Evaluation of LoRa Mesh Networks for Disaster Response

by

Brenton Poke

A thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science
(Computer Science and Information Systems)
in The University of Michigan - Flint
2023

Committee:

Advisor:

Suleyman Uludag, Ph.D.

Professor of

Computer Science

Signature

Date

Reader:

Stephen Turner, Ph.D.

Associate Professor of

Computer Science

Signature

Date

FLINT

COMPUTER SCIENCE AND INFORMATION SYSTEMS

TABLE OF CONTENTS

LIST OF FIGURES

Figure

LIST OF TABLES

Table

ABSTRACT

Evaluation of LoRa Mesh Networks for Disaster Response

by

Brenton Poke

Natural and man-made disasters are becoming more prevalent and increasing in danger as climate change continues unabated and resources get scarce. Whether it be flood or terrorism, disasters displace people and it remains difficult to reach those in need when the disaster does strike. For years, mobile phone networks have been integral in responding to emergencies, but they rely on costly infrastructure that is prone to outages or outright attack. We propose a long-range, low-power mesh network infrastructure based on a new, still-developing networking protocol and embedded software implementation that seeks to fill the need of disaster communications telemetry gathering for a fraction of the cost of mobile phone networking infrastructure such as 4G LTE or 5G. While the protocol is simple, it was found to have an inadequate implementation at this stage. Error rates were high and bugs were found in implementation that led to an abnormal amount of time spent processing corrupted data, though power consumption was encouraging in spite of these conditions. We examine the strengths and weaknesses of this new protocol and suggest improvements to harden the implementation.

CHAPTER I

Introduction

Natural disasters are becoming more frequent and more dangerous as climate change continues unabated[2]. There is a need to manage the fallout from these disasters by creating network communication infrastructure that is both inexpensive and reliable. Wireless communication is an attractive avenue for creating this infrastructure, but mobile wireless networks run by telecommunications companies and publicly funded infrastructure are prone to outages[3], while being expensive to set up and maintain. There are other options involving handheld radio communications using wireless bands reserved for public safety or satellite phones that can be used anywhere in the world. However, these options all require active human engagement and are only distributed to relief workers to communicate amongst each other. This method of disaster communication is heavily dependent on linguistic competence and ability of the responding entities to coordinate among themselves. There has been much research into this effort and telecommunications companies have been devising their own schemes to solve the problem[4, 5, 6]. While robust and comprehensive, these solutions are costly and require resources beyond what many smaller, less wealthy nations have. The need for more useful information in the localities not well-served by telecommunications infrastructure has been made clear by Dailey and Starbird, who found that the social media network Twitter was instrumental in disaster communications[7]. Getting local information from traditional broadcast media is difficult due to response personnel's unfamiliarity with local regions and a lack of communications infrastructure in rural areas[8].

An ideal wireless solution would be one that is low-cost, far-reaching and easy to set up[9]. Lowering cost will be integral to helping distribute emergency communications infrastructure where none or very little currently exists. Longer range communication will be necessary for maintaining information links to remote areas, and ease of setup and deployment will encourage participation in the network[7, 8]. This research seeks to provide one axis of communication between disaster victim and disaster responder by providing communications infrastructure for mobile phones as connection vectors for gathering information on where victims are and what help they may need. This work seeks to contribute the following:

- Long-Range wireless communication protocol for embedded systems
- Empirical experimentation design for the communication infrastructure
- Analysis of effectiveness of network design and implementation
- Analysis of vectors for improvement of the system

In this paper, a new long-range wireless standard will be introduced as well as two competing methods of utilizing it. An experimental design will be outlined along with its reasoning, gathered metrics, analysis of those metrics and future work planned to address any short-comings. The prime use case of this paper will be disaster response, paying special attention to communication systems.

CHAPTER II

Related Work

This chapter presents an overview of alternatives and how they pertain to the problem of emergency communications. While many have played a role, there are issues that must be addressed within all technologies or gaps in the research literature. This chapter serves to identify these issues and provide background knowledge of systems currently under consideration by the research community.

2.1 Electronic Communication Systems & Disaster Response

The topic of Emergency Communication Systems (ECS) using various technologies has been studied quite extensively. ECS encompasses any communications used to establish one-way or two-way communications to address emergency situations. In the ECS domain, it is very common to use wireless devices such as broadcast radios, cellular phones and satellite phones[10, 11, 12, 13, 14]. Mobile phone communication is often handled by telecommunications providers in the area, both publicly and privately owned or operated. Some of these providers - often referred to as "telecoms" - have proposed solutions that, while robust, can nonetheless be very expensive to implement. Such a system was proposed by Japanese telecom NTT Corporation, in which small-scale data centers would be mobilized to the disaster area to collect image and video information coded into layers into a layered storage system comprised of various storage technologies[6]. For medical assistance in disasters, work has been done on the viability of using mobile devices to link doctors to patients in remote areas where disaster victims wear wireless vitals sensors connected to the mobile devices[15]. This sort of solution would require broad appeal and active participation from the public only after the disaster victims are found. While this contributes to the overall readiness of disaster response, there is still a gap left in finding the victims. To that end, the American telecom AT&T has used unmanned aerial vehicles (UAVs) tethered to their mobile phone service network to allow those in need of assistance to make phone calls with their own devices[16]. While this would allow high data rate connections to be established for voice and and image data, it still requires proactive use of the mobile phone, making it of limited help to one who may be injured and incapable of initiating contact. Further, such calls require far more battery power than what is necessary just to find the location of individuals. Relying on victims to initiate contact would drain the batteries quicker, reducing the time emergency responders have to reach those who may be wounded.

2.2 Internet of Things

Internet of Things (IoT) describes the ability for devices not originally made for the internet to connect to it for various applications. Devices are able to connect to each other through a number of different protocols such as Bluetooth, WiFi, Zigbee and Z-Wave. Wired infrastructure is common and will continue to be for certain applications, such as power systems, manufacturing and any other application with security concerns. Hardware manufactures often combine these technologies in different ways to achieve the goal of inter-connectivity, leading many implementations open to interpretation. Recently, a new standard for connected consumer electronics has emerged in the form of Matter, which seeks to unify all of the hardware and software vendors under a common protocol[17]. This standard is still new and the result of industry collaboration by various consumer electronics companies that once relied primarily on WiFi and Bluetooth for their device communication. As of this writing, many companies including Google, Amazon, Samsung and Apple have committed to the standard, but only a hand full of devices fully supporting Matter are on the market. It is not likely to enter far into the industrial space given its focus on commodity consumer hardware.

In the commercial and industrial space, IoT is believed to hold promise in the areas of smart cities and infrastructure. A prominent area of research to this end is vehicle-to-everything (V2X), in which vehicles are outfitted with wireless transceivers to communicate with other vehicles and the infrastructure around them. The possibilities are vast and offer potent solutions in the area of safety. Wang proposes an algorithm for predicting intersection collisions based on information that would have to come from IoT infrastructure embedded in the road system or communicated directly from other vehicles[18]. Traffic efficiency is another area of IoT research where Skoufas proposes a traffic system using traffic lights that dynamically route traffic by communicating directly to vehicles[19]. However, the security implications of these systems involves many considerations of privacy due to the constant exchange and validation of information using certificates[20]. Vehicles and people moving through space has necessitated that these networks be wireless, which has generated a thriving research community focus on wireless networking strategies that can cope with dynamic environments. In the realm of ECS, this technology has been proposed to decrease response times to vehicle accidents by use of a device installed in the vehicle that communicates location and severity information to a server[21].

2.3 Wireless Networks

Wireless networks comprise many different protocols, standards and bands on which to implement them. The most commonly used band is the Very High Frequency

(VHF) band, used by a number of public safety services like EMS weather radio and fire services. IoT has led to interest in Wireless Sensor Networks (WSNs) as an avenue of research in the emergency response realms of navigation[22] and medical [23, 24]. WSNs and Wireless Ad-hoc Networks (WANs) for ECS has seen interest from the research community related to natural disaster scenarios [25, 26]. These studies often use WiFi and Bluetooth[10, 14, 27, 28]. But a promising young technology called LoRa - standing for **Lo**ng **Ra**nge - is gaining traction in the research community [29, 30, 31, 32].

2.3.1 Wireless Sensor Networks

Wireless Sensor Networks (WSNs) are networks intended to monitor physical spaces or things in those spaces. Common applications are agriculture and smart homes, for monitoring crop health and home safety respectively. In agricultural IoT, a common theme is monitoring sunlight, humidity and soil composition, with a number of solutions being devised using embedded systems and wireless technologies such as Zigbee [33, 34, 35, 36, 37] and WiFi[38, 39]. These studies show the practical design of low-cost systems but lack rigor in their analysis of the networking aspect of what was implemented. Extending the life of these systems is the primary goal of energy harvesting, which typically involves using a combination of batteries and solar cells, as demonstrated by Saxena [40]. Further research has been done outlining numerous ways to reduce the energy cost of the sensor nodes. The main contributions were through managing the transmission of data through limiting the length of duty cycles of the networking and application hardware and using different hardware for different tasks based on the power necessary for those tasks[41]. This would look like using two different channels on different radios for transmission and reception, where the receive radio is of lower power and the transmission radio is turned on only when necessary.

2.3.2 Mobile Networks

Mobile networks include 4G, 5G and in the future, 6G. All of these technologies are intended to be widely available and in the case of 5G, accessible from virtually any device through what is called the 5G Core network. In this design, a service-based architecture is used that allows third parties to implement the services they intend to use or make available to others[42]. The appeal of mobile networks is primarily rooted in how expansive they are while providing general access to the internet. This allows for higher data rates and cloud-based applications to enhance data-intensive processes like machine learning (ML). The drawbacks are in power consumption, cost and security. Cost of backhaul technologies showed wireless to be most cost-effective only when fiberoptic wiring was over $25 USD per foot[43]. While cost of networking material is certainly a factor, reliance on prices being high is a suboptimal motivator for using less resource-intensive methods of providing connectivity in remote areas, as mobile networks as expected to do. Power consumption can be reduced by mitigating consumption of the physical layer, compressing data and decreasing interference in communication channels[44]. The main contribution in proposed guidelines is at the level of infrastructure. While comprehensive in approach it requires broad adoption by industry to make it a reality. Security is possibly the most harrowing concern. Security issues include improper network slicing, resource sharing risks where multiple applications can run on 5G-accessible hardware and exposure to the internet, exposing the network to denial-of-service attacks[45]. These problems are all inherent to the highly interconnected and complex nature of mobile networks, and solving them will require similarly complex enhancements.

2.3.3 Mobile Ad-hoc Networks and Mesh Networks

MANETS are of increasing relevance in research given the prevalence of mobile phones that can act autonomously and connect opportunistically to other devices. A node in this network is assumed to be capable of moving from one place to another, so there is no static network in place for data to travel. In several studies, Unmanned Aerial Vehicles (UAVs) are used to construct temporary relays for users below, focusing on the optimal placement for of the UAVs[46, 47]. Additional proposals have been made regarding the architecture of the UAV systems in relation to the people they are intended to help[48]. This and similar approaches have the advantage of not being reliant on existing infrastructure and can be deployed by military and security personnel that typically have access to advanced drone hardware. The specific wireless technologies used by the MANETs formed by drones create can range from using IEEE 802.11a [49] to IEEE 802.11s [50] and Bluetooth Low Energy (BLE)[51]. The designs are simple enough to be deployable using less expensive UAVs that can be found in the commercial space. There is a current gap in such research however, as these studies currently do not take cost or energy consumption into account. While governments and other publicly funded entities can often be counted on to have the necessary funding in wealthy countries such as the United States, the same cannot be said of less-wealthy nations, and their individual communities may be stretched thin in terms of resources.

