HELPER: Heterogeneous Efficient Low Power Radio for enabling ad hoc emergency public safety networks

Jithin Jagannath*, Sean Furman, Anu Jagannath, Luther Ling, Andrew Burger, Andrew Drozd

ANDRO Advanced Applied Technology, ANDRO Computational Solutions, LLC, Rome, NY 13440, United States

ABSTRACT

Natural and man-made disasters have been causing destruction and distress to humanity all over the world. In these scenarios, communication infrastructures are the most affected entities making the rescue and emergency response operations extremely challenging. This invokes a need to equip the affected people and the emergency responders with the ability to rapidly set up and use independent means of communication. Therefore, in this work, we present a complete end-to-end solution that can connect survivors of a disaster with each other and the authorities using a completely self-sufficient ad hoc network that can be setup rapidly. Accordingly, we develop a Heterogeneous Efficient Low Power Radio (HELPER) that acts as a WiFi (Wireless Fidelity) access point for end-users to connect using website application developed by us. These HELPERs then coordinate with each other to form a LoRa based ad hoc network. To this end, we propose a novel cross-layer optimized distributed energy-efficient routing (SEEK) algorithm that aims to maximize the network lifetime. This aspect is critical especially in energy constrained scenarios after a disaster.

To prove the feasibility of the solutions, we prototype the HELPER using WiFi enabled Raspberry Pi and LoRa module that is configured to run using Li-ion batteries. We implement the required cross-layer protocol stack along with the SEEK routing algorithm and develop a website application that an end-user can avail to connect using any device such as smartphones, tablets, laptops etc. We have conducted demonstrations to establish the feasibility of exchanging of text messages over the HELPER network, live map updates, ability to send distress messages (like 9-1-1 calls) to authorities. In the context of authorities, we have shown how they can leverage this technology to remotely monitor the connectivity of the affected area, alert users of imminent dangers and share resource (water, food, first aid) availability information. We have also conducted an extensive numerical evaluation of SEEK algorithm against a greedy geographical routing algorithm using the HELPER testbed. Results showed up to 53% improvement in network lifetime and up to 28% improvement in throughput. Overall, we hope this technology will become instrumental in improving the efficiency and effectiveness of public safety activities.

1. Introduction

In the past years, several lives have been devastated by hurricanes, tsunami, floods, earthquakes, and other natural disasters. Similar natural and man-made disasters are undesirable but sometimes unavoidable [2,3]. Even if the disasters may vary in intensity, nature, and duration of occurrences some of the challenges faced during this period are similar. In these scenarios, one of the most critical infrastructures affected is often communication networks [4,5]. Today's world is heavily reliant on wireless communication. This is evident from the fact that 99% of the population is covered by at least 3G network in the United States [6]. Similarly, authorities like Emergency Response Center (ERC) setup by agencies like Federal Emergency Management Agency (FEMA) are heavily reliant on wireless communication for information gathering, command, and control. There are also several disaster alert system [7] that relies on wireless communication to relay the message. For example, Grillo sensor network (Mexico) is a network of seismic sensors that will sense and alert local users about the seismic activity. MyShake (U.S) is a mobile application based solution which leverages the accelerometers of smartphones to detect seismic vibrations and sent information for analysis to the Berkeley Seismological Laboratory for a final check before alerting the user. Citizen Flood Detection (U.K) network is based on sensors installed under water bridges to keep a tab on the water levels using echolocation and update the flood maps while alerting the connected users over the Internet. Clearly, wireless communication is an essential component to

* Corresponding author.
E-mail addresses: jjagannath@androcs.com (J. Jagannath), sfurman@androcs.com (S. Furman), ajagannath@androcs.com (A. Jagannath), lling@androcs.com (L. Ling), aburger@androcs.com (A. Burger), adrozd@androcs.com (A. Drozd).

maintain connectivity for such alert systems, Emergency Responders (ERs) as well as affected individuals when traditional infrastructures like cell towers are affected or unavailable.

Several steps have been taken to enable wireless communication between ERs in such situations [8–13] with an objective to improve interoperability, reliability, and accessibility. In comparison, there are few solutions designed to connect the affected survivors to the ERs and the ERCs [14–18]. Further limited are the solutions that have been implemented and prototyped to establish feasibility [14,16,18]. This aspect of emergency communication is critical to enable rapid assistance, recovery and ensure the safety of the people in the affected area. Realizing this gap, Mozilla and National Science Foundation (NSF) launched Wireless Innovation for a Networked Society (WINS) challenge to enable rapid off-the-grid connectivity during the times of disaster. Motivated from the challenge, in this work, we focus on designing a solution (hardware and software) that can be deployed by civilians (in their households) and ERs (roadside or other locations) to establish an infrastructure-less network that enables communication between End Users (EUs), ERs, and ERC during the aftermath of a disaster.

There are several challenges and requirements that have to be considered to enable such technology that can be accessed by everyone in an emergency scenario. Since there is a high probability that pre-existing infrastructure like base stations, cables etc. may be partially or completely damaged during the disaster, the solution proposed for emergency communication must be self-sustained. The solution should be readily accessible to EU such that there is no learning time or contingencies for them to be connected to the network. In other words, it should be as simple as people walking into an airport terminal and connecting to a Wireless Fidelity (WiFi) Internet network within seconds. There might also be a shortage or absence of electricity during this period leading to the demand for energy efficient solution. Another critical aspect will be the ease of deployment and cost associated with the technology and the coverage it provides. Since the topology of the target area may vary from tens to thousands of km^2 based on the magnitude of the disaster, the network must be designed in a distributed manner to ensure scalability. Due to the ad hoc nature of the network, there could arise network holes which may isolate parts of the network. One way to mitigate this problem is by deploying dense networks where density is defined as the average number of neighbors for each node in the network. This can be accomplished by using a physical layer solution that provides extremely long range links while maintaining energy efficiency. Finally, an ideal solution should be portable, low cost and energy efficient such that large networks can be deployed and sustained within a short period of time.

In this paper, we develop a Heterogeneous Efficient Low Power Radio (HELPER) ad hoc network for enabling emergency wireless communication as shown in Fig. 1. The proposed end-to-end ad hoc networking solution and supporting software is capable of establishing an independent, low cost, lower power wireless network for off-the-grid users during the aftermath of a disaster. One of the objectives of this work is to restrict the cost of the proposed device as much as possible such that each household can have one in their emergency kit and easily set it up when other communication infrastructures are disrupted. These HELPERs will form a wireless ad hoc network connecting users among themselves and to a ERC. The goal is not to provide a network with the highest throughput or minimize delay rather maximize sustained connectivity through energy efficient operation and provide key services. These services will include text and voice messages within the neighborhood, the ability to share resource information (water, food, gas, electricity, and internet) and the ability to send distress messages to the ERC. On the other hand, the ad hoc network will also be used by the ERC to remotely monitor the connectivity of

the affected area and send alerts regarding imminent dangers to the connected EU.

2. Related work

The need for a robust communication system during the recovery period after a disaster is evident from the previous discussions. Accordingly, several wireless communication technologies have been developed for public safety [9,19]. TETRA [11] is a telecommunication standard for the private mobile digital radio system that provides an interoperability standard for equipment from multiple vendors. The services provided include voice calls (individual call, group call, broadcast call) with a data rates from 2.4 kbit/s to 28 kbit/s. TETRA Release 2, known as TETRA Enhanced Data Service (TEDS) [13] has been deployed in United Kingdom [12] consisting of 3000 base stations providing national coverage. Similarly, in the United States, Project 25 (or APCO 25) is a standard setup for public safety communication by Telecommunication Industry Association (TIA) that ensures interoperability, spectral efficiency and is gaining acceptance worldwide for public safety application [10]. The protocol supports encrypted communication with a range of few km with data rates up to a maximum of 9.6 kbit/s. Additionally, there are several instances where commercial cellular wireless communication systems have been used as an emergency network. For example, Federal Communications Commission (FCC) in a white paper [20] and the authors of [21] recommends an approach for public safety broadband communications that leverage the advantages of Long-Term Evolution (LTE) technologies. All these approaches mentioned above requires fixed infrastructure, which can be significantly degraded or destroyed during the disaster rendering these services infeasible. An alternative solution that does not require pre-existing on-ground infrastructure is the use of satellite networks. The satellite network can provide access to mobile or fixed terminals using various frequency bands including C-Band and Ku Band. While the fixed terminal can achieve up to 1.5 Mbits/s data rate, the mobile terminals can achieve 256 kbit/s [9]. Another approach to sustaining communication when on-ground infrastructure is damaged is airborne communication using avionic communication through helicopters. The traditional avionic communication is in the Very High Frequency (VHF) band and can be used in three main configurations [22], (i) the system deployed as an aircraft repeaters, (ii) a base transceiver station on an aircraft or (iii) a complete system on an aircraft. Avionic communication is not cost-effective but is generally used after a large natural disaster in a rural area that does not currently have any alternative. Overall, these solutions are designed for ERs and cannot be cost-effectively extended to enable communication between EUs and ERs during an emergency. During the aftermath of the earthquake in Haiti, connectivity was enabled for 100 holding centers for displaced people using Worldwide Interoperability for Microwave Access (WiMAX) and WiFi [23]. WiMAX was also used during the 2004 tsunami in Indonesia and after hurricane Katrina in the Gulf Coast in 2005 [9]. This requires setting up a centralized WiMAX system to provide connectivity to EU. Any such centralized network will limit the scalability of the network in cases where the affected area is large.

As discussed earlier, it is essential to overcome the reliance on infrastructure and hence ad hoc networks have been identified as a preferable solution for such scalable networks [14,15,17,18,24–29]. The great east Japan earthquake and tsunami motivated authors of [14] to develop Device-to-Device (D2D) communication capabilities between smartphones to send emergency messages in areas with affected infrastructure. Accordingly, the authors develop a prototype and conduct an experimental evaluation in Sendai city which was one among the affected areas. They used Optimized Link State Routing (OLSR) and epidemic routing protocols to achieve commu-
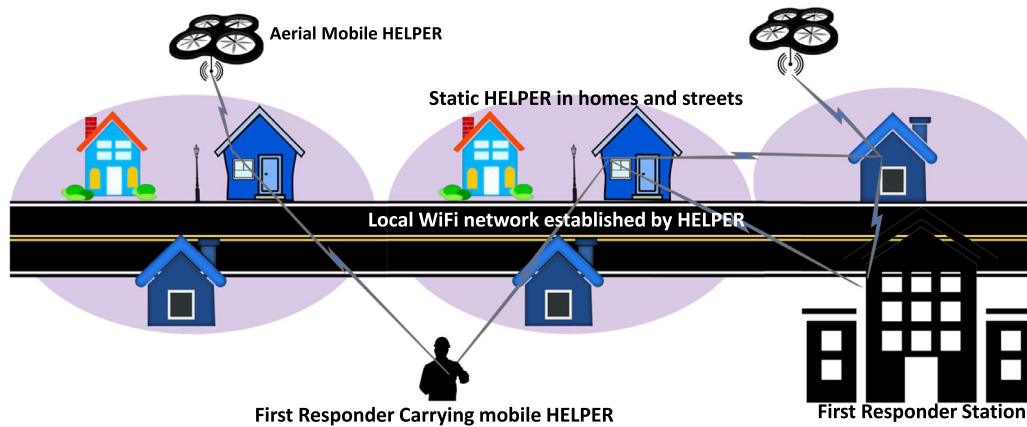
Fig. 1. HELPER development prototype.

nication between source and destination. In [30], the authors propose a location-aware wireless mesh network to assist ER in providing medical support. Zigbee technology that operates in 2.4 GHz band was used as the physical layer. An emergency and disaster relief system called Critical and Rescue Operations using Wearable Wireless sensors networks (CROW^2) is proposed in [26]. The authors propose an end-to-end system that employs an Optimized Routing Approach for Critical and Emergency Networks (ORACE-Net) routing protocol. ORACE-Net accesses every end-to-end link with regards to its quality (end-to-end link quality estimation) to perform multi-path routing. CROW^2 is designed specifically to offload data from the disaster area to ERC but does not provide any provisions for the EU to use the network to gather information for themselves. The authors of [17] propose to use smartphones to form an ad hoc network using a reliable routing mechanism. Reliable Routing Technique (RRT) [17] uses a broadcast based routing technique to improve reliability. Broadcasting every message to determine routes may lead to excessive and non-uniform energy consumption leading to some devices being excessively drained of energy causing network holes. In contrast, WIreless DEployable Network System (WIDENS) [27] is a European Project aimed to set up rapidly deployable emergency services. The system architecture uses a cross-layer interface to provide enhanced Medium Access Control (MAC) and physical layer interaction. It uses OLSR protocol at the network layer. OLSR is also used by [28] that aims to provide an efficient broadcast algorithm to reduce network overhead induced by the control packets. The proposed prototype in [28] uses a hybrid of satellite and WiFi connectivity to connect the ERC to the affected sites. In addition to considering energy-efficient routing, the choice of the physical layer will also be essential in determining the network lifetime and the operational feasibility in energy constrained scenarios. Therefore, the use of WiFi can be an ideal choice to connect to the EU but may not be the ideal choice to form ad hoc network that might have to span over several hundreds of km^2. The choice will always be a trade-off between energy consumption, range and data rate and hence should be made considering the requirement at hand.

