Toward a Unified Routing Framework for Delay-Tolerant Networking

S. Burleigh§, C. Caini°, J. J. Messina°, M. Rodolfi°

§NASA -Jet Propulsion Laboratory California Institute of Technology, Pasadena, CA, USA

°DEI/ARCES, University of Bologna, Italy

scott.burleigh@jpl.nasa.gov, carlo.caini@unibo.it, jakojo.messina@studio.unibo.it, michirod@gmail.com

***Abstract*— Routing in Delay-/Disruption-Tolerant Networking (DTN) has long been recognized as a challenging research topic. The difficulty lies in the fact that link intermittency and network partitioning, possibly coupled with long delays, prevent the use of Internet solutions based on an up-to-date comprehensive knowledge of network topology, as communicated by routing protocols. In the literature on DTN routing, there is a dichotomy between solutions designed for deterministic (e.g., space flight) networks, such as Contact Graph Routing (CGR), and the wide variety of protocols designed for opportunistic terrestrial networks. After a discussion of the origin and motivations of this duality, the paper presents an opportunistic extension of CGR (OCGR). The aim is to try to resolve the DTN routing dichotomy by providing a unified approach suitable for all DTN environments.**

Keywords-component; Routing, Delay-/Disruption- Tolerant Networking, CGR, Challenged Networks, Space Networks.

I. Introduction

Delay-/Disruption- Tolerant Networking (DTN) evolved from InterPlanetary Networking research, when became clear that the problems to be faced in space communications had much in common with terrestrial "challenged networks" (mobile ad-hoc networks, emergency networks, sensor networks, tactical networks, underwater networks, etc.). In challenged networks the ordinary TCP/IP architecture and related protocols cannot provide satisfactory performance, because of the presence of at least one of the following impairments: long delays, communication disruptions, high error rates, asymmetric link rates, and lack of end-to-end connectivity. The aim of the DTN architecture [1], [2], based on the introduction of the Bundle protocol layer between Application and lower layers, is to offer a common general solution instead of a variety of specific solutions limited in scope. The application of DTN architecture and of the related Bundle Protocol (BP) [3] makes possible communication in challenged networks, which thus become DTN networks.

Shortly after the development of DTN architecture began, it was recognized that routing in DTN-based networks would be a central research challenge [4]. The difficulty lies in the fact that, while routes in the Internet may be computed based on contemporaneous and nearly comprehensive knowledge of network topology as communicated by routing protocols, DTN routes cannot. By the very nature of DTN, it can never be assumed that information about changes in topology has been distributed to the relevant network entities rapidly enough to be relied upon in route computation: the length of time that a given item of topology information remains true may commonly be less than the length of time required to propagate that information to all route-computing nodes, due either to long delays and intermittent connectivity, as in space networks, or to network partitioning as in terrestrial DTNs. A different approach is required.

Many routing algorithms for DTNs have been proposed, investigated in simulation, and in some cases tested in operation, but the field remains generally open: no single routing system has emerged as the consensus choice of the DTN research and deployment community, in part because the constraints on route computation are very different in different DTN deployment environments [5]. As discussed later, space networks are characterized by intermittent scheduled connectivity: opportunities for of transmission between nodes are known in advance, and paths are thus deterministic. By contrast, most terrestrial DTNs are characterized by random intermittent connectivity, as contacts typically arise from casual encounters.

Contact graph routing (CGR) [6] is possibly the sole DTN routing algorithm designed to cope with deterministic scheduled connectivity, while for opportunistic networks there are many proposed schemes; see [7] for a survey of the field. Here we will summarize only a few of the most widely studied approaches to give the reader at least an introductory idea of opportunistic routing mechanisms.

In Epidemic routing [8], mobile nodes simply forward each *bundle* (i.e., Bundle Protocol data unit; in effect, a message) to all nodes with which they come into contact other than those that have already received a copy of that bundle. The probability of delivering the bundle to its destination is maximized but transmission overhead is high.

