Replication Probability-based Routing Scheme for Opportunistic Networks

Salem Sati
Technology of Social Networks
University of Düsseldorf, Germany
sati@cs.uni-duesseldorf.de

Andre Ippisch
Technology of Social Networks
University of Düsseldorf, Germany
ippisch@cs.uni-duesseldorf.de

Kalman Graffi
Technology of Social Networks
University of Düsseldorf, Germany
graffi@cs.uni-duesseldorf.de

***Abstract*—Opportunistic networks offer a delay-tolerant end-to-end message delivery in networks with intermittently connected nodes and communication islands. In this paper, we propose a probabilistic forwarding scheme for opportunistic networks, called Replication Probability-based Routing Scheme (RPRS), based on controlled replication of messages aiming to have a high delivery ratio while drastically reducing the message overhead. It uses from each single message the replication count and hop count to calculate the desired replication probability, which is prioritizes the message for replication. Also, it has its own drop policy whose utility function is a function of replication count, hop count, and the buffer time of a message, being an estimate of the end-to-end delay. Through this, RPRS allows us to decide when it is desirable to further spread a message and thus reach a high delivery ratio without congesting the network with unnecessary message copies. In simulation results, we analyze the performance of RPRS and compare it with well-known routing protocols such as Epidemic routing and Spray & Wait, with different scheduling and drop policies based on the same variables. Conducted scenarios show that our scheme has better performance regarding delivery ratio, delay, buffer delay and overhead.**

***Index Terms*—Opportunistic Networks, Epidemic Replication Routing Implementation, Controlled Probability**

I. INTRODUCTION

Opportunistic Networks (OppNets) are a subclass of delay tolerant networks, which both aim at supporting message delivery while facing intermittent connectivity and limited resources in terms of storage, bandwidth and power. Due to the intermittent connectivity, a direct path from source to destination is typically not given in an instance of time. Messages are delivered based on hop-by-hop routing via a store-carry-forward fashion. Over time, thus delay-tolerant, the messages reach their destination. In OppNets, source or relay nodes store messages in their buffer for a long time until they encounter the message destination or a suitable node to copy the message to. OppNet routing is mainly classified into two main categories, namely *flooding-based* routing and *utility-based* routing. One of the utility-based routing protocols is PRoPHET [1] which forwards the messages based on a delivery predictability, deciding whether a opportunistic contact is better suitable to carry on a copy of a message.

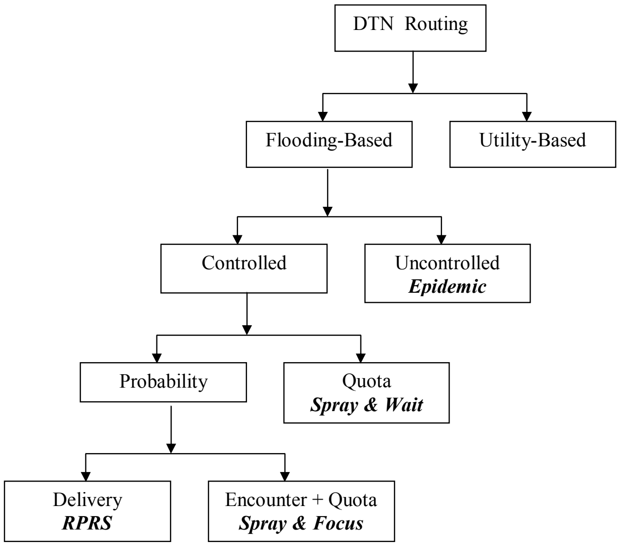


Figure 1. Classification of Opportunistic Networking Protocols

As shown in the Figure 1, the flooding-based routing protocols do not differentiate and replicate the message to every encountered node. Flooding-based protocols are further divided into two main types which are *Controlled* and *Uncontrolled replication*. One of the simplest greedy replication protocols of flooding-based routing is the Epidemic [2] routing protocol, which uses *uncontrolled replication*. The concept of Epidemic is that the node replicates the message to every encountered node. This is efficient when the buffer space of nodes is unlimited, however, if not, the replication of unlimited message copies leads to high overhead. To solve the problem of message overhead and the node's resources consumption, several controlled replication protocols have been proposed, such as Spray & Wait [3] which is considered a quota-based routing protocol. The Binary Spray & Wait marginally solves the problem of the overhead but is still suffering from the long time the messages reside in the buffer inactively. In addition to the quota-based approach, the Spray & Focus [4] protocol considers the *Inter-Contact time* of the last encounter event as probability of selecting a suitable relay. In both protocols of Spray & Wait and Spray & Focus, the total number of message copies (quota) present in a network is limited by a certain number of hops.

In this paper, we propose a replication controlled forwarding scheme for Opportunistic Networks, termed *Replication Probability-based Routing Scheme* (*RPRS*). This proposed forwarding scheme integrates an replication probability, allowing to keep an optimal replication count for a message in the network, which maximizes the message delivery ratio, but avoid congesting the network. The probability function uses the message's hop-count, the node's local replication counter, both characterizing the message's overhead in the network, as well as the buffer time as storage and transmutation costs for the integrated drop policy. The replication and dropping criteria of *RPRS* are calculated based on local message information as indication of the minimum single message delivery probability. The idea of *RPRS* relies on the assumption that the best delivery ratio is achieved when the rate of forwarding is adapted to the dropping rate, thus the buffer stay optimally filled. The forwarding rate is be adapted through a replication probability to control the number of messages induced and spread in the network, which may then congest the buffers and be dropped. Therefore, *RPRS* considers the message's hop count and replication count for both forwarding and dropping as invert functions, as we suggest that the forwarding and dropping are complement rates.

