Analysis of Hop Limit in Opportunistic Networks by Static and Time-Aggregated Graphs

Suzan Bayhan*, Esa Hyytiä†, Jussi Kangasharju*, and Jörg Ott†
* Department of Computer Science, University of Helsinki, Finland
Email: bayhan@hiit.fi, jakangas@cs.helsinki.fi
†Department of Communications and Networking (COMNET), Aalto University, Finland
Email: {esa, jo}@netlab.tkk.fi

***Abstract*—Hop count limitation helps controlling the spread of messages as well as the protocol complexity and overhead in a distributed network. For a mobile opportunistic network, we examine how the paths between any two nodes change with increasing number of hops a message can follow. Using the *all hops optimal path* (AHOP) problem, we represent the total delay of a route from a source node to a destination node as *additive weight* and use the number of encounters as a representation of *bottleneck weight*. First, we construct a *static (contact) graph* from the meetings recorded in a human contact trace and then analyze the change in these two weights with increasing hop count. Alternatively, we aggregate all the contact events in a time interval and construct several time-aggregated graphs over which we calculate the capacity metrics. Although, we observe differences in the properties of the static and the time-aggregated graphs (e.g., higher connectivity and average degree in static graph), our analysis shows that second hop brings most of the benefits of multi-hop routing for the studied networks. However, the optimal paths —path that provides the most desirable bottleneck/additive weight— are achieved at further hops, e.g, hop count ≈ 4. Our finding, which is also verified by simulations, is paramount as it puts an upper bound on the hop count for the hop-limited routing schemes by discovering the optimal hop count for both additive and bottleneck weights.**

I. Introduction

Mobile opportunistic networks rely on short-range radios (e.g., Bluetooth, WiFi Direct) to transmit data between two nodes and exploit the mobility of nodes to physically carry messages from one location to another. This kind of operation is favourable for several reasons: (i) interconnection without dependency on the infrastructure, (ii) *hop gain* due to direct link between the transmitter and the receiver [1], (iii) spectrum reuse gain, and (iv) mobile data offloading. On the other hand, it is more challenging to provide guaranteed performance in such a *dynamic* network. Mobility of nodes results in intermittent connections and raises uncertainty in the network topology. The major challenge is hence the lack of global knowledge at the nodes to decide on the optimal forwarding paths or other networking tasks.

Epidemic protocol [2] is the simplest protocol that does not rely on any information about the network (e.g., about nodes and connections) but greedily replicates a message to every node that does not have it. Albeit being desirable due to its simplicity, epidemic protocol over-consumes network resources (e.g., waste of bandwidth due to too many replications) and may not perform well under resource-restricted networks (e.g., short contact duration). A less greedy solution is *hop-limited routing* [3], [4] which limits the journey of a message in the network to maximum h hops. More sophisticated protocols aim to balance the tradeoff between delivery ratio and resource consumption by tuning the protocol parameters, e.g., the maximum number of replications [4], lifetime of a message [5], replication/forwarding logic [6], and so on. However, no matter how optimized the protocol is, the performance of a mobile opportunistic network also strongly depends on the *node mobility*. More specifically, two properties related to node mobility are paramount: *contact duration* and *inter-contact time duration*. Contact duration is the time two nodes stay connected while inter-contact time is the time elapsed between two consequent contacts of two particular nodes. Both determine the transmission capacity (i.e., how much data can be transmitted) as well as the speed of change in the network topology.

Although opportunistic routing protocols have been studied extensively from many perspectives, the effect of hop limitation is not yet fully understood. While [3] and [7] provide theoretical analysis on hop-limited forwarding, our work differs from them in two ways. First, rather than a pure theoretical approach, we explore the effect of hop count using the real human contact traces. Second, we research if the analysis approach – modelling the network as static or time-aggregated graphs– makes a difference in our conclusions.

We start with constructing the connectivity graph from a trace and explore the change in the network performance indicators, e.g., fraction of reached nodes, using the solution for *all hops optimal path problem* (Section III). This *static* graph aggregates all the contacts recorded in the trace into a contact graph. Alternatively, we sample the network on regular time intervals and analyze each *time-aggregated graph* separately (Section IV). Social-aware opportunistic routing schemes largely tend to aggregate contacts either using a sliding window approach (e.g., BubbleRap [8]) or accounting for the whole past events (e.g., SimBet [9]) to derive metrics about the nodes, e.g., centrality. However, [10] shows that both short and long time windows may fail to differentiate the nodes according to their social metrics. Similarly, we compare the two approaches for hop analysis to see how local observations agree/disagree with the analysis of the static graph.

