A Human-Centric Cloud and Fog-Assisted IoT Platform for Disaster Management

Bikesh Maharjan
Computer Science Dept.
North Dakota State Univ.
Frago, USA
bikesh.maharjan@ndsu.edu

Juan Li
Computer Science Dept.
North Dakota State Univ.
Frago, USA
j.li@ndsu.edu

Shadi Alian
Computer Science Dept.
North Dakota State Univ.
Frago, USA
shadi.alian@ndsu.edu

Yan Bai
School of Eng. & Tech
Univ. of Washington Tacoma
Tacoma, USA
yanb@uw.edu

Abstract— **Natural and man-made disasters pose an ever-present threat to our community and society. Effective information and communication platform are important for disaster response. However, under disaster scenarios, network infrastructure, such as power grid, cellular tower and backbone networks can be damaged. In addition, the communication channels often become congested by exceptionally high levels of data traffic, as people impacted by the disaster are trying to contact with their family and friends and seeking for help; disaster responders are working to deploy rescue resources, and lots of citizens are broadcasting the disaster by uploading pictures or videos. In this paper, we propose an emergency information platform that is suitable for a challenged environment where the traditional communication paradigms either completely fail or poorly perform. We utilize the vast deployment of Internet of Things (IoT) and citizens in the community to bring benefits in terms of data network resilience in face of disaster. In particular, we propose a software-defined communication network that combines IoT technology, Fog computing, Cloud computing, peer-to-peer computing and delay-tolerant network together to enable an effective emergency-response information platform. It integrates the power of community, and citizen with IoT devices and existing partially damaged communication platform to address critical disaster communication problems.**

Keywords— cloud, disaster management, fog computing, delay tolerant networking

I. Introduction

In the past year, we have witnessed a growing number of hazard events, such as Hurricane Michael, Hurricane Florence, the Indonesia Earthquake and Tsunami, Super Typhoon Mangkhut, California Wild Fire, and the Pittsburgh Shooting. To respond to disasters in a fast and an effective manner, decision makers need to have an up-to-date disaster situational picture. Many crowdsensing/crowdsourcing technologies, such as social media and sensor networks, are used in collecting disaster information. Sensors and humans collect disaster data and report them to a centralized cloud server where data related to the disaster are processed, analyzed. Decisions are made based on the analysis results. However, it may not be possible to maintain a well-connected network during the disaster: natural or man-made disasters cause structural damage not only to buildings, roads, and transportations, but also to the power grid, cellular tower and communication backbone networks. Furthermore, information channels often become extraordinarily congested with significantly increased data traffic caused by people seeking for assistance, disaster responders responding the disaster, and others querying for disaster situation. When data from numerous sources flows into a central Cloud processing point, bottlenecks will occur. Information transmission can be can be intensely constrained or even shut off. Critical information may get lost while waiting for the congestion to be cleared. Consequently, it would affect rescue decisions and create differences between life and death for victims in the disaster. Therefore, availability of minimal information and communication services is crucial to allow efficient and coordinated disaster response.

In this paper, we propose an effective emergency information and communication system that is suitable for a challenged environment where the traditional disaster management Information and Communications Technology either completely fails or poorly performs. We utilize the vast deployment of Internet of Things (IoT) and citizens in the community to bring benefits in terms of data network resilience in face of disaster. IoT entities, such as sensors, mobile devices, vehicles and humans (with smart phones or wearable devices) act as crowdsensing tools to sense and deliver various disaster information. Leveraging their battery-powered property and spontaneous wireless networking capabilities, IoT devices could enable minimal communication services while the conventional communication infrastructure is out of service [1]. Like typical IoT communication architectures, the proposed platform enables IoT devices to connect to the Cloud through communication backbone (i.e., the Internet) using an infrastructure-based wireless network paradigm. We also propose a novel Fog Computing infrastructure to enable communications between edge nodes and fog nodes autonomously using a spontaneous wireless network paradigm, with or without the help of any infrastructure.

To further improve communication efficiency, an efficient disruption-tolerant data propagation mechanism is developed

2573-7562/19/$31.00 ©2019 IEEE
DOI 10.1109/MobileCloud.2019.00009

which uses moving people and vehicles to forward data between IoT devices and Fog servers. The idea behind relies on the concept of no reliance on end-to-end path connectivity at any point in time. This communication is suitable for challenged environments where the traditional communication paradigms either fail completely or rather perform poorly. In addition, Fog computing brings Cloud resources close to the underlying IoT devices, making it an ideal technology for latency sensitive services in disaster scenarios. Moreover, we prioritize network traffic based on the messages' ontology coding. The most critical information can get higher priority to be processed and transported.

The rest of the paper is organized as follows. Section II surveys related work on communication and information technologies used to manage disasters. Section III describes our proposed platform in details. Section IV presents the experimental results. Finally, in Section V, we provide conclusions and future work directions.

II. Related Work

The Internet is vital to enable communication and management of disaster. In normal situation, the Internet is highly connected providing high redundancy. However, during or after a disaster Internet resilience would be very limited because of its dependency on fixed infrastructure components [2]. As pointed out by Petersen et al. [1], adjusting the Internet to a disaster adaptive infrastructure with existing service capabilities is an unrealistic challenge. Therefore, more and more research works are working to provide minimal communication in disasters by reducing networking complexity and following the major requirements in disaster scenarios.

