Pers Ubiquit Comput
DOI 10.1007/s00779-016-0974-0

ORIGINAL ARTICLE

Participant selection for data collection through device-to-device communications in mobile sensing

Yu Wang[1,2] · **Hanshang Li**[2] · **Ting Li**[2]

Received: 1 September 2016 / Accepted: 30 September 2016

Abstract The appearance of smart mobile devices with communication, computation and sensing capability and increasing popularity of various mobile applications have caused the explosion of mobile data recently. In the same time, mobile sensing has been emerging as a new sensing paradigm where vast numbers of mobile devices are used for sensing and collecting huge amounts of mobile data in cities. One of the challenges faced by mobile sensing is how to efficiently collect the huge amount of mobile data beyond the existing capacity of 4G networks. In this paper, we investigate the feasibility of collecting data packets from mobile devices through device-to-device communications by carefully selecting the subset of relaying (or/and sensing) devices. We formulate these problems as optimization problems and propose a set of solutions to solve them. Our experiments over a real-life mobile trace confirm the effectiveness of the proposed idea.

Keywords Participant selection · Data collection · Device-to-device communication · Mobile crowd sensing · Mobile sensing

✉ Yu Wang
yu.wang@uncc.edu

Hanshang Li
hli39@uncc.edu

Ting Li
tli8@uncc.edu

[1] College of Information Engineering, Taiyuan University of Technology, Taiyuan 030024, China

[2] Department of Computer Science, University of North Carolina at Charlotte, Charlotte, NC 28223, USA

1 Introduction

With the increasing popularity of mobile applications and services for smart devices, we are currently facing the challenges of mobile big data explosion. Based on the most recent Cisco's report [1], mobile data traffic grew 74% in 2015 and reached 3.7 exabytes per month at the end of 2015, which was nearly 4000 times the one in 2005. Cisco also forecasts that mobile data traffic will surpass 30.6 exabytes per month in 2020. Even though smart devices only represent 36% of the total mobile devices and connections, they account for 89% of the mobile data traffic. The widespread availability of smart devices equipped with a rich set of built-in sensors has also enabled a new sensing paradigm, *mobile crowd sensing* (MCS) [2], for collecting and sharing sensing data from surrounding environment. MCS has been widely used for different sensing applications, such as public safety [3, 4] , traffic planning [5, 6], localization [7, 8], environment monitoring [9, 10] and urban dynamic mining [11, 12]. In the same time, this new sensing paradigm makes the mobile data explosion severer.

The current cellular networks do not have enough capacity to support all of the fast-growing mobile big data from these smart devices and Internet of Things. Different offloading solutions (such as WiFi networks [13, 14] or femtocells [15]) have been adopted. According to Cisco [1], 51% of total mobile data traffic was offloaded onto the fixed network through WiFi or femtocell in 2015, and this is the first-time offload traffic exceeded cellular traffic. Recently, offloading cellular traffic through opportunistic device-to-device (D2D) communications [16–18] among mobile phones becomes a new and possible solution. Compared with current WiFi or femtocell solutions, this method uses occasional D2D contact opportunities to deliver data rather than the fixed network infrastructure.

Published online: 11 November 2016

The major advantage is low cost and easy to deploy. Han et al. [16] study how to select the initial set of mobile users to push the content to all users in the networks via D2D, and their proposed heuristics can improve the delivery efficiency and offload a large fraction of data from the cellular network. Li et al. [17] study the problem of multiple mobile data offloading through D2D among different data subscribers under resource constraints. Zhu et al. [18] study offloading peer-to-peer traffic among mobile users with D2D relays. In this paper, we focus on offloading data collection for mobile sensing data via D2D relays instead of broadcasting traffic from the service provider to all subscriber users (as in [16, 17]) or peer-to-peer traffic between any two users (as in [18]).

In most existing mobile sensing systems [19–23], the sensing data are collected via cellular networks with the assumption that the size of sensing data is not large. However, with the new types of multimedia sensing (videos, audios, high-resolution images, real-time streaming, etc.) and increasing number of sensing devices (smartphones, smart watches, smart glasses, smart meters, smart vehicles, RFIDs, etc.), the amount of mobile sensing data grows to a scale that traditional cellular methods may not handle. Wang et al. [24] first consider leveraging the delay-tolerant mechanisms by offloading the data to Bluetooth/WiFi gateways or data-plan users. The major goal for their method is to reduce the energy consumption and data cost of data-plan users. Karaliopoulos et al. [25] consider a joint user recruitment problem for both sensing and data collection, where the data collection is done via D2D communications. They formulate the selection of users as a minimum cost set cover problem and propose greedy heuristics to solve it. However, the solution has large time complexity due to the huge search space over all space–time paths across the network, which makes it not suitable for large-scale data collection. In this paper, we focus on the data collection phase of mobile data sensing by carefully selecting a few mobile participants as relay nodes to help with data propagation via D2D relays. By doing so, we limit the search space and make our algorithm more efficient. In addition, since we use multiple space–time paths for data collection from the source (in [25] only one space–time path is selected for one source), our method can achieve better delivery ratio too. We also consider the joint problem where the selected participants perform both sensing and data collection.

In summary, in this paper, we study how to select a small subset of participants as relaying (or/and sensing) devices so that the data propagation via these D2D relays can achieve certain level of delivery ratio in mobile crowd sensing. We formulate the problem as various optimization problems in Sect. 2. Then we propose simple but efficient solutions in Sects. 3 and 4 for relay selection and joint relay/sensing selection, respectively. In Sect. 5, we conduct experiments over a real-life mobile trace to confirm the effectiveness of the proposed algorithms. Section 6 provides a brief review of related work. A preliminary version of this paper appeared in [26].

2 System model and problem statement

2.1 System model

We consider the relay node selection problem for sensing data collection. We assume that a mobile user set $User = \{u_1, u_2, \ldots, u_n\}$, which includes n mobile users who are willing to participate into sensing and the delivery of the sensing data. This candidate set is assumed very large given the popularity of smartphones. Each mobile users can visit m different locations, denoted as $Location = \{l_1, l_2, \ldots, l_m\}$. The whole time period is evenly divided into T sequential time slots; thus, time $t \in [1, T]$. Each user has her own visiting pattern over both temporal and spacial domains. We use P to denote the visiting probability matrix of all users, which its element $p(u_i, l_j, t)$ (or $p(i, j, t)$ for short) represents the probability of mobile user u_i to make a visit at location l_j during time slot t. There are various methods to estimate the visiting probability of each user based on historical traces, and our proposed solution can use any such existing method. In our simulations, we utilize a simple statistic-based method, which is illustrated in Sect. 5.1. We assume that these visiting probabilities are independence to the others for each particular mobile user. We also assume that there is a set of sensing tasks $Task = \{q_1, q_2, \ldots, q_o\}$, which includes o sensing tasks. Each task i has a tuple of target $(l(q_i), t(q_i))$, which represents the temporal and spacial target of this task. Note that here each task only has a single interest point in the temporal and spacial domain; however, it could be easily extended to the case where each task has multiple interest points.

