximity.

In traditional self-organizing wireless sensor networks, communication depends on mobile terminals with a routing function. However, when users wear wireless sensors and randomly walk in the network, the mobility of users is out of our control in mobile crowdsensing, i.e., users don't gather information in a specific location. For example, smartphones contain many sensors to support their performance, including sensors for light levels to adjust screen brightness, pressure sensors to complement the GPS for an accurate (vertical) location estimation, and thermometers for the battery to avoid damage from overheating. Readings from such sensors can be used for opportunistic environmental sensing by collecting them through mobile applications (apps). and then when users carry the mobile phones, they can collect the information at any time by the sensors embedded in mobile phones. Although the trajectories of users are predictable if there exists some historical data, we should exploit the feature of the crowd and propose a method in a macroscopic way.

We consider the scenario that all the collected data should be transferred to a sink. For example, the sink should process all the sensor data and make some decisions after an earthquake. In this scenario, we propose the data transferring process of mobile device, store-carry-forward, which is illustrated in Fig. 1. The gray lines represent the collection tree. The black lines represent the trajectories of mobile devices. The blue dot represents that the mobile device stores the information. The yellow dot represents that the mobile device is carrying the information. The orange dot represents that the current node is not the parent of the previous node. The mobile device would keep its information. The green dot represents that the mobile device reaches the parent node of the previous node and it would forward the information to the static node.

Our contribution mainly includes:

- We propose a new wireless coverage problem in mobile crowdsensing. The objective is to minimize the average delay of receiving messages.
- We propose two kinds of algorithms to solve the problem. Greedy Collection Tree (GCT) is constructed by a greedy strategy based on geographical information. Improved Greedy Collection Tree (IGCT) is a sub-optimal solution of minimizing the maximum working time. Dynamic

978-1-6654-2678-7/22/$31.00 ©2022 IEEE

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on December 19,2022 at 06:34:13 UTC from IEEE Xplore. Restrictions apply.

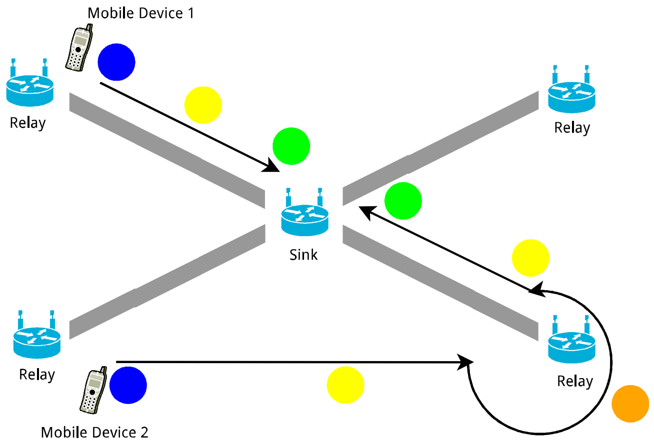


Fig. 1: Structure of Network and Data Transferring Process.

Greedy Collection Tree (DGCT) is generated based on the prediction for the crowd's circulation. The candidate nodes' entropies are calculated. The node with the maximum increase of entropy is chosen as the parent node.

The remainder of the paper is organized as follows. Section II introduces the related works of wireless coverage and mobile crowdsensing. Section III formulates the problem. Section IV analyzes the problem and proposes three algorithms. Section V evaluates the performance of our algorithms through simulation. Section VI concludes the paper.

II. Related Works

Data dissemination and collection in crowdsensing are based on several techniques. [3] improved mobile crowdsensing by collecting sensing data from other devices through opportunistic communication. [9] proposed a routing protocol for data collection which can determine a packet's value. [18] disseminates the data using existed infrastructure in the vehicular network. Piggyback crowdsensing [19], [20], [22] reuses the existed network link to decrease energy consumption. [8] improved the efficiency of mobile networks in cooperative transmission, where crowd members are treated as relays to extend the network coverage.

Collection Trees are applied as routing protocols in sensor networks. Collection Tree Protocol (CTP) [5] is a Distance Vector routing protocol designed for dynamic wireless sensor networks. [12] analyses and improves CTP [5] in mobile sensor networks. [10] proposed a time slot schedule algorithm that avoids the communication interference in dense tree-based sensor networks. [2] proposed collection trees for concurrent data streams in the scenario of IoT. [13] designed a data collection protocol for unmanned aerial vehicles (UAV) to reach difficult sensing areas.

