

# Estimation based Erasure-coding Routing in Delay Tolerant Networks

Yong Liao<sup>†</sup>, Kun Tan<sup>\*</sup>, Zhensheng Zhang<sup>‡</sup>, Lixin Gao<sup>†</sup>

<sup>†</sup>Dep. of Electrical & Computer Eng.  
University of Massachusetts  
Amherst, MA 01003, USA  
email: {yliao, lgao}@ecs.umass.edu

<sup>\*</sup>Microsoft Research Asia  
Sigma Center, No. 49 Zhichun Rd  
Beijing 100080, China  
email: kuntan@microsoft.com

<sup>‡</sup>San Diego Research Center  
Suite A, 6696 Mesa Ridge Rd  
San Diego, CA 92121, USA  
email: zzhang@ieee.org

**Abstract**— Wireless Delay Tolerant Networks (DTNs) are intermittently connected mobile wireless networks. Some well-known assumptions of traditional networks are no longer true in DTNs, which makes routing in DTNs a challenging problem. We observe that mobile nodes in realistic wireless DTNs may always have some mobility pattern information which can be used to estimate one node’s ability to deliver a specific message. This estimation can greatly enhance the routing performance in DTNs. Furthermore, we adopt an alternative way to generate redundancy using erasure coding. With a fixed overhead, the erasure coding can generate a large number of message-blocks instead of a few replications, and therefore it allows the transmission of only a portion of message to a relay. This can greatly increase the routing diversity when combined with estimation-based approaches. We have conducted extensive simulations to evaluate the performance of our scheme. The results demonstrate that our scheme outperforms previously proposed schemes.

## I. INTRODUCTION

Delay Tolerant Networks (DTNs) [1] are intermittently connected mobile wireless networks. Examples of this type of network include inter-planetary networks [1] [2], and wildlife tracking and habitat monitoring sensor networks [3] [4]. In the typical DTN scenario, nodes are mobile and have wireless networking ability. The connectivity between nodes change frequently due to the movement of nodes.

Generally, the DTN routing should work in an extended store-and-forward way. When two nodes encounter and a wireless link is established between them, messages can be sent over this link. Nodes cache the messages before getting the chance to send them to the next hop nodes. In this way, the messages are forwarded from node to node until they reach the destinations.

Because the exact contact information between any two nodes in DTN cannot be known in advance, it is very challenging to make routing decisions. Many existing approaches dispatch multiple copies of identical messages to different relays in order to improve the delivery rate or reduce the delivery delay [5] [6]. The previous work generally assumes nodes in the network is identical and independent, which implies these nodes have similar delivery capabilities as well as mobility patterns. This assumption, although simple, may not be realistic. Recent measurements on the *Pocket Switched*

*Networks* [7] illustrate that nodes in a DTN have heterogeneous mobility patterns. For example, some nodes visit certain areas more frequently than other nodes; or a group of nodes with common properties may meet with each other more often. Similarly, the *PeopleNet* [8] also observes that people with similar interests may form a bazaar, where they meet with each other with high probabilities.

In this paper, we explore the routing in DTNs where nodes are heterogenous. More specifically, we try to exploit the periodic movement patterns of nodes to estimate one node’s ability to deliver a specific message to a destination. We argue that this estimation information can be used to select the message routes and greatly improve the performance. Moreover, instead of full duplication of messages as used in [5] and [6], we adopt an alternative way to generate redundancy with erasure coding. With a fixed overhead, the erasure coding can generate a large number of message-blocks instead of a few replications, and therefore it allows the transmission of only a portion of message to a relay. This ability is critical to increase the diversity in routing when combined with the estimation-based approach, which can be shown in the following example.

Assume node *A* has a message to deliver to node *D* and *A* can spread two copies of the message to relays. Also assume each node has an estimation on its ability (e.g. the probability) to deliver a message to *D*. When *A* meets a node *B*, if *B* has higher probability to deliver the message, it may be reasonable for *A* to spread one copy of message to *B*, since *B* is considered as “better” than *A*. However, it may be a bit complicated if *B* is estimated to be “worse” than *A*. It would be a risk to spread a copy to *B*, as *B* is estimated to have less probability to deliver the message. However, it would be also a risk not to spread a copy, as *A* may miss a chance to spread copies to more relays. Note that more relays in the network will also increase the delivery probability of the message. With erasure-coding, we can achieve a better tradeoff here. Instead of making “yes or no” decisions, we can forward only “part” of one message (i.e. a few message blocks) to *B*. The portion to spread should depend on the estimation value of *B*.

