J Inf Process Syst, Vol.17, No.6, pp.1097~1114, December 2021
https://doi.org/10.3745/JIPS.03.0169

ISSN 1976-913X (Print)
ISSN 2092-805X (Electronic)

JOURNAL OF INFORMATION PROCESSING SYSTEMS JIPS

Wireless Mobile Sensor Networks with Cognitive Radio Based FPGA for Disaster Management

G. A. Preethi Ananthachari*

Abstract
The primary objective of this work was to discover a solution for the survival of people in an emergency flood. The geographical information was obtained from remote sensing techniques. Through helpline numbers, people who are in need request support. Although, it cannot be ensured that all the people will acquire the facility. A proper link is required to communicate with people who are at risk in affected areas. Mobile sensor networks with field-programmable gate array (FPGA) self-configurable radios were deployed in damaged areas for communication. Ad-hoc networks do not have a centralized structure. All the mobile nodes deploy a temporary structure and they act as a base station. The mobile nodes are involved in searching the spectrum for channel utilization for better communication. FPGA-based techniques ensure seamless communication for the survivors. Timely help will increase the survival rate. The received signal strength is a vital factor for communication. Cognitive radio ensures channel utilization in an effective manner which results in better signal strength reception. Frequency band selection was carried out with the help of the GRA-MADM method. In this study, an analysis of signal strength for different mobile sensor nodes was performed. FPGA-based implementation showed enhanced outcomes compared to software-based algorithms.

Keywords
Cognitive Radio, Disaster Management, FPGA, GRA, Sensors, Signal Strength

1. Introduction

Our Earth has observed a large number of natural disasters recently, resulting in enormous fatalities for people and other living organisms, building infrastructure, and more or less the entire region involved. Catastrophes like cyclones, earthquakes, and floods, have constant effects in several places at irregular times, leading to the intensification of the destruction of life and public/individual property. Even though science and technology have made immense progress in many areas, scientists are still unable to precisely forecast the time and place of these disasters and the level of hazard that might occur. Some of the intense hurricane floods in recent times include Hurricane Dorian, which thrashed the Bahamas with its severe vicious winds on September 1, 2019. In 2015, north east monsoon generated heavy rainfall in South India especially in the region of Tamilnadu and Andhra, were there was a drastic flood that affected the daily life. In 2013, in the north Indian region Uttarkhand a flash flood, caused a huge catastrophic disaster in the form of a cloud burst and landslide. For all these, the focus has been on mitigating the harm that might

Manuscript received May 18, 2020; first revision November 7, 2020; accepted March 5, 2021.
Corresponding Author: Preethi Ananthachari (preethi.ga@wsu.ac.kr)
* Technology Studies, Endicott College of International Studies, Woosong University, Daejeon, South Korea (preethi.ga@wsu.ac.kr)

be caused by natural adversity and to remain prepared for recovery management and reform operations. Calamity recovery and reconstruction operations have always been difficult tasks for the administration, confined authorities, and the Indian nation.

The major plan of the army, police officials, local volunteers, and other rescue officers immediately after an event is to look for survivors and to aid the wounded. The initial responders who reach a site immediately after a disaster have to deal with many problems and confrontations. One of the most vital requirements in disaster recovery and crisis situations is to establish consistent and uninterrupted communication between the army officers, medical aid team, and other rescue teams.

Effective communication is inevitable in coordinating the rescue team and survivors of the disaster. To establish efficient and swift recovery, the rescue team, police officials, and individual volunteers involved need to be moving at a fast pace through diverse locations within the area to diminish the effects and to discover more survivors from the catastrophe. However, in many circumstances, the communication infrastructure breaks down and proper communication cannot be achieved in the area which leads to longer latencies in crisis operations and increased damage to humankind. Mobile sensor networks play an important role in risk management systems. Floods will affect the communication systems which can make it not possible to have regular communication. Thus a few infrastructure networks are implemented as a temporary arrangement. Since regions in the proximity also have the possibility of being affected by the flood. There may be disruption of signals, which are deployed using mobile sensor networks. At the destination end, signal reception will be interrupted, and result in noise signals. To prevent those compliances, FPGA is incorporated with cognitive radio and implemented. Cognitive radio embedded with FPGA substantially overcomes the interruptions incurred in communications. In turn, it paves way for the increased survival rate of people.

2. Related Work

A variety of wireless sensor studies have been conducted, starting from typical small-scale sensors to advanced mobile sensors (both wired and wireless types), temperature, underwater sensors, etc., in various research institutes such as in the Centre of Excellence in Wireless Technology, Chennai, India. A disaster detection and alerting system was presented by Kaur et al. [1]. The authors discussed the basic architecture of wireless sensor networks (WSNs) and how it will be helpful in disaster management schemes. The MANET or mobile ad hoc networks are capable of forming wireless networks with different topologies. The radios, which form the network use a specific band, data rate, and frequency. In [2], the authors focused on the implementation of an intelligent MANET (iMANET) in which wireless networks are established using reconfigurable radios so that intelligent networks can be formed. The nodes involved in the network formation are capable of detecting spectrum holes and are self-configurable to specific bandwidth and data rates based on the channel conditions. The cooperative sensing capability of the network helps to make accurate decisions. The entire iMANET is implemented in Xilinx Zynq, which is a GPU-FPGA system-on-a-chip (SoC).

Menon et al. [3] used the reliable routing technique (RRT) to ensure reliable data delivery at a destination device even when people with mobile devices are moving in the network, broadcasting property of the wireless network and a priority list of probable forwarding candidates at each device ensure increased throughput. Jayakumar and Gopinath [4] stated that the introduction of Bluetooth, 802.11, and hyper-LAN are helping commercial MANET deployments outside the military domain. To facilitate communication, routing protocols have been used to discover nodes nearby [4]. In [5], the author presented remote sensing application which plays a vital role in disaster management activities. It provides a database from which the evidence left behind disasters that have occurred before can be inferred. Hazard maps can be derived from remote sensing activities. Cyclone monitoring and warning, inundation mapping and damage assessment and drought management have been the major works carried out with the help of remote sensing techniques. Abba and Lee [6] suggested a low-power, low-cost, and highly accurate monitoring and control mechanism using autonomous sensor agents to dynamically reconfigure control of on-chip sensor networks by FPGAs. In [7], the authors proposed a "DistressNet," an ad-hoc wireless architecture that supports disaster response with distributed collaborative sensing, topology-aware routing using a multichannel protocol, and accurate resource localization.

However, designing a sensor network on FPGA is a tedious job to accomplish in a short period since designers are limited to extension work of the FPGA producer design requirement. Likewise, the amount of intricacy, as well as complexity of devise mechanization restricted in the FPGA, make it complicated for designers to twitch the FPGA circuitry to design and implement sensor networks. A cluster-based decentralized orchestration cooperative sensing scheme was proposed in [8], which was as entitled "Decentralized cooperative spectrum sensing for ad-hoc disaster relief network clusters" in another conference held in 2010 IEEE 71st Vehicular Technology Conference, Taipei, Taiwan. Choi et al. [9] proposed a distributed medium access control protocol that uses successive multiple collision detection phases for dense WSN environments by enhancing the typical carrier sense multiple access with collision resolution protocol that uses only a single a collision detection phase. Colliding stations are filtered so that only surviving stations are allowed to compete. The collision probability becomes considerably reduced. As a result, throughput increases.