Next, we look at some of the emergency ad hoc networking solutions that aim to achieve the required energy efficiency. In [29], the author emphasizes on the importance of energy efficient operation in emergency conditions and thereby designs Minimum Power Routing (MPR) protocol that chooses routes that require minimum power using Bellman-Ford algorithm which adapts to the changing channel conditions (noise and interference). While this may provide optimal energy efficiency in terms of energy consumed per bit delivered, this may not be the optimal routing strategy for maximizing the entire network's lifetime. In other words, this may lead to some nodes being over-utilized for routing packets in the network depleting these nodes of energy causing network holes. The authors of [18] propose TeamPhone that uses smartphones to form ad hoc networks using WiFi. TeamPhone uses opportunistic routing or Ad hoc On-Demand Distance Vector (AODV) for routing and propose to employ grouping technique along with a wake-up schedule to conserve energy. This sleeping technique can be adopted by emergency ad hoc network but cannot substitute an energy efficient distributed routing algorithm. An interesting framework is proposed in [15] that enables nodes to harvest energy from an undamaged base station (source) and then act as a relay to carry the information to an area that does not have direct access to the source. The authors propose an optimal communication route for networks during an emergency to minimize end-to-end disconnection and reduce energy consumption while introducing the concept of Radio Frequency (RF)-based energy harvesting. Clustering is an ideal choice for their framework as the energy harvesting and coordinated operation is assumed between nodes but in deployments where energy harvesting is infeasible, clustering may lead to uneven consumption of energy or require frequency re-clustering procedure that will eventually lead to larger overhead. We have summarized the above discussion in Table 1.

Therefore, in this work, we significantly extend [1] to develop an emergency ad hoc networking solution, HELPER Network with an objective to connect EUs of an affected community to each other and the responding authorities. The significant contributions are summarized below,

- We propose an end-to-end solution that includes Website Application (Web App) that connects EU's mobile devices to the HELPER using WiFi links. The HELPERs form an energy efficient ad hoc network using low power, long-range LoRa links to connect all EUs to each other and ERC.
- The proposed capabilities include resource information sharing, emergency distress messages to ERC, imminent danger alert from ERC to all connected EUs and the ability to send text and voice messages between EUs.
- To accomplish this, we design and implement a cross-layer protocol stack that is used by each HELPER to perform optimized routing by using information acquired from different layers.
- Additionally, we propose and implement a novel distributed energy efficient routing algorithm that aims to maximize the network lifetime.
- Finally, we prototype a portable, cost-effective and energy efficient solution to conduct proof-of-concept demonstration. We

Table 1
Summary of technology.

System	Users	Standard/ band	Design focus	Operation/ routing	Hardware eval
TETRA [11]	ER	TETRA Release 1	Interoperability	Centralized	Yes
TEDS [13]	ER	TETRA Release 2	Interoperability	Centralized	Yes
APCO 25 [10]	ER	Project 25	Interoperability	Centralized	Yes
Satellite [9]	ER	C & Ku	Remote Connectivity	Centralized	Yes
Avionic [22]	ER	VHF	Remote Connectivity	Centralized	Yes
Haiti [23]	EU	WiMAX and WiFi	Connectivity	Centralized	Yes
BRCK [16]	EU	3G or LTE WiFi	Connectivity	Centralized	Yes
Gomez et al. [21]	Both	LTE	Distributed LTE	Centralized and D2D	No
Nishiyama et al. [14]	EU	WiFi	Smartphone Relay	OLSR and Epidemic	Yes
RRT [17]	Both	Not Specified	Reliability	RRT	No
TeamPhone [18]	Both	WiFi	Energy Efficiency	AODV and Opportunistic	Yes
WIDENS [27]	ER	Enhanced 802.11	Rapid Deployment	OLSR	No
Kanchansut et al. [28]	ER	WiFi and Satellite	Efficient Broadcast	OLSR	Yes
Chandra et al. [30]	ER	Zigbee	Energy Efficiency	Zigbee Mesh	Yes
Ali et al. [15]	EU	LTE	Energy Harvesting	D2D	No
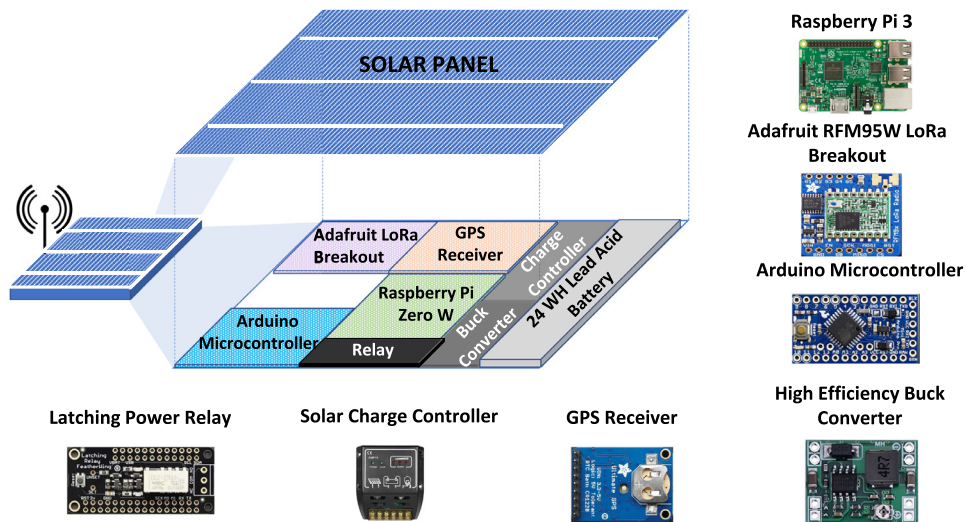
Fig. 2. Static HELPER design.

use the six-node network to conduct extensive numerical evaluations of the proposed routing algorithm.

3. Concept of operation

3.1. Types of HELPERs

As shown in Fig. 1, we envision three types of HELPERs in the proposed network. These three HELPERs have the same capabilities in terms of wireless communication and networking but differ in the context of mobility, size, and survivability (duration of operation).

Static HELPER (SH): These HELPERs are reasonably portable yet considered relatively static as they are envisioned to be operated in a relatively fixed location (terrace of household, hospitals, roadside, public buildings etc.) with abundant sunlight or other energy sources. These HELPERs have the largest battery and solar panel that supports 24/7 operation. The design goal of the SH is to survive at least a day or two in the absence of sunlight and to extend for multiple days in presence of ample sunlight. The components used to prototype the proposed SH is depicted in Fig. 2. The main board used is a Raspberry Pi (RPI) 3b [31]. The choice was motivated from the low cost, size, and large open community support for RPI development. Additionally, it is enabled with WiFi (802.11 b/g/n) and will be set up to operate as an access point for EU. The WiFi link provides a comparatively lower range of coverage but is

an essential choice taking into account the widespread usage of WiFi by today's devices (smartphones, tablets, laptops etc.). This will ensure a seamless connection from a users point of view due to the abundant familiarity in accessing WiFi.

To establish networks covering larger areas, we choose LoRa [32] to set up low power, long-range links (2 − 5 km in urban areas and 15 km in suburban areas [33]) between HELPERs. LoRa is emerging as a viable communication choice for Internet of Things (IoT) devices that strive to operate at low power yet achieve long-range. The long range of LoRa ensures dense networks because a larger number of these nodes may be deployed within the communication range. This ensures the availability of multiple routes to choose from such that energy based optimization can be performed. A Global Positioning System (GPS) receiver is also attached to the RPI. This will be used to acquire the location information to perform geographical routing and to indicate the location of the node when assistance needs to be dispatched. Lastly, we propose to use an Arduino microcontroller and power relay attached to the RPI. The Arduino and power relay can be used to put the system in a deep sleep mode to conserve energy when multiple devices are deployed in the same vicinity. Once the RPI has decided to sleep for a given duration of time, the signal is sent over to the Arduino board which in turn shuts the power relay which disables the RPI. The Arduino board uses its power management watchdog timers to keep the RPI in a low energy state for the entire sleep duration. The Arduino will then flip the power relay back on effectively
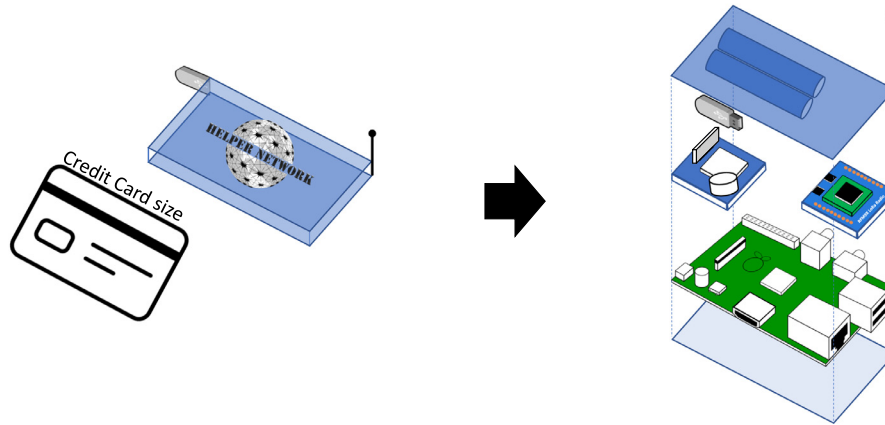
Fig. 3. Envision final mobile HELPER design.

restarting the RPI. This mechanism will enable energy preservation and cooperation between multiple HELPERs in the same vicinity.

Next, we discuss the envisioned power supply for the prototype. We have estimated that the system requires about 1Wh to run at a 25% duty cycle. Choosing the appropriate solar panel relies on current weather conditions, amount of daily sunlight and historical weather trends for that location. In a mostly sunny area, a 3 or 4 W solar panel would be sufficient for 24/7 operation yet some areas may need a 10–15 W solar panel. The solar panel will be attached to a solar controller. This solar controller has a 24 Wh lead-acid battery and a high-efficiency buck converter to the load. The 24 Wh battery will last an entire day on a full charge. The more expensive buck converter can be used to supply power to the system because a buck converter can commonly get up to 90% efficiency, whereas a simple voltage regulator would have a 59% loss of power coming from 12 V down to 5 V. A factor that will affect the portability of the design is the battery system designed to operate 24/7. Li-Ion Batteries are the ideal choice to ensure portability but needs a complicated charging and discharging circuit whereas Lead-acid batteries have a simpler charge-discharge circuit but tend to be on the heavier side. The prototype designer can make a studied choice based on the deployment requirements.

Mobile HELPER (MH): We envision two versions of MHs. The first version that is indented to go on vehicles will not require a battery as it will draw energy from the vehicle itself. This version will have the smallest form factor but will have to operate within the vehicle itself. The second version of the MH will have the same design as the SH with the exception of eliminating solar panel and using a smaller battery to ensure more portability for ER and EU. The MH will use lightweight 18650 Li-Ion batteries as a power source with a boost converter to up the voltage to the required 5 V input of our devices. Li-Ion batteries have a much higher energy density compared to lead-acid batteries. Since the batteries will be easily swappable, discharged batteries can be removed to recharge while other charged ones can be used in deployed nodes. As the batteries do not need to supply a load while being charged, any off-the-shelf charger can be used. There is a trade space between the battery capabilities with energy storage and max amperage draw. For the MH, two 3000 mAh 18650 batteries will be used to power the device. Given the 1 Wh load from the device and two batteries that supply around 22 Wh, the ER devices will be operable for a total of at least $8 - 9$ h (22 h in ideal scenarios considering only 25% duty cycle and no loss) before needing to be recharged. In this work, we prototype six MHs for our experiments which we will see in the later part of this paper.

Fig. 4. Aerial HELPER (Erle Copter).

Aerial HELPER (AH): When the network is set up, based on accessibility, there might be parts of the network that is disconnected due to node failure, locally disruptive channel conditions or uneven distribution of HELPERs during the setup period. We refer to these gaps in the network as network holes. The goal of an AH is to identify these isolated HELPERs and act as a temporary sink node that retrieves information. The isolated HELPER can upload the information about the users currently connected to the given HELPER and this information is carried by the AH to the ERC. The AH also indicates which locations need more HELPERs to be deployed in order to fill the network holes. The AH are the most costly and least energy efficient (taking into consideration the energy for flight) among the three types of HELPERs but is required in critical scenarios where road access might be completely cut-off. More about the different deployment scenarios are discussed in the next section. To meet the needs of the AH, we propose to use an open style drone (see Fig. 4) that allows for the flexibility of programming a completely autonomous drone while minimizing the cost. The Erle Robotics Drone Kit also has the added benefit that the Erle Brian [34] uses RPI. Therefore, the development toolchain is the same as the MH and the SH. Therefore, these AHs will use the battery and RPI that are inherent to the drone itself.

3.2. Deployment scenarios

We divide the deployment scenarios into three major cases based on accessibility and available resources which are discussed in detail below.

