An Eco-Friendly Routing Protocol for Delay Tolerant Networks

Tamer Abdelkader  
Department of Electrical and Computer Engineering  
University of Waterloo  
Waterloo, Ontario N2M 3G1  
Email: tammabde@uwaterloo.ca

Kshirasagar Naik  
Department of Electrical and Computer Engineering  
University of Waterloo  
Waterloo, Ontario N2M 3G1  
Email: knaik@swen.uwaterloo.ca

Amiya Nayak  
School of Electrical and Computer Engineering  
University of Ottawa  
Ottawa, Ontario K1N 6N5  
Email: anayak@site.uottawa.ca

Abstract—In sparse mobile networks, nodes are connected at discrete periods of time. This disconnection may last for long periods in suburban and rural areas. In addition, mobile nodes are energy and buffer sensitive, such as in mobile sensor networks. The limited power and storage resources, combined with the intermittent connection have created a challenging environment for inter-node networking. This type of networks is often referred to as Delay Tolerant networks (DTN). Routing protocols developed for DTN focused on minimizing the end-to-end delay as a means of maximizing number of delivered packets. Therefore, they tend to spread many copies of the same packet into the network, assuming the availability of sufficient storage space and power. A key factor to help maintain a clean environment, is the reduction of energy consumption which can be achieved by decreasing number of transmissions in the network. In this paper, we formulate a mathematical model for optimal routing in DTN to minimize number of transmissions. In addition, we study and analyze the DTN heuristic routing protocols. After that, we propose an eco-friendly routing protocol, EFR-DTN, that efficiently uses simple information provided from the network to deliver packets with higher delivery ratio and minimum energy consumption than the other protocols. Simulation results show the outperformance of the proposed protocol under different buffer capacities, traffic loads, packet TTL values, and number of nodes in the network.

I. INTRODUCTION

The widespread usage of communication devices, such as laptops, cell phones, and different types of sensors, and the need to connect these devices wherever they are, and however the surrounding environment is, created a challenging task for network engineers. Over the next few years, wireless services are expected to extend to remote rural areas. These areas, not only they lack the network infrastructure, but they even lack an electrical grid to operate such infrastructure. Currently, establishing a wireless network in remote areas relies on diesel generator-powered stations that are environmentally-unfriendly and incurs high maintenance costs. An eco-friendly and a cost effective solution is to rely on renewable energy sources such as wind and solar power. However, renewable energy sources are still in their preliminary phases and only limited power is expected to be drawn from them. Therefore, saving energy becomes a critical issue in designing and maintaining network devices and protocols.

Another factor which affects networking in suburban and rural areas is the sparse distribution of network devices. This results in the absence of complete end-to-end connections for long periods of time. Intermittently connected networks are referred to in literature as Delay or Disruption Tolerant Networks (DTN) [1]. One of the interesting applications of DTNs is providing Internet access to residential properties in remote villages [2]. However, the majority of its applications can be found in sensor networking, such as monitoring and tracking wildlife animals [3] and whales in oceans [4], and environmental monitoring [5]. DTNs can be applied in a variety of other fields ranging from healthcare to education to economic efficiency [6]. In addition, they have their strong presence in vehicular networking, such as providing Internet access to vehicles [7], and the virtual warning signs [8]. Moreover, they are used in space networking such as the interplanetary network [9] which was historically the first application of a DTN.

Traditional routing protocols developed for MANETs assumed the presence of continuous end-to-end connections between sources and destinations. This assumption is violated in the DTN, and there becomes a need to develop new protocols that can survive the new challenging conditions. To overcome the intermittent connectivity and provide reliable communication in DTN, routing protocols developed for DTN spread multiple copies of the same packet in the network so that one of them may succeed in reaching the destination. Nodes receiving the packets store them until they meet other nodes or meet the destinations. Routing protocols depend on the available network information to decide about which nodes to select and number of copies to spread. Simple DTN routing protocols blindly send data packets to the nodes they meet without having a selection criteria. They range from the full network flooding to the limited flooding. Other protocols restrict the forwarding of data packets to selected nodes that are predicted to be on the route to destination.

Our objective from this work is to study and analyze routing protocols developed for DTN, and to develop a new protocol which:

- Reduces number of transmissions in the network which decreases energy consumption and, therefore, maintains
a clean and eco-friendly environment.