Under the scenario of opportunistic network, several works are proposed. [17] proposed an adaptive sampling scheme in opportunistic sensor networks. Sparse area will be more sampled while oversampling will be avoided around the static nodes. Hybrid Crowd Sensing [4] deploys the static nodes in the way of triangular tessellation. Sensing tasks are distributed between the static nodes and mobile nodes opportunistically. Lower-bound of transferring probability is calculated by using the Markov chain.

Traditional crowdsourcing topics are also discussed in mobile crowdsensing, including incentive mechanisms, worker selection and task assignment. To attract workers to finish the tasks, incentive mechanisms are proposed. [16] offers augmented reality game to induce the action of users in crowdsensing. [7] adjusts the benefit for each user according to revenue and sensor amount to achieve the market equilibrium.

Worker recruitment and task assignment are considered to accomplish the crowdsensing task. [22] recruits the workers according to their utilities until satisfying the probabilistic target coverage. [6] considered the worker recruitment in vehicular crowdsensing to maximize the coverage rate while minimizing the number of vehicles. [11] selects the most popular nodes in social networks as the initial sensing nodes. Then the sensing nodes with the lowest utilities and the non-sensing nodes with the highest observability are exchanged. [15] constructs route for the tasks which must be done. Optional tasks with higher profits are added to the list if the time constraint is satisfied. [21] allocates the tasks while minimizing the total time finishing all the tasks by calculating the encounter probability of two nodes based on the historical data.

III. Problem Statement

For a region $\mathcal{R}$, suppose that there exists a set of static relay nodes $\mathcal{S} = \{s_0, s_1, \dots, s_k\}$ with their location $L_i = (x_i, y_i)$, where s_0 is the sink as the destination of all the messages and $i = 0, 1, \dots, k$. The former Region $\mathcal{R}$ is partitioned in the way of

$$\mathcal{R}(s_j) = \{P = (x, y) | \arg\min_i \|P - L_i\|_2 = j\}, \quad (1)$$

where P is the location of each node in region $\mathcal{R}$. Nodes within one hop count of s_i are noted as

$$\mathcal{N}(s_i) = \{s_j | \mathcal{R}(s_i) \cap \mathcal{R}(s_j) \neq \emptyset\}. \quad (2)$$

Obviously, $s_i \in \mathcal{N}(s_i)$ and $\mathcal{N}(s_i) \neq \emptyset$. Meanwhile, users (or mobile devices) wear wireless sensors and randomly walk in the network. Let $U(s_i, t)$ represent the population in $\mathcal{R}(s_i)$ at the instant t, and $p(s_i, s_j, t)$ represent the estimated probability of users going to s_i from s_j at the instant t. The actual number of users going to s_i from s_j at the instant t is $F(s_i, s_j, t)$.

We suppose that each relay node and user (or mobile device) keep a heap respectively. The construction of each heap is based on the comparison of messages. The properties of the heap ensure that all messages popped by the heap will be the most "important" messages in the current heap. Here, we take the generating time of messages as the criterion.

Messages will be exchanged when a user (or mobile device) and a relay are close, and the distance is smaller than a threshold d_{th}. The exchanging process is described in Algorithm 1. Since each user has limited storage space, it would push all the messages it had into the relay's heap. Then a relay would

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on December 19,2022 at 06:34:13 UTC from IEEE Xplore. Restrictions apply.

Algorithm 1: Exchanging Process

input : Heap of user H_u, heap of relay node H_n, the next location of user O

output: An improved collection tree, B_{opt}

push(new message,H_u);(users keep gathering new messages before they exchange messages with relay nodes.)
while $H_u.size > 0$ ***and*** $distance(H_u, H_n) < d_{th}$ **do**
  push(pop(H_u),H_n);
if $O = parent(n)$ **then**
  while $H_u.size \leq max_{buf}$ ***and*** $H_n.size > 0$ **do**
    push(pop(H_n),H_u);

push the messages with the smallest collection time to the heaps of users which will visit the node's parent at the next time period. The number of messages is up to users' storage space max_{buf}.

