Social Based Throwbox Placement in Large-Scale Throwbox-Assisted Delay Tolerant Networks

Zhu Ying Chao Zhang Yu Wang
Department of Computer Science, University of North Carolina at Charlotte, Charlotte, NC 28223, USA
Emails: {yzhu17,czhang25,yu.wang}@uncc.edu.

***Abstract*—Recent advances in Delay Tolerant Networks (DTNs) allow delivering packets among mobile devices via opportunistic communications during intermittent contacts. However, the lack of rich contact opportunities still causes poor delivery ratio and long delay of DTN routing, especially for large-scale networks. Deployment of additional stationary throwboxes can create a greater number of contact opportunities, thus improve the performance of DTN 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 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.**

I. Introduction

Delay Tolerant Networks (DTNs) have a wide range of applications in challenging environments, such as space communications, vehicular networks, mobile sensor networks, and mobile social networks. 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 [1]–[8] 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.

One way to improve DTN performance is to deploy additional stationary nodes, called *ThrowBoxes* (TBs), to create a greater number of contact opportunities [9]–[14]. 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. As shown in Fig. 1, when two 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 [10]–[14] have demonstrated that introducing small amount of throwboxes can indeed improve the routing performances and overall throughputs.

For a large-scale DTN (in term of amount of mobile users), it is impossible to deploy a huge amount of throwboxes due to budget constaint. 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 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 [15], [16] has been well studied. However, in 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 [9], 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 DTN so that the performance of DTN routing can be improved and maximized. Since mobile devices are usually carried by people, whose behaviors are better described by their social characteristics, 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 deployed locations. Simulation results based on real-life tracing data of large-scale mobile users demonstrate the efficiency of the proposed methods.

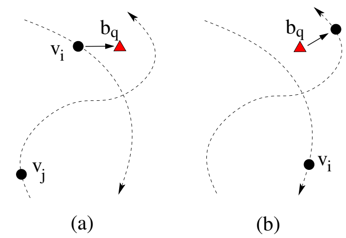


Fig. 1. **Illustration of throwbox-assisted DTNs:** (a) when a mobile node v_i passes by a throwbox b_q, it can left a copy of the message to b_q; (b) later, throwbox b_q can then forward the message to another passing by node v_j.

The rest of this paper is organized as follows. Section II introduces our models and formally defines the throwbox placement problem. Section III describes the D4D dataset [17], [18] we used. A set of greedy algorithms are presented in Section IV and simulation results are reported in Section V. Finally, Section VI provides a briefly review on throwbox-assisted DTNs and Section VII concludes this paper.

This work is supported in part by the US National Science Foundation under Grant No. CNS-1319915 and CNS-1343355.

II. Throwbox Placement Problem and Models

A. Throwbox Placement Problem

Throwboxes can be used in variety of scenarios. As shown in Fig. 1, stationary TBs can relay data between mobile nodes via a "store-and-forward" fashion. Assume that $V = \{v_1, \cdots, v_n\}$ and $B = \{b_1, \cdots, b_m\}$ 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[1]. However, our proposed throwbox placement algorithms work for any other candidate location sets.

With the helps from throwboxes there will be more forwarding opportunities among mobile devices, thus it increases 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 DTNs with n mobile users V 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.

B. 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 is limited up to N_{max}. If the total number of copies for a message already reaches N_{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_{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.

[1]It is mainly due to: (1) based on our analysis from the large-scale cellular data 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 implement throwboxes there; (3) with the cellular tower access record available in the cellular dataset, it is easy for us to get the location and social characteristics of these locations.