To enable device-to-device communications, we assume that two nodes can discover each other and transfer sensing data to each other when they both visit the same location within a particular time slot. For each piece of sensing data, it is generated at a source node s (a mobile device which performs the sensing task and generate the data) and needs to be delivered to a sink node d (a mobile device or a static device at certain location). For simplicity, we only focus on the selection of relay nodes for the collection of sensing data to a single sink. However, all proposed methods are general enough to handle multiple sources/sinks. To enhance the delivery probability, we assume that restricted flooding (i.e., epidemic [27]) is used within the selected relay nodes. Figure 1 shows an example of data delivery

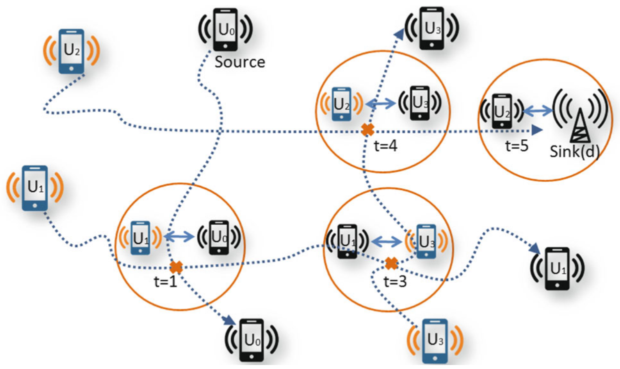

Fig. 1 Example of data delivery via multi-hop D2D communications

via multi-hop D2D relays. In this figure, dashed curves are trajectories of devices, a circle represents an encounter between two devices, and the device in black indicates that it has a copy of the data. The one marked with "source" is the device performing sensing and generating the data, and the sink is an access point or tower. During the encountering among multiple devices, the data could be transmitted from one device to another. Through this type of multi-hop D2D transmission, sensing data could be delivered to a sink which is not able to directly communicate with the source. Note that there we assume the data collection is through only device-to-device communications, while in reality, a hybrid solution (combining D2D and direct communication with cellular tower) could be desired.

2.2 Relay selection problems

First, we only consider the participant selection for D2D data collection in MCS. For simplicity, we only focus on the selection of relay nodes for the collection of a single piece of sensing data. However, the method is general enough to handle multiple data pieces. The key challenging is how to identify a small set of relay nodes from the huge candidate pool *User* while guarantee certain level of data delivery. This is different with traditional DTN routing, in which relay nodes are dynamically selected during the routing. We can formally define the relay selection problem as the following optimization problem.

Definition 1 Given the volunteering users *User* (with their historical call and location traces), and the source s (who generates the sensing data) and destination d of the sensing data, *minimum relay problem* is to find a subset $U(s, d)$ of mobile users from *User* as the relay nodes with the objective to

$$\begin{aligned}&\min |U(s,d)|\\ &\text{s.t. } p_r(U(s,d),s,d) \geq \gamma.\end{aligned}$$

in which $p_r(U(s,d),s,d)$ is denoted as the probability that the sensing data can be delivered to its destination sink and γ represents a threshold of the probability.

Similarly, the optimization problem can be defined as another formulation as well.

Definition 2 Given the volunteering users *User* (with their historical call and location traces), and the source s and destination d of the sensing data, *K relay problem* is to find a subset $U_{s,d}$ of K mobile user from *User* as the relay nodes with the objective to

$$\begin{aligned}&\max p_r(U(s,d),s,d)\\ &\text{s.t. } |U(s,d)| = K.\end{aligned}$$

Note that both versions of the problem are computational challenging, since even with perfect predication of visiting patterns this problem can be reduced to a set cover problem which is NP-hard. Therefore, in the next section, we are looking for efficient heuristics to tackle them.

2.3 Joint sensing and relay selection problems

In the problems above, we focus on the selection of participants who will only participate the D2D data collection, by assuming that the participants for sensing tasks have been selected and fixed via existing participant selection methods [19–23]. However, it is very natural to consider the selection of the same group participants for both sensing and data collection purposes. Then, we can define the following joint problems.

Definition 3 Given the volunteering users *User* (with their historical call and location traces), and the sensing task q and destination d of the sensing data, *minimum sensing and relay problem* is to find a subset $U(q, d)$ mobile user from *User* as the selected participants with the objective to

$$\begin{aligned}&\min |U(q,d)|\\ &\text{s.t. } p_s(U(q,d),q,d) \geq \gamma.\end{aligned}$$

in which $p_s(U(q,d),q,d)$ is denoted as the probability that the target information can be collected **and** the sensing data can be delivered to the destination sink.

Definition 4 Given the volunteering users *User* (with their historical call and location traces) , and the sensing task q and destination d of the sensing data, *K sensing and relay problem* is to find a subset $U(q, d)$ of K mobile user from *User* as the selected participants with the objective to

$$\max p_s(U(q,d),q,d)$$
$$\text{s.t. } |U(q,d)| = K.$$

Note the problems above are defined for a single sensing task q. We can also consider the optimization over the whole sensing task set $Task$, i.e., we select the common set of participants $U(Task, d)$ to perform all sensing tasks in $Task$. The only difference in the definitions is using $\sum_{q_i \in Task} p_s(U(Task,d),q_i,d) \geq \gamma$ as the constraint in the minimum sensing and relay problem or $\max \sum_{q_i \in Task} p_s(U(Task,d),q_i,d)$ as the optimization goal of the K sensing and relay problem. All of these problems are obviously NP-hard.

3 Relay selection for D2D collection

Recall that we use a flooding/epidemic strategy to deliver the sensing data via multiple hops among selected relay nodes. The selection criteria for relay nodes may rely on how to estimate the delivery probability of a particular group of relay nodes. To achieve this, we first introduce a space–time graph-based method; then, we propose our greedy-based algorithm for relay node selection.

3.1 Estimation of delivery probability via space–time graphs

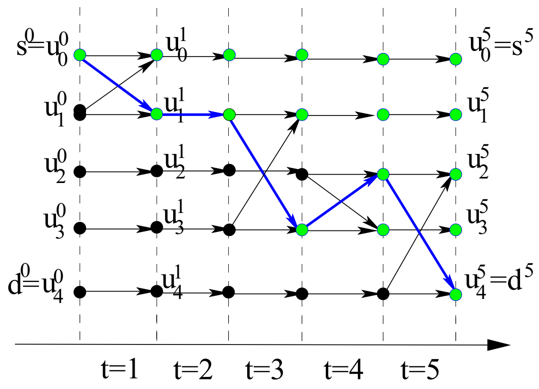

Fig. 2 *Space–time graph* the corresponding space–time graph $\mathcal{G}$ of Fig. 1, where a space–time path from the source s to the sink d is highlighted

