UAV Routing Protocol (URP) for crop health management

Ammad Uddin Mohammad

THÈSE / UNIVERSITÉ DE BRETAGNE OCCIDENTALE

sous le sceau de l'Université Bretagne Loire

pour obtenir le titre de

DOCTEUR DE L'UNIVERSITÉ DE BRETAGNE OCCIDENTALE

Spécialité : Sciences et Technologies de l'Information et de la Communication

École Doctorale MATHSTIC

Presented by

Ammad Uddin MOHAMMAD

Prepared at Sensor Networks & Cellular Systems (SNCS) Research Center and ENSTA-Bretagne

UAV Routing Protocol (URP) For Crop Health Management

Thesis is defended on 19 December 2017 at ENSTA Bretagne

In front of the following committee :

Prof. Christian JUTTEN
University Joseph Fourier, President,

Prof. Yannick DEVILLE
University Paul Sabatier, Toulouse 3, Reviewer,

Prof. Denis HAMAD
University of Littoral Opal Coast (ULCO), Reviewer,

Prof. Stephane AZOU
National School of Engineers Brest (ENIB), Examiner,

Dr. Angelique DREMEAU
ENSTA-Bretagne, Examiner,

Prof. Ali MANSOUR
ENSTA-Bretagne, Director of Thesis,

Dr. Denis LE JEUNE
ENSTA-Bretagne, Supervisor

Abstract

The Kingdom of Saudi Arabia (KSA) is one of the largest countries in world having 2149690 Km^2 of area but not highly populated. The Kingdom is spending 24 billion dollars annually to import food items to fulfill the demand of 32 million residents. Although, it has potential to grow most of the crops within the country, the major hurdle is scarcity of water especially when underground water is not sweetened with the rare rainfall. KSA hardly manages to fulfill water demand for its industry and domestic use by going through an expensive treatment of sea water. After such an expensive treatment, only a small portion of treated water is spared for agriculture. Considering these issues, we initiated a project with the collaboration of KACST and University of Tabuk. The objective was to modernize the agriculture sector with state of the art technologies to get quality and quantity of food by utilizing precise amount of resources. Our project got more attention with the launch of KSA vision 2030, where the aim is to reduce the economy dependency on oil reserves. After the launch of vision 2030, agriculture is considered as one of sectors that needed to be developed on priority basses hence this project got more worth.

In this project, we developed a crop health monitoring system. We used all state of the art technologies to collect the actual data from crop field that will be processed further to take appropriate timely action which ultimately leads to optimize the resources. To achieve the desire goals, we harness IoT (Internet of Things) and drones in Saudi agriculture to establish instant deployment. Out project focuses mainly on two dimensions, first is data collection from crop field by using clusters of heterogeneous IoT devices and other is localization of these IoT devices for data harvesting. Formation of clusters by considering the path of UAV, sensors heterogeneity, weather harsh condition, fluctuation of sensors nodes, cost of IoT devices, etc. addressed in this research. Furthermore, carrying of larger a heavy localization system including an array of antenna and receiver by a small size UAV was an issue related to localization taken in consideration. Hence, we introduced a dynamic clustering and virtual antenna array to develop a complete data collection scheme. The proposed system is also validated through experiments including simulation models and test equipment the results obtained were encouraging especially in term of deployment time, energy efficiency and throughput.

To Dr. el-Hadi M. Aggoune (My Inspiration)

Acknowledgement

I am very grateful to Almighty ALLAH Subhanahu Wa Ta'ala for all His favors and blessings particularly enhancing my courage to write this dissertation. I would like to thank my parents, wife and children for what they have contributed towards my education. Their unconditional love, constant support, great advice and prayers over the past years are priceless. Knowing that my success in completing the Ph.D. would be a source of happiness for them, I was further motivated to work hard.

I would like to thank Professor Christian JUTTEN of University Joseph FOURIER for heading the jury being a president of my Ph.D. committee. Further, I would like to thank Professor Yannick DEVILLE of University Paul Sabatier Taulouse and Professor Denis HAMAD of University of Littoral Opal Coast for honoring me by accepting to be reviewers of my dissertation. I also want to thank Professor Stephane AZOU of National School of Engineers Brest, Assistant Professor Angelique DREMEAU of ENSTA-Bretagne and for accepting to be examiners of my dissertation.

I am deeply indebted to the director of my thesis Prof. Ali MANSOUR for his valuable encouragement, comments and timely guidance along supervising this doctoral thesis. He has always been tough but fair when criticizing my work and I greatly appreciate him for maintaining such a difficult balance. I will remember our long discussions we have had to improve this work as he always sat and listened me patiently no matter how tight his schedule was. I thank him for always being available and helpful.

I must also thank my supervisors Dr. Denis Le JEUNE and Dr. Muhammad AYAZ for the time they spare to evaluate my contributions, the problems they noticed and the discussions we had which helped me to proceed my work and results.

This work has been carried out at Sensor Networks and Cellular Systems (SNCS) Research Center, University of Tabuk. The financial support provided by SNCS is gratefully acknowledged. I would like to gratitude ex-President of the University of Tabuk, Dr. Abdulaziz Bin Saud ALANAZY (late); the President, Dr. Abdullah bin Mofreh AL-DHIAIBI; and Vice President of Higher Studies and Scientific Research, Dr. Rashed bin Muslit ALSHARIF for their kind support. I would also like to express my thanks to Professor el-Hadi M. AGGOUNE the director of SNCS, Dr. Sami Salah Al WAKEEL and Dr. Mohammed Al WAKEEL for their constant support and help during this entire journey.

Table of contents

List of Figures

List of Table

List of Acronyms

2D: 2 Dimensions

3D: 3 Dimensions

AAAS: American Association for the Advancement of Science

ADC: Analog to Digital Converter

AGI-STK: Analytical Graphics System Tool Kit

AIS: Automatic Identification System

AoA: Angle of Arrival

APM: ArduPilot Mega

ASP: Array Signal Processing

BIL: Band Interleaved by Line

CCA: Concentric Circular Array

CCH: Candidate Cluster Head

CDMA: Code Division Multiple Access

CH: Cluster Head

CM: Cluster Member

CSMA/CA: Carrier Sense Multiple Access with Collision Avoidance

CSTDMA: Carrier Sense Time Division Multiple Access

CTS: Clear To Send

CW: Clock Wise

CCW: Counter Clock Wise

DoA: Direction of Arrival

DSA: Displaced Sensor Arrays

DV-Hop: Distance Vector Hop

EDC: Error Detection Code

EMS: Electro Magnetic Spectrum

EO-1: Earth Observing 1

ESC: Electronic Speed Control

ESPIRT: Estimation Signal Parameters via Rotational Invariance Technique

EVD: EigenValue Decomposition

FIR: Far infra-Red

fps: Frames Per Seconds

FTSP: Flooding Time Synchronization Protocol

GA: Getaway Nodes

GMC: Grand Master Clock

GPS: Global Positioning System

HEED: Hybrid Energy-Efficient and Distributed

rHEED: RSSI based Hybrid Energy-Efficient and Distributed

HRTS: Hierarchical Referencing Time Synchronization

ICT: Information and Communication Technology

INRES: Institute of Crop Science and Resource Conservation

ICT: Information and Communication Technology

INA: Intermediate Navigation Agents

INRES: Institute of Crop Science and Resource Conservation

IoT: Internet of things

IR: Infra-Red

KSA: kingdom of Saudi Arabia

KACST: King Abdul Aziz City of Science and Technology

LEACH: Low-energy adaptive clustering hierarchy

LoS: Line of Sight

LTS: Lightweight Time Synchronization

LVDT: Linear Variable Differential Transducer

MCM: Million Cubic Meters

MIMO: Multiple Inputs Multiple Outputs

MIR: Mid Infra-Red

ML: Maximum Likelihood

MSSN: Mobile Sink Static Node

MSMN: Mobile Sink Mobile Node

MUSIC: MUltiple SIgnal Classification

MVDR: Minimum Variance Distortion-less Response

NA: Navigation Agents

NADC: Network Assisted Data Collection

NDII: Normalized Difference Infrared Index

NIC: Network Interface Card

NIR: Near Infra-Red

NLOS: Non-Line of Sight

NTP: Network Time Synchronization

OMNeT ++: Objective Modular Network Test bed in C++

PAA: Physical Antenna Array

RBS: Reference Broadcast Synchronization

RBTSP: Ratio Based Time Synchronization Protocol

PEL: Path Loss Exponent

PTP: Precision Time Protocol

RM: Rectifier Module

RGB: Red Green Blue

RSSI: Received Signal Strength Indicator

RVP: Rendez-Vous Points

PTP: Precision Time Protocol

RTP: Round Time Trip

RTS: Request to Send

RTT: Round Trip Time

SA: Smart Agriculture

SA-UAV: Synthetic Aperture UAV

SC: Slave Clocks

SINR: Signal to Interference plus Noise Ratio

SNR: Signal to Noise Ratio

SNCS: Sensor Networks and Cellular Systems

SOTDMA: Self-Organized Time Division Multiple Access

SSSN: Static Sink Static Node

SSMN: Static Sink Mobile Node

SVD: Signal Value Decomposition

SWIR: Short Wave Infra Red

TADCO: Tabuk Agriculture Development Company

TDMA: Time Division Multiple Access

TDoA: Time Direction of Arrival

TGR: Third Generation Revolution

ToA: Time of Arrival

ToF: Time of Flight

TSP: Traveling Salesman Problem

TSPN: Timing-Sync Protocol for Sensor Network

UAV: Unmanned Aerial Vehicle

USB: Universal Serial Buss

UCA: Uniform Circular Array

ULA: Uniform Linear Array

URA: Uniform Rectangular Array

URP: UAV Routing Protocol

VAM: Virtual Antenna Module

VCXO: Voltage Controlled Crystal Oscillators

VPA: Virtual Phase Array

VULA: Virtual Uniform Linear Array

Wh: Watt Hours

WSF: Weighted Subspace Fitting

WSN: Wireless Sensor Network

1

Introduction

1.1 Background

Nowadays, Smart Agriculture (SA) [1]–[4] is not only a luxury technology to ease the human life but it has rather become a necessity or even a compulsion to cope with rapidly increasing food demand by the growth of world population. With the passage of time the agriculture sector is facing more problems and greater challenges such as: falling land fertility, dwindling water reservoirs, and desertification. In addition, some of the wildlife losing their habitat and thus being pushed to the verge of extinction. Furthermore, arable lands are being replaced by urban population and industrial units at an alarming rate. Environmental pollution, excessive use of pesticides and contaminated water are as well additional factors compounding further the problems of agriculture.

Smart agriculture in the Kingdom of Saudi Arabia (KSA) is targeted in this research thesis. The agriculture sector in the KSA faces even greater challenges because of scarcity of water, very extreme climatic conditions characterized by high temperatures, dry air, dust /sand storms, vast expanse of desert and lack of infrastructure in very remote geographical locations. The only plausible solution to overcome the above-mentioned challenges successfully lies in making an effective use of modern tools and technologies in the context of classical agriculture such as: sensors and wireless communication, and Unmanned Aerial Vehicles (UAVs) [4] to ensure optimum usage of available resources in achieving better quality and higher yield of crops. Smart systems may have six different levels of intelligence:

1. Sensing: The ability to sense changes in its surrounding,
2. Self-Organizing: The capability of different sensing units to organize themselves spontaneously as per need,
3. Adapting: Smart systems should be able to adapt itself to meet any particular requirement in terms of smart agriculture,
4. Inferring: It basically refers to conclusion which is based on results and observations,
5. Learning: After observing the environment, the learning can be used to improve the methodologies used previously,
6. Anticipating: Smart sensors could have the capability to anticipate the change of the environment.

In this study, smartness in agriculture is achieved in four steps:

1. Deployment of heterogeneous sensor nodes in crop field: Wide ranges of heterogynous sensors are used to monitor different parameters related to crop, soil and environment.
2. Use of UAVs for smart agriculture: UAVs are utilized to build cheap, handy and instant communication infrastructure between sensing devices and end-user.
3. Clustering of sensor nodes for efficient data collection: Clustering is the most important aspect of our study. It is the process to arrange different sensing devices in groups according to geographical area, required data, path of UAV, communication limitation, similarities or any other criterion. Once available alive sensors arranged themselves in a cluster formation then the challenge becomes to select a node as Cluster Head (CH) which will collect all the data from neighboring nodes and transmit it to UAV. Selection of CH is a tricky part, the node having best specification and more suitable for UAV (Near to UAV path) has to be selected as CH.
4. Localization and data collection from field sensors by UAV: Localization is the technique used to explore ground sensing devices by an aerial UAV and connect them to collect data.

1.2 Thesis Contribution

The problem addressed in this research project is data gathering from a large number of sensors that are unable to establish an ad-hoc communication due to widespread deployment of heterogonous sensors, geographical constraints, weather conditions or power considerations. The whole proposed system is only composed of an UAV and field sensor nodes, no prerequisite like communication infrastructure, network updates in term of routes and clusters, special CH nodes for data gathering and knowledge of UAV path are required. All sensor nodes are considered as heterogynous, location unaware, cheap in cost and left unattended. The function of field sensor nodes is only limited to sense and sleep in every day routine life to preserve the sensor node's energy and prolong the network life as maximum as possible. In this regard, the high-power transmitter radio unit is always switched off and a very low power radio receiver always senses and waits for the UAV activation call. The life cycle of proposed system is composed of seven steps as shown in Figure 1.1:

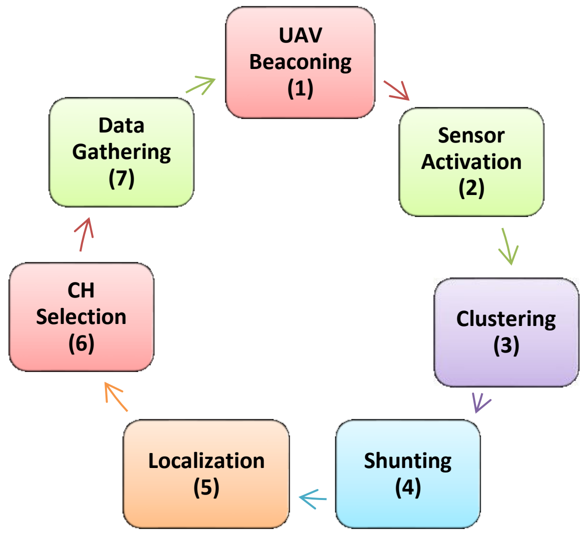


Figure 1.1: Proposed system life cycle.

UAV is the main part, acts like data mole and the mean of communication among the sensors. The system life cycle starts when UAV initiates the process of data gathering by sending a beacon message in step-1. Only selective sensor nodes that are addressed in this UAV beacon become activated in Step-2

Step-3 is clustering; clusters are made of selective sensors in the way of UAV to preserve its predefined path.

There is a possibility that none of the desire sensors have the capability to communicate with UAV because of limited resources, the other way around UAV may get many responses from activated ground sensors and unable to handle at a time. To tackle both of these conditions step-4 is introduced to select some reasonable amount of field sensors for further processing. We name this process "shunting" to push or pull some nodes from process to make them in a reasonable range.

Localization is the setep-5 to find out sensor nodes installed in crop field by UAV, a special light weight energy efficient antenna is designed for the said purpose.

Best node among all will be selected as CH in step-6. Many parameters like energy, antenna size, energy consumption rate and distance with UAV are investigated before selecting a node as CH.

The final step-7 is data collection in which CH collects data from all neighboring nodes and aggregated data is transmitted to UAV by using point to point dedicated link. This lifecycle keeps on continue till whole or selected area of the crop field is scanned and data is harvested successfully.

1.2.1 Dynamic clustering

Clustering is an important aspect to increase network lifetime and reliability. Many clustering techniques are proposed and they can be classified into 4 broad categories: Static sink static nodes (classical LEACH, HEED [5], [6]), mobile sink static nodes (Rendez-vous base routing [7]–[10]), static sink mobile nodes (cellular Network [11]), and mobile sink mobile nodes (ad-hoc routing [12]). This thesis focuses on mobile sink static nodes clustering fashion as all agriculture sensors are assumed to be static and we are considering a mobile UAV to collect data from crop field. For static sensor nodes, researchers are proposing predefined clusters and cluster head schemes to collect data. This type of clustering is not feasible in our case as large number of sensor nodes may become unavailable due to weather conditions or harsh environment (covered by sand, water or plant follicles). Situation becomes more critical if CH is included and whole network becomes un-functional. In addition, path of UAV is dynamic and sensor nodes are unaware of it; in this case, it rarely happens that predefined CH resides in the path of UAV and has good link to it. Network defined and Rendez-vous base clustering is also proposed in the literature, where all the nodes send periodic updates to maintain up-to-date CH or Rendez-Vous Point (RVP) from where UAV can collect data. In this situation, the main drawback becomes the overhead of all nodes to update CH continuously which results in battery drainage and reduces network life time. Besides, UAV should have to search and track network assigned CH/ Rendez-vous that will affect the throughput of the system and deflect the UAV form its path. As per our best of knowledge, none of previously published clustering schemes considers the UAV path as a clustering criterion.

In this thesis, we developed a dynamic clustering scheme. All the field sensor nodes initially considered indistinguishable (no potential CH), UAV sends a beacon message to activate all nodes reside in its vicinity, made a cluster by considering path of UAV and type of required data. The next step is to choose one node as the CH, merge whole cluster data on this point, locate and connect it with UAV at reasonable height and distance.

1.2.2 Dynamic cluster head selection

Once UAV assisted cluster is made, developed system will grade cluster nodes into two types: First type contains the nodes don't have capability to approach UAV called Cluster Members (CMs); Candidate Cluster Heads (CCHs) are the other type. The CCHs further shunted by the developed system to keeps them in range from *1* to *N* (*N* is maximum capacity of UAV to locate sensor nodes). All CCHs and UAV will collectively take part in selection process to nominate a node as CH. Many parameters (like remaining energy, available renewable energy, energy consumption rate, antenna size and distance with UAV) are considered in this selection process. In the proposed system, the tasks of cluster formation and the CH selection are conducted dynamically at runtime according to the context and then backbone reliable point to point connection is established between CH and UAV to collect all required data for further processing and decision making. The proposed dynamic clustering scheme is illustrated in Figure 1.2.

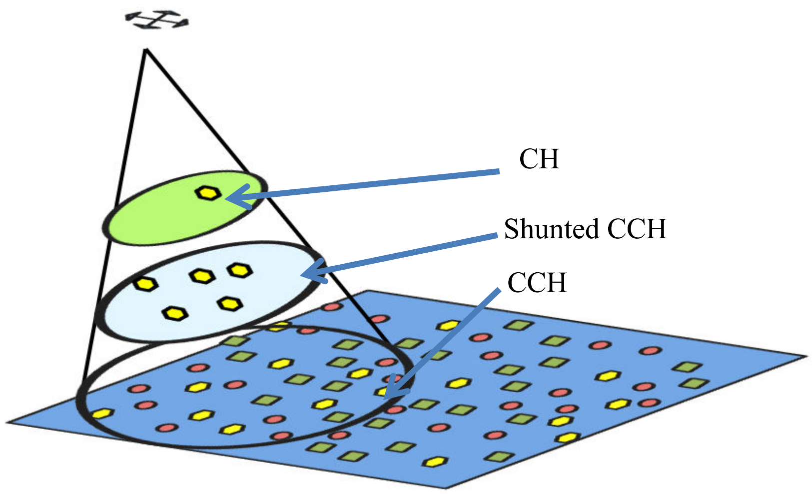


Figure 1.2: Dynamic clustering.

1.2.3 Virtual antenna for localization

In our proposed system, many ground sensor nodes are installed to monitor a crop field, soil and environmental parameters, and an UAV is used to harvest data from these sensors. To collect data in an efficient way, localization of sensor nodes by UAV becomes an important part. UAV should know the exact number and locations of sensor nodes to collect data in efficient way. Many schemes are proposed for the localization like [13], [14] but in all existing schemes multiple antennas are used to measure Angle of Arrival (AoA) for incoming signals to estimate the location of field sensors. However, the drawbacks of mounting such multiple antennas on UAV outweigh the benefits. The challenge is that adding multiple antennas and receivers on an UAV increases its weight and ultimately decreases its payload capacity, flight time, speed and agility.

In this thesis, we are proposing a new virtual antenna system where a single moving antenna can replace exactly the physical array of many antennas. This proposed virtual antenna system can operate at different precision levels and multi-frequencies which are very difficult to be implemented in a physical array of antennas. In physical antenna array the number of antennas and the spacing among them cannot be easily adjust to fit the required wave length and precision; especially if it is on onboard. The proposed virtual phase array antenna will be light weight and energy efficient as well, which will make it the best choice for UAVs.

1.3 Project Approval

This project is approved by King Abdulaziz City for Science and Technology (KACST), Riyadh and Sensor Network and Cellular System (SNCS) research center at the University of Tabuk, KSA in 2015 for 3 years. The whole project is conducted with the grant given by KACST and resources, facilities and expertise provided by SNCS. Project details and approval granted is provided in chapter 2 and Appendix G.

1.4 Thesis Organization

Whole thesis is organized in two phases clustering and localization of ground sensor nodes.

Chapter 2 is the state of art involved in both technologies clustering and localization. Initially in this chapter, potential of agriculture and challenges faced in KSA environment are discussed to prove the intensity of need for smart agriculture. Further survey of available sensor devices, UAVs, clustering schemes and localization procedures are given to describe how we can integrate and modify all the existing components to develop a complete agriculture monitoring solution.

After giving introduction of both phases (clustering and localization) in chapter 2, segregate chapters (chapter 3 and 4) are given to describe our contributions.

Chapter 3 is all about clustering, how we developed proposed dynamic clustering protocol and what is its performance comparing to other similar schemes.

Chapter 4 is about the localization process, in this chapter, we are proposing a new virtual antenna array system that will be mounted on a small sized UAV and will be used for localization of field sensor nodes.

Chapter 5 is more interesting and practical oriented. The way we developed UAV and proof of concept deceives to explain the working of developed system is given in this chapter in such a way that predecessor student can easily rebuild and enhance it in future. Conclusion of this thesis and suggestions for future work is given in Chapter 6. Finally, at the end of the thesis, we can find several appendices such as:

Appendix A Simulation Algorithm,

Appendix B AGI STK (Systems Tool Kit) simulation tutorial,

Appendix C OMNeT++ (Objective Modular Network Testbed in C++) Simulation for clustering,

Appendix D MATLAB Simulation for clustering,

Appendix E Proof of concept design and development,

Appendix F Agriculture Supporting WNS (Wireless Sensor Network) material,

Appendix G Evaluation and Approval of Project by AAAS (American Association for the Advancement of Science).

2

State of the Art

2.1 Introduction

Recent advances in microelectronic and micro electromechanical systems have produced new battery-powered sensor devices that have capabilities for detecting and processing physical information. These devices (nodes) can be connected to form a Wireless Sensor Network (WSN) that performs a variety of operations. WSNs provide sensing accuracy and fault tolerance. It can be deployed in harsh environments to provide continuous monitoring and processing capabilities. WSNs collect various types of data from a monitored area. Depending on the application, sensing parameters may include moisture, temperature, nutrients, and pollutants. Sensed information is carried over multi-hop from node to node to a Base Station (BS) for further processing and action taking.

Given the numerous benefits of WSNs, a case study is proposed for their potential implementation in the farming sector in Kingdome of Saudi Arabia (KSA). Tabuk is considered as one of the favorite site of agriculture situated in the northwestern region of KSA shown in Figure 2.1.

Figure 2.1: Tabuk region, Saudi Arabia [15].

Water utilization in Saudia is very critical as there is little permanent storage for it such as: reservoirs or dams. At the same time, the Saudi land is fertile and has the potential to produce several crops such as: wheat, dates, fruits, vegetables, flowers, and alfalfa. This case study focuses on use of WSNs to control resources (like water, pesticide and nutrients; however, water used for irrigation is the most important one) as well as for monitoring the quantity and quality of crops. In majority of crops, an excess of water may have negative effects, this fact motivated us for this project to monitor crop health to control resources especially water. This will help us to perverse resources to improve yield production and quality.

Proper usage of WSNs requires a deployment plan upon which it can operate with very little or no human supervision. WSNs are usually deployed in remote farms where weather and geographical conditions vary significantly. Despite all the problems and hardships, KSA has great potential for agriculture that need to be flourished with the help of modern technologies. Now, we are presenting the scope of agriculture in KSA, the challenges giving resistance to its bloom and the tools that can be used as resistance emollient.

2.2 Scope of Agriculture in Saudi Arabia

According to [16], Saudi Arabia imports almost 70 % of its food requirements at the cost of $24 billion per year. On the other hand, the KSA has large available un-populated area that can be used for agriculture but unfortunately still remains un-utilized or underutilized.

KSA is the 15th largest country in the world having area of 2,149,690 km^2 but it is 205th in world according to population density (persons/ Km^2) [17], [18]. Only 1.6% is inhabited and 2.63% is arable land of the total area, the rest largest part (85%, 1827236 km^2) is desert. This huge un-populated and un-arable area is not being utilized properly; the main hurdles for cultivation are the shortage of water, geographical landscape, adverse weather and atmospheric conditions. The most critical factor is the deficiency of water because major portion of mentioned area is desert having few reservoirs and low average precipitation per year on the other hand, ever-increasing demand of sweet water for fast growing population and industry superseded the demand of agriculture sector [19]–[21]. KSA is the world largest oil producing country but unfortunately its underground water is not sweet, therefore it cannot be used directly. Sweet water reservoirs are very limited, although 230 small and large dams exists but can amass only an estimated 1.138 km^3 of runoff annually [22].

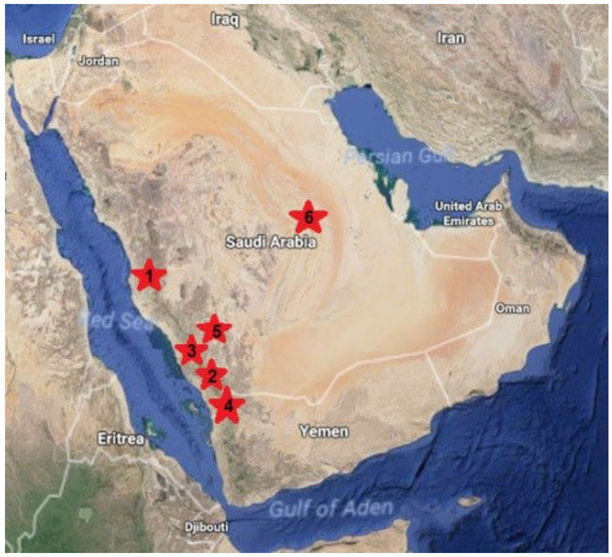
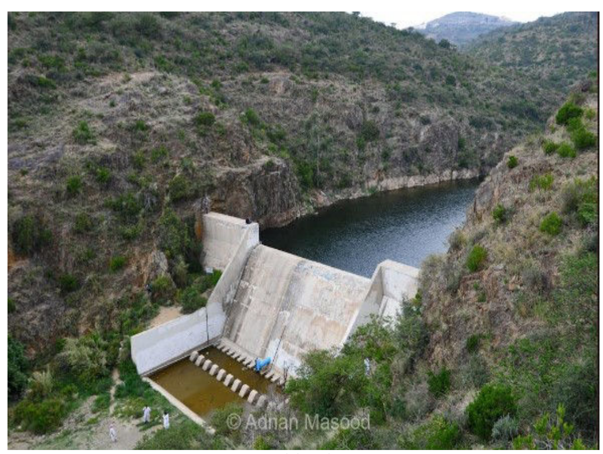


Figure 2.2: Famous dams in Saudi Arabia [15].

Some of the famous dams are indicated in Figure 2.2 and shown in Figure 2.3 [23], [24]:

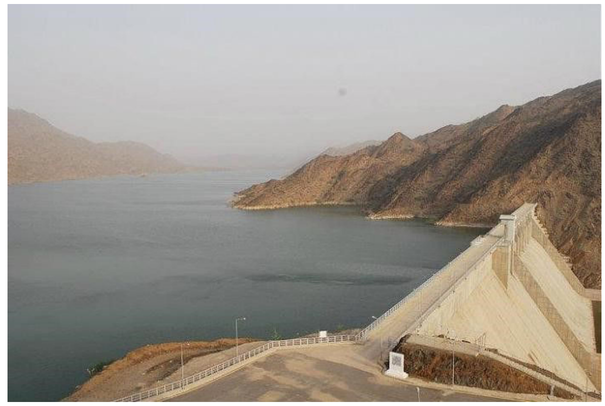
1- Murwani dam is in Makkah region having the capacity of 150,000,000 m^3,
2- Abha dam is in Abha capital of Asir province having the capacity of 2,130,000 m^3,
3- The Hali dam is about 14 km east of Keyad in Makkah having the capacity of 690,000 m^3,
4- Jizzan dam is about 16 km northeast of Jizan Province having the capacity of 51,000,000 m^3,
5- Bisha dam is the biggest dam in Saudia, it has the capacity of 325,000,000 m^3. Bisha is a south-western city of KSA, situated about 1002 Km from Riyadh and 212 Km from the city of Abha,
6- Namar Dam is situated near the capital Riyadh having the capacity of 12,000,000 m^3.

1- Murwani dam in Makkah.



2- Abha dam.

3- Hali dam in Makkah.



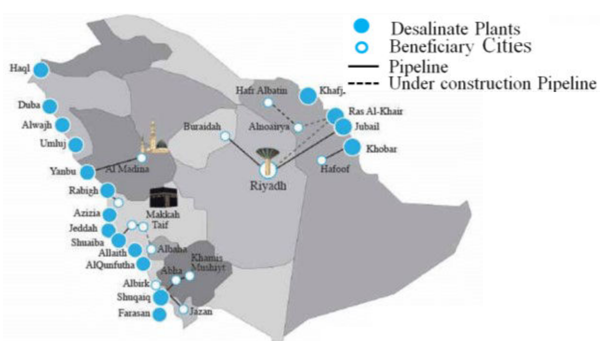
4- Jizzan dam.

5- Bisha dam.

6- Namar dam Riyadh.

Figure 2.3: Famous dams in KSA [15].

Twenty-five sea water desalination plants are constructed to recuperate sweet water resources, these plants are shown on map in Figure 2.4-A and a picture of largest one is show Figure 2.4-B. Jubail is an Industrial Zone at 481 Km from Riyadh in Eastern Province. Jubail desalinate water plant is the world's largest plant, supplying 70% of the country's drinking water as well as more than 28 million megawatts of electricity to KSA.

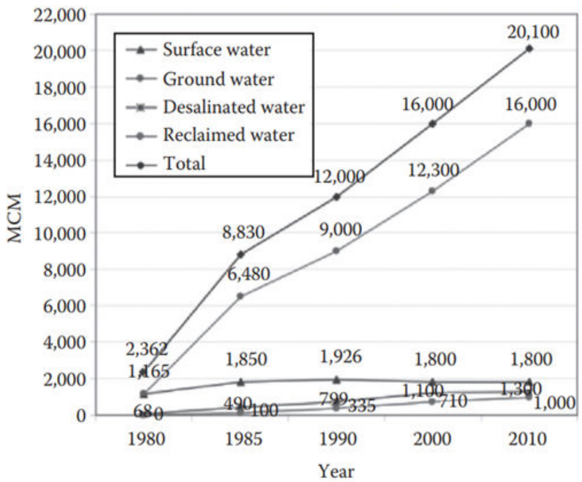


(A) Desalinate plants in KSA [25].

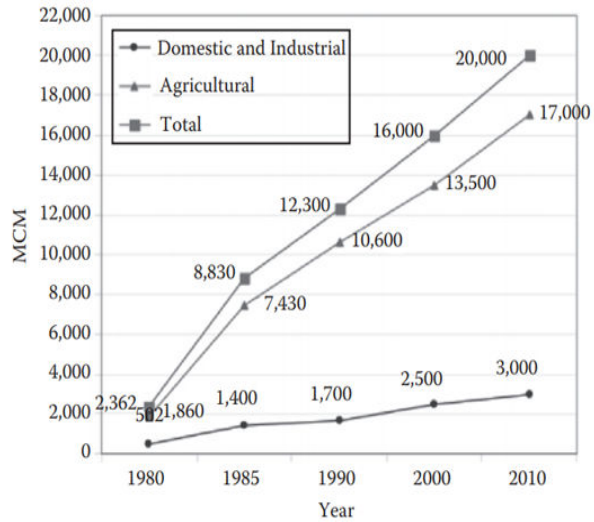
(B) World's largest desalination plant Jubail [26], [27]. Jubail is an industrial zone situated about 481 Km from the capital Riyadh in the eastern province.

Figure 2.4: Water desalinate plants in KSA.

Water demand and supply trends are shown in Figure 2.5-a and 2.5-B [20]. In 2010, the total water demand was 20 Km^3, while the total water availability from the different resources was 20.1 km^3. There is a narrow difference suggesting a low reserve margin, warranting a state of alert [28]–[30]. It is also shown in Figure 2.5-B that most water demanding sector is agriculture.



(A) 20100 MCM (Million Cubic Meter) water resources was available in KSA in 2010.



(B) In 2010, total water demand was 20000 MCM.

Figure 2.5: Water demand and supply trend in Saudi Arabia in 2010 [31], [32].

By exploring more natural and alternative water resources mentioned above, agriculture in KSA has been developed very fast in last 20 years as shown in NASA's satellite Figure 2.6. Agriculture in KSA has enormous potential including very large available area and very fertile soil. In addition, hot and dry environments are required to ripe well, most of the fruits. All these are the factors opt best to produce quantity and quality of crops. There are two parallel ways to utilize this potential properly by exploring more alternatives of fresh water and to utilize available resources in more precise way. In KSA, crops are grown in dispersed circular shaped (Figure 2.7) parcels to forbear the limited water resources and exposure of harsh environmental conditions including excessive heat or cold weather and sandstorms. Furthermore, the farming parcels have limited or no communication infrastructures.

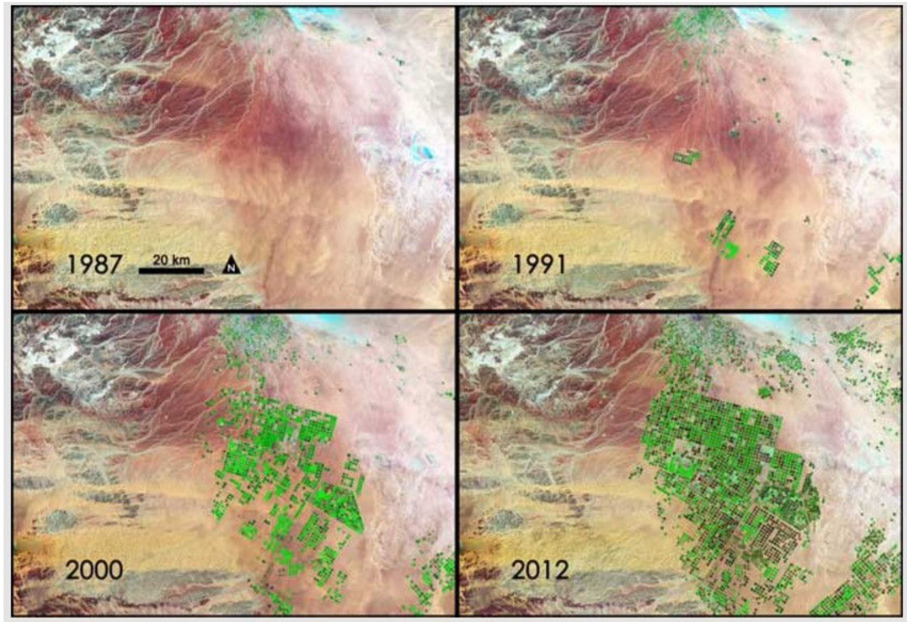
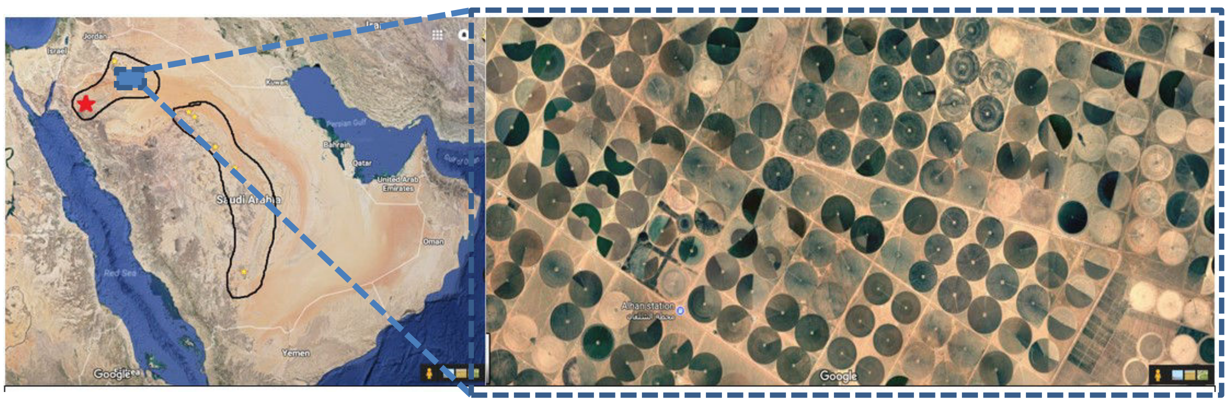
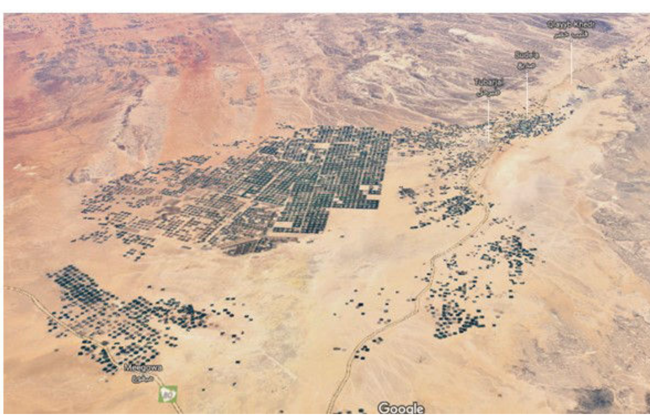


Figure 2.6: Historical data of Saudi agriculture from 1987 to 2012 [33].

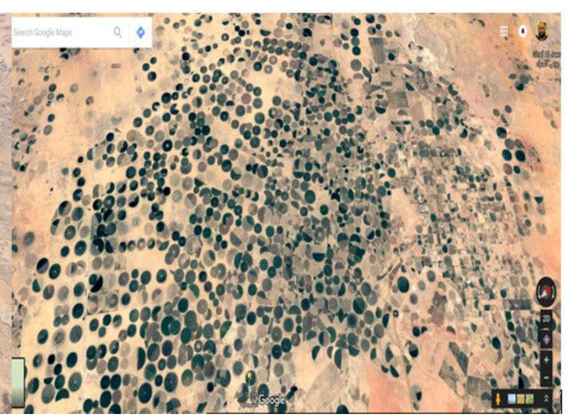


(A) Tabuk City indicated by a red star.

(B) Zoomed picture of farm fields.



(C) The agriculture site "Tubarjal" is the biggest town in Al Jouf region north of KSA. It is one of the largest agriculture site.



(D) Wadi Dawasar situated in Najad area of KSA which is the most famous for agriculture especially for the production of olive, tomato, potato and date. Many large food companies like NADIC and Watania are producing their items in this area.

Figure 2.7: Agriculture areas in KSA [15].

To cope with the scarcity of water and other challenges, there is a need to equip the agricultural sector with modern tools and scientific approaches that rely on WSNs to achieve sustainability. More recently, with the advent of unmanned vehicles and the accompanying progress in research and development in ad-hoc and vehicular communication, WSNs are positioned to gain further functionality as some of the nodes can become dynamic (carried by UAVs) facilitating both data collection and wireless communication in areas that are not equipped with fixed communication infrastructures. The next generation of agriculture is smart agriculture with less human interaction and more resource specific with precise monitoring.

