Social based throwbox placement schemes for large-scale mobile social delay tolerant networks

Ying Zhu, Chao Zhang, Xufei Mao, Yu Wang

Department of Computer Science, University of North Carolina at Charlotte, Charlotte, NC 28223, USA
School of Software, Tsinghua University, Beijing 100084, China

Abstract

Mobile phone sensing is a new paradigm which takes advantage of smart phones to collect and analyze data at large scale but with a low cost. Supporting pervasive communications among mobile devices in such a large-scale mobile social network becomes a key challenge for this new mobile sensing system. One possible solution is allowing packet delivery among mobile devices via opportunistic communications during intermittent contacts. However, the lack of rich contact opportunities still causes poor delivery ratio and long delay, especially for large-scale networks. Deployment of additional stationary throwboxes can create a greater number of contact opportunities, thus improve the performance of routing. However, the locations of deployed throwboxes are critical to such improvement. In this paper, we investigate where to deploy throwboxes in a large-scale throwbox-assisted mobile social DTN. By leveraging the social properties discovered from real-life tracing data, we propose a set of social-based throwbox placement algorithms which smartly pick the location of each throwbox. Extensive simulations are conducted with a real-life wireless tracing dataset and a wide range of existing DTN routing methods. The results confirm the efficiency of the proposed methods.

1. Introduction

With the increasing popularity of smart phones equipped with a set of versatile sensors, it becomes possible to perform ubiquitous sensing via a mobile sensing network formed by a large amount of mobile phones [1]. Such a new paradigm of mobile sensing, where each mobile device is held by an individual, can provide abundant data about individuals, human society and environments. Some examples of mobile sensing applications include traffic information [2,3], environmental monitoring [4,5], human mobility modeling [6,7], important place extraction [8,9], urban sensing and planning [10,11], parking information [12], sociology [13–15], ecology and epidemiology [16].

One of the challenges for large-scale mobile sensing or mobile networking is how to efficiently transfer data among mobile devices. For smart phones, we are currently facing the challenge of mobile data explosion. Based on the most recent Cisco’s report [17], mobile data traffic grew 70% in 2012 and reached 885 petabytes per month at the end of 2012, which was nearly 12 times the size of the entire Internet in 2000 (75 petabytes per month). Cisco also forecasts that mobile data traffic will surpass 10 exabytes per month in 2017. In addition, the recent advance in machine-to-machine (M2M) communications may potentially add billions of devices into mobile Internet. By the end of 2013, the number of mobile-connected devices will exceed the number of people on earth [17]. However, the current cellular networks do not have enough capacity to support all of the fast-growing mobile data from these mobile devices and the new applications of mobile sensing [18–20]. "The technology evolution of radio access networks is limited by the laws of physics, and significant growth in radio frequency efficiency can no longer be expected. Long-Term Evolution (LTE) radio access is reaching the limits of Shannon’s law, the spectrum available for mobile data applications is limited", regarding to [18]. Therefore, looking for new ways to increase the overall capacity and satisfy fast-growing mobile data is emerging as a new topic for cellular industry.

To avoid overloading the cellular networks, one of the possible offloading solutions is using opportunistic communications to deliver data packets [19–21] via a mobile Delay Tolerant Network (DTN). The major advantage of this solution is low cost and easy to deploy even in a large-scale network. However, intermittent connectivity in DTNs results in the lack of instantaneous end-to-end paths, large transmission delay and unstable network topology. To overcome these challenges, many DTN routing algorithms [22–33] have been proposed by relying on intermittent contacts.
between mobile nodes to deliver packets. However, the lack of rich contact opportunities in many DTNs (especially those with sparse deployments) still causes poor delivery ratio and long delay of DTN routing [34,35]. For example, as shown in [35], the delivery ratios of existing DTN routings in a large-scale mobile social network are only around 30–40%. Therefore, there are still spaces to further improve the delivery ratio of opportunistic routing in large-scale mobile networks.

One way to improve mobile DTN performance is to deploy additional stationary nodes, called *ThrowBoxes* (TBs), to create a greater number of contact opportunities [34,36,38–41]. Throwboxes are usually small, battery-powered, and inexpensive devices equipped with wireless interfaces and storage. They are stationary and can relay data between mobile nodes in a *store-and-forward* way.