Scenario I (Full Accessibility and resources): In the first scenario, accessibility is not restricted and the ERs have all the
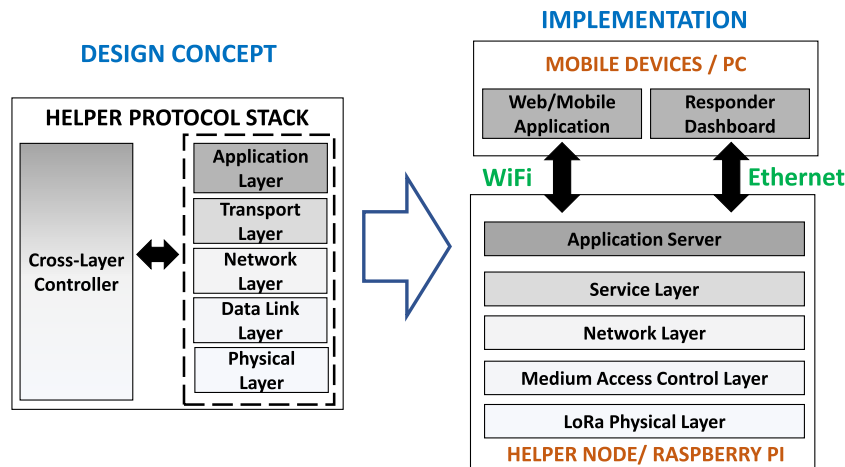
Fig. 5. HELPER's cross-layer protocol stack design and implementation.

required resources (vehicles, drones and a large number of ERs) to set up a HELPER network. In this scenario, the SHs can be placed in a strategical manner to ensure full coverage of the affected area with minimum deployment cost. The placement of the SHs will be under direct sunlight to enable 24/7 operation. The deployment of SHs can be relatively sparse as the HELPERs can be arranged optimally to ensure full coverage and extended lifetime. The MHs and AHs will also supplement this network during the rescue operation. AHs will use BEACON packets (more about BEACON packets is described in Section 4) while flying over the affected areas to determine the HELPERs that might need replacement due to depletion of battery or absence of sunlight. Overall, there is more control over deployment of HELPERs and hence easier to provide full coverage and repair disconnected parts of the network.

Scenario II (Limited Accessibility): In the second scenario, the accessibility is highly limited during the initial stages. This implies that there will be limited options (few vehicles with HELPERs during initial stages) to deploy HELPER network. Therefore, a denser deployment of SHs will be necessary to ensure maximum connectivity and network lifetime. These HELPERs may be present in the emergency kits of households, or other buildings before the disaster strikes. Additionally, a large number of supplementary SHs can be deployed via air. The role of AH will also be critical in these scenarios to determine network holes (areas without coverage or isolated HELPERs). When isolated HELPERs are determined, AH will act as a temporary sink and will fly back to the ERC with this information. This will enable ERs to have access to survivor information in isolated areas and prepare rescue efforts. The AHs will also be able to plug the detected network holes by promoting ERC to deploy HELPERs to provide complete network connectivity.

Scenario III (Limited Accessibility and resources): In the third scenario, the assumption is the lack of access and resources. There is no availability of the costlier AH. Since the proposed SH and MH is highly cost-effective, multiple HELPERs can be deployed in a dense manner such that maximum area is covered for connectivity. The dense network will operate in an ad hoc manner bolstered by the proposed routing algorithm that aims to maximize the network lifetime.

4. HELPER design and implementation

In this section, we discuss the overall HELPER framework consisting of the communication protocol stack, novel routing protocol and discuss the corresponding packet structure, and packet handling while determining the necessary interactions between differ-

Table 2
Heterogeneous wireless link parameters.

Features	WiFi (802.11 b/g/n)	LoRa
Frequency range	2.4 GHz	915 MHz
Bandwidth	20 MHz − 40 MHz	7.8 kHz − 500 kHz
Transmission range	Medium	High
PHY techniques	DSSS, OFDM, MIMO-OFDM	CSS, FSK [44]

ent layers to enable a cross-layered approach to optimize the network performance.

4.1. HELPER cross-layer protocol stack

The significance of cross-layer optimization in wireless communication has been widely studied [35–40] across various domains and optimization problems. Identifying the advantages of cross-layer optimization there has been some work recently to develop cross-layer platforms to facilitate these technologies [41–43]. Fig. 5 depicts the design concept of a HELPER and how the design is currently implemented in a modular manner on the selected platform. We will discuss the design considerations for each of these layers in detail in the upcoming sections.

4.1.1. Physical layer

As discussed earlier, HELPER is enabled using two wireless technologies WiFi (802.11 b/g/n) and LoRa which gives it the heterogeneous nature of operations. The prominent reason behind using both these well established wireless technologies are as follows, (i) WiFi is ubiquitous in today's devices and this will ensure seamless access for EUs, (ii) LoRa is becoming a prominent communication technology enabling IoT devices that requires low power, long-range wireless links and (iii) it is extremely cost-effective to use off-the-shelf physical layer to ensure low (Size, Weight, and Power) (SWaP). The features of the physical layer are shown in Table 2. Though we use both WiFi and LoRa as wireless technologies to enable HELPER, only LoRa can be considered as the physical layer from the point of view of the ad hoc HELPER network. WiFi can be considered as the interface between the application layer and the service layer of the HELPER's protocol stack as shown in Fig. 5.

The HELPER network stack interfaces to the LoRa radio module using the Radio Head and BCM2835 C++ libraries. This interface is implemented in an Application Programming Interface (API) that is utilized by the MAC layer to send and receive packets over-the-air.
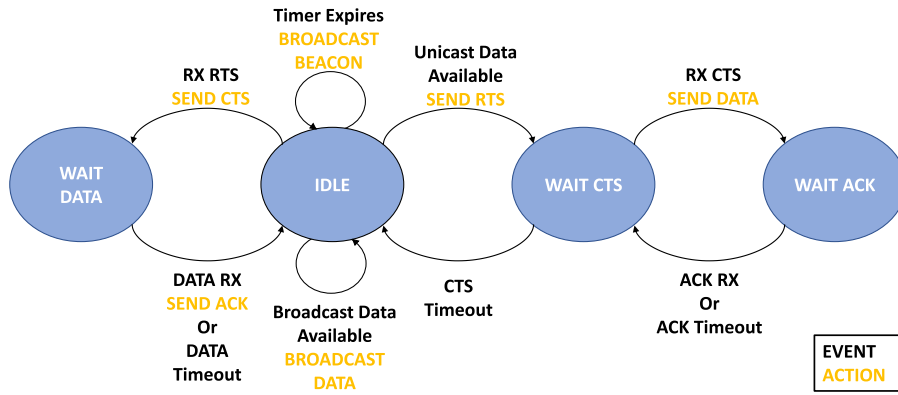
Fig. 6. FSM of the MAC protocol.

The API also provides functionalities for reading and writing radio parameters.

4.1.2. Data-link layer

One of the primary function of the data-link layer is negotiating the medium access. In the proposed HELPER network, just as we discussed the physical layer, we have two levels of medium access, (i) local WiFi links between HELPER and devices (phone, laptop, and tablets) of users and (ii) LoRa links between different HELPERs that form the ad hoc network. We use the standard off-the-shelf MAC protocol employed by WiFi (IEEE 802.11) to allow multiple users access to HELPERs within the local area. We have implemented a similar Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA) based MAC protocol to setup multihop ad hoc network using LoRa with the intention to utilize the Channel Activity Detection (CAD) offered as a hardware feature on the RF95 LoRa. CAD is a valuable tool since LoRa uses spread spectrum transmissions. The spread spectrum is known to operate at low signal to noise ratio making traditional approaches like power detection with Received Signal Strength Indication (RSSI) unreliable. CAD helps to detect if there is ongoing transmission in the channel chosen within two symbols according to the RF95 hardware documentation. This feature can be leveraged to implement the MAC protocol for the multihop LoRa based network. Since WiFi and LoRa operate on different parts of the Industrial, Scientific and Medical (ISM) band (as shown in Table 2) they do not interfere with each other's operation.

Therefore, the data-link layer shown in Fig. 5 contains the control logic used by HELPERs to negotiate access to the wireless medium. It houses the Finite State Machine (FSM) used to implement the CSMA/CA like MAC protocol used by the HELPER network. As seen in Fig. 6, the traditional RTS (request-to-send), CTS (clear-to-send) handshake is used before transmitting a unicast data packet. The successful reception is followed by the receiver transmitting ACK (Acknowledgement) packet. In addition to this, a BEACON packet is broadcasted periodically by a HELPER that has not transmitted any control packet for a pre-determined duration. Each of these control packets (RTS, CTS, BEACON) carry information including, instantaneous backlogged queue length, residual energy, location and the observed goodput per neighbor, which we refer to as Optimization Assisting Information (OAI). In this manner, each HELPER gathers OAI from its neighbor and uses this updated information to perform optimized energy efficient routing (which will be discussed in detail in upcoming sections). Therefore, in implementation, the MAC layer continuously monitors the physical layer receive queue for inbound messages and handles them according to the current state of the FSM. All OAI received from the control packets are used to update the inputs to perform cross-layer optimization. Once the network layer has performed the required optimization and chosen the optimal next hop, the MAC layer negotiates the medium and forwards the data packets.

4.1.3. Network layer

The network layer is responsible for packet queuing and routing. As shown in Fig. 5, the network layer interfaces to the service layer and data-link layer. When packets are received from the service layer or data-link layer, the network layer encapsulates/decapsulates network layer fields as needed and places packets in the appropriate queue. The network layer maintains two transmit queues: one for priority traffic and a second for best effort traffic. Each packet is sorted into one of these queues depending on application message type and other fields in the header. The energy efficient routing algorithm is used for routing unicast packets which ensure maximum network lifetime. Every broadcast packet contains a Hop-To-Live (HTL). Packets with HTL greater than zero are broadcasted by the receiving HELPER. In addition, the network layer uses shared memory to manage neighbor lists and OAI information in order to perform optimized cross-layer routing. The network layer also has access to HELPER's current GPS location via libgps and stores it in its local OAI data which is then shared with neighboring HELPERs.

A critical aspect of the proposed HELPER network is its energy efficiency. Since the majority of energy consumption is attributed to the transmission of packets, routing becomes a significant aspect of the design. Accordingly, we define a utility function that takes into account energy efficiency, goodput and a measure of congestion (using differential backlog) to formulate an optimization problem with the objective to maximize network lifetime while maintaining reliable connectivity. Since the goal is to deploy a scalable network, we formulate a distributed version of the optimized routing algorithm such that each node can make its own routing decision based on the limited OAI gathered from its neighborhood.

System Model: To design the routing algorithm, we consider the most constrained scenario (scenario III of Section 3.2) which has restricted access and minimal resources. Accordingly, consider a dense multihop wireless ad hoc network comprising of several N HELPERs (which we refer to as nodes in this section) modeled as a directed connectivity graph $\mathcal{G}(\mathbb{N}_{net}, \mathbb{E})$, where $\mathbb{N}_{net} = \{H_1, H_2 \ldots, H_N\}$ is a finite set of wireless transceiver (nodes), and $\mathfrak{L}(i, j) \in \mathbb{E}$ represents unidirectional wireless link from node H_i to node H_j (for simplicity, we also refer to them as node i and node j). We assume $\mathcal{G}$ is link symmetric, i.e., if $\mathfrak{L}(i, j) \in \mathbb{E}$, then $\mathfrak{L}(j, i) \in \mathbb{E}$. Each node is assumed to have the transmission range R based on the chosen transmit power P_t. As seen before, all the nodes are equipped with GPS and therefore the location (longitude/latitude) coordinates are known. The knowledge of node locations is important for a geographical/position based routing algorithms proposed in this work. In Fig. 7, the nodes within the transmission radius of
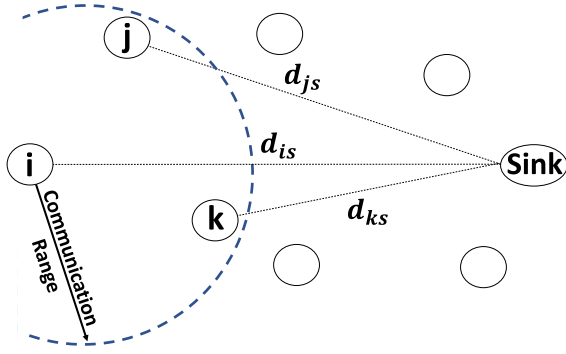
Fig. 7. Network diagram.

i will constitute its neighbors. Let us denote the set of neighboring nodes of node i as $\mathbb{NB}_i = \{j, k\}$ and the sink (ERC) node as s. The location of s can be predefined in every node or as in our case, this information is flooded at the time of network setup. In this formulation, we consider packets that have to be transmitted from node i to sink s but this can be extended to any source-destination pair.