To limit the high transmission overhead that characterizes raw Epidemic routing, Spray-and-wait [9] stops flooding a given bundle after a given number of copies of that bundle have been forwarded.

Similarly, the PRoPHET [10] system reduces transmission overhead by forwarding a copy of a given bundle only to nodes that are expected to have higher probability of delivering the bundle to its destination, as derived from propagated encounter history information.

In this paper we propose Opportunistic CGR (OCGR), a simple extension to Contact Graph Routing aimed at enlarging its applicability from deterministic space networks to opportunistic terrestrial networks. The idea, and the hope, is that OCGR could serve as a unified routing framework encompassing all DTN environments. A few preliminary results, obtained by means of the ONE DTN simulator [11], are presented.

The paper is organized as follows: the DTN routing problem is discussed in Section II; CGR and OCGR in Sections II and IV; simulation results are presented in Section V; and conclusions are drawn in Section VI.

II. DTN ROUTING

It is possible to regard DTN routing not as a single problem but as two quite different problems distinguished by the nature of transmission opportunities in the network.

Most DTN research to date has focused on "terrestrial" networking, where signal propagation latencies are very small but delivery latencies may be arbitrarily large due to lapses in end-to-end connectivity. Where such lapses are the transient network partitions that result when fixed Internet infrastructure temporarily fails in place, DTN can usefully preserve information flow and routing is not an issue: existing Internet routes remain valid, though momentarily inaccessible. The real routing problem in this environment arises from network partitions caused by the mobility of Internet devices in nominal operation, communicating by means of radio interfaces with limited range, such as Bluetooth, or WiFi. The physical movement of these nodes continually changes the topology of the network, as communications between a pair of nodes (or a node and the infrastructure) is possible only when they are mutually in communication range. Messages are exchanged not over fixed, known paths but rather over paths that emerge spontaneously from unplanned episodes of node proximity. Routing over such paths is sometimes termed *opportunistic DTN routing* and is non-deterministic, due to the analogous nature of node mobility.

The other prominent domain of DTN research is "space networking", where lapses in end-to-end connectivity are again routine – and moreover signal propagation latencies may be very large (e.g., 1.2 seconds one-way propagation delay from Earth to Moon; from 3 to 21 minutes from Earth to Mars) – but where paths may be assembled from planned episodes of radio connectivity. Episodes of connectivity ("contacts") can be planned because (a) the orbital movements of network nodes in space are well understood and (b) the operations of space flight assets, including the exercise of their radios, are typically scheduled in detail to maximize functional impact within severe resource constraints. Routing over such paths may be termed *schedule-aware DTN routing*.

In general, for a given network node, we might say that to "route" data in the network (in "unicast" fashion; multicast routing is beyond the scope of this paper) is simply to answer the following two questions once for each outbound data item, for each opportunity to transmit directly to some other network node (that is, for each contact):

1. Do I transmit a copy of this data item during this contact?
2. Do I continue considering additional opportunities to transmit copies of this data item?

For Internet routing, where contacts are continuous and are known with relative certainty, the node consults propagated routing information in order to compute the optimum route through the current known network topology and, if the contact under consideration is the first contact in that route, then the answer to (1) is yes and to (2) no; otherwise the answer to (1) is no and to (2) yes.

For opportunistic DTN routing, the impossibility of timely distribution of current network topology information makes this approach untenable; some other basis for answering these questions must be adopted. In the extreme, a flooding strategy stipulates that the answers are always (1) yes (except when the contact is with a node that is known to have already had a copy of this data item) and (2) yes. Such a strategy minimizes delivery delay and maximizes success (at least in an uncongested network) but has the obvious drawback of generating a high volume of unnecessary transmission. For this reason, the opportunistic routing systems developed to date have been based on a variety of plausible heuristics, all aimed at reducing the volume of transmission without too severely reducing the rate of end-to-end data delivery or increasing the delivery delay; for these schemes the answers to routing questions (1) and (2) are typically a function of the previously computed answers.