Based on the above observations, we propose a novel routing scheme for DTNs that combines erasure coding with

estimation-based approach in this paper. We conduct extensive simulations to evaluate our proposed scheme, and the results demonstrate that our combined approach generally has much less delivery delay compared to existing approaches with same overhead.

The rest of this paper is organized as follows. We present a brief survey on related work on DTN routing in Section II. In Section III, we present our routing scheme in detail. The simulation results are shown in Section IV. Section V concludes this paper.

## II. RELATED WORK

There have been many research activities in the area of DTN routing [9]. The knowledge-based routing scheme for DTNs is proposed in [10]. In their work, the DTN routing problem is formulated as moving messages across a graph with time-varying connectivity but whose dynamics is predictable. The authors of [11] propose the space-time routing scheme. They assume the network profile can be accurately predicted over certain period of time. The dynamic of the networks is model as *space-time graph*. Finding the best route for message is done by looking ahead. Both of them are also deterministic routing schemes.

An epidemic routing is proposed in [5] which works like flooding the message to all nodes in the networks. Obviously, it has the highest probability of delivery (or least latency) and the most overhead in terms of bandwidth and energy as well [3]. On the contrary, *direct transmission* [12] only lets the source hold the data until it comes in contact with the destination. This scheme, although has minimal overhead, may incur considerable long delays.

Other routing schemes can be deemed as some tradeoff between those two schemes. Approaches are proposed to control the flooding overhead by selectively forwarding to a few nodes. In [13] the authors use the *delivery predictability* to indicate the chance a relay node can deliver messages to the destination. A message is flooded to a new relay only if it has higher *delivery predictability*. Similarly, ref. [14] proposes the *expected path length* as the metric in forwarding messages, which is defined as the expectation of message transmission latency through multi-hop relays. And in [15], the authors propose to forward messages to relay nodes with similar mobility pattern as the destination. The node mobility pattern is defined as the coordinates in “virtual space” which reflexes the probability of a node appearing at some certain check-points. The similarity of two nodes’ mobility pattern is defined as the Euclidean distance between their virtual coordinates.

The aforementioned schemes, although generally can reduce much overhead, do not bound the exact overhead to deliver a message. In the worst case, a message may still spread to every node in the network. Therefore, in [16] and [6], the authors propose to spray only a fixed  $r$  number of copies of the message into the network to reduce the delivery latency. The number of identical copies  $r$  is called *replication factor*. The *Spray and Wait* scheme presented in [6] makes every node

transfer half of its message copies to the contacts until there is only one copy left. In [17], instead of simply duplicating the message, the authors propose to erasure code the messages. In their scheme, a message is divided into  $K$  blocks and erasure coded to total generate  $K \times R$  coded blocks. A receiver can decode the original message with any  $K$  blocks from the coded blocks. With erasure coding, it will potentially use more relays and is much more robust to failures of a few relays or some bad choice of long delay nodes. It is also shown in [17] that the average delay using erasure-coding can converge to a constant in an extreme case. However, these schemes are all based on the assumption that nodes in the network is identical.

## III. OUR ESTIMATION-BASED SCHEME

As stated earlier, the basic idea of our scheme is to combine erasure-coding with the estimation-based routing scheme and selectively distribute message blocks to relay nodes. In this paper, we use the contact frequency with the destination as the metric to estimate a node’s ability to delivery the message, which is defined as the number of contacts between two nodes in one unit time. For each message, the source takes a replication factor  $R$  and erasure codes  $R \times K$  equal sized blocks. When two nodes encounter, these message blocks are re-dispatched between them according to their ranking values. The message can be fully decoded at the destination if at least  $K$  generated blocks are received. Note that since each message block is  $1/K$  of the size of the original message, the proposed erasure-coding will generate same overhead as simply replicating  $R$  copies of the message. In the following subsections, we describe our scheme in more details.