The distance between any two nodes i and j is represented by d_{ij}. If a node j exist within the transmission range of node i, there exists a link $\mathfrak{L}(i, j)$, i.e., a wireless communication link $\mathfrak{L}(i, j)$ exists when $d_{ij} \leq R$. The power consumed over $\mathfrak{L}(i, j)$ or the power required by the source node (i) to transmit to a neighboring node (j) is denoted by P_{ij}. The initial and residual battery energy at node i can be denoted as E_0^i and E_r^i respectively. Every node maintains a queue that holds the outbound packets. Let q_i represent the instantaneous number of packets retained in the queue of node i, also called the queue backlog. The transmission bit rate and Bit Error Rate (BER) over $\mathfrak{L}(i, j)$ are denoted by R_b^{ij} and e_b^{ij} respectively.

Routing Algorithm: The proposed *diStributed Energy Efficient bacKpressure (SEEK)* routing algorithm utilizes the geographic information of nodes, differential queue backlog, residual battery energy and transmission power levels to compute the optimal next hop. In this section, we will present a formal derivation of our utility function ($\mathcal{U}_{ij}$ with respect to $\mathfrak{L}(i, j)$) and formulate the network optimization problem.

The utility function considers the following parameters associated with potential next-hop; (i) proximity to sink, (ii) differential queue backlog, (iii) residual battery energy, (iv) power required to transmit over the link and (v) the corresponding link throughput. This information is gathered from traditional control packets like RTS, CTS and BEACON packets. As discussed earlier, these packets will contain updated OAI and a PROBE field. In a realistic scenario, the real-time Signalto-Noise Ratio (SNR) is unknown to the device. Therefore, measures derived from SNR estimated based on a radio propagation model might not be a suitable guideline for signifying transmission reliability over a link $\mathfrak{L}(i, j)$. We, therefore, propose to use a PROBE field in the control packets to perform link probing [45,46]. The PROBE field would contain a bit sequence known by the nodes in the network. Upon receipt of the control packet, each node will compute effective throughput (goodput) measure in bits per second (bps). In our analysis, we prefer to use the term goodput to signify the effective number of bits successfully received. For example, once node i receives a control packet from node j, it will compute the corresponding goodput measure (G_{ij}) with respect to $\mathfrak{L}(i, j)$ and transmission strategy, T_{ij} (which includes choice of R_b^{ij} and P_{ij}). The energy efficiency of a given link can be expressed as a ratio between goodput and transmission power as [47],

$$\eta_{ij} = \frac{\mathrm{R}_b^{ij}\left(1 - \mathrm{e}_b^{ij}\right)}{\mathrm{P}_{ij}} = \frac{\mathrm{G}_{ij}}{\mathrm{P}_{ij}}$$

where η_{ij} gives the measure of number of bits successfully transmitted over $\mathfrak{L}(i, j)$ per Joule of transmission energy. Another key factor that needs to be considered in routing is the differential queue backlog ($\Delta\mathcal{Q}_{ij} = \mathrm{q}_i - \mathrm{q}_j$) with respect to the source node (i) and next-hop (j) [48–50]. The queue backlog at the destination node is considered to be zero. Considering the queue backlog is necessary to mitigate congestion in the network and traditional backpressure algorithms has been shown to be throughput optimal [48]. Since achieving maximum throughput is not the sole objective of HELPER network, differential backlog is just one parameter in our utility function. The effective progress made by a packet can be represented as $d_{is} - d_{js}$. Choosing nodes that provide larger progress implies fewer hops to the sink node which in turn could lead to smaller energy consumption. Finally, to ensure uniform depletion of energy per node, we need to consider the E_r^j of potential next hops [51]. Therefore, we define our utility function as follows,

$$\mathcal{U}_{ij} = \eta_{ij}\left(\frac{\max\left[\Delta\mathcal{Q}_{ij}, 0\right]}{\mathrm{q}_i}\right)\left(\frac{d_{is} - d_{js}}{d_{is}}\right)\left(\frac{\mathrm{E}_r^j}{\mathrm{E}_0^j}\right), \forall j \in \mathbb{NB}_i$$

η_{ij} aims to improve the energy efficiency of the network. It is also interesting to note that the maximum value of $\mathcal{U}_{ij} = \eta_{ij}$ when each of the three normalized terms is 1. This implies that each of the other terms penalizes the utility function based on the instantaneous value. For example, a small differential backlog ($\mathrm{q}_i - \mathrm{q}_j < \mathrm{q}_i$) will dampen the value of $\mathcal{U}_{ij}$. Both $d_{is} - d_{js}$ and E_r^j will have similar effects on $\mathcal{U}_{ij}$.

The objective of the network is to maximize the summation of $\mathcal{U}_{ij}$ for all possible links $\mathfrak{L}(i, j)$ in order to maximize the overall energy efficiency of the network. This, in turn, will ensure reliable communication while maximizing the network lifetime (which is defined as the time when the first node in the network depletes its energy leading to a network hole). The optimization problem is subject to residual battery energy, queue backlog, bit error rate, and capacity constraints. This is formulated as Problem $\mathcal{P}_1$ shown below,

$\mathcal{P}_1:$ Given: $\mathcal{G}(\mathbb{N}_{net}, \mathbb{E})$, $\mathbb{G}$, $\mathbb{E}_r$, $\mathbb{Q}$

$\quad\quad$ Find: $\mathbb{NH}^*$, $\mathbb{T}^*$

Maximize: $\displaystyle\sum_{i \in \mathbb{N}_{net}} \sum_{j \in \mathbb{NB}_i} \mathcal{U}_{ij}$ $\hfill$ (1)

subject to:

$$\mathrm{R}_b^{ij} \leq C_{ij}, \quad \forall i \in \mathbb{N}_{net}, \; \forall j \in \mathbb{NB}_i \tag{2}$$

$$\mathrm{e}_b^{ij} > \mathrm{e}_{b*}^{ij}, \quad \forall i \in \mathbb{N}_{net}, \; \forall j \in \mathbb{NB}_i \tag{3}$$

$$\mathrm{E}_r^i > 0, \quad \forall i \in \mathbb{N}_{net} \tag{4}$$

$$\mathrm{q}_i \geq 0, \quad \forall i \in \mathbb{N}_{net} \tag{5}$$

where the objective is to find the set of next-hop and transmission strategy for all nodes in the network which can be represented as $\mathbb{NH}^* = [\mathrm{NH}_i^*]$ and $\mathbb{T}^* = [\mathrm{T}_{ij}^*]$ respectively, $\forall i \in \mathbb{N} \; j \in \mathbb{NB}_i$. In the above optimization problem $\mathcal{P}_1$, $\mathbb{G} = [\mathrm{G}_{ij}]$, $\mathbb{E}_r = [\mathrm{E}_r^i]$ and $\mathbb{Q} = [\mathrm{q}_i]$, $\forall i \in \mathbb{N}_{net}, \; \forall j \in \mathbb{NB}_i$ denote the set of goodput measure, residual battery energy and queue backlogs respectively. The constraint (2) restricts the total amount of data rate in link (i, j) to be lower than or equal to the physical link capacity. Constraints (3) impose that any transmission should guarantee the required BER. Finally, constraints (4) and (5) ensure the residual energy and queue backlog of each node will not have negative values. It can be seen that for solving the above optimization problem, nodes would

Table 3
Neighbor table of node i.

Node ID	Distance to destination	Queue backlog	Residual battery energy	Goodput
j	d_{js}	q_j	E_r^j	G_{ij}
k	d_{ks}	q_k	E_r^k	G_{ik}

HELPER RECEIVE
- Message Type
- Source
- Time To Live
- Source Location
- Payload

HELPER SEND
- Message Type
- Destination
- Time To Live
- Payload
- Reserve

HELPER PACKET
- Message Type
- Source
- Destination
- Time To Live
- Packet ID
- Sequence ID
- Sequence Total
- Beacon Data
- MAC Msg Type
- MAC Source
- MAC Destination
- Payload

Fig. 8. Packet formats.

require global knowledge of the network. Since the centralized optimization method is not a scalable solution, it motivates the need for a scalable distributed solution. We propose SEEK which will operate in a distributed fashion and enable each node to find the next-hop based on the local information available to them. Each node with a packet to transmit chooses an optimal next-hop and transmission parameters such that it maximizes its own local utility function. The probability of channel access will be controlled by utility based random backoff. This can be considered as a divide-and-conquer approach to solving the optimization problem in a distributed manner. Accordingly, every source node (i) will aim to maximize the utility function $\mathcal{U}_{ij}$ and select the optimal next-hop and transmission strategy as follows,

$$[j^*, T_{ij}^*] = \arg \max_j \mathcal{U}_{ij}, \forall j \in \mathbb{NB}_i \tag{6}$$

Each node will maintain a neighbor table with node parameters of its neighbors and will update the table as needed based on information from the control packets. Considering the scenario in Fig. 7, the source node i will listen to control packets and maintain a neighbor table as in Table 3.

The ERC (sink) collects and disseminates vital information like availability of resources, drop-off locations, emergency updates for EU among others. ERC needs to strategically flood this information in the network to enable all HELPERs to obtain the updated information. This implies the requirement to implement one of the energy efficient flooding technique that has been widely studied in literature [52,53].

4.1.4. Service layer

The Service Layer provides a common interface between HELPER applications and the lower layers of the protocol stack. This layer communicates to HELPER applications using local sockets and the Network Layer via direct function calls. Messages received from applications are translated from HELPER Send format to HELPER Packet format (shown in Fig. 8) and are passed to the network layer. Messages received from the network layer are translated from HELPER Packet structure to HELPER Receive format and are then passed to the application. In the implementation,

the Service Layer uses an MQTT messaging socket to communicate with the Web Application and messages are encoded using JSON. The Paho MQTT CPP and Rapid JSON libraries are used to implement the messaging to the application. The implementation is such that more application message types and message handling can be added in the future to expand the capabilities of the HELPER network.

4.1.5. Application server

Users can join the HELPER network by connecting to a HELPER node via a WiFi or LAN connection. The HELPER nodes are configured to act as a WiFi access point, allowing users to connect their smartphones, laptops, tablets, etc. to the network. A wired connection to a HELPER node is also possible and is utilized by the ERC. Each node runs a web server that hosts the web applications. Once connected to a HELPER node, a user can launch a web application. Currently, two web applications are developed, one of EU and the other for ERC. These are discussed in detail in Sections 4.4 and 4.5.

The web server interfaces to the HELPER network stack to send and receive data across the network. This interface is implemented using MQTT sockets. The web server and service layer both connect to the MQTT broker and setup two publish-subscribe channels. One channel is for sending messages from the web server to the HELPER stack and the other is for sending messages from the HELPER stack to the web server. The data sent on these MQTT sockets are in the Helper Send and Helper Receive formats of Fig. 8 and are encoded with JSON.

4.2. HELPER packet handling

The HELPER network consists mainly of two kinds of HELPERs; (i) HELPERs that are deployed in households, hospitals, and other building that residents (survivors of disaster) connect to and usually have limited power supply and (ii) HELPER that forms ERC and usually has an unlimited power supply. Accordingly, messages can be classified as EU messages and ERC messages. We describe each message types used by HELPER below,

4.2.1. End-user messages

Emergency HELP Messages: A HELP message is used to indicate that an individual is in need of immediate assistance. This is similar to or a substitute for a 9-1-1 call when cell phone and other services are disrupted or inaccessible. All HELP messages are handled by HELPER in two ways. First, the HELP message is send destined for the ERC with maximum HTL. Additionally, at the service layers, these HELP messages are also broadcasted with a predefined HTL (currently set to 2). The intention of broadcasting the HELP message with HTL = 2 is to find a first responder who may be in the vicinity of the individual in distress to provide faster response rather than waiting for ERC to react. The HTL is limited to avoid excessive energy consumption and mitigate problems of congestion. Overall, the HELP message will enable users to alert authorities of their location, need, and situation when all other communication infrastructures are down.

Local Messages: Every user connected to a HELPER is able to chat with each other using Local Chat messages. These messages are exchanged using WiFi itself and do not have to use the LoRa on Physical Layer. These links can achieve high data rates and in future support video chatting as long as HELPER is plugged in and
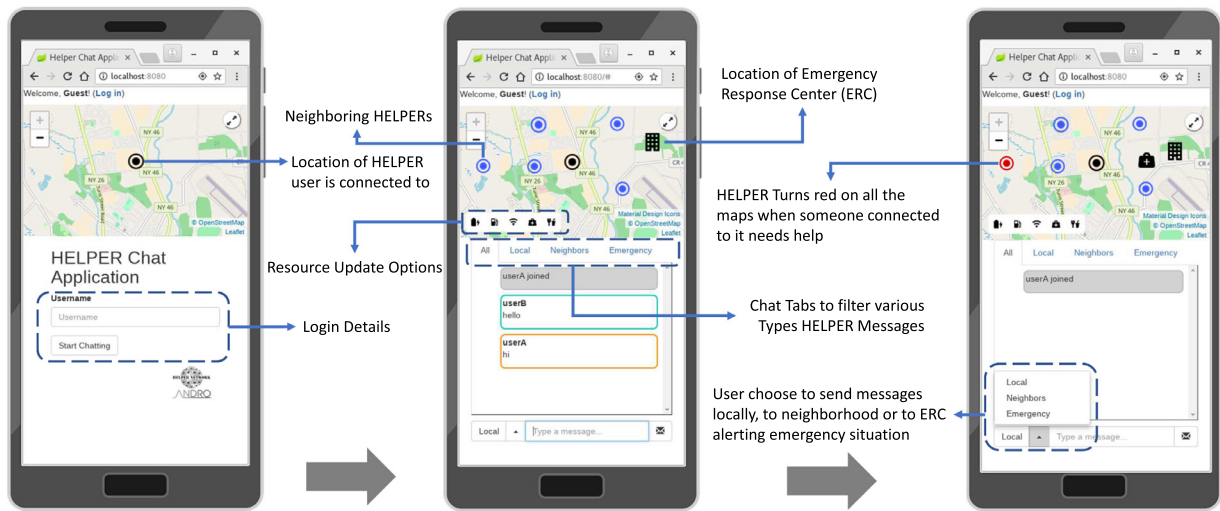
Fig. 9. Website application.

does not have energy constraints. Therefore, everyone in the range of a single HELPER can use local messaging to remain connected to each other. These messages are handled by the service layers itself and are not passed to the lower layers of the HELPER protocol stack.

