IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 5, 2022, accepted May 20, 2022, date of publication May 27, 2022, date of current version June 3, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3178437

A Novel LoRa LPWAN-Based Communication Architecture for Search & Rescue Missions

MELVIN P. MANUEL, (Student Member, IEEE), MARIAM FAIED, (Senior Member, IEEE), AND MOHAN KRISHNAN, (Life Senior Member, IEEE)
Department of Electrical & Computer Engineering and Computer Science, University of Detroit Mercy, Detroit, MI 48221, USA
Corresponding author: Melvin P. Manuel (manuelmp@udmercy.edu)

ABSTRACT To avoid risking the lives of rescue team personnel in the event of disasters like earthquakes, volcanic eruptions, hurricanes, etc., Search and Rescue (SAR) robots are increasingly incorporated into the operation. One of the major challenges in integrating SAR robots into rescue operations is the potentially severely damaged infrastructure within the disaster site. A functional communication system is critical for exchanging real-time information between the robots and the base station. Given the limited coverage or absence of communication systems in a severely affected disaster site, a novel communication architecture for search & rescue missions based on Long Range (LoRa) Low Power Wide Area Network (LPWAN) and a SAR robot called *Rescuer* are proposed. *Rescuer* is a SAR robot that can operate in worst-case disaster sites where all communication infrastructure has been wiped out. It has been tested in a Gazebo simulated environment as well as an actual test setup inside the University of Detroit Mercy's lab facility and showed great promise. In this test, the *Rescuer* robot was monitored and controlled from a remote base station.

INDEX TERMS Disaster management, LPWAN, LoRa, Pioneer-P3DX, rescue, ROS, robotics, SARs, UGVs.

I. INTRODUCTION

The planet faces a growing frequency and severity of natural and man-made disasters with dangerous impacts. As reported by Centre for Research on the Epidemiology of Disasters (CRED), over half a million people have been killed, and more than 3.9 billion people have been affected by disasters over the last ten years [1]. Disasters create emergencies that necessitate the urgent provision of essential services to victims, which requires close cooperation. People may be trapped alive within damaged structures if the disaster involves structural collapse. Their survival is contingent on immediate assistance, and the longer they wait, the more likely they will die. Those that have been seriously injured must be given immediate medical attention and evacuated to a safe place. Rescue activities in severely damaged disaster situations are made even more difficult when the affected area is large. Mobile robots have been employed in disaster relief for several years as a solution to this problem, specifically to tackle duties that people are unable to perform [2]. Modern robot and sensor-based SAR technologies will only hasten the discovery of the disaster environment, including the identification of casualties and volatile, dangerous sources.

The associate editor coordinating the review of this manuscript and approving it for publication was Shaohua Wan.

A. RELATED WORK

Many countries have begun to use Unmanned Ground Vehicles (UGVs) and Unmanned Aerial Vehicles (UAVs) as a vital component of rescue operations in order to prevent endangering the lives of victims, rescue team workers, and volunteers. According to the Texas-based Center for Robot-Assisted Search and Rescue (CRASAR), more than 60 deployments of SAR robots have been documented around the world to date [3]. The following are some of the published articles in the field of robot-based SAR operations. Mitchell *et al.* [4] present a review of how SAR robots helped in search and rescue operations in the recent natural disasters that hit North America. Geert De Cubber *et al.* [5] proposed using two SAR robots for search and rescue operations in a highly affected disaster site. One of them was a large UGV equipped with a powerful manipulator arm used for debris removal, shoring operations, and remote structural operations on rough terrain. The SAR robots used in this study were created as part of the Integrated Components for Assisted Rescue, and Unmanned Search operations (ICARUS) project [6], [7]. The

ICARUS project was a European project that built integrated components to support SAR robotics teams coping with rescue missions in dangerous and life-threatening situations in order to save survivors' lives. Q. Ren *et al.* [8] demonstrated a rescue operation in a real disaster scenario, where they used UGVs to clear rocks and debris from the robot's path. H. Kuntze *et al.* [9] explained the SENEKA project's proposed integrated private robot sensor network that a SAR team could use to communicate with robots carrying different sensor systems in order to expedite the rescue mission.

One of the significant challenges in integrating robotics into a SAR project is the possibly badly damaged communication infrastructure, in addition to highways, bridges, and power supplies. For a SAR mission, these types of worst-case scenarios pose the most significant challenge. A functional communication system is critical for exchanging real-time information between robots and base stations for proper coordination of multiple operations, as would be needed at a disaster site. Current techniques for communicating with the UGVs in such a disaster scenario are based purely on 3G/4G, GSM, and WiFi networks, whose availability and coverage might be limited or absent in a severely affected disaster site. Keeping all these limitations of current methods in mind, a novel disaster management system which is based on a SAR robot called *Rescuer* and LoRa technology is proposed. *Rescuer* is a SAR robot that can operate in worst-case disaster sites where all communication infrastructure has been wiped out.

Some recent studies on implementing LoRa-based communication systems in different engineering applications are outlined below. Godoy *et al.* [10] proposed a LoRa-based communication system for UAVs to expand their coverage area. Experiments were developed to test the performance of the developed system, which required equipping two UAVs with the necessary gear and sending data between two distant locations. These experiments have confirmed that the suggested system fits the criteria of long-range communication coverage extending over a distance of 10 kilometers with transmitted power of 0 dBm. To change the existing farm management system, Mehran *et al.* [11] presented a farm monitoring system that includes UAVs, LPWAN, and IoT technologies. The proposed approach assists farmers in acquiring actionable data from farm activities. A multi-channel LoRaWAN® gateway was developed and incorporated into a vertical takeoff and landing drone based on LoRaWAN® technology to transmit data received from various sensors to the cloud for further analysis. The greatest feasible LoRa coverage when operating on-air through the drone was around 10 kilometers, according to measurement data. Hsieh *et al.* [12] proposed a vehicle monitoring system (VMS) based on LoRa technology. The VMS can monitor a variety of environmental characteristics, including ambient temperature and humidity, as well as air quality metrics such as PM2.5, NO_2, CO, and O_3. The detected data from the automobiles is relayed to the cloud server using the LoRaWAN® protocol. A user interface was used to display the sensor data stored on the cloud server. Chou *et al.* [13] suggested the i-car system, a LoRa-based LPWAN vehicle diagnostic system for driving safety. A remote diagnostic system, a LoRa gateway, and a cloud platform are all part of the planned i-car system. The OBD-II Bridge can read some car information to determine whether there is a problem. If there is an abnormality in the parameters, this information will be sent to the monitoring station through the LoRaWAN® gateways. T. Maneekittichote *et al.* [14] proposed a LoRaWAN-based bidirectional communication system that allows the robot to send and receive information/commands from a user. The experiment was aimed to test whether the mobile robots can successfully reach the target location by avoiding obstacles and by continuously communicating with the user. Alfin Junaedy *et al.* [15] proposes a LoRaWAN-based communication system along with WiFi for Real-Time 2D SLAM and object localization for teleoperating a robot. The proposed two-LoRa configuration enhances the teleoperation capabilities of the robots to use in SAR missions.

B. LPWAN & LoRa BACKGROUND

The Low Power Wide Area Network (LPWAN) is a type of wireless telecommunication wide area network designed to allow long-range communications using less bandwidth between connected devices. Long-range, low power, and low cost are the three factors that distinguish this type of network from other networks. There are several competing standards in the LPWAN domain, the most prominent of which are: DASH7, Sigfox, LoRa, etc. But LoRa has the lowest power consumption rate, as well as the longest communication range [16] [17]. LoRa is able to connect devices up to 30 miles apart in rural areas and can penetrate through dense urban, and deep indoor environments [18], [19] [20]. LoRa is based on spectrum modulation techniques derived from Chirp Spread Spectrum (CSS) technology [21], [22].