To capture the evolving characteristics in both spacial and temporal spaces, we adopt the *space–time graph* [28, 29] to model the time-evolving D2D links among selected relay nodes. Let $U(s,d) = \{u_1,\ldots,u_r\}$ is the relay nodes selected for source s and sink d. We can define a space–time graph $\mathcal{G}^{U(s,d)} = (\mathcal{V},\mathcal{E})$, which is a directed graph defined in both spacial and temporal spaces. Hereafter, we simply use $\mathcal{G}$ to represent $\mathcal{G}^{U(s,d)}$. In $\mathcal{G}$, $T+1$ layers of nodes are defined and each layer has $r+2$ nodes (corresponding to $\{u_0 = s, u_1,\ldots,u_r,u_{r+1} = d\}$), thus the whole vertex set $\mathcal{V} = \{u_j^t | j = 0,\ldots,r+1 \text{ and } t = 0,\ldots,T\}$ and there are $(r+2)(T+1)$ nodes in total. Figure 2 illustrates the corresponding space–time graph for the network shown in Fig. 1. Two kinds of links (spacial links and temporal links) are added between consecutive layers in the edge set $\mathcal{E}$. A temporal link $\overrightarrow{u_j^{t-1}u_j^t}$ (those horizontal links in Fig. 2) connects the same node u_j across consecutive $(t-1)$th and tth layers, which represents the node carrying the data in the tth time slot. A spacial link $\overrightarrow{u_j^{t-1}u_k^t}$ represents a forwarding possibility from one node u_j to its encountering node u_k in the tth time slot (i.e., u_j encounters u_k in time slot t). By defining the space–time graph $\mathcal{G}$, any communication operation in the time-evolving network can be simulated on this directed graph. For example, the propagation path in Fig. 1 is highlighted in Fig. 2.

To estimate the delivery probability, we need first define the link probability $p(e)$ of each link $e \in \mathcal{E}$), i.e., the probability of existing such a link. For each temporal link $\overrightarrow{u_j^{t-1}u_j^t}$, its link probability is set to 1 since a node can always hold the data. For a spacial link $\overrightarrow{u_j^{t-1}u_k^t}$, its link probability is calculated as follows.

$$p\left(\overrightarrow{u_j^{t-1}u_k^t}\right) = \left(1 - \prod_{i=1}^{m}(1 - p(j,i,t)p(k,i,t))\right) \cdot r\left(\overrightarrow{u_j^{t-1}u_k^t}\right),$$

where $1 - \prod_{i=1}^{m}(1 - p(j,i,t)p(k,i,t))$ is the probability that node u_j and u_k are colocated at any location and $r(\overrightarrow{u_j^{t-1}u_k^t})$ is the link reliability (representing the successful transfer over the encounter). If u_k is a location l_k instead of a mobile user, $p(\overrightarrow{u_j^{t-1}u_k^t}) = p(j,k,t) \cdot r(\overrightarrow{u_j^{t-1}u_k^t})$.

We then define the delivery probability of a space–time graph $\mathcal{G}$ as $p^{\mathcal{G}}(s^0, d^T)$ regarding the source s and destination d. It is the probability that a packet sent from node s over the routing topology $\mathcal{G}$ reaches node d under flooding-based routing. Similar definition is used in [30] as broadcast reliability. To efficiently calculate this delivery probability is not an easy job. Actually, it is known that the computation of such reliability over general graphs is a problem of NP-hard [31]. Fortunately, the nice loop-free property of our space–time graph model allows us to compute the reliability very efficiently with a dynamic programming (DP) algorithm [30]. Basically, for any node u_i^t in $\mathcal{G}$, its delivery probability from the source node s can be calculated as follows:

$$p^{\mathcal{G}}\left(s^0, u_i^t\right) = 1 - \prod_{\overrightarrow{u_j^{t-1}u_i^t}\in\mathcal{G}} \left(1 - p^{\mathcal{G}}\left(s^0, u_j^{t-1}\right)p\left(\overrightarrow{u_j^{t-1}u_i^t}\right)\right).$$

Given the structure $\mathcal{G}$ defined by r relay nodes, starting from a source node, the dynamic programming algorithm can compute the delivery ratio of all other nodes within time of $O(rT(\log(rT)+r))$. Notice that the time complexity of DP algorithm is the same with that of Dijkstra's algorithm. Given the relay node set $U(s, d)$ for source s and sink d, we can estimate the delivery probability based on the space–time graph $\mathcal{G}$ as follows: $p_r(U(s,d),s,d) = p^{\mathcal{G}}(s^0, d^T)$.

3.2 Relay selection algorithm

Then the relay selection algorithm is quite straightforward. In each step, we greedily select the user u_i which leads to maximal improvement of $p_r(U(s,d),s,d)$ into $U(s, d)$. Repeat this until either the delivery probability reaches the threshold γ for minimum relay problem or $U(s, d)$ has K users for K relay problem. However, there is still a starting problem, since initially when $U(s, d)$ is empty or just with a few users the space–time graph $\mathcal{G}^{U(s,d)}$ may not be connected at all (i.e., $p_r(U(s,d),s,d) = 0$). In this case, adding any single user may not improve the delivery probability. Therefore, instead of considering improvement of $p_r(U(s,d),s,d)$, we simply pick the user who is the most active (in term of visited locations). Detailed algorithm is given in Algorithm 1.

Next we consider the time complexity of Algorithm 1. Since the complexity of the initial step to form a connected space–time graph (lines 2–3) is much smaller than the complexity of relay selection step (lines 4–7), we only focus on the latter. First, the while loop will be performed K rounds in K relay problem and n rounds in minimum relay problem since in the worst case we need to select all participants to achieve the threshold. Second, the for loop is bounded by n rounds. Third, the complexity of DP algorithm for the estimation of delivery probability is $O(rT(\log(rT)+r))$ and r is the size of selected relay group which is bounded by K or n in K relay problem and minimum relay problem. Therefore, the time complexity of Algorithm 1 is bounded by $O(nKT(\log(KT)+K))$ or $O(n^2T(\log(nT)+n))$ for K relay problem or minimum relay problem, respectively.

Algorithm 1 Relay Selection Algorithm

```
Input: potential user set $User$, visiting probability matrix
    $P$ of all users in $User$, the source $s$ and the sink $d$.
Output: selected relay nodes $U(s,d)$.
1: $U(s,d) = \emptyset$
2: while $\mathcal{G}^{U(s,d)}$ is not connected do
3:     Choose the most active user and add it into $U(s,d)$
4: while $|U(s,d)| < K$ or $p_r(U(s,d),s,d) < \gamma$ (for $K$ relay
   problem or minimum relay problem, respectively) do
5:     for all $u_i \in User$ and $\notin U(s,d)$ do
6:         Calculate the improvement of $p_r(U(s,d),s,d)$ by
           adding $u_i$ in to $U(s,d)$ (Section 3)
7:     Select the user $u_i$ with the largest reliability improvement and add it into $U(s,d)$
8: return $U(s,d)$
```

4 Joint sensing and relay selection

When we consider the joint sensing and relay selection problems (defined in Definitions 3 and 4), we have to first estimate the sensing capability of each participants and then integrate it into the participant selection procedure.