2.3 Smart Farming

Smart farming is a Third Green Revolution (TGR) in the agriculture field, evolved with modern Information and Communication Technologies (ICT) following by plant breeding to genetics revolutions; TGR is taking over the agriculture sector by harnessing both ICT and IoTs (Internet of Things) in it, including sensors and actuators, geo-positioning systems, big data, UAVs, robotics, digital and wireless communication.

Smart farming has a real potential to grow more productive and sustainable agricultural yields, based on more precise monitoring and resource-efficient approaches. Farmer expect from smart farming to provide added services in the form of aid in decision making for better crop management. In this study, smartness in agriculture is achieved in 4 steps as described in chapter 1:

1. Deployment of heterogeneous sensor nodes in crop field,
2. Use of UAVs to establish a network for smart agriculture,
3. Clustering of sensor nodes for an efficient data collection,
4. Localization of field sensors by an UAV for clustering and data collection purposes.

Hereinafter, we elaborate each step.

2.4 Deployment of Heterogeneous Sensor Nodes in Crop Field

Sensors traditionally have four layers including: sensing, communication, control and power layers. Their main functions are to detect, monitor, and measure physical parameters such as: temperature, brightness, relative humidity, precipitation, sunshine, soil fertilizer/ moisture, speed and direction of wind, and fruit/ stem sizes. Heterogeneous sensors are required to be installed in a crop field; they may vary in size, resources and functionality. For example, some sensor nodes are so small (with very limited resources like power, processor, and memory) that can be installed on plant leaves. Other sensors can be big enough (having better resources) can be fixed on tree's truck to monitor its parameters. Large varieties of sensors are available for agriculture uses (details are provided in Appendix F or our survey paper [34]), and they can be divided into 5 categories:

1. Bug monitoring and control sensors using
    a. Photo sensor array device,
    b. Light spectrum analyzers,
    c. Sound devices;
2. Crop health monitoring
    a. Plant leaves,
    b. Plant stem and trunk,
    c. Fruit size;
3. Soil monitoring;
4. Environment monitoring;
5. Crop health monitoring by aerial view using multispectral imaging.

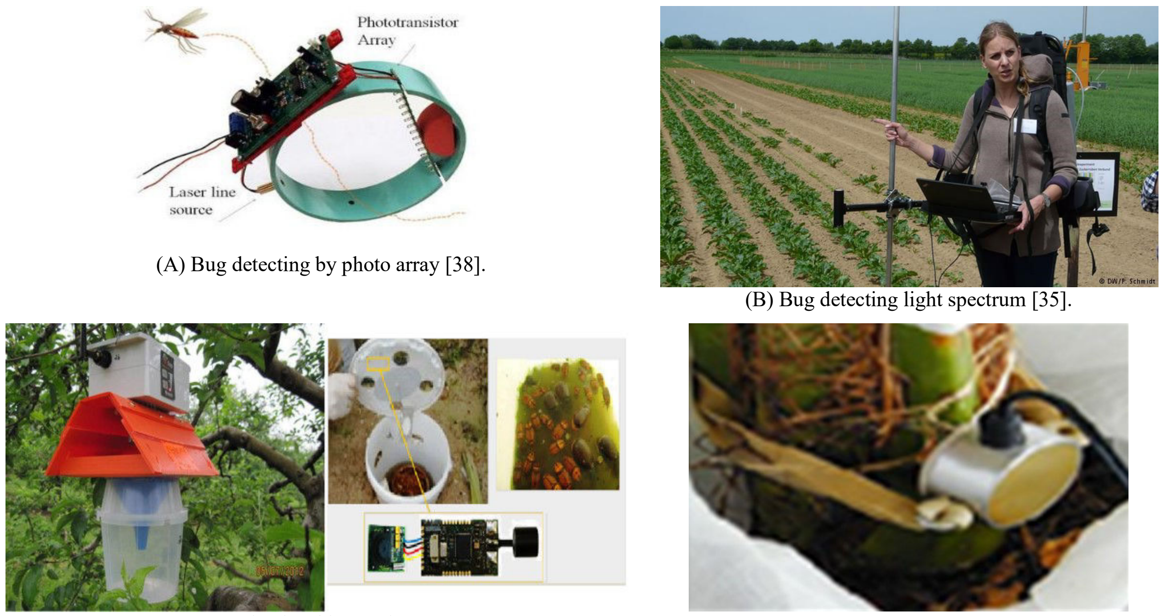
2.4.1 Bug monitoring and control sensors

Insects are responsible for two major kinds of damage to growing crops. First is direct injury done by eating leaves, fruits, roots or burrows in stems. There are hundreds of pest species like orthopterans, homopterous, heteropterans, coleopterans, lepidopterans or dipterans; All these species can damage the crop by eating it at different stages of their life in the form of larvae, pupa and adults. The second type of damage is indirect in which the insect itself does little or no harm but transmits some bacterial, viral, or fungal infection to the crop, e.g. the viral diseases of sugar beets and potatoes. Different types of sensors are available to detect the bug attack, such as:

A. **Bug detector sensor:** Detect and count number, type and sound of bugs passing by the sensor device, see Figure 2.8-A. An array of photo sensors and a microphone are used to monitor the number of bugs passing by using their wings beat pattern or bugs' sound that will be further used for bug classification.
B. **Light sensor:** In this technique, different wavelength combinations including visual light, infrared and ultraviolet are used to detect bugs, as it is well established that different combinations of wavelength have different impact on healthy green leaves and infected/ rotten leaves (Figure 2.8-B) [35].
C. **Bug Visual Inspection:** It is very common and mostly used technique to inspect type and number of bugs in the crop. In this way, bug traps are being hanged in crop and some smelly material or sound device (generate the sound of opposite gender) is used to attract the insects inside the trap. These traps are made in such way that once bug get into the trap cannot escape. Conventionally bug traps are monitored physically by visiting and seeing it by eyes this hassle is now superseded

by placing digital cameras, applying digital image processing and transmitting data or decision by means of wireless communication, see Figure 2.8-C [36].

D. Bug detection by sound: The most difficult task is to detect a bug eating the plant from inside of its trunk, for this purpose sound devices are being used as described in [37] and shown in Figure 2.8-D.



(A) Bug detecting by photo array [38].

(B) Bug detecting light spectrum [35].

Automatic visual inspection [36].

(D) Acoustic sensor [37].

Figure 2.8: Different bug detection sensors.

2.4.2 Crop health monitoring

Timely and precise assessment of the crop health is critical in ensuring good agricultural productivity. Health assessment includes measuring quality of crop and stress associated with, for example, water deficiencies, insects, weed and fungal infestations must be detected early enough to mitigate it with precise amount of resources. This process requires continuous remote sensing by using intelligent devices. Crop heath can be monitored by assessment of leaves, trunk and fruit size.

A. Monitoring Crop health by plant leaf: Leaves are an important part of the plant. Leaf monitoring and assessment is important because effects of any disease or deficiency reflected on leaves characteristics very soon. Different wireless sensors can be installed on the leaf to monitor different parameters like humidity, thickness, water deficiency, temperature and color these sensed parameters can be transmitted wirelessly to remote side where this data can be analyzed to evaluate the plant health. Different kinds of sensors are made as per leaf size and parameters need to be monitored. Some leaf sensors are shown in Figure 2.9. Figure 2.9-A is leaf moisture monitoring sensor, Figure 2.9-B is Leaf temperature monitoring and Figure 2.9-C is multi senor can monitor humidity, temperature and light intensity at the same time. Affixing leaf sensor to a crop can conserve 20% or more water that is required to complete its life cycle. Besides using less water, this type of monitoring also leads to less energy and nutrients utilization as well.

(A) SG-1000 leaf sensor by Agri-biotech company agriHouse, inc to monitor water deficit in plant[39].

(B) LT-1M sensor is a subminiature touch probe that measures absolute temperature of a leaf in the range of 0 to 500C with the accuracy of 0.150C developed by phyto-sensor [40].

(C) DTU NanoTECH developed greenhouse sensor can monitor temperature, humidity and light intensity at 3 different wave lengths all together [41].

Figure 2.9: Leaf monitoring sensors.

B. Monitoring Crop health by plant stem and trunk: Another way to monitor the crop health is by monitoring the stem growth rate and quantity of water / flux passing through. These monitored parameters can later be used to control water or other nutrients supplying to plant. Some of the developed sensors for this purpose are shown in Figure 2.10. (A) SD-5M sensor to monitor variations of stem diameter in micron range [42], (B) DE-1M dendrometer highly precise incremental LVDT[1] based sensor for monitoring micro-variations of trunk radius in micron range with an accuracy of 0.22 mm [43] (C) SF-3 measures sap flow in a small stem [43], (D) SF-4M/ SF-5M sensors are designed for monitoring relative variations of sap flow rate in a leaf petiole or small shoot [44].

[1] Linear Variable Differential Transducer (LVDT) is an absolute position transducer that makes accurate measurements with a good resolution and repeatability [184].

(A) SD-5M diameter sensor [42].

(B) DE-1M dendrometer [43].

(C) SF-3 sap flow sensor [43].

(D) SF-4M and SF-5M sensors for sap flow [44].

Figure 2.10: Stem monitoring sensors.

C. **Monitoring Crop health by fruit size:** As markets around the world became more particular about fruit size, and its ripeness especially when transport to long distance. Fruit price premiums result and huge financial benefits are related to send right size fruit to the right market at the right time. Fruit size monitoring sensors are shown in Figure 2.11 description is given in caption and more detail specifications are available in [45], [46]. These sensors track the fruit development throughout the season and provide the opportunity to adjust different management strategies like;

- Thinning strategies
- Irrigation strategy
- Growth and maturity regulators
- Selection of an exporter marketer.

(A) FI-SM small size (7-30 mm) fruit diameter monitoring sensor.

(B) FI-MM medium Size (30-190 mm) fruit monitoring (C).

Figure 2.11: Fruit size monitoring sensors made by Phyto-Sensor Group [45], [46].

2.4.3 Soil parameter monitoring

Soil is a natural resource which has been taken overlooked and for granted; to fulfill the massive demand of crops, there is a need to monitor the soil parameters from very early stage (land preparation) to the end (harvesting of fruits). All other resources like water and fertilizer are given to the crop according to the soil condition that can be helpful for economical production of quality and quantity of crops. Different types of soil parameters can help to control crop growth like temperature, moisture, CO_2 flux. Soil moisture monitoring sensor is shown in Figure 2.12 and details can be retrieved from [47], [48].

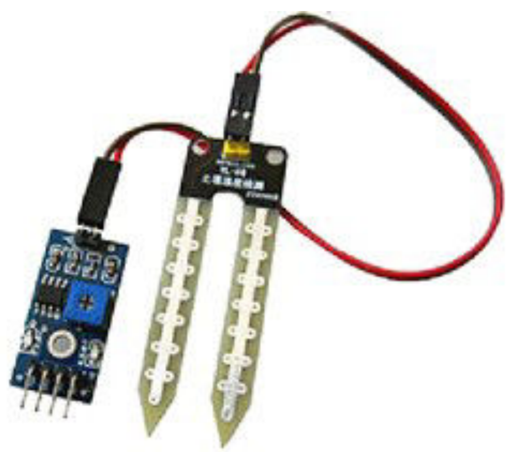


Figure 2.12: ESP-12 soil moisture probe [48].

2.4.4 Monitoring crop health by aerial view using hyperspectral imaging

Hyperspectral imaging is the study of amount of reflected light from a ground surface. Each pixel of the image is then assigned some numerical values (spectral radiance) by utilizing a range of wavelengths across the electromagnetic spectrum, including visible and infrared regions. These pixels are sorted and characterized through the use of specialized software and statistical analysis, to distinguish among different plant characteristics and environmental conditions.

Satellites, airbornes, and UAVs are used to carry visual light (Red Green Blue (RGB)), Infra-Red (IR), or thermal cameras to capture multi or hyperspectral images of crop fields to manage agricultural lands more precisely. The difference between hyperspectral and multispectral imagery is the spectral resolution, as hyperspectral imagery captures more narrow bands than multispectral. Light spectrum is divided into hundreds of small bands in hyperspectral imaging to get very detailed information of the reflecting surface but multi-spectral deals with only few bands. Every pixel has a complete spectrum in it and this can be used for a variety of applications including mineralogy, agriculture, astronomy, and surveillance. In agriculture, this higher spectral resolution enables us to distinguish between different crop characteristics, and to see details hidden from necked eyes.

During the last decade, spaceborne has become very popular to conduct geological survey using multi and hyperspectral sensing. Quickbird, Landsat, and Spot satellites are well-known satellite having sensors that can capture multispectral imagery. Multispectral sensors contain broad spectral bands ranging from the RGB, Near-InfraRed (NIR), Mid-InfraRed (MIR) and Far-InfraRed (FIR) of the Electro Magnetic Spectrum (EMS) are used in Landsat 5TM while SPOT 4 having spectrum ranging green to the MIR and the Short-Wave Infrared (SWIR). Earth Observing 1 (EO-1) [49] is an example of hyperspectral satellite covering the spectral range of 450-2600 nanometers (nm) consisting of 220 bands at a spatial resolution of 30 meters [50]. Table 2 reveals the main differences between multispectral and hyperspectral imaging.

TABLE 2-1: HYPERSPECTRAL AND MULTISPECTRAL CROP ANALYSIS [51]

Characteristics	Multispectral		Hyperspectral
Satellite	SPOT 4	LS5TM	EO-1 Hyperion
Lunching Date	August 2007	July 2008	August 2007
Number of Bands	4	6 & Thermal	220
Spectral Range	500-1750 nm	400-2400 nm	400-2600 nm
Spatial Resolution[2]	1165m	68 m	30 m
Swath Width[3]	60 km	185 km	7.5 km

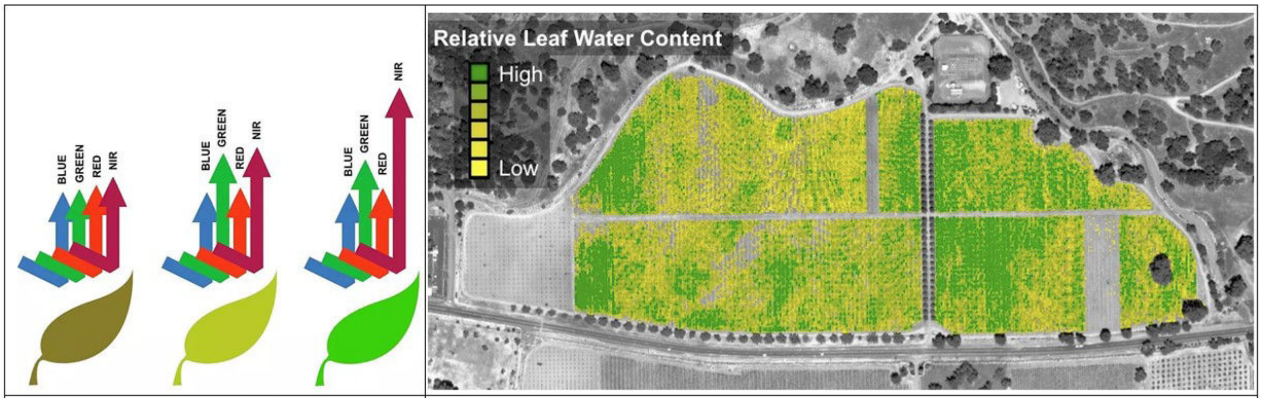
[2] Spatial resolution refers to the sharpness level of spatial detail shown in an image. It is the measure of the smallest object on the ground set by the sensor representing a single picture element (pixel) in the image. As a result, distance is associated with pixel size describing the side length of a pixel. Thus, the finer spatial resolution is associated with a smaller distance.

[3] Swath width refers to the width of any repetitively cut or treated strip. For example, the strip of the Earth's surface from which geographic data are collected by a moving vehicle such as a satellite, aircraft or ship in the course of swath mapping.

Multispectral have more swath coverage then Hyperspectral while less details. Multispectral is mostly used to observer overall view for example to map forested areas, while hyperspectral imagery can be used to map tree species within the forest.

A lightweight Hyperspectral imaging system can also be mounted on a fixed wing aircraft or small size multi router UAV to get accurate data over very large areas. These systems can effectively monitor the health of crops by observing water contents, nutrient levels and presence of hard-to-spot diseases. Hyperspectral imaging has enabled many advances in smart agriculture and has provided access to difficult approaching areas such as: snow, swamps, and mountainous regions. It has a great potential in all crop applications such as: area-wide crop management (like weed control or detection of insect damage), crop monitoring for nutrients, water-stress, disease, overall plant health, characterization of soils, vegetative cover and yield estimation. It provides farmer to rely on site-specific management tactics to maximize yield production while minimizing utilization of resources and impact on environmental. It can pin-point areas requiring attention like watering, weed or pathogen treatment, or nutrient adjustments and make it possible to perform spot applications rather than whole-field treatment. Some of the famous applications of hyperspectral imaging are:

A. **Water stress detection:** Hyperspectral imaging can be used to evaluate the health conditions of crops, in this regard, interaction and relationship between light spectrum and vegetation is very important. Different wavelengths of light may be absorbed, transmitted or reflected by the crop structure these phenomena can be interpreted into many vital crop parameters. To monitor general crop health, it is observed that a red and blue parts of the visible spectrum can be absorbed by the photosynthetic pigments. Reflection of Mid-infrared (MIR) is the primarily influenced by water content. NIR is influenced by the shape and condition of air spaces in the spongy mesophyll and significant reduction in reflection of mid-infrared spectral indicates some kind of stress (nutrient stress, pathogen and insect infestation). Some other crop health indicators like fungal pathogens, excess salt and nutrient deficiencies can be estimated using vegetation index of IR and visual light spectrum. An example of visual light and NIR behaviors on various nature of a leaf is shown in Figure 2.13-A. Figure 2.13-B shows the water stress in crop field measured by hyperspectral imaging.



(A) Behavior or different wavelengths of light on plant leaf.

(B) Normalized Difference Infrared Index (NDII) was used to estimate relative leaf water content at 1649 nm wavelength [52].

Figure 2.13 : Hyperspectral imaging for water deficiency analysis.

B. **Crop pathogens monitoring:** Plant pathogens are the major cause that may destroy whole crop in few days. Plant pathogens can be prevented if early detected. In case of delay, effectiveness of recovery and quality of yield may decrease and cause more expense. Hyperspectral areal imaging is a useful tool to avoid or minimize the effect of different pathogens. Infect, it can detect decrease reflectance for blue and green regions of the visible spectrum caused by infection and a strong decreased in near-infrared reflectance is a typical signature of leaf rust infected wheat. Apple fruit yield commonly damage by venturia inaequaliss, Orange rust (caused by Puccinia kuehnii) is a threat for sugarcane, Phytophthora infestans is a major risk to tomato and potato production; all these and many more diseases can be detected and classified even before symptoms become visible to the human eye by using hyperspectral imaging with varying vegetation indexes, further details are available in [53], [54].

C. **Crop nutrients stress estimation:** Hyperspectral or multispectral imaging technology can capture various symptoms caused by nutrient stress in plants; include both heavy metal contamination and deficiencies in nutrients of the soil. Many different techniques are being proposed to detect different types of nutrients for example in [55] and [56]: zinc deficiency and mercury contamination/ toxicity are measured by monitoring the symptoms in different crops including Bahia grass and mustard plants. The authors of [57] describe how hyperspectral imaging can be used to estimate nitrogen and phosphorous concentrations and condition of yield under these nutrient stresses.

D. **Analysis of Soil Properties:** Hyperspectral and multispectral imaging can also analyze and map soil characteristics even if it is under vegetation, which improves precision agriculture technologies and enhances capabilities. In [58], soil properties like moisture, temperature, salinity are determined using hyperspectral. In [59], soil organic carbon is predicted in some areas of Australia such as: Tasmania, Jervis Bay Territory, Victoria, New South Wales, etc. In [60], several soil parameters including salinity was mapped in several locations in Israel areas like Alfula, Jenin Best-Shean, etc.

2.4.5 Issues and challenges in hyperspectral imaging

Although hyperspectral imaging is being used since last four decades and progressing very fast but have some pros and cons. Some of the important factors limiting its performance and acceptability and their replacements are given below.

1- Although we can estimate the impact and size of insects effected area by using aerial hyperspectral images but we cannot visualize the actual plant condition and type/ shape of bugs, which is also equally important to apply more specific pesticide. Use of broad spectrum general purpose pesticides may kill useful insects which badly damage pollination process, as results manual and artificial pollination are increasing. Some bug detector sensors, traps or digital cameras installed in crop field can fix this issue.

2- Insect species like Phyllium or Phylliidae shown in Figure 2.14 are exactly the same color and shape of leaves and light waves reflected the same pattern as reflected by leaves. These kinds of bugs are difficult to be monitored by hyperspectral imaging. The solution is again visual inspection by some digital cameras. Some images of this family of insects are shown in Figure 2.14 A, B and C [61].

(A) Phylliidae insect on a leaf. (B) Phylliidae insect exists in different colors matching with crop lifecycle. (C) Phyllium celebicum inset.

Figure 2.14: Different Phylliidae family insects in picture [61].

3- If bug attack is in initial stage and crop is not yet affected: in this case, hyperspectral images are not much useful; while, crop sensor can generate alarm at very early stage.

4- Some bugs eat plant stem or trunk from inside which is very common in date and palm trees; in this type of bug attacks, there are no symptoms unless tree is completely dead but it is possible to detect these attacks at early stages by using sound sensors.

5- It is difficult and costly to acquire, store and process hyperspectral historical data to see disease, bugs and deficiencies spreading trends, that can be further analyzed to foresee something. Sensor with logging facility can maintain months even years of data that can be taken with a single contact which requires less storage, processing, computation and cost.

6- Individual plant level measurement is not possible like water circulation in plant stem, thickness and temperature of leaf, fruit size, etc.

7- Images capturing time is also critical in hyperspectral imaging to make accurate estimations of yields.

8- Large data storage requirement, intensive image processing, expensive equipment (like high profile computer and camera) and more technical expertise are additional challenges faced in hyperspectrum crop monitoring. It is extremely expensive exercise to purchase equipment, hiring developers, and analyzing the terabytes of data which is mainly suitable for large research institutes.

The best solution to monitor a crop to get better yield lies in the combination of both techniques ground WSN and aerial hyperspectral imaging. In our project, ground WSN are used to monitor real time data and generate early warnings to prevent crop from huge damage. Besides that, aerial view imaging can be also used to see the overall effect and pattern of diseases spreading, bugs or any kind of deficiencies.

2.4.6 Categories of crop sensors

Varieties of sensors are available for crop monitoring; they differ in size, monitoring parameter, reliability, sensitivity, sensing mechanism and cost. We categorized crop sensors into four sections according to data size and their power consumptions.

Catagory 1. Small sized data and low power consumption sensors:

Sensors with few bytes of data to transmit are considered as cat-1 sensors. There are varieties of such type of sensors that can be installed in the field to monitor different attributes related to environment, soil, and crop like:

- Air temperature / humidity,
- Wind speed and direction,
- Soil temperature / humidity,
- Leaf thickness /color (chlorophyll),
- Trunk thickness / flux flow,

- Fruit size.

The output value range of each sensor can be a number between $[-127,+127]$ and 1byte is enough to represent it. If the considered sensor is taking five samples a day and has one-month buffer storage then each sensor may have up to 150 bytes of data to transfer to UAV. As per datasheet of sensor [40] average energy consumption of these sensors is about 0.21 Wh (watt hours) knowing that 1 Wh = 3600 Joule, we conclude that 0.21 × 3600 = 756 Joule. A small size battery can make these sensors alive for more than a year.

Catagory 2. Medium data size and power consumption sensors:

These sensors are bit complicated required more processing, energy and can produce large amount of data (5 to 10 MB) to transfer to the sink node (UAV). Some examples are given bellow.

Sound sensors: Sound sensors can be of two types: intelligent or simple. If the sound sensor is intelligent, then it can detect and process a sound, can generate and transmit alarm message; but simple sensor can record a specific wavelength of sound (of our interest) and transmit it to remote location for further processing. Intelligent sound sensors can be considered as small data size cat-1 sensor that have few data to transmit but require more processing capabilities and consume more battery power.

Bug monitoring and control is the most challenging task in crop farming. Bugs' damages can be massive, the control of that damage becomes tough if not detected in an early stage. In Saudi Arabia, the biggest crop yield is date. It is almost impossible to see the bugs or insects damaging the date tree from inside the trunks. For that reason, an acoustic sound device can be installed on each tree to monitor such type of activities. That acoustic devices can record any abnormal sound produced by bug biting inside the trunks, and transmit this sound file to a remote-control system. In our project, we assume that all these sensors have limited processing and battery capabilities, and they are unable to process or analyze different recorded sounds.

According to [37], frequency range between 100 Hz and 10 kHz is enough for the bug sound acquisition and the minimum specification of a sound file that can be used to detect bugs by its sound is as under

Sample Rate (Hz)	100 samples per second
Word Length (bits)	8 bits = 1 byte /sample
Channels	1 = mono
Bit Rate	$100 \times 1 \times 1 = 100$ bytes per Sec
File size in	$100 \times 60/1024 \cong 5.86$ KB per min

For a sound clip of 1 hour that required to be investigated, this sensor needs to transmit $5.86 \times 60 \cong 351$ KB to the BS through UAV.

According to [62], a typical energy consumption of this sensor node is about 2.25 Wh (8100 Joule).

Picture camera sensor: A camera is used to capture a snap of crop progress and as well as bug traces.

Many still cameras can be installed in the field to inspect the crop health by visualizing it or can use some image processing techniques to monitor any kind of deficiency or disease. A still camera can also be used to detect presence of insects, their traces or impact on the crop. An example is already shown in previous section Figure 2.8-C, where species and number of attacking bugs are monitored by installing bug collector buckets in the field. A bucket traps the bugs and camera mounted over it takes a snap of inside view, which required to send to the BS for further analysis. As described in [36], the specifications of acceptable image quality are:

Resolution	640×480 pixels
Byte per pixel (RGB)	24 bits =3 bytes
Size of image	$640 \times 480 \times 3/1024 = 900$ KB

According to [63], the visual characteristics of a crop do not change very frequently; therefore one sample per day is sufficient for the analysis purpose. As per specifications the energy consumption of this sensor is estimated as 0.381 Wh for 14 fsp = 0.381/14 = 0.0272 Wh (~98 Joule) when activated.

Catagory 3. Large data size and high-power consumption sensors:

It is also observed that crop yield may decrease by insufficient control of the production process and another source of yield waste is intruders, including human or animals. Video-surveillance is a solution to detect and identify intruders as well as to better take care of the production process. Video streaming cameras are an example of cat-3 sensors that required high energy consumption and large data transfer to the sink node (UAV).

A famous camera node CMUCam2 developed by Carnegie Mellon University, mounted over IMB400 main board is used in many applications such as: [64] and [65]. Specifications are given below.

Frame size	640×480 pixels
Color depth	24 bits = 3 bytes
Frame rate (frames per second fps)	25 fps (as the PAL standard of TV)
Clip duration	30 s

$$\text{Video File size} \quad 640 \times 480 \times 3 \times 25 \times 30 = 691200000 \cong 659MB \text{ MB}$$

Estimated size of a 30 second video clip is 659 MB and Average energy consumption is same as still camera in estimated as 0.381 Wh for 14 fps = 0.68 Wh for 25 fps (2448 Joule) when activated [66].

Catagory 4. Very large data size and high-power consumption sensors:

Hyperspectral and multispectral images are used in wide range of agriculture applications through all stages from soil preparation to crop harvesting. Different vegetation indexes are used for various purposes; a complete list (more than 500 indexes) is given in a database [67] published by Institute of Crop Science and Resource Conservation (INRES). A combination of variety of cameras (thermal, visual light, infrared, X-ray, gamma rays) are used to monitor different crop parameters and a large size of database and high processing power system is required to analyze these types of images. Specification of a hyperspectral camera is given in [68] and camera is shown in Figure 2.15.

Figure 2.15: BaySpec OCI™-F series hyperspectral cameras [68].

Spectral Range	400-1000 nm = 0.4-1 μm
Number of spectral bands	60, 110, or 220 bands
Spectral Resolution	10-12, 5-7, 3 nm
Spatial Pixel	20 × 40 pixels
Exposure Time	20 μs -1s
Frame Rate	Up to 60 fps
Power consumption	~ 3 W (USB 3.0 power)
Weight	~ 570 g
Price of a complete system	40000 to 50000 €
Raw data pixel size	2 bytes
File format	BIL Interleaves[4]

If a frame of image of specifications 1063 × 1520 × 220 ×2 (rows × cols × spectral bands × pixel size) is captured, it will take about 0.66 GB space to store. The detail survey and comparison of

[4] BIL image file (.bil), which means "band interleaved by line," is an uncompressed file containing the actual pixel values of an image. An image is a type of spatial data based on rows and columns, where a single piece of information is stored in each pixel (or grid-cell).

existing agriculture sensors, manufactures, technologies used and sensing parameters are given in Appendix-F.

2.5 Use of UAVs to Establish a Network for Smart Agriculture

UAVs have gained a lot of attention in recent years. Their use nowadays is not limited to defense or military applications rather, civilian applications have taken advantage of the advance scored in the defense sector. One can find very successful applications in such areas as forest, ocean, environment, weather monitoring, topography, rescue, safety and farming, etc. Lately, suggestions on the use of UAVs have included airplane inspection. The success of UAVs is due to their versatility. They can be very small, carry a customizable payload, and may not necessarily require takeoff or landing strips. Furthermore, they are becoming very affordable to the point where a group of them can be used as a swarm in a coordinated structure to take on a variety of participatory tasks or serve for redundancy and backup. UAV command and control, condition and capability in terms of self-awareness, situational awareness, self-organization, reconfiguration, and adaptation are well-established concepts. It is worth mentioning that today; a simple UAV in the markets can exhibit many of these capabilities with added features including controllability through wearable computing devices such as smart-phones. UAV command and control interfaces along with trajectory planning options are also available. One can select a group of UAVs on a computer screen, designate a mission, specify the payload, assign a path or a destination, and launch a real-time scenario. In [13], UAVs are used in order to minimize the excessive usage of pesticides and fertilizers in agricultural areas. The process of applying the chemicals is controlled by means of the feedback obtained from the WSN planted in the crop field. The UAV trajectory is adjusted on base of feedback from the sensors. In [14], UAVs are used to serve as a relay network to eliminate the disconnection of parts of a WSN and guarantee the delivery of data. In [15], cooperative Multiple Input Multiple Output (MIMO) techniques are used to support communication among static sensors in a sparse WSN and a relay network composed of UAVs in order to keep the WSN connected. In [16], a customizable virtual environment to display conditions and capabilities of unmanned vehicles is disclosed. The virtual environment is highly customizable that can be used to generate mission scenarios and edit with the help of planning and strategy based on real-time data gathering and processing. UAV technology has given a high-technology makeover to agriculture industry. UAV can be utilized in every phase of crop cycle like:

A. **Soil and field analysis:** UAV can produce precise 3-D maps for soil analysis that is useful in seeding and planting patterns making, irrigation and nitrogen-level management.

B. **Plantation:** UAV plantation systems can decrease planting costs by 85 percent. These systems shoot pods with seeds and plant nutrients into the soil, providing the plant all the nutrients necessary to sustain its life.
C. **Crop spraying:** UAV can spray pesticide to selected areas even selected plant that can save resources, environment and wildlife all together.
D. **Irrigation:** UAV can identify water deficit areas by using hyperspectral, multispectral and thermal sensors by calculating vegetation index and heat signature. This information is very useful to get quality along quantity of crop and to save water resources.
E. **Health assessment:** Healthy and sick plants can be identified using visible and near-infrared light inspection. Sensors carried by UAV can differentiate between healthy and bacterial or fungal infected plant as they reflect different amounts of visual and NIR light.
F. **Crop monitoring:** Inadequate, inaccurate, and outdated crop information can cause crop damage and waste of resources at the same time. UAV is the best solution for crop monitoring and can generate accurate data on timely basis. It will reduce the labor cost as well as increase the precision level of monitoring and amount of resources.

The above-mentioned applications clearly suggest that UAVs can be used in a meaningful way to augment WSNs capabilities. Furthermore, we postulate the following assumptions with regard to farming and especially KSA farming:

1. The agricultural parcels are disjointed and separated by roads and right of ways for utilities such as power transmission lines.
2. There are one or more WSNs in each parcel. Some of the nodes are coupled with actuators that control the flow of water, pesticide, or fertilizer.
3. The communication between WSNs is either nonexistent or very limited to be useful.

The suggested use of UAVs can provide following capabilities:

1. In most of the cases, field sensors are location unaware and an UAV is used to locate and identify sensor nodes,
2. Gather data from the WSNs,
3. Relay the information to BS computers residing in a control center,
4. UAV will assist the ground sensors to arrange themselves in the form of cluster and route their data to cluster head.
5. Provide help to select a node as cluster head.

Some of the best available UAVs that are being used for agriculture can be seen in Table 3.

TABLE 2-2 : UAVS AND THEIR SPECIFICATIONS USED FOR AGRICULTURE

UAV	Weight (Kg)	Speed (Km/h)	Range (Km)	Wind Resistance (Km/h)
senseFly eBee SQ [69] 	1	40 - 110	3	45
PrecisionHawk Lancaster 5 [70] 	2.4	43- 58	2	45
Phantom 4 quad copters [71] 	1.4	72	5	16

We developed a S500 Dragan flyer to help us getting real data in this research project by using local resources and expertise. Complete procedure to rebuild and customize this UAV is given in section 5.2. We equipped our S500 with an open source autopilot APM (ArduPilot Mega) 2.6 to control it with onboard developed routing and data gathering protocol. To evaluate the performance of the proposed system, the specifications of this UAV are mentioned below:

- Max Turn Rate: 90°/second,
- Maximum Speed: 25 Km/h,
- Minimum Speed: 0 Km/h,
- Maximum Altitude: 1.5 Km,
- Helicopter Weight: 2.5 Kg,
- Payload Capacity: 0.5 - 1 Kg
- Maximum Flight Time: Approx. 20 min (without payload), 15 min with 0.5 Kg payload.

The change in flight time with respect of varying battery power, payload, wind speed and atmosphere temperature, can be calculated using an interesting calculator published by [72]. UAV is performing a vital role in this research project, its main role is to collect vital crop parameters from a specific area and selected sensors. For that purpose, UAV has to activate the required sensor of that area and form them in a cluster shape to save energy.

2.6 Clustering of Sensor Nodes for Efficient Data Collection

Clustering is one of the important means to prolong the network lifetime of wireless Sensor Networks (WSNs). Clustering mean dividing wireless sensor nodes into virtual groups or arranging them in hierarchical structure and electing one of them as a Cluster Head (CH) for each cluster. All Cluster Members (CM) should send their data to the corresponding CH which forwards the aggregated data to the Base Station (BS). A major challenge in WSN clustering is to select a suitable node as the CH. Advantages of cluster based WSN over flat network are energy efficiency, better network communication, minimized delay, efficient topology management and so forth.

Wireless sensor networks are always highly resource constrained having limited power, storage, bandwidth, and computational capabilities. Therefore, in case of energy depletion in sensor nodes, it becomes inoperable and irreplaceable. Increasing network lifetime and sustainability are the key issues for the contemporary research areas in sensor domain. Normally, energy depletion is highly dominated by radio transmission and even more for far-distances. Clustering techniques increase sensor network lifetime by limiting the number and range of radio transmissions. In contrast, performance of the flat network degrades with the growth of network size because increasing the network size and control overhead are directly proportional to each other. We can classify WSN clustering techniques into four categories (see section 1.2.1): Static Sink Static Nodes (SSSN) routing where base station, CH and sensor nodes all assumed to be stationary; the second type is Mobile Sink Static Nodes (MSSN) data collection which is more of our interest because our project lies in this category. One of the famous MSSN technique is rendez-vous based clustering where some data collection centers are established to collect data by a moving vehicle considered as sink node. Another MSSN type of clustering is network assisted clustering where predefined network is scanned by some moving sinks to collect data. In both these techniques WSN assists moving sink node to navigate and collect data. There is also exists UAV assisted routing where UAV assist the network to farm clusters. A general overview of SSSN, MSSN and more specific UAV assisted data collection techniques are given in next section and more detail and technicalities are presented in clustering chapter number 3. The remaining two types of data collection; Static Sink and Mobile Nodes (SSMN) such as cellular system and Mobile Sink and Mobile Node (MSMN) like ad-hoc communication, are not considered in this thesis. As in both cases, sensor nodes are considered as moving and more focus is given to manage the clusters and data routes according to the predicted or monitored mobility pattern of sensor nodes. In our scenario, crop sensors are always static.

2.6.1 Static sink routing

A large number of research studies are conducted for this type of sink node. The most famous are Low-Energy Adaptive Clustering Hierarchy (LEACH) and Hybrid Energy-Efficient and Distributed (HEED) protocols [5], [6]. A detail survey is conducted in [73] where about 30 different types of protocols are discussed. All these protocols are being used from many years, but the issue is not accommodating the mobility of sink node.

LEACH was designed to prolong the network lifetime be rotating CH node so that node near to BS should not die faster. Cluster-heads can be chosen on a random based is;

$$CH_{prob}(n) = \begin{cases} \frac{N_c}{n-N_c\,(r \bmod \left\lfloor\frac{n}{N_c}\right\rfloor)} & if\ n \in G \\ 0 & otherwise \end{cases} \tag{2.1}$$

Where;

N_c is Number of desire clusters, n is total nodes in the area, r is the current round, G is the set of nodes that have not been taken as cluster heads in the last $\left\lfloor\frac{n}{N_c}\right\rfloor$ rounds, $\lfloor X \rfloor$ is the integer part of X, and R_n is a random number uniformly distributed between 0 and 1.

If, $CH_{prob} > R_n$ then that node becomes a cluster-head

The algorithm is designed so that each node must become a cluster-head in $\left\lfloor\frac{n}{N_c}\right\rfloor$ rounds.

LEACH-C (or LEACH Centralized) is a modified version of LEACH where the amount of energy of the node is also taken into account.

$$CH_{prob}(n) = \frac{P \times E_{n\_current}}{1-P\left(r \bmod \left\lfloor\frac{1}{P}\right\rfloor\right) \times E_{n\_max}} \tag{2.2}$$

Where, $E_{n\_current}$ is the current amount of node energy, E_{n_max} is its initial amount of energy and P is desire percentage of cluster heads.

HEED is another example of static sink routing; it was proposed to select CH in a field according to the amount of energy as compare to its neighboring. HEED was designed to achieve four primary goals to produce well-distributed cluster heads, make compact clusters, prolong network life-time by distributing energy consumption, minimizing control overhead and terminate the clustering process within a constant number of iterations/steps. The following are the assumptions for HEED protocol:

1) Each node performs neighbor discovery,

2) Broadcasts its cost to the detected neighbors.

3) Each node sets its probability of becoming a cluster head CH_{prob} as follows:

$$CH_{prob}(n) = \max\left(P \times \left(\frac{E_{resiual}}{E_{max}}\right), P_{min}\right) \quad (2.3)$$

Where P is the initial percentage of cluster heads among n nodes, while $E_{resiual}$ and E_{max} are the residual and the maximum energy of a node (corresponding to the fully charged battery), respectively. The value of CH_{prob} is not allowed to fall below the threshold P_{min}.