Our proposed forwarding scheme *RPRS* differs from the existing controlled quota and probability routing protocols in four important points.

1) *RPRS* is a heuristic forwarding scheme, which considers the local information of the message to estimate the network traffic and nodes resource consumption.
2) *RPRS* calculates the replication criteria based on the message delivery probability. The function is dynamic as it considers hop count and replication count as power consumption metric. While Spray & Wait make them equal values and we agree that this is good to minimize the overhead, but at the same time it leads to high delay as the OppNet environment is highly dynamic.
3) *RPRS* considers the buffer time of the messages as delay component in addition to the overhead variables, which is missing in Spray & Wait and Spray & Focus.
4) *RPRS* has no administrative (quota) values such as Spray & Wait and Spray & Focus.

In conclusion, *RPRS* is a resource focused routing scheme from the concept of its forward and drops decisions, where those decisions impact on each other as relation between the replication and hop count of the message. Finally, the rest of this paper will be organized as follow. In Section II, we give an overview on existing work and in Section III we present the *RPRS* Protocol. In Section IV, we present the comparison of *RPRS* with Epidemic and Spray & Wait protocols.

II. RELATED WORK

Wireless multihop networks have been researched for the past decade, mainly in form of ad hoc networks [5] and since recent years also in form of opportunistic networks. While the first two assume a connected graph, routing in Opportunistic Networks is based on the Store-Carry-Forward scheme due to the regular partitions of the network. Therefore most of routing research efforts focus on the two issues of buffer management and routing decisions criteria. The buffer management is considered to have the main impact on the OppNet routing performance, as typically only a single or even non neighboring node is available. Thus the ordering of scheduled buffered messages is related to the routing decision, i.e. which message to send once there is a contact opportunity. On the another hand, the dropping buffer decision complements with the routing forward decision when the buffer is full. It is relevant which message to drop if the buffer is full and a new message is added. Clearly, the dropping rate of the messages which are removed from the node buffer should be proportional with the number of forwarded messages, i.e. to maintain a high number of messages in the buffer. Therefore, unlimited replication does not increase the delivery ratio and minimize the delay as a positive value, as the buffers are unnecessarily filled. Due to the limitation of the networking opportunity time as well as the limitation of the buffer capacity and node power of the physical nodes, it is more reasonable to control the message replication and only to replicate if the chances for delivery are increasing. The routing algorithm should consider the overhead as one main issue to minimize in mobile OppNets. Several publications try to find a dynamic number of replication to make a balancing between the delay and the overhead while keeping the desired delivery ratio. Optimal Probabilistic Forwarding [6] aims to maximize the delivery ratio of each message based on its hop count and message life. Their probabilistic forwarding metric is derived by modeling each forwarding task as an optimal stopping rule problem. Jia Xu provides in his paper [7] an Optimal Joint Expected Delay Forwarding (OJEDF) protocol which minimizes the expected delay based on the number of forwarding times per message. Their paper proposes a comprehensive forwarding metric called Joint Expected Delay (JED) which is calculated based on remaining hop-count (or ticket) and message residual lifetime. The aim of this approach is to achieve a near-optimal replication of the message which both provides a maximum possible message delivery ratio while keeping the overhead for this purpose as small as possible. The authors of the paper suggest the RAPID [8] as OppNet routing protocol that can optimize a routing metric such as the delivery delay or the ratio of messages that delivered to a deadline. The key of RAPID is a resource allocation protocol that calculates the routing metric per-message utilities which determine how many messages should be replicated in the system. Lo and Liou propose in [9] a quota-based routing protocol which has the feature of limiting the number of message copies and control network traffic in OppNets. They propose an enhanced mechanism to dynamically adjust quota values, the dynamic-quota value observed by local buffer occupancy. For our approach, we build on our systematic evaluation of scheduling and drop policies for OppNets which we presented in [10], where we systematically analyzed the impact of a message's TTL, arrival time, hop count, replication count and size when considered in the message scheduling and message drop policies. Furthermore, we only use information that is gathered locally at the nodes. This is necessary due to that nodes do not have a global view on the network.

Our approach, presented next, aims to maintain a high delivery ratio while drastically reducing the overhead. Here we focus on controlled replication control through an optimized message scheduling and message drop policy.

III. THE *RPRS* FORWARDING SCHEME

In this section, we describe our proposed forwarding scheme, termed *Replication Probability-based Routing Scheme (RPRS)* in more detail. *RPRS* uses message information to prioritize the replication order of the messages based on the value of the calculated replication probability. Once a node contact encountered happens, the message with the most highly rated replication probability is copied to the other node. To calculate the replication probability of the message locally gathered information is used, which describe the situation of the network environment. It uses three parameters, namely replication count, message hop count and message buffer time. The first two variables are used as replication criteria, where the integrated drop policy utility uses the message buffer time and the two overhead variables of replication and hop count.

