Patrick Lieser, Nils Richerzhagen, Simon Luser, Björn Richerzhagen, Ralf Steinmetz. *Understanding the Impact of Message Prioritization in Post-Disaster Ad Hoc Networks (accepted for publication).* In Proc. Conference on Networked Systems (NetSys), IEEE, 2019. ISBN:

Understanding the Impact of Message Prioritization in Post-Disaster Ad Hoc Networks

Patrick Lieser, Nils Richerzhagen, Simon Luser, Björn Richerzhagen, Ralf Steinmetz
Multimedia Communications Lab (KOM), Technische Universität Darmstadt, Germany
{patrick.lieser|nils.richerzhagen|simon.luser|bjoern.richerzhagen|ralf.steinmetz}@kom.tu-darmstadt.de

***Abstract*—In the aftermath of disasters, access to communication infrastructure is often impaired or fully unavailable. Smartphone-based ad hoc networks can be utilized to re-enable basic communication services and foster coordination and self-help capabilities of those affected. However, their capacity is limited as they need to operate in a disruption-tolerant fashion. At the same time, the communication demand increases significantly after a disaster, potentially overloading the ad hoc network and requiring message prioritization mechanisms.**

In this work, we contribute insights into the communication behavior and resource demand in a post-disaster ad hoc network based on a large field trial and a survey of related works. We identify—potentially undesired—interactions between delay-tolerant networks (DTNs) with message prioritization and the specific dynamics of a disaster scenario. To study these interactions in greater detail, we propose a generic architecture for the evaluation of prioritized DTNs in disaster scenarios. We identify key issues w.r.t. static and adaptive prioritization approaches based on a proof-of-concept evaluation and outline directions for future research on prioritization in DTNs.

***Index Terms*—Delay-tolerant Networks, Disaster Communication, Adaptive Prioritization**

I. INTRODUCTION

Due to climate change, the scope and scale of natural disasters increased in the last few years [12]. Furthermore, the number of man-made disasters such as power outages rose as well [17]. Managing such disasters becomes increasingly challenging due to the rise of population densities in urban areas [22] and an interconnected global economy. Especially for coordination and rescue efforts in disasters, information and communication technology (ICT) is essential. After disasters the infrastructure for communication is either partially or completely destroyed. The available bandwidth is only a fraction of that available right before the disaster. For a short moment right after the incident, the communication demand also drops, as people are focusing on getting themselves in safety. However, shortly afterwards the communication demand rises significantly as illustrated in Figure 1. During this phase, people ask for help, coordinate rescue efforts, and try to contact their friends and family, leading to a communication demand even higher than right before the disaster.

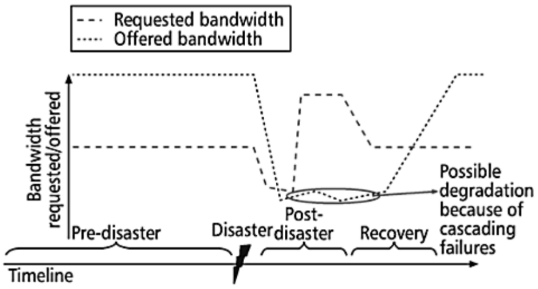

Figure 1: Communication demand during a disaster [10].

This leads to a huge gap between network resource demand and available communication supply [5], [9]. To address this issue, our previous research has successfully shown, that smartphone-based ad hoc networks can be used to provide the affected population with basic communication capabilities [1], without the need for any central infrastructure. We enabled ad hoc functionality of ordinary phones and designed and implemented an application for responsive emergency communication systems, providing people with the ability to, e.g., ask for help, share resources, and other relevant services [14]. The application relies on an underlying ad hoc communication service for delay-tolerant networks (DTNs). DTNs distribute data in a store-carry-forward fashion. Data is stored by the mobile network nodes, and data is distributed whenever two nodes are within reach of each other. To achieve good data dissemination over larger areas, flooding based routing protocols are often used. The necessity of data retransmissions, that comes along with flooding-based DTNs, further increases the network resource demand.

To test our application and the underlying DTN protocol under real-world conditions, we conducted a large field test with 125 participants during a scripted disaster event [1]. During that field test we recorded user mobility, communication characteristics, and the interactions with the different services offered by the application. Similar to [18], we observed that smartphone based ad hoc networks cannot fully replace the infrastructure based communication and can only provide limited means of communication. Data prioritization within such a network is essential to cope with the restricted communication capabilities and the increased demand for communication.

However, prioritization is a challenging issue, given that the service usage changes throughout the course of the post-disaster scenario. As a consequence, the relevance of a specific message determined by its type, content, and the user's context further changes over time and is hard to quantify. Prioritization of relevant data always results in penalizing other data. Especially in a post-disaster scenario, it is important to understand the potential impact of prioritization on the performance of the network and its ability to distribute relevant information in a timely manner.

In this paper, we identify interactions between message prioritization in DTNs and the specific dynamics of a disaster scenario based on insights from our conducted field trial. Based on the discussion of the scenario in Section II, we formulate three hypotheses for prioritization in post-disaster DTNs. We further elaborate on related work on prioritization and post-disaster communication in Section III. To study the interactions between prioritization and DTN in greater detail, we propose a generic architecture for the evaluation of prioritized DTNs in disaster scenarios. We discuss this architecture and its potential to further examine our hypotheses in Section IV. Section V contains a proof-of-concept evaluation of our approach, highlighting benefits and drawbacks of three prioritization approaches being compared within our architecture. Our findings motivate the need for a more adaptive, scenario-specific approach towards prioritization in resource-constrained post-disaster communication networks. The paper is concluded in Section VI.