The CSS technique was developed in the 1940s for radar applications. Due to the long communication distances that can be achieved and the robustness to interference, CSS has been widely used in military and space communication applications for decades, but LoRa is the first low-cost implementation for commercial use [21]. The chirp signal results from intentionally spreading the information to be encoded over the allocated frequency domain through the use of a chirp carrier, whose frequency systematically increases and decreases with time. This CSS modulation system is also known for its robustness against interference, multi-path fading, and channel degradation phenomena. A number of data rates for various frequency ranges are available with this technique. The standard protocol named LoRaWAN® (LoRa Wide Area Network) specification defines the frequency bands and standards for LoRa communication [22]. These radio bands were originally set aside for electromagnetic radiation produced by Industrial, Scientific, and Medical (ISM) equipment. In the early 1990s, the US Federal Communications Commission (FCC) allowed the use of three of the

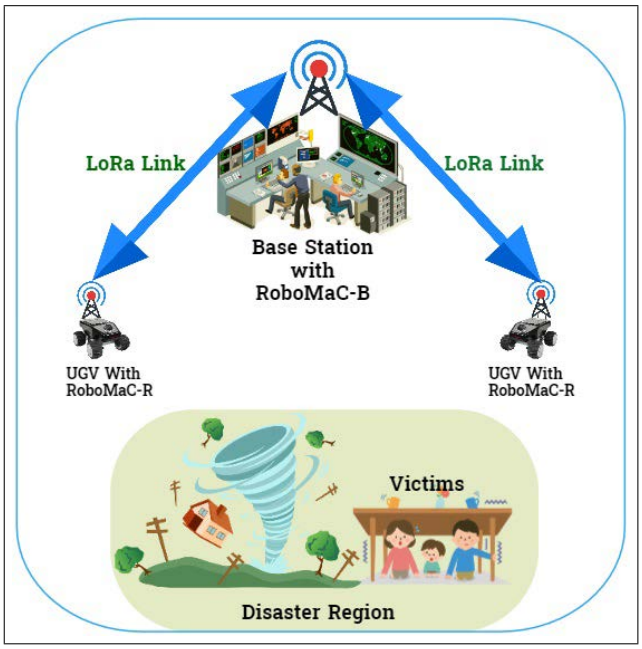


FIGURE 1. LoRa based robot control in disaster region.

ISM bands for unlicensed communication equipment. These are: 902 MHz to 928 MHz, 2.400 GHz to 2.4835 GHz and 5.725 GHz to 5.875 GHz [23]. LoRa operates in the United States, Canada, and South America in the 902 to 928MHz range [24].

Despite its long-range, low power, and low cost, one major disadvantage of a LoRa-based communication system is the lack of data security between the devices communicating with one another. This key feature necessary to use with LoRa is actually incorporated into LoRaWAN®, which is commonly used in IoT-based applications. The LoRa physical layer (Wireless) is used to communicate between LoRa devices/nodes and the gateway, and an Internet Protocol (IP)-based network LoRaWAN® is then used to link the gateways to the central network servers. All of the LoRaWAN® protocol's security features are highly dependent on the applications and how the LoRa-based connection is implemented [25]. The receiver/gateway must be wired to a standard network in order to gain access to the LoRaWAN® security features. However, as discussed above, network connectivity cannot be guaranteed due to the damaged infrastructure within a disaster site. Therefore, the standard LoRaWAN® protocol's security features might not be accessible. Manuel *et al.* [26] proposed a cryptography protocol that can be used with LoRa end devices to ensure confidentiality, integrity, and authenticity (CIA) of data. This cryptography protocol has been used in this work as a security feature for ensuring data security. To be clear, since the base station is required to manage the rescue robot activities locally at the disaster site and under the assumption of communication network unavailability in the worst-case disaster site, the LoRaWAN® part cannot be used in this study. Instead, the LoRaWAN® protocol's physical layer, the LoRa, is utilized to carry out this work in order to achieve device-to-device communication.

C. CONTRIBUTIONS

According to the literature study in Section 1-A, prior research used the LoRaWAN® protocol to send data to a monitoring station at an assigned time period. Outside of the allocated time, there was no continuous established communication link between the devices. On the other hand, this study presents a LoRa-based communication system that maintains a continuous communication link between the robot and the base station. Using the proposed method, disaster relief teams can track and control *Rescuer* robots from a remote or standoff position as they visit and provide help to victims while reporting their own locations to the base station. This can be done without relying on the availability of traditional communication networks. The complete system architecture is conceptualized in Figure 1. This paper introduces a robot named *Rescuer* for SAR missions. This *Rescuer* robot is suitable for deployment in disaster sites under worst-case communication blackout conditions using Long-Range LPWAN technology. Using the proposed novel LPWAN LoRa-based communication device (RoboMaC), the *Rescuer* robot can be monitored and controlled from a remote base station. RoboMaC is a full-duplex communication device that can serve as a transceiver unit to transmit control commands to the *Rescuer* robot while at the same time receiving the robot's location information at the base station. The functionality of *Rescuer* robot and RoboMaC devices are demonstrated through test results. Testing is initially carried out in a Gazebo simulated robot environment and then extended to actual deployment inside the University of Detroit Mercy's lab facility, using a Pioneer-P3DX robot. The philosophy behind all algorithms proposed in this study is applicable not just to Pioneer-P3DX, but also to other heavy-duty SAR robots outfitted with the same communications hardware. In addition, although the implementation was done on one robot, the communication architecture is extendable to multiple robots operating in SAR teams, using different ISM frequency bands.

The rest of the paper is organized as follows - Section II presents the hardware details of the *Rescuer* robot and the RoboMaC devices, Section III outlines the RoboMaC parameter selection and packet format, Section IV discusses the details of software implementation, Section V provides details of the experiments carried out and the results obtained, and Section VI concludes the paper.

II. RESCUER ROBOT AND ROBOMAC HARDWARE DETAILS

The *Rescuer* robot and RoboMaC devices were developed using a variety of hardware modules and sensors. The following is a broad discussion of its features. Pioneer-P3DX [27] is the mobile robot platform used for implementing this work. The P3DX is a small, lightweight, two-drive wheel plus caster, differential drive robot suitable for research

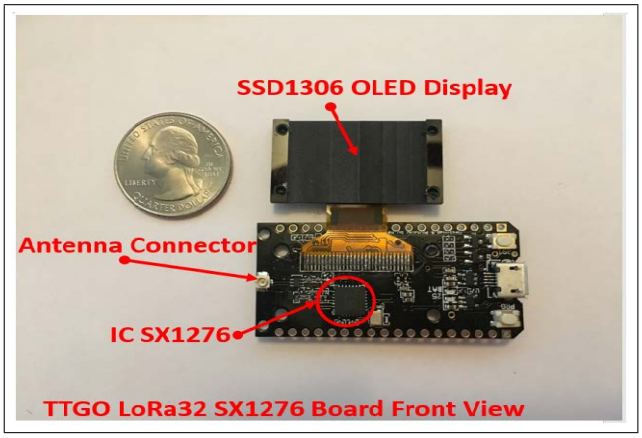


FIGURE 2. Development board used for RoboMaC.

applications. The robot was outfitted with a Microsoft Kinect camera [28], which was used to capture footage of the environment once the *Rescuer* arrived at the victims' location. The Kinect camera is an advanced visual sensor for high-performance 3D imaging facilitated by a depth sensor. The computer used on the *Rescuer* robot is an Intel Next Unit of Computing (NUC) computer [29] suitable for high-speed robotic applications. The LiDAR used for detecting obstacles along the *Rescuer* robot's path is a Hokuyo UST-10LX [30] [31].

The RoboMaC devices were developed using the TTGO LoRa ESP32 (LoRa32) development board. This board can operate safely in harsh industrial conditions and has an operating temperature range of $-40°C$ to $+125°C$, making it ideal for use in disaster environments. Furthermore, the LoRa ESP32 supports three communication protocols: WiFi, Bluetooth, and LoRa. The LoRa ESP32 board used for developing RoboMaC is shown in Figure 2. The brain of the development board is an ESP32-PICO-D4 module that is based on the ESP32 microcontroller, providing complete 2.4 GHz WiFi and Bluetooth functionalities [32]. The IC SX1276 is the LoRa chip used in the LoRa32 development board. The transceivers on the SX1276 feature the LoRa modem that provides ultra-long range spread spectrum communication and high interference immunity while minimizing current consumption [33]. With just the in-built antenna, the LoRa32 module's communication range is limited to 0.8 to 1.2 miles [34]. To increase the range further, an Omnidirectional antenna Model A904 from Data Alliance was used [35]. A wireless radio was used to establish audio communication with the victim(s) at the disaster site. For proof of concept, a low-cost walkie-talkie model T-388 from Funkprofi has been mounted on the *Rescuer* robot. To enable victim communication with the base station, the operator at the base station will remotely enable the radio's Push-To-Talk feature via the RoboMaC device.