A. System Model

This section describes our model and assumptions. We consider an OppNet model based on the Markov Chain model with Ordinary Differential Equation as in [11], where this model consists of a set of mobile OppNet nodes. The nodes transmit the message to each other, once they encounter other nodes within communication range. During this transfer, the sender or relay node replicates the message while keeping a copy of the message in the buffer. A node can deliver messages to a destination node either directly or the message is delivered with intermediate nodes in a hop-by-hop manner. There are limited node and channel resources regarding storage and transfer bandwidth respectively. The mobility model of the nodes is assumed to be i.i.d distributed. Node meetings are assumed to be exponentially distributed. The nodes are assumed to be homogeneous nodes. In Table I, we summarize the notations and quantities that we use throughout our forwarding scheme design. From the modeling of the OppNet, we can write the Ordinary Differential Equation as:

$$\grave{R(t)} = \frac{\partial r}{\partial t} = \lambda_c \cdot r \cdot (N - r) \tag{1}$$

Solving equation (1) by integrating, we can write the number of message copies at instant time as:

$$r_t = \frac{N}{1 + (N-1)e^{-t \cdot N \cdot \lambda_c}} \tag{2}$$

In addition, we can calculate the probability [12] for message delivery for all messages in the system as follows:

$$P(t)_N = 1 - \frac{N}{e^{t \cdot N \cdot \lambda_c} + N - 1} \approx 1 - e^{-TTL \cdot \lambda_c \cdot N} \tag{3}$$

Please note, that nodes only hold a single copy of any message in maximum, but not more copies of the same message. Thus the probability is 1 in maximum. Furthermore, we can find the first differentiation of Equation (3) as follow:

$$\grave{P(t)}_N = TTL \lambda_c e^{-TTL \cdot \lambda_c \cdot N} \tag{4}$$

Table I
USED NOTATIONS AND QUANTITIES

No	Parameters	Description
1	N	Number of nodes in the network
2	n	Max relayed nodes in the network
3	ICT	Inter-contact time
4	r	Number of message
5	TTL	Time-To-Live for the message
6	$P(t)_N$	Probability of message copies delivery
7	r_t	Number of message copies in the entire network at time t
8	T_{BUF}	Queue time at the node buffer
9	nr	Current relayed nodes of the message
10	$\grave{P(t)}_N$	First derivative of delivery probability function
11	$R'(t)$	The rate of “infected” nodes carrying the message
12	H_c	Message hop counter
13	R_c	Node message replication counter
14	λ_c	Average meeting rate between two nodes
15	F_p	Replication Probability of the message
16	S_r	Stopping rule function replication
17	$S_{c(t)}$	Storage cost function
18	$T_{c(t)}$	Transmission cost function
19	D_p	Utility function of integrated drop policy

The RPRS forwarding scheme considers the message hop count and replication counter which affect the overhead of the message. For example, the replication counter impact on the hop count and the overhead of the message at the same time. The probability of successful replication depends on the sum of replication count and the hop count. The delivery decision and forward probability are based on local information: the message with highest replication probability value is going to be replicate on the next opportunistic encounter.

B. RPRS Replication Control Strategy

RPRS becomes active when two nodes meet. The scheme considers the message information to calculate the cost of every replicated message. *RPRS* also adapts to storage and bandwidth restrictions for the OppNet environment, where messages with the maximum utility are deleted first by the integrated drop policy. From Eq. (2), we can approximate the probability of a single message replication at the node, where the total number of nodes in the network is equal to the maximum number of intermediate nodes plus source and destination of the message. Therefore, $N = n + 2$, we calculate the forward probability function of the message by the following equation:

$$F_p = \frac{r_t}{N} = \frac{1}{1 + H_c + R_c} \tag{5}$$

From Eq. (5), we formulate the replication order strategy of *RPRS*, where this forwarding probability is a function of the overhead variables. The criteria of *RPRS* is how to determine the ordering of the messages, so that the most promising message to replicate is identified and what is the maximum number of the message copies required to achieve the desired delivery probability. The optimal replication criteria is derived under the assumption that, at any time, all the nodes have information about the number of relays carrying the message as hop count and the number of relays that have received the message as replication counter. We aim to determine the optimal replication stopping rule of each message, i.e. to identify when it does not make sense to further replicate a message, as a function of the message forwarding probability (F_p). The decision criteria decides whether which message node shall transfer/replicate when it meets an encountered node. The optimization goal is to minimize the cost based on the forwarding probability function. Further more, we want to order the messages as weighted messages based on the forwarding probability of Eq. (5). As shown from the equation, we found that the message originator gives the greatest weight to its own created messages to be replicated. This is shown in the value of (1) at the denominator of the function. Also, the messages in the send queue will be ordered based on that the high message cost will be in the front of the queue. This means that the message with lowest replication counter and hop count will have the highest chance to be replicated and those with high replication and/or hop count will stay behind. From Eq. (3), Eq. (4) and Eq. (5), we determine the stopping rule as an optimal stopping replication decision S_r with a threshold behavior as follows:

$$S_r = \frac{\grave{P(t)}_{N+1} - P(\grave{t})_N}{P(t)_{N+1} - P(t)_N} = F_p \tag{6}$$

Eq. (6) states that the optimal stopping function S_r for each node is to copy the message every time it is possible until the replication probability F_p of “infected” nodes is equal to the value of the forwarding probability F_p, or the message is delivered to the destination. Note that the threshold value depends on their replication counter R_c and message hop count H_c as the cost function.

C. Integrated Drop Policy

We assume that all the nodes of the network have the same replication strategy by running *RPRS*. The buffer management, aiming to optimize the delivery ratio, is thus of high interest. The number of message copies spread by *RPRS* in the network is related to the number of message drops when the buffer is full. Therefore, the integrated drop policy D_p considers the same variables of the forward probability function F_p, the variables related to the overhead as hop count H_c and replication counter R_c. Existing controlled routing protocols consider the overhead, but suffer from a high buffering delay resulting from the fixed number of the initial copies. Therefore our drop policy considers the buffer time in addition to a dynamic hop count and replication of the message. This dynamic behavior of the integrated drop policy considers the high dynamism of OppNet environments.