Neighborhood Messages: A neighborhood message is a chat message that is transmitted to all immediate neighboring HELPERs. In the network layer, this is a broadcast message with an HTL of 1. This message is intended to enable communication between the community in the close neighborhood (within 2 km radius). These messages will be used by the community members to help each other and mark themselves safe even if they are not connected to the same HELPERs. The design is flexible enough such that the broadcast can be extended beyond immediate neighbors by setting appropriate HTL.

Resource Messages: A resource message is sent by a user to indicate that a resource (food, water, gas, medicine, internet, etc.) is available in proximity to the local HELPER. This message is transmitted to the ERC with an HTL set to maximum. The Responder Station aggregates these messages, approves them and transmits periodic HELPER resource update messages to let all the HELPERs in the network get the updated Map.

4.2.2. Emergency response center messages

Network Discovery Message: A Network Discovery (ND) message is used at network initialization. The ERC broadcasts this ND message to its immediate neighbors. All HELPERs that receive this message use an efficient flooding technique to broadcast ND messages to other HELPERs. As the deployed HELPERs receive a ND, they reply with a HELPER Update Message containing their information (Node ID, location etc.). These are unicast messages to the ERC with a maximum HTL. In this manner, the ERC performs network discovery to map all the nodes that are actively deployed in an affected area.

HELPER ALERT Message: Similar to the Wireless Emergency Alerts (WEA) that is used over the cellular network, the ALERT message is intended to inform every connected user about an imminent threat like high winds, rising water level, flash flood, fires etc. This message is also distributed using an efficient flooding technique. Every user connected to a HELPER sees a message labeled from ERC and hence are aware of the steps to take to remain safe during the upcoming situation. This will be highly beneficial in situations where the cellular network is not operable due to infrastructural damage.

Resource Update Message: Once the ERC receives a resource message from EU connected to any of the HELPERs in the network, it has to first approve the resource update. Upon approval, the ERC floods the resource update message to the entire network using an efficient flooding technique. In this manner, every HELPER in the network receives updated resource information for users to access.

4.3. Applications

As discussed, to provide a complete end-to-end solution, we have developed two applications one for EU to connect to the HELPER network using their mobile devices and the second for ERC to remotely monitor the network and provide assistance and alerts to the EU. In this section, we describe the functionalities that have been enabled through these applications.

4.4. End-user application

Every user connected to a HELPER via WiFi will be prompted to access the services of HELPER network by logging in to the Web App shown in Fig. 9. As you can see, the login page consists of the location of the HELPER (marked using a black marker) that the user is currently connected. The final version of the App will also have a short message describing the network and utilities to encourage people to use the HELPER network. Next, we describe features availed by the Web App for the connected EU to use.

Text and Voice Messaging: The primary goal of the HELPER Network is to keep individuals in the affected community connected. Therefore, an intuitive chatbox is developed for EUs to interact with and help each other and ERC. As shown in Fig. 9, there are three kinds of messages that can be sent/received by a user (as indicated by the 3 tabs or options in the pop-up menu).

 i The *local* messages are exchanged between users connected to the same HELPER. These messages go directly over WiFi and do not need to interact with the LoRa physical layer. These links can achieve high throughput since it is not bottle-necked by the lower data-rates of LoRa. In the future, video call and higher throughput applications can be enabled based on the availability of energy in the affected area.

 ii Next, the message sent using *Neighbor* tab are broadcasted to *h*-hop neighbors (where *h* is predetermined by the operator). The choice of *h* would be a trade-off between energy consumption and range of connectivity. In this case, a
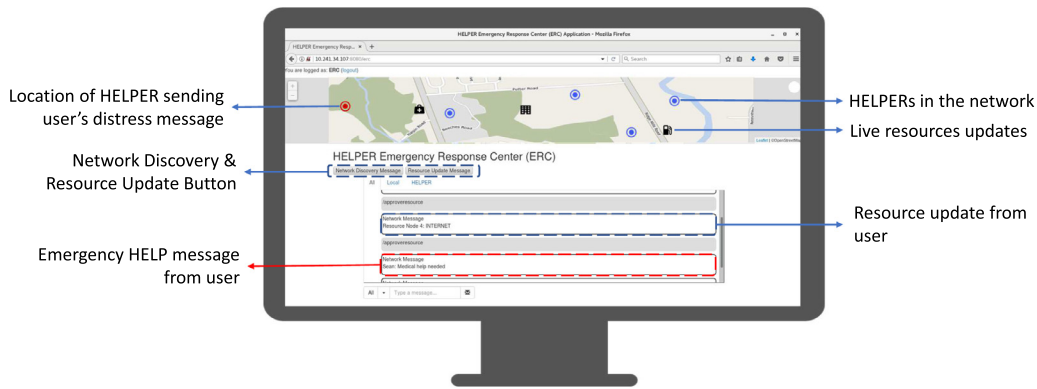
Fig. 10. ERC application.

message sent by a user to the neighbors is received by all users connected to all the HELPERs (black and blue markers) within *h* hops from the source node.

iii Finally, and most importantly, the *Emergency* tab is used to send distress messages directly to the ERC to seek help during distress. These messages will be carried over a multi-hop path to the ERC and inform the ERC of the location where help is required. This serves as an alternative to 9-1-1 calls when the degradation of infrastructure renders traditional 9-1-1 calls infeasible. Similarly, ERC can broadcast ALERT messages so that each user is alerted to situations like high winds, rising water level, flash floods etc. The *All* tab displays all the above messages in one place.

Live Map Updates: A regional map with live updates on the availability of resources like gas stations, operational hospitals, food and water gas station, internet access, electricity etc are accessible to the connected users. The ERC will collect information about the availability of resources using HELPERs deployed in hospitals, stores, gas stations, households etc. Periodically, this information is flooded by the ERC in the ad hoc network to update the map at each HELPER. The periodicity of this flooding can be controlled by the ERC based on the update information and status of the network. This information sharing is accomplished as follows,

- A connected user (ER or EU) who has information about available resources to share with rest of the users, drags and drops the corresponding resource on the known location on their local map. (Fig. 10).
- This action triggers a packet that is directed towards the nearest ERC. This packet is routed using the proposed SEEK algorithm towards ERC.
- Upon reception of the packet, a message containing the information about the type of resource and its location shows up on the ERC Application. Once the operator verifies this information, it is flooded to the rest of the HELPER network. The method of verification will be controlled by the agencies. This can be based on trusted nodes, the number of similar requests or physical verification using an on-field ER.

It can be argued that the above three-step process may incur a delay in disseminating information as compared to the information being flooded by the source HELPER itself without going through the ERC. While this may be true, authorizing any node to update resource information may lead to the propagation of misinformation, duplicate information, and overall larger energy consumption.

4.5. Emergency response center dashboard

As shown in Fig. 5, at the ERC, a HELPER is connected to a PC using an Ethernet cable. ERC Dashboard with some critical features have been developed with the following capabilities,

Remote Monitoring: A ND phase can be initiated by using the *Network Discovery Message* button on the ERC Dashboard. Accordingly, the ERC broadcasts ND packet to its immediate neighbors. All HELPERs that receive this message use an efficient flooding technique to broadcast ND packet to other HELPERs. As HELPERs receive a ND packet, they reply with a *HELPER Update* packet. The HELPERs deployed in the field use this unicast *HELPER Update* packet during ND phase to reply to the ND packet with their information (Node ID, location etc). The operator can perform this remote monitoring intermittently to ensure all the HELPERs in the network are active. If some of the HELPERs do not show up during these intermittent monitoring phases, the operator will be aware of the lack of connectivity in those areas and can deploy more nodes or take other corrective actions to keep the network fully connected.

Critical Alert Message: Similar to the Wireless Emergency Alerts (WEA) that is used over the cellular network, the *ALERT* message is intended to inform every connected user about an imminent threat like high winds, rising water level, flash flood, fires etc. This message will also be distributed using an efficient flooding technique. Every user connected to a HELPER sees a message labeled from ERC and hence are aware of the steps to take to remain safe during the upcoming situation. This will be highly beneficial in situations where cellular network is not operable due to damage and a large number of individuals need to be informed about imminent dangers.

5. Evaluation

In this section, we discuss the HELPER prototype that was developed to establish proof of concept and conduct some initial development. (Fig. 14).

5.1. Operational proof-of-concept

In this section, we have focused on developing the prototype for MH as shown in Fig. 11. We decided to prototype the MH because of its portability, cost and because it can be used to demonstrate the envisioned operation of the entire HELPER network. This concept can be easily extended when more SH and/or AH are added to the HELPER network. The proof-of-concept demonstrations conducted by us were recorded using mobile screen recorders and compiled in the form of a video [54]. In this section, we use parts
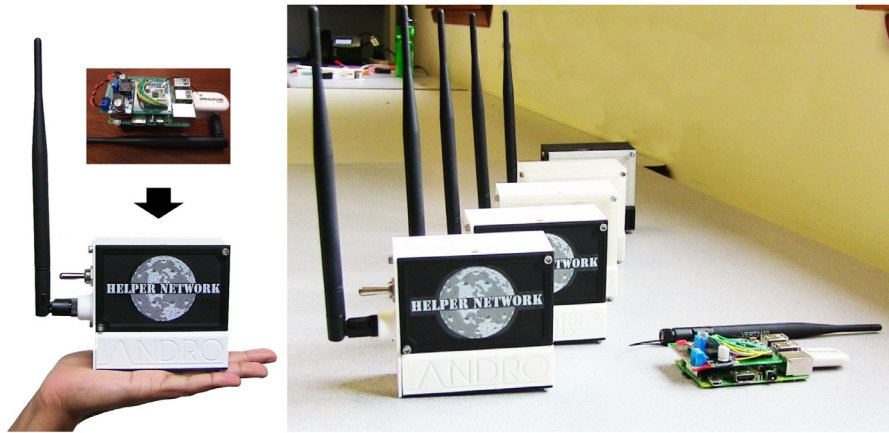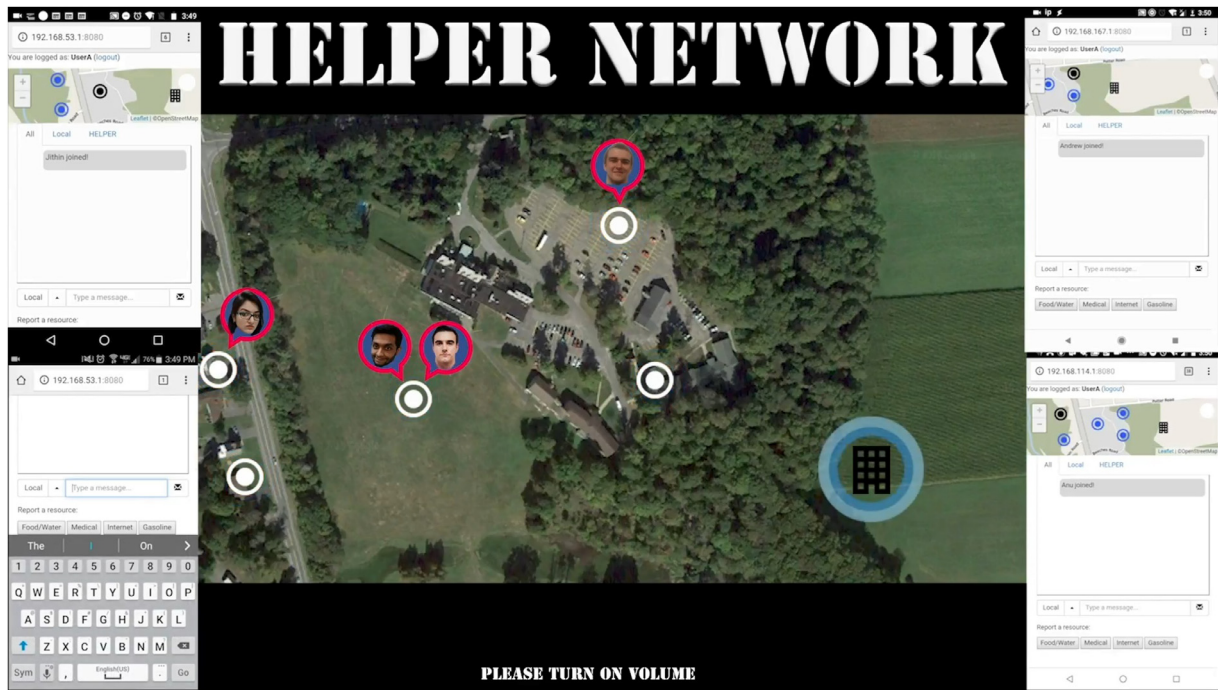
Fig. 11. HELPER protoype.

