Abstract—Disasters result in disruptions to the communication infrastructure, typically forming a multitude of fragmented networks. This makes it difficult to exchange even simple while critical messages with large portions of the affected population, including first-responders and government authorities. In this paper, we focus on a content driven data retrieval model and propose an enhanced information-centric network (ICN) approach to provide communication resilience to such disruption-prone, delay-tolerant networks. The message exchange between communities and mules takes into account the limited available resources and the fact that the interest and the data might traverse completely different paths, unlike the assumption in many of the existing ICN solutions. Moreover, we argue for the separation of the the logical faces from the actual physical interfaces taking into account the fact that the data mules behave as mobile routers and are connected to different nodes at different time periods, unlike the assumptions made in existing ICN solutions. Our preliminary evaluations show that our solution is able to outperform other approaches such as “Epidemic” and “SprayAndWait” with respect to the latency, response probability and overall performance while improving the performance (latency and response probability) for high priority messages.

I. INTRODUCTION

Disasters have often disrupted communications because of damages to critical infrastructure. For instance in the aftermath of the Japanese Earthquake in 2011, approximately 1.2 millions of fixed telephone lines and 15,000 base-stations were not functioning. On average, 22% (with peaks up to 65% in some areas) of the base-stations had to shut down due to the lack of power or damages to the infrastructure. During and in the aftermath of a disaster, there is also substantial increase in the amount of traffic generated because of the natural anxiety and panic among people and the need to organize rescue and emergency services. In the Japanese case, congestion resulted in restrictions in voice traffic up to 95%, including emergency priority calls [1], [2].

Disruptions to the communication infrastructure also results in fragmenting the network where clusters of nodes are disconnected from the rest of the communications infrastructure. Multiple fragments may exist, with the potential for limited communication capability within the fragment, but without infrastructure based connectivity to others. This makes it difficult to exchange even simple, and critical messages with large portions of the affected population, first-responders and government authorities. Figure 1 illustrates such a scenario where the aftermath of a disaster results in the creation of fragmented networks (represented as A–H) in a metropolitan area. This may be the result of failures to the backhaul, metro-area network or even connectivity to the backbone in a cellular network. Each of these fragmented local regions could include one or more emergency shelters such as hospitals, schools, first-aid centers, police stations and/or isolated group of people. People in one fragmented network might want to I) urgently communicate with family members far away or even just in a different region in the metro area (hence different fragment); II) contact first responders for help; III) know when a supply of medicine/food is expected to arrive; IV) receive information/warning of impending danger from government authorities V) know the state (e.g., hospitalized) of people in another fragmented network. Moreover, the information being sought by people (status) or being communicated by the authorities (warnings, critical information) are mostly with regard to the current status of the disaster and warnings for impending danger. It does not matter where this information has to be obtained from as long as it is the appropriate information that can be trusted.

This calls for an information centric solution [3]–[7] that enables communication without regard to the specific location or network address of an entity that can provide the desired information. It is also very desirable for this communication to be delay-tolerant, because it may highly depend on the need to take advantage of data-mules that could be used to provide intermittent, infrequent connectivity between these fragmented networks. While this form of communication can be of sub-
stantial value, it is likely that the amount of information that can be carried may be limited because of bandwidth limitations (a mule visits a fragmented community for a short period of time) or storage (the mule has limited storage). As a result, it is also necessary to consider prioritizing the information that is to be delivered over the limited communication facilities.

Information Centric Networking (ICN) [5]–[8], a future Internet architectural approach focuses on accessing and delivering information based on its name, and is a natural fit for communication in disaster situations, providing both the timeliness and coverage needed for such critical communications. Moreover, ICN is designed to provide authenticity and integrity of the data that is identified by name. These properties are useful for communication during disaster situations where many ad-hoc devices that could potentially be untrusted and unreliable are used to transfer very important data such as emergency warnings. Furthermore, the content name based store and forward capability significantly eases data retrieval in highly distributed environments, since it allows data to be replicated at several locations for better information dissemination and fast data access even in adverse conditions. Unfortunately, most of the existing ICN solutions do not consider the scenarios we consider, which are characterized by: i) a need to support Delay Tolerant communications; ii) result in a disconnect between when the interest is expressed to when the data is actually received; iii) the data-mule that carries the interest could be different from the data-mule that could potentially carry the data back since they are travelling in different directions and probably have different movement patterns; and iv) the data-mule behaves as a mobile ICN router whose physical interfaces connect to different nodes at different time periods.