For schedule-aware DTN routing, advance planning makes it possible to know the network topology a priori, despite the absence of Internet-like routing protocols. In this context, contact graph routing (CGR) [6] behaves somewhat like Internet routing: the node consults a schedule of planned contacts in order to compute the optimum route through the network topology as it will vary over the near future and, if the contact under consideration is the first contact in that route, then the answer to (1) is yes and to (2) no; vice versa otherwise.

In this paper we propose that relaxing the requirement for certainty in knowledge of future network topology could enable CGR to be applied to opportunistic DTN routing as well. Specifically, we suggest that the insertion of predicted contacts into the topology forecast can enable CGR to make good opportunistic DTN routing decisions

III. CONTACT GRAPH ROUTING

Routing in the Internet may be viewed as analogous to planning a road trip: links (analogous to highway segments) form the arcs of a graph; hosts and routers (analogous to towns and highway interchanges) form the vertices; costs are associated with traversing each of the arcs, and the problem is to find the lowest-cost route from one graph vertex to some other graph vertex.

For Internet routing this model works well because both highway topology and Internet topology are generally time-insensitive: the connections between hosts/routers are generally continuous and of notionally unlimited capacity, at least for the duration of any single graph traversal, just as highways are very likely to be in place and open to all potential drivers throughout the duration of any single road trip. But in a DTN-based space network the connections between nodes may routinely appear and disappear at scheduled times, and there may never be continuous connectivity from a bundle's source all the way through to its destination at any moment.

Contact graph routing is instead similar to booking airline flights for a business trip. A single airline flight constitutes transit from some identified airport to some other identified airport, characterized by departure time, arrival time, and the number of passengers the aircraft can carry. The problem is to select, for each traveler, a sequence of flights that results in the earliest final arrival time, regardless of which airports are on the route. The airports constrain the selection of flights – the traveler cannot land in Nashville and then take off from Frankfurt – but they are not the vertices of the graph. The flights are the vertices of the graph, and the arcs of the graph are the connections between flights, i.e., the periods of time during which a traveler arriving on one flight must wait before departing on the next.

Similarly, a DTN network contact constitutes transmission from some identified node to some other identified node, characterized by transmission time, reception time, and *volume* – the maximum amount of data that can be transferred during the contact, given by the difference between contact start time and contact stop time, multiplied by the transmission data rate. The problem is to select, for each bundle, a sequence of contacts that results in the earliest final arrival time, regardless of which nodes are on the route. The nodes constrain the selection of contacts – the bundle cannot be received at node A and then be transmitted from node B – but they are not the vertices of the graph. The contacts are the vertices of the graph, and the arcs are the periods of time when a bundle resides in storage at some node while awaiting the next transmission opportunity. See Figures 1-2 and Table I for an illustration.

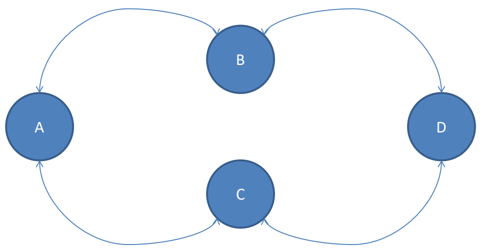

Figure 1: Network Topology Example

TABLE I. Contact Plan Example

Contact	*Sender*	*Receiver*	*From time (s)*	*Until time (s)*	*Rate (kbps)*
1	A	B	1000	1100	1000
2	B	A	1000	1100	1000
3	B	D	1100	1200	1000
4	D	B	1100	1200	1000
5	A	C	1100	1200	1000
6	C	A	1100	1200	1000
7	A	B	1300	1400	1000
8	B	A	1300	1400	1000
9	B	D	1400	1500	1000
10	D	B	1400	1500	1000
11	C	D	1500	1600	1000
12	D	D	1500	1600	1000