### A. Metric for Estimation Relay Nodes

We assume the chance of a node to encounter a specific node is strongly correlated with the meeting history of these two nodes. Therefore, one node may estimate its chance to meet any other node using the *average contact frequency* it observed in the past. Formally, we define this *average contact frequency* as follows:

*Definition 1 (Average Contact Frequency):* The average contact frequency (ACF)  $\tau_{i,j}$  is defined as the number of encounters between node  $i$  and node  $j$  within a unit time. ACF can be approximated with  $\tau_{i,j} = \frac{N_{i,j}}{T}$ , where  $T$  is pre-defined fixed length of time; and  $N_{i,j}$  is the number of contacts node  $i$  and  $j$  have during  $T$  time.

We assume that each node in DTN has a unique ID, and each node can maintain an ACF table including every node it encounters. Every  $T$  time, a node takes a sample of ACF for every other node. These samples are exponentially smoothed. Note that in our previous work [14], nodes may also exchange their ACF tables when they meet, and therefore each node can estimate its chance of delivery based on end-to-end (multi-hop) information. Such a scheme will give more accurate estimations but also incur much more control plane overhead,

and therefore it is only suitable for small scale DTNs. However, in this paper, we target at large scale networks and later our simulations show that with only one-hop information, it already can greatly help the routing decisions.

### B. Estimation-based Erasure-coding Routing

The distribution of the message blocks is partially inspired by the *binary spray and wait* in [6]. Any node may work at the *spreading state* or the *forwarding state*. A node is in the *spreading state* if it holds more message blocks than a pre-defined threshold  $G$ ; otherwise, it works in the *forwarding state*.

Initially, the source holds all  $R \times K$  message blocks and is working in the *spreading state*. When it meets a new node, it will dispatch a portion of message blocks to that node, and this node becomes a new “relay” of the message (blocks). The portion of message blocks of the new relay hold should be proportional to its ACF value. On the other hand, when a relay working in the *forwarding state*, it will not generate a new relay when it meets a new node. This time, it will either keep all message blocks, if the new node has less ACF value. Or if the new node has higher ACF value, it will forward all message blocks to the new node.

This two-states routing scheme is simple yet efficient to deliver messages to the destinations. In the *spreading state*, message blocks are quickly distributed to diverse relays, while in the *forwarding state*, message blocks are directed to the most “suitable” relays based on the estimation of nodes. The pseudo-code of our routing scheme is summarized in Algorithm 1.

Note that the parameter  $G$  is a tunable parameter. It is easy to see that  $G \leq K$ . It is because  $K$  blocks are enough to decode the original message. Therefore it is unnecessary for a relay to keep more than  $K$  blocks in the *forwarding state*. In this work, we choose  $G = K$ .

---

#### Algorithm 1 The message block forwarding operation process

**Precondition:** Node  $A$  encounters  $B$  and destination node is  $D$ . Node  $A$  has  $m_A$  messages blocks and node  $B$  has no block.  $\tau_{i,D}$  is node  $i$ ’s frequency of contact with destination node  $D$ .

```

1: if  $\tau_{A,D} < \tau_{B,D}$  then
2:   if  $m_A > G$  then
3:      $m_B = \lceil m_A \times \frac{\tau_{B,D}}{\tau_{A,D} + \tau_{B,D}} \rceil$ ;
4:      $m_A = m_A - m_B$ ;
5:   else
6:      $m_B = m_A$ ;
7:      $m_A = 0$ ;
8:   end if
9: else
10:  if  $m_A > G$  then
11:     $m_B = \lfloor m_A \times \frac{\tau_{B,D}}{\tau_{A,D} + \tau_{B,D}} \rfloor$ ;
12:     $m_A = m_A - m_B$ ;
13:  end if
14: end if
```

---

## IV. SIMULATION EVALUATIONS

### A. Simulation Model and Methodology

In our simulation, we use a *Restrict Random Waypoint* mobility model, which is similar to the model used in [14]. Different from the traditional random waypoint model, each node in the restrict random waypoint model has a given set of waypoints. After the node arrives at a waypoint, it will stay at that point for a random “thinking time”. After that, the node will randomly pickup one destination from its waypoint set and move towards that point. Therefore, instead of having uniform visiting probability on the whole area, each node may only have high chance to visit points in its waypoint set. By carefully setting the distribution of each node’s waypoints, we can simulate different types of movements.