A wireless mesh network can be thought of as a type of Wireless Ad-hoc Network (WANET) or MANET. The key difference is reliance on pre-existing infrastructure such as routers. These networks are prevalent in homes and businesses, often making up the telecommunications and internet infrastructure. This is the type of network that makes up the aforementioned Matter,[17] Z-Wave and Zigbee protocols. These networks have numerous routing protocols and have been studied in the context of disaster response scenarios[52]. In disaster response, many of the forwarding protocols are for delay tolerant networks (DTNs). This is reasonable as most disasters see people moving from one place to another in search of help or resources. Wireless meshes and MANETs in particular are very suited for this due to their ability to expand widely.

Some use routing tables to predict the probability of reaching another node or use some criterion for whether to pass a message to another node that is encountered. The protocols are broadly in two categories, routing and flooding. Routing protocols require some table of information about nodes that are nearby or are quick to reach. This often includes an ID and information about latency, but it can optionally include Received Signal Strength Indicator (RSSI) information in the case of wireless meshes to determine the best link state. These tables are integral to the implementation of such algorithms but can incur overhead if the tables are large. The flooding category comprises algorithms that are largely heuristic in nature, meaning they make simple calculations about whether to broadcast a packet to other nodes in the network. These algorithms require less memory in general, but they can easily congest the network if not implemented well.

2.3.4 LoRa

LoRa, standing for Long Range is a low-frequency wireless band that can reach much farther than typical 2.4 Ghz or 5 GHz bands, and is currently being used in maritime and wildlife conservation applications. Frequency ranges are on the sub-GHz band with values dependant on region for standards such as LoRaWAN. The technology is intended to enable communication over Low-Power Wireless Area Networks (LPWANs), and will be explained in more depth subsequently.

A LoRa-based mesh network has been evaluated in simulation by Pham et al [29] and found to be promising. The simulation predicted that bandwidth (BW), coding rate (CR) and spreading factor (SF) would be key predictors in range of communication and packet delivery ratio (PDR). This is in agreement with previous experimental work on LoRa's characteristics[53]. LoRa has also been evaluated for its potential as an ECS, with researchers finding success for limited communication equivalent to SMS messaging[31, 30].

Due to the extreme environment any ECS must endure, research into power consumption is the next logical avenue of inquiry. Pursuant to that, several contributions have been made examining the energy consumption of the different stages of node operation[54, 41, 55, 56]. These studies modeled the energy consumption of the physical layer and evaluated the parameters that affected it. Payload size, spreading factor (SF) and transmit power of the radio were shown to be particularly relevant in the efficiency per bit and overall power consumption[57, 53]. It has also been found that energy per useful bit goes down exponentially with smaller coding rates[55]. This suggests LoRa could be a viable technology to transmit data from nodes on the ground that collect information from people and infrastructure with the proper parameters. In a disaster scenario where traditional wireless networks such as mobile 4G could be unavailable, a LoRa-enabled network not reliant on mobile towers could be very valuable if an independent power source such as solar cells with batteries is used. What has not yet been discussed in literature is robust systems using LoRa technology in the disaster response space. In this paper, we examine the use of LoRa nodes in a mesh network to glean useful information about data rates in an empirical setting. The Federal Emergency Management Agency (FEMA) outlines four phases of disaster response: Mitigation, Preparedness, Response and recovery[58], which can be expressed as a cycle illustrated in Figure 2.1.

A LoRa-based network should ideally be in place before an emergency evolves into a disaster, so the successful system would be implemented in the Preparedness phase where an organization would set up systems for the Response phase. Response is when the network would be active and receiving telemetry. LoRa is a good candidate for this area of disaster response and preparedness due to its uniquely low power consumption and ability to reach long ranges on low power. While mobile networks can achieve this currently, they require very expensive infrastructure that must have support from power grids in the area. LoRa has potential to address issues that arise

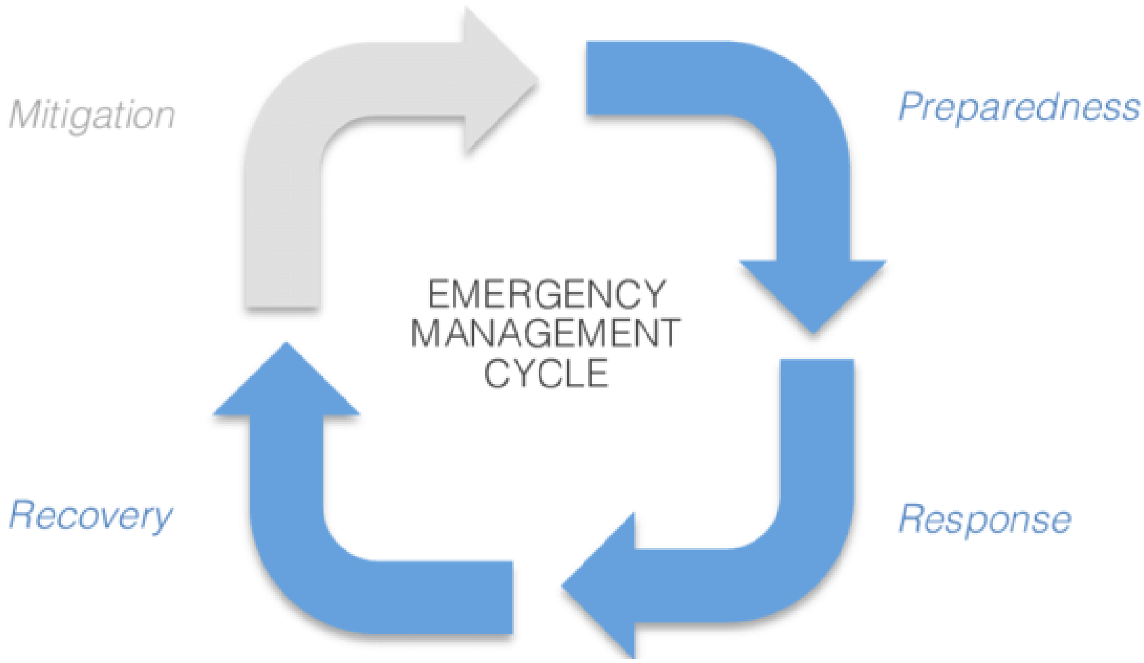


Figure 2.1: The four phases of disaster management

in rural areas or places where mobile network infrastructure is scarce or unreliable.

CHAPTER III

Background

This chapter goes into more depth about the LoRa wireless technology and the options for hardware using it. LoRaWAN is explained along with a new protocol that aims to achieve the same goals in a mesh networking context. While this new protocol is not LoRaWAN compliant, the standard heavily influences said protocol's design and is important to understand.

3.1 LoRa

To understand the problem in question, we must first gain some background knowledge of LoRa and what it really is. As elucidated by Liando et al, LoRa is a wireless technology that uses unlicensed ISM bands and employs a variation of Chirp Spread Spectrum[53, 59]. LoRa's physical layer is described by chirps, which incorporate channel, bandwidth, spreading factor and transmission power. Spreading Factor is a value that describes how spread out a chirp should be over a given bandwidth. Spreading Factors range from SF6 to SF12, with SF6 having the highest data rate and SF12 having the lowest. SF6 requires the highest signal-to-noise ratio while SF12 requires much lower at the same transmission power. [53] The relationship to bandwidth is given as a description of the duration of a chirp:

$$T_{sym} = \frac{2^{SF}}{BW} \tag{3.1}$$

T_{sym} is the time to communicate a symbol, or chirp. Bandwidth BW is the width of the transmitted signal. It should be noted that higher transmission power can counteract the effects of attenuation caused by environmental obstructions[53].

3.1.1 LoRaWAN

In addition to the physical layer of LoRa, there is a need to regulate the manner by which each of the physical devices interact. To make use of LoRa's physical layer, there is a need for defining networking parameters using the hardware. To address this, a standard called LoRaWAN was devised. Building on the LoRa physical characteristics, the LoRa Alliance has developed a specification that defines three classes of device[60]. The classes are Class A, Class B, and Class C, with Class A being the default that must be supported by all LoRaWAN-compliant devices.

3.1.1.1 Class A

Described as an ALOHA-type protocol, class A is bi-directional and up-link can be initiated at any time by an end device. That initial up-link phase is then followed by two down-link phases, enabling the bi-directional communication. This is the lowest power device type because it is allowed to go into a sleep phase whenever it is not transmitting or receiving.

3.1.1.2 Class B

This class of LoRaWAN device is also bi-directional, but deterministically so. This means that Class B devices open periodic down-link slots at scheduled times regardless of the state of other nodes. This latency can be up to 128 seconds.

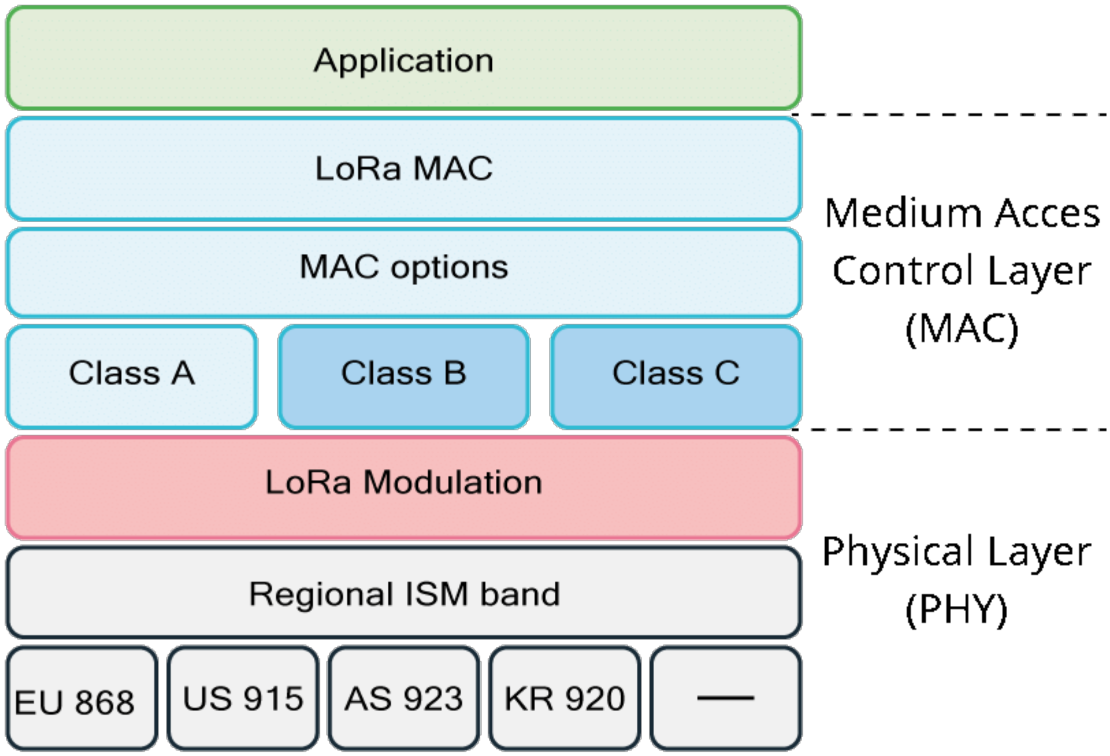


Figure 3.1: LoRaWAN PHY layers as outlined by Kim, et al [1]

3.1.1.3 Class C

This class of device has the lowest latency of the three, keeping the down-link receiver open at all times the device is not transmitting. This results in the highest power consumption of all classes, and thus is only suited to use with continuous power sources.

