Multicasting in Delay Tolerant Networks: Semantic Models and Routing Algorithms [*]

Wenrui Zhao, Mostafa Ammar and Ellen Zegura
College of Computing, Georgia Institute of Technology, Atlanta, Georgia 30332
{wrzhao, ammar, ewz}@cc.gatech.edu

ABSTRACT

Delay tolerant networks (DTNs) are a class of emerging networks that experience frequent and long-duration partitions. These networks have a variety of applications in situations such as crisis environments and deep-space communication. In this paper, we study the problem of *multicasting* in DTNs. Multicast supports the distribution of data to a group of users, a service needed for many potential DTN applications. While multicasting in the Internet and mobile ad hoc networks has been studied extensively, due to the unique characteristic of frequent partitioning in DTNs, multicasting in DTNs is a considerably different and challenging problem. It not only requires new definitions of multicast semantics but also brings new issues to the design of routing algorithms. In this paper, we propose new semantic models for DTN multicast and develop several multicast routing algorithms with different routing strategies. We present a framework to evaluate these algorithms in DTNs. To the best of our knowledge, this is the first study of multicasting in DTNs. Our objectives are to understand how routing performance is affected by the availability of knowledge about network topology and group membership and to guide the design of DTN routing protocols. Using *ns* simulations, we find that efficient multicast routing for DTNs can be constructed using only partial knowledge. In addition, accurate topology information is generally more important in routing than up-to-date membership information. We also find that routing algorithms that forward data along multiple paths achieve better delivery ratios, especially when available knowledge is limited.

Categories and Subject Descriptors

C.2.1 [**Network Architecture and Design**]: Wireless communication; C.2.2 [**Network Protocol**]: Routing protocols

General Terms

Algorithms, Design, Performance

Keywords

Multicast, delay tolerant networks, semantic model

1. INTRODUCTION

Delay tolerant networks (DTNs) are a class of emerging networks that experience frequent and long-duration partitions [9, 12]. There is no end-to-end path between some or all nodes in a DTN. These networks have a variety of applications in situations that include crisis environments like emergency response and military battlefields, deep-space communication, vehicular communication, and non-interactive Internet access in rural areas [1, 4, 10, 14, 19, 20, 21, 22, 23].

In this paper, we study the problem of *multicasting* in delay tolerant networks. Multicast service supports the distribution of data to a group of users. Many potential DTN applications operate in a group-based manner and require efficient network support for group communication. For example, in a disaster recovery scene, it is vital to disseminate information about victims and potential hazards among rescue workers. In a battlefield, soldiers in a squad need to inform each other about their surrounding environment. Although group communication can be implemented by sending a separate unicast packet to each user, this approach suffers from poor performance, which is confirmed in our simulations. The situation is especially acute in DTNs where resources such as connectivity among nodes, available bandwidth and storage are generally severely limited. Thus efficient multicast services are necessary for supporting these applications.

Multicasting in the Internet and mobile ad hoc networks (MANETs) has been studied extensively in the past [3, 6, 7, 8, 15, 16, 18]. However, due to the unique characteristic of frequent partitioning in DTNs, multicasting in DTNs is a considerably different and challenging problem. It not only requires new definitions of multicast semantics but also brings new issues to the design of routing algorithms.

The semantics of multicasting in traditional networks such as the Internet and MANETs are straightforward, specifying that packets sent to a multicast group be delivered to members of the group. Since data transfer delay in these networks is short (on the order of milliseconds), group membership changes during data transfer are rare and can be ignored. Thus the receivers of a multicast packet are well defined, i.e., all current group members. This, however, is no longer valid in DTNs. Due to frequent partitions and consequently large transfer delays in a DTN, membership changes during data transfer are the norm rather than the exception. Under these situations, it is not obvious how to define the receivers of a multicast packet, relative to the group membership over time.

Consider a simple example where a source sends a message to a group at time t. Let t' be the earliest time that other nodes could

---

[*] This work was supported by the NSF Grant ITR-0313062 and AFOSR MURI Grant F49620-00-1-0327.

SIGCOMM'05 Workshops, August 22–26, 2005, Philadelphia, PA, USA.

possibly receive this message according to network topology limitations. Suppose that node A joins the group at time $t_1 < t$ and leaves at time t_2, $t < t_2 < t'$. Node B joins at time t_3, $t < t_3 < t'$ and never leaves. From the perspective of traditional multicasting, it is not clear which nodes should receive this message, whether A, B, both or neither of them. For node A, it is a group member at the time of message generation but no longer a member at the earliest time of potential message delivery. The reverse is true for node B. To address this problem, new semantic models are needed for DTN multicasting.

In this paper, we develop new multicast semantic models for DTN environments that have explicit constraints on group membership and delivery action. These semantic models unambiguously define the receivers of a multicast packet and have various applications in DTN environments.

With these semantic models, we study the problem of multicast routing in DTNs. DTNs introduce several challenges for routing. First, there may be no end-to-end path between nodes in DTNs. Traditional routing algorithms would fail to deliver data because no route is found to reach the destinations. Thus multicast routing in DTNs needs to operate in the presence of network partitions. Second, as proposed in [9], data transfer in DTNs is in application data units called *messages* (or bundles). This is different from the use of flows in traditional multicasting. Third, information about nodes joining or leaving a group may be available to nodes only after significant delays because of network partitions. Multicast routing algorithms need to handle these highly delayed join or leave requests. Finally, the multicast semantic models developed in this paper also introduce new requirements for message forwarding.

