Data-Intensive Routing in Delay-Tolerant Networks

Kazuya Sakai*, Min-Te Sun†, and Wei-Shinn Ku‡

*Dept. of Electrical Eng. and Comput. Sci., Tokyo Metropolitan University, 6-6 Asahigaoka, Hino, Tokyo 191-0065, Japan.

†Department of Computer Science and Information Engineering, National Central University, Taoyuan 320, Taiwan.

‡Department of Computer Science and Software Engineering, Auburn University, Auburn, Alabama 36849–5347.

ksakai@tmu.ac.jp, msun@csie.ncu.edu.tw, and weishinn@auburn.edu

***Abstract*—Mobile users and wireless devices are now the sources of a large volume of data. In such data-intensive mobile and wireless computing systems, delay-tolerant network (DTN) routing plays a critical role in data routing, dissemination, and collection. In this paper, we first introduce a new routing problem in DTNs - data-intensive routing - where data transmitted from one node to another is very large with respect to the size of data which can be transmitted in a single contact and available buffer size at relay nodes. In the proposed opportunistic path model, the contact frequency, contact duration, and buffer constraint are all integrated into a single routing metric. Then, we design the data-intensive routing (DIR) protocol where the path with the highest bottleneck link capacity is defined as the path weight. In addition, we propose the advanced DIR (A-DIR) protocol which focuses on the probability that the last message block will be delivered to its destination within the time constraint. Both the DIR and A-DIR protocols forward messages to better relays or to their destinations based on a greedy strategy with the proposed path metric. Simulations using real mobility traces demonstrate that the proposed DIR and A-DIR protocols achieve their design goals.**

***Index Terms*—Delay tolerant networks, DTNs, routing, data-intensive protocols.**

I. Introduction

Delay-tolerant networks (DTNs) prove useful in many value-added mobility-based applications such as pocket-switched networks [1], disaster recovery [2], mobile crowd sensing [3], and data offloading [4]. The network model of DTNs is different from traditional wireless networks because the data transmission opportunities are limited by the disruptive nature of opportunistic networks. To enable end-to-end communications in such an environment, a number of DTN routing protocols [5]–[10] have been proposed based on the principle of *store-and-carry*, where nodal mobility is exploited in the message forwarding process. These protocols assume that the size of packets exchanged among mobile nodes is relatively small with respect to the size of data which can be transmitted in a single contact.

In the past few years, trends in DTN protocol designs have gotten more data-intensive to adapt to the data-driven mobile computing era. As a result, mobile users and wireless devices are now the sources of a large volume of data including social information and sensory-generated data [11] (e.g., image and video files). To effectively route, disseminate, and collect big data, we involve the manipulation of data in protocol designs. For example, Gao et al. [12], [13] designed collaborative caching and cache maintenance mechanisms to quickly reply to consumer's queries in data dissemination. Zhao et al. [14] developed data replication schemes with erasure codes for a DTN-based mobile cloud.

However, to the best of our knowledge, there is no study on how to route big data over multi-hop opportunistic networks. Therefore, in this paper, we introduce a new class of routing protocols in DTNs, *data-intensive routing*. The objective of data-intensive routing is to maximize the delivery rate of big data from the source to destination. Our work differs from traditional DTN routing in several respects. First, the message routed from one node to another is very large with respect to the size of data which can be transmitted in a single contact and the size of available buffer at relay nodes. Even compared with the existing data-intensive DTN protocols [12]–[14], the size of the data considered in this paper is large. Second, in a manner counter-intuitive to the traditional DTN routing problem, this paper asserts that duplicating more message copies does not necessarily improve message delivery. To be specific, the first message block of the first copy of a message at the head of the source node's buffer has most likely been delivered to its destination by the time the first message block of the second copy is polled from the buffer. Although this issue can be alleviated by buffer manipulation, we conclude that introducing more than two copies of a message does not help message delivery in our simulations.

These differences impose new design challenges upon us. The key challenge of data-intensive routing protocol designs is how to model multi-hop opportunistic paths and define better forwarding relays by incorporating the pairwise contact rate, contact duration, and buffer constraints. While many probability models including exponential, hypoexponential, Poisson, and Pareto distributions have been used to understand the fundamental performance issues in DTNs, the existing models cannot be applied to data-intensive routing. To tackle this challenge, we first build *the contact-duration-aware opportunistic path* model from scratch. Then, depending on how to quantify the path weight, we design two distributed data-intensive routing protocols for DTNs. The contributions of this paper are as follows:

- First, we introduce a new class of routing problems for DTNs, data-intensive routing, in which the message size is much larger with respect to the amount of data which can be transmitted during a contact and be stored in the buffer of relay nodes. Therefore, in data-intensive routing, the contact duration, buffer constraint, and limited forwarding opportunities must be taken into account for protocol designs.

- Second, we formulate the contact-duration-aware opportunistic path for data-intensive routing, which is modeled by the Poisson and Pareto distributions. Since such a model unfortunately does not have a closed-form expression, by simplifying assumptions, we derive a closed-form solution to quantify the path weight by the min-max-based metric.
- Third, we propose a data-intensive routing (DIR) protocol that, for a given message overhead constraint, delivers a large amount of data to its destination. In our DIR, a forwarding decision is made at every contact based on the proposed metric and the principle of the greedy strategy. In addition, we incorporate buffer manipulation techniques to improve message delivery.
- Fourth, we further propose the advanced data-intensive routing (A-DIR) protocol, in which we employ a smarter metric. The A-DIR protocol further improves performance by optimizing the delivery probability of the last set of message blocks.
- Finally, we conduct extensive simulations using a well-known real trace, CRAWDAD dataset Cambridge/haggle [15], in order to demonstrate that the proposed protocols outperform the baseline protocol based on the existing spray-and-wait solution.

The rest of paper is organized as follows. In Section II, we formulate the problem of data-intensive routing. With the contact-duration-aware opportunistic path model, we propose the DIR and A-DIR protocols in Sections III and IV, respectively. We evaluate the performance of the proposed protocols in Section V. Section VI reviews the existing works in DTNs, and Section VII concludes this paper.

II. Problem Formulation

A. Network Model

A DTN is represented by an undirected contact graph, denoted by $G = (V, E)$, where V is a set of nodes and E is a set of links. Let v_i be node i and $e_{i,j}$ be the link between nodes v_i and v_j. Link $e_{i,j}$ exists in the graph if and only if nodes v_i and v_j have at least one contact in the past. The inter-meeting time between v_i and v_j is defined as $1/\lambda_{i,j}$. The probability of v_i meeting v_j at time t is assumed to follow the exponential distribution, i.e., $\lambda_{i,j}e^{-\lambda_{i,j}t}$. Hence, the probability of v_i meeting v_j within time constraint T is obtained by Equation 1.

$$\int_0^T \lambda_{i,j}e^{-\lambda_{i,j}t}dt = 1 - e^{-\lambda_{i,j}T} \tag{1}$$

The entire file to be transmitted is called a *message* and is denoted by M. Message M consists of l chunks, i.e., $M = \{m_1, m_2, ..., m_l\}$, and each chunk is called a *message block*. Two nodes are assumed to be able to send/receive message blocks while they are in the communication range.

