Toward a Unified Routing Framework for Delay-Tolerant Networking

S. Burleigh, Jet Propulsion Laboratory, California Institute of Technology

C. Caini, J. J. Messina, and M. Rodolfi, DEI/ARCES, University of Bologna

26 September 2016

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. (c) 2016 California Institute of Technology. Government sponsorship acknowledged.

Problem: Routing in a DTN

- Routes in the Internet can be computed from current knowledge of network topology, noted by routing protocols.
- In a network built on Delay-Tolerant Networking (DTN), no node's network topology knowledge is sure to be current.
  - Propagation of new topological data via routing protocols may take so long that by the time the information is received it is no longer true.
- If future changes are ***planned***, then propagation of the contact plan can close before the changes occur. Routes are computed from this knowledge: Contact Graph Routing (CGR).
- But most DTN routing must accommodate ***discovered*** contacts, resulting in opportunistic forwarding: Epidemic, Spray and Wait, PRoPHET, and others.

Opportunistic CGR

- Hypothesis: if a contact plan includes not only planned contacts but also discovered and ***predicted*** contacts, then the CGR procedures can support opportunistic forwarding.
- Motivation: if CGR can support both scheduled and opportunistic forwarding, then this single routing framework can be universally deployed in DTN-based networks.
  - Simpler implementation and deployment.
  - More predictable network behavior, easier troubleshooting.

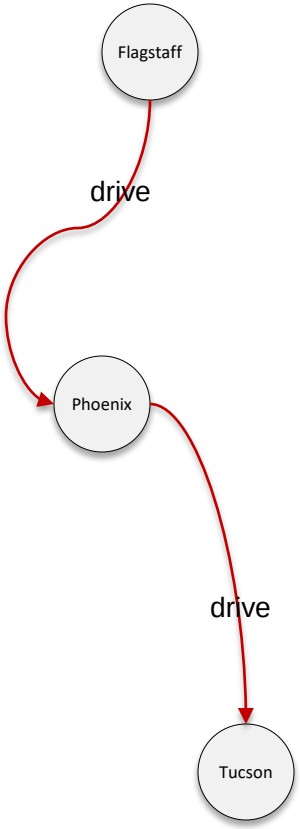
What is routing?

- For each outbound bundle, for each transmission opportunity (i.e., contact with topologically adjacent node), we must answer two questions:
  - Do I transmit a copy of this bundle in the course of this contact?
  - Do I discard my local copy of the bundle, or do I retain it for possible transmission in the course of other contacts?

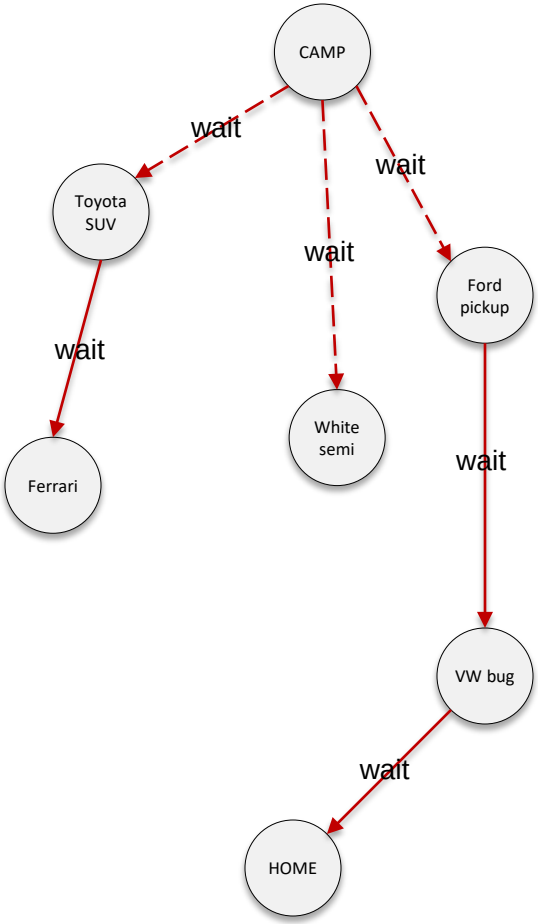
	Internet routing		Spray & Wait, etc.	Contact Graph Routing		
	Current path to destination via this node	**No current path to destination via this node**	**Discovered contact with neighbor**	**Future path to destination via this node**	**No future path to destination via this node**	**Discovered contact with neighbor**
Transmit?	yes	no	maybe	yes	no	maybe
Retain?	no	If more neighbors, try again; else no.	maybe	no	yes	maybe