To promise the quality of data aggregation, multiple messages are needed. The knowledge $K(i, t_1, t_2)$ is defined as the number of the messages about s_i at t_1 which the sink s_0 has received until t_2. Noticing that $K(i, t_1, t_2)$ is an non-decreasing function for t_2, we note that

$$T(i, t_2) = \arg\max_{t_1} K(i, t_1, t_2), \tag{3}$$

$$\max_{t_1} K(i, t_1, t_2) \geq K_0, \tag{4}$$

where K_0 is the threshold and

$$\forall i, t_2, K(i, 0, t_2) = K_0. \tag{5}$$

The delay of s_i at t_2 can be calculated as

$$D(i, t_2) = t_2 - T(i, t_2). \tag{6}$$

We need to construct a series of collection trees for different instants to control the flow of sensing data. The data will be transferred from low-priority nodes to high-priority nodes. The priority is determined by the hop count to the access point, i.e. the root in the determined collection tree. We define the parent of the node s_i at the instant t as $parent(i, t)$. The set of children for the node s_i at the instant t is noted as

$$parent^{-1}(i, t) = \{j | parent(j, t) = i\}. \tag{7}$$

The function of hop count, hop, is defined under the condition

$$\forall i,\ parent^{hop(s_i)}(i) = 0, \tag{8}$$

i.e., $hop(s_i)$ is the number of hop between s_i and the sink s_0.

We need to determine the matrix $\mathbf{Parent} = \{parent(i, t)\}$ to construct the collection trees. They should satisfy the connectivity and minimize the average delay from each point to the sink. The schedule can be formulated as:

$$\arg\min_{\mathbf{Parent}} \mathbb{E}\left(\sum_{t_2}\sum_{i} D(i, t_2)\right). \tag{9}$$

IV. Algorithm

The population of s_i at $t+1$, $U(s_i, t+1)$, is the sum of $F(s_i, s_j, t)$, where $s_j \in \mathcal{N}(s_i)$. And $\mathbf{P}_{(t)}$ is the transferring probability matrix at time t. So the expectation of $U(s_i, t+1)$ can be calculated by

$$\mathbb{E}(\mathbf{U}_{(t+1)}) = \mathbf{P}_{(t)}\mathbf{U}_{(t)}, \tag{10}$$

where

$$\mathbf{P}_{(t)} = [p(s_i, s_j, t)]_{k\times k} \tag{11}$$

and

$$\mathbf{U}_{(t)} = (U(s_0, t), U(s_1, t), \ldots, U(s_k, t))^T. \tag{12}$$

After t, $\mathbf{P}_{(t+1)}$ is generated,$\forall i, j$,

$$p(s_i, s_j, t+1) = \frac{F(s_i, s_j, t)}{U(s_j, t)}. \tag{13}$$

A. Greedy Collection Tree

We consider a simple case at first. Assuming that there are infinite users transferring from any node to any of its neighbors at any instant. Infinite messages are generated at a node and transferred to the node's parent. Therefore, the delay of s_i, $D(i, t_2)$, equals to the hop count from s_i to s_0. The problem (9) is converted to determine a tree which minimizes the total hop count, where s_0 is the root. Mathematically, the problem is defined as

$$\min \sum_i hop(s_i). \tag{14}$$

A greedy collection tree is constructed in Algorithm 2. It keeps the connectivity to the sink s_0. We first set the status of the sink s_0 as 1. Others' status are set as 0. And we define a variable, $level$, as 1. For all the nodes whose status are 0, find their neighbors with the current status' level. Then set itself a status of $level + 1$. $level$ is increased by one when none of the nodes have a neighbor with the current status level. The process is cycled until all the nodes' status are not 0. We note the collection tree constructed by the algorithm as $B_g(t)$.

In the algorithm, f is a metric of two adjacent nodes at the instant t. Then we propose two following algorithms based on two different objective functions f:

- Greedy Collection Tree (GCT)

  A node would simply choose the node with the largest transferring probability among the candidate nodes as its parent node. Here,

  $$f(i, j, t) = p(s_j, s_i, t). \tag{15}$$

- Dynamic Greedy Collection Tree (DGCT)

  To optimize the distribution of the messages, the entropy is considered. $K_{H_j}(i, k, t)$ is defined as the number of the messages from s_i at time t_k entering the heap H_j and still being H_j at time t ($t_k \leq t$). We note the frequency of messages, which is from s_i entering H_j at the time t_k and still being H_j at time t, as $\omega_{H_j}(i, k, t) = |H(j, t)|^{-1} K_{H_j}(i, k, t)$, where $|H(j, t)|$ is

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on December 19,2022 at 06:34:13 UTC from IEEE Xplore. Restrictions apply.