Lotze et al. [10] developed a virtual architecture for hardware abstraction, an adaptive run-time system for managing cognition, and high-level design tools for cognitive radio development for embedding cognitive radio technology with a FPGA board. They proposed a framework for a receiver node that worked in two modes as discovery and communication. In [11], requirements for disaster management and innovative technology for an integrated disaster management communication and information system, addressing, in particular, network, configuration, scheduling, and data management issues during the response and recovery phases were discussed. The authors of [12] presented an experiment on ring oscillators in low voltage using thermal sensors. The sensibility of frequency increase was shown with the proposed system in which a quadratic polynomial function was utilized. A direct sensor FPGA interface was used by Oballe-Peinado et al. [13] to acquire data-parallel. FPGA does not have an analogue to digital converter. However it can be reprogrammed. Different capture modules have been analyzed.

In [14], the authors proposed FPGA-based on-chip sensors for network monitoring, which utilize autonomous reconfigurable sensors for controlling. It collects the signals of voltage and power; by self-aware capability, it reduces the power consumption of the FPGA. A 1,000 ms refresh time increases the on-chip sensor readings. Perera et al. [15] proposed a 1-wire protocol for on-chip sensor-based FPGA. Reprogrammable smart sensor nodes have been used with Zigbee. Processing and transducer functionalities are given in a single core that increases the power speed due to inter-process communication between sensors. In [16], the authors presented an autonomous RFID sensor node using a single ISM band, which was used for both power transfer and data communication. A rectenna harvests the electromagnetic energy transported by the dedicated radiofrequency source for charging a few-mF super-capacitor. For super-capacitors of 7 mF, the proposed autonomous sensor nodes were able to wirelessly communicate with the reader at 868 MHz for 10 minutes without interruption for a tag-to-reader separation distance of 1 m. The authors obtained the result from effective radiated powers of 2 W during the super-capacitor charging of 100 mW during wireless data communication. In [17], the authors proposed a Markov decision process-based approach for seamless mobility of heterogeneous networks. The rest of the sections describe our work plan, methodology, signal strength implementation based on the hidden Markov model (HMM), the results and analysis, signal strength based on FPGA, conclusion.

3. Work Plan

Disaster management and recovery plans have become inevitable because of global warming. Mobile sensor networks play a crucial role in data transmission and communication. They can be very helpful in disaster-prone areas. Mobile sensor networks make use of the technologies which can warn and provide an alert message for the instantaneous rescue operation to activate, whenever a disaster. The aim was to assess the technological solutions for controlling a disaster using mobile sensor networks via a disaster discovery and preparedness system and search and rescue operations. Perhaps the focus was on the fundamental architecture of mobile sensor networks, which are very helpful for communicating with survivors during disaster management and for the FPGA models that can be employed for diverse disaster-prone locations. Disaster Relief Teams (DFT) were sent to all the devastated areas and coastal regions, which need quick aid during this study.

Voice communication may become interrupted in due course at an affected site. A suitable routing type should be used in which distress tolerant networks have been used. This overcomes the problem of disturbances in communication in congested network areas. In disaster-affected zones survivors and responders (people) move periodically for rehabilitation. Collective-channels medium access control usage enables continuous communication without commotion. Protocols, such as low rate wireless personal area networks, Zigbee, and Wi-Fi are used for communication. Problems, such as link breaks, deviation in networks, and an inability to access the protocol are significant challenges in networks. Congestion and collision occur while sensor nodes compete to send data through available channels. Optimized link state routing protocols will be used when it is a proactive protocol. It periodically updates the routing path and the time consumption is less compared to reactive protocols. Here mobile sensor networks were implemented using cognitive radios with effective channel selection, which was quite helpful in communicating in emergencies. A real execution is proposed for disaster management for observing consistent situations through a low pervasive sensing method. The network architecture and the interconnecting devices for consistent measurement of constraints by sensors and data communication through sensor networks were considered in this framework. At the base level, huge numbers of sensors were incorporated. A huge quantity of real-time device values was produced through the sensor. Frequently, small amounts of data were often produced by the sensors.

Various sensors can be incorporated for measuring environmental conditions. Sensors connected to the FPGA board through a cognitive radio will get accurate data for further processing. The records are then distributed towards the system for serial monitoring. With the help of cognitive radio, sensor networks, and FPGA data analysis, the DFT can communicate rapidly and evacuate the people to a safer zone. FPGA sensor data is used to alert people for continuous automated monitoring of the affected regions.

This layer manages the infrastructure among all the sensor nodes. The interference of signals is mitigated through cognitive techniques of spectrum sensing. Cognitive radio technology increases channel utilization. The architecture design is shown in Fig. 1. In Algorithm 1, people from emergency areas will communicate through movement or by other means. Sensor data is collected through it and stored. Further processing is accomplished through cognitive ratio (CR) and FPGA analysis.

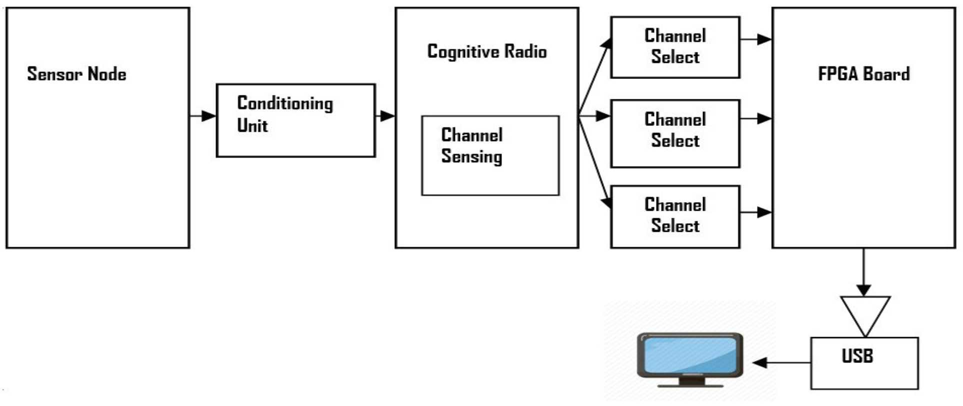

Fig. 1. Architecture of the proposed cognitive radio based FPGA.

Algorithm 1. Channel selection by cognitive radio

Begin:

1. Accumulate: ObtaineEmergency caution (Signal) information and store it in the sensornNodes.

2. Extract: a. Communicate through the mobile node (CR-augmented) and obtain emergency location data through the FPGA

b. Select users where user location information = Emergency location address

3. Transform: Combine the FPGA emergency user information with emergency warning information in the sensors.

4. Send: a. Obtain the emergency user information device address (MAC address of the sensors)

b. Make a link with the user device

c. Send a warning message over the sensor networks to selected user & devices

5. Display: Display warning information through a message.

End

4. Methodology

Simulation: The FPGA board incorporated with sensors was used for the simulation. ISim is an FPGA simulator that helps program the board with various sensors and transducers. Having developed the On-chip sensor boards with FPGA [14], we used it as the basic concept for embedding the sensors in the simulation. CR incorporation with sensors will enhance channel utilization and prevent communication disconnection. CRs enable current WSNs to overcome the scarcity problem of the spectrum, which is shared with many other successful systems such as Wi-Fi and Bluetooth. It has been shown that the coexistence of such networks can significantly degrade a WSN's performance. Moreover, cognitive technology could provide access not only to the new spectrum but also to the spectrum with better propagation characteristics. Channel selection methodology using the CR is depicted in Fig. 2.

