DEVELOPMENT OF A MULTI-MODEL SENSOR NODE WITH DATA RECOVERY ABILITY TO MONITOR ENVIRONMENTAL FACTORS WHICH COULD BE USED FOR FOREST FIRES DETECTION

T. P. D. Pieris

198103P

Master's Degree

Department of Mechanical Engineering
Faculty of Engineering

University of Moratuwa
Sri Lanka

February 2023

DEVELOPMENT OF A MULTI-MODEL SENSOR NODE WITH DATA RECOVERY ABILITY TO MONITOR ENVIRONMENTAL FACTORS WHICH COULD BE USED FOR FOREST FIRES DETECTION

T. P. D. Pieris

198103P

Thesis submitted in partial fulfillment of the requirements for the degree
Master's Degree

Department of Mechanical Engineering
Faculty of Engineering

University of Moratuwa
Sri Lanka

February 2023

DECLARATION

I declare that this is my own work and this Thesis does not incorporate without acknowledgement any material previously submitted for a Degree or Diploma in any other University or Institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text. I retain the right to use this content in whole or part in future works (such as articles or books).

Signature: UOM Verified Signature Date: 30/04/2024

The supervisor should certify the Thesis with the following declaration.

The above candidate has carried out research for the Master's Degree Thesis under my supervision. I confirm that the declaration made above by the student is true and correct.

Name of Supervisor: K V D S Chathuranga

Signature of the Supervisor: UOM Verified Signature Date: 01/05/2024

DEDICATION

To my loving parents, your unwavering support and boundless love have been the driving force behind my journey, and I dedicate this to you with deep gratitude and profound appreciation.

ACKNOWLEDGEMENT

I am profoundly grateful to my supervisor, Dr. Damith Chathuranga, for the unwavering support, guidance, encouragement, and patience he extended throughout this research endeavor. Beyond being a conventional supervisor, he assumed the role of a mentor, comprehending my strengths and offering the necessary guidance and advice that played a pivotal role in the success of this research. I hold a deep appreciation for his invaluable dedication of time and expertise to my growth and development.

I extend my heartfelt gratitude to Dr. Asitha Kulasekera, whose unwavering support and expert insights have been instrumental in the realization of this research. Without his invaluable guidance, this project would not have come to fruition. Furthermore, I would like to express my appreciation to my progress review panel Dr. Thilina Lalitharatne and Prof. R. A. R. C. Gopura, whose invaluable inputs and constructive comments propelled me in the right direction throughout the research journey. Special mention goes to the dedicated members of the Computational Sensing and Smart Machines Laboratory, Mr. Chanaka, Mr. Ransimal, and Mr. Lihini, whose invaluable assistance has contributed significantly to my progress.

I wish to acknowledge the generous funding provided by Indo-Sri Lanka Joint Research Programme, MSTR/TRD/AGR/3/02/12 and Senate Research Committee Grant, SRC/TP/2017/14, University of Moratuwa without which the realization of this research would not have been possible. My heartfelt appreciation goes out to them for their crucial financial support, which has played a pivotal role in the success of this endeavor. Additionally, I am deeply grateful to the dedicated individuals in the laboratories and workshops of the Department of Mechanical Engineering, University of Moratuwa, whose unwavering commitment and selfless service have been invaluable throughout this research. Their willingness to assist, without hesitation, has been instrumental in the smooth progression of this project.

Lastly, I want to express my profound gratitude to my parents and my brother, whose unwavering belief in me has been an unwavering source of strength and encouragement. Their unwavering faith in my abilities has been a constant driving force, and I am forever indebted to them for their unwavering support.

ABSTRACT

Forest fires induce significant threats to ecosystems, human lives, and the climate. Human activities, fuel accumulation, and climate change are the primary factors contributing to their occurrence. To effectively manage and prevent the spreading of forest fires, it is essential to take support from improved forest fire monitoring systems. Existing fire detection systems have limitations, including high costs, limited coverage, and false alarms. Wireless Sensor Networks offer a promising solution due to their ability to cover large areas, provide rapid response, and collect relevant ambient data. This research proposes the design and development of a multi-model sensor node for forest fire detection. The sensor node integrates temperature and humidity sensors, along with a communication system and data recovery method. An artificial intelligence-based algorithm is employed for fire detection. The performance of the sensor node detection algorithm is evaluated through experiments. Also, the system life, data recovery ability, and packet loss of the sensor node are also evaluated experimentally. As the results of these evaluations, there is a system life of more than 4 years, excellent recovery ability for sensor module failure, and promising fire detection ability for a 30 cm x 30 cm wide artificial fire with 2 Kg of forest fire fuel compared to the existing systems giving better deployment ability.

Keywords: forest fire detection, wireless sensor nodes, artificial intelligence, multi-AI

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

Forest fires occur around the world every year and they have a significant impact on ecosystems, human life, and the climate. There are many primary factors contributing to the occurrence of forest fires. One major cause is climate change [1]. As a result of this climate change, frequent droughts, higher temperatures, and altered precipitation patterns occur. There is a significant impact on an increased risk of fire-prone conditions in many regions. Also, recent studies have indicated that there is a clear link between anthropogenic climate change and the escalation of forest fires [2]. These findings highlight the urgent need to address climate change mitigation and adaptation strategies to effectively manage and prevent forest fires.

Another factor influencing forest fire occurrence is the accumulation of combustible materials, such as dry vegetation and deadwood, in forests. Forest management practices and changes in land manipulation have resulted in the collection of fuels through the past. This fuel build-up creates sufficient conditions for fire ignition and a high risk of fire spreading situation. Recent studies have stated the importance of fuel management strategies, such as burning and fuel reduction treatments, in reducing forest fire risk [3]. But in practical situations, this is not useful for large areas due to high cost, risks of implementation, and not efficiency.

The third major cause of forest fires is human activities, both intentional and unintentional. Campfires, fireworks, and equipment sparks are the major reasons for a significant amount of forest fires globally. Also, human-related factors are a major impact on propagating wildfires. For instance, research conducted by Balch et al. emphasized that humans were responsible for igniting most wildfires in the United States [4]. Moreover, socioeconomic factors, such as population growth, urban expansion, and agricultural practices, can increase human-caused fires. Therefore, there is a need for improved monitoring systems of forest fires.

Due to the causes of forest fires, there are many ways of forest fire detection. They are human observations [5], satellite based methods ,[6] optical camera based methods [7], wireless sensor nodes, animals based methods [8] and unmanned areal vehicles based methods [9]. But in these existing fire detection systems have some major drawbacks [10], [11]. Human observations are not practical for large area of coverage. Satellite-based systems have longer response time and cannot cover the total area at once and are not cost-effective. Camera-based systems can only respond to line-of-sight fires, which may be limited due to trees, rocks, mountains, etc. Covering a large area using optical camera systems requires a longer setup time, is more complex and expensive, and is difficult to monitor. Cameras can also cause false alarms due to rapid changes in ambient lighting caused by changes in weather. Additionally, cameras require mounting on a tower, which increases the cost of the system. As mentioned above, the other methods have the many drawbacks.

WSN-based solutions have shown promise in providing a better solution for forest fire detection [10], [11]. These WSN-based systems can easily cover a large area, respond rapidly, can be easily customized to collect the relevant ambient data, do not need specialized mounting platforms, and hence are cheaper compared to camera-based solutions. WSNs can be easily mounted in unreachable places compared to other contemporary methods.

An effective data sampling strategy is the result of the properly designed and developed sensor node. The sensor nodes have major primary subunits as follows [10]. The primary component of a sensor node is the sensing unit, which captures environmental data related to its application. Various sensors, such as temperature sensors, humidity sensors, smoke detectors, and air quality sensors, can be integrated into the sensor node to collect relevant information. These sensors enable the continuous monitoring of the environment. The sensor nodes consist a microcontroller to manage all the tasks in the sensor nodes such as measuring data, sending data, etc. There is a power supply is the most important component that ensures the uninterrupted operation of the sensor node. The choice of power sources, such as batteries or energy harvesting techniques, must consider the longevity and reliability of the sensor node in remote forested areas. But if the sampling rate of the sensor node is very low, the sensor node can work few years without an energy harvesting unit.

In the communication methods of sensor nodes, several wireless technologies are used for efficient data transmissions. The sensor node can employ wireless communication protocols, such as GSM, RF, Wi-Fi, LoRa, etc. to transmit collected data to a base station or a network of interconnected nodes. These communication methods enable data sharing and command sharing [12]. In the forest environment, the sensor node's communication range and reliability need to be optimized to ensure seamless connectivity in dense vegetation and remote locations.

Wireless Sensor Nodes (WSNs) are widely used in various fields such as climate monitoring, military applications, health monitoring, building structure monitoring [13], air quality monitoring [14], and agriculture [15]. In forest fire detection, WSNs have emerged as a promising solution over human-based observations, satellite-based systems, and optical camera-based systems [10]. In these applications, the sensor node's ability and requirements are different from each other. As the different abilities of the sensor nodes, the sensors of the climate monitoring sensor nodes are different from building structure monitoring sensors. As per the requirements, the power supply of the sensor nodes for the building structure monitoring can be powered by the building for continuous operation but the air quality sensor nodes are remote sensors and they should be powered by batteries for continuous applications for a long time.

Various wireless sensor networks based forest fire detection systems have been introduced previously. Commonly used systems are direct data thresholding [9, 16, 17] and the 30-30-30 rule [12], which compare predetermined data values for detection. Another approach is early fire detection [18], where WSN data is fed into a predefined mathematical model. Neural network (NN) based algorithms, such as convolutional neural networks (CNN) and deep neural networks (DNN) [19], as well as multiple artificial intelligence methods [19], have also been developed for fire detection using WSN data.

In the existing thresholding methods provide limited information on fire behavior patterns and exhibit low stability due to simplistic threshold comparisons [19]. Early fire detection methods require significant time for environment modeling and prediction [18, 20, 21]. Although complex neural networks like CNN have been used for fire detection [19], they suffer from high detection delays and computational resource demands. To overcome these limitations, multiple AI methods employing multiple neural networks have emerged, offering improved stability and reduced detection delays [19].

1.1 Research gap

The challenge is to use a minimum number of sensors to minimize node complexity, allowing the large-scale deployment of nodes across a large forest area for effective forest fire detection. This research proposes the use of a WSN system that combines a less complex multisensor node with an AI-based detection algorithm for forest fire detection.

1.2 Objective of the Thesis

- Design and develop a multi-model sensor node to detect environmental conditions such as temperature, humidity, smoke, etc. with a communication system and data recovery method.

- Evaluate the performance of the sensor node for detecting artificially created forest fires.

A multi-model sensor node to detect environmental conditions such as temperature and humidity with a communication system and data recovery method has been designed and developed for the node recovery. Data have been collected creating artificial forest fires by the sensor nodes and verified that the developed sensor nodes can detect forest fires. To detect forest fires artificial intelligence-based algorithm has been used.

1.3 Thesis Overview

This thesis consists of seven chapters and their contents have been summarized as follows. The first chapter is an introduction that describes an overview of this research. The second chapter is the literature survey which describes the existing methods of forest fire detection, why wireless sensor nodes are important for forest fire detection, existing methods of forest fire detection by wireless sensor nodes, the best method for forest fire detection, and the construction of the sensor nodes. The third chapter methodology describes how the sensor node has been designed and developed. Conducted experiments and their experiment setups have been mentioned in chapter four. Based on the conducted experiments, validation of the forest fire detection ability of the sensor node has been mentioned in the results. Then the discussion chapter describes the current progress and comparison of the proposed solution with a comparison of existing methods. The last chapter describes the conclusion of this research and the future works.

CHAPTER 2

LITERATURE SURVEY

2.1 Existing Methods of Forest Fire Detection

The common forest fire detection methods are the human-based observations [5], satellite-based systems [6], optical camera-based systems [7], unmanned areal vehicle (UAV) [9], animals based mobile sensing systems [8], radio acoustic-based systems [22] and wireless sensor nodes (WSN) and their networks [10]. In this research, the WSNs have been used to detect fires but the advantages of the WSNs over other methods have been discussed as follows.

To compare each method of fire detection several parameters have been suggested by several review papers [10] & [11]. They are cost, practicality, false alarms, fire localizing accuracy, detection delay, fire behavior information, and flexibility to adapt for an ability to integrate with another application. The best fire detection method should be low cost, practical, low false alarm rate, high fire localizing accuracy, low detection delay, and high fire behavior information. As an additional ability, good flexibility to adapt another system.

In human-based observations have these properties. As the positive aspects, these systems have low cost and low false alarms. But low practicality, low fire localizing, high detection delay, and no fire behavior information are the negative aspects of this method. In this method, the watch towers should be accommodated in the forests to stay a person. This is the most traditional way of forest fire detection.

The positive aspect of the satellite-based methods is the good fire behavioral information. But the availability of medium-level false alarms and practicality, high cost, and detection delay are some negative aspects of this method. An additional advantage is this has the flexibility to adapt to another system.

As the above details, all the parameters have been summarized in Table 2.1 as a comparison.

According to the above comparison, the sensor nodes are the best method of forest fire detection considering the high fire good practicality, localizing accuracy, low detection delay, high fire behavior information, and medium cost. As an additional advantage, WSNs can be used for other applications such as precision agriculture, weather monitoring, etc.

2.2 Wireless Sensor Nodes

The function of a sensor node is to detect, process, and periodically transmit sensor data to another location. The sensor node architecture should include a sensing unit,

TABLE 2.1: COMPARISON OF THE EXITING FIRE DETECTION METHODS

Parameter	Humans	Satellites	Cameras	WSNs	Animals	UAV
Cost	Low	Very high	High	Medium	High	High
Practicality	Low	Low	Medium	High	Very Low	High
Faulty alarms	Low	Low	Medium	Medium	High	Medium
Fire localising accuracy	Low	Medium	Medium	High	High	High
Detection delay	Long	Very long	Long	Small	Long	Long
Fire behaviour information	—	Yes	—	Yes	Yes	Yes
Can be used for other purposes	No	Yes	No	Yes	No	Yes

a processing unit, a transceiver, and a power unit to enable these functions [23], [24] (Figure 3.1). The sensors are the primary subunit in this architecture and are typically connected to the processor through an analog-to-digital converter (ADC) to convert analog data into digital signals [25]. The transceiver unit is responsible for establishing a connection between the sensor node and the network, enabling data transmission and receipt of acknowledgment instructions. The processing unit manages all tasks, while the power unit supplies power to all subunits. Some sensor nodes operate using non-rechargeable cells [26], allowing them to operate for extended periods with high wake-up times, while others use rechargeable batteries and energy harvesting methods such as solar cells [27]. However, sensor nodes must consume minimal power to operate for extended periods [24]. The processor of the sensor nodes may be a microcontroller, a field-programmable gate array (FPGA) for high processing capabilities such as vision, a digital signal processor (DSP) for high data processing capabilities, or a hybrid architecture [28] to meet specific requirements with low power consumption and high processing capabilities [23].

2.2.1 Sensor Networks

A wireless sensor network comprises numerous sensor nodes connected by wireless links, with the primary components consisting of the sensor field and base station [29]. The sensor field denotes the vast assemblage of deployed sensor nodes, while the base station constitutes a centralized node that manages the network, collects information from the sensor nodes, and disseminates commands to the sensor nodes. Access to the base station is possible through satellite or internet communication for the purpose of data retrieval.

2.2.2 Energy Sources for the Wireless Sensor Nodes and their Configurations

Energy harvesting techniques such as solar energy, wind, and RF power capture can be utilized to recharge batteries, provided that appropriate rechargeable battery systems are employed. Also, due to the charging circuits, the cost of the sensor nodes will be advanced and the circuit will be more complex. For a large-scale deployment of sensor nodes such as forest fire detection, this additional cost will be highly affected. For wireless sensor networks that operate in harsh environments, including underground [30], underwater, or within building walls [31], non-rechargeable batteries are preferred [26]. This approach is only feasible when the sensor nodes have a very low-duty cycle. Environmental monitoring applications that require low-duty cycles can benefit from non-rechargeable batteries [32]. Effective power management techniques are crucial when using non-rechargeable batteries for wireless sensor networks [26], [33]. Among the power management techniques for wireless sensor nodes, avoiding voltage regulators, using power gating, and power matching are few of them [26].

2.3 Existing Methods of Fire Detection with Wireless Sensor Nodes (WSN)

The use of several algorithms based on data from WSNs for fire detection has been presented in recent literature. The commonly used methods are the direct data thresholding [9, 16, 17] and the 30-30-30 rule [12]. Predetermined data values are obtained for both algorithms to obtain a detection. Early fire detection [18] is also another method where it inputs the data from WSNs from a predetermined mathematical model for the final decision. Pre-collected data are used for the training of neural network (NN) [19] based algorithms. Convolutional neural networks (CNN), deep neural networks (DNN), etc. [19] and multiple AI methods [19] are fire detection-based NN.