Two pairs of TTGO LoRa ESP32 (LoRa32) modules were utilized to achieve full-duplex communication between the *Rescuer* robot and the base station. Each pair consists of

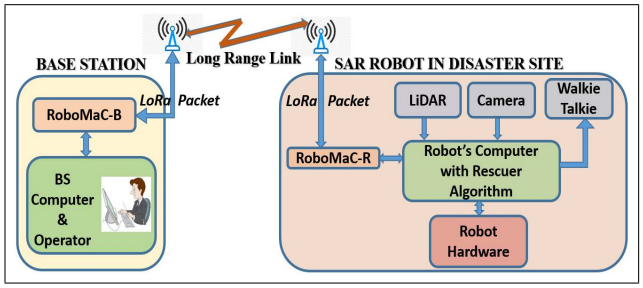


FIGURE 3. Block diagram of the proposed communication architecture.

a transmitter and a receiver. One pair named RoboMaC-B was deployed at the base station, and another pair named RoboMaC-R on the robot. Frequency allocations as well as algorithms for each module of the RoboMaC units are discussed in the upcoming sections. A complete block diagram of the proposed system is shown in Figure 3.

III. RoboMaC PARAMETERS SELECTION & DATA PACKETS FORMAT

RoboMaC-B, the device deployed at the base station, functions primarily as a transceiver unit (TxRx). It consists of two LoRa32 modules. The first one was programmed as a transmitter module (Tx) to send control commands to the *Rescuer* robot. This module uses 915.5MHz. The second module was tuned to 915.0MHz and served as a receiver for collecting robot position information. Another RoboMaC module named RoboMaC-R, was mounted on the *Rescuer* robot. RoboMaC-R transceiver device also contains two LoRa32 modules, one transmitting the robot's current position information to the base station at 915.0MHz and the other one receiving base station commands at 915.5MHz. Given that the USA LoRaWAN® frequency spectrum has 64 up-link channels available with a bandwidth of 125KHz each [36], [37], a bandwidth of 250KHz has been allotted for each transceiver unit (TxRx) to avoid interchannel interference. The frequencies are suitably chosen within the ISM frequency band. The frequency allocation process within the ISM frequency band can be extended to accommodate multiple robots in the rescue mission. Some of the parameters for optimizing performance and fine-tuning the LoRa communication system are discussed below.

A. SPREADING FACTOR (SF)

LoRa converts binary data to chirp signals that span the frequency range. The chirp time is roughly proportional to double the spreading factor. So, each step increment in *SF* doubles the time on-air to transmit binary data. The larger the spreading factor, the greater the range but slower the data rate. The spreading factor for both Tx and Rx radios needs to be the same for reduced packet loss [38] [39] [40]. The spreading factor *SF* is represented as in [41]:

$$SF = log_2 \left(\frac{R_c}{R_s}\right) \tag{1}$$

where R_c and R_s are the chirp rate and the symbol rate, respectively. SF reflects the number of chirps per symbol and ranges between 7 and 12.

B. SIGNAL BANDWIDTH (BW)

The data rate depends on the bandwidth and spreading factor. Higher bandwidth corresponds to a higher data rate and is more power-efficient, but it has more congestion and less range. LoRa can use channels with a bandwidth of either 125KHz, 250KHz, or 500KHz, depending on the region or the frequency plan. Since these parameters influence the modulation and demodulation of LoRa packets, these parameters must be collectively set for the transmitter and receiver for effective communication [41].

C. DATA RATE (DR)

The following is how the DR was determined:

$$DR = SF * \frac{BW}{(2^{SF})} * CR \tag{2}$$

The number of chirps per symbol is calculated by SF, which is inversely proportional to the modulation rate of a chirp. The ratio of non-redundant data to all data within the send and receive frames is between 4/5 and 4/8 and the selected chirp rate (CR) for the RoboMaC devices is 4/5. The chirp rate is calculated from:

$$CR = \frac{BW^2}{2^{SF}} \tag{3}$$

If the BW of LoRa is constant, the chirp rate differs according to the SF. The orthogonality of the chirp for each SF avoids interference from other devices, when two or more transmitters use the same channel to concurrently transmit with different SF [42].

A time analysis has been conducted for the optimal selection of these parameters. At 600 meters distance between LoRa transmitter and receiver, we averaged the time delay between transmission and reception of 24 LoRa data packets. Our time analysis chart shown in Figure 4 is a monothetic analysis in which we change only one parameter at a time. As we can see from Figure 4, for the CR value of 4/5 and 10 SF, the delay between sending and receiving the data is 1.86 seconds. Thus, we choose these values in our experiments. For BW, we choose the middle-frequency 250KHz to balance the tradeoff between bandwidth and data rate.

D. RoboMaC DATA PACKET FORMAT

Figure 5 shows the payload format of a LoRa packet framed and sent by the base station from RoboMaC-B to RoboMaC-R. The X and Y coordinates are the *Rescuer* robot's destination. We proposed 9 bits for the X and Y coordinates, enabling the system to accept destination location in GPS coordinates format. The minimum digits for a GPS coordinate are 9 bits for the latitude and longitude. The C bit is a single bit used for enabling the camera to record video once the *Rescuer* robot reaches the victims' location. The W bit is to enable/disable the walkie-talkie attached to the robot. The last bit is the $E - Stop$ bit, which allows the operator at the base station to stop the robot in an emergency.

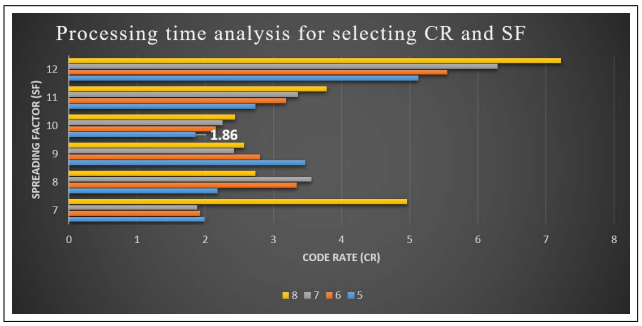


FIGURE 4. Time analysis chart.

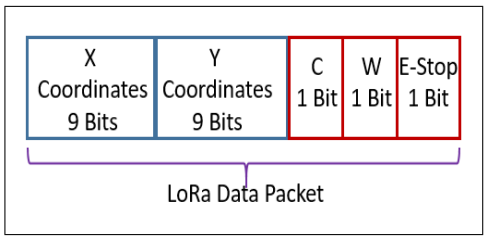


FIGURE 5. LoRa payload packet structure sending from base station.

IV. SOFTWARE IMPLEMENTATION

Through Transmitter, Receiver, and Rescuer algorithms, the *Rescuer* robot is able to receive information like victims' location, an emergency stop signal, a camera enables bit and wireless radio activation command from the base station, and send back its current location to the base station while navigating to a victim. The set of algorithms discussed below enables the coordination of *Rescuer* from the base station. All data communicated between the base station and the robot are secured using our earlier cryptography protocol discussed in [26].

After ensuring that the data was not hacked, the victim's Location Information (LI) will be extracted from the received LoRa packet. The destination coordinates will then be given as goals for the navigation algorithm. Our predeveloped navigation algorithm named FISVFH [43] was used to drive the robot to the destination while avoiding obstacles along its way. The FISVFH is a fuzzy-rule-based navigation algorithm that can navigate a robot to the assigned destination through an unknown environment, incorporating the discovery of emerging sections of the environment, as the robot moves, into the navigation strategy. FISVFH does not need a prior map for navigation, which makes it suitable for navigation in a disaster environment. As the robot moves to its destination, it uses the ROS Gmapping package to create a map of the environment through which it is traveling. The algorithms enabling the functionality described are distributed between the RoboMaC modules and the robot's computer. These algorithms are discussed further below.