We study four classes of multicast routing algorithms for DTNs with different routing strategies. To understand routing performance in DTN environments where available routing information may be significantly limited by network partitions, we present an evaluation framework that models different levels of available knowledge about network topology and group membership. This is an extension of the framework for unicast routing developed in [12]. Our objectives are to understand the impact of the availability of knowledge on routing performance and to guide the design of DTN routing protocols. With extensive *ns* [17] simulations, we evaluate various routing algorithms. We find that efficient routing for multicast can be constructed using only partial knowledge, as in the case of unicast [12]. In addition, accurate topology information is generally more important in routing than up-to-date membership information. Furthermore, routing algorithms that forward data along multiple paths achieve better delivery ratios, especially when available knowledge is limited. Finally, our results confirm that unicast-based approaches that send a separate copy of messages to each receiver perform poorly in DTNs.

The rest of this paper is structured as follows. Section 2 describes the network model and briefly reviews unicast routing in DTNs. In Section 3, we present new semantic models for DTN multicasting. We describe an evaluation framework for multicast routing in Section 4. Section 5 presents the four classes of multicast routing algorithms. We present simulation results in Section 6 and review related work in Section 7. The paper is concluded in Section 8.

2. DTN NETWORK MODEL

In this section, we present the network model considered in this paper and briefly review unicast routing in DTNs.

2.1 Network Model

We assume that nodes in DTNs are identified by a unique ID. An *endpoint* is an entity at a node that acts as the source or destination of communication, e.g., an application at the node. An endpoint is identified by an *endpoint ID* which is a tuple $(node\_id, entity\_id)$ where $entity\_id$ uniquely identifies an endpoint within a node[1]. We assume a message-oriented service, where endpoints communicate using application data units called *messages*. In addition, nodes are assumed to have loosely synchronized clocks.

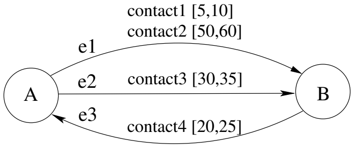


Figure 1: An example of DTN graphs.

In our model, multicasting disseminates messages to a group of endpoints that are identified by a *group ID*. A group ID is a globally unique ID that has the same form as endpoint IDs. A multicast message encodes a group ID as the destination endpoint. In order to receive messages destined to a specific group, endpoints join the group by indicating a JOIN request with the group ID to the DTN routing agent at the node. Similarly, an endpoint leaves a group using a LEAVE request to stop receiving messages for a group. Routing agents in a DTN may authenticate endpoints and authorize JOIN or LEAVE requests according to administrative policies. However, in this paper, we consider a general multicast model in which endpoints can join and leave groups autonomously.

In a DTN, network partitions may occur frequently. To overcome disconnections, data is forwarded in a *store-carry-and-forward* fashion, i.e., a node buffers messages in its storage until connections with other nodes become available. We assume that node storage is used for holding in-transit messages only. Delivered messages are stored in separate application buffers.

2.2 Unicast Routing in DTNs

We now briefly describe the unicast routing in DTNs developed by Jain et al. [12]. A DTN is represented as a directed multi-graph. Thus there may exist multiple edges between two nodes. Each edge represents a connection between nodes and has time-varying capacity and propagation delay that represent the properties of the connection over time. The capacity of an edge is zero when the corresponding connection is unavailable. A *contact* is defined as an opportunity to send data between nodes, i.e., an edge and the time interval during which the edge capacity is positive. Fig. 1 shows a DTN graph in which there are three edges between node A and node B. Contacts are shown along each edge with their time intervals. We can see that there are two contacts for edge e_1, from time 5 to 10 and from time 50 to 60 respectively.

Given the time-varying capacity and delay of edges in a DTN, routing decisions vary with time. Suppose that node A sends messages to node B and the routing objective is to minimize the message transfer delay. For simplicity of presentation, we ignore the transmission delay and propagation delay at the edges. If a message arrives at node A at time 0, the optimal route to node B is via $contact1$ of edge e_1 which has the minimum delay of 5. If a message arrives at time 15, however, the optimal route would be via $contact3$ of edge e_2 since $contact1$ is no longer available. To compute the shortest (or minimum delay) paths in a DTN graph, Jain et al. [12] develop a modified Dijkstra's algorithm. The key difference in the modified algorithm is to take into account of the time of message arrivals at each node and only consider contact opportunities after message arrivals. In this paper, we assume that this

---

[1] Other addressing schemes are certainly possible, however, we focus on this fairly standard scheme in this paper.

algorithm is used to compute routes in a DTN graph and use the term “shortest path” and “minimum delay” of a message to refer to the path computed by this algorithm and the corresponding delay of forwarding the message along the computed path respectively. Readers should refer to [12] for more details about this algorithm.

3. MULTICAST SEMANTIC MODELS

In this section, we will present three new semantic models for multicasting in DTNs[2]. As discussed earlier, due to large transfer delays in DTNs, group membership may change during a message transfer, introducing ambiguity in multicast semantics. Under these situations, it is necessary to make a distinction between *group members* and the *intended receivers* of a message, i.e., endpoints to which the message should be delivered. Group members may change with time as endpoints join and leave the group. The intended receivers, on the other hand, should be fixed for a message, even though they are defined based on group membership. We develop three multicast semantic models that allow users to explicitly specify temporal constraints on group membership, unambiguously defining the intended receivers of a message. These models also specify constraints on the action of message delivery and have important applications in DTN environments.

3.1 Temporal Membership Model

To determine the receivers of a multicast message, we need to explicitly specify the time during which the intended receivers are defined. One straightforward approach is to define the receivers of a message as the group members at the time of message generation. In this paper, we consider a more general semantic model, called *Temporal Membership (TM)*. In the TM model, a message includes a *membership interval* that specifies the period during which the group members are defined. For a message with group ID G and membership interval $[t_1,t_2]$, the intended receivers of the message consist of endpoints who are members of group G at *any* time during period $[t_1,t_2]$[3]. Under the TM model, the receivers of a message are well defined. In the TM model, there is no delivery constraint so messages can be delivered at any time. Note that the intended receivers of a message may be different from the *actual receivers* which actually receive the message. The actual receivers are a subset of the intended receivers and dependent on the routing algorithm used and the traffic condition in the network.