A fast recovery of communication infrastructure is of utmost importance. A popular strategy nowadays is to deploy provisional connectivity using simple mobile stations (such as 3G or GSM), satellite communication links and spontaneous links with mobile generators for power supply [3]. For example, Google's project Loon can provide Internet access to disaster affected regions. The project uses upper air balloons positioned in the stratosphere to create an aerial wireless network with up to 4G-LTE speeds [4]. However, these types of approaches need lots of time to set the infrastructure up, as lots of communication equipment need to be delivered, installed and initialized at the disaster location. The connectivity created by the temporary communication infrastructure is very limited. Moreover, the temporary infrastructure may cause communication conflicts with existing infrastructure.

IoT technologies have been considered to overcome the problems of the traditional Internet infrastructure which depends heavily on fixed communication components and network links. IoT technologies can complement existing networking infrastructure with wireless transmission and battery power.

Considerable IoT devices have sensors which can sense and monitor different kinds environmental factors. Utilizing their sensing capabilities and their resilience of network failure, IoT devices are able to provide real-time information about disaster areas, which can help decision makers to better understand the impact of a disaster and react more appropriately. For example, BRINCO [5] is an IoT-enabled beacon system which can sense seismic waves and then notify users about potential earthquake or tsunami. BRINCO system has a sensing unit which includes an accelerometer, signal processing unit and audio alarm units. When the sensing unit senses any vibrations from the ground, it will send the result to a cloud, i.e. the Brinco Data Center, to further process this result. Very similarly, Grillo [6] is another IoT earthquake warning system, which can send warnings to people about potential hit by any seismic waves. Grillo is generally connected with Grillo Sensor Network, a proprietary network of Grillo devices in Mexico. BRCK (www.brck.com) is another IoT-enabled device, which can connect with low connectivity areas using 2G communication. BRCK is also supported by an backend cloud server, where sensed data can be transported to and result can be collected from. It is powered by solar energy, therefore, it is suitable for disaster areas where power supplies can be damaged. Citizen Flood Detection Network (www.flood.network) is an IoT-supported open crowd-sensing network. Sensors in the network are placed in different locations such as under bridges. Periodically, sensor nodes measure the water level. Sensing result is sent to a cloud which can be tagged in a map.

From the aforementioned IoT applications in disaster scenarios, we can see that although wireless sensor-based IoT networks are widely deployed in (potential) disaster affected areas, they have not yet been used to improve the resilience of communication networks in face of large-scale disasters. Instead, they are mainly used to provide sensing information to a centralized cloud to assist situation awareness and decision-making.

Besides the wireless sensor network (WSN)-based communication, ad hoc networks have been proposed as an alternative communication technology to deal with the unexpected communication network conditions emerged during and after a disaster. There have been many Mobile Ad hoc NETworks (MANETs) such as vehicular ad hoc network proposed to providing networking and communication support for disaster areas [7]. Researchers have evaluated several popular routing protocols for MANETs, including Ad Hoc On-Demand Distance Vector (AODV), Dynamic Source Routing (DSR), Dynamic Manet on demand (DYMO) and Better Approach To Mobile Ad hoc Networking (B.A.T.M.A.N.) in disaster scenarios, under different communication patterns [7]. They found that reactive routing protocols such as AODV are more suitable for disastrous scenarios. Vehicular Ad-Hoc Network (VANET) has been used for safety applications. For example, in the work proposed by Chen et al. [8], accident information can be exchanged via broadcasting messages in VANETs. Similarly, broadcast communications for emergency scenarios have been studied in other works [9].

Disruption Tolerant Networking (DTNs) are also utilized for improving network performance in disaster areas. For example, Tornell et al. evaluated the possibilities for applying existing DTN protocols [10] in disaster areas, Chakrabarti and Shibata et al., proposed new routing protocols and systems specifically for disaster situation [11].

Wireless mesh networks (WMN) have been used for disaster response networks as backbone network to extend Internet connectivity or to extend local connectivity in a WLAN. WMNs

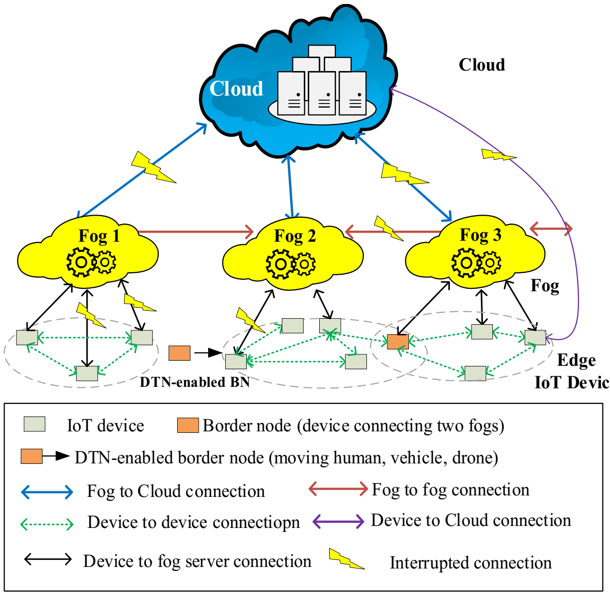
Cloud
Cloud
Fog 1
Fog 2
Fog 3
Fog
Edge
IoT Device
DTN-enabled BN
IoT device
Border node (device connecting two fogs)
DTN-enabled border node (moving human, vehicle, drone)
Fog to Cloud connection
Fog to fog connection
Device to device connectiopn
Device to Cloud connection
Device to fog server connection
Interrupted connection

Fig. 1. System Architecture

can be used to quickly form an ad hoc infrastructure when a disaster happens. For example, SKYMESH, a mesh network based on WiFi access points can provide rapid access to a WLAN network in a disaster area [12]. A similar mesh network, SwanMesh [13] can provide multimedia services in a disaster area.

