Improving Message Delivery in Opportunistic Networks with Fragmentation and Network Coding

Raphael Bialon Jan Tölkes Kalman Graffi
Department of Computer Science, University of Düsseldorf, Germany
bialon@hhu.de, jan.toelkes@hhu.de, graffi@hhu.de

***Abstract*—In opportunistic networks, no guarantees about network connectivity and node behavior can be made. Routing protocols aim at delivering messages by choosing promising paths throughout the network. These paths are created by nodes replicating messages and forwarding them to other nodes upon encountering them. The first replicated message reaching its destination node is the successful delivery, while others routed along different paths might arrive at a later time. The order and possibility of arrival depend on many factors such as the path length and sufficient transfer time for a complete message upon inter-node contact. We motivate the use of distinct paths throughout the network for Fragmentation and Network Coding and propose Fragmentation Strategies on the edge of the opportunistic network improving general message delivery. By carefully choosing appropriate fragment sizes and the amount of network coding added, we show that our Fragmentation Strategies are able to improve message delivery for large messages regardless of the routing protocol in use. As the main contribution of this paper, we provide the mathematical equations needed for fragment creation, and make available an implementation of our Fragmentation Strategies.**

I. Introduction

Opportunistic networks, as given in [1], form a subclass of Delay Tolerant Networks. A heterogeneous or homogeneous set of participating nodes forms the not fully connected network by spontaneously connecting to other nodes upon encountering them. The topology of the network changes dynamically and is influenced by every nodes' movement and message transmissions. No guarantees on connections can be made, neither on connection time nor message throughput. These properties create new challenges for routing protocols when compared to regular computer and ad-hoc networks.

Opportunistic networks may be applicable in regions with impaired network infrastructure or no infrastructure at all, such as not having sufficient Wi-Fi or mobile cellular internet connection. Beyond, disruption of network infrastructure in the context of a catastrophe or censored internet access may lead to a loss or to restriction of internet connection. These situations make it challenging or even impossible to gain network access. If an opportunistic network is established in such a region, providing adequate delivery probabilities of messages leads to some level of networking ability. In those situations, opportunistic networks are able to provide a best-effort network access. The applicability does not stop at providing basic network functionality, but is also able to support a more sophisticated usage when combined with or used in addition to existing infrastructure. Especially increased distribution of mobile applications with focus on delivering photo and video to the user lead to a higher cellular mobile internet traffic. Sending and receiving such large messages through opportunistic networks removes for example the costs when using a centralized infrastructure administered by some central provider.

Routing protocols can make no assumptions about predetermined paths through the network, leading to an individual forwarding decision to be made by each node routing a message. In recent years, we have seen vast efforts to introduce new and improved routing strategies in the context of opportunistic networks. Much effort has been put into improving single routing protocols focusing on utilizing network characteristics in their favor. At the same time, various improvements in general network usage were researched independently of opportunistic networks. Still, not every message to be sent through an opportunistic network has the same delivery probability. Large messages will always be less likely to be successfully forwarded to their destination as intermittent links do not provide necessary properties caused by the dynamic behavior.

With this paper, we want to provide an improvement of message delivery for any routing protocol utilized in the network for large message having generally lower delivery probabilities.

Our general idea is based on the multiple paths a message can take through the network due to individual forwarding decisions as shown in fig. 1. The only difference leading to different sets of paths this specific message takes is the buffer management strategy, the routing protocol stayed the same. Also, general node behavior and properties such as its movement, buffer levels, and transmission speed and range influence message forwarding.

We present strategies which can be applied in all use cases of opportunistic networks where larger message sizes may occur. These strategies aim at being independent of other network parameters such as the routing protocol and thus are not only applicable in the previously named use cases, but in a vast area of network parameter constellations. In general, our strategies aim at improving single node contact interactions which previously cannot always handle large message transfers, leading to general improvement of message delivery.

Therefore, we provide the necessary background information on message paths throughout the network, message fragmentation and network coding which are later applied in our strategies.

Throughout this paper, the following questions are answered in the context of opportunistic networks:

- Can message delivery benefit from different paths a message can take through the network?
- Can proactively fragmenting messages improve delivery probability?
- Can network coding improve message delivery for fragmented messages even more?
- Within which thresholds does fragmentation allow for an improved message delivery?

As our main contribution of this paper, we provide a mathematical model supporting our strategies and a simulator implementation allowing the evaluation carried out in this paper as well as further usage in other use cases.

The remainder of this paper is structured as follows: In section II, we give an insight on the fundamental observations our strategies are derived from. Following in section III, an overview on existing related work is given, outlining common parts and differences between our work and others. Our methodological approach followed by a description of our simulation and evaluation setup is depicted in section IV. Section V addresses our simulation results. Finally, we give a conclusion of our work and an outlook on future work in section VI.