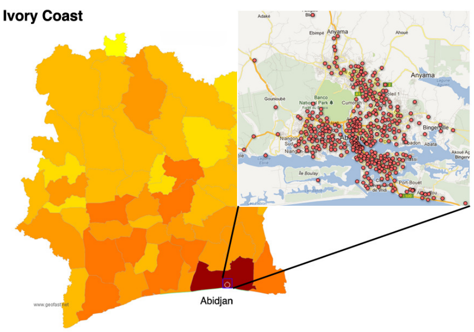


Fig. 2. **Visualization of D4D dataset**: the traffic distribution among all regions are shown in color where darker color indicates heavier traffic loads; the small blue rectangle around Abidjan shows the limited region with high traffic we select for our simulations; and a zoomed view of the detailed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

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_{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 V-A. The maximum number of copies N_{max} can be implemented via tokens, similar with [3].

III. D4D Datasets

To simulate the large scale DTNs, we use a real life wireless tracing data from the cellular operator Orange for the *Data for Development (D4D) challenge* [17]. The released D4D datasets [18] 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. We use the dataset of individual trajectories with high spatial resolution (*SET2* in D4D datasets), which contains the access records of antenna (cellular tower) of each mobile user over every two-week period during the whole duration of D4D data collection 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.

Since D4D datasets do not have direct encounter information between phones via short range communications (such as Bluetooth or WiFi), 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, $1,097$ cellular towers, and $6,787,594$ encounters between users in total. In a previous study [19], 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 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. Thus, 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 $6,318$ active mobile users, 496 cellular towers, and $327,717$ encounters.

IV. 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.

A. Social Characteristics of Locations/Users

We first introduce different social properties that we obtain from the mobile tracing data. For each mobile users v_i, we analyze its access records to cellular towers and construct the location profile of this user as $L(v_i) = \{b^i_1, b^i_2, \cdots, b^i_m\}$, 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 for 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 a top 10 places (in term of visiting frequencies or durations) which v_i visits. Let $T(v_i)$ be the set of such places. We then can define a *user-location* graph G_{ub} (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_i, we can also define a top 10 places (in term of visiting frequencies or durations) which visits b_i, denoted by $T(b_i)$. Here, when a user v_p visits location b_i with certain duration and $b_j \in T(v_p)$, we treat b_j visits b_i once for the same duration. By this way, we basically defines a two-hop relationship between two places b_i and b_j via a mobile user v_p. High frequency or duration among two locations indicates high frequency or duration of user mobility between these two locations. A *location-location* bipartite graph G_{bb} 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.

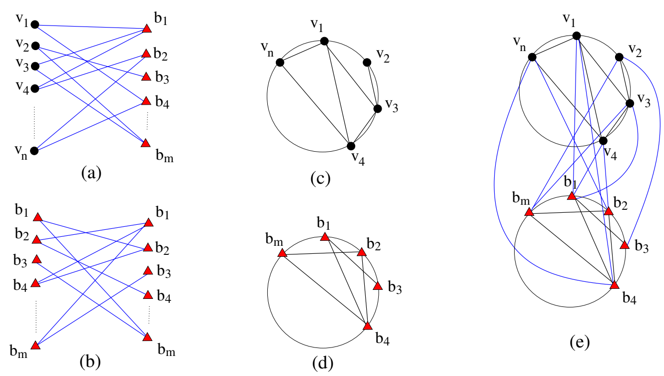


Fig. 3. **Social graphs in throwbox-assisted DTNs:** (a) the *user-location* bipartite graph G_{ub} shows every user's top 2 locations; (b) the *location-location* bipartite graph G_{bb} shows every location's top 2 locations; here we assume that the total visiting frequencies or durations of b_2 and b_m to b_1 are larger than that of b_4. (c) the social graph among users G_u; (d) the social graph among locations G_b; (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.

- 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 top 10 places. Fig. 3(c) shows the example.
- A social graph among locations can be defined as G_b, where there is an edge between b_i and b_j if they share a common top 10 places. Fig. 3(d) shows the example.
- A social graph among both users and locations can be defined as G, which combines G_u and G_b using the edges from G_{ub}, i.e., connecting 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 $B(v_i)$ 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 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_{ub} 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.