Although different approaches have been proposed to solve various aspects of the communication problems in disaster scenarios, they lack interoperability to work together to form an effective service or communication platform. Effective communication in disaster scenario is still a challenging problem.

III. System Design

We propose a disruption-tolerant Fog-Cloud IoT computing paradigm to enable sustained IoT information networking in disaster scenarios as shown in Fig. 1. IoT devices at the edge of a network can sense and collect disaster-related information. The information will eventually be sent to the cloud, where it is processed and analyzed. End users and disaster management officials can query the cloud server to obtain the information. IoT devices may also provide services to local community, they can register their services to the Cloud. At the same time, IoT devices (or humans using IoT devices) can query the Cloud to get relevant information or contact of service providers.

Fog extends the Cloud service to the "ground," where services and contents located on the remote Cloud servers are now distributed to a multitude of fog servers located just in local communities (indoor or outdoor). When disaster happens, first-responders can deploy more fog servers to the disaster site. These fog servers have process, storage, and network transmission capabilities. Since many of the fog servers only provide a short-range communication access, they can be spontaneously deployed to the disaster site. Mobile users access Fog servers with just one-hop wireless connection. Therefore, the majority IoT services can become available even in areas with poor Internet coverage. Clearly, the transmission speed from a fog server to end devices is much faster than the speed from a remote Cloud to end users. Disaster data collected from edge IoT devices can be first sent to fog nodes for pre-processing, including filtering, aggregating and initial analysis. As a proxy of the cloud, part of queries related disaster can be resolved by a local fog server. The fog server may forward the queries to the Cloud on behalf of the requester if necessary.

Fog servers form a P2P-based structure to offload the centralized Cloud server, thus reducing the Cloud bottleneck problem in disaster scenarios. However, today's Fog computing is not capable of providing local services when Internet connectivity is impaired. To solve this problem, we propose to utilize the movement of humans, vehicles, or even drones to create an opportunistic network to assist data propagation. The opportunistic fog network is infrastructureless. Messages are transferred from a sender to the destinations by nodes in between, which store, carry and forward messages. This kind of networks is tolerant to delays and disruptions. Nodes can communicate with each other even if there is no route connecting between them as they can build routes dynamically. It is very important in emergency situations as it ensures that the messages and data generated in the disaster area reach their destinations without any loss.

As shown in Fig. 1, our Cloud-Fog-IoT platform consists of multiple layers. The Fog server can be hosted in devices at the access layer (such as a community-wide Internet gateway), at the gateway layer (such as within a building or in the nearby area), or within various objects (such as vehicles, laptops, or smartphones) that serve as IoT end points. There are five kinds of communication links between different layers, namely, device to fog, device to cloud, device to device, fog to fog, and fog to cloud. These links use different communication connections: Some use short-range wireless connections (device to fog, device to device); some use Internet connections (device to cloud, fog to cloud), others use the movement of objects such as (human, vehicle) plus short-range wireless connections (device to fog, fog to fog). Some of the communication connections can be damaged by disasters. The goal of this communication platform is to enable data propagation successfully using all the available links, including the delay-tolerant network (DTN)-enabled links. For example, due to interrupted connections, data created in Fog 1 (Fig. 1), cannot be sent to the cloud. A moving node (orange node) previously in Fog 1 now moving towards Fog 2, can forward the data to Fog 2, which may be able to process and pass the data to the Cloud.

A. Service and Communication Information Model

To effectively process and route the network messages, it is important for the system to express and process the semantics of the messages. For example, in order to evaluate the relative urgency of messages and give urgent messages more priority, the communication platform needs to understand the semantics of heterogeneous messages. However, it is hard to achieve a single common syntactic agreement between various IoT service providers and requesters. We choose ontology to realize this goal, as Ontology can represent resources in machine-understandable format.