- Achieves the highest possible delivery ratio of data packets among all the other protocols.

Our road map to achieve these objectives can be summarized as follows:

- Developing a mathematical model for optimal routing in DTN that assumes the availability of full knowledge. This is used as a performance benchmark to compare with.
- Investigating the heuristic routing protocols developed for the DTN, and conducting a performance comparison among the different types of routing protocols.
- Designing and implementing an eco-friendly routing protocol, EFR-DTN, that combines the advantages of guided and blind routing protocols to achieve minimum number of transmissions without sacrificing the delivery ratio.
- Comparing the proposed protocol with a full flooding protocol (Epidemic), a limited flooding protocol (Spray-And-Wait (SnW)), and a guided routing protocol (PROPHET) together with the optimal protocol in terms of delivery ratio, and number of transmissions. The comparison is conducted under different buffer capacities, the traffic loads, packet time-to-live (TTL) values, and number of nodes in the network.
- Simulation results show that the proposed protocol achieves lower number of transmissions with higher delivery ratio compared to other heuristic protocols.

The rest of the paper is organized as follows. Section II gives a brief overview of DTN routing protocols. Optimal routing formulation is explained in Section III. The proposed protocol is presented in section IV. The performance comparisons and simulation results are discussed in Section V. Finally, conclusions and future work are drawn in section VI.

II. DTN ROUTING PROTOCOLS

Routing protocols are classified, according to the amount and type of information used to take the routing decision [10], into three classes: No-Knowledge, Partial-Knowledge and Full-Knowledge. In the No-Knowledge level, routing protocols choose the next node in a route blindly. Therefore, it is required to spread many copies of the same packet to increase the chance of one of them reaching the destination. Further classification inside this class is by the replication method [11], which ranges from a single copy to full flooding. Partial-Knowledge protocols use some network information, such as nodes meeting times, locations and mobility patterns, to guide the packets to their destinations. This information can be collected using additional infrastructure, such as rovers or basestations [12], or without infrastructure by exchanging information during nodes contacts. The third level of knowledge is the full knowledge, which is a theoretical case used as a reference to compare with. If we have full knowledge, then we can find optimal solutions to the routing problem.

A. Routing Protocols with No Knowledge about the Network (Blind Routing)

These are the simplest protocols in terms of communication overhead and processing power consumed to take a routing decision. There is no data exchange between contacting nodes, no infrastructure used and little processing required for the routing decision. Routing protocols in this category range from the single-copy to the full flooding [13].

1) Epidemic Routing: The simplest of the blind protocols is the full flooding technique, or so called Epidemic Routing [13], where nodes broadcast all its data packets to all the nodes they meet. Broadcasting stops when the packet expires or is deleted. Assuming that the contact duration between two nodes is long enough to transfer all the non-common packets, then the two nodes will have the same packet list after their contact. This protocol proves to provide the highest delivery ratio, if the buffers did not overflow. However, since buffers are limited, the protocol performance drops significantly with the high traffic rates. In addition, it guarantees the lowest end-to-end delay, because each packet is routed on all possible paths from the source including the shortest path. The main drawback of Epidemic routing is its huge consumption of data buffers and the tremendous number of transmissions that occurs in the network which translates into huge energy consumption.

2) Spray and Wait Routing: Energy consumption is mostly incurred in the communication process (transmission and reception). To save energy, it is required to decrease the number of transmissions and receptions. Motivated by this concept, the authors in [14] proposed the Spray-and-Wait (SnW) routing protocol. The idea of SnW is to limit the number of packet copies in the network. A packet, transferred from a node to another, is associated with the number of copies allowed for the next node to spread. This number is decreased by the number of transmissions of this packet at every node. When the allowed number of transmissions reaches one, the carrying node stop transmitting the packet until it either meet the destination or the packet is deleted due to buffer overflow or lifetime expiry. A binary version of SnW (BSnW) permits each node to use half the number of transfers allowed for the packet and the other half is left for the receiving node.