TABLE I
Definition of notations.

Symbol	Definition
v_i	Node v_i
$N(i)$	The open neighbor set of v_i
$\lambda_{i,j}$	The contact frequency between v_i and v_j
$\delta_{i,j}$	The contact duration between v_i and v_j
M	A message $M = \{m_1, m_2, ..., m_l\}$ with l blocks
T	The message deadline
L	The number of copies of a message
B_i	The available buffer of v_i for a msg., $B_i \in [1, B_{max}]$
η	The number hops between two nodes
$p_{i,j}(.)$	The weight of the link between v_i and v_j
$H_{i,j}(.)$	The path weight between v_i and v_j

The transmission during a contact is assumed to be reliable, while wireless links among nodes are intermittently disrupted. We define the average link duration of a contact between nodes v_i and v_j by $\delta_{i,j}$. For simplicity, we assume that a node can send one block of a message to another node in one unit time. For example, if two nodes v_i and v_j are connected for 60 seconds, then v_i can send 60 blocks of M, say m_1, m_2, ..., m_{60}, to v_j. Since we aim to address data-intensive routing, the size of message M is relatively large with respect to l, i.e., $l >> \delta_{i,j}$ for any $v_i, v_j \in V$.

Each node has a buffer with a limited size to temporally store message blocks. For simplicity, we assume that the buffer at each relay is of the same size. The maximum buffer size and the available buffer size at relay node v_i are denoted by B_{max} and B_i, respectively. The size of the buffer is quantified with respect to the number of message blocks. For instance, when $B_i = 100$, v_i can store 100 message blocks. In data-intensive routing, the size of M is relatively large with respect to the maximum buffer size, i.e., $l >> B_{max}$.

The notations used in this paper are listed in Table I.

B. Problem of Data-Intensive Routing

To the best of our knowledge, this paper is the first to address data-intensive routing in DTNs, which is formally defined below. Let v_s be the source node who wishes to deliver a large message $M = \{m_1, m_2, ..., m_l\}$ to the destination v_d. The traffic constraint is given by the number of message copies, denoted by L, and the buffer constraint is determined by the maximum buffer size B_{max} at relay nodes. Given a message deadline, denoted by T, routing is said to be successful if all the blocks of a message are delivered to v_d within T. Otherwise, routing is considered failed.

A message can be duplicated to up to L copies, and message blocks with the same index of different copies are identical. Let $m_i^{(j)}$ be the i-th block of the j-th copy of M. Here, we refer to i and j as the message block ID and the copy ID, respectively. For any j and j' (where $j \neq j'$), two blocks $m_i^{(j)}$ and $m_i^{(j')}$ are identical. Thus, v_d can assemble the original data M from a set of blocks from different copies.

Data-intensive routing differs from previous DTN routing problems in that the size of messages is much larger than both the buffer size, i.e., $l >> B_{max}$, and the amount of data that a node can transmit at one contact, i.e., $l >> \delta$. In addition, the message size considered in this paper is large compared to existing data-intensive DTN protocols. For example, in [14], one data item is assumed to consist of 32 blocks and, in their simulations with real datasets [15], the buffer size that each node offers for one data item is set to be 96. Our data-intensive routing handles bigger data; each message consists of 3,000 blocks and each node offers a buffer of 300, as shown later in Section V.

In this paper, a large message is sequentially divided into l message blocks. We claim that our work can be easily extended to incorporate coding techniques, e.g., erasure codes in which the original message can be assembled from k blocks out of l blocks.

C. Research Challenges

Data-intensive routing is a new class of routing in DTNs, and it differs from existing DTN routing problems. As such, we face new challenges in our research and list them below.

- **Challenge 1:** The original DTN routing problem generally assumes that each contact duration is long enough for one node to forward a message to another node. In other words, only the contact frequencies between each pair of nodes affect the routing performance. Thus, the link weight (or capacity) is simply quantified based on the contact frequency. However, this is not the case in data-intensive routing. The link duration plays a critical role in the link quality. Our first design challenge is determining how to integrate the contact frequency and link duration into a single link and multi-hop path capacity metric.
- **Challenge 2:** Most DTN routing protocols do not consider the buffer constraint. This is because a DTN routing module is implemented at the Bundle layer, which is located between the transport and application layers. Each application designer shall define the buffer size based on her application requirement. Consequently, it is reasonable for protocol designers to assume that each node has sufficient buffer space. However, the available buffer at each node is of significant concern, since a message can be too big for intermediate nodes to store the message pieces in their buffer. Therefore, our second challenge is incorporating the buffer constraint into a link and path metric.
- **Challenge 3:** Additionally, most of existing routing protocols do not consider path diversity. This is because the weight of a path at given time instance is always the same in the existing solutions. For example, Figure 1 shows a contact graph with four nodes. There exist two paths between v_s and v_d, i.e., $\{v_s, v_1, v_d\}$ and $\{v_s, v_2, v_d\}$. Based on the contact frequency at each hop of these paths, $\{v_s, v_2, v_d\}$ will be considered the better path. However, when the path weight is evaluated, the buffer at v_2 may not be sufficient, and as a result, the first path, $\{v_s, v_1, v_d\}$, may be the better path at some time instances. Thus, our third challenges is utilizing the path diversity for faster delivery.

III. Protocol Design

A. Protocol Overview

In this section, we present a data-intensive routing protocol (DIR) by utilizing the contact history as well as contact duration. To optimize the delivery rate, we first introduce the multi-hop contact-duration-aware opportunistic path model, which quantifies the probability of all the blocks of a message being delivered to the destination within deadline T. Unfortunately, the precise delivery probability has no closed-form solution. Hence, we build a link weight for single-hop with simplified assumptions and then propose a multi-hop path weight based on the min-max metric to define a better path. This approach integrates the contact frequency, contact duration, and buffer constraint. With this model, we design a data-intensive routing (DIR) protocol. For the given deadline and number of copies constraints, a node with some chunks of a message makes a forwarding decision based on the proposed path model.