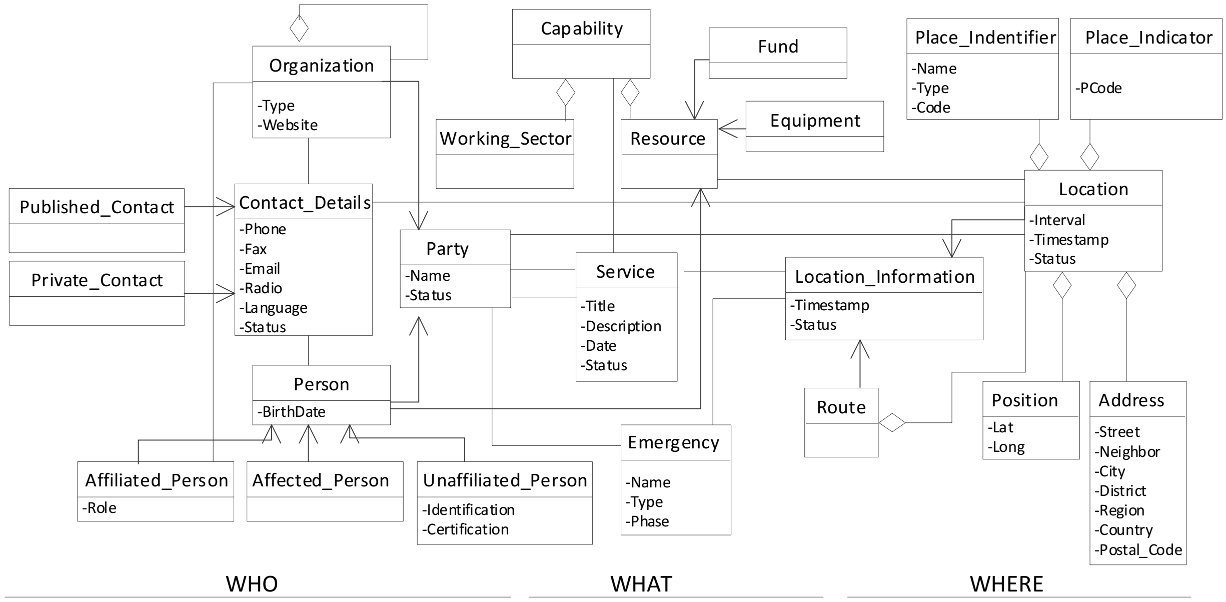

Fig. 2 Part of the Emergency Ontology adopted from the W3 Information Model

a) Disaster Ontology Model

We generalize and define emergency-related ontology mostly from the existing ontologies (such as [7], [14], [15] [16]) Completely formalizing all emergency ontology is likely to be an insurmountable task. Instead, we model the most fundamental concept, i.e., a set of upper-level entities, and provide flexible extensibility to add specific concepts in different application domains. Common emergency concepts shared by different domains are modeled as a general model. The separation of application domains encourages the reuse of general concepts, and provides a flexible interface for defining application-specific knowledge. In particular, we adopt the "who-what-where" model proposed by W3C Emergency Information Interoperability Framework [17] and extend it with other important directions including "When" the time dimension. To cope with the openness and extensibility requirements, we adopt two W3C recommendations: The Resource Description Framework (RDF) and the Web Ontology Language (OWL) as our ontology language. Fig. 2 presents part of the ontologies we defined following the W3C information framework.

b) Optimization with Ontological Encoding

Much research work (e.g. [18], [19]) has been conducted in matching Ontologies. However, most proposed technologies and tools for semantic reasoning and matchmaking are computationally expensive. Therefore, we cannot use them in IoT environments in which devices have only limited computational resources. We follow the idea of signature matching [20], which is a simple method to identify the subsumption relationships between concepts describing web services. To further reduce matching cost, we numerically encode ontologies, thus transforming the costly semantic reasoning into a simple numeric computation. Specifically, ontologies are classified into hierarchies offline. A prime-based encoding technique which we proposed in a previous research is applied to allow for subsumption testing of classes in ontologies [21]. For example, we can assign a unique prime number Pi to each class Ci in the ontology hierarchy. Assigning unique prime numbers to class will ensure the encoding to be conflict-free. Encoding of a class Ci, E(Ci) is defined as $E(Ci) = \prod j\ Pj$, where Pj is the assigned prime number of class Ci and its ancestors. As shown in our previous research, this encoding can efficiently identify whether two concepts in an ontology are related without performing costly semantic reasoning online [21].

B. Community Disaster Service and Request

Devices in an IoT network may provide various services for the community. We listed several typical service types: (1) sensing services that are provided by sensors (2) actuator services where actuators act and activate (3) tagging service that uses RFID tags, (4) content services that provide multimedia contents, and (5) gateway services that provide routing and internet connection services to other IoT devices.

Different IoT services may have different serving scopes: some services are local such as sensing results from a field that can be processed in a local processing point; vital sign data collected from human wearable devices that can be collected and analyzed by a local smart phone. Some services are regional or community-wide, for example, community-wide information forum, education and content sharing. Other service scope is outside of the community which need a Cloud support, such as reporting sensing result to the Cloud. Correspondingly, the system is required to support requests with different scopes. Different services and requests may have different urgency level, which is defined based the nature of the services request and response time requirements.

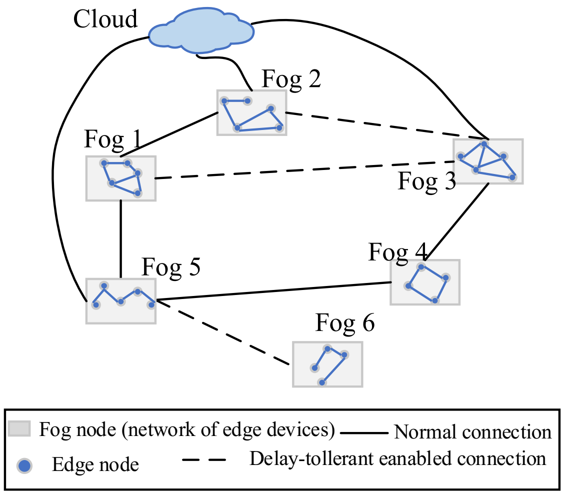
Cloud
Fog 2
Fog 1
Fog 3
Fog 5
Fog 4
Fog 6
▨ Fog node (network of edge devices)
—— Normal connection
● Edge node
– – Delay-tollerant eanabled connection