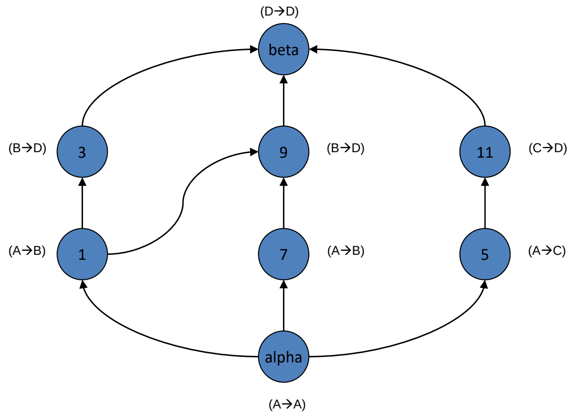

Figure 2: Resulting Contact graph for A→D Transmission

For a detailed explanation of CGR please see [6]. Very briefly:

- At each node, for each bundle destination, the comprehensive list of all scheduled contacts among nodes in the network (the *contact plan*, constructed by network management, analogous to the union of all airlines' flight schedules) is searched in order to compute all plausible routes to destination. The receiving node for the first contact in each route is termed the route's *entry node*. For each node that is an entry node, the route through that node which offers the earliest final arrival time is deemed the best route through that node.
- Each bundle is queued for transmission to the entry node whose best route offers the earliest arrival time, subject to various constraints including the bundle's priority and the prior claims on the volumes of the contacts on that route (bundles previously queued for transmission to that entry node).

IV. Opportunistic Contact Graph Routing

Opportunistic Contact Graph Routing (OCGR) is still a work in progress, and all elements of the design clearly remain open to discussion and revision. For now, OCGR is defined as follows.

To extend CGR in support of opportunistic routing, we extend the contact plan in two ways:

- Non-scheduled contacts may be automatically discovered in real time, offering immediate connectivity to newly discovered neighboring nodes. When these discovered contacts end, their start and stop times and volumes are recorded in a contact log.
- Our *confidence* in both scheduled and discovered contacts is always 1, but the contact plan may also include predicted contacts in which we have much less confidence.

Additionally, we note for each outbound bundle our confidence that the forwarding activities performed so far will result in delivery of the bundle at its destination prior to bundle expiration. This bundle delivery confidence value is initialized to 0.

In the course of the initial handshaking for any newly discovered contact, the communicating nodes exchange all contact log entries. They then discard all previously computed predicted contacts and use the updated contact history to compute new predicted contacts. Contact prediction is performed for each sender/receiver node pair for which contact log entries exist:

- Mean and standard deviation are computed for the durations of the applicable contact log entries and also for the durations of the gap periods between those logged discovered contacts.
- Our *base confidence* B in contact prediction for this node pair is High (a tuning parameter value in the range 0 to 1, currently .2) if the standard deviations for contact and gap duration are less than the corresponding means, otherwise Low (another tuning parameter value, less than High, currently .05). That is, our confidence in the prediction is higher if contact history exhibits a degree of regularity than if it does not.
- A single predicted contact is inserted. That contact's start time is the current time, its duration is computed as the current time less the start time of the earliest applicable log entry, and its data rate is the sum of the volumes of all applicable log entries, divided by the duration of the predicted contact. Our net confidence C in this predicted contact is given by $C = 1 - (1 - B)^N$ where N is the number of applicable log entries.

The result is a contact plan that can be used for contact graph routing in the usual way, except that our confidence in the resulting forwarding decisions is less than total. That is, bundles are forwarded as follows at each of the communicating nodes:

- The updated contact plan, including all newly added discovered and predicted contacts, is used to compute all plausible routes to all destinations. Our confidence D in a given route is the product of the computed confidences for all contacts in that route.
- All bundles that are not currently queued for transmission to any entry node are re-examined. For each such bundle, if the newly discovered peer neighboring node is the entry node for at least one *pertinent route* (a route that results in arrival of the bundle at the destination prior to its expiration), then (a) the bundle is queued for transmission to that node and (b) bundle delivery confidence K is increased by increment J, given by $J = 1 – ((1 – K) * (1 – D))$ where D is the maximum confidence value among all pertinent routes for which this node is the entry node. When a bundle's delivery confidence reaches a predefined threshold (another tuning parameter, currently .8), further attempts to forward this bundle cease.