LoRaWAN baud rates range from 0.3 kbps to 50 kbps[60]. Security has two layers, a 128-bit session key shared between the end device and network server, and an application session key shared end-to-end at the application layer. AES is used for encryption at both layers[61]. LoRaWAN is a standard that primarily regulates the physical layer, uplink and downlink communication of the device. However, it also mandates back-end connected infrastructure such as network and application servers responsible for managing the network and the data transmitted through it. [60]

3.1.2 ClusterDuck Protocol

The ClusterDuck Protocol (CDP) [62] is a software implementation for allowing LoRa radios to send, receive, and relay messages to a sink node. There are a number of different types of nodes referred to as "ducks" that can be created through this software, but three main types will be discussed here. These different types of nodes are not LoRaWAN compliant and do not map neatly to the device class structure outlined by the specification. Where LoRaWAN defines how nodes must handle messages and outlines specific commands to be implemented, the CDP does not make mandates on how often nodes communicate or process data. This implementation features several LoRaWAN-like capabilities such as built-in encryption, bi-directional communication and mesh networking as shown in Table 3.1.

The nomenclature for the different node types is themed around ducks. There are three main types, differentiated by role and capability when deployed. Though the hardware used for each type could be the same, what type they are is determined by function.

3.1.2.1 DuckLink

This type of node is intended for communication of sensor or status information from a device it is connected to. They optionally can have Wifi or Bluetooth on to allow other devices to connect and communicate information over the LoRa network. This class of device typically requires the least amount of power and can be put to sleep without disrupting communication elsewhere in the network.

3.1.2.2 MamaDuck

This node inherits all the capabilities of a DuckLink, but adds the ability to relay messages coming from other ducks to the rest of the network as long as it is not in a sleep state and can access the LoRa radio. Bluetooth and WiFi can optionally be enabled on this node type.

3.1.2.3 PapaDuck

This node type is intended to be the message sink for the network. It can receive messages like any Mama Duck but is capable of sending data over Wifi to enable internet connectivity. This type is optionally allowed to communicate via LTE or satellite networks if additional hardware allows.

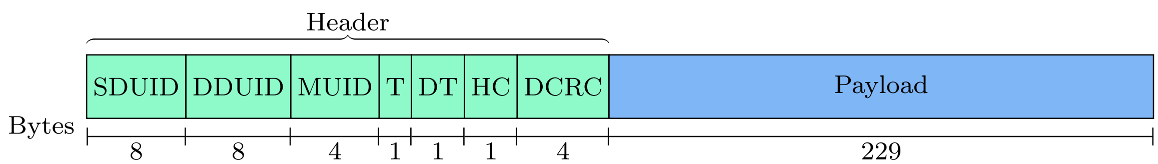


Figure 3.2: The message structure is byte-delineated and variable

To enable message propagation, the ClusterDuck Protocol defines a packet structure consisting of header and payload fields. Described in Figure 3.2, a 256-byte packet is used. The header consists of several fields contributing telemetry to the protocol stack.

- Destination Device Unique ID (DDUID)
- Source Device Unique ID (SDUID)
- Message Unique ID (MUID)
- Topic (T)
- Duck Type (DT)
- Hop Count (HC)
- Data Section CRC (DCRC)

The Topic refers to a message type system. The requirement of this is built into the communication standard due for future support for things like status messages,

Feature Parity		
Feature	LoRaWAN	CDP
Mesh networking	Yes	Yes
Multiple node types	Yes	Yes[1]
Encryption	Yes	Yes
Two-way communication	Yes	Yes

Table 3.1: How the CDP compares to the standards of LoRaWAN

commands and other specialized communiques. The DATA field is variable in length but must start after the 27-byte header. The ClusterDuck Protocol is implemented with a bloom filter to reduce network congestion, with each node maintaining its own record of what messages it has seen using the MUID. The CRC will only be computed on the payload section in this implementation. These features together achieve a managed flooding technique for a mesh network.

An important note about CDP is that it is purpose-built for telemetry and two-way emergency communications, so it goes beyond LoRaWAN in some ways. While lacking the specific node up-time and response behavior dictated by LoRaWAN, it is nonetheless a meaningful contribution to its intended problem space. The motivation for using the CDP project is two-fold. The first concern is the ease with which nodes can be implemented and distributed. Due to the lack of resources possessed by many communities beset by climate change, it is paramount that the technology used be low-cost and relatively easy to program. The CDP is open source under Apache License 2.0 and the LoRa boards are inexpensive at $20-$40 depending on capability. The second concern is technical availability. The CDP project has been active since 2018 and an open source project since March, 2020. [63] That has produced a software package that can be used by anyone around the world to meet the immediate demands of regions of the world where low-cost computing can make the highest impact.

[1]While both the standard and CDP have 3 node types, CDP's node types do not currently map to the standards set by LoRaWAN

CHAPTER IV

Deploying a LoRa Mesh Network for Post-Disaster Communications

This chapter defines the axes on which the LoRa mesh network will be evaluated and equations that define those axes. A brief set of assumptions is also made, outlining what type of equipment should be expected in experimentation.

4.1 Problem Definitions and Assumptions

Definition 1. ***Duck Network***. *The Duck Network is a mesh network* $G = (V, E)$ *where* G *is a connected graph of a set of LoRa nodes* V *where* $v_i \in V$ *and a set of wireless connections* E *where* $e_i \in E$ *and* : $|e_i| = d_{v_i,v_j}$ *where* d_{v_i,v_j} *is the distance between any two nodes* $(v_i, v_j) \in V$

Definition 2. ***Delay***. *The time committed to the database will govern the network travel time and can be computed simply as the difference of that time and transmission time.*

$$T_{sink} - T_{node} \tag{4.1}$$

where T_{sink} *is the time a packet is received, and* T_{node} *is the time transmitted from the originating LoRa node as indicated from the timestamp within the packet as collected from GPS.*

Due to the non-trivial latency incurred by the Micro-Control Unit (MCU), calculating values is done taking into account the delay due to MCU processing time, where the delay in milliseconds is added to the timestamp value received by the source node and subtracted for the value at the sink node. This ensures the delay measured is that of the time messages spend in transit without inflation due to hardware latency. We can then revise the general equation by refining T_{sink} and T_{node}:

$$T_{sink} - T_{node} \tag{4.2}$$

such that

$$T_{node} = T_{GPS} + D_{MCU} \tag{4.3}$$

and

$$T_{sink} = T_{NTP} - D_{MCU} \tag{4.4}$$

where D_{GPS} is the timestamp according to the GPS system, D_{MCU} is the millisecond delay spent processing data on the MCU and T_{NTP} is the timestamp according to NTP accessed through the internet-connected router serving the sink node.

Definition 3. ***Data Rate***. *Simple calculation of bits per second where the number of bytes for the full packet is considered.*

$$\frac{B_{packet}}{(T_y - T_x)} \tag{4.5}$$

where B_{packet} is the number of bytes that make up the packet, and $(T_y - T_x)$ is the time of commit to database minus the time of transmission from LoRa node.

Definition 4. ***Loss Rate***. *Loss rate LR is found using the number of packets not successfully committed to the database, either due to corruption upon receipt or loss after transmission.*

$$LR = \frac{R_x}{T_x} \tag{4.6}$$

where

$$Rx = T_x - (C_x + L_x) \tag{4.7}$$

where R_x is the number of packets successfully committed, T_x is the number transmitted, C_x is corrupt packets and L_x is packets lost in transmission.

Definition 5. ***Jitter***. *Jitter J is the mean difference in delay between packets in a collection transmitted each interval.*

$$J = \frac{\sum_{n=1}^{n} T_{n+1} - T_n}{n} \tag{4.8}$$

where T_n is the time in delay in milliseconds for a packet in an n sequence of packets.

4.2 Assumptions

In the event of a natural disaster, it is paramount that EMS personnel respond quickly and efficiently. To do this, information is needed on where people are and how long it might take to reach them. To do this, a mesh network using LoRa nodes communicating via the CDP is proposed with a laptop acting as a sink for the data generated by each node. The data generated is a stand-in for real people who would be assumed to have a mobile smartphone capable of GPS and Bluetooth.

The research presented here will attempt to provide insight on LoRa's suitability for relaying data under the constraints of a natural disaster. This means constraining all LoRa nodes to battery operation.

A wireless LAN created for the PapaDuck to connect a timeseries database is permitted, given that such can be created by simple consumer routers trivially. One concern that has been made clear by past research is the impact of payload size on power consumption of the LoRa physical layer[53, 55, 56]. The packets transmitted will conform to the CDP packet structure outlined in Figure 3.2. Details on the data transmitted, network design and hardware used will be in Chapter V.

CHAPTER V

System Model & Methods

This chapter outlines the possible avenues of experimentation while indicating which avenue was taken. It then discusses the general network design that would be used regardless of method along with a data serialization format to be used.

5.1 Preliminaries

To properly simulate usage of the network by end-devices, it is important to capture the unpredictability of natural disasters so that the data can be viewed in the proper context. In ideal circumstances, each node on the network would be a mobile device with Bluetooth that can anonymously send a payload to the network upon detecting a node. This would realistically require cooperation from mobile phone manufacturers to allow the end users of their devices the ability to volunteer certain non-identifying information to be periodically transmitted via Bluetooth to disaster response infrastructure in the area. While a mobile app might suffice from a technical standpoint in the short term, ensuring maximum coverage of this capability would necessitate a standardized communication payload and ability to modify the information to be inserted into that payload from the user interface. Furthermore, it will be necessary for the nodes of the network to be in place beforehand or at least rapidly deployable in the aftermath of a disaster.

5.1.1 Experimental Design Options

There are four types of experimentation that can be done, and this study will choose one type in section 5.4. Due to varying degrees of involvement with the public, some experiment implementations have more ethical considerations than others. These ethical considerations are weighed against the current needs of the study to find the most suitable course of action to meet the goals.

5.1.1.1 Full Simulation

This option would use a network simulation framework such as OMNet++ to simulate the packets traveling in a mesh network with the proper data rate propagation rules accounted for in the simulation software. This would have the advantage of being the least expensive option monetarily, requiring no hardware to be acquired and not involving the public in any way. Bluetooth communication to the LoRa nodes from clients would also be simulated, leaving no ethical implications to be considered because the public would never be involved. The drawback to this option is the lack of concrete data it would provide given the lack of attenuating structures that would realistically impact results. This method would also only speak for network communication and leave out important data about computation time of the low-power compute node that is integral to this study. This means the network communication and compute node investigations would have to be done divorced from one another. While that is possible to do, synthesizing the results together to form a more complete conclusion would still eventually require more rigorous empirical confirmation later.