Experimental set-up: To verify the above methodology implemented simulation results, an FPGA board was used. Mobile sensors send the signals and those signals are captured through cognitive techniques, which helps with sensing the spectrum of frequency bands for channel utilization.

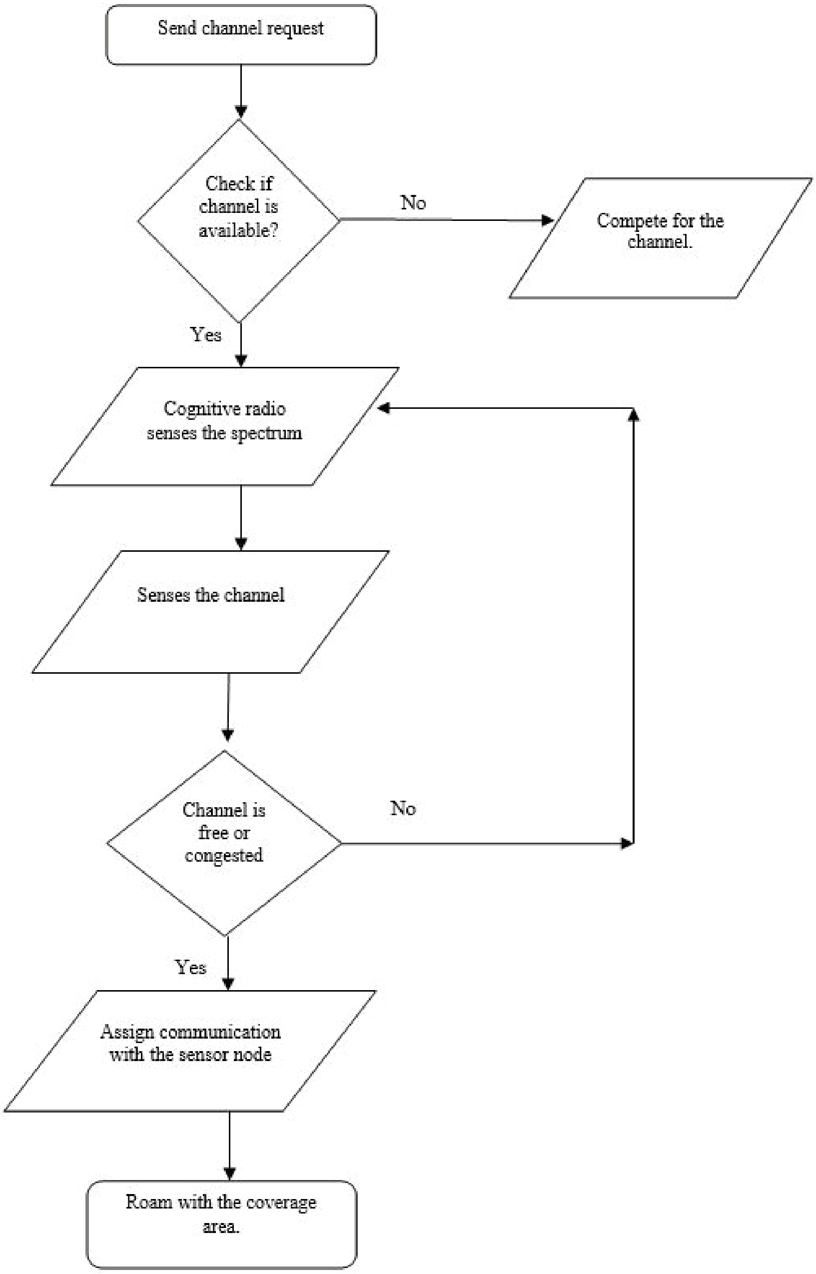

Fig. 2. Channel selection methodology.

Study and comparison: The computer simulation and experimental results were compared for validation of the FPGA-based mobile networking model and working nature. The computer simulation was carried out based on HMM. HMM was implemented and compared with the FPGA-based signal strength obtained through cognitive sensing. Signals were selected through spectrum sensing. However, the available spectrum could not be guaranteed because a long duration call could be interrupted and disconnected during the congestion of channels. All these issues should be overcome by spectrum management, which involves spectrum detection (sensing), spectrum decision, and spectrum sharing. To ensure seamless spectrum availability, spectrum handover should be carried out successfully. Heterogeneous spectrum sharing involves many aspects, such as target channel, switching delay, channel latency, channel interference, and channel capacity.

Spectrum detection: Spectrum detection is sensing the unused spectrum. First, it should check for primary users and available free channels. Channels should be sensed without any interference. Many spectrum sensing schemes are available such as energy-based, radio identification-based schemes, and waveform-based schemes [18].

Spectrum decision: This involves selecting the spectrum from the available free spectrum. The spectrum hole is decided by the received signal strength, spectrum congestion, and the number of primary users sharing the spectrum. Spectrum selection is more challenging as there are many different routes available between the source and destination. Quality-of-service (QoS) parameter consideration will help to select the spectrum. Here, the multiple attribute decision making method was employed for spectrum selection.

Spectrum sharing: Here network maintains the QoS by avoiding collision with other CR's. This also involves resource allocation. As CR's require frequent movement from one band of frequency to another band, fairness in the handoff from one band to another band is mandatory. In this work, there were four different bands and for each band, there was its respective primary user. Whenever a primary user appears, the CR should leave the channel and search for a free channel. This will be accomplished by the heterogeneous handoff algorithm. Grey relational analysis (GRA) was used for spectrum selection [19].

5. Signal Strength Estimation based on the Hidden Markov Model

Normally, Markov chains are based on random selections. The basic Markov process involves states, transition probabilities, and actions. Actions are based on the transition probabilities, which can be calculated with the help of states. Transition probability represents moving from one state to another state. In contrast, HMM involves hidden states. If our Markov process represents signal strength from different nodes, we consider states as sensor nodes with RSS $\{S_1, S_2 \ldots S_n\}$. Data from one node to another node travel through the hops. If the source node to destination node is far apart, more hop counts will be added. The destination node has no idea about the traversal history of its data. This can be illustrated with the help of the HMM [20].

The Markov process property can be shown as

$$P(X_t = j | X_1 = i_1, \ldots, X_{t-1} = i_{t-1}) = P(X_t = j | X_{t-1} = i_{t-1}) \quad (1)$$

The probability of moving from state *i* to state *j*.

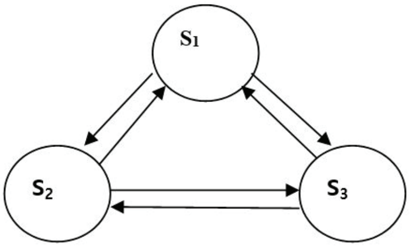

Fig. 3. States and transitions.