B. 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 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.

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. 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, $D(v_i)$ is normalized to a value $\in [0, 1]$. For each location b_j, we then define a new metric 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., $\sum_{any\ v_p\ s.t.\ b_j \in T(v_p)} D(v_p) = D(b_i)$ where only the number of mobile users who has 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 $B(v_i)$ is normalized and the metric is defined 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.

V. Simulations Results

To test our proposed throwbox placement schemes, we implement four representative DTN routing algorithms.

- Fresh [5]: 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 [2]: 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 [1], [4].
- 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 [6].

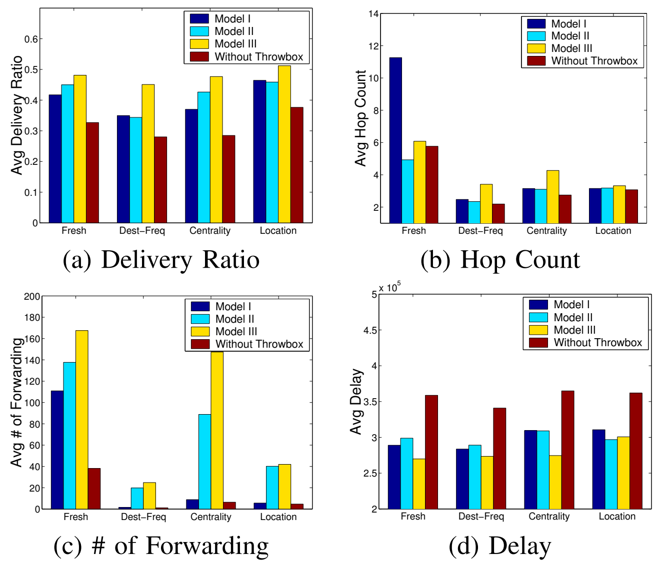
We conduct extensive simulations of proposed throwbox placement schemes with these routing algorithms on the D4D data set described in Section III and measure the following four metrics: *average successful delivery ratio*, *average hop count* of successfully delivered message, *average number of forwarding*, *average delay* of successfully delivered message. For all experiments, we perform $5,000$ random routing tasks among the selected participators in each two-week period. All results reported here are the average over these tasks and over different two-week periods. For all routing algorithms, we use their multiple copy versions where the number of duplicates of a message on mobile users is limited by 10, i.e., $N_{max} \leq 10$.

A. Communication Models of Throwboxes

We first evaluate the three different communication models introduced in Section II-B for TB-assisted DTNs and compare their performances with the case without any TB deployed. For each model, we pick 100 active mobile users and use Method A to select 20 TBs 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. Fig. 4 illustrates the detailed results. It is clear that with deployed TBs all routing methods can achieve significantly higher average successful delivery ratio (than those without TBs). Model III achieves the best performances (with the highest successful delivery ratio and the smallest delay) 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.

B. Number of Throwboxes

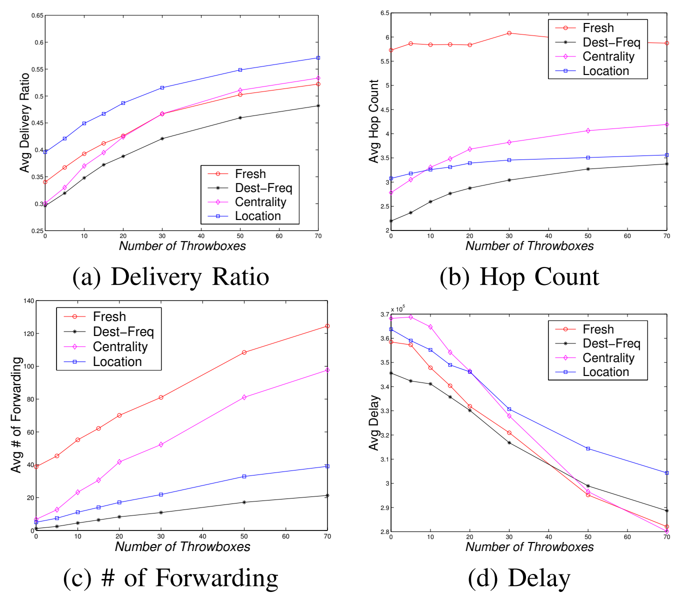
We also study the effect of the number of deployed TBs with both random deployment and proposed social-based deployment. For both sets of simulations, we select 100 active mobile users and vary the number of deployed TBs from 5 to 70. Fig. 5 and Fig. 6 illustrate the simulations with random deployed TBs and TBs 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 TBs increases. Thus, more TBs 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