5.1.1.2 Hybrid Simulation & Experimentation

A hybridized experimentation is one other option afforded to us that would see use of an actual LoRa mesh network and compute node deployed within downtown Flint, MI in order to study the realistic effects of a built-up area on LoRa communication.

Bluetooth is left out in favor of the most relevant portion of the network, the LoRa mesh. To send data over the network, GPS coordinates and metadata attached to them would be generated by deployed LoRa node with GPS capability to be sent. The GPS coordinates generated would be within a certain radius of the node that generated them to simulate people spontaneously being found in an area as the LoRa nodes make contact with their Bluetooth-enabled mobile devices. A compute node would be deployed on campus with a connection to a PapaDuck that could receive the information from the network. This method has the advantage of providing more empirical evidence for the study, while continuing to leave the public out of the study, so no ethical concerns need to be addressed. The drawback is the lack of data regarding Bluetooth's suitability as a beaconing method for client devices. With this experimental design, the core of the research problem is accounted for with a wireless network to adequately evaluate suitability for relaying distress messages across a landscape.

5.1.1.3 Hybrid Simulation & Experimentation with External Bluetooth Data

This option is nearly identical to the previous, only adding a second experiment regarding Bluetooth communication in downtown Flint, MI. The Bluetooth experiment would involve scanning the area for Bluetooth-enabled devices to glean information about people's movement in the area, then using that data to inform how to simulate clusters of people on the LoRa nodes. This is the first option to trigger an ethical consideration that must be overseen by and Internal Review Board (IRB) because of the public's unwitting involvement in a scientific study. Gathering any information about mobile phones present in a vicinity should be treated as ethically suspect, requiring more time by the researcher to maintain the privacy of individuals and ensure ethical boundaries are not breached. This method would speak to the suitability of

Bluetooth as a client-side wireless technology for reaching the mobile phones that would be carried by the average disaster victim.

5.1.1.4 Full Empirical Experimentation

This option is the most expansive, and is essentially the empirical conductance of the experiment previously outlined, with no simulation of clients or GPS information. Semi-randomized payload data would still be made, but all data would be received from actual mobile devices running an application or a Bluetooth-enabled, GPS-enabled beaconing device given to volunteers specifically for this experiment. Depending on what the client device is, there would be different ethical concerns for this option. If a mobile application were used, an IRB would require time to be assured there is no capability for the app to submit personal information through the network or collect any impertinent data from the mobile device the application runs on. If a stand-alone beacon device is used, the concern about data collection and retention is eliminated, but effort must still be made to reassure any volunteer will only have their position tracked while in the downtown area, and with no personal information about them bound to the device identifier. The benefit to this mode of experimentation could be high, ensuring the most empirically accurate representation possible for the study, enhancing its usefulness to the scientific community. The most robust of the four, this would provide the greatest level of insight into the system's efficacy in solving the proposed problem.

To get the most robust experimental data possible, a fully empirical experiment was performed. Empirical experimentation is the only way to capture fully the strengths and weaknesses of an implementation, and what is most important in emergency communications is getting unsimulated data. Certain obstacles like the effects of attenuating infrastructure are very difficult to simulate accurately, and data regarding imperfections is often just as valuable as data about perfect scenarios. For the sake of privacy and ethics, Bluetooth communication with pedestrians was not attempted.

5.2 Network Design

The network topology is a mesh network as described in Definition 1 in which each node is capable of relaying messages, but a subset of those nodes are capable of generating them. Figure 5.1 illustrates the connectivity of the mesh, with each MamaDuck capable of relaying messages from other MamaDucks, while the PapaDuck is the sink. Each node is to have line of sight to at least one other node. While any node in a ClusterDuck network can transmit telemetry, a subset of them will not generate simulated telemetry.

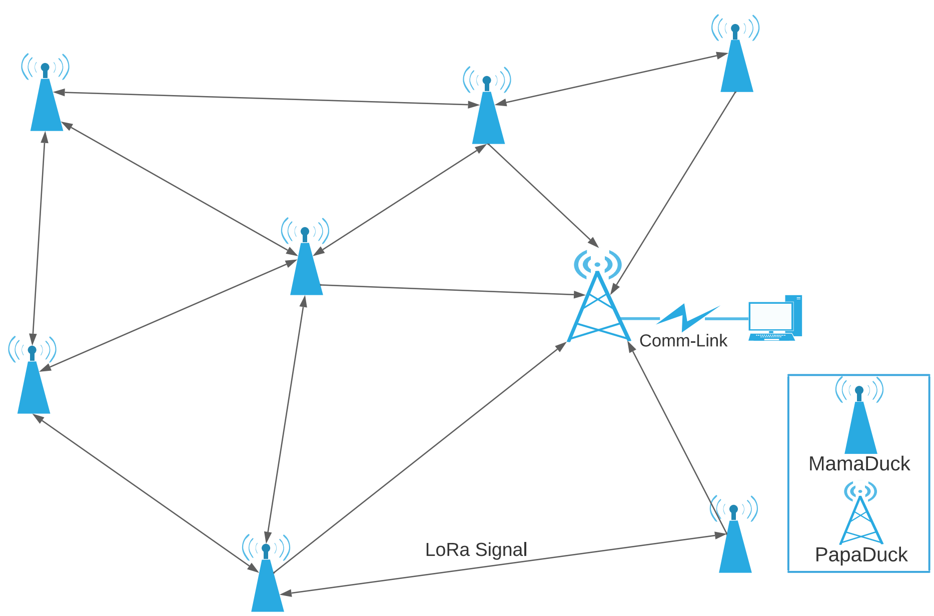


Figure 5.1: Nodes in a network broadcast to any other node, while PapaDuck acts as a sink

5.3 Data Model

To accurately simulate the type of messages that would both be helpful in an emergency scenario, protect the privacy and personhood of an individual and still be small enough to not congest a low-bitrate network, a data model is proposed in JSON as shown in Listing V.1. JSON is a very common data representation standard well-known in industry around the world for its ease of use and logical structure.

Listing V.1: JSON representing various metadata about a person


```json
{
  "Device": "MAMAGPS4",
  "seqID": "JNE6T3",
  "seqNum": 2,
  "MCUdelay": 2338,
  "GPS": {
    "lon": 42.7610285,
    "lat": -84.6006325,
    "satellites": 8,
    "time": 1679887867,
    "alt": 269.4
  },
  "Voltage": 3042,
  "level": 1
}
```


5.4 Methodology

This section explains in detail what has been discussed over Chapter IV with a fully empirical experiment being chosen as the evaluation methodology. As noted earlier, empirical data is required to get a complete insight into the suitability of a technology, thus this experiment will use physical hardware and a full implementation of the ClusterDuck Protocol. The hardware, software, node placement and method of data collection are all discussed with special attention to reasoning for the experimental design.

5.4.1 System Implementation

To construct the network, the two node varieties detailed in Section subsubsection 5.4.1.1 will be distributed throughout a wide area of the University of Michigan - Flint campus. To provide realistic coverage of the area that is most likely to be occupied by the average human, we place most of the nodes near the ground, while some are positioned higher in the air. The manner in which this is done attempts to make use of line-of-site placement as much as possible. The TTGO LoRa modules are placed at an elevation close to the ground, between approximately 3 to 4 meters in height. It is important to note that these nodes must remain within Bluetooth range of the ground, but out of reach of tampering by the typical adult to ensure the continued function of the network. The Heltec LoRa module that lacks GPS capability is used for routing messages sent by the TTGO LoRa nodes. These Heltec modules must be placed at a much higher elevation to mitigate interference from low-standing infrastructure. The height for most Heltec modules is 10 meters, situated in the windows of campus buildings or the top floor of parking garages. The PapaDuck remains outside of any buildings but uses a battery power source on the university campus and connected to the database via an 802.11 network. This implementation strategy assumes that some infrastructure is still standing, and the structures for affixing the LoRa nodes must be chosen carefully to provide the most realistic representation of what can be accomplished after a natural disaster.

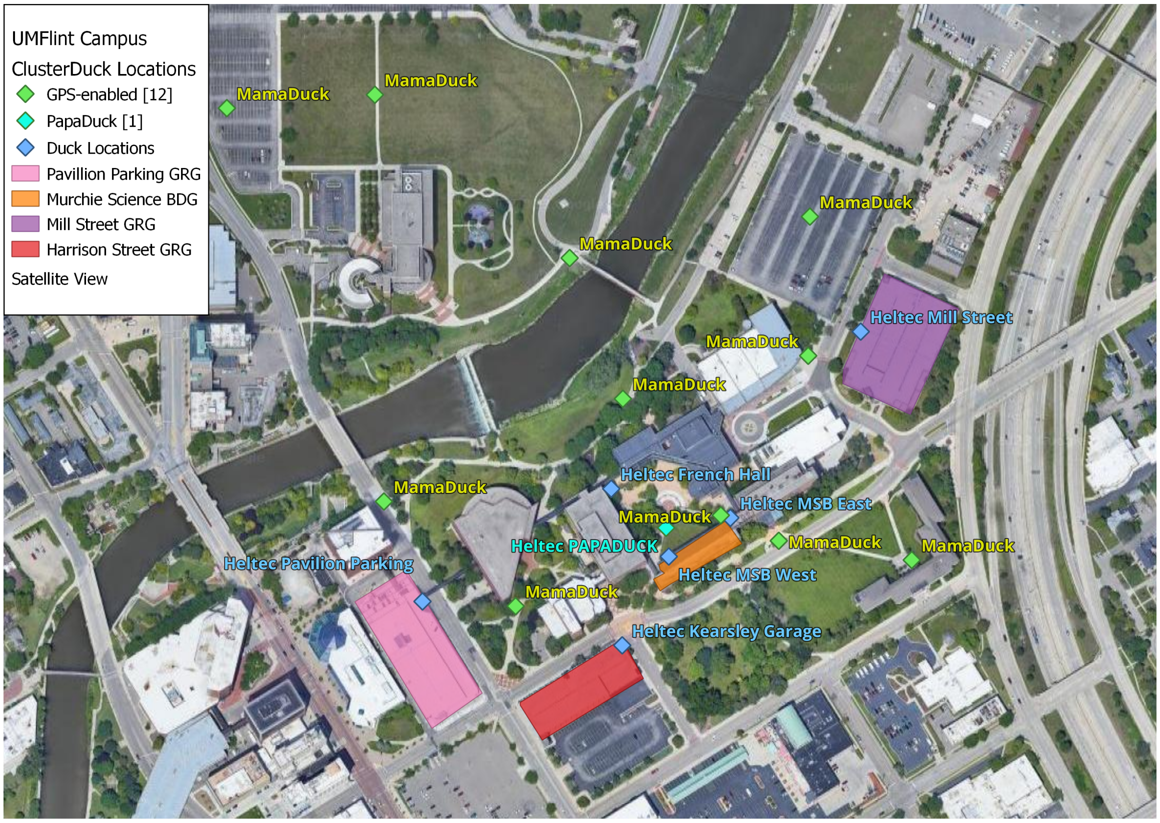


Figure 5.2: Map of Duck Network layout across UMFlint campus

5.4.1.1 Hardware

There are two types of LoRa radios that will be used. One, the LILYGO TTGO[64], will have GPS functionality. The other, the Heltec LoRa 32[65], will only have LoRa functionality. Each node will be equipped with a battery for power. Batteries to be used are 18650 cells for the TTGO modules and Lithium Polymer batteries with SH.125 connectors for the Heltec modules. In this experiment, all nodes will be MamaDucks with the exception of one Heltec LoRa 32 acting as a sink node, which will be a PapaDuck.