Algorithm 2: Greedy Collection Tree

input : A set of static relay nodes $\mathcal{S}$, nodes set within one hop count $\mathcal{N}$, the instant t, the objective function f

output: The collection tree $B_g(t)$

Set $status(s_0) \leftarrow 1$;
$\forall s_i \in \mathcal{S} \backslash \{s_0\}$, *Set* $status(s_i) \leftarrow 0$;
Set $level \leftarrow 1$;
Set $B_g(t) = \varnothing$;
while $\neg(\forall s \in \mathcal{S}, status(s) \neq 0)$ **do**
  $\forall s_i \in status^{-1}(0)$;
  if $\exists s_j \in \mathcal{N}(s_i), status(s_j) = level$ **then**
    $B_g(t) = B_g(t) \cup (s_i, s_{\arg\max_j f(i,j,t)})$;
    $status(s_i) \leftarrow level + 1$;
  $level \leftarrow level + 1$;
return $B_g(t)$;

defined as the the number of messages in heap H_j at time t. And the frequency of messages, which are originally stored in heap H_j and still in heap H_j at time t, is defined as $\omega_{H_j}(t) = |H(j,t)|^{-1} K_{H_j}(t)$, where $K_{H_j}(t)$ is the number of message originally stored in heap H_j and still being in heap H_j at time t.

The entropy of the heap H_j at time t is

$$S(H_j(t)) = -\sum_{\substack{\{(i,k,t)| \\ \omega_{H_j}(i,k,t) \neq 0\}}} \omega_{H_j}(i,k,t) \ln \omega_{H_j}(i,k,t) - \omega_{H_j}(t) \ln \omega_{H_j}(t). \tag{16}$$

The estimated population transferring from s_i to s_j at the instant t is $p(s_j, s_i, t)U(s_i, t)$, which is noted as $\tilde{F}(s_i, s_j, t)$. The set of the first $\tilde{F}(s_i, s_j, t)$ messages in $H_i(t)$ is noted as $C(s_i, s_j, t)$. To decide where to go next, s_i would testify the entropy of $H_j(t) \cup C(s_i, s_j, t)$. At the instant t, the next hop of s_i is the node s_j which maximizes

$$f(i,j,t) = \frac{S\left(H_j(t) \cup C(s_i, s_j, t)\right) - S(H_j(t))}{S(H_j(t))}, \tag{17}$$

where $s_j \in \mathcal{N}(s_i)$.

From the description above, we can know that GCT and DGCT are only different in their objective functions f.

B. Improved Greedy Collection Tree

Now we consider an another specific case that the environment is sensed only at the instant 0. Our objective is sending all the messages to the sink as soon as possible. Assuming that the transferring matrix is independent of time. Assuming that $\forall i, t, U(s_i, t) = K_0$, which means that we need to collect all the data and the fluidity is low. We neglect the change of population.

Algorithm 3: Improved Greedy Collection Tree

input : The greedy collection treeB_g, the transferring probability matrix P

output: An improved collection tree B_{opt}

Function *[s_m,$D_e(s_m)$]=Delay(B)*
  for i = *1 to* k **do**
    $D_e(s_i) = \frac{|sub(s_i)|}{p(s_{parent(i)}, s_i)} + hop(s_i)$;
  Find s_m *which maximizes* $D_e(\cdot)$;
  return *[s_m,$D_e(s_m)$];*
$Flag \leftarrow True$;
while $Flag = True$ **do**
  [s_g,t_g]=Delay(B_g);
  for $s_i \in sub(s_g) \cup \{parent^n(s_g), n \in [0, hop(s_g)]\}$ **do**
    for $s_j \in \mathcal{N}(s_i) \backslash sub(s_i)$ **do**
      $B_{temp} \leftarrow B_g$;
      In B_{temp}, *parent*$(i) \leftarrow j$;
      [s_{temp},t_{temp}]=Delay(B_{temp});
      if $t_{temp} < t_g$ **then**
        $B_{res} \leftarrow B_{temp}$;
        $t_g \leftarrow t_{temp}$;
  if $B_{res} \neq B_g$ **then**
    $B_g \leftarrow B_{res}$;
  else
    $Flag \leftarrow False$;
    $B_{opt} \leftarrow B_g$;
return B_{opt};

We define $t_{fin}(s_i)$ as the retiring time of s_i. After $t_{fin}(s_i)$, node s_i will neither receive messages from its children nor send messages to its parent. For the node s_i which has children, i.e., $parent^{-1}(i) \neq \varnothing$, we have the inequality

$$t_{fin}(s_i) \geq \max_{j \in parent^{-1}(i)} t_{fin}(s_j) + 1. \tag{18}$$

Because the node s_i must handle with the messages sent from its children.