Finally, we simulate a hop-limited forwarding scheme to see the change in performance during the actual operation (Section V). Obviously, the third analysis explains the real effect of the hop count. However, the formers are also useful for estimating the performance if they provide similar conclusions.

Our main contribution is that we provide answers to the following research questions for the considered opportunistic networks: **Q1**: How is the average time to send a packet from one arbitrary node to another arbitrary node affected by hop restriction h?; **Q2**: How is the fraction of nodes reachable from one arbitrary node affected by h?; **Q3**: How is the delivery ratio from one arbitrary node to another arbitrary node affected by h?

II. System Model

We consider a network of n mobile nodes. A *contact* starts when two nodes come in transmission range of each other, and ends when they cannot maintain a connection. Nodes can exchange messages only during contact time.

In a hop-limited routing scheme, each message has a *hop count* field that shows the number of routers this message has followed. When a message is forwarded from one node to another, the hop count of this message is incremented by 1 at the receiving node. If the hop limitation is h, the message can only be forwarded h hops.

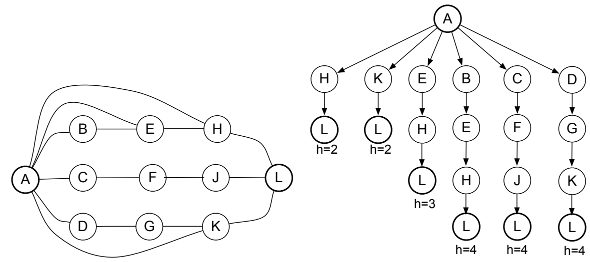
We use the following human contact traces in our analysis Infocom05 [11], Cambridge and Infocom06 (collected by Haggle project [12] and downloaded from [13]), whose basic properties are listed in Table I.

TABLE I
Human contact traces used in the analysis.

Trace	Duration	# of devices (n)	Context
Infocom05	≈3 days	41	Conference participants
Cambridge	≈11 days	36	1st and 2nd year undergraduate students
Infocom06	≈4 days	78 (and 20 stationary devices)	Conference participants

III. All Hops Optimal Paths (AHOP)

One common practise in exploring the network capacity is to analyze the network topology. In opportunistic networks, the topology changes frequently. However, human contact traces that are collected for a long duration can reflect the stationary contact probabilities which then can be used for constructing the expected network topology. Given the network's stationary meeting characteristics, we can model the encounter events as a graph $G = (\mathcal{V}, \mathcal{E})$ where vertex set $\mathcal{V}$ is the set of nodes and edge set $\mathcal{E}$ is the set of all possible pairwise contacts among nodes. In this graph, there are $n = |\mathcal{V}|$ vertices and maximum $|\mathcal{E}| = O(n^2)$ edges for which we assign weights (w) based on the contact characteristics (e.g., inter-contact time, number of meetings) of the two connecting vertices. Given a source node s, our aim is to find the shortest (e.g., fastest) path to a destination d that has at most k hops for each k, $1 \leq k \leq h$. In the most general form, this problem is known as *all hops optimal path problem* (AHOP) [14]. Fig. 1(a) shows an example connectivity graph. An edge in this graph indicates that the two nodes has met at least once. Fig. 1(b) depicts some of the paths from A to L (the leaf nodes). Number near each leaf shows the hop count of the path. The optimal path among all these paths depends on the edge weights.



(a) Static graph. (b) Paths from A to L. No path exists for $h = 1$.

Fig. 1. (a) Static graph derived from the encounters in the contact trace, (b) paths from A to L in $h = \{1, 2, 3, 4\}$ hops.