![Fig. 1. Illustration of throwbox-assisted mobile social DTNs: (a) when a mobile node \( v_i \) passes by a throwbox \( b_q \), it can leave a copy of the message to \( b_q \); (b) later, throwbox \( b_q \) can then forward the message to another passing by mobile node \( v_j \).](image)

![Fig. 2. Visualization of D4D dataset: the traffic distribution among all regions are shown in color where darker color indicates heavier traffic loads; the small blue rectangle around Abidjan shows the limited region with high traffic we select for our simulations; and a zoomed view of the detailed tower distribution within the selected region is also provided.](image)

![Fig. 3. Illustration of social graphs in mobile social DTNs: (a) the user-location bipartite graph \( G_{ul} \) shows every user’s top 2 locations; (b) the location-location bipartite graph \( G_{ll} \) shows every location’s top 2 locations; here we assume that the total visiting frequencies or durations of \( b_2 \) and \( b_m \) to \( b_1 \) are larger than that of \( b_n \); (c) the social graph among users \( G_u \); (d) the social graph among locations \( G_l \); (e) the whole social graph among users and locations \( G \). In all examples, we use top 2 locations instead of top 10 for simple illustration.](image)
shown in Fig. 1, when two mobile nodes pass by the same location at different time, the throwbox can act as a relay, creating a new contact opportunity. Simulations and real deployments [36,38–41] have demonstrated that introducing small amount of throwboxes can indeed improve the routing performances and overall throughputs.

For a large-scale mobile social DTN with a huge amount of mobile users, it is impossible to deploy a huge amount of throwboxes due to budget constraint. With a limited number of deployed throwboxes, their locations become critical to the performance. In this paper, we study a key design problem in throwbox-assisted mobile social DTNs: throwbox placement problem. Given a set of potential locations for throwboxes and a fixed number of throwboxes, we need to find where to deploy these throwboxes to maximize the network performance. General relay placement in static wireless networks [42,43] has been well studied. However, in mobile DTNs, the network topology evolves over time due to node mobility. This brings new challenges into the problem and makes existing relay placement algorithms useless in DTNs. To our best knowledge, there is not much study on throwbox deployment in DTN except for [34], which addresses a joint throwbox deployment and routing optimization problem. However, their focus is only on the long term average capacity. In this paper, instead we study how to deploy throwboxes in a large-scale mobile DTN so that the performance of DTN routing can be improved. In mobile DTNs or mobile sensing applications, mobile devices are usually carried by people, whose behaviors are better described by their social characteristics. Therefore, we propose several social-based throwbox placement schemes in which the knowledge of social characteristics of mobile users and candidate locations are used to smartly pick the deployment locations. Simulation results based on real-life tracing data of large-scale mobile users demonstrate the efficiency of the proposed methods.

The rest of this paper is organized as follows. Section 2 introduces our models and formally defines the throwbox placement problem within a mobile social DTN. Section 3 describes the Data for Development (D4D) dataset [45] we used. A set of greedy algorithms are then presented in Section 4 and simulation results of these methods are reported in Section 5. Finally, Section 6 provides a brief review on throwbox-assisted DTNs and Section 7 concludes this paper. A preliminary version of this paper appeared in [44].

2. Throwbox placement problem and network models

We first describe the throwbox placement problem which we study and the network models which we assume within this paper.

2.1. Throwbox placement problem

Throwboxes can be used in variety of scenarios. As shown in Fig. 1, stationary throwboxes can relay data between mobile nodes via a “store-and-forward” fashion. Assume that $V = \{v_1, \ldots, v_n\}$ and $B = \{b_1, \ldots, b_m\}$ be the set of all individual mobile users (wireless devices) and the set of all potential locations of throwboxes in the network, respectively. In this paper, we use the locations of cellular towers as the potential candidate locations of throwboxes for mobile users. It is mainly due to the following three reasons: (1) based on our analysis from the large-scale cellular data (D4D dataset [45]) these locations are often the hot spots of mobile nodes; (2) all cellular towers already have existing infrastructure and
equipments for cellular systems, thus it is convenient to install and implement throwboxes there; (3) with the cellular tower access record available in this D4D cellular dataset, it is easy for us to get the location and social characteristics of these cellular towers. However, our proposed throwbox placement algorithms work for any other candidate location sets, as long as the number of candidate locations is finite and certain user’ access records are available.