In this paper, we propose Delay-tolerant ICN for Disaster-management (DID), that primarily targets interest-based content retrieval between fragmented networks in the aftermath of a disaster. The main contributions of this paper are:

- **Design of DID to improve communication resilience** in the aftermath of a disaster where networks may be fragmented, requiring delay tolerant communication between fragments (detailed scenario is illustrated in Section III).

- **We identify the need for separation between a logical face of a node and a physical interface.** This is key to enabling the operation of data mules that are mobile and may connect to different logical points such as fragmented communities at different time periods in the disruption prone environment typical of disaster situations.

- **DID enhances ICN protocols to enable the interest and data to travel via completely different paths and tolerate large and variable delay between when an interest is expressed and when the data is available** (see Section IV).

- **DID efficiently utilizes the limited available communication resources by allocating more resources to higher priority data, but not starving lower priority data** (see Section IV). Priority of message exchange between two nodes takes into consideration factors such as the characteristics of the data (e.g., if it is a safety-critical message, data size), number of people interested in the data, the mobility pattern of both the data-mules and the data consumer/producer.

- **Preliminary evaluations show that DID is in fact more efficient than approaches such as Epidemic [9] and SprayAndWait [10] with respect to the delay, response probability and overall performance. It succeeds in delivering high priority messages without starving low priority ones** (see Section V).

## II. RELATED WORK

Delay tolerant networks (DTN) are networks in which the communication ends have to tolerate much larger delays in message delivery than in usual networks. This is due to the impossibility to achieve better performances, either because of external factors or for some specific convenience, e.g., energy saving. They are useful in a variety of scenarios and since its inception [11], many different protocols have been proposed (see [12]–[14] for a comprehensive overview). The most common scenarios considered in literature include networks where the high node mobility makes impractical the use of traditional routing approaches and networks where the end-to-end connectivity is not guaranteed. In the first case, upon encounter, nodes may exchange information. Instead of defining a routing strategy, i.e. associate a set of nodes to each possible destination as next hops, at each encounter the nodes decide whether to forward or not the messages they carry, according to some protocol-specific strategy.

A popular approach is represented by zero-knowledge protocols, including among the others Epidemic and SprayAndWait [9], [10], [15]. In this approach, the general trade-off is between transmitting messages often (flooding in the extreme case), minimizing the delay, and sending more rarely, minimizing the network usage. The peculiar aspect of these protocols is that the choice is not based on any previous collected information. This class of protocol doesn’t fit well the discussed scenario, because it assumes all the nodes can independently transmit and receive and must keep moving to spread messages. But in a disaster scenario there could be energy constraints and for nodes to continuously transmit could lead to a rapid exhaustion of the available energy. Furthermore, the high overhead these protocols usually introduce doesn’t cope well with a highly loaded network and could lead to severe network congestion.

Another broad class of protocols could be called statistical, in which a node tries to better estimate if another node can help the message delivery through some probabilistic analysis, either based on past data [16](e.g., frequency and length of the encounters, visited locations, etc.) or other information [17](e.g., node location, movement direction). The heuristics’ efficiency is often dependant on scenario specific assumptions [18]. Our work builds upon the previous work, adapting it to the disaster scenario to exploit the fact that communities do not move and can thus be used as a reference to determine movement patterns and as safe relays to enhance message deliver probabilities.

A further novelty element introduced is the adoption of a content centric model in a delay tolerant network. The concept
of mobile, delay-tolerant nodes. Until now most related proposals have targeted wired networks like the Internet [5]–[8]. Most of the work to bring information centricity on MANET, vehicular networks or DTN is still at early stages [19]–[22]. Recently, Psaras et. al. proposed the use of Information Centric Networking for disaster scenarios [23], but unlike DID, they rely on a flooding strategy where the data is spread to all the nodes that encounter each other and make use of priority if there is limited space in the nodes. Our work compliments these works and introduces concepts such as the logical face/interface, no dependency on reverse path forwarding of data, priority, encounter probability based forwarding, etc.

III. Scenario Description

A. Fragmented Networks

DID is designed to facilitate communication resilience in a DTN scenario similar to the aftermath of a disaster. Note that in the case of a complete breakdown of the communication infrastructure (e.g., all the base-stations in that area are down), each of the shelters, buildings and etc. (shown in Figure 1) could be fragmented from one another leading to a large number of fragmented network communities in a densely populated urban area. These fragmented networks, may generate and require significant amounts of information from people within those networks and thus be opportunistic communication hubs. Moreover, these fragmented networks also represent fixed points in the topology, where information can be collected, cached or further spread. As shown in Figure 1, it is assumed that in each fragmented network, there is a gateway that collects messages to and from external communities. In our work, we consider each of these fragmented networks as a single node. The communication within a fragmented network could use any mechanism such as Ad-hoc modes, Wifi or wired LAN for communication, and is not the focus of this work. Although the members of a fragmented network may join and leave (e.g., people getting admitted to a fragmented network that hosts a hospital), it is assumed that the fragmented network remain at fixed locations.