Note that the data-intensive routing is a new class of routing in DTNs. Although some of existing DTN routing protocols [16]–[18] consider conatct durations, they are not intended to handle large data. While there is data-intensive protocol incorporating a contact duration into a metric [14], it is primarily designed for packet-level file replications in DTNs and multi-hop fowarding is not considered. In the existing contact duration aware routing [19], the delivery probability model is limited up to two hops and block level message forwarding is not considered.

B. Contact-Duration-Aware Opportunistic Path

In this paper, we model multi-hop contact-duration-aware data forwarding between two nodes, v_s and v_d, within deadline T. According to [14], the number of contacts and the contact duration between two nodes within time constraint T can be modeled by the Poisson and Pareto distributions, respectively. Let $X_{i,j}$ be the random variable that represents the number of contacts between nodes v_i and v_j. Then, $\Pr[X_{i,j} = k]$ is computed by Equation 2.

$$\Pr[X_{i,j} = k] = \frac{(\lambda_{i,j} T)^k \, e^{-\lambda_{i,j} T}}{k!} \tag{2}$$

Let $Y_{i,j}$ be the random variable that represents the contact duration between v_i and v_j for a contact. The shape and scale parameters are denoted by $\alpha_{i,j}$ and $y_{i,j}$, which are determined by mobility traces or simulations. Then, the probability density function (PDF) is defined by $f_{Y_{i,j}}(y) = \frac{\alpha_{i,j} y_{i,j}^{\alpha_{i,j}}}{y^{\alpha_{i,j}+1}}$ if $y > y_{i,j}$, and $f_{Y_{i,j}}(y) = 0$, otherwise. Let $Z_{i,j}^{(k)}$ be the random variable that represents the summation of k contact duration, i.e., $Z_{i,j}^{(k)} = Y_{i,j}^1 + Y_{i,j}^2 + ... + Y_{i,j}^{(k)}$, where $Y_{i,j}^{(k)}$ is the contact duration of the k-th contact between v_i and v_j. The summation of the contact duration between v_i and v_j within T, denoted by $F_{Z_{i,j}}(T)$, is obtained by the following:

$$F_{Z_{i,j}}(T) = \sum_{k=1}^{\infty} \Pr[X_{i,j} = k] \cdot f_{Z_{i,j}}^{(k)}(T). \tag{3}$$

Here, $f_{Z_{i,j}}^{(k)}(T)$ is the convolution of the Pareto distribution with k contacts. Equation 3 can be seen as the link weight of two nodes, v_i and v_j. Let η be the number of hops between v_s and v_d. Then, the amount of data delivered from v_s to v_d within T, denoted by $H_{s,d}(T)$, is derived by the convolution of $F_{Z_{i,j}}(T)$. Therefore, the weight of a contact-duration-aware opportunistic path between v_s and v_d can be computed:

$$H_{s,d}(T) = F^{(\eta)}_{Z_{i,j}}(T). \tag{4}$$

Unfortunately, there is no closed-form solution for Equations 3 or 4. In the subsequent section, we will propose a path metric with simplified assumptions to quantify the probability of data-intensive being delivered from v_s to v_d within time constraint T.

C. Link Weight

To simplify the analyses, we assume the following conditions hold. First, the link from nodes v_i to v_j is used as a routing path only when $\lambda_{i,j}T > 1$, which means that nodes v_i and V_j would have to meet more than once within time constraint T. For each contact, v_i and v_j can communicate for $Y_{i,j}$ unit times. Since $Y_{i,j}$ is modeled by the Pareto distribution, the expected value of $Y_{i,j}$ can be obtained:

$$E[Y_{i,j}] = \begin{cases} \infty & \text{if } y > y_{i,j} \\ \dfrac{\alpha_{i,j} y_{i,j}}{\alpha_{i,j} - 1} & \text{otherwise.} \end{cases} \tag{5}$$

Since we assume that one message block can be transmitted in one time unit, if the contact duration equals $Y_{i,j}$, then node v_i can send $Y_{i,j}$ message blocks to v_j as long as the buffer size at v_j is large enough. Otherwise, the number of message blocks that v_i sends to v_j is limited by the receiver's buffer size. Therefore, the expected number of transmitted message blocks is obtained by $c = \min\{E[Y_{i,j}], B_j\}$.

Let $p_{i,j}(T, l)$ be the probability that node v_i delivers M with l blocks ($|M| = l$) to v_j within T. At every contact between nodes v_i and v_j, node v_i can send c blocks to v_j on average. Hence, at least l/c contact events need to occur within T. Therefore, we may derive $p_{i,j}(T, l)$ as follows.

$$p_{i,j}(T, l) = 1 - \sum_{k=0}^{l/c-1} \frac{(\lambda_{i,j}T)^k \, e^{-\lambda_{i,j}T}}{k!} \tag{6}$$

D. Extended Contact Frequency

While the link weight modeled in Equation 6 integrates the contact frequency, contact duration, and buffer constraint, it does not consider the path diversity. Since the state of available buffer at the receiver side dynamically changes, a better path can have a smaller delivery probability at some time instances. In other words, a node with some message blocks cannot forward them at a contact with a better relay because the relay's buffer is full. Consider a path that consists of three nodes, v_i, v_j, and v_k, where v_j offers a buffer space for relaying message blocks. Assume that v_i sends x blocks to v_j at the first contact. Before v_i sends another x' blocks to v_j, the v_j's buffer must be cleared by forwarding some blocks to v_k. Thus, the forwarding opportunity from v_i to v_j is limited by the available buffer at v_j.

Since the available buffer size is time-varying, it is difficult to incorporate it into the link weight. Instead, we will define the effective contact frequency between two nodes, where the value of $1/\lambda_{i,j}$ increases when B_j (node v_j's available buffer) is small. We define t_f as the expected time that v_j forwards x blocks to the next node, which can be formulated as follows. The expected number of contacts that v_j sends out x blocks from its buffer is computed by dividing x by the average contact duration $E[Y_{j,k}]$, where $Y_{j,k}$ denotes the random variable of contact duration between v_j and v_k. Let $N(j)$ denote the open neighbor set of v_j. We define the average inter-contact time between v_j and $v_k \in N(j)$ by $\overline{\lambda}_j = \frac{1}{|N(j)|} \sum_{\forall v_k \in N(j)} \frac{1}{\lambda_{j,k}}$. By multiplying the resulting value with $\overline{\lambda}_j$, the additional forwarding time t_f can be obtained, and we have Equation 7.

$$t_f = \frac{x}{\min\left\{\overline{\lambda}_j \cdot \dfrac{1}{E[Y_{j,k}]}, B_{max}\right\}} \cdot \overline{\lambda}_j \tag{7}$$

Let $\lambda'_{i,j}(t_e, x)$ be the effective contact frequency between v_i and v_j, where t_e is the elapsed time after v_i sent x blocks to v_j. We may derive the effective inter-contact time, $1/\lambda'_{i,j}(t_e, x)$, as follows.

$$\frac{1}{\lambda'_{i,j}(t_e, x)} = \begin{cases} t_f + \dfrac{1}{\lambda_{i,j}} & \text{if } t_f < t_e \\ \dfrac{1}{\lambda_{i,j}} & \text{otherwiese} \end{cases} \tag{8}$$

By using the effective contact frequency instead of the contact frequency in Equation 6, the effective delivery probability can be obtained.