From Fig. 3, there are four states, the transition probability from S_1 to S_2 is given as 0.2 and from S_2 to S_3, it is given as 0.3. $S_3 \rightarrow S_1$ as 0.5. In matrix form,

$$p = \begin{matrix} 0.2 & 0.3 & 0.5 \\ 0.4 & 0.4 & 0.2 \\ 0.3 & 0.3 & 0.4 \end{matrix}$$

Likewise, note that the sum of each row is equal to 1. It represents weights. It shows the stochastic matrix. The *i,j* corresponds to $p_{i,j}$ which is the transition from state i to state j. Normally, probabilities are given at the initial stage as$q = (q_1, \dots, q_m)$ to be at each state at time $t = 0$. State i at time t will be equal to the i^{th} entry of the vector $p^m q$, which is the probability measure. Suppose that the signal S_1, S_2, S_3 has the transition probabilities, such as 0, 0.3 and 0.6 at some given time. The probability of the 50th node receiving the data could be calculated as

$$p = \begin{matrix} 0.2 & 0.3 & 0.5^{50} \\ 0.4 & 0.4 & 0.2 \\ 0.3 & 0.3 & 0.4 \end{matrix} . (0, 0.3, 0.6)^t$$

In the Hidden Markov Model, there is an invisible Markov chain that cannot be observed. Each state randomly generates one out of *m* observations that is visible. Dense deployment of sensor nodes produces signals at the same time and routes the signal through the hop count. For routing the data signal, sensors use some algorithms. During transmission of data, signals deteriorate, and not all the sensors data are received at the destination point. The transmissions of the data signal are depicted in terms of the Hidden Markov Model. As sensor node routes are unpredictable, the Hidden Markov Model suggests the best data reception method. Let us consider the Markov chain, with five states S1, S2, S3, S4, and S5. Suppose there are 2 observations, S1 and S2, in which,

$$P(S_1|S_3) = 0 \,, P(S_2|S_3) = 1 \quad (2)$$

$$P(S_1|S_4) = 0.4 \,, P(S_2|S_4) = 0.6 \quad (3)$$

$$P(S_1|S_5) = 0.2 \,, P(S_2|S_5) = 0.8 \quad (4)$$

By applying the Hidden Markov policy, consider that the data signal arrives from the sensor node S2. Suppose the data has been transmitted two times; then, the probability is 3x3, which has 9 options. The probability calculation is defined below in Eq (5);

$$P\big((S_2,S_2),(S_4,S_3)\big) = P\big((S_2,S_2)\big|(S_4,S_3)\big).P(S_4,S_3)) \\ = P(S_2|S_4).P(S_2|S_3).P(S_3|S_4).P(S_4) \tag{5}$$

5.1 Viterbi Algorithm: For Finding the Hidden States

Based on the observations, hidden states should be found. The straight forward approach will take multi-exponential time, like the traditional statistical approach that was discussed in an earlier section. A well-organized approach is the Viterbi algorithm [21]. The idea is as follows.

A series of observations are provided $m_1, \dots \dots \dots . m_t$. For each state $s, t = 1 \dots \dots T$

$$\vartheta_t(s) = \max_{s_1,\dots,s_{t-1}} P\{x_1 = s_1, \dots, x_{t-1} = x_{t-1}, x_t = s, m_1, \dots, m_t\} \tag{6}$$

From the above Eq (6), the maximum probability of the path at state *s* that ends at time *t* is provided with the observations. By using the Markov property, the observation is the perceptive path the ends with state *s* at time *t,* which equals to s^* at time *t*-1. This confirms that s^* is the value of the final state of the perceptive path that ends at time *t*-1. The forward recursion is given by

$$\vartheta_t(s) = \max_s P_{s,n}\alpha_n(m_t)\vartheta_{t-1}(s) \tag{7}$$

$\alpha_n(m_{t)}$ is the probability to acquire m_t. provided that the hidden Markov state is *j*. Consider 15 different paths through which the sensor nodes send data. Here, 1 symbolizes successful data and 0 is more likely for unsuccessful data. By using the Viterbi algorithm, from the below code in Fig. 4, the simulated observations are shown.

```
Simulated Observations:
    Obs_code         Obs_seq
0          1          Signal
1          1          Signal
2          0  Average_signal
3          1          Signal
4          0  Average_signal
5          0  Average_signal
6          1          Signal
7          0  Average_signal
8          1          Signal
9          1          Signal
10         0  Average_signal
11         0  Average_signal
12         0  Average_signal
13         1          Signal
14         1          Signal
```

Fig. 4. Observed code and sequence.

The HMM matrix and observable signal transition probabilities are given in Fig. 5.

From the outcome shown in Fig. 6, s is the observed probability (Here we have 3 observations, such as s_3, s_4, and s_5). Symbol "t" here represents different paths (hop) the signal traverses.

Nu[0,2] = 2.0 signifies that the sensor node s_3 with path 2, emits a strong signal. Path results are shown in Fig. 7.

```
HMM matrix:
      S3   S4   S5
S3   0.2  0.3  0.5
S4   0.4  0.4  0.2
S5   0.3  0.3  0.4

Observable matrix:
      S1   S2
S3    0    1
S4   0.4  0.6
S5   0.2  0.8
```

Fig. 5. Hidden Markov Model matrix and observable matrix.

```
Start Sending Signal

s=0 and t=1: Nu[0, 1] = 2.0
s=1 and t=1: Nu[1, 1] = 2.0
s=2 and t=1: Nu[2, 1] = 2.0
s=0 and t=2: Nu[0, 2] = 2.0
s=1 and t=2: Nu[1, 2] = 2.0
s=2 and t=2: Nu[2, 2] = 0.0
s=0 and t=3: Nu[0, 3] = 1.0
s=1 and t=3: Nu[1, 3] = 1.0
s=2 and t=3: Nu[2, 3] = 2.0
s=0 and t=4: Nu[0, 4] = 1.0
s=1 and t=4: Nu[1, 4] = 0.0
s=2 and t=4: Nu[2, 4] = 0.0
s=0 and t=5: Nu[0, 5] = 1.0
s=1 and t=5: Nu[1, 5] = 1.0
s=2 and t=5: Nu[2, 5] = 2.0
s=0 and t=6: Nu[0, 6] = 1.0
s=1 and t=6: Nu[1, 6] = 1.0
s=2 and t=6: Nu[2, 6] = 1.0
s=0 and t=7: Nu[0, 7] = 1.0
s=1 and t=7: Nu[1, 7] = 0.0
s=2 and t=7: Nu[2, 7] = 0.0
s=0 and t=8: Nu[0, 8] = 1.0
s=1 and t=8: Nu[1, 8] = 1.0
s=2 and t=8: Nu[2, 8] = 2.0
s=0 and t=9: Nu[0, 9] = 1.0
s=1 and t=9: Nu[1, 9] = 0.0
s=2 and t=9: Nu[2, 9] = 0.0

s=0 and t=10: Nu[0, 10] = 2.0
s=1 and t=10: Nu[1, 10] = 2.0
s=2 and t=10: Nu[2, 10] = 2.0
s=0 and t=11: Nu[0, 11] = 1.0
s=1 and t=11: Nu[1, 11] = 1.0
s=2 and t=11: Nu[2, 11] = 2.0
s=0 and t=12: Nu[0, 12] = 1.0
s=1 and t=12: Nu[1, 12] = 1.0
s=2 and t=12: Nu[2, 12] = 1.0
s=0 and t=13: Nu[0, 13] = 1.0
s=1 and t=13: Nu[1, 13] = 1.0
s=2 and t=13: Nu[2, 13] = 1.0
s=0 and t=14: Nu[0, 14] = 1.0
s=1 and t=14: Nu[1, 14] = 0.0
s=2 and t=14: Nu[2, 14] = 0.0
```

Fig. 6. Signal emission through a different path.

```
Start Backtrace

path[13] = 0
path[12] = 1
path[11] = 1
path[10] = 1
path[9]  = 2
path[8]  = 0
path[7]  = 1
path[6]  = 0
path[5]  = 1
path[4]  = 1
path[3]  = 0
path[2]  = 1
path[1]  = 2
path[0]  = 2
```

Fig. 7. Aggregation of the best path.