II. Background

A. Fragmentation

Fragmentation can be used in a proactive and reactive manner. For proactive fragmentation, both sender and receiver have to fragment and defragment the message, while the other nodes are not involved in the fragmentation process. Fragment sizes are chosen on the node creating the message before sending out fragments. Fragments routed through the network are not changed within the network regarding their size.

In the course of reactive fragmentation, the size of message fragments routed through the network is dynamically adjusted by participating nodes, resulting in different fragment sizes when routing messages from start to destination node. Reactive fragmentation therefore requires participating nodes to be able to actively fragment the messages being routed.

Thus, proactive fragmentation does not require additional logic on participating nodes except for sender and receiver, whereas reactive fragmentation distributes the fragmentation decision making process along participating nodes.

B. Network coding

Network coding describes utilizing the possibility of sending multiple messages as one single message. This single message m embodies n encoded messages m_i where $i \in \{1, 2, \ldots, n\}$. Every node to receive message m_i may generate m_i out of m if the encoding scheme and all $m_j, j \in \{1, 2, \ldots, n\} \setminus \{i\}$ are known to the specific node [2].

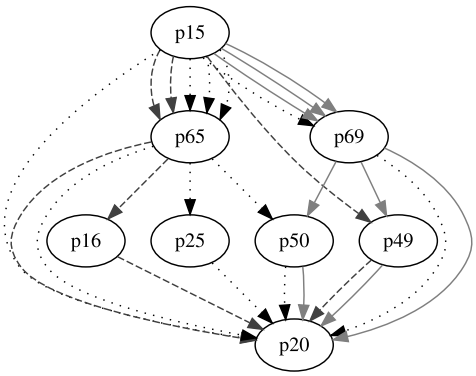


Figure 1. Various message paths for FIFO (dashed), RANDOM (dotted) and ROUNDROBIN (solid) buffer sorting.

C. Forward error correction

Forward error correction may be used to restore data which is corrupt or only partly known. In this case, the missing or corrupt data can be restored when a sufficient amount of data is known and not corrupt.

D. Network coding and forward error correction

The forward error correction based approach we use in the course of this work constitutes network coding like forward error correction to fragment the payload of a message to be sent as follows:

Payload ρ of message A is fragmented in n fragments $\{A_1, A_2, \ldots, A_n\}$ of size ϱ such that $n \cdot \varrho = \rho$. In addition m fragments $\{A_{n+1}, A_{n+2}, \ldots, A_{n+m}\}$ of size ϱ may be created. All fragments encode each other, so that ρ may be retrieved from any n fragments $\{A_i\}, i \in \{1, 2, \ldots, n+m\}$. In this case, not all fragments of a fragmented message must be received to retrieve the payload but only a share given by $\frac{n}{n+m}$. This exploits the possibility of not successfully delivering all fragments in order to receive the message A.

To be able to effectively compare the fragmentation- with the forward error correction based approach proactive fragmentation must be chosen due to the need to receive full payload of all successfully delivered fragments in order to retrieve the total payload of the source message. Thus proactive fragmentation is the only suitable choice to be able to compare results of fragmentation with the forward error correction approach we investigate in the course of the present paper.

No additional requirements are introduced for the network except for the origin and destination node. In particular no additional logical requirements on the vehicle nodes are introduced.

E. Multi-Path routing

Figure 1 shows the various paths a message sent from node `p15` to node `p20` takes, depending on the given buffer management strategy. The underlying message is exemplary for other messages as well. Possible ways a message can take throughout the network are affected by various parameters, such as node movement, the buffer management strategy,

transmission speed and range, and many more. We emphasize this possibility of profiting from various successful routes depending on network parameters to be used for our Fragmentation Strategies. It is known that different buffer management strategies influence message delivery probabilities as shown in [3]. We focus on buffer management strategies allowing for a uniform distribution of messages selected from the buffer to send out different fragments along the distinct paths taken throughout the network. By exploiting different possible routes traffic of one message may be distributed on the various routes taken by successfully delivered fragments. Thus, fragmentation may lead to increased utilization of the network's resources.

III. Related Work

Packet fragmentation is defined since the beginning of modern computer networks, it is already part of the IP standard [4]. Network coding, being an albeit newer concept, was first defined in [5]. It has quickly shown to provide great improvements in computer networks. For opportunistic networks, several evaluations on different fragmentation and network coding strategies have been conducted. In the following, we give a short overview on those publications and state the main difference between their and our work.