The TM model allows users to flexibly specify the time-based characteristics of the receiving group of a message, which has some interesting applications in DTNs. One potential application of the TM model is in mobile sensor networks where mobile sensors record sensory data along their movement trajectories. Each region of interest is associated with a multicast group and sensor nodes join or leave multicast groups based on their locations. To query the status of a given region during a specific period, a user can send a multicast message to the group that is associated with the region with a specific membership interval. The message will be delivered to sensors that are in the region during the specified period.

Consider an example in Fig. 2 which shows a DTN at time 0 when a message is generated at node S. If the membership interval of the message is $[0,1]$, the intended receivers are $\{R_1,R_2,R_3,R_4\}$. If the membership interval is $[15,20]$, the intended receivers become $\{R_1,R_3,R_4\}$ since R_2 is no longer a group member during this period.

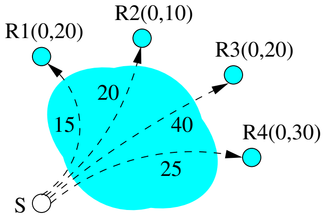


Figure 2: An example of multicast semantic models. The figure shows a DTN at time 0 when a message for group G is generated at node S. The dashed lines are the shortest paths from S to other endpoints with the minimum delay shown along each path. The time interval during which an endpoint is a member of group G is shown next to the endpoint.

3.2 Temporal Delivery Model

Our second model is the *Temporal Delivery (TD)* model. In this model, messages specify additional constraints on the action of message delivery beyond the unconstrained TM model. A message specifies both a membership interval and a *delivery interval*. The delivery interval indicates the time period during which the message should be delivered to the intended receivers, as will be defined below. Note that the message can be delivered to nodes hosting the intended receivers *before* that period since nodes can delay forwarding the message to endpoints[4].

To be consistent with this delivery constraint, the intended receivers of a message should *exclude* endpoints that are not able to receive the message during the delivery interval. Let R be the set of all endpoints in the network and $member(r,t',t'')$ be a predicate on whether endpoint r is a group member during period $[t',t'']$. Let t_0 be the message generation time and $d(t,r)$ be the minimum delay from the source of the message to endpoint r starting at time t. As described in Section 2, $d(t,r)$ can be computed using the modified Dijkstra’s algorithm in [12]. For a message with group ID G, membership interval $[t_1,t_2]$ and delivery interval $[t_3,t_4]$, the set $I_{\mathcal{TD}}$ of intended receivers is defined as

$$I_{\mathcal{TD}} = \{r | member(r,t_1,t_2) = true \textbf{ and } d(t_0,r)+t_0 < t_4, r \in R\}. \tag{1}$$

Note that while the delivery interval specifies that the message be delivered no earlier than t_3, the definition of $I_{\mathcal{TD}}$ does not require an earliest time for the message to reach a node. This is because nodes can delay forwarding the message to endpoints. The TD model is more general than the TM model, which is a special case with delivery interval $[t_0,\infty)$.

The TD model enables users to have additional control on when messages are delivered. In addition, a delivery interval specifies an expiration time for a message. This enables routing algorithms to remove messages that are not able to meet the delivery intervals and reclaim storage space, which is crucial in DTNs since nodes may need to buffer messages for a significantly long period.

Consider an example using Fig. 2. For a message with membership interval $[0,1]$ and delivery interval $[0,35]$, the intended receivers are $\{R_1,R_2,R_4\}$. R_3 does not meet the delivery interval

---

[2] Other models are possible, however these models seem to capture the needs of many applications. Further experience with DTN applications will help clarify which semantics are most useful.

[3] An alternative and complementary model would require endpoints to be group members throughout period $[t_1,t_2]$. As an initial effort, we focus on the TM model in this paper.

[4] An alternative model would require the message be delivered to nodes hosting the intended receivers only during the delivery interval. Under this model, a node can act as a relay for messages only when there is no receiver at the node. This requires that either nodes have knowledge about future group membership of local endpoints, or all relaying nodes are not data sources or destinations.

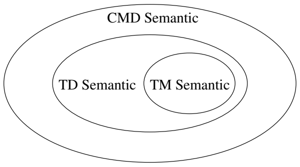


Figure 3: DTN multicast semantic models.

since it could receive the message no earlier than time 40, hence R_3 is not an intended receiver of this message.

3.3 Current-Member Delivery Model

In both the TM and TD models, receivers of a message are not required to be group members at the time of message delivery. In our third model, the *Current-Member Delivery (CMD)* model, messages explicitly specify whether this requirement should be met. A message includes a *CMD flag* as well as a membership interval and a delivery interval. When the CMD flag is set, the receivers of the message should be group members at the time of message delivery. In addition, the message should be delivered during the delivery interval as in the TD model. When the CMD flag is not set, the CMD model reduces to the TD model, thus the CMD model is a more general model. Fig. 3 depicts the relationship among these semantic models.

We now define the intended receivers $I_{\mathcal{CMD}}$ of a message in the CMD model. When the CMD flag is set, $I_{\mathcal{CMD}}$ should exclude endpoints that are not able to be group members at the time of message delivery. Using the same notations as in the previous section, we define $I_{\mathcal{CMD}}$ as follows

$$I_{\mathcal{CMD}} = \{r | r \in I_{\mathcal{TD}} \textbf{ and } member(r, t_m, t_4) = true\} \qquad (2)$$

where t_m is $\max(d(t_0, r) + t_0, t_3)$, the earliest time that the message could be delivered to endpoint r because of the transfer delay from the source to r and the delivery interval constraint. In order to meet the CMD constraint, it is necessary that r be a group member during period $[t_m, t_4]$.