E. Path Weight

When the number of hops, denoted by η, between v_s and v_d is one, the path weight is simply obtained using Equation 6. When $\eta \geq 2$, we need to compute the convolution of $p_{i,j}(T)$ to approximate Equation 4. However, doing this is still too complicated and the computational cost is too high for mobile devices to perform. Therefore, by further simplifying assumptions, we derive $\hat{H}_{s,d}(T)$ to approximate the order of paths quantified by Equation 4. Then, the path with the highest value of $\hat{H}_{s,d}(T)$ is defined as the shortest path between v_s and v_d.

For DIR, we propose a simple path metric, called min-max. For a given path, the link with the smallest $p_{i,j}$ is considered the path weight and can be formulated by Equation 9.

$$\hat{H}_{s,d}(T) = \min_{\forall p_{i,j}(T) \texttt{ in a path}} \{p_{i,j}(T)\} \tag{9}$$

The above equation metric does not approximate Equation 4. Instead, the min-max metric can filter out the paths which contain low capacity links.

Example of the path weight Figure 1 shows a contact graph with four nodes. The contact frequencies and durations of each pair of nodes are depicted in the figure. There exist two paths from v_s to v_d, i.e., path 1 $= \{v_s, v_1, v_d\}$ and path 2 $= \{v_s, v_2, v_d\}$. Assume that the number of message blocks l is set as 3000, the message deadline is set as 1000, and each node can store up to 300 message blocks. At the beginning (i.e., $t = 0$), each link weight is initialized to be $p_{s,1}(T) \simeq 0.977$, $p_{1,d}(T) \simeq 0.934$, $p_{s,2}(T) \simeq 0.999$, and $p_{2,d}(T) \simeq 0.997$,

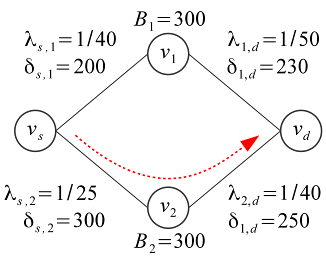

Fig. 1. A graph at $t = 0$.

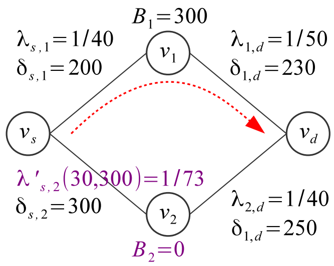

Fig. 2. A graph at $t = 30$.

respectively. Since link $e_{1,d}$ is the bottleneck, the resulting weight of path 1 is 0.93. Thus, path 2 is better than path 1 at $t = 0$.

Assume that v_s forwards 300 message blocks to v_2. At 30 unit times later (i.e., $t = 30$), the effective contact frequency between v_s and v_2 is smaller than the original $\lambda_{s,2}$. From Equation 8, we have that $\lambda'_{s,2}(30, 300) = 73$ and the resulting contact graph is given in Figure 2. The residual time is now $T' = 970$. Each link weight would be $p_{s,1}(T - t_e) \simeq 0.969$, $p_{1,d}(T - t_e) \simeq 0.916$, $p_{s,2}(T - t_e) \simeq 0.772$, and $p_{2,d}(T - t_e) \simeq 0.995$, respectively. Link $e_{s,2}$ becomes the bottleneck at $t = 30$, and therefore, path 1 would be considered the better path.

F. Data-Intensive Routing Protocol

Based on the proposed path model, we design the data-intensive routing (DIR) protocol whose pseudocode is given in Algorithm 1. Let v_s be the source node that wishes to deliver M ($|M| = l$) to v_d within T. Up to L copies of a message can be duplicated and each intermediate relay provides a buffer with size up to B_{max}. At first, v_s adds L copies of M to its buffer $v_s.buff$ by calling the $push(.)$ function.

Let v_i be the intermediate node (or the source node) holding m_x for some x in its buffer, and v_i meets v_j at t. If v_j is the destination v_d, v_i simply sends as many m_x as possible, as described in lines 3 to 10. That is, at every unit time, v_i polls message block m_x from $v_i.buff$ by calling $poll(.)$ and then transmits m_x to v_d. This continues until the link between v_i and v_d is disconnected or until v_i sends the last message block of a particular copy.

If v_j is not the destination, a forwarding decision is made based on the proposed path model provided in lines 11 to 18. Let pr_j be the shortest path from v_j to v_d. The path weight from v_j to v_d is computed by $\hat{H}_{j,d}(T-t)$ for up to η hops. In addition, v_i computes the weight pr_i of the shortest path from v_i to v_d in the cases where v_j is not used as a relay. If pr_j is greater than or equal to pr_i, v_j is a better relay to forward the message blocks. In this case, v_i forwards as many message blocks in its buffer as possible to v_j, until the link between v_i and v_j is disconnected, v_j's buffer exceeds its capacity, or v_j receives the last message block for a particular copy.

If v_d collects all the message blocks, m_x for all $1 \leq x \leq l$, by the deadline $t < T$, routing is considered successful.

G. Buffer Manipulation

Because a message exchanged among nodes is large in data-intensive routing, buffer management plays a critical role in efficient delivery. In this subsection, we propose two buffer manipulation techniques to improve performance. Our first strategy is the elimination of unnecessary forwarding. When the source node, v_s, meets the destination, v_d, it identifies the message block IDs that v_d has already received. The corresponding blocks in the source buffer at v_s are then removed in order to eliminate redundant message transmissions.

Our second technique is improving the delivery probability of the last message block in the source buffer. Figure 3 shows the inside of the source buffer, where $m_i^{(j)}$ denotes the i-th message block of the j-th copy of a message. Assume that the first and last blocks, $m_1^{(1)}$ and $m_l^{(1)}$, are transmitted at t_1 and t_2, respectively, from the source to an intermediate node. As it takes a long time for a source node to send a large message, $t_2 >> t_1$ most likely holds. In other words, $m_1^{(1)}$ has likely reached v_d by the time t_3 that the first block of the second copy, $m_1^{(2)}$, is transmitted from v_s to either v_d or an intermediate node. In order to take advantage of duplicate message blocks, the message blocks of the j-th copy ($j \geq 2$) are reordered in decreasing order of the block IDs, as shown in Figure 4.