A. TRANSMITTER ALGORITHM

The RoboMaC Transmitter algorithm is explained in Algorithm 1. It starts by initializing various parameters in lines 1-4. Then, the data packet parameters (X, Y, C, W, and E-stop) bits are loaded and framed as a LoRa packet. After that, the LoRa packet is encrypted. Once the packet is ready for transmission, the algorithm will determine whether the specific channel to be used is free or not. If the channel is free, the encrypted data packet will be sent. If the channel is not free, then it will wait for the channel to become free.

Algorithm 1: RoboMaC Transmitter

```
1  Set the LoRa frequency 915/915.5MHz ;
2  Set the spreading factor to 10 ;
3  Set the signal bandwidth 250KHz ;
4  Set the Tx Power to 15 ;
5  Frame the LoRa Message [X, Y, C, W, E-Stop];
6  [X= xx.xxxxxx, Y= yy.yyyyyy, C= 0/1, W= 0/1,
   E-Stop= 0/1] ;
7  Apply Cryptography protocol on the data frame ;
8  if LoRa channel is free, then
9  |   Send the LoRa Packet
10 else
11 |   Wait for channel to be free ;
12 |   Go to step 8;
13 End Transmission;
```

Algorithm 2: RoboMaC Receiver

```
1  Set the LoRa frequency 915/915.5MHz ;
2  Set the spreading factor to 10 ;
3  Set the signal bandwidth 250KHz ;
4  Set the Tx Power to 15;
5  while LoRa packet available, do
6  |   Read the incoming packet;
7  |   Decrypt and check the received packet is hacked or
   |     not;
8  |   if Data Hacked then
9  |   |   Display that the data is hacked ;
10 |   |   Go to step 6 ;
11 |   else
12 |   |   Send the data to serial port;
13 |   End Reception ;
14 Go to step 5;
```

B. RECEIVER ALGORITHM

Algorithm 2 explains the code running on the RoboMaC receiver modules. The algorithm continuously monitors the LoRa channel for any incoming LoRa packet. Once it receives a packet, it will read it and then decrypt it. If the data is found to be hacked, a message will be displayed on the OLED saying that the data was hacked [26]. If the integrity of the incoming data is validated, it will send it to the serial port of the LoRa32 module. If the receiver is a part of the RoboMaC-R module, the navigation algorithm reads the serial port and drives the robot accordingly. On the other hand, if the receiver module is in the RoboMaC-B module, which is at the base station, it will continuously monitor and display the robot's location, using the received location information.

C. RESCUER ALGORITHM

Algorithm 3 details the steps followed to start, initialize and drive the *Rescuer* robot. All the required ROS nodes to bring up the robot and sensors must be launched first. Then, the FISVFH algorithm has to be launched. The FISVFH algorithm monitors the serial port every 5 seconds for any incoming LoRa packets. Once a data packet arrives, the algorithm first decrypts it. Additional actions will be initiated if the incoming data passes the security check. Otherwise, a message will be displayed on the OLED, indicating that the incoming data was hacked or otherwise corrupted. In this case, the navigation algorithm will not be launched, and the robot will not move. If the data integrity of the incoming data is confirmed, further processing steps are carried out by the FISVFH navigation algorithm. The status of the E-Stop bit is checked, and if it is a '1', the algorithm will wait until the E-Stop bit is disabled. The victim's location information (LI) will be assigned as the input to the navigation algorithm. Once the robot starts moving, its current location will be continuously recorded every 5 seconds and sent back to the base station. This real-time location information enables the base station to be aware of the *Rescuer* robot's position. Once the robot has reached the victim's location, the algorithm starts monitoring the camera bit C and wireless radio bit W to activate these devices, as needed.

V. EXPERIMENTATION AND RESULTS

Testing of the system was done in two phases. The first phase was with a simulated disaster environment in Gazebo, but with ESP32 hardware modules for controlling and monitoring the simulated robot. The second experimentation phase was performed in an actual environment constructed inside the University's indoor testing facility. For the first phase of testing RoboMaC and all algorithms, a simulated disaster environment with scattered obstacles was developed in the Gazebo simulator. The main intention of this test was to control the simulated robot from a remote location. The simulated disaster site is shown in Figure 6.

The second phase of the work was to test the system's functionality on a real robot. A Pioneer P3DX robot was used for this purpose. A test environment was created inside our indoor lab facility, in which the communication between the base station and the robot was tested. As in Phase 1, RoboMaC modules were positioned at the base station and on the robot, enabling the former to send commands to and receive information from the robot. The *Rescuer* robot could be successfully controlled over a range of approximately

Algorithm 3: Rescuer Algorithm Running on Robot

```
Power ON the LoRa32 modules and wait for system initialization;
Launch all relevant ROS nodes :-
2(a)Launch the robot's ROS nodes (Simulation/Real time). This will publishes and subscribes different topics to drive the robot;
2(b)Kinect camera node will be launched and wait for incoming commands to start recording;
2(c) Gmapping will be launched to start constructing map ;
Execute the Matlab code for FISVFH :-
3(a) Open the serial ports where the RoboMaC-R is connected;
3(b) Read and save the incoming data from the LoRa receiver every 5 seconds; ;
while LoRa packet available do
    Decrypt the received packet ;
    if Data Hacked then
        Display that the data is hacked ;
        Go to step 5 ;
    if E-Stop bit is 0, then
        Go to step 13 ;
    else
        Go to step 5;
    Read the Location Information(LI) ;
    if LI available, then
        Assign LI to FISVFH and start navigation;
        if LI ≠ Odometer data, then
            Continue navigation using FISVFH;
            Encrypt the odometer data;
            Send the Encrypted data to the base station;
        else
            Stop navigation and go to step 24
    else
        Wait for LI from LoRa ;
    Read the Camera enable C Bit ;
    if Camera enable bit is 1 then
        Start Recording video;
    else
        Do not record ;
    Read the Wireless Radio W enable bit ;
    if W bit is 1, then
        Send HIGH to GPIO-25 of LoRa32 Rx module.;
    else
        Send LOW to GPIO-25 of LoRa32 Rx module ;
Go to step 4;
```

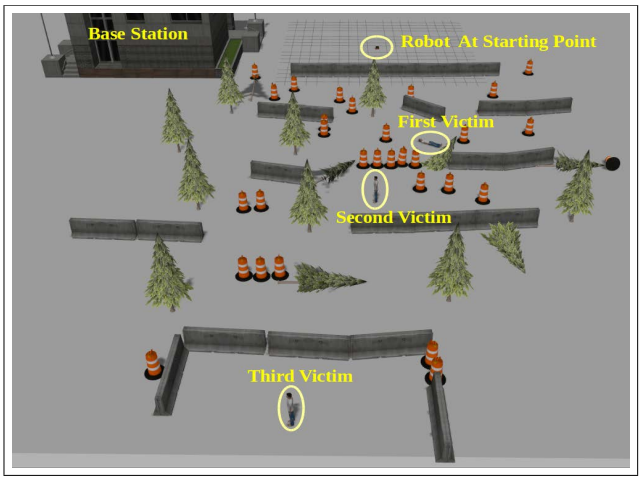


FIGURE 6. Simulated disaster site in gazebo.

1.6 miles, which spanned the University campus and the surrounding urban areas of the City of Detroit. The details of the simulation and real-time testing of RoboMaC modules and the *Rescuer* robot follow.

A. GAZEBO SIMULATION ENVIRONMENT

A disaster world was developed in the Gazebo simulator with many scattered obstacles, positioned to require the robot to adopt zigzag paths to reach the victims, as shown in Figure 7(a). The simulated environment was about 70 meters long by 55 meters wide. It was assumed that there were three victims needing help at different locations. A simulated Pioneer-P3DX robot model was used to test the system. The physical base station unit, the RoboMaC-B, was inside a car parked about 1.6 miles away from the computer on which the Gazebo world was launched. From the RoboMaC-B LoRa device, the control command was sent to the simulated robot. A RoboMaC-R module, which was connected to the computer's USB port in which the simulated environment was launched, received the destination information from RoboMaC-B. Once received, the simulated robot started navigating to the destination, as explained in Algorithm 3.