Consider an example using Fig. 2. For a message with membership interval $[0, 1]$, delivery interval $[0, 35]$ and the CMD flag set, the intended receivers are $\{R_1, R_4\}$. R_2 is not an intended receiver because it could not be a group member at the time of message delivery which is at least time 20.

4. MULTICAST ROUTING FRAMEWORK

Given these semantic models, we now turn to the problem of multicast routing in DTNs. In this section, we present a framework for evaluating multicast routing in DTNs. This is an extension of the framework for unicast routing developed in [12]. Routing algorithms generally use various information about network conditions to achieve better performance. Due to network partitions in DTNs, however, there might not be complete or current knowledge available, degrading routing performance. In this paper, we study the fundamental trade-off between the amount of available knowledge and the achieved performance. To model the availability of knowledge, we use abstract *knowledge oracles* that encapsulate particular knowledge about network status to be used in routing algorithms [12]. While the use of knowledge oracles does not consider how such knowledge is actually disseminated in the network and how much overhead it causes, this approach isolates the effects of knowledge availability on routing performance, which would provide insight to guide the design of routing protocols in DTNs.

In the following, we first discuss the routing objectives in DTN multicasting. We then describe various knowledge oracles and present an overview of four routing approaches.

4.1 Routing Objectives

For any routing algorithm, a basic objective is to maximize the probability of delivering messages. In this paper, we evaluate multicast routing algorithms by the *message delivery ratio* which is the ratio between the number of endpoints that receive a message and the number of intended receivers of the message according to the semantic model used. This metric measures how successful a routing algorithm is in delivering messages. In addition, we define the *routing efficiency* of an algorithm as the ratio between the total amount of delivered messages and the total amount of traffic generated in the network. This metric measures how efficient a routing algorithm is in utilizing resources.

In this paper, we study several classes of routing algorithms that are expected to achieve different balance of delivery ratio and routing efficiency. For each class of algorithms, we focus on minimizing the delay for each intended receiver.

4.2 Knowledge Oracles

We consider knowledge oracles about both contact opportunities and group membership. Contact oracles provide information about network topology, while group membership oracles answer questions about group dynamics, e.g., the events of an endpoint joining or leaving a group. Due to space limits, we consider the following oracles in this paper. More oracles are considered in a longer version of this paper [24].

- *Contact Summary Oracle.* This oracle can answer questions about the long-term statistics regarding network topology, i.e., average time between contact occurrences and average contact duration.

- *Complete Contact Oracle.* This oracle can answer any question about network topology at any time, including the exact time when a contact occurs, the duration, capacity and delay of the contact.

- *Delayed Membership Oracle.* For an endpoint r and a node S that queries the oracle, this oracle can answer questions about membership of endpoint r up to a specific time t. t is the latest time that satisfies $d(t, S) + t \leq t_0$ where t_0 is the current time and $d(t, S)$ is the minimum delay from endpoint r to node S starting at time t[5]. In other words, if endpoint r joins or leaves a group at or before time t and sends this information to other nodes by flooding, assuming no contending traffic in the network, node S should have received this information by the time of querying the oracle.

- *Complete Membership Oracle.* This oracle can answer questions about group membership of all nodes at any time.

4.3 Routing Approaches

We now describe four approaches for multicast routing in DTNs which are adopted from multicasting in the Internet or MANETs. Fig. 4 depicts simple examples of these approaches.

- *Unicast-Based Routing (UBR).* This approach implements multicast service by using unicast transfer, i.e., the source will send a copy of the message to every intended receiver.

---

[5] The minimum delay depends on the message size since it affects the transmission delay. In the delayed membership oracle, we try to model the availability of knowledge that is limited by the network topology. Thus we are not concerned about the actual transmission of membership information and use a message size of zero in computing this delay.

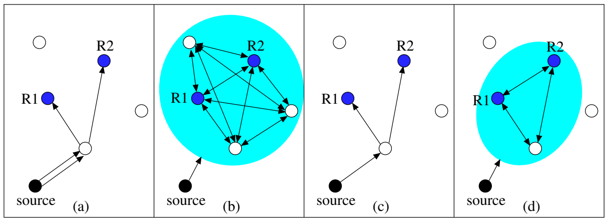


Figure 4: Routing approaches in DTNs. (a) unicast-based routing (b) broadcast-based routing (c) tree-based routing (d) group-based routing.

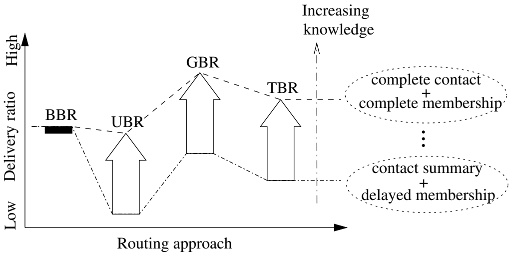


Figure 5: Conceptual performance of various routing approaches under different levels of knowledge.

- *Broadcast-Based Routing (BBR).* In BBR or epidemic routing [20], messages will be flooded throughout the network in order to reach the intended receivers.

- *Tree-Based Routing (TBR).* In TBR, messages are forwarded along a tree in the DTN graph that is rooted at the source and reaches all receivers. Messages are duplicated only at nodes that have more than one outgoing path.

- *Group-Based Routing (GBR).* GBR uses the concept of *forwarding group* [6] which is a set of nodes that are responsible for forwarding the message. Messages will be flooded within the forwarding group to increase the chance of delivery.

Fig. 5 summarizes the conceptual performance of various routing approaches under different levels of available knowledge. BBR is expected to achieve the same delivery ratio under different amount of available knowledge. The expected delivery ratio of other approaches (i.e., UBR, GBR and TBR) would improve with the increasing knowledge. GBR is expected to perform best while UBR is the worst. These approaches would also achieve different routing efficiency.