Let $\mathbf{p}$ be a path from s to d consisting of vertices $\langle s = n_0, n_1, \cdots, d = n_k \rangle$ in the given order, and let $w(n_i, n_j)$ be the weight of the edge between n_i and n_j. Denote by $w(\mathbf{p})$ the weight of a path $\mathbf{p}$ and define it as a function of weights of edges along this path:

$$w(\mathbf{p}) = f(w(n_0, n_1), w(n_1, n_2), \ldots, w(n_{k-1}, n_k))$$

where $k \leqslant h$ and $n_i \in \mathcal{V}$ for all i. If f is additive, the weights are said to be *additive weights*; if f is maximum/minimum, the weights are *bottleneck weights* [14], [15]. Both metrics are of our interest as they reflect different performance requirements in networks; bottleneck weight matches the minimum bandwidth available for a service whereas additive weight is more appropriate measure for the total delay of a path. As for opportunistic forwarding, we consider both weights:

(i) *additive weight*: expected time to reach a target node is the total expected inter-contact time of the selected path. Hence, we compute the total delay between s and d as the additive weight of the path $\mathbf{p}$:

$$w(\mathbf{p}) = \sum_{(n_i, n_j) \in \mathbf{p}} w(n_i, n_j)$$

where $w(n_i, n_j)$ is the expected inter-contact time between n_i and n_j.

(ii) *bottleneck weight*: as encounters are probabilistic, we aim to select the edges that will appear with high probability. In a sense, given the number of encounters among nodes, a routing scheme tries to decrease the risk of relying on less probable encounters by selecting the *most probable* paths. An additive weight may not properly choose the most probable paths if only the total delay is considered. Therefore, we define the weight of an edge as the inverse of the number of encounters between the corresponding nodes. Then, the weight of $\mathbf{p}$ is the maximum weight of the edges along this path:

$$w(\mathbf{p}) = \max_{(n_i, n_j) \in \mathbf{p}} w(n_i, n_j).$$

Let $l(\mathbf{p})$ denote the length of the path, i.e., the number of edges. Then, we define an h-hop constrained optimal shortest path as the path with at most h hops that yields the minimum weight among all paths between s and d. More formally, $\mathbf{p_h}*$ is defined as follows:

$$\mathbf{p_h}* = \arg\min_{\mathbf{p}} w(\mathbf{p}) \text{ and } l(\mathbf{p}) \leqslant h.$$

Once we find $\mathbf{p_h}*$, we can find the hop count $h*$ realizing the optimal path. We refer to it as the *optimal hop count* and define as follows: $h^* = l(\mathbf{p}^*_\mathbf{h})$.

In [14], it was shown that Bellman-Ford algorithm provides the best solution with complexity $O(h|\mathcal{E}|)$ for additive weights whereas for the simpler case of bottleneck weights authors propose an improvement upon Bellman-Ford with a lower complexity. We should note that for human contact networks $h \ll n$ due to the *small world* structure [16]. Hence, although the worst case complexity is *cubic* for additive weights, the average complexity would be much lower, e.g., $O(n^2)$. We compute the solution for AHOP by modifying the Bellman-Ford algorithm for hop restricted shortest paths [17]. Even though we do not focus on efficiency of this calculation, we note that the running time of the algorithm can be decreased by pruning some of the edges via thresholding.

IV. Analysis on the Network Snapshots

Solving the AHOP problem as discussed above provides us the shortest paths and their capacities for given hop restrictions. With this knowledge, we can grasp the network characteristics and the capacity limitations. One of the shortcomings of such an analysis is the loss of temporal network dynamics [18], [19]. In other words, we aggregate the snapshots of the network as if an occurring edge always exists there. On the other hand, *temporal reachability graph* [20] is not identical to the aggregated connectivity graph. In the above AHOP analysis, the time of appearing of the edges is lost. Take the graph in Fig.1(a) as example. Node `B` can be a good relay for `A` in reaching `E`, only if `B` meets `E` after meeting `A`. Fig. 2 illustrates two snapshots of the same graph. Neither of the networks is connected contrary to the network that is formed by aggregating these two snapshots. The encounter between `A` and `B` occurs after `B`-`E` meeting, which makes the path `A` $\rightarrow$ `B` $\rightarrow$ `E` impossible. The static network derived from the encounters does not express this property of the network. Hence, some of the paths discovered by our AHOP solution may be causally impossible due to the time ordering of the edges in the temporal network. As a result, the connectivity of the network is overestimated and thereby also the resulting network capacity [10].