(a) Delivery Ratio (b) Hop Count

(c) # of Forwarding (d) Delay

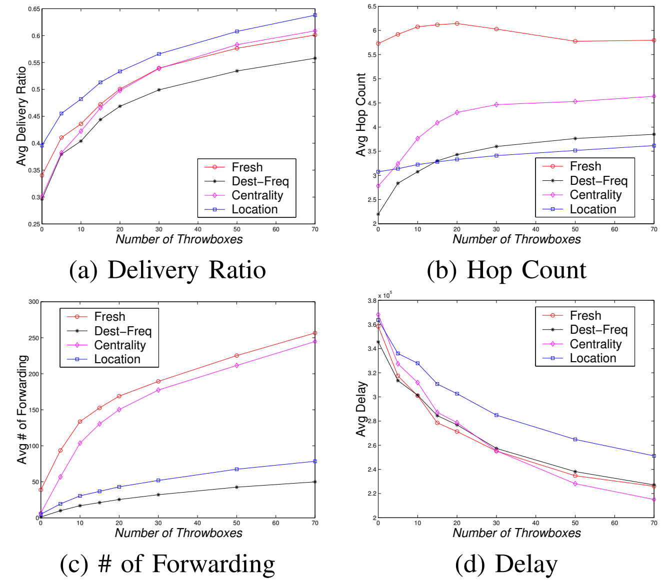
Fig. 4. Comparison on various communication models in TB-assisted DTNs.



(a) Delivery Ratio (b) Hop Count

(c) # of Forwarding (d) Delay

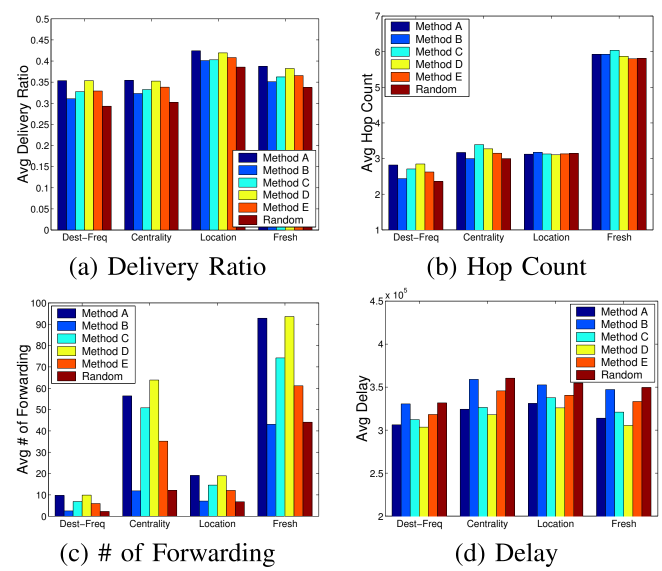
Fig. 5. Results with different number of TBs deployed randomly.



(a) Delivery Ratio (b) Hop Count

(c) # of Forwarding (d) Delay

Fig. 6. Results with different number of TBs deployed by Method A.