At the same time, the RoboMaC-B at the base station was receiving the odometer data sent from the RoboMaC-R module. Initially, the robot was assigned the first victim's destination (in local coordinates), which was X=22, Y= 4. The FISVFH navigation algorithm used this information to navigate the robot through the devastated environment to the first victim's location. As the robot moved, the Robot Operating System (ROS) Gmapping package started creating a map of the environment traversed by the robot. This map enabled the base station to be aware of the nature and condition of the disaster environment, information that could

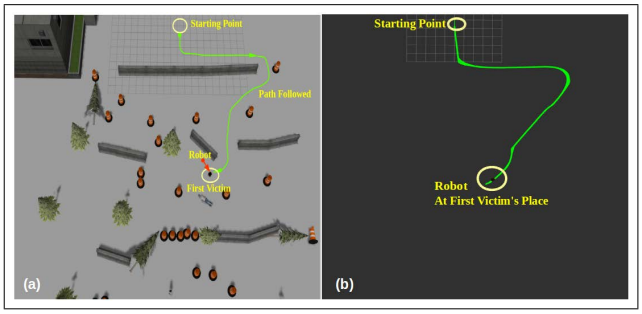


FIGURE 7. (a) Gazebo world and first victim's location. (b) Robot's path plotted in Rviz.

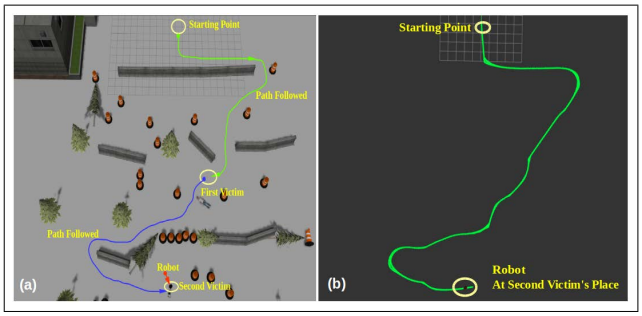


FIGURE 8. (a) Gazebo world and second victim's location. (b) Robot's path plotted in Rviz.

be useful if necessary for subsequent active involvement of human rescuers in rescue and recovery.

The navigation path followed by the robot from the starting point to the first victim's location is plotted in Rviz and is shown in Figure 7(b). Rviz is a 3D visualizer [44] for displaying sensor data and state information obtained from the Gazebo simulator [45]. It can show the robot's position and plot valuable data, such as the path adopted by the robot, based on sensor information. As seen from Figure 7, the robot successfully adopted a smooth path to the first victim's location while avoiding obstacles along the way.

When the robot reached the first victim's site and provided needed services, the base station sent the second victim's location information (X= 30, Y = −1). The robot then started to move towards the new destination. Figure 8(a) shows the path adopted to reach the second victim. Figure 8(b) shows the Rviz plot of the robot's path. Once the RoboMaC-B received confirmation that the robot had reached the second victim's location, the operator could remotely activate the enable/disable switch for the camera and the wireless radio. Once the robot completes management of the second victim, the base station operator assigns the third victim's location (X = 48 and Y = −6). The robot then proceeded to navigate to where the third victim was, as shown in 9(a). The Rviz plot of the associated path followed is shown in Figure 9(b).

After reaching the third victim's location, the base station operator directed the robot to return to the starting location

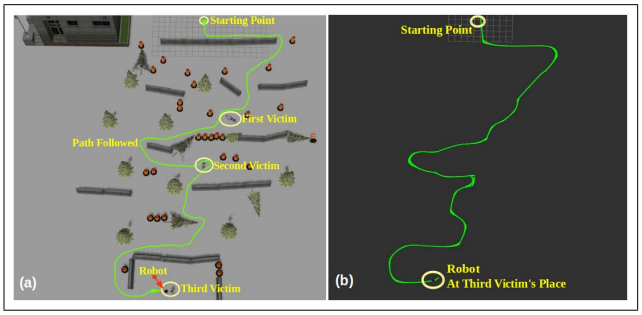


FIGURE 9. (a) Gazebo world and third victim's location.(b) Robot's path plotted in Rviz.

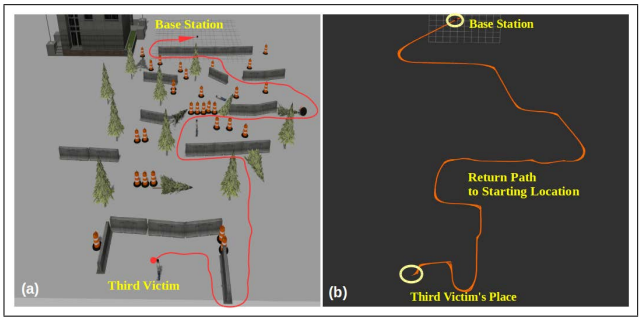


FIGURE 10. (a) Gazebo world and robot's return path. (b) Robot's return path plotted in Rviz.

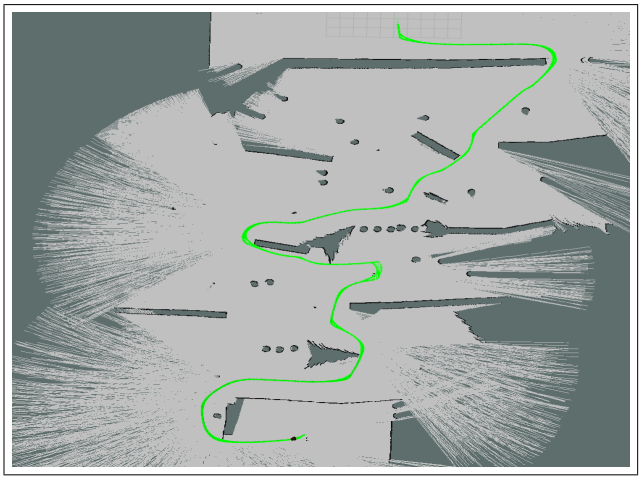


FIGURE 11. Map of the environment.

(X=0, Y=0). Figure 10(a) shows the return path of the robot, and Figure 10(b) shows the return path as plotted in Rviz. Figure 11 shows the overall map of the disaster site plotted with Rviz. When the robot arrived at the base station, the map it had created during its journey was available to the rescue team to develop additional strategies.

B. TESTING WITH REAL ROBOTS

A mock disaster environment was put together in our lab facility to test the system on a real robot. The size of the disaster environment was 60 meters long by 15 meters wide and is shown in Figure 12. Three destinations were assigned to the

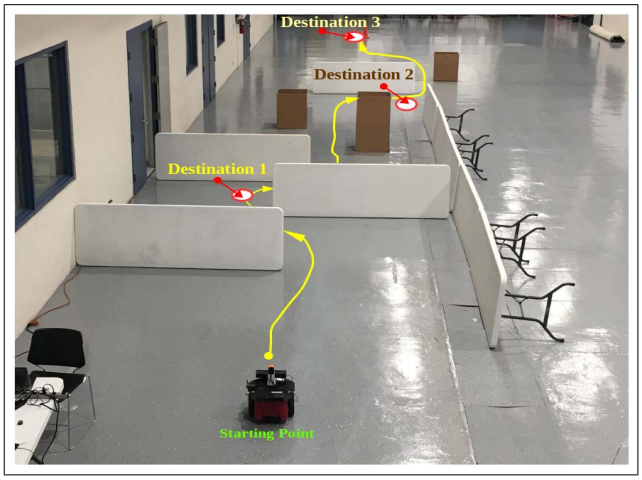


FIGURE 12. Disaster scenario set up in LAB.

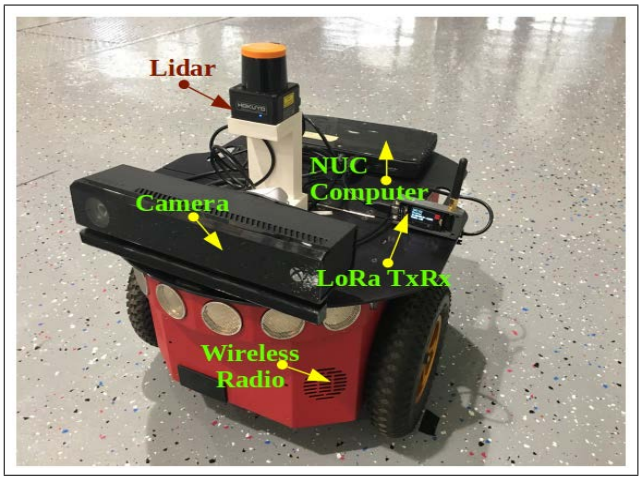


FIGURE 13. Pioneer-P3DX Robot used for testing.