The delay for messages sensed in the region $\mathcal{R}(s_i)$ can be formulated as

$$D_e(s_i) = \max_{j \geq 0} \left(hop(s_{parent^j(i)}) + t_{fin}(s_{parent^j(i)})\right). \tag{19}$$

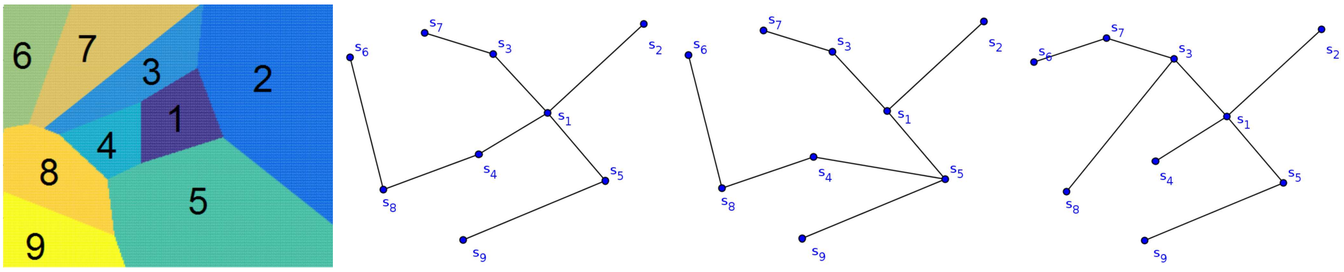
According to Eq. (18), $\forall j \in parent^{-1}(i)$,

$$hop(s_i) + t_{fin}(s_i) \geq hop(s_i) + t_{fin}(s_j) + 1 = hop(s_j) + t_{fin}(s_j). \tag{20}$$

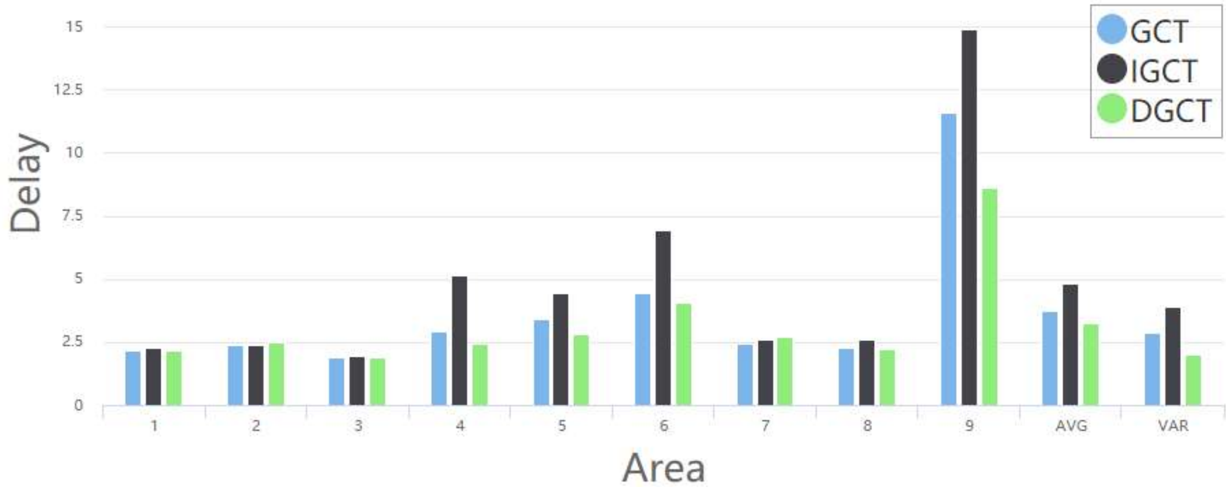
So, we have $D_e(s_i) = t_{fin}(s_0)$ for each node, i.e., our objective is equivalent to minimize $t_{fin}(s_0)$.

The total messages, which can be received by s_i, are products of the cardinality of its sub-tree and K_0. For each

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on December 19,2022 at 06:34:13 UTC from IEEE Xplore. Restrictions apply.



(a) RegionMap (b) GCT (c) IGCT (d) DGCT



(e) Delay in Each Area

Fig. 2: An instance

instant, it is estimated that there will be $K_0 \cdot p(s_{parent(i)}, s_i)$ users transferring from the node s_i to its parent node $s_{parent(i)}$.

The minimum t which makes $|H_i(t)| = 0$ is noted as $t_{clear}(s_i)$, i.e., the number of messages in s_i at $t_{clear}(s_i)$ is 0. We have

$$t_{clear}(s_i) \geq \frac{|sub(s_i)|}{p(s_{parent(i)}, s_i)}, \tag{21}$$

where $|sub(s_i)|$ is the node number of s_i's sub-tree. As the node s_i transfers messages to its parent node at the rate of $K_0 \cdot p(s_{parent(i)}, s_i)$ and the maximal number of receiving messages is $K_0 \cdot |sub(s_i)|$.