(a) Delivery Ratio (b) Hop Count

(c) # of Forwarding (d) Delay

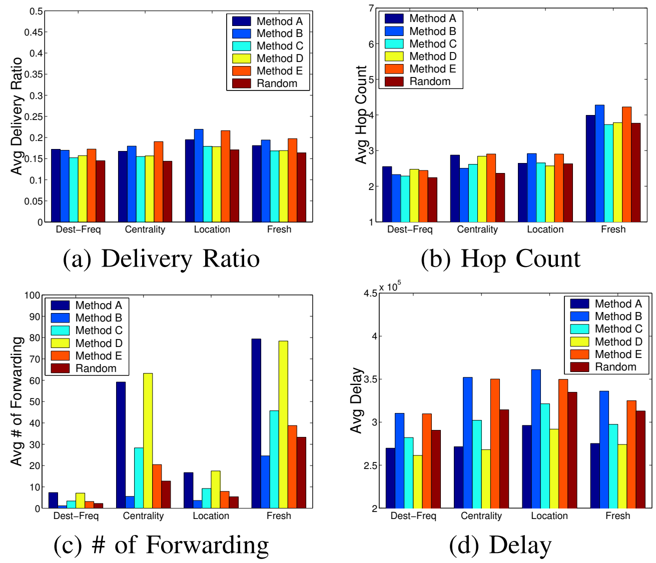
Fig. 7. Performance comparison of different throwbox placement schemes.

TBs is small, i.e., the successful delivery ratio of Method A increases faster than the one with random deployment. When the number of deployed TBs is large enough, there is no significant improvement over random deployment.

C. Throwbox Placement Schemes

Finally, we compare the proposed five TB placement methods and random deployment by fixing the number of deployed TBs to 5. Fig. 7 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 in most of routing methods. Method B, which considers the locations' betweenness centrality, has significantly less number of forwarding than other methods. This may due to that putting TBs at "bridge" locations (locations with high betweenness centrality in the social graph G) reduces unnecessary forwardings among TBs and mobile users.

Besides the experiments with 100 selected active users (well connected in social graph of users), 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). Results in this scenario are given in Fig. 8. Clearly, the performances are much poorer than those in the previous simulations since the connectivity between two components are loose. Now Methods B and E, which consider betweenness centrality, have better successful delivery ratio than others. This is due to that the locations selected by these two methods



(a) Delivery Ratio
(b) Hop Count
(c) # of Forwarding
(d) Delay

Fig. 8. Performance comparison of different throwbox placement schemes over a not well connected network.

can act as "bridge" nodes to connect the separate components.

Overall, our proposed social-based methods can indeed improve the performances for all routing methods by smartly picking the locations of deployed throwboxes.

VI. Related Work

TB-assisted DTNs are first proposed in [9] where the gain on the network throughput of deploying TBs is studied. A joint TB 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 TB placement to optimize the overall routing performances (e.g. delivery ratio and delay). Banerjee *et al.* [10] consider energy efficiency inside each TB for TB-assisted DTNs. They not only propose an energy-efficient architecture for TBs, but also build such architecture in a real testbed. Their energy optimization is only performed within each individual TB. There are also other studies on providing analytical models for delay distribution [11], [12] and designing/evaluating routing strategies [13], [14] for TB-assisted DTNs. These works do not consider how to deploy TBs. Notice that different relay placement problems in static wireless networks have been well-studied, such as static relay placement [15] or mobile relay planning [16] in static wireless sensor networks. However, the networks studied in this paper are time-evolving DTNs where wireless devices are mobile and the network topology evolves over time. Relay placement for such dynamic networks has never been studied except for [9].

Social based approaches have been used for DTN routing [8], where the knowledge of social characteristics and relationships among mobile users is used for better forwarding decisions. For example, *SimBet* [4] prefers a relay node with high social centrality and more common neighbors with the destination; *Group* [7] try to forward packets to a node within the same social group of the destination; while *Bubble Rap* [1] forwards data via a hierarchical community structure and chooses the node with higher centrality in the community. In this paper, we apply social-based approaches to pick the deployment sites of TBs.