II. ANALYSIS OF USER AND NETWORK BEHAVIOR

To understand the potential interactions between message prioritization in resource-constrained DTNs and the application-specific usage patterns, as well as the resulting implications on the performance of the ad hoc network, we first discuss insights obtained during our large-scale field test [1]. In the following, we discuss relevant findings from the field test and formulate hypotheses with respect to potential interactions. For an in-depth discussion of the field test setup and measurement methodology, we refer to [1].

One main challenge when it comes to smartphone-based DTNs in post-disaster scenarios is the limited bandwidth of the network. Regardless of the utilized communication interface—e.g., Bluetooth or WiFi—the offered bandwidth of the overall network over multiple hops is not sufficient to cater for the increased demand after a disaster due to short interconnection times and the store-carry-forward nature of the DTN. The amount of stored and carried data messages, as well as the amount of their duplicates generated in the network over time, is highly dependent on the message lifetime. This parameter is configured either within the DTN protocol or provided by the application when generating new messages. Figure 2 shows the message propagation delay of the conducted field test with a limited message lifetime of 60 minutes. The Figure shows that, on the one hand, newly generated messages are propagated almost immediately in the close neighborhood of their originator. However, on the other hand, messages still reached additional—and potentially highly relevant—nodes until the very end of their lifetime due to network fragmentation and node mobility. This shows that a wide message propagation in DTNs requires sufficiently large message lifetimes, especially considering mobility in a post-disaster scenario. This has a direct impact on the resulting load in the network caused by message duplicates of the huge amount of stored and carried messages.

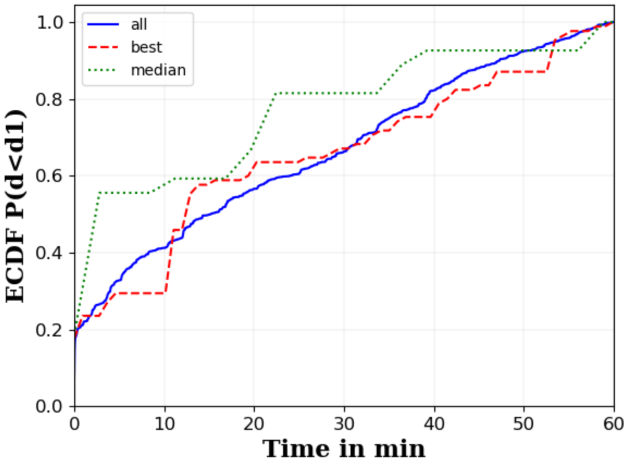

Figure 2: Absolute field test message delay of 1835 unique messages with a lifetime of 60 minutes [1].

To reduce the load on the network while still ensuring that messages reach a sufficiently large fraction of the network, prioritization can be used to favor specific messages and postponing or completely dropping less important ones. Previous studies of social media postings and tweets during and after a disaster showed that messages can be grouped into distinct categories [11], [22]. Figure 3 shows the categories and their message volume during the field test.

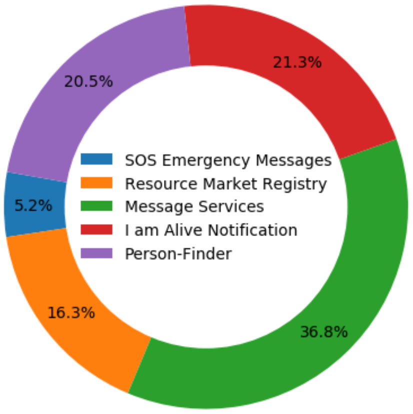

Figure 3: Field test application service usage [1].

The amount of messages within a certain category and their importance changes during the course of a disaster, as discussed in [21], [22] and visualized in Figure 4. In the aftermath of a disaster, for example, emergency messages are considered to be most relevant. After some time, safety-related information and guidance information for the remaining population becomes more important and takes over a larger fraction of the load in the network. It is important to note that emergency messages are still created at later stages—albeit at lower frequency—when the network is already dealing with huge numbers of messages of other types.

Consequently, to ensure the delivery of highly relevant messages, they need to be prioritized in the network. One way to achieve this is to utilize static prioritization based on previously known message types. In our previous example one could, e.g., always prioritize emergency messages over messages of other types. While this property is desired, there arise some implications from the store-carry-forward nature of DTNs and the potentially long message lifetimes. As illustrated in Figure 4 the currently trending message type changes over time. In this case, older messages of a highly prioritized type might have already been delivered to a large fraction of nodes in the network while their duplicates are still blocking new messages with lower priority from being forwarded. Therefore, more flexible and adaptive prioritization approaches are needed.