Algorithm 1 DIR(v_s, v_d, M, L, T, B_{max})

1: /* Initialization: the source node v_s does the following. */
2: $push(v_s.buff, m_i^{(j)})$ for $1 \leq i \leq l$ and $1 \leq j \leq L$.
3: /* Forwarding: v_i holding m_x for some x does the following when it meets v_j at t. */
4: **if** $v_j = v_d$ **then**
5:     **repeat**
6:         $m_x \leftarrow poll(v_i.buff)$, where $1 \leq x \leq l$ holds.
7:         v_i sends m_x to v_d.
8:     **until** The link is disconnected or m_x is the last block
9:     **if** v_d collects m_x for all $1 \leq x \leq l$ **then**
10:         returns $SUCCESS$.
11: **else**
12:     v_i computes $pr_j \leftarrow \text{argmax}\{\hat{H}_{j,d}(T-t)\}$ up to η hops.
13:     v_i computes $pr_i \leftarrow \text{argmax}\{\hat{H}_{i,d}(T-t)\}$ up to η hops.
14:     **if** $pr_j \geq pr_i$ **then**
15:         **repeat**
16:             $m_x \leftarrow poll(v_i.buff)$, where $1 \leq x \leq l$ holds.
17:             v_i sends m_x to v_j.
18:         **until** The link is disconnected, $v_j.B_j = B_{max}$, or m_x is the last block
19: /* The failure to deliver M within T. */
20: **if** $t \geq T$ and v_d has not received m_x for some x **then**
21:     returns $FAIL$.

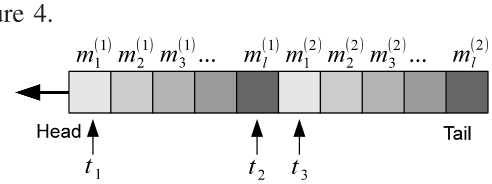

Fig. 3. The state of the source buffer with $L = 2$.

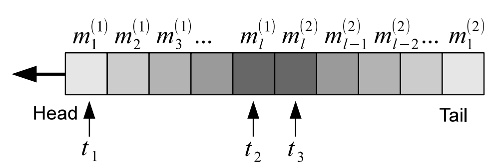

Fig. 4. The state of the source buffer with the buffer manipulation.

IV. Advanced DIR Protocol

A. Overview of Advanced DIR

In this section, we propose the advanced data-intensive routing (A-DIR) protocol by incorporating a smarter forwarding scheme into the DIR protocol in Section III-F. The idea of A-DIR is based on a simple observation: even though the first message block is forwarded via a non-shortest path (i.e., an inefficient path) at a given time instance, that block will likely have reached the destination by the time the last message block of the corresponding copy reaches its destination. Thus, in A-DIR, a forwarding decision is made as long as the probability that the first block at the head of the buffer will be delivered to the destination is greater than or equal to the probability that the last block of the corresponding copy of a message in the buffer will be delivered within the time constraint. A-DIR incurs slightly more number of message block forwarding than the original DIR, since some message blocks may travel a longer path. However, this observation yields higher delivery rate compared with the original DIR, as shown by the simulations using real mobility traces in Section V.

Note that the last message block does not mean the one at the tail of the buffer. Assume that the message blocks $m_i^{(j)}$ are pushed into the buffer in increasing order of the block IDs ($1 \leq i \leq l$) and the copy IDs ($1 \leq j \leq L$). If the message block at the head is $m_{100}^{(1)}$, the last message block for this copy of a message is defined as $m_l^{(1)}$, but not as $m_l^{(L)}$. On the other hand, a relay node does not have all the copies of a message, and thus, the last message block refers to the one with the largest block ID of the corresponding copy of the first block in its buffer. For example, assume that a relay node has 120 blocks with ID ranging from $m_{100}^{(1)}$ to $m_{179}^{(1)}$ and from $m_{80}^{(2)}$ to $m_{119}^{(2)}$. If $m_{100}^{(1)}$ is at the head of the buffer, the first and last blocks refers to $m_{100}^{(1)}$ and $m_{179}^{(1)}$, respectively.

B. Path Metric

The probability that the last block of a copy of a message will be delivered to the destination within a given time constraint is derived as follows. Let v_s be the source node (or the intermediate node holding some message blocks), and let v_s meet an intermediate node v_j at time t. Since we assume that one message block can be forwarded to a receiver in one unit time, the residual time to the deadline is computed by $T' = T-t-1$. Therefore, the probability that the first message block at the head of the source buffer will be delivered via an η-hop path from v_j to v_d within $T-t-1$, denoted by $\hat{H}_{j,d}^1(T')$ is obtained by the hypoexponential distribution as Equation 10.

$$\hat{H}_{j,d}^1(T') = \int_0^{T'} \sum_{k=1}^{\eta} A_k^{(\eta)} \lambda_k e^{-\lambda_k (T')} dt \tag{10}$$

$$= \sum_{k=1}^{\eta} A_k^{(\eta)} \left(1 - e^{-\lambda_k (T'-t-1)}\right) \tag{11}$$

Here, λ_k is the contact frequency of the k-th hop at the path. In addition, $A_k^{(\eta)}$ is the coefficient of the hypoexponential distribution, which is defined as Equation 12.

$$A_k^{(\eta)} = \prod_{j=1, j\neq k}^{\eta} \frac{\lambda_j}{\lambda_j - \lambda_k} \tag{12}$$

Next, we will approximate the probability that the last message block will be delivered to the destination within T under the opportunistic assumption that all the preceding blocks have already been delivered to v_d when the last group of blocks is polled from the buffer. Let v_j be one of the v_s's neighbors in a contact graph. Let l' be the residual number of blocks of a particular copy of a message in the v_s's buffer. As Equation 6, the probability that l' message blocks will be transmitted from one node to another is modeled by the Poisson process derived in Equation 5. After sending out all the blocks, v_s can forward a set of message blocks containing the last block. Assuming that the number of blocks of the last set is relatively small with respect to the average contact duration, the hypoexponential distribution can be again applied to model a path. Consequently, the probability that the last packet will be delivered to v_d within T' is computed in Equation 13.

$$\hat{H}_{s,d}^l(T, l') = \int_{t=0}^{T} p_{s,j}(T, l') \cdot \sum_{k=1}^{\eta} A_k^{(\eta)} \left(1 - e^{-\lambda_k (T-t)}\right) dt \tag{13}$$

A similar argument holds for modeling path weight from an intermediate node, say v_i, to v_d.