Rescuer robot. They were: i) X=11, Y= 0.7; ii) X=27, Y = −2; iii) X= 48, Y = −1 and are shown in Figure 12. As in the simulation phase, the base station RoboMaC-B module communicated with the RoboMaC-R on the robot from a distance of 1.6 miles. Figure 13 shows a Pioneer-P3DX robot used for testing, equipped with a Hokuyo LiDAR, Kinect Camera, RoboMaC-R, Wireless radio, and NUC computer.

The base station sent various destinations sequentially to the robot. The RoboMaC-R on the Robot received the LoRa packet, which contained the LI, camera enable bit, wireless radio enable bit, and the E-Stop bit. The algorithm read these bits and made decisions accordingly, as discussed in Section IV. Once the RoboMaC-R received the location information of the first victim, the robot checked the status of the E-Stop bit. If the E-Stop bit was 0, the received LI (X=11; Y=0.7) was given to the FISVFH navigation algorithm, and the robot started moving to the first destination, as shown in Figure 14(a). At the same time, the RoboMaC-R transmitted

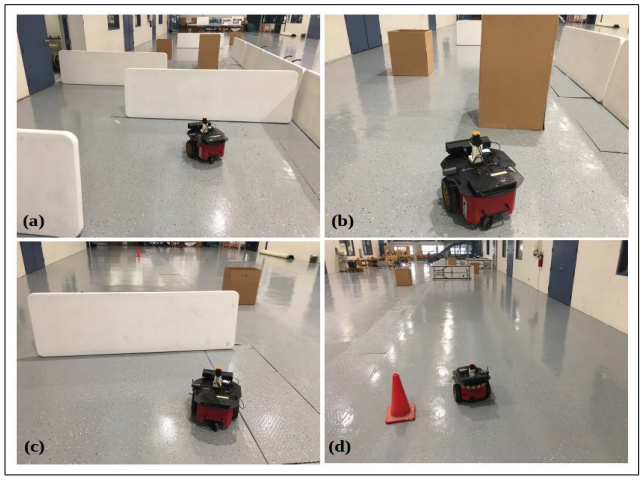


FIGURE 14. Pioneer-P3DX navigate to destinations.

location data back to the base station. This enabled the operator at the base station to continuously monitor the robot's location. Once the robot reached the first destination, the base station could enable/disable the camera bit and wireless radio enable/disable bit, as required. This enabled video recording on the robot's computer, and using the wireless radio, the victim could communicate directly with the operator at the base station. After providing needed assistance to the first victim, the operator at the base station assigned the location (X=27, Y = −2) of the next victim to the robot. Figure 14(b) shows the robot navigating to the second destination. After assisting the second victim, the robot waited for instructions from the base station on the next destination. The base station then transmitted the location information of the third victim (X=48 and Y = −1). Figure 14(c) shows the robot on its way to this destination, while Figure 14(d) captures its arrival. The path followed through the environment from start to finish is shown in Figure 12. Figure 15 shows the base station RoboMaC-B module sending the location information of the third destination to the robot while receiving the odometer data from the robot. Figure 16 shows the RoboMaC-R module on the robot receiving the LI of the third destination from the base station while transmitting the current odometer data back to the base station. Once the robot has visited all destinations, the base station instructs the robot to return to the starting point. The rescue team can then use the information collected by the robot for additional operational planning.

An initial study has been conducted to compare the proposed communication architecture with the private WiFi network that is commonly used in disaster management systems. To diversify our results, WiFi network was established twice with two different routers: Netgear router model number N600 WNDR3700 and another high-end Asus router model number AX88u. These routers support both 2.4GHz and 5GHz, frequency bands. Communication range with the SAR robot's computer across a range of

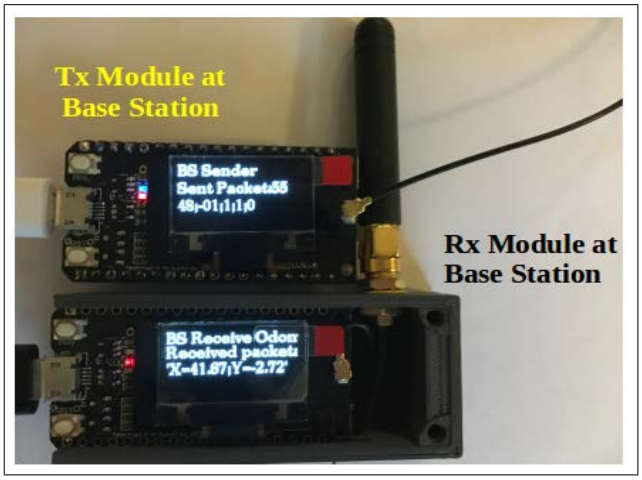


FIGURE 15. RoboMaC-B module at the base station.

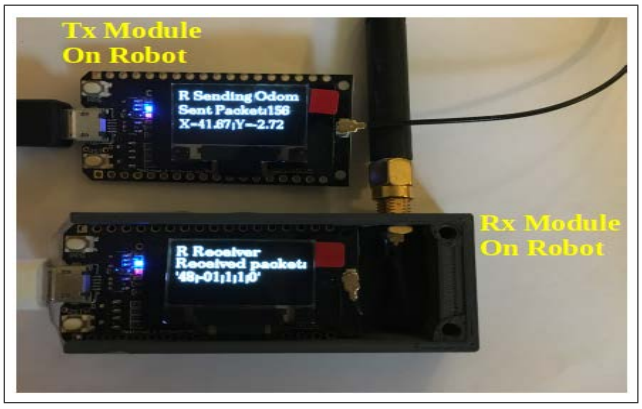


FIGURE 16. RoboMaC-R module on the Robot.

18-22 meters with the Netgear N600 and 150-180 meters with the Asus–ax88u with 5GHz and 2.4GHz, respectively, were achieved. In addition, we compare our results with SENEKA DMS system [9]. In SENEKA DMS, they employed a combination of Fraunhofer IIS low-power, low-data-rate snet® technology and commercially available IEEE 802.11 components in a multihop topology to establish a private network for managing the rescue mission in a disaster site. Their devices were able to communicate up to a range of 100 meters. With our proposed communication architecture (LoRa-based), the range achieved was 1.6 miles.

A performance analysis was carried out by testing the system's repeatability in order to ensure the quality of service and system performance. The same location information was sent to the robot a dozen times via the RoboMaC devices, and the robot was able to receive the data without any errors at all times, and the FISVFH navigation algorithm was able to drive the robot to the destination along the same path every time as shown in Figure 12.

VI. CONCLUSION AND FUTURE WORK

A Novel LoRa LPWAN-based communication architecture for search & rescue missions has been developed and tested. A SAR robot named *Rescuer*, equipped with a communication device named RoboMaC has been deployed. RoboMaC is a novel full-duplex LoRa-based communication device that can function as a transceiver unit, transmitting control commands from the base station to the *Rescuer* robot while also receiving the robot's location information at the base station. Testing was initially carried out in a simulated disaster environment in Gazebo, and then it was extended to an actual test setup inside Detroit Mercy's lab facility using a Pioneer P3DX robot. The signal range achieved with the RoboMaC devices was 1.6 miles. As a future extension to this study, high-end LoRa modules [46] can be used to extend the range of the proposed system up to 70%. By leveraging the available ISM frequency bands, more RoboMaC devices could be integrated into the SAR operations to incorporate more *Rescuers* in the mission. Also, by using an Interoperability Profile (IOP) [47], various advanced SAR robots can be deployed in the mission. Currently, the authors are working on extending the work to include various teams of SAR robots performing SAR missions in outdoor environments.