In [6], the authors implement several both proactive and reactive fragmentation strategies which work independently of the utilized routing protocol. They motivate the use of fragmentation using a reasoning sharing common observations with ours: Large messages have lower delivery probability due to longer transfer times and therefore higher probability of failed transfers. They evaluated their implementations using the ONE simulator and showed that reactive fragmentation leads to a higher delivery probability. In their conclusion, they also outline that a message may travel along multiple distinct paths but conclude that this leads to an impaired routing process due to premature message fragmentation. In our paper, we utilize the multiple paths a message travels along to improve delivery probability using proactive fragmentation. Thus, we elude the impairment of premature fragmentation by transmitting fragments right from the beginning with additional network coding to allow for a better fragment assembly at the receiver.

In [7], Feng et al. published their investigations on the use of fragmentation in opportunistic networks. Their fragmentation strategy called SADF operates on two levels of fragmentation: One general fragmentation with static sizes, which is applied onto every message when it is first introduced, and a second fragmentation using variable sizes according to the current network status. Also, this fragmentation strategy is embedded in a novel routing protocol and cannot directly be used independently of it. While their findings also shown an improvement in message delivery, it is bound to their routing protocol.

Wang et al. show similar improvements using another novel routing protocol based on erasure coding in [8]. Additionally, the size of their network coded packets is not related to the network characteristics, possibly missing out another improvement in message delivery.

Other routing protocols using network coding to achieve an improved message forwarding are given in [9], [10]. Again, they introduce novel routing protocols showing improvements in message delivery. Both require the defined routing protocols to be present on participating network nodes and cannot be directly applied onto other routing protocols.

In summary, a variety of network coding routing protocols and fragmentation strategies exist. Each provides improvements in message delivery for opportunistic networks, but most solutions rely on novel routing protocols or an active participation of network nodes in the fragmentation process. With our approach, we provide a routing protocol-independent strategy for optimized message delivery in opportunistic networks using fragmentation and network coding imposing no further requirements for participating nodes in the middle of a network.

IV. Methodological Approach

This section covers the integration of the concepts presented in section II into *Fragmentation Strategies* that aim at solving the problem statement given in section I. Having compared our work with already existing related work, we want to outline that our contribution provides additional functionality at the edge of the network. The middle of the network with all other participating nodes, besides sender and receiver, needs not to participate in the fragmentation process, while still contributing to an improved message delivery. Our solution therefore does not depend on the chosen routing protocol and enables a broad application onto many different use cases. Additionally, we show that our solution is also independent from other network parameters as well in the evaluation part further on.

A. Strategies

In the following we model strategies based on the rationale motivated in section II.

1) Exploiting Multi-Path Routing: In section II-E, we motivated that different messages may take distinct paths throughout the network. Due to the dynamic topology of opportunistic networks, contacts to a set of different nodes occur. These nodes themselves again experience various encounters with other nodes. This leads to various opportunities for a variance of possible routes a message may take through the opportunistic network.

This variance in potential ways through the network is utilized for the fragmentation strategies. On a node-based view, the fragments are treated as independent messages. Normally, bandwidth being used on one route from sender to receiver in order to successfully deliver the message equals a multiple of the whole message. With fragments, having used bandwidth from various routes for smaller message sizes as compared to the larger message on each route, the potential total bandwidth in the network can be used more effectively.

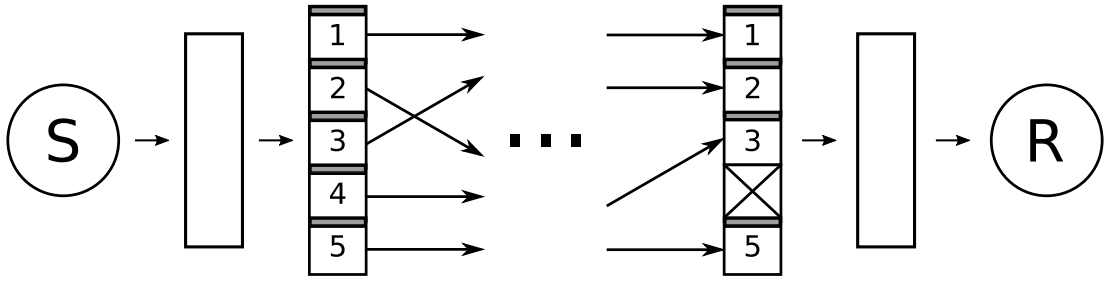


Figure 2. Workflow of fragmentation and network coding process on sender and receiver

Table I
ESTIMATED THRESHOLD

transmit speed [kB/s]	threshold [MB]
250	2.5
500	5
1000	10
2000	20

2) Fragmentation Thresholds: Our fragmentation strategies allow for an improved delivery probability of messages in opportunistic networks. We argue that it should only be applied for messages larger than a given threshold, calculated from the product of average inter-node contact time and average transmit speed. This product resembles the amount of data that, on average, can be successfully transmitted between two nodes upon contact. Therefore, our threshold calculation is independent of other network parameters such as the routing protocol in use. Messages with sizes below this threshold can, on average, be transmitted successfully without fragmentation. Thus, we follow that our fragmentation strategies should be applied for messages with sizes above this threshold and that the threshold should be calculated from measured data about the network. If the network is too heterogeneous for average measurements to be correct, a certain quantile of the measured data can be chosen, as fragmentation still achieves similar delivery probability below the defined threshold. Above the aforementioned threshold, our fragmentation strategies help improve message delivery.