2.6.2 Mobile sink routing techniques

In static sink routing, some serious issues can not be prevented, such as: communication overhead one of the leading issue where sensor nodes have to relay whole data towards BS. The second issue is that the nodes near the BS become effected anyhow because these nodes are the only option to approch BS. In the presence of these issues, the performance of a static sink routing can not be enhanced much. The only solultion to overcome these issues is the use of mobile sink nodes. This type of sink nodes are capable to harvest data by visiting the sensor nodes, especially the use of UAV becomes a trend and emerging technology. The simplest example of data gathering with mobile sink is a direct contact. In direct contact, UAV has to visit individually all nodes in the network to collect data [74]. In this scheme, many ways can be adapted to optimize the network traversing such as: square Grid tessellation, triangle tessellation, Snake like traversal, Boundary traversal, Traveling salesman problem, etc. more details can be grabbed from [75] or [76]. As we have already mentioned, clustering is a better choice than the direct contact, therefore we will only focus on UAV supported clustering schemes.

We can divide UAV supported clustering in two broad categories according to the UAV control:

1) WSN clustering with controlled sink path. There are three possibilities for controlled sink path
    a. Fixed/static: In fix/static path, the sink always follows the same path that is known to all ground sensors.
    b. Controlled by WNS: In WSN controlled path, the sink has to follow the path instructed by ground sensors to collect the data from predefined CHs.
    c. Random. In random path, the sink has to search the CHs in the field to harvest the data where CM nodes can help to find it.

    In either case fixed, controlled or random sink can't go independently by its own.
2) WSN clustering with an independent sink path. In this category, sink node can move independently/dynamically to complete its mission. Little attention is given in this direction and our project also lice in this dynamic.

A. WSN clustering with controlled sink path: A famous mobile sink data collection technique is the Rendez-vous based clustering [7]–[9], [74], [77]–[82]. Rendez-Vous Points (RVPs) are fixed meeting points defined by the network, from where mobile sink can collect data. In [79], a public transport (bus, truck or vehicle) is used to collect data while passing by a remote location using fixed route (see Figure 2.16).

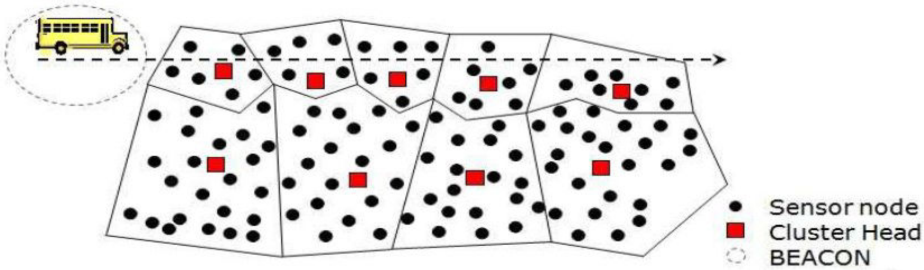


Figure 2.16: variable size clustering [80].

Variable sizes (small and large) clusters are made in the path of a moving vehicle in such a way that small size clusters are made closer to the vehicle path. Clusters are formed on the basic of Received Signal Strength Indicator (RSSI) value. As closer nodes utilize more energy to relay data, they are given compensation by reducing inter-classed communication of a small sized cluster. All the sensor nodes are considered as homogeneous and location aware plus path of UAV are also fixed and known in advance. Multiple rounds carried by multiple vehicles are required to make clusters and collect data. In the first round, conducted by the first bus, clusters are made, and in successive rounds data is collected. RVPs are also made in addition with CHs. Relatively centered node is selected as CH and the closer node to the sink path and having more energy is selected as RVP. The sensor nodes estimate distance by RSSI value from received beacon message. Suppose two sizes small and big $(R, \bar{R})$ of clusters are required to construct based on a threshold distance Dt from the sink. Each sensor node S_v estimates the distance $dist(S_v, MS)$ with a Mobile Sink MS and decides its cluster type as:

$$Type = \begin{cases} \bar{R} & if\ dist(S_v, MS) < Dt \\ R & if\ not \end{cases} \tag{2.4}$$

Each node S_v sends a broadcast message to inform all neighbors about its energy and cluster type. A node S_V maintains a list of tentative CHs when receiving a broadcast from the node S_u.

$$List_{sv} = \{tentative\ CH\ S_u \mid dist(S_u, S_v) < \max(S_u.Type\ ,\ S_v.Type)\} \tag{2.5}$$

A node S_u finds its maximum distance with sink and set its range, then broadcasts CH competition message. If the node S_v receives this message, it will include this node in its candidate cluster head $List_{uv}$ if it lies within its communication range as per equation (2.5). The node in the $List_v$ having the highest residual energy will be selected as CH. The next step is the selection of RVP, if $E_{resedent}$ is the remaining energy of the node, E_{max} is maximum energy in the system, the number of beacons received by the node is $n_b$and the average of RSSI values is $\frac{\sum_{i=1}^{n_b} RSSI_i}{n_b}$ than the node will calculate a cost value CH_{prob} as:

$$CH_{prob}(n) = \frac{E_{resedent}}{E_{max}} \times n_b + \frac{\sum_{i=1}^{n_b} RSSI_i}{n_b} \tag{2.6}$$

The node having more energy and remains in sink coverage for longer time will have highest CH_{prob}value and selected ad CH. These vehicles (public buses/logistic trucks) collect data during their journeys to destinations. In this scheme, the path of a vehicle is fixed and known to GPS enable sensor nodes. Another variation of this type is [83] Joint routing and navigation where UAV (sink node) has to follow the network assigned path; details are given in the 3rd chapter.

B. WSN clustering with independent UAV path: In most of the Rendez-vous based data collection schemes, path of mobile sink is controlled and it can't move freely. Few research studies have been conducted about UAV assisted routing and data gathering protocols like [84]. In this study, it is suggested that nodes far away from UAV and getting out of range are served at priority bases. In this case, long distance nodes always get priority over closer ones which will ultimately consume energy and produce fewer throughputs because closer node required less energy and can do fast communication. As expressed in Figure 2.17 where $R_{UAV(t)}$, $R_{UAV(t+1)}$, $R_{UAV(t+2)}$ are the list of sensor nodes at time $t, (t+1), (t+2)$ having priority 1, 2 and 3.

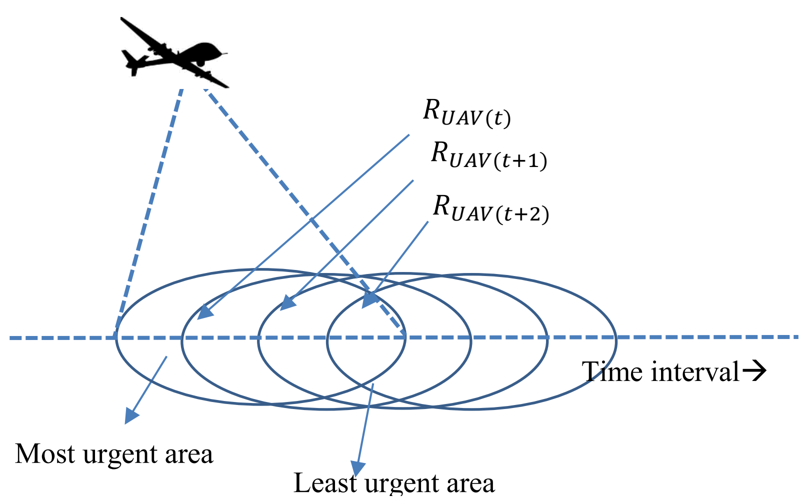


Figure 2.17: UAV assisted data collection [84].

Another example is given in [9], it is the extension of HEED called RSSI based HEED (rHEED) shown in Figure 2.18.

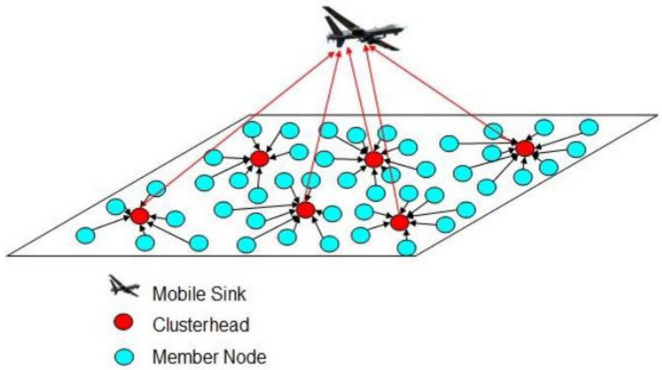


Figure 2.18: RSSI based HEED routing and data collection scheme [9].

In this scheme, sensor nodes are taken as unaware of self-location and UAV path. Sensor nodes i estimate distance with UAV based on RSSI value and computes a cost $Cost_i$ as:

$$Cost_i = \max(RSSI_i, \Phi) \tag{2.7}$$

Where Φ is the UAV beacon connection duration. Based in this $Cost_i$, each node computes a probability to become a cluster head similar to HEED protocol.

$$CH_{prob}(n) = \begin{cases} \max\left(C_{Prob} \times \left(\frac{E_{residual}}{E_{max}}\right), P_{min}\right), & if\ Cost_i > 0 \\ 0 & if\ Cost_i = 0 \end{cases} \tag{2.8}$$

In most of the MSSN routing schemes, moving sink either has fixed or controlled route for data collection. Little attention is also given to the UAV assisted routing and data collection where UAV can move independently and cluster are made accordingly. In all UAV assisted routing techniques, RSSI values are used to form a cluster and establish data transmission link. We have developed an UAV assisted routing technique where clusters formation and CHs selection depend on the health of sensor nodes, UAV path and the required data. In addition, localization by virtual phase array antenna is more precise as compare to RSSI value.

2.7 Localization of Field Sensors by UAV

The main function to establish a sensor network is to collect and forward data to destination. It is very important to know about the location of the nodes to collect data in efficient way. This kind of information can be obtained using localization techniques in WSNs. It is highly desirable to design low-cost, scalable, and efficient localization mechanisms for WSNs. There are many ways to find the location of a node like:

A. GPS Based and GPS Free: In Global Positioning System (GPS) based schemes; localization accuracy is very high but is very expensive as well in terms of cost and resources. Embedding GPS receiver into small size energy sensitive sensors, availability of GPS signal in very remote area, indoor environment and covered area are major problems of GPS based systems. In the scenario under consideration where sensor nodes are not changing its position, GPS receiver is an extra burden on it to monitor the same location every time. An energy efficient (311 Joule tracking and 389 Joule during acquisition) and highly sensitive (-165 dBm), PmodGPS module made by Digilent is shown in Figure 2.19 and its specifications are available in [85]. The average energy consumption of small size sensors (see section 2.6.1) is 756 Joule in this regard, size and power consumption of this GPS module is significant for this sensor.

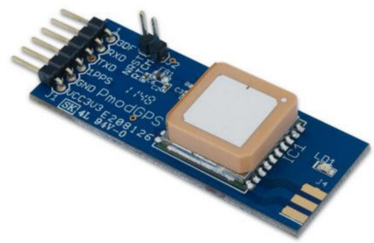


Figure 2.19: PmodGPS module made by Digilent Inc. [85].

In WSN, where energy is a strong constraint, GPS free localization is preferred and the simplest two model log path loss[5] and Walfisch-Ikegami [6]are described in [86] and [87], where localization is done by received signal strength and then applying a path loss model, to calculate the distances among wireless sensor nodes. The distances can also be determined based on the time-of-arrival, round-trip time and time-of-flight measurement of a radio signal. In some research studies, radio and ultrasound signal both are employed to estimate distance and location [88]. GPS enable and GPS free both schemes have some pros and cons in terms of complexity, weight / size of equipment, processing and energy efficiency.

B. Centralized and distributed localization: In centralized schemes, all information is passed to one central point or node, which is usually called sink node or base station. Computation cost of centralized based algorithm is less as compared with an individual node.

[5] The Path Loss Exponent (PLE) is a key parameter in the distance estimation based localization algorithms, where RSS is used for the distance estimation. The PLE is the rate at which the RSS decreases with distance, and its value depends on the specific propagation environment.

[6] Walfisch-Ikegami model is based on considerations of reflection and scattering of signal above and between buildings in urban environments. It considers both line of sight (LOS) and non-line of sight (NLOS) situations for localization.

In distributed schemes, sensor nodes individually calculate and estimate their positions by communicating directly with anchor nodes or estimate their own positions based of GPS sensing or through the measurement of distances to reference points. To carry out all these activities, more processing and energy are required at sensor nodes' end where the replacement of a battery or maintenance are almost not possible.

C. **Range based schemes and range-free schemes:** The distributed localization can be further grouped into range based and range-free. In the range based approach, information like Time of Arrival (ToA), Angle of Arrival (AoA), or time difference of arrival are required; while in alternative, absolute point to point distances are estimated to calculate the location of a certain point. In range-free algorithms, several location aware seed nodes are distributed in the whole WSN, these seed nodes periodically broadcast control messages with their location information. Sensor nodes estimate their own locations based on these control messages by observing the traces that left on these messages because of multiple hops and channel characteristics. Some basic concepts for localization are described in the next subsection.

2.7.1 Lateration

This method is based on the knowledge of reference point positions and distances to them. In this concept, intersection of multiple circles is calculated, in case of three circles called trilateration and more than three is known as multilateration. An example is provided in Figure 2.20 where, two reference nodes S_1, S_2 and their distance are known then position of node S_3 can be calculated as:
Let's assume two reference nodes S_1, S_2 are known.

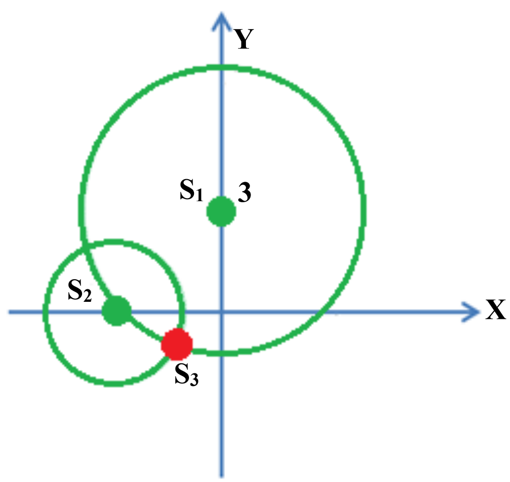


Figure 2.20: Lateration example.

Position of $S_1 = (0,3)$; radius $rS_1 = 4\ m$; Equation of circle S_1 is:

$$X^2 + (Y-3)^2 = 4^2 \Rightarrow X^2 + Y^2 - 6Y + 9 = 16 \tag{2.9}$$

Position of $S_2 = (-3, 0)$; radius $rS_2 = 2\ m$; Equation of circle S_2 is:

$$(X+3)^2 + Y^2 = 2^2 \Rightarrow X^2 + 6X + Y^2 + 9 = 4 \tag{2.10}$$

Position of the red circle S_2 is not known:

S_2 is the intersection of the two circles ⇒

Subtracting equation (2.9) and (2.10)

$$Y = -X - 2 \tag{2.11}$$

Substituting Yin equation (2.10)

$$2X^2 + 10X + 9 = 0 \tag{2.12}$$

By solving equation (2.12), we have two solutions for S_3:

$$(X_1, Y_1) = (-3.8, 1.8) \text{ or } (X_2, Y_2) = (-1.2, -0.8)$$

2.7.2 Angulation

In this mechanism, at least two angles of an un-localized node, from two localized nodes are measured to estimate its position. Example of this localization schemed is shown Figure 2.21 and calculation is as under:

Position of two nodes S_1, S_2 and the distance D between them is known. Point S_3 is the target and its position (X, Y) is required. Using simple geometrical rules, we can prove that:

$$S_3(X,Y) = \left\{\frac{D\ \cos(\alpha)\sin(\beta)}{\sin(\alpha+\beta)}, \frac{D\ \sin(\alpha)\sin(\beta)}{\sin(\alpha+\beta)}\right\} \tag{2.13}$$

We can mention the existing of two possible solutions: S_3 and $\overline{S_3}$

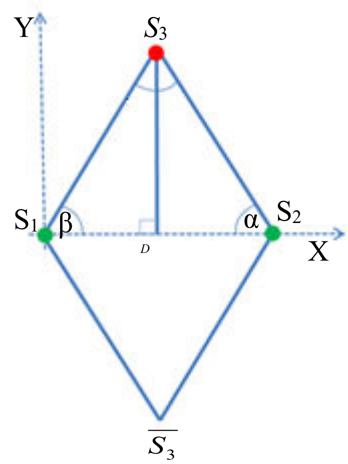


Figure 2.21: Angulation example.

2.7.3 RSSI based localization technique

RSSI based localization is discussed in many research studies like [89], [90]. It is a practical, self-organizing scheme addressing any outdoor environments. The limitation of this scheme is its power consumption because it needs to forward much information to the central unit. In RSSI based system, distance between transmitter and receiver is estimated by measuring signal strength and propagation loss at the receiver end. Distance between transmitter and receiver, and power of signal strength are inversely proportional to each other. If P_r is the power of received wireless signal, it can be represented as:

$$P_{r(d)} = \frac{P_t G_t G_r \lambda^2}{(4\pi)^2 D n^2 L} \tag{2.14}$$

Where P_t is the wireless signal transmitted power, G_t and G_r are the antenna gains of the transmission and receiving signal, L is the system loss parameter and D is the distance between the sending and receiving nodes.

Constant signal attenuation throughout travel distance is a strong assumption in the presence of fast/ multipath fading and shadowing effects. Furthermore, because of mentioned above factors accuracy of RSSI based system may reduce up to 50% which caused low acceptability of the scheme. In addition, obstacles between transmitter and receiver may cause extra signal delay/attenuation as compared to line of sight transmission, which results measurement of a larger distance than actual size. Channel path-loose due to environment is estimated by calibration which has to complete before starting localization. This type of deployment is not feasible in many cases like Warfield and dynamic clustering/ data collection, etc. All these mentioned above factors restricting the use of RSSI based localization and alternatives may exit like signal ToA, DV-Hop[7], etc.

2.7.4 Time of arrival

Time of Arrival (ToA) or Time of Flight (ToF) based localization is built on measuring the travelling speed and time of radio signals between anchor node and the target node [91]. TOA is the one-way propagation time of the signal travelling between a transmitter and a receiver. The measured ToA is then multiplied with the propagation speed to get distance between the receivers and the sender. A beaconing node sends signal at time t_1 to reference node and receive reply at t_2, distance D is measured as:

$$D = (t_2 - t_1)c \tag{2.15}$$

From this distance, a circle is made in such a way that receiver keeps in center and source must lie on the circumference in a Two-Dimensional (2D) space. Three or more circles are required to make distinct intersection point which represents the source position see section 2.7.1. The limitations of this scheme are:

1. Sender and receive much be accurately synchronized to measure the ToA information
2. Minimum three sensors are necessary for two-dimensional position estimate

The problem of synchronization between receive and sender can be fixed by using two-way ToA, where propagation time is estimated by using Round Trip Time (RTT) of the transmitted signal. Request-to-

[7] DV-HOP (Distance Vector Hop) is the localization scheme, where sensor nodes try to find their location by counting number of hops between them and reference nodes. Reference nodes broadcast location information packet in the network that is time stamped again at each hop and the target node can estimate its location from the reference by counting the hops and estimate, the delay in between.

Send (RTS) and Clear-to-Send (CTS) mechanism of CSMA/CA[8] is used avoid collision while accessing the shared medium to exchange ToA information between transmitter and receiver. Working of two-way ToA is shown in Figure 2.22.

Node S_1sends message at t_{1S_1} and receive reply at t_{2S_1}, and Node S_2 receives message at t_{2S_2}, process it and reply back at t_{2S_2}. Distance between node $S_1$and node S_2 can be calculated as follow:

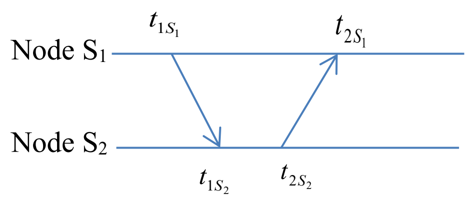


Figure 2.22: Two-way ToA.

let T be the shift time between Ref S_1 and Ref S_2.

$t_{2S_1}$and t_{2S_2} should be adjusted in the ref S_2 as follow:

$$T_{iS_2} = t_{iS_2} - T \tag{2.16}$$

As by equation (2.15), the distance between the two nodes Dn is given by $i = 1, 2$

$$D = \left(T_{1S_2} - t_{1S_1}\right)C = \left(T_{2S_1} - t_{2S_2}\right)C \Rightarrow \tag{2.17}$$

$$\frac{D}{C} = t_{1S_2} - t_{1S_1} - T = t_{2S_1} - t_{2S_2} + T \Longrightarrow \tag{2.18}$$

Then we can conclude that

$$T = \frac{t_{2S_2} + t_{1S_2} - t_{2S_1} - t_{1S_1}}{2} \tag{2.19}$$

Substituting T in equation (2.18).

$$\frac{D}{C} = \frac{t_{1S_2} - t_{2S_2} - t_{1S_1} + t_{2S_1}}{2} = \frac{\left(t_{2S_1} - t_{1S_1}\right) - \left(t_{2S_2} - t_{1S_2}\right)}{2}$$

Finally, we get the distance:

$$D = \frac{\left(t_{1S_1} - t_{2S_1}\right) - \left(t_{1S_2} - t_{2S_2}\right)}{2} \times C \tag{2.20}$$

[8] Carrier-sense multiple access with collision avoidance (CSMA/CA) is a communication protocol to access a shared medium in such a way, nodes attempt transmitting only when the channel is sensed to be idle, to avoid collisions.

2.7.5 Time difference of arrival

Synchronization problem faced in one-way ToA (Time of Arrival) is avoided in two-way ToA, another more accurate method is TDoA [91]. Where two signals having different velocities, are used to measure distance between sender and receiver. For example a radio signal traveling with c_1 speed, sent at time t_1 and received at t_2 , followed by an acoustic signal of speed c_2 sent at t_3 and received at t_4. In this technique, the time difference of arrival radio and ultrasound signal is used. Each node is equipped with radio transmitter, receiver, microphone and speaker. TDoA is described in Figure 2.23 and calculation is as:

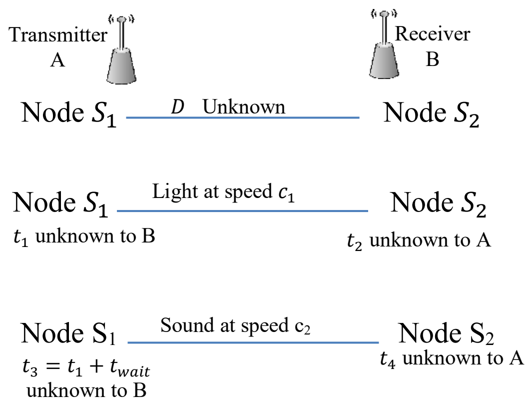


Figure 2.23: TDoA example.

For a light signal, we can write that:

$$D = (t_2 - t_1)c_1 \Rightarrow t_1 = t_2 - \frac{D}{c_1} \tag{2.21}$$

For an acoustic signal, we have similar equation:

$$D = (t_4 - t_3)c_2 = (t_4 - t_1 - t_{wait})c_2 \tag{2.22}$$

Putting value of t_1 from (2.21) in equation (2.22), we conclude:

$$D = \left(t_4 - t_2 + \frac{D}{c_1} - t_w\right)c_2 \tag{2.23}$$

The later equation can be simplified as follows:

$$D\left(\frac{1}{c_2} - \frac{1}{c_1}\right) = t_4 - t_2 - t_w \tag{2.24}$$

Finally, we can conclude that:

$$D = \frac{c_1 c_2 (t_4 - t_2 - t_w)}{c_1 - c_2} \tag{2.25}$$

Where, D is the distance between two nodes S_1 and S_2, that node S_2 wants to discover. c_1 and c_2 are the two different speeds belong to light and sound and are well known. $t_2, t_4,$ and t_w are the times S_2 sends radio signal, then transmitted sound signal and waiting time in between respectively. As all these parameters are known to node S_2 so it can calculate distance easily. The drawbacks are requirement of extra hardware and more energy consumption.

2.7.6 Angle of arrival

Angle of Arrival (AoA) is a method to find out the direction of propagation of a radiofrequency wave incident on an antenna array. Time Difference of Arrival (TDoA) at individual elements of the array is measured to determine AoA; from these delays, the AoA can be calculated.

Most famous method for AoA estimation is Multiple Signal Classification (MUSIC). Assuming M elements antenna array each separated by d distance is used for AoA, shown in Figure 2.24.

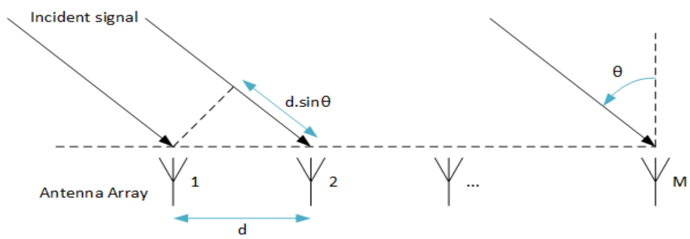


Figure 2.24: AoA method for localization.

A signal transmitted by a faraway source, is received by an antenna array in the shape of L plane waves. If we consider the first antenna element as reference then it is observed that signal received at the second element after traveling larger path by $(d\, sin(\theta)/c)$, which caused phase shift of $(2\pi d \sin(\theta)\, f/c)$. This phase shift difference, observed at m^{th} antenna is $(2\pi d(m-1)\sin(\theta)\, f/c)$. This regular phase changing phenomena at each antenna element is used to estimate the AoA of this signal.

In a classical MUSIC, M physical antenna elements can detect only $(M-1)$ sensor nodes only and carrying of multiple antennas and receivers by small size UAV is another issue. In addition, multiple antennas require more energy. In our project, UAV is the only mean of communication and data collection from a large size of many crop fields. In this scenario, we have to preserve UAV energy that will utilize to fly for longer periods and for data collection. In this research study, we are presenting virtual antenna array where a single antenna mounted over moving UAV is used to form a virtual linear array. The benefits achieved by developing this antenna are: small size, lightweight, energy efficient, multi frequency support, adaptive steering mechanism, adjustable number of antennas, reconfigurable inter-elements spacing and many more such as minimizing the effect of inter-element mutual coupling and induced current. Detail discussions about this new scheme are given in chapter 4.

2.8 Uniform Linear Antenna for Localization

The accurate estimation of Angle of Arrival (AoA) of a signal is very important in many applications such as those involving Radar [92], Sonar [93], Emergencies and surveillance [94], and cellular systems [95]. To find AoA, one should use a set of multiple antennas which is either fixed to form an array, or rotating in case of radar except some exception like multi cell static radar in aircrafts. An array of antenna system with having signal processing capabilities can be used to detect many parameters of the incoming signal including range, frequency, polarization and the most important is AoA. Array antenna system not only improves the resolution of AoA of incoming signal but also makes it possible to identify multiple sources that are emitting these signals. AoA can be described as the direction in terms of angle (azimuth θ and elevation Φ), created by multiple plane wave signals (narrowband or wideband) incident on a single or array of antennas.

Array geometry is another important factor in AoA estimation accuracy (resolution), which may composed of a set of antennas organized in a particular formation such as: Uniform Linear Array (ULA), Uniform Circular Array (UCA), Concentric Circular Arrays (CCAs), Uniform Rectangular Arrays (URAs), L-shaped array, V-shaped array, Displaced Sensor Arrays (DSAs), Parallel linear arrays and Y-shaped arrays for details see [96], [97] and [98]. Uniform Circular Array (UCA) is proposed in [96] to provide two-dimensional coverage and uniform performance in all azimuth directions at the cost of adding complexity as compare with ULA as shown in Figure 2.25. AoA is estimated using a rectangle geometry with 8 elements [97] and shown in Figure 2.26.

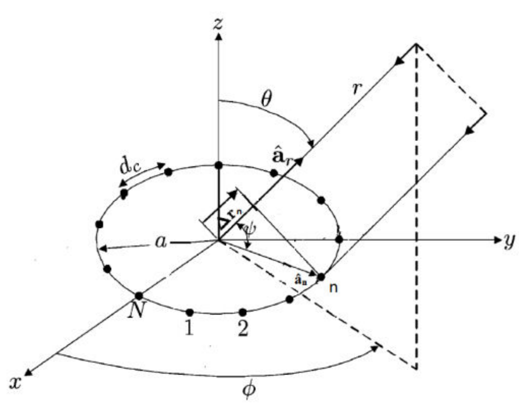


Figure 2.25: Uniform circular array [96].

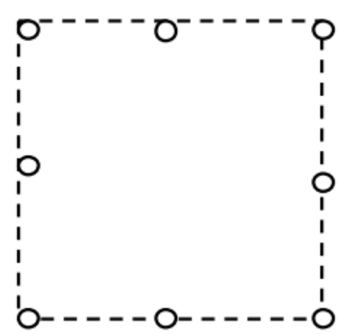


Figure 2.26: Rectangle shape antenna array [97].

ULA is the simplest possible array geometry working on narrowband signals, delivers acceptable resolution and accuracy during beamforming and AoA estimation. The strength of ULA is its simple structure, less computational / processing requirements and a good resolution for the AoA estimation.

2.9 Virtual Antenna Array

Our research group is working on beamforming and antenna array system since 2010 [13], [14], [99]–[102]. We have also worked on circular antenna array [102]. As per our previous knowledge, we can say that existing classic localization techniques require the physical existence of multiple antennas. This system becomes a huge problem in the application considered in this thesis, because we can't mount such heavy and large antenna on a small size UAV that can carry a limited payload battery onboard. In addition, mutual coupling, induced current, large size, energy utilization, and fixed steering mechanism are factors that limiting the performance of classical localization antenna. A little intention is also given to the virtual array formation [103], virtual antenna array is formed for indoor environments only, by considering line-of-sight (LoS) requirements. Multi path incoming radio signals deflected by indoor walls and items are used at the receiver end to measure AoA. In [104], circular virtual array is formed by moving single antenna in an arbitrary/circular path. As per our knowledge, none of the existing localization scheme proposed single antenna act virtually like a uniform linear array.

In this project, we developed a virtual antenna system, which can be installed over a small size UAV to locate ground sensors and harvest data. The distention of this virtual antenna system is as follows:

1) Physical ULA antenna is replaced with a single Virtual Phase Array (VPA) antenna
2) Proposed VPA antenna system can be mounted on an UAV moving with constant speed and height
3) Proposed antenna system should have enough light weight so that can be carried by UAV easily without disturbing its flight specifications
4) Proposed antenna system carrying by UAV can localize multiple sensors installed on ground level

The proposed system will be capable to operate with adaptive steering precession and support multiple frequencies to preserve maximum energy of the UAV and sensor nodes as well by adjusting number of antenna elements and spacing among them.

2.9.1 MUSIC algorithm for localization

An antenna array is used in such a way that the signals received at each element are combined and processed in order to achieve improved performance. The improvement is mentioned in term of increase in the overall gain, cancelation of interference from a particular set of directions, diversity of reception, resistance to Signal-to-Interference plus Noise Ratio (SINR), steering direction, AoA and other parameters estimation of incoming signals.

Array Signal Processing (ASP) has vital applications in sonar, biomedicine, astronomy, wireless communication system, seismic event prediction, radar, etc [105]. Various algorithms like Multiple

Signal Classification (MUSIC), Estimating Signal Parameters via Rotational Invariance Technique (ESPRIT), Weighted Subspace Fitting (WSF), Minimum Variance Distortion-less Response (MVDR), Maximum Likelihood (ML) techniques [106] and others can be used for the estimation process. Signals are considered distributed homogenously in space and its spatial spectrum is exploited to obtain the AoA. MUSIC [107] and ESPRIT [108] are the two widely used spectral estimation techniques based on Eigenvalues Decomposition (EVD) or Signal Value Decomposition (SVD). These subspace based estimation methods depend on the signals covariance matrices. The most classic example is MUSIC algorithm which gives accepted parameter estimation for both uniform and non-uniform linear antenna arrays. In ULA, the array elements are placed in linear format to satisfy the Nyquist sampling criteria and in this case conventional MUSIC estimation algorithm works well. MUSIC algorithm can compute different parameters including: number, direction and strength of incident signals. In this AoA assessment, large number of uniform samples of bearing space are taken to get many discrete angles.

The key point of MUSIC is its data model. Let's assume N to be the number of signal sources and M is the number of antenna elements $M \geq N+1$. The output signal of array element m, at any time t is given by:

$$x_m(t) = \sum_{k=1}^{N} a_m(\theta_k) s_k(t) + n_m(t) \tag{2.26}$$

Where a_m is the response function of the m array element to the K^{th} signal source; $S_k(t)$ is a narrowband signal and $n_m(t)$ is the white noise component. Equation (2.26) can be represented in matrix form as follows:

$$X = AS + N \tag{2.27}$$

Where $X = [x_1(t), x_2(t), \dots, x_m(t)]^T, s = [s_1(t), s_2(t), \dots, s_D(t)]^T$ and the Steering vector is given by: $A = [a(\theta_1), a(\theta_2), \dots, a(\theta_D)]^T$

MUSIC algorithm can well perform with fix a length and number of antenna elements as described earlier. The classical MUSIC algorithm is not suitable in our application. MUSIC is very sensitive to sensor position, gain, phase errors and needs careful calibration. In virtual antenna, all the samples of data are not collected at once. This property modified the signal characteristics more than usual. To compensate these additional changes, some additions and modifications are suggested in this research study to adapt the MUSIC algorithm to be functional for the proposed virtual antenna array system. Hereinafter we are proposing virtual antenna where a single moving antenna is used to form ULA. Complete detail of MUSIC algorithm and our proposed modification are given in the 4th chapter.

2.9.2 ESPRIT algorithm for localization

MUSIC algorithm has some limitations. It works well with ULA and narrowband signal but for other array shapes; it requires the knowledge of sensor positions. Furthermore, it is extremely sensitive to sensor position, gain, and phase errors and needs careful calibration. The ESPRIT algorithm overcomes such shortcomings to some degree by relaxing the calibration task. ESPRIT takes much less computation but it needs twice as many sensors as MUSIC, Figure 2.27.

The ESPRIT algorithm based on identical pairs of sensors called doublets that are aligned in a row to makes calibration a little easier.

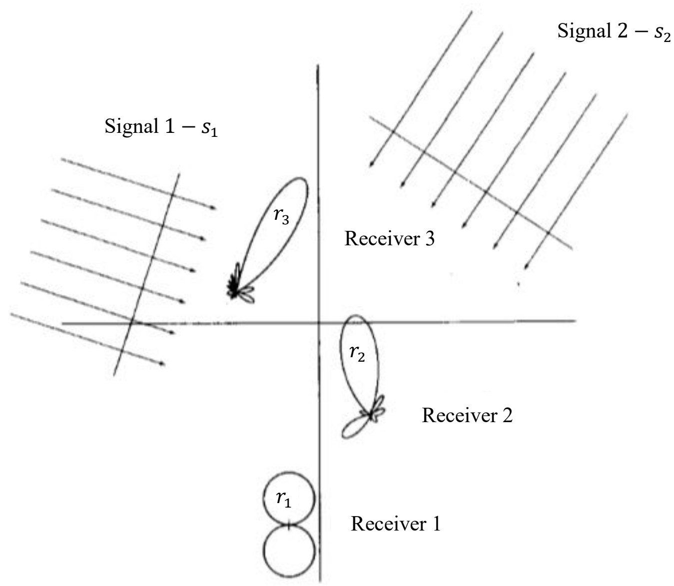


Figure 2.27: AoA by ESPIRT.

Assume there are $2M$ sensors making M sets of doublets. An incoming signal caused a delay $(d\ sin(\theta)/c)$ between two consecutive sensors; this delay is same for all doublets. As incoming signal is measured by doublet (two sensor nodes) and generate two out responses:

$$x_m(t) = \sum_{k=1}^{N} a_m(\theta_k)s_k\ (t)\ +\ n_m(t) = \mathrm{AS(t)} + \mathrm{n}_m(t) \tag{2.28}$$

and

$$Y_m(t)\ =\ \sum_{k=1}^{N} a_m(\theta_k)\ e^{j\gamma_k}\ s_k\ (t) +\ n_m(t) = \mathrm{A\Phi S(t)} + \mathrm{n}_m(t) \tag{2.29}$$

Where $\gamma_i = \frac{\omega_o d sin\theta_i}{c}$ and $\Phi = \mathrm{diag}\{e^{j\gamma_1}, e^{j\gamma_2}, \dots., e^{j\gamma_D}\}$

Steering vector $A = [a(\theta_1), a(\theta_2), \dots, a(\theta_D)]^T$ similar to the one used in MUSIC

MUSIC and ESPRT methods are both high-resolution, able to detect multiple sources. However, ESPRT is more accurate on the expense of more antennas and computation efforts and more elements. MUSIC is suitable with fewer and simple antenna elements.

2.10 Project Worthiness

Saudi Vision 2030 [109] is a plan to reduce Saudi Arabia's dependence on oil, diversify its economy, and develop service sectors such as health, education, agriculture and tourism. Saudi Arabia's need for water is a foremost challenge facing the government. Development of agriculture is also a top priority as the Kingdom has to pay heavy price to import food items. Concerning the future of water, agriculture and environment in Saudi Arabia, the Kingdom has approved a budget of $24.5 billion in its vision 2030 [110].

As our project addresses these issues has a great value and importance. Our project started by a research proposal approved and accepted by King Abdulaziz City for Science and Technology (KACST)[9] on the bases of reviews done by AAAS[10] (American Association for the Advancement of Science) for a grant worth 1.8 Million Saudi Riyals with a reference number 15-AGR5298-48 for three years starting from spring 2015. The acceptance reviews are attached in Appendix-G and working team is listed in Table 2.3. The project is conducted at SNCS (Sensor Network and Cellular Systems) research center University of Tabuk, Kingdom of Saudi Arabia with the collaboration of ENSTA-Bretagne. The goal of this project was to modernize the agriculture sector in KSA and to utilize sweet water resources more efficiently. The objectives were to get more revenue, fulfill ever increasing food demands and reduce the import load of the country.

In 2013, SNCS started to work on underdeveloped projects including water quality monitoring of red sea, visual light communication, etc. As this research center and university is located in Tabuk city, which is famous by its agriculture and producing large amount of fruit, vegetable and animal feed. Considering the importance of this region for agriculture, we initiate a smart farming project to address the real issues and

[9] KACST is a scientific government institution that supports and enhances scientific applied researches. It coordinates the activities of government institutions and scientific research centers in accordance with the requirements of the development of the Kingdom. KACST is a major program included in Saudi Arabian vision for 2030.

[10] AAAS (American Association for the Advancement of Science) is an international non-profit organization dedicated to advancing science for the benefit of all people. The AAAS seeks to advance science, engineering, and innovation throughout the world for the benefit of all people.

challenges faced in this area. We submitted this project to KACST for funding and presented it to Tabuk Agriculture Development Company (TADCO)[11] for implementation.

This project is led by me from SNCS as a team lead and completed with the consultancy of Dr. Ali MANSOUR and Dr. Denis Le JEUNE from ENSTA Bretagne. Currently, the theoretical work of this project is completed and proof of concept/ prototype devices are being developed which will be further tested in TADCO real crop fields in Tabuk to get acceptance for implementation.

TABLE 2-3: ASSIGNED TEAM FOR KACST PROJECT

Team Member	Role
Senior Personal:	
Muhammad Ayaz Arshad	PI
Mohammad Al Wakeel	CO-PI-1
Sami Al Wakeel	CO-PI-2
Other Personnel:	
Administrative including accountant-1	Administrative including accountant
Administrative including accountant-2	Administrative including accountant
Researcher -1(**M. Ammad uddin**)	Researcher
Researcher -2 (M. Rouf Abaza)	Researcher
PhD Student (Full time)	PhD Student
Technician-1	Technician
Dr. Ali Mansour	CONSULTANT
Dr. Denis Le Jeune	CONSULTANT

2.11 Major Challenges

The problem addressed in this research is data gathering from a large number of sensors unable to establish an ad-hoc communication due to widespread deployment of heterogonous sensor nodes, geographical constraints and power considerations. The highlights for the proposed system are:

1. Weather condition: As described earlier, KSA has severe weather conditions. Temperature can reach up to 50^{o} C in summer and fall down up to -1^{o} C in winter (in some areas like Tabuk, it can reach -10^{o} C). Air is dry with an average precipitation of 59 mm per year and dust storms are very often 15 to 20 major events per year. In this study, we developed a system to resist this environment and help crops to fight with.