With the help from throwboxes there will be more forwarding opportunities among mobile devices, thus they increase the chances of final delivery. However, the deployment of throwboxes has certain cost (either the hardware cost or the deployment cost) and the network operator may have a fixed budget only allowing a limited number of deployed throwboxes. Therefore, a key problem is where to put these throwboxes to maximize the network performance. We now formally define the throwbox placement problem as follows:

**Definition 1.** Given a time-evolving mobile social DTN with \( n \) mobile users and \( m \) potential locations of throwboxes \( B \), the aim of **throwbox placement problem** is to find \( k \) locations to place throwboxes, such that the routing performance is maximized. Here, \( k \ll m \) is a small constant.

### 2.2. Communication models of throwboxes

We then introduce three different communication models which define how packets can be transferred between normal mobile users and throwboxes during the DTN routing process.

**Model I:** In this model, we treat any throwbox exactly the same as a normal mobile user. In other words, during an encounter between a throwbox and a mobile user, the packet can be transferred in both directions (throwbox-to-user or user-to-throwbox) based on the forwarding decision made by underlying routing algorithm. The routing algorithm does not distinguish throwboxes from normal mobile users. The total number of copies of a packet in the network (hold by either throwboxes or mobile users) is limited up to \( N_{\text{max}} \). If the total number of copies for a message already reaches \( N_{\text{max}} \), the current node will delete its copy after forwarding it to the encounter.

**Model II:** In this model, we treat throwboxes and mobile users differently. The maximum number of copies \( N_{\text{max}} \) is only applied to copies of a packet hold by mobile users. There is no constraint on the number of copies on throwboxes. When a mobile user encounters a throwbox, it always gives a copy to the throwbox while keeps a copy on itself. The throwbox will keep the copy permanently but it cannot forward its copies to mobile users except for forwarding to the destination node.

**Model III:** Similar to Model II, there is no constraint on the number of copies on throwboxes. The only difference is that throwboxes are now allowed to forward a copy of the packet to encountered mobile users. If the total number of copies on mobile users is less than \( N_{\text{max}} \) and the encountered mobile user has a “better” metric than this throwbox, it will give a copy of the packet to the mobile user.

Obviously, from Model I to Model III, more forwarding opportunities can be utilized by the routing algorithms. We will compare the performance of classical DTN routing over these models in Section 5.1. Notice that the maximum number of copies \( N_{\text{max}} \)
can be implemented using either tokens or a centralized counter as in [25].

3. D4D datasets

To simulate the large scale mobile social DTNs (especially for mobile phone sensing), we use a real life wireless tracing data from the cellular operator Orange for the Data for Development (D4D) challenge [46]. The reason we pick the D4D dataset is that it is the only mobile networking tracing dataset available to us which has a large-scale and diverse set of mobile users. The released D4D datasets [45] are based on anonymized Call Detail Records (CDR) of phone calls and SMS exchanges between 50,000 Orange mobile users in Ivory Coast between December 1, 2011 and April 28, 2012 (150 days). We use the dataset of individual trajectories with high spatial resolution (SET2 in D4D datasets), which contains 10 groups of the access records of antenna (cellular tower) of each mobile user. Each group of records are collected over a two-week period. The time ranges of these 10 groups of records are sequentially and add up equal to the whole duration of D4D data collection period. But unfortunately, in each group of records, the user IDs were renumbered and anonymized, which makes impossible to merge them together. So all of our experiments are preformed within a two-week period. We use the sequences of visited cellular towers of all users to generate both contact encounters among mobile users and location/social profiles of each mobile user and location (i.e., cellular tower). Note that though we use the D4D dataset to evaluate our proposed methods, the proposed throwbox placement solutions also work for any other datasets and DTN networks (such as campus-based DTNs and vehicular networks), as long as the participants have regular social or spacial characteristics among them.