Due to the fact that we try to take advantage of most contacts in the network from a statistical point of view, we anticipate the possibility of successfully exchanging at least one fragment during most contacts. Thus, we estimate the threshold via the 40^{th} percentile $P_{40,\texttt{contacttime}}$ of the average time each contact between nodes last via

$$\texttt{threshold} = P_{40,\texttt{contacttime}} \cdot \texttt{transmitspeed}. \quad (1)$$

With $P_{40,\texttt{contacttime}} = 10\,\mathrm{s}$ in the case of the present simulation we obtain estimated thresholds given in table I.

Additionally, we suggest that there is another, lower threshold for which fragment sizes should not be chosen below this lower threshold. This consideration has two reasons: First, if the fragment size is chosen below the threshold given by eq. (1), a majority of contacts is already sufficient for fragment exchange. This proposition is true, independent of the displacement between threshold and actual chosen fragment size. Second, an increased number of fragments leads to dependence on a higher number of successful transfers of those fragments to the receiving node, which leads to a decrease in delivery probability.

3) Adding Network Coding: We motivated the improvements on multi-path routing through fragmentation of messages in sections II-A and II-E and outlined the applicability in the previous section. This allows for an improved resource usage, but loss of one or more fragments along the paths becomes more likely. To counter this and also allow receiving nodes to assemble the whole message from a sufficient amount of already received fragments, we add network coding as detailed in section II-B to fragments created from one message. Message fragmentation and network coding in combination can benefit from each other, as both strategies each augment the effects of the other.

B. Combined Model

This subsection describes fragmentation and additional network coding in opportunistic networks from a mathematical perspective and shows how one can approach the issue of optimizing delivery probability of messages in opportunistic networks. To understand the order in which the single steps of our model are applied, fig. 2 depicts the fragmentation and network coding on the senders' side, and reconstruction of fragments — as fragment no 4 never arrived — as well as defragmentation on the receivers' side. The present subsection is divided into section IV-B1, where we describe the fundamental idea for fragmentation in our use cases, and section IV-B2, where we depict our forward error correction approach.

1) Fragmentation: For an opportunistic network consisting of n nodes, a node N_s in the network creating a message got $n-1$ possible next nodes to forward this message to. The next node has $n-2$ possible next nodes, and so forth. In each of these cases, one of the next nodes, N_r, is the receiving node of the underlying message. If N_r is the next node to receive the message, the message is successfully delivered and this path is closed. Thus, only the other paths need further consideration and all paths may be closed at any hop count

Table II
SCENARIO SETTINGS

Simulation duration	126 400 s
Warm-up time	40 000 s
World size	4500 m × 3400 m
Number of hosts	80
Movement model	`ShortestPathMapBasedMovement`
Movement speed	0.5 m/s to 1.5 m/s
Router	`SprayAndWaitRouter`, `ProphetRouter` and `EpidemicRouter`
Interface type	`SimpleBroadcastInterface`
Interface speed	250 kB/s to 2 MB/s
Interface range	10 m
Buffer size	1000 GB and 100 MB
Buffer sorting	FIFO, RANDOM, ROUNDROBIN
Message TTL	300 min = 5 h
Event interval	various
Event size	500 kB to 32 MB

$|H| \in \{1, 2, \ldots, n-1\}$. Following, we need to consider every set of possible paths for every possible $|H|$ and arrive at

$$\sum_{i=0}^{n-2} \frac{(n-2)!}{i!} \tag{2}$$

possible successful routes a message may take from start to receiving node. If a message is fragmented, the created fragments may take routes varying from each other. If this is the case, network resources may be exploited better, thus leading to improved network performance.

2) Network Coding: In the case of fragmentation, the successful delivery of a message A is dependent on the delivery of all fragments $A_i, i \in \{1, 2, \ldots, n\}$ the message consists of. This leads to cases, where the message is not successfully delivered because a fraction of those fragments has not been received by the destination node. Thus, the delivery probability might be increased if this dependence does not require the successful delivery of all fragments, but the successful delivery of a share ω of fragments. This is the case in the network coding approach described in section II-B.

