Performance Analysis of a Hybrid Flooding-Probabilistic DTN Protocol using Logged Contact Data

Constantino Antunes, Ricardo Morla
INESC Porto and Faculdade de Engenharia, Universidade do Porto
Campus da FEUP, Rua Dr. Roberto Frias, 378, 4200–465 Porto, Portugal
Phone: +351-222094000, Fax: +351-222094050, e-mail: cantunes|ricardo.morla@inescporto.pt

Abstract — **In this paper we analyze packet delivery, buffer space, and transmission performance of a hybrid flooding-probabilistic DTN protocol using the DTN2 framework and one month of third-party logged contact data. This new protocol switches from probabilistic to flooding when the number of node contacts is below a threshold. Results show only 0.52% fewer messages delivered compared to flooding and only 0.34% and 0.16% higher buffer occupancy and bandwidth usage costs compared to probabilistic and suggest that flooding could be used to improve probabilistic DTN performance at times when nodes are more isolated and contacts scarce.**

I. Introduction

Delay tolerant networks (DTN) [1] are useful in cases where there is no end-to-end connectivity and nodes are mobile. DTN nodes take a carry-and-forward approach to deliver data. Several protocols have been proposed to deliver data in these networks, for example based on flooding [2] and on probabilistic delivery [3]. Flooding protocols forward data to all nodes they get in contact with as they move. Probabilistic protocols compute their probability of delivering data and only forward data if the delivery probability of the nodes they encounter is higher.

Because they forward data to all nodes they encounter, flooding protocols provide maximum packet delivery but fall short on other metrics such as buffer space and wireless transmission requirements. By forwarding packets only to nodes with higher delivery probability, probabilistic protocols have low buffer space and transmission requirements than flooding protocols at the cost of not being able to guarantee the delivery of the same number of packets as flooding protocols.

Node contact information is used to compare the performance of flooding and probabilistic protocols. While this information is typically generated based on simulation mobility models, several projects have published more realistic node contact data that is logged on devices that are actually used by people. In this paper we use real contact data from the UMass Trace Repository [4] and the Reality Mining project [5] and combined buffer-transmission-delivery metrics to analyze the performance of a new DTN protocol that is a hybrid between the Epidemic flooding protocol [2] and the Prophet probabilistic routing protocol [3]. This Hybrid protocol tries to provide the best of both worlds – higher packet arrival typical in flooding protocols and lower buffer space and transmission typical in probabilistic protocols.

The authors would like to acknowledge the availability of the UMass Trace Repository (http://traces.cs.umass.edu/) and Reality Mining Project (http://reality.media.mit.edu/) contact data.

II. Hybrid Flooding-Probabilistic Protocol

The new Hybrid protocol calculates the current average of contacts per hour that a node has observed since the beginning of the day (L). At any time, if this value is lower than a switching threshold ε, then the node behaves according to the Epidemic protocol; if L changes and becomes higher than the threshold, then the node immediately starts behaving according to the Prophet protocol. When $\varepsilon=0$ the Hybrid protocol behaves similarly to Prophet and when $\varepsilon=\infty$ it behaves similarly to the Epidemic protocol.

The Hybrid protocol implementation was built on top of the DTN2 framework; the Hybrid protocol extends DTN2's BasicRouter, which hides low level operations from the Hybrid router module. This module has three major functions: (1) process control messages, which are the same regardless of the Hybrid protocol being in Epidemic or Prophet mode and are used to exchange information about which data bundles need to be delivered to whom; (2) manage neighborhood and bundle lists, which includes updating routing tables, acknowledged bundles, and neighbor lists; (3) calculate network statistics by updating a counter each time the node encounters another node and dividing this counter by the total elapsed time since the beginning of the day to yield the current value of L; and (4) L value-based protocol decider and Epidemic and Prophet codes, which access the same bundle list managed in (2) to guarantee data coherence between the two modes.

III. Methodology

A. Metrics

The key metrics that we consider in this analysis are: (1) the total number of packets that arrive to their final destination; (2) the average per hour of the total buffer space used in all nodes; (3) and the average per hour of the number of bytes that are transmitted to other nodes; the latter include DTN control and payload bytes.

Because we must typically consider tradeoffs between these metrics, we prefer to use two cost metrics that indicate how many bytes (buffer space or transmission) the protocol uses on average to deliver one data packet. In the traffic model that we use, nodes send a fixed-length 1 Kbyte packet every simulation hour to a single node of the network which acts as the network gateway; thus the cost of delivering one packet is equivalent to the cost of delivering 1 Kbyte of data.

B. Logged Data

We used node contact data from the UMass Trace Repository [4] and the Reality Mining project [5] to evaluate our Hybrid protocol. The UMass data that we used was collected on 26, 27, and 28 March 2007 and includes bus and access point contact data. The Reality Mining data that we used was collected in October 2004 and includes participant Bluetooth mobile phone and cell contact data. We chose a small number of consecutive days from all the available UMass data to get an initial understanding of how the Hybrid protocol would behave and then randomly chose a whole month from all the available Reality Mining data to more systematically evaluate it. The UMass and Reality Mining data used have an average per day of approximately 25 nodes and 250 contacts, and 300 nodes and 4000 contacts, respectively.