The existing thresholding methods as described in references [9, 16, 17] and [12] provide limited information on the pattern of fire behavior, since they merely compare the measured data with a pre-set threshold value, resulting in low stability [19]. Additionally, early fire detection methods require a significant period of time to model the environment and predict forest fires [18, 20, 21]. While complex neural networks, such as convolutional neural networks, have been developed for fire detection [19], they suffer from high detection delays and require substantial computational resources for implementation. Nevertheless, these techniques still demonstrate greater stability than thresholding methods [19]. To overcome these limitations, multiple AI methods employing more than one neural network have been developed for fire detection, yielding a promising approach with improved stability and reduced detection delays [19]. Thus, the multi-AI approach represents an innovative and effective solution for enhancing fire detection accuracy. The above-mentioned methods have been summarized in Table 2.2.

TABLE 2.2: COMPARISON OF RECENT WORK ON EXISTING FIRE DETECTION ALGORITHMS WITH WSNS

Method	**Special Notes**	**Delay for fire detection [19]**	**Is it stable? [19]**
Thresholding data (temperature, humidity, etc.) [9, 16, 17]and[12]	Simple to implement	High	No
30-30-30 rule [12]	Used in Spain, such as thresholding, simple	-	-
Early fire detection [18, 20, 21]	Used in USA, Canada and Korea, complex	-	-
Fuzzy logic [34]	A simple fuzzy algorithm is used	-	-
Neural networks (CNN, DNN, etc.) [19]	Complex, need separate resources	CNN: High	CNN: Yes
Multiple AI methods [19]	Complex, need separate resources	Low	Yes

Park et al. [19] proposed a novel multi-AI-based algorithm for fire detection, which utilizes various sources of input such as image data obtained through cameras, temperature data, humidity data, etc. In this approach, a convolutional neural network is employed to analyze the image data, while a deep neural network is used to analyze the sensor data. The pre-processed input data are then fed to the deep neural networks instead of the raw sensor data. The output from the neural networks is then fed to an adaptive fuzzy module, which adjusts its fuzzy factors based on the neural network output. The deep neural networks consist of an input processing block, hidden layers, and output data.

However, in forest fire detection, the deployment of large-scale sensor nodes is often impractical in wilderness areas. Furthermore, the sensor nodes should be simple to reduce power consumption and facilitate deployment. If cameras are included in the sensor nodes, they consume a considerable amount of power, and deployment becomes much more complex. Hence, intelligent detection algorithms should be employed in conjunction with simple sensor nodes to provide an effective forest fire detection solution.

The proposed system by Park et al.[19] has temperature, humidity, smoke sensors, and video feed by cameras as the input data. For a simpler design, temperature, humidity, and smoke data are useful without the cameras.

But, recent literature mentioned that smoke sensors have several drawbacks in outdoor environments such as less ability to cover wide areas and low response time [35], [36]. Also, outdoor smoke detection is more challenging due to rapid changes in weather conditions, fog, and low illumination [36]. For a rapid significant difference of data in smoke sensors, the fire conditions should be at enough level [37]. The solution for the above smoke sensor drawback is using image processing-based smoke detection [35], [37]. But the integration of a camera with a wireless sensor node is a more challenging task due to the increase in node complexity and high power consumption. Therefore, a smoke sensor-free design is a reliable solution to detect outdoor fires, then, the sensor node will be less complex than proposed. In this research, temperature, and humidity data are used for fire detection the results also confirmed that the proposed node can detect outdoor forest fires [38].

In recent literature, forest fire detection with sensor nodes and machine learning algorithms has been studied [39]. The used technologies such as processors, sensors, and communication have been summarized as follows.

TABLE 2.3: COMPARISON OF RECENT WORK ON EXISTING FIRE DETECTION SENSOR NODE TECHNOLOGIES

Recent work	Processor	Sensors for temperature and humidity measurement	Communication
[40]	LoRa32u4	BME280	LoRa
[41]	Arduino UNO	DHT11	WIFI
[42]	Arduino nano	DHT22	GSM/RF

The sensors BME280, DHT11, and DHT22 have been compared as follows to select the suitable sensor. The communication methods LoRa, WiFi, GSM and RF have been compared for the selection of suitable methods.

As the above comparison in Table 2.4, the DHT22 sensor has better range and resolution but there is no validation or recommendation for outdoor environmental monitoring. Therefore, the BME280 sensor module has been selected for the temperature and humidity measurement due to the recommendation and validation of the performance as meteorological equipment [43].

TABLE 2.4: COMPARISON OF SENSORS

Sensor	Power Consumption	Capability of environment monitoring	Performance as meteorological equipment	Temperature range (°C)	Humidity range (%)	Accuracy (+/-)
BME280	Low	Yes [43]	Yes [43]	-40 to 85	0-100	1 °C & 3 %
DHT11	Low	No	No	0-50	20-90	2 °C & 5 %
DHT22	Low	No	No	-40 to 80	0-100	0.5 °C & 2 %

Arduino platform is also compatible with machine learning-based forest fire detection according to Table 2.3. Therefore, as the processor of the sensor node ATmega328P micro-controller has been used.

TABLE 2.5: COMPARISON OF COMMUNICATION METHODS

Communication method	Power Consumption	Range	Cost
LoRa	Low	High	High
WiFi	High	Low	High
GSM	High	High	High
RF	Low	Low	Low

According to the above comparison in Table 2.3, LoRa is the best communication method considering power consumption and cost. But considering the cost RF is the suitable solution. If the LoRa is used for all sensor node communications, the cost will be high. But the combination of RF and LoRa communication will be cost-effective. Therefore, a cluster-based network has been used. Sensor nodes send the data to a cluster head via RF communication. Then, the cluster head sends the data to the LoRa gateways, and the LoRa gateway sinks data via the internet to the base station where the data analysis happens. Due to RF communication, the cost can be reduced and due to the LoRa, the communication range can be improved.

CHAPTER 3

METHODOLOGY

3.1 Design and development of the sensor node

The proposed WSN consists of a low-power, high-performance general-purpose AVR 8-bit ATmega328P microcontroller, a BME280 sensor module for temperature, pressure, and humidity measurement, and a nRF24l01+ 2.4 GHz RF transceiver as the primary components. This WSN is battery-powered by two AAA alkaline batteries. The microcontroller features additional interfaces such as SPI, I2C, Serial communication ports, power ports, three analog ports, and five digital general-purpose input-output ports to accommodate other devices or sensors like pH monitors, carbon dioxide sensors, and moisture sensors, depending on the application requirements. Two LEDs are also provided for debugging purposes to indicate power and transmitter acknowledgment or another purpose of indication such as low battery. In continuous operation, the LEDs can be disabled using jumpers to conserve energy.

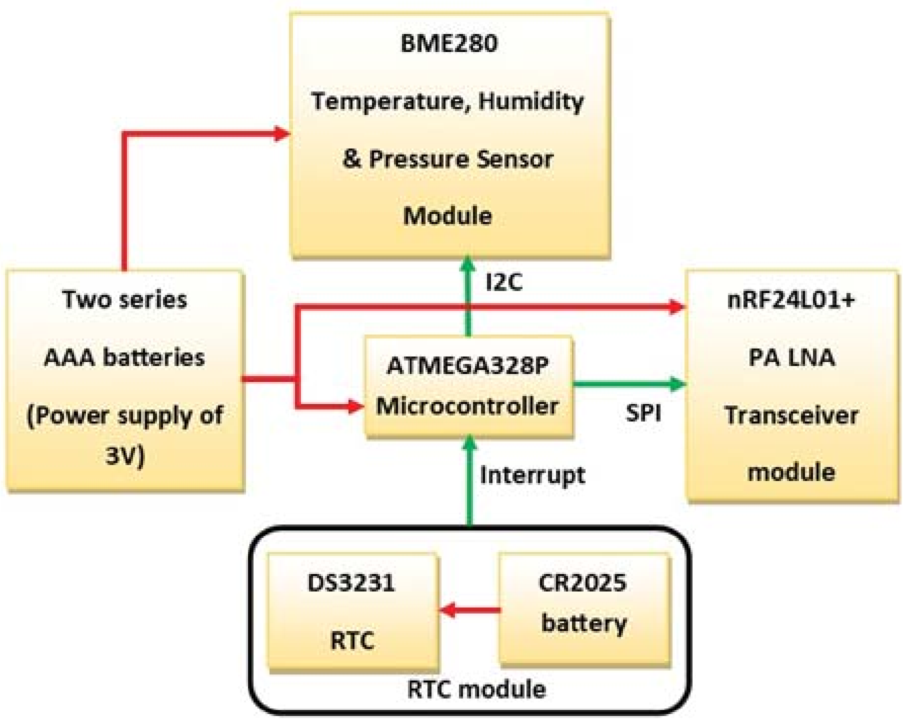

Fig. 3.1: Block diagram of the WSN

3.2 Components of the WSN and their configurations of the sensor node

3.2.1 Sensor Module: BME280

According to the parameters required for fire detection in the introduction and the literature survey, temperature and relative humidity should be studied. To measure these two parameters, suitable sensors should be used. The required sensors should have the ability to take data samples similar to the meteorological equipment's performance for better fire detection. The Bosch BME280 environment monitoring sensor module has been used. This module can measure temperature and relative humidity in the environment. This sensor has greater than 95 % of similar characteristics as the meteorological equipment's performance [43]. Also, This has been specially designed for weather monitoring and it has very low power consumption and is pre-calibrated.

This BME280 sensor module operates from 1.71 V to 3.6 V and has a built-in regulator and voltage reference for operation. This sensor can be used with primary cells, and according to the datasheet, it consumes less than 4 uA for all measurements and 0.1 uA sleep in sleep mode. The sensor offers three sensor modes for the data sampling and they are sleep mode, forced mode, and normal mode. In the sleep mode, no operation takes place, but all registers are accessible. In forced mode, the sensor performs one measurement, stores results, and returns to sleep mode. The sensor has perpetual cycling of measurements and inactive periods in normal mode. In this WSN implementation, the sensor modules have been configured to operate in the forced mode.

3.2.2 Micro-controller: ATmega328P

The function of the sensor node is wake up from sleep mode, sample the data, and send them to the required location. Therefore, a general-purpose microcontroller is sufficient for the to do these tasks. An AVR 8-bit ATmega328P microcontroller was selected for the proposed node. This is able to operate down to 1.8 V with a maximum frequency of 4 MHz in the safe operating region, as stated in the datasheet [44].

If the frequency is reduced below 4 MHz, power consumption can also be reduced further. To accomplish this, the fuse bytes in the microcontroller can be used. Three bytes of fuse bits are available and they are extended, high, and low. To reduce the clock speed, the configuration of the low fuse byte is necessary, and changing the related bits can achieve this. The bits were configured to select the internal 8 MHz oscillator and divide the clock by 8 as per the datasheet. The extended fuse bytes were set to disable brown-out detection, and the microcontroller can now operate at a 1 MHz clock speed. Disabling brown-out detection can save even more power, which was already turned off by the fuse byte settings. Thus, only the ADC is turned off while in power-down mode.

After sending the data, the microcontroller goes to sleep mode, but before the microcontroller enters sleep, the other modules are turned into sleep mode by the microcontroller, minimizing the power consumption of the whole sensor node further.

3.2.3 Real time clock (RTC) module: DS3231 RTC module

For waking up the node from power-down mode, a DS3231 RTC module with a few modifications is used. , The LED indicator and EEPROM of the module are removed to save power further. This module is powered by its own CR2032 coin cell battery. The alarm interrupt pin of the RTC module was connected to the microcontroller interrupt pins to wake up the microcontroller. The registers of the DS3231 were set to generate an interrupt when the alarm is activated, as per the datasheet.

3.2.4 Transceiver Module: nRF24L01 + PA LNA Transceiver Module

nRF24L01 + PA LNA Transceiver Module was used as the main transceiver module. This ability to communicate simultaneously with 6 transceivers without a routing scheme due to integrated "Enhanced ShockBurst" [45] proprietary communication module. Therefore, this is most suitable for star topology with a cluster-based sensor network which is used in this proposed WSN.

This transceiver operates in the worldwide 2.4 GHz ISM band and has ultra-low power operation, with 900 nA in power-down mode. It also has on-chip voltage regulation for a 1.9 V to 3.6 V supply range. The nRF24L01+ chip can work from 1.9V to 3.6 V, as mentioned in the datasheet [45], and the RFX2401C chip can work from 2.0 V to 3.6V.

The RF24 network library was used to establish the network used in the proposed sensor node with a tree topology [46]. This transceiver module operates at a low bit rate of 250 kbps to increase receiving sensitivity.

3.2.5 Power supply: Two AAA batteries

As mentioned in the above details all the components can operate between the 2 V – 3.6 V range. The operation voltage levels have been summarized in Table 3.1. The RTC module AAA alkaline batteries have 1.5 V – 1 V voltage reduction while discharging into its 95 % of full capacity. Therefore, two series AAA series batteries are able to provide 2 V – 3 V voltage range for the load using its full capacity. Two AAA batteries are used to power the sensor node. The RTC module has been powered by its own CR2025 battery therefore the main power supply power up the other components only.

TABLE 3.1: OPERATIONAL VOLTAGE RANGES OF COMPONENTS OF THE SENSOR NODE

Selected Component for the sensor node	Operational voltage range
ATMEGA328P microcontroller	1.8 V - 5.5 V
BME280 sensor module	BME280 sensor module
nRF24L01 + PA LNA Transceiver Module	2.0 V – 3.6 V

3.2.6 Enclosure of the sensor node

The designed and developed sensor node circuit has been mounted in a waterproof enclosure. The enclosure has a small opening to mount the sensor modules sensor. Figure 3.2 is the sensor node with the enclosure mounted at outdoor.

Fig. 3.2: Sensor node with enclosure

The proposed sensor node's architecture is shown in (Figure 3.1), and the fabricated sensor node is given in Figure 3.3.

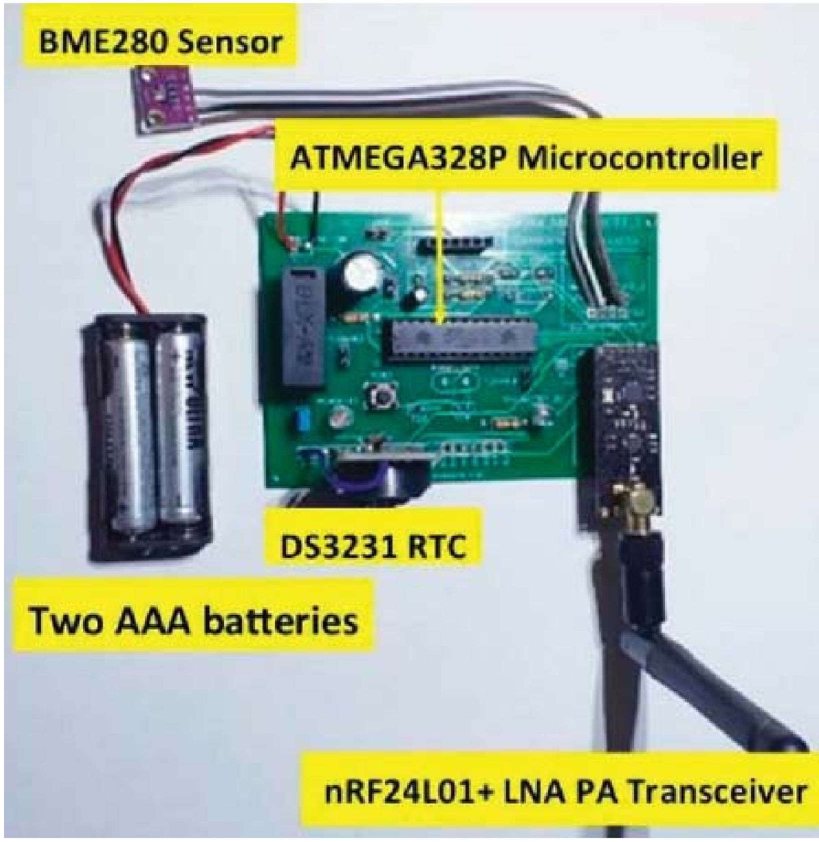

Fig. 3.3: Fabricated sensor node

3.3 Data recover ability unit of the sensor node

These sensor nodes are deployed in forest areas (remote areas) and under any malfunctions there is no convenient way of debugging them efficiently without reaching the malfunctioned sensor node. These malfunctioning situations can be subdivided into failures of subunits such as sensor modules, processors, transceivers, and power supplies. Under this kind of failure, if the sensor node can recover to its previous function, it is called the self-recovery ability of the sensor node.

Due to these subunit failures, various situations can occur as a result. They are tabulated in Table 3.2 as follows.

TABLE 3.2: RESULTANT FAILURE SITUATIONS UNDER SUB UNIT FAILURES

Subunit	Result of failures
Sensor module	No data or wrong data
Processor	Interrupt the functioning
Transceiver	Can not send data or receive commands
Power supply	Totally malfunction the node

To solve this failure situation the most convenient method is adding a functional spare sub-unit for functioning when the primary unit fails [47].

Under this solution, various disadvantages will be added to the sensor node such as the cost of the node will be double, the node will be fully duplicated in the same node, idle power consumption will be doubled, more additional parts will be needed to give the ability to recover the failures then another cost will be added. (MOSFETs, complex PCB, etc.) and Size will be double.