4.1 Estimation of sensing and delivery probability

Recall that each task has a target tuple of location and time. Therefore, given a specific task q_i and a set of selected participant $U(q_i,d)$, the probability that this task can be performed by one participant $u_j \in U(q_i,d)$ can be obtained. Basically, the probability of u_j making a visit to location $l(q_i)$ at time $t(q_i)$ is $p(u_j, l(q_i), t(q_i))$.

Here, we assume that the probability of sensing and data delivery is independent to each another. Therefore, the probability of sensing and delivery of a sensing task q_i with a particular source can be calculated by multiply the sensing probability (at time $t(q_i)$) with the delivery probability (from time $t(q_i)$ to T over the space–time graph). Thus, given a sensing task q_i and a selected participant $u_j \in U(q_i,d)$ as the source (the sensing performer), the sensing and delivery probability from this participant is

$$p_s(u_j, U(q_i,d), q_i, d) = p(u_j, l(q_i), t(q_i)) \cdot p^{\mathcal{G}}\left(u_j^{t(q_i)}, d^T\right)$$

Then overall sensing and relay probability of task q_i by $U(q_i,d)$ is:

$$p_s(U(q_i,d), q_i, d) = 1 - \prod_{u_j \in U(q_i,d)} (1 - p_s(u_j, U(q_i,d), q_i, d)).$$

4.2 Sensing and relay selection algorithm

The participant selection algorithm basically is still the same. The only difference is now the joint sensing and relay probability is considered instead of relay probability.

Algorithm 2 shows the detailed algorithm. If we consider the selection over the whole sensing task set *Task*, the improvement of sensing and relay probability should be over the summation of all sensing tasks, i.e., the improvement of $\sum_{q_i \in Task} p_s(U(Task, d), q_i, d)$. Since Algorithm 2 is similar to the one for relay selection, the time complexity of this algorithm is bounded by $O(nKT(log(KT) + K))$ or $O(n^2T(log(nT) + n))$ for K relay problem or minimum relay problem as well.

Algorithm 2 Sensing and Relay Selection Algorithm

```
Input: potential user set $User$, visiting probability matrix
    $P$ of all users in $User$, the sensing task $q$ and the sink $d$.
Output: selected relay nodes $U(q, d)$.
$U(q, d) = \emptyset$
while $\mathcal{G}^{U(q,d)}$ is not connected do
    Choose the most active user and add it into $U(q, d)$
while $|U(q, d)| < K$ or $p_s(U(q, d), q, d) < \gamma$ (for $K$ sensing & relay problem or minimum sensing & relay problem, respectively) do
    for all $u_i \in User$ and $\notin U(q, d)$ do
        Calculate the improvement of $p_s(U(q, d), q, d)$ by adding $u_i$ in to $U(q, d)$ (Section 4)
    Select the user $u_i$ with the largest sensing and relay improvement and add it into $U(q, d)$
return $U(q, d)$
```

5 Experiments over D4D dataset

To evaluate the proposed algorithms, we conduct extensive simulations over a real-life cellular dataset, D4D dataset [32]. To make comparisons, we also implement two simple heuristics: *random selection* and *activity-based selection*. Random selection randomly chooses a user at each step until the algorithm ends, while activity-based selection greedily chooses a user who is most active (visiting most locations) at each step. In our simulations, we use the real delivery ratio and the number of selected users as the metrics of measurement.

5.1 D4D dataset and experiment settings

The D4D dataset [32] is a dataset of large-scale cellular users released by Orange S.A. for the Data for Development (D4D) Challenge in 2013, which is based on Call Detail Records of cell phone calls and SMS exchanges among Orange mobile users in Ivory Coast between December 1, 2011, and April 28, 2012. The number of the users is more than 50,000 for each week. Each record is associated with a cellular tower which provides the service and there are more that 1000 cellular towers in this dataset.

We select 20 most popular towers, i.e., the towers with largest number of associated records, to implement our simulations. Thus, $m = 20$. We choose a 100 candidate user set as *User*, i.e., $n = 100$. For each sensing task, we randomly generate its request at one of the towers and at one of the time slots from 1 to T. We assume that the whole sensing period T is one week and treat one hour as the smallest time unit. For each data collection task, we randomly select one location as the sink. If two users make phone calls associated with same tower at same time, we assume that they are close to each other and could transfer data between them via D2D links. For simplicity, we set the link reliability as 0.5, i.e., the successful transferring over a pair of nodes is 50% during their encountering.

To calculate the probability $p(u_i, l_j, t)$ of a particular user u_i visiting a location l_j at certain time t, we have to leverage the knowledge from the historical call traces. Here, we assume that for each user we have multiple rounds of call traces (e.g., X weeks), and each round of data denoted as D_i, $i = 1, \ldots, W$. Let $c_x(u_i, l_j, t)$ indicate whether u_i made one or more phone call at l_j and t in D_x (1 if it made, 0 otherwise). Then we simply estimate the visiting probability as follows,

$$p(u_i, l_j, t) = \frac{\sum_{x=1}^{W} c_x(u_i, l_j, t)}{W}.$$

Instead of this simple model, we can also consider Markov model [33] or Poisson process [20].

5.2 Experiments on D2D data collection

We first test our proposed algorithm (Algorithm 1) for D2D data collection of sensed data (K relay problem and minimum relay problem). For each data collection task, we randomly select a mobile user as the data source. For each set of experiments, we test 15 tasks and 100 rounds per tasks. The average performances over 1500 rounds are reported.

5.2.1 *Experiments on K relay problem*

In the first set of simulations, we consider the K relay problem. We vary the number of selected relay nodes from 10 to 20. Figure 3a shows the delivery rate achieved by each algorithm. It is clear that our proposed algorithm achieves the highest delivery ratio among the three algorithms when the number of selected relay nodes is the same. In addition, we can find that the delivery ratio of all the three algorithms increase as the number of selected relay nodes increase. This is obvious since more selected relay nodes provide more possible routes for the data to reach the sink node. Figure 3b shows the comparison between expected delivery ratio and real delivery ratio of our proposed algorithm. The real delivery ratio is always lower than the expected one since the estimation is based on the historical records. Although it is not 100% accurate,