B. Data-Mules

We assume that vehicles such as ambulances, police-cars, helicopters, people that naturally move between points (e.g., shelters) with mobile phones could be used as data mules to connect or facilitate communication between these fragmented networks. The mules are assumed to be moving along streets or certain paths (unless it is a drone flying from one fragmented network to another) and have a limited interaction time with each contact point. The contact point could be other mules it meets on its path or fragmented networks that are on its path acting as source and/or destination. Different data-mules are expected to have different movement patterns, e.g., ambulances return to one of the hospitals with a very high probability whereas first-responders might move around with a set plan or based on orders/needs. For instance, as shown in Figure 1, the helicopter moves from E -> D -> G, whereas the police car moves from F -> E and back. The movement pattern of the mules are critical to determine the potential interactions between the fragmented networks. Moreover, if a fragmented network is close to an important or forced physical path, or lies near the center of the network topology, then it is likely to encounter mules more often and thereby have many more interactions. Based on the movement pattern of the mules, they can be categorized into the following groups:

- Constrained mules: Their movement and encounters are more predictable. For instance, this category could include ambulances that return to a particular hospital or one of the many adjacent hospitals, soldiers that return to a barrack or storehouse to collect more food and medicine, etc. Therefore, the destination of these mules might be predictable to a certain extent.
- Unpredictable mules: Their movements are difficult or impossible to predict. It can be a person or vehicle only occasionally acting as a mule, or a mule for which the prediction algorithm does not yet have enough information to make an estimation.
- Mules with pre-defined paths: We can assume that certain mules are launched by the rescue team to facilitate data delivery. Since they follow a pre-defined path, it is easier to determine the communities they will encounter next.

IV. Solution

Unlike standard ICN solutions [5]–[8] and ICN solutions that exist for DTN environments [19]–[22], we believe that there is a clear need for separation between the physical interface and the logical face of an ICN entity. In NDN, the FIB maintains the face to the next hop that the interest needs to be sent, to reach one of the sources of the data, whereas the Pending Interest Table (PIT) maintains the face to the next hop that the data needs to forwarded to reach the final destination. As the interest traverses the network, entries are stored in the PIT to help the data follow the reverse path, using reverse-path-forwarding (RPF) rules, towards the requester.

In our DTN environment, the mules are mobile. We view them as mobile routers whose physical interface comes in contact with different communities at different points in time. Therefore, we introduce the need to associate entries in the FIB and PIT tables with the logical faces to the different communities instead of the physical interface. As and when a mule connects to the fragmented network community’s gateway or another mule, the logical face becomes active and data transfer can proceed based on the entries in the corresponding table. Similarly, from the point of view of the community, the physical interface of its gateway interacts with different mules at different time periods. Therefore, storing the logical face to the different mules in its FIB and PIT is needed, rather than the physical interface. In DID, the mules as well as the gateways to the communities are seen as routers with logical faces to one another and these logical faces become active at different time periods based on the entity it encounters.

Below, we describe the design of DID by first describing the communication model and then detailing the protocol.

A. Communication model

The primary focus of DID is to facilitate the exchange of information among people belonging to different fragmented communities. The communication model chosen is
interest based retrieval which refers to both a single response addressing the query or the presence of a standing query (i.e., similar to pub/sub, so that when information is published, it is sent to those who have issued the standing query). Similar to NDN [6], interest and data are the two kinds of messages exchanged. Data messages have variable size, while interest messages typically have a fixed size. An interest requests data by a name prefix. The name prefix is assumed to identify either a specific data item or a set of data within the network. These messages, generated by members of a fragmented community, are delivered to a specific gateway node responsible for communicating with the external world (see fragmented community D shown in Figure 1). It is assumed that the gateway has a relatively large cache where messages may be retained until it is possible to deliver them or they expire.

Data mules take on the responsibility to transport messages between communities; when a mule encounters a fragmented community gateway, it delivers messages believed to be destined to the gateway’s community or is believed to be a good next hop for the message; it also collects messages from the community that it can help deliver. The probability of message delivery by a given mule depends on the mule’s path and future encounters with other gateways, but these are not always pre-defined. In order to transport interest messages, the mule also maintains a table that is similar to the Forwarding Information Base (FIB) in NDN [6], where the name prefix and next hop fragmented community to obtain it from are stored. Due to the heuristics involved, the table includes an additional column that is the estimate of the probability of that next hop being the optimal next hop for the message.