Avoiding these limitations as much as possible these data recoverability can be limited as follows. The cost of each subunit of the sensor node is available in Figure 3.4. The cost of the forest fire detection sensor node deployment is not more cost-effective than other application areas of the sensor nodes. Therefore the cost is one of the major parameters in this application. [47], [48] and [49] are several applications of sensor nodes but they have used dual component methods but all are limited due to a single sub-unit duplication.

In this method, the recovery ability is limited to the sensor module considering the cost, space, and power. The sensor module consumes very low power at idle compared to others. Now the sensor node consists of dual BME 280 sensor modules. If one fails, the other one will be activated by the firmware. Evaluation of this system is available in the Experiments section.

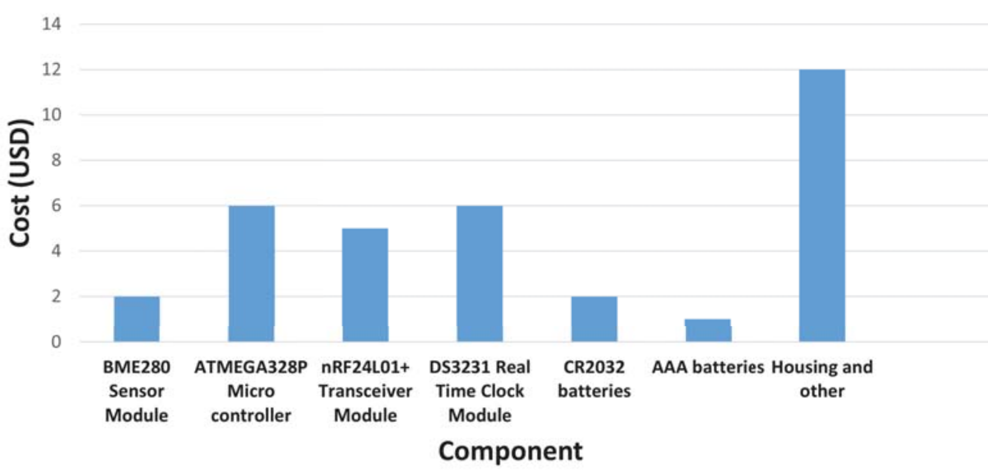

Fig. 3.4: The cost of each sub-unit of the sensor node

3.4 Function of the sensor node

The sensor node spends most of its time in sleep mode after activation. Once every five minutes, it wakes up to measure the temperature, relative humidity, and atmospheric pressure in its immediate surroundings. The gathered data is then transmitted as a data array to the receiver before the node returns to sleep mode. This cycle is repeated until the battery is drained. Department of Meteorology in Sri Lanka is taking observations for every 3 hours routine therefore the sampling rate of 5 minutes is most sufficient for the observation of data [50].

3.5 Deasign and Development of The Cluster Head

The network architecture of the proposed sensor node as mentioned in the literature review is a cluster-based network architecture. The cluster has the ability to take the data from many sensor nodes and send them into the base station.

The designed and developed cluster head is also a similar sensor node but has the ability to send the receiving data from other sensor nodes as well as the data taken from the sensor of the cluster to a base station via a Lo-Ra communication-based router. For a large-scale sensor network, this router should have the ability to send that via GSM GSM-based internet facility. For this proposed system, the base station is also another Lo-Ra-based receiver.

This needs more power to operate than a sensor node because the wake-up time is higher than the sleep time. The reason for the high wake-up time is the routing scheme. In the routing scheme, a specific time has been dedicated to the routing protocol for the sensor node to connect. In this way, through the duty cycle of the cluster, the sensor nodes can connect and send the data.

This is powered by a lithium-ion battery pack and a solar panel has been used to charge the battery pack due to high power demand. In this implementation, the routing scheme has been developed for 8 sensor nodes. The cluster head used is available in Figure 3.5.

3.6 Design and development of Artificial Intelligence (AI): Proposed multi-ai based architecture

The proposed AI algorithms are based on the multi-AI framework mentioned in [19] and [38]. It employs temperature and humidity data as input from sensor node data, using two shallow neural networks for simplicity of sensor node deployment in large-scale forest coverage. Each sensor node of the deployed sensor networks is used to find the possibility of the fire inference.

The proposed framework does not use image data or other sensor data, making it suitable for use with low-power and simple sensor nodes. In (Figure 3.6), the temperature and humidity data are collected and processed by a data processing unit to obtain the average and maximum of the buffered data.

Then, the shallow neural networks (Figure 3.7) receive the final-first difference (FFD), the final-maximum difference (FMD), and the final-average difference (FAD)

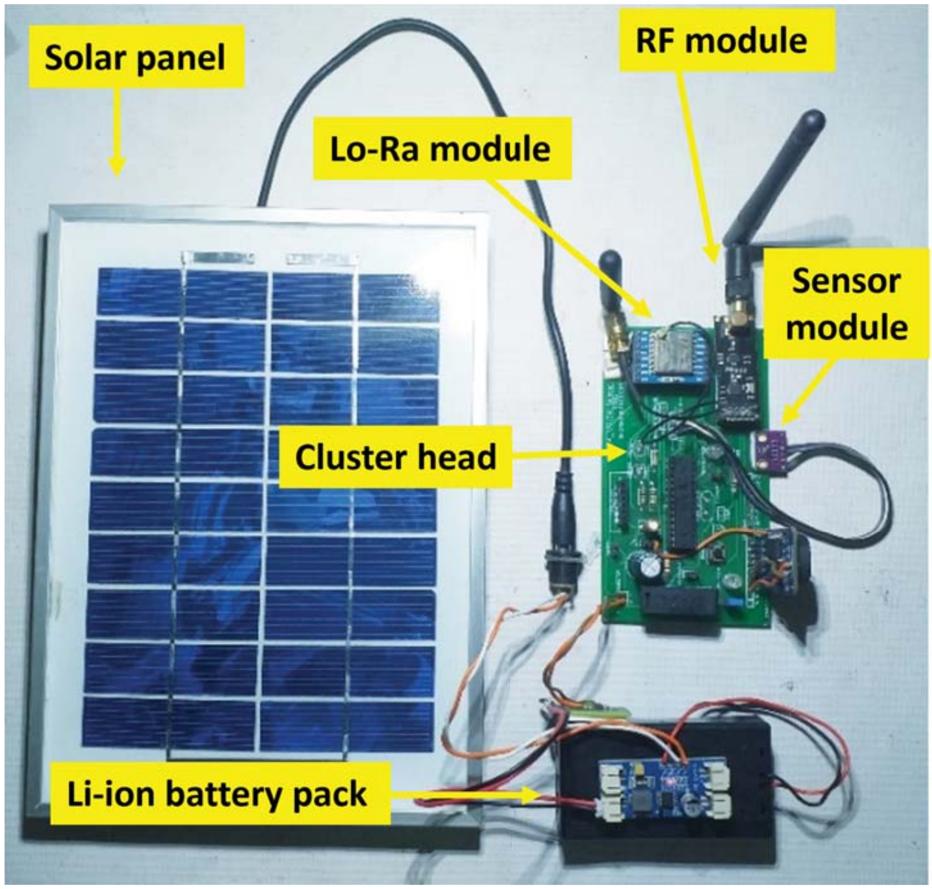

Fig. 3.5: Cluster Head

as input, with separate networks used for temperature and humidity data by the data processing unit. Each neural network has one hidden layer with 15 nodes (the number of nodes is determined by conducted experiments) and a single output that identifies the probability of fire occurrence between 0 and 1. These values are then passed through an output processing module and a fuzzy module. The output processing module cut off the values into the values 0 to 1. After that, these values are fed into the fuzzy module and finally, the fuzzy module produces the final probability of fire.

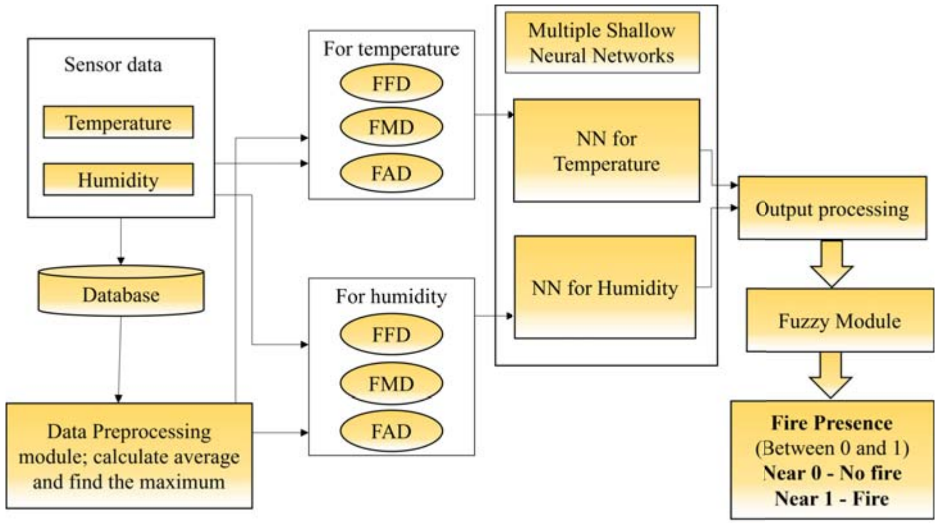

Fig. 3.6: Proposed multi-AI architecture

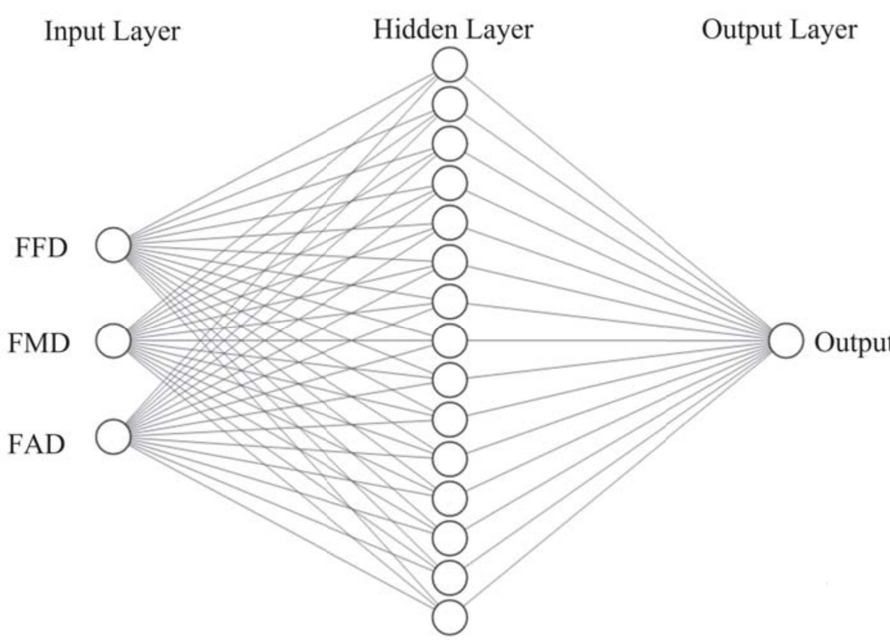

Fig. 3.7: The architecture of the deep neural networks

CHAPTER 4

EXPERIMENTS

4.1 Spatial Size Determination for Fire Detection

A fire can occur anywhere. Therefore the sensor nodes should be deployed (or placed) as they have the ability to detect a fire. The maximum dimension for detecting fire-related data is called spatial size. Spatial size is considered as a hypothetical cuboid (Figure 4.1) for placing the nodes to identify a significant change (more than the sensitivity of the sensors) of fire-related data (temperature and humidity). Nodes are placed on top corners of the spatial cuboid. As mentioned in Figure 4.1, the spatial size has the volume of s x s x h. Here d is the maximum distance between the sensor nodes. These d and h values should be determined to identify the a fire.

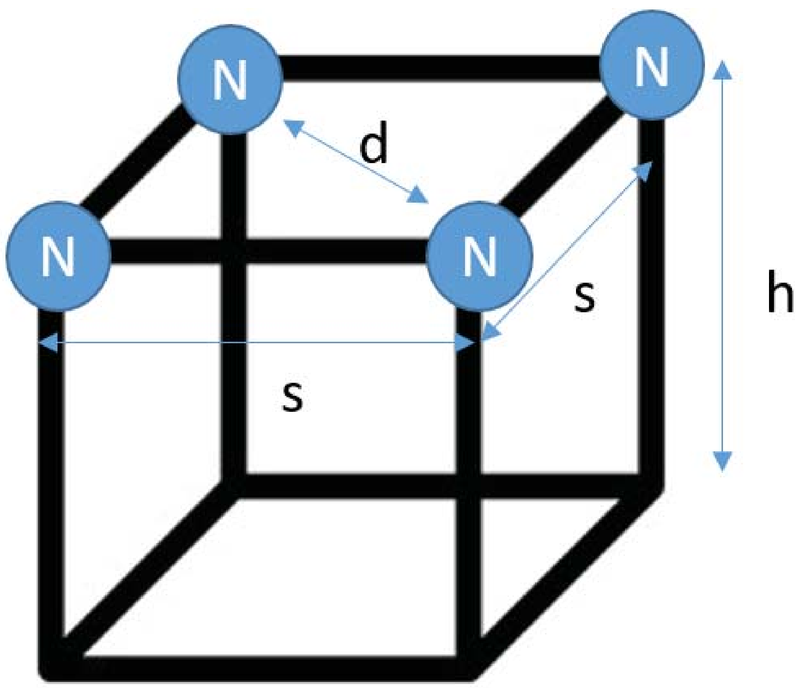

Fig. 4.1: Spatial Cuboid

The spatial size was determined using the procedure as follows. At first, two sensor nodes were placed in a straight line as in Figure 4.2, and a fire was created between them. The distance between the sensor nodes is d and the node height is h. Then, the temperature and humidity data were observed while a fire is occurring. The experiment was repeated for different distances between the two sensor nodes and node placement heights. The changes of received data were studied and the detectable spatial size was determined. The fire was created with 2 Kg of teak wood on a 30 cm x 30 cm area. Another sensor node was placed far away from the fire to determine the ambient conditions. The absolute temperature of the fire was measured by a thermocouple.

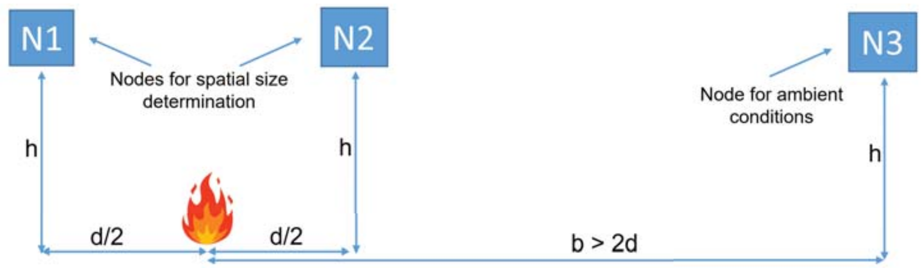

Fig. 4.2: Experiment setup for the determination of the spatial size

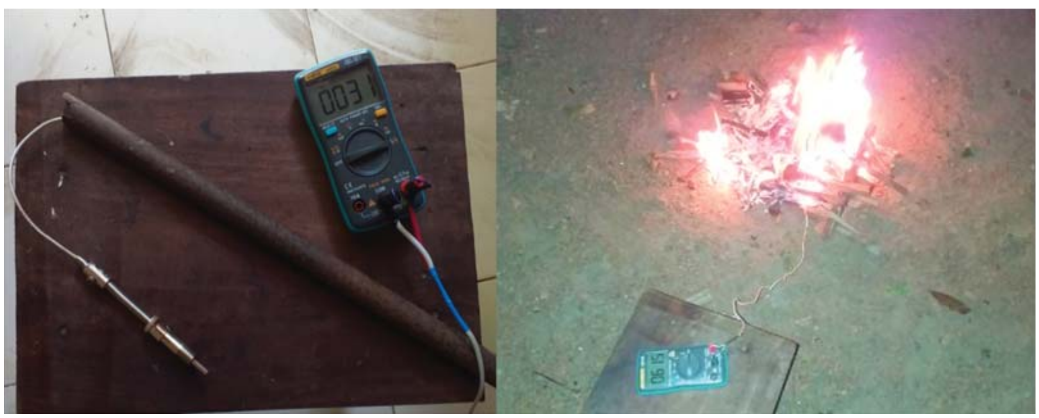

Fig. 4.3: Thermocouple for actual temperature measurement of the fire

4.2 Experiment setup to train the proposed AI's neural networks

For the training data set for fire detection, binary fire presence (1 for fire and 0 for not fire) and related dependant variables under a fire presence (temperature and humidity data) have been used. After the training of the neural networks, the input data are temperature and humidity values and the output is the fire presence (0 or 1).

For the training data set as mentioned above, a test setup in Figure 4.4 was used. 4 sensor nodes were used and the fires were started in the middle of the spatial cuboid base. This experiment was repeated changing the distance between the sensor node and the fire. That was changed from 60 cm to 200 cm for a few different values. Therefore, four sets of data can be observed once by four sensor nodes.