The network used in our simulations consists of 150 nodes moving in a  $10000 \times 10000$  unit<sup>2</sup> square area and there is no node failure. Each node has 20 waypoints. We use  $0 \sim 149$  as each node’s unique ID. Among those nodes, half of them move in the speed of  $V_{max} = 50$  unit/s and the other half move in the speed of  $V_{min} = 20$  unit/s. In the initial state, nodes are randomly located in the entire square area. Three quarters of each node’s 20 waypoints are randomly distributed in a  $200 \times 200$  unit<sup>2</sup> square whose center is the node’s initial position; the other one fourth of the 20 waypoints are randomly distributed in the entire network. The thinking time at each waypoint is uniformly selected in  $[0, 20s]$ . Each node’s communication range is set to 100 units. For each message, it is encoded into  $R \times K$  message blocks.  $R$  is the replication factor and its default value is set to 10.  $K$  is equal to  $M/b$ , where  $M$  is the message size and  $b$  is the block size.  $K$  has the default value of 10 in our simulation as well. The maximum number of message blocks one relay can have is set to  $G = 10$  by default, so a relay can keep one entire message in its *forwarding state*.

At the beginning of each simulation run, we have all the nodes moving in the network for a certain period of time, so that nodes can build and stabilize their AFC tables. After that, messages are randomly generated with randomly chosen sources and destinations.

We evaluate our proposed estimation-based erasure-coding routing scheme (EBEC) against some previously proposed schemes, namely the *binary spray and wait* (BSW) [6] and erasure-coding routing scheme (EC) [17]. All three schemes are compared under the same replication factor  $R$ .

### B. The Estimation Metric

The performance of EBEC scheme depends on proper estimation of nodes’ delivery abilities to given destination. We use AFC as the metric to estimate a node’s possibility of successfully delivering messages to the destination. We believe that using this historical information can pretty well predict the future chance to encounter the destinations. With the aforementioned restrict random waypoint mobility model, we run a simulation and dump the number of contacts between node 0 and other nodes after the simulation runs for 2000,

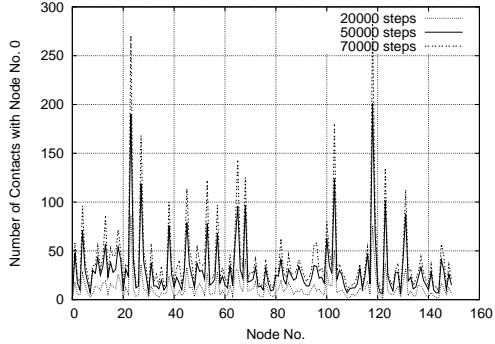


Fig. 1. Frequencies of contacts between node 0 and all other nodes when running the simulation for different number of rounds. The contact frequency is roughly proportional to the number of simulation rounds.

5000, and 7000 rounds, respectively. The number of contacts between node 0 and other nodes is plotted in Fig. 1, where the  $x$  axis is the node ID and the  $y$  axis is the number of contacts between node 0 and node  $x$ .

From this figure we can see that different nodes have quite different number of contacts with node 0. But the number of contacts is roughly proportional to the simulation time. For instance, after running the simulation for 2000 rounds, we found that node 0 encountered node 23 and node 118 much more frequently than other nodes. As the simulation runs more rounds, node 0 still encounter with node 23 and node 118 more frequently than other ones.