5.4.1.2 Data Collection

A payload of JSON formatted data will be communicated by the nodes to simulate an emergency beacon that could be sent by a smartphone or specialized beaconing

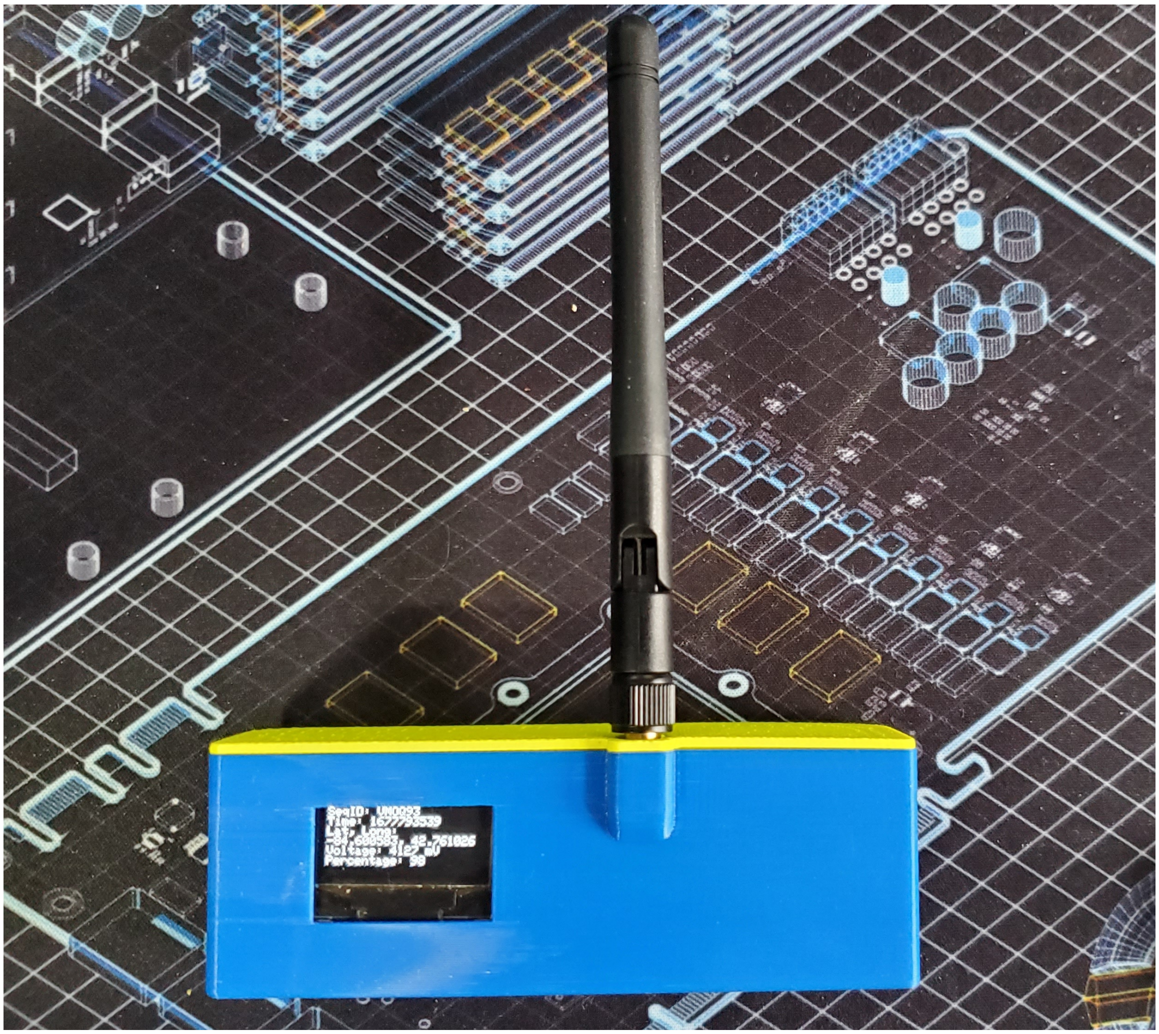


Figure 5.3: A LILLYGO TTGO LoRa module with an enclosure

device. Two experiments will be run that simulate random arrival and inter-arrival times of people wandering through a disaster area seeking help. The first will have TTGO nodes generate telemetry in sequences of four packets according to the uniform distribution and the second under an exponential distribution. These distributions have been chosen to best simulate the circumstances that each is adept at modeling. A uniform distribution is often used for modeling the likelihood a certain number of arrivals will happen in an interval. The interval used will be 5 minutes to simulate a periodic sensory scan of the environment for active mobile phones of users. To get a meaningful jitter calculation and ensure an appropriate comparison, 4 packets are sent each transmission period in both experiments. The radios transmit 4 packets to ensure there are enough packets to calculate a jitter value in the event any are lost. The exponential distribution is best suited for modeling when something will happen given an average. This serves to simulate new arrivals since the last dispatch of packets, with telemetry regarding those arrivals being sent immediately. The mean is 2.5 to simulate fragmented groups of people congregating among each other before arriving within range of a node. 2.5 also happens to be the average household size in the United States, increasing the value of this experiment.[66] All values chosen are pursuant to a best-effort modeling of behavior in a disaster scenario. All database communication is handled on the PapaDuck acting as the sink node. The timeseries database is InfluxDB, a column-oriented database written in the Go programming language with a very low memory footprint running on a consumer-grade laptop. No additional software is used to "clean" the data and metrics persisted within it are raw values directly from the PapaDuck.

5.4.1.3 Software

The ClusterDuck Protocol is written in C++ using a number of Arduino-compatible libraries including RadioLib and the espressif IoT Development Framework. To gather telemetry on the true error rate due to attenuating infrastructure, it is necessary to make one minimal modification to the ClusterDuck Protocol by disabling the CRC-based error-checking so corrupt packets can be counted. This is due to the Protocol's current behavior of silently throwing away corrupt packets, which would not allow gathering of information about them and their prevalence. The packets will be generated on random intervals using two distributions outlined in subsubsection 5.4.1.2. These distributions are implemented in the C++ standard library and are used to generate a delay in milliseconds until the next transmission period. The nodes will be programmed not to sleep for the entire duration of the experiment in order to get data on the worst case scenario of battery life. This is also done to make sure all nodes are able to relay packets that come from other nodes. The data is left unencrypted over LoRa transmission for all packets.

CHAPTER VI

Results

This chapter outlines all data found and delivers the meaning behind all data collected.

6.0.1 Jitter

For the sake of comparing jitter with and without delay due to MCU and local network time to the database, included is the jitter for an experimental run that did not account for time spent on MCU or LAN delay to the database in Table 6.1 and Table 6.2. The jitter calculation is easily determined per-node showing a roughly 2.1 to 2.2 second mean jitter for every node when transmitting on a delay modeled by a uniform distribution. When transmission times follow an exponential distribution, the mean jitter increases for most nodes in the network to as much as 2.7 seconds. On a subsequent run with more nodes, the jitter mean and median values increase, as shown in Table 6.3 & Table 6.4. a low mean of 2.029 seconds and high of 2.57 seconds is observed for the uniform distribution, with medians keeping pace at a low of 1.996 seconds and high 2.593 seconds. The exponential distribution sees a wider range of values with the lowest mean being 1.883 seconds and the highest being extremely close to 3 seconds. The median low and high are also skewed farther apart, with observed values of 1.477 and 2.611 seconds, respectively.

Node	Mean (ms)	Median (ms)
MAMAGPS3	2,116.45	2129
MAMAGPS4	2,027.89	1907
MAMAGPS6	2,147.67	2135
MAMAGPS7	2,149.70	1845.5
MAMAGPS9	2,149.08	1882
MAMAGPSB	2,246.36	2241
MAMAGPSC	2,195.78	2115
MAMAGPSE	2,200.66	2068

Table 6.1: Per-node jitter values including MCU delay: Uniform Distribution

Node	Mean (ms)	Median (ms)
MAMAGPS3	2,619.10	2201.5
MAMAGPS5	1,946.53	1845.25
MAMAGPS6	2,715.83	2452
MAMAGPS7	2,480.22	2266
MAMAGPSB	2,577.39	2157
MAMAGPSC	2,202.79	1776
MAMAGPSE	2,032.25	2032.25

Table 6.2: Per-node jitter values including MCU delay: Exponential Distribution

Node	Mean (ms)	Median (ms)
MAMAGPS1	2,569.54	2593.5
MAMAGPS3	2,357.91	2290
MAMAGPS5	2,405.29	2559
MAMAGPS6	2,592.29	2599
MAMAGPS7	2,049.58	2058
MAMAGPS8	2,167.46	1991
MAMAGPS9	2,439.66	2066
MAMAGPSA	2,466.97	2100
MAMAGPSB	2,357.78	2078.5
MAMAGPSC	2,028.56	1765
MAMAGPSD	2,119.28	1996

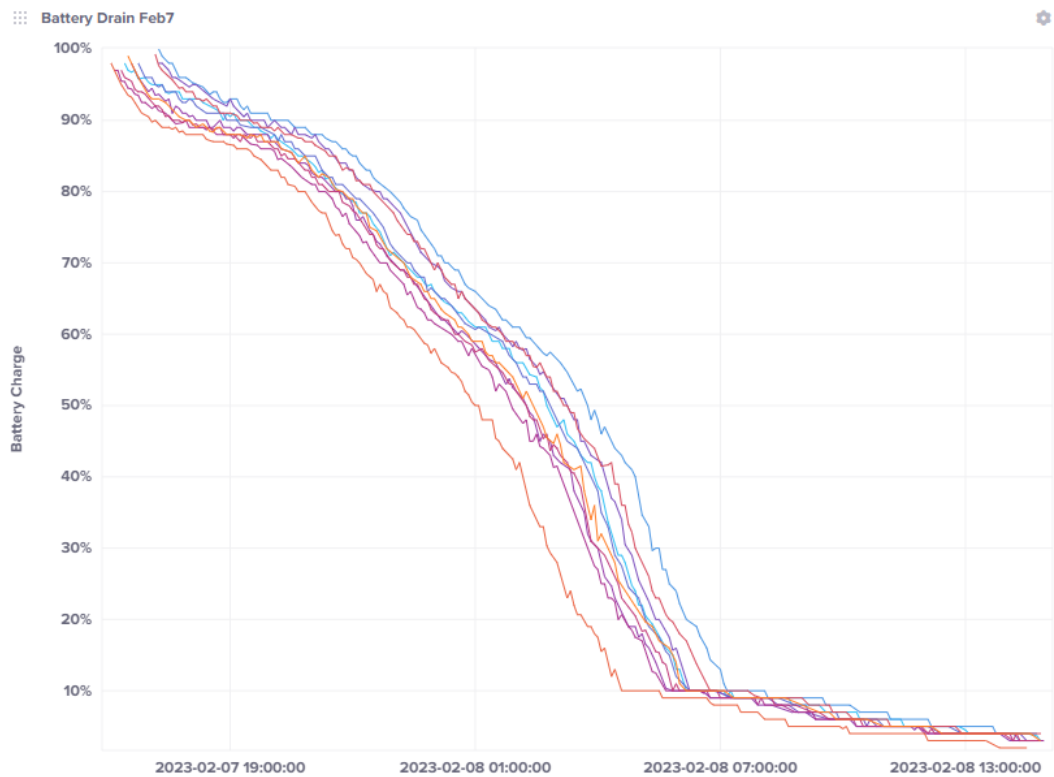
Table 6.3: Per-node jitter values excluding MCU delay: Uniform Distribution