B. Routing Protocols with Partial Knowledge about the Network (Partially-Guided Routing)

Following the traditional routing protocols developed for the wired and wireless networks, this category protocols redesign the old protocols to suit the new environment of intermittent connection. They use information about the network to assign weights to links or nodes to help better choose the next nodes in the path. The network information include the contacts times and durations, the buffers availability, number of packet copies, location and mobility pattern of nodes, and many others. This information can be collected using special infrastructure [4] or by exchanging data between mobile nodes [15]–[17]. Exchanging data between mobile nodes is the low-cost alternative for having special infrastructure.
1) PROPHET Routing Protocol: The Probabilistic Routing Protocol using History of Encounters and Transitivity (PROPHET) is proposed in [15]. The protocol estimates a node metric, named delivery predictability, at each node for each other node, which is the predicted probability to deliver a packet to a certain node. When two nodes meet, they strengthen their delivery predictability towards each other. Then each node sends its delivery predictability list to the other node to update its delivery predictability towards the other nodes. PROPHET provides a partial guiding towards the destination by tracing the contacts between nodes and assigning weights to these contacts whether they were direct contacts or through intermediate nodes. Therefore, PROPHET is expected to outperform blind routing protocols in delivery ratio. On the other side, it is expected that the average packet delay may increase due to waiting for a better next node in the path.

III. OPTIMAL ROUTING FOR DTN

Our objective is to find a route for each message, if it exists, which minimizes the number of hops for all the routes. Other objectives have been investigated in [18]. A route is an ordered set of contacts from source to destination. Each contact has four attributes: Sender, Receiver, Time, and Duration, where

- Sender: The node which sends the messages.
- Receiver: The node which receives the messages.
- Time: The time at which the two nodes meet.
- Duration: The period length in which the two contacting nodes are able to transfer messages.

Each node buffer has two attributes: Capacity and size, where

- Capacity: The maximum number of messages the buffer can carry.
- Size: The actual number of messages at the time of optimization.

The problem is divided into two subproblems:

- Finding all the possible paths for a message. The problem is solved for each message individually. The output is a set of paths for each message.
- Finding the optimal path for each message. This step requires considering all the paths for all the messages in the optimization problem.

A. Finding all the possible paths for a message

The problem is to find all the possible paths for each message. The output is a set of paths \( P_m \), for each message \( m \in M \). The symbols used in the formulation are described in Table I. There is no objective function other than satisfying the following set of constraints:

- Flow Conservation: Every node included in the selected path, except for the source and destination, should be associated with two contacts:
  - A contact in which the message is received from another node.
  - A contact in which the message is sent to the next node in the route.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( N )</td>
<td>Set of nodes in the network</td>
</tr>
<tr>
<td>( M )</td>
<td>Set of messages in the network</td>
</tr>
<tr>
<td>( x_c )</td>
<td>A binary variable indicating the message is transferred during contact ( c ), 1=yes, 0&gt;No</td>
</tr>
<tr>
<td>( C )</td>
<td>Set of contact objects</td>
</tr>
<tr>
<td>( t_c )</td>
<td>The starting time of contact ( c )</td>
</tr>
<tr>
<td>( L_m )</td>
<td>The lifetime (TTL) of message ( m )</td>
</tr>
</tbody>
</table>

\[
\sum_{c_1 \in C} x_{c_1} - \sum_{c_2 \in C} x_{c_2} = 0, \quad \forall n \in N, \text{if } n = c_1.r = c_2.s \quad (1)
\]

- Source and Destination: The source node is associated with only one contact in which the message is sent by that node. The destination node is associated with only one contact in which the message is received by that node.

\[
\sum_{c \in C} x_c = 1, \text{if } c.s \text{ is the source},
\sum_{c \in C} x_c = 1, \text{if } c.r \text{ is the destination} \quad (2)
\]

- Contacts order: Contacts in the message route should have increasing order of their starting times.

\[
t_{c_1}x_{c_1} + (H - t_{c_2})x_{c_2} < H,
\text{if } c_1.r = c_2.s, \forall c_1, c_2 \in C, \quad H > t_{c_1}, H > t_{c_2} \quad (3)
\]

where \( H \) is any number greater than the starting times of both contacts.