The drop function of *RPRS*’s drop policy consists of two parts in which the first part considers the storage cost function $S_{c(t)}$. This function computes the cost of the buffered message as the sum of the buffer time at the node that generated the message and the buffer time at the relaying nodes which is based on the hop count H_c. Therefore the value of the storage cost for buffering messages is calculated based on the following equation:

$$S_{c(t)} = T_{BUF} + H_c \cdot T_{BUF} \tag{7}$$

Algorithm 1 *RPRS* Message Replication

```
procedure READ($[H_c, R_c] \leftarrow Message$)
    $Sorting$                                  ▷ RPRS sorting based on $F_p$
    while $F_p \neq S_r$ do
        Calculate $F_p$ for $m_1$ and $m_2$
        if $(m_1, F_p) > (m_2, F_p)$ then
            $Message \leftarrow (m_1, F_p)$
        else
            $Message \leftarrow (m_2, F_p)$
    return $Message$
    $Replicate \leftarrow Message$
```

The second part of the drop policy considers the transmission cost function $T_{c(t)}$. As nodes are suffering from limited bandwidth, this function includes the cost of replicated message copies per node and is calculated as follows:

$$T_{c(t)} = R_c \cdot T_{BUF} \tag{8}$$

The drop policy D_p computes the weight of each message as sum of both storage and transmission costs as follow:

$$D_p = S_{c(t)} + T_{c(t)} \tag{9}$$

In *RPRS*, as we mentioned in Eq. (6) and Eq. (9), the forwarding policy is based on the F_p function with the replication stopping decision calculated as follows:

$$\text{send-queue} = \begin{cases} max\ F_p & S_r \neq F_p \\ stop & otherwise \end{cases} \tag{10}$$

In *RPRS*, we use the forwarding probability defined in Eq. (10) and the integrated drop policy from Eq. (9) in addition to the stopping rule of Eq. (6). Concluding, we show that *RPRS* depends on the three message parameters H_c, R_c and T_{BUF} as variables to control the replication of the message as an overhead problem from one side, on the other side it even considers the buffer delay as the main component of the end-to-end delay.

When two nodes come in each others communication range, the node replicates the message with the highest forward probability F_p for which the condition $F_p \neq S_r$ is true. The message replication process is shown in Algorithm 1.

IV. EXPERIMENT SETUP AND RESULTS

This section presents the evaluation of the implemented forwarding scheme *RPRS*. This evaluation considers metrics related to resource consumption for deep analysis. We consider the end-to-end delay and buffer delay at relaying nodes as main component of end-to-end delay. The comparison of the performance uses metrics related to an application as delay metric and resource consumption as overhead metric. The following list of performance metrics is used for comparison:

1) *Delivery Ratio* is the ratio of the number of delivered messages to the total number of generated messages.
2) *Overhead* is the average number of intermediate nodes used for one delivered message.

3) *Average End-To-End Delay* is the average delay of successfully delivered messages.
4) *Average Buffer Delay* is the average storage time spent by the messages at the nodes' buffers.

The performance of *RPRS* is compared with two types of flooding-based OppNet routing protocols, namely greedy uncontrolled Epidemic routing protocol and Spray & Wait as Quota-based controlled OppNet routing protocol. Both protocols are compared with different replication and drop policies. *RPRS* uses the send queue based on the Eq. (10), the drop policy which is applied based on the Eq. (9) and the replication stopping rule function based on Eq. (6). *RPRS* uses its replication probability F_p and its drop policy D_p.

For comparison, we select different forward and drop policies, the selected policies are based on the idea of *RPRS* parameters. Obviously, the selection is based on the overhead variables, replication counter and hop counter, of the message. We consider the buffer time as delay component for the drop policy. Therefore, the selected policies are based on the three variables of the message, from the suggested policies of those variables, we select *FIFO* which selects the message with the minimum arrival time, *MOFO* which selects the message with the maximum replication counter using the higher hop count as a tie breaker, *MaHo* which selects the message with the maximum hop count, *MiHo* which selects the message with the minimum hop count, *MaFo* which considers the maximum replication and *MiFo* which considers the minimum replication of the message. We use those different policies as replication or drop policies for the comparison of Epidemic and Spray & Wait routing.

We run all different policies using Epidemic and Spray & Wait routing protocols with the listed parameters in Table II and compare the performance with regards to the above mentioned metrics. A random waypoint model has been used as mobility model to allow a better comparison to other routing protocols in literature. In future, however, also more advanced movement models, such as Schelling's model [13] will be considered. With Epidemic routing, three scenarios were conducted under different message TTL values and otherwise with the default settings of the ONE Simulator [14]–[16]. For extending *RPRS*'s evaluation, we compare the performance of *RPRS* with the controlled routing protocol Spray & Wait with various drop policies based on the three variables of *RPRS*. Where, in all of the comparisons with the Epidemic and Spray & Wait routing protocols, we use the form of *RPRS* of $F_p - D_p$ as forwarding and drop policies.