If the share ω of fragments got a delivery probability of at least X %, we state that for X % of messages the share ω of fragments is successfully delivered. For this case, one can choose the payload sent in the network coding based approach as

$$\texttt{payload} = \omega \cdot \texttt{messagesize} \tag{3}$$

where

$$\texttt{messagesize} = \texttt{F} \cdot \texttt{fragmentsize} \tag{4}$$

with `F` as the total amount of fragments. The share of redundancy resulting from this is given by $1 - \omega$.

C. Scenario Definition

Parameters of the chosen scenario are shown in table II. Based on the ranking given in [11], we chose the Opportunistic Network Environment simulator (ONE) [12] for our evaluations. We implemented our Fragmentation Strategies as a message gateway conforming to the existing structure of ONE. The chosen simulation scenario configuration resembles nodes moving along streets and paths in the town of Helsinki, as given by the default scenario of ONE. We conducted simulations employing the three given routers referring to Spray&Wait [13], PRoPHET [14] and Epidemic [15] routing each. We chose these routing protocols to show the applicability of our Fragmentation Strategies onto routing protocols from probabilistic and flooding categories. For the available buffer size on simulated nodes, we chose both a way too large buffer size to rule out effects from filled buffers, and a plausible buffer size easily available on todays' mobile devices.

To achieve comparable results from applying our forward error correction approach, we utilize a uniform message generation within our simulations. The event interval for generating messages is given through

$$\frac{\texttt{event size}}{\texttt{dataCreationRate}} \cdot \texttt{number of hosts} \tag{5}$$

with

$$\texttt{dataCreationRate} \stackrel{!}{=} 100\,\text{kB/s} \tag{6}$$

as a result derived from the event size and data creation rate. The event interval describes the interval, after which a message on each node is created. In summary, message creation events are uniformly distributed over time.

Our simulations were computed using OpenJDK 1.8.0_212 on Debian 9 machines running on Intel Xeon E5-2630L v4 and took an average of 2 min per simulation run to complete. For further research and verification, our implementation and simulation scenarios can be found on Github[1].

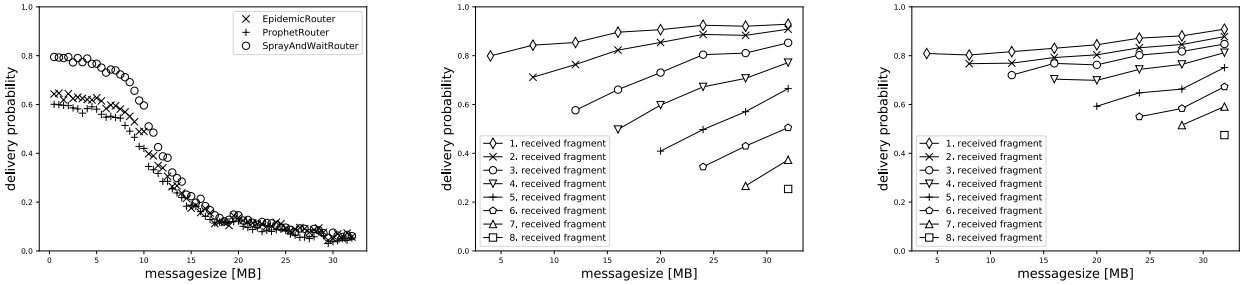
V. EVALUATION

In this section, we evaluate our simulation outcomes and describe the improvements gained by using our Fragmentation Strategies implemented in our Gateway. First, we describe the metrics used in the following evaluations. After detailing evaluations of single observations, we give an overall evaluation of our Fragmentation Strategies in section V-F. Please note that our evaluations focus on several simulation runs we conducted and cannot provide a general evaluation. We argue that our findings are applicable for other scenarios, too, if the requirements stated in section IV are met.

A. Metrics

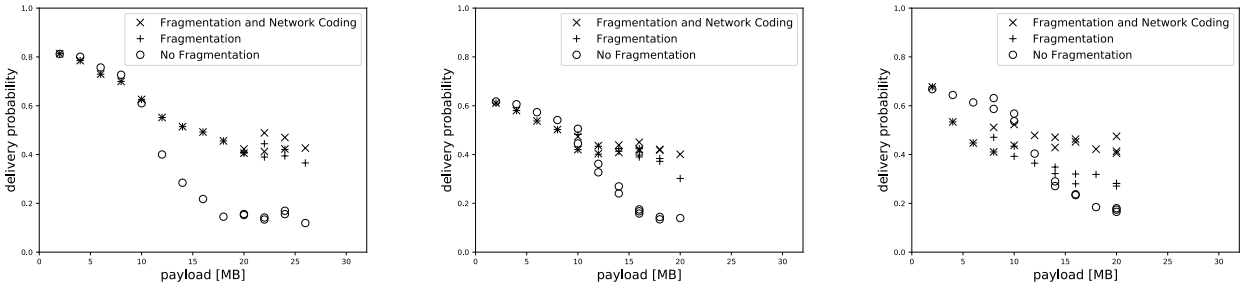
In the following, we explain the metrics used for our evaluations and their relevance to general network performance. These four metrics are common for evaluations using the ONE simulator. Each metric outputs details about one specific network characteristic. While it is favorable to improve one metrics' output, it has to be put into context by also observing the other metrics. An improvement for one metric can possibly impair others. All evaluated metrics are averaged over all measurements taken into account for its calculation.