- Message Lifetime: The starting time of the last contact in the message route should be less than the message lifetime (Time-To-Live or TTL).

\[
t_{c}x_{c} < L_m, \quad \forall c \in C \quad (4)
\]

The output of this problem, the set of paths \( P_m \), is fed as an input to the second problem which is to find the optimal path for each message explained in Section III-B.
**Table II**

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_m$</td>
<td>Set of valid paths for message $m$</td>
</tr>
<tr>
<td>$x_p$</td>
<td>A variable indicating if path $p$, $p \in P_m$, is selected for message $m$, $1$=yes, $0$=No</td>
</tr>
<tr>
<td>$B_i$</td>
<td>The buffer capacity of node $i$</td>
</tr>
<tr>
<td>$b_i$</td>
<td>The buffer size at the time of optimization</td>
</tr>
<tr>
<td>$D_c$</td>
<td>The time duration of contact $c$</td>
</tr>
<tr>
<td>$d_m$</td>
<td>The transmission time for message $m$</td>
</tr>
</tbody>
</table>

### B. Finding the optimal path for each message

The input to this problem is the set of paths, $P_m$, for each message $m$. The output is one path for each message. The new symbols used in this formulation are defined in Table II.

The problem should satisfy the following set of constraints:

- **Buffer:** The messages transferred during each contact should not exceed the receiver buffer capacity.

\[
\sum_{m \in M} \sum_{p \in P_m} \sum_{c \subseteq C \cap p_1} x_{p1} - \sum_{m \in M} \sum_{p \in P_m} \sum_{c \subseteq C \cap p_2} x_{p2} \leq B_{c,r} - b_{c,r,0} \\
\forall c \in C, if c.r = p1.c.r = p2.c.s, \\
t_{p1,c}, t_{p2,c} < t_c
\]

- **Contact duration:** The sum of all message transfer times during a contact should be less than the duration of that contact.

\[
\sum_{m \in M} \sum_{p \in P_m} d_m x_p \leq D_c, \forall c \in C
\]

- **Paths/Message:** For each message, there should be only one path selected.

\[
\sum_{p \in P_m} x_p = 1, \forall m \in M
\]

The objective is to set the minimum number of hops for all the routes, which is equal to minimizing the number of contacts.

\[
\begin{align*}
\text{Minimize} & & \sum_{m \in M} \sum_{p \in P_m} \sum_{c \subseteq C \cap p_1} x_{p1} - \sum_{m \in M} \sum_{p \in P_m} \sum_{c \subseteq C \cap p_2} x_{p2} \leq B_{c,r} - b_{c,r,0} \\
& & \forall c \in C, if c.r = p1.c.r = p2.c.s, \\
& & t_{p1,c}, t_{p2,c} < t_c \quad (5)
\end{align*}
\]

- **Contact duration:** The sum of all message transfer times during a contact should be less than the duration of that contact.

\[
\sum_{m \in M} \sum_{p \in P_m} d_m x_p \leq D_c, \forall c \in C \\
\text{(6)}
\]

- **Paths/Message:** For each message, there should be only one path selected.

\[
\sum_{p \in P_m} x_p = 1, \forall m \in M \\
\text{(7)}
\]

The objective is to set the minimum number of hops for all the routes, which is equal to minimizing the number of contacts.

\[
\begin{align*}
\text{Minimize subject to} & & \sum_{m \in M} \sum_{p \in P_m} \sum_{c \subseteq C \cap p_1} x_{p1} - \sum_{m \in M} \sum_{p \in P_m} \sum_{c \subseteq C \cap p_2} x_{p2} \leq B_{c,r} - b_{c,r,0} \\
& & \forall c \in C, if c.r = p1.c.r = p2.c.s, \\
& & t_{p1,c}, t_{p2,c} < t_c \quad (5)
\end{align*}
\]

The objective is to set the minimum number of hops for all the routes, which is equal to minimizing the number of contacts.

\[
\begin{align*}
\text{Minimize subject to} & & \sum_{m \in M} \sum_{p \in P_m} \sum_{c \subseteq C \cap p_1} x_{p1} - \sum_{m \in M} \sum_{p \in P_m} \sum_{c \subseteq C \cap p_2} x_{p2} \leq B_{c,r} - b_{c,r,0} \\
& & \forall c \in C, if c.r = p1.c.r = p2.c.s, \\
& & t_{p1,c}, t_{p2,c} < t_c \quad (5)
\end{align*}
\]

- **Contact duration:** The sum of all message transfer times during a contact should be less than the duration of that contact.

\[
\sum_{m \in M} \sum_{p \in P_m} d_m x_p \leq D_c, \forall c \in C \\
\text{(6)}
\]

- **Paths/Message:** For each message, there should be only one path selected.

\[
\sum_{p \in P_m} x_p = 1, \forall m \in M \\
\text{(7)}
\]