A. Comparison with the Epidemic Routing Protocol

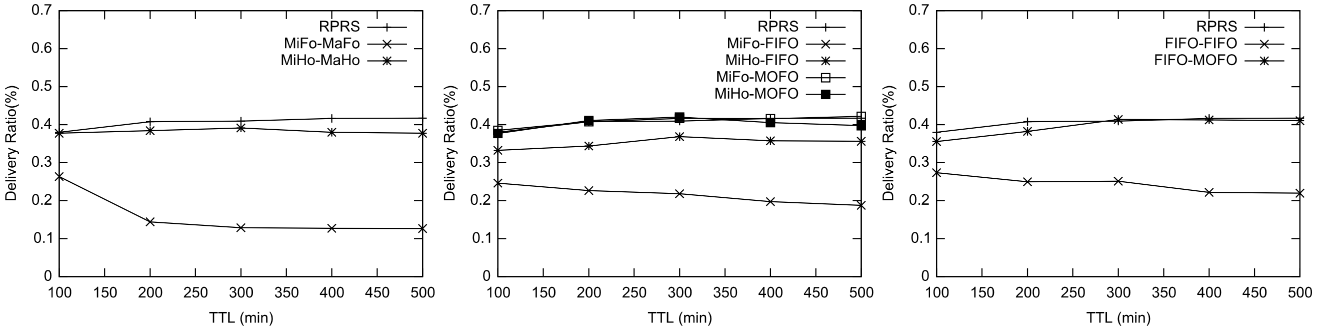
For the comparison of *RPRS* with Epidemic routing, we apply three different scenarios. For the first scenario, we consider the two variables used in F_p as shown in Eq. (5). Therefore we compare the Epidemic routing with the forward and drop policies MiFO-MaFo which replicates the minimum forwarded and drops the most forwarded messages, as well as the combination MiHo-MaHo which replicates the message with the minimum hop count and drops the message with the maximum hop count. For the second scenario, we add the third variable of buffer time as FIFO policy in addition to the drop policy of MOFO which considers the replication count as main criteria and hops count as the tie breaker. Finally, for the third scenario, we consider FIFO and MOFO as policies for Epidemic routing in comparison to *RPRS*.

Table II
SIMULATION SETTINGS

No	Settings	Value(s)
1	Simulation area	Helsinki, Finland Map
2	Simulation time	12 h
3	Number of devices (n)	126
4	Group Type with speed	80 Pedestrians (0.5-1.5 km/h) 40 Cars (10-80 km/h) 6 Trains (10-80 km/h)
5	Routing protocols	*RPRS*, Epidemic and Spray & Wait
6	Interface type	Simple Broadcast
7	Transmission range	250 m
8	Bandwidth	250 KBps
9	Drop policies used	FIFO, MOFO, MaHo, MaFo
10	Message size range	0.5-1 MB
11	Message creation interval	25-35 s
12	Time-to-live (TTL)	100, 200, 300, 400, 500 min
13	Default buffer size	Pedestrians: 5 MB Cars, Trains: 50 MB

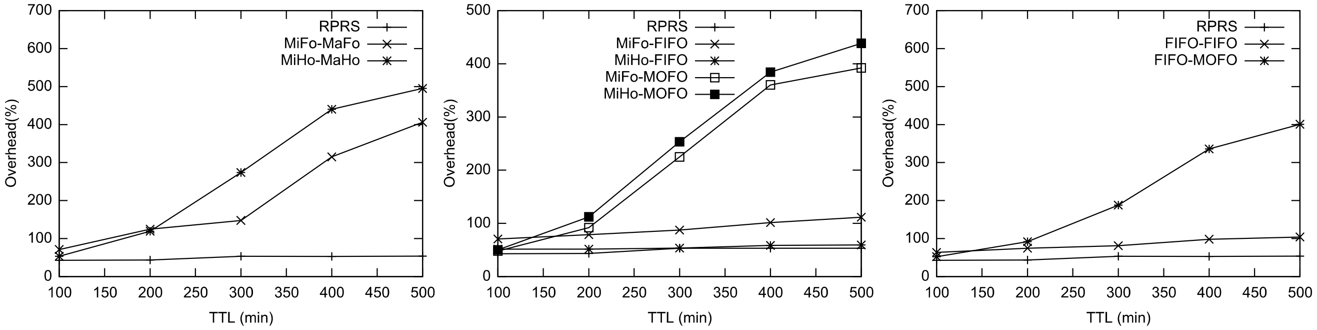
1) Delivery Ratio: *RPRS* uses as replication rule the F_p function for the send queue and its integrated drop policy defined in D_p. We compare *RPRS* with the Epidemic routing which uses the polices FIFO, MOFO, MaHo, MaFo, MiHo and MiFo as shown in Figure 2. The traffic pattern of Scenario 1 is shown in Table II and Figure 2 with different TTLs Values. From Scenario 1 the delivery ratio of *RPRS* is better than the delivery ratio of the Epidemic with different policies. Scenario 1 shows that the only the delivery ratio of Epidemic routing with MiHo-MaHo polices is close to the performance of *RPRS*, specially at TTL = 100 min the *RPRS* is better than Epidemic routing with 1% and at TTL = 500 min is the greatest difference, where the delivery ratio of *RPRS* routing is 4% higher than in Epidemic routing. The high delivery ratio of *RPRS* can also be seen in Scenarios 2 and 3, which use different forward and drop policies. The delivery ratio of *RPRS* has better performance in comparison to any policies for the Epidemic routing protocol.

At scenario 2, we can see that both MiFo-MOFO and MiHo-MOFO have close values to *RPRS*, but both policies are suffering from high overhead compared with overhead results of *RPRS* as shown in Figure 3. The same we see in Scenario 3, where we compare Epidemic routing using FIFO-MOFO with *RPRS*. Figure 2 shows that the delivery ratio efficiency is low, when using Epidemic Routing with MiFO-MaFO, MiFO-FIFO or FIFO-FIFO policies, as in some cases when selecting the message with longer buffer time or minimum replication counter this is not the suitable replication criteria. The message might have lower replication times or a longer buffer time at the node, but the message is not in the correct path to the message destination. This is considered by *RPRS* by using the message's hop count H_c and replication counter R_c in the replication probability function F_p.



