Implementing Forward and Drop Policies for Improving PRoPHET's Routing Performance

Salem Sati
Technology of Social Networks Group
University of Düsseldorf, Germany
Email: Salem.Sati@hhu.de

Christopher Probst
Technology of Social Networks Group
University of Düsseldorf, Germany
Email: Christopher.Probst@hhu.de

Kalman Graffi
Technology of Social Networks Group
University of Düsseldorf, Germany
Email: Graffi@hhu.de

***Abstract*—Routing protocols for delay tolerant network are designed to operate in an environment where there is no stable end to end path. The DTN research has focused primarily on the performance of routing protocols that build on a Store-Carry-Forward manner without consideration of a physical buffer limitation. Therefore, the need of a message sorting and drop policy according to priorities is very critical and favorable in such restricted environments. In this paper, we present effective forward and drop policies to improve the performance of PRoPHET routing protocol. These forward and drop policies are applied on the PRoPHET version 2 routing protocol, which is currently commonly used as routing protocol for practical implementations of DTN middlewares. In this paper, we propose a stopping rule for the forward strategy in PRoPHET together with a dynamic drop policy. By this, we optimize the routing performance based on reducing the number of relayed messages while keeping a desired delivery ratio. To further improve the resource conservation, we propose new rules for message forwarding when encountering nodes. A forwarding error approximation is used to give the optimal threshold of the forwarding decision. In addition, we propose a dynamic drop policy which considers delay and overhead of the message based on multiple parameters. Simulation results show that the PRoPHET protocol with our new forward and drop policies gives better performance improvement in terms of relayed messages.**

Keywords: Opportunistic Network, PRoPHET Routing, Dropping Policies, Message Forwarding, Routing Performance.

I. Introduction

In Delay Tolerant Networks (DTN) an end-to-end path between a source and a destination node is not guaranteed. Messages are delivered based on hop-by-hop routing from a source node to the destination node via Store-Carry-Forward fashion. In DTNs, a source node or an intermediate node stores messages in its buffer and carries them while it moves around. An opportunistic network environment, as described in [1], faces long link absence which may partition the network due to node mobility or limited power. Routing in DTN has therefore to be adapted for variable messages and connections available. One main point in DTN routing is the assumption that each node has a local buffer that stores a limited number of messages. The limited number of messages in the buffer and long delays in an intermittently connected network, requires a prioritization of the stored messages for dropping and transmission. Such ordering and dropping conditions issues are very challenged in DTN environments as only a limited number of messages can be transmitted during the limited connection duration and due to the limitation of the physical node buffer that immediately impacts on routing performance in terms of message delay and delivery ratio.

DTN routing solutions can be categorized into flooding-based routing and utility-based routing. In flooding-based routing approaches such as Epidemic [2], message copies are replicated to every encountered node. However, the epidemic approach leads to high overhead and a degrading of the routing performance. On the other hand, in the utility-based routing, the messages are forwarded to other nodes based on calculated costs and eventually are delivered to the destination node via multiple hops. The main idea of utility-based protocols is to reduce the node resources consumption by reducing the number of message replication. This type of utility-based routing offers a limited message copy, as the DTN node is suffering in reality from limited node resources in terms of storage, bandwidth and power. There are a lot of proposed utility-based routing approaches which aim to consider the node buffer limitation by controlling the number of the message copies spread on the network. These protocols, still use multi-copy spreading in the network to achieve a high message delivery ratio, cause serious congestion and consumes nodes' storage space. Several utility-based routing schemes have been proposed, such as the PRoPHET (Probability ROuting Protocol using History of Encounters and Transitivity) protocol [3]. The forwarding option in PRoPHET is calculated based on the predictability cost between two encountered nodes, where the calculations are based on contact information history between connected nodes, where higher delivery predictability leads to higher probability of future contacts between encountered nodes. With the PRoPHET protocol, a message is copied to a contact node only when the delivery predictability cost to the destination node of encountered node is larger than that cost of the node which is carrying the message. By doing this, the PRoPHET protocol achieves good delivery probability as well as satisfying low overhead. Clearly, PRoPHET's forward strategy reduces the number of message copies and thus minimizes the node's resource consumption in terms of storage, bandwidth and power. Despite the idea of PRoPHET's utility function to improve the performance of the protocol, it still

This work was partially supported by the BMWi Grant KF2846503KM4 - within the program Zentrales Innovationsprogramm Mittelstand (ZIM)

assumes that the nodes have infinite buffer size. In addiction, the predictability cost value comparison between encountered nodes still has some errors of calculations. When PRoPHET uses the utility function, there is a lot of unnecessarily message copies which leads to high overhead ratio. Furthermore. the drop and forward policies play a very important role in the routing performance.