Since D4D datasets do not have direct encounter information between phones via short range communications (such as Bluetooth or WiFi), to support opportunistic DTN communications we assume that two phones can direct communicate with each other if they share the same cellular tower at a particular time. Though this assumption may not be true in reality, it gives us an approximated environment for opportunistic communications in such a large scale network. We use all access records for every two-week period, but only perform the opportunistic communications within each two-week period. For one two-week period, there are already huge number of users and encounters. For example, for the first two-week period, there are 46,254 active mobile users, 1097 cellular towers, and 6,787,594 encounters between users in total. Notice that the number of users in our generated encounter database is less than the number of users in original SET2 (such as 46,254 < 50,000). This shows that there are some mobile users who do not share any cellular towers with other users in these two weeks. In a previous study [35], we have shown that the smaller size of user set and encounter database could accelerate the execution time of DTN simulations while conclusions from simulation results are still consistent with those using the whole

Fig. 6. Performance comparison on different communication models with throwbox placement Method C in throwbox-assisted DTNs.
Fig. 7. Performance comparison on different communication models with throwbox placement Method D in throwbox-assisted DTNs.

Fig. 8. Performance comparison on different communication models with throwbox placement Method E in throwbox-assisted DTNs.
user set. Therefore, in this paper, we choose a subset of users from the whole user set by requiring that the users must be from the first 15,000 users in our encounter database and the physical locations of encounters must be within a small region. Fig. 2 shows the number of calls during a period of two weeks (where darker color indicates heavier traffic loads). Clearly, the traffic load distribution within Ivory Coast is unbalanced. Therefore, we choose a small region (a blue rectangle region in Fig. 2, whose longitude and latitude range from $[-8.49, -2.69]$ and $[4.41, 10.47]$, respectively) with the heaviest traffic load. This region is around Abidjan, the economic and former official capital of Ivory Coast and the largest city in the nation. A zoomed view of this region in Fig. 2 shows that it holds a large number of cellular towers. For the first two-week period, this selected dataset has 6318 active mobile users, 496 cellular towers, and 327,717 encounters between users.

4. Social based throwbox placement schemes

In this section, we propose a set of throwbox placement algorithms which consider social properties of potential locations of throwboxes or/and users' social properties. Since each mobile device is held by an individual, the underlying social tie among individuals and locations will play an important role in our placement algorithms.

4.1. Social characteristics of locations/users

We first introduce different social properties that we obtain from the mobile tracing data. For each mobile user $v_i$, we analyze its access records to cellular towers and construct the location profile of this user as $L(v_i) = \{b_1, b_2, \ldots, b_m\}$, which is an ordered list of locations visited by user $v_i$ based on their visiting frequencies (or durations). Notice that here we use the same set of towers as the locations to build location profile and the potential candidate locations of throwboxes, but they could be different. From the location profiles of all users, we then can construct a serial of social graphs among mobile users and locations.

- First, for each user $v_i$, we can have top 10 places (in term of visiting frequencies or durations $L(v_i)$) which $v_i$ visits most. Let $T(v_i)$ be the set of such places. We then can define a user-location graph $G_{ul}$ (a bipartite graph) where there is an edge between a user $v_i$ and a place $b_j$ if and only if $b_j \in T(v_i)$. Fig. 3(a) shows an example.
- For each location $b_j$, we can also define its top 10 places (in term of visiting frequencies or durations) whose users have highest possibility to visit $b_j$ next, denoted by $T(b_j)$. When there is a record user $v_i$ visits location $b_j$ with certain duration and his previous location is $b_k$, we add such a visit (with the same duration) of $b_j$ by $b_k$. Note that high frequency or duration among two locations indicates high possibility and frequency the users will travel from one location to the other. By this way, we defines a two-hop relationship between two places $b_i$ and $b_j$ via a mobile user $v$. A location–location bipartite graph $G_{bg}$ can be defined where there is an edge between $b_i$ to $b_j$ if and only if $b_j \in T(b_i)$. Fig. 3(b) shows the example.
A social graph among users can be defined as $G_u$, where there is an edge between two users $v_i$ and $v_j$ if they share a common list of top 10 places, which represents high possibility of direct or indirect contacts between two users. Here an indirect contact may be via a static throwbox at their common place. Fig. 3(d) shows the example.

A social graph among locations can be defined as $G_b$, where there is an edge between locations $b_i$ and $b_j$ if they share a common top 10 places. Fig. 3(c) shows the example.

A social graph among both users and locations can be defined as $G$, which combines both $G_u$ and $G_b$ using the edges from $G_{ub}$. In other words, connect a user $v_i$ and a place $b_j$ if and only if $b_j \in T(v_i)$. Fig. 3(e) shows the combined graph.