[1] https://github.com/hhucn/the-one-fragmentation-network-coding



(a) Different routing and message sizes (b) Fragment counts for $1000\,\mathrm{kbit/s}$ (c) Fragment counts for $2000\,\mathrm{kbit/s}$

Figure 3. Observations on different routing protocols and message sizes, and fragments of $4\,\mathrm{MB}$



(a) Spray&Wait (b) PRoPHET (c) Epidemic

Figure 4. Fragmentation Strategies used with different routers and $2\,\mathrm{MB}$ fragments for $1000\,\mathrm{kbit/s}$

1) *Delivery Ratio:* The delivery ratio is the probability of a successful delivery of a message. It is calculated by comparing the number of delivered, i.e. successfully assembled, messages to the number of created message.
2) *Overhead Ratio:* The overhead-ratio represents the efficiency of the network by considering the number of total transmissions and delivered messages. In the different contexts of fragments and messages, its outputs are not directly comparable. We explain the details in the evaluations using overhead ratio.
3) *Hop Count:* While this metric usually gives an insight on how many nodes participated in a successful delivery, this metric is not applicable for our fragmentation strategies. As each message fragment can travel along different paths and thus having different hop counts, an exact output cannot be given. Multiple fragments can travel along the same or different paths, meaning that an average is not appropriate. Therefore, we leave this metric out of our evaluations as its results are not directly comparable.
4) *Latency:* The latency is the time a message needs to reach its destination. In our simulations, it is calculated as the time between message creation and the first time a message can be successfully assembled.

Using these three defined metrics, we can compare simulation outputs and give a reasoning on how our improvements work.

B. Fragment delivery

Before we evaluate the impact of our Fragmentation Strategies on different routing strategies, we detail observations on single fragment delivery. As Fragments form the base of our Fragmentation Strategies, it is important to note network performance differences for different fragment sizes.

We evaluated the performance of three selected routing protocols for different message sizes, representing different fragment sizes for our evaluation. Figure 3a shows the delivery probabilities for message sizes between $500\,\mathrm{kB}$ to $32\,\mathrm{MB}$ using different routers with a transmit speed of $1000\,\mathrm{kbit/s}$. All three routing protocols show a gradual decline in delivery probability the higher the message size. The same simulations using different transmit speeds show similar results, with the gradual decline shifted along the x-axis depending on transmit speed increase or decrease.

Figures 3b and 3c show the delivery probabilities of different fragments of one message using Spray&Wait. Displayed are fragments of $4\,\mathrm{MB}$ each, therefore every multiple of four on the x-axis represents one message consisting of at least one fragment. Then, from top to bottom, each point resembles the average delivery probability for the first, second, and so on fragment received from this message. The order is given by the time the fragment was received, not sent. According to the data generation formula given in eqs. (5) and (6), we maintain an average data creation rate of $100\,\mathrm{kbit/s}$ for all message sizes. We observe that for larger messages, while the same amount of data is generated within the network and all

fragments have the same size, the fragments arriving first have a higher delivery probability than for smaller messages. For explaining this effect, we argue that the more fragments for one message are underway, the more likely it is for one of these fragments to arrive at their destination.

Comparing overhead ratio and average latency for fragments to the outcomes for messages is not useful for this evaluation. We inspect single fragments and not wholly delivered messages, therefore one can identify single fragments as smaller messages. This leads to equal overhead ratio and average delay as for other messages.

C. Message delivery

Figures 4a to 4c show delivery probabilities of various message sizes for our Fragmentation Strategies and no Fragmentation at all. The fragment size is $2\,\mathrm{MB}$ and transmit speed is $1000\,\mathrm{kbit/s}$.

We observe that for the first few message sizes, fragmentation leads to a deteriorated delivery probability. Our evaluations regarding this behavior is given in section V-D, in this section we focus on the improvements gained when utilizing our Fragmentation Strategies.