C. A-DIR Protocol

The A-DIR protocol is basically the same as the DIR protocol, whose pseudocode is provided in Algorithm 1. The difference is the forwarding decision described in lines 11 to 18.

The forwarding decision in A-DIR shall be made as follows. Let v_i be the source node or intermediate node with some message blocks in its buffer, and let v_i meet v_j at t. If v_j is the destination, v_i simply sends as many m_x in $v_i.buff$ as possible to v_j. When v_j is an intermediate relay, v_i computes the probability that the message block at the head of $v_i.buff$ will be delivered from v_j to v_d within T', where $T' = T - t - 1$ since sending one message block costs one unit time. That is, the weight of the η-hop shortest path from v_j to v_d, i.e., $pr_j \leftarrow \hat{H}_{j,d}^1(T')$ in Equation 10, is computed. Then, v_i computes the probability that the last message block will be delivered via any path from v_i to v_d within $T - t$. Let l' be the number of remaining message blocks for a particular copy in $v_i.buff$. Thus, the η-hop shortest path, i.e., $pr_i \leftarrow \hat{H}_{i,d}^l(T', l')$, is computed. The forwarding decision is made when the delivery probability of the first block is greater than or equal to the delivery probability of the last block. In other words, if $pr_j \geq pr_i$ holds, v_i forwards message blocks to v_j. Otherwise, it refrains from message forwarding.

If v_d collects all the message blocks within T, routing is considered to be successful, and otherwise message delivery fails.

Example of A-DIR Figure 5 illustrates a contact graph with four nodes, where v_s wishes to deliver M with l blocks

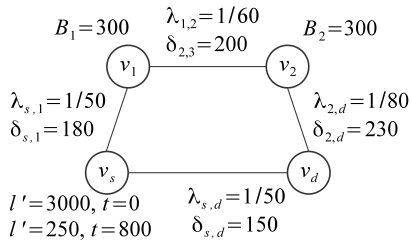

Fig. 5. An example of A-DIR.

to v_d. There are two paths from v_s to v_d, i.e., path 1 $= \{v_s, v_1, v_2, v_d\}$ and path 2 $= \{v_s, v_d\}$, and the pairwise contact frequency and duration are provided as shown in the figure. Clearly, path 2 is a better path than path 1. However, this is not necessarily the case when path diversity is considered. At the beginning, v_s has 3000 blocks in its buffer, i.e., $l' = 3000$ at $t = 0$. Assume that v_s meets v_1 at $t = 0$, and the remaining time to the deadline will be $T' = 3000 - 1 = 2999$. Then, v_s computes the delivery probabilities that the first block at the head of its buffer as well as the last block of the first copy will be delivered to their destination by Equations 10 and 13. We will have $\hat{H}^1_{1,d}(T', l') \simeq 0.999$ and $\hat{H}^l_{i,d}(T', l') \simeq 0.895$. Thus, though path 1 is not the shortest path, v_s concludes that it is better to forward message blocks to v_1 at this time.

Later, v_s meets v_1 again at $t = 800$. The remaining time and number of blocks are $T' = 200$ and $l' = 250$, respectively. The delivery probability of the first block at the buffer via path 1 is obtained by $\hat{H}^1_{1,d}(T', l') \simeq 0.779$. That of the last block is computed by $\hat{H}^l_{s,d}(T', l') \simeq 0.927$. Hence, v_s concludes that it would be better not to forward message blocks to v_1, and it will refrain from forwarding until it meets v_d.

V. Performance Evaluation

For performance evaluation, the proposed DIR and A-DIR protocols are implemented along with the source spray-and-wait (SW) as the baseline. Note that the DTN data-intensive protocols proposed so far are primarily designed for file replications and dissemination, and thus, they cannot be fairly compared as routing protocols.

A. Simulation Configurations

As a real mobility trace, the Infocom traces in CRAWDAD dataset Cambridge/haggle [15] are applied in our simulation. The traces contain contacts among stationary nodes (access points) and mobile nodes (iMotes), which are recorded over a five to six day period during the Infocom conference. Each contact contains two node IDs and two timestamps. One refers to the time when two nodes establish a connection; the other refers to the time when the connection fails. The contact duration can be computed by taking the difference between these two timestamps. However, since the traces are not well-refined, there are many contact events whose durations are zero (i.e., two timestamps are the same). In these cases, the contact duration is considered to be one, i.e., a node can forward one message block. In addition, we only consider contacts among iMotes, since our focus is on mobility-based protocol. There are 41 iMotes in Infocom 2005 (Experiment 3).

TABLE II
Simulation parameters.

Parameter	Value (default value)
The number of nodes	41 iMotes
The inter-contact time	Given by a trace
The contact duration	Given by a trace
The message deadline	1,000 to 300,000 unit time
The num. of message copies	1 to 5 (3)
The message size	500 to 5,000 blocks (3,000)
The buffer size	100 to 3,000 (300)
The num. of simulations	100

While simulation parameters related to node mobility and contacts are determined by a given real trace, the protocol parameters are set by ourselves. Our justification for this is that the message size is set to be larger than the size of data which can be transmitted in average contact duration and available buffer of relay nodes, i.e., at least 10 times larger than them on average. The protocol parameters are set as follows. One message consists of 500 to 5,000 blocks, and the number of copies of a message ranges from 1 to 5. Each intermediate node provides a buffer of 100 message blocks for a source and destination pair. The deadline of a message ranges from 1,000 to 200,000 unit time. Given a network realization, source and destination nodes are randomly selected, and the source node starts message transmission in the middle of a day. Routing is said to be successful if the destination receives all the blocks of a message within the deadline. For each configuration, 100 simulations are conducted.

The delivery rate is the ratio at which routing succeeds. The percentage of delivered message blocks is the number of message blocks that the destination node receives within the deadline divided by the total number of blocks of a message. Note that in some data types, e.g., media files, a portion of a file is still useful even when not all the blocks are collected. That is why we employ the percentage of delivered message blocks in addition to the delivery rate.

B. Simulation Results

Figure 6 shows the delivery rate for different protocols with respect to the message size. It is intuitive that the delivery rate decreases as the message size increases, since routing a large message requires more forwarding opportunities. The A-DIR protocol achieves the highest delivery rate. To be specific, the delivery rate of A-DIR is two times higher than SW when the message size is 5,000.

Figure 7 illustrates the delivery rate for different protocols with respect to the number of message copies. As the figure shows, neither the DIR nor A-DIR protocols benefit much from having more than two copies of a message. This is because the first message block of the first copy of a message likely reaches the destination when the first message block of the second copy of a message is sent out from the source node. As a consequence, increasing the number of message copies does not necessarily improve the delivery rate for data-intensive routing.