V. SIMULATION

OCGR has been implemented in an experimental version of the ION DTN package [12], and that implementation has been integrated into the ONE DTN simulator [11]. To emphasize this point: the native ION CGR software (including the OCGR adaptations), written in C, has been imported directly into the Java-based ONE simulator, without modification, by means of Java Native Interface (JNI) classes. CGR is not simulated in ONE, it is executed. Our findings from ONE-based simulations will be directly applicable to operational deployment of OCGR.

At the time of this writing, the study of OCGR in simulation has only begun: just two comparative simulations have been completed, one for 50 kB bundles and one for 100 kB bundles. Key parameters of the simulations were as follows:

- A network of 15 nodes was simulated: one group of 5 autos and two groups of 5 pedestrians each. Mobility characteristics of both autos and pedestrians, as well as road topology, were left to ONE defaults.
- Buffer size at each node was set to 5MB.
- Radio range for each node was set to 10 meters (i.e. we chose short range interfaces).
- Transmission rate on each contact was set at 2 Mbps.
- Once each 40-60 seconds a single bundle with randomly selected source and destination was inserted into the network, 436 bundles in all.

Four alternative routing schemes were compared: Epidemic routing, PRoPHET, base CGR, and opportunistic CGR. Note that base CGR was included only as a performance baseline, as we provided it with a comprehensive contact plan that gave advance knowledge of all contacts that would occur during the simulation. Although this would of course be impossible in actual network operations, it gave us a loose upper bound on achievable performance. Results for the two simulations are shown in Table II and Table III.

TABLE II. Transmission of 50KB Bundles

	Epidemic	*PRoPHET*	*Base CGR*	*OCGR*
Bundles started	2810	1795	829	1206
Bundles relayed	2732	1746	819	1177
Bundles aborted	78	49	10	29
Bundles dropped	1964	1153	448	1006
Bundles delivered	303	299	346	133
Delivery probability	0.6950	0.6858	0.7936	0.3050
Overhead ratio	8.0165	4.8395	1.3671	7.8496
Average latency (s)	4356.8426	4525.5709	3805.8419	5570.3684
Median latency (s)	3671.3000	4037.0000	3364.2000	4988.0000
Average hop count	1.9142	1.7592	2.2688	2.0752

TABLE III. Transmission of 100KB Bundles

	Epidemic	*PRoPHET*	*Base CGR*	*OCGR*
Bundles started	1660	1324	780	918
Bundles relayed	1525	1223	742	845
Bundles aborted	135	101	38	73
Bundles dropped	1188	897	474	743
Bundles removed	0	0	0	0
Bundles delivered	246	251	283	104
Delivery probability	0.5642	0.5757	0.6491	0.2385
Overhead ratio	5.1992	3.8725	1.6219	7.1250
Average latency (s)	4201.5045	4166.6171	4680.5675	5579.9510
Median latency (s)	3566.8000	3667.2000	4193.0000	4988.9000
Average hop count	1.6667	1.5458	2.2049	1.7212

On all measures, base CGR yielded the best performance in these simulations. It was able to outperform even Epidemic routing because not all contacts lasted long enough to enable the exhaustive bundle exchange on which the Epidemic algorithm relies. The high performance of base CGR is unsurprising, as the algorithm was given perfect knowledge on which to base its forwarding decisions. Such a configuration is obviously unrealistic, serving only as a performance target against which to evaluate OCGR.