Starting from message sizes above $10\,\mathrm{MB}$, average delivery probabilities for our Fragmentation Strategies lie mostly or always above the average delivery probability for when using no fragmentation. Spray&Wait is able to achieve the highest average delivery probabilities when combined with our Fragmentation Strategies. PRoPHET in combination with our Fragmentation Strategies also delivers higher average delivery probabilities compared to using no Fragmentation, while Epidemic shows the least improvements when using fragmentation. Still, for larger messages, the superior average delivery probabilities of Epidemic combined with fragmentation is clearly visible. To explain these different measurements, one has to look at the functionality behind those routing protocols. For Epidemic, the routing tries to send all fragments over all possible paths throughout the network by exchanging all messages not yet received upon each node encounter. One would think that this allows for an overall improved delivery, as more paths throughout the network could be utilized for our Fragmentation Strategies. As participating nodes each get copies of all possible fragments, their buffers fill up quicker than with the other routing protocols, leading to greater overhead ratio and impaired delivery probability. Spray&Wait and PRoPHET use different message replication and place lesser load on participating nodes. PRoPHET uses a probabilistic decision based on node encounter history for message replication. If encountered nodes have a higher probability of meeting other nodes that can forward a given message to its destination (including the receiving node itself), the message is replicated and forwarded. The amount of potential paths throughout the network is thereby reduced, resulting in an reduced improvement of delivery probability through our Fragment Strategies. Spray&Wait, on the other hand, is a flooding based routing protocol like Epidemic. While Epidemic sets no constraints for flooding except for sending a message to one node only once at most, Spray&Wait limits the amount of message replications in the network. Following this behavior only the maximum number of replications is set, enabling Spray&Wait to utilize all possible ways throughout the network until the maximum number of replications has been reached.

Another important observation is the largest message size used in our evaluations for all three routing protocols. For Spray&Wait, it is $26\,\mathrm{MB}$, while for PRoPHET and Epidemic it ends at $20\,\mathrm{MB}$. This due to our choice of calculation for the number of fragments as detailed in section IV-B. PRoPHET and Epidemic achieve general lower delivery probabilities than Spray&Wait in our simulations. This is reflected in the number of $2\,\mathrm{MB}$ fragments possibly being send out with a high enough attempted message delivery probability.

The overhead ratio seen per-fragment is comparable to the overhead ratio per-message. In absolute numbers, the amount of total fragment overhead copies per original message is well above the amount of overhead copies. This follows because every fragment has multiple copies and every original message has multiple fragments, while for the per-message overhead, only copies of the whole message are counted. Once it is broken down to the amount of bytes needed for copies, the overhead ratios equalize again. If Network Coding is used in addition, the overhead in total bytes increases by the amount of additional error correction bytes times the overhead ratio.

For the transmission latency, simulations using our Fragmentation Strategies tend to have a higher average latency. This is an effect of the higher delivery probability: As more messages can be successfully delivered by assembling fragments on the receiving side, fragments taking longer paths through the network delay the successful delivery of the whole original message. For message transfers without our Fragmentation Strategies, direct deliveries are more likely to occur, lowering the average latency by a great amount.

D. Fragmentation Thresholds

Left out from the evaluation conducted in the previous section, we focus on the delivery probabilities depicted in fig. 4 for message sizes below $10\,\mathrm{MB}$. Inspecting the Fragmentation Threshold as given in section IV-A2, we can validate the calculated value of $10\,\mathrm{MB}$ from eq. (1). Delivery Probabilities for messages below this Threshold are higher for regular message sending than for our Fragmentation Strategies. The average node contact time, and thereby also the 40^{th} percentile of those, is sufficient for transmitting the whole message at once. This allows the complete message to be delivered faster, as the receiving node does not have to wait for further fragments to arrive. Additionally, the delivery probability relies on the single message itself instead of the combined probabilities of several fragments.

For messages larger than the Threshold, our Fragmentation Strategies enable a higher delivery probability, with additional network coding allowing for a greater delivery probability than fragmentation alone in some cases. There is only a slight improvement between both Fragmentation Strategies, as for

network coding additional error correction data has to be sent along the actual payload, resulting in an even higher network load with better performance regarding message delivery.

E. Influence of Buffer properties

In section II, we emphasized that a message routed through an opportunistic network by sending replications to different participating nodes can take various distinct paths throughout the network. We have also shown the influence of different Buffer Management Strategies on the set of distinct paths a message can take. While utilizing these distinct paths is one of the fundamental observations our work is based on, our evaluations also show that our Fragmentation Strategies work independently of the applied Buffer Management Strategy if it follows a uniform distribution. We evaluated our Fragmentation Strategies with the three Buffer Management Strategies listed in table II. All simulation outcomes show similar results, with minor differences due to a different message ordering in the buffers. We argue, that all three Buffer Management Strategies lead to a uniformly distributed selection of messages, allowing our Fragmentation Strategies to utilize multiple distinct paths through the network in an analogical manner. For other buffer management strategies, the level of improvement depends on how fair the selection of buffer entries works out.