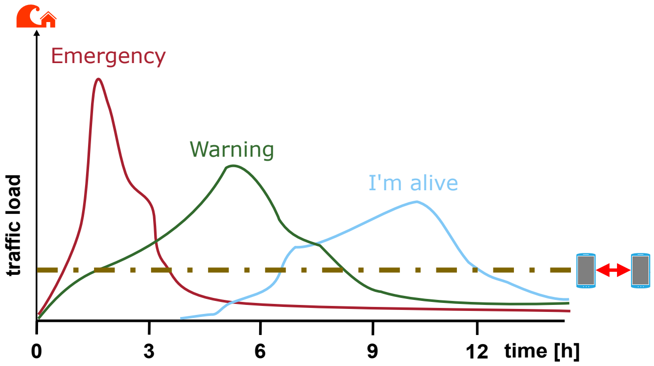

Figure 4: Illustration of time-dependent load during a disaster (based on [9]) separated by message type.

Messages need to be prioritized under consideration of environmental and contextual changes, e. g., the message content, the transmission history, location of the user, or even the health status of a person [13]. Due to the lack of a central coordination unit, prioritization within the DTN needs to be performed in a fully decentralized fashion, knowing that a global consensus is not achievable. Understanding the potential impact of the respective prioritization approach on the performance of the DTN in disaster scenarios is a crucial requirement for the design of adaptive algorithms. In the following, we formulate three hypotheses on the expected impact and inter-dependencies of prioritization in DTNs. The hypotheses motivate the design of our architecture detailed in Section IV and are later revisited in the proof-of-concept evaluation of the resulting prioritized DTN approaches.

H1: *Static prioritization can lead to the delay of potentially urgent data and may instead amplify the delivery of outdated—or duplicate—information.* As presented in Figure 4 based on [9] the frequency and importance of a message type can change over type. In networks with insufficient bandwidth, static prioritization is unable to assign sufficient resources to messages with lower priority. Instead, available resources are largely consumed by duplicates and re-transmissions of messages with higher priority, decreasing the overall performance of the network significantly.

H2: *Adaptive prioritization can prevent relevant—and especially urgent—data from spreading in the network.* Adaptive prioritization mechanisms can change the prioritization order of different message types, for example, by detecting message type trends in the network. Under the assumption, that the frequency of a specific message type also reflects the relevance of the data type, such adaptive prioritization mechanisms have the advantage of dynamically prioritizing data and, thereby, increasing the propagation and decreasing the dissemination delay of the according type. Consequently, this can result in prevailing other data, that is not (yet) trending in the overall network. The length of the window used to detect message trends highly correlates with the ability to detect changes (long- and short-term) in the network load and needs to be carefully tuned.

H3: *Prioritization needs to consider a type-, content-, or even context-specific validity time of messages.* The real value of a message from a sender's and receiver's point of view strongly depends on the respective person's context, the content of the message, and—equally important—the age of the respective information. If the outdated information is still propagated due to a high static priority being assigned to the message, this consumes resources that could otherwise be used to transmit urgent, fresh messages of lower priority but higher perceived value. Especially in disaster scenarios (as considered in our field trial), the spatial distance between sender and receiver of a message can further determine the value of a given type of information. Consider, for example, the ability to offer and request resources as proposed in [1]: the relevance of a respective offer depends on the receiver's ability and willingness to travel the distance and to risk being too late to consume a resource that is located farther away. A more elaborate prioritization algorithm should take the respective contextual information into account.

Before discussing our proposed architecture to asses the aforementioned hypotheses, we discuss relevant related work in the following section.

III. RELATED WORK

To make prioritization possible, messages need to have some sort of message specific attributes such as size, type (e. g. emergency, warning) or additional context information. For our work, we assume that these attributes are preassigned through a mobile application with different services running on the mobile nodes. If this is not possible, because i. e. an already existing system wants to be utilized (for example a social network as Twitter), natural language processing methods can be applied [2]. They used a Naive Bayes Classifier to categorize WhatsApp messages that were exchanged between first responder teams during post-disaster relief operations after a major earthquake in Nepal in 2015. They classified the messages in predefined types and prioritized accordingly to the situation. Messages which were related to resource requirements were assigned with a high priority, while messages with sentimental content were prioritized low. Afterward, a static priority has been assigned to each of them and forwarded inside the network correspondingly. Related to the disaster scenario, extensive categorization of message content has been done in the *CrisisLex* [20]. It is a lexicon with crisis-related terms that frequently appear in relevant messages posted during different types of crisis situations. These terms can then be used to perform message typecasting during a disaster. Categorizing messages with various machine learning techniques is a current research field [4] and also newer approaches such as Deep Learning [19] has been applied. Techniques like this can also be applied to multimedia content such as picture or videos [6], so that the user's phone is capable of detecting critical content such as fires or road blocks.

Besides the message content, the context of the user can be used for data prioritization. Context information could be vital signs of a person [7], the battery level of the phone or the location [15]. In previous work [13] we showed, that it is even possible to detect different disaster-related activities of a user, like crawling on the floor or walking with an injured leg, solely based on the user's smartphone or smart-watch sensor readings. For our work, the important aspect is the opportunity to apply these algorithms beforehand to enable prioritization based on the message content and the user's context characteristics.