We have

$$t_{fin}(s_i) = \max\left(\frac{|sub(s_i)|}{p(s_{parent(i)}, s_i)}, \max_{j \in parent^{-1}(i)} t_{fin}(s_j) + 1\right) \tag{22}$$

where $i \neq 0$, and

$$t_{fin}(s_0) = \max_{j \in parent^{-1}(0)} t_{fin}(s_j) + 1. \tag{23}$$

We conclude that the collection tree should minimize

$$t_{fin}(s_0) = \max_i \left(\frac{|sub(s_i)|}{p(s_{parent(i)}, s_i)} + hop(s_i)\right), \tag{24}$$

where $i \neq 0$.

We modify the solution $B_g(t)$ of GCT algorithm. And we note the node which maximizes $\frac{|sub(s_i)|}{p(s_{parent(i)}, s_i)} + hop(s_i)$ as s_m. To decrease $t_{fin}(s_0)$, we have three following strategies.

- Change $parent$ of $parent^j(m), j \in [1, hop(s_m)]$ to a node with lower hop count to decrease $hop(s_m)$.
- Change $parent(m)$ to a node with higher transferring probability to increase $p(s_{parent(m)}, s_m)$.
- Change $parent(j)$, $s_j \in sub(s_m) \backslash \{s_m\}$ to other node to decrease $|sub(s_m)|$.

During the modification, a node should not choose the nodes in its sub-tree as its new parent node to promise the connectivity. Each choice should be enumerated. The solution which decreases $t_{fin}(s_0)$ most would be taken as the modification. The process cycles until the tree does not change and the final tree is noted as B_{opt}. The detailed algorithm, IGCT, is given in Algorithm 3.

V. Performance Evaluation

In this section, we observe the performance of our algorithms under different initial conditions. The set of nodes S, where $|S| = 9$, is deployed in a 25×25 area. The initial population $U(s_i)$ in each region and the initial transferring probability matrix are generated randomly. We will observe the delay of each area and its average and standard deviation. Under this situation, the Dynamic Greedy Collection Tree

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on December 19,2022 at 06:34:13 UTC from IEEE Xplore. Restrictions apply.

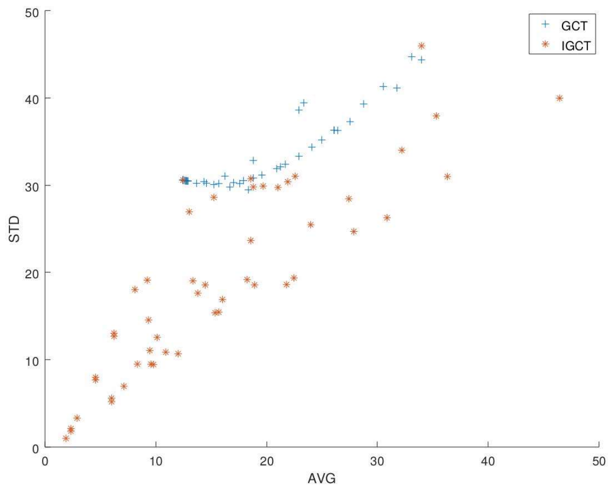


Fig. 3: Average Delay and its Standard Deviation among Regions

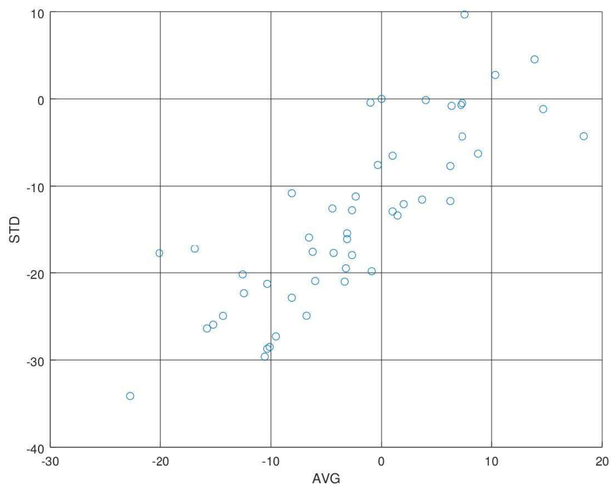


Fig. 4: Shift of IGCT according to GCT

(DGCT) has the performance of the lowest average delay and the smallest standard deviation of average delay.