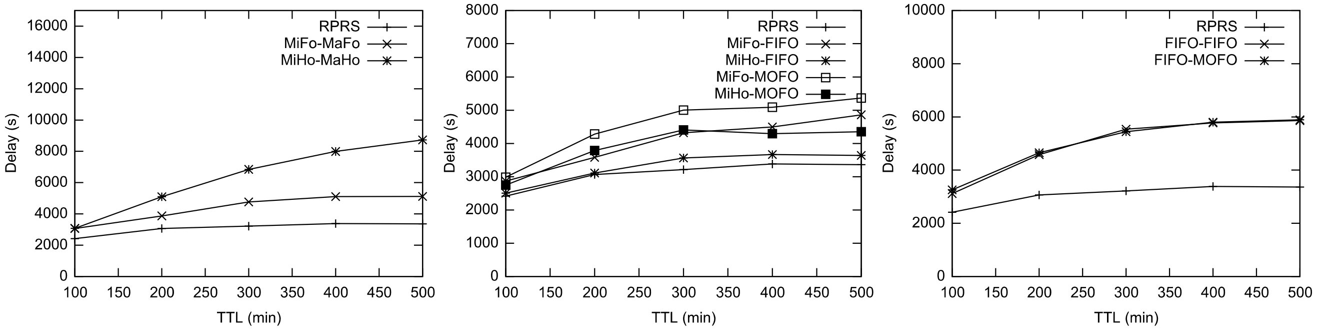
(a) Scenario 1 (b) Scenario 2 (c) Scenario 3

Figure 2. Delivery Ratio



(a) Scenario 1 (b) Scenario 2 (c) Scenario 3

Figure 3. Overhead

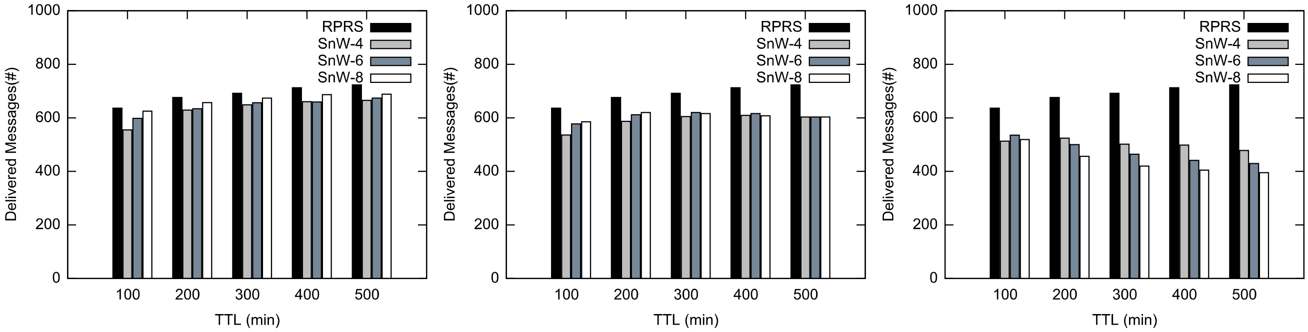


(a) Scenario 1 (b) Scenario 2 (c) Scenario 3

Figure 4. Delay

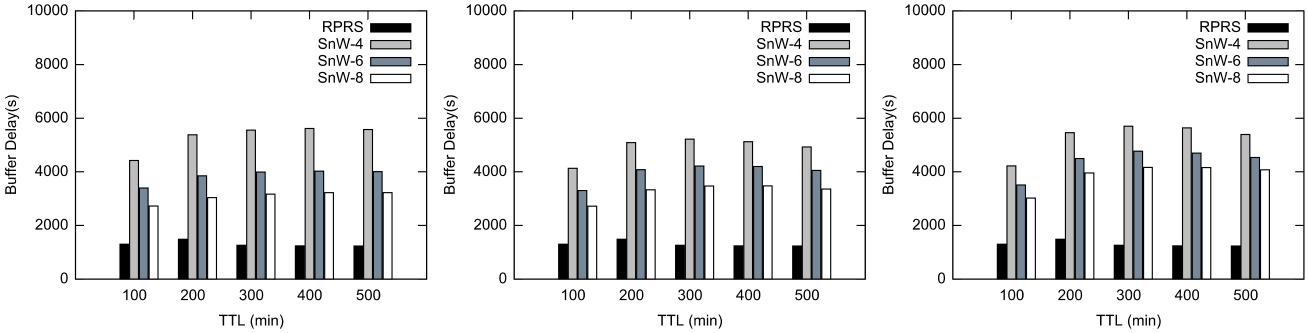
2) Overhead Ratio: The main performance factor to compare *RPRS* with Epidemic using different forward and drop polices is the overhead, as this factor is related to the resources of both a single node and the network. Due to the unlimited replication of the messages, the Epidemic routing protocol is suffering from the unnecessary consumption of resources. *RPRS* considers the resource consumption regarding storage and transmission metrics. Figure 3 shows that *RPRS* has the lowest overhead when compared to Epidemic routing protocol using different forward and drop policies. This is due to the replication probability, F_p of *RPRS*, which considers the overhead variables H_c and R_c. Also, the replication is controlled by the replication stopping rule S_r based on the probability of message replication. This message replication probability is calculated as a function of hop count (as storage metric) and replication counter (as transmission metric) of the message. The two variables $H_c and R_c$ imply the resource consumption in the corresponding equations. Hence, from Scenario 1 of Figure 3, we notice that *RPRS* has a stable and minimum overhead ratio. This stability derives from both the applied replication probability F_P and the drop policy. Scenario 1 and 3 show that a minimum overhead occurs through *RPRS* compared on Epidemic routing. Even in Scenario 2 of Figure 3, we can see that *RPRS* has a higher delivery ratio while having a lower overhead compared to Epidemic routing with MiHO-FIFO at TTL = 100 with about 7% and it reaches 12% at a TTL of 500 min.