Based on these social graphs, we can easily define the following social metrics for mobile nodes or locations.

- **Degree/betweenness centrality of a mobile user**: The degree and betweenness centralities of a mobile user $v_i$ can be computed from the social graph of users $G_u$, and denoted as $D(v_i)$ and $B(v_i)$ respectively. Here $D(v_i)$ is equal to the total number of neighbors of $v_i$ in $G_u$ while $B(v_i)$ is measures the number of shortest paths passing via $v_i$ in $G_u$. A node with a high degree centrality is a popular node with a large number of possible direct or indirect "contacts", and a node with high betweenness centrality can control or facilitate many connections between other nodes.

- **Degree centrality of a location**: The degree centrality of a location $b_i$ can be computed from $G_b$ and denoted as $D(b_i)$. It basically shows how many mobile users have $b_i$ as one of their top 10 locations. Larger $D(b_i)$ shows $b_i$ is more popular among mobile users.

- **Betweenness centrality of a location**: The betweenness centrality of a location $b_i$ can be computed from $G$ and denoted as $B(b_i)$. It basically shows how important of the location of $b_i$ for the entire social graph $G$. A location with larger $B(b_i)$ can act as a bridge node during message exchanges.

4.2. Throwbox placement algorithms

To choose the appropriate throwbox locations to maximize the benefits of throwboxes, especially when there is only a small amount of throwboxes available as defined in the throwbox placement problem, we propose several social-based throwbox placement algorithms. Our algorithms consider degree/betweenness centrality of both mobile users and candidate throwbox locations. We believe that putting throwboxes at the popular locations (such as top 10 locations or locations with high centrality) of important mobile users (active users with high popularity or betweenness centrality) may have large contributions on routing performance. Our detailed social-based throwbox placement algorithms are presented as follows.

- **Method A – Most Popular Locations**: We simply choose the $k$ locations with highest degree centrality (i.e., $D(b_i)$) to deploy throwboxes. We use $G_{ub}$ to calculate $D(b_i)$. 

---

**Fig. 10.** Performance results with different number of throwboxes deployed by one of the proposed throwbox placement methods (Method A).
Fig. 11. Performance comparison of different throwbox placement schemes over a well connected network.

Fig. 12. Performance comparison of different throwbox placement schemes over a not well connected network.
Method B – Most Important Locations: We consider the betweenness centrality of locations and choose the k locations with highest betweenness centrality (i.e., $B(b_i)$) to deploy throwboxes. To calculate $B(b_i)$, we use the entire social graph $G$, which includes all mobile users and candidate locations.

Method C – Top Locations of Important Users: We consider the betweenness centrality among users $B(v_i)$ calculated from $G_u$. We choose all top 10 locations of the mobile users with highest betweenness centrality. Start from those users with the highest $B(v_i)$, then add those users with the second highest $B(v_i)$, and keep going until we have enough k deployment locations.

Method D – Weighted Popular Locations with User Degree Centrality: We consider the degree centralities of both locations and mobile users. First, $D(v_i)$, calculated from $G_u$, is normalized to a value $e \in [0, 1]$. For each location $b_j$, we then define a new metric base on $G_{ub}$ to describe the popularity of a location among mobile users weighted by users’ degree centrality:

$$\sum_{any \ v_p \ s.t. \ b_j \in T(v_p)} D(v_p).$$

We choose the k locations with highest metric values to deploy throwboxes. Intuitively, if a location has frequent visitors with high degree centrality, it will have a high metric value, thus have a high probability to be picked. Note that this method regresses to Method A if we let all $D(v_i) = 1$ (i.e., $\sum_{any \ v_p \ s.t. \ b_j \in T(v_p)} D(v_p) = D(b_j)$ where only the number of mobile users who have the location as one of their top 10 locations is considered).

Method E – Weighted Popular Locations with User Betweenness Centrality: This method is similar to Method D except that we consider users’ betweenness centrality instead of their degree centrality. Again $B(v_i)$, calculated from $G_u$, is normalized and the metric is defined base on $G_{ub}$ as

$$\sum_{any \ v_p \ s.t. \ b_j \in T(v_p)} B(v_p).$$

We choose the k locations with highest metric values to deploy throwboxes.