B. Encounter Prevision

To estimate future encounters, mules maintain a data structure, the Encounter Table, where the logical face to each known fragmented community gateway is associated with a value that expresses the probability that they will meet a particular community gateway in the future. When a mule meets a fragmented community, the logical face identifier associated with that community becomes active and the messages are transferred from the mule to the gateway. Similarly, the logical face on the fragmented community becomes active and the data is transferred from the community gateway to the mule. The advantage of associating the entry with the community’s logical interface is that even if the physical gateway changes (e.g., the chance of this may be high in a disaster affected area), some other node can take up this role seamlessly without the network having to drop messages it has already transported with a different gateway.

For mules with pre-defined paths, the table could be manually set up. For the others, a heuristic could be used to populate the table. This heuristic would consider the length and frequency of past encounters, assuming that if a mule met a community often in the past, it is likely to also meet it in the future. The geographical position of the mule and of the communities would also be considered, if available.

If the Encounter Table is manually populated, fully reflecting the mule’s itinerary, the data dissemination is expected to be similar to the message ferrying model [17], minimizing network usage. When the Encounter Table is filled based on a set of default values, the mule accepts to transport all messages with the same probability, irrespective of their destination, and the behavior is similar to a flooding (or epidemic) approach. When the probabilistic approach based on the heuristics is used to populate the table, the behavior is expected to be better than the “default values” approach, although it likely will not perform as well as the model based on the perfect knowledge of the mule’s path.

C. Protocol Overview

When a Community and a Mule meet, the mule transmits its Encounter Table, so that the Community knows which destinations are more likely to be reached by the mule. Based on this information and on the priority of each message, the community assign a transmission priority to each outgoing message. Outgoing messages can thus be sorted and the message with the highest transmission priority is the first to be transmitted.

If the mule correctly receives the message, it is removed from the outgoing queue and its priority is decreased proportionally to the probability that the mule will deliver it to its final destination. When a message priority reaches zero, the message is either discarded or kept in the local cache, but no more transmitted.

When the buffer is full and a generic message $m$ has to be stored, if enough messages with priority lower than $m$’s are in the buffer, they are discarded to make space for $m$, otherwise $m$ is dropped.

D. Protocol Details

1) Compute the Message Priority: To decide the transmission order, communities assign a priority to each message. The message priority is directly proportional to the number of community members who requested it and is inversely proportional to its size.

$$priority_m = \frac{\text{requesters}_m}{\text{size}_m} \tag{1}$$

While other, more sophisticated, priority measures could be defined, it is important to ensure that prioritization does not lead to starvation.

2) Compute the Transmission Priority: To determine the transmission order of the outgoing messages from the community, the message priority and the probability that the mule will reach the message’s destination are considered, to compute the transmission priority ($tp$) of each message $m$.

$$tp_m = priority_m \cdot et_m \tag{2}$$

where $et_m \in [0, 1]$ is the probability associated to the message’s intended destination in the mule’s Encounter Table. The messages with higher transmission priority are uploaded first, thus making optimal utilization of the mule-community encounter time and ensuring that the most important messages have been transmitted from the community first. A “mule-to-mule” encounter is also similar to the mule-community
A destination is associated with each interest (of size 1). Figure 2 shows an example handshake and transmission priority computation between a mule and a particular community. Note that it is not necessary to compute $tp$ for all stored messages before starting the transmission, since the community keeps messages sorted according to priority and the computation of $tp$ is a monotonic function for messages with the same destination. Thus, it is only necessary to compute it for the message with the highest priority for each destination.

3) Decrease the Message Priority: After a message is transmitted, it is important to prevent it from being retransmitted with the same (potentially high) priority on the subsequent visit by the same or different mule. It could either be removed from the list of outgoing messages, or its priority be decreased. One reason to not simply remove the message from the list of outgoing messages is because the delivery of a message by a particular mule is only probabilistic.

$$requesters_m = requesters_m \cdot (1 - et_m/2) \quad (3)$$

By reducing the number of requesters (Equation 3) associated with the message, the priority of the message is also reduced, proportionally to the probability $et_m$ that the mule will correctly deliver the message.