The map is shown in Fig. 2(a). The index of each region $\mathcal{R}(s_i)$ is indicated on the map. An instance of collection trees generated by our algorithms under one same transferring probability matrix is shown in Fig. 2(b), 2(c), 2(d). In Fig. 2(e), we show the average delay in each region under three algorithms.

The simulation is conducted 50 times with the same map and different transferring probability matrix.We compare the performance of GCT and IGCT in the metrics of average delay and its standard deviation among all the regions. In Fig. 3, we find that IGCT can reach low latency and low jitter in some cases. In Fig. 4, we observe the shift of IGCT according to GCT, i.e. the difference of two algorithms in aspect of average delay and its standard deviation. We find that in most cases, IGCT has lower standard deviation, which means that IGCT makes the knowledge for each region more balanced.

To indicate the freshness of messages in the heap H_0, we propose an entropy-like metric $S_\Delta(H_j(t))$ to indicate the freshness of messages in the heap H_j.

$$S_\Delta(H_j(t)) = S(H_j(t)) - S(H_j(t-\Delta)) \tag{25}$$

where t represents the current time. Messages generated in the period of $[t-\Delta, t]$ are considered.

The evolution of $S_\Delta(H_0(t))$ in three algorithms for the above instance is shown in Fig. 5, in which $\Delta = 10$. The curve of $S_\Delta(H_0(t))$ proves that the messages are fresher under the dynamic collection tree(DGCT).

Therefore, DGCT performs better with low and stable latency, which verifies the rationality of the modified objective function. IGCT shows a good advance in standard deviation and the freshness of messages in general compared with GCT. The improvement contributes to better efficiency and stability. Complex situations can be simplified according to the conditions of static nodes, mobile nodes and information

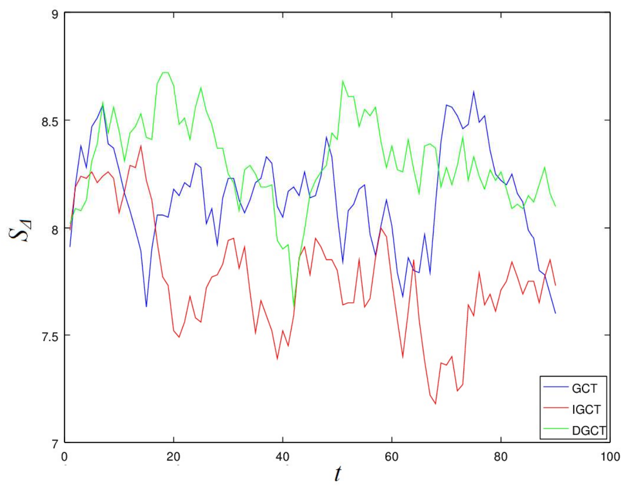


Fig. 5: The evolution of S_Δ

collection requirement so that the hypothesis of the algorithms can be satisfied.

VI. Conclusion

In this paper, we consider the application of wireless coverage problem in mobile crowdsensing. The scenario is that the sensing information needs to be delivered to the sink through existed relays and mobile devices. Without cellular network, the wireless communication is based on wireless sensor network. Collection trees should be founded to minimize the average delay. We propose two kinds of algorithms for constructing collection trees to optimize the average delay. First we propose two algorithms based on greedy strategy with different objective functions, GCT and DGCT. Then under the condition of low fluidity, IGCT is proposed. The simulation proves that IGCT performs generally better than GCT in the aspect of jitter. And DGCT has the best freshness of messages for the sink.

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on December 19,2022 at 06:34:13 UTC from IEEE Xplore. Restrictions apply.

Acknowledgment

This work was supported by the National Key R&D Program of China [2019YFB2102200]; the National Natural Science Foundation of China [61872238, 61972254], Shanghai Municipal Science and Technology Major Project [2021SHZDZX0102], and the Huawei Cloud [TC20201127009]. The authors would like to thank Ailun Song, Jiahao Fan, Yue Ma and Shiqing Tang for their contributions on this paper. Xiaofeng Gao is the corresponding author.