3) Delay: In this section, we look at the average end-to-end delay as a metric used as a performance metric for different OppNet applications and scenarios. *RPRS* considers the delay at the integrated drop policy D_p. The drop function



(a) S&W with MaFo drop policy (b) S&W with MaHo drop policy (c) S&W with FIFO drop policy

Figure 5. Delivered Messages



(a) S&W with MaFO drop policy (b) S&W with MaHo drop policy (c) S&W with FIFO drop policy

Figure 6. Buffer Delay

D_p uses the buffer time as variable to select the message to drop when the buffer of the node is full. Scenario 1 of Figure 4 shows that *RPRS* has a lower delay compared to the Epidemic routing protocol using a different forward and drop policies. As we see from Figure 4 (a) the delay of delivered messages using *RPRS* ranges from about 2416s to 3363s. The closest Epidemic routing configuration uses MiFo-MaFo, leading to delay ranges from 3064s to 5118s. The message delay is lower when using *RPRS* in comparison to using Epidemic routing using MiFo-MaFo with a range of about 27% to 52.2% with different message TTL values. This reduction of delay results from the fact that *RPRS* considers the buffer time as a third variable in its integrated drop policy D_p. The message TTL is the sum of all buffer times and transmission times. Therefore, we consider the buffer times as the highest impact on the delivered message end-to-end delay. A message that is held in short in the buffer will have a higher chance of replication. Therefore, we consider the message buffer time as main criteria of the storage and transmission costs. As shown in Scenario 2 of Figure 4, *RPRS* has a lower delay when compared with Epidemic routing using MiHo-FIFO forward and drop policies with 1% at TTL = 100 min and a lower delay at TTL = 500 min with about 4%. In Scenario 3, *RPRS* has lower delay values compared to Epidemic using both FIFO-MOFO and FIFO-FIFO policies. In *RPRS* , we have less packets in the buffer and focus on the delivery of messages, which add most to the delivery probability. Thus, messages arrive quicker and are not delayed by repetitive message transfers of old messages, which might already have been delivered. Thus, *RPRS* has a higher delivery ratio at lower overhead and delay costs in comparison to Epidemic routing.

B. Comparison with Spray & Wait

A comparison with the quota-based Spray & Wait routing protocol was conducted to extend the performance evaluation of *RPRS* using different scenarios. The Spray & Wait routing protocol uses the settings in Table II with the binary mode of Spray & Wait and varying the number of initial message copies as 4, 6 and 8. We conduct scenarios for comparison of Spray & Wait using different buffer policies based on the three variables. The selected buffer drop policies were MaFo which drops maximum forwarded messages, MaHo which drop messages with maximum hop count and the FIFO policy. The goal of the scenarios is to observe the changes in the number of buffered messages and the rate of delivered messages. The various drop policies allow to evaluate the impact of fix and dynamic message copies in the routing performance on the delivery ratio and the buffer time.

1) Delivered Messages: To evaluate *RPRS* with Spray & Wait's routing performance, it is important to measure the delivered messages. The drop policy varies for Spray & Wait, where *RPRS* uses its integrated drop policy considering the buffer time, hop count and replication counter. Figure 5(a) shows that when the initial number of the messages copies increases the delivered messages will decrease. Spray & Wait suffers this problem from fixing the number of initial copies, where there is no number fit for all destinations. For all of the applied scenarios, we found that our *RPRS* forwarding scheme has a higher delivery ratio compared to Spray & Wait. Figure 5 shows that as the message's TTL increases the more messages are delivering to the destination by *RPRS* compared with Spray & Wait. It has with a message TTL of 500 min up to 24% higher delivery ratio at the setup of 4 initial copies and up to 32% higher delivery ratio at the setup of 8 initial copies. Figure 5 (b) and (c) shows that the implemented *RPRS* has a higher delivery ratio compared with Spray & Wait with both of MaHo and FIFO drop polices.

2) Buffer Delay: The buffer time, i.e. storage delay, is one of the critical variables together with the transmission delay which impact on the end-to-end delay. The available quota-based scheme of routing protocols considers the number of copies from the view of the overhead. The number of the copies even has an impact on the delay. In general, the routing protocol's replication decision should consider that the increase of the number of initial message copies increases the overhead. In opposite direction, decreasing or fixing the number of copies minimizes the overhead, while increasing the delay, specially the delay resulting by the buffering time. Therefore, we compare *RPRS* with Spray & Wait of different initial copies using the metric of buffer delay. As shown in Figure 6, Spray & Wait with all applied drop policies has a higher delay compared with *RPRS*. This is as *RPRS* uses its integrated drop policy which considers the buffer time in addition to the two overhead variables. Furthermore, we notice that the amount of the buffer delay caused by the Spray & Wait routing protocol decreases with regards to the number of initial message copies with any of the drop policies for Spray & Wait and with any TTL message values. Concluding, also here *RPRS* has a higher delivery ratio while coming with less costs in comparison to Spray & Wait.