2. Scarcity of water: The main hurdle for agriculture in KSA is the shortage of sweet water. In this study, we presented a system to overcome this problem by monitoring crop vitals that will be further used for decision making to manage resources especially water.

[11] TADCO is one of the largest agricultural companies in the Middle East that spans over 35,000 hectares.

3. No/limited communication infrastructure: In the scenario under consideration, monitoring area is very remote and fix infrastructure is not possible or very costly to build. In this research project, a fast deployed, handy and autonomous system is proposed; our system needs no prerequisites to deployed and can work solely.

4. Heterogeneity of sensor nodes: In our case study, heterogeneous sensor nodes are involved including crop monitoring, soil and environmental sensors. Some sensors are very small and can be installed on leaves, while others are very large and need to be fixed on tree trunk. Our study takes into consideration all these heterogeneities.

5. Localization of ground sensor nodes: In our Project, as all field nodes are considered cheap in cost and location unaware; then, the localization of these nodes to harvest data from them becomes an issue that should be addressed.

6. Need base data- A large number of sensors is installed in a big crop field and large number of crop fields are dispersed in a huge area. In this scenario, data from whole area and all sensors are rarely needed. This study proposed a system to collect data from selected nodes in area.

2.12 Suggested Solutions

In this study, we are focusing on dynamic cluster formation, virtual antenna for localization and data harvesting. Proposed system will have following properties.

1- No predefined cluster heads
2- Dynamic routing protocol
3- Diversified data collection
4- Dynamic route of UAV
5- Location unaware sensor nodes
6- Cheap and instant deployment
7- Self-convergence
8- Energy efficient

Our major contribution composed of two major parts:

A. **Dynamic Clustering-** Some snapshots of the proposed system are shown in Figure 2.28. An UAV or a swarm of UAVs is used to scan selective or all crop circles (crop fields are mostly in circular form in KSA, because single pivot system is generally used for watering), when an UAV reached a crop field, it will initiate UAV based clustering and routing protocol to collect data form all/selective sensors installed in crop field. Proposed protocol is described in detail in chapter 3.

A set of UAVs can scan all/selective gateway nodes in all circular/rectangle farm fields by taking a flight over predefined way. These UAVs may have equipped with receiving devices, cameras, Navigation/GPS system.
Figure 2.28: Proposed model for WSN using UAV's.

B. Virtual Antenna for localization- In this research study, we are proposing a virtual antenna system to overcome the challenges and limitations faced in conventional antenna system such as: large size, heavy weight, energy demanding, fixed steering, inter-element mutual coupling, etc. In the proposed system, single antenna is used for the localization procedure which will not affect much the specification of UAV. The detail model and working of proposed system is given in chapter 4.

2.13 Conclusion

In this chapter, we described the potential of smart agriculture in KSA and the challenges ahead to explore this potential. It was observed that real time monitoring of crop to take timely actions with precise amount of resources is the solution to overcome these challenges. The main focus of the chapter was to build a crop monitoring system comprise of state of art existing sensor devices, WSN and UAVs. Furthermore, we have explained how this crop monitoring system will be developed, what will be the required modification and additions to alien it with KSA environment and what modules are needed to make a complete system. Use of UAV in agriculture for communication and data gathering was another interesting topic of this chapter. Considering the worth of this study for KSA, KACST and SNCS have approved our research and development project. Some basic knowledge about smart forming and related technologies, we wrote in a CRC book chapter 2 title "Wireless Sensor Networks with Dynamic Nodes for Water and Crop Health Management" [111]. Some constituents of this thesis chapter are already published in IEEE 7th International Conference on Computer Science and Information Technology (CSIT) in 2016 [112]. The advanced and enhanced version of this paper is accepted in IEEE Sensors Journal [34].

3

Clustering

3.1 Introduction

UAVs are becoming the tool of choice in many applications. They come in a variety of shapes and capabilities. Because of their agility and low cost, they are good candidates for many applications such as: farming, rescue operations, persistent surveillance, aircraft inspection and maintenance, many other civil or military applications. It is anticipated that one of their most successful deployments will be in conjunction with smart farming. Smart farming relies heavily on information collected by unattended sensors embedded in open field strategic locations near and on crops. As described in chapter 1 and 2, the focus of this research is crop monitoring in Saudi agriculture where agriculture is facing more challenges than usual. In this situation, UAV is the best available choice to develop a low cost, fast deployed, portable solution requiring no prerequisite of infrastructure for data harvesting. Data gathering from sensors deployed in very hard conditions (like war field, disaster area, underwater/under sea and agriculture environment) is always a challenging task. Fix network infrastructure is not possible to build because of restriction including prohibitions, high initial cost and time constraints. Establishing ad-hoc communication is also not practical in many cases due to widespread deployment of heterogeneous sensor nodes, geographical constraints and power considerations.

Wireless Sensor Networks (WSNs) are highly resource conscious with limited power, bandwidth, processing, storage and computational capabilities. Therefore, sensor nodes are mostly inoperable and irreplaceable when failure occurs due to energy depletion. Increasing network sustainability and lifetime are the key issues for the contemporary studies in sensor domain. Normally, energy depletion is highly dominated by length and number of radio transmissions that are controlled by communicating protocol. Flat network (flooding [113]) is highly energy demanded way of communication where the entire network is used for a single operation, in case of large size network situation becomes worst. To achieve the small network features in a large sensor network, clustering is introduced. Clustering is the process to break down whole network into smaller groups and adding benefits to WSN like data aggregation, controlled communication, ease of management, better overall power consumption and prolonging network life time. Some existing clustering schemes are described in details in the following sections.

3.2 LEACH

In WSN, if a fixed CH node is used to collect all the data from the whole area then many energy depletion problems arise for example CH dies faster because it has to aggregate, process and forward the whole data; power down CH may cause the disconnection of whole area; In addition, nodes near to CH mostly used to relay the data routed toward CH also exhausted soon. All these problems effect the network life time and its performance badly. By randomly choosing a Cluster Head (CH), this problem can be reduced. LEACH stands for Low Energy Adaptive Clustering Hierarchy routing protocol for data fusion [114]. LEACH is an example of clustering protocol for homogeneous sensor networks where all sensor nodes are equipped with the same battery energy (see Figure 3.1). All the nodes in a network, organize themselves into local cluster and one node is randomly selected as CH on rotation bases. Every node will get chance to become a CH in $\left\lfloor \frac{n}{N_c} \right\rfloor$ rounds, N_c is desire number of clusters heads and n is total nodes in the area. Rotating the CH responsibilities among all member nodes help to balance the energy utilization in whole network. CH node maintains a list of members to establish a TDMA schedule based on allocated time slots for CMs. TDMA prevents intra-cluster collision and limits the inter cluster communication which further reduces the node energy consumption. LEACH cycle is divided into two rounds begins with set-up phase when the clusters are organized. Data is transferred from the nodes to the CH and to the Base Station (BS) in as second round steady-state phase.

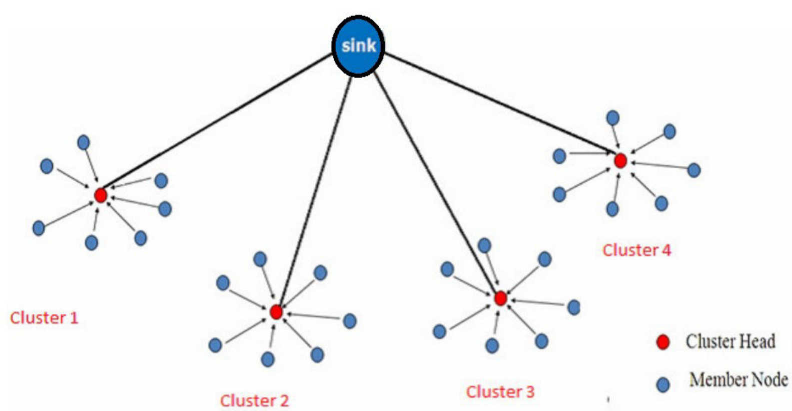


Figure 3.1: LEACH clustering.

The assumptions to run LEACH algorithm are as follows:

- All nodes are homogeneous,
- All nodes and base stations are static,
- All nodes start with the same initial energy,
- The BS is situated at the center of the area space,
- Normal nodes transmit directly to their respective cluster heads within a particular cluster,
- Routing is dynamic, no predefined CH and data routes.

Set-up phase: In this phase, the clusters are made and cluster-heads are selected. Each node gets a random number R_n in [0,1] and compares it to the threshold CH_{prob} given in equation (3.1) and if the threshold is higher than R_n, then the node becomes a CH. Cluster-heads can be chosen on randomly based as;

$$CH_{prob}(n) = \begin{cases} \frac{N_c}{n - N_c\,(r \; mod \left\lfloor \frac{n}{N_c} \right\rfloor)} & if\ n \in G \\ 0 & otherwise \end{cases} \tag{3.1}$$

Where N_c is the number of desired clusters, n is total number of nodes in the area, r is the current round, G is the set of nodes that have not been taken as cluster heads in the last $\left\lfloor \frac{n}{N_c} \right\rfloor$ rounds and R_n is a random number (uniform distribution between 0 and 1).

If $CH_{prob} > R_n$, then that node becomes a cluster-head.

Steady-state phase: All elected CHs broadcast advertisement using CSMA/CA protocol with same energy. Based on the received advertisement signal strength, Cluster Member (CM) nodes choose their cluster. The node chooses the CH whose advertisement has the highest received power in case of tie, again CH is selected on random basis. Each node informs the CH about its choice using CSMA/CA protocol. CHs create TDMA schedule and picks up a CDMA[12] code randomly and broadcast both, back to their nodes. TDMA is selected to avoid an inter-cluster collision and CDMA is used to avoid an intra-cluster interference.

[12] Code-division multiple access (CDMA) is a shared channel access method. Each transmitter assigned a separate code to send information simultaneously over a single communication channel. This allows several users to share a band of frequencies.

Energy Model: Energy model for transmission is shown in Figure 3.2.

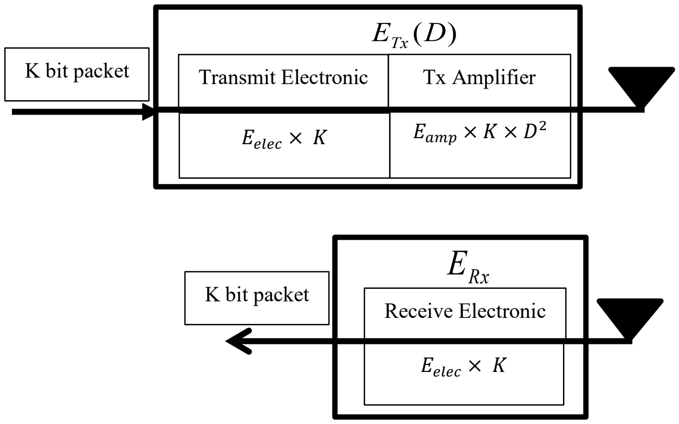


Figure 3.2: Energy model for LEACH [114].

If a sensor node wants to transmit K bits of data, energy is dissipated in two ways, for transmission and amplification. The energy required to process the data for transmission is not much, and only depends upon the amount of data, that is not controllable. The energy required for amplification depends upon data size and distance as well. In our case, this transmitter sender separation which causes major source of energy dissipation, can be managed. At receiving end, almost the same energy is required to receive the data as it is required for transmission process (excluding amplification). Although, a small amount of energy is also required for pre-amplification but we are not considering the receiving energy as whole because at receiving end we are using UAV that is a chargeable unit.

Transmission method also have a big impact on energy utilization, for example direct communication is shown in Figure 3.3.

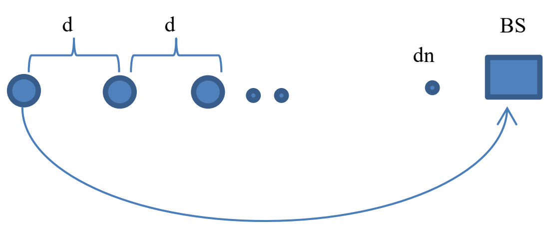


Figure 3.3: Direct transmission

Let us denote by K the length of the message in bits, D the distance between the transmitter and the receiver, E_{elec} the energy being dissipated to run the transmitter and E_{amp} is the energy dissipation to amplify the message.

As per Figure 3.2, E_{Tx} is the energy required to transmit K bits of data over D meters:

$$E_{Tx} = (E_{elec} \times K) + (E_{amp} \times K \times D^2) \tag{3.2}$$

Data transmitted by direct link is shown in Figure 3.3. Assume that all nodes are equally distributed with the same distance d, if a node is n hop away then the energy needs to transmit k bits becomes:

$$E_{Tx} = K(E_{elec} + E_{amp}(nD)^2) \tag{3.4}$$

Minimum Transmission Energy (MTE)[13] is considered as shown in Figure 3.4.

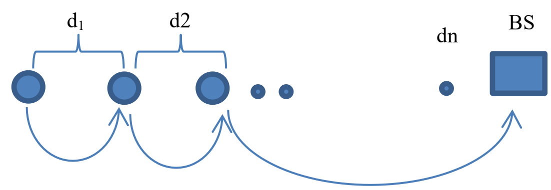


Figure 3.4: MTE transmission

The energy cost for a single transmission becomes:

$$E_{transmitting} = K((2n-1)E_{elec} + E_{amp}nD^2) \tag{3.5}$$

An improved version of LEACH is proposed in [115]. In this paper, two modifications are suggested:

1. Introducing a threshold for each round, any node above this threshold can be a candidate for the next round. In this way, if a node was CH in the last round and it still have enough energy above the threshold, then it can contest for CH in the next round too.
2. Different power levels are introduced to transmit signals within the and outside it.

3.3 HEED

HEED was designed to select different CHs in a field according to the amount of energy that is distributed in relation to a neighboring node. It is established that energy is the most critical resource in WSN and the LEACH was not considering it. In this case, there is possibility that a node, already suffering from lack of energy is selected as CH randomly which degrades the network performance and accelerates the WSN node to die faster. This problem was tried to overcome in Hybrid Energy-Efficient and Distributed clustering approach called HEED.

In HEED, clusters formation is based on two parameters: residual energy is the primary parameter and network features like node degree or distance are secondary one. Secondary parameter is only used to break the ties. Each node calculates its probability CH_{prob} of becoming a CH performs neighbor discovery and broadcasts its cost to neighbors. Probability CH_{prob} , is calculated as:

[13] In Minimum Transmission Energy (MTE) consumption routing protocol [185] Traffic is routed through intermediate nodes rather than direct communication.

$$CH_{prob} = \max\left(P \times \left(\frac{E_{residual}}{E_{max}}\right), N_c\right) \tag{3.6}$$

Where N_c describes the percentage of network nodes (e.g., 5%) to become CHs, $E_{residual}$ describes the present node energy and E_{max} describes maximum energy.

$$CH_{prob} = C_{Prob} . E_{Node} \tag{3.7}$$

Where, E_{Node} is the % of node energy.

The primary parameter is node energy percentage. Thus, a node with high energy has a higher chance to be elected as a cluster head. To increase energy efficiency and further prolong network lifetime, intra-cluster "communication cost" is also taken into consideration as a secondary clustering parameter. Cost can be a function of neighbor proximity or cluster density. If the power level used for intra-cluster communication is fixed for all nodes, then the cost can be proportional to node degree. This means that a node joins the cluster head with minimum degree to distribute cluster head load (possibly at the expense of increased interference and reduced spatial reuse), or joins the one with maximum degree to create dense clusters. The terms minimum degree cost and maximum degree cost is used to denote these cost types. HEED assumptions are:

- Nodes in the network are quasi-stationary,
- Nodes are location unaware,
- Nodes have similar processing, communication capabilities and equal significance.

3.4 Network Assisted Clustering

In [83], network assisted routing and data gathering schemes are proposed, mobile sink is used to collect data from ground sensors. All the ground sensors will make a connected network and maintain routes by periodic beaconing. All the nodes in the network always have up-to-date route and neighboring information. Data collection and relay nodes are also known to everyone. UAV has to find and follow pre-defined data collection centers and need to follow ground network defined path to collect data from all the nodes. Once UAV keeps in contact with any one of the node in the network, it will guide him to the next data collection center and its location. The problem with this type of schemes is that UAV should have to follow the network defined path strictly and network updates are also an overhead.

This algorithm works in four phases. Navigation Agents (NA) and Intermediate Navigation Agents (INA) are selected in first phase as shown in Figure 3.5. NAs are selected in such a way that all network nodes can be approached up to n hops ($n = 2$ in this case). Shortest path between two NAs is selected as intermediate navigators. Traveling Salesman Problem (TSP) scheme is used to compute shortest route between all NAs. Remaining nodes that have direct connection with NAs declared themselves Getaway Nodes (GA). Mobile sink will collect data from these GAs, NAs and INAs help UAV to navigate/ approach nearest GA. The algorithms used in our thesis are given in appendix A:

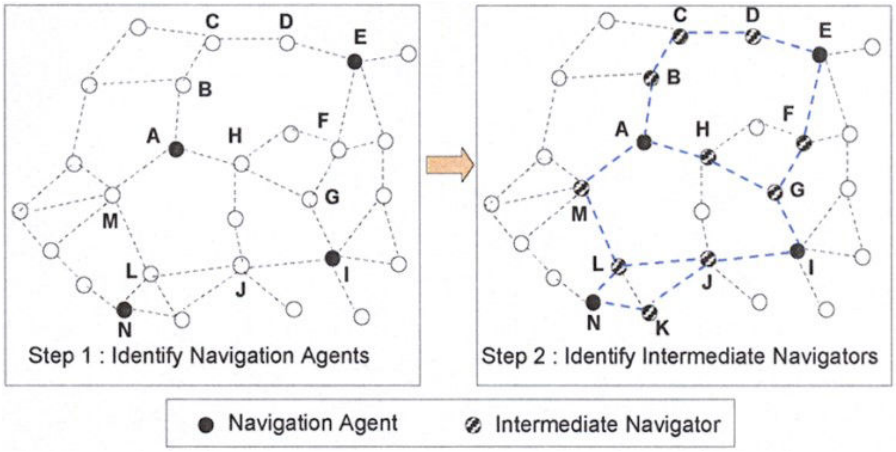


Figure 3.5: Network assisted clustering and routing technique.

3.5 Proposed UAV Assisted Dynamic Clustering

As far as we know, the existing clustering and data gather schemes, do not full fill all the requirements of our case study (agriculture in KSA); Some serious issues have been described in detail in section 2.11. Here, we are proposing a dynamic cluster and data gathering scheme that is developed to satisfy not only the under-consideration scenario but all other like constraints as well e.g. war field, security and severance, wild life monitoring, etc.

We are presenting here a dynamic clustering that consists of two components, sensor nodes and UAVs. Each component of proposed system is defined in next subsections.

3.5.1 UAV

In the developed system, UAV plays a vital role and performs major tasks of localization, communication and data gathering and whole the system is trigged by UAV beacon. Considering the importance of UAV in our system, we build a special customized quadcopter S500, detail design and step wise procedure are given in chapter 5. The developed S500 should be equipped with a sink node which consists of a designed software protocol (algorithm 3.7.1), hardware (see chapter 5) and developed virtual antenna system (Chapter 4).

Assumptions and specifications of our developed UAV are:

- Quad copter with hovering functionality,
- Minimum flight height = 20 meter,
- Maximum flight height = 500 meter,
- Maximum speed = 7 m/s.

UAV sends beacon message to activate sensor node, the parameters set in the beacon are as:

- **Sensor type:** Type of sensor nodes or combination that needed to be activated in response of this beacon.
- **Height:** Current data collection height of UAV not the flying height. The sensor nodes that are capable to communicate at least this distance will be considered as candidate for a CH.
- **Threshold:** This threshold will be used to limit the number of sensor nodes that will contest for CH.
- **Trailer:** A trailer contains Error Detection Code (EDC) or any other information.

Type 6 byte	Height 2 byte	Threshold 1byte	Others 2 bytes
Header		**Payload**	**Trailer**

3.5.2 Sensor node

Field sensors installed to monitor a specific parameter about crop, soil or environment has the following properties and developed prototype is shown in chapter 5.

- Location unaware, cheap in cost and left un-attendant.
- Support multiple frequencies, at least 2 frequencies 433 MHz and 2.1 GHz for localization and data transmission.
- By default, 433 MHz is switch on to hear from UAV, afterward it will be used for localization and synchronization. While 2.1 GHz transmitter / receiver will be activated on demand for communication only.
- Can hear UAV if in range, but all might not be able to connect with because of their internal parameters.
- Maximum communication range = 500 m.
- processing + memory enable.

All sensor nodes considered to have a unique ID of format as given bellow:

- **Type:** We are considering 3 basic types: crop, soil and environment. However, we are allocating 1 byte for further and future extensions. For example, to represent 3 types, we need (2^3-1=7) possible combinations.
- **Subtype:** Leaf, stem, root and any combination.
- **Purpose:** Temperature, humidity, thickness, flow and all possible combinations.
- **Unique id:** Unique number of each sensor node.

The field sensor nodes will have 6 bytes of unique id. UAV will select any particular type or any combination with the help of this id also called prefix. Sensor nodes will also use it to send beacon acknowledgement/reply.

Circle No	Type	Sub Type	Purpose	Unique No	Total size
1byte	1byte	1byte	1byte	2 byte	6 byte
Prefix					

In our proposed system, initially all sensor nodes are considered as undisguisable (no special CH). A sensor node has to keep 5 parameter values about its health. All activated sensor nodes will calculate a probability value to become a CH based on these health indicators:

1. **Energy:** How much remaining energy,
2. **Consumption rate:** The consumption rate of energy to perform its job,
3. **Renewable energy:** Renewable energy is available or not,
4. **Antenna size:** Antenna size to estimate its communication range,
5. **Data size:** How much data it has to transmit.

The sensor node reply to UAV will consist of 9 Bytes containing prefix (node id) in packet header, probability value (probability to be selected as CH, calculated by Bayesian formula by putting all its health indicator's values) in payload and trailer. The trailer 1 byte is allocated for CRC and future use. A sensor node activated in response of UAV beacon, will send a reply as:

ID 6 byte	Probability 2 byte	Trailer 1 byte
Header	**Payload**	**Trailer**

The UAV and sensor nodes having mentioned above specificities and functionality will complete data gathering proposed in three layers, the top layer consists of UAVs taking care of all UAVs related tasks. Middle layer composed of CHs acts like connection between low power small size sensors and UAV. Lower most layer is a sensing layer which consists of many heterogeneous sensors engaged in the assessment of crop, soil and environment.

3.6 Three-Layer Architecture of the Developed System

The proposed UAV assisted routing and data gathering scheme is developed in multi-layers and multi phases. The assumptions for this protocol are as follows:

1. All the sensor nodes are taken undistinguishable, no predefined CHs. The advantage taken from this assumption is that data will be collected form all alive and available sensors only. Any number and type of nodes that are unavailable due to any reason (cover by sand, water, plant follicle, running out of battery, etc.) does not effect on data gathering process.
2. No predefined data routes and clusters are considered, as result no periodic updates for route discovery and maintenance is required, the benefit achieved is maximum energy saving for sensor nodes.
3. Developed UAV and sensor nodes have dual frequency support UHF (433 MHz) for localization and WiFi (4.2 GHz) for data collection.
4. Independent UAV path has great impact in the sense that UAV can move freely to scan desire area and can instruct the ground sensors to activate and aggregate require data on a suitable point.

Developed 3-layer architecture of proposed system is defined as under:

1. Layer-1 UAV: UAVs are the main part and the top layer,
2. Layer-2 CH: One node per cluster will be selected as CH and they will form a 2nd middle layer,
3. Layer-3 CM: Composed of ground sensor nodes.

Each layer further divided in three phases as shown in Figure 3.6.

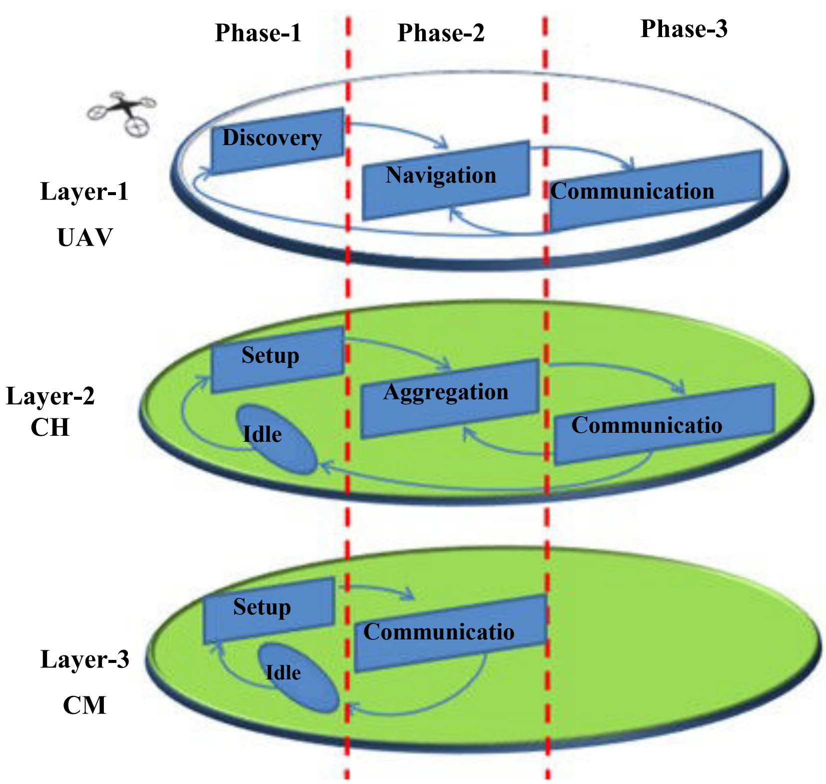


Figure 3.6: Multi-layers and multi-phases proposed system architecture.

The main part of this system is UAV that acts like a data mole, and the process is started with its beacon message. The UAV can operate at two different frequencies: very low power and long-range UHF F_1, for localization and high data rate WiFi F_2. UAV is performing localization at distance D_1 which is considered as constant and should be high enough to satisfy the far field conditions of F_1. Data collection is conducted at distance D_2 which can be adjusted from 20 to 170 meters by considering CH transmission ability, safe flight distance and WiFi limitations. The working life cycle of UAV spans over three phases as seen in Figure 3.6.

3.6.1 Discovery

UAV performs five main tasks in this phase beaconing, estimating the number of connected nodes, shunting, localizing and CH nominating as explain in figure 3.7.

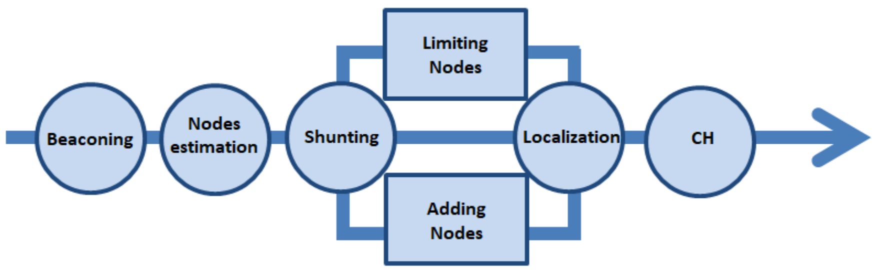


Figure 3.7: UAV discovery phase.

Let us suppose that UAV is flying at a height D_1 and initiates data gathering process by sending a beacon $B(T_{ys}, D_2, Th)$ containing type T_{ys} of nodes required to activate for data collection, distance D_2 at which it will collect data and Th is, the threshold to limit the activated node who will contest for CH. Initially $Th = 0$ mean all nodes can participate in CH selection process. Suppose that there are n sensor nodes $S = \{s_1, s_2, \dots, s_n\}$ installed in a crop field. Assume O nodes become activated in response of UAV beacon such that $1 \le O \le n$ and $SS = \{SS \subseteq S \mid SS \in T_{ys}\} = \{ss_1, ss_2, \dots, ss_O\}$. SS is the set of nodes that will make cluster in response of this beacon.

If each active node SS_i has $\mathscr{b}_i$ bit data to transmit then $T\mathscr{b}$ represents the total size of cluster data, that is expected to be transmitted by CH to UAV:

$$T\mathscr{b} = T\mathscr{b}(SS) = \sum_{i=0}^{O} \mathscr{b}_i \tag{3.8}$$

The energy required to transmit $T\mathscr{b}$ to UAV for distance D_2 is;

$$TE_{Tx} = (E_{elec} \times T\mathscr{b}) + \left(E_{amp} \times T\mathscr{b} \times (D_2)^2\right) \tag{3.9}$$

While UAV is in discovery phase, activated ground sensors SS will be in setup phase to make a cluster. Every member of SS will calculate a Bayesian probability P_i considering all its parameters (energy, consumption, antenna size, data size, etc). Detail discussion on Bayesian probability is given in section 3.6.4. Based on E_i, the energy of the node SS_i, and P_i, the probability to be selected as CH, a further subset called Candidate Cluster Heads $CCH = \{cch_1, cch_2, \dots, cch_q\}$, where $1 \leq q \leq O$, is formed such that

$CCH = \{CCH \subseteq SS \mid \forall E_{i_{current}}$ $T\mathcal{E}$ & $P_i > Th\}$. As described earlier, $Th = 0$ means any activated node having enough energy can be a member of CCH.

All CCH nodes start sending beacon replies to UAV in the form of narrow band signals. At this point, UAV estimates the number of CCH $|CCH| = q$. If UAV estimates that $q = 0$ or $q \geq m$, where m is the number of antenna elements on board, then shunting process is started to keep the candidate cluster heads in a reasonable range; Otherwise, UAV will locate all CCH nodes by using a special virtual phase array antenna developed for said purpose and detail model is given in chapter 4. Based on CCH location and P_i values, UAV nominates a final CH and broadcast a message to all nodes of set SS to let them informed about the CH nomination.

Shunting is the important process of discovery phase to handle the situation when:

$q = 0$ or $q \geq m$

If $q = 0$ or $CCH = \emptyset$ it means no CCH member has the capability to send aggregated data to UAV in this case, shunting decreases D_2 in steps up to a minimum height depending upon safe flight constraint.

If $q \geq m$ means there are many good CCH nodes and UAV cannot locate all at once, in this case, shunting can take 3 steps.

1. Increase antenna capacity from m to $2m$,
2. Increase D_2 in steps up to the limit of F_2,
3. Increase Th.

Once cluster head is selected the next phase of UAV is navigation.

3.6.2 Navigation

Once CH is selected UAV enters in next phase called navigation. In similar way, CH and CM switch their phases from setup to aggregation and communication respectively. In this phase, all active nodes will switch on their frequencies from F_1 to F_2 (433 MHz to 2.4 GHz). Only CH will operate on both frequencies. CH will use F_2 to collect data from CMs and F_1 for UAV navigation. While UAV approaches CH and attains an agreed height D_2 ; All CM nodes must transmit their data to CH which will aggregate it and get ready to make a link with UAV. As soon as UAV approaches D_2 and starts handshaking, CH will switch off its F_2 module and navigation phase becomes over. Navigation diagram is elaborated in a diagram Figure 3.8.

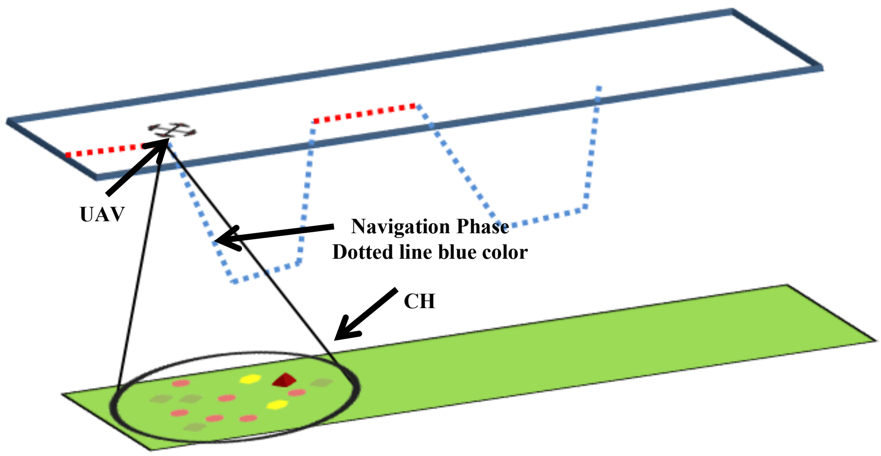


Figure 3.8: Navigation phase.

3.6.3 Communication

In the last phase, only the top two layers (UAV and CH) of our developed system will participate. CH will transmit the whole data to UAV at frequency F_2; Once transmission is done, CH will go to sleep for a specific time T, while all CMs had already been shifted to sleep. To elaborate the exact sequence of steps the state diagram of the developed system is shown in Figure 3.13.

3.6.4 CH selection

Each node will participate in CH selection process depending upon its probability calculated by using a Bayesian classifier.

There are n sensor nodes $S = (s_1, s_2, \dots, s_n)$ and each sensor node s_i has z attributes (independent variables) represented by a vector $\mathcal{A} = (a_1, a_2, \dots, a_z)$. A sensor node s_i can be in one of two states:

a) Cluster Head (CH),

b) Cluster Member (CM), representing by $State = (CH, CM)$.

$P(s_i = CH | a_{ij})$ is the posterior probability of a given sensor node s_i to be a CH knowing its attribute a_{ij}.

$P(s_i = CH)$ is the prior probability of given node to be a CH.

$P(a_{ij} | s_i = CH)$ is the likelihood probability that the highest value of the attribute a_{ij} is in CH node.

$P(s_i = CM)$ is the prior probability that a given node s_i is a member node.

$P(a_{ij} | s_i = CM)$ is the probability that the highest value of attribute a_{ij} is in CM and not in CH

$P(a_{ij})$ is the prior probability of the attribute value a_{ij} is the highest one.

Equation (3.11) is the probability of the node s_i to be cluster head by knowing only one parameter a_{ij}. If all parameters $\mathcal{A}_i = (a_1, a_2, \ldots, a_z)$ of node s_i are independent in nature then the conditional probability of this node, by considering whole set $\mathcal{A}_i$ can be calculated as:

$$\mathrm{P}(a_{\mathrm{i1}}, a_{\mathrm{i2}}, \ldots, a_{\mathrm{iz}} | \mathrm{s}_i = \mathrm{CH}) = \prod_{j=1}^{z} \mathrm{P}\big(a_{\mathrm{ij}} \big| \mathrm{s}_i = \mathrm{CH}\big) = \prod_{j=1}^{z} \frac{\mathrm{P}\big(\mathrm{s}_i = \mathrm{CH}\big|a_{\mathrm{ij}}\big)\,\mathrm{P}\big(a_{\mathrm{ij}}\big)}{P(\mathrm{s}_i = \mathrm{CH})} = \frac{\prod_{j=1}^{z} \mathrm{P}_{ij}\,\mathrm{P}\big(a_{\mathrm{ij}}\big)}{[P(\mathrm{s}_i = \mathrm{CH})]^z} \tag{3.12}$$

$$\mathrm{P}(a_{\mathrm{i1}}, a_{\mathrm{i2}}, \ldots, a_{\mathrm{iz}} | \mathrm{s}_i = \mathrm{CM}) = \prod_{j=1}^{z} \mathrm{P}\big(a_{\mathrm{ij}} \big| \mathrm{s}_i = \mathrm{CM}\big) = \prod_{j=1}^{z} \frac{\mathrm{P}\big(\mathrm{si} = \mathrm{CM}\big|a_{\mathrm{ij}}\big)\,\mathrm{P}(a_{\mathrm{ij}})}{P(\mathrm{si} = \mathrm{CM})} = \frac{\prod_{j=1}^{z} (1 - \mathrm{P}_{ij})\,\mathrm{P}(a_{\mathrm{ij}})}{[1 - P(\mathrm{s}_i = \mathrm{CH})]^z} \tag{3.13}$$

If:

P_i is the probability $P(s_i = CH|\mathcal{A}_i)$ of the i[th] node, s_i to be a CH by knowing a set of all its parameters $\mathcal{A}_i$,

P_{ij} is the probability $\mathrm{P}\big(\mathrm{s}_i = \mathrm{CH}\big|a_{ij}\big)$ of the i[th] node, s_i to be a CH by knowing its j[th] parameter a_{ij} from the parameters set $\mathcal{A}_i$ and a_{ij} is the j[th] attribute of the i[th] sensor node.