Fig 3. Fog layer overlay network

C. *Multidimensional Disruption-Tolerant Fog IoT Disaster Networking*

a) Networking Architecture

We propose a generic overlay network architecture that is capable of exploiting the aforementioned Fog infrastructure (cf. Fig. 1). It consists of three layer's hierarchical networking components: (i) a centralized Cloud layer, (ii) a P2P-based disruption-tolerant Fog layer, and (iii) a MANET-based edge layer. The Cloud communication layer is responsible for accepting, processing, and replying requests from the Fog servers and edge devices. At the network edge, IoT devices report to and request services from the Fog servers and the Cloud server. At the same time, IoT devices can spontaneously connect with each other to form a MANET using short-range communication channels such as Wi-Fi or Bluetooth. Existing MANET routing approaches, such as [22]–[24] can be applied to this network. In disaster scenarios, different rescue organizations and individuals will set up their own IoT MANETs using their specific equipment.

In the Fog layer, we categorize Fog servers as two different types, namely static nodes and mobile nodes based on their mobility levels. Static nodes do not move or move infrequently, for example, routers, community CCTV, and roadside unites. Mobile Fog nodes are more dynamic, for example, smart phones, vehicles etc. Here, we treat each Fog server and its clients as a Fog node. As shown in Fig. 3, a fog node will connect to the Cloud if there is Internet connection between them. In addition, fog nodes connect with each other forming an overlay network based on their physical proximity. The Fog network adopts an unstructured P2P architecture. P2P networking enables communication between Fog nodes without the support of the Cloud. This will alleviate the heavy traffic load of the centralized Cloud during and after a disaster. Some of the overlay connections maybe interrupted caused by the disaster. We utilize moving objects (e.g., human, vehicle, drone) to support communication with interrupted communication connectivity.

b) Message Flow

- **Edge Node**. Most events (e.g., sensing report, service advertisement, service request) are generated from the edge IoT devices. Based on the event types and the network connection status of the IoT devices, the IoT device will decide whether the event message is sent to the Fog server or the Cloud server. For example, if there's no Internet connection or if the service or request is a local one, the IoT device will contact the local fog server. If the service or request is an urgent global one and the IoT device has Internet connections, the device may contact the Cloud server directly. It is possible that there is no direct connection from the edge device to either fog servers or cloud servers. In this case, the request/report will be stored locally in the IoT device and wait for opportunities to be transported to a fog server. For example, the message can be forwarded though a moving vehicle or human.

- **Fog Node**. A Fog server processes messages (such as filtering, computing, and analyzing) from edge devices and other fog servers. For some local service and request, the fog server may accept and serve these requests directly. For some community-based requests, the fog server cannot serve them directly, it may forward the requests to neighboring fog servers to process. For some other sensing reports and requests that need cloud involvement, the fog server may accumulate and forward these requests to the cloud for further processing.

  To maintain the P2P-based fog overlay network, periodically fog nodes probe their neighboring fog peers and exchange content-based information with them. If a fog node loses contact of a neighboring peer, it will find another peer to add to the neighbor list either though the cloud or other neighbors.

- **Border Node**. A moving node can pass the borders of different fog nodes; therefore, it is called a boarder node. A border node can store and forward data for other fog nodes and edge nodes when they are in its transmission range. A Border node will allocate storage with certain size to store messages and forward these messages to other moving nodes or to a fog node when it gets to the transmission range of the fog node.

c) Traffic Prioritization

By default, network resources are allocated on a first-come-first-served basis. During and after a disaster, network loads on a community network can reach up to several times higher than the normal traffic [25]. If important traffic receives equally poor access to resources as low-priority traffic, lots of urgent information (e.g., victims' distress message, first responder's command message) will be overwhelmed by tons of unimportant information. This may cause serious problems in disaster response. To solve this problem, the system distinguishes network traffic and gives different traffic different priorities. This scheme is based on the encoded ontological disaster information model presented in Section III.A. Message

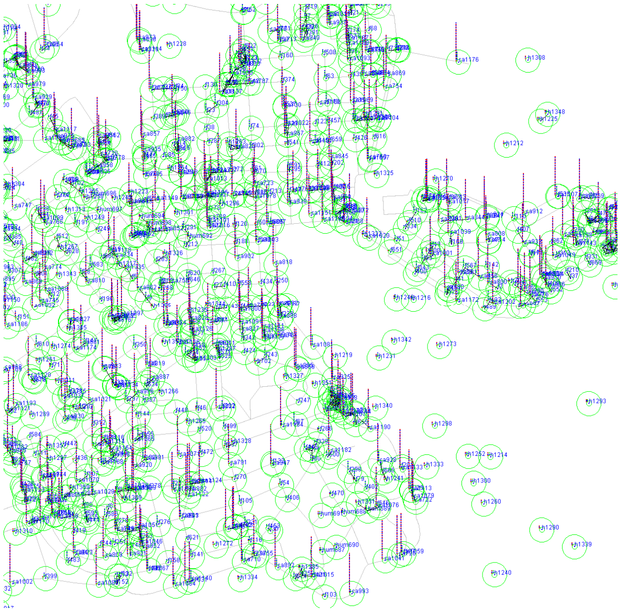

Fig. 4. Simulation GUI

forwarding is based on the priority info encoded in the message. Using this model, a policy decision point can reason over these ontology tags and infer the correct set of operations to forward the messages based on the semantic tag.