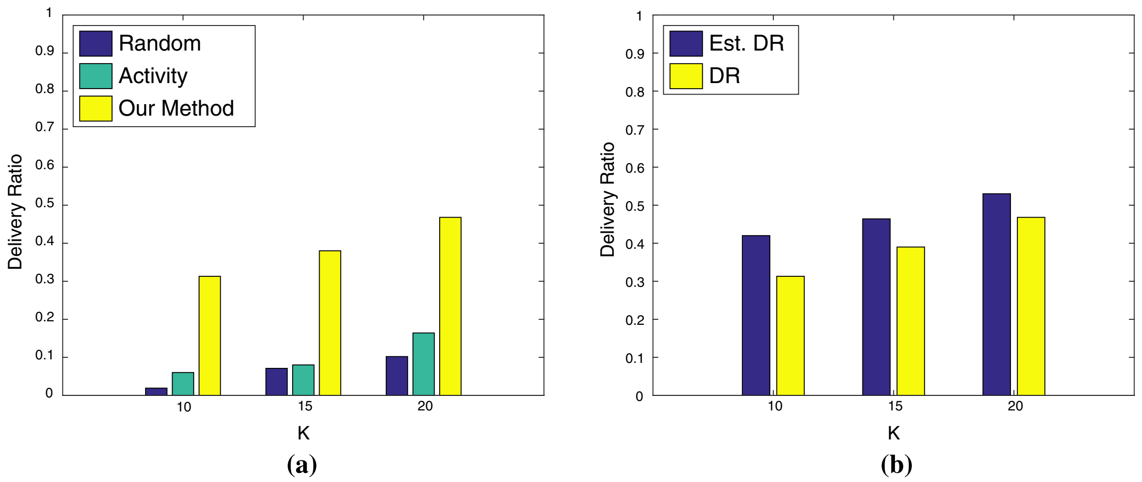

Fig. 3 Results for K relay problem where $K = 10$, 15 or 20. **a** Delivery ratio (DR), **b** estimated DR versus DR

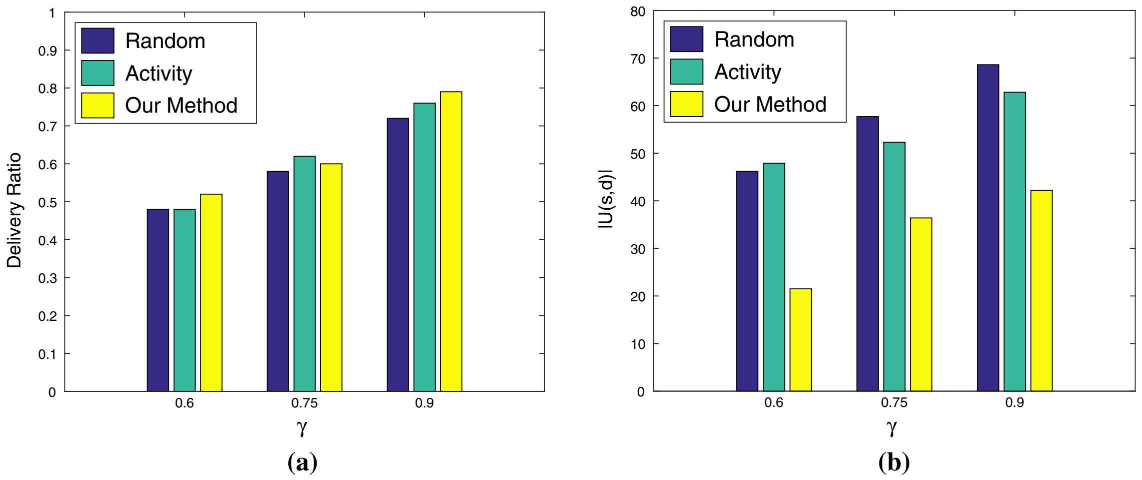

Fig. 4 Results for minimum relay problem where $\gamma = 0.6$, 0.75 or 0.9. **a** Delivery ratio, **b** number of relay nodes

the expected delivery ratio still provides us the guidance to pick the relay nodes.

5.2.2 Experiments on minimum relay problem

In this set of simulations, we evaluate the performance of algorithms over minimum relay problem. Here we vary the delivery ratio threshold γ from 0.6 to 0.9. Figure 4a shows that the delivery ratios of the three algorithms are similar to each others. Recall that all algorithms will continue to add new relay nodes until the estimated delivery probability reaches the threshold. Since the threshold is the same for the three algorithms, the overall delivery ratios are similar. However, in Fig. 4b, we can see that the number of selected relay nodes of our algorithm is much fewer than those of the other algorithms when achieving similar level of delivery ratios. This confirms that our proposed algorithm is more efficient than the other two simple heuristics.

5.3 Experiments on joint sensing and D2D relay

We also test our proposed algorithm (Algorithm 2) for joint sensing and relay problems. Here we consider two cases: (1) the optimization of participant selection is done per task based and (2) the optimization of participant selection is done for multiple tasks. For each set of experiments for the first case, we test 15 different tasks and 100 rounds per task. For each set of experiments for the second case, we test several task sets with various numbers of tasks and perform participant selection over 100 rounds per task set. For each sensing task, its interested location is generated randomly.

5.3.1 Experiments on single-task selection problem

For K sensing and relay selection, we vary the number of selected relay nodes H from 10 to 20. Figure 5a shows the

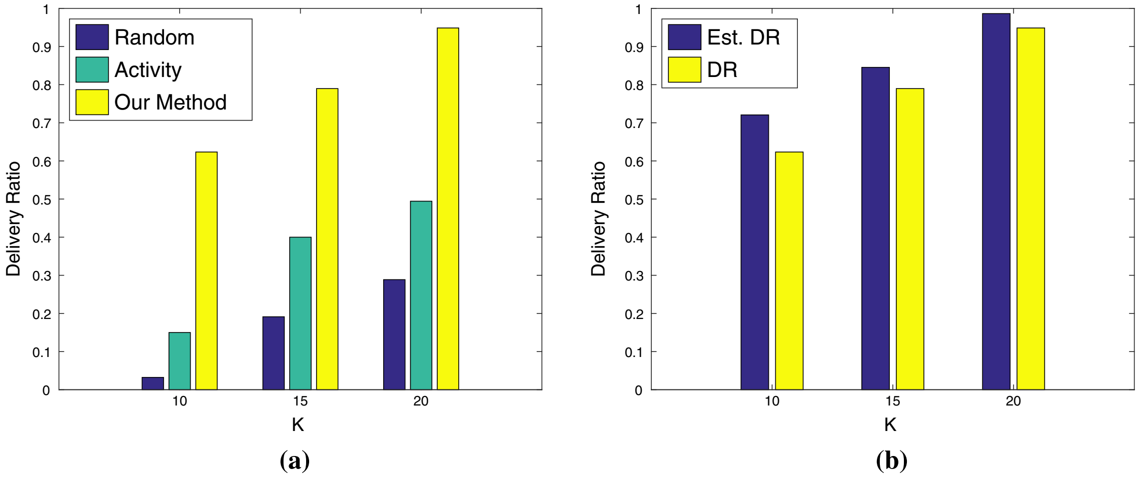

Fig. 5 Results for K sensing and relay problem with a single sensing task where $K = 10$, 15 or 20. **a** Delivery ratio (DR), **b** estimated DR versus DR