Then:

$$\mathrm{P}(\mathrm{s}_i = \mathrm{CH}|\mathcal{A}_i) = \frac{\mathrm{P}(\mathrm{s}_i = \mathrm{CH})\,\mathrm{P}(\mathcal{A}i|\mathrm{s}_i = \mathrm{CH})}{\mathrm{P}(\mathcal{A}_i)}$$

$$= \frac{\mathrm{P}(\mathrm{s}_i = \mathrm{CH})\,\mathrm{P}(\mathcal{A}_i|\mathrm{s}_i = \mathrm{CH})}{\mathrm{P}(\mathrm{s}_i = \mathrm{CH})\,\mathrm{P}(\mathcal{A}_i|\mathrm{s}_i = \mathrm{CH}) + \mathrm{P}(\mathrm{s}_i = \mathrm{CM})\,\mathrm{P}(\mathcal{A}_i|\mathrm{s}_i = \mathrm{CM})} \tag{3.14}$$

$$\mathrm{P}\big(\mathrm{s}_i = \mathrm{CH}\big|a_{\mathrm{ij}}\big) = \frac{\mathrm{P}(\mathrm{s}_i = \mathrm{CH})\,\mathrm{P}\big(a_{\mathrm{ij}}\big|\mathrm{s}_i = \mathrm{CH}\big)}{\mathrm{P}(a_{\mathrm{ij}})} \tag{3.10}$$

$$\mathrm{P}\big(\mathrm{s}_i = \mathrm{CH}\big|a_{\mathrm{ij}}\big) = \frac{\mathrm{P}(\mathrm{s}_i = \mathrm{CH})\,\mathrm{P}\big(a_{\mathrm{ij}}\big|\mathrm{s}_i = \mathrm{CH}\big)}{\mathrm{P}(\mathrm{s}_i = \mathrm{CH})\,\mathrm{P}\big(a_{\mathrm{ij}}\big|\mathrm{s}_i = \mathrm{CH}\big) + \mathrm{P}(\mathrm{s}_i = \mathrm{CM})\,\mathrm{P}\big(a_{\mathrm{ij}}\big|\mathrm{s}_i = \mathrm{CM}\big)} \tag{3.11}$$

Putting Values of (3.12) and (3.13) in equation (3.14).

$$\mathrm{P}(\mathrm{s}_i = \mathrm{CH}|\mathcal{A}_i) = \frac{\mathrm{P}(\mathrm{s}_i = \mathrm{CH})\,\frac{\prod_{j=1}^{z} \mathrm{P}_{ij}\,\mathrm{P}(a_{\mathrm{ij}})}{[P(\mathrm{s}_i = \mathrm{CH})]^z}}{\mathrm{P}(\mathrm{s}_i = \mathrm{CH})\,\frac{\prod_{j=1}^{z} \mathrm{P}_{ij}\,\mathrm{P}(a_{\mathrm{ij}})}{[P(\mathrm{s}_i = \mathrm{CH})]^z} + [1 - P(\mathrm{s}_i = \mathrm{CH})]\,\frac{\prod_{j=1}^{z} (1 - \mathrm{P}_{ij})\,\mathrm{P}(a_{\mathrm{ij}})}{[1 - P(\mathrm{s}_i = \mathrm{CH})]^z}}$$

$$\mathrm{P}(\mathrm{s}_i = \mathrm{CH}|\mathcal{A}_i) = \frac{\frac{1}{\mathrm{P}(\mathrm{s}_i = \mathrm{CH})^{z-1}}\,\prod_{j=1}^{z} \mathrm{P}_{ij}\,\mathrm{P}(a_{\mathrm{ij}})}{\frac{1}{\mathrm{P}(\mathrm{s}_i = \mathrm{CH})^{z-1}}\,\prod_{j=1}^{z} \mathrm{P}_{ij}\,\mathrm{P}(a_{\mathrm{ij}}) + \frac{1}{[1 - P(\mathrm{s}_i = \mathrm{CH})]^{z-1}} \prod_{j=1}^{z} (1 - \mathrm{P}_{ij})\,\mathrm{P}(a_{\mathrm{ij}})}$$

Let us consider that $P(a_{ij})$ are constant then the previous equation can be simplified as follows:

$$\mathrm{P}(\mathrm{s}_i = \mathrm{CH}|\mathcal{A}_i) = \frac{\prod_{j=1}^{z} \mathrm{P}_{ij}}{\prod_{j=1}^{z} \mathrm{P}_{ij} + \left(\frac{\mathrm{P}(\mathrm{s}_i = \mathrm{CH})}{1 - P(\mathrm{s}_i = \mathrm{CH})}\right)^{z-1} \prod_{j=1}^{z} (1 - \mathrm{P}_{ij})} \tag{3.15}$$

If considering "not biased" condition where all nodes in the network have the same probability to become cluster head then equation (3.15) can be written as:

$$\mathrm{P}(\mathrm{s}_i = \mathrm{CH} | a_{\mathrm{ij}}) = \frac{\mathrm{P}(a_{\mathrm{ij}} | \mathrm{s}_i = \mathrm{CH})}{\mathrm{P}(a_{\mathrm{ij}} | \mathrm{s}_i = \mathrm{CH}) + \mathrm{P}(a_{\mathrm{ij}} | \mathrm{s}_i = \mathrm{CM})} \tag{3.16}$$

And equation (3.16) can be written as:

$$\mathrm{P}_i = \mathrm{P}(\mathrm{s}_i = \mathrm{CH} | \mathcal{A}_i) = \frac{\mathrm{P}_{i1}.\,\mathrm{P}_{i2} \dots \dots \mathrm{P}_{iz}}{\mathrm{P}_{i1}.\,\mathrm{P}_{i2} \dots \dots \mathrm{P}_{iz} + (1 - \mathrm{P}_{i1}).\,(1 - \mathrm{P}_{i2}) \dots \dots (1 - \mathrm{P}_{iz})} \tag{3.17}$$

As equation (3.17) may cause floating point underflow problem, converting the equation in log domain.

$$\frac{1}{\mathrm{P}_i} - 1 = \frac{(1 - \mathrm{P}_{i1}).\,(1 - \mathrm{P}_{i2}) \dots \dots (1 - \mathrm{P}_{iz})}{\mathrm{P}_{i1}.\,\mathrm{P}_{i2} \dots \dots \mathrm{P}_{iz}} \Longrightarrow$$

$$\ln\left(\frac{1}{\mathrm{P}_i} - 1\right) = \sum_{j=1}^{z} \left(\ln(1 - \mathrm{P}_{ij}) - \ln \mathrm{P}_{ij}\right) \tag{3.18}$$

$\ln(P_{ij})$ can produce a problem when P_{ij} is close to zero; the results will be asymptotically correct [14].

Let $\mu = \sum_{j=1}^{z} \ln(1 - \mathrm{P}_{ij}) - \ln \mathrm{P}_{ij}$

$$\mu = \sum_{j=1}^{z} \ln\left(\frac{1 - \mathrm{P}_{ij}}{\mathrm{P}_{ij}}\right) = \sum_{j=1}^{z} \ln\left(\frac{1}{\mathrm{P}_{ij}} - 1\right)$$

Finally, we can write that:

$$\mathrm{P}_i = \frac{1}{\mathrm{e}^{\mu} + 1}$$

3.6.5 Time synchronization

---

[14] According to equation (3.17), If $P_{ij} \rightarrow 0^{+}$ Then $P_i \rightarrow 0^{+}$. In this case, the log in the right hand part of equation (3.18) will be asymptotically as follows:

$$P_{ij} \rightarrow 0^{+} \Rightarrow ln(P_{ij}) \rightarrow -\infty \tag{A.1}$$

Then the right hand part of equation (3.18) will tend to

$$\sum_{j=1}^{z} \left(\ln(1 - \mathrm{P}_{ij}) - \ln \mathrm{P}_{ij}\right) \xrightarrow[P_{ij \to 0^{+}}]{} +\infty$$

Therefore, we can conclude that

$$\ln\left(\frac{1}{P_i} - 1\right) \xrightarrow[P_{ij \to 0^{+}}]{} +\infty \Rightarrow \frac{1}{P_i} - 1 \xrightarrow[P_{ij \to 0^{+}}]{} +\infty \tag{A.2}$$

$$\Rightarrow \frac{1}{P_i} \xrightarrow[P_{ij \to 0^{+}}]{} +\infty \Rightarrow P_i \xrightarrow[P_{ij \to 0^{+}}]{} 0^{+}$$

In the proposed system, data delivery among cluster head and member nodes are conducted by using TDMA scheme to preserve their energy as maximum as possible and to avoid collision to get better throughout. TDMA is also used for coordinated sensor wakeup in order to wakeup sensor nodes in coordinated way, to preserve maximum energy, rather than activated all the time. To use TDMA scheme all the sensor nodes need to be well synchronized so that each one knows about its time slot to send data without collision. Time synchronization in wireless sensor nodes is not as simple as in wired network because of unreliable sensor nodes, random wakeup time, unavailability of centralize clock, timing server and GPS module.

Many self-organized TDMA schemes are already proposed that works well for WSN networks like [116]–[119]. Self-Organized Time Division Multiple Access (SOTDMA)[15] is the most complex scheme made for Automatic Identification System (AIS[16]) that can establish TDMA in distributed way. Each node can assign a time slot without any global knowledge. SOTDMA is not suitable in our scenario because almost all of its variation (SOTDMA, RATDMA, ITDMA, FATDMA, PATDMA uses satellite as an external source of common time of reference. In our case, sensor nodes are not equipped with GPS module. CSTDMA[17] (Carrier Sense Time Division Multiple Access) do not require GPS synchronization. In CSTDMA, a node needs to monitor channel background noise level continuously to select a TDMA slot for itself. The continuous sniffing of the channel is not a good choice for our small battery nodes.

There are three basic types of synchronization methods for wireless networks [120]. The first and simplest is relative timing for event sequencing. Clock synchronization is not important, only local clock is used to ensure sequence of event (event 1 occurred before event 2) is maintained. The next method is relative timing with offset. The nodes have the ability to synchronize their local time with other nodes at any moment and the nodes keep track of drift and offset correspondence to neighboring nodes. The last method is global synchronization, where there is a constant global timescale throughout the network. It is mostly not required because of complexity and heavy processing requirement that a small sized energy conscious sensor node cannot afford. Many Synchronization protocols are being developed in this regards such as: Network Time Synchronization (NTP), Timing- Sync Protocol for Sensor Network (TSPN), Flooding Time Synchronization Protocol (FTSP), Ratio Based time Synchronization Protocol (RSP), Reference Broadcast Synchronization (RBS), Hierarchical Referencing Time Synchronization (HRTS),

---

[15] GPS based distributed time synchronization where all stations share a common time reference, derived from GPS to ensure accurate estimation of start time of each TDMA slot [186].

[16] AIS, Automatic Identification System is an automatic tracking system for collision avoidance used in ships [186].

[17] CSTDMA is a kind of SOTDMA, where AIS base station transmissions within receiver range is used to allocate slot timing. GPS based timing is not required, background noise level is used as a reference for a received signal strength measurement at the start of each TDMA slot [187].

etc. for more details see survey paper [121]. We are discussing two famous one IEEE 1588 [122] and Lightweight Synchronization (LTS) [123].

A. IEEE 1588 Synchronization: IEEE 1588 is called Precision Time Protocol (PTP), provides a standard protocol for clock synchronization. It can operate with heterogeneous network of clocks containing different characteristics such as: clock source and stability. In this protocol, the reference clock is called Grand Master Clock (GMC) which synchronizes all the Slave Clocks (SCs). Bidirectional multicast communication is used to synchronize the SCs (see figure 3.9 for protocol and 3.10 for example).

The GMC issues a sync call to initiate synchronization at time $T1$ and then sends a follow-up at time $T2$. SC receives these calls at $T3$ and $T4$ consecutively and updates its clock.

Let $T3$ and $T4$ be known by SC and $T1$ is received through a message from GMC; However, the propagation time is not known for both. In order to synchronize SC with GMC, first fix should be done at $T4$ on SC. Remember that propagation time is not known but as we know is same for both transmissions, see Figure 3.9. A solve example of this synchronization is given in Figure 3.10:

$$\Delta T = T2 - T1 = T4 - T3 \Rightarrow \tag{3.19}$$

$$T2 = T4 - T3 + T1 \tag{3.20}$$

SC will change its local clock to $T2$ and sends a delay request at time $T5$ and gets an immediate reply from GMC at $T6$ and updates its clock once again as second fix:

$$T7 = \frac{T6 - T5}{2} = Propagation\ time \tag{3.21}$$

At $T7$, SC will add this calculated Propagation time to its clock and SC will become successfully synchronized with GMC.

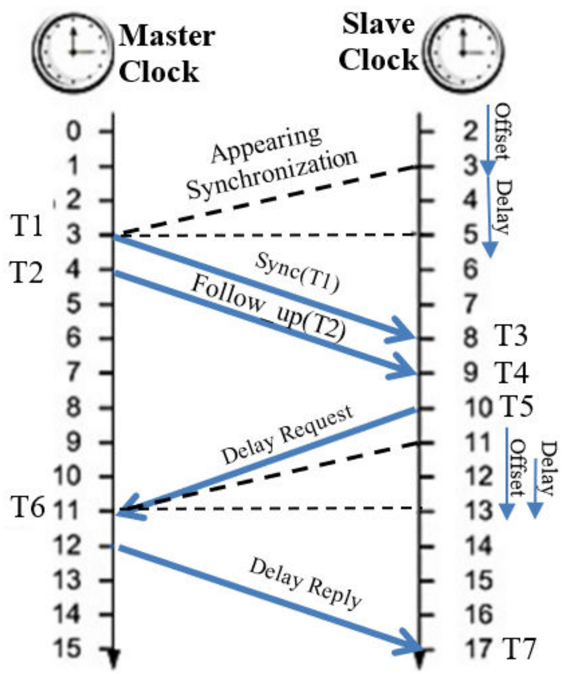


Figure 3.9: Running example of PTP.

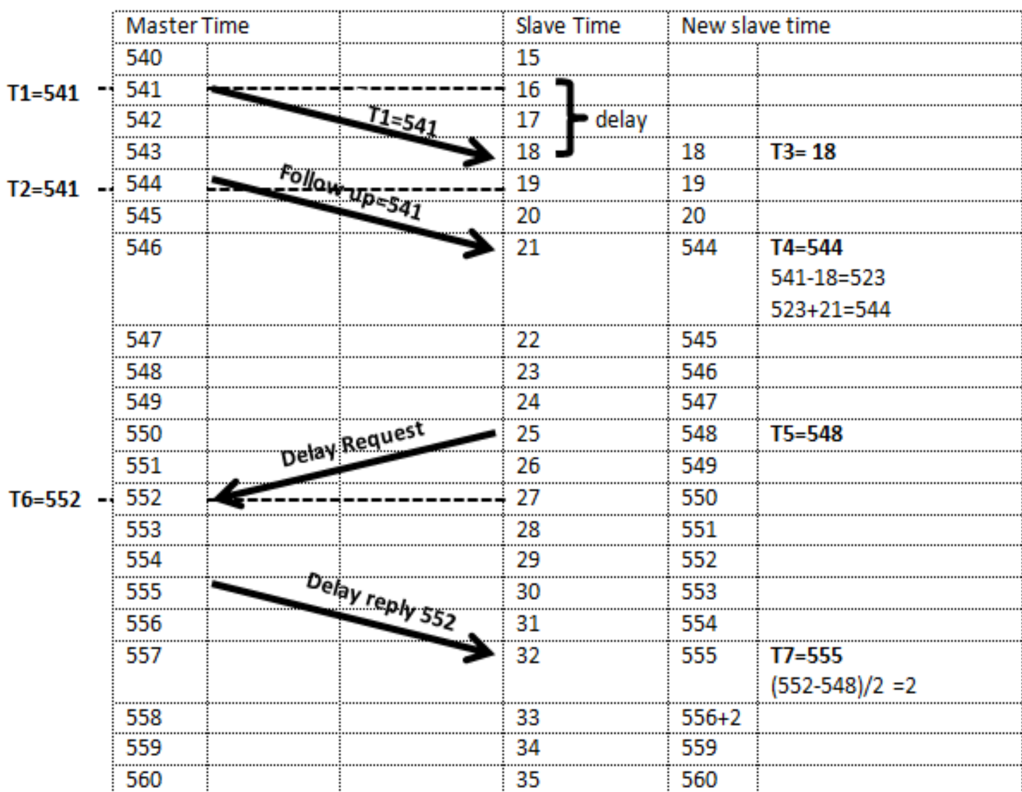


Figure 3.10: IEEE 1588 synchronization protocol [122].

Let us initially suppose that, GMC is at clock 540 seconds and SC is started later on and its clock is at 15 seconds. GMC sends synchronization call at $541(T1)$ seconds and sends follow-up at 544 $(T2)$. SC receives this message at 18 $(T3)$ seconds and the follow-up at $21(T4)$ seconds.

Now at this stage, SC knows $T3$ and $T4$ and came to know about $T1$ by sync and follow-up messages. Now SC calculates $T2 = 544$ as per equation (3.20). SC will change its clock to 544 seconds. This new time has inaccuracy equal to propagation delay. Now to fix this propagation delay SC sends a "Delay Request" message which is received by GMC at 552 $(T6)$. GMC immediately sends "Delay Reply" including $T6$ in payload as time stamp. Now SC will receive this reply at $T7$ and makes its 2nd fix as per equation (3.21) where, propagation time = 2 seconds. By adding this delay in its time clock it gets synchronized with GMC.

B. Lightweight Time Synchronization: Lightweight Time Synchronization mechanism (LTS) can correct the clock of wireless nodes or subset of nodes applying minimum overhead. Only a single request and reply message is required to synchronize a slave node clock as shown in Figure 3.11 and the solved example of Figure 3.12.

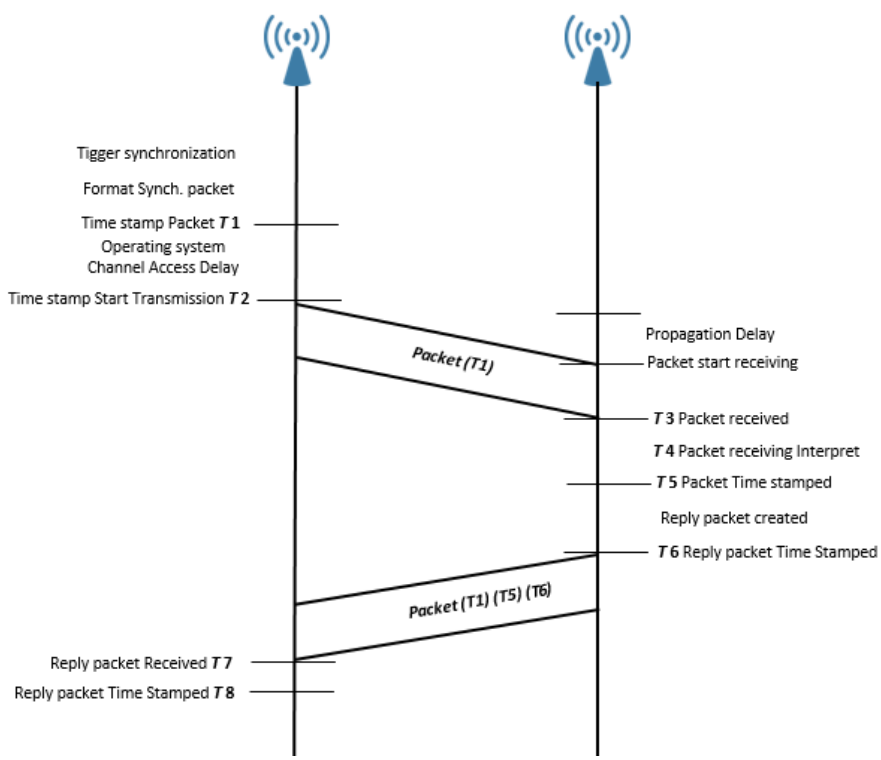


Figure 3.11: Time synchronization by LTS [123].

Node i			**Node j**		
Trigger syn.	540		18		
T1-Stamp	**541**		19		
Format packet,	542		20		
T2 – start	543		21		
	544		22		
	545	(T1)	23		
	546		24		
	547		25	T3	End transmission
	548		26	T4	Receive
	549		27	T5	Stamp message
	550		28	T6	**New message stamp**
	551		29		Handed over
	552		30		Starts
	553		31		
	554	(T1) (T5) (T6)	32		
	555		33		
T7-Receive	556		34		
Interrupt	557		35		
T8- Stamp	558				
	559				
	560				

Figure 3.12: Working of LTS.

Figure 3.11 shows all steps taking by the system to synchronize its clock. A node starts the synchronization before stamping its first message at time $T1$ and ends up the process at $T8$. Between $T1$ and $T8$, the sequence of steps is shown in Figure 3.11 and time activities are also mentioned. All times mentioned in Figures 3.11 and 3.12 may not be used to evaluate the time offset. A solved example is shown in Figure 3.12 for better understanding. More details about figure 3.11 and calculation of 3.12 are shown stepwise as under:

1. The i[th] node should synchronize its clock with the clock of j[th] node
2. Timeline:
   - Node i triggers resynchronization, formats packet, time stamps it at $T1$ and hands it over to transmission (with $T1$ as payload).
   - At $T2$ the first bit appears on the channel, at $T3$ last bit received, packet reception is signaled at $T4$, and at $T5$ node j timestamps it.
   - Node j formats answer packet, timestamps it at time $T6$ and hands it over for transmission, as payload the timestamps $T1, T5$ and $T6$ are included.
   - The arrival of the answer packet to node i is signaled at time $T7$, and node i timestamps it afterwards with $T8$.

If t is transmission time and Δt is offset between node i and j and $T5$ is the time when node j stamps the packet received from node i, can be expressed as:

$$T5 = T1 + \Delta t + t \tag{3.22}$$

And $T8$ is the time when node receive and stamp reply packet as:

$$T8 = T6 + t - \Delta t \tag{3.23}$$

The offset Δt can be calculated by subtracting $T8$ and $T5$:

$$\Delta t = \frac{T_5+T_6-T_1-T_8}{2} \tag{3.24}$$

- Node i can figure out the offset to node j based on the known values , $T1$, $T5$, $T6$ and $T8$.
- This synchronization exchanges only two packets. if node j should also learn about the offset, a third packet is needed from i to j carrying Δt.

$$\Delta t = \frac{27+28-541-558}{2} = -522$$

3.7 Developed System Algorithm

The overall working of the developed system is explained in state diagram 3.13 where homological sequences of steps are shown. The variables used in the algorithm are listed in Table 3-1.

TABLE 3-1: THE LIST OF VARIABLES USED IN STATE DIAGRAM

Variable	Meaning
CCH	Set of cluster heads
CID	Id of CH broadcast by UAV
D_1	Distance for localization at least 120 m we are taking it 200m
D_2	Distance for data collection [95 m]
D_{2L}	Minimum possible distance for data collection [170 m]
D_{2H}	Maximum possible distance for data collection [170 m]
E_i	Energy of node i
F_1	1st Frequency 433 MHz
F_2	2nd Frequency 4.2 GHz
g	GPS accuracy
ID_i	Id of crop sensor i
$\mathscr{b}_i$	Data of node i
$\mathcal{L}_i$	Location of node i
m	Number of antenna elements
n	Number of Sensor nodes ($S1, S2, \ldots, Sn$)
$P\mathcal{L}_i$	Probability of node i by know its location
q	Number of CCH nodes
$T\mathscr{b}$	Total data of a cluster
Th	Threshold set by UAV, default = 0
TE_{Tx}	Total energy requires to send $T\mathscr{b}$ bits to D_2 distance
T_{ys}	Type of sensor node announce by UAV
P_i	Probability of node i to become a cluster head

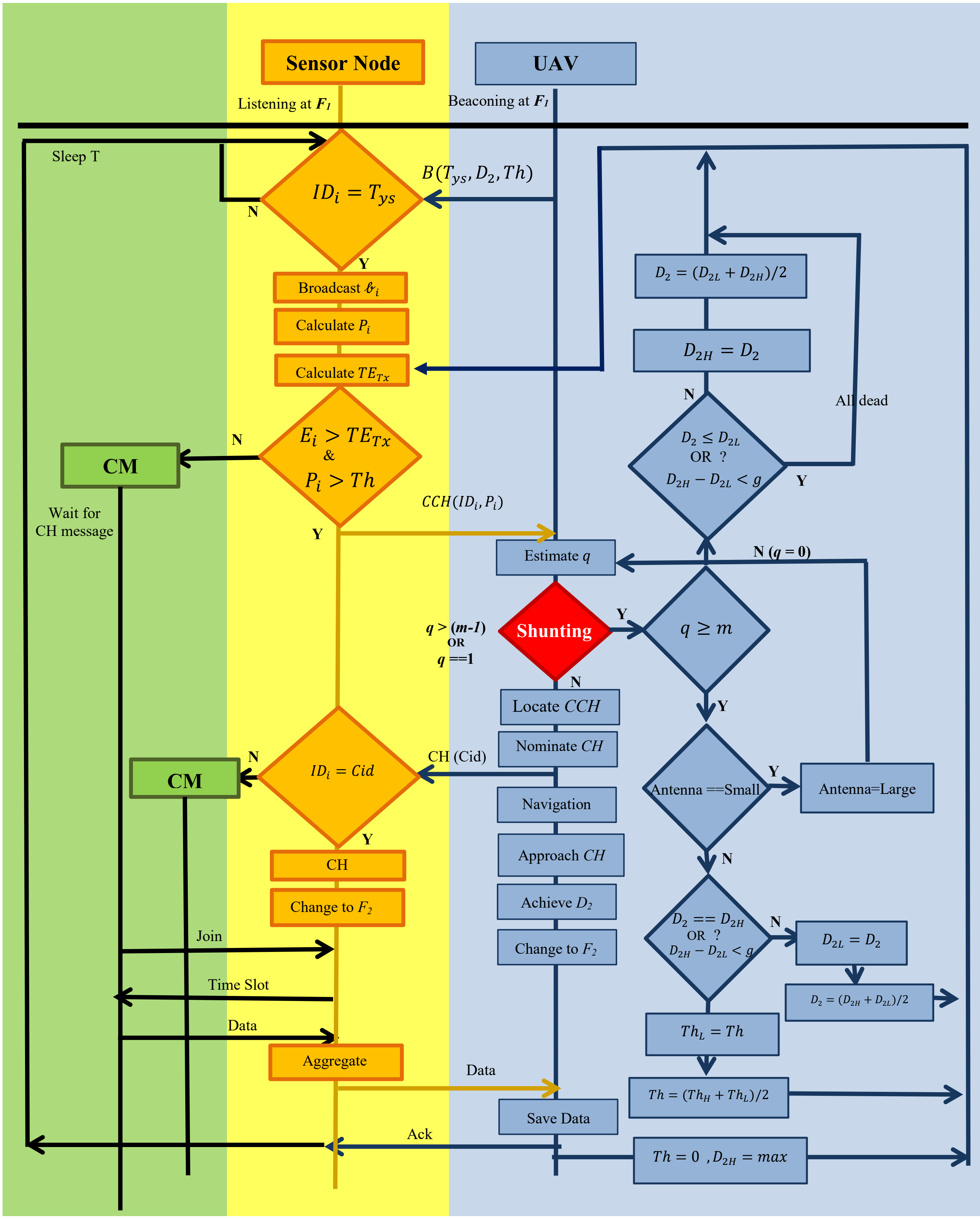


Figure 3.13: State diagram of the developed system.

As shown in Figure 3.13, state diagram blue color represents the top layer, Yellow is middle one and green represents the last layer. UAV needs following inputs to start mission and data collection procedure.

1) Path of UAV consists of waypoints $\{(x1\,,y1)\,,(x2\,,y2)\,,(x3\,,y3)\ldots.\}$,

2) D_1 height, UAV normally flies at this height and conducts localization and navigation (e.g. 200),

4) D_{2h} is maximum possible height for data collection (e.g. 170 m),

5) D_{2L} is lowest possible height for data collection (e.g. 20 m),

3) D_2 data collection height, UAV only obtain this height to establish a backbone high data rate link with CH to collect data, after that UAV goes to its normal height D_1. Initially D_2 should mainly be set to an average height (e.g. 170+20/2 = 95 m),

6) th the threshold value $0 \leq th \leq 1$,

7) Th_L denotes minimum threshold,

8) Th_H stands for maximum threshold and

9) g is the GPS accuracy (e.g. 5 m).

UAV is equipped with virtual phase array antenna which can operate at two different modes Low/High, Low means with least specification can locate fewer targets and High means maximum specifications.

UAV sends beacon message to activate ground sensors. Beacon is composed of three basic kinds of information T_{ys} is the type of nodes need to be activated, D_2 distance at which UAV will collect data and Th is threshold. Algorithm steps are as under:

1. The nodes not mentioned in this beacon message continue sleeping, while other will become activated.
2. The activated nodes will broadcast a message to all its neighbors to let them know how much data and energy they have.
3. All activated nodes will calculate cost of transmission TE_{Tx} as per equation (3.9) and their probability to be cluster head P_i as per equation (3.17). All the nodes having energy greater than TE_{Tx} and get probability higher than threshold will declare themselves Candidate Cluster Heads (CCHs) and remaining will be Cluster Members (CM).
4. CCHs will proceed further and start sending a narrowband signal to UAV having P_i and their ids.
5. When UAV got reply from ground sensor nodes, it will use its, on board virtual antenna and estimates the number of replying sensor nodes q. If UAV found $0 < q < m$ then it will go for localization, to estimate the location of these CCHs, otherwise it will starts shunting process.
6. Shunting is a process to pull or push some sensor nodes from the process to keep their number in some reasonable range $(1\ to\ m-1)$, m is the total number of virtual antenna elements. Shunting

process is shown by red diamond in state diagram (see Figure 3.9). For shunting, UAV have 3 options that it will use in steps:

a. The first option: UAV is equipped with adjustable virtual antenna, that can operate at two different modes fast mode with minimum localization ability and high specification mode. If UAV found many CCH nodes contesting for Cluster Head (CH), it will switch its mode to high performance mode.
b. If UAV found that number of replying CCHs are even greater than high performance antenna capacity then it will try to reduce the number of CCHS by increasing data collection height D_2. The D_2 can also be used in reverse ordered if no sensor node is contesting for CH, it means no one has the ability to send aggregated data to UAV at this distance in this case, D_2 can be decreased up to a minimum level (safe flight).
c. D_2 is always kept in middle of heights height D_{2H} and Lowest D_{2L} as $D_2 = (D_{2H} + D_{2L})/2$. We are shrinking and expanding low and high values as per requirements, if found many CCHs and want to limit them by increasing the height as Low value is shifted to middle and middle value is calculated again. Same treatment in reverse order if no CCH found, data collection height is decrease by shifting high value to middle and middle is calculated again.
d. This D_2 height tuning process keeps on till:
    i. Number of CCH becomes in range (1 to m-1)
    ii. D_2 reaches the boundary condition extreme high or low
    iii. The distance between high and low becomes less then GPS accuracy let say 5 meters.
e. The third option is the variation of the threshold Th, by increasing Th from 0 to 0.5 roughly half of the nodes will withdraw from CH selection. UAV can increase Th at every iteration.

7. UAV will localize the activated candidate cluster heads and select the best one as CH on the basis of P_i information received and distance with next waypoint. The node having highest P_i value and closer to waypoint will be selected as CH.
8. UAV will send CH message to all activated nodes.
9. The nodes receiving CH message will send join request.
10. The CH receives join request and allocate time slot.
11. The nodes receive time slot and send its data.
12. CH aggregates the data and transmit it to UAV.
13. UAV moves to next cluster.

14. All active nodes finish their transmission, and go to sleep for a fixed time.

3.7.1 Development of the sensor node

The most important component of our developed system is the sensor node. The whole the system is developed in such a way to facilitate the tasks of this component and let it optimize its resources to monitor the crop parameters for longer time. In our developed system, daily routine working of sensor node is limited to sense and sleep. It becomes only active for communication after receiving beacon message from UAV. The sensor node working process is expressed in an algorithmic form as follows:

Sensor node algorithm

Default Values:

$F_1 = On$, $F_2 = Off$

Procedure:

$Listening\ Beacon\_msg(\ )$

$If\ \left(Get\ Beacon(T_{ys}, D_2, Th)\ \&\&\ (ID_i \neq T_{ys})\right)$

$\quad Sleep(T_{time})$

$Else$

$\quad Boradcast\ (E_i\ , K_i)$

$\quad T\mathcal{b} = T\mathcal{b}(SS) = \sum_{i=0}^{O} \mathcal{b}_i$

$\quad TE_{Tx} = (E_{elec} \times T\mathcal{b}) + \left(E_{amp} \times T\mathcal{b} \times (D_2)^2\right)$

$$\mathrm{P}_i = \frac{\mathrm{P}_{i1}.\,\mathrm{P}_{i2} \ldots \ldots \mathrm{P}_{iz}}{\mathrm{P}_{i1}.\,\mathrm{P}_{i2} \ldots \ldots \mathrm{P}_{iz} + (1 - \mathrm{P}_{i1}).\,(1 - \mathrm{P}_{i2}) \ldots \ldots (1 - \mathrm{P}_{iz})}$$

$\quad If(E_i < TE_{Tx}\ OR\ P_i < th)$

$\quad\quad$ Set $Status = CM$

$\quad\quad Wait\_untill\left(Get\ msg\_CH(CID)\right)$

$\quad Else$

$\quad\quad Loop\ while\left(Get\ msg\_CH(CID)\right)$

$\quad\quad\quad Set\ Status = CCH$

$\quad\quad\quad Signaling\_UAV(ID_i, P_i)$

$\quad\quad End\ while\ Loop$

$\quad End\ If$

$\quad If\ (Get\ msg\_CH(CID))$

$If(CID == ID_i)$

$Set\ Status = CH$

$Switch\ on(F_2)$

$Loop\ While(\tau_{time})$

$Get\ msg\_Join\&Syn(S_{id}\ , T_1)$

$Send\ msg\_Reply\ Join\&Syn(T_2)$

N_CM^{++}

$end\ While$

$Send\ msg\ (TDMA\ schedule)$

$Loop(\ 1\ to\ N\_CM)$

$Get\ data(Id, payload)$

$End\ Loop$

$Agrigate\ and\ compress(Data)$

$Send\ msg(to.UAV\ , data)$

$Sleep(T_{time})$

$Else\ Send\ msg\_Join\&Syn(to.CID\ , from.ID_i)$

$Get\ msg\_Reply\_Join\&Syn(T_2)$

$corruct\ (clock)$

$get(TDMA\_slot)$

$Send\ data(to.CID\ , from.ID_i, Payload)$

$Sleep(T_{time})$

$End\ If$

$End\ If$

$End\ If$

The graphical or pictorial representation of above-mentioned algorithm is shown in Figure 3.14. The process starts when the node receives a beacon then it calculates its Bayesian probability to be a CH and inform the UAV about it. UAV selects one of the activated node as CH, which should collect data from the whole cluster, aggregate and forward it to UAV.

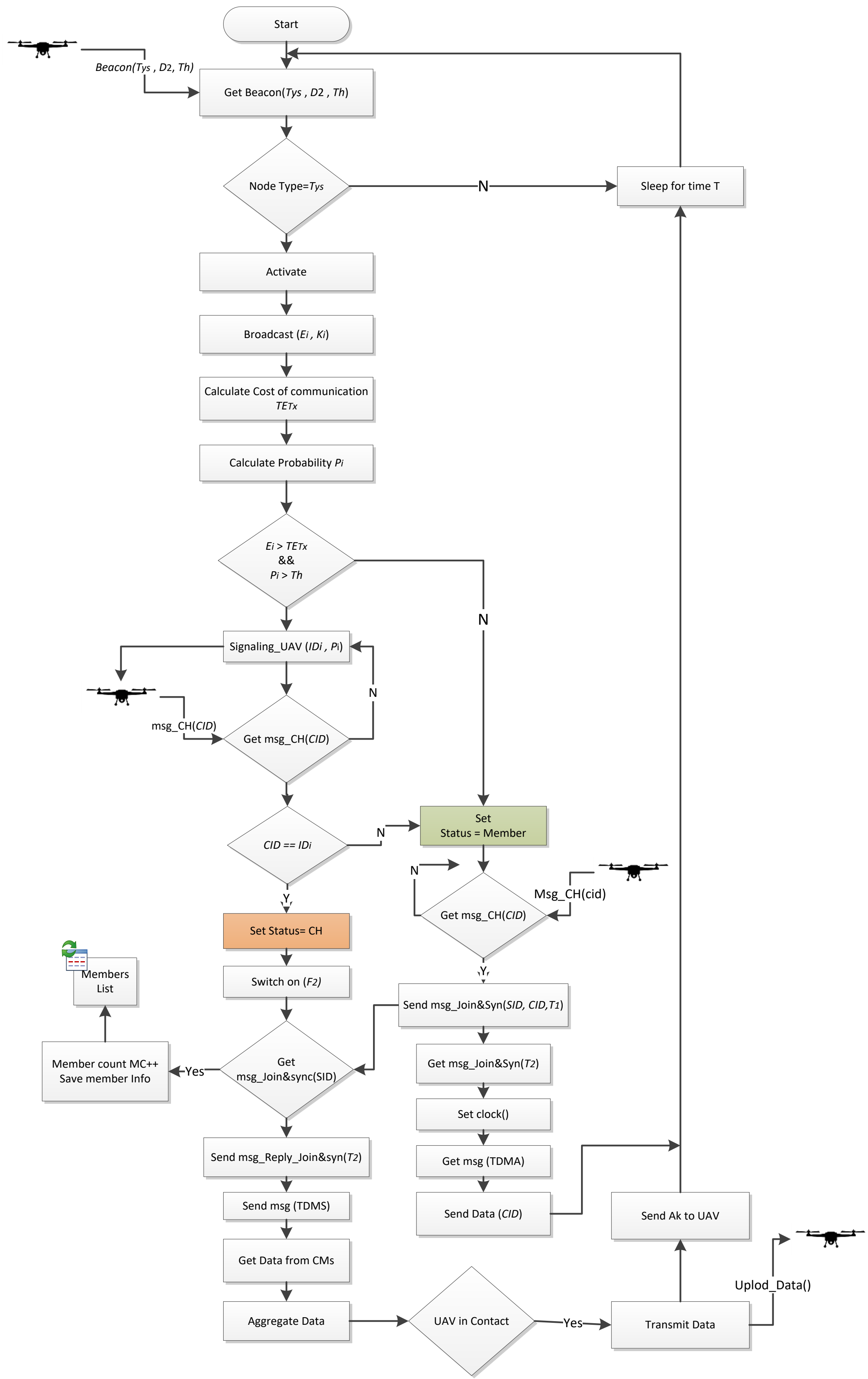


Figure 3.14: Working flow for sensor node.

3.7.2 UAV functionality

UAV role is very important in this developed system. It is designed in such way that it bears most of the processing burden to simplify the tasks of the sensor nodes. It performs many key functions throughout its working cycle: activating the sensor noes, shunting them for further processing, getting nodes initial information, locating all activated nodes, evaluating them, nominating the CH and finally collecting data from the selected CH. Working cycle of UAV is given in an algorithmic form as follows:

UAV Algorithm

Inputs:

$Path\ , F_1\ , D_1, F_2, D_2, D_{2L}, D_{2H}, Th, Th_L, Th_H$

Process:

$\boldsymbol{Point\ A}$:

$Loop(1\ to\ L_n)$

$\quad Send\_msg\left(Beacon(T_{ys}, D_2, th)\right)$

$End\ loop$

$\boldsymbol{Point\ B}$:

$Estimate\ (q)$

$If(q == 0)$

$if\big((D_2 \leq D_{2L})OR(D_{2H} - D_{2L} \leq g)\big)$

$\quad D_{2H} = D_2$

$\quad D_2 = \left(\frac{D_{2L} + D_{2H}}{2}\right)$

$End\ If$

$\boldsymbol{Go\ to\ Point\ A}$

$If\ (q \geq m)$

$\quad If(Antenna == Low)$

$\quad\quad Antenna = High)$

$\quad\quad \boldsymbol{Go\ to\ point\ B}$

$\quad End\ If$

$\quad Else\ If(D_2 \leq D_{2H})AND(D_{2H} - D_{2L} > g)$

$\quad\quad D_{2L} = D_2$

$$D_2 = \left(\frac{D_{2L} + D_{2H}}{2}\right)$$

$\boldsymbol{Go\ to\ point\ A}$

$Else\ if$

$Th_L = Th$

$$Th = \left(\frac{Th_L + Th_H}{2}\right)$$

$\boldsymbol{Go\ to\ point\ A}$

$End\ if$

$End\ if$

$Loop(j = 1\ to\ q)$

$\mathcal{L}_j = Location\ CCH(j)$

$P_j = CCH(j).P$

$ID_j = CCH(j).ID$

$$\mathrm{P}\mathcal{L}_j = \mathrm{P}(\mathrm{CCH}_j = \mathrm{CH}|\mathcal{L}_\mathrm{j}) = \frac{\mathrm{P}(\mathrm{CCH}_j = \mathrm{CH})\ \mathrm{P}(\mathcal{L}_\mathrm{j}|\mathrm{CCH}_j = \mathrm{CH})}{\mathrm{P}(\mathrm{CCH}_j = \mathrm{CH})\ \mathrm{P}(\mathcal{L}_\mathrm{j}|\mathrm{CCH}_j = \mathrm{CH}) + \mathrm{P}(\mathrm{CCH}_j = \mathrm{CM})\ \mathrm{P}(\mathcal{L}_\mathrm{j}|\mathrm{CCH}_j = \mathrm{CM})}$$

$$TP(j) = P_j \times 100 \times \frac{z-1}{z} + P\mathcal{L}_j \times \frac{100}{z}$$

$End\ Loop$

Final_CH = max(TP)

$Broadcast\_msg(Final\_CH(CID))$

$Navigate\ to\ \mathcal{L}_{CID}$

$Set\ Height\ D_2$

$Switch\_on\_\ Frequency(F_2)$

$Receive\_data(from.CID)$

$Switch\_off\_Frequency(F_2)$

$Set\ D_{2H} = max$

$Set\ Th_H = 1$

$\boldsymbol{Go\ to\ step\ A}$

This algorithm is shown in the form of flow chart diagram to observer different program branches and sequence steps. Furthermore, flow chat gives us better understanding of the system.