From the above result, 0 → Noise, 1 → Signal, 2 → Strong Signal. Finally, it can be concluded that, from the above Table 1, the signal node (S_5) shows strong signal traversal and reaches the signal in (S_2). Signals are prone to deteriorate. On the account of such mitigation, the strong signal becomes less intense and reaches a normal signal. In path 2, the sensor node (S_4) sends an average signal to S_2. Here average means a, signal near the threshold level. In path 3, S_3 represents noise, which could be damaged data, and it results in signal (S_2). Here, the resultant signal would have lost some data packets. A large number of sensor nodes as well as, its signals could not be handled very easily. Therefore, FPGA usage paves the way for getting the manipulation of huge sensor signals. A huge number of sensor nodes are multiplexed and integrated to receive it as whole dataset. The advantage of multiplexing is to fault-tolerate some weak signal or no signal.

Table 1. Observation of states and outcome

Path (hop)	Observed state	Best path
0	Signal (S_2)	Strong signal (S_5)
1	Signal (S_2)	Strong signal (S_5)
2	Average signal (S_1)	Signal (S_4)
3	Signal (S_2)	Noise (S_3)
4	Average signal (S_1)	Signal (S_4)
5	Average signal (S_1)	Signal (S_4)
6	Signal (S_2)	Noise (S_3)
7	Average signal (S_1)	Signal (S_4)
8	Signal (S_2)	Noise (S_3)
9	Signal (S_2)	Strong signal (S_5)
10	Average signal (S_1)	Signal (S_4)
11	Average signal (S_1)	Signal (S_4)
12	Average signal (S_1)	Signal (S_4)
13	Signal (S_2)	Noise (S_3)
14	Signal (S_2)	Noise (S_3)

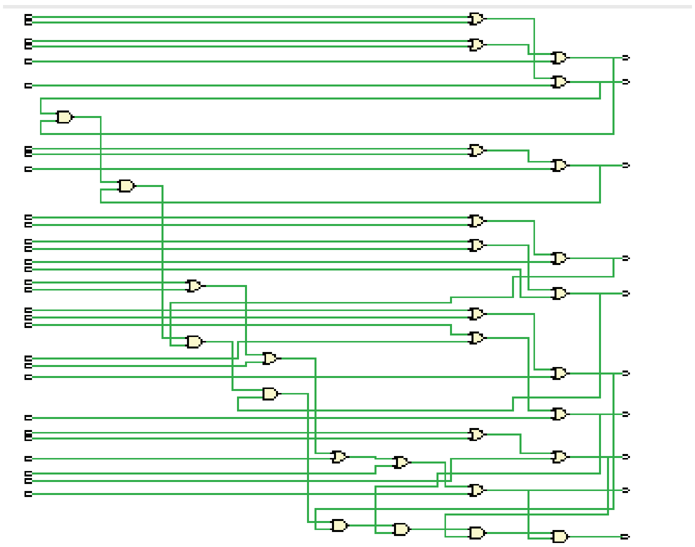

Fig. 8. FPGA implementation using 40 sensor nodes.

Fig. 8 shows different input signals multiplexed through the OR gate to form integration and results as a single signal output. A 4:1 multiplexer type was used. Here the FPGA board uses 29 cells, 40 I/O ports, and 59 Nets. Nets represent the logical connection of more than a one pin symbol instance. Usually, mobile sensor communication is prone to attenuation. As it is deployed in an emergency where there are flood-affected areas, the sensor nodes deploy their own infrastructure and act as a base station at times. Fig. 9 shows the in-depth view of logic cells deployed in FPGA.

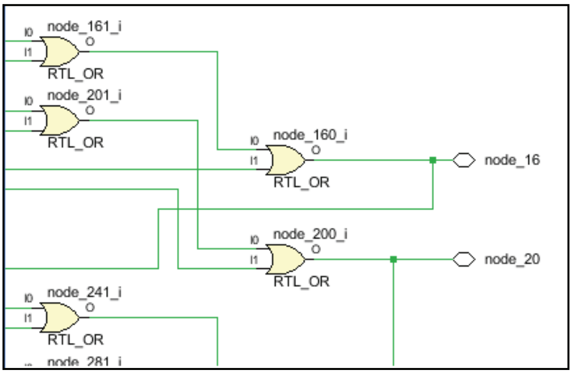

Fig 9. Zoom-in view of nodes.

6. Results and Analysis

As shown in Figs. 10–18, sensor nodes signal strength was measured and visualized. Received signal strength is represented in decibels (dB). Normally, the received signal strength depends on the transmitted power and received power. Overall, -60 dB to -70 dB is the signal strength range in mobile nodes. In sensor nodes, it might be less. As shown in Figs. 10–18, Node 1 measured a peak signal of 1.0; Node 2 showed its range from 20–60 –dBs; Node 3 signal range is from 0 to 5.5, which is a weak signal; Node 4 showed 0 to 10.5 at 500 ms and, -5 dBs at 1,250 ms; Node 5 displayed a peak of -3 dBs at 550 ms and fluctuated from -3 to +3 at 1,000 ms to 1,750 ms; Node 6 showed a low of 0.4 and a peak of -0.2; Node 7 showed -0.2 as a constant streaming signal; Node 8 showed a strong signal as -50 measured; Node 9 had a range from -30 to -70; Node 10 showed a range from -30 to -80; Nodes 11 and 12 consequently had their range as -20 to -80 and -30 to -70 dBs, respectively. Nodes 8 and 9 showed stronger signals compared to other nodes.

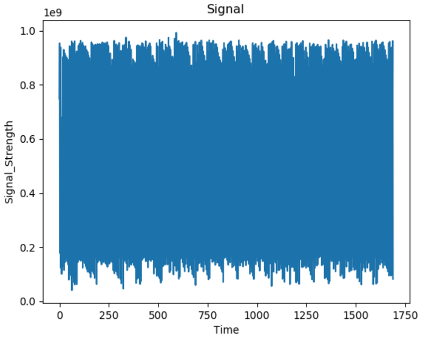

Fig. 10. Signal strength (Node 1).

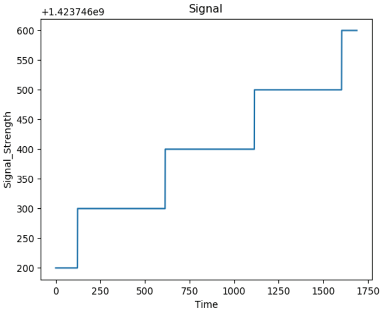

Fig. 11. Signal strength (Node 2).

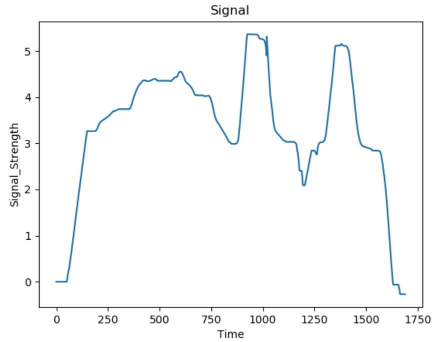

Fig. 12. Signal strength (Node 3).

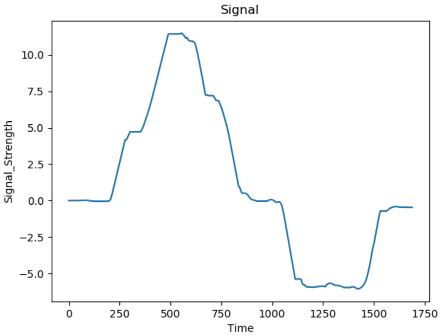

Fig. 13. Signal strength (Node 4).