Different Metaphors

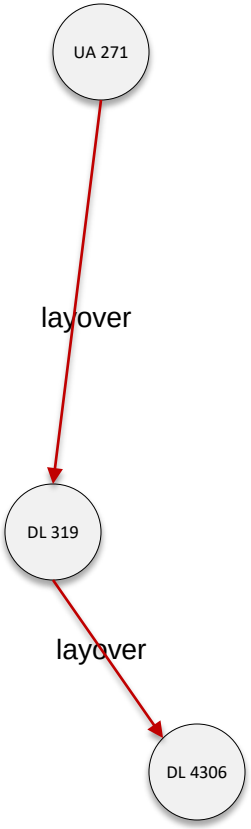
Internet routing:
like a road trip



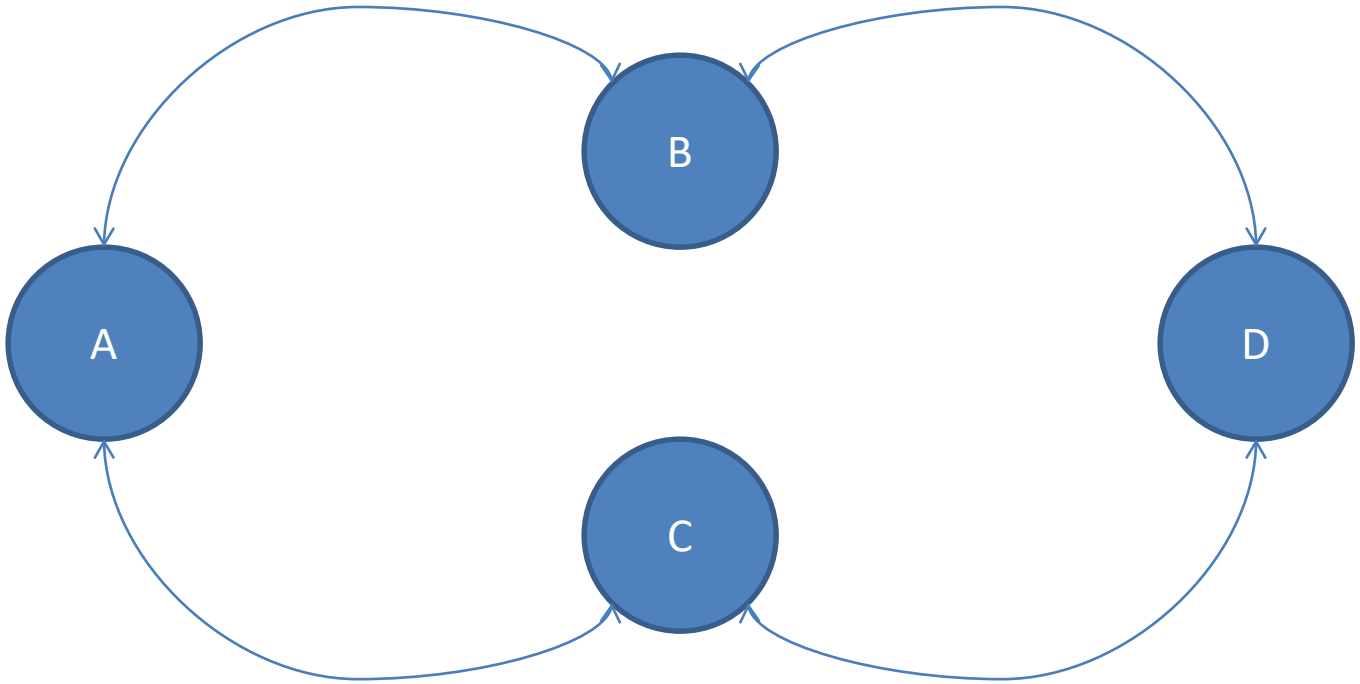
Spray & Wait:
like hitch hiking



CGR: like airline
connections



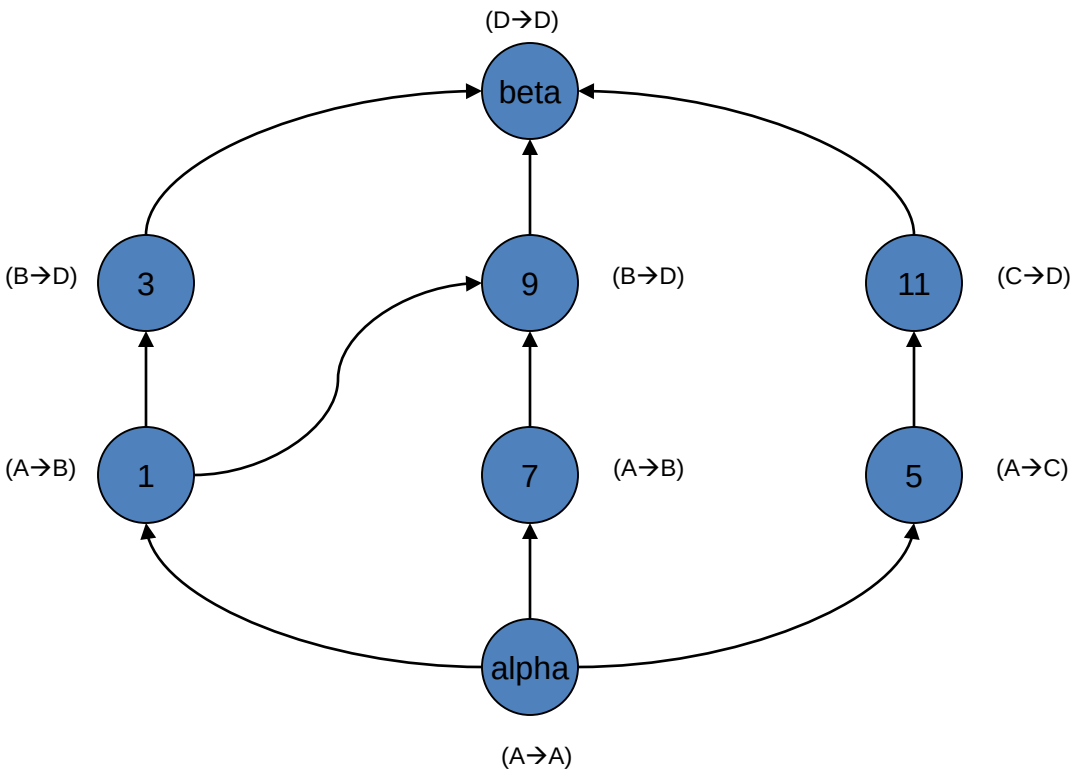
Sample Network Topology



Contact Plan for this network

Contact	Sender	Recvr	From	Until	Rate
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

Contact Graph for A→D traffic



Opportunistic CGR (OCGR)

- Basic CGR procedure:
  - On any contact plan change, re-compute all routes in the contact graph and rank computed routes by earliest arrival time.
  - For each bundle, select the best route subject to route backlog and bundle priority; forward via the first contact in that route.
- OCGR extensions:
  - Log all discovered contacts; exchange contact logs among nodes.
  - “Doubt” level D for each scheduled or discovered contact is 0.
  - From contact log, compute predicted contacts between pairs of nodes. Doubt for a predicted contact is between .10 and .99.
  - For each bundle, on each discovered contact, forward via this contact if net doubt for the best route through that contact (based on computed doubt for all other contacts on the route) is low.

Evaluation in the ONE simulator

Simulation parameters:

- Network of 15 simulated nodes, one group of 5 autos and two groups of 5 pedestrians each. 5 MB of buffer space at each node. Radio range 10 m. Transmission rate 2 Mbps.