Node	Mean (ms)	Median (ms)
MAMAGPS1	2,587.26	2611
MAMAGPS2	2,250.88	2051.5
MAMAGPS3	2,999.57	2690
MAMAGPS5	2,030.40	2004.5
MAMAGPS7	1,883.03	1477
MAMAGPS8	2,581.17	2462
MAMAGPS9	2,451.07	2589
MAMAGPSA	2,643.35	2335
MAMAGPSB	2,258.94	2005
MAMAGPSC	2,323.49	2072
MAMAGPSD	2,461.36	2527.5

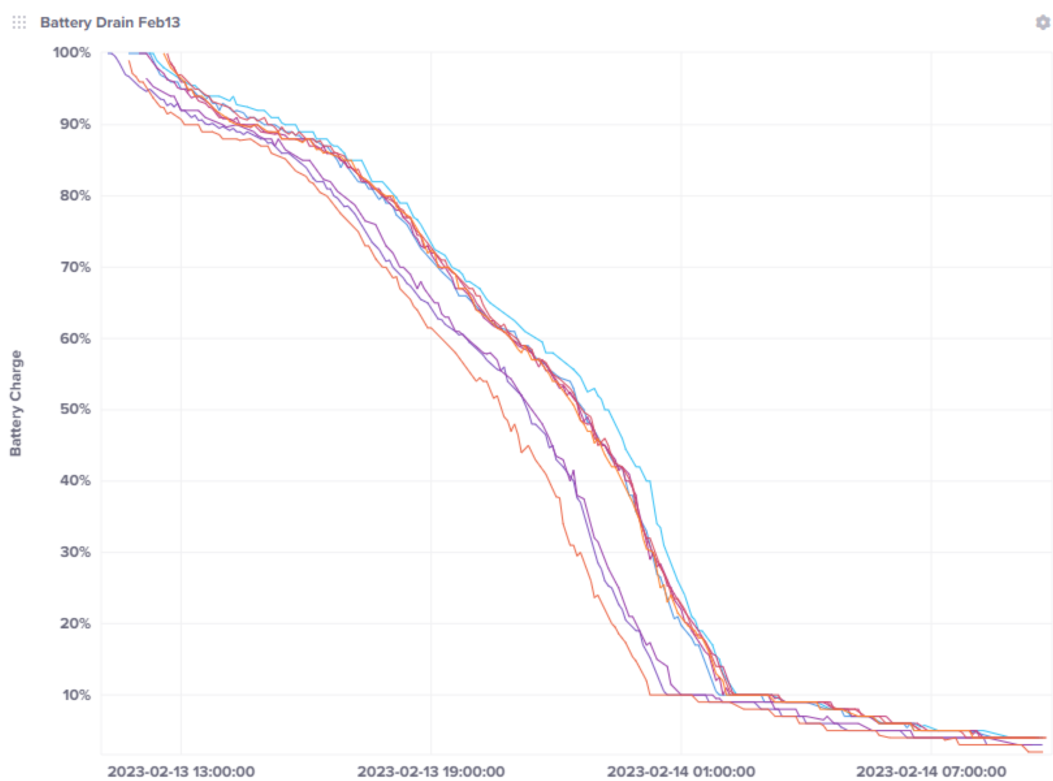
Table 6.4: Per-node jitter values excluding MCU delay: Exponential Distribution

6.0.2 Battery Usage

The battery usage of the TTGO GPS-enabled nodes gives us insight as to how long the nodes could possibly last in a scenario where power resources may be limited. When triggering transmission of packets according to the uniform distribution, the batteries drain on a predictable path down to approximately 10% of a 3000mAh battery's limit. This 10% value corresponds to the 3.7v rating of the 18650 batteries, but the MCU can persist much longer before dying. While the overall drain pattern is very similar between distributions, the discharge rate varies more widely under an exponential distribution. This is consistent with the expected behavior of such a distribution where some nodes will encounter more transmission-triggering events than others.



(a) Uniform distribution



(b) Exponential distribution

Figure 6.1: Battery drain pattern by percentage for T-Beams for message frequency (UTC-5)

6.0.3 Data Rate

Data rate is modest due to the low frequencies LoRa operates in, so it is unsurprising to see a range from 604b/s to 1175b/s for the mean data rate of uniformly distributed transmissions. For exponentially distributed transmissions, the mean tightens slightly, between 606b/s and 1129b/s. The same phenomena is observed for both median ranges of their respective distributions, indicating neither was biased toward the low or high ends of the ranges.

Node	**Mean (Packet)**	**Median (Packet)**
MAMAGPS1	1,071.11	1,032.90
MAMAGPS3	871.10	807.57
MAMAGPS5	1,102.23	1,045.45
MAMAGPS6	887.57	818.68
MAMAGPS7	664.59	631.34
MAMAGPS8	604.58	625.53
MAMAGPS9	1,140.84	1,090.18
MAMAGPSA	1,068.23	1,021.17
MAMAGPSB	1,070.70	1,007.41
MAMAGPSC	1,175.93	1,130.22
MAMAGPSD	660.82	626.85

(a) Packet Size

Node	**Mean (Payload)**	**Median (Payload)**
MAMAGPS1	929.93	897.02
MAMAGPS3	756.52	702.74
MAMAGPS5	957.11	908.16
MAMAGPS6	770.93	711.38
MAMAGPS7	577.21	548.39
MAMAGPS8	525.13	543.04
MAMAGPS9	990.56	945.89
MAMAGPSA	927.93	886.67
MAMAGPSB	929.50	874.07
MAMAGPSC	1,021.21	981.32
MAMAGPSD	574.22	545.04

(b) Paylod Size

Table 6.5: Per-node data rates calculated using packet size vs payload size in bits per second: Uniform Distribution

Node	Mean (Packet)	Median (Packet)
MAMAGPS1	787.60	693.31
MAMAGPS2	636.24	617.25
MAMAGPS3	855.74	793.14
MAMAGPS5	674.56	619.45
MAMAGPS7	1,129.57	1,103.92
MAMAGPS8	668.78	598.84
MAMAGPS9	1,063.50	1,029.36
MAMAGPSA	1,060.32	1,034.88
MAMAGPSB	967.28	950.09
MAMAGPSC	1,100.71	1,086.35
MAMAGPSD	1,054.43	1,033.65

(a) Packet Size

Node	Mean (Payload)	Median (Payload)
MAMAGPS1	684.16	603.08
MAMAGPS2	552.67	535.55
MAMAGPS3	743.12	690.18
MAMAGPS5	586.01	538.33
MAMAGPS7	980.82	956.79
MAMAGPS8	580.98	520.68
MAMAGPS9	923.67	894.44
MAMAGPSA	920.85	898.37
MAMAGPSB	839.71	826.16
MAMAGPSC	955.74	943.09
MAMAGPSD	915.47	898.19

(b) Payload Size

Table 6.6: Per-node data rates calculated using packet size vs payload size in bits per second: Exponential Distribution

6.0.4 Loss Rate

The loss is relatively high at 31.34% for the uniformly distributed transmissions. The total number of messages transmitted would be 12,272, or 3068 $\times$ 4 due to the static number of four packets transmitted each cycle. The number of corrupt packets received is actually greater than that of intact packets received, meaning there is likely a great deal of interference from attenuating infrastructure. The same behavior can be observed for the exponential distribution, with a very similar loss rate of 32.64%. The number of total packets sent is further broken down per node in Table Table 6.8a and Table 6.8b.

Metric	**Uniform Distribution**	**Exponential Distribution**
Transmitted	12,272	12,248
Packet Sequences	3,068	3,062
Corrupt	4,341	4,222
Complete	3847	4,028
Received	8,188	8,250
Not Received	4084	3,998
Loss Rate	**0.31348**	**0.32642**

Table 6.7: Loss metrics for uniform and exponential distributions

To get the number of corrupt packets, they were deliberately received so they could be counted along with some basic telemetry about them. However, the number could also be derived by rewriting Equation 4.7:

$$C_x = T_x - (L_x + R_x) \tag{6.1}$$

where R_x and T_x have the same definitions and L_x is equal to the packets not received by the PapaDuck.

To gain some intuition on how many packets were transmitted per-node, we can refer to Table 6.8a and Table 6.8b and see that number of transmissions can differ greatly. In the uniform distribution, the lowest count was 217 while the highest

Node	Mean
MAMAGPS1	423
MAMAGPS3	284
MAMAGPS5	394
MAMAGPS6	217
MAMAGPS7	236
MAMAGPS8	241
MAMAGPS9	462
MAMAGPSA	424
MAMAGPSB	373
MAMAGPSC	533
MAMAGPSD	260

(a) Uniform Distribution

Node	Mean
MAMAGPS1	260
MAMAGPS2	303
MAMAGPS3	199
MAMAGPS5	317
MAMAGPS7	563
MAMAGPS8	233
MAMAGPS9	441
MAMAGPSA	408
MAMAGPSB	426
MAMAGPSC	451
MAMAGPSD	421

(b) Exponential Distribution

Table 6.8: Per-node number of packets transmitted

was well over twice that amount at 533. Even greater extremes can be seen for the exponential distribution, between 199 and 563 packets. However, due to the chaos inherent to disaster scenarios and the complications of finding safe areas to congregate, these ranges of transmissions are not unreasonable.

CHAPTER VII

Discussion & Conclusion

7.1 Summary

The key findings that should be taken from the data are that while mesh networks are very achievable via the sub-gigahertz LoRa medium, there are fundamental principles of wireless networking that must be taken into account, such as infrastructure, power consumption and implementation, particularly programming. The data rate was low as expected, but the battery consumption was encouraging given that the first 36 hours are the most important in a natural disaster scenario. Tests have shown us that a generous window of over 24 hours is achievable, and with further optimizations multiple days are well within reach. While the error rate was high, this can serve as a warning to future LoRa deployments that validity and error checking for packets communicated is an imperative. The loss rate of 31% and 32% is not acceptable for a disaster response solution with the software implementation in its current form. In addition, there were imperfections in the way the experiment was designed, which contributed further to the problem, which we discuss in detail later in this section.