Another approach is to decrease the time window of the analysis. Instead of aggregating the whole contact events, we analyze the network several times and average the statistics over these observations. In this approach, events occurring inside a time window are aggregated, and at the end of the time window we have a snapshot of the network. The proper choice of the time window size is paramount and it

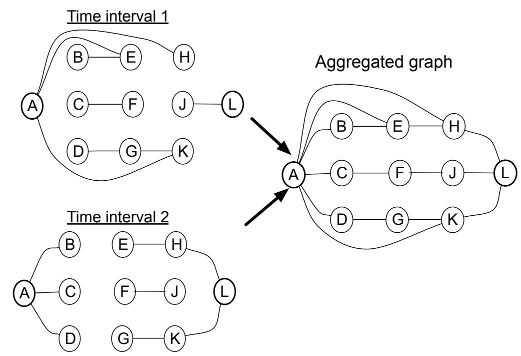


Fig. 2. Connectivity graph (snapshot) in two consequent time intervals and the aggregated graph.

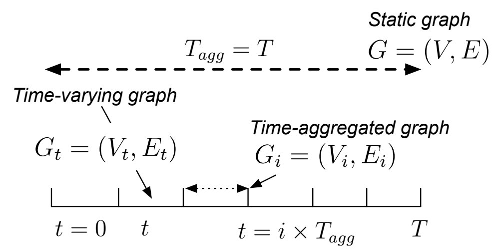


Fig. 3. Static (G), time-aggregated (G_i), and time-varying graph (G_t).

depends on the network dynamics which can be estimated online [21]. Understanding the effect of time window length is of our interest as many opportunistic routing protocols collect encounter information to predict future meetings or make informed forwarding decisions. However, the collected information becomes stale due to the change in the network dynamics. Therefore, the metrics that are derived from the aggregated network may be misleading for these protocols and result in performance degradation. Hence, *time-varying graph* (TVG) and related metrics (e.g., temporal shortest path) are argued to be more powerful [18], [20].

Time-varying graphs (TVG) have been studied under different contexts and with different terminology (see [18]). For example, ad hoc networks [22], animal networks such as ant colonies [23], and social networks [24] entail connections among the network entities that are changing (appearing, disappearing) over time and are complicated compared to the static graphs. TVGs extend the static graphs on time dimension by a *presence function* that shows whether a particular edge exists at a specific time. An opportunistic network is obviously a TVG and previous works [16], [20] highlighted the benefits of TVG modelling for opportunistic networks. In this work, we take a midway approach and analyze the traces by breaking down the whole history of contacts into regular time intervals. This approach is obviously not as accurate as TVGs, however it alleviates the deficiency of the information loss by static graphs to some extent.

Suppose that the aggregation time is T_{agg} for a trace of T time units. Then, we have $\lfloor T/T_{\text{agg}} \rfloor$ intervals. A contact event occurring in a specific interval is reflected as an edge in the corresponding contact graph G_i where i is the index of the graph. For each edge in G_i, we calculate average inter-contact time and number of meetings. Finally, we apply the previous analysis in Section III to each G_i. Fig. 3 summarizes the time granularity of static, TVGs, and time-aggregated graphs.

Lastly, we observe the change in network capacity while the network is under operation. We simulate a flooding-based forwarding protocol that is subject to hop limitations.

V. Numerical Evaluation

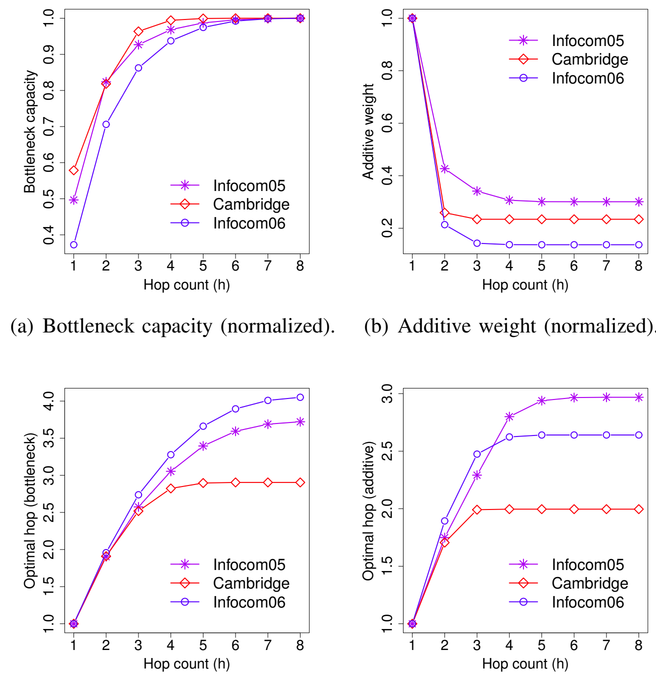
We implemented AHOP algorithm in R and used ONE [25] for simulations. In R, we generate network snapshots from a trace and a list of time intervals using *Timeordered* package [23].