The data collection was started 30 minutes before the fire start point and continued for 30 minutes after the fire died (by extinguishing manually), with data being sampled every 30 seconds. Controllable fires were created by burning 2 kg of teak wood in a 30 cm x 30 cm area. The collected data was then used to train the neural network. The number of nodes in the hidden layer of the neural network was determined by studying the coefficients of correlation (R-squared) values of the training, validation, and test data. The required value of nodes in neural networks was selected according to the received R values greater than 0.9 [39].

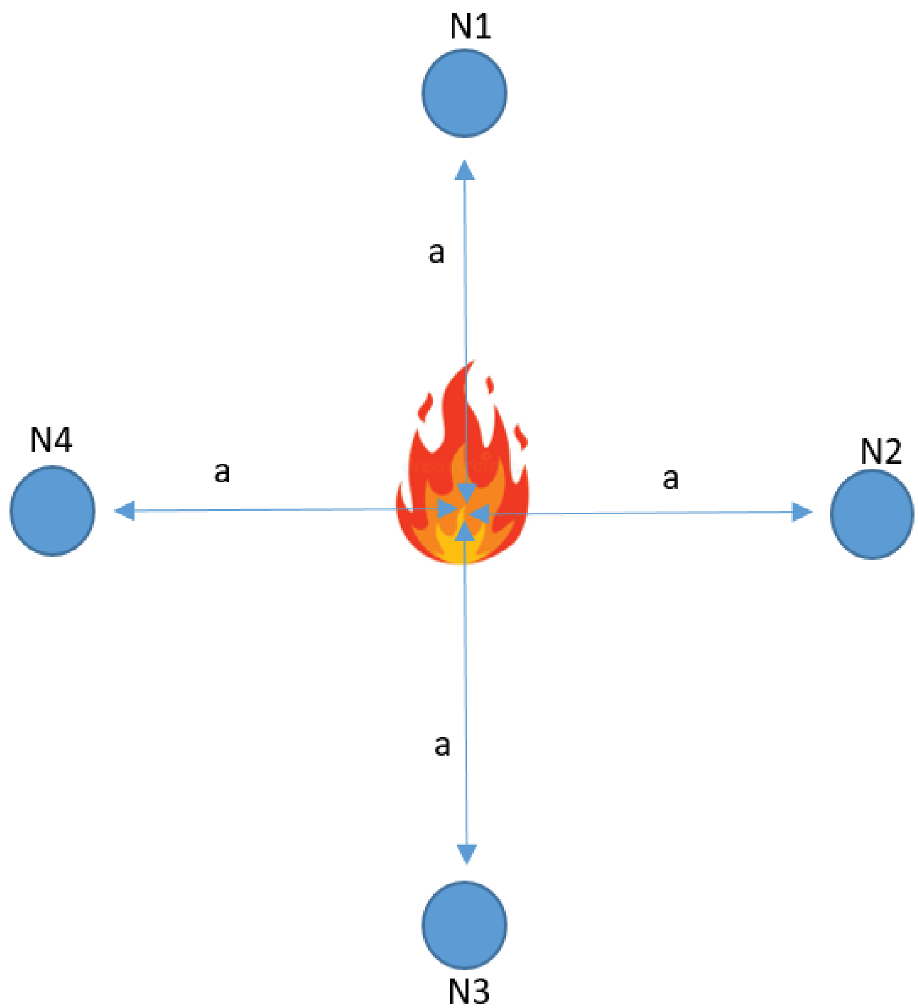

Fig. 4.4: Experiment setup to train the AI

4.3 Experiment setup for the evaluation of AI

The evaluation of AI performance for fire detection was carried out using the setup available in Figure 4.5, consisting of 4 sensor nodes for fire detection and an additional sensor node for ambient conditions monitoring (temperature and humidity). The task of this ambient monitoring sensor node is to collect the data to check the possibility of fire presence by the ambient data while taking the fire data. This sensor node has been placed a long distance exceeding the spatial size dimensions. the main sensor node. The sensor nodes were placed in a spatial cuboid with the determined dimensions of 150 cm x 150 cm x 100 cm by the spatial size determination. The fires, labeled as F1 and F2, were created at the locations mentioned in Figure 4.5. Then, the AI's performance in detecting the fires was evaluated.

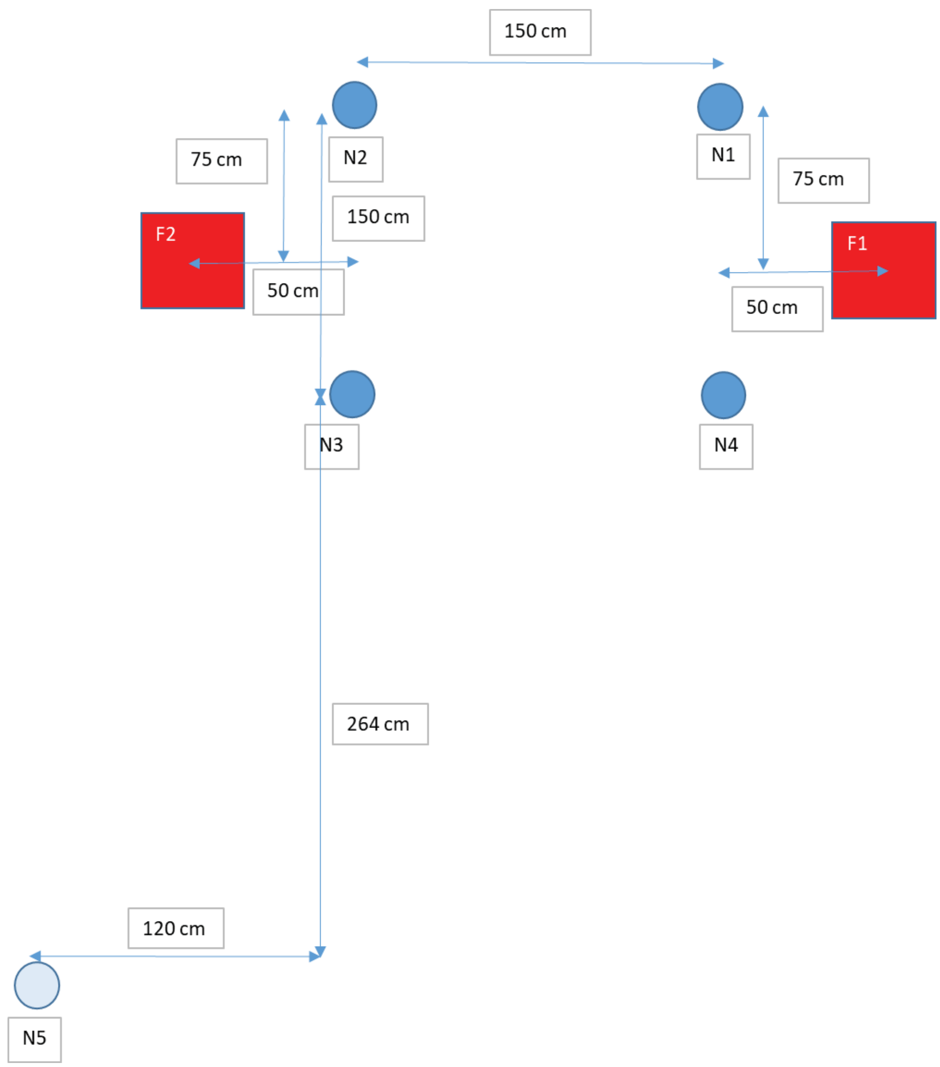

Fig. 4.5: Experimental setup for testing the trained algorithm

4.4 Evaluation of Packet loss of the Sensor Node

Packet loss of the sensor node has been analyzed for 50 data samples from a node to the final base station. The first receiver unit is the cluster head of the sensor node. That is

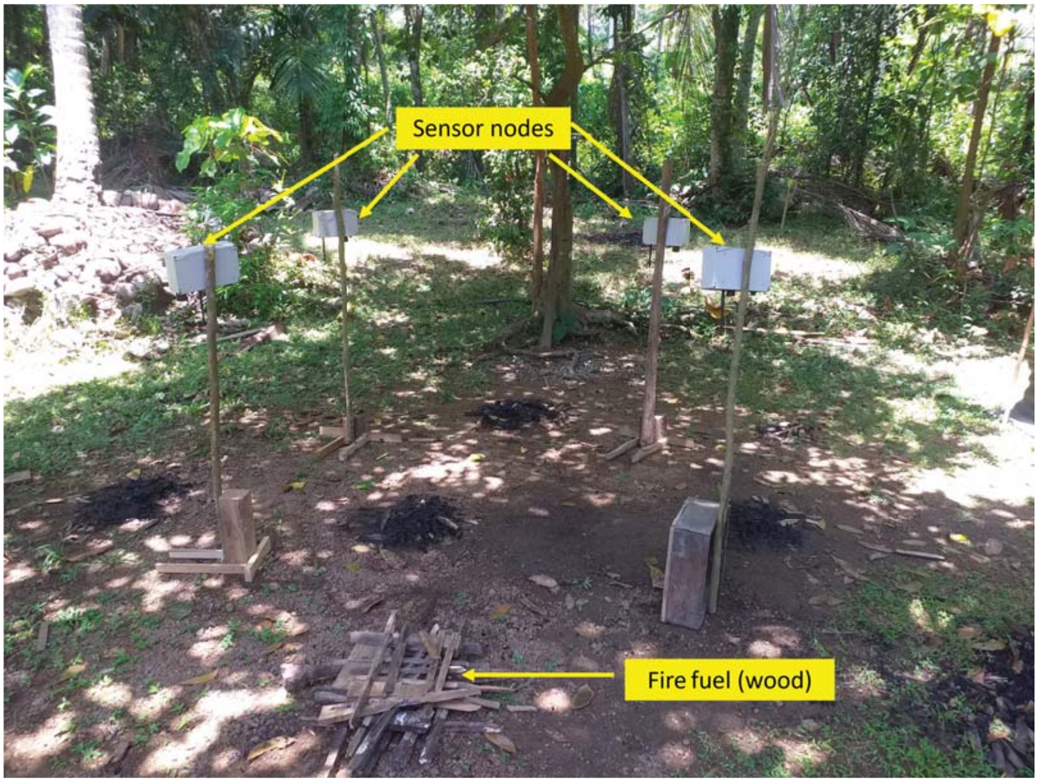

Fig. 4.6: Actual experimental setup for testing the trained algorithm

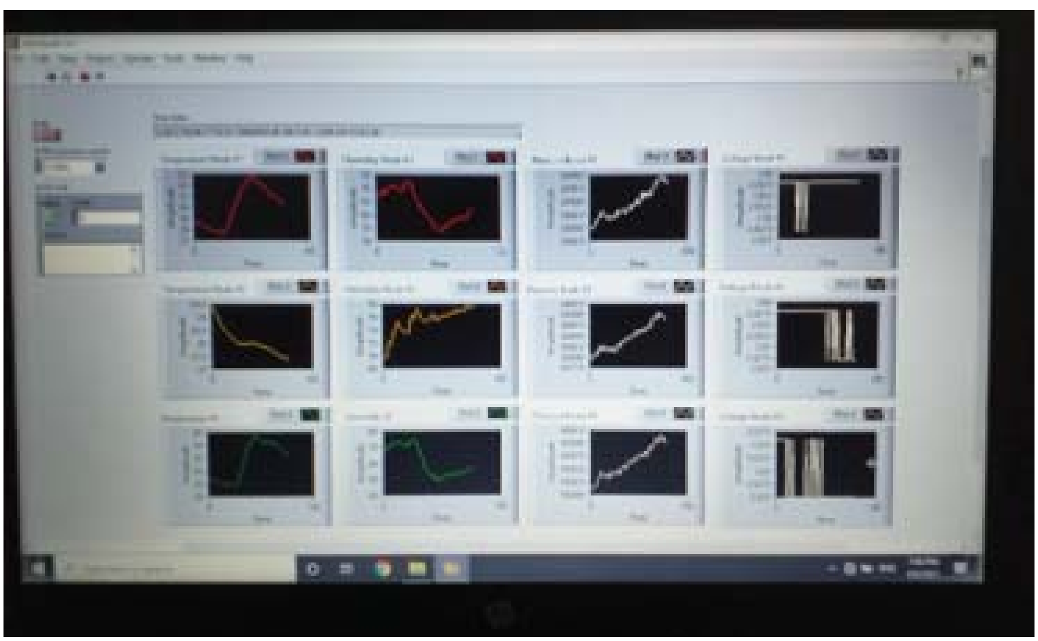
Fig. 4.7: Graphical user interface for data gathering

sending the data to the final base station. The packet loss for the total communication has been analyzed here.

4.5 Evaluation of Power Consumption of the Sensor Node

Sensor node has two main power consumption states which are sleep mode power consumption and active state power consumption. These two states of power consumption have been measured by two types of methods. The active state has been measured by using a power analyzer module (INA219) and the sleep mode power with the support of an 8-digit benchtop multimeter (measuring voltage and current).

For the active state power consumption measurement of a wireless sensor node, Texas Instruments INA219, and Node MCU were employed in [51] and a similar method was used for this implementation. In this implementation, an INA219 digital bidirectional current/power monitor was utilized to analyze the active power of the two sensor nodes mentioned earlier. INA219 modules were connected to an Arduino Uno through the I2C interface to achieve an average sampling time of 2.2 milliseconds. Power was supplied to the sensor nodes through the INA219 module. (Figure 4.8).

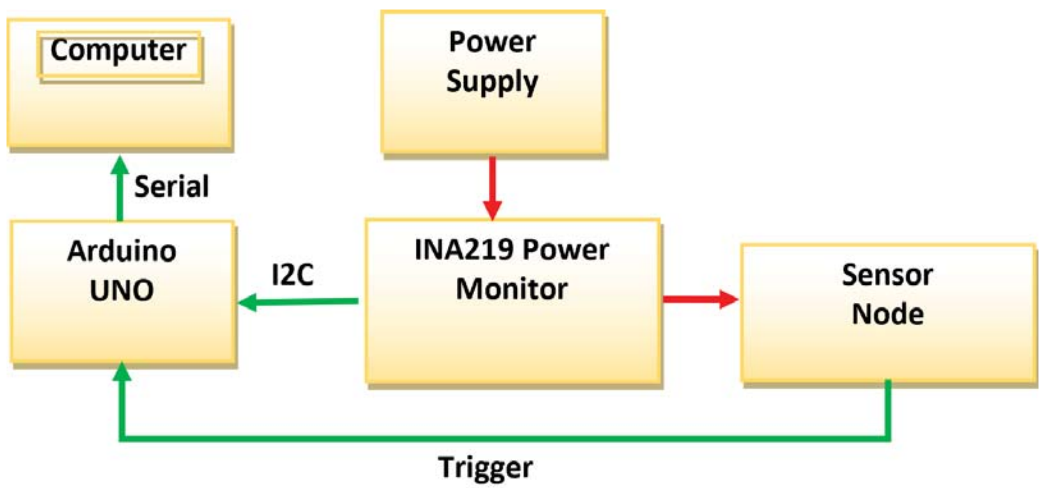

Fig. 4.8: Block diagram of the power measurement setup

When the sensor node entered to the active state, it initiated active power consumption-based measurements (voltage and current) and transmitted the data via a serial interface to a computer every 2.2 milliseconds. The active state of the node was identified by the power measurement setup upon receiving a trigger pulse from the node. The active power consumption was recorded over 50 active periods and data has been saved in a CSV file. This experiment was repeated for different voltage levels of the power supply. 50 active periods were averaged and observed a generalized power consumption curve (power vs time) for the sensor node.

In Figure 4.9 is the generalized active state power consumption of the sensor node at 3.0 V. To calculate the consumed energy of the sensor node trapezoidal integration method has been used on Matlab taking these data as a matrix. This was repeated for different supply voltages between the maximum and minimum battery voltage range (2 V - 3 V).

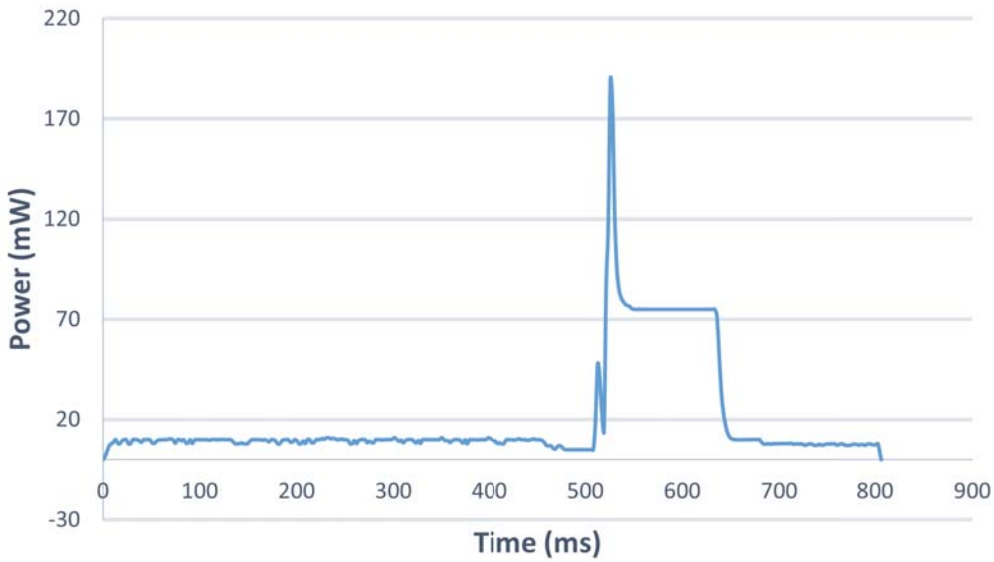

Fig. 4.9: Generalized active state power consumption of the sensor node at 3.0 V

To measure the power consumption in the sleep state by the sensor node, a microammeter, and a voltmeter have been used. After taking these measurements, power consumption was calculated. In the sleep state, the current consumed by the sensor node is constant because there is no function in the microcontroller and that does not have any function.