Figure 8 presents the delivery rate for different protocols with respect to the buffer size. The delivery rate of DIR and

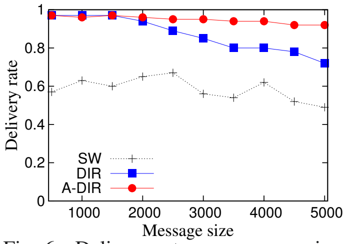

Fig. 6. Delivery rate vs. message size.

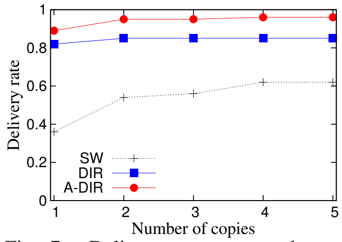

Fig. 7. Delivery rate vs. number of copies.

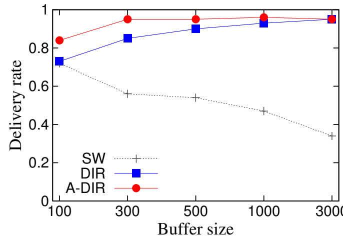

Fig. 8. Delivery rate vs. buffer size.

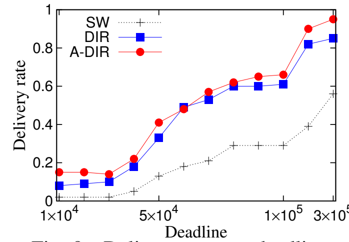

Fig. 9. Delivery rate vs. deadline.

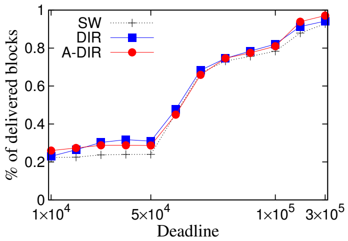

Fig. 10. Percentage of delivered message blocks vs. deadline.

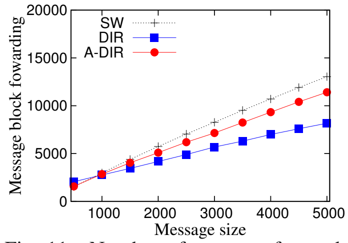

Fig. 11. Number of message forwarding vs. message size.

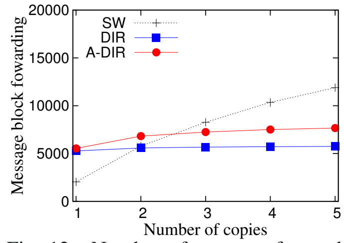

Fig. 12. Number of message forwarding vs. number of copies.

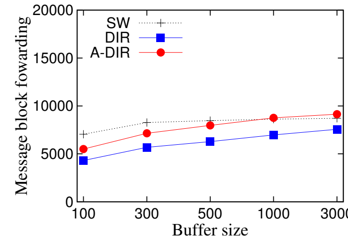

Fig. 13. Number of message forwarding vs. buffer size.

A-DIR increases when the buffer size increases from 100 to 500. However, significant improvement is not observed when the buffer size is more than 500. Note that the contact duration of most contact events ranges from 100 to 500 seconds. Therefore, we conclude that the buffer size is sufficient for data-intensive routing as long as it is greater than the average contact duration. On the other hand, the delivery rate of SW decreases as the buffer size increases. This is because the source node may forward message blocks to a worse relay node which will not have sufficient contact duration with the destination. As a result, the coupon collector's problem [20], i.e., some message blocks are not delivered to the destination, occurs with a higher probability. DIR avoids this issue by taking the bottleneck link as the path weight, and A-DIR addresses this by always relaying the last message block to a better relay.

Figure 9 gives the delivery rate for different protocols with respect to the message deadline. The delivery rate increases as the message deadline increases. Both the DIR and A-DIR protocols show significant improvement compared to SW. From this figure, we can say that smarter forwarding decisions are made in the proposed protocols.

Figure 10 depicts the percentage of the message blocks delivered to the destination with respect to the deadline. Unlike the delivery rates shown in Figure 9, all the protocols show similar performance. This indicates that, in SW, some blocks enter deadlock state and never reach the destination. In addition, the percentage of delivered message blocks rapidly increases between 70,000 and 100,000 seconds, because there are many contacts during this period. In real traces, there are many contacts during business hours, and almost no contact events occur during the night.

Figure 11 presents the number of message block forwardings for different protocols with respect to the message size. The total number of message blocks for complete message delivery increases in proportion to the message size. As a result, the number of message fowarding of all the protocols increases as the message size increases. The A-DIR protocol incurs a smaller message overhead than SW, even though it provides a higher delivery rate.

Figure 12 gives the number of message block forwardings for different protocols with respect to the number of copies. In SW, the source node must refrain from forwarding message blocks until it meets the destination when the number of message copies equals one. Thus, the number of message forwardings in SW linearly increases as the number of copies increases. Message blocks are forwarded any time a better intermediate node is found. Furthermore, as discussed in Section III-G, unnecessary message blocks are eliminated from the source buffer. Therefore, the message overheads of DIR and A-DIR are mostly constant, regardless of the number of message copies.

Figure 13 depicts the number of message block forwardings for different protocols with respect to the buffer size. In general, there will be more forwarding opportunities, when the buffer constraint is alleviated. As seen in the figure, the message overhead of DIR is smaller than that of the other protocols. The A-DIR protocol incurs slightly a smaller overhead than SW does. Figures 11, 12, and 13 demonstrate that the proposed DIR and A-DIR protocols achieve higher delivery rates with lower overhead than a straight-forward approach.

VI. Related Works

A. DTN Routing Protocols

Any DTN routing protocol relies on the principle of *store-and-carry* which utilizes nodal mobility to accommodate the limited transmission opportunities that result from the disruptive nature of DTNs. The simplest routing is achieved by a flooding-like protocol called Epidemic [5] which forwards a message at every contact. While Epidemic maximizes the delivery rate without buffer constraint, it also causes a high level of message overhead. To alleviate this, the number of message transmissions can be controlled by tickets. In the spray-and-wait protocol [6], each node can duplicate copies of a message up to the number of tickets it has. A better forwarding decision can be made based on the knowledge oracle [7]; the use of past contact history improves the delivery rate [8] and reduces redundant message replications [9]. To optimize a particular metric, e.g., the average delay, missed deadlines, and maximum delay, Balasubramanian et al. [10], [18] proposes a set of utility functions. PRoPHET [16] and MaxProp routing [17] consider the diversity of paths to improve the delivery rate. However, none of them was intended to handle large data. CAD routing [19] incorporates the contact duration into path metric, but their delivery probability estimation is limited to up to two hops. In addition, only message level forwarding is considered in CAD, i.e., a node can forward a message to a relay node, only when the contact duration is long enough for the relay node to receive the entire message during one contact.