IV. EXPERIMENTS

We use the ONE simulator [26] to evaluate the performance of the proposed platform. The ONE is an opensource probabilistic DTN networking simulator. It offers tools for us to create the disaster environment and mobility scenarios that is close to reality.

A. Environment Setup

We use the default map (the Helsinki downtown map) as the simulation area. In the simulation environment, we deploy 1000 IoT nodes in a community where disaster happens. Among the 1000 IoT nodes, 140 are Fog nodes and 860 are Edge nodes. We assign edge IoT devices from 1-10MB of free RAM for buffering messages in these devices. Citizens, volunteers, and first responders may travel on foot or in cars. Fog devices have higher storage capacity ranging from 10MB-10GB are deployed at different regions in the community. Due to the impact of the disaster, 20%-80% of the network Infrastructure may get damaged. We simulate different types of mobile IoT nodes: pedestrians (with wearable devices and/or smart phones) moving with speeds of 2-3 miles per hour (mph) and pause times of 0-180 seconds), emergency vehicles (fire trucks, ambulance, police cars) moving at speeds of 30-45 mph, pausing for 0-500 seconds. Citizen's vehicles moving at speeds of 10–30 mph, pausing for 0-500 seconds. The simulation last for 6 hours. Fig. 4 shows a small part of the simulated area, in which the green circle represents each node's transmission range. The purple bar on top of every node represents a node's message queue.

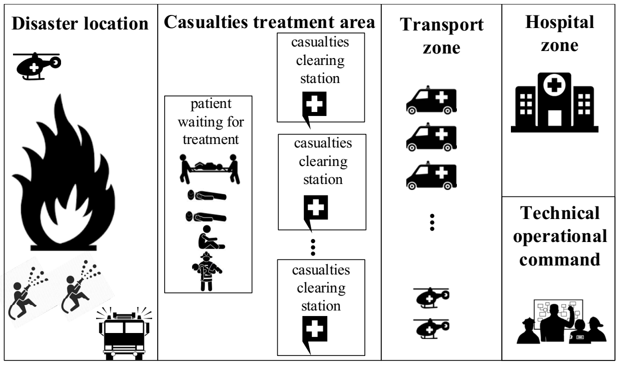
Disaster location
Casualties treatment area
casualties clearing station
patient waiting for treatment
casualties clearing station
casualties clearing station
Transport zone
Hospital zone
Technical operational command

Fig. 5. Separation room modeling of the disaster affected area

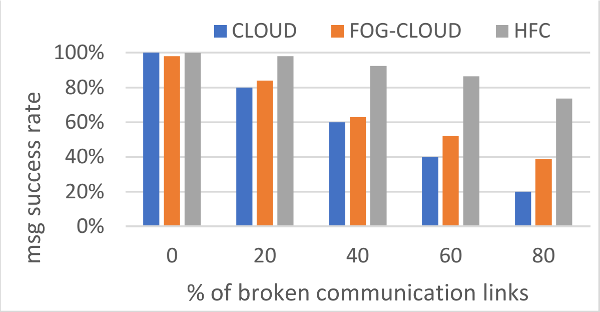

Fig. 6. Transmission success rate vs. percentage of broken network links

To model the traffic pattern in a disaster scenario, we follow the separation of the room method [25] and divide the affected community into areas: the incident location, patients waiting for treatment area, casualties clearing stations, the rescue vehicles parking point, and the technical operational command as shown in Fig. 5. Nodes can move inside these areas and from one area to another. Basically, these areas are set as Points of Interest (POIs). For each node group (e.g., ambulance, fire truck, volunteers) have different probabilities to access these POIs. For example, ambulance may have very high probability to travel between a disaster location and a hospital zone. Probabilistic Routing Protocol using History of Encounters and Transitivity (PRoPHET) [27] routing algorithm is applied based on the separation of these regions. Moving nodes adopt the shortest path-based movement model, SPMBM [28], in which nodes use Dijkstra's shortest path algorithm to calculate shortest paths from the current location to the destination location.

B. Experimental results

In the first set of experiments, we verify that the proposed *Human-Centric Fog Cloud-assisted IoT* platform (HFC) enhances network resilience, and thus improves the communication success rate. For this purpose, we compare the message success rate of three platforms (namely, traditional centralized cloud platform, classical Fog-Cloud platform, and HFC platform) under various degrees of damaged

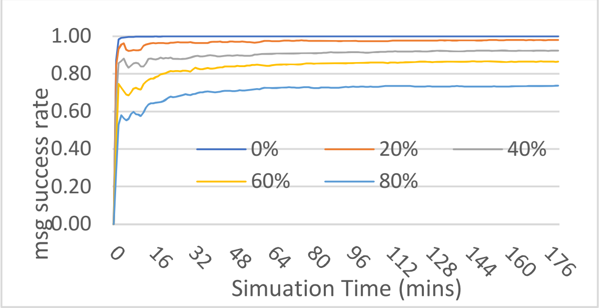

Fig. 7. Transmission success rate vs. simulation time for networks with various percentage of broken links