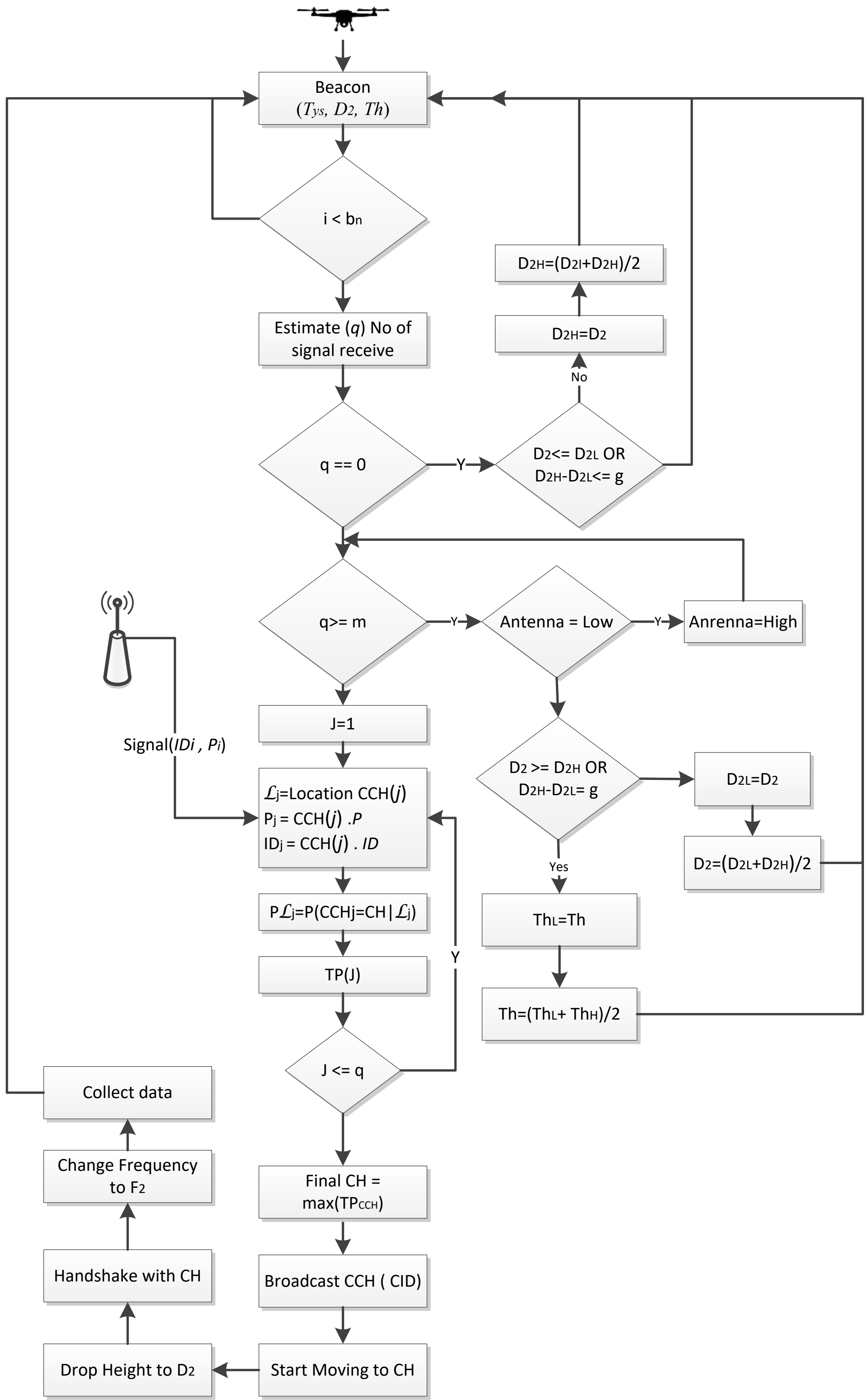


Figure 3.15: Flow chat for UAV working.

3.8 Link Budget

The critical parameter in our application is the energy, in particular the required energy for the header nodes to transmit their beacon and useful information to the UAV. Relationship among energy, distance and frequency is given in equation (3.25) [124].

$$Path\ loss = 32.45\ dB + 20\ Log_{10}(frequency\ in\ MHz) + 20\ Log_{10}(distance\ in\ Km) \quad (3.25)$$

We realize that at a large distance from UAV, it is preferable to use a lower frequency to minimize the power consumption of the header node.

In our developed system, UAV and sensor nodes are performing two major types of communication: one is long range communication for identification/ Localization/ Navigation and other is short range for the data exchange. For Long range communication, we don't need a high bandwidth. We are using a bit lower frequency that it can penetrate to larger distance with very low energy. High frequency (WiFi) is only uses for data exchange, as data size may be large enough so we need high bandwidth. Short-range communication is more energy demanding and increased exponentially with increasing the distance. In our developed system, we do care about this transmitter receive distance. Link budget for both the frequencies are as under:

3.8.1 *F_1 433 MHz UHF signal*

Sensor node energy is the most important parameter. Let us assume a small radiated power ($100\ \mu W$) is considered for transmission.

$$PW = 100\ \mu W = 10\ log_{10}\ (PW\ mW)dBm = 10\ log_{10}(0.1)dBm = -10dBm$$

Masking factor (by objects like vegetation or other) for UHF long distance is considered very small $MF =\ 20\ dB$

IF receiver sensitivity of a UAV is $\xi\ =\ -130\ dBm$ and a signal to noise ratio necessary to demodulate is *SNR*= 15dB.

The maximal path loss constraint is calculated as:

$$M_{PL} = PW -\ \xi - SNR - MF = -10 + 130 - 15 - 5 = 100\ dB$$

Based on equation (3.25), we can calculate the maximum distance covered by UHF communication:

$$20\ log_{10}(D_1) = 100 - 32.45 - 20\log(F_1) = 100 - 32.45 - 52.7 = 14.85$$

$$D_1 = 10^{0.7425} \cong 5.527\ Km$$

The maximum range between the UAV and the header node is $D_1\ =\ 5.527\ km.$

In our developed system, we don't need to communicate over a long distance of 5 Km. If we are considering a UAV is flying at a height of 300 m and it wants to activate nodes in 500 meter area then only 0.825 μW Power is require to transmit F_1 signal by UAV and sensor nodes to communicate.

3.8.2 F_2 2.4 GHz WiFi signal

In a similar way, we can find the maximum distance covered by WiFi signal taking all parameters as considered for F_2 frequency except higher masking factor that is 20.
Receiver sensitivity of a UAV is $\xi = -130\ dBm$
A signal to noise ratio necessary to demodulate is $SNR = 15dB$
The maximum path loss constraint is:

$$M_{PL} = PW - \xi - SNR - MF = 85\ dB$$

Based on equation (3.25), we can calculate the maximum distance covered by WiFi communication

$$20\ log_{10}(D_2) = 85\ dB - 32\ dB - 20\ log_{10}(F_2) = (85 - 32.45 - 67.6)dB = -15.05dB$$

$$Log_{10}(D_2) = -\frac{15.05}{20} = -0.7525 \Longrightarrow$$

$$D_2 = 10^{-0.7525} \cong 0.176 \text{ Km}$$

Maximum range between the UAV and the header node for data collection is $0.177\ km$. We will then consider a maximum distance in the (x, y) plane of the order of $xy_{UAV} = 150m$.

3.9 OMNeT ++ Simulation

OMNeT++ (Objective Modular Network Test bed in C++) is an object-oriented modular discrete event simulation framework, publicly available since 1997. It has a generic architecture, so it can be used in various problem domains including wired and wireless communication networks, on-chip networks, queueing networks, and so on. OMNeT++ has eclipse based IDE and graphical runtime environment that makes it more attractive and easy to use. For more details and basic learning please see Appendix C.
We chose OMNeT++ to simulate developed system. It is module mode network simulator and different modules of our project included UAV, sensor nodes, dynamically selected CCHs and CHs can be build, simulate and visualize easily to evaluate the proposed system. In addition, mobility of UAV and its impact on field sensors can be easily managed, seen and evaluated. Our developed simulation is consisting of two components: sensor nodes and UAV.

3.9.1 Sensor node simulation model

Simulation model for wireless sensor node is developed using OMNeT++ components in “node.ned” file and required algorithm given in section 3.7.1 is implemented using C++ programming in “node.cpp” file. As per our designed protocol, wireless sensor node is equipped with two different frequencies 433 MHz for localization and identification of nodes and 2.4 MHz for data transmission. We developed sensor node simulation model using two NICs CC2420 [125] and CC1021[126] (see Figure 3.16). CC2420 is 802.15.4 compliant Network Interface Card (NIC) operating at 2.4 GHz frequency and having built-in CSMA/CA in MAC[18] layer. While CC1021 is Low Power RF Transceiver for Narrowband Systems operating at 433 MHz

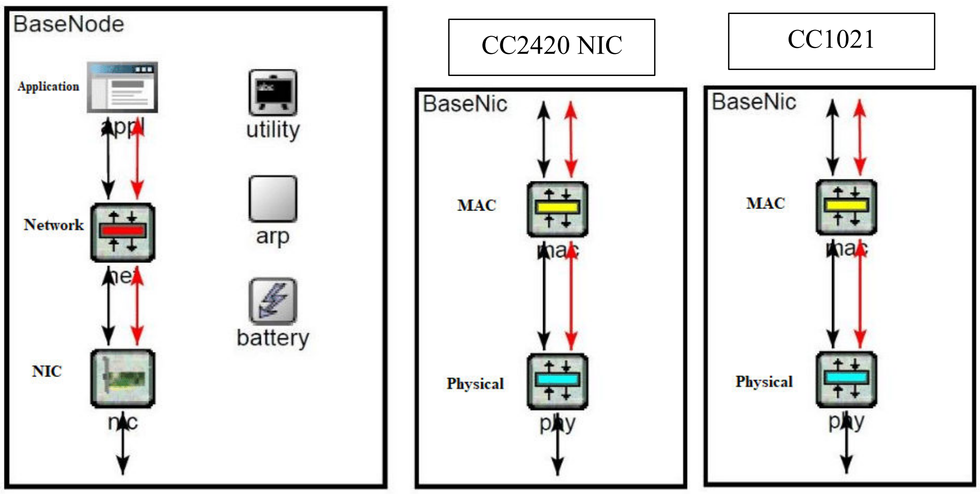


Figure 3.16: Sensor Node and installed NIC.

The sensor node is considering as smart enough that it can negotiate communication height with UAV by considering amount of data needed to transmit and remaining energy power level. It utilizes only 0.825 μW energy while operating with CC1021 for localization. The largest energy depletion factor is data transmission between CH and UAV using higher frequency e.g. WiFi. If considered node is selected as CH then energy consumption of CC2420 is optimized by limiting CH-UAV agreed height. UAV is taking data at a lower height suitable to CH.

CM nodes consume 100 μW energy for inter-cluster communication.

In this way, sensor node was expected to utilize its energy more precisely as it is the only critical parameter of wireless sensor node that may affect the whole network life time. Our expectation becomes true after analyzing the simulation results.

[18] Medium Access Control (MAC) is a 2nd level protocol in WSN protocol stack. It regulates access to the shared medium.

3.9.2 UAV simulation model

UAV model is created with same NICs as in sensor nodes. The only difference is addition of mobility components in the main module as shown in Figure 3.17.

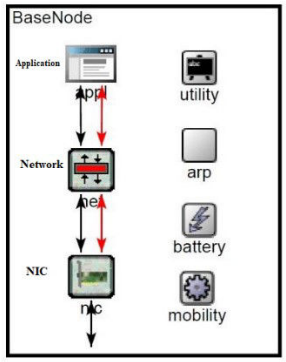


Figure 3.17: UAV simulation model with mobility.

UAV is simulated to fly with a constant speed of 20 m/h and a height of 200 m. Initially it switches on its CC1021 module and starts sending beacon message. Once UAV identifies the location of the selected CH, it starts moving towards it and reduces its height up to an agreed level. When UAV approaches the CH, it will change its communication module from CC1021 to CC2420.

3.9.3 Simulation cases

As described in section 2.6, our case scenario lies in the category of MSSN (Mobile Sink Static Node) network where sink is traversing the network to collect data while all sensor nodes are kept constant. If we are taking sink as mobile node then there are three possibilities (see section 2.6.1):

1) Direct data collection,

2) Cluster based data collection with controlled path,

3) Cluster based data collection with independent UAV path.

In this project, we have developed UAV based data collection where UAV is used to collect data from a specific area and selected sensors. In this scheme, UAV is moving freely and instruct the ground sensor nodes to make clusters and help them to select a cluster head. To evaluate this developed system, we made three simulation cases, one for each type:

1) Direct Data Collection (DDC) by UAV, as shown in Figure 3.18 and described in [74],

2) Network Assisted Data Collection (NADC), see Figure 3.20 as given in [83]

3) Our developed UAV Routing Protocol (URP), Figure 3.19.

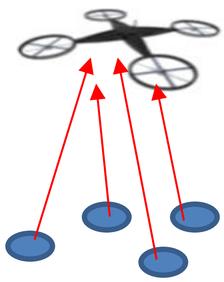


Figure 3.18: Direct Data Collection DDC [74].

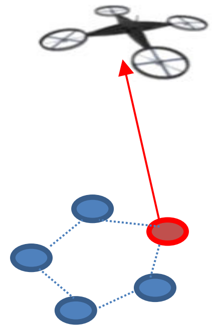


Figure 3.19: Proposed UAV Routing Protocol (URP).

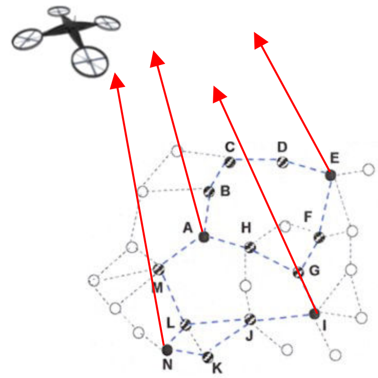


Figure 3.20: NADC [83].

100 sensor nodes are deployed randomly with uniform distribution in 2Km^2 area. A snapshot of simulation is shown in Figure 3.21. Red nodes represent the CH selected in last round, and big blue circle shows the communication range of UAV.

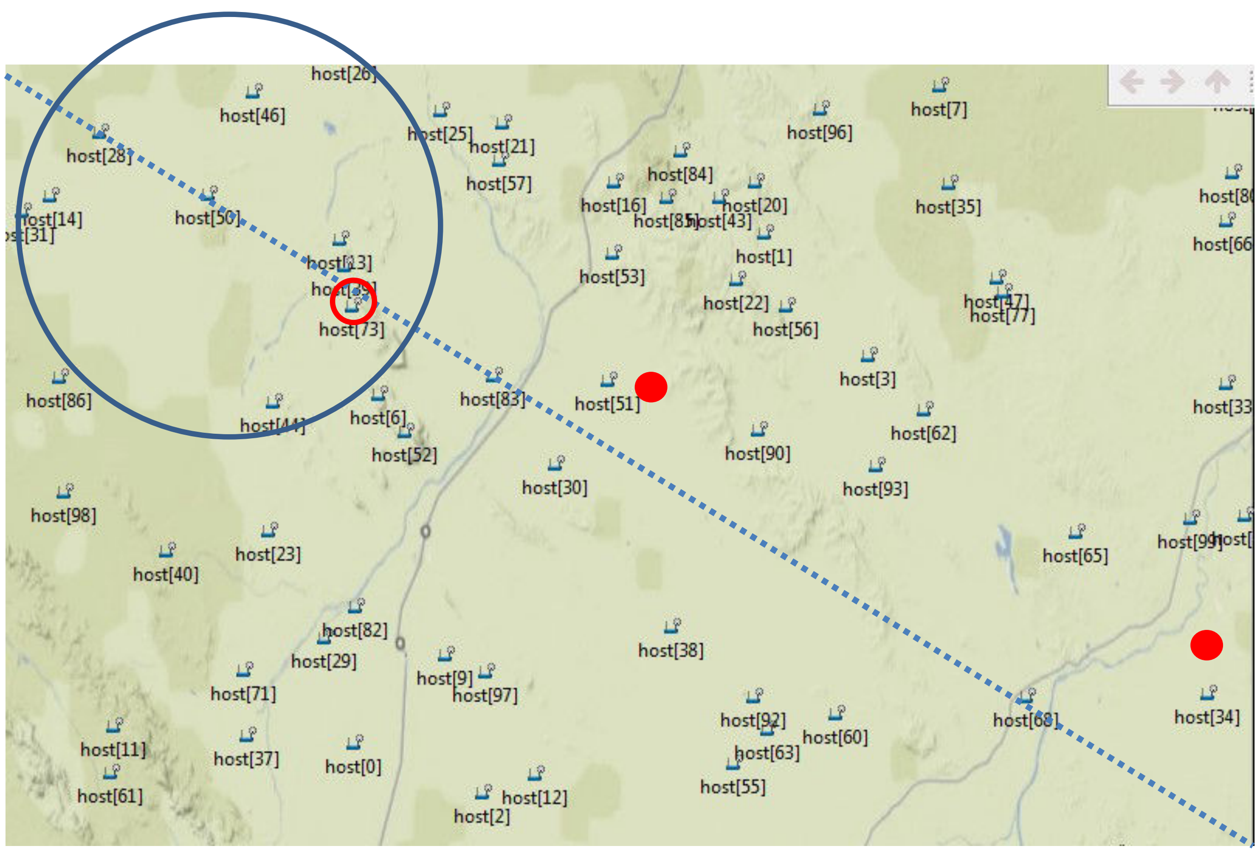


Figure 3.21: OMNET simulation snapshot.

3.9.4 Simulation results

A. Number of dead nodes VS simulation time: In this simulation run, we wanted to check how energy is utilized and how fast sensor nodes become exhausted completely. We simulate each case (DDC, NADC, URP) separately with same parameters. To run this simulation, each node is given 1 Joule battery and 100 nodes are installed in the 2Km^2 area. Initially UAV is flying at height of 200 m and sending beacon every 2 seconds. Each simulation run is executed at least three times to take an average value. The results of each simulation case are shown in Figure 3.22. X-axis is representing the number of dead nodes and Y-axis shows the simulation time in hours, means higher the curve higher the performance.

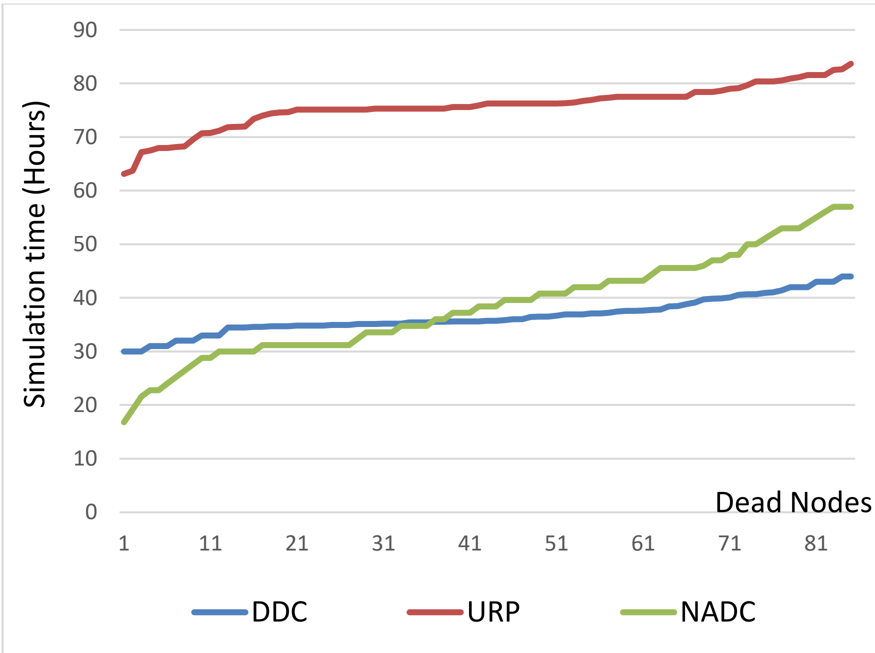


Figure 3.22: Number of dead nodes VS simulation time.

From Figure 3.22, it is observed that in our protocol, the first node becomes dead after 60 rounds. It is much improved as compared to NADC and direct case. In direct contact, data is taken by every node individually and the most energy demanding activity is the long-range communication. The second factor is the height of UAV as UAV is flying at a constant height, every node has to transmit the data to 200 m to deliver it to UAV otherwise it will be considered as dead.

In case of NADC, the first factor is maintenance of network and exchange of frequent updates to keep it updated. The 2nd factor is again the height of UAV as UAV is taking data at certain height and every CH has to transmit aggregated data up to that height.

The life time of our system is observed about 80 simulation hours (red Curve) and in NADC it is 58 (green curve) and we can conclude that the overall performance is improved by 22%. The reason is no periodic updates and the most important is UAV data collection at CH's affordable height. As long as, sensor nodes become aged and running shortage of battery, UAV is decreasing its height accordingly, it can go down to 20 meters from ground level. Another important factor is context aware CH selection, means CH is selected by evaluating all its parameters by Bayesian formula. If a sensor node is equipped with renewable energy (solar panel), the system will prefer it to be a CH. In similar way, if a node has less consumption and more power then it will also get better chance to be selected as CH. Finally, the use of dual frequencies, one with very low power demanding UHF 433 MHz and long range communication also help us to improve the system performance.

We can conclude that using our developed system network, life time can be improved by 20% and more on the expense of adding more intelligence in sensor nodes, dual frequency support dual, adding virtual antenna in UAV and UAV energy/battery.

B. Effect of beacon sending period:

UAV starts data collection process by sending beacon messages to activate the sensor nodes. This beaconing period decides the size of cluster, longer the beaconing period larger the cluster size. This phenomenon can also affect the performance of the system. The developed system is tested against varying beacon sending period and results are shown herein after.

Figure 3.23 represents the total number of dead nodes on X- axis, simulation time in Y-axis and different curves show the performance of system using different beacon periods. Lower the curve better is the performance.

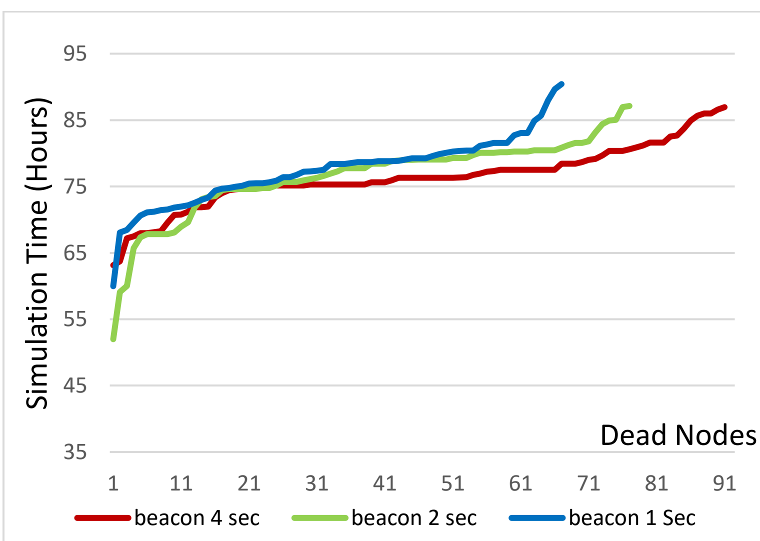


Figure 3.23: Beacon sending period.

Figure 3.24 shows the number of rounds on X-axis and number of cluster heads on Y-axis and different curves reflects the behavior of system with varying number of beacon periods. Higher the curve higher the number of beacons.

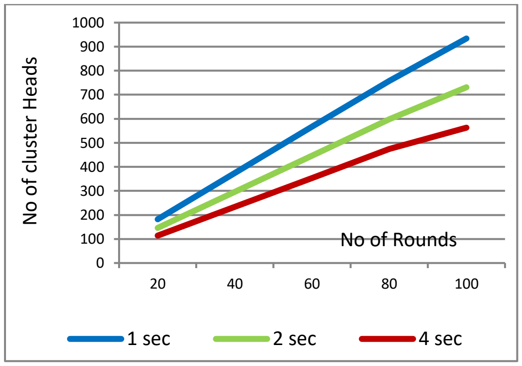


Figure 3.24: Number of clusters VS. Rounds with varying beaconing period.

Figure 3.25 shows the remaining energy of each node at the end of simulation while using 2 second beacon. More the high peaks mean more nodes are unutilized.

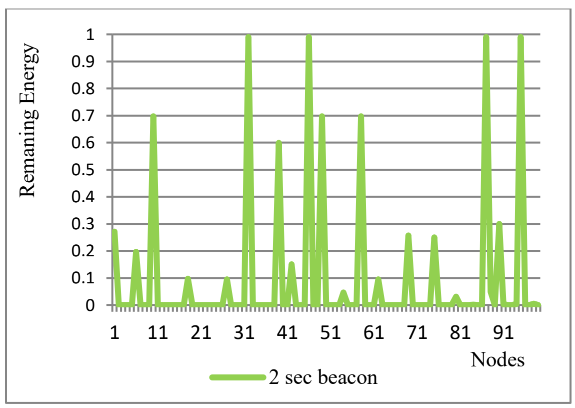


Figure 3.25: Energy utilization of each node in 2 seconds beacon period.

Figure 3.26 shows similar results as Figure 3.25 but we considered have 4 second beacon.

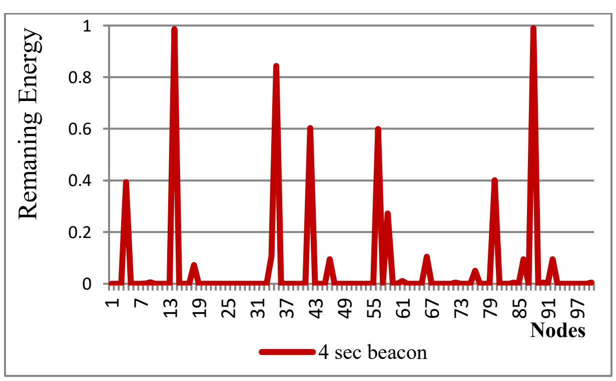


Figure 3.26: Energy utilization of each node in 4 seconds beacon.

The system is evaluated with changing the beaconing period from 1 second to 4 second. It is observed that by increasing the beaconing period more clusters are made as per Figure 3.24; but if 4 second beacon is used about 580 clusters are made in 100 rounds and in case of 1 second beacon number of clusters are 980. Figure 3.25 shows that about 10 nodes are left unattended or unutilized at end of the simulation, while in Figure 3.26 energy utilization is improved in the sense, only two nodes are observed unutilized and few underutilized.

The overall picture of system performance with varying beacon period is shown in Figure 3.23 and it is concluded that maximum performance is achieved with 4 second beacon period where maximum number of nodes are explored.

3.10 MATLAB Simulation

In the starting of this chapter, we have discussed in detail three existing schemes LEACH, HEED and network assisted for data gathering (see sections 3.2, 3.3 and 3.4). Then, we presented our developed UAV based routing and data gathering scheme in section 3.5. We conducted also MATLAB simulations of all these schemes and compare them with our developed system for performance evaluation.

The simulation input parameters are as under:

We took 1 Km^2 area to disperse 100 nodes randomly. Initially, UAV is set to fly at 400 m high and covering 10 m radius on ground. Snapshot of simulation test bed is shown in Figure 3.27 and the simulation procedure/code is given in appendix D.

Sensor nodes are represented by 100 green color dots, UAV is represented with yellow color dot and red color circle is its range. In the figure, S shape lines pattern shows the path of UAV in the form of 10 waypoints. X-axis and Y-axis represents the x, y position of sensor nodes and UAV.

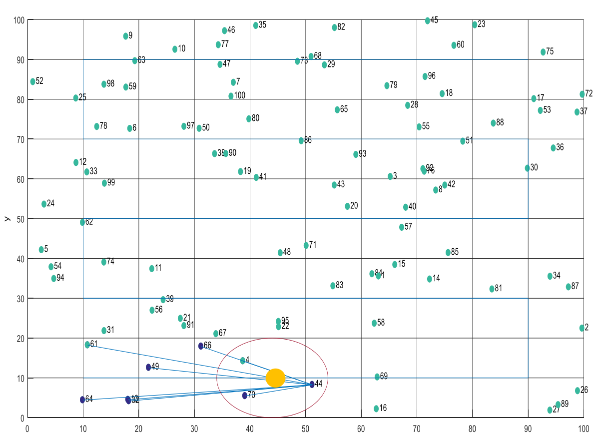


Figure 3.27: MATLAB simulation 2D view.

MATLAB simulation was made in 3-D plane. To explain the working of developed system and cluster formation, some 3-D views of the system are shown in Figures 3.28, 3.29 and 3.30 from different angles.

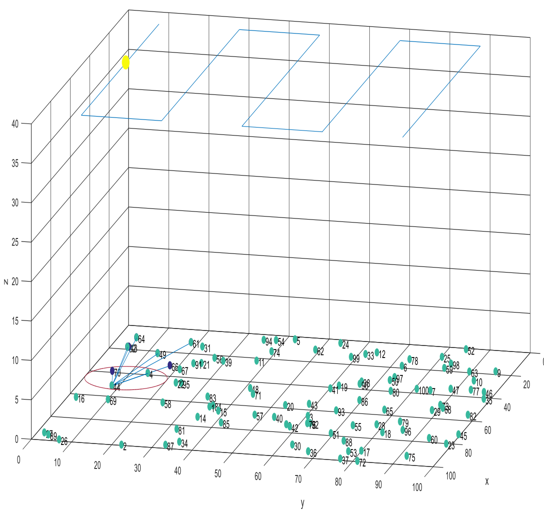


Figure 3.28: MATLAB simulation 3D view-1.

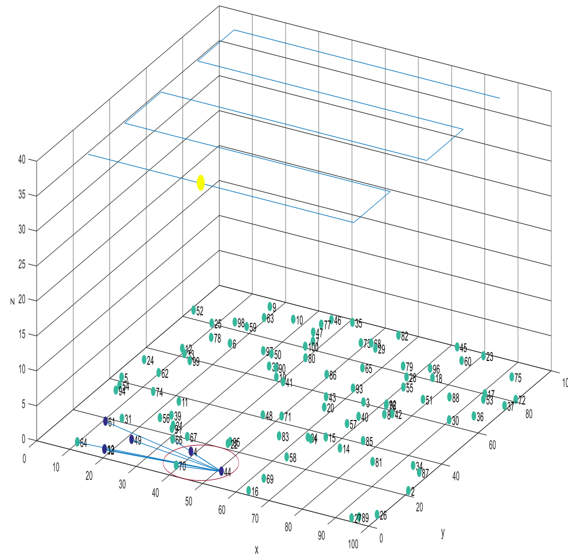


Figure 3.29: MATLAB 3D simulation view-2.

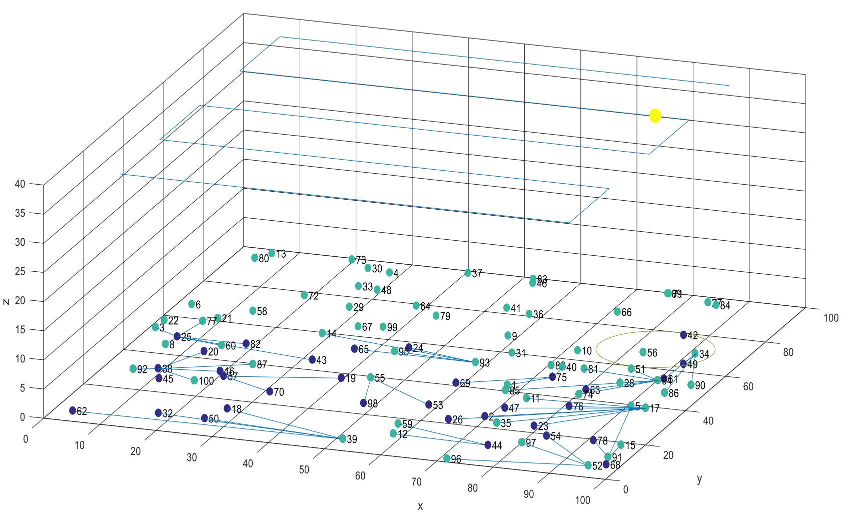


Figure 3.30: MATLAB simulation 3D view-3.

UAV is represented by a yellow dot and straight lines on the same level represent its path. On the ground level, blue dots with numbering shows the installed sensor nodes and star line connected lines are the clusters. Each cluster is shown by blue connected lines. Different simulation models are executed with varying parameters and some results are presented here.

A. **Dead nodes investigation:** Figure 3.31 shows the comparison of dead nodes VS UAV rounds in different routing schemes. Number of rounds are shown in X-axis, number of dead nodes are shown on Y-axis. Different colored curves are representing different data gathering schemes. Lower the curve higher the performance. Red and purple lines are showing the performance of proposed UAV Routing Protocol (URP). We tested the developed system at constant and fixed height (UAV is not going down to take data) as in case of other networks (LEACH and HEED) and results are shown in red color curve. When UAV is flying at a fix constant height even then our proposed system is performing quite good thanks to the facts that there are no periodic updates, no flooding of information, better CH selection, duel frequency use, etc. Purple line shows the performance of proposed system with adaptive height. UAV and CH negotiate a suitable height for data collection. As long-range communication is the main source of energy depletion, adjusting it results in a good impact on overall system life time.

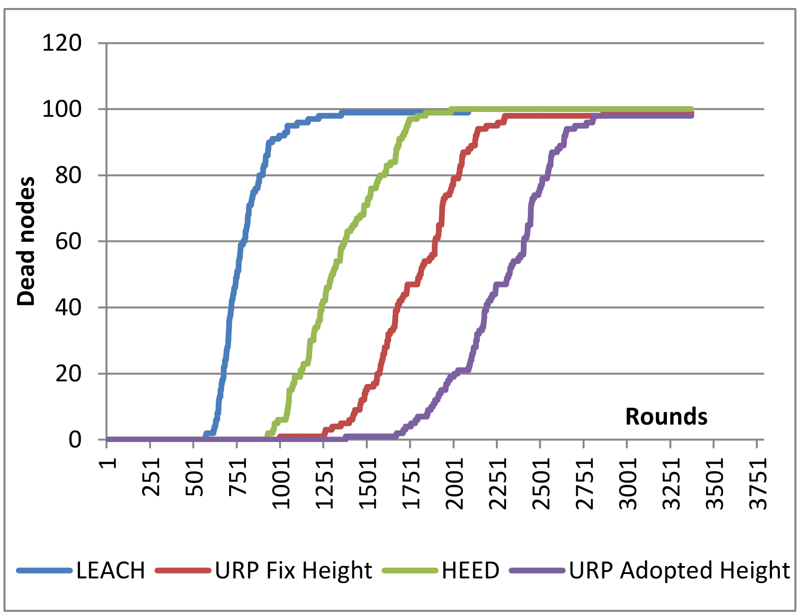


Figure 3.31: Number of dead nodes in proposed system VS HEED and LEACH.

B. **Inter-cluster communication assessment:** Number of packets exchanged among clusters to build a network are shown in Figure 3.32. Number of rounds are shown on X-axis, packets delivered are represented by Y-axis and colored curves shows different protocols. Lower the curve better is the system. It is observed that about 140k packets are exchanged among cluster members and cluster heads in 1000 rounds just only to keep network live and updated in the form of periodic updates, clustering information and CH notifications. In the proposed system, only 20k messages are exchanged, it is the great factor that improved the system performance a lot.

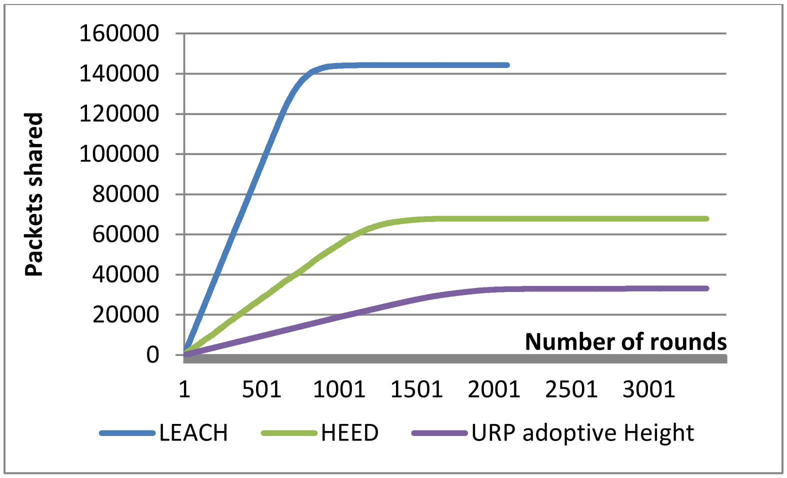


Figure 3.32: Inter-cluster packet exchange VS rounds.

Proof of concept of the proposed system is developed using two devices ground sensor nodes and UAV mounting data sink and the results extracted from real test data are shown in next section.

3.11 Proof of Concept

We used Arduino microcontroller to build the devices because:

1. It is an open source micro controller, easy to program and build.
2. It has many compatible modules and applications.
3. It has different sizes (Uno, Micro control, Mini, etc.) with same interface, if a component is built on UNO it's very easy to change it to micro.
4. Different shieldes (GSM, Motor, MP3, Zigbee, joystick, etc.) are easily available to make a component in layers.

Our project is composed of three main components UAV, ground sensor nodes and UAV sink node. Both of these components are made by Arduino microcontroller. Specialized UAV is also made to carry this equipment and can operate as per instructions given by onboard our developed system, details about UAV preparation is given in chapter 5. Sensor nodes and UAV sink node are shown in Figures 3.33 and 3.34. Corresponding wiring diagrams are shown in Figures 3.35 and 3.36

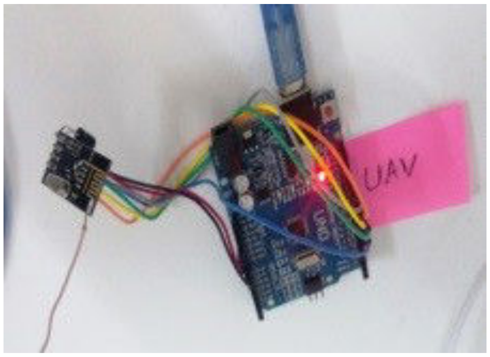


Figure 3.33: UAV sink device .

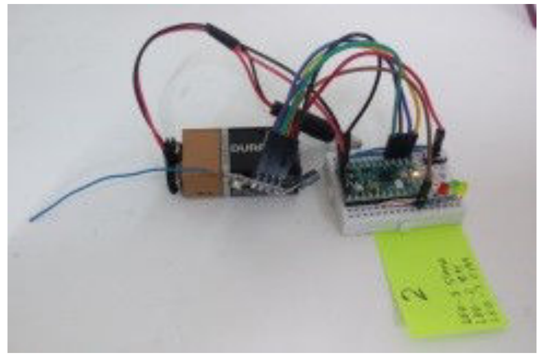


Figure 3.34: Sensor node device.

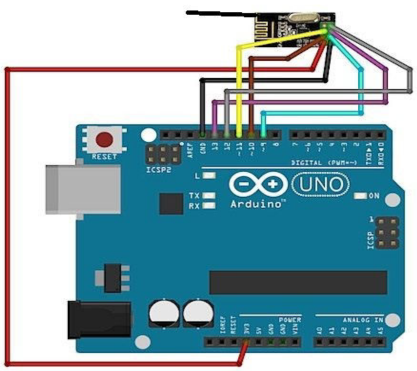


Figure 3.35: UAV Sink unit wiring diagram.

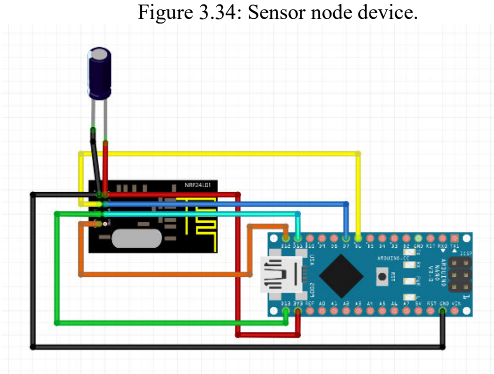


Figure 3.36: Ground sensor nodes wiring diagram.

Both devices are build using nRF24L01 transceiver. nRF24L01 is a low power consumption transceiver operating at 2.4 GHz frequency and capable to transmit data at the rate up to 2 Mbps. Circuit diagram for nRF24L01[127] and wiring information are shown in Figure 3.27. The hardware details are given in chapter 5 and develop software codes for Arduino are given in Appendix E.