4.6 Evaluation of Data Recover Ability

To analyze the data recovery ability, the sensor module should be failed. Therefore, the sensor module failures were created by interrupting the I2C communication line. In this experiment, communication interruptions were only studied. These interrupts are created by another microcontroller randomly. Then the ability of the sensor module recovery was studied by comparing the number of failures, and the number of recoveries.

If the first sensor module is malfunctioned, the second module should be activated by the recovery system. In this situation, the expected recovery is the second module activation. If the second sensor module is malfunctioning or functioning and sensor module 1 is well functioning, nothing should be done. In this situation, the expected recovery is taking data from sensor module 1. If both sensor nodes are malfunctioned, the sensor node can not be recovered. In this situation, there is no expected recovery. These states have been summarized in Figure 4.10.

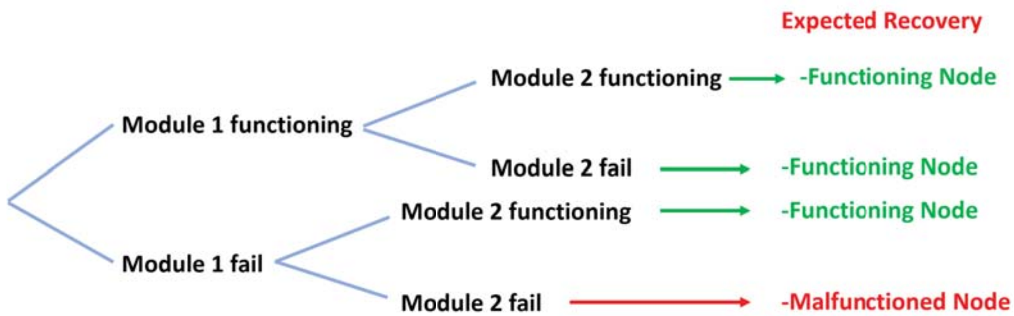

Fig. 4.10: Malfunctioning States of The Sensor Modules

The experiment setup for analyzing the recovery ability was created by Arduino Uno with a relay module. Relays interrupt the sensor I2C communication while reading the data. The block diagram of the experiment setup is available in Figure 4.11. The sensor node has been configured to sample the data every 30 seconds. Under this sampling rate, the above-mentioned experiment setup has interrupted the sensor module communication. After taking 50 samples of data and repeating the same method 6 times, the recovery ability has been evaluated and these results are available in the results and discussion section.

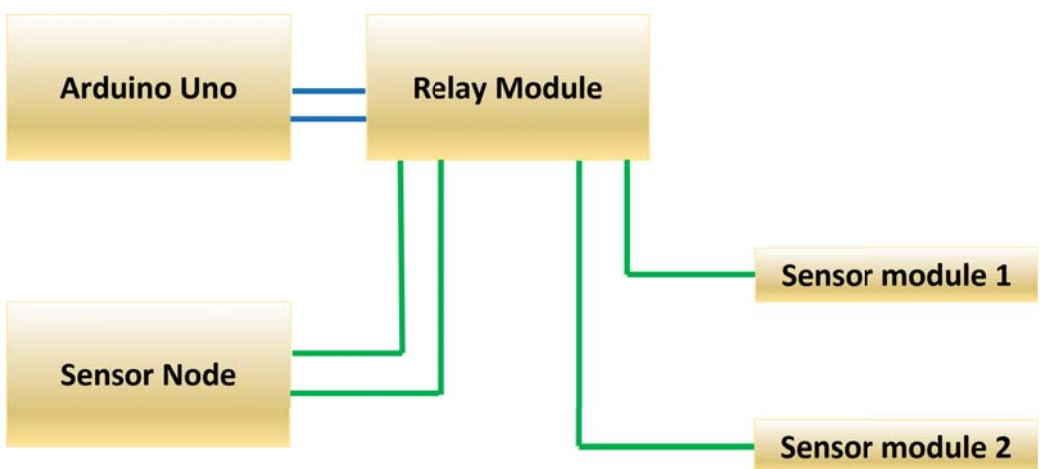

Fig. 4.11: The block diagram of the experimental setup for evaluation of the data recovery system

CHAPTER 5

RESULTS

5.1 Results for Power Consumption and System Life of the Sensor Node

5.1.1 Power Consumption Results

Power consumption of the sensor node is available in Table 5.1 and Table 5.2. Table 5.1 contains the power consumption at wake up state of the sensor node and Table 5.2 contains the power consumption at the sleep state of the sensor node.

These measurements were taken placing the sensor node and the receiver (the cluster head) 10 m apart from each other. There were no other operating nodes. The consumed energy by the shunt resistor is negligible compared to the active consumed energy by the nodes due to the shunt resistor value being 0.1 ohms is the first assumption we have taken. The energy consumed to generate a trigger pulse by nodes is negligible due to the short trigger period is the second assumption.

TABLE 5.1: CONSUMED ENERGY OF SENSOR NODE AT WAKE UP FOR DIFFERENT SUPPLY VOLTAGE LEVELS

Voltage of the Battery (V)	Consumed energy (mJ)
3	16.076
2.5	13.144
2	10.127
Averaged consumed energy	13.116

TABLE 5.2: CONSUMED ENERGY OF SENSOR NODE AT SLEEP FOR DIFFERENT SUPPLY VOLTAGE LEVELS

Voltage of the Battery (V)	Consumed energy (mJ)
3	5.4
2.5	4.5
2	3.6
Averaged consumed energy	4.5

5.1.2 System Life

According to the above sub-section details, the system life of the sensor node can be calculated as Table 5.3. According to these calculations, the estimated system life of the sensor node is 4.8 years. In a real situation, due to the self-discharge of alkaline batteries, this value will not be able to achieve completely.

TABLE 5.3: SYSTEM LIFE CALCULATION

Calculation requirement	Method of calculation	Result
Consumed energy for a single cycle (5 minutes)	Energy consumed at wake up + Energy consumed at sleep	13.116 + 4.5 = 17.616 (mJ)
5 minute cycle based samples for a day	(24 hours x number of minutes per hour)/5	(24 x 60)/5 = 288
Total power consumption per day	number of samples per day x consumed total energy per day	288 x 17.616 = 5.073 J/day
Capacity of a general alkaline AAA battery	-	9000 J
System life	Battery capacity/energy consumption per day	**9000/5.073 = 1774 days = 4.8 years**

5.2 Packet Loss Results of the Sensor Node Communication

In Table 5.4, the packet loss of the sensor node is available for different supply voltages and, distances. Here, we assumed that there is no packet loss through the communication between the cluster head and base station. The cluster head and the base station are apart from each other with 20 m for every experiment.

TABLE 5.4: PACKET LOSS PERCENTAGE OF THE SENSOR NODE COMMUNICATION

Supply	Distance(m)		
voltage (V)	30	40	50
3	0 %	0 %	0 %
2.5	0 %	0 %	0 %
2	0 %	0 %	6 %

5.3 Results of recover ability system

As previously described experiments in the experiments section, the expected number of recoveries and happened actual recoveries have been compared. The results are available in Table 5.5 and Figure 5.1.

TABLE 5.5: RESULTS OF RECOVER ABILITY SYSTEM

No.	Only 1st module failures out of 50 samplings	Only 2nd module failures out of 50 samplings	Both modules failure out of 50 samplings	Expected recoveries	Actual Recover-ies	Success (%)
1	5	4	1	4	4	100
2	9	12	4	5	5	100
3	10	14	2	8	8	100
4	9	11	1	8	8	100
5	13	12	3	10	10	100
6	6	12	3	3	3	100

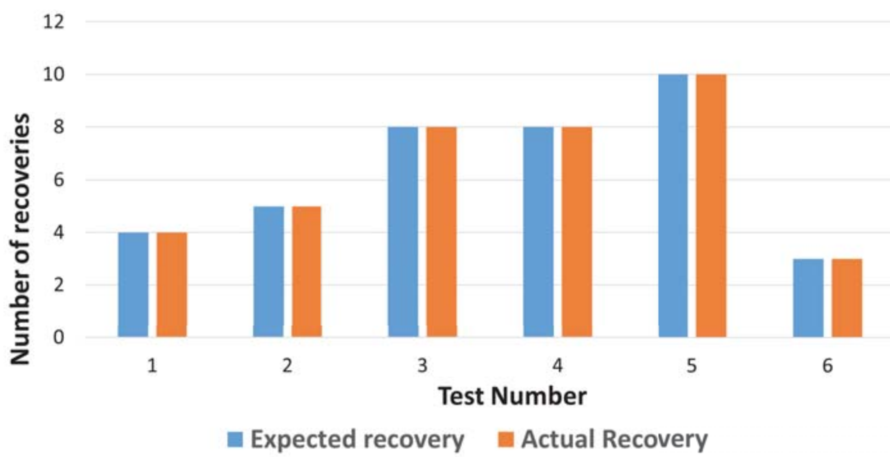

Fig. 5.1: Expected number of recoveries and the actual number of recoveries per 50 data samples

5.4 Results of Spatial Size Determination and Training the Neural Networks of the AI

5.4.1 Results of the spatial size determination

The spatial size determination results are available in the following Figure 5.2, Figure 5.3, Figure 5.4 and Table 5.6. The significant difference in temperature and humidity are available in Figure 5.3 and Figure 5.4. The actual temperature of the created fire is available in Figure 5.2. These results have been observed for the values of d = 150 cm and h = 100 cm.

TABLE 5.6: RESULTS OF THE SPATIAL SIZE DETERMINATION

d (cm)	h (cm)	Significant change in sensor data
120	100	Yes
150	100	Yes
210	100	Yes
300	100	Yes
420	100	No
100	200	No
150	200	No
200	200	No
300	200	No

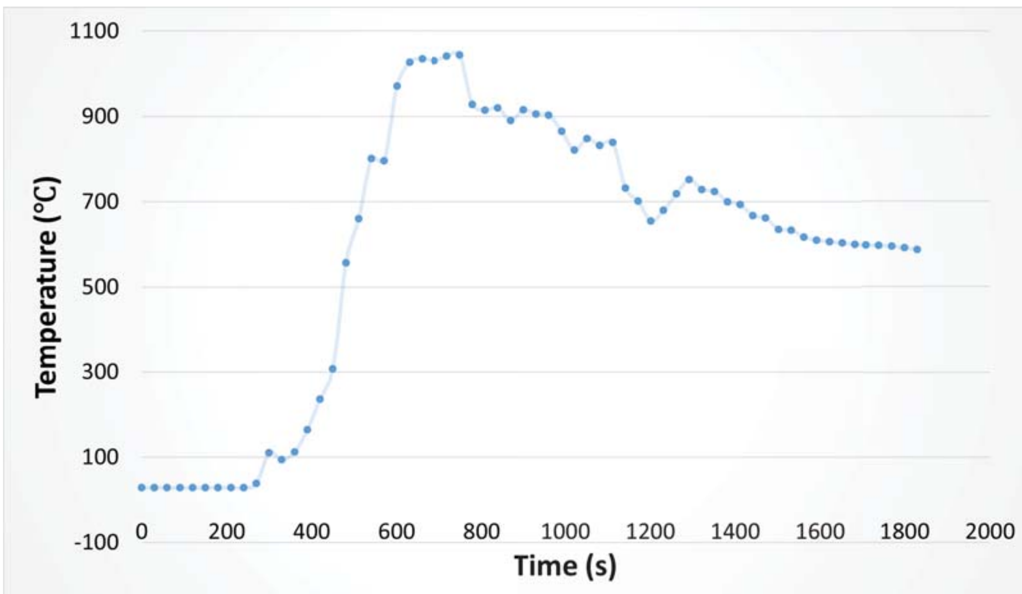

Fig. 5.2: Absolute Temperature of the Fire vs Time (d = 150 cm and h = 100 cm)

5.4.2 Results of training of the AI's Neural Networks

The observed coefficients of correlation (R-squared) values for training, validation, and test are available in the following Table 5.7 after training of the neural networks for both temperature and humidity based neural networks which consists of 15 nodes.

5.4.3 Results of Trained AI Evaluation

Results for the trained AI evaluation are available in the figures from Fig.5.5 to Fig.5.9. Each consist results of two pair of sensor nodes. The top two graphs are interpreting the NNs outputs after the cutoff with respect to the time and bottom two graphs are

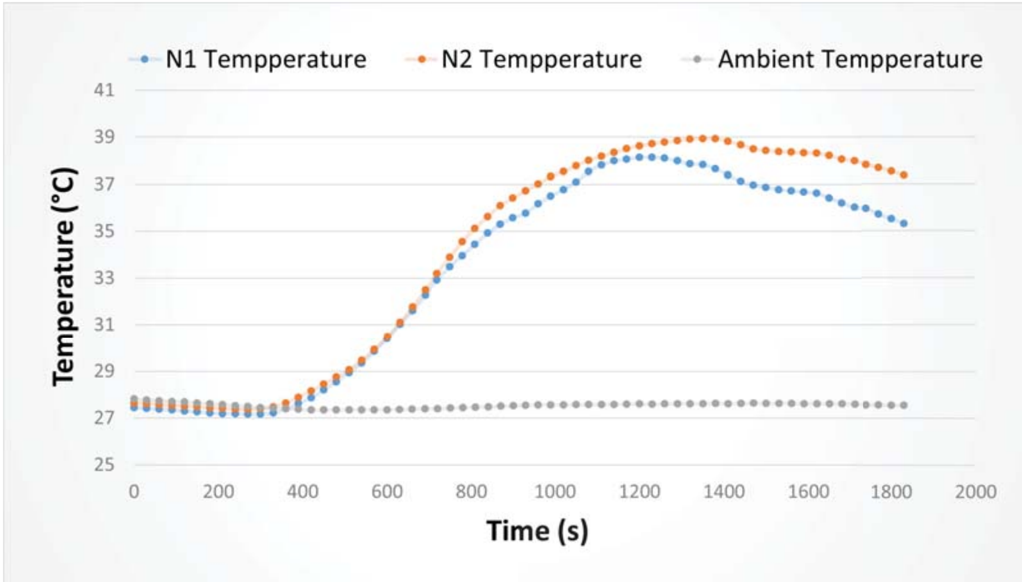

Fig. 5.3: Temperature Variation vs Time (d = 150 cm and h = 100 cm)

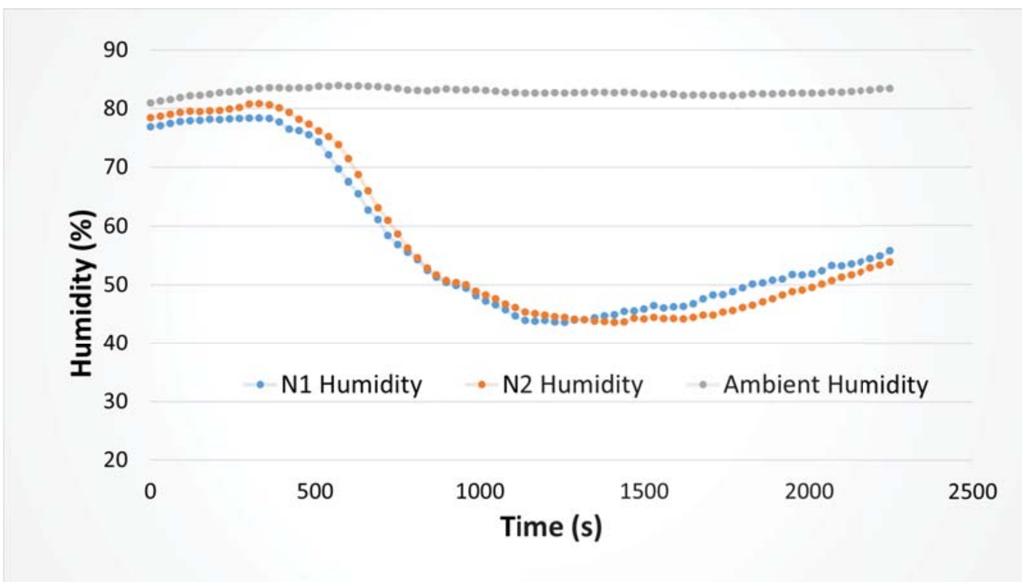

Fig. 5.4: Humidity Variation vs Time (d = 150 cm and h = 100 cm)

showing related final final fire probability output of the trained AI with respect to the time. The NNs outputs are available for both temperate and humidity and actual fire occurrence in binary form (occur or not).