The question of how to design efficient and effective forward and drop policies, i.e. which message to select for forwarding upon a contact opportunity and which message to drop once the buffer is full, becomes a challenge. We choose the PRoPHET routing protocol as a basis for these policies, as it is commonly used by a lot of implemented practical middlewares such as IBR-DTN, JDTN and N4C. In [4], we show that most of the buffer management research focuses on one parameter either based on the node mobility or message information for both the scheduling or drop policies. Currently, most drop policies in the DTN buffer management, such as drop tail, drop front and drop largest are performing insufficiently in opportunistic networks since each one of these drop policies consider one of performance metrics as most relevant and take only one parameter as decision criteria. We believe that a lot of parameters should be considered which are related to the opportunistic environment. These parameters consider the mobility pattern as traffic indicator and message information as application indicator. We propose a drop policy which focuses on improving the message delivery ratio while also minimizing delay and overhead. In addition, we propose a new forward strategy based on the predictability difference of PRoPHET routing. This strategy keeps the number of delivered messages (delivery ratio) high, while decreasing the number of message copies in the network, which is more suitable regarding the resource limitation of mobile DTN nodes. We call our framework *Difference Forwarding Predictability with Dynamic Drop policy* (DFP-DDP).

The organization of this paper is as follows. Related work is presented in Section 2, while Section 3 explains the PRoPHET routing protocol operation and assumptions. The proposed forward and drop policies are described in Section 4. Section 5 illustrates the data sets as well as simulation setting, and it explains the experimental metrics and results. Section 6 concludes the results and suggests the future work.

II. Related Work

The PRoPHET protocol uses a Predictability Delivery metric to copy messages to other nodes in order to improve the routing performance. PRoPHET minimizes the message copies by selecting the appropriate node which has a higher predictability value. There are many approaches to improve PRoPHET's performance. In PRoPHET+ [5], the cost is calculated as sum of buffer size, battery power, location, popularity, and the delivery predictability. PRoPHET+ is not always applicable, such as when there are factors that are critical to the node, specially when the corresponding value is equal to zero. The authors of [6] propose a probabilistic routing protocol based on the message history. In the their approach, the cost is calculated using the messages hop count and delivery predictability to determine the relay node. In [7] the delivery predictability is calculated based on the distance between the nodes. In [8], the authors propose a controlled forward strategy for improving the performance of PRoPHET protocol that can optimize overhead metrics. The authors suggest new rules for message forwarding to encounter node. In [9], the authors propose an improvement for PRoPHET that helps to minimize the amount of unnecessary message copies by determining the relay capability of deleting messages that have already been delivered.

Many drop policies for challenging networks [10] are implemented to minimize the effect of the buffer limitation and to increase the performance of the network without increasing the delivery delay. In [4], we give an overview on the effect of 121 drop and scheduling policy combinations. Current drop policies concentrate on the message information or node mobility in case of a congested buffer, these available policies do not look at the dynamic behavior of the environment when to decide to drop the messages from the node's buffer. Clearly, these policies do not reduce the drop rate as needed by the DTN application. Rani [11] has evaluated the performance of the MOFO drop policy with different routing protocols. The paper shows that MOFO improves the message delivery ratio in opportunistic networks. Lindgren [12] has studied a few collections of available buffer management policies for PRoPHET. His paper shows that by selecting suitable buffer management policies, the performance of routing can improve. Global Knowledge-based Scheduling Drop (GBSD) proposed by Krifa et al. [13] further improves the performance, but requires global network state. Therefore, GBSD is unsuitable to be practical in actual environments. Daru Pan [11] propose to use a comprehensive integrated buffer, which takes all information relevant to message delivery and network resources into consideration. Shin [14] uses the estimated total number of replicas of a message to improve the performance. It considers two utility functions by message properties, particularly the number of replicas, the age, and the remaining TTL. Rashid et al. [15] propose a buffer management which takes message sizes into account. The weight criteria of the message is calculated based on the message's properties, which are message size, remaining TTL, message buffer time, hop count, and replication count. In [4], we take all available message parameters and consider them for drop and scheduling policies. We show the impact of their selection on delivery ratio, delay and overhead. Therefore, we pay more attention to improve the performance of PRoPHET, specially PRoPHET version two as it is commonly used in most available middlewares such as IBR-DTN, JDTN and N4C.

III. PRoPHET Routing Protocol

Before presenting our proposed forward and drop policies for PRoPHET, we firstly introduce the PRoPHET routing protocol. PRoPHET is one of the routing protocols for opportunistic networks which uses a costs function to determine the best suitable next node of a message to forward. It is a

Probabilistic Routing Protocol [16], [17]. It has been updated from version one which was based on the event of the contact to version two which considers the contact information, i.e. duration, frequency and history, as well. This section will consider the updated version two. In general, PRoPHET uses the node encounter history and the transitivity as metrics to calculate delivery predictability for an encountered node. The delivery predictability is computed based upon the number of encounters and duration of the connection at each node for a known destination to select the best next-hop for the buffered message. Hence, the node with frequent encounters and with stable connectivity holds high probability to be encountered again in near future. The nodes calculate the delivery probability using the following equation.