5. MULTICAST ROUTING ALGORITHMS

In this section, we describe multicast routing algorithms for DTNs based on the four routing approaches and various knowledge oracles described in the previous section. We first describe the common operations and then the specifics of each algorithm. More details can be found in [24].

5.1 General Operations

The general operations of these algorithms are sketched in Fig. 6. When a message arrives, either generated by a local endpoint or received from another node, a node estimates the intended receivers of the message. If there are local intended receivers, the message is forwarded to these receivers according to the semantic model. The message is then buffered in node storage and forwarded to other nodes when contacts become available. Since data transfer in DTNs is based on messages, nodes maintain forwarding state for each buffered message, which is updated as group membership changes. We now describe how nodes buffer messages, maintain forwarding state and forward messages.

5.1.1 Message Buffering

In DTN multicast routing, messages will be buffered in node storage until being deleted due to buffer overflows or being expired according to the semantic models. This improves the availability of messages such that nodes other than the source can handle join requests and send buffered messages to new receivers. In this paper we adopt an age-based buffering policy which removes the oldest message when the buffer overflows, thus giving new messages opportunities to be delivered.

5.1.2 Forwarding State

Nodes maintain local forwarding state for each message buffered in node storage. Each message is associated with a NEXT-HOP list $\mathcal{L}_n$ that records nodes to which this message should be sent, and a SENT list $\mathcal{L}_s$ that consists of nodes that have already received this message. $\mathcal{L}_n$ is initialized upon the message arrival and updated when group membership changes. To compute $\mathcal{L}_n$, nodes need to estimate the intended receivers of a message based on the current available knowledge. We will describe how to calculate $\mathcal{L}_n$ for various algorithms in the next section. $\mathcal{L}_s$ is initially set to empty. In the rest of this paper, we use $\mathcal{L}_n(m)$ and $\mathcal{L}_s(m)$ to denote the NEXT-HOP and SENT lists for message m respectively.

5.1.3 Message Forwarding

Given message forwarding state $\mathcal{L}_n(m)$ and $\mathcal{L}_s(m)$, nodes forward messages as follows. Suppose that a contact between node A and B becomes available. For each buffered message m, node A will try to forward message m to node B if B is in $\mathcal{L}_n(m)$ and not in $\mathcal{L}_s(m)$. In other words, node B should be a next hop for this message and node A has not transmitted this message to node B before. After transmission, node A will add node B into $\mathcal{L}_s(m)$. So node A will not send duplicate messages to node B. However, node B may still receive duplicate messages from different nodes, due to network dynamics or message flooding. To address this problem, nodes exchange control information first to determine which messages should be sent[6]. Specifically, node A will first send an ADV message including information about messages it wants to transmit. Upon reception of the ADV message, node B replies with a REQ message which lists only messages it currently does not have. Then A will send the messages listed in the REQ message. Since meta data is much smaller in size than the actual message, the overhead of ADV and REQ messages is generally not significant.

5.2 Specific Operations

In the following, we present the specific operations of each multicast routing algorithm.

5.2.1 Static Tree-Based Routing (STBR)

In STBR, nodes construct a shortest path tree in the DTN graph from the source to the estimated intended receivers of a message starting at the message generation time. $\mathcal{L}_n$ of a message would include nodes that are the next hops in the tree. This can be computed using the modified Dijkstra's algorithm in [12]. As group membership changes, nodes update the shortest path tree and $\mathcal{L}_n$. Messages are then forwarded along the tree.

---

[6] Here we assume that the delay of the contact is small as compared to the duration of the contact. If contacts are short in duration, other strategies are needed, which is a topic for future work.

```
1. On arrival of message $m$
    Forward $m$ to local receivers if any;
    Insert $m$ in node storage;
    Initialize forwarding state $\mathcal{L}_n(m)$ and $\mathcal{L}_s(m)$;
2. On contact with node $B$
    For each message $m$ in storage
        IF $B \in \mathcal{L}_n(m)$ and $B \notin \mathcal{L}_s(m)$
            Send a copy of $m$ to node $B$;
            Add $B$ into $\mathcal{L}_s(m)$;
3. On join/leave request for group $G$
    Update $\mathcal{L}_n$ for messages destined for $G$;
```

Figure 6: Message forwarding.

In STBR, the route from the source to an intended receiver is static. Thus if a message misses a contact with a node in $\mathcal{L}_n$, the message needs to wait for the next opportunity to connect to this node, which may significantly increase the message delay. In addition, the use of static routes disallows nodes to utilize local or more accurate information to forward messages along better paths.

5.2.2 Dynamic Tree-Based Routing (DTBR)

DTBR addresses the above problems with STBR using the explicit addressing approach [2], i.e., messages include the endpoint IDs of the receivers as well as the group ID in the message header. This way, nodes can determine the next-hops of a message dynamically based on current available information, such as local queuing information or newly available contact information. Specifically, DTBR determines $\mathcal{L}_n$ by computing the shortest paths from the current node to endpoints embedded in the message. Nodes which are the next hops in these paths are added to $\mathcal{L}_n$. When a message is forwarded, each copy of the message contains only the IDs of endpoints to which it will be delivered.

5.2.3 Group-Based Routing (GBR)

In GBR, nodes construct a forwarding group for each message by computing a shortest path tree as in STBR and setting the forwarding group as the set of nodes in the tree including the receivers. Messages are then forwarded by flooding within the forwarding group. In other words, $\mathcal{L}_n$ consists of all nodes in the shortest path tree.

5.2.4 Broadcast-Based Routing (BBR)

In BBR, $\mathcal{L}_n$ always includes all nodes in the network. So messages are flooded throughout the network.