B. Data-Intensive Protocols in DTNs

To disseminate a large amount of data in DTNs, Gao et al [12], [13] introduced collaborative caching. In [12], an opportunistic multi-hop path is modeled by the hypoexponential distribution, which is the convolution of the exponential distribution. Then, the shortest path is defined as the path with the highest probability of a message being delivered within the deadline. Gao et al. [13] further adressesed opportunistic data update to maintain the freshness of cache in DTNs. To handle a large amount of data, Zhao et al. [14] addressed the data replication problem, where consideration of contact duration at each contact is incorporated into modeling the link weight by applying the Pareto distribution. However, their model covers only the one-hop link, not the multi-hop opportunistic path.

There are some works [21], [22] on delay-tolerant routing primarily designed for the Internet and data center networks. However, these researches do not address the mobility issue, which is the primary concern for DTN routing.

VII. Conclusion

In this paper, we first introduce a new routing paradigm in DTNs, namely data-intensive routing, which differs from the traditional DTN routing because the data routed from the source to the destination is large with respect to the link capacity and the buffer size at relay nodes. For efficient delivery, we propose a new contact-duration-aware opportunistic path model that integrates contact-frequency, contact duration, and buffer constraint into single routing domain. In addition, we incorporate the expected contact frequency into the path metric to utilize the path diversity. Based on the proposed path model, we propose a DIR protocol in which the path weight is defined as the min-max metric. In addition, we propose the advanced DIR (A-DIR) protocol with a smarter path metric, which focuses on the probability that the last message block will be delivered to the destination within the time constraint under the opportunistic assumption that the other blocks will have been delivered by the time the last block reaches its destination. The performance evaluation of the proposed DIR and A-DIR protocols are conducted by simulations with a well-known real mobility trace, CRAWDAD dataset Cambridge/haggle [15], and we demonstrate that both DIR and A-DIR protocols achieve their design goals.

Acknowledgments

This research has been funded in part by the JSPS KAKENHI Grant Number JP17K12675, by the Taiwan MOST grants 107-PFA2-550-448, 107-2218-E-001-006, and 107-2221-E-008-082-MY2, and by the U.S. National Science Foundation grants IIS-1618669 (III) and ACI-1642133 (CICI).

References

[1] S. Wang, M. Liu, X. Cheng, and M. Song, "Routing in Pocket Switched Networks," *IEEE Trans. Wireless Commun.*, vol. 19, no. 1, pp. 67–73, 2012.
[2] Z. Lu, G. Cao, and T. L. Porta, "Networking Sartphones for Disaster Recovery," in *PerCom*, 2016, pp. 1–9.
[3] H. Jin, L. Su, B. Ding, K. Nahrstedt, and N. Borisov, "Enabling Privacy-Preserving Incentives for Mobile Crowd Sensing Systems," 2016, pp. 344–353.
[4] X. S. Zongqing Lu and T. L. Porta, "Cooperative Data Offloading in Opportunistic Mobile Networks," in *Infocom*, 2016, pp. 1–9.
[5] A. Vahdat and D. Becker, "Epidemic Routing for Partially-Connected Ad Hoc Networks," Tech. Rep., 2000, CS-200006, Duke University.
[6] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks," in *SIGCOMM Workshop on Delay-Tolerant Networking*, 2005, pp. 252–259.
[7] S. Jain, K. Fall, and R. Patra, "Routing in a Delay Tolerant Network," in *SIGCOMM*, 2004, pp. 145–158.
[8] C. Liu and J. Wu, "An Optimal Probabilistic Forwarding Protocol in Delay Tolerant Networks," in *Mobihoc*, 2009, pp. 105–114.
[9] W. Gao, Q. Li, and G. Cao, "Forwarding Redundancy in Opportunistic Mobile Networks: Investigation and Elimination," in *Infocom*, 2014, pp. 2301–2309.
[10] A. Balasubramanian, B. Levine, and A. Venkataramani, "DTN Routing As a Resource Allocation Problem," in *SIGCOMM*, 2007, pp. 373–384.
[11] M. Chen, S. Mao, and Y. Liu, "Big data: A survey," *MONET*, vol. 19, no. 2, pp. 171–209, Apr. 2014.
[12] W. Gao, G. Cao, A. Iyengar, and M. Srivatsa, "Cooperative Caching for Efficient Data Access in Disruption Tolerant Networks," *IEEE Trans. Mob. Comput.*, vol. 13, no. 3, pp. 611–625, 2014.
[13] W. Gao, G. Cao, M. Srivatsa, and A. Iyengar, "Distributed Maintenance of Cache Freshness in Opportunistic Mobile Networks," in *ICDCS*, 2012, pp. 132–141.
[14] J. Zhao, X. Zhuo, Q. Li, W. Gao, and G. Cao, "Contact Duration Aware Data Replication in DTNs with Licensed and Unlicensed Spectrum," *IEEE Trans. Mob. Comput.*, vol. 15, no. 4, pp. 803–816, 2016.
[15] J. Scott, R. Gass, J. Crowcroft, P. Hui, C. Diot, and A. Chaintreau, "CRAWDAD dataset cambridge/haggle (v. 2009-05-29)," Downloaded from http://crawdad.org/cambridge/haggle/20090529, May 2009.
[16] A. Lindgren, A. Doria, E. Davies, and S. Grasic, "Probabilistic Routing Protocol for Intermittently Connected Networks," in *RFC*, 2012, p. 6683.
[17] J. Burgess, J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine, "MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks," in *Infocom*, 2006, pp. 1–11.
[18] A. Balasubramanian, B. Levine, and A. Venkataramani, "Replication Routing in DTNs: A Resource Allocation Approach," *IEEE/ACM Trans. Netw.*, no. 2, pp. 596–609, 2010.
[19] T. Le and M. Gerla, "Contact Duration-Aware Routing in Delay Tolerant Networks," in *Infocom*, 2017, pp. 1–8.
[20] R. Motwani and P. Raghavan, *3.6 The Coupon Collector's Problem, Randomized Algorithms*. Cambridge University Press, 1995, no. 57–63.
[21] N. Laoutaris, M. Sirivianos, X. Yang, and P. Rodriguez, "Inter-Datacenter Bulk Transfers with Netstitcher," in *SIGCOMM*, 2011, pp. 74–85.
[22] N. Laoutaris, G. Smaragdakis, P. Rodriguez, and R. Sundaram, "Delay tolerant bulk data transfers on the internet," *SIGMETRICS Perform. Eval. Rev.*, vol. 37, no. 1, pp. 229–238, 2009.