TABLE 5.7: TRAINING RESULTS OF NEURAL NEWORKS; COEFFICIENT OF CORRELATION VALUES (R-SQUARED)

	Neural network for Prepossessed Temperature Data	**Neural network for Prepossessed Humidity Data**
Training	0.9402	0.9221
Validation	0.9543	0.9464
Test	0.9332	0.9104

TABLE 5.8: Summarized Results

	F1 Fire		F2 Fire	
Node	Expected Detection	Actual Detection	Expected Detection	Actual Detection
N1	Yes	Yes	No	No
N2	No	Yes[a]	Yes	Yes
N3	No	Yes[a]	Yes	Yes
N4	Yes	Yes	No	No
N5[b]	No	No	No	No

[a] less than N1 and N4 detection

[b] reference node, far away from rest

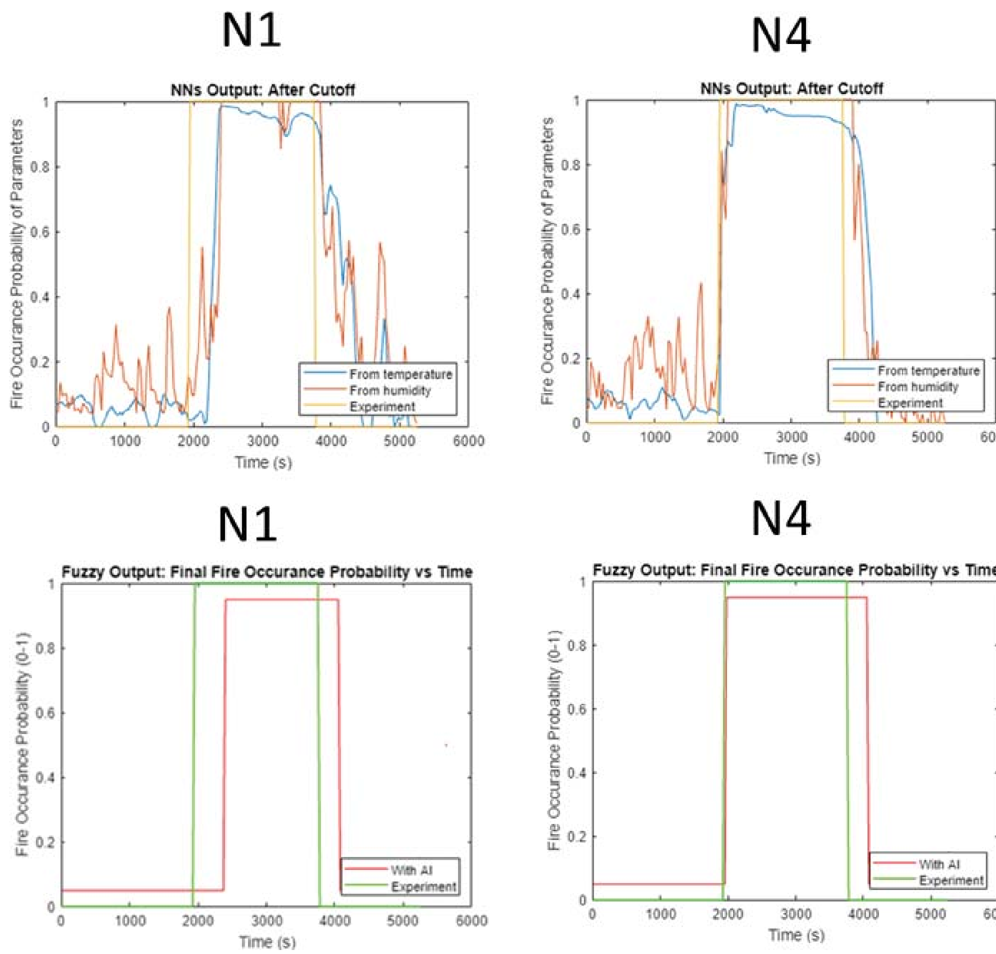

Fig. 5.5: Fuzzy output and final inference of the AI under the data from N1 and N4 nodes at F1

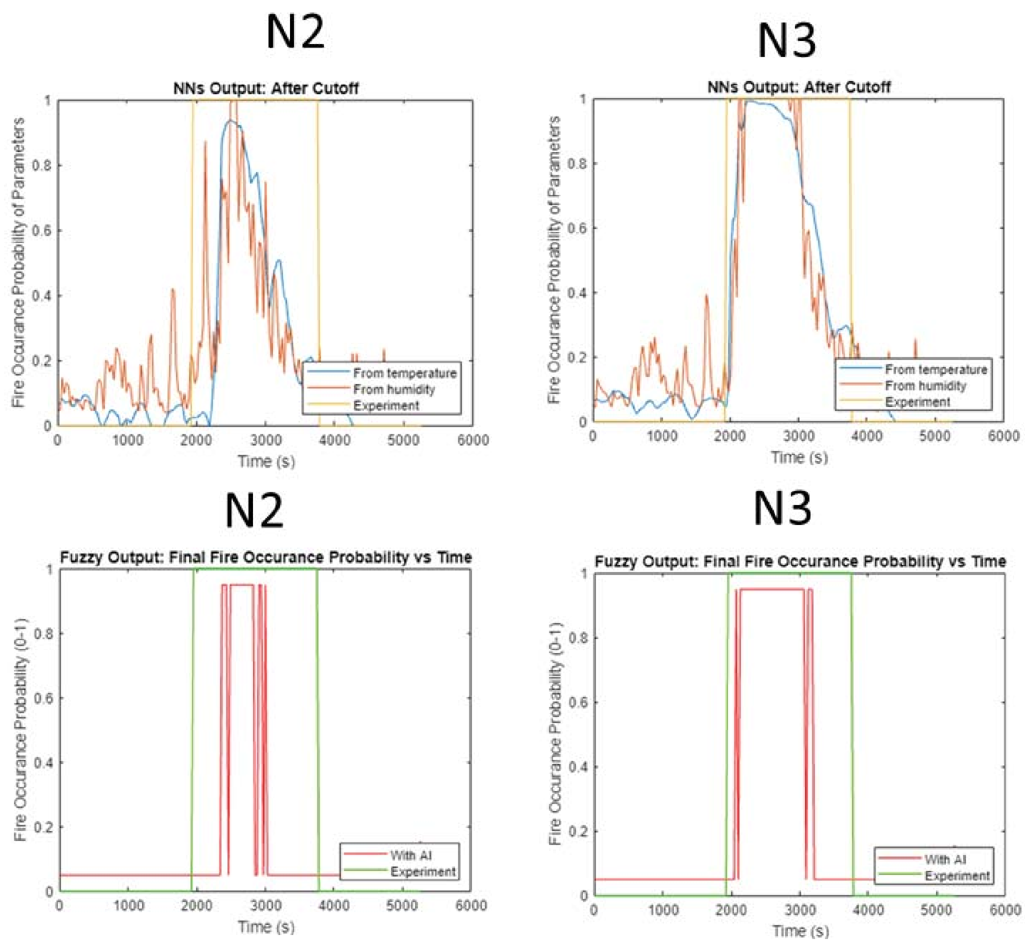

Fig. 5.6: Fuzzy output and final inference of the AI under the data from N2 and N3 nodes at F1

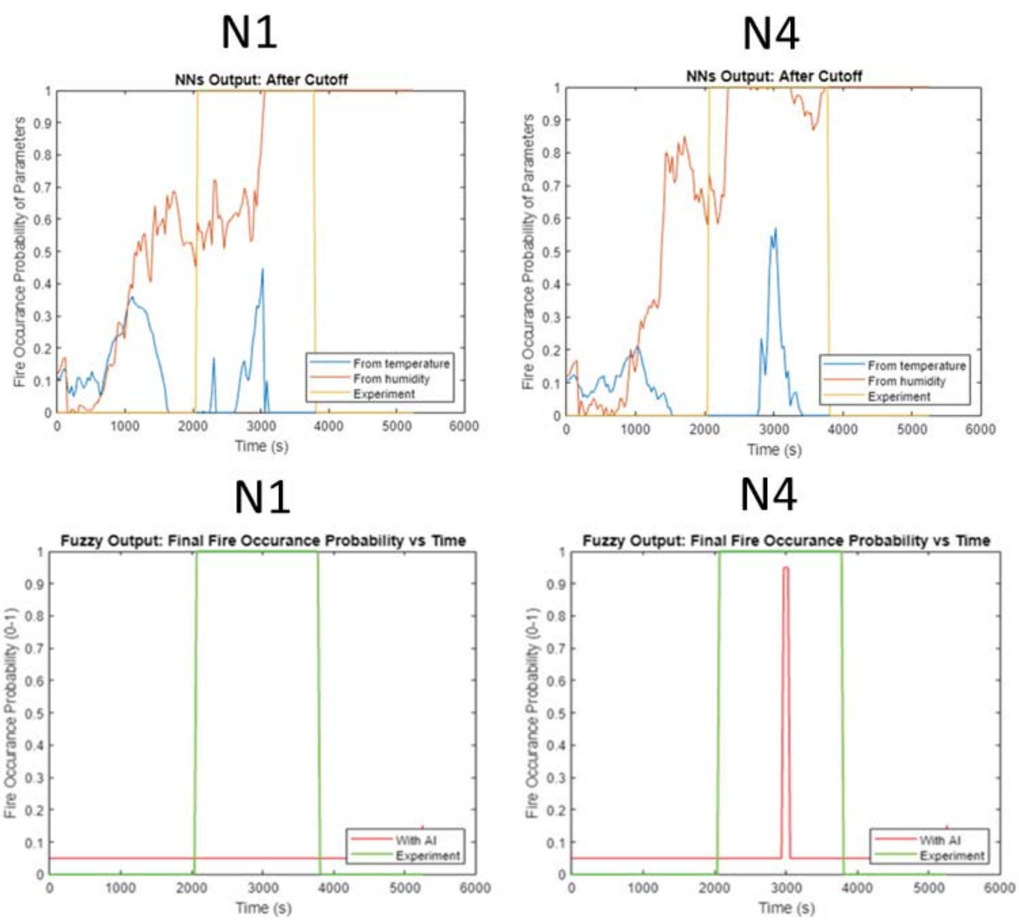

Fig. 5.7: Fuzzy output and final inference of the AI under the data from N1 and N4 nodes at F2

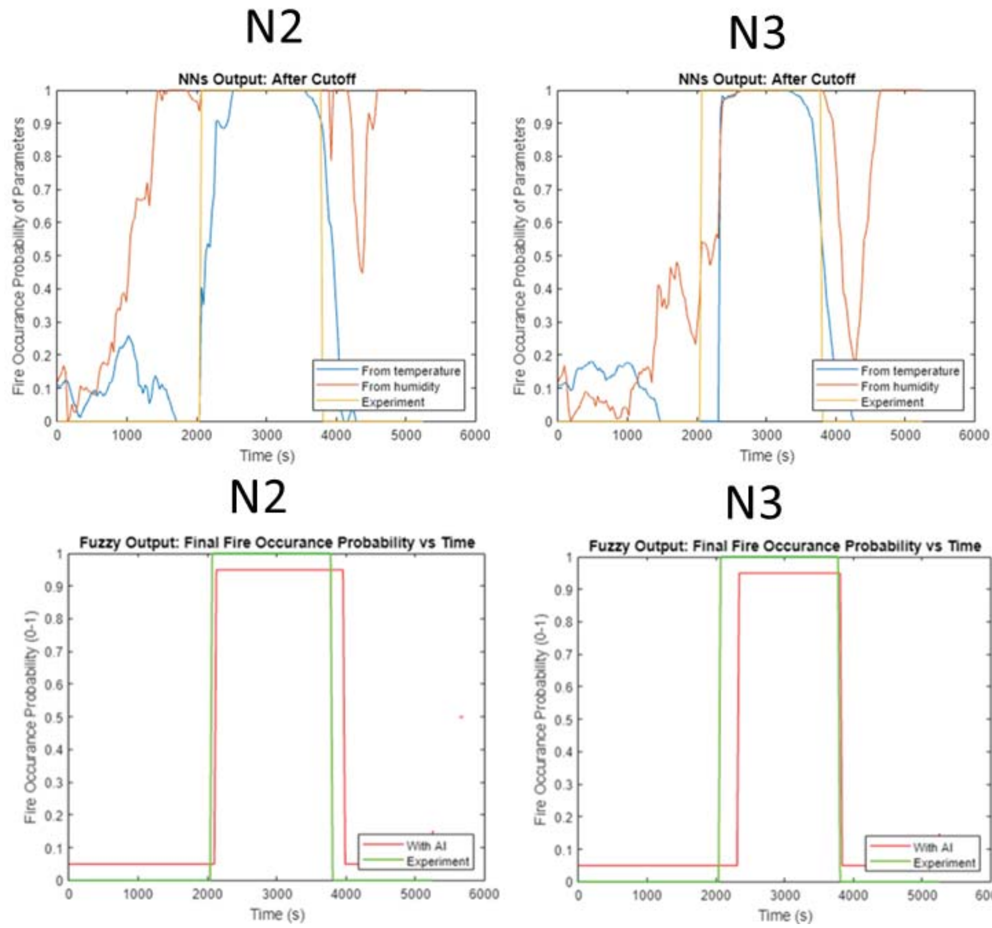

Fig. 5.8: Fuzzy output and final inference of the AI under the data from N2 and N3 nodes at F2

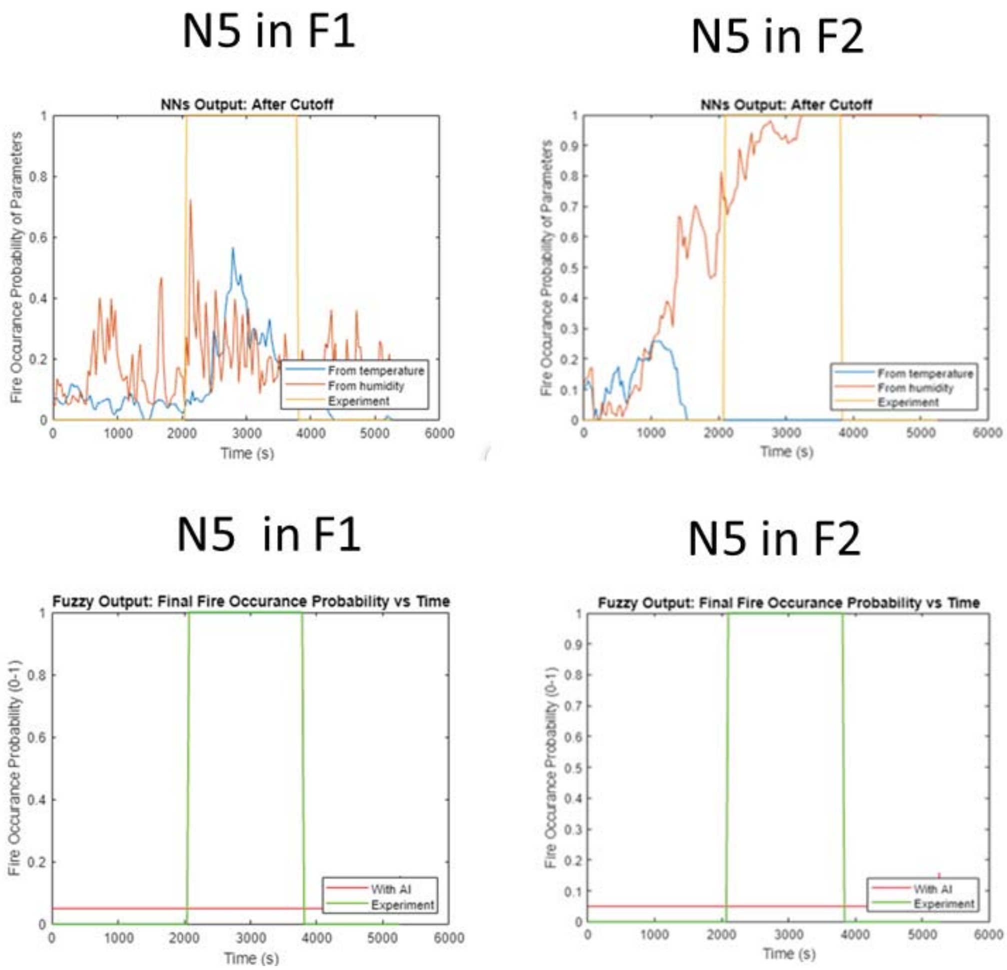

Fig. 5.9: Fuzzy output and final inference of the AI under the data from N5

CHAPTER 6

DISCUSSION

6.1 Discussion for Power Consumption and System Life of the Sensor Node

6.1.1 Power Consumption of the Sensor Node

The consumed energy has been measured for different voltage levels of the battery for two states of the node which are the sensor node at wake-up and the sensor node at sleep mode in the operation cycle of the sensor node. According to the above results of the consumed energy, most of the energy is consumed in the wake-up mode of the operating cycle of the sensor node compared to the consumed energy in the sleep mode of the sensor node. The reason for this high energy consumption is, that the required energy needed by the transceiver module is considerably high when the data is transmitted. Also, the sensor node's power consumption is getting low when the battery voltage is reduced when the sensor node is discharging. The major reason for this is the power loss of the components of the sensor node is less when the power supply voltage is low. The estimated total consumed energy is the summation of consumed energy in wake-up and consumed energy in sleep mode.