Fig. 4 shows the effect of hop count for all traces. We normalize the bottleneck and additive weights for each trace as we aim to show the change in the network dynamics rather than the actual values of the related metrics. As Figs.4(a) and 4(b) show, relaxing the hop limitation improves the network capacity: higher bottleneck capacity indicating the higher probability of path's existence and lower additive weight indicating lower delays among the nodes. The most significant gain is achieved by letting two hop routing rather than a direct delivery (i.e., $h = 1$). Increasing to $h = 4$ still brings benefits especially for bottleneck capacity, however after $h \approx 4$ change in the considered metrics is negligible. Please note the consistency of the behaviour for all traces. Regarding connectivity (not depicted), we observe that only two hops are sufficient to reach all the nodes in the network from any node. For $h = 1$, the reached fraction of nodes is $(0.96, 0.82, 0.88)$ for Infocom05, Cambridge, and Infocom06, respectively. However, the existing path is not optimal as shown in Figs. 4(c) and 4(d).

Fig. 4(c) and Fig. 4(d) illustrate optimal hop count for bottleneck and additive weights, respectively. We have two observations: all networks represented by the traces achieve optimal operation at very few hops, three to four. Second, generally speaking, the optimal bottleneck capacities are achieved at higher hops compared to additive weights. This is due to the *strength of weak ties* [26] for the additive weights; in bottleneck capacity calculation our algorithm avoids the weak links (with low number of meetings) whereas in additive weights these links may be a fast move towards the target.

In summary, for the research questions we listed in Section I, we have the following conclusions from the AHOP analysis. As for (Q1), nodes can be reached faster by relaxing hop count, however the improvement vanishes after several hops. More specifically, optimal hop counts considering the total path delay are $h \approx 3$ for Infocom05, $h \approx 2$ for Cambridge, and $h \approx 2.6$ for Infocom06. As for (Q2), the first two hops are sufficient to reach every node from every other node. As for (Q3), indirectly from our bottleneck weight analysis, we can conjecture that delivery ratio increases significantly if at least two hops are allowed. However, for all traces the performance increase tends to stabilise after $h \approx 4$.

Next, we set $T_{\text{agg}} = \{1, 6, 24\}$ hours to analyze how time-aggregation affects the network dynamics. We refer to these settings as short, medium, and long aggregation windows.



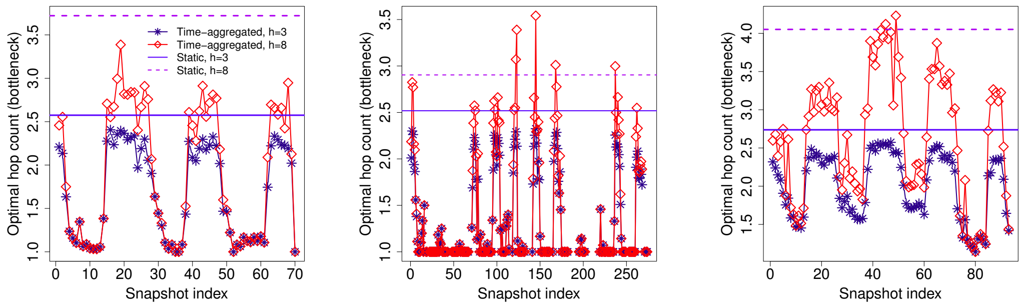
(a) Bottleneck capacity (normalized). (b) Additive weight (normalized).

(c) Optimal hops, bottleneck capacity. (d) Optimal hops, additive weight.

Fig. 4. AHOP analysis for Infocom05, Cambridge, Infocom06.