$$P_{(a,b)} = (1 - P_{(a,b)_{old}}) \cdot P_{enc}$$

Here $P_{(a,b)}$ is the current contact probability between Nodes a and b, $P_{(a,b)_{old}}$ is the previous contact probability and P_{enc} the probability for an encounter. Hence, the equation uses the P_{enc} instead of P_{ini} of the first version which clears that the version two is considering more contact information to calculate the delivery probability. In addition, PRoPHET uses information aging which describes that nodes with large elapsed encountering interval are expected not to be encountered in the future. Hereby, the delivery probability between two nodes is aging. It decreases the delivery probability based on the time interval and aging constant gamma - γ. The aging constant determines the reduction rate and is represented by the following equation.

$$P_{(a,b)} = P_{(a,b)_{old}} \cdot \gamma^k$$

The third part of the PRoPHET routing calculations is transitivity, the following equation is used to calculate the transitivity between the three nodes A, B and C.

$$P_{(a,c)} = (1 - P_{(a,c)_{old}}) \cdot P_{(a,c)_{old}} \cdot P_{(a,b)} \cdot P_{(b,c)} \cdot \beta$$

PRoPHET employs two different parameters when defining a forward strategy, the cost calculation determines the next hop of a message and how many copies of the message are to be forwarded. The combination of both parameters allows for different proposed forward strategies according to different scenarios and applications. Therefore, PRoPHET routing is considering a trade-off between its performance in terms of delivery ratio, delay, and node resource limitations in terms of energy and storage. The default strategy proposed by the PRoPHET Internet-Draft, GRTR, forwards the message if the delivery predictability to the destination of the message for the encountered node is higher than the node taking the decision. The Internet-Draft proposes many other strategies, such as GRTRMax which applies ordered GRTR taking into account the delivery predictability of encountered node towards the message destination, where GTMX adds a threshold (NFmax) that bounds the maximum number of forwards that the node is allowed to make for the specific message. Considering a node set with interconnected node A, a node set with interconnected node D and a mobile commuting node B between these two node sets, we can summarize the previous PRoPHET forward strategies as follows

- GTRT : $P_{(b,d)} > P_{(a,d)}$
- GTRTMax : $P_{(b,d)} > P_{(a,d)}$ & $DESC(P_{(b,d)})$
- GTMX : $P_{(b,d)} > P_{(a,d)}$ & $NF < NF_{max}$

However in the PRoPHET routing protocol, the dissemination speed of a message is relatively constant along the message's lifetime TTL, since a message is copied only when a delivery predictability condition is met. This leads to insufficient node resources consumption especially when the message lifetime is close to zero. In this paper, we improve the performance of the PRoPHET protocol by

1) Difference Forward Predictability (DFP) - a new forward strategy which considers the order of the delivery predictability of encountered node towards the message destination. In addition it takes the forward error approximation as forwarding decision for minimizing the overhead.
2) Dynamic Drop Policy (DDP) - a new drop policy which considers the delivery probability of the message, the probability of buffering at the relay node which carries the message and the overhead by the single message.

IV. Proposed Forward and Drop Policies for PRoPHET

The performance of a routing algorithm depends mostly on the forwarding decision, further more, there is some impact on the buffer management as drop policy on the efficiency of the routing algorithm, specially in DTN environment. PRoPHET exchanges messages between two encountered nodes according to the delivery predictability of the destination of the message. When two nodes meet, a message is forwarded from message carrier to encountered node only if the delivery predictability of the message destination is higher at the encountered node. We propose a forward strategy and drop policy for PRoPHET, those two approaches are implemented to improve the performance of PRoPHET protocol.

A. Proposed Difference Forward Predictability DFP

Most of PRoPHET forward strategies determine whether the message should be forwarded to the encountered node, which is done if the encounter node has a greater delivery predictability to the destination node than the message carrier. The inconvenience of this approach lies in the difference of the delivery predictability values of the encountered nodes. We investigate the value and the quantity of this difference. Further we trace whether there are any errors that would result in unnecessarily message copies. Finally we clarify whether this difference and the impacting error are variable or constant. For the optimization of the number of message copies forwarded by the PRoPHET protocol, we assume that we have N mobile nodes and the number of message copies R changed based on the following equation

$$R = r + \delta_r \tag{1}$$

Where R is the number of message copies at instance time (its value is in range of 1 to N), r is the desired number of copies to achieve the recommended delivery probability, and δ_r is the exchange rate of the message copies. According to the performance analysis of DTN routing presented in [18],

if R copies of a message are distributed in the network and the TTL is the message life time, then the expected delivery probability of the message can be formulated as

$$P = 1 - e^{-TTL\lambda_c R} \tag{2}$$

Where $\lambda_c$is the contact rate, this rate is the inverse of the average inter-contact time between N mobile nodes, From Eq. (2), we can find the first differential of the delivery probability as follow

$$\grave{P} = \lambda_c TTL e^{-TTL\lambda_c R} \tag{3}$$

Furthermore, we can formulate the number of message copies as function of P from the Eq. (4)