### C. Message Delivery Delay

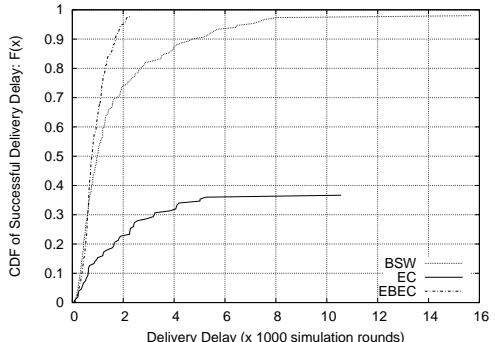


Fig. 2. CDF of message delivery latencies for BSW, EC, and EBEC routing schemes.

For each of the simulated routing schemes, we have run 150 different simulation instances and plot the CDF (Cumulative Density Function) of the message delivery delays in Fig. 2. From this figure we can see that the EBEC scheme has better performance than BSW and EC schemes in term of delivery delay. EBEC is better than BSW because the message blocks are forwarded to selected relay nodes instead of blindly choosing relay nodes. From Fig. 2 we can also see that the EC routing scheme has rather longer delivery delays than the BSW scheme. This can be explained as a long time is needed to spray the large number of message blocks ( $R \times K$  blocks)

to more relay nodes and also a long time is needed for the destination to collect enough blocks to decode the message. This result is consistent with the work in [17]. As stated in [17], EC is suitable only for DTNs where relay failures are prevalent, delays are unpredictable, and minimizing the worse-case delay is important. However, in our simulations, nodes never fail or drop messages.

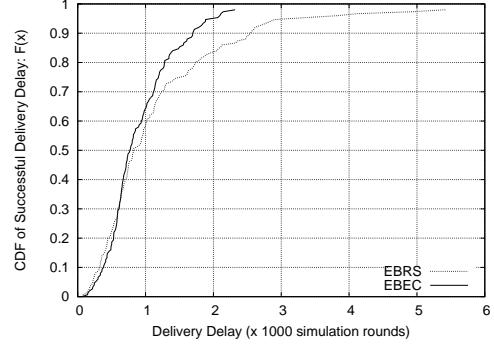


Fig. 3. CDF of message delivery delay when setting the coding factor  $K$  to different values.

### D. The Gain of Adopting Erasure Coding

In order to evaluate the gain we have by adopting erasure coding in the routing scheme, we conduct the simulations to evaluate EBEC with different  $K$  values. Note that when we set  $K = 1$ , EBEC is reverted to an estimation-based replication scheme (EBRS), where each message block contains the whole messages. Similarly, we run the simulation of each scheme for 150 different instances with EBEC and EBRs, respectively. The CDF of message delivery delays of each scheme is plotted in Fig. 3. From this figure we can see that adopting the erasure-coding scheme does have much gain in term of delivery delay. It is because with erasure coding, nodes with lower ranking values can still be exploited properly to increase the speed to spread more relays as well as the routing diversity.

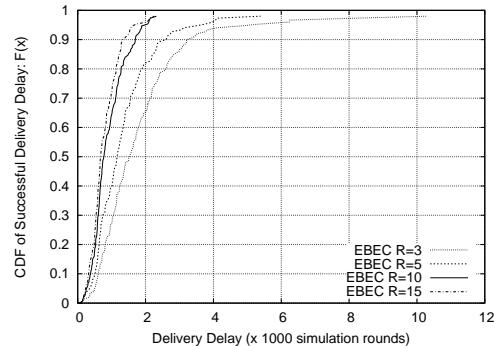
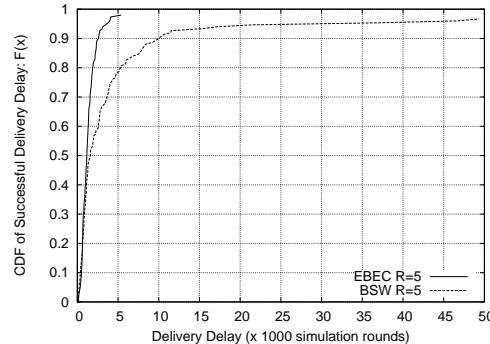


Fig. 4. CDF of message delivery delay in EBEC scheme when setting the replication factor  $R$  to 3, 5, 10, and 15, respectively.