V. CONCLUSION AND FUTURE WORK

This paper studies the replication issues in opportunistic networking environments by formulating the problem of routing replication decisions. The paper here considers the replication as a heuristic problem to obtain the optimal replication decision which is taken based on the resource constraints. We solve the problem based on three message characteristics which are replication counter, hop counter and buffer time. *RPRS* considers the trade-off between replication and resource consumption in opportunistic networks. The replication decision for the messages is taken based on the dynamic values of the hop count and replication. Furthermore, *RPRS* uses a drop policy which considers the storage and transmission costs. The performance evaluation in this paper shows that *RPRS* performs better regarding performance metrics such as delivery ratio as well as cost metrics such as overhead and buffer delay in comparison to Epidemic routing as an uncontrolled replication scheme and Spray & Wait as controlled (quota-based) scheme. For future work, we aim to extend our proposed forwarding scheme to be a gradient routing protocol which considers mobility or social variables. Also we aim to investigate security for routing in opportunistic networks in PeerfactSim.KOM [17] based on an Watchdog behavior, as proposed for wireless mesh networks in [18] and [19].

REFERENCES

[1] A. Lindgren, A. Doria, E. Davies, and S. Grasic, "RFC 6693: Probabilistic Routing Protocol for Intermittently Connected Networks," *IETF*, 2012.

[2] A. Vahdat and D. Becker, " Epidemic Routing for Partially-Connected Ad Hoc Networks," *Technique Report, Department of Computer Science, Duke University, USA*, vol. 20, no. 6, 2000.

[3] T. Spyropoulos and K. Psounis, " Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks ," *Proceedings of the ACM SIGCOMM Workshop on Delay-Tolerant Networking*, , 2005.

[4] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Spray and Focus: Efficient Mobility-Assisted Routing for Heterogeneous and Correlated Mobility," in *In Proceedings of the IEEE International Conference on Pervasive Computing and Communications - Workshops (PerCom Workshops)*, 2007, pp. 79–85.

[5] W. He, Y. Huang, K. Nahrstedt, and B. Wu, "Message Propagation in Ad-hoc-based Proximity Mobile Social Networks," in *Proc. of IEEE PerCom'10*, 2010, pp. 141–146.

[6] J. Xu, X. Feng, W. Yang, R. Wang, and B. Q. Han, "Optimal Joint Expected Delay Forwarding in Delay Tolerant Networks," *International Journal of Distributed Sensor Networks* , 2013.

[7] C. Liu and J.Wu, " An Optimal Probabilistic Forwarding Protocol in Delay Tolerant Networks," *International Journal of Distributed Sensor Networks* , pp. 105–114, 2009.

[8] A. Balasubramanian, B. N. Levine, and A. Venkataramani, "Replication routing in dtns: a resource allocation approach," *IEEE/ACM Trans. Netw.*, vol. 18, no. 2, pp. 596–609, 2010.

[9] S. Lo and W. Liou, "Dynamic quota-based routing in delay-tolerant networks," in *Proceedings of the 75th IEEE Vehicular Technology Conference, VTC Spring 2012, Yokohama, Japan, May 6-9, 2012*, 2012, pp. 1–5.

[10] S. Sati, C. Probst, and K. Graffi, "Analysis of Buffer Management Policies for Opportunistic Networks," in *IEEE ICCCN '16: Proceedings of the International Conference on Computer Communications and Networks*, 2016.

[11] X. Zhang, G. Neglia, J. F. Kurose, and D. F. Towsley, "Performance modeling of epidemic routing," *Computer Networks*, vol. 51, no. 10, pp. 2867–2891, 2007.

[12] E. Bulut, Z. Wang, and B. K. Szymanski, "Cost-effective multiperiod spraying for routing in delay-tolerant networks," *IEEE/ACM Trans. Netw.*, vol. 18, no. 5, pp. 1530–1543, 2010.

[13] L. Vu, K. Nahrstedt, and M. Hollick, "Exploiting Schelling Behavior for Improving Data Accessibility in Mobile Peer-to-Peer Networks," in *Proc. of Int. Conf. on Mobile and Ubiquitous Systems: Computing, Networking, and Services (MobiQuitous '08)*, 2008.

[14] A. Keränen, J. Ott, and T. Kärkkäinen, "The ONE Simulator for DTN Protocol Evaluation," in *Proceedings of the International Conference on Simulation Tools and Techniques for Communications, Networks and Systems (SimuTools)*. ICST/ACM, 2009.

[15] A. Keränen, T. Kärkkäinen, and J. Ott, "Simulating Mobility and DTNs with the ONE (Invited Paper)," *Journal of Communications*, vol. 5, no. 2, pp. 92–105, 2010.

[16] A. Cheraghi, T. Amft, S. Sati, P. Hagemeister, and K. Graffi, "The State of Simulation Tools for P2P Networks on Mobile Ad-Hoc and Opportunistic Networks," in *IEEE ICCCN '16: Proc. of the International Conference on Computer Communications and Networks*, 2016.

[17] K. Graffi, "PeerfactSim.KOM: A P2P System Simulator - Experiences and Lessons Learned," in *IEEE P2P '11: Proceedings of the International Conference on Peer-to-Peer Computing*, 2011.

[18] P. S. Mogre, K. Graffi, M. Hollick, and R. Steinmetz, "AntSec, WatchAnt and AntRep:Innovative Security Mechanisms for Wireless Mesh Networks," in *IEEE LCN '07: Proceedings of the International Conference on Local Computer Networks*, 2007.

[19] ——, "A Security Framework for Wireless Mesh Networks," *Wireless Communications and Mobile Computing*, vol. 11, no. 3, pp. 371–391, 2011.