$$R = -\frac{ln(1-P)}{\lambda_c TTL} \tag{4}$$

We can found the relationship between both of δ_r and δ_P exchange rate with successive delivery probability of the message from the differential of Eq. (4) as follow

$$\frac{\delta_P}{\delta_r} = \lambda_c TTL(1-P) \tag{5}$$

Where $\delta_r = 1$ for every single forwarding decision, now we formulate our new forward strategy rule when two nodes encountered each other, the forwarding decision will be based on the following

$$P_{(b,d)} > P_{(a,d)} + \varepsilon \tag{6}$$

Now we want to calculate the difference or error ε of single forward decision of $\delta_r = 1$, the value of ε is calculated based on the principles of differentiation which consider the forward error approximation as follow

$$\grave{P} = \lim_{\delta_r \to 0} \frac{P(r+\delta_r) - P(r)}{\delta_r} \tag{7}$$

We have that $\delta_r = 1$, therefore, from Eq. (5), Eq. (7) and Eq. (1), we drop from both sides the constant value of $e^{-TTL\lambda_c r}$ as desired delivery probability, we get the following

$$\delta_P = 1 - e^{-TTL\lambda_c} \tag{8}$$

Finally our DFP implemented based on that the value of ε is constant for all of the messages and equal to the value of δ_P From Eq. (6) and Eq. (8) the implemented forward strategy DFP forward decision will be as follow

$$P_{(b,d)} > P_{(a,d)} + \delta_P \tag{9}$$

This means that for every single message, the message is forwarded by the PRoPHET protocol to the encounter node when the encounter node has higher a delivery predictability towards the message destination plus the value of the single message probability than the message carrier.

B. Proposed Dynamic Drop Policy DDP

The main aim of buffer management policies is to maximize the global delivery ratio of the network. Therefore, our proposed drop policy determines its drop decisions based on the message information and mobility patterns of nodes contacts. The proposed Dynamic Drop Policy (DDP) is designed based as utility function to be applied for each single message in the network. However, this information approximates the number of message copies disseminated at an instant time in the network. This information helps to decide when a message will be dropped and which message should be selected to drop. This drop message decision is made by the node based on its local information. There is the dropping criteria which was determined by the message originator in form of the value of the time-to-live (TTL_r), when the message was created. Normally, copies of the message will be dropped when its TTL_r is expired. This assumption relates to the maximum buffering time T_{BUF} = TTL_r. In some scenario the message with the longer buffering time at a relay node has not necessarily been replicated many times. Therefore, we would like to give a message higher chances of delivery if there is a free buffer space at the node. This motivates the normalizing by the ratio of message TTL only as shown in the utility function Eq. (10). In addition, to increase the delivery ratio, the system should minimize the delay and overhead of the messages in the system. With the Dynamic Drop Policy, at the time when a message needs to be dropped, the drop policy will discard the message m with maximum utility function value $U(m)$.

$$U(m)_{DDP} = \frac{TTL_e \cdot T_{BUF}}{TTL} \cdot (1 + H_c + R_c) \tag{10}$$

From Eq. (10), the TTL_e is the delay from the source node. T_{BUF} is the time the message spent in the buffer of the current node, while TTL is the initial life time of the message when it created. The parameter H_c is the hop count of the message, where R_c is the replica counter, i.e. the number of message copies this current node has already spread in the network. The first part of the function considers the time and delay of the message caused by the mobility pattern of the node movement. Here, $\frac{TTL_e \cdot T_{BUF}}{TTL}$ considers the delay from the source as TTL_e and the buffer delay at the current node. The two delay values are multiplied to reflect their impact on the end-to-end delay. They are set in relation to the message life time which is the maximum delay, i.e. TTL, allowed by the message originator. The second part of the utility function, i.e. $(1 + H_c + R_c)$, is related to the overhead of the message as we believe that the hop count H_c of the message is the overhead from the message source to the current message carrier R. On the other hand, the number of transmissions, i.e. replications R_c, will impact the overall network overhead. In addition, the constant one (1) in $(1+H_c+R_c)$ gives priority to new messages created at the node itself.

V. Experiment and Results

For the evaluation of quality of a routing protocol in opportunistic networks, we consider three main metrics: *Delivery Ratio* is the ratio of the number of delivered messages to the number of generated messages. *Overhead Ratio* is the average number of intermediate nodes used for one delivered message. *Average Delay* of all messages delivered successfully.