Before presenting the average capacity and optimal hop count, let us see how optimal hop count changes over time for each trace. In Fig. 5(a), we plot the results for short T_{agg}. As Infocom05 trace is approximately 3 days long, we have 70 snapshots while Cambridge has 274, and Infocom06 has 93 snapshots. The figures show the hop counts providing the best bottleneck weights for $h = 3$ and $h = 8$ for both the time-aggregated graph and static graph. The reachable fraction of nodes is not presented here due to space restrictions, but our results show that static graph overestimates the connectivity of the network. All nodes can be reached in two hops if waited sufficiently long. However, the actual reachable fraction is much lower according to the results of our snapshot analysis in Fig. 6(a). This is not primarily because of the hop restriction but the inherent network dynamics. Albeit setting $h = 8$ leads to a higher connectivity than with $h = 3$, the connectivity is still drastically lower than with the static graph. Figures also capture the difference in time of the day: connectivity during daytime is higher compared to the night times. This change in the connectivity manifests itself as the change in the optimal hop count over time. During some periods, connectivity is so low that although the protocol lets 8 hops, single hop routing achieves the optimal performance (which is very low). This change in hop count may call for protocols that adapt the parameters according to the time of the operation. Regarding the amount of change, the standard deviation is 0.54 hops for $h = 2$ and 0.77 hops for $h = 8$ in Infocom05, while they are $(0.45, 0.62)$ hops in Cambridge and $(0.42, 0.83)$ hops in Infocom06.

Fig. 6 demonstrates the average results collected from

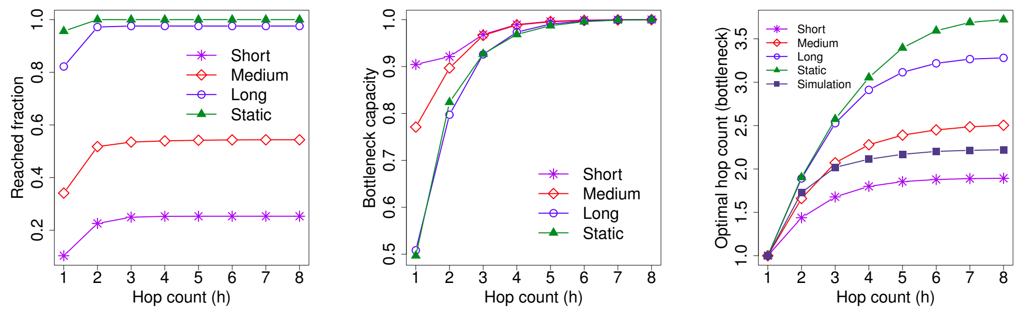


(a) Infocom05.

(b) Cambridge.

(c) Infocom06.

Fig. 5. Change in optimal hops over time.



(a) Fraction of reachable nodes.

(b) Bottleneck capacity (normalized).

(c) Optimal bottleneck hop count.

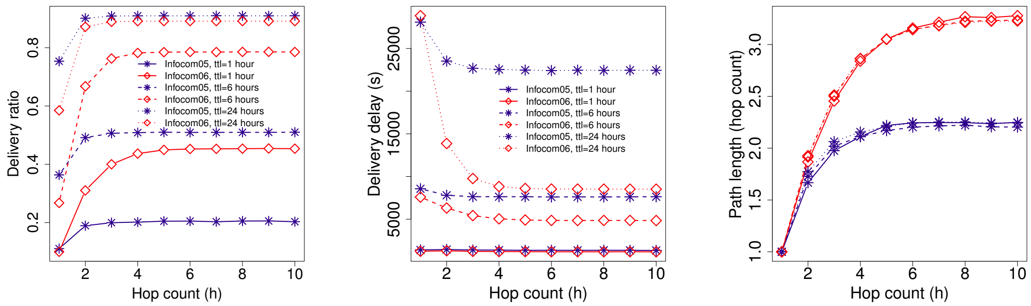
Fig. 6. AHOP analysis for $T_{\text{agg}} = \{1, 6, 24\}$ hours. Infocom05 trace.

multiple snapshots of the network. We present only results for Infocom05 as the rest behave similarly. As our simple example showed, reachable fraction of nodes in static graph is higher than what actual graph G_t could provide. The overestimation is clear in Fig. 6(a): static and long T_{agg} have the highest reachable fraction ($\approx 0.9 - 1$) whereas short T_{agg} is significantly lower (≈ 0.3). Regarding the improvement with increasing h, all exhibit the same behaviour. In line with the results of Section III, we see that $h = 2$ provides most of the benefits of multi-hop routing. However, our previous conclusion that two hops are sufficient to provide full connectivity among all nodes does not hold. Increasing the hop count does not help as the dynamics of the network put a limit on the ratio of reachable nodes in the given time period. Regarding the bottleneck capacity, increasing hop count ($h > 2$) leads to a higher bottleneck capacity for all cases. Similar to our previous results, the benefits diminish after $h \approx 4$. As Fig. 6(c) illustrates, the optimal number of hops are on the average higher for higher T_{agg}. This is due to the added edges that may violate the time ordering in reality but increasing the bottleneck capacity. The closest results to the simulation results are achieved by medium aggregation window while long T_{agg} and static overestimate the hop count and short T_{agg} underestimates it. Please note that best aggregation window size depends on how fast the network evolves and how the protocol operates. Hence, our analysis should not be generalized but rather highlight the change in results with change in T_{agg}.