VII. Conclusion

Recent studies have shown that deployment of TBs can significantly enhance the DTN routing performances. This paper studies throwbox placement problem in a large-scale mobile 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 TBs 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. We would like to thank Orange and the D4D challenge organizers to provide us the D4D datasets and allow us to continue working on them after the D4D challenge.

References

[1] P. Hui, J. Crowcroft, and E. Yonek, "Bubble rap: Social-based forwarding in delay tolerant networks," in *Proc. of ACM MobiHoc*, 2008.
[2] V. Erranmilli, M. Crovella, A. Chaintreau, and C. Diot, "Delegation forwarding," in *Proc. of ACM MobiHoc*, 2008.
[3] T. Spyropoulos, et al., "Spray and wait: an efficient routing scheme for intermittently connected mobile networks," in *Proc. ACM WDTN*, 2005.
[4] E. M. Daly and M. Haahr, "Social network analysis for routing in disconnected delay-tolerant manets," in *Proc. of ACM MobiHoc*, 2007.
[5] H. Dubois-Ferriere, et al., "Age matters: efficient route discovery in mobile ad hoc networks using encounter ages," in *ACM MobiHoc*, 2003.
[6] J. Leguay, T. Friedman, and V. Conan, "DTN routing in a mobility pattern space," in *Proc. of ACM WDTN*, 2005.
[7] L. Zhao, F. Li, and Y. Wang, "Routing with multi-level social groups in mobile opportunistic networks," in *Proc. of IEEE Globecom*, 2012.
[8] Y. Zhu, B. Xu, X. Shi, and Y. Wang, "A survey of social-based routing in delay tolerant networks: Positive and negative social effects," *IEEE Communication Survey and Tutorials*, 15(1):387–401, 2013.
[9] W. Zhao, Y. Chen, M. Ammar, M. Corner, et al., "Capacity enhancement using throwboxes in DTNs," in *Proc. of IEEE MASS*, 2006.
[10] N. Banerjee, M. D. Corner, and B. N. Levine, "Design and field experimentation of an energy-efficient architecture for DTN throwboxes," *IEEE/ACM Trans on Networking*, 18(,2):554–567, 2010.
[11] M. Ibrahim, et al., "Delay and resource analysis in MANETs in presence of throwboxes," *Performance Evaluation*, 64(9-12):933–947, 2007.
[12] B. Gu and X. Hong, "Latency analysis for thrown box based message dissemination," in *Proc. of IEEE GLOBECOM*, 2010.
[13] M. Ibrahim, P. Nain, and I. Carreras, "Analysis of relay protocols for throwbox-equipped DTNs," in *Proc. of IEEE WiOPT*, 2009.
[14] B. Gu and X. Hong, "Capacity-aware routing using throw-boxes," in *Proc. of IEEE GLOBECOM*, 2011.
[15] E. L. Lloyd and G. Xue, "Relay node placement in wireless sensor networks," *IEEE Trans on Computers*, 56(1):134–138, 2007.
[16] F. El-Moukaddem, E. Torng, and G. Xing, "Mobile relay configuration in data-intensive wireless sensor networks," *IEEE Trans. on Mobile Computing*, 12(2):261–273, 2013.
[17] Data for Development (D4D) Challenge, http://www.d4d.orange.com.
[18] V. D. Blondel, M. Esch, C. Chan, F. Clerot, P. Deville, E. Huens, F. Morlot, Z. Smoreda, and C. Ziemlicki, "Data for development: The d4d challenge on mobile phone data," in *arXiv.1210.0137v2*, 2013.
[19] Y. Zhu, C. Zhang, and Y. Wang, "Mobile data delivery through opportunistic communications among cellular users: A case study for the D4D challenge," in *Proc. of NetMob*, 2013.