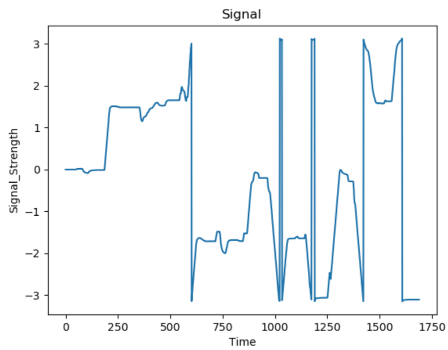

Fig. 14. Signal strength (Node 5).

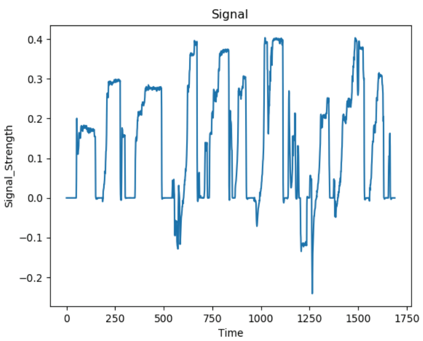

Fig. 15. Signal strength (Node 6).

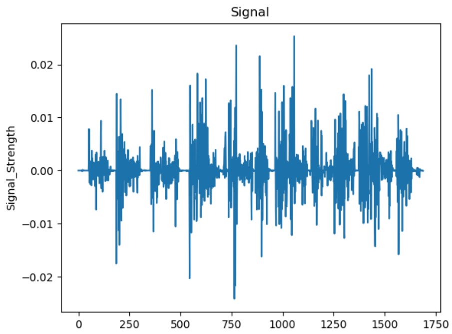

Fig. 16. Signal strength (Node 7).

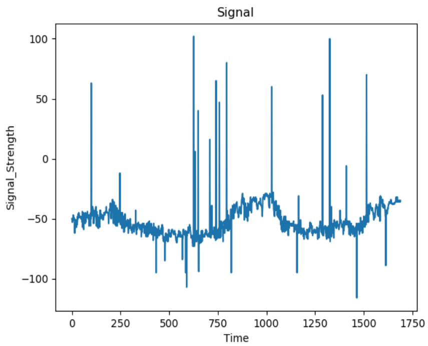

Fig. 17. Signal strength (Node 8).

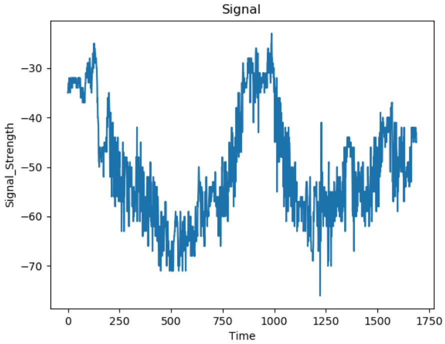

Fig. 18. Signal strength (Node 9).

6.1 Signal Processing through FPGA

In Figs. 19–29, for Nodes 1–11, signal strengths which were captured via FPGA are shown: Node 1, showed a standard stream in which its signal is weak; Node 2 showed a peak of -2.5 at nearly 2,000 ms. Node 3 had an increase from 0 to 120 dBs, which could be noise or an interruption from other sensor nodes; Node 4 had its range from 7.75 to 9.5; Node 5 had a peak of -1; Node 6 had a streaming signal of 0.3 for the entire duration. Therefore, for Node 7, this also had a streaming signal. Node 8 had a stronger signal of -40 to -80 dBs; Nodes 9–11 are shown in Figs 27–29. These nodes exhibit stronger signals compared to previous simulation results. However, these particular nodes are executed for a short duration, the same as a computer simulation.

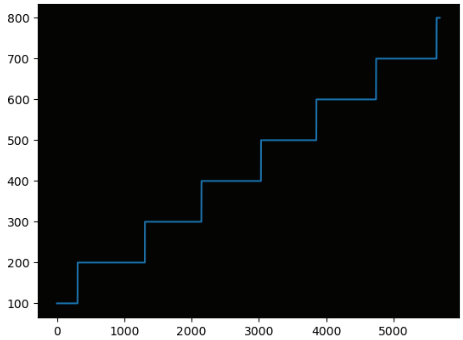

Fig. 19. FPGA signal (Node 1).

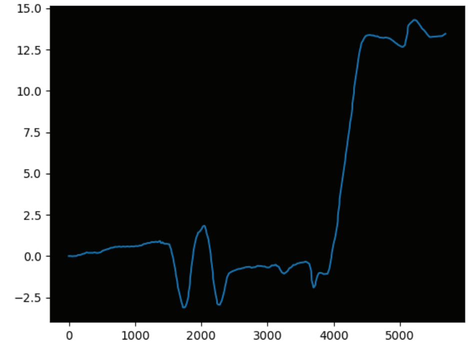

Fig. 20. FPGA signal (Node 2).

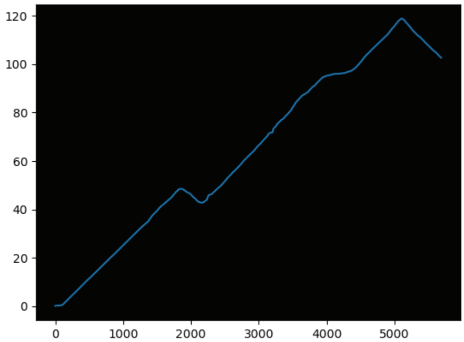

Fig. 21. FPGA signal (Node 3).

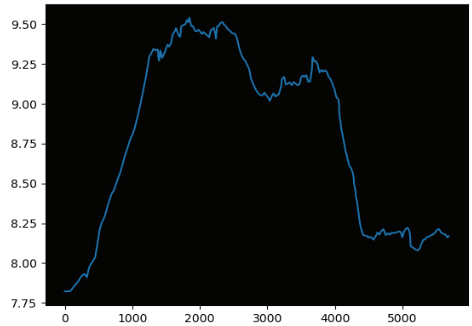

Fig. 22. FPGA signal (Node 4).

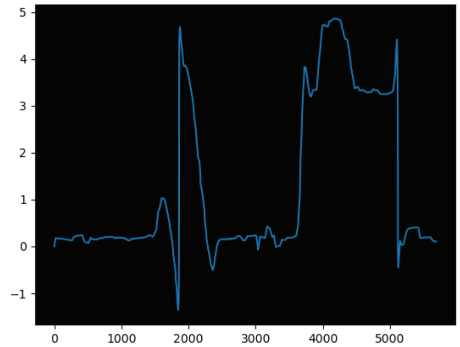

Fig. 23. FPGA signal (Node 5).

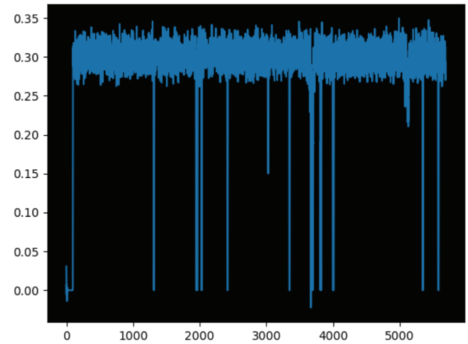

Fig. 24. FPGA signal (Node 6).