Luqman et al. [16] introduced a framework called "TRIAGE" which prioritizes messages based on different content and context information in case of an emergency. *TRIAGE* is also monitoring network congestion to determine the necessity of data prioritization. Every message is routed through a central component (Emergency Operations Center (EOC)) that coordinates the data prioritization if the network is congested. To reduce traffic, the EOC node can determine critical, disaster dependent information. Since all data is collected by the EOC, it has global knowledge and can, therefore, prioritize data very efficiently. Compared to our considered scenario and the utilization of mobile ad hoc delay-tolerant networks, the lack of a central coordination unit makes the *TRIAGE* framework not applicable.

IV. ARCHITECTURE FOR PRIORITIZED POST-DISASTER COMMUNICATION

As motivated in the previous section, scenario-specific message types are either pre-determined by the respective application or can be derived based on, e.g., textual analysis of the message content. This application-specific information can be used in conjunction with data gathered by the DTN approach itself, such as neighborhoods or hop counters, to prioritize messages if the network is overloaded. Our architecture proposes a generic message storage approach for DTNs, allowing us to separate the workings of a specific DTN protocol from the prioritization of messages. Therefore, prioritization is realized as a function operating solely on the message storage, as illustrated in Figure 5 and explained in more detail in the following. This separation and interaction over the shared message storage allows us to later evaluate the impact of different prioritization algorithms combined with the chosen DTN approach on the performance of the communication network.

A. Message Storage and Generalized DTN Operation

DTN protocols rely on the store-carry-forward-principle, requiring nodes to store messages for later propagation. Depending on the utilized DTN protocol, the respective messages are annotated with additional meta information, such as the hop count, a duplicate counter, or the timestamp the message was last sent by the node. We refer to this as *DTN-specific* meta information that is updated by the DTN protocol upon reception or transmission of a message. This process is illustrated in the lower part of Figure 5: whenever a new message is to be sent from the application or received for the first time from another node, it is inserted into the storage. In case a received message is already contained in the storage, the node simply updates the associated *DTN-specific* meta information.

In typical DTN approaches, the storage operates similar to a stack, with new messages being added on top. Whenever the protocol is to forward messages, it picks the topmost N messages, with N referring to the current capacity of the network. In basic DTN protocols, this process is executed periodically, leading to the N most recent messages being broadcasted to nearby nodes. Often, this involves a probabilistic filtering of messages to further reduce network load [3], [25]. More elaborate protocols rely on the exchange of message availability prior to the forwarding phase to optimize the transmission process [26], using unicast for the actual data exchange. Instead of selecting the first N messages, those messages that are already known to the counterpart are skipped. To cater for these different types of DTN protocols, we generalize the forwarding phase as an iteration procedure over the (sorted) message storage selecting up to N messages in total. During an iteration, messages can be skipped based on *DTN-specific* meta information or, e.g., a probabilistic function.

The DTN protocol interacts with the message storage through two basic functions as previously described: (i) inserting new messages into the storage and (ii) updating the associated DTN-specific meta information of a message. During the forward phase, the protocol can iterate over the storage, accessing the message itself and associated *DTN-specific* meta information. Thereby, the process of message prioritization is modeled as a re-ordering of the message storage. To limit the amount of messages being stored on a node, DTN protocols rely on a Time-to-Live (TTL) as one additional meta information. If a message is older than its TTL, it is removed from the message storage during an iteration. However, as discussed in Section II, a reasonable TTL is highly dependent on the scenario and application and might even differ depending on the content and type of a message. Therefore, we argue that the removal of messages from the storage should instead be part of the prioritization mechanism, as discussed in the following.

B. Application- and Scenario-specific Prioritization

Our proposed architecture separates the DTN protocol itself from the prioritization algorithm through the message storage. To this end, a prioritization algorithm can (i) reorder elements in the storage and (ii) remove elements from the storage.

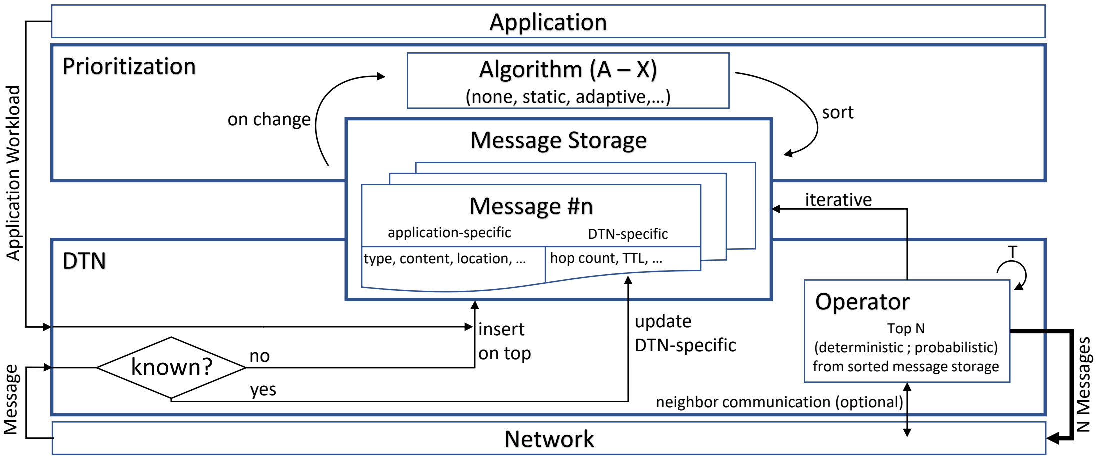

Figure 5: Architecture for prioritized post-disaster communication

The most important aspect, however, is access to *application-specific* meta information from within the prioritization algorithm. Such information is either contained within the application-defined structure of the message (i.e., fields) or can be derived by executing higher level operations on the contents of the message, such as text classification [4] as described in Section III. This requires prior knowledge of how the application creates and structures messages, which we believe to be a substantial requirement for the design of efficient prioritization algorithms in resource constrained scenarios as discussed in Section II.