Additionally, our Fragmentation Strategies are not influenced by the given buffer size from a reasonable buffer size onwards. By reasonable, we expect a buffer to be able to hold at least one complete message, while multiple are preferred. As fragment sizes are smaller than message sizes, we can fit more fragments than messages into a buffer, also meaning that fragments from many different messages all can reside in the same buffer. Combined with transmitting fragments over various paths through the network, this allows for an improved message probability.

F. Combined Evaluation

The evaluations of simulations we conducted support our proposed Fragmentation Strategies. Metric outcomes show that, if applied within the given requirements, our Fragmentation Strategies enable regular opportunistic routing protocols to achieve a higher delivery probability. For the defined Fragmentation Threshold, we are also able to show that it holds independently of the employed routing protocol, albeit the improvement gained depends on the routing protocol.

VI. Conclusion

In this work we motivated the applicability and advantages of Fragmentation and Network Coding for opportunistic networks. We introduced two Fragmentation Strategies utilizing fragmentation by itself and fragmentation and Network Coding combined. By defining a Fragmentation Threshold, we give a range of application where our Fragmentation Strategies are able to improve message delivery. Using the Opportunistic Network Environment simulator ONE, we were able to show paradigmatic for three routing protocols that our Fragmentation Strategies allow each routing protocol to achieve significantly higher message delivery probabilities than without fragmentation.

A. Future Work

We used statistically determined measurements for employing our Fragmentation Strategies. Opportunistic Networks are highly dynamic and the environment one node is located in can notably change throughout its participation in the network. Therefore, it might be interesting to investigate sampling network environment measurements and vary the fragmentation parameters accordingly.

References

[1] L. Pelusi, A. Passarella, and M. Conti, "Opportunistic networking: data forwarding in disconnected mobile ad hoc networks," *IEEE communications Magazine*, vol. 44, no. 11, pp. 134–141, 2006.

[2] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Médard, and J. Crowcroft, "XORs in the air: practical wireless network coding," *IEEE/ACM Transactions on Networking (ToN)*, vol. 16, no. 3, pp. 497–510, 2008.

[3] S. Sati, C. Probst, and K. Graffi, "Analysis of buffer management policies for opportunistic networks," in *2016 25th International Conference on Computer Communication and Networks (ICCCN)*. IEEE, 2016, pp. 1–8.

[4] J. Postel *et al.*, "RFC 791: Internet protocol," Tech. Rep., 1981.

[5] R. Ahlswede, N. Cai, S.-Y. Li, and R. W. Yeung, "Network information flow," *IEEE Transactions on information theory*, vol. 46, no. 4, pp. 1204–1216, 2000.

[6] M. Pitkanen, A. Keränen, and J. Ott, "Message fragmentation in opportunistic DTNs," in *2008 International Symposium on a World of Wireless, Mobile and Multimedia Networks*. IEEE, 2008, pp. 1–7.

[7] L. Feng, Y. Zhang, and H. Li, "Large file transmission using self-adaptive data fragmentation in opportunistic networks," in *2015 Fifth International Conference on Communication Systems and Network Technologies*. IEEE, 2015, pp. 1051–1055.

[8] Y. Wang, S. Jain, M. Martonosi, and K. Fall, "Erasure-coding based routing for opportunistic networks," in *Proceedings of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking*. ACM, 2005, pp. 229–236.

[9] L.-J. Chen, C.-H. Yu, T. Sun, Y.-C. Chen, and H.-h. Chu, "A hybrid routing approach for opportunistic networks," in *Proceedings of the 2006 SIGCOMM workshop on Challenged networks*. ACM, 2006, pp. 213–220.

[10] J. Widmer and J.-Y. Le Boudec, "Network coding for efficient communication in extreme networks," in *Proceedings of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking*. ACM, 2005, pp. 284–291.

[11] 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 *2016 25th International Conference on Computer Communication and Networks (ICCCN)*. IEEE, 2016, pp. 1–7.

[12] A. Keränen, J. Ott, and T. Kärkkäinen, "The ONE Simulator for DTN Protocol Evaluation," in *SIMUTools '09: Proceedings of the 2nd International Conference on Simulation Tools and Techniques*. New York, NY, USA: ICST, 2009.

[13] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Spray and wait: an efficient routing scheme for intermittently connected mobile networks," in *Proceedings of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking*. ACM, 2005, pp. 252–259.

[14] A. Lindgren, A. Doria, and O. Schelen, "Probabilistic routing in intermittently connected networks," in *International Workshop on Service Assurance with Partial and Intermittent Resources*. Springer, 2004, pp. 239–254.

[15] A. Vahdat, D. Becker *et al.*, "Epidemic routing for partially connected ad hoc networks," Tech. Rep., 2000.