Running the Hybrid protocol on this data proved to be a lengthy process and due to restrictions on available resources we decided to use only the morning of each day, from 6am to 12pm. All contact data was introduced in a MySQL database and appropriate simulation files automatically generated for DTNSim – DTNSim is the DTN2 simulation platform that uses the same DTN router code used for non-simulation prototypes.

C. Experiments with the Epidemic and Prophet protocol on the logged data

In order to explore the metrics and the logged data, we ran the Epidemic and Prophet protocols on the three days of the UMass data and on 10 days of the Reality Mining data. Figures 1 and 2 show the data generated from these experiments. Notice how Prophet has lower arrivals in all days of both figures – as expected given that Epidemic is optimal regarding packet delivery; how the arrivals, buffer cost, and transmission cost in the UMass data in figure 1 are approximately 10 times larger than those of the Reality Mining data in figure 2; and how data significantly changes from one day to the other in both figures.

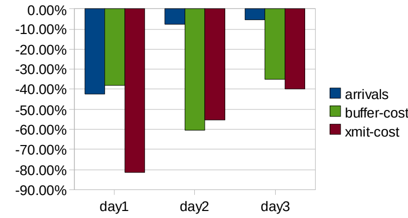

Fig. 1. Arrivals, buffer cost, and transmission cost of the Prophet protocol relative to Epidemic (3 days of UMass data)

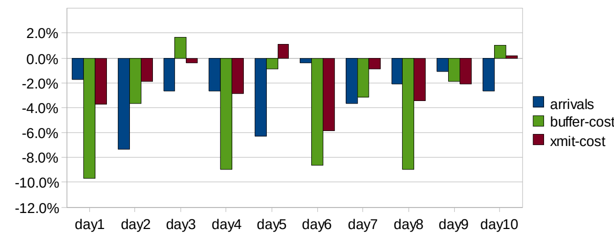

Fig. 2. Arrivals, buffer cost, and transmission cost of the Prophet protocol relative to Epidemic (10 days of Reality Mining data)

IV. Results

To get an idea of how the Hybrid protocol behaves, we started by running the protocol on the three days of the UMass data with threshold values ε ranging from 0 (equivalent to the Prophet protocol) to 2, with 0.2 increments. We also ran the Epidemic protocol on the same days, resulting in the maximum number of delivered packets by 12pm being 40, 108, and 89 for days 1, 2, and 3 respectively. The data gathered from this initial experiment is shown in figures 3-5.

By 12pm on day 1 Prophet delivers 23 packets, a reduction of 42.5% compared to Epidemic's packet delivery (40 packets), yet has 38.2% lower buffer space cost (a reduction from 5.58 Kbytes to 3.45 Kbytes of total buffer space per packet delivered) and 81.4% lower transmission cost (a reduction from 2.42 Kbytes to 0.45 Kbytes transmitted for each packet that is delivered). In the same conditions, Hybrid with $\varepsilon=0.4$ delivers the maximum number of packets (40), uses 2.72 Kbytes total buffer space per delivered packet, and has 0.68 Kbytes transmitted for each delivered packet. This corresponds to 43.9% and 60.8% lower buffer space and transmission costs, respectively, than Epidemic.

On this day, Hybrid offers better or equal performance than Epidemic when comparing packet delivery and buffer space and transmission costs. Hybrid also offers better performance than Prophet on this day when comparing packet delivery (73.9% more packets) and buffer space cost (9.3% lower total buffer space per delivered packet). However, Prophet has better performance than Hybrid when we compare transmission costs (52.6% lower transmission costs than Hybrid). When directly comparing the Hybrid and Prophet protocols on this data, tradeoffs between packet delivery, buffer space, and transmission must be considered.

For the values of ε between 0.4 and 2 that we tested on day 1 and day 2, Hybrid is able to deliver the maximum number of packets. This is not the case on day 3 regardless of the value of ε that we tested, although Hybrid does present lower buffer space and transmission costs than Epidemic. These initial results were a motivation for running the Hybrid protocol on a larger set of data in order to get more insights on its behavior; results follow.

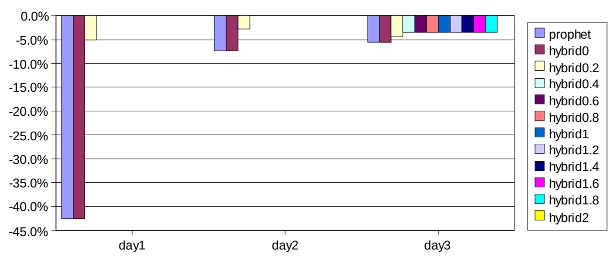

Fig. 3. Arrivals of the Prophet and Hybrid ($0 < \varepsilon < 2$, 0.2 intervals) protocols relative to the Epidemic protocol on 3 days of UMass data.