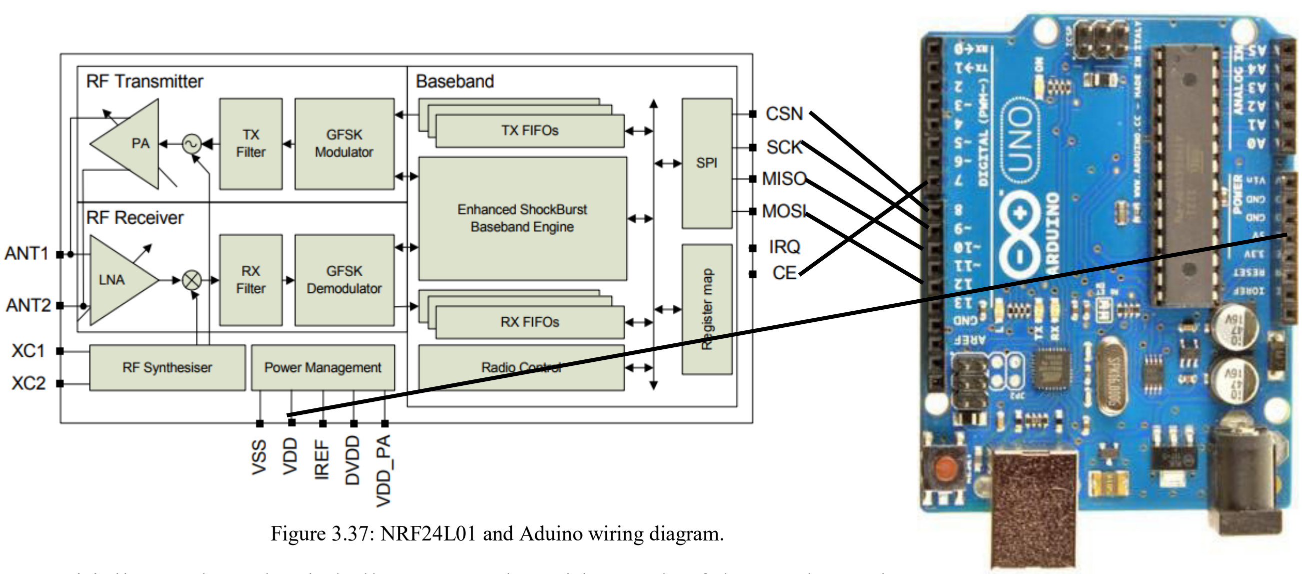


Figure 3.37: NRF24L01 and Aduino wiring diagram.

Initially, we have loaded all sensor nodes with 20 Kb of data and 5 nodes are deployed to install in the field. When UAV approached the field, one of them is elected as CH dynamically and it collects all the data from neighbor nodes and finally transmit 100KB data to UAV.

Localization components including virtual antenna, UHF 433 MHz transceiver for UAV and sensor nodes are still in development process and not installed so far. First test run results are given below. All these results are not including the UAV navigation time.

A. **Working of cluster member node**: The algorithm given in section 3.7.1 is developed in Arduino language and mostly based on C++. The developed code then builds on Arduino mini platform shown in Figure 3.37, that is developed to make a field sensor node. We chose Arduino mini to build a small size of node. Figure 3.38 is the activity graph of ground sensor node that acted as cluster member. It is observed that it takes about 5.4 seconds to complete the cluster formation and data delivery. In that figure, X-axis represents the CM sequence of activities, Y-axis shows time in second and curve represent the relationship of time and activities. The parameters of this node were set as it doesn't have good specification and its CH probability is almost 0, so it decides immediately to set its status as CM.

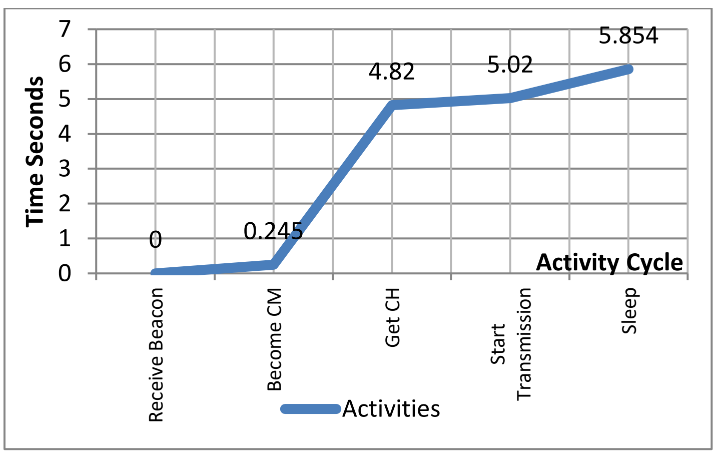


Figure 3.38: Cluster member activities VS time.

Cluster head selection takes about 4.5 seconds and finally data transmission takes less than a second to complete its process.

B. **Cluster head node activities:** Figure 3.39 shows the graph of CH working. It takes about 10 seconds to complete its working cycle. It is observed that this node becomes CH in 5 seconds, during this period, it becomes activated, contacts neighboring nodes and share information with UAV. As this node has been selected as a CH, it has to collect data from all other members and transmit aggregated data to UAV. This whole procedure is conducted in 5 seconds.

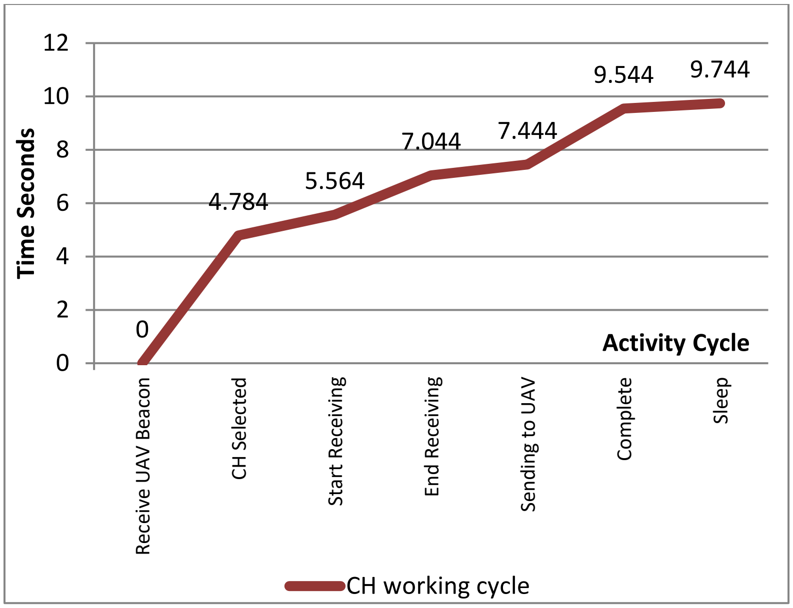


Figure 3.39: Cluster head activities VS time.

C. Working cycle of UAV: Life cycle of UAV is shown in Figure 3.40. It sends beacon message in the first 5 seconds, then switch to discovery phase to search for a suitable CH. Once CH is selected, it will navigate to approach it and takes data at some reasonable height. The whole procedure takes about 10 seconds.

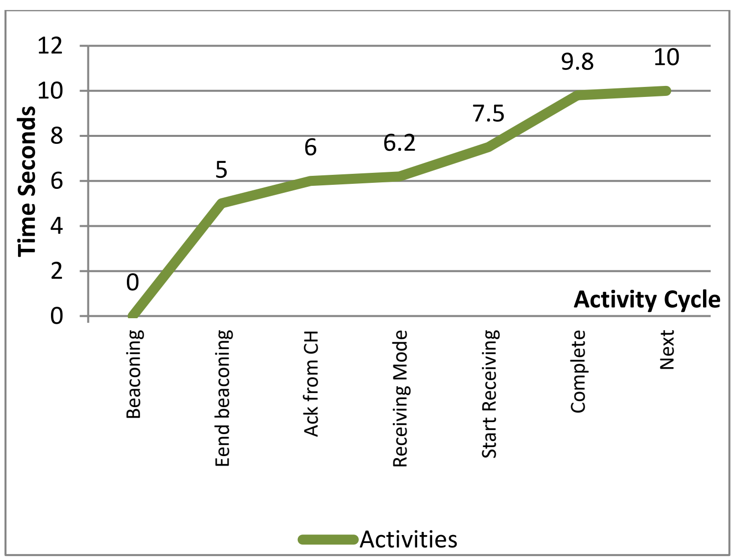


Figure 3.40: UAV activities VS time.

D. Combined activities analysis: Comparison of UAV activities with respect to the CM and CH are shown in Figure 3.41. X-axis represents UAV activities, Y-axis shows time in second and different color curves are different components (UAV, CH and CM).

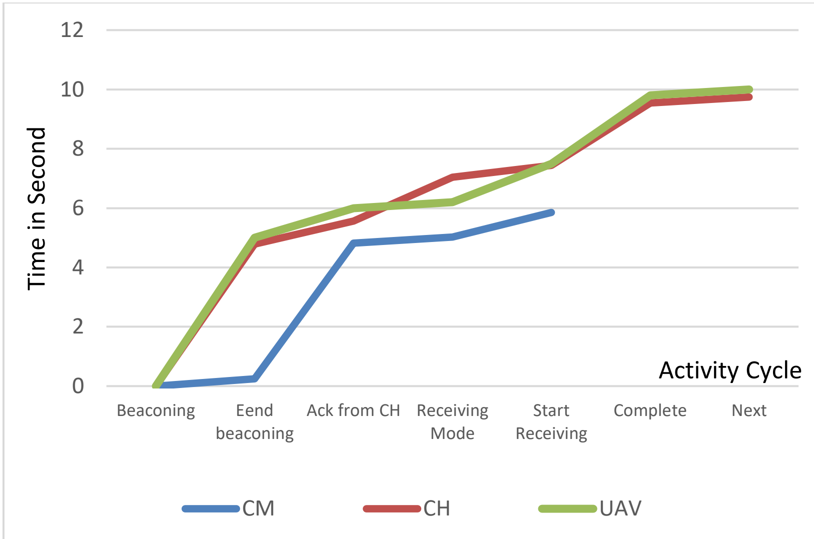


Figure 3.41: UAV activities with respect to other nodes.

E. Effect of varying height on data collection: we evaluate the effect of changing height on the system performance and results are shown in figure 3.42. X-axis shows the height of UAV from 1.2 m to 20 m. and Y axis is representing both, time consumed in second and energy utilization in PJ (Peta Joul).

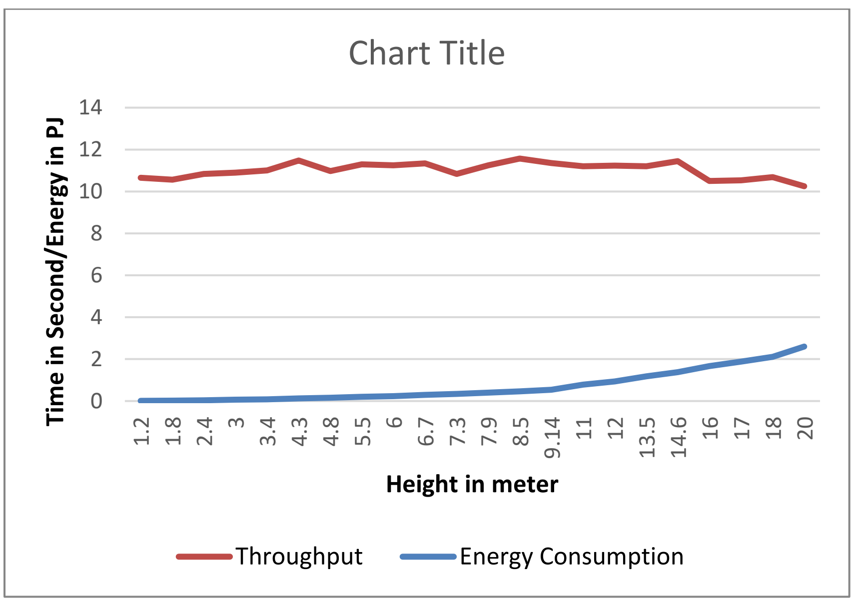


Figure 3.42: UAV data collection at different heights.

By increasing the height of UAV, we could not notice any effect on data collection time but it may have massive impact on CH energy utilization.

3.12 Conclusion

In this chapter, we proposed UAV based routing protocol for crop monitoring. It is considered that heterogynous sensor nodes are installed in the large crop field and only selective data from selective sensors is harvested by UAV. The distinction of this study is that all the alive and active sensor nodes are arranging themselves in clusters dynamically. UAV instructs the ground sensor nodes to make clusters in its way. UAV also nominates one node as CH based on information given by node itself and UAV's own information. All parameters related to sensor nodes and UAV mission are put in Bayesian formula to evaluate the probability of each node to be a cluster head. The best node according to this cost value in term of energy and connectivity is selected as CH. The CH node will aggregate the data from all neighboring nodes and transmit it to the UAV. The proposed system is evaluated by using simulation models and proof of concept devices; It is found that proposed system can optimize the energy utilization of the sensor nodes, does not disturb much the UAV path and enhance the throughput of the system.

The proposed data gathering scheme is published in EUSIPCO 2016 [128]. The modified advanced version is accepted and will be presented in ITNAC 2017 [129].

4

Localization

4.1 Introduction

The problem addressed in this research is data collection from remote agriculture field, where infrastructure in not available. UAVs are chosen to build an instant mean of communication. UAVs solve the most of our data gathering problems as it is cheap in cost, requires no preinstalled infrastructure and can go closer to the sensor node to get data. Additionally, operation is not highly technical. The UAV must be capable to identify and locate the ground sensor nodes accurately. Crop sensors are mostly not furnished with GPS module to save cost and battery. In such kind of sensors deployment, all the burden shifted to UAV to locate a node and collect data. A short introduction about sensor node localization is already given in section 2.7. In chapter 3, we presented a dynamic data gathering scheme to harvest data from heterogeneous crop sensors. In this scheme, UAV can scan a specific area to collect data from selective sensors. The key function of this scheme is the use of UAV to find the data centric nodes to collect their data in an efficient way. We studied many existing localization techniques, some of related ones are introduced in chapter 2, but none of these were fully solving our problem (see section 2.7). The considered schemes, are based on physical multiple antennas. Adding multiple antennas on a small size UAV is not practical, where UAV has to scan many huge crop fields, each one may have average area of 2 Km^2. Considering all these factors and problems, we decided to design a light weight, energy efficient and small size localization system that can easily be installed over a commonly used UAV. In this chapter, we are presenting our developed virtual antenna array system. In this system, we used a single antenna moving with constant speed and height, acting like multiple antennas. Furthermore, we adapted MUSIC algorithm to our new application; in fact, we adjust and modified existing MUSIC algorithm to estimate the angle of arrival of incoming signals from sensor nodes. Analyzing our simulation results, we conclude that moving a single antenna can achieve our goal, with same precision.

4.1.1 Uniform linear array

A Uniformed Linear Array (ULA) of multiple antennas can be used to measure the Angle of Arrival (AoA) of incoming signals. An antenna array can attain higher gain / directivity and narrower the beam of radio waves than it could be achieved by a single antenna. In general, more the antenna elements, higher the gain and narrower the beam. However, the drawbacks of mounting such multiple antennas on an Unmanned Aerial Vehicle (UAV) outweigh the benefits. The use of UAVs in wireless sensor networks is increasing rapidly as it improves the networks in terms of lifetime, connectivity and reliability. The main challenge is to affix multiple antennas and receivers on the UAV which increases its weight and ultimately decreases its payload capacity, flight time, speed and agility. Another challenge is to adapt the precision levels by varying the number of antenna elements or modifying the space among them is not possible in a fixed physical antenna system. In this chapter, we are proposing a new method for angle of arrival estimation based on MUSIC algorithm. This new method takes into consideration the UAV speed and adjusts the AoA accuracy accordingly. Virtual Phased Array (VPA) antenna is proposed to model a typical ULA antenna system by using a single antenna mounted on a moving UAV. This VPA mitigates the challenges resulting from limiting UAV payload and communication throughput, under certain conditions. The proposed system is evaluated through many simulations. Suggested modifications and additions in classical MUSIC algorithm make it possible to operate the virtual antenna system with same precision as the physical antenna and add more flexibility, ease of use, cost economy, more reliability and better throughput.

Background knowledge of developed system is already given in section 2.9. Let us consider an Uniform Linear Array (ULA) antenna with $M = (m_1, m_2, \dots, m_M)$ elements as shown in Figure 4.1, where λ = wavelength, c = speed of light and f_0= frequency of wave

$$c = \lambda f_0 \tag{4.1}$$

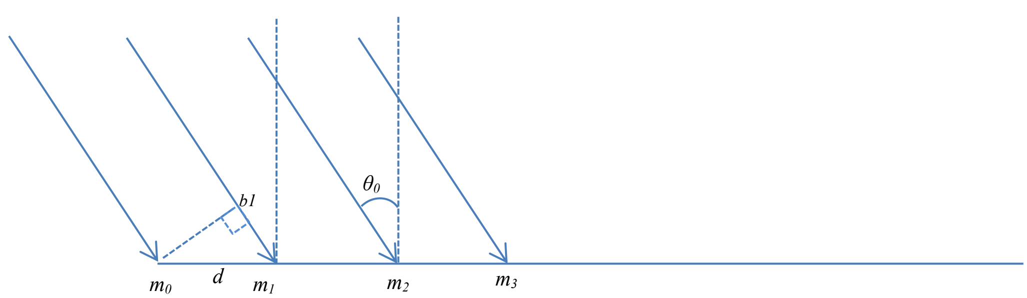


Figure 4.1: Classical direction of arrival by ULA.

Let m_i be the i[th] antenna element and $d = \frac{\lambda}{2} = \overline{m_0 m_1}$ is the distance between two adjacent antennas, in this case, we can write:

$$\overline{m_1 b_1} = l = d \sin(\theta_0) \tag{4.2}$$

Let $s(t)$ be the reference signal received by the first antenna:

$$x_1(t) = s(t) = \cos(\omega_0 t) = \cos(2\pi f_0 t) \tag{4.3}$$

The signal received by the 2nd antenna $x_2(t)$ is delayed by τ with respect to the 1st antenna.

$$x_2(t) = s(t-\tau) = \cos\big(\omega_0(t-\tau)\big) = \cos(\omega_0 t - \omega_0 \tau) \tag{4.4}$$

The delay τ as per equation (4.2) can be written as:

$$\tau = \frac{l}{c} = d\,\frac{\sin(\theta_0)}{c} \tag{4.5}$$

The obtained phase difference due to τ between the array elements becomes:

$$\varphi_0 = \omega_0 d\,\frac{\sin(\theta_0)}{c} \tag{4.6}$$

In similar way, we can prove that phase difference at the ith antenna element is:

$$\Phi_i = (i-1)\omega_0 \frac{d\sin(\theta_0)}{c} = (i-1)2\pi\,\frac{f_0}{c}\,d\sin(\theta_0) = (i-1)\,\frac{2\pi d}{\lambda}\,\sin(\theta_0) \tag{4.7}$$

For a complex signal $x_i(t) = s(t)e^{-j\Phi_i}$ (4.8)

The steering vector of the signal with M elements is defined by:

$$\mathrm{A}(\theta) = \begin{pmatrix} 1 \\ e^{-j\theta_0} \\ e^{-2j\theta_0} \\ \vdots \\ e^{e^{-j}(\mathrm{M}-1)\theta_0} \end{pmatrix} \tag{4.9}$$

4.1.2 Multi sources

Let N be the number of independent sources. The output signal of the m^{th} antenna element is given by the following equation:

$$x_m(t) = \sum_{k=1}^{N} s_k(t)e^{-j(m-1)\varphi_k} + b_m(t) \qquad \text{Where } \varphi_k = \frac{2\pi d}{\lambda}\sin(\theta_k) \tag{4.10}$$

$b_m(t) = AWGN$ (AWGN is Additive White Gaussian Noise) and φ_k is the incident angle of the K^{th} source. In this case, the observation vector can be defined as:

$$X(t) = \begin{pmatrix} x_1(t) \\ \vdots \\ x_M(t) \end{pmatrix} = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ e^{-j\varphi_1} & e^{-j\varphi_2} & & e^{-j\varphi_N} \\ \vdots & \vdots & & \vdots \\ e^{-j(M-1)\varphi_1} & e^{-j(M-1)\varphi_2} & \cdots & e^{-j(M-1)\varphi_N} \end{pmatrix} \begin{pmatrix} s_1(t) \\ \vdots \\ s_N(t) \end{pmatrix} + \begin{pmatrix} b_1(t) \\ \vdots \\ b_M(t) \end{pmatrix} \tag{4.11}$$

In this case, the observation vector can be defined as follows:

$$X(t) = \begin{pmatrix} A_1(\theta_1) & A_2(\theta_2) \dots . A_{M-1}(\theta_N) \end{pmatrix} \begin{pmatrix} s_1(t) \\ \vdots \\ s_N(t) \end{pmatrix} + \begin{pmatrix} b_1(t) \\ \vdots \\ b_M(t) \end{pmatrix} = AS + B \tag{4.12}$$

Where $X = [x_1(t), x_2(t), \dots, x_M(t)]^T$, $S = [s_1(t), s_2(t), \dots, s_N(t)]^T$,

$B = [b_1(t), b_2(t), \dots, b_M(t)]^T$ and $A = [A_1(\theta_1),\ A_2(\theta_2), \dots, A_{M-1}(\theta_N)]$ (4.13)

4.1.3 Multiple signal classification

The author of [130] proposes the Multi Signal Classification (MUSIC) algorithm to find AoA of incoming signals. MUSIC is based on Eigenvalue Decomposition (EVD) of the covariance matrix of observed data which can define the signal and noise subspaces. The two orthogonal subspaces are used to constitute a spectrum function. Finally, AoA of signal are detected by maximizing the spectral criterion.

The eigenvalue decomposition of the covariance matrix R_X, of the observation vector is defined by:

$$R_X = E[XX^H] \tag{4.14}$$

Where H denotes the Hermitian. As assumed before, the transmitted signals and the noise are uncorrelated and they are zero mean.

$$E[B] = E[S] = 0 \tag{4.15}$$

Based on equation (4.13), we can conclude that:

$$R_X = A\, R_s A^H + R_B \tag{4.16}$$

Where $R_s = E[SS^H]$ is the signal correlation matrix and R_B denotes the covariance matrix of the noise. As the noise is AWGN, then we can write:

$$R_B = E[BB^H] = \sigma^2 I \tag{4.17}$$

Where σ is the standard deviation of the noise and I is $M\ \times M$ unit matrix.

As the number of sources N is considered less then number of antenna elements M, then we can conclude that:

$$Rank[R_S] = N \tag{4.18}$$

Let A be a full column rank, then

$$Rank[R_{AS}] = Rank[R_S]$$

Matrix $R_{AS} = AR_S A^H$ should have $(M - N)$ zero eigenvalues.

Let g_m be a vector belongs to the kernel of R_{AS}:

$$g_m \in Ker(R_{AS})$$

In this case, one can write

$$R_{AS} g_m = A R_S A^H g_m = 0 \Rightarrow g_m^H A R_S A^H g_m = (A^H g_m)^H R_S A^H g_m = 0 \tag{4.19}$$

As R_S is a full Rank and positive definite matrix, then we can conclude that:

$$A^H g_m = 0 \tag{4.20}$$

g_m becomes perpendicular to the space of signal steering vectors.

Let us denote by $Q_m = (g_m)$ a $N \times (N - M)$ matrix,

The AoA can be finally obtained by maximizing the following criterion

$$max\ P_{MUSIC}(\theta) = \frac{1}{\sum_{k=1}^{N-M} (A^H(\theta) g_k)^2} = \frac{1}{\|Q_n^H A(\theta)\|^2} \tag{4.21}$$

$A(\theta)$ is orthogonal with each column of noise matrix Q_n^H

Let q_m be an eigen vector of R_{AS}:

$$R_{AS} q_m = \lambda_m q_m \tag{4.22}$$

Based on equations (4.16) and (4.17), we can write the following equation:

$$R_X q_m = R_{AS} q_m + R_B q_m = \lambda_m q_m + \sigma^2 q_m = (\lambda_m + \sigma^2) q_m \tag{4.23}$$

According to equation (4.23), we can conclude that q_m is also an eigen vector of R_X for the eigenvalue $(\lambda + \sigma^2)$. By applying an EVD on R_{AS}, we can write:

$$R_{AS} = Q \Lambda Q^H = Q\ diag\ (\lambda_1, \dots, \lambda_N) Q^H$$

By applying an EVD on R_X, we can prove that:

$$R_X = Q \begin{bmatrix} \lambda_1 + \sigma^2 & 0 & 0 \\ & \ddots & \vdots \\ 0 & \lambda_N + \sigma^2 & 0 \\ & & \ddots \ \vdots \\ 0 & 0 & \dots \ \sigma^2 \end{bmatrix} Q^H \tag{4.24}$$

Using the above equation, one can divide the obtained matrix Q into two matrices Q_S and Q_n such that Q_S is the $M \times N$ matrix and Q_n is $M \times (M - N)$ matrix. In this case, MUSIC becomes equal to:

$$max\ P_{MUSIC}(\theta) = \frac{1}{\sum_{k=1}^{N-M} (A^H(\theta) g_k)^2} = \frac{1}{\|Q_n^H A(\theta)\|^2} \tag{4.25}$$

MUSIC algorithm

1) Estimate covariance matrix: $\hat{R}_X = \frac{1}{K} \sum_{1=1}^{K} X(i) X^H(i)$; $K =$ number of samples
2) Evaluate EVD of $\hat{R}_X \Rightarrow \hat{R}_X = Q \wedge Q^H$
3) Separate Q into two matrixes: $Q = (Q_S \quad Q_n)$
4) Plot a function of θ: $\left\|Q_n^H A(\theta)\right\|^2$

5) The M larger peaks of P_{MUSIC} represent the AoA

4.2 Proposed System

Antenna array systems for AoA estimation have been studied since several decades. Many algorithms and techniques have been proposed so far (see section 2.8), but still have the potential to improve and develop smarter and smaller antennas to make it compatible with rapidly improving WSN and emerging UAV technologies.

In our application, agriculture is taken as a case study where a large number of different sensors are installed in a crop field to monitor various parameters related to plant health, soil and atmospheric conditions. All these sensors do not have GPS and left unattended in the field. Farm fields are in very remote area where communication infrastructures are not available. UAV is used as a mean of communication and to harvest data from these field sensors. For this purpose, it is very important for the UAV to have the complete knowledge about the number of connected sensors and their locations to visit all of them individually or in groups for data collection. Here, we are proposing a virtual phase array (VPA) antenna system installed over UAV to estimate the location of sensor nodes scattered on the ground. This VPA antenna could overcome many existing challenges in classic systems, which are described below one by one;

1. In a conventional way, many physical antennas are collectively used to estimate the AoA of the signal emitted by a sensor. Adding multiple antennas and receivers on UAV increases its weight and results in decreasing its payload capacity, flight time, speed and agility. The first challenge ahead is, how to carry the physical antenna array system by an UAV which is not feasible due to heavy weight and complex structure. In this proposed system, ULA of multiple antennas will be replaced by a single moving antenna acting as a virtual linear array and called hereinafter VLA.
2. Another issue is, how to adapt the steering mechanism which is not possible in a conventional ULA system. For example, during the UAV flight if more precision is required to stare some suspected objects or areas, then proposed system should have the ability to increase the steering precision level by adjusting the length of antenna.
3. How to add an agility feature to our system. This proposed VLA will introduce precession capability to improve the throughput of the system. For example, in normal conditions UAV will fly at a high speed with a minimum steering precession (with minimum array elements), as soon as it found something suspected or of interest, it can increase its steering precession by sampling more data and increasing virtually the size of antenna.
4. In classical ULA, the total number of detectable targets is limited to total number of antenna elements. In VLA, the number of antenna elements is virtually adjustable.

5. The proposed system will be capable to operate on multiple frequencies, and the inter element spacing of ULA antenna can be adjusted accordingly.
6. Typical challenges of the physical antenna that are limiting its performance are as follows: mutual coupling induced current gains and inter-channel phases between array elements does not exist in our proposed VLA.

In the proposed system, UAV will be equipped with a single antenna acting as a linear array of M antennas by exploiting the motion of the UAV. Moving with a constant speed, the UAV can take snapshot of the signals at the fixed time intervals. Finally, the M Snapshots are considered as the outcomes of a physical antenna with M elements, as shown in Figure 4.2.

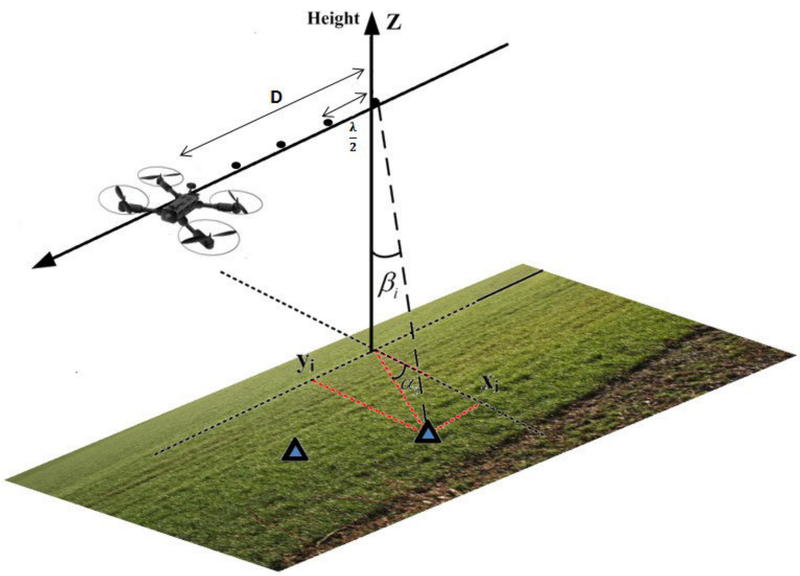


Figure 4.2: Localization of sensor nodes, using virtual phased array antenna system.

4.3 Virtual Antenna Array

A VPA carrying by UAV, we name it Synthetic Aperture UAV (SA-UAV), will be capable to accumulate the data that VPA collects in response of impinging signals with different phases at different times. The parameters (e.g. ToA, AoA) of incoming signals can be estimated if the communication channel during each period of data collection is quasi-stationary. In our proposed system, N target nodes are emitting narrowband signals that are being monitored by UAV at M different places (snapshots) at a fix time period of Δt where $N = M - 1$. UAV will collect R_{over} samples of data between two snapshots. All the received signals are considered to be plane wave as UAV is flying high enough to satisfy far field condition. Designed VPA is shown in Figure 4.3.

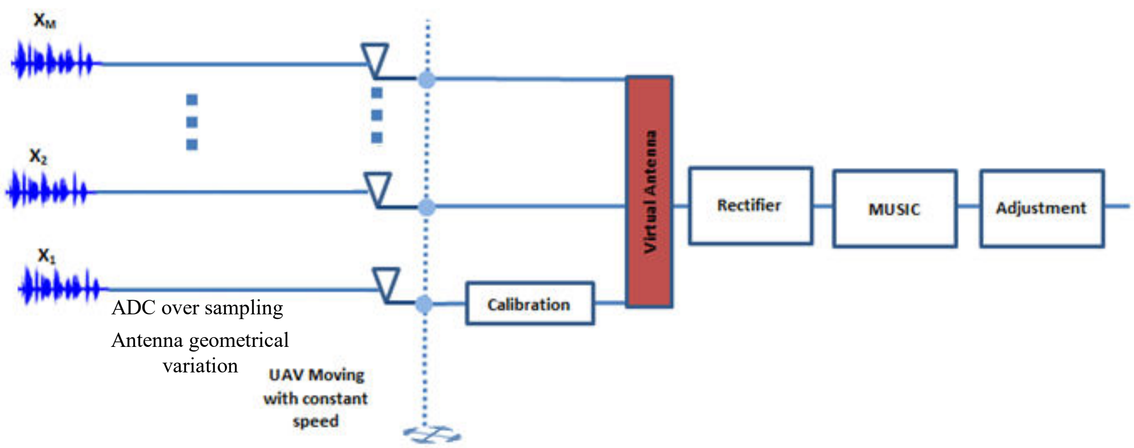


Figure 4.3: Proposed virtual antenna array system.

As shown in Figure 4.3, the signal $x_i(t)$ is transmitted by the ith target node s_i. Each signal is travelling independently to approach the virtual antenna. All the factors such as: desynchronized UAV and ground sensor clock, Signal to Noise Ratio (SNR) and geometrical variation of antenna for each snapshot, caused an extra deviation in signal phase that will be received at the VPA antenna elements. In the newly developed system, the calibration phase is introduced during the first snapshot. It is used to estimate the phase difference F_{offset} between two consecutive samples which is caused by desynchronized clock of devices. Once the first snapshot has been made, we will consider similar style to gather for the whole data. Data of each snapshot will be fed to the Virtual Antenna Module (VAM) for further processing. VAM produces a steering vector where the response of each signal is stored in the form of a matrix and will be handed over to the Rectifier Module (RM). RM is introduced to rectify the whole data according to the F_{offset} measured in the calibration phase and to estimate the covariance matrix R_{AS} of the equation (4.24). This covariance matrix will be introduced to MUSIC algorithm. The output of the MUSIC algorithm (angle θ) will finally be tuned by the adjustment module. The proposed system model and detail description of each module are as follows:

4.3.1 Geometrical variation

The main difference between VPA antennas as compared to Physical Antenna Array (PAA) is that, in VPA all the snapshots are not taken at once. The position of antenna is continuously changing while taking data snapshots, which causes classical MUSIC algorithm directly inapplicable. First of all, this geometrical variation of antenna is needed to be considered. Let's suppose N sensor nodes $(n_1, n_2, \dots, n_N)$ are installed in a field and signal transmitted by each Node n_i is making angle $(\theta_1^{n_1}, \theta_2^{n_1}, \dots, \theta_M^{n_1})$ with M virtual antenna elements installed on an UAV (see Figure 4.4).

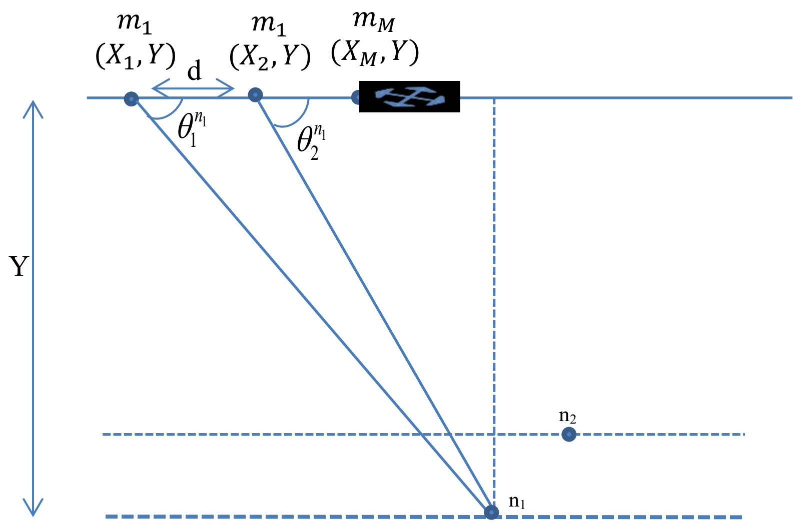


Figure 4.4: Geometrical variation of angles in VPA antenna.

For a single antenna formation UAV is considered as moving in a straight path. The X axis corresponds to the UAV trajectory, considered linear.

Let's suppose UAV want to locate a node n_i, where $1 \leq i \leq (N = M - 1)$, installed in the field and transmitting a narrowband signal. Y is ordinate of UAV that is same for all nodes and X_k is the abscissa of UAV at time t_k where $1 < k \leq M$. UAV is taking M samples for an array snapshot each after a fixed distance:

$$d = X_{k+1} - X_k = c_u \times \Delta t \tag{4.26}$$

where d is the length of the linear trajectory of the UAV between 2 snapshots and c_u is the speed of the UAV for a period of Δt (time difference between 2 successive snapshots). The angle of first antenna element X_1 with the sensor node n_i is:

$$\tan \theta_1^{n_1} = \frac{X_1}{Y} \tag{4.27}$$

If we consider first snapshot as reference, the second angle at second snapshot is as:

$$\tan \theta_2^{n_1} = \frac{X_1 - d}{Y} \tag{4.28}$$

In similar way and for linear flight of UAV, we can write:

$$\theta_k^{n_i} = \theta^{n_i}(t_k) = \tan^{-1}\left(\frac{X_k - C_u\, t_k}{Y}\right) \tag{4.29}$$

As we know, the separation between two antenna elements should be $d = \lambda/2$ and time interval required to cover this distance is $\Delta t = \frac{d}{c_u}$ (where c_u is speed of UAV), then we can conclude that UAV will make virtual array by taking snapshots at times with time interval $\Delta t = \frac{\lambda}{2c_u}$ second.

UAV having a single antenna is moving in a straight line at constant height and speed. It can sample the received signals at different times and construct a $M \times 1$ steering vector, or array manifold, for the node i of θ angle. We can construct observation matrix as:

$$A(\theta) = [a_1(t_1), a_2(t_2), \dots a_M(t_N)]^T \tag{4.30}$$

If t_i and observations $A(\theta)$ are known then virtual phase array can model the physical array of the antenna. As data sampling frequency is much higher than the frequency of snapshotting related to the speed of UAV, then we can assume that UAV is making uniform VPA antenna of M elements by moving with a constant speed of c_u for one antenna length. Array is referenced to the first element and steering vector model can be written as, where $i = 1,2, \dots, M$

$$A = \left[e^{j\beta d_i \sin(\theta_i)}\right] = \left[e^{j\beta c_u t_i \sin(\theta_i)}\right] = \left[1 \quad e^{j\beta\,(c_u \times t_1)\sin(\theta)} \;\cdots\; e^{j\beta(c_u \times t_N)\sin(\theta+\Delta\theta_M)}\right] \tag{4.31}$$

Where the inter-snapshot spacing $d = \lambda/2$, $\beta = \frac{2\pi}{\lambda}$, λ is the wavelength and θ is the AoA of the sensor, $t_1 = 0$ is the time when UAV starts taking snapshots and t_m is the time when it approaches the M^{th} snapshot. $\Delta\theta$ is the change of angle of incedent signal array on snapshot N due to movement of UAV.

One of the key benefit of our proposed system is that it can operate at different frequencies and can adapt inter element spacing and number of sampling as per requirement. Two examples of antenna formation with different frequencies are given below.

4.3.2 Example -1, long range localization

If a 433 MHz frequency is used for localization, it can cover long distance with very low power, see subsection 3.8.1. The specifications of virtual antenna are as under:

Operational frequency = $f_1 = 433\ MHz \qquad \lambda_1 = \frac{3\,.10^8}{4.33\,.\ 10^8} = 0.6928\ m \sim 0.7\ m$

Speed of UAV is = $c_u = 25\ km/h = 6.944\ m/s \sim 7\ m/s$

UAV will take a sample after every $\Delta t = \lambda 1/(2C_u) = 0.05\ s = 50\ ms$

As per [131] cross range resolution of an antenna is $\Delta\theta = \frac{\lambda 1}{L_{f1}}$ in radian. The antenna resolution $\Delta\theta$ in degree can be calculated as:

$$\Delta\theta = 57\frac{\lambda 1}{L_{f1}} \text{ in degrees}$$

Where, L_{f1} is length of antenna with frequency f_1

If we want an antenna resolution $\Delta\theta_{3dB} = 6^0$ (3dB beamwidth), then the antenna length for a long-range communication should be:

$$L_{f1} = 57\frac{\lambda 1}{6°} = 6.65\,m$$

Spacing between two virtual antenna elements is $\lambda/2 = 0.35\,m$

Number of antenna elements for a complete virtual array snapshot: $M = \left\lceil \frac{L_{f1}}{\lambda/2} \right\rceil = 20$

Time of observation for a complete array snapshot: $Tobs = L_{f1} / c_u = 6{,}65/7 \sim 0.95\,sec$

4.3.3 Example -2, short range localization

If considering 2.4 GHz for localization, then the specifications of virtual antenna will form as:

Operating frequency $f_2 = 2.4\,GHz$; $\lambda_2 = \frac{3\,x\,10^8}{24\,x\,10^8} = 0.125$ m

$\Delta t = \frac{\lambda_2}{2C_u} \sim 0.009\,s = 9\,ms$ (UAV will take a sample after every 9 ms)

If the resolution $\Delta\theta_{3dB} = 6^0$, the minimum required an antenna length $L_{f2} = 57\frac{\lambda_2}{6°} = 1.1875\,m$

Number of antenna elements for a complete virtual array snapshot should be $M = 20$.