7.1.1 Challenges

While the ClusterDuck Protocol and LoRaWAN both have their own implementations for this, it is now clear that the ClusterDuck Protocol in particular must have its error checking active at all times. One other thing to note is the number of packets that weren't received at all. Due to the PapaDuck being modified to capture telemetry about all packets that come in rather than silently dropping corrupt packets as designed, the data overwhelmed the Heltec32's modest hardware when communicating with the database. More database communication necessitates more time on the MCU. When the MCU is performing a database commit, it does not have access to the LoRa radio in order to receive more transmissions from the LoRa mesh. This causes many messages to be missed completely, as we can see from the data in Table 6.7. To remediate this, one can simply slow down transmissions between packets. While this was noticed in a small, contained setting of only a few nodes even before the full empirical experiment with accurate delay times, it was necessary to see if the behavior would replicate itself at that scale. The determination was made to keep the rapid transmission behavior to judge the network's ability to handle packet bursts of this nature. It is now confirmed that the loss rate this yields is untenable and should be remediated. When attempting to triage a solution, a delay was introduced at the time after each packet transmission. In a very localized setting with just a few nodes, this resulted in all messages being seen and processed by the PapaDuck.

7.1.2 Errors in Experiment Design

Over the course of experimentation, there was a desire to collect as much metadata as possible. In order to do so, some code relating to catching bad packets was moved out of the protocol's implementation and into the message handler to capture that metadata. As mentioned earlier, this put strain on the MCU and caused the radio to miss new incoming packets, ultimately inflating the loss rate. With the benefit of hind-sight, we now know better to not interfere with the inner-workings of the software implementation, however, an updated experiment will be necessary.

Another finding after all testing was done was a critical bug in the implementation of the Bloom filter used to determine whether a packet has already seen and prevent flooding of the network. The ClusterDuck Protocol project found that the bloom filter's hash function was not seeded with sufficient entropy to yield random outputs for the filter. The specific problem was non-optimal bloom filter implementation and the use of a call to the C Standard Library's TIME() function as input to the srand() function. Since the hardware is not guaranteed to have the time via GPS or some other mechanism, using the current time is not an adequate solution for determining whether to relay an incoming packet via this method. Future remediation is planned through an update that will use a more robust re-implementation of the comparison mechanism to possibly not use a different container object. The current finding in these discussions is that a hash table implementation would be better by resulting in fewer collisions.

Beyond deficiencies with the ClusterDuck Protocol's implementation, there is one important aspect that will need to be addressed in future work. While the number of corrupt packets can be derived by simply subtracting the well-formed "complete" packets and not received packets from the total transmitted, telemetry about corrupt packets can still be useful. A way to get that telemetry without hindering the MCU's ability to receive more of the transmitted packets will have to be devised. Currently, the only solution seems to be having a second PapaDuck acting as a sink solely for the collection of corrupt packets. While the first attempt at this with only one PapaDuck impacted the experiment brutally, the metadata about corrupt packets that were caught was complete. This tells us that the method for doing this can be easily offloaded to a separate PapaDuck node that specifically handles inverse case of a well-formed packet, i.e. an irrecoverably corrupt one.

7.1.3 Future Work

While future work is planned to address all the deficiencies with the experimental design, there are still more questions to be answered regarding LoRa mesh networks for ECS as a whole. One avenue is in the realm of security, where we need to know whether it is feasible to detect malicious modification of packets by a rogue node in the network. Future work must assess data safety and integrity in more detail than it has been here. Another question in the realm of security is how much of a burden encryption imposes on network communication. Current consumer routers come with built-in hardware encryption for common standards, reducing the burden of such security on other components. No such hardware encryption is yet availably for hardware the ClusterDuck Protocol is designed to run on, and there is a practical concern regarding export restrictions placed on cryptographic hardware by some governments, including the United States. Discovering whether enhanced security could be burdensome on such low-power devices is necessary for eventual adoption.

One realm not delved into in this research was resilience. While the mesh network design clearly works insofar that packets are received in a reasonable time, nodes becoming unusable or unavailable in a disaster is a distinct possibility that must be tested for. An experimental design could take the form of having each node in a network disable itself for a certain period of time stochastically to simulate an event were a node in the network is damaged or otherwise unusable after deployment. The random variable that determines this behavior will have to be chosen carefully if multiple experimental phases at different failure rates are infeasible.

This work has set out to show the strengths, weaknesses and opportunities for improvement in using LoRa mesh networking in the realm of disaster response. While the current state of the software that drives the ClusterDuck Protocol is inadequate, the areas for improvement are known and can be solved. The experimental design, while imperfect, has nonetheless shown promise in the area of battery consumption when transmitting information over long distances. The ClusterDuck Protocol is young, but promising entrant in a space known as "Internet of Things". Exciting as the space may be, a closer, concerted effort toward resilience and security should be top of mind going forward.

7.2 Acknowledgments

The authors would like to thank Project Owl and the ClusterDuck Protocol project for their assistance with hardware, software expertise and time spent debugging. Special gratitude is extended to Dr. Chris Pearson and Greg Keller of the College of Information for funding and experiential support of this research. We also thank Information Technology Services (ITS) and Dept. of Public Safety (DPS) of the university for infrastructural and material support.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] D.-H. Kim, E.-K. Lee, and J. Kim, "Experiencing lora network establishment on a smart energy campus testbed," *Sustainability*, vol. 11, p. 1917, 03 2019. [Online]. Available: https://doi.org/10.3390/su11071917

[2] FAO, *The impact of disasters and crises on agriculture and food security: 2021*, 2021. [Online]. Available: https://doi.org/10.4060/cb3673en

[3] T. Frank, "Cell phone service must be restored quicker after hurricanes," *Scientific American*, Oct 2019. [Online]. Available: https://www.scientificamerican.com/article/cell-phone-service-must-be-restored-quicker-after-hurricanes/

[4] K. Andersson and V. Kafle, "Disaster-resilient mobile network architecture," 01 2014, pp. 157–162.

[5] E. Blumenthal. (2020, Nov) Staying connected when the world falls apart: How carriers keep phones working. [Online]. Available: https://www.cnet.com/features/staying-connected-when-the-world-falls-apart-how-carriers-keep-phones-working/

[6] T. Nakachi, S. Kim, and T. Fujii, "A disaster-resilient layered data transmission based on movable and deployable resource units," 09 2013, pp. 517–522. [Online]. Available: https://doi.org/10.1109/ISCIT.2013.6645913

[7] D. Dailey and K. Starbird, "Social media seamsters: Stitching platforms & audiences into local crisis infrastructure," in *Proceedings of the 2017 ACM*

Conference on Computer Supported Cooperative Work and Social Computing, ser. CSCW '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 1277–1289. [Online]. Available: https://doi.org/10.1145/2998181.2998290

[8] I. Shklovski, L. Palen, and J. Sutton, "Finding community through information and communication technology in disaster response," in *Proceedings of the 2008 ACM Conference on Computer Supported Cooperative Work*, ser. CSCW '08. New York, NY, USA: Association for Computing Machinery, 2008, p. 127–136. [Online]. Available: https://doi.org/10.1145/1460563.1460584

[9] J.-S. Huang and Y.-N. Lien, "Challenges of emergency communication network for disaster response," in *2012 IEEE International Conference on Communication Systems (ICCS)*, 2012, pp. 528–532.

[10] Z. Lu, G. Cao, and T. L. Porta, "TeamPhone: Networking SmartPhones for Disaster Recovery," ser. IEEE Transactions on Mobile Computing, 2016.

[11] I. Santos-Gonzalez, P. Caballero-Gil, A. Rivero-Garcia, and C. Hernandez-Goya, "Distributed Communication System for Emergencies," *2016 24th Euromicro International Conference on Parallel, Distributed, and Network-Based Processing (PDP)*, pp. 309–312, 2 2016.

[12] Z. Shao, Y. Liu, Y. Wu, and L. Shen, "A rapid and reliable disaster emergency mobile communication system via aerial ad hoc bs networks," in *2011 7th International Conference on Wireless Communications, Networking and Mobile Computing*, 2011, pp. 1–4.

[13] Y. Zuo, Q. Luo, C. Cao, and J. Ge, "A design for emergency communication based on wireless laser communication," in *2019 International Conference on Robots Intelligent System (ICRIS)*, 2019, pp. 122–124.

[14] R. Sato, O. Oyakhire, and K. Gyoda, "Performance evaluation of Disaster Information Communication System using Message Ferry," *2019 34th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC)*, vol. 00, pp. 1–4, 2019.

[15] R. K. Megalingam, R. Vineeth, D. Krishnan, D. C. Jacob, and K. S. Akhil, "Wireless intelligent gadget for disaster relief," *Proceedings of the 1st International Conference on Wireless Technologies for Humanitarian Relief - ACWR '11*, pp. 515–518, 2011.

[16] R. Staff, "U.s. agency approves atamp;t drone in puerto rico for cellular service," Nov 2017. [Online]. Available: https://www.reuters.com/article/storm-maria-telecoms/u-s-agency-approves-att-drone-in-puerto-rico-for-cellular-service-idUSL1N1NN1HM

[17] J. P. Tuohy, "Matter: What you need to know," Oct 2022. [Online]. Available: https://www.theverge.com/23390726/matter-smart-home-faq-questions-answers

[18] R. Wang, F. He, and B. Gan, "Anti-Collision Warning Algorithm of Vehicle Intersection Based on V2X," *Proceedings of the 7th International Conference on Cyber Security and Information Engineering*, pp. 24–29, 2022. [Online]. Available: https://doi.org/10.1145/3558819.3558824

[19] K. Skoufas, E. D. Spyrou, and D. Mitrakos, "Low cost v2x traffic lights and vehicles communication solution for dynamic routing," in *Proceedings of the Eighth International Conference on Telecommunications and Remote Sensing*, ser. ICTRS '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 56–65. [Online]. Available: https://doi.org/10.1145/3357767.3357775

[20] T. Yoshizawa, D. Singelée, J. T. Muehlberg, S. Delbruel, A. Taherkordi,

D. Hughes, and B. Preneel, “A survey of security and privacy issues in v2x communication systems,” *ACM Comput. Surv.*, vol. 55, no. 9, jan 2023. [Online]. Available: https://doi-org.libproxy.umflint.edu/10.1145/3558052

[21] S. Habib, Z. Afnan, S. A. Chowdhury, S. A. Chowdhury, and A. S. M. Mohsin, “Design and development of iot based accident detection and emergency response system,” in *Proceedings of the 2020 5th International Conference on Cloud Computing and Internet of Things*, ser. CCIOT ’20. New York, NY, USA: Association for Computing Machinery, 2020, p. 35–42. [Online]. Available: https://doi.org/10.1145/3429523.3429526

[22] Z. Li, X. Liu, and S. Wu, “Dynamic emergency navigation based on prediction via wireless sensor networks,” in *2020 The 8th International Conference on Information Technology: IoT and Smart City*, ser. ICIT 2020. New York, NY, USA: Association for Computing Machinery, 2020, p. 210–215. [Online]. Available: https://doi.org/10.1145/3446999.3447634

[23] N. Hashmi, D. Myung, M. Gaynor, and S. Moulton, “A sensor-based, web service-enabled, emergency medical response system,” in *Proceedings of the 2005 Workshop on End-to-End, Sense-and-Respond Systems, Applications and Services*, ser. EESR ’05. USA: USENIX Association, 2005, p. 25–29.

[24] A. Filippoupolitis, L. Hey, G. Loukas, E. Gelenbe, and S. Timotheou, “Emergency response simulation using wireless sensor networks,” in *Proceedings of the 1st International Conference on Ambient Media and Systems*, ser. Ambi-Sys ’08. Brussels, BEL: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008.