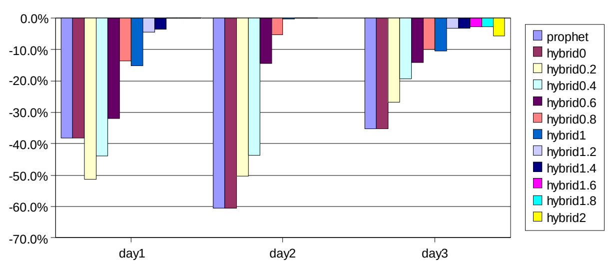

Fig. 4. Same as figure 1 except showing buffer cost.

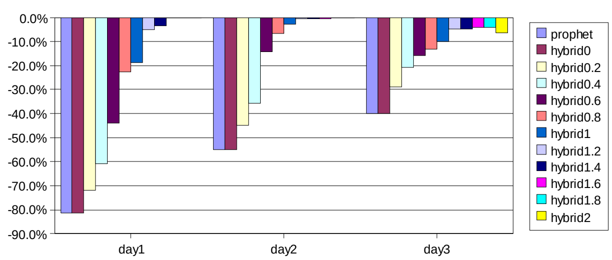

Fig. 5. Same as figure 1 except showing transmission cost.

A. Threshold decision

To help choose a switching threshold for using Hybrid on the Reality Mining data set, we started by running both Epidemic and Hybrid protocols on the first 10 days of the Reality Mining data with ε = 0.2, 0.4, 0.5, and 0.6. We did not observe as much changes of the average of arrivals, buffer cost, and transmission cost with varying ε as we did with the UMass data. A differentiating factor is that in 7 out of 10 days, only ε=0.5 and ε=0.6 delivered the maximum number of packets; from these two values we arbitrarily selected ε =0.5.

B. Performance results

We ran 31 days of logged contact data from the Reality Mining project using the Epidemic, Prophet, and Hybrid ε=0.5 protocols. Table 1 presents the average of these metrics for the three protocols throughout the 31 days. This table also shows the Hybrid to Epidemic and Hybrid to Prophet ratios averaged over the 31 days. Although the average Hybrid to average Epidemic ratios that can be obtained by dividing the average values shown in the table yield similar values, these are conceptually different from ratio averages which we believe are better suited for comparing these protocols.

We observe that on average Hybrid ε=0.5 has 4.75% and 1.79% lower buffer space and transmission costs compared to Epidemic while failing to deliver only 0.52% of the maximum number of packets. Compared to Prophet, Hybrid ε=0.5 delivers 3.01% more packets while having only 0.34% and 0.16% more buffer space and transmission costs. We can also observe that the difference between the average number of delivered packets registered on this data set for Prophet and Epidemic is 23 out of 676. This is an order of magnitude smaller than the 17 out of 40 that we observed on the first day of the UMass data and may explain why the performance of the Hybrid protocol using the small data set of the UMass data was much better than using the Reality Mining results.

Table I
Hybrid vs. Epidemic and Prophet protocols on the 31 days of Reality Mining data.

	Delivered Packets (# Packets)	**Buffer Cost (Kbytes / Packet)**	**Xmit cost (Kbytes / Packet)**
Averages:			
Epidemic	676	1.63	5.57
Prophet	653	1.57	5.46
Hybrid	672	1.57	5.47
Hybrid relative to:			
Epidemic	99.48%	95.25%	98.21%
Prophet	103.01%	100.34%	100.16%
Hybrid better(+) or worse(-) than:			
Epidemic	-0.52%	4.75%	1.79%
Prophet	3.01%	-0.34%	-0.16%

V. CONCLUSIONS

Results show that when compared to its original protocols in a set of logged contact data, the hybrid flooding-probabilistic DTN protocol proposed in this paper has improved performance on some metrics while only slightly underperforming in others. These results suggest that probabilistic DTN routing may benefit from switching to flooding at times when contacts are scarce and, more generically, point to further work on dynamically adjusting the type of DTN protocol used according to the varying contact characteristics of the network.

REFERENCES

[1] S. Jain, K. Fall, and R. Patra. "Routing in a delay tolerant networking." *SIGCOMM*, Aug/Sep 2004.

[2] K. Harras, K. Almeroth, and E. Belding-Royer. "Delay tolerant mobile networks (DTNMS): Controlled flooding schemes in sparse mobile networks." *IFIP Networking*, May 2005.

[3] A. Lindgren, A. Doria, and O. Schelen. "Probabilistic routing in intermittently connected networks". *Workshop on Service Assurance with Partial and Intermittent Resources (SAPIR 2004)*, August 2004.

[4] X. Zhang, J. Kurose, B. Levine, D. Towsley, and H. Zhang. "Study of a bus-based disruption tolerant network: Mobility modeling and impact on routing." *ACM International Conference on Mobile Computing and Networking (Mobicom)*, September 2007.

[5] A. Miklas, K. Gollu, K. Chan, S. Saroiu, K. Gummadi, and E. de Lara. "Exploiting social interactions in mobile systems". *UbiComp'07*, September 2007.