- 6 hours of simulated time; 436 bundles of 100KB each generated with random source and destination.
- OCGR v.1 used original doubt computation algorithm, tested in June. OCGR v.2 used an improved algorithm, tested in August.

	Epidemic	PRoPHET	Base CGR	OCGR v.1	OCGR v.2
Bundles started	1660	1324	780	918	303
Bundles relayed	1525	1223	742	845	291
Bundles aborted	135	101	38	73	12
Bundles dropped	1188	897	474	743	97
Bundles removed	0	0	0	0	0
Bundles delivered	246	251	283	104	180
Delivery probability	0.5642	0.5757	0.6491	0.2385	0.4128
Overhead ratio	5.1992	3.8725	1.6219	7.1250	0.6167
Average latency (s)	4201.5045	4166.6171	4680.5675	5579.9510	4755.1167
Median latency (s)	3566.8000	3667.2000	4193.0000	4988.9000	3567.2000
Average hop count	1.6667	1.5458	2.2049	1.7212	1.1278

Outlook and Conclusions

- OCGR has been shown to be functional, successfully delivering bundles to their destinations.
- Performance of the first-generation doubt computation algorithm was poor (low delivery probability, high overhead).
- Second-generation, though still lagging PRoPHET and Epidemic, is much better. Improvements in the algorithm are continuing.
- We think it remains plausible that a further optimized doubt computation algorithm will make OCGR a competitive routing solution for opportunistic forwarding, enabling a unified DTN routing framework.

Thanks!

- Questions?