The respective meta information may include, for example, a message type, an application-defined TTL, the user's location, the device's battery capacity, or the intended set of recipients. The combination of *DTN-specific* and *application-specific* meta information can be utilized by different prioritization algorithms to update the message storage. Thereby, algorithms can realize an application- or scenario-specific prioritization order. In the following, we introduce three examples to reorder the data that mimics the behavior of currently utilized prioritization approaches.

One possibility is not to prioritize at all. As already mentioned in the previous section, in this case, the prioritization only has the basic task of periodically removing messages with expired TTL from the message storage. This basic functionality is contained in every prioritization algorithm, as it prevents the ongoing dissemination of outdated messages. As briefly mentioned beforehand, the TTL can either be set directly within the DTN protocol irregardless of message type or content, or it might be part of the *application-specific* meta information. In our evaluation, we consider the TTL to be set within the DTN protocol.

The second example is a static prioritization that considers the message type included in the *application-specific* meta information. This algorithm relies on a predefined order of message types. In case of the considered post-disaster scenario, this order can be defined by disaster relief organizations [14], for example. Every time a new message is inserted into the message storage, the static prioritization reorders the messages according to the predefined prioritization order. Within one message type, messages are sorted by age, with newest messages on top. The age of a message is assumed to be provided as a *DTN-specific* meta information as either the hop count or an absolute timestamp.

The last example is an adaptive prioritization algorithm. Whenever the message storage is updated, the algorithm ranks message types according to their frequency in the storage. The highest priority is allocated to the most prominent message type. The algorithm considers the number of message of a given type that were received within a defined time period (window). To this end, a *DTN-specific* meta information containing the timestamp of the last reception of each message is utilized. If for two or more types have the same prioritization rank, the algorithm falls back to the aforementioned static prioritization. Thereby, the algorithm tries to cater for the change in message frequency and importance over time as motivated in Section II. To ensure that upcoming trends are detected by other nodes, the adaptive prioritization propagates locally created messages at least once irregardless of their type.

In the following section, we discuss a proof-of-concept evaluation of our framework, highlighting the impact of prioritization on key performance metrics in a DTN with varying information relevance.

V. EVALUATION

The goal of the evaluation is to assess the impact of data prioritization in DTNs. Our architecture can be used to observe the interactions between DTNs with data prioritization and the specific dynamics of a disaster scenario. Our prototype is realized within the Simonstrator framework [23], relying on the IEEE 802.11g model of ns-3 [8] for ad hoc Wi-Fi communication between mobile nodes.

A. Evaluation Setup

We simulate 100 mobile nodes on an area of 2x2 km^2 relying on real-world map data from OpenStreetMap. Node movement is restricted to walkways, taking obstacles such as buildings into account. Nodes move with a speed between 1.5 and 2.5 m/s and randomly select Points of Interest contained in the map—e.g., market places or public parks—as their next target destination, as presented in [24]. We configured the damping factor of the Wi-Fi model such that the maximum communication range of a broadcast is 100 m, with the effective communication range in dense scenarios being lower as determined by the 802.11 MAC model [8]. We used *Epidemic Routing* [26] based on the exchange of availability vectors as DTN protocol in our simulations. Only messages that are not yet known to the receiver are actually transmitted. Additionally, unicast is used during the transmission, leading to a more efficient channel usage due to better modulation schemes compared to broadcast. The protocol is configured with a beaconing interval of 15 s for the neighbor detection.

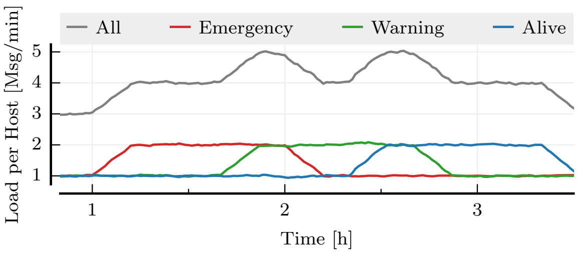

Figure 6: Workload distribution for different message types for one host over time.

We integrate three prioritization approaches: *None*, *Static*, and *Adaptive* with a window size of 20 minutes, as described in Section IV-B. The capacity of the DTN is limited in terms of the number of messages each host is allowed to send in one minute, the storage size on each node is not limited. Thereby, we can directly control the resource scarcity (and, thus, the influence of prioritization) by setting the amount of messages being generated in one minute. For simplicity, we assume that messages have the same size—however, the size of a message is another factor one could also explicitly consider during prioritization at a later stage. For our evaluation, each node can send 10 messages every 60 seconds, with nodes generating messages of a specific type following the distribution shown in Figure 6. By varying the volume of each message type (*Emergency*, *Warning* and *Alive*) over time, we capture a fundamental property of post-disaster communication as discussed in Section II. The TTL is set to 12 minutes irregardless of the message type.