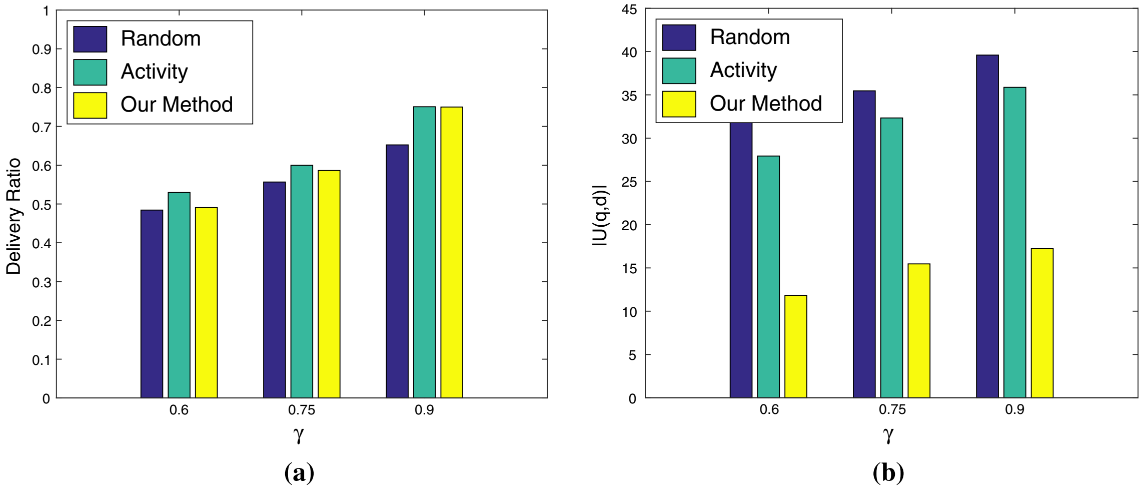

Fig. 6 Results for minimum sensing and relay problem with a single sensing task where $\gamma = 0.6$, 0.75 or 0.9. **a** Delivery ratio, **b** number of selected nodes

delivery rate achieved by each algorithm. Similar to the results of the experiments on K relay selection, our proposed algorithm achieves the highest delivery ratio among the three algorithms when the number of selected nodes is the same. The delivery ratio increases as the number of selected participants increases. Figure 5b also shows that there are still differences between the expected delivery ratio and real delivery ratio of our proposed algorithm since the estimation is based on the historical records. In the simulations of minimum sensing and relay selection, we vary the delivery ratio threshold γ from 0.6 to 0.9. Figure 6a shows that the three different algorithms achieve similar delivery ratio under the same delivery ratio threshold. However, our proposed algorithm selects fewer participants than those of the other two algorithms (Fig. 6b).

5.3.2 Experiments on multitask selection problem

We then test our proposed algorithm in multitask scenario where the selection of participants is optimized for the whole task set *Task*. Firstly, we test our algorithm over task sets with 5 sensing tasks. Figures 7 and 8 show the performance of K sensing and relay problem and minimum sensing and relay problem, respectively. The trends are similar to those in the single-task cases.

We also vary the number of sensing tasks in *Task* from 1 to 10 to evaluate the performance over different size of task set. Figure 9a shows the changing of delivery ratio when we fix the K to 20 and γ to 0.6 for K sensing and relay problem. We find that the delivery ratio drops when the number of tasks increases. Figure 9b shows the number of selected participants increases along the increasing of tasks

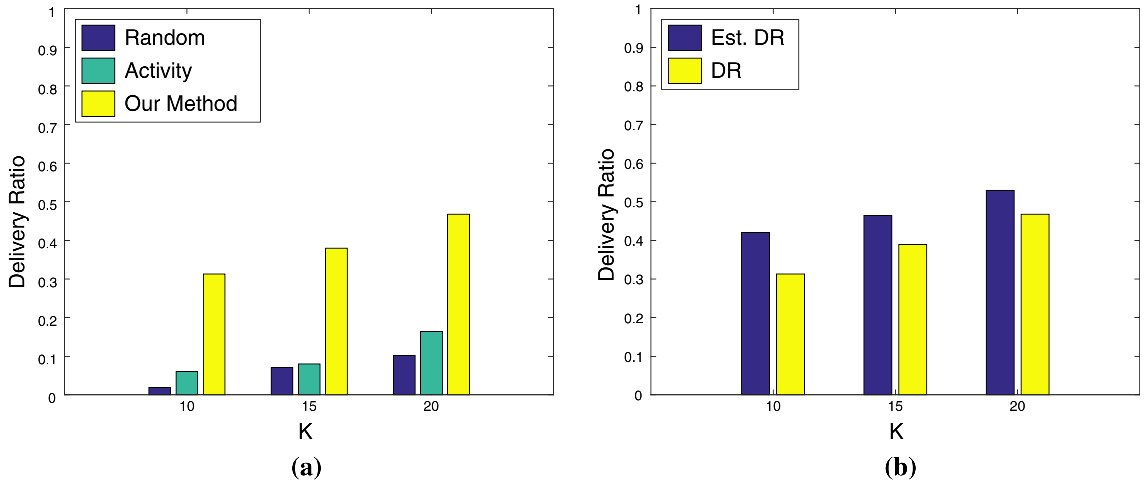

Fig. 7 Results for K sensing and relay problem with 5 sensing tasks where $K = 10$, 15 or 20. **a** Delivery ratio (DR), **b** estimated DR versus DR

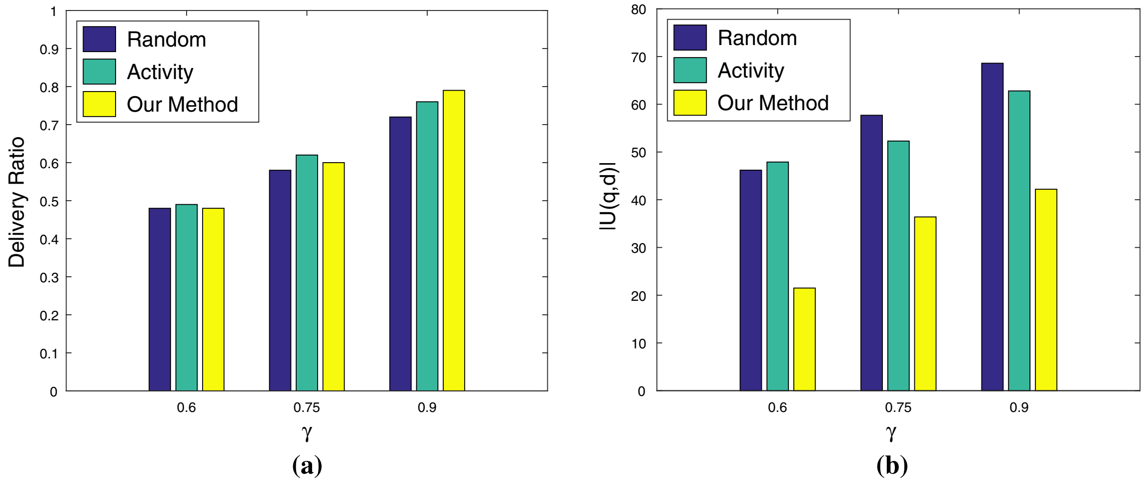

Fig. 8 Results for minimum sensing and relay problem with 5 sensing tasks where $\gamma = 0.6$, 0.75 or 0.9. **a** Delivery ratio, **b** number of selected nodes