The objective is to set the minimum number of hops for all the routes, which is equal to minimizing the number of contacts.

\[
\begin{align*}
\text{Minimize subject to} & & \sum_{m \in M} \sum_{p \in P_m} \sum_{c \subseteq C \cap p_1} x_{p1} - \sum_{m \in M} \sum_{p \in P_m} \sum_{c \subseteq C \cap p_2} x_{p2} \leq B_{c,r} - b_{c,r,0} \\
& & \forall c \in C, if c.r = p1.c.r = p2.c.s, \\
& & t_{p1,c}, t_{p2,c} < t_c \quad (5)
\end{align*}
\]

- **Contact duration:** The sum of all message transfer times during a contact should be less than the duration of that contact.

\[
\sum_{m \in M} \sum_{p \in P_m} d_m x_p \leq D_c, \forall c \in C \\
\text{(6)}
\]

- **Paths/Message:** For each message, there should be only one path selected.

\[
\sum_{p \in P_m} x_p = 1, \forall m \in M \\
\text{(7)}
\]

**IV. ECO-FRIENDLY ROUTING PROTOCOL FOR DELAY TOLERANT NETWORKS (EFR-DTN)**

From the analysis of DTN routing protocols, we can come out with the following:

- Full flooding protocols has the highest delivery ratio and the lowest average packet delay if provided large buffers. However, they incur the highest number of transmissions leading to huge consumption of energy and buffer space.
- Limited flooding has the lowest number of transmissions and reasonable average packet delay. However, the delivery ratio drops down as the flooding limit decreases.
- Guided routing protocols provide better delivery ratio than limited flooding protocols, but with higher number of transmissions.

According to this analysis, an efficient routing protocol should exploit the available information in the network to guide the packets to the shortest paths towards their destinations. In addition, it should limit the number of copies for each packet in the network to decrease the number of transmissions and, therefore, help in saving the nodes energy.

Our proposed protocol can be explained as follows:

- Each packet generated is assigned a unique ID that is associated with it and all its copies till they are deleted or they reach the destination. The list of all the packets IDs in a node’s buffer is called the summary vector.
- When two nodes meet, they exchange their summary vectors. All data packets that are stored in one node and not in the other are ordered on a first come first serve (FCFS) basis to be transferred to the other node. Packet transfer then starts until the contact duration ends.
- We used the PROPHET node selection criteria, so that a packet is transferred only if its delivery predictability at the other node is higher than that at its current node. The delivery predictability is updated as follows:

  - **Aging:** The two meeting nodes decrease the delivery predictability to all nodes in the network, according the the time passed since last updating, as follows:

\[
P_{(a,b)} = P_{(a,b),old} \gamma^k \quad (8)
\]

  - **Direct Delivery:** The two meeting nodes update the delivery predictability to each other as follows:

\[
P_{(a,b)} = P_{(a,b),old} + (1 - P_{(a,b),old}) P_{init} \quad (9)
\]

  - **Transitive Delivery:** The two meeting nodes update the delivery predictability to the other nodes, seen by the corresponding, node as follows:

\[
P_{(a,b)} = P_{(a,b),old} + (1 - P_{(a,b),old}) P_{(a,c)} P_{(c,b)} \beta \quad (10)
\]

where $P_{(a,b)}$ is the delivery predictability at node $a$ for destination $b$, $P_{init} \in [0,1]$ is an initialization constant, $\gamma \in [0,1]$ is the aging constant, $k$ is the number of time units that have elapsed since the last time the metric was updated, and $\beta \in [0,1]$ is the transitivity constant which reflects the impact of transitivity on the delivery predictability.
The flooding of data packets is limited using the Binary SPRAY-and-WAIT (BSnW) mechanism, so that a node keeps half the number of copies and assigns the other half to the receiving node.

When the expiry time of a data packet approaches, the carrying node switches to partial flooding of the network with limited copies of the packet. This gives a last chance to the dying packet to catch the destination before it is dropped.

If a packet is transferred to another node, it may be dropped from the sender node. The higher the delivery predictability of the sender node, the less is the probability to drop the packet from its buffer. This simple buffer management scheme helps provide space for new packets coming to the node without decreasing its delivery probability.

The pseudo-code for the proposed protocol is presented in procedure 1.