6.1.2 System Life

In the operating cycle of the sensor node, The sensor node is operating for only a very short time (low-duty cycle). Therefore, the system life of 4.8 years has been achieved very conveniently. This amount of system life can be improved by increasing the capacity of the battery pack accordingly. Also, due to the result of this, the node replacement time and service period can be increased. When increasing the battery capacity the self-discharge of the battery pack should also be considered.

6.1.3 Packet loss

The results of packet loss have been obtained from the sensor node to the cluster head and cluster head to the base station. According to these results, if the battery has a good state of charge and is closer to the sensor node, there is no packet loss. But 6 % is the only available packet loss at 2 V of supply voltage for the sensor node and any other situations are not available. The sensor node can be guaranteed less than 50 m with a considerable amount of the packet loss-free design. Also, in research, the deployment of sensor nodes is less than 10 m. This packet loss result is mostly sufficient.

6.1.4 Data Recovery Ability

In the first experiment of the recovery ability system, for the 50 data samples, 1st sensor module 5 times, the second module 4 times, and both modules a single time have been failed by the external setup. Therefore expected number of recoveries is 4 because recovery should be done only when the first module fails and no need to recover if the first module is functioning whether the second module malfunctioned or not. As the result of actual recovery is also 4, the recovery system has 100 % of success. According to other results and considering an evaluation similar to the above discussion, the recovery system has recovered every possible recovery because they have the same 100 % success rate.

6.2 Discussion for Spatial Size Determination and Training the Neural Networks of the AI

6.2.1 Spatial Size Determination

According to the observed results of the experiments for spatial size determination, there is a significant difference in the data (temperature and humidity) in a maximum range of 0 cm – 300 cm for the distance between two sensor nodes and at a maximum height of 100 cm. Therefore, this sensor node has a detectable range of 0 cm - 150 cm for node-to-node distance and 100 cm of node height. When deploying a sensor node in a cuboid the maximum node-to-node distance will be along the diagonal. Therefore, the diagonal length of the base of the cuboid should be 300 cm and the maximum spatial size should be s = 200 cm and h = 100 cm.

6.2.2 Training of the AI's Neural Networks

The R-squared values are less than 0.9 for less than 15 nodes or greater than 15 nodes of both neural networks of temperature and humidity data-based neural networks. But for the nodes of 15 both neural networks have an R-squared value greater than 0.9. Hence, The suitable number of nodes for the neural networks of the AI is 15 as mentioned in Table 5.7.

6.2.3 Trained AI Evaluation

In the F1 fire, the N1 and N4 nodes were closer to the fire, while N2 and N3 were farther away (as shown in Figure 4.5). Similarly, for the F2 fire, the N2 and N3 nodes were closer to the fire, while N1 and N4 were farther away.

As shown in Figure 5.5, N1 and N4 data should have a significant inference by the AI under the F1 fire. In the results, the AI showed a clear fire inference as expected, but N2 and N3 also showed some fire inferences, but less than N1 and N4 (as shown in Figure 5.6). Here we have assumed that these inferences occurred due to the impact of fire data propagating due to wind from N1 to N2 area. However, the faraway sensor node measuring the ambient condition and their data did not infer the fire. Results for the N5 node are shown in Figure 5.9.

In the F2 fire creation, The expected outcome should be the data from N2 and N3 should have a clear inference and the data from N1 and N4 should not have a fire inference. The data from N2 and N3 showed clear fire inferences as expected (as shown in Figure 5.7), while N1 did not have any fire inference also as expected. However, N4 data showed a negligible spike-like fire inference (as shown in Figure 5.8), which could be disregarded. As expected, the data from ambient sensor nodes did not infer any fire (as shown in Figure 5.9). All these results are summarized in Table 5.8.

The obtained results confirmed [52] that the sensor node's data have the ability to detect forest fires. The existing recent forest fire detection systems [19], [40], [41] & [42] are much more complex due to the integration of more sensors and cameras limiting the deployment ability and increasing the serviceability. Then the proposed system has the ability of forest fire detection with a convenient deployment and less complex design.

The stability of the proposed system is acceptable for only some situations (Figure 5.5) but the unstable situations are also acceptable after taking some time, as well as before probability goes to a lower value. When comparing the system complexity of the proposed system with the literature as discussed above, the is much less and therefore these lower ability issues can be discarded. A summarized comparison is available in Table 6.1.

According to the comparison in Table 6.1, the proposed system has a longer delay than the other systems. However, the other systems have much more complexity than the proposed system due to cameras and additional sensors. In forest fire detection, a large number of sensor nodes should be deployed. Therefore, the proposed system is more advanced in the aspect of deployment even though the detection delay is high.

One of the most recent developments in forest fire detection is the BME688 sensor-based forest fire detection sensor node from BOSCH company [53]. Compared to the proposed system, they use gas sensors additionally in a single sensor module. Also, the most specific feature of this newly developed sensor module is that it consists of an AI-based algorithm to process the data on the spot allowing low power consumption and less processing time. This simplifies the development procedure of the sensor node as well. Therefore, this BME688 sensor module can be used to reduce the currently observed delay in the proposed system, by keeping the sensor node in low power consumption and reducing the data processing time further.

TABLE 6.1: COMPARISON OF THE PROPOSED METHOD WITH EXISTING METHODS

Method	Complexity	Stability	Practicality for a large scale deployment	Detection delay (s)
Thresholding [16], [19]	Low (only sensors)	Low	Yes	Low (55 [19])
CNN [19]	High (cameras)	Medium	No	High (70)
Multi-AI [19]	High (sensors and cameras)	High	No	Very low (5)
Proposed	Low (only sensors)	High (not as [19])	Yes	Very high (825)

CHAPTER 7

CONCLUSION AND FUTURE WORKS

The present study introduces a multi-AI architecture and WSNs-based system that offers a less complex approach to detecting outdoor fires. This system stands out among existing approaches discussed in the literature, as evidenced by the observed results. Unlike deep neural networks utilized in related studies, the proposed system relies exclusively on temperature and humidity data, rendering it simpler in design. However, there is considerable potential for system improvement and reduction of current delays through the integration of additional sensors. Incorporating more sensors could increase the complexity of the node. The streamlined setup allows for direct fire alarm generation by the WSNs, thereby enhancing the power consumption of the sensor nodes can be decreased and extend their operational lifespan due to there is no data periodic transmission as the existing methods and proposed method. In this case of future work, the fire probability is the only transmission parameter and the transmission payload is only a single binary value for the fire detection which is whether a fire is available or not.

The estimated maximum lifespan of the sensor node is 4.8 years. This is a promising result for forest fire detection with large-scale deployment and long service periods in the case of replacing the batteries. However, it is important to consider the possibility of diminished battery life due to self-discharge in alkaline AAA batteries under certain conditions. To ensure the expected system lifespan, a similar Lithium thionyl chloride battery can be used. These Lithium thionyl chloride batteries can extend system life and ability to maintain optimal performance.

The packet loss study demonstrates highly satisfactory results for the sensor node to make reliable communication for the device with the base station. Specifically, the packet loss rate remains at less than 6% when the distance is below 50 m, even at a battery voltage as low as 2V. This distance value exceeds the minimum requirement established through spatial size determination, and no packet losses were observed within this range. Such reliability is immensely valuable for dependable forest fire detection.

Regarding the recovery system, our findings indicate a 100 % recovery rate when communication failure occurs between the microcontroller and the sensor modules. As part of future works, additional key components can be incorporated into the recovery system to enhance its capabilities. However, the cost per sensor nod can be doubled in this implementation.

Moreover, this sensor node exhibits versatility beyond fire detection, lending itself to various other applications such as environmental monitoring, precision agriculture, weather monitoring, and more. The availability of supplementary interfaces on the sensor node enables seamless integration with necessary sensors, devices, and other components, making it an adaptable solution.

The practical implementation (practicality) of the proposed system is a considerably complex task. That can be subdivided into several aspects. They are the deployment of sensor nodes, communicating with the sensor node via the network, and maintenance of the sensor networks for a long time. The current sensor network's final spatial size (150 cm x 150 cm x 100 cm) is very low but that is for small-scale fire conditions (2 kg of fire fuel: wood). However, the detection range can be improved further by optimizing the algorithm with high-scale fire conditions. After a spatial size increment, the deployment will be simplified further. The propagation property of forest fires can be considered to reduce the complexity of deployment, by reducing the number of sensor nodes. The cost can also be reduced further due to this reduction. That means if the fire fuel in the area that has been planned to deploy the sensor nodes has the property of being highly combustible and the propagation property of the fire is high, then the minimum dimensions for the node deployment can be increased further. As a result of that, the whole deployment process will be simplified further.

The communication (from sensor node to base station) through the network should be simple. That is possible due to this AI-based algorithm is much simpler and can be implemented in the microcontroller. Then the communication only happens when a fire is detected. The sensor node only sends the final state of the fire occurrence probability, instead of the periodic data. Then, a much simpler communication method can be achieved. In the sensor node, most of the power is consumed by the transceiver module. Therefore, power consumption can also be reduced further. That will improve the system life further as well. Also, communication can be further optimized with a proper routing protocol.

In summary, the proposed multi-AI architecture and WSNs-based system provide a more straightforward and effective approach to outdoor fire detection than the existing systems in the literature. By relying on temperature and humidity data, our system achieves simplicity with compromising accuracy with long life, recoverable, and reliable communication. Although the integration of additional sensors holds promise for further improvement, careful consideration must be given to the accompanying increase in complexity. The sensor node demonstrates commendable performance in terms of packet loss, meets deployment requirements, and incorporates a robust recovery system. Furthermore, its flexibility enables deployment in diverse applications beyond fire detection. Future work can focus on refining and expanding these capabilities to enhance the overall system.

LIST OF PUBLICATIONS

1. T. P. D. Pieris, K. V. D. S. Chathuranga, A. L. Kulasekera, P. Guha and P. Mukhija, "Energy and Power Consumption Analysis of a Wireless Sensor Node without a Voltage Regulator," 2020 5th International Conference on Information Technology Research (ICITR), Moratuwa, Sri Lanka, 2020, pp. 1-6, doi: 10.1109/ICITR51448.2020.9310847.

2. T. P. D. Pieris, A. L. Kulasekera and K. V. D. S. Chathuranga, "Multi-AI Based Wireless Sensor Node for Forest Fire Detection," 2022 7th International Conference on Information Technology Research (ICITR), Moratuwa, Sri Lanka, 2022, pp. 1-6, doi: 10.1109/ICITR57877.2022.9992560.

REFERENCES

[1] N. J. Abram, B. J. Henley, A. Sen Gupta, T. J. Lippmann, H. Clarke, A. J. Dowdy, J. J. Sharples, R. H. Nolan, T. Zhang, M. J. Wooster *et al.*, "Connections of climate change and variability to large and extreme forest fires in southeast australia," *Communications Earth & Environment*, vol. 2, no. 1, p. 8, 2021.

[2] A. P. Williams, J. T. Abatzoglou, A. Gershunov, J. Guzman-Morales, D. A. Bishop, J. K. Balch, and D. P. Lettenmaier, "Observed impacts of anthropogenic climate change on wildfire in california," *Earth's Future*, vol. 7, no. 8, pp. 892–910, 2019.

[3] P. M. Fernandes, G. M. Davies, D. Ascoli, C. Fernández, F. Moreira, E. Rigolot, C. R. Stoof, J. A. Vega, and D. Molina, "Prescribed burning in southern europe: developing fire management in a dynamic landscape," *Frontiers in Ecology and the Environment*, vol. 11, no. s1, pp. e4–e14, 2013.

[4] J. K. Balch, B. A. Bradley, J. T. Abatzoglou, R. C. Nagy, E. J. Fusco, and A. L. Mahood, "Human-started wildfires expand the fire niche across the united states," *Proceedings of the National Academy of Sciences*, vol. 114, no. 11, pp. 2946–2951, 2017.

[5] S. Mathews, P. Ellis, and J. Hurle, "Evaluation of three systems," *Australia: Bushfire Cooperative Research Centre*, 2010.

[6] P. Coppo, "Simulation of fire detection by infrared imagers from geostationary satellites," *Remote Sensing of Environment*, vol. 162, pp. 84–98, 2015.

[7] R. Rosas-Romero, "Remote detection of forest fires from video signals with classifiers based on k-svd learned dictionaries," *Engineering Applications of Artificial Intelligence*, vol. 33, pp. 1–11, 2014.

[8] Y. G. Sahin, "Animals as mobile biological sensors for forest fire detection," *Sensors*, vol. 7, no. 12, pp. 3084–3099, 2007.

[9] W. Krüll, R. Tobera, I. Willms, H. Essen, and N. von Wahl, "Early forest fire detection and verification using optical smoke, gas and microwave sensors," *Procedia Eng.*, vol. 45, pp. 584–594, 2012.

[10] A. A. A. Alkhatib, "A review on forest fire detection techniques," *Int. J. Distrib. Sens. Netw.*, vol. 10, no. 3, p. 597368, Mar. 2014.

[11] V. Chowdary, M. K. Gupta, and R. Singh, "A review on forest fire detection techniques: A decadal perspective," *Networks*, vol. 4, p. 12, 2018.

[12] R. Vega-Rodriguez, S. Sendra, J. Lloret, P. Romero-Diaz, and J. L. Garcia-Navas, "Low cost LoRa based network for forest fire detection," in *2019 Sixth International Conference on Internet of Things: Systems, Management and Security (IOTSMS)*. IEEE, Oct. 2019.

[13] W. Quinn, P. Angove, J. Buckley, J. Barrett, and G. Kelly, "Design and performance analysis of an embedded wireless sensor for monitoring concrete curing and structural health," *J. Civ. Struct. Health Monit.*, vol. 1, no. 1-2, pp. 47–59, Jun. 2011.

[14] D. Oletic and V. Bilas, "Design of sensor node for air quality crowdsensing," in *2015 IEEE Sensors Applications Symposium (SAS)*. IEEE, Apr. 2015.

[15] D. D. K. Rathinam, D. Surendran, A. Shilpa, A. S. Grace, and J. Sherin, "Modern agriculture using wireless sensor network (WSN)," in *2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS)*. IEEE, Mar. 2019.

[16] S. Abdullah, S. Bertalan, S. Masar, A. Coskun, and I. Kale, "A wireless sensor network for early forest fire detection and monitoring as a decision factor in the context of a complex integrated emergency response system," in *2017 IEEE Workshop on Environmental, Energy, and Structural Monitoring Systems (EESMS)*. IEEE, Jul. 2017.

[17] A. R. Ulucinar, I. Korpeoglu, and A. E. Cetin, "A WI-fi cluster based wireless sensor network application and deployment for wildfire detection," *Int. J. Distrib. Sens. Netw.*, vol. 10, no. 10, p. 651957, Oct. 2014.

[18] K. Bouabdellah, H. Noureddine, and S. Larbi, "Using wireless sensor networks for reliable forest fires detection," *Procedia Comput. Sci.*, vol. 19, pp. 794–801, 2013.

[19] J. H. Park, S. Lee, S. Yun, H. Kim, and W.-T. Kim, "Dependable fire detection system with multifunctional artificial intelligence framework," *Sensors (Basel)*, vol. 19, no. 9, p. 2025, Apr. 2019.

[20] B. Lawson and B. Armitage, "Weather guide canadian forest fire danger rating system," 01 2008.

[21] P. Schlobohm and J. Brain, "Gaining an understanding of the national fire danger rating system." Idaho, USA: NWCG Fire Danger Working Team, July 2002.

[22] Y. G. Sahin and T. Ince, "Early forest fire detection using radio-acoustic sounding system," *Sensors*, vol. 9, no. 3, pp. 1485–1498, 2009.

[23] F. Karray, M. W. Jmal, A. Garcia-Ortiz, M. Abid, and A. M. Obeid, "A comprehensive survey on wireless sensor node hardware platforms," *Computer Networks*, vol. 144, pp. 89–110, 2018.

[24] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer networks*, vol. 38, no. 4, pp. 393–422, 2002.

[25] B. Sensortec, "Bme280 combined humidity and pressure sensor," *Bosch Sensortec*, 2015.

[26] A. Silva, M. Liu, and M. Moghaddam, "Power-management techniques for wireless sensor networks and similar low-power communication devices based on nonrechargeable batteries," *Journal of Computer Networks and Communications*, vol. 2012, 2012.

[27] C. M. Nguyen, J. Mays, D. Plesa, S. Rao, M. Nguyen, and J.-C. Chiao, "Wireless sensor nodes for environmental monitoring in internet of things," in *2015 IEEE MTT-S International Microwave Symposium*. IEEE, 2015, pp. 1–4.

[28] O. Berder and O. Sentieys, "Powwow: Power optimized hardware/software framework for wireless motes," in *23th International Conference on Architecture of Computing Systems 2010*. VDE, 2010, pp. 1–5.

[29] S. Yessad, L. Bouallouche-Medjkoune, and D. Aïssani, "Comparison of routing protocols in wireless sensor networks." in *VECoS*. Citeseer, 2014, pp. 135–142.