As anticipated, OCGR was more economical in its forwarding decisions than either Epidemic Routing or PRoPHET, but because its rate of successful delivery was so low its overhead ratio was very high. We suspect that OCGR performance can be significantly improved with minimal effort by:

- Adjusting the values of the various tuning parameters.
- Omitting from performance statistics collection the period during which the algorithm is navigating its "learning curve", i.e., acquiring sufficient contact history to make informed routing decisions. (Until at least minimal history is acquired, no bundles are forwarded at all.)

On this latter point, we speculate that exercising Epidemic routing during this learning period, enabling nodes to make the transition from Epidemic routing to OCGR as they accumulate contact log records, may prove to be a successful deployment strategy.

VI. Conclusions and Future Work

The addition of opportunistic forwarding procedures to CGR is still in its infancy and has not yet been shown to improve network performance beyond the benchmarks established by other systems for opportunistic DTN routing. As noted above, however, we believe that such improvement remains possible. We therefore continue this investigation because the potential reward seems compelling: if OCGR proves effective in opportunistic routing environments, it should be possible to deploy over any DTN-based network a single system that unifies scheduled, discovered, and continuous network contacts in a seamless efficient routing fabric. We believe this is a goal worth pursuing.

Acknowledgments

Part of this research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

References

[1] S. Burleigh, A. Hooke, L. Torgerson, K. Fall, V. Cerf, R. Durst, K. Scott, and H. Weiss, "Delay-tolerant networking: an approach to interplanetary Internet," IEEE Communications Magazine, vol. 41, no. 6, June 2003, pp. 128-136.

[2] V. Cerf , A. Hooke, L. Torgerson, R. Durst, K. Scott, K. Fall, H. Weiss "Delay-Tolerant Networking Architecture", Internet RFC 4838, Apr. 2007.

[3] K. Scott, S. Burleigh, "Bundle Protocol Specification", Internet RFC 5050, Nov. 2007.

[4] Jain, K. Fall, and R. Patra, "Routing in a Delay Tolerant network", in Proc. of ACM SIGCOMM 2004, Portland, Aug/Sept. 2004, pp. 145-157.

[5] C. Caini, H. Cruickshank, S. Farrell, M. Marchese, "Delay- and Disruption-Tolerant Networking (DTN): An Alternative Solution for Future Satellite Networking Applications", Proceedings of IEEE, Vol. 99, N. 11, pp.1980-1997, Nov. 2011.

[6] G. Araniti, N. Bezirgiannidis, E. Birrane, I. Bisio, S. Burleigh, C. Caini, M. Feldmann, M. Marchese, J. Segui, and K. Suzuki, "Contact graph routing in DTN space networks: overview, enhancements and performance," IEEE Communications Magazine, vol. 53, no. 3, March 2015, pp. 38-46.

[7] R. J. D'Souza and Johny Jose, “Routing Approaches in Delay Tolerant Networks: A Survey”, International Journal of Computer Applications (0975 - 8887), Volume 1 – No. 17, 2010.

[8] Amin Vahdat and David Becker, “Epidemic routing for partially connected ad hoc networks”, Technical Report CS-2000-06, Department of Computer Science, Duke University, April 2000.

[9] T. Spyropoulos, K Psounis, and C. S. Raghavendra, “Spray and wait: An efficient routing scheme for intermittently connected mobile networks”, in Proc. of 2005 ACM SIGCOMM workshop on Delay-tolerant networking, WDTN’05, 2005, pp. 252-259.

[10] A. Lindgren, A. Doria E. Davies, and S. Grasic, “Probabilistic Routing Protocol for Intermittently Connected Networks”, Internet RFC 6693, Aug. 2012.

[11] A. Keränen, J. Ott, and T. Kärkkäinen, “The ONE Simulator for DTN Protocol Evaluation”, in Proceedings of the 2nd International Conference on Simulation Tools and Techniques. New York, NY, USA: ICST, 2009.

[12] S. Burleigh, “Interplanetary overlay network design and operation V3.3.1,” JPL D-48259, Jet Propulsion Laboratory, California Institute of Technology, CA, May 2015. [Online]: http://sourceforge.net/projects/ion-dtn/files/latest/download