[25] R. Beuran, S. Yasuda, T. Inoue, S. Miwa, and Y. Shinoda, “Using emulation to validate post-disaster network recovery solutions,” in *Proceedings of the*

7th International ICST Conference on Simulation Tools and Techniques, ser. SIMUTools '14. Brussels, BEL: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2014, p. 92–97. [Online]. Available: https://doi.org/10.4108/icst.simutools.2014.254619

[26] S. Kumar, R. K. Rathy, and D. Pandey, "Design of an ad-hoc network model for disaster recovery scenario using various routing protocols," in *Proceedings of the International Conference on Advances in Computing, Communication and Control*, ser. ICAC3 '09. New York, NY, USA: Association for Computing Machinery, 2009, p. 100–105. [Online]. Available: https://doi.org/10.1145/1523103.1523125

[27] E. Chissungo, H. Le, and E. Blake, "An electronic health application for disaster recovery," in *Proceedings of the 2010 Symposium on Information and Communication Technology*, ser. SoICT '10. New York, NY, USA: Association for Computing Machinery, 2010, p. 134–138. [Online]. Available: https://doi.org/10.1145/1852611.1852637

[28] H.-C. Jang, Y.-N. Lien, and T.-C. Tsai, "Rescue information system for earthquake disasters based on manet emergency communication platform," in *Proceedings of the 2009 International Conference on Wireless Communications and Mobile Computing: Connecting the World Wirelessly*, ser. IWCMC '09. New York, NY, USA: Association for Computing Machinery, 2009, p. 623–627. [Online]. Available: https://doi.org/10.1145/1582379.1582514

[29] V. D. Pham, V. Kisel, R. Kirichek, A. Koucheryavy, and A. Shestakov, "Evaluation of A Mesh Network based on LoRa Technology," *2021 23rd International Conference on Advanced Communication Technology (ICACT)*, pp. 1–6, 2 2021.

[30] L. Sciullo, A. Trotta, and M. D. Felice, "Design and performance evaluation

of a LoRa-based mobile emergency management system (LOCATE)," *Ad Hoc Networks*, 2020.

[31] K. C. V. G. Macaraeg, C. A. G. Hilario, and C. D. C. Ambatali, "LoRa-based Mesh Network for Off-grid Emergency Communications," *2020 IEEE Global Humanitarian Technology Conference (GHTC)*, pp. 1–4, 2020, summary in article in Notion.

[32] A. Zomaya, G. Min, A. F. Loureiro, M. C. Bor, U. Roedig, T. Voigt, and J. M. Alonso, "Do LoRa Low-Power Wide-Area Networks Scale?" ser. Proceedings of the 19th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems. acm, 2016. [Online]. Available: https://doi.org/10.1145/2988287.2989163

[33] J. M. Núñez V, F. Fonthal R, and Y. M. Quezada L, "Design and implementation of wsn and iot for precision agriculture in tomato crops," in *2018 IEEE ANDESCON*, 2018, pp. 1–5.

[34] N. Fahmi, S. Huda, E. Prayitno, M. U. H. Al Rasyid, M. C. Roziqin, and M. U. Pamenang, "A prototype of monitoring precision agriculture system based on wsn," in *2017 International Seminar on Intelligent Technology and Its Applications (ISITIA)*, 2017, pp. 323–328.

[35] L. Al-Tarawneh, A. Mehyar, S. E. Alasasaf, and M. Al-Mariat, "Environmental tracking system using iot based wsn: Smart agriculture," in *2022 4th IEEE Middle East and North Africa COMMunications Conference (MENACOMM)*, 2022, pp. 147–152.

[36] Santoshkumar and U. R.Y, "Development of wsn system for precision agriculture," in *2015 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)*, 2015, pp. 1–5.

[37] S. Bhagyashree, S. Prashanthi, and K. M. Anandkumar, "Enhancing network lifetime in precision agriculture using apteen protocol," in *2015 IEEE Technological Innovation in ICT for Agriculture and Rural Development (TIAR)*, 2015, pp. 44–48.

[38] G. R. Mendez, M. A. Md Yunus, and S. C. Mukhopadhyay, "A wifi based smart wireless sensor network for an agricultural environment," in *2011 Fifth International Conference on Sensing Technology*, 2011, pp. 405–410.

[39] O. N. Samijayani, R. Darwis, S. Rahmatia, A. Mujadin, and D. Astharini, "Hybrid zigbee and wifi wireless sensor networks for hydroponic monitoring," in *2020 International Conference on Electrical, Communication, and Computer Engineering (ICECCE)*, 2020, pp. 1–4.

[40] M. Saxena and S. Dutta, "Improved the efficiency of iot in agriculture by introduction optimum energy harvesting in wsn," in *2020 International Conference on Innovative Trends in Information Technology (ICITIIT)*, 2020, pp. 1–5.

[41] H. Patel and V. Shah, "A review on energy consumption and conservation techniques for sensor node in WSN," ser. 2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES), 12 2015, pp. 594–599.

[42] A. Ghosh, A. Maeder, M. Baker, and D. Chandramouli, "5g evolution: A view on 5g cellular technology beyond 3gpp release 15," *IEEE Access*, vol. 7, pp. 127 639–127 651, 2019.

[43] W. Xie, N.-T. Mao, and K. Rundberget, "Cost comparisons of backhaul transport technologies for 5g fixed wireless access," in *2018 IEEE 5G World Forum (5GWF)*, 2018, pp. 159–163.

[44] L. Dash and M. Khuntia, "Energy efficient techniques for 5g mobile networks in wsn: A survey," in *2020 International Conference on Computer Science, Engineering and Applications (ICCSEA)*, 2020, pp. 1–5.

[45] X. Zhang, J. Fei, H. Jiang, and X. Huang, "Research on power 5g business security architecture and protection technologies," in *2021 6th International Conference on Power and Renewable Energy (ICPRE)*, 2021, pp. 913–917.

[46] I. Rubin and R. Zhang, "Placement of UAVs as Communication Relays Aiding Mobile Ad Hoc Wireless Networks," pp. 1–7, 2007. [Online]. Available: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=4455114

[47] N. Matsumura, K. Nishimori, R. Taniguchi, T. Mitsui, and T. Hiraguri, "Performance improvement of drone mimo relay station using selection of drone placement," in *2018 IEEE International Workshop on Electromagnetics:Applications and Student Innovation Competition (iWEM)*, Aug 2018, pp. 1–1. [Online]. Available: https://doi.org/10.1109/iWEM.2018.8536637

[48] S. Ganesh, V. Gopalasamy, and N. B. Sai Shibu, "Architecture for drone assisted emergency ad-hoc network for disaster rescue operations," in *2021 International Conference on COMmunication Systems & NETworkS (COMSNETS)*, 2021, pp. 44–49. [Online]. Available: https://doi.org/10.1109/COMSNETS51098.2021.9352814

[49] E. Yanmaz, S. Hayat, J. Scherer, and C. Bettstetter, "Experimental performance analysis of two-hop aerial 802.11 networks," in *2014 IEEE Wireless Communications and Networking Conference (WCNC)*, April 2014, pp. 3118–3123.

[50] A. Kurt, N. Saputro, K. Akkaya, and A. S. Uluagac, "Distributed connectivity maintenance in swarm of drones during post-disaster transportation applica-

tions," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 9, pp. 6061–6073, Sep. 2021.

[51] L. Ferranti, S. D'Oro, L. Bonati, E. Demirors, F. Cuomo, and T. Melodia, "Hiro-net: Self-organized robotic mesh networking for internet sharing in disaster scenarios," in *2019 IEEE 20th International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM)*, June 2019, pp. 1–9. [Online]. Available: https://doi.org/10.1109/WoWMoM.2019.8793029

[52] D. G. Reina, M. Askalani, S. L. Toral, F. Barrero, E. Asimakopoulou, and N. Bessis, "A survey on multihop ad hoc networks for disaster response scenarios," *Int. J. Distrib. Sen. Netw.*, vol. 2015, jan 2015. [Online]. Available: https://doi.org/10.1155/2015/647037

[53] J. C. Liando, A. Gamage, A. W. Tengourtius, and M. Li, "Known and Unknown Facts of LoRa: Experiences from a Large-scale Measurement Study," *ACM Transactions on Sensor Networks (TOSN)*, vol. 15, no. 2, p. 16, 2019.

[54] L. Casals, B. Mir, R. Vidal, and C. Gomez, "Modeling the Energy Performance of LoRaWAN," *Sensors*, vol. 17, no. 10, p. 2364, 2017. [Online]. Available: https://doi.org/10.3390/s17102364

[55] T. Bouguera, J.-F. Diouris, J.-J. Chaillout, and G. Andrieux, "Energy consumption modeling for communicating sensors using LoRa technology," *2018 IEEE Conference on Antenna Measurements & Applications (CAMA)*, pp. 1–4, 2018.

[56] N. Pukrongta and B. Kumkhet, "The relation of LoRaWAN efficiency with energy consumption of sensor node," ser. 2019 International Conference on Power, Energy and Innovations (ICPEI), vol. 00, 2019, pp. 90–93.

[57] G. Kazdaridis, S. Keranidis, P. Symeonidis, P. Tzimotoudis, I. Zographopoulos, P. Skrimponis, and T. Korakis, "Evaluation of lora performance in a

city-wide testbed: Experimentation insights and findings," in *Proceedings of the 13th International Workshop on Wireless Network Testbeds, Experimental Evaluation & Characterization*, ser. WiNTECH '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 29–36. [Online]. Available: https://doi.org/10.1145/3349623.3355474

[58] J. H. Adams, M. Southard, S. Forbes, B. Leinweber, and D. Nix, "Effective training improves disaster response," *Opflow*, vol. 48, pp. 10–14, 01 2022.

[59] C. Goursaud and J. M. Gorce, "Dedicated networks for iot: Phy / mac state of the art and challenges," *EAI Endorsed Transactions on Internet of Things*, vol. 1, no. 1, 10 2015.

[60] "What is lorawan?" Sept 2022. [Online]. Available: https://lora-alliance.org/about-lorawan

[61] "Lorawan security faq," Oct 2021. [Online]. Available: https://lora-alliance.org/resource_hub/lorawan-security-faq

[62] "Call-for-code/clusterduck-protocol: Firmware for an ad-hoc mesh network of internet-of-things devices based on lora (long range radio) that can be deployed quickly and at low cost." Dec 2021. [Online]. Available: https://github.com/Call-for-Code/ClusterDuck-Protocol

[63] T. L. Foundation, "The linux foundation open sources hardware of disaster relief project," Mar 2020. [Online]. Available: https://www.linuxfoundation.org/press-release/the-linux-foundation-open-sources-hardware-of-disaster-relief-project-that-won-first-call-for-c

[64] "Shenzhen xin yuan electronic technology co., ltd." [Online]. Available: http://www.lilygo.cn/

[65] Info, "Wifi lora 32 (v2)." [Online]. Available: https://heltec.org/project/wifi-lora-32/

[66] "America's families and living arrangements: 2022," Nov 2022. [Online]. Available: https://www.census.gov/data/tables/2022/demo/families/cps-2022.html