### Procedure 1 OnContact

**Input:** node: n1,n2; ContactDuration

1. Exchange Summary Vector(n1,n2)
2. UpdateDeliveryPredictability()
3. if ContactDuration > 0 then
   4. if pkt=GetPacket(n1) then
      5. if NotExpired(pkt) and NotReceivedBefore(pkt,n2) then
         6. if IsDestination(pkt) and NotReceivedBefore(pkt,n2) then
            7. ConsumePacket(pkt,n2)
         8. else if ExpiryTime < ExpiryThreshold then
            9. NrCopies=GetLastChance(pkt,n1)
            10. if NrCopies > 1 then
                11. StorePacket(pkt,n2)
                12. SetLastChance(pkt,n1,NrCopies − 1)
                13. SetLastChance(pkt,n2,1)
            14. end if
            15. else
               16. DPn1=DeliveryPredictability(pkt,n1)
               17. DPn2=DeliveryPredictability(pkt,n2)
               18. NrCopies=GetNrOfCopies(pkt,n1)
               19. if DPn2 > DPn1 and NrCopies > 1 then
                  20. StorePacket(pkt,n2)
                  21. SetNrOfCopies(pkt,n1,NrCopies/2)
                  22. SetNrOfCopies(pkt,n2,NrCopies/2)
                  23. U = UniformRandomNumber
                  24. if U > DPn1 then
                     25. DeletePacket(pkt,n1)
               26. end if
               27. end if
               28. end if
         29. ContactDuration = ContactDuration − 1
      30. end if
   31. end if
32. DeletePacket(pkt,n1)
33. end if

V. PERFORMANCE COMPARISON AND SIMULATION RESULTS

We consider a sparse mobile network where nodes are connected to each other at discrete time intervals via wireless links. Nodes communicate when they get into the communication range of each other. In such events, they are said to be in contact. The inter-contact time, that is the interval between two contacts of the same pair of nodes is modeled using the power law distribution [19]. The cumulative distribution function (CDF) of the power law distribution is as follows

\[ P(X \geq x) = \left( \frac{x}{x_{\text{min}}} \right)^{-\theta} \]  

where \( x \) is the inter-contact time, \( x_{\text{min}} \) is the minimum inter-contact time and \( \theta \) is the parameter that characterizes the power law, \( \theta > 0 \). In our experiments, we use \( \theta = 0.9 \) as mentioned in [19]. To avoid partitioning, each node is made to contact with at least one node. The maximum number of nodes to contact with is drawn from a uniform distribution up to one fifth of the total number of nodes in the network. Packets are created at every node using Poisson distribution, and assigned a random destination. When generated, each packet is associated with a unique identifier, time of creation and a time to live (TTL). Table III shows the values used for the network parameters.

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALL</td>
<td>Simulation time</td>
<td>12 hours</td>
</tr>
<tr>
<td></td>
<td>Minimum packet arrival</td>
<td>1 minute</td>
</tr>
<tr>
<td></td>
<td>Minimum inter-contact</td>
<td>4 hours</td>
</tr>
<tr>
<td>PROPHET, EFR-DTN</td>
<td>Initialization constant, ( P_{\text{init}} )</td>
<td>0.75</td>
</tr>
<tr>
<td>EFR-DTN</td>
<td>Transitivity constant, ( B )</td>
<td>0.25</td>
</tr>
<tr>
<td>BSnW, EFR-DTN</td>
<td>Aging constant, ( \gamma )</td>
<td>0.98</td>
</tr>
<tr>
<td>EFR-DTN</td>
<td>Initial number of copies</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>Last-Chance number of copies</td>
<td>10</td>
</tr>
</tbody>
</table>

Each node records all the packets it receives to avoid receiving two copies of the same packet. We study the impact of varying the buffer capacity, the traffic load, packet time-to-live (TTL) values, and number of nodes in the network on the performance of the routing protocols: Epidemic, PROPHET, Binary SPRAY-AND-WAIT (BSnW) and our proposed protocol (EFR-DTN). The performance measures considered are:

- Delivery ratio: the number of delivered packets to the number of packets generated.
- Number of transmissions: the total number of packets divided by the number of delivered packets. This metric is used an indicator for the energy consumption. The less the number of transmissions, the less is the energy consumption.