[30] A. R. Silva and M. C. Vuran, "Empirical evaluation of wireless underground-to-underground communication in wireless underground sensor networks," in *Distributed Computing in Sensor Systems: 5th IEEE International Conference, DCOSS 2009, Marina del Rey, CA, USA, June 8-10, 2009. Proceedings 5*. Springer, 2009, pp. 231–244.

[31] W. Quinn, P. Angove, J. Buckley, J. Barrett, and G. Kelly, "Design and performance analysis of an embedded wireless sensor for monitoring concrete curing and structural health," *Journal of Civil Structural Health Monitoring*, vol. 1, pp. 47–59, 2011.

[32] B. J. Stocks, B. Lawson, M. Alexander, C. V. Wagner, R. McAlpine, T. Lynham, and D. Dube, "The canadian forest fire danger rating system: an overview," *The Forestry Chronicle*, vol. 65, no. 6, pp. 450–457, 1989.

[33] E. Popovici, M. Magno, and S. Marinkovic, "Power management techniques for wireless sensor networks: a review," in *5th IEEE International Workshop on Advances in Sensors and Interfaces IWASI*. IEEE, 2013, pp. 194–198.

[34] H. A. R. M Udin, E. Depandi, and S. Ferry Astika, "Smart home system for fire detection monitoring based on wireless sensor network," in *2019 International Electronics Symposium (IES)*. IEEE, Sep. 2019.

[35] G. N. Pham, "Smoke detection for outdoor environment based on image processing."

[36] S. Chaturvedi, P. Khanna, and A. Ojha, "A survey on vision-based outdoor smoke detection techniques for environmental safety," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 185, pp. 158–187, 2022.

[37] S. Saponara, A. Elhanashi, and A. Gagliardi, "Exploiting r-cnn for video smoke/fire sensing in antifire surveillance indoor and outdoor systems for smart cities," in *2020 IEEE International Conference on Smart Computing (SMARTCOMP)*. IEEE, 2020, pp. 392–397.

[38] T. Pieris, A. Kulasekera, and K. Chathuranga, "Multi-ai based wireless sensor node for forest fire detection," in *2022 7th International Conference on Information Technology Research (ICITR)*. IEEE, 2022, pp. 1–6.

[39] M. Grari, M. Yandouzi, I. Idrissi, M. Boukabous, O. Moussaoui, M. AZIZI, and M. MOUSSAOUI, "Using iot and ml for forest fire detection, monitoring, and prediction: a literature review," *Journal of Theoretical and Applied Information Technology*, vol. 100, no. 19, 2022.

[40] W. Benzekri, A. El Moussati, O. Moussaoui, and M. Berrajaa, "Early forest fire detection system using wireless sensor network and deep learning," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 5, 2020.

[41] V. Zope, T. Dadlani, A. Matai, P. Tembhurnikar, and R. Kalani, "Iot sensor and deep neural network based wildfire prediction system," in *2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS)*. IEEE, 2020, pp. 205–208.

[42] U. Dampage, L. Bandaranayake, R. Wanasinghe, K. Kottahachchi, and B. Jayasanka, "Forest fire detection system using wireless sensor networks and machine learning," *Scientific reports*, vol. 12, no. 1, p. 46, 2022.

[43] J. S. Carranco, F. D. Salgado, C. Sellers, and H. Torres, "Comparative analysis of meteorological monitoring using an integrated low-cost environmental unit based on the internet of things (iot) with an automatic meteorological station (aws)," in *2017 IEEE Second Ecuador Technical Chapters Meeting (ETCM)*. IEEE, 2017, pp. 1–6.

[44] P. Six and P. serial USART, "8-bit avr microcontroller with 32k bytes in-system programmable flash," *Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet. pdf*, 2015.

[45] N. Semiconductor, "nrf24l01+ single chip 2.4 ghz transceiver preliminary product specification v1. 0," *nRF24L01+ datasheet*, 2008.

[46] C. Cambra, S. Sendra, L. Garcia, and J. Lloret, "Low cost wireless sensor network for rodents detection," in *2017 10th IFIP Wireless and Mobile Networking Conference (WMNC)*. IEEE, 2017, pp. 1–7.

[47] J. Beutel, S. Gruber, A. Hasler, R. Lim, A. Meier, C. Plessl, I. Talzi, L. Thiele, C. Tschudin, M. Woehrle *et al.*, "Permadaq: A scientific instrument for precision sensing and data recovery in environmental extremes," in *2009 International Conference on Information Processing in Sensor Networks*. IEEE, 2009, pp. 265–276.

[48] O. M. Salim, K. M. Fouad, and B. M. Hassan, "Dual-level sensor selection with adaptive sensor recovery to extend wsns' lifetime," *HUMAN-CENTRIC COMPUTING AND INFORMATION SCIENCES*, vol. 12, 2022.

[49] W. Liu, G. Yang, H. Jia, and J. Wang, "A novel uasb-mfc dual sensors system for wastewater treatment: On-line sensor recovery and electrode cleaning in the long-term operation," *Chemosphere*, vol. 246, p. 125751, 2020.

[50] D. of Meteorology, *Sri Lanka Journal of Meteorology*, vol. 3, 2018.

[51] K. Khanchuea and R. Siripokarpirom, "An iot-enabled platform for power consumption measurement of low-power wireless sensor devices," in *2018 2nd International Conference on Engineering Innovation (ICEI)*. IEEE, 2018, pp. 57–62.

[52] T. P. D. Pieris, K. V. D. S. Chathuranga, A. L. Kulasekera, P. Guha, and P. Mukhija, "Energy and power consumption analysis of a wireless sensor node without a voltage regulator," in *2020 5th International Conference on Information Technology Research (ICITR)*. IEEE, Dec. 2020.

[53] B. Global, "Early Forest Fire Detection Using Sensors — bosch.com," https://www.bosch.com/stories/early-forest-fire-detection-sensors/, [Accessed 28-04-2024].

APPENDIX A

ARDUINO CODE OF THE SENSOR NODE

1. Sensor_Node.ino

```
#include "temp_humidSensor.h"
#include "data_transmitter.h"
#include "LowPower.h"
#include <Wire.h>
#include <DS3231.h>

temp_humid tempHumidSensor;
data_trans transmitter;

DS3231 clock;
RTCDateTime dt;
boolean isAlarm = false;
boolean alarmState = false;

unsigned long last = 0;

int counter = 0;

unsigned long starts = 0;
unsigned long stops = 0;

int packets = 0;

void wakeUp()
{
  isAlarm = true;
}

void setup() {
  pinMode(9, OUTPUT);
  digitalWrite(9, HIGH);
  delay(1000);
  digitalWrite(9, LOW);
```

```
  Serial.begin(9600);
  clock.armAlarm1(false);
  clock.armAlarm2(false);
  clock.clearAlarm1();
  clock.clearAlarm2();

  clock.setOutput(1);
  pinMode(2, INPUT_PULLUP);
  attachInterrupt(digitalPinToInterrupt(2), wakeUp, FALLING);
}

void loop() {
  transmitter.sendData();
  if (isAlarm)
  {
    alarmState = !alarmState;
    clock.clearAlarm1();
  }
  clock.setAlarm1(0, 0, 0, 0, DS3231_MATCH_S);
  LowPower.powerDown(SLEEP_FOREVER, ADC_OFF, BOD_OFF);
}

boolean condition(int interval) {
  unsigned long now = millis();

  if ((now - last) >= interval) {
    last = now;
    return false;
  }
  else {
    return true;
  }
}
```

2. data_transmitter.h

```
#ifndef data_transmitter
#define data_transmitter

class data_trans
```

```
{
  public:
    void sendData();
    void getTime();
    void receivedTimeDataUpdate();
    void relay();
    void sleepTrans();

  private:
    const uint16_t this_node = 04;
    const uint16_t other_node = 00;

    const int channel = 90;

    bool ok;

    struct payload_t {
      float node;
      float temperature;
      float humidity;
      float altitud;
      float co2;
      float portValue;
      float batLevel;
    };
};

#endif
```

3. data_transmitter.cpp

```
#include "Arduino.h"
#include "data_transmitter.h"
#include "temp_humidSensor.h"
#include <RF24Network.h>
#include <RF24.h>
#include <SPI.h>
#include <Wire.h>
#include <DS3231.h>
```

```
RF24 radio(7, 8);
RF24 wirelessSPI(7, 8);

RF24Network network(radio);

temp_humid tempHumidSensorRF;

DS3231 clock1;
RTCDateTime dt1;

void data_trans::sendData()
{
  tempHumidSensorRF.setuping();
  tempHumidSensorRF.measureData();
  analogReference(INTERNAL);
  float sensorValue = analogRead(A3);

  payload_t payload = { this_node, (tempHumidSensorRF.temperatu
  RF24NetworkHeader header(other_node);

  SPI.begin();
  wirelessSPI.powerUp();
  radio.begin();
  network.begin(channel, this_node);
  network.update();
  radio.setDataRate( RF24_250KBPS );
  radio.setPALevel(RF24_PA_MAX);

  ok = network.write(header, &payload, sizeof(payload));

  if (ok) {
    Serial.println("Sent data");
    pinMode(9, OUTPUT);
    digitalWrite(9, HIGH);
    delay(100);
    digitalWrite(9, LOW);
  }
  else {
    Serial.println("Failed sending data");
    for (int i = 0; i < 3; i++) {
```

```
      pinMode(9, OUTPUT);
      digitalWrite(9, HIGH);
      delay(1000);
      digitalWrite(9, LOW);
      delay(1000);
    }
    delay(100);
    ok = network.write(header, &payload, sizeof(payload));
    if (ok) {
      Serial.println("Resent data");
    } else {
      Serial.println("Failed resending data");
      for (int i = 0; i < 3; i++) {
        pinMode(9, OUTPUT);
        digitalWrite(9, HIGH);
        delay(1000);
        digitalWrite(9, LOW);
        delay(1000);
      }
    }
  }
  tempHumidSensorRF.temperature = 0;
  tempHumidSensorRF.humidity = 0;
  tempHumidSensorRF.pressure = 0;
  wirelessSPI.powerDown();
}

void data_trans::getTime()
{
  SPI.begin();
  if (!radio.begin()) {
    Serial.println(F("Radio hardware not responding!"));
    while (1) {
      // hold in infinite loop
    }
  }
  network.begin(channel, this_node);
  network.update();
  radio.setDataRate(RF24_250KBPS);
  radio.setPALevel(RF24_PA_MAX);
```

```
  payload_t payload = {this_node, 1111, 1111, 111111,
  111, 11111, 11111};
  RF24NetworkHeader header(other_node);
  ok = network.write(header, &payload, sizeof(payload));
  if (ok) {
    Serial.println("Sent time request");
    pinMode(9, OUTPUT);
    digitalWrite(9, HIGH);
    delay(100);
    digitalWrite(9, LOW);
  }
  else {
    Serial.println("Request failed");
    for (int i = 0; i < 3; i++) {
      pinMode(9, OUTPUT);
      digitalWrite(9, HIGH);
      delay(1000);
      digitalWrite(9, LOW);
      delay(1000);
    }
  }
}

void data_trans::receivedTimeDataUpdate()
{
  network.update();

  while ( network.available() ) {

    RF24NetworkHeader header;
    payload_t payload;
    network.read(header, &payload, sizeof(payload));

    int y = payload.temperature;
    int mo = payload.humidity;
    int d = payload.altitud;
    int h = payload.co2;
    int m = payload.portValue;
    int s = payload.batLevel;
```

```
    clock1.setDateTime(y, mo, d, h, m, s);

    dt1 = clock1.getDateTime();

    Serial.print("The time updated: ");
    Serial.print(dt1.year);   Serial.print("-");
    Serial.print(dt1.month);  Serial.print("-");
    Serial.print(dt1.day);    Serial.print(" ");
    Serial.print(dt1.hour);   Serial.print(":");
    Serial.print(dt1.minute); Serial.print(":");
    Serial.print(dt1.second); Serial.println("");

    for (int i = 0; i < 5; i++) {
      pinMode(9, OUTPUT);
      digitalWrite(9, HIGH);
      delay(300);
      digitalWrite(9, LOW);
      delay(300);
    }
  }
}
```

APPENDIX B

SCHEMATIC OF THE SENSOR NODE

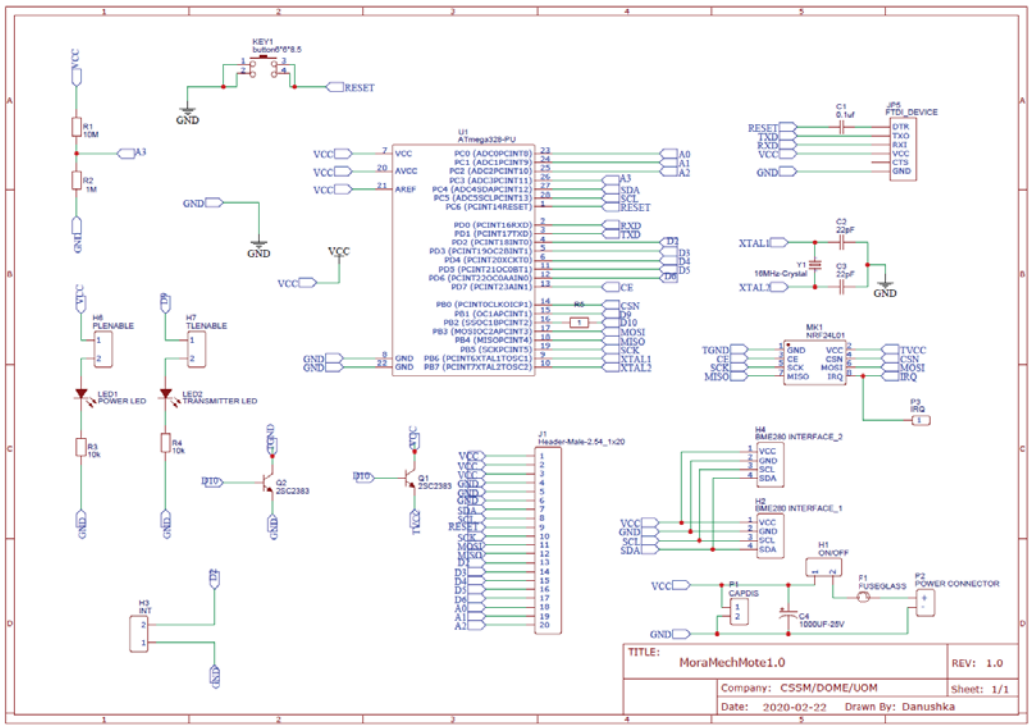

Fig. B.1: Schematic of the sensor node

APPENDIX C

PRINTED CIRCUIT BOARD (PCB) OF THE SENSOR NODE

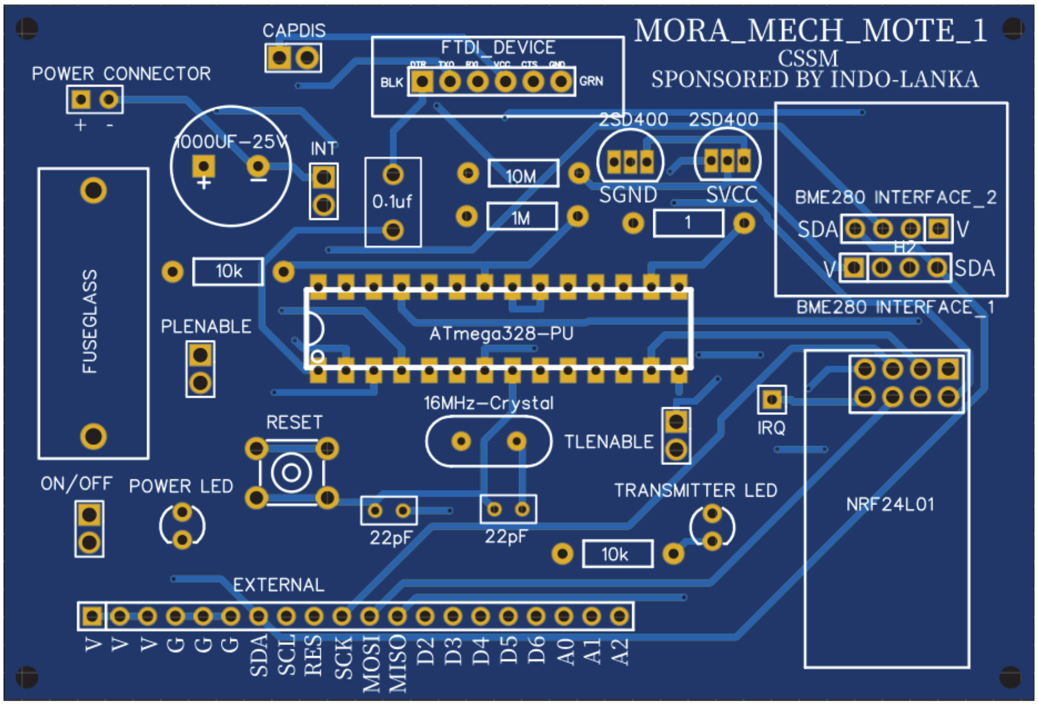

Fig. C.1: Printed Circuit Board (PCB) of the Sensor Node