5.2.5 Unicast-Based Routing (UBR)

In UBR, when a multicast message is generated, the source node sends a unicast message, which encapsulates the original multicast message, to each of the estimated intended receivers. The source node also buffers the multicast message and sends out new unicast messages when being informed of new intended receivers. In this paper, we assume that unicast messages are forwarded using the shortest paths to the destinations. Unicast messages are removed from node storage after being transmitted to the next hop. Upon receiving a unicast message, the destination node will decapsulate the message and forward the original multicast message to the intended receiver according to the delivery constraints of the specified semantic model.

6. PERFORMANCE EVALUATION

In this section, we evaluate various multicast routing algorithms using *ns* simulations. We aim to compare these routing algorithms and understand how the availability of knowledge affects routing performance.

We simulate a specific type of DTNs, *sparse mobile networks* that consist of mobile nodes communicating via wireless radios. In these networks, nodes are sparsely distributed such that the networks experience frequent and long-duration partitions. We implement the four classes of routing algorithms and the contact and membership oracles in the *ns* simulator. Our simulations use the IEEE 802.11 MAC layer. The radio range and data rate are $250m$ and 2Mbps respectively.

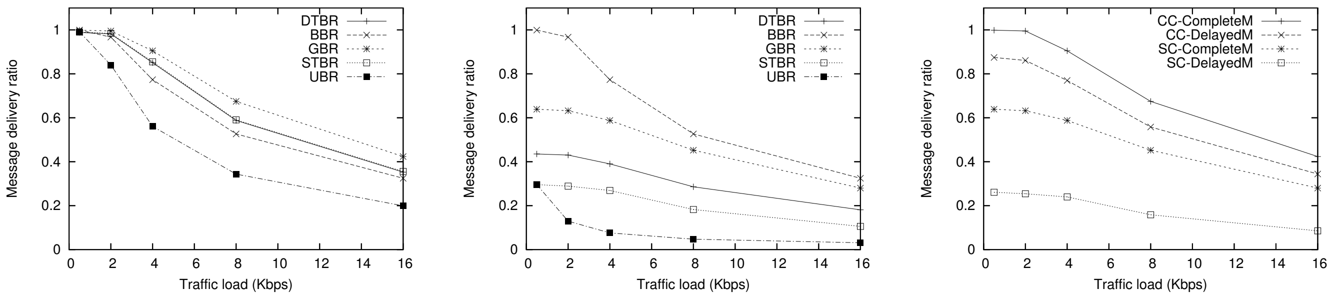
We use the following default settings unless specified otherwise. All simulations have 40 nodes on a $5000m \times 5000m$ area. Nodes move in the area according to the Random Way-Point (RWP) model [13] with a maximum speed 5m/s and a minimum speed 1m/s. The node storage capacity is 400 messages. In order to discover other nodes for communication, each node sends out beacon messages every 3 seconds. Each simulation lasts for 10000 seconds and each result is averaged over five runs with random seeds.

To understand how these routing algorithms perform, we consider only multicast traffic in the network. By default, there are 4 multicast sessions and each session consists of a single source which transmits messages to a multicast group. Each multicast group has 10 potential members which join and leave the group dynamically. Both the source and the potential group members are chosen randomly. Messages are generated at each source according to a Poisson process with mean inter-arrival time 4 seconds. Each message has 1000 bytes, thus the traffic rate of each source is 2kbps. After an endpoint joins (leaves) a group, it will leave (join) the group after a duration that is exponentially distributed with mean 200 seconds. Messages use the TD semantic model with membership interval $[t_0, t_0 + 100]$ and delivery interval $[t_0, t_0 + 3000]$ where t_0 is the message generation time.

Simulation Results: With extensive *ns* simulations, we have evaluated these routing algorithms under different traffic rates, numbers of sessions, session sizes, mobility patterns, node buffers and semantic models. Due to space limits, we present only the results under different traffic rates in this paper, which illustrate the representative performance. Please refer to [24] for more details.

In these simulations, the average message inter-arrival for all sources combined varies from 16, 4, 2, 1, to 0.5 seconds. Thus the total traffic load ranges from 0.5, 2, 4, 8, to 16 kbps. We first compare the performance between various algorithms. Fig. 7(a) and (b) show the delivery ratio when different oracles are used. We make the following observations. First, the delivery ratio decreases for all algorithms as the traffic load increases, which is as expected. Second, among the routing algorithms that utilize knowledge in computing routes, GBR achieves the best performance. This is because in GBR, messages may be forwarded to receivers via multiple paths, which is better in exploiting available contact opportunities. UBR, on the other hand, has the worst delivery ratio because a separate unicast message is sent to each receiver which significantly increases contention for node storage and transmission opportunities, and results in message drops. This result confirms the intuition that providing multicast service by sending multiple unicast messages is very inefficient in DTNs. The performance of both DTBR and STBR is between that of GBR and UBR. Since DTBR can adapt to network conditions, it performs slightly better than STBR. Third, GBR and BBR achieve the highest delivery ratio, depending on the level of available knowledge. GBR typically performs best when complete contact knowledge is available, while BBR achieves the best delivery ratio when the contact summary oracle is used. Both BBR and GBR utilize some form of flooding in message forwarding, which suggests that forwarding messages via multiple paths is a promising approach to achieve high delivery ratio in DTNs.

We now study how each routing algorithm performs under dif-



(a) Complete contact, Complete membership

(b) Contact summary, Complete membership

(c) Delivery ratio

Figure 7: Message delivery ratio under different message generation rates.

ferent amount of available knowledge. Fig. 7(c) shows the results for GBR, which are representative of other algorithms that utilize knowledge, i.e., UBR, STBR and DTBR. The labels "CC-xM"("SC-xM") in the figure represent scenarios where the complete contact (contact summary) oracle is used. We can see that the availability of up-to-date membership or exact contact knowledge has significant effect on routing performance. GBR performs poorly when such knowledge is not available. This suggests that a minimum amount of knowledge is required to achieve efficient routing for these approaches. In addition, the marginal improvement in performance for accurate contact information is more significant than that for up-to-date membership information.