In Fig. 7, we plot the simulation results: the delivery ratio, delivery delay, and average hop count of the delivered messages for Infocom05 and Infocom06 traces. In these scenarios, we set time-to-live (TTL) of a message to $ttl = \{1, 6, 24\}$ hours. A message exceeding its ttl is dropped. For the sake of comparison, we set buffer capacity and contact capacity large so that the performance is not restricted by these factors. From Fig. 7(a), we can see the improvement facilitated by increasing h. In all scenarios, we see two regions; in the first region the delivery ratio increases with increasing h which later changes marginally in the second region. The turning point is $3 - 4$ hops. This behaviour agrees with our previous AHOP analysis. Unsurprisingly, delivery delay in Fig. 7(b) shows a similar trend to that of additive weights in Fig. 4(b). In this figure, the results are not normalized to give an idea about the delay of communication in such opportunistic networks. For $ttl = 1$ h, the delay changes slightly across different h, which indicates that the performance is primarily determined by the time restriction (and network mobility) rather than the hop count restriction. Lower average hop count in Fig. 7(c) corroborates this claim. For other settings, we observe a significant drop in delivery delay from $h = 1$ to $h = 2$. Another thing to note is that messages are routed in $h \approx 2.3$ hops in Infocom05 trace independently of ttl, whereas $h \approx 3.2$ for Infocom06.

VI. Conclusions

We have explored the effect of hop count restriction on opportunistic routing by modelling our problem as *all hops*



(a) Delivery ratio. (b) Delivery delay. (c) Hop count.

Fig. 7. Simulation results for $ttl = \{1, 6, 24\}$ hours. Infocom05 and Infocom06 traces.

optimal path problem (AHOP). We analyzed human contact traces using three approaches: static graph, network snapshots, and simulations. AHOP analysis on static graph is optimistic as it disregards the time ordering of the links. To decrease the deviation from the actual graph, we observed the network on multiple time points and aggregated all events occurring in a time interval into a network snapshot. We derived the performance by averaging related metrics over the snapshots. Finally, we simulated hop-limited routing. Although characteristics (e.g., connectivity) of the network snapshots may be different than the static graph, they are similarly affected by the hop count limitation. Connectivity of the network as well as the performance (e.g., lower delay) improves with increasing hop count. Our simulation experiments strongly suggest that while two hops are sufficient to achieve the highest connectivity the network can provide given the time restrictions, routing protocols can perform better by increasing hop count to $h \approx 4$. After this point, we did not observe significant improvements in the performance.

In this work, we have studied *small-world* networks that exhibit some community structure which reflected this property as a low number of hops achieving a good connectivity for the static graph. As future work, we plan to provide a formal framework which can guide us to a better understanding of the effect of the hop count for general opportunistic networks.

VII. Acknowledgments

This work was supported by the Academy of Finland in the PDP project with grant number 260389.