REFERENCES

[1] (Oct. 12, 2020). *The Human Cost of Disasters—An Overview of the Last 20 Years 2000–2019—World*. ReliefWeb. Accessed: May 24, 2022. [Online]. Available: https://reliefweb.int/report/world/human-cost-disasters-overview-last-20-years-2000-2019

[2] S. Gossett. (Jun. 25, 2019). *10 examples of Rescue Robots*. Built In. Accessed: May 24, 2022. [Online]. Available: https://builtin.com/robotics/rescue-robots

[3] P. Feuilherade. (Jul. 19, 2017). *Robots to the Rescue!*. Robohub. Accessed: May 24, 2022. [Online]. Available: https://robohub.org/robots-to-the-rescue/

[4] O. Mitchell. (Oct. 18, 2017). *Disaster Recovery Robots Working Overtime*. Accessed: May 24, 2022. [Online]. Available: https://becominghuman.ai/disaster-recovery-robots-working-overtime-62bcbd520ab8

[5] G. De Cubber, D. Doroftei, K. Rudin, K. Berns, A. Matos, D. Serrano, J. Sanchez, S. Govindaraj, J. Bedkowski, R. Roda, E. Silva, and S. Ourevitch, "Introduction to the use of robotic tools for search and rescue," in *Search and Rescue Robotics—From Theory to Practice*. London, U.K.: IntechOpen, 2017, doi: 10.5772/intechopen.69489

[6] G. De Cubber, D. Doroftei, D. Serrano, K. Chintamani, R. Sabino, and S. Ourevitch, "The EU-ICARUS project: Developing assistive robotic tools for search and rescue operations," in *Proc. IEEE Int. Symp. Saf., Secur., Rescue Robot. (SSRR)*, Oct. 2013, pp. 1–4, doi: 10.1109/SSRR.2013.6719323.

[7] S. Govindaraj, K. Chintamani, J. Gancet, P. Letier, B. van Lierde, Y. Nevatia, G. De Cubber, D. Serrano, M. Esbri Palomares, J. Bedkowski, C. Armbrust, J. Sanchez, A. Coelho, and I. Orbe, "The ICARUS project–command, control and intelligence (C2I)," in *Proc. IEEE Int. Symp. Saf., Secur., Rescue Robot. (SSRR)*, Oct. 2013, pp. 1–4, doi: 10.1109/SSRR.2013.6719356.

[8] Q. Ren, K. L. Man, E. G. Lim, J. Lee, and K. K. Kim, "Cooperation of multi robots for disaster rescue," in *Proc. Int. SoC Design Conf. (ISOCC)*, Nov. 2017, pp. 133–134, doi: 10.1109/ISOCC.2017.8368834.

[9] H.-B. Kuntze, C. W. Frey, I. Tchouchenkov, B. Staehle, E. Rome, K. Pfeiffer, A. Wenzel, and J. Wollenstein, "SENEKA–sensor network with mobile robots for disaster management," in *Proc. IEEE Conf. Technol. for Homeland Secur. (HST)*, Nov. 2012, pp. 406–410, doi: 10.1109/THS.2012.6459883.

[10] J. A. Godoy, F. Cabrera, V. Arana, D. Sanchez, I. Alonso, and N. Molina, "A new approach of V2X communications for long range applications in UAVs," in *Proc. 2nd URSI Atlantic Radio Sci. Meeting (AT-RASC)*, May 2018, pp. 1–4, doi: 10.23919/URSI-AT-RASC.2018.8471484.

[11] M. Behjati, A. B. M. Noh, H. A. H. Alobaidy, M. A. Zulkifley, R. Nordin, and N. F. Abdullah, "LoRa communications as an enabler for Internet of Drones towards large-scale livestock monitoring in rural farms," *Sensors*, vol. 21, no. 15, p. 5044, Jul. 2021, doi: 10.3390/s21155044.

[12] C. Hsieh, Z. Ye, C. Huang, Y. Lee, C. Sun, T. Wen, J. Juang, and J. Jiang, "A vehicle monitoring system based on the LoRa technique," *Int. J. Transport Vehicle Eng.*, vol. 11, no. 5, pp. 1100–1106, 2017.

[13] Y.-S. Chou, Y.-C. Mo, J.-P. Su, W.-J. Chang, L.-B. Chen, J.-J. Tang, and C.-T. Yu, "I-car system: A LoRa-based low power wide area networks vehicle diagnostic system for driving safety," in *Proc. Int. Conf. Appl. Syst. Innov. (ICASI)*, May 2017, pp. 789–791, doi: 10.1109/ICASI.2017.7988549.

[14] T. Maneekittichote and T. Chanthasopeephan, "Mobile robot swarm navigation and communication using LoRaWan," in *Proc. 6th Int. Conf. Mechatronics Robot. Eng. (ICMRE)*, Feb. 2020, pp. 22–25, doi: 10.1109/ICMRE49073.2020.9064973.

[15] A. Junaedy, H. Masuta, K. Sawai, T. Motoyoshi, and N. Takagi, "LPWAN-based real-time 2D SLAM and object localization for teleoperation robot control," *J. Robot. Mechatronics*, vol. 33, no. 6, pp. 1326–1337, Dec. 2021.

[16] L. M. Fernández-Ahumada, J. Ramírez-Faz, M. Torres-Romero, and R. López-Luque, "Proposal for the design of monitoring and operating irrigation networks based on IoT, cloud computing and free hardware technologies," *Sensors*, vol. 19, no. 10, p. 2318, May 2019, doi: 10.3390/s19102318.

[17] M. Sidorov, J. H. Khor, P. V. Nhut, Y. Matsumoto, and R. Ohmura, "A public blockchain-enabled wireless Lora sensor node for easy continuous unattended health monitoring of bolted joints: Implementation and evaluation," *IEEE Sensors J.*, vol. 20, no. 21, pp. 13057–13065, Nov. 2020, doi: 10.1109/JSEN.2020.3001870.

[18] A. Augustin, J. Yi, T. Clausen, and W. Townsley, "A study of LoRa: Long range & low power networks for the Internet of Things," *Sensors*, vol. 16, no. 9, p. 1466, Sep. 2016, doi: 10.3390/s16091466.

[19] R. P. Centelles, F. Freitag, R. Meseguer, L. Navarro, S. F. Ochoa, and R. M. Santos, "A LoRa-based communication system for coordinated response in an earthquake aftermath," *Multidisciplinary Digital Publishing Institute Proceedings*, vol. 31, no. 1, p. 73, Nov. 2019, doi: 10.3390/proceedings2019031073.

[20] *LoRa (LoRaWAN) Modules: LPWA Products*. Murata Manufacturing. Accessed: May 24, 2022. [Online]. Available: https://www.murata.com/en-us/products/connectivitymodule/lpwa/overview/lora

[21] E. Gambi, L. Montanini, D. Pigini, G. Ciattaglia, and S. Spinsante, "A home automation architecture based on Lora technology and message queue telemetry transfer protocol," *Int. J. Distrib. Sensor Netw.*, vol. 14, no. 10, Oct. 2018, Art. no. 155014771880683, doi: 10.1177/1550147718806837.

[22] *What is LoRa*. Semtech. Accessed: Jan. 23, 2021. [Online]. Available: https://www.semtech.com/lora/what-is-lora

[23] (Sep. 9, 2020). *FCC Rules*. Accessed: May 24, 2022. [Online]. Available: https://afar.net/tutorials/fcc-rules/

[24] E. Team. (May 22, 2019). *Everything RF*. LoRa Frequency Bands in North America—Everything RF. Accessed: May 24, 2022. [Online]. Available: https://www.everythingrf.com/community/lora-frequency-bands-in-north-america

[25] (Nov. 17, 2020). *LoRawan is Secure (but Implementation Matters)*. LoRa Alliance. Accessed: May 24, 2022. [Online]. Available: https://lora-alliance.org/resource hub/lorawan-is-secure-but-implementation-matters/

[26] M. P. Manuel and K. Daimi, "Implementing cryptography in Lora based communication devices for unmanned ground vehicle applications," *Social Netw. Appl. Sci.*, vol. 3, no. 4, pp. 1–14, Apr. 2021, doi: 10.1007/s42452-021-04377-y.

[27] *Robots/AMR Pioneer Compatible—ROS Wiki*. Accessed: Apr. 2, 2021. [Online]. Available: https://wiki.ros.org/Robots/AMR Pioneer Compatible

[28] K. Khoshelham and S. O. Elberink, "Accuracy and resolution of Kinect depth data for indoor mapping applications," *Sensors*, vol. 12, no. 2, pp. 1437–1454, Feb. 2012, doi: 10.3390/s120201437.