A. Data and Experimental Settings

The performance evaluation is performed using the ONE Simulator [19]. This evaluation compares the proposed DFP-DDP to PRoPHET using GRTRMax as forward strategy with different three representative drop policies: FIFO (First-in-First-Out), SHLI (Evict Shortest Life Time First) and

MOFO (Evict Most Forwarded First). In addition, we consider MaxRep and MaxHop drop policies which Maximize Replication (MaxRep) and Hop Count (MaxHop) as a message-related parameters to decide which message to drop. We compare the performance and costs of these drop policies in combination with PRoPHET V2 with GTRTMax forward strategy. As the TTL and the workload have a very large influence on the quality and costs, we varied the TTL between 5 values, namely 100, 200, 300, 400 and 500 minutes. Also we varied the workload to match different traffic patterns. The performance evaluated under different message TTLs shows the performance of the selected forward and drop policies under increasing congestion and increasing expected delay. Furthermore, the comparison with the increase of message TTL shows that the replication probability of all messages increase, therefore, the congestion and drop rate (overhead) in addition to delay will rise gradually. This is reasonable as a large TTL of a message gives more time for message copies to be buffered in intermediate nodes with free space without being discarded. Thus, with higher TTL, the delivery ratio raises, but also the delay and overhead raise. Normally when buffer spaces runs out, the message copies will be dropped based on the policy criteria, which will lead to a degrading in delivery ratio. The simulation setup is presented in Table I.

To reflect different traffic situations, we consider 3 scenarios. Scenario 1 only considers 80 pedestrians with a buffer size of 5MB, a message creation interval of 25s to 35s and message size ranging from 500KB to 1 MB. In Scenario 2, in addition to those 80 pedestrians from Scenario 1 also 40 cars join in. The scenario reflects a information-centric use case in which small (64KB - 512KB and 512B - 2KB) messages are quickly exchanged. Here the cars have a message creation interval of 1s - 5s. Finally, in Scenario 3 we investigate the effects of larger files and even faster nodes. 80 Pedestrians generate 1MB - 5MB large messages every 60s to 120s. 40 cars generate at an interval of 25s to 35s messages of 64KB to 512KB, while 6 nodes in trains generate every 1s to 5s messages of size 512B to 2KB. Through the variety of scenarios, in combination with the 5 values for the TTL we are able to fully reflect the quality of the proposed DFP-DDP Policies in comparison to the other forward and drop policies.

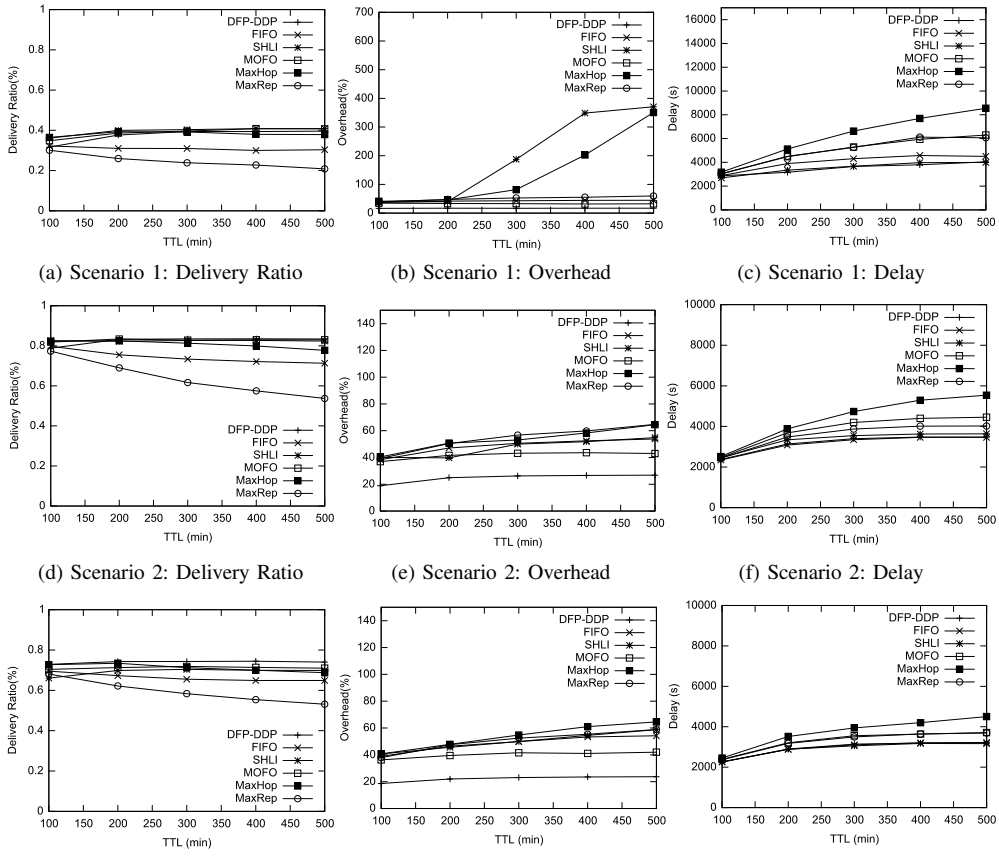
TABLE I
SIMULATION SETTINGS

No	Settings	Map of downtown Helsinki, Finland
1	Simulation time	12 h
2	Number of devices (n)	126
3	Group Type with Speed	80 Pedestrians (0.5-1.5 km/h) 40 Cars (10-80 km/h) 6 Trains (10-80 km/h)
4	Simulation area	Helsinki, Finland Map
5	Routing protocols	PRoPHET V2 with GTRTMax
6	Interface type	Simple Broadcast
7	Transmission range	250 m
8	Bandwidth	250 KBps
9	Drop policies used	FIFO, MOFO, SHLI, MaxHop, MaxRep
10	Message size ranges	Scenario 1: 0.5-1 MB Scenario 2: 64-512 KB & 0.5-2 KB Scenario 3: 1-5 MB & 64-512 KB & 0.5-2 KB
11	Message creation interval	Scenario 1: 25-35 s Scenario 2: 25-35 s & 1-5 s Scenario 3: 60-120 s & 25-35 s & 1-5 s
12	Time-to-live (TTL)	100, 200, 300, 400, 500 min
13	Default buffer size	Pedestrians: 5 MB Cars, Trains: 50 MB