4.3.4 Sampling

VPA will take M snapshots and within two snapshots m_k and m_{k+1}, it collects R_{over} samples of data. R_{over} depends on the sampling frequency of the Analog to Digital Convertor (ADC) device used. If the ADC sampling frequency is F_{ADC} Hz, then oversampling of the one virtual array element (number of samples between 2 snapshots of the virtual array) will be:

$$R_{over} = \lfloor F_{ADC} \times \Delta t \rfloor \tag{4.32}$$

where $\lfloor \iota \rfloor$ is the integer of ι and Δt is the time between two snapshots.

If a standard Voltage Controlled Crystal Oscillators (VCXO) is considered as an ADC convertor, it has an accuracy of:

$$10\,ppm < \text{Frequency accuracy} < 50\text{ ppm} \qquad \text{(ppm= part per million)}$$

If the communication radio frequency is F_{radio} = 433 MHz, then the accuracy gives a frequency offset of:

$$4.33\,KHz < \Delta F < 21.65\,KHz$$

By sampling period $\frac{1}{F_{ADC}}$ the frequency offset generates a rotation of phase as:

$$\Delta\psi_{offset} = \frac{\Delta F}{F_{ADC}}$$

$$0.02165 \times 2\pi < \left|\Delta\psi_{offset}\right| < 0.10825 \times 2\pi$$

So, the poorest accuracy gives $\frac{2\pi}{10}$ phase rotation between two consecutive samples. It means, in above mentioned conditions of the order of every consecutive data samples may have a $\frac{2\pi}{10}$ extra phase difference rather than usually expected. In physical antenna, as all the snapshots are taken at once, this phase difference is considered constant and the same for all, but in case of moving antenna samples are taken one by one independently and each sample is facing addition $\Delta\psi_{offset}$ separately, that needs to be corrected.

4.3.5 Calibration

Due to desynchronized clocks of UAV and sensor nodes, the precedent ΔF applies to each clock and the resultant inaccuracy (relative to a theoretical) clock frequency is the sum of both which can be of opposite directions. ADCs oversampling presents a phase difference between two consecutive data samples. The fact that the phase rotation between two consecutive samples $< \frac{2\pi}{10}$, allows the calibration to estimate this rotation to be compensated before array processing procedure

To fix our idea and without loose of generality, let us suppose the case with a singular source and let $s(k)$ denotes the kth sample of that source received by the antenna at k discrete time.

$$s(k) = e^{-j2\pi\Delta F_{total}\frac{k}{F_{ADC}}}$$

Where $k = 1, 2, \dots, R_{over}$ and ΔF_{total} is the frequency offset difference of the 2 clocks, UAV and node:

$$\Delta F_{total} = F_{UAV} - F_{node}$$

The angle rotation between 2 samples because of ΔF_{total} is:

$$\Delta\Phi = \Phi_k - \Phi_{k-1} = \frac{2\pi\,\Delta F_{total}\,k}{F_{ADC}} - \frac{2\pi\,\Delta F_{total}\,(k-1)}{F_{ADC}} \tag{4.33}$$

From equations (4.32) and (4.33), the frequency offset difference ΔF_{total} can be written as:

$$\Delta F_{total} \approx \frac{\Delta\Phi R_{over}}{2\pi\,\Delta t} \tag{4.34}$$

ΔF_{total} can be considered as constant during an observation time. For a unitary observation time of Δt (between 2 samples of VPA), we can make the following approximation:

$$\Delta F_{total} \approx \widetilde{\Delta \mathrm{F}}_{\mathrm{offset}} \cong \frac{\sum_{k=2}^{R_{over}}(\Phi_k - \Phi_{k-1})}{R_{over}-1} \tag{4.35}$$

4.3.6 Rectification

After the estimation of $\widetilde{\Delta \mathrm{F}}_{\mathrm{offset}}$, the rectification is introduced to compensate the phase offset estimated as:

$$S_r(k) = e^{-j2\pi\widetilde{\Delta F}_{offset}\frac{k}{F_{ADC}}} \times s(k) \tag{4.36}$$

4.3.7 Adjustment of incident angle

In our scenario, the incident θ is varying at each sample by the ratio of UAV speed and covered distance. By considering this characteristic, the final out put θ value needs to be adjusted as:

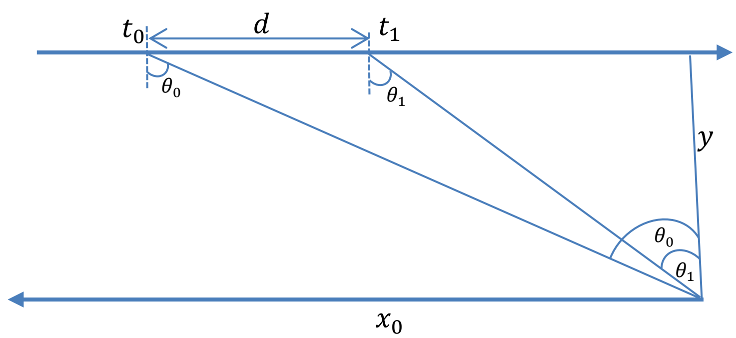


Figure 4.5: Incident angle

Let θ_i be the incident angle at the i[th] snapshot, x_i is the horizontal distance between the UAV and the node at the i[th] snapshot, y is the height of UAV. In this case, we can conclude that:

$$x_i = x_{i+1} + d$$

$$\theta_i = tan^{-1}\left(\frac{x_i}{y}\right)$$

Where d is the distance between 2 successive snapshot:

$$D = c_u \Delta t$$

Here c_u is the speed of the UAV and Δt is time between two snapshots. Using a first order approximation, we write the following equation:

$$\Delta\theta = \theta_i - \theta_{i+1} = \tan^{-1}\left(\frac{x_i}{y}\right) - \tan^{-1}\left(\frac{x_i + 1}{y}\right)$$

$$\approx \frac{x_i}{y} - \frac{x_i - d}{y} = \frac{c_u \Delta t}{y}$$

The total adjustment $\Delta\theta$ should be:

$$\Delta\theta_T = M\Delta\theta \tag{4.37}$$

4 .4 Virtual Antenna Array in Case of UAV Non-linear Movement

To construct a virtual antenna array, it is assumed that UAV is moving at constant speed and height, and all the snapshots are taken at equal intervals of time in a straight line, as shown in Figure 4.6. UAV is taking its first snapshot $m_1(X_1, Y_1)$ at point $\mathbb{A}$ and a second snapshot $m_2(X_2, Y_2)$ at point $\mathbb{B}$.

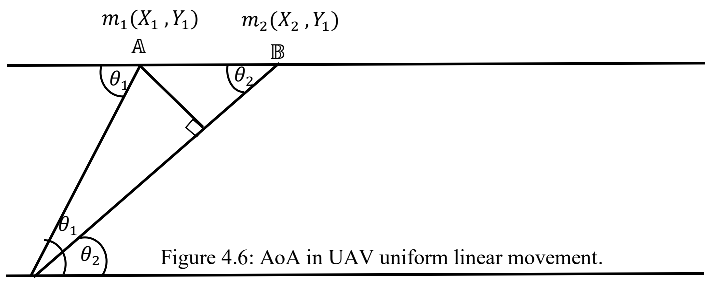


Figure 4.6: AoA in UAV uniform linear movement.

By considering air pressure or GPS inaccuracies, the resulting scenario is the one shown in Figure 4.7. UAV is deflected from a straight path, at point $\mathbb{B}$ by angle an α and it takes the second snapshot at a point $\mathbb{C}$ but the distance between the two points ($\mathbb{A}$ and $\mathbb{C}$) is still $\lambda/2$ because snapshots are taken after fix intervals of time.

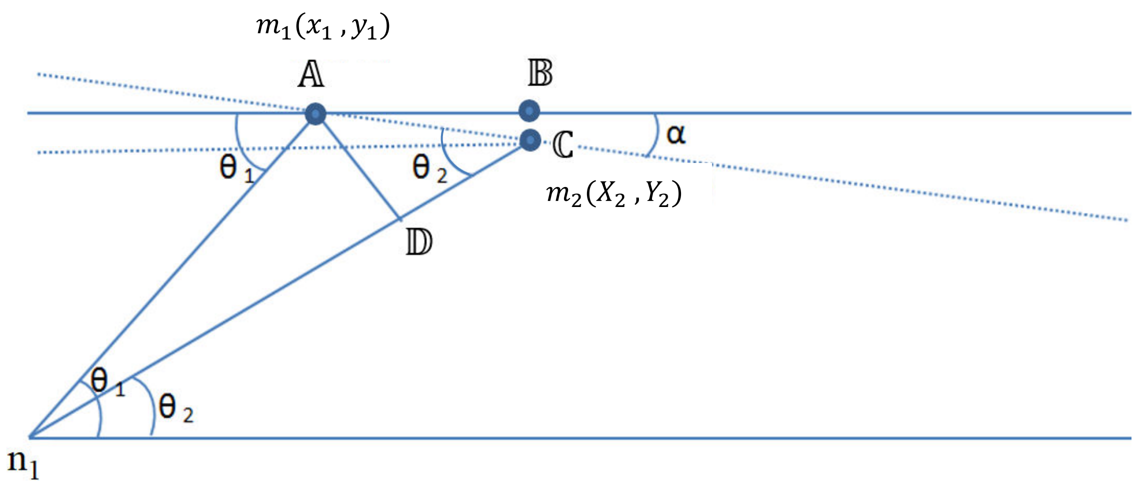


Figure 4.7: AoA in UAV non-linear movement.

Let $D_i = distance$ from n_1 to the point $m_i = \sqrt{x_i^2 + y_i^2}$, the transmission time required to cover the distance D_i is:

$$\tau_i = \frac{D_i}{c} \tag{4.38}$$

where c = speed of light. Let us consider a relative delay; i.e. the origin of time for the antenna is the time when the signal is received m_1.

If (x_1, y_1) is the position of the first point $\mathbb{A}$, (x_2, y_2) is the position of second point $\mathbb{C}$ which is deflected by angle α from straight path and the point $\mathbb{B}$ has (x, y). in this case we can prove that:

$$tan\ \alpha = \frac{\overline{\mathbb{BC}}}{\overline{\mathbb{AC}}} = \frac{y_1 - y_2}{x_2 - x_1}$$

$$tan\ \theta_i = \frac{y_i}{x_i} \tag{4.39}$$

The real delay between the two antennas is given by:

$$\tau = \frac{D_2 - D_1}{c} = \frac{\sqrt{{x_2}^2 + y^2} - \sqrt{{x_1}^2 + y^2}}{c} \tag{4.40}$$

As $d = \overline{\mathbb{AC}} = \overline{m_1 m_2}$, the distance between $\overline{\mathbb{BC}}$ is:

$$\overline{\mathbb{CD}} = d\ cos\ (\theta_2 + \alpha) \tag{4.41}$$

Where, $y_1 - y_2 = d\ sin\alpha$, $x_2 - x_1 = d\ cos\alpha$.

In this case, we can write:

$$\tau C = \sqrt{(x_1 + d\cos\alpha)^2 + (y_1 - d\sin\alpha)^2} - \sqrt{{x_1}^2 + {y_2}^2} \tag{4.42}$$

Let D be the distance between UAV and the sensor:

$$D = \sqrt{{x_1}^2 + {y_1}^2} \tag{4.43}$$

We can assume that $D >> d$

$$\tau C = \sqrt{D^2 + d^2 + 2d(x_1\cos\alpha - y_1\sin\alpha)} - D$$

$$\tau C \cong D\sqrt{1 + \left(\frac{d}{D}\right)^2 + \frac{2d}{D}\cos(\alpha + \theta)} - D$$

$$\cong D\sqrt{1 + \frac{2d}{D}\cos(\alpha + \theta)} - D$$

$$\cong D\left(1 + \frac{d}{D}\cos(\alpha + \theta)\right) - D$$

$$\cong d\cos(\alpha + \theta) \tag{4.44}$$

As the antenna length is small and the total time required to collect all of data samples is also small, in this case, α is considered as negligible.

4.5 Simulation Model and Analysis

Simulation of the proposed system was conducted in MATLAB. A WiFi single antenna operating at 2.4 GHz is mounted over an UAV that is moving at the constant speed of *20* m/s. Three targets (sensor nodes) are placed on ground level where the different azimuth angles are respectively 20^0 and -60^0 and the elevation angle is 0 for all. All the targets are transmitting narrowband signals $a(\theta)$ of frequency 4.3 MHz periodically from a wave field which incident on the VPA. A VPA antenna installed over UAV is used to stare and locate the targets that are placed on the ground. For the simulation, the following input parameters are taken:

```
F       = 4.3 x 10^8      %Hz radio frequency for transmission
F_ADC   = 2 x 10^5        %Hz analog to digital converter frequency
F_offset = 2 x 10^4   %Hz frequency offset between transmitter and receiver
V       =7            % speed of UAV
dt      =λ /(2*v)     % time difference between 2 snapshots (2 virtual
antennas)
R_over   = F_ADC*dt   %oversampling of the virtual array= number of samples
                      %between 2 snapshots of the virtual array
M       =20;          %Number of snapshots for 6 degree accuracy
y       = 100;         % Height of UAV
dt_over =1/fADC;      %time between 2 ADC samples.
```

Some simulation results are shown below. As shown in Figure 4.8, the AoA of two target nodes making angles of 20^0 and -60^0 with the UAV, is estimated. Direction of both the targets is measured with and without our proposed system. Black curve shows the estimation by MUSIC without our modification and found not working well. Red curve is showing the working of MUSIC after adding our developed modules including; calibration and rectification. It is observed that estimation gives accurate results.

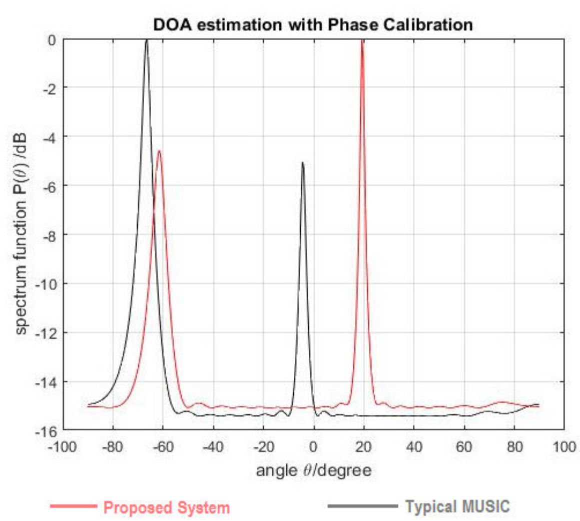


Figure 4.8: Proposed system VS classical MUSIC.

Effect of AoA estimation accuracy with increasing SNR is evaluated in Figure 4.9. It is found that the proposed system is working well in low value of SNR up to 8 dB.

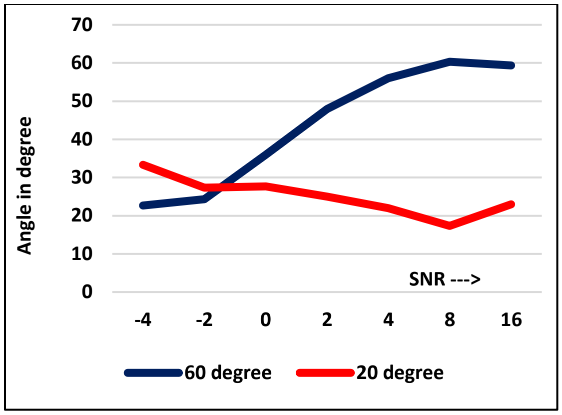


Figure 4.9: Effect of increasing SNR on proposed AoA estimation system.

Original received signal before rectification is shown in Figure 4.10 and the phase offset observed in each signal is shown in Figure 4.11. Since the rotation of phases cause major deviations in AoA estimation, for that reason the received signal needs special rectification to deal with this issue.

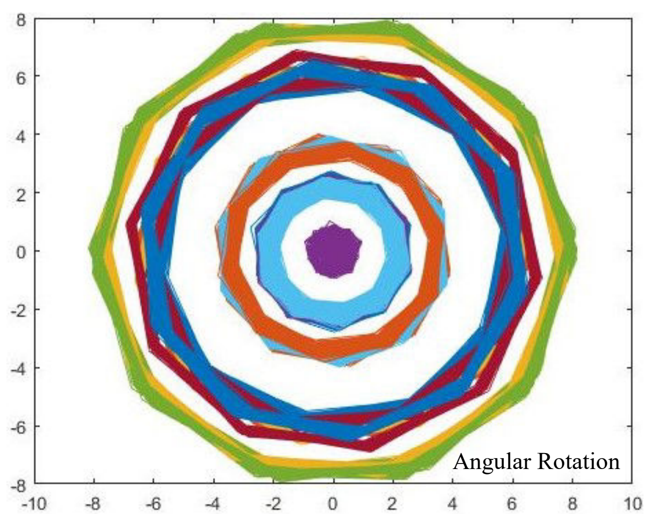


Figure 4.10: Original received signal on VPA without rectification.

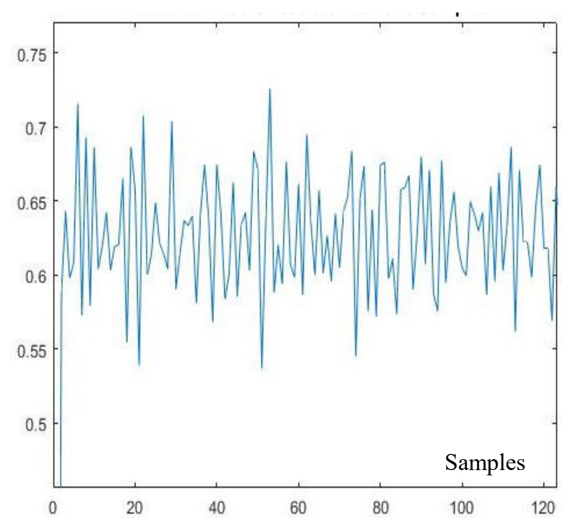


Figure 4.11: Phase offset observed.

Figure 4.12 represents the received signals after applying the proposed rectification. This figure shows that the problem of the rotation of phases due to ADC oversampling is fixed by applying our proposed rectification module. The proposed system is also found working well in the condition when AoA angle is transforming from positive to negative value, as shown in Figure 4.13. In this figure, it is shown that both the angles are measured accurately, especially the angle 0 that actually varies from +2 to -2^0.

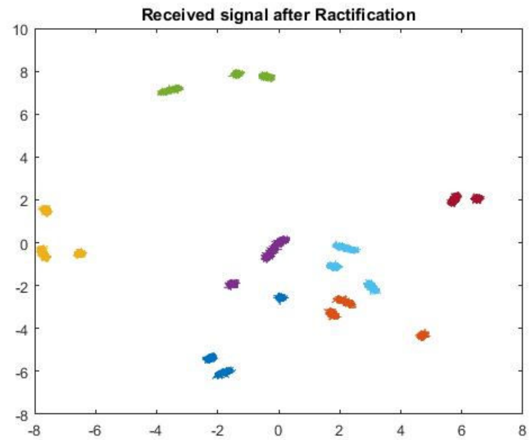


Figure 4.12: Rectified received signal.

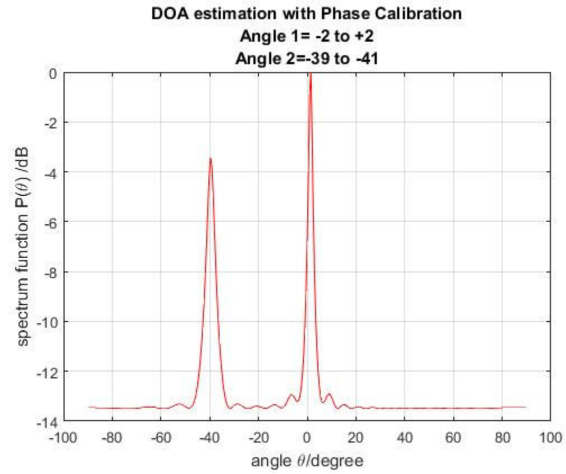


Figure 4.13: AoA estimation when angle varies from positive to negative value.

All the simulation results shown above prove that if a UAV is moving at a constant speed, the single antenna can act as ULA of multiple antenna elements. As shown in figure 4.13, both the angles are measured accurately and the performance of the virtual array antenna system is found to be outclassed. Classical MUSIC algorithms are not practical in case of a single antenna (virtually making antenna array) and gives a very poor performance, if it is used without the suggested modifications. The proposed system will make it possible to use a single antenna as a virtual phase array of *M* antennas, which will add more benefits such as the ease of use, cost effectiveness, light-weightiness, energy efficiency, flexibility, and adaptability.

4.6 Conclusion

In this chapter, a virtual phase array (VPA) antenna system for AoA estimation of narrowband signal is presented. It has been mathematically proven that a single antenna installed on a moving UAV can act as a virtual linear array of multiple antennas and can completely replace the physical array of antennas. The benefit is that, it is a light weight single antenna system that can be carried by an UAV very easily to estimate AoA of the signals received from ground sensors. VPA will also provide adaptive steering capabilities which will help to increase the performance and throughput of the system. By using the proposed system, an UAV can easily estimate the location of sensors installed on the ground and can communicate with them more efficiently and precisely. The proposed system is evaluated by using simulation model and found working out class. It has the potential to work better than a conventional ULA antenna.

5

Hardware Development

5.1 Introduction

A proof of concept is built to test the proposed system in the real environment. All components are designed and prepared by using local resources and expertise. We faced many challenges to achieve the desire functionality. We developed three main modules UAV, sink device and sensor node. The development of sink and sensor nodes are already explained in chapter 3. We have assembled and customized our own UAV to carry the designed data gathering devices, and to operate according to developed routing protocol. Stepwise tutorial about UAV making is given in the upcoming section.

5.2 UAV

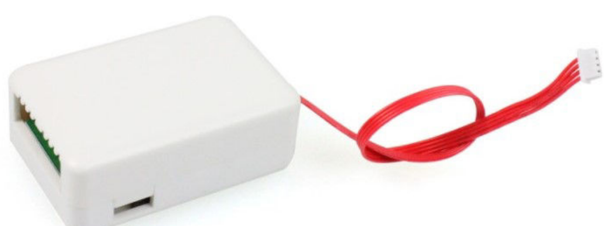
UAV is the main and major component of this project. Although, variety of such machines are available in market at affordable price but choice was a bit difficult task. Our humble gratitude to SNCS and KACST for funding this project. Initially, SNCS and University of Tabuk arranged a very good machine DJI Phantom 4 [71] for said purpose (see Figure 5.1).

Figure 5.1: DJI Phantom 4.

It is a very good, much stable and accurate machine. However, phantom is difficult to integrate with Arduino and other similar or compatible components. Besides, DJI is not an open source and is hard to be customized and controlled as per functionality of our developed protocol. The alternative of DJI is Cheerson CX-20 [132] shown in Figure 5.2.

Figure 5.2: Cheerson CX-20.

Cheerson CX-20 has an open source flight controller. It can be controlled by open source mission planner available for windows, android, IOS and other operating systems. While attaching our developed system on this machine, some problems restrict us to continue further with it which included: its small size flight controller has a limited number of interfaces and restricts the number of connected devices and modules (see Figure 5.3-A). It also suffers from an inaccurate GPS module (Figure 5.3-B). It is difficult to control as not equipped with an altitude hold sensor and also don't have support to upgrade. Furthermore, it does not have the provision to add collision avoidance system that we may need in future.



(A) CX-20 APM V2.1 Flight Controller.

(B) CX-20 GPS module.

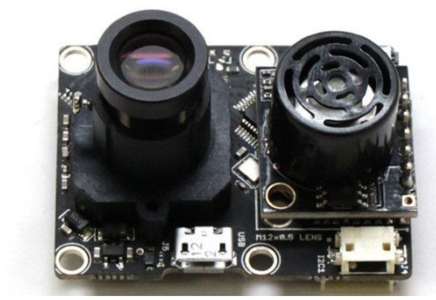
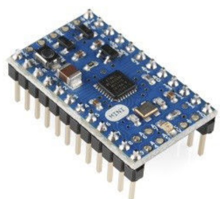
Figure 5.3: CX-20 APM flight controller and GPS module.

One of the solutions for these limitations was the up gradation of CX-20 with latest APM 2.6 flight controller, specialized GPS module, altitude hold sensor, etc.

Finally, we decided to build our own customized UAV to overcome shipping delays, custom clearance and up gradation hassles, furthermore to achieve economy. As known, to construct an UAV, the most important parameter is the expected weight carried by UAV and its components should be selected/purchased accordingly. The expected weight of required UAV is estimated around 1.3 Kg, details are listed in Table 5-1. All these components are available at hobbyking.com, Amazon.com, ebay.com, Aliexpress.com, etc.

TABLE 5-1: UAV COMPONENTS

Name	Pictures	Weight (Gram)	Specifications
Fiberglass frame		405	Height =18 cm Motor to moter distence = 35 cm
4x Motors and ESC		200	**Motors:** Speed = 920 rpm/v Max Power = 370W Max thrust =1.2 Kg **ESC:** 30A **Propellers** :1045
Flight controller		20	APM 2.6
R/C Receiver		15	2.4 GHZ 6 channels (Throttle, Aileron, Elevator, Rudder, Gear, Aux)
Battery		150	1.8 Ah

Px-4 Altitude hold sensor optical flow + Ultrasonic module		10	168 MHz Cortex M4F CPU (128 + 64 KB RAM) 752×480 MT9V034 image sensor, L3GD20 3D Gyro 16 mm M12 lens (IR block filter) Onboard 16bit gyroscope Onboard sonar (to estimate the ground distance)
Arduino mini (to attach routing protocol)		20	9 v Input 3, 5 v Outputs 4 Analog pins 9 Digital pins Tx and Rx interface
Telemetry		20	Telemetry Kit 433Mhz Receiver sensitivity -121 dBm Transmit power up to 20dBm (100mW) Air data rates up to 250kbps
Localization antenna and receiver		20	Will add later on for localization
Optional and future hardware			
FPV Camera and Transmitter		250	Camera=GoPro Hero 4 Still picture 12 Mpix Video = HD 1080p120 2 Axis Brushless Gimbal
Collision avoidance		210	In Future
	Total weight	**1.3 Kg**	

5.2.1 UAV thrust calculation

The downward thrust of motor that is used to lift up the UAV depends upon many factors such as: Battery power applied, revolutions per minute (rpm) of motor, length/width/weight of propellers and air density. First of all, we need to calculate the power W in watt applied on propellers (UAV fan blades) [133]

$$W = B_c \times (R_{pm})^{B_f} \tag{5.1}$$

Where, B_c and B_f are blade constant and blade factor respectively and R_{pm} is a thousand revolutions per minute of the motor. Values of both B_c and B_f can be taken from aircraft datasheet shown in Table 5-2.

TABLE 5-2: AIRCRAFT DATASHEET [133]

Prop Size	Prop Constants	Power Factor
6x4	0.015	3.20
7x5	0.042	3.20
8x4	0.060	3.20
8x6	0.106	3.20
8x8	0.148	3.20
9x4.5	0.090	3.20
9x6	0.129	3.20
9x7.5	0.352	2.90
9x9	0.448	2.90
10x5	0.144	3.20
10x7	0.223	3.20
10x10	0.680	2.90
11x5.5	0.222	3.20
11x7	0.301	3.20
11x8	0.357	3.20
11x8.5	0.383	3.20
11x10	0.589	3.20
12x6	0.322	3.20
12x10	0.706	3.20
12x12	1.528	2.90
13x4	0.299	3.20
13x6.5	0.516	3.20
13x8	0.590	3.20
13x10	0.825	3.20
14x7	0.715	3.20
14x10	1.118	3.20

The mass of air m which propeller blades throws in opposite direction to get acceleration can be calculated as per equation given in [133].

$$m = \frac{\left(\frac{\pi}{2} B_l{}^2 D_{air} W^2\right)^{1/3}}{g} \tag{5.2}$$

Where, B_l is propeller diameter (meters), D_{air} is air density (Kg/m3), g is gravitational force (m/s^2) and W is power (Watt).

As mentioned in Table 5-1, we are using 1045 propellers. According the aircraft datasheet the values of Bc and B_f are 0.144 and 3.2 respectively. The motor specification is 920 rpm/v motor [134]. We can make a table to show the relationship of applied voltage and performance of UAV.

Propeller type =1045

Propeller constant = 0.144

Propeller factor =3.2

TABLE 5-3: UAV THRUST AND VOLTAGE RELATION SHIP

Volts	rpm 920 × Volts	W Equation (5.1)	M Equation (5.2)
6	5520	34.1	0.5
7	6440	55.8	0.7
8	7360	85.6	1.0
9	8280	124.8	1.2

If motor is spinning with 9 v, it can produce 1.2 × 4 = 4.8 Kg of thrust by using all four throttles. If we set half of the thrust for carrying weight and rest half for navigation then it can hold 2.4 kg of weight easily. As per specifications, DJI Phantom-4 can carry (850 × 4) / 2 = 1.7 Kg while CX-20 can only carry about 0.64 Kg.

5.2.2 UAV development

The steps wise UAV assembling procedure is provided below:

1. The most important component is the motor which should be obtained as per required specifications including size of UAV, carrying weight, required thrust to hold the payload and resist the environment, etc. To make a classical quadcopter UAV, we need two pairs of motors, one pair rotating clock wise (CW) and another rotating Counter Clock Wise (CW). As a first step, we arranged all the required components to make a quadcopter as per above mentioned specifications.
2. To install motors, we should identify the spinning direction of motors. Same motors should be installed on opposite UAV wing as per Figure 5.4.

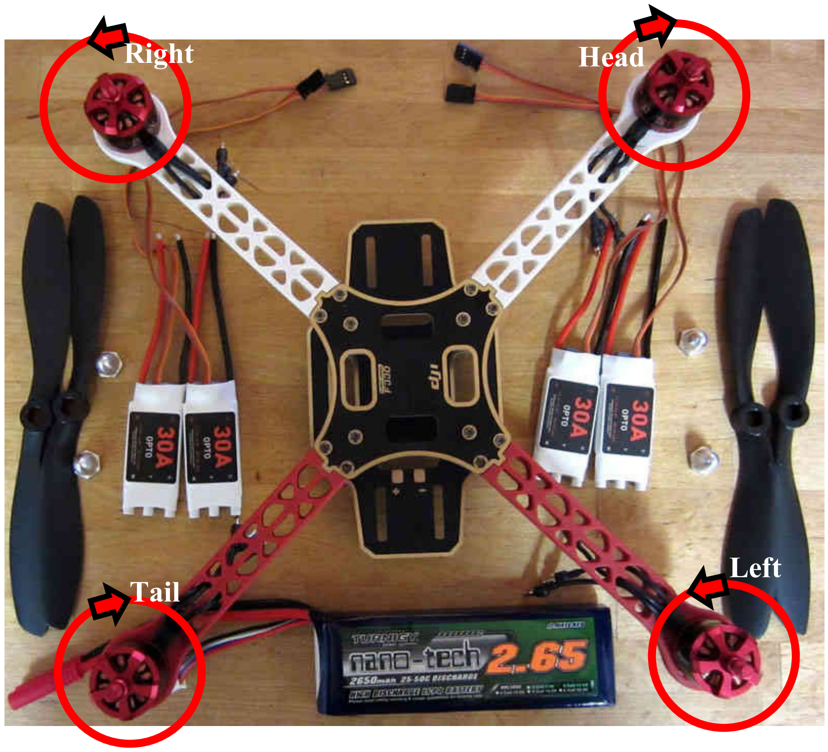


Figure 5.4: UAV assembly.

3. In the same way, two propeller blades should be of clockwise type remaining two anticlockwise. Should be installed same as motors.
4. There is no special clockwise and anti-clockwise ESCs (Electronic Speed Control). Two ESCs should be installed in head up direction, on head and tail wings and remaining two in the opposite way.
5. ESC should be connected with APM Autopilot in such a way: Right wing ESC at APM output pin pair 1, Left wing ESC at 2, head wing ESC at 3 and tail at 4 (see Figure 5.5).

Figure 5.5: Arduino output pins for ESCs

6. GPS is needed to be connected on APM's GPS port and 12C port. When fixing GPS on the frame it should be noticed that GPS arrow is pointed towards the front (head).
7. Remote control receiver pins are labeled as A, E, T, R, 1, 2, 3. All these pins should be connected with APM input pins 1, 2, 3, 4, 5, 6, 7 respectively. The receiver's pins labels stand for Aileron, Elevator, Throttle, Rudder, Gear and Aux.
8. Power distribution is simple. One battery cable will go to each ESC and one will plug in APM power port through APM Power Module 5.3V DC BEC XT60.
9. Many other UAV optional components can be installed as per requirements like telemetry, camera, altitude sensor, collision avoidance sensors, radar, etc.

5.2.3 Firmware installation

We used Mission Planner[19] application to install firmware, software, calibration, trouble shooting and planning waypoints. Many types of firmware are available in mission planner to build different kinds of UAVs, here we selected a quadcopter (see Figure 5.6).

[19] Mission planner is a free, open-source, community-supported application developed by Michael OBORNE of Western Australia Country Health Service, for the open-source APM autopilot project [188].

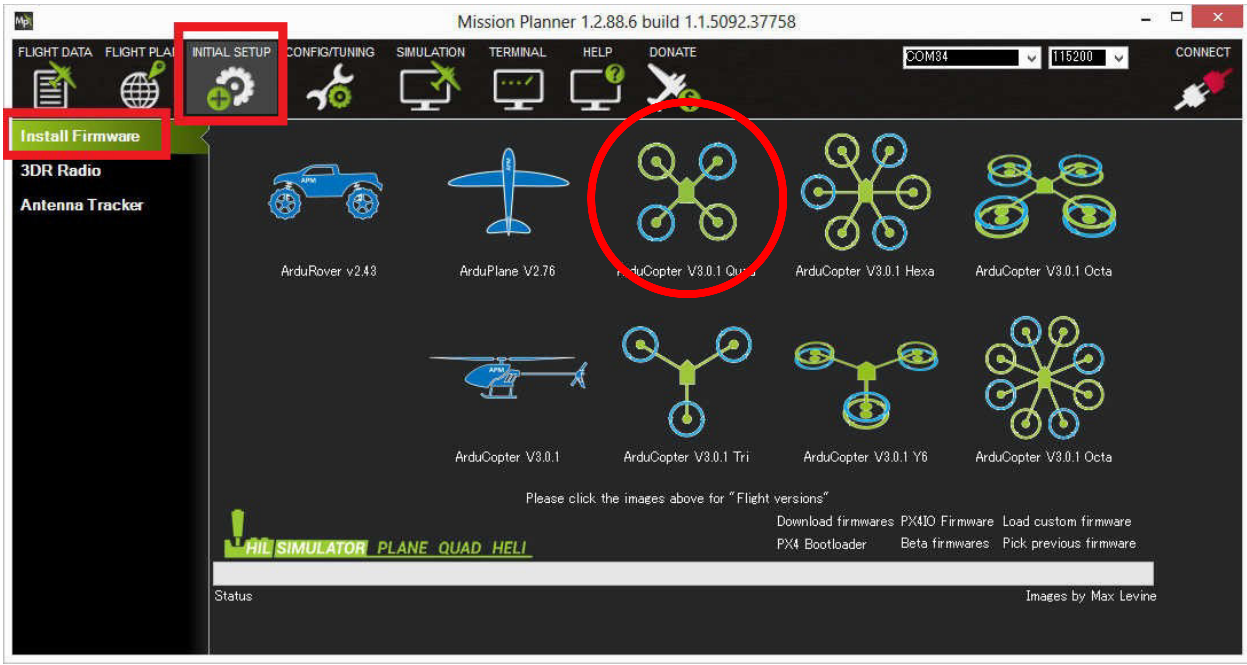


Figure 5.6: UAV firmware installation.

5.2.4 Accelerometer calibration

Once UAV is ready with installed firmware then next step is the calibration of different UAV components such as: GPS, Transmitter / Receiver, Motors etc. The first step is accelerometer calibration (see Figure 5.7). To complete calibration process, onboard APM flight controller should be connected with computer using USB cable[20].

[20] It is better to remove battery before plugin the USB cable because USB itself has power, doubling the power source may harm the APM

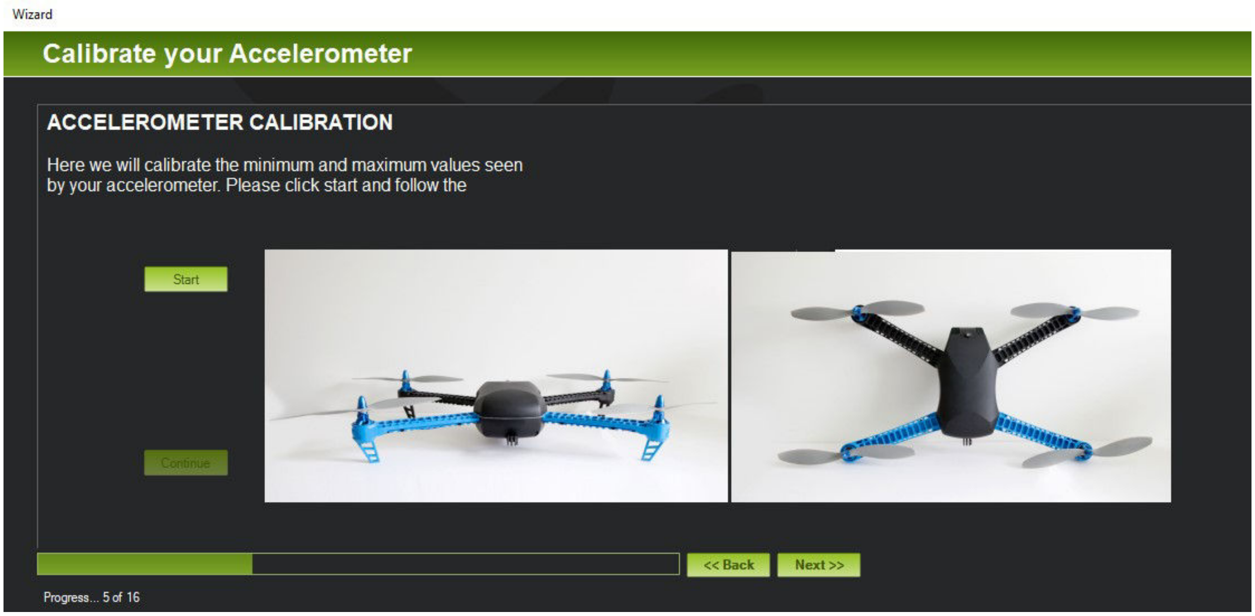


Figure 5.7: Mission planner accelerometer calibration

In this calibration, UAV accelerometer is calibrated by placing UAV in 6 different positions (Head up, Head down, right up, right down, left-right up and left-right down)

5.2.5 Compass calibration

Live calibration is next conducted to calibrate the compass/ gyroscope. A snapshot of live calibration is shown in Figure 5.8.