References

[1] Free community-based mapping, traffic & navigation app. https://www.waze.com/

[2] Cheng, C.T., Ganganath, N., Fok, K.Y.: Concurrent data collection trees for iot applications. IEEE Transactions on Industrial Informatics pp. 1–1 (2016)

[3] Chessa, S., Corradi, A., Foschini, L., Girolami, M.: Empowering mobile crowdsensing through social and ad hoc networking. IEEE Communications Magazine **54**(7), 108–114 (2016)

[4] Ding, S., He, X., Wang, J., Qiao, B., Gai, K.: Static node center opportunistic coverage and hexagonal deployment in hybrid crowd sensing. Journal of Signal Processing Systems **86**(2-3), 251–267 (2017)

[5] Gnawali, O., Fonseca, R., Jamieson, K., Moss, D., Levis, P.: Collection tree protocol. In: Proceedings of the 7th International Conference on Embedded Networked Sensor Systems, SenSys 2009, Berkeley, California, USA, November 4-6, 2009 (2009)

[6] Han, K., Chen, C., Zhao, Q., Guan, X.: Trajectory-based node selection scheme in vehicular crowdsensing. In: IEEE/CIC International Conference on Communications in China (ICCC), Shenzhen, China, November 2-4. pp. 1–6 (2015)

[7] Jiang, C., Gao, L., Duan, L., Huang, J.: Economics of peer-to-peer mobile crowdsensing. In: IEEE Global Communications Conference (GLOBECOM), San Diego, CA, USA, December 6-10. pp. 1–6 (2015)

[8] Kong, Q., Yu, J., Lu, R., Zhang, Q.: Incentive mechanism design for crowdsourcing-based cooperative transmission. In: IEEE Global Communications Conference (GLOBECOM), Austin, TX, USA, December 8-12. pp. 4904–4909 (2014)

[9] Liu, J., Bic, L., Gong, H., Zhan, S.: Data collection for mobile crowdsensing in the presence of selfishness. EURASIP Journal on Wireless Communication and Networking **2016**, 82 (2016)

[10] Lo, SM, Lin, WH, Chen, CY, Tseng, YG: Optimal coloring for data collection in tree-based wireless sensor networks. Theor Comput Sci (2017)

[11] Nguyen, P., Nahrstedt, K.: Crowdsensing in opportunistic mobile social networks: A context-aware and human-centric approach. CoRR **abs/1704.08598** (2017)

[12] Ottman, N.B., Liscano, R., Heydari, S.S.: An analysis of the collection tree protocol (ctp) in mobile sensing environments. In: 2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC) (2017)

[13] Ramdhan, Nihel, Sliti, Maha, Boudriga, Noureddine: A tree-based data collection protocol for optical unmanned aerial vehicle networks. Computers & Electrical Engineering (2018)

[14] Regalia, B., McKenzie, G., Gao, S., Janowicz, K.: Crowdsensing smart ambient environments and services. Transaction in GIS **20**(3), 382–398 (2016)

[15] Rui, L., Zhang, P., Huang, H., Qiu, X.: A location-dependent task assignment mechanism in vehicular crowdsensing. International Journal of Distributed Sensor Networks (IJDSN) **12**(9) (2016)

[16] Rula, J.P., Bustamante, F.E.: Crowdsensing under (soft) control. In: IEEE Conference on Computer Communications (INFOCOM), Kowloon, Hong Kong, April 26 - May 1. pp. 2236–2244 (2015)

[17] Wennerström, H., Rohner, C.: Towards even coverage monitoring with opportunistic sensor networks. In: ACM Workshop on Challenged Networks (CHANTS@MobiCom), New York City, New York, USA, October 3-7. pp. 81–86 (2016)

[18] Wu, D., Zhang, Y., Luo, J., Li, R.: Efficient data dissemination by crowdsensing in vehicular networks. In: IEEE International Symposium of Quality of Service (IWQoS), Hong Kong, China, May 26-27. pp. 314–319 (2014)

[19] Xiong, H., Zhang, D., Chen, G., Wang, L., Gauthier, V.: Crowdtasker: Maximizing coverage quality in piggyback crowdsensing under budget constraint. In: IEEE International Conference on Pervasive Computing and Communications (PerCom), St. Louis, MO, USA, 23-27 March. pp. 55–62 (2015)

[20] Xiong, H., Zhang, D., Wang, L., Chaouchi, H.: EMC^3: Energy-efficient data transfer in mobile crowdsensing under full coverage constraint. IEEE Transactions on Mobile Computing (TMC) **14**(7), 1355–1368 (2015)

[21] Yao, H., Xiong, M., Liu, C., Liang, Q.: Encounter probability aware task assignment in mobile crowdsensing. Mobile Networks and Applications (MONET) **22**(2), 275–286 (2017)

[22] Zhang, D., Xiong, H., Wang, L., Chen, G.: Crowdrecruiter: Selecting participants for piggyback crowdsensing under probabilistic coverage constraint. In: ACM Conference on Ubiquitous Computing (UbiComp), Seattle, WA, USA, September 13-17. pp. 703–714 (2014)

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on December 19,2022 at 06:34:13 UTC from IEEE Xplore. Restrictions apply.