V. PRELIMINARY EVALUATION

We now present the results of a simulation-based evaluation of DID using the ONE [24] simulator which is specifically designed to recreate scenarios with high node mobility. The nodes move on a map representing an urban area (similar to Figure 1). There are five communities at fixed locations and five mules follow pre-defined paths. There is also an additional mule that periodically returns to the home community after visiting a random community. For each experiment, roughly 10000 messages are generated over a period of twelve hours in simulation time. Interest messages are 50 bytes long, and data messages have variable length ranging from 5 byte to 100 bytes. The movements performed by the mules and the generation of the messages are identical for each of the protocols tested. Each message is randomly assigned a priority at the time of creation and the priority ranges from 1-100 with 100 being the highest priority. Communities have larger buffers of 500MB each, while the mules have a buffer of only 70KB.

In the experiments DID is compared to Epidemic [9] and SprayAndWait [10]. Special attention is given to estimating the effectiveness of the priority algorithm used in DID. It is important to deliver important messages with higher probability and lower latency, especially when the network is congested (likely in the case of disaster). In addition to the priority of the messages, other factors such as the movement pattern of the mules, their encounter probability, number of interests expressed for the message and message size are taken into account (see Section IV) to decide the relative priority to transfer a message when a mule encounters a community. Although the communication is delay tolerant, it is important to deliver at least some of the messages in a timely manner, either because of the urgency for the user or because its relevance decreases after a certain amount of time. For this reason, each message has a time to live (TTL) that expresses the message’s validity and can vary according to the message type. The TTL of all messages is set to 300 minutes (5 hours).

Figure 3 compares the results of the three approaches in terms of average time to deliver (since the message was created), the probability of receiving a response to a user in a community sending an interest. The figure also provides an overview of overall performance for all messages, ignoring their priority. In Figure 3a, the messages (both interests and data) are grouped according to their priority into ten priority classes for better visualization. We observe in Figure 3a that the average delivery time of DID is lower than SprayAndWait in most cases. More importantly, it is clearly lower for higher priority messages. Although the tests were performed with a slightly overloaded network, the fact that only messages with lower priorities (for 0 and 10) have a higher delay means the prioritization algorithm works as expected. The higher average delays for Epidemic and SprayAndWait can be justified by the higher overhead they introduce, which causes an increase in the network congestion.

Figure 3b depicts the response probability in terms of the percentage of interests that received the requested data before the expiration time. If more than 5 hours pass between the time an interest is expressed and when the corresponding data is returned, the interest is considered no longer valid and gets dropped. DID performs significantly better than both Epidemic and SprayAndWait in achieving a much higher response probability, thereby making efficient use of the limited network resources. Furthermore, the response probability is lower than 100% only for messages with low priority (again, from 0 to 30).

Figure 3c is an overview of the performance of the three protocols in terms of i) overhead: the ratio of the total number of transmitted messages to the number of messages delivered to the final destination; ii) delivery probability: the overall proportion of messages that are correctly delivered; iii) average latency; iv) the number of dropped messages. The values are normalized. Note here that the priority of the messages is not considered. The results illustrate that DID outperforms
In this paper, we enhance ICN to support disrupted delay tolerant networks, particularly for disaster situations. We argue for the need to reconsider the assumptions made in the design of the earlier ICN architectures. Our design takes into consideration the fact the data mules, that travel between fragmented communities in a disaster scenario, behave as mobile routers. When they encounter different nodes (fragmented community gateways, other mules) at different time periods with varying probabilities, it is important to develop a solution that prioritizes what data gets exchanged. Moreover, it is important to design the solution such that the data does not necessarily need to take the reverse path in which the interest was delivered. Preliminary evaluations show that DID works as expected in such a disruption-prone network. It is superior to other popular approaches such as “Epidemic” and “SprayAndWait”. In the preliminary evaluations we conducted, we did not consider prioritization for other approaches. We are currently pursuing a more detailed evaluation and exploring avenues to further improve the solution.

VI. SUMMARY

In this paper, we enhance ICN to support disrupted delay tolerant networks, particularly for disaster situations. We argue for the need to reconsider the assumptions made in the design of the earlier ICN architectures. Our design takes into consideration the fact the data mules, that travel between fragmented communities in a disaster scenario, behave as mobile routers. When they encounter different nodes (fragmented community gateways, other mules) at different time periods with varying probabilities, it is important to develop a solution that prioritizes what data gets exchanged. Moreover, it is important to design the solution such that the data does not necessarily need to take the reverse path in which the interest was delivered. Preliminary evaluations show that DID works as expected in such a disruption-prone network. It is superior to other popular approaches such as “Epidemic” and “SprayAndWait”. In the preliminary evaluations we conducted, we did not consider prioritization for other approaches. We are currently pursuing a more detailed evaluation and exploring avenues to further improve the solution.

VII. ACKNOWLEDGEMENT

This research was funded by the joint EU FP7/NICT GreenICN Project, Contract No. 608518 and NICT No. 167.

REFERENCES