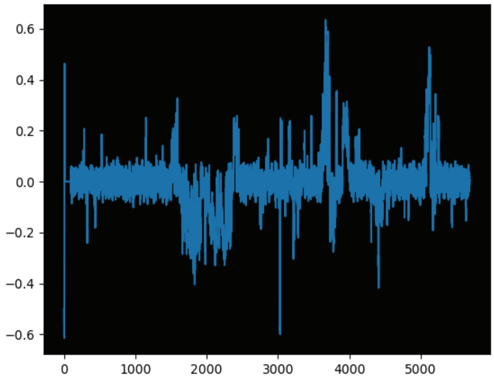

Fig. 25. FPGA signal (Node 7).

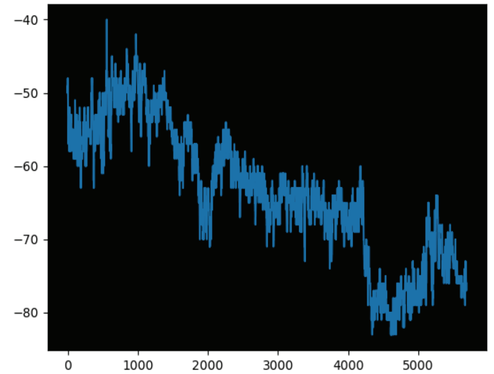

Fig. 26. FPGA signal (Node 8).

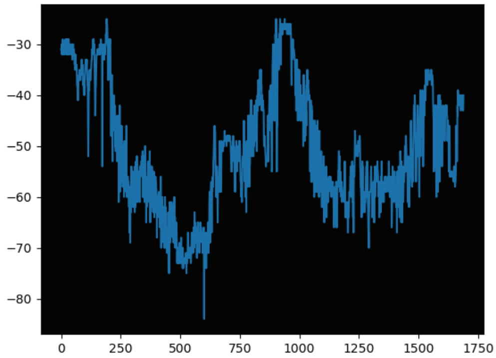

Fig. 27. FPGA signal (Node 9).

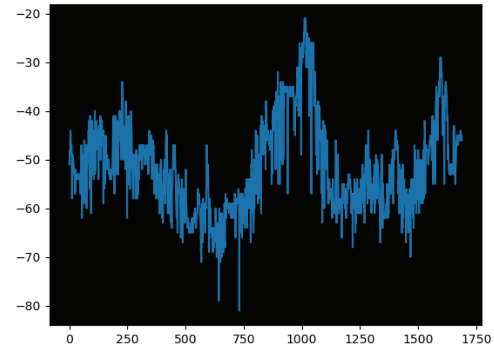

Fig. 28. FPGA signal (Node 10).

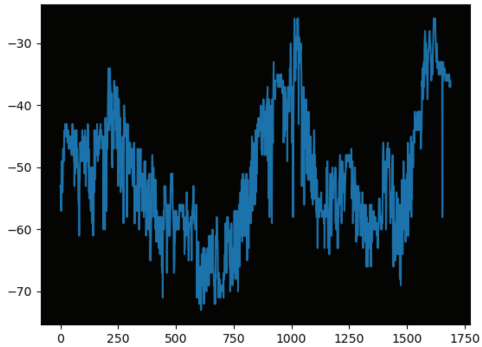

Fig. 29. FPGA signal (Node 11).

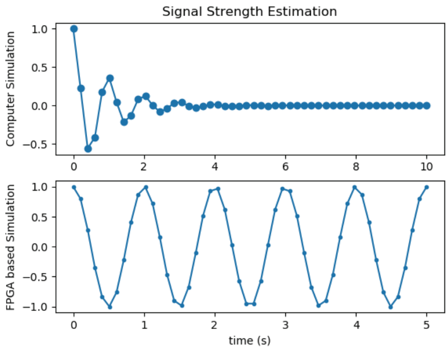

Fig. 30. Comparison of simulation results.

As shown in the above Fig. 30, the FPGA-based implementation incorporates a proper sine wave, whereas simulated signals are prone to interruptions and cancel each other. This results in no available signal. The capabilities of proper signal waves are achieved by cognitive spectrum sensing. The Vivado Simulator has been used for simulating FPGA sources. It supports VHDL as well as Verilog code files. Parsers for VHDL are xvhdl and for Verilog files, they are xvlog, which stockpile the parsed files into an HDL library on the memory disk. The HDL elaborator is xelab and the linker command. For a specified high-level unit, xelab invokes all sub-level units, carries out static embellishment, and associates the produced executable code with the simulation core to generate an executable simulation scenario. The

Vivado simulation command xsim loads a simulation scenario to achieve a batch mode simulation, a graphical user interface, or Tcl-based interactive simulation environment. Vivado Integrated Design Environment (IDE) is an "interactive design-editing environment that provides the simulator a user-interface and common waveform viewer."

CR channel selection could be statured as a multiple attribute decision-making problem (MADM). For a MADM problem, if there are 'm' alternatives and 'n' attributes, the i'th alternative could be expressed as $Y_i = (y_{i1}, y_{i2}, \dots y_{in})$. Here alternatives are referred to as frequency bands and attributes are the QoS parameters. The comparability sequence is given by $X_i = (x_{i1}, x_{i2}, \dots x_{in})$. The grey relational coefficient is calculated by [19],

$$\gamma(x_{0j}, x_{ij}) = \frac{\Delta_{\min} + \varsigma\Delta_{\max}}{\Delta_{ij} + \varsigma\Delta_{\max}}. \tag{8}$$

Here $\Delta_{\min}$ represents the minimum-better option. This applies to parameters, such as the delay in which should be at a minimum. $\Delta_{\max}$ represents the maximum better option. Channel capacity is considered to be maximum. ς represents the co-efficient value that is used for normalizing the values (0 to 1). Δ_{ij} is the difference between one attribute and another. The performance of GRA yields improved outcomes in relation to other MADM methods. The GRA method converges rapidly compared to other methods. From Figs. 31 and 32, shown that the computational complexity of GRA is less compared to the traditional MADM method. On the other hand, GRA converges at the optimal time, whereas MADM requires more iteration for channel selection. In Figs. 31 and 32, time is shown in terms of milliseconds.

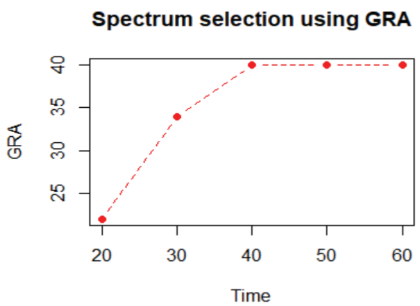

Fig. 31. Channel selection using GRA.

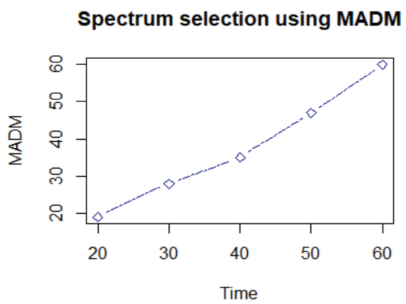

Fig. 32. Channel selection using MADM.