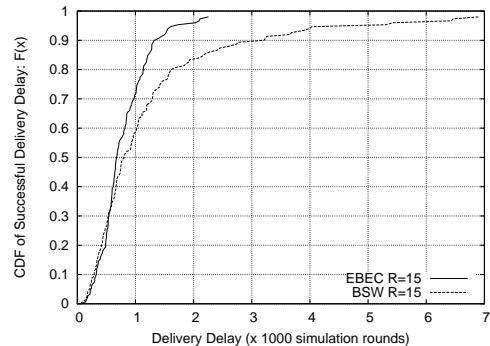
### E. The Impact of Replication Factor $R$

As we have discussed, the replication factor  $R$  is an important parameter that affects the delivery delay. When  $R$  is

equal to 1, the BSW scheme reverts to the direct transmission scheme. Generally, the delivery delays will decrease when  $R$  increases. We set the replication factor to different values and again run 150 different simulation instances of the EBEC scheme. The CDF of message delivery delays is plotted in Fig. 4. This figure shows that the delivery delays of EBEC are smaller when a larger replication factor  $R$  is used, which is consistent with our intuition. We also compare EBEC with BSW, as plotted in Fig. 5(a) and Fig. 5(b), where the replication factor is set to 5 and 15, respectively. From Fig. 5 we can see that the EBEC is more superior to BSW, especially when  $R$  is smaller. This is mainly because our scheme is based on the estimation scheme and message blocks are transferred to low-delay relay nodes, so that the message can be delivered to the destination quickly even the message redundancy is small.



(a) BSW EBEC,  $R=5$

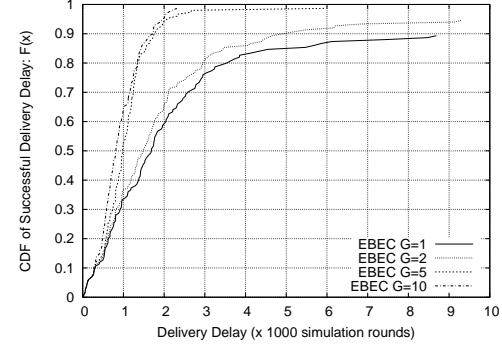


(b) BSW EBEC,  $R=15$

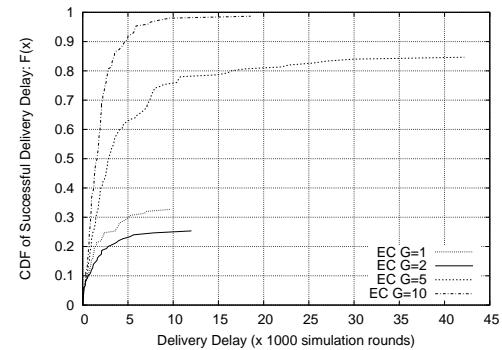
Fig. 5. Comparation between the CDFs of BSW scheme and EBEC scheme when setting the replication factor  $R$  to 5 and 15, respectively.

#### F. The Impact of $G$

In this section, we evaluation the impact of threshold  $G$  on the delivery delay of EBEC. The parameter  $G$  defines the minimal number of message blocks a node has before it turns into the *forwarding state*.



(a) EBEC scheme



(b) EC scheme

Fig. 6. CDF of message delivery delay when setting the maximum number of message blocks each node can keep to  $G = 1$ ,  $G = 2$ ,  $G = 5$ , and  $G = 10$  respectively.

We have run a set of simulations of the EBEC scheme as well as the EC scheme. In each simulation run, both the replication factor  $R$  and the number of blocks one message has  $K$  are set to 10. We vary  $G$  with different values. The results of EBEC scheme and EC scheme are plotted in Fig. 6(a) and Fig. 6(b), respectively. The plot in Fig. 6 shows that the EBEC scheme always outperforms EC scheme with the same  $G$  parameter. The message delivery delays of EBEC scheme are high when  $G$  is small. However, when  $G$  is larger than half of  $K$ , the delivery delays will not change too much for EBEC. This implies that we may not distribute the message blocks to too many relays. Otherwise, the destination will take longer to collect enough blocks to decode the original message.