[29] (Mar. 17, 2016). *Meet Intel NUC NUC6i7KYK—Skull Canyon Arrives With Thunderbolt 3*. Legit Reviews. Accessed: Apr. 2, 2021. [Online]. Available: https://www.legitreviews.com/meet-intel-nuc-nuc6i7kyk-skull-canyon-arrives-thunderbolt-3180007

[30] *Hokuyo UST-10LX Scanning Laser Rangefinder | Acroname*. Accessed: Apr. 2, 2021. [Online]. Available: https://Acroname.com/store/r359-ust-10lx downloads

[31] *Hokuyo node/Tutorials/Using The Hokuyo Node—ROS Wiki*. Accessed: Sep. 10, 2021. [Online]. Available: https://Wiki.ros.org/hokuyo-node/Tutorials/UsingTheHokuyoNode

[32] (Feb. 1, 2021). *ESP32-PICO-D4 Datasheet*. [Online]. Available: https://www.espressif.com/sites/default/files/documentation/esp32-pico-d4-datasheet-en.pdf

[33] *SX1276 | 137 MHz to 1020 MHz Long Range Low Power Transceiver | Semtech*. Accessed: May 22, 2010. [Online]. Available: https://www.semtech.com/products/wireless-RF/lora-transceivers/sx1276

[34] Koyanagi, Fernando. (Apr. 2018). *ESP32 LoRa: You Can Reach up to 6.5 Km! 20*. Accessed: Nov. 27, 2020. [Online]. Available: https://www.instructables.com/ESP32-LoRa-You-Can-Reach-Up-to-65-Km/

[35] *Antenna 900 MHz 7dbi Magnetic-Mount Omni W/2-Meter Cable to RP-SMA/SMA*. [Online]. Available: https://www.data-alliance.net/antenna-900mhz-7dbi-magnetic-mount-omni-w-2-meter-cable-to-rp-sma-sma/

[36] *LoRaWAN USA Frequencies, Channels and Sub-Bands for IoT Devices*. BARANI DESIGN Technologies. [Online]. Available: www.baranidesign.com/faq-articles/2019/4/23/LoRaWAN-usa-frequencies-channels-and-sub-bands-for-iot-devices: : text=The USA LoRaWAN frequency spectrum

[37] J. Barani. (Apr. 23, 2019). *LoRaWAN USA Frequencies, Channels and Sub-Bands for IoT Devices*. (May 5, 2020). [Online]. Available: https://www.baranidesign.com/faq-articles/2019/4/23/LoRaWAN-usa-frequencies-channels-and-sub-bands-for-iot-devices

[38] *Spread Factor VS Payload Size on LoRa. (n.d.). I LoRa You LoRa*. Accessed: Nov. 18, 2019. [Online]. Available: https://myLoRaWAN.blogspot.com/2016/05/spread-factor-vs-payload-size-on-lora.html

[39] M. Zachmann. (Dec. 30, 2019). *The Best LoRa Settings for Range and Reliability*. [Online]. Available: https://medium.com/home-wireless/testing-lora-radios-with-the-limesdr-mini-part-2-37fa481217ff

[40] E. Bäumker *et al.*, "Minimizing power consumption of LoRa and LoRaWAN for low-power wireless sensor nodes," *J. Phys., Conf. Ser.*, vol. 1407, Nov. 2019, Art. no. 012092, doi: 10.1088/1742-6596/1407/1/012092.

[41] S. Kim, H. Lee, and S. Jeon, "An adaptive spreading factor selection scheme for a single channel Lora modem," *Sensors*, vol. 20, no. 4, p. 1008, Feb. 2020, doi: 10.3390/s20041008.

[42] *Confidentiality, Integrity, Availability: The three components of the CIA Triad*. Accessed: Dec. 7, 2019. [Online]. Available: https://security.blogoverflow.com/2012/08/confidentiality-integrity-availability-the-three-components-of-the-cia-triad/

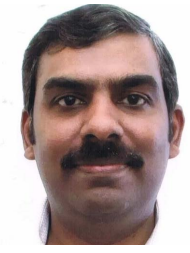
[43] K. Balan, M. P. Manuel, M. Faied, M. Krishnan, and M. Santora, "A fuzzy based accessibility model for disaster environment," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, Montreal, QC, Canada, May 2019, pp. 2304–2310.

[44] D. Hershberger, D. Gossow, and J. Faust. (Jun. 16, 2018). *RViz*. Accessed: Jan. 15, 2021. [Online]. Available: http://wiki.ros.org/rviz

[45] N. Koenig and A. Howard. (Oct. 10, 2014). *Gazebo*. Accessed: Jan. 10, 2021. [Online]. Available: http://wiki.ros.org/gazebo

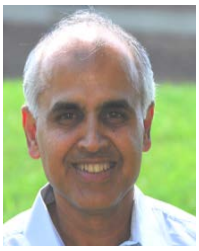
[46] *Sub-1 GHz. (n.d.)*. Accessed: Dec. 10, 2019. [Online]. Available: https://www.ti.com/wireless-connectivity/simplelink-solutions/sub-1-ghz/overview.html

[47] D. Serrano. *Introduction to JAUS for Unmanned Systems Interoperability—Joint Architecture for Unmanned Systems*. Accessed: May 24, 2022. [Online]. Available: https://www.sto.nato.int



MELVIN P. MANUEL (Student Member, IEEE) received the B.Sc. degree in electronics from Mahatma Gandhi University, Kerala, India, in 2002, the M.Sc. degree in electronics from Bharathidasan University, Tamil Nadu, India, in 2004, and the M.Tech. degree in sensor systems technology from the Vellore Institute of Technology, Tamil Nadu, in 2007. He is currently pursuing the Ph.D. degree in electrical engineering with the University of Detroit Mercy, MI, USA. From 2008 to 2011, he was a worked as an Assistant Professor with the School of Electronics Engineering, Vellore Institute of Technology. From 2012 to 2014, he was with the Fluid Control Research Institute (FCRI) as a Senior Research Fellow. His research interests include robot-based disaster management, sensor fusion, automobile electronics, and network security systems.

MARIAM FAIED (Senior Member, IEEE) was a Visiting Research Professor with the University of Michigan Ann Arbor. During Ph.D. degree, her research was with the Aerospace, Robotics and Control (ARC) Laboratory, University of Michigan on collaborative control of multiple unmanned vehicles, adversarial strategies, and advanced mission planning. In 2010, she worked as a Postdoctoral Researcher with the University of Michigan. She supervised undergraduate, master's, and Ph.D. students and served as a Ph.D. Committee Member. She worked as an Interim Director of the ARC Laboratory for six months. In 2012, she was appointed as an Assistant Professor with Fayoum University, Egypt. She developed the Mechatronics Engineering Department curriculum, which was audited by the Egyptian Supreme Council of Universities and approved. In 2013, she was selected to serve as the Mechatronics Program Chair for Fayoum University. She received Outstanding Program Coordinator Honor and Best Paper in Session Award at AIAA GNC Conference. Her research interests include the broad area of design, analysis and optimization of planning, and control algorithm for robotics.



MOHAN KRISHNAN (Life Senior Member, IEEE) received the B.Tech. degree in electrical engineering from the Indian Institute of Technology, Madras, India, the M.Tech. degree in electrical engineering from the Indian Institute of Technology, Kanpur, India, and the Ph.D. degree in electrical engineering from the University of Windsor, Windsor, ON, Canada. He is currently a Professor Emeritus with the Electrical & Computer Engineering & Computer Science Department, University of Detroit Mercy (UDM). His research interests include the applications of digital signal processing, including pattern recognition problems involving both 1-D and 2-D signals, such as signature verification and identification of shape contours of objects, problems involving the use of computational intelligence techniques, such as fuzzy logic and neural networks in intelligent control and autonomous vehicle navigation, modeling of mechatronic systems, and engineering education. He has served as a Consultant for Army Tank Command under the Army's SFRE Scientific Services Program on the use of neuro fuzzy techniques to predict engine oil quality degradation. He has published extensively in the area of computational intelligence, in particular in modeling handwritten signatures using neural networks for the purpose of authentication.