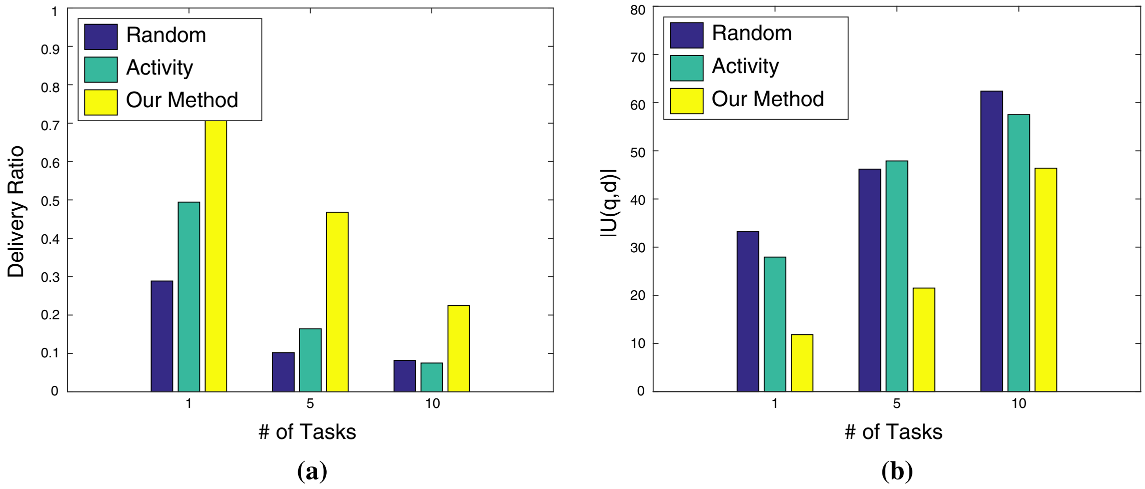

Fig. 9 Results for K sensing and relay problem ($K = 10$) and minimum sensing and relay problem where the number of sensing tasks $o = 1$, 5 or 10. **a** K sensing and relay, **b** minimum sensing and relay

when we fixed γ in minimum sensing and relay problem. Both of the trends above are reasonable since more tasks usually take more participants to perform and relay the data. One the one hand, it needs more people to perform the sensing tasks. On the other hand, it also needs more possible path to relay the sensing data to the destination.

6 Related work

Mobile sensing (especially mobile crowd sensing, MCS) which leverages large amount of mobile users has been widely used for different sensing applications [2]. To handle participant selection in large-scale MCS, different algorithms and systems [19–23] have been proposed recently. For example, Zhang et al. [21] study participant selection in a piggyback MCS for probabilistic coverage, which aims to select minimum number of participants to guarantee that the selected participants will make enough number of calls at certain percentage of the target locations over a fixed sensing period. Xiong et al. [19] have investigated how to assure the asymptotically full coverage over a 13 tower region with the minimum number of users. The task coverage is defined as whether the total number of calls is equal to or more than a threshold at these 13 towers in a fixed time period. Li et al. [22, 23] consider how to select minimum number of participants in an online manner while guarantee certain level of coverage of tasks. They assume that the sensing tasks can come at any time and may have various length and spacial/temporal requirement. Pournajaf et al. [33] also study task assignment in MCS aiming to assign moving participants with uncertain trajectories to static sensing tasks. There are also other studies on participant selection with consideration of energy efficiency [19, 34, 35], privacy [36] or user incentive [37–39]. However, all of these studies assume that the collected sensing data can be directly sent to MCS platform without much cost.

Recently, with the advances in mobile opportunistic networks or delay-tolerant networks [40–43] and D2D offloading [16–18], D2D data collection for mobile sensing becomes a new trend. Wang et al. [24] first consider leveraging the delay-tolerant mechanisms by offloading the data to Bluetooth/WiFi gateways or data-plan users. Their objective is to reduce the energy consumption and data cost of data-plan users. Karaliopoulos et al. [25] consider a joint user recruitment problem for both sensing and data collection, which is very similar to the one we study since the data collection is also done via D2D communications. However, the selection of users is formulated as a minimum cost set cover problem and single greedy heuristics are proposed to solve it. Since the solution space over all space–time paths is huge, their method may not be suitable for large-scale data collection. In this paper, instead we carefully select a few mobile participants as relay nodes to help with data propagation via D2D relays. By doing so, we limit the search space and make our algorithm more efficient. In addition, by leveraging multiple space–time paths for data collection, our method can achieve better delivery ratio too.

7 Conclusion

In this paper, we investigate the feasibility of collecting data packets from mobile devices in mobile sensing through device-to-device communications, by carefully selecting the subset of participant devices. We formulate the problem as several optimization problems (K relay selection, minimum relay selection, K sensing and relay selection, and minimum sensing and relay selection) and propose simple greedy algorithms to solve them. The proposed algorithms use historical information to obtain the estimated sensing, and delivery probability of a given participant set and greedily selects the participant based on this estimated probability. Our experiments over the real-life D4D mobile traces confirm the effectiveness of the proposed algorithms.

As for future works, we plan to continue our study on data collection in mobile sensing along the following directions: (1) hybrid data collection schemes which combine D2D and direct communications; (2) implementation of the proposed methods over a real testbed with mobile devices and experiments with real-world sensing tasks; and (3) modeling of energy consumption in mobile sensing and designing new energy efficient solutions.

Acknowledgements The work is partially supported by the US National Science Foundation under Grant Nos. CNS-1319915 and CNS-1343355, and the National Natural Science Foundation of China under Grant Nos. 61428203 and 61572347. The authors would like to thank Orange and the D4D challenge organizers for providing them the D4D datasets and allowing them to continue working on the datasets after the D4D challenge.

References

1. Cisco visual networking index: global mobile data traffic forecast update, 2015–2020 (February 3, 2016). http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.html
2. Guo B, Wang Z, Yu Z, Wang Y, Yen N, Huang R, Zhou X (2015) Mobile crowd sensing and computing: the review of an emerging human-powered sensing paradigm. ACM Comput Surv 48(1):7
3. Bengtsson L, Xin Lu, Thorson A, Garfield R, Schreeb JV (2011) Improved response to disasters and outbreaks by tracking population movements with mobile phone network data: A post-earthquake geospatial study in Haiti. PLoS Med 8(8):e1001083