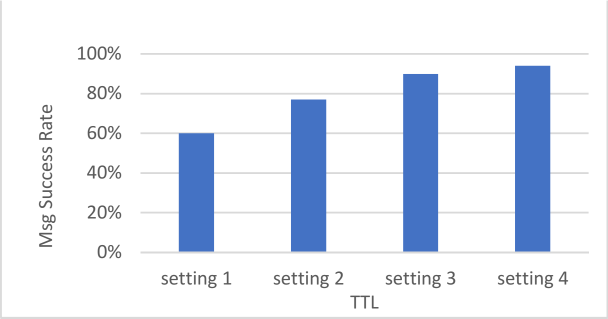

Fig. 8. Transmission success rate vs. percentage of broken network links

communication networks. As shown in Fig. 6, for an ideal network without any damaged network infrastructure, all three platforms may get message success rate close to 100%. When network damage increases, Our HFC dramatically outperforms centralized cloud platform and classical Fog-Cloud platform. The P2P-based Fog overlay network along with the delay and disruption-tolerant mobile gateway nodes coordinately improve the network connectivity. Fig. 7 shows the average message success rates of HFC during a 3-hour simulation time when network links are broken with different levels.

In the second set of experiments, we studied how the message's time to live (TTL) requirement affects the network performance. In these experiments, we have four different settings of messages' TTL. For each setting, each different message category accounts for 25% of the total number of messages in the experiment. The messages' TTL are as follows: Setting 1: 5 seconds, 10 seconds, 30 seconds, and 60 seconds; Setting 2: 10 seconds, 30 seconds, 60 seconds, and 120 seconds; Setting 3: 30 seconds, 1 minute, 1.5 minutes, 3 minutes; Setting 4: 3 minutes, 5 minutes, 10 minutes, 15 minutes. All of these experiments were conducted under a situation where 40% of network links are broken. As can been from Fig. 8, for smaller TTL setting e.g., Setting 1, the message success rate is lower. As we increase the TTL the message success rate increases. These experiments indicated that our HFC platform can achieve better message success rate when longer lifetime of data existing in a network is given.

In the third set of experiments, we verify that our semantics-based traffic priority mechanism that gives high priority to urgent messages improve the system performance. In this experiment, we simplify the semantics of messages. We only use message's TTL to represent its urgency. For comparison, two other traffic routing mechanisms are employed for comparison, PRoPHET and Epidemic that is a flooding-based traffic forwarding scheme. Both of these two routing mechanisms do not consider traffic semantics. As shown in Fig.9, our semantic traffic priority scheme improves system performance in terms of message success rate by prioritizing more urgent messages.

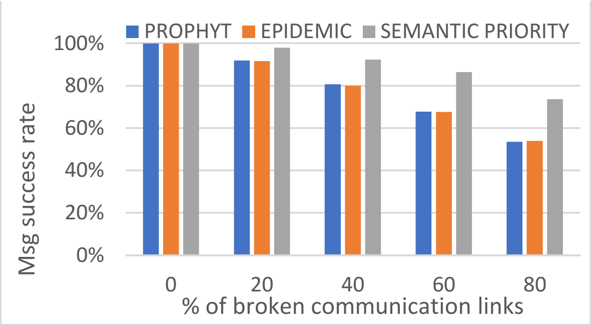

Fig. 9. Transmission success rate vs. percentage of broken network links

V. Conclusions

It is important to provide best-effort communication when disaster happens. In this paper, we propose an IoT-based cloud-fog communication platform that utilizes the power of IoT devices and human movements to enable delay and disruption-tolerant communication when normal network infrastructure is not available. Based on this communication platform, A community can better respond to disasters to support a sustainable livelihood by protecting lives, property and the environment.

Our proposed work poses a few fundamental design issues that set this project apart from current practices in developing communication systems for disaster use:

First, we emphasize the "human" factor in communication and computing, and propose a "human-centric" communication platform. Humans in this platform are not treated as passive users, but active computing components of the system. They are involved in the information collection, network formation and message propagation to make the system more scalable, robust and sustainable.

- Second, we combine existing communication infrastructure in a disaster-affected community with unstructured IoT overlay networking. In such a way, the proposed Fog computing platform removes the infrastructure and broadband-dependency limitation of classical Fog computing. Therefore, it can be quickly and easily deployed as compared to existing disaster communication systems.

- Third, we propose a multi-dimensional overlay routing approach that seamlessly integrates locality-preserving P2P routing with a delay-tolerant routing to realize a disruption-tolerant routing mechanism. We modelled the disaster area as different functioned regions. The delay-tolerant network routing algorithm is based on this model. It enables effective disruption-tolerant routings and makes the platform applicable to environment with a limited and intermittent internet connection.

- Fourth, we prioritize information delivery based on their critical levels. We develop an ontology-based disaster message information model and a light-weight ontology encoding scheme. A message's semantic type can be succinctly encoded. Network traffic is prioritized based on the coding. The most important information will be delivered with the highest priority.

The proposed platform has been evaluated with simulation experiments. The experimental results demonstrate the effectiveness of the platform. In the future, we plan to deploy real IoT devices including fog servers to a real world to further test its performance.

ACKNOWLEDGMENT

This work was supported by the National Science Foundation (NSF) under Div. Of Information & Intelligent Systems (IIS) with award number: 1722913.

REFERENCES

[1] H. Petersen, E. Baccelli, M. Wählisch, T. C. Schmidt, and J. Schiller, "The role of the Internet of Things in network resilience," in *International Internet of Things Summit*, 2014, pp. 283–296.