Fig. 12. 6-node HELPER network.

of the video to discuss the experiments and functionalities it intended to display. Accordingly, we have developed six HELPERs (see Fig. 11), five of which are deployed with given location value as shown in Fig. 12 (white markers) for EU to connect using their mobile devices. The sixth one is connected to a PC and acts as the ERC which is indicated as a building in Fig. 12. We had four EUs connected to the network through three of the deployed HELPERs. As you can see two users (Jithin and Nick) are connected to the same HELPER. The Web App with the chatbox corresponding to each connected users are displayed at the edges of Fig. 12.

First, we tested the operation of the *local* messaging using the HELPER where two users were connected. Several text messages were exchanged between Nick and Jithin, as you can see in Fig. 13 connected to the same HELPER. As mentioned earlier, these messages are exchanged over WiFi and do not need to use LoRa. Since these are local messages, the other two users (Andrew and Anu) connected to their respective HELPERs will not receive these messages. Next, Jithin switches his chat option from *local* to *neighbor* which implies all users connected to HELPERs within h hop

will receive his messages. In this deployment $h = 1$, which implies one-hop neighbors will receive chat messages. Accordingly, the text message sent by Jithin is received by the other three users even if they are not connected to the same HELPER as Jithin. This part of the demonstration proves how HELPER can be used to keep community members connected by exchanging text messages with each other. The EU is completely abstracted from the ad hoc networking operation that happens in the background. To the EU, they are just sending messages to two different groups, one local group, and other more extended community groups.

As mentioned before, ERC has the ability to flood the HELPER network with ALERT messages to inform connected EU of imminent dangers. In this case, the ERC sends an *ALERT* message regarding the high winds. In Fig. 15, it can be seen that three users connected to different HELPERs have received the "HIGH WIND" alert message transmitted by ERC. The packet was dropped for one user since we do not have a transport layer currently implemented that ensures end-to-end reliability. During this period of demonstration, multiple users have shared the availability of resources like a gas
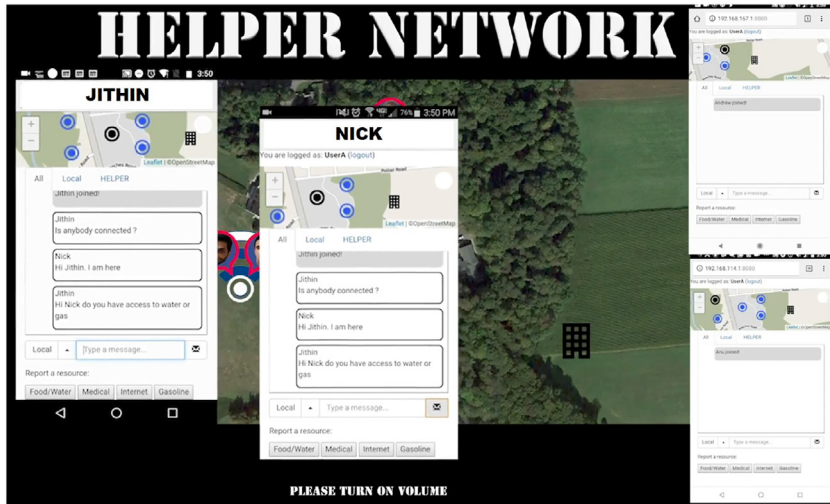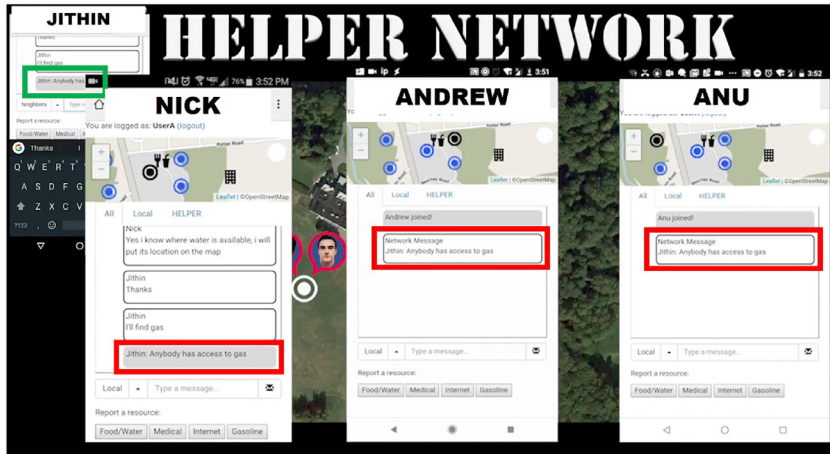
Fig. 13. Local text messaging.

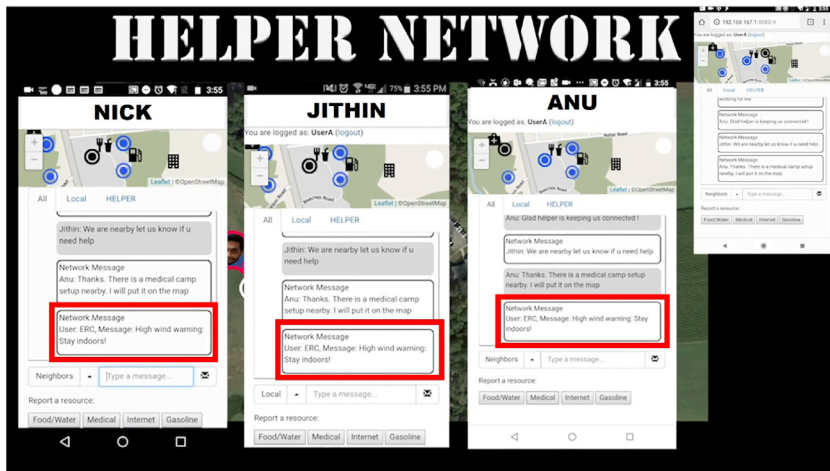**Fig. 14.** Neighbor text messaging.

Fig. 15. ERC's ALERT.

station, water, and food etc. These packets are first sent directly to ERC and upon approval, the information is flooded in the network. Accordingly, the connected users are able to see the location of the resource on their local map in their Web App as shown in Fig. 16.

The final part of the demonstration was to evaluate how distress messages can be sent directly to the ERC. In this case, Jithin realized Nick needs medical attention and uses the "help" option to send a message. This message is sent directly to ERC and nearby HELPERS simultaneously just in case there is ER or others in the vicinity who can provide assistance as compared to the ERC itself. Accordingly, the connected users see the HELPER which Jithin is connected to turn red indicating distress at that location. This will
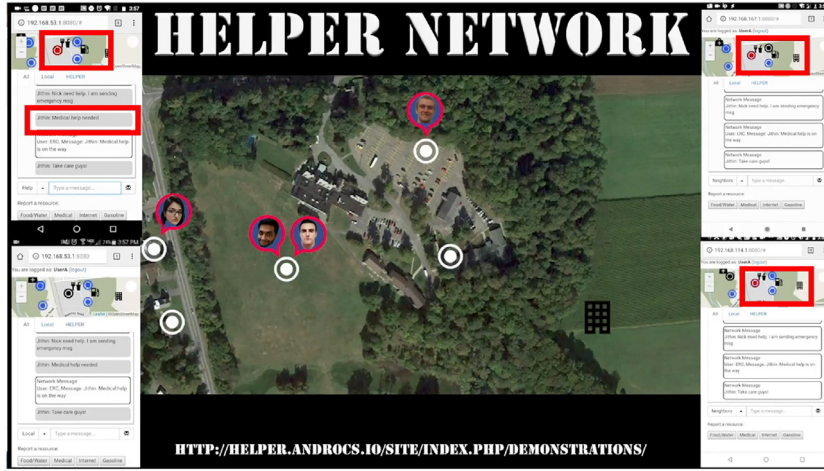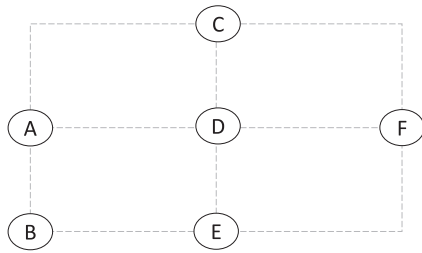
Fig. 16. Distress message.

Fig. 17. 6-node HELPER network for quantitative evaluation.

Table 4
Evaluation parameters.

Parameters	Values
Payload size	200 Bytes
Packet interval	100 ms
Physical layer	LoRa
Bandwidth	125 kHz
E_{ini}	25 J
Duration	120 min

enable community members to reach out to the nearest ER and provide the required assistance. This location information is also available at the ERC which instantly dispatch help to the given location. Overall, this service acts as the replacement for 9-1-1 calls when traditional infrastructures like cell towers or the internet are unavailable. We hope this technology will enable low cost, efficient public safety system. We also provide the demonstration [55] from the point of view of an ERC.

5.2. Testbed evaluation

In the previous section, we have established the feasibility of the proposed HELPER network. Here, we try to perform an extensive evaluation of the underlying SEEK algorithm and analyze various aspects of its operation in a unicast setting. To accomplish this, we set up the six HELPER prototypes in a grid topology as shown in Fig. 17. We compare the proposed algorithm against the shortest path routing algorithm. We implement this shortest path routing using a greedy geographical forwarding technique. In this algorithm, nodes that have a packet to forward elects the node closest to the destination as the next hop. This can also be seen as a greedy distributed version of MPR used in [29] discussed earlier in Section 2 with the assumption that paths with the smallest number of hops may indeed be the path with minimum energy consumption. Both protocols have similar complexity. In other words, all the possible next hops are considered by both. Greedy algorithm calculates the forward progress of each next-hop and SEEK calculates the utility function for each next-hop. Both then chooses the neighbor providing highest value. In terms of complexity for a given number of transmission strategies, the complexity of both the algorithms are $O(|\mathbb{NB}|)$.

The parameters used in the evaluation are depicted in Table 4. As discussed earlier, LoRa consumes extremely low power. This

means that for a realistic battery to drain completely, we may have to run the evaluation over multiple days. To save time and yet without loss of rigor, we use a virtual energy level to evaluate the HELPER network so that we can see the network behavior in experiments lasting less than 120 min. Each node is assumed to start at a total energy of 25 J and is depleted as each packet (control or data) is transmitted.

In the first experiment, F is set as the destination (would represent ERC in a real-life scenario) and HELPER A and B are the source nodes. As shown in Table 4, packets are generated at the source node at a constant rate and it has to choose appropriate routes to reach the destination. The first metric we evaluate is the minimum residual energy ($E_r^{\min}$) among all HELPERs in the network. In other words, at any given time instant t, we plot the residual energy value of the HELPER that has consumed the highest energy. The second metric under evaluation is the normalized throughput of the network calculated with respect to observed point-to-point link throughput (Th_l) and can be referred to as,

$$\overline{Th}_{net} = \frac{Th_{net}}{Th_l} \tag{7}$$

First, let's look at the initial 14 minutes of the experiments. As you can see in Fig. 18, the $E_r^{\min}$ in both cases are the same since SEEK operates similarly to the greedy algorithm in this stage even after gathering information from immediate neighbors. This is because at the beginning most of the possible next hops have similar parameters including backlog length and residual energy. Additionally, it can be seen from Fig. 19 that during the same period, the greedy algorithm seems to marginally outperform SEEK. This can be attributed to the overhead involved in SEEK to compute the optimal next hop from the gathered information. This marginal superiority is short-lived as SEEK starts learning about the environment and begins to exploit spatial diversity to choose multiple paths to the destination. This provides HELPER network with two advantages, (i) the energy consumption is evenly spread between nodes
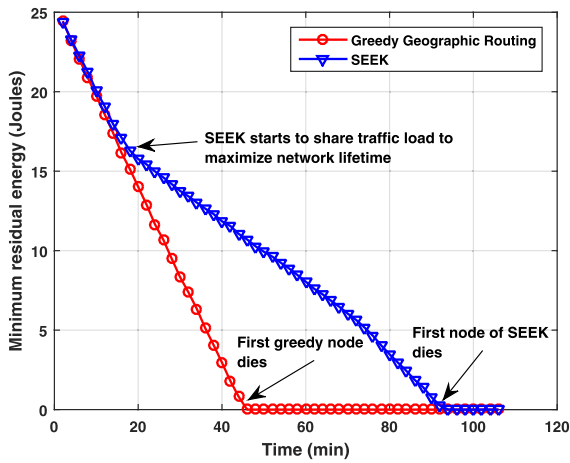
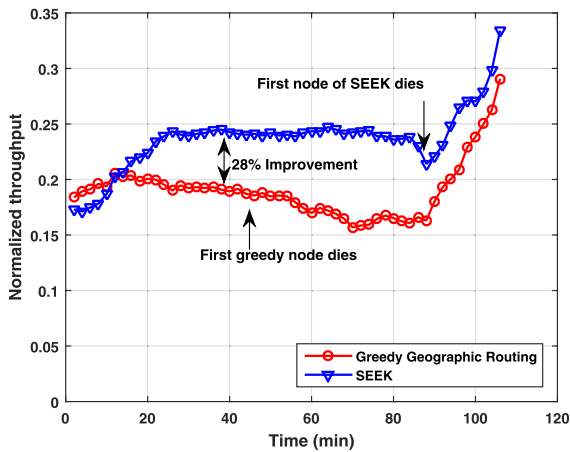**Fig. 18.** Maximum energy consumed by a node.