### A. Impact of Varying the Buffer Capacity (B)

As shown in Figure 1(a), the delivery ratio for the optimal routing is higher than that of the heuristic protocols. This
can be justified by the ability of the optimal protocol to find routes which optimally distribute packets among the network and avoid congestion. In addition, the optimal routing spreads only a single copy of the packet which helps in avoiding the redundant filling of the packet buffers. Regarding the heuristic protocols, except for Epidemic, they almost have the same or near values for the delivery ratio, Figure 1(b). Because of the limited buffer capacities, Epidemic cannot compete with the other protocols. Epidemic performance should be better at large buffer capacities and less traffic loads.

Figure 2 shows the impact of varying the buffer capacity on the total number of transmissions per one delivered packet. The number of transmissions include those for the packets delivered, dropped and those that are still in the buffers. Because the optimal protocol minimizes the number of contacts (hops), and spreads a single copy of each packet, it significantly reduces the number of transmissions. Among the heuristic protocols, EFR-DTN and PROPHET extremely outperforms the others because of their intelligent routing. Our proposed protocol, EFR-DTN, proves its lowest energy consumption among all the heuristic protocols.

B. Impact of Varying the Traffic Load (TL)

As shown in Figure 3(a), injecting more packets into the network causes the dropping of many of stored packets, and therefore, decreases the delivery ratio for all the heuristic protocols. The optimal protocol is capable of handling the difficult situation even at higher traffic loads. Our proposed protocol, EFR-DTN, have the same behavior as the other protocols, while maintaining its outperformance in the delivery ratio, Figure 3(b).

Number of transmissions, Figure 4, are almost constant with increasing traffic loads for all the protocols. This behavior can be explained by the the continuous fullness of the buffers because of the high traffic loads and the low buffer capacities. The buffers are full most of the time and the number of transmissions depend on the packets stored in the queues, not those injected into the network. Therefore, it is almost constant with all the introduced traffic rates. It is worth noting that EFR-DTN maintains the lowest number of transmissions with all the traffic loads.
C. Impact of Varying the Packets Time-To-Live (TTL)

It is observed from Figure 5(a) that the delivery ratio increases with increasing the packets TTL. This is because increasing the packet lifetime gives it a better chance to reach destination, instead of being dropped earlier. This increase in delivery ratio is small in the heuristic protocols because the effect of dropping packets due to buffer overflow is more significant than giving longer lifetime for the packets, Figure 5(b).

As explained in Section V-B, the number of transmissions are almost constant for all the protocols, Figure 6. Except for Epidemic, the other heuristic protocols have near values. Our proposed protocol, EFR-DTN, maintains the lowest energy consumption among all the heuristic protocols.

D. Impact of Varying the Number of Nodes (N)

Increasing the number of nodes increases their contact frequency and the total network buffer space. This leads to increasing both the delivery ratio, Figure 7, and the number of transmissions, Figure 8. EFR-DTN and PROPHET have the highest delivery ratio among heuristic protocols, and the lowest number of transmissions (energy consumption). Epidemic performs poorly with increasing the number of nodes.

VI. Conclusion

Delay Tolerant Networks (DTN), lack end-to-end connections between data sources and destinations. This require the intermediate nodes to store data packets for long periods of time which violates one of the basic assumptions of traditional routing protocols and triggers the development of new ones. DTN routing protocols developed before focused on decreasing the end-to-end delay and maintaining high delivery ratio of packets. These objectives required increasing the packet copies in the network to guarantee their delivery with minimum delay. However, increasing transmissions consumes more energy which means more pollutants to be produced and, therefore, becomes environmentally unfriendly. In this paper, we formulated an optimization problem that finds the optimal routes in DTN with minimum number of hops. Reducing the number of hops leads to minimizing the number of transmissions and, therefore, the energy consumption. In addition, we studied and analyzed the well-known DTN routing protocols. From the analysis, it was found that by limiting the number of packet copies, while efficiently guiding them to their destinations, we can achieve the minimum energy consumption without sacrificing the delivery ratio. Taking that into consideration, we designed and implemented an eco-friendly routing protocol, EFR-DTN, that combines the advantages of guided and blind routing protocols to achieve an acceptable delivery ratio of packets while reducing the energy consumption. Simulation results show that the proposed protocol consumes lower energy compared to the routing protocols while maintaining higher delivery ratio.
Fig. 7. Impact of number of nodes, N, on the delivery ratio

Fig. 8. Impact of number of nodes, N, on the number of transmissions

REFERENCES