5. Simulations results

To test our proposed throwbox placement schemes in the large-scale D4D mobile network, we implement four representative DTN routing algorithms.

- Fresh: [27] the message is only forwarded from the current node $v_i$ to the encountered node $v_j$ if $v_j$ has met the destination more recently than $v_i$ does.
- Destination Frequency: [24] the message is only forwarded from $v_i$ to $v_j$ if $v_j$ has met the destination more often than $v_i$ does.
- Centrality-Based: the message is only forwarded from $v_i$ to $v_j$ if $v_j$ has higher centrality than $v_i$ does. Here, we use the degree centrality as the centrality metric, similar to many existing social-based methods [23,26]. Note that this degree centrality is not computed from $G_u$ but from traditional contact graph generated from direct contact history.
• Location-Based: the message is only forwarded from $v_i$ to $v_j$ if $v_j$ has more similar location profile with the destination than $v_i$ does. The similarity is defined as the distance between their location profiles. Similar ideas have been used in [28,29].

We conduct extensive simulations of proposed throwbox placement schemes with these routing algorithms on the D4D data set described in Section 3 and measure the following four metrics:

- **Average Successful Delivery Ratio**, the average percentage of successfully delivered messages from the sources to the destinations.
- **Average Hop Count**: the average number of hops during each successful delivery from the sources to the destinations.
- **Average Number of Forwarding**: the average number of messages forwarding in the network during the whole period.
- **Average Delay**, the average time duration of successfully delivered messages from the source node to the destination nodes.

For all experiments, we perform 5000 random routing tasks among the selected participators in each two-week period. For each routing task, we randomly choose the source node and the destination node, thus the task aims to deliver the message from the source to the destination node. The start time of each task is also randomly chosen from the two-week period. All results reported here are the average over these tasks and over different two-week periods. We assume a perfect MAC layer and reliable message transmission between any two encountering nodes. For all routing algorithms, we use their multiple copy versions where the number of duplicates of a message on mobile users is limited by $N_{\text{max}}$, whose default value is set at 10.

### 5.1. Communication models of throwboxes

We first evaluate the three different communication models introduced in Section 2.2 for throwbox-assisted DTNs and compare their performances with the case without any throwbox deployed. For each model, we pick 100 active mobile users and use Methods A–E to select 20 throwboxes from 268 candidate locations (the whole set of top 10 locations of these 100 mobile users). Thus, there are total 120 nodes participated in the opportunistic communications. Figs. 4–8 illustrate the detailed results. It is clear that with deployed throwboxes all routing methods can achieve significantly higher average successful delivery ratio (than those without throwboxes). Model III achieves the best performances (with the highest successful delivery ratio and the smallest delay) for all Methods but uses largest number of forwarding as we expect. Thus, there is always a tradeoff between routing performance and communication overhead. In the remaining simulations, we fix Model III as the communication model.

### 5.2. Number of throwboxes

We also study the effect of the number of deployed throwboxes with both random deployment and proposed social-based deployment. For both sets of simulations, we select 100 active mobile users and vary the number of deployed throwboxes from 5 to 70. Figs. 9 and 10 illustrate the simulations with random deployed...
throwboxes and throwboxes deployed by our proposed Method A, respectively. For both scenarios, the successful delivery ratio increases and the average delay decreases as the number of deployed throwboxes increases. Thus, more throwboxes usually can further improve the routing performances. In addition, compared with random deployment, the social-based approach can achieve better performance especially when the number of deployed throwboxes is small, i.e., the successful delivery ratio of Method A increases faster than the one with random deployment. When the number of deployed throwboxes is large enough, there is no significant improvement over random deployment.

5.3. Throwbox placement schemes

We then compare the proposed five throwbox placement methods and random deployment by fixing the number of deployed throwboxes to 5. Here, we use a very small number of available throwbox because in this scenario it is more meaningful to choose suitable location smartly. Fig. 11 shows the results. All of our social-based methods have similar successful delivery ratios which are higher than that of random deployment. Among the five methods, Method A and Method D have the slightly better delivery ratios than most of other routing methods. Method B, which considers the locations’ betweenness centrality, has significantly less number of forwarding than other methods. This may be due to that putting throwboxes at “bridge” locations (locations with high betweenness centrality in the social graph) reduces unnecessary forwarding among throwboxes and mobile users. But obviously it also hurts the delivery ratio.