B. Performance of the Resource-constrained Network

Prioritization is introduced to ensure message delivery in a resource-constrained and overloaded network. Therefore, we first consider the recall of messages as an indicator of the performance of message delivery under such conditions. The recall is defined as the fraction of reached receivers and interested receivers. In our evaluation, all 100 nodes in the network should receive each message. Figure 7 shows the recall achieved for different message types and prioritization algorithms. Boxes show the 25th and 75th percentile, with whiskers indicating the 2.5th and 97.5th percentile of all data points. The circle next to each box indicates the mean. As expected, if no prioritization occurs (*None*), the recall does not differ significantly for individual message types. On average, approximately 65 % of all nodes are reached, with more than 75 % of all messages reaching at least half the population. At this point, the network becomes saturated and nodes are less likely to forward older messages, as new (locally created) messages are inserted at the top of their local queue. This limits the spread of messages in the network and, thus, the achievable recall.

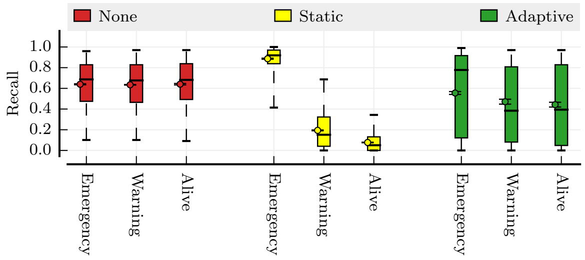

Figure 7: Message Recall.

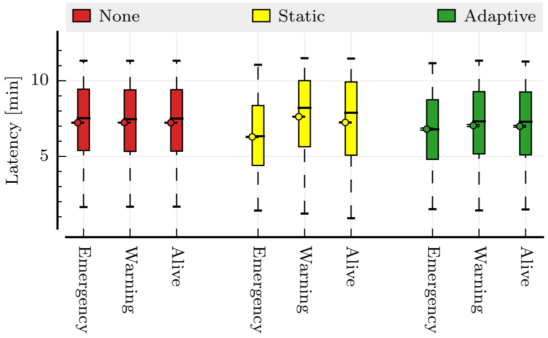

Figure 8: Message Latency.

The impact of static prioritization is severe: as expected, *emergency* messages are favored over other message types, counteracting the saturation effect described previously. In this case, the mean recall for highly prioritized messages reaches 0.9, with more than 75 % of messages reaching at least 80 % of all nodes. However, this has a significant impact on the recall for other message types: messages with lower priority only achieve a recall of below 0.2 on average. The respective information cannot spread in the network, as all available resources are occupied.

The adaptive approach achieves something in-between, with highly skewed distributions. This is because priorities change over time, a fact that is not observable in the box plot. The exact behavior is discussed in more detail in the following sections.

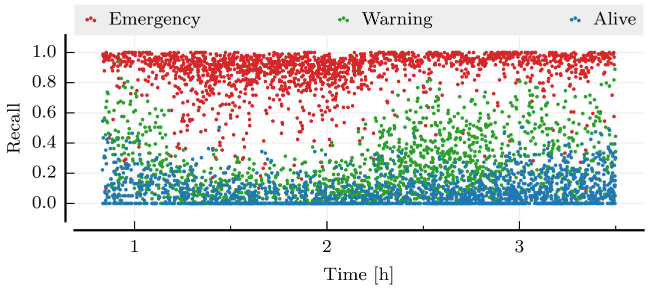

Figure 9: Recall over time with static prioritization.

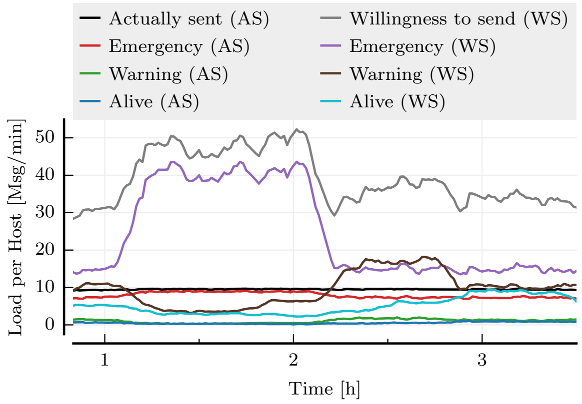

Figure 10: Load over time with static prioritization.

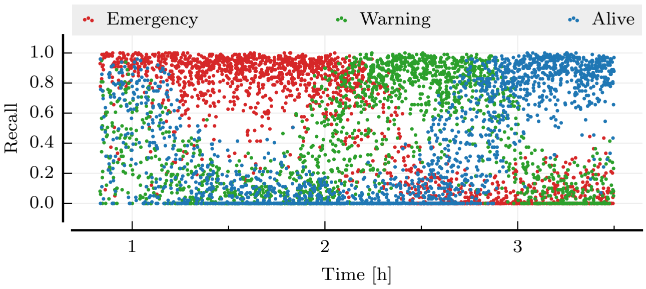

Figure 11: Recall over time with adaptive prioritization.