7. Conclusion

FPGA mobile sensor nodes that are combined with cognitive radio amplify the communication channel in disaster regions. Cognitive radio-based FPGA sensor networks increase the scalability of nodes and interruptions in the communications are discarded extensively. This can be achieved with the help of spectrum sensing and effective channel utilization is established by cognitive radios. For the effective usage of CR's, the Handoff algorithm is used. Several algorithms are available for Handoff. MADM methods are characterized for decision-making. The GRA method illustrated significant results compared to the traditional method in terms of channel selection. As the versatility and consistency of FPGA enlarge, self-governing sensors can be embedded in FPGA to evaluate several dynamic parameters. This approach will considerably reduce power consumption and increase usability. It can be fully used for remote sensing applications. Sensor nodes will monitor the temperature, power, communication channel, etc. These mobile sensor nodes have better coverage capacity, enhanced energy efficiency, and outstanding channel capacity. The sensors connected to FPGA will detect the temperature and environmental conditions in close proximity. It will send an alert signal so that the evacuation process can proceed rapidly. This will help to save numerous lives. The planned architecture and layered work will support hardware engineers and system architects by supplying supple and proficient real-time observation and a control design for huge and complex embedded sensor networks and remote-sensing applications.

- FPGAs have a high amount of multipliers and internal memory. As such they are highly suitable for signal processing systems. Thus we can find them in hardware that performs signal conditioning and muxing/demuxing, e.g., wireless networking gear, such as base stations.
- The smallest logic element in an FPGA is called a logic block, which is an ALU+flip-flop at its minimum. Therefore, FPGAs are used extensively for computing problems that can benefit from SIMD types of architectures, e.g. cleaning up images being received from an image sensor, point, or local processing of image pixels, e.g. computing difference vectors in H.264 compression.
- Finally, ASIC emulation, or hardware/software in loop testing, etc. is another application. Logic design for FPGA shares the same flow and tools as ASIC design. As such, FPGAs are also used to verify some test cases during ASIC development where the interactions between hardware and software may be too complicated or time-consuming to model.

References

[1] H. Kaur, R. S. Sawhney, and N. Komal, "Wireless sensor networks for disaster management," *International Journal of Advanced Research in Computer Engineering & Technology*, vol. 1, no. 5, pp. 2278-1323, 2012.

[2] K. A. Kumar, "FPGA-ARM implementation of an intelligent mobile ad-Hoc network," in *Proceedings of 2017 International Conference on Networks & Advances in Computational Technologies (NetACT)*, Thiruvananthapuram, India, 2017, pp. 111-115.

[3] V. G. Menon, J. P. Pathrose, and J. Priya, "Ensuring reliable communication in disaster recovery operations with reliable routing technique," *Mobile Information Systems*, vol. 2016, article no. 9141329, 2016. https://doi.org/10.1155/2016/9141329

[4] G. Jayakumar and G. Gopinath, "Ad hoc mobile wireless networks routing protocols: a review," *Journal of Computer Science*, vol. 3, no. 8, pp. 574-582, 2007.

[5] B. Manikiam, "Remote sensing applications in disaster management," *Mausam*, vol. 54, no. 1, pp. 173-182, 2003.

[6] S. Abba and J. A. Lee, "An autonomous self-aware and adaptive fault tolerant routing technique for wireless sensor networks," *Sensors*, vol. 15, no. 8, pp. 20316-20354, 2015.

[7] S. M. George, W. Zhou, H. Chenji, M. Won, Y. O. Lee, A. Pazarloglou, R. Stoleru, and P. Barooah, "DistressNet: a wireless ad hoc and sensor network architecture for situation management in disaster response," *IEEE Communications Magazine*, vol. 48, no. 3, pp. 128-136, 2010.

[8] N. Pratas, N. Marchetti, N. R. Prasad, A. Rodrigues, and R. Prasad, "Decentralized cooperative spectrum sensing for ad-hoc disaster relief network clusters," in *Proceedings of the 71st IEEE Vehicular Technology Conference*, Taipei, Taiwan, 2010.

[9] H. H. Choi, H. Lee, S. Kim, J. R. Lee, and I. H. Lee, "Distributed medium access control protocol based on successive collision detection for dense wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 12, 2016. https://doi.org/10.1177/1550147716664238

[10] J. Lotze, S. A. Fahmy, J. Noguera, B. Ozgul, L. Doyle, and R. Esser, "Development framework for implementing FPGA-based cognitive network nodes," in *Proceedings of 2009 IEEE Global Telecommunications Conference (GLOBECOM)*, Honolulu, HI, 2009, pp. 1-7.

[11] A. Meissner, T. Luckenbach, T. Risse, T. Kirste, and H. Kirchner, "Design challenges for an integrated disaster management communication and information system," in *Proceedings of the 1st IEEE Workshop on Disaster Recovery Networks (DIREN)*, New York, NY, 2002, pp. 1-7.

[12] J. J. L. Franco, E. Boemo, E. Castillo, and L. Parrilla, "Ring oscillators as thermal sensors in FPGAs: experiments in low voltage," in *Proceedings of 2010 VI Southern Programmable Logic Conference (SPL)*, Ipojuca, Brazil, 2010, pp. 133-137.

[13] O. Oballe-Peinado, F. Vidal-Verdu, J. A. Sanchez-Duran, J. Castellanos-Ramos, and A. Hidalgo-Lopez, "Smart capture modules for direct sensor-to-FPGA interfaces," *Sensors*, vol. 15, no. 12, pp. 31762-31780, 2015.

[14] S. Abba and J. A. Lee, "FPGA-based design of an intelligent on-chip sensor network monitoring and control using dynamically reconfigurable autonomous sensor agents," *International Journal of Distributed Sensor Networks*, vol. 12, 2016. https://doi.org/10.1155/2016/4246596

[15] M. D. R. Perera, R. G. Meegama, and M. K. Jayananda, "FPGA based single chip solution with 1-wire protocol for the design of smart sensor nodes," *Journal of Sensors*, vol. 2014, article no. 125874, 2014. https://doi.org/10.1155/2014/125874

[16] A. Okba, D. Henry, A. Takacs, and H. Aubert, "Autonomous RFID sensor node using a single ISM band for both wireless power transfer and data communication," *Sensors*, vol. 19, no. 15, article no. 3330, 2019. https://doi.org/10.3390/s19153330

[17] G. A. Preethi and C. Chandrasekar, "Seamless mobility of heterogeneous networks based on Markov decision process," *Journal of Information Processing Systems*, vol. 11, no. 4, pp. 616-629, 2015.

[18] K. Kumar, A. Prakash, and R. Tripathi, "Spectrum handoff in cognitive radio networks: a classification and comprehensive survey," *Journal of Network and Computer Applications*, vol. 61, pp. 161-188, 2016.

[19] U. Caydas and A. Hascalik, "Use of the grey relational analysis to determine optimum laser cutting parameters with multi-performance characteristics," *Optics & Laser Technology*, vol. 40, no. 7, pp. 987-994, 2008.

[20] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257-286, 1989.

[21] A. J. Viterbi and J. K. Omura, *Principles of Digital Communications and Coding*. New York, NY: McGraw-Hill, 1979.

G. A. Preethi Ananthachari **https://orcid.org/0000-0001-5740-1006**

She received her Ph.D. from Periyar University, Salem, TN, India. Currently, she is working as an assistant professor in the Technology Studies department, Endicott College of International Studies, Woosong University, Daejeon, South Korea. She has been awarded a distinction in her M.Phil. (Computer Science). She worked as an Assistant Professor at VIT Bhopal University, MP, India. She worked as a guest lecturer in the Department of Computer Science at Periyar University, Salem for a year from 2011-2012. She joined as a Project Fellow in the same department under the UGC Major Project in 2012. Her research interests are mobile computing, wireless technology, data analytics, machine learning, and robotics. She received the best paper award in an International Conference on Advances in Engineering and Technology (ICAET) held in E.G.S. Pillai Engineering College, Nagapattinam, Tamil Nadu, India.