Fig. 20. Network lifetime vs no. of sessions.

Fig. 19. Normalized throughput of the network.

Fig. 21. Normalized throughput vs no. of sessions.

Fig. 22. Analysis of packet delivery.

and (ii) higher throughput is achieved. Accordingly, from Fig. 19, it is evident that the death of the first node in the aggressive greedy algorithm happens much earlier than the death of the first node in SEEK. This provides a proof-of-concept that SEEK can be applied to maximize the network lifetime in a distributed manner.

Next, to extend the experiments further, we evaluate the performance of SEEK while increasing the number of sessions in the network to 4. This is to evaluate if SEEK can adapt to multiple traffic partners in the network which is expected behavior in a large distributed network. These sessions include $A \rightarrow F$, $B \rightarrow C$, $C \rightarrow E$ and $F \rightarrow A$ and are chosen to ensure no source in a session has it's destination via direct link (i.e. destination is not the source's immediate neighbor). Each source in the session is set up to generate packets at a constant rate as mentioned in Table 4. Both residual energy of each node and packets received are constantly monitored. We first analyze the network lifetime which is defined as the duration of operation until the first node in the network dies. This is important for such emergency networks as the death of a node would imply unconnected users. Fig. 20 shows how SEEK outperforms the greedy algorithm regardless of the number of sessions. The experiments show an improvement of up to 53% in terms of network lifetime. One interesting finding is that the network lifetime seemed to increase with the increase in sessions which might be counter-intuitive at first sight. Further evaluation using Fig. 21 will reveal that the small network is saturated even with two sessions in the network as portrayed by the throughput decline. This implies that more collision may occur at the MAC layer leading to a
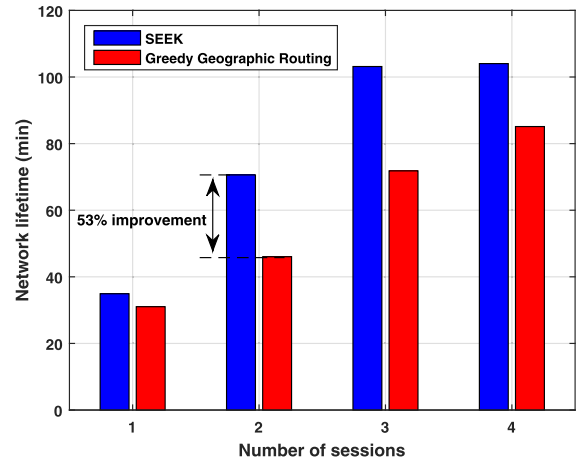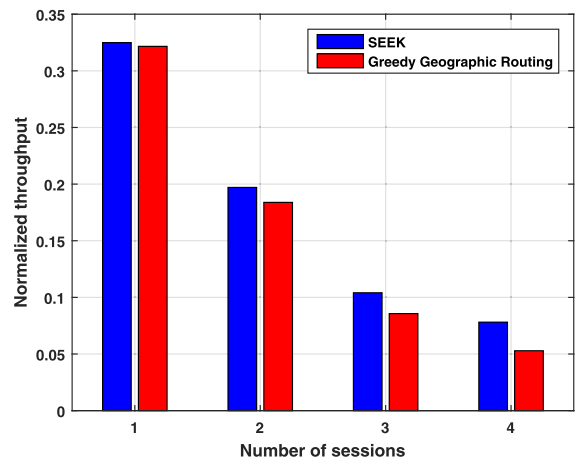
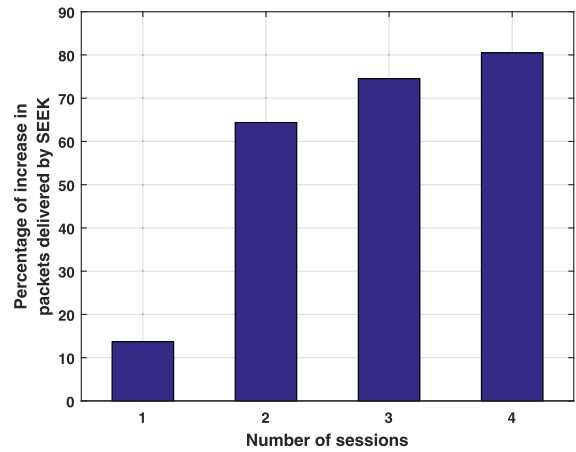larger backoff and lower throughput as the number of sessions increase. In a saturated network, the overall throughput even while operating for a longer period of time is better for SEEK compared to the greedy algorithm. To further substantiate the importance of network lifetime, we plot the percentage increase in packets delivered by SEEK as compared to the greedy algorithm in Fig. 22. This keeps increasing as the number of sessions in the network grows which can be related to delivering critical information from survivors to the ERC during the aftermath of the disaster.
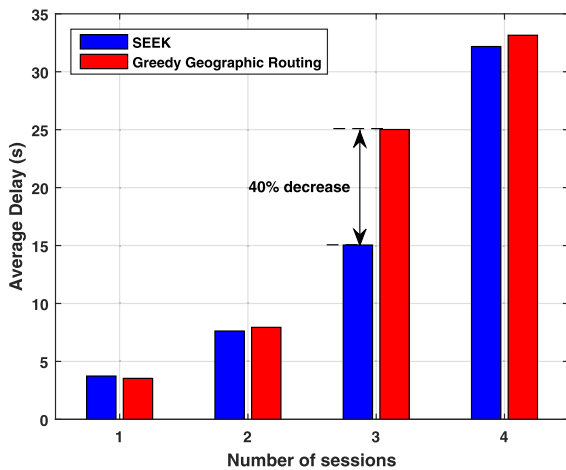
Fig. 23. Average delay vs no. of sessions.

Finally, we look at the average delay per packets as the number of sessions in the network increases. To accomplish this, we set each session to transmit 100 packets while keeping the rest of the setting similar to the earlier experiment. As expected, the delay per packet of both the schemes increases as the number of sessions in the network increases due to congestion. The more critical observation from Fig. 23 is that the delay incurred by packets serviced using SEEK is up to 40% less than greedy algorithm especially when the traffic increases (3 sessions). This is because SEEK is able to use multiple paths to distribute traffic spatially among nodes to reduce congestion at the bottleneck nodes. This is further substantiated by the fact that the advantage in terms of lower delay diminishes as the network saturates (4 sessions) since all the nodes are involved in either case (greedy and SEEK) leaving no extra nodes for SEEK to distribute traffic load.

Overall, the experiments showed how nodes in SEEK share information among each other using the control packets which is then used to perform cross-layer optimization to choose optimal routes that ensure all nodes share the load of the traffic to maximize the network lifetime. We expect the improvement in the performance to be significantly higher on a larger network consisting of hundreds of nodes. Here, we have prototyped HELPER and set up a small yet effective testbed with a limited number of nodes to perform extensive testing. The results provide proof-of-concept that the proposed HELPER network can be deployed in near future to enable off-the-grid connectivity.

6. Conclusion

In this work, we have proposed, prototyped and established the proof-of-concept of a complete end-to-end solution to enhance and enable public safety communication systems. The proposed HELPER uses heterogeneous wireless communication techniques; (i) WiFi which enables EU to connect to the HELPER like any WiFi access point thereby ensuring easy and widespread adoption, and (ii) LoRa, that provides extremely low power, long range wireless link to implement the ad hoc operation. The HELPER network is used to set a completely self-sustained network that does not require the support of any traditional communication infrastructure like cell towers or satellite. The HELPER network is designed to serve a dual purpose; (i) enable affected individuals to stay connected and maintain situational awareness, and (ii) equip authorities to remotely monitor the situation, provide assistance and warnings in an efficient manner.

The proposed solution provides connected EU with live map updates to share the location of known resources. It enables text messages between community members and equips EU with an alternative to traditional 9-1-1 like emergency calls. Similarly, it provides ERC with the capability to monitor the network connectivity, manage resource sharing information and send out ALERT messages to connected users. Additionally, numerical evaluations using HELPER testbed showed up to 53% improvement in network lifetime and up to 28% improvement in network throughput as compared to a greedy scheme that routes using shortest path. All these demonstrated capabilities will enhance the state-of-the-art public safety response system.

References

[1] J. Jagannath, S. Furman, A. Jagannath, A. Drozd, Energy efficient ad hoc networking devices for off-the-grid public safety networks, in: Proc. of IEEE Consumer Communications & Networking Conference (CCNC), Las Vegas, NV, USA, 2019.

[2] FEMA: disaster list, (https://www.fema.gov/disasters). Accessed: April. 27, 2018.

[3] The deadliest natural disasters of 2017, (https://www.telegraph.co.uk/education/stem-awards/design/deadliest-natural-disasters/). Accessed: April. 27, 2018.

[4] T. Gomes, J. Tapolcai, C. Esposito, D. Hutchison, F. Kuipers, J. Rak, A. de Sousa, A. Iossifides, R. Travanca, J. André, L. Jorge, L. Martins, P.O. Ugalde, A. Paši?, D. Pezaros, S. Jouet, S. Secci, M. Tornatore, A survey of strategies for communication networks to protect against large-scale natural disasters, in: 8th International Workshop on Resilient Networks Design and Modeling (RNDM), 2016, pp. 11–22.

[5] When communications infrastructure fails during a disaster, (https://www.drj.com/articles/online-exclusive/when-communications-infrastructure-fails-during-a-disaster.html). Accessed: April. 27, 2018.

[6] Proportion of population covered by a mobile network, by technology, (https://sdg.data.gov/9-c-1/). Accessed: April. 27, 2018.

[7] P.P. Ray, M. Mukherjee, L. Shu, Internet of things for disaster management: state-of-the-art and prospects, IEEE Access 5 (2017) 18818–18835.

[8] O.A. Dobre, A. Abdi, Y. Bar-Ness, W. Su, Survey of automatic modulation classification techniques: classical approaches and new trends., IET Commun. 1 (2) (2007) 137–156.

[9] G. Baldini, S. Karanasios, D. Allen, F. Vergari, Survey of wireless communication technologies for public safety, IEEE Commun. Surv. Tut. 16 (2) (2014) 619–641.

[10] Project 25 technology interest group (PTIG), (http://www.project25.org/). Accessed: April. 27, 2018.

[11] ETSI EN 300 392-1 V1.4.1 (2009-01), Terrestrial trunked Radio (TETRA); Voice plus Data (V+D); Part 1: General network design (2009).

[12] D. Rogers, Interoperability: ensuring joined-up communication, 2007, (https://www.geoconnexion.com/articles/interoperability-ensuring-joined-up-communication/). Accessed: April. 27, 2018.

[13] ETSI TR 102 580 V1.1.1, Terrestrial trunk radio (TETRA); Release 2; designer's guide; TETRA high-speed data (HSD); TETRA enhanced data service (TEDS) (2007).

[14] H. Nishiyama, M. Ito, N. Kato, Relay-by-smartphone: realizing multihop device-to-device communications, IEEE Commun. Mag. 52 (4) (2014) 56–65.

[15] K. Ali, H.X. Nguyen, Q.T. Vien, P. Shah, Z. Chu, Disaster management using D2D communication with power transfer and clustering techniques, IEEE Access 6 (2018) 14643–14654.

[16] BRCK, (https://www.brck.com/connectivity/). Accessed: April. 27, 2018.

[17] V.G. Menon, J.P. Pathrose, J. Priya, Ensuring reliable communication in disaster recovery operations with reliable routing technique, Mob. Inf. Syst. (2016).

[18] Z. Lu, G. Cao, T.L. Porta, Teamphone: networking smartphones for disaster recovery, IEEE Trans. Mob. Comput. 16 (12) (2017) 3554–3567, doi:10.1109/TMC.2017.2695452.

[19] K. Ali, H.X. Nguyen, Q.T. Vien, P. Shah, Z. Chu, Disaster management using D2D communication with power transfer and clustering techniques, IEEE Access 6 (2018) 14643–14654, doi:10.1109/ACCESS.2018.2793532.

[20] FCC White Paper, The public safety nationwide interoperable broadband network, a new model for capacity, performance and cost, June 2010.

[21] K. Gomez, L. Goratti, T. Rasheed, L. Reynaud, Enabling disaster-resilient 4G mobile communication networks, IEEE Commun. Mag. 52 (12) (2014) 66–73, doi:10.1109/MCOM.2014.6979954.