Besides the set of experiments with 100 selected active users (well connected in social graph of users since they are very active in term of contacts), we also perform a set of experiments with carefully selected 100 users which form two separate components in the social graph (each has 50 users). The criterion for creating the separated components is as follows: first we choose the node with the highest centrality and put it into the first component; then we choose the node with highest centrality from the nodes with no connection to the first component and put it into the second component; and then we choose the node with highest centrality from nodes with no connection to the second component and put it into the first component. We repeat this process until we have 100 nodes inside these two components. Results in this scenario are given in Fig. 12. Clearly, the performances are much poorer than those in the previous simulations since the connectivity between two components are very loose. Now Method B and Method E, which consider betweenness centrality, have better successful delivery ratio than others. This is mainly because the locations selected by these two methods can act as “bridge” nodes to connect the separate components. Higher delivery ratios also lead to longer delays since they deliver more messages, some of them takes longer delays.

5.4. Number of duplicates

Finally, the number of duplicates \(N_{\text{max}}\) is also a controllable parameter for multi-copy DTN routing. Therefore, we also conduct two sets of simulations to observe the effect of \(N_{\text{max}}\) by varying its value from 5 to 50. Other simulation settings are the same as

![Fig. 15. Performance comparison of throwbox selection methods as the number of duplicates increases with Destination Frequency.](image-url)
previous experiments. Figs. 13–16 show the results for the 100 selected active users, while Figs. 17–20 show the results for the case of two separate components. In each set of the simulations, we either fix the throwbox placement method (A or B) or fix the routing method (Destination Frequency or Location-based). All these results have the same conclusions as before. In addition, they all show that with larger $N_{\text{max}}$ all methods can achieve better delivery ratios and delays.

5.5. Summary

To summarize all these simulation results, we can conclude the following discoveries.

- With deployed throwboxes, all routing methods can achieve significantly higher average successful delivery ratio (than those without throwboxes).
- Among three communication models, Model III can achieve the best performances (with the highest successful delivery ratio and the smallest delay) but uses the largest number of forwarding.
- More throwboxes can further improve the routing performances (higher successful delivery ratio and smaller average delay).
- With a larger number of duplicates (larger value of $N_{\text{max}}$) allowed in the system, all methods can achieve larger delivery ratios and smaller delays.

- Our proposed social-based methods can indeed improve the performances for all routing methods by smartly pick the locations of deployed throwboxes.
- When the number of deployed throwboxes is large enough, there is no significant improvement of the proposed method over random deployment.
- Among five proposed social-based methods, Method A and Method D have slightly better delivery ratios than most of other routing methods while Method B, which considers the locations’ betweenness centrality, has significantly less number of forwarding than other methods.
- When the network is not well connected, Method B and Method E, which consider betweenness centrality, have better successful delivery ratio than others.

6. Related work

Throwbox-assisted DTNs are first proposed in [34] where the gain on the network throughput of deploying throwboxes is studied. A joint throwbox deployment and routing optimization problem is formulated and a greedy algorithm is proposed which relies on network flow techniques to solve multiple linear programming problems. However, this study only focuses on the average capacity, i.e., the maximum data rate that can be sent between two nodes in long term. Different from them, we consider the throwbox placement to optimize the overall routing performances (e.g. delivery ratio and delay). Banerjee et al. [36] consider energy
Fig. 17. Performance comparison of routing algorithms as the number of duplicates increases with throwbox placement Method A over a not well connected network.

Fig. 18. Performance comparison of routing algorithms as the number of duplicates increases with throwbox placement Method B over a not well connected network.
efficiency inside each throwbox for throwbox-assisted DTNs. They not only propose an energy-efficient architecture for throwboxes, but also build such architecture in a real testbed. Their energy optimization is only performed within each individual throwbox. Li et al. [37] formulate a $k$-throwbox placement problem which aims to find places to put $k$ throwboxes so that the network reliability is maximized over time. They assume that the positions of individual nodes and the evolving topology are known, thus such time-evolving networks can be modeled by space–time graphs. Several greedy algorithms are proposed to solve the deployment problem. There are also other studies on providing analytical models for delay distribution of throwbox-assisted DTNs [38,39] and designing/evaluating routing strategies for throwbox-assisted DTNs [40,41]. These works do not consider how to deploy throwboxes in the network.