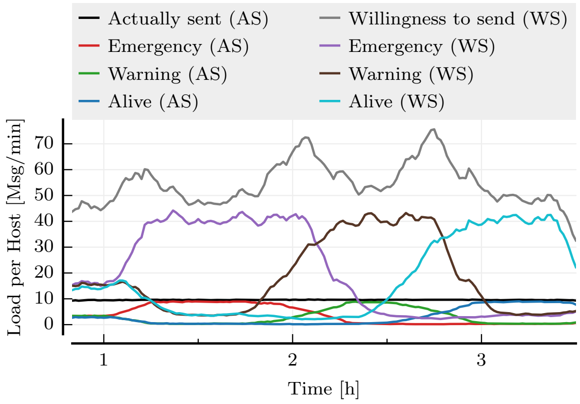

Figure 12: Load over time with adaptive prioritization.

Regarding the dissemination latency as shown in 8, we can observe that message dissemination in some cases requires the full TTL, supporting our analysis in Section II. Other than that, the prioritization only has a minor impact on the latency—only static prioritization slightly reduces the latency for messages with high priority. This is expected, as the latency can only be analyzed in a meaningful fashion when also considering the recall. However, one can still note that the respective algorithms behave as expected: if messages are not prioritized, there is no difference in observed latency for individual messages.

C. *Amplification of Outdated or Duplicate Information*

As motivated in Section II, we expect static prioritization to favor duplicates and outdated information of a message type with high priority over newer and potentially more relevant messages of other types (Hypotheses H1). To study this, we need to analyze the network behavior and performance of message forwarding over time.

Figure 9 shows the recall of individual messages as dots over the duration of the scenario in case static prioritization is applied. One can clearly observe the previously discussed overall characteristics. It is important to note that even though there are less emergency messages in the system after around 2.5 hours, they still significantly hinder the propagation of other message types. This can be observed in Figure 10, where the average load regarding messages of a given type waiting to be sent is shown. The capacity of the network is 10 messages per minute per host, as indicated by the *actually sent* line. For the whole duration of the scenario, the number of emergency messages that should be forwarded—*Emergency (WS)*—exceeds this limit. Only very few new messages of other types are sent, illustrating the problem of an amplification of the DTN-induced long-living duplicates of high priority messages.

D. *Impact of Adaptive Prioritization*

Next, we look at the propagation characteristics of the adaptive prioritization over time. If we look at the individual message recall in Figure 11 and the buffered (WS) as well as the actual forwarded (AS) messages per host in Figure 12 we clearly see, that the utilized bandwidth and the high recall values follow the applied workload with the different message type peaks in Figure 6.

Figure 11 also confirms hypothesis H2 from Section II, that claims that adaptive prioritization prevents urgent information, such as emergency messages, when other message type trends are currently detected and prioritized by the nodes.

Another observation from Figure 12 is the increased overall amount of buffered messages, compared to the static prioritization in Figure 10. The increased amount of buffered messages results in the fact that Nodes in DTNs are not able to build a consensus of trending message types. The nodes have most likely calculated different prioritization orders, and therefore a higher amount of messages are spread in the network while fewer duplicates are generated. The 20 minute time window that is used to calculate the priority based on the buffered messages can also be observed in Figure 12 as a delay until the corresponding type is prominent in the buffered and actually send messages. Using adaptive prioritization, in the first-hour emergency messages have the highest recall. This results from the fact that nodes switch to static prioritization if they have calculated the same rank for different messages types. This happens, especially at the beginning of the simulation, when message types are generated equally, and the amount of messages spread in the network is still low.

VI. CONCLUSION AND FUTURE WORK

In this paper, we studied the implications of message prioritization in resource constrained DTNs for a post-disaster communication system. Based on the analysis of a large-scale field trial and a survey of related work in the area, we formulated three hypotheses regarding potential dependencies and—potentially undesired—interactions. We then proposed an architecture to study the impact of prioritization algorithms on resource-constrained DTNs, relying on a common abstraction for the message storage to decouple the DTN protocol from the utilized prioritization algorithm.[1] Our architecture allows prioritization algorithms to access scenario-specific meta information provided by the respective mobile application, potentially including further contextual information. We conducted a proof-of-concept evaluation confirming our hypotheses and further motivating research on adaptive, scenario-specific prioritization algorithms.

We plan to study the impact of prioritization algorithms under more realistic workloads relying on traces gathered during our field trial. Further, we explore the benefit of locally available contextual information, such as readings from smart watches or fitness trackers, which would allow us to consider, e.g., the physical health of a user during prioritization.

ACKNOWLEDGMENT

This work has been co-funded by the following projects: the LOEWE initiative (Hesse, Germany) within the NICER[2] and the Natur 4.0 project. The German Federal Ministry of Education and Research within the SMARTER[3] and the German Research Foundation (DFG) as part of project B1 within the Collaborative Research Centre (CRC) 1053 – MAKI[4].

REFERENCES

[1] F. Alvarez, L. Almon, P. Lieser, T. Meuser, D. Yannick, B. Richerzhagen, M. Hollick, and R. Steinmetz, "Conducting a Large-scale Field Test of a Smartphone-based Communication Network for Emergency Response," in *ACM CHANTS*, 2018.

[2] S. Bhattacharjee, S. Basu, S. Roy, and S. Das Bit, "Best-effort delivery of emergency messages in post-disaster scenario with content-based filtering and priority-enhanced prophet over dtn," in *Proceedings of the COMSNETS*, 2016.