[2] C. Bach, A. K. Gupta, S. S. Nair, and J. Birkmann, "Critical infrastructures and disaster risk reduction," *Natl. Inst. Disaster Manag. Dtsch. Gesellschaft für Int. Zusammenarbeit GmbH (GIZ), New Delhi, 72p*, 2013.

[3] J. Freeman and L. Hancock, "Energy and communication infrastructure for disaster resilience in rural and regional Australia," *Reg. Stud.*, 2017.

[4] S. Katikala, "Google Project Loon," *Rivier Acad. J.*, 2014.

[5] P. P. Ray, M. Mukherjee, and L. Shu, "Internet of Things for Disaster Management: State-of-the-Art and Prospects," *IEEE Access*, 2017.

[6] J. Hayes, "Shaking earthquake warning up systems [Sensors Disaster Prevention]," *Eng. Technol.*, vol. 12, no. 10, pp. 66–69, 2017.

[7] D. G. Reina, M. Askalani, S. L. Toral, F. Barrero, E. Asimakopoulou, and N. Bessis, "A Survey on Multihop Ad Hoc Networks for Disaster Response Scenarios," *International Journal of Distributed Sensor Networks*. 2015.

[8] R. Chen, W. L. Jin, and A. Regan, "Broadcasting safety information in vehicular networks: Issues and approaches," *IEEE Netw.*, 2010.

[9] M. C. Chuang and M. C. Chen, "DEEP: Density-aware emergency message extension protocol for VANETs," *IEEE Trans. Wirel. Commun.*, 2013.

[10] S. M. Tornell, C. T. Calafate, J. C. Cano, and P. Manzoni, "DTN protocols for vehicular networks: An application oriented overview," *IEEE Commun. Surv. Tutorials*, 2015.

[11] F. R. Segundo, E. Silveira E Silva, and J. M. Farines, "A DTN routing strategy based on neural networks for urban bus transportation system," *Journal of Network and Computer Applications*. 2016.

[12] H. Suzuki, Y. Kaneko, K. Mase, S. Yamazaki, and H. Makino, "An Ad Hoc network in the sky, SKYMESH, for large-scale disaster recovery," in *IEEE Vehicular Technology Conference*, 2006.

[13] M. Iqbal, X. Wang, D. Wertheim, and X. Zhou, "SwanMesh: A multicast enabled dual-radio wireless mesh network for emergency and disaster recovery services," *J. Commun.*, 2009.

[14] P. Delir Haghighi, F. Burstein, A. Zaslavsky, and P. Arbon, "Development and evaluation of ontology for intelligent decision support in medical emergency management for mass gatherings," *Decis. Support Syst.*, 2013.

[15] A. Galton and M. Worboys, "An Ontology of Information for Emergency Management," *Proc. 8th Int. ISCRAM Conf.*, 2011.

[16] S. H. Jihan and A. Segev, "Humanitarian assistance ontology for emergency disaster response," *IEEE Intell. Syst.*, 2014.

[17] "W3C Emergency Information Interoperability Frameworks."

[18] M. Achichi *et al.*, "Results of the ontology alignment evaluation initiative 2017?," in *CEUR Workshop Proceedings*, 2017.

[19] P. Shvaiko and J. Euzenat, "Ontology matching: State of the art and future challenges," *IEEE Transactions on Knowledge and Data Engineering*. 2013.

[20] A. Doan, J. Madhavan, P. Domingos, and A. Halevy, "Ontology Matching: A Machine Learning Approach," in *Handbook on Ontologies*, 2004.

[21] J. Li, Y. Bai, N. Zaman, and V. C. M. Leung, "A Decentralized Trustworthy Context and QoS-Aware Service Discovery Framework for the Internet of Things," *IEEE Access*, 2017.

[22] S. Lu, L. Li, K. Y. Lam, and L. Jia, "SAODV: A MANET routing protocol that can withstand black hole attack," in *CIS 2009 - 2009 International Conference on Computational Intelligence and Security*, 2009.

[23] B.-C. Seet, G. Liu, B.-S. Lee, C. Foh, K.-J. Wong, and K.-K. Lee, "A-STAR: A Mobile Ad Hoc Routing Strategy for Metropolis Vehicular Communications," *Third Int. IFIP-TC6 Netw. Conf. Athens, Greece, May 9-14, 2004, Proc.*, 2004.

[24] L. Pelusi, A. Passarella, and M. Conti, "Opportunistic networking: Data forwarding in disconnected mobile ad hoc networks," *IEEE Commun. Mag.*, 2006.

[25] S. George *et al.*, "DistressNet: A wireless ad hoc and sensor network architecture for situation management in disaster response," *IEEE Commun. Mag.*, 2010.

[26] A. Keränen, J. Ott, and T. Kärkkäinen, "The ONE simulator for DTN protocol evaluation," in *Proceedings of the Second International ICST Conference on Simulation Tools and Techniques*, 2009.

[27] F. C. Lee and C. K. Yeo, "Probabilistic routing based on history of messages in delay tolerant networks," in *IEEE Vehicular Technology Conference*, 2011.

[28] A. B. Altamimi and T. A. Gulliver, "On routing protocols using mobile social networks," *Int. J. Wirel. Mob. Comput.*, 2013.