References

[1] G. Fodor, E. Dahlman, G. Mildh, S. Parkvall, N. Reider, G. Miklós, and Z. Turányi, "Design aspects of network assisted device-to-device communications," *IEEE Comm. Mag.*, vol. 50, no. 3, pp. 170–177, 2012.
[2] A. Vahdat, D. Becker *et al.*, "Epidemic routing for partially connected ad hoc networks," CS-200006, Duke University, Tech. Rep., 2000.
[3] M. Vojnovic and A. Proutiere, "Hop limited flooding over dynamic networks," in *Proceedings IEEE INFOCOM*, 2011, pp. 685–693.
[4] C. Liu and J. Wu, "On multicopy opportunistic forwarding protocols in nondeterministic delay tolerant networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 6, pp. 1121–1128, 2012.
[5] J. Karvo and J. Ott, "Time scales and delay-tolerant routing protocols," in *Proc. of ACM workshop on Challenged networks*, 2008, pp. 33–40.
[6] E. Jones and P. Ward, "Routing strategies for delay-tolerant networks," *ACM Computer Communication Review (CCR)*, 2006.
[7] E. Hyytiä, S. Bayhan, J. Ott, and J. Kangasharju, "Searching a needle in (linear) opportunistic networks," in *Proc. of ACM MsWiM*, 2014.
[8] P. Hui, J. Crowcroft, and E. Yoneki, "Bubble rap: Social-based forwarding in delay-tolerant networks," *IEEE Transactions on Mobile Computing*, vol. 10, no. 11, pp. 1576–1589, 2011.
[9] E. M. Daly and M. Haahr, "Social network analysis for information flow in disconnected delay-tolerant MANETs," *IEEE Transactions on Mobile Computing*, vol. 8, no. 5, pp. 606–621, 2009.
[10] T. Hossmann, T. Spyropoulos, and F. Legendre, "Know thy neighbor: Towards optimal mapping of contacts to social graphs for DTN routing," in *2010 Proceedings IEEE INFOCOM*, 2010, pp. 1–9.
[11] P. Hui, A. Chaintreau, J. Scott, R. Gass, J. Crowcroft, and C. Diot, "Pocket switched networks and human mobility in conference environments," in *Proc. of the ACM SIGCOMM Workshop on Delay-tolerant Networking*, 2005, pp. 244–251.
[12] "EU FP6, jan.2006–jun.2010," http://www.haggleproject.com.
[13] "Encounter traces for the ONE simulator," http://www.shigs.co.uk/index.php?page=traces, Oct.2, 2014.
[14] R. Guérin and A. Orda, "Computing shortest paths for any number of hops," *IEEE/ACM Transactions on Networking (TON)*, vol. 10, no. 5, pp. 613–620, 2002.
[15] D. H. Lorenz and A. Orda, "QoS routing in networks with uncertain parameters," *IEEE/ACM Transactions on Networking (TON)*, vol. 6, no. 6, pp. 768–778, 1998.
[16] A. Chaintreau, A. Mtibaa, L. Massoulie, and C. Diot, "The diameter of opportunistic mobile networks," in *Proc. of the 2007 ACM CoNEXT conference*, 2007, p. 12.
[17] T. H. Cormen, C. E. Leiserson, and R. L. Rivest, *Introduction to Algorithms, 2009*. MIT Press, 2009, ch. Single-Source Shortest Paths.
[18] A. Casteigts, P. Flocchini, W. Quattrociocchi, and N. Santoro, "Time-varying graphs and dynamic networks," *Int. Journal of Parallel, Emergent and Distributed Systems*, vol. 27, no. 5, pp. 387–408, 2012.
[19] P. Holme and J. Saramäki, "Temporal networks," *Physics reports*, vol. 519, no. 3, pp. 97–125, 2012.
[20] J. Whitbeck, M. Dias de Amorim, V. Conan, and J.-L. Guillaume, "Temporal reachability graphs," in *Proc. of ACM Mobicom*, 2012.
[21] V. Iyer, Q. Liu, S. Dulman, and K. Langendoen, "Adaptive online estimation of temporal connectivity in dynamic wireless networks," in *Int. Conf. on Self-Adaptive and Self-Organizing Systems (SASO)*, 2013.
[22] F. De Pellegrini, D. Miorandi, I. Carreras, and I. Chlamtac, "A graph-based model for disconnected ad hoc networks," in *Proc. of IEEE INFOCOM*, 2007, pp. 373–381.
[23] B. Blonder, T. W. Wey, A. Dornhaus, R. James, and A. Sih, "Temporal dynamics and network analysis," *Methods in Ecology and Evolution*, vol. 3, no. 6, pp. 958–972, 2012.
[24] C. Aggarwal and K. Subbian, "Evolutionary network analysis: A survey," *ACM Computing Surveys (CSUR)*, vol. 47, no. 1, p. 10, 2014.
[25] A. Keränen, J. Ott, and T. Kärkkäinen, "The ONE Simulator for DTN Protocol Evaluation," in *Proc. of SIMUTools*, 2009.
[26] M. Granovetter, "The strength of weak ties: A network theory revisited," *Sociological theory*, vol. 1, no. 1, pp. 201–233, 1983.