[3] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine, "Maxprop: Routing for vehicle-based disruption-tolerant networks," in *Proceedings of the IEEE INFOCOM*, 2006.

[4] S. R. Chowdhury, M. Imran, M. R. Asghar, S. Amer-Yahia, and C. Castillo, "Tweet4act: Using incident-specific profiles for classifying crisis-related messages," in *Proceedings of the ISCRAM*, 2013.

[5] P. Earle, D. Bowden, and M. Guy, "Twitter earthquake detection: earthquake monitoring in a social world," *Annals of Geophysics*, 2012.

[6] J. T. B. Fajardo, K. Yasumoto, and M. Ito, "Content-based data prioritization for fast disaster images collection in delay tolerant network," in *Mobile Computing and Ubiquitous Networking (ICMU), 2014 Seventh International Conference on*. IEEE, 2014, pp. 147–152.

[7] T. Gao and D. White, "A next generation electronic triage to aid mass casualty emergency medical response," in *Engineering in Medicine and Biology Society,EMBS'06. 28th Annual International Conference*. IEEE, 2006.

[8] T. R. Henderson *et al.*, "ns-3 Project Goals," in *ACM WNS2*, 2006.

[9] R. D. Hofmann, "Smart operation of a mobile network in event of disaster," in *Proceedings of the International Telecommunications Energy Conference (INTELEC)*, 2008.

[10] P. Klinsompus and N. Nupairoj, "Critical message scheduling for disaster response and recovery phases," in *IEEE ICTC*, 2015.

[11] A. Kongthon, C. Haruechaiyasak, J. Pailai, and S. Kongyoung, "The Role of Twitter during a Natural Disaster: Case Study of 2011 Thai Flood," in *Proceedings of PICMET: Technology Management for Emerging Technologies*, 2012.

[12] J. Leaning and D. Guha-Sapir, "Natural disasters, armed conflict, and public health," *New England Journal of Medicine*, 2013.

[13] P. Lieser, A. Alhamoud, H. Nima, B. Richerzhagen, S. Huhle, D. Böhnstedt, and R. Steinmetz, "Situation detection based on activity recognition in disaster scenarios," in *ISCRAM*, 2018.

[14] P. Lieser, F. Alvarez, P. Gardner-Stephen, M. Hollick, and D. Boehnstedt, "Architecture for Responsive Emergency Communications Networks," in *IEEE GHTC*, 2017.

[15] F. Luqman, "Triage: Applying context to improve timely delivery of critical data in mobile ad hoc networks for disaster response," in *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2011 IEEE International Conference on*. IEEE, 2011.

[16] F. Luqman, F.-T. Sun, H.-T. Cheng, S. Buthpitiya, and M. Griss, "Prioritizing data in emergency response based on context, message content and role," in *Proceedings of the International Conference on Wireless Technologies for Humanitarian Relief (ACWR)*, 2011.

[17] J. Mclinn, "Major power outages in the us, and around the world," IEEE Reliability Society, Tech. Rep., 2009.

[18] B. Mukherjee, M. Habib, and F. Dikbiyik, "Network adaptability from disaster disruptions and cascading failures," *IEEE Communications Magazine*, vol. 52, no. 5, pp. 230–238, 2014.

[19] D. T. Nguyen, S. Joty, M. Imran, H. Sajjad, and P. Mitra, "Applications of online deep learning for crisis response using social media information," in *Proceedings of the International Workshop on Social Web for Disaster Management (SWDM)*, 2016.

[20] A. Olteanu, C. Castillo, F. Diaz, and S. Vieweg, "Crisislex: A lexicon for collecting and filtering microblogged communications in crises," in *Proceedings of the International AAAI Conference on Web and Social Media (ICWSM)*, 2014.

[21] Y. Qu, C. Huang, P. Zhang, and J. Zhang, "Microblogging after a major disaster in china: A case study of the 2010 yushu earthquake," in *Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW)*, 2011.

[22] F. Ranghieri and M. Ishiwatari, *Learning from Megadisasters: Lessons from the Great East Japan Earthquake*. Washington, DC: World Bank, 2014.

[23] B. Richerzhagen, D. Stingl, J. Rückert, and R. Steinmetz, "Simonstrator: Simulation and Prototyping Platform for Distributed Mobile Applications," in *SIMUTOOLS*, 2015.

[24] N. Richerzhagen, B. Richerzhagen, D. Stingl, and R. Steinmetz, "The human factor: A simulation environment for networked mobile social applications," in *Proceedings of the International Conference on Networked Systems (NetSys)*, 2017.

[25] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Spray and wait: An efficient routing scheme for intermittently connected mobile networks," in *Proceedings of the ACM SIGCOMM Workshop on Delay-tolerant Networking*, 2005.

[26] A. Vahdat and D. Becker, "Epidemic routing for partially-connected ad hoc networks," Duke University, Tech. Rep. CS-200006, 2000.

---

[1] The source code and simulation environment is available online (please contact the authors for full access): https://dev.kom.e-technik.tu-darmstadt.de

[2] Networked Infrastructureless Cooperation for Emergency Response

[3] Smartphone-based Communication Networks for Emergency Response

[4] Multi-Mechanisms Adaptation for the Future Internet