4. Bo C, Jian X, Jung TJ, Han X-Y, Li Mao X, Wang Y (2016) Detecting driver's smartphone usage via non-intrusively sensing driving dynamics. IEEE Internet Things J (99):1. doi:10.1109/JIOT.2016.2552399
5. Zhou P, Zheng Y, Li M (2012) How long to wait? Predicting bus arrival time with mobile phone based participatory sensing. In: Proceedings of ACM MobiSys
6. Nawaz S, Efstratiou C, Mascolo C (2013) Parksense: a smartphone based sensing system for on-street parking. In: Proceedings of ACM Mobicom
7. Gao R, Zhao M, Ye T, Ye F, Wang Y, Bian K, Wang T, Li X (2014) Jigsaw: indoor floor plan reconstruction via mobile crowdsensing. In: Proceedings of ACM MobiCom
8. Bo C, Jung T, Mao X, Li X-Y, Wang Y (2016) SmartLoc: sensing landmarks silently for smartphone based metropolitan localization. EURASIP J Wirel Commun Netw 2016:e111
9. Rana RK, Chou CT, Kanhere SS, Bulusu N, Hu W (2010) Earphone: an end-to-end participatory urban noise mapping system. In: Proceedings of ACM/IEEE IPSN
10. Mun M, Reddy S, Shilton K, Yau N, Burke J, Estrin D, Hansen M, Howard E, West R, Boda P (2009) PEIR, the personal environmental impact report, as a platform for participatory sensing systems research. In: Proceedings of ACM MobiSys
11. Lathia N, Pejovic V, Rachuri KK, Mascolo C, Musolesi M, Rentfrow PJ (2013) Smartphones for large-scale behavior change interventions. IEEE Pervasive Comput 12(3):66–73
12. Noulas A, Scellato S, Lambiotte R, Pontil M, Mascolo C (2012) A tale of many cities: universal patterns in human urban mobility. PLoS ONE 7(5):e37027
13. Balasubramanian A, Mahajan R, Venkataramani A (2010) Augmenting mobile 3G using WiFi. In: ACM MobiSys 2010
14. Dimatteo S, Hui P, Han B, Li VOK (2011) Cellular traffic offloading through WiFi networks. In: Proceedings of IEEE MASS
15. Chandrasekhar V, Andrews JG, Gatherer A (2008) Femtocell networks: a survey. IEEE Commun Mag 46(9):59–67
16. Han B, Hui P, Kumar VSA, Marathe MV, Shao J, Srinivasan A (2012) Mobile data offloading through opportunistic communications and social participation. IEEE Trans Mobile Comput 11(5):821–834
17. Li Y, Qian M, Jin D, Hui P, Wang Z, Chen S (2014) Multiple mobile data offloading through disruption tolerant networks. IEEE Trans Mobile Comput 13(7):1579–1596
18. Zhu Y, Zhang C, Wang Y (2013) Mobile data delivery through opportunistic communications among cellular users: a case study for the D4D challenge. In: Proceedings of NetMob
19. Xiong H, Zhang D, Wang L, Chaouchi H (2015) EMC^3: energy-efficient data transfer in mobile crowdsensing under full coverage constraint. IEEE Trans Mobile Comput 14(7):1355–1368
20. Xiong H, Zhang D, Chen G, Wang L, Gauthier V (2015) Crowdtasker: maximizing coverage quality in piggyback crowdsensing under budget constraint. In: Proceedings of IEEE Percom
21. Zhang D, Xiong H, Wang L, Chen G (2014) Crowdrecruiter: selecting participants for piggyback crowdsensing under probabilistic coverage constraint. In: Proceedings of ACM UbiComp
22. Li H, Li T, Wang Y (2015) Dynamic participant recruitment of mobile crowd sensing for heterogeneous sensing tasks. In: Proceedings of IEEE MASS
23. Li H, Li T, Li F, Wang W, Wang Y (2016) Enhancing participant selection through caching in mobile crowd sensing. In: Proceedings of ACM/IEEE IWQoS
24. Wang L, Zhang D, Xiong H (2013) Effsense: energy-efficient and cost-effective data uploading in mobile crowdsensing . In: Proceedings of ACM UbiComp
25. Karaliopoulos M, Telelis O, Koutsopoulos I (2015) User recruitment for mobile crowdsensing over opportunistic networks. In: Proceedings of IEEE INFOCOM
26. Li H, Li T, Shi X, Wang Y (2016) Data collection through device-to-device communications for mobile big data sensing. In: Proceedings of 1st workshop of mission-critical big data analytics (MCBDA 2016)
27. Vahdat A, Becker D (2000) Epidemic routing for partially connected ad hoc networks. Technical Report CS-200006, Duke University, Technical Report
28. Merugu S, Ammar M, Zegura E (2004) Routing in space and time in networks with predictable mobility. Technical Report GIT-CC-04-07
29. Huang M, Chen S, Zhu Y, Wang Y (2013) Topology control for time-evolving and predictable delay-tolerant networks. IEEE Trans Comput 62(11):2308–2321
30. Li F, Chen S, Huang M, Yin Z, Zhang C, Wang Y (2015) Reliable topology design in time-evolving delay-tolerant networks with unreliable links. IEEE Trans Mobile Comput 14(6):1301–1314
31. Agrawal A, Barlow RE (1984) A survey of network reliability and domination theory. Oper Res 32:478–492
32. Blondel VD, Esch M, Chan C, Clerot F, Deville P, Huens E, Morlot F, Smoreda Z, Ziemlicki C (2013) Data for development: the D4D challenge on mobile phone data. arXiv.1210.0137v2
33. Pournajaf L, Xiong L, Sunderam VS (2014) Dynamic data driven crowd sensing task assignment. In: Proceedings of ICCS
34. Zhao D, Ma H, Liu L (2014) Energy-efficient opportunistic coverage for people-centric urban sensing. Wirel Netw 20(6):461–1476
35. Li F, Tian C, Li T, Wang Y (2016) Energy efficient social routing framework for mobile social sensing networks. Tsinghua Sci Technol 21(4):363–373
36. Jin H, Su L, Ding B, Nahrstedt K, Borisov N (2016) Enabling privacy-preserving incentives for mobile crowd sensing systems. In: Proceedings of IEEE ICDCS
37. Yang D, Xue G, Fang X, Tang J (2012) Crowdsourcing to smartphones: incentive mechanism design for mobile phone sensing. In: Proceedings of ACM Mobicom
38. Liu Y, Li F, Wang Y (2016) Incentives for delay-constrained data query and feedback in mobile opportunistic crowdsensing. Sensors 16(7):1138. doi:10.3390/s16071138
39. Feng Z, Zhu Y, Zhang Q, Ni LM (2014) Vasilakos AV TRAC: truthful auction for location-aware collaborative sensing in mobile crowdsourcing In: Proceedings of INFOCOM
40. Zhu Y, Xu B, Shi X, Wang Y (2013) A survey of social-based routing in delay tolerant networks: positive and negative social effects. IEEE Commun Surv Tutor 15(1):387–401
41. Zhu Y, Zhang C, Li F, Wang Y (2015) Geo-social: routing with location and social metrics in mobile opportunistic networks. In: IEEE ICC
42. Liu Y, Bashar AMAE, Li F, Wang Y, Liu K (2016) Multi-copy data dissemination with probabilistic delay constraint in mobile opportunistic device-to-device networks. In: Proceedings of 17th IEEE international symposium on a world of wireless, mobile and multimedia networks (WOWMOM 2016)
43. Li Y, Wu H, Xia Y, Wang Y, Li F, Yang P (2016) Optimal online data dissemination for resource constrained mobile opportunistic networks. IEEE Trans Veh Tech (99):1. doi:10.1109/TVT.2016.2616034