B. Numerical Results

We compare our DFP-DDP with different drop policies in Figure 1 for the three scenarios while in Figure 2 we consider the number of relayed message to evaluate the resource concentrate in detail. The delivery ratio is the success indicator of the routing protocol's performance. Corresponding figures for the three Scenarios 1, 2 and 3 are (1)a, (1)d and (1)g respectively. We observe that only the proposed DFP-DDP, SHLI and MOFO achieve good results. The policy MaxHop yields also acceptable, but not consistently good results on par with DFP-DDP, SHLI and MOFO. The achievements of DFP-DDP are better than those of SHLI and MOFO. For Scenario 1, we observe only a maximum percentage difference in the delivery ratio of 2.2553% for Scenario 2, TTL=300 min, where MOFO is able to deliver 2.2553% more messages than DFP-DDP. In general, the differences in the delivery ratio are quite small with DFP-DDP being in the top grade. In Scenario 1, MOFO and SHLI result in up to 1.3% more delivered messages with a maximum of 40.87% delivered messages with SHLI in comparison to 39.58% delivered messages for DFP-DDP in Scenario 1, TTL=500 min. In general, the average difference in the delivery ratio between proposed DFP-DDP policy regarding SHLI and MOFO in scenario 1 are around 1,2 % respectively. The delivery ratios are rather low here for all drop policies. This is due to the low mobility of the nodes and potentially large distances to travel for a message for a successful delivery. Figures 1(a), 1(d) and 1(g) reflect the delivery ratio for the various TTLs and scenarios. MOFO and SHLI are very low compared to proposed DFP-DDP, there are in maximum 4.5% more messages delivered in Scenario 3 by proposed DFP-DDP. We compare the delivery ratio to the resource consumption. Figure 2 shows the delivery ratio compared regarding the number of required relayed nodes for all of DFP-DDP and other drop policies. This is coming from that the proposed DFP-DDP has the ability to adapt and consider the overhead error in both decisions of forward and drop decisions as the TTL of the message increases or traffic pattern rate increases.

The overhead ratio for varying the TTL of the messages and the traffic rate through the scenarios is the main metric regarding the resource consumption. Here it is worthwhile to mention that the proposed DFP-DDP has the lowest traffic overhead compared with the other different drop policies, while MaxRep and MaxHop are the worst from the optimal DFP-DDP specially in Scenario 2 and 3. Figures 1(b), 1(e) and 1(h) show that the overhead of SHLI grows greatly in Scenario 1. In three mentioned overhead figures we see that with increasing TTL both policies SHLI and MaxHop are affected. In Scenario 1 with TTL=500 min, SHLI produces enormous traffic overhead of up to 370.81%, MOFO produces an overhead of 32.28%, while the proposed DFP-DDP remains throughout the scenarios and TTL variations at 16.42% to 17.12%. MOFO requires in best case 16.16% greater traffic overhead than DFP-DDP but in the worst case 2.4 times more than proposed DFP-DDP. The overhead ratio of the proposed