## V. CONCLUSION

In this paper, we have presented a routing scheme for DTNs in which nodes are not identical and the mobility pattern has certain structures. Our routing scheme combines erasure-coding with estimation-based routing scheme. Our scheme speeds up distributing the message blocks to different relays by letting the low ranked nodes have the opportunity to carry a small portion of message blocks. In the meanwhile, after

the message blocks are distributed to different relays, they can still be forwarded to nodes that are estimated as “better” relays, which further leads to less delivery delay. We used simulation experiments to evaluate the performance of our routing scheme. The results demonstrate that our scheme can delivery message faster than existing schemes with the same overhead.

## REFERENCES

- [1] K. Fall, “A delay-tolerant network architecture for challenged Internets,” in *SIGCOMM ’03: Proceeding of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*. New York, NY, USA: ACM Press, 2003, pp. 27–34.
- [2] S. Burleigh, A. Hooke, L. Torgerson, K. Fall, V. Cerf, B. Durst, Keith Scott, and H. Weiss, “Delay-tolerant networking: An approach to interplanetary Internet,” *IEEE Communications Magazine*, June 2003.
- [3] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. S. Peh, and D. Rubenstein, “Energy-efficient computing for wildlife tracking: design tradeoffs and early experiences with zebranet,” *SIGOPS Oper. Syst. Rev.*, vol. 36, no. 5, pp. 96–107, 2002.
- [4] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson, “Wireless sensor networks for habitat monitoring,” in *WSNA ’02: Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*. New York, NY, USA: ACM Press, 2002, pp. 88–97.
- [5] D. B. Amin Vahdat, “Epidemic routing for partially connected ad hoc networks,” Duke University, Technical Report CS-200006, April 2000.
- [6] T. Spyropoulos, K. Psounis, and C. Raghavendra, “Spray and wait: an efficient routing scheme for intermittently connected mobile networks,” in *WDTN ’05: Proceeding of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking*. New York, NY, USA: ACM Press, 2005, pp. 252–259.
- [7] P. Hui, A. Chaintreau, J. Scott, R. Gass, J. Crowcroft, and C. Diot, “Pocket switched networks and human mobility in conference environments,” in *WDTN ’05: Proceeding of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking*. New York, NY, USA: ACM Press, 2005, pp. 244–251.
- [8] M. Motani, V. Srinivasan, and P. S. Nuggehalli, “Peoplenet: engineering a wireless virtual social network,” in *MobiCom ’05: Proceedings of the 11th annual international conference on Mobile computing and networking*. New York, NY, USA: ACM Press, 2005, pp. 243–257.
- [9] Z. Zhang, “Intermittently connected mobile ad hoc networks and delay tolerant networks: Overview and challenges,” 2004, to appear on IEEE Survey and Tutorial.
- [10] S. Jain, K. Fall, and R. Patra, “Routing in a delay tolerant network,” in *SIGCOMM*, Aug 2004.
- [11] S. Merugu, M. Ammar, and E. Zegura, “Routing in space and time in networks with predictable mobility,” Georgia Institute of Technology, Technical Report GIT-CC-04-7, 2004.
- [12] T. Spyropoulos, K. Psounis, and C. Raghavendra, “Single-copy routing in intermittently connected mobile networks,” in *Secon ’05: The First IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks*. October 2004.
- [13] A. Lindgren, A. Doria, and O. Schelen, “Probabilistic routing in intermittently connected networks,” *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 7, no. 3, pp. 19–20, 2003.
- [14] K. Tan, Q. Zhang, and W. Zhu, “Shortest path routing in partially connected ad hoc networks,” in *Globalcom*, 2003.
- [15] J. Leguay, T. Friedman, and V. Conan, “Dtn routing in a mobility pattern space,” in *WDTN ’05: Proceeding of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking*. New York, NY, USA: ACM Press, 2005, pp. 276–283.
- [16] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, “Multi-copy routing in intermittently connected mobile networks,” USC, Technical Report, 2004.
- [17] Y. Wang, S. Jain, M. Martonosi, and K. Fall, “Erasure-coding based routing for opportunistic networks,” in *WDTN ’05: Proceeding of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking*. New York, NY, USA: ACM Press, 2005, pp. 229–236.