Social based approaches have been used for DTN routing [33], where the knowledge of social characteristics and relationships among mobile users is used for better forwarding decisions. For example, SimBet [26] prefers a relay node with high social centrality and more common neighbors with the destination; Label [30] and Group [31] try to forward packets to a node within the same social group of the destination; Bubble Rap [23] forwards data via a hierarchical community structure and chooses the node with higher centrality in the community; while Geo-Social [52] leverages both social and location characteristics among users to make routing decisions. Note that these social characteristics among mobile users are usually long-term characteristics and less volatile than node mobility. In this paper, we apply social-based approaches to pick the deployment sites of throwboxes for large-scale mobile DTNs.

Notice that in this paper we study a hybrid DTN network, where static throwboxes are combined with dynamic mobile nodes to enhance data forwarding. Beside of such throwbox-assisted DTNs, a routine-based strategy is proposed to consider the restrictions imposed by transportation modes to people trajectories and the space–time interaction between people and urban locations. However, the WiFi access points studied in these problems is different with throwboxes, since they can directly transfer the messages via internet after receiving them from mobile users. However, throwboxes do not have any wired connection to internet, they still use contact opportunities to forward messages to other mobile users. Therefore, the methods proposed in [50,51] cannot be used here.

Social based approaches have been used for DTN routing [33], where the knowledge of social characteristics and relationships among mobile users is used for better forwarding decisions. For example, SimBet [26] prefers a relay node with high social centrality and more common neighbors with the destination; Label [30] and Group [31] try to forward packets to a node within the same social group of the destination; Bubble Rap [23] forwards data via a hierarchical community structure and chooses the node with higher centrality in the community; while Geo-Social [52] leverages both social and location characteristics among users to make routing decisions. Note that these social characteristics among mobile users are usually long-term characteristics and less volatile than node mobility. In this paper, we apply social-based approaches to pick the deployment sites of throwboxes for large-scale mobile DTNs.
there are other types of hybrid networking. For example, Guo et al. [53] consider to leverage the combination of online social networks and opportunistic networks for data dissemination. How to tackle challenging communication tasks via hybrid networking with diverse information resources becomes an important research topic. Our study on social-based throw box placement is just a small step towards this direction.

7. Conclusion

Recent studies have shown that deployment of throwboxes can significantly enhance the DTN routing performances, which can be utilized to offload the overcrowded cellular system and support large-scale mobile ubiquitous sensing. This paper studies a throwbox placement problem in a large-scale mobile social DTN. By leveraging the social properties discovered from the real-life tracing data, we propose a set of social-based throwbox placement algorithms in which the locations of deployed throwboxes are carefully picked based on social properties of mobile users and/or locations. We show the efficiency of the proposed methods through extensive simulations over the D4D mobile tracing data. Last, we believe that social-based approaches can be found very useful for many tasks (beyond the throwbox placement) in large-scale mobile sensing.

Our proposed throwbox deployment problems, solutions and evaluations have certain limitations. First, the number of potential candidate locations \( m \) is finite. In reality, the number of such locations could be infinite depending on the deployment regions and environments. That will make the deployment problem more challenging, and certain geometric techniques and analysis have to be considered. Second, our proposed social-based throwbox placement algorithms have to obtain the social properties from certain amount of historical data. This is the same weakness for all social based approaches. How to achieve sufficient accuracy social properties with minimum historical data will be an interesting problem. Third, due to the limits from the D4D dataset, we do not have direct contact records, real traffic loads, and available link bandwidths among all devices to make our evaluations more realistic. We hope that we can make-up such weaknesses when a more completed large-scale dataset is available to the research community. We believe that such realistic scenarios would provide more significant performance improvements of the proposed placement schemes. Four, in this paper we do not provide any theoretical analysis on the proposed methods. We leave such a study and addressing the previous limitations as our future work.

Acknowledgments

This work is supported in part by the US National Science Foundation under Grant No. CNS-1050398, CNS-1319915 and CNS-1343355, and the National Natural Science Foundation of China under Grant No. 61428203. 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