[22] J. Deaton, M. Schmitt, S. Cherry, C. Papke, Analyzing Options for Airborne Emergency Wireless Communications, Idaho National Laboratory, 2008.

[23] WiMAX in Haiti, (http://www.dailywireless.org/2010/02/12/wimax-in-haiti/). Accessed: April. 27, 2018.

[24] A. Yarali, B. Ahsant, S. Rahman, Wireless mesh networking: a key solution for emergency & rural applications, in: Proc. of Second International Conference on Advances in Mesh Networks, 2009.

[25] S.S. Anjum, R.M. Noor, M.H. Anisi, Survey on MANET based communication scenarios for search and rescue operations, in: Proc. of 5th International Conference on IT Convergence and Security (ICITCS), 2015.

[26] D. Ben Arbia, M.M. Alam, A. Kadri, E. Ben Hamida, R. Attia, Enhanced IoT-based end-to-end emergency and disaster relief system, J. Sensor Actuator Networks 6 (3) (2017).

[27] H. Aïache, V. Conan, G. Guibé, J. Leguay, C. Le Martret, J.M. Barcelo, L. Cerda, J. Garcia, R. Knopp, N. Nikaein, X. Gonzalez, A. Zeini, O. Apilo, A. Boukalov, J. Karvo, H. Koskinen, L.R. Bergonzi, J.C. Diaz, J. Meessen, C. Blondia, P. Decleyn, E. Van de Velde, M. Voorhaen, WIDENS: advanced wireless ad-hoc networks for public safety, in: Proc. of IST Summit, 14th Mobile & Wireless Communication Summit, 2005.

[28] K. Kanchanasut, A. Tunpan, M. Awal, T. Wongsaardsakul, D.Das, Y.Tsuchimoto, Building a Long-distance Multimedia Wireless Mesh Network for Collaborative Disaster Emergency Responses, Internet Education and Research Laboratory, Asian Institute of Technology, 2007.

[29] M.W. Subbarao, Dynamic power-conscious routing for MANETs: an initial approach, J. Res. Natl. Inst. Stand. Technol. 104(6) (1999) 587–593.

[30] A.K. Chandra-Sekaran, A. Nwokafor, P. Johansson, K.D. Mueller-Glaser, I. Krueger, ZigBee sensor network for patient localization and air temperature monitoring during emergency response to crisis, in: Proc. of Second International Conference on Sensor Technologies and Applications (SENSORCOMM), 2008.

[31] Raspberry Pi 3 model B, (https://www.raspberrypi.org/products/raspberry-pi-3-model-b/). Accessed: April. 27, 2018.

[32] LoRa (RF95), (https://www.digikey.com/catalog/en/partgroup/rfm95w-868s2/50906). Accessed: April. 27, 2018.

[33] F. Adelantado, X. Vilajosana, P. Tuset-Peiro, B. Martinez, J. Melia-Segui, T. Watteyne, Understanding the limits of lorawan, IEEE Commun. Mag. 55 (9) (2017) 34–40.

[34] Erle Robotics, (https://erlerobotics.com/blog/erle-copter/). Accessed: April. 27, 2018.

[35] D. Pompili, M.C. Vuran, T. Melodia, Cross-layer design in wireless sensor networks, Sensor Network and Configuration: Fundamentals, Techniques, Platforms, and Experiments, Springer-Verlag, Germany, October 2006.

[36] R. Mehta, D.K. Lobiyal, Energy efficient cross-layer design in MANETs, in: Proc. of 4th International Conference on Signal Processing and Integrated Networks (SPIN), 2017, pp. 448–453, doi:10.1109/SPIN.2017.8049991.

[37] S. Colonnese, F. Cuomo, T. Melodia, I. Rubin, A cross-layer bandwidth allocation scheme for HTTP-based video streaming in LTE cellular networks, IEEE Commun. Lett. 21 (2) (2017) 386–389.

[38] A.L. Drozd, T. Arcuri, J. Jagannath, D.A. Pados, T. Melodia, E. Demirors, G. Sklivanitis, Network throughput improvement in cognitive networks by joint optimization of spectrum allocation and cross-layer routing, in: Proc. of NATO Symp. on Cognitive Radio and Future Network (IST-123), 2014. The Hague, The Netherlands.

[39] J. Jagannath, T. Melodia, A. Drozd, DRS: distributed deadline-based joint routing and spectrum allocation for tactical ad-hoc networks, in: Proc. of IEEE Global Communications Conference (GLOBECOM), 2016. Washington D.C., USA

[40] J. Jagannath, S. Furman, A. Drozd, T. Melodia, Design and experimental evaluation of a cross-layer deadline-based joint routing and spectrum allocation algorithm, IEEE Trans. Mob. Comput. (2018).

[41] Q. Jiang, Y. Zhou, J. Wang, Y. Wang, X. Wang, A lightweight cross-layer cooperative testbed for evaluation of connected vehicles, in: Proc. of 11th International Conference on Mobile Ad-hoc and Sensor Networks (MSN), 2015, pp. 194–201.

[42] E. Demirors, G. Sklivanitis, T. Melodia, S.N. Batalama, RcUBe: real-time reconfigurable radio framework with self-optimization capabilites, in: Proc. of IEEE Intl. Conf. on Sensing, Communication, and Networking (SECON), 2015. Seattle, WA.

[43] J. Jagannath, H. Saarinen, T. Woods, J. OBrien, S. Furman, A. Drozd, T. Melodia, COmBAT: cross-layer based testbed with analysis tool implemented using software defined radios, in: Proc. of IEEE Conf. on Military Communications (MILCOM), 2016. Baltimore, MD, USA.

[44] Semtech LoRa documentation, (https://www.semtech.com/uploads/documents/an1200.22.pdf). Accessed: April. 27, 2018.

[45] T. Nadeem, S. Banerjee, A. Misra, A.K. Agrawala, Energy-Efficient Reliable Paths for On-Demand Routing Protocols, Springer US, Boston, MA, pp. 485–496.

[46] S. Dawans, S. Duquennoy, O. Bonaventure, On link estimation in dense RPL deployments, in: Proc. of 37th Annual IEEE Conference on Local Computer Networks - Workshops, 2012. Clearwater, FL, USA.

[47] X. Chen, Y. Xu, A. Liu, Cross layer design for optimizing transmission reliability, energy efficiency, and lifetime in body sensor networks, Sensors 17 (4) (2017) 900.

[48] L. Tassiulas, A. Ephremides, Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks, in: Proc. of IEEE Conference on Decision and Control (CDC), 1990. Honolulu, HI.

[49] S. Liu, E. Ekici, L. Ying, Scheduling in multihop wireless networks without back-pressure, IEEE/ACM Trans. Networking 22 (5) (2014) 1477–1488.

[50] J. Jagannath, T. Melodia, A. Drozd, DRS: distributed deadline-based joint routing and spectrum allocation for tactical ad-hoc networks, in: Proc. of IEEE Global Communications Conference (GLOBECOM), 2016. Washington DC, USA.

[51] K. Hao, Z. Jin, H. Shen, Y. Wang, An efficient and reliable geographic routing protocol based on partial network coding for underwater sensor networks, Sensors 15 (6) (2015) 12720–12735.

[52] H. Song, Z. Xu, X. Wang, Energy-efficient flooding with minimum latency for low-duty-cycle WSNs, in: Proc. of 10th International Conference on Sensing Technology (ICST), 2016, pp. 1–6.

[53] S. Ahn, Y. Lim, H. Yu, Energy-efficient flooding mechanisms for the wireless sensor networks, in: Proc. of International Conference on Information Networking, 2008, pp. 1–5.

[54] Demonstration video: HELPER network 6 node demonstration (user perspective), (https://youtu.be/wfXkuAtFlfQ). Accessed: April. 27, 2018.

[55] HELPER network 6 node demonstration (ERC perspective), (https://youtu.be/TtWm3oX-4z8). Accessed: April. 27, 2018.

Mr. Jithin Jagannath is an Associate Senior Scientist/Engineer at ANDRO Computational Solutions, LLC. He is also a Ph.D. Candidate in the Department of Electrical and Computer Engineering at Northeastern University. Mr. Jagannaths expertise and research interests include cognitive radio, optimized networking, automatic modulation classification (AMC), machine learning, testbed design using SDRs, communication protocols for wireless sensor networks including terrestrial, and underwater sensor networks. He joined ANDRO in 2013, previously he was at the Wireless Networks and Embedded Systems Laboratory at the University at Buffalo from where he received his MSEE degree specializing in communication, control and signal processing in 2013. Mr. Jagannath has extensive experience developing SDR based prototypes. This includes designing, implementing and testing different communication protocols and optimized algorithms on hardware for wireless sensor networks. Recently, he was the Principal Investigator (PI) of DHS SBIR with the objective to design and develop a portable low-cost solution that detects and localizes radio frequency interference for first responders. Previously, he has been the PI of HELPER, a project that aimed to provide off-the-grid connectivity to survivors for disasters through wireless ad hoc network. He has also been the technical lead in multiple SBIR/STTR efforts including cross-layer networking technology, AMC (which employed ANN based solution), cybersecurity, and compressive sensing. He has been the lead author of several peer-reviewed journal and conference publications. He also renders his review services to several leading Elsevier and IEEE Journals and conferences.

Sean Furman received his BSECE degree in 2017 from the State University of New York Polytechnic Institute in Utica, New York. He is currently pursuing a masters degree in electrical engineering at Clarkson University in Potsdam, New York. He joined ANDRO Computational Solutions in 2015 and has worked on projects involving software defined radio networks and Dynamic Spectrum Access (DSA) policy. His research interests include wireless communication, signal processing, and automatic control systems.

Anu Jagannath received her M.S. degree (2013) from State University of New York at Buffalo in Electrical Engineering. She is currently an Associate Scientist Engineer at ANDRO Computational Solutions, LLC. Her research focuses on MIMO communications, Deep Machine Learning, Adaptive signal processing, Software Defined Radios, spread spectrum systems, LPI/LPD communications, spectrum sensing, adaptive Physical layer and cross layer techniques, medium access control and routing protocols, underwater wireless sensor networks, signal intelligence and so on. She has rendered her reviewing service for IEEE Annual Consumer Communications & Networking Conference 2019. She has played the role of Technical Lead and Lead developer in multiple SBIR/STTR efforts involving but not limited to Towards Maximal Spectral efficiency networking, Developing low cost, portable interference detection and analysis device. She has been the lead author of several conference publications and co-author of a book chapter.

Luther Ling received his B.A. in Computer Science at Hunter College in 2015, pursuing electives in artificial intelligence and web-based technologies. After graduating, he assisted in the development of a large-scale autonomously flocking mobile ad-hoc network testbed. He has worked at ANDRO Computational Solutions, LLC. since 2016 and has developed portable applications for several software defined radio (SDR) platforms including front-end and back-end components, kernel modules, network stacks, and emulation environments. He is cultivating his interest in topics including software optimization methods, hardware design, and digital signal processing. He enjoys snowboarding whenever the weather allows for it, and prefers tea over coffee.
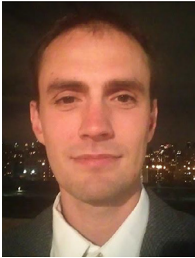
Andrew Burger is Senior Software Engineer/Security Analyst at ANDRO Computational Solutions, LLC. He has been a software engineer for twenty years in various fields including software engineering design, software architecture among others. He received his B.S.C.S. East Stroudsburg University of Pennsylvania in 2004.

Mr. Drozd established ANDRO Computational Solution, LLC in 1994. His fields of expertise include wireless communication, dynamic spectrum access, computational electromagnetics (CEM), development of electromagnetic environmental effects (E3) tools for cosite analysis, spectrum exploitation and interference rejection technologies. He has gained extensive mobile radio communication experience through various working relationships with the DoD and the Federal government supported clients include the U.S. Army, U.S. Air Force, U.S. Navy, Department of Homeland Security (via a CSC CMaaS IDIQ contract vehicle), Office of the Secretary of Defense (OSD), Missile Defense Agency (MDA), and the DISA JSC. Over the past several years, Mr. Drozd has led efforts to develop/mature new technologies under Small Business Innovative Research (SBIR) and Commercialization Readiness Program (CRP) contracts namely the Adaptive Cyber-secure Cross-layer Communications-Classification System (AC4S) for enhancing the throughput, spectral efficiency, speed, reliability and security performance of software defined radio (SDR) communication networks. Mr. Drozd was President of the IEEE EMC Society (2006–2007), a past member of the EMC-S Board of Directors (1998–2008), and is an IEEE Fellow. He was also on the Board of Directors of the Applied Computational Electromagnetics Society (ACES) (2004–2010). He contributed a chapter on EMC Modeling and Simulation for the 2nd Ed. of the IEEE Press book titled Engineering Electromagnetic Compatibility: Principles, Measurements, Technologies, and Computer Models by V. Prasad Kodali and published a chapter on radar spectrum management in the textbook Principles of Waveform Diversity and Design. He is an iNARTE certified EMC Engineer and has authored over 160 technical papers, reports, and journal articles.