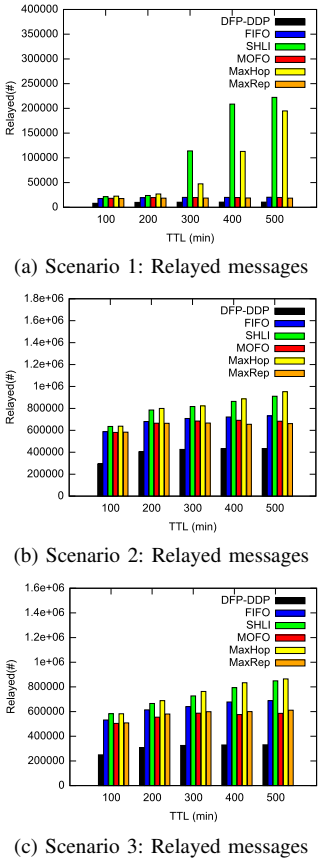


(a) Scenario 1: Delivery Ratio (b) Scenario 1: Overhead (c) Scenario 1: Delay

(d) Scenario 2: Delivery Ratio (e) Scenario 2: Overhead (f) Scenario 2: Delay

(g) Scenario 3: Delivery Ratio (h) Scenario 3: Overhead (i) Scenario 3: Delay

Fig. 1. Measurement Results with regard to the three Scenarios



(a) Scenario 1: Relayed messages

(b) Scenario 2: Relayed messages

(c) Scenario 3: Relayed messages

Fig. 2. Relayed Messages with regard to the three Scenarios

DFP-DDP stays with few exceptions always less than all compared different drop policies at a very constant level of around 16.41-17.12%. This is as DFP-DDP considers the overhead, in terms of Hop Count and Replication Count for the drop policy, for the history of the message from the message originator to the message's final destination (as shown in Eq. (10)). Also, the proposed forwardin strategy consider the forward error approximation as decision which eliminate unnecessary message copies for minimize the relayed messages (as shown in Eq. (9)). SHLI does not consider the overhead at all as it uses the message TTL as decision-making condition for the message to be dropped. On the other hand, the MOFO policy creates less overhead than SHLI as the MOFO decision is considering the Hop Count and Replication Count only indirectly. While SHLI is to be considered as not practicable for this reason, MOFO has problems with delay. A decision on the quality of the remaining drop policies is given through the observation of the delay of the delivered messages.

The delay for the different Scenarios 1 to 3 with varying values for TTL and traffic rate are shown in Figures 1(c), 1(f) and 1(i). The figures show that the delay increases as message TTL increases in all three scenarios. While for Scenario 1, SHLI and FIFO yield the closest average delay compared to proposed DFP-DDP delay, DFP-DDP has better values of around 1-2% and has the best delay values for Scenario 1 and 2. The quality of DFP-DDP becomes even more clear when compared with MOFO, MaxHop and MaxRep in Scenario 1 and 2. The efficiency of the proposed DFP-DDP for minimum delay comes from the drop policy considering the time as message buffer and TTL (as in Eq. (10)). For Scenarios 1 to 3, MaxHop drop policy has the worst delay, as this policy dose not consider any message time indicator in terms of buffer or life time. In Scenario 3, MOFO has more delay than proposed DFP-DDP, the delay difference of MOFO varies as message TTLs vary. MOFO has in average much higher delays, ranging from 2% to 5% more than proposed DFP-DDP, for the delivered messages.

SHLI considers the delay from the message originator to the current message carrier. The proposed DFP-DDP considers both the TTL as well as the buffer time and thus gains an additional advantage. The MOFO policy does not consider the delay at all, as it uses the message overhead. MaxHop and MaxRep ignore the delay completely, as we see in Scenario 1.

C. Relayed Message Optimization

The main idea of the proposed new forward strategy DFP is to optimize the number of message copies. This metric, in terms of relayed messages, is presented in Figure 2 in comparison to the performance of the PRoPHET protocol. Figure 2 summarizes the three scenario results. We compare PRoPHET protocol using GTRTMax with different drop policies and our new forward and drop policies DFP-DDP with different message TTLs. At first, from Scenario 1 of Figure (2)a, PRoPHET exchanges more messages than our new proposed

forward strategy DFP, while PRoPHET with MaxRep, FIFO and MOFO exchanges 17576, 17809 and 18143 messages at TTL = 100 min to 18526, 20336 and 19837 messages at TTL = 500 min respectively, our DFP forward strategy exchanges 8063 messages at TTL=100 min and 10494 messages at TTL= 500 min. This means that DFP reduces the relayed message to about 50% of relayed messages by MaxRep, FIFO and MOFO. The worst values for relayed messages in Figure (2)a were by the SHLI and MaxHop as they give more time to messages to stay in the network. As indication of the efficiency of proposed forward strategy DFP, we found that in Scenario 1 our approach required in maximum 12 copies of a message, while other drop policies require 50 copies, this mean that our approach removed around 75 % of overhead of this message. The other Scenarios 2 and 3 in Figures (2)b and (2)c show that the new proposed forward strategy remove around 50% of message copies while being still best or at par with the message delivery. Therefore, we confirm that our proposed forward strategy DFP optimizes the forwarding decision based on the difference of delivery predictability. This difference considers the forward error approximation of message delivery probability. This advantage of the proposed forward strategy is to enhance or at least save the number of delivered messages regardless of increasing the rate of relayed messages. Obviously, relayed messages are reduced, which is more suitable concerning the battery power limitation of mobile DTN nodes. The results of Figure 2 show that our new forward strategy DFP performs better than other solution of the PRoPHET routing, which are well known by the research community of mobile DTN.

VI. Conclusion and Future Work

Opportunistic networks target to provide the communication in intermittently connected networks. The major challenge in this environment is to increase the message delivery and to minimize both overhead and delay for the routing protocol.