We also evaluate the routing efficiency of various algorithms. Fig. 8(a) illustrates the results when the complete contact and complete membership oracles are used. We can see that BBR, which uses flooding to forward messages, has the lowest routing efficiency because it generates many redundant messages. Thus BBR is not suitable for mobile networks where nodes are equipped with limited power supplies. UBR is also inefficient in utilizing resources since it sends a separate copy of a multicast message to every receiver. STBR and DTBR achieve the best routing efficiency among all algorithms. The routing efficiency for GBR is slightly lower than that of the TBR algorithms. Fig. 8(b) depicts the routing efficiency when the contact summary and complete membership oracles are used. GBR achieves better efficiency than both TBR algorithms in this case. In addition, UBR achieves the highest routing efficiency, which, however, is obtained with a very low delivery ratio.

Fig. 8(c) shows the average delay for delivered messages when the complete contact and complete membership oracles are used. We can see that for all algorithms, the message delay decreases as the traffic load increases. This is because as the network becomes more congested, messages of the same age are more likely to be removed from node storage. Thus messages tend to reach only receivers that are on a shorter forwarding path, resulting in lower message delay. BBR achieves slightly lower delay than other algorithms because messages are flooded to all nodes and it is more likely that messages follow a shorter path to the receivers. The delay of GBR is slightly larger than that of STBR and DTBR.

7. RELATED WORK

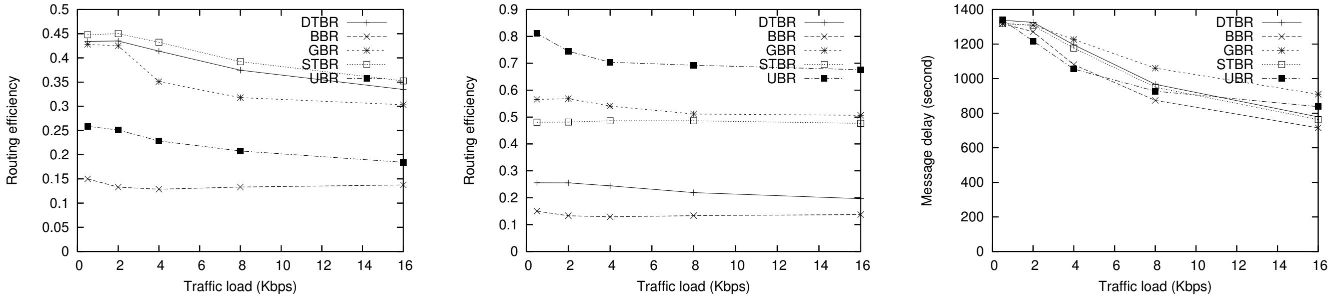
In this section, we review some related work on DTNs and multicasting in traditional networks. DTNs are a class of emerging networks that experience frequent and long-duration partitions, such as military ad hoc networks [1], deep space communication [4] and vehicular communication [22]. To achieve interoperability between various types of DTNs, Fall [9] proposes an architecture that is based on an asynchronous message forwarding paradigm. This architecture operates as an overlay above the transport layers to connect different DTNs.

Routing in frequent-disconnected networks has been studied relatively recently. In [12], Jain et al. study unicast routing in DTNs and develop several routing algorithms for scenarios where different levels of knowledge about network are available. The authors present a framework to evaluate these algorithms and find that efficient routing can be achieved using only limited amount of knowledge. There is also other work that focuses on sparse mobile networks and exploits node mobility to deliver data. For example, in Epidemic Routing [20], mobile nodes carry data and exchange data when they meet, essentially flooding data throughout the network. The Data Mules project [19] exploits mobile entities for data transportation to conserve energy in sensors. In the Message Ferrying project [23], special nodes called *message ferries* are used to provide communication services and controlled mobility is exploited to improve routing performance. Other work includes [10, 14, 21].

Multicasting has been studied extensively in the past, both in the Internet and in MANETs. Deering and Cheriton [7] first introduce the concept of IP multicasting. IP multicast assumes an open group model in which sources do not need to know the group membership or be group members to send data to a group. In addition, nodes can join or leave a multicast group at will. Various multicast protocols have been developed for the Internet, including DVMRP, MOSPF, PIM and CBT [3, 7, 8, 16]. These protocols construct a multicast tree to forward packets, using either a broad-and-prune (dense mode) or an explicit join (sparse mode) mechanism. In MANETs, node mobility introduces frequent topological changes which is different from the wired Internet. In addition, MANETs are resource-constrained in terms of bandwidth and energy supplies. MANET multicast protocols (e.g., FGMP, MAODV and ODMRP [6, 15, 18]) have been designed to address these issues, which normally use on-demand routing, localized repair of broken paths, and a mesh structure for packet forwarding.

In this paper, we study multicasting in DTNs. The semantic models we developed are based on the open group model used in IP multicasting but with additional temporal constraints to uniquely identify receivers of a message. Multicast routing in DTNs shares some commonalities with that in MANETs, e.g., dynamic topology and limited resources. On the other hand, DTNs differ from MANETs in the following aspects, namely frequent partitions and large message delivery delay. These factors affect not only data forwarding, but also the dissemination of control information. It would be interesting to study how MANET routing protocols or techniques can be adapted to DTNs, which is a topic of our future work.

In [11], Huang et al. propose *mobicast* for sensor networks in which applications can specify spatiotemporal constraints on a mobile delivery zone for a packet. In contrast, the multicast mod-



(a) Complete contact, Complete membership (b) Contact summary, Complete membership (c) Complete contact, Complete membership

Figure 8: Routing efficiency and delay under different message generation rates.

els proposed in this paper define the intended receivers of messages, which are time-invariant, by specifying temporal constraints on group membership, instead of geographic regions that change over time.