In this paper, we proposed DFP-DDP as forward and drop policies to improve the performance of the PRoPHET routing protocol version 2. Using the ONE simulator, we evaluated our proposed DFP-DDP with different PRoPHET routing protocol settings. For comparison, we use PRoPHET with GRTRMax forwarding combined with different drop policies such as FIFO, MOFO and SHLI, in addition to maximum hop count and replication drop policies. The performance of the proposed DFP-DDP policies was analyzed with different message TTLs and different traffic scenarios. Evaluation shows that proposed DFP-DDP improves the performance of PRoPHET in terms of message delivery ratio, overhead and delay. The proposed DFP-DDP is better that the existing approaches as it is able minimize the overhead ratio and the delay while being best in the message delivery. while it uses the forward error approbation as forwarding decision and dynamic drop policy considering the overhead and delay. Thus, we improved the performance of PRoPHET protocol version 2 by employing the proposed DFP-DDP. An interesting extension is the analysis of social-based forwarding, laying a focus on heterogeneous optimization. In the future, we aim to simulate distributed applications on opportunistic networks in large-scale simulators [20], and to consider the security challenges in fully mobile [21] and infrastructure-supported [22] hybrid opportunistic networks.

References

[1] M. Conti, S. Giordano, M. May, and A. Passarella, "From Opportunistic Networks to Opportunistic Computing," *IEEE Communications Magazine*, vol. 48, no. 9, pp. 126–139, Sept 2010.

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

[3] A. Lindgren, A. Doria, and O. Scheln, "Probabilistic Routing in Intermittently Connected Networks," in *Service Assurance with Partial and Intermittent Resources*. Springer, 2004, vol. 3126, pp. 239–254.

[4] S. Sati, C. Probst, and K. Graffi, "Analysis of Buffer Management Policies for Opportunistic Networks," in *Proc. of the IEEE International Conference on Computer Communication and Networks (ICCCN)*, 2016.

[5] T. Huang, C. Lee, and L. Chen, "PRoPHET+: An Adaptive PRoPHET-Based Routing Protocol for Opportunistic Network," in *Proc. of the IEEE International Conference on Advanced Information Networking and Applications, AINA*, 2010, pp. 112–119.

[6] F. C. Lee and C. K. Yeo, "Probabilistic Routing Based on History of Messages in Delay Tolerant Networks," in *Proc. of the IEEE Vehicular Technology Conference, VTC Fall 2011*, 2011, pp. 1–6.

[7] P. Sok and K. Kim, " Distance-based PRoPHET Routing Protocol in Disruption Tolerant Network ," in *Proc. of the International Conference on ICT Convergence (ICTC '13)*, October 2013, p. 159164.

[8] A. E. Ouadrhiri, M. El-Kamili, M. R. El-Fenni, and L. Omari, "New Forwarding Strategy for PROPHET Routing in Delay Tolerant Networks," in *Proc. of the International Conference on Networked Systems, NETYS 2013, Revised Selected Papers*, 2013, pp. 300–305.

[9] Y. Vang, A. Saavedra, and S. Yang, "Ferry Enhanced Improved PRoPHET Routing Protocol," in *Proc. of the IEEE International Conference on Mobile Ad Hoc and Sensor Systems*, 2015, pp. 567–572.

[10] K. Graffi, K. Pussep, S. Kaune, A. Kovacevic, N. Liebau, and R. Steinmetz, "Overlay Bandwidth Management: Scheduling and Active Queue Management of Overlay Flows," in *IEEE LCN '07: Proceedings of the International Conference on Local Computer Networks*, 2007.

[11] D. Pan, Z. Ruan, N. Zhou, X. Liu, and Z. Song, "A Comprehensive-integrated Buffer Management Strategy for Opportunistic Networks," *EURASIP J. Wireless Comm. and Networking*, vol. 2013, p. 103, 2013.

[12] F. Legendre and A. Helmy, Eds., *Proc. of the ACM workshop on Challenged networks, CHANTS@MOBICOM'11*. ACM, 2011.

[13] A. Krifa, C. Barakat, and T. Spyropoulos, "Message Drop and Scheduling in DTNs: Theory and Practice," *IEEE Transactions on Mobile Computing*, vol. 11, no. 9, pp. 1470–1483, 2012.

[14] K. Shin and S. Kim, "Enhanced Buffer Management Policy that utilises Message Properties for Delay-tolerant Networks," *IET Communications*, vol. 5, no. 6, pp. 753–759, 2011.

[15] S. Rashid, Q. Ayub, and A. H. Abdullah, "Reactive Weight Based Buffer Management Policy for DTN Routing Protocols," *Wireless Personal Communications*, vol. 80, no. 3, pp. 993–1010, 2015.

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

[17] S. Grasic, E. Davies, A. Lindgren, and A. Doria, "The Evolution of a DTN routing protocol - PRoPHETv2," in *Proc. of the 6th ACM Workshop on Challenged Networks*, 2011, pp. 27–30.

[18] 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.

[19] A. Keränen, T. Kärkkäinen, and J. Ott, "Simulating mobility and dtns with the ONE (invited paper)," *JCM*, vol. 5, no. 2, pp. 92–105, 2010.

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

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

[22] P. S. Mogre, K. Graffi, M. Hollick, and R. Steinmetz, "A Security Framework for Wireless Mesh Networks," *Wireless Communications and Mobile Computing*, vol. 11, no. 3, pp. 371–391, 2011.