8. CONCLUSION

In this paper, we studied the problem of multicasting in DTNs. We developed three multicast semantic models that allow users to explicitly specify temporal constraints on group membership and message delivery. These semantic models unambiguously define the intended receivers of messages and have various applications in DTN environments. We then developed four classes of routing algorithms for DTNs with different routing strategies. With extensive *ns* simulations, we compared these multicast algorithms and studied how routing performance is affected by the availability of knowledge.

Based on our simulations, we obtained the following results. First, efficient routing for multicast can be constructed using only partial knowledge. In addition, the marginal improvement in performance for accurate contact information is generally more significant than that for up-to-date membership information. Second, GBR and BBR achieve the best delivery ratios depending on the amount of knowledge available. Both algorithms use some form of flooding, which suggests that forwarding messages along multiple paths is a promising approach for multicasting in DTNs. Third, UBR performs poorly in DTNs, confirming that multicast routing using multiple unicast messages is not efficient in DTNs.

In this paper, we studied the impact of available knowledge on routing performance using knowledge oracles which do not consider the overhead of disseminating such knowledge in the network. We are currently studying information dissemination in DTNs and how it might affect routing performance. We plan to develop multicast routing protocols for DTNs based on the semantic models and routing algorithms presented in this paper. In addition, we would like to extend our evaluation of the multicast algorithms in different networks, e.g., environments where node mobility is more predictable [12] or follows power-law distributions [5]. Furthermore, we are interested in extending our semantic models to incorporate spatial constraints as in geocast or mobicast.

9. REFERENCES

[1] DARPA Disruption Tolerant Networking Program: http://www.darpa.mil/ato/solicit/dtn.

[2] L. Aguilar. Datagram routing for internet multicasting. In *ACM SIGCOMM'84*, 1984.

[3] T. Ballardie, P. Francis, and J. Crowcroft. Core Based Trees (CBT): An architecture for scalable inter-domain multicast routing. In *ACM SIGCOMM*, Sept. 1993.

[4] S. Burleigh, A. Hooke, L. Torgerson, K. Fall, V. Cerf, B. Durst, K. Scott, and H. Weiss. Delay-tolerant networking – an approach to interplanetary internet. *IEEE Communications Magazine*, June 2003.

[5] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott. Pocket switched networks: Real-world mobility and its consequences for opportunistic forwarding. Technical Report UCAM-CL-TR-617, University of Cambridge, Computer Lab, February 2005.

[6] C.-C. Chiang, M. Gerla, and L. Zhang. Forwarding group multicast protocol (FGMP) for multihop, mobile wireless networks. *Cluster Computing*, 1(2):187–196, 1998.

[7] S. Deering and D. Cheriton. Multicast routing in datagram internetworks and extended LANs. *ACM Transactions on Computer Systems (TOCS)*, 1990.

[8] S. Deering, D. Estrin, D. Farinacci, V. Jacobson, C.-G. Liu, and L. Wei. An architecture for wide-area multicast routing. In *ACM SIGCOMM 1994*, 1994.

[9] K. Fall. A delay-tolerant network architecture for challenged internets. In *ACM Sigcomm 2003*, 2003.

[10] A. A. Hasson, R. Fletcher, and A. Pentland. DakNet: A road to universal broadband connectivity. *Wireless Internet UN ICT Conference Case Study*, 2003.

[11] Q. Huang, C. Lu, and G.-C. Roman. Spatiotemporal multicast in sensor networks. In *ACM SenSys 2003*, pages 205–217, 2003.

[12] S. Jain, K. Fall, and R. Patra. Routing in a delay tolerant network. In *ACM Sigcomm 2004*, Portland, OR, 2004.

[13] D. Johnson and D. Maltz. Dynamic source routing in ad-hoc wireless networks. In *ACM SIGCOMM*, August 1996.

[14] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. Peh, and D. Rubenstein. Energy-efficient computing for wildlife tracking: design tradeoffs and early experiences with ZebraNet. In *ASPLOS'02*, October 2002.

[15] S. Lee, M. Gerla, and C. Chiang. On-demand multicast routing protocol (ODMRP) for ad hoc networks. *Internet draft*, June 1999.

[16] J. Moy. Multicast extensions to OSPF. *RFC 1584*, Mar. 1994.

[17] Network simulator 2. http://www.isi.edu/nsnam/ns/.

[18] E. Royer and C. Perkins. Multicast operation of the ad-hoc on-demand distance vector routing protocol. In *ACM Mobicom*, Seattle, WA, Aug. 1999.

[19] R. Shah, S. Roy, S. Jain, and W. Brunette. Data MULEs: Modeling a three-tier architecture for sparse sensor networks. In *IEEE SNPA Workshop*, 2003.

[20] A. Vahdat and D. Becker. Epidemic routing for partially-connected ad hoc networks. *Technical report, Duke University*, 2000.

[21] R. Wang, S. Sobti, N. Garg, E. Ziskind, J. Lai, and A. Krishnamurthy. Turning the postal system into a generic digital communication mechanism. In *ACM SIGCOMM 2004*, August 2004.

[22] H. Wu, R. Fujimoto, and G. Riley. Analytical models for data dissemination in vehicle-to-vehicle networks. In *IEEE VTC 2004/Fall*.

[23] W. Zhao, M. Ammar, and E. Zegura. A message ferrying approach for data delivery in sparse mobile ad hoc networks. In *ACM MobiHoc 2004*, Tokyo Japan, 2004.

[24] W. Zhao, M. Ammar, and E. Zegura. Multicasting in delay tolerant networks: Semantic models and routing algorithms. Technical report, College of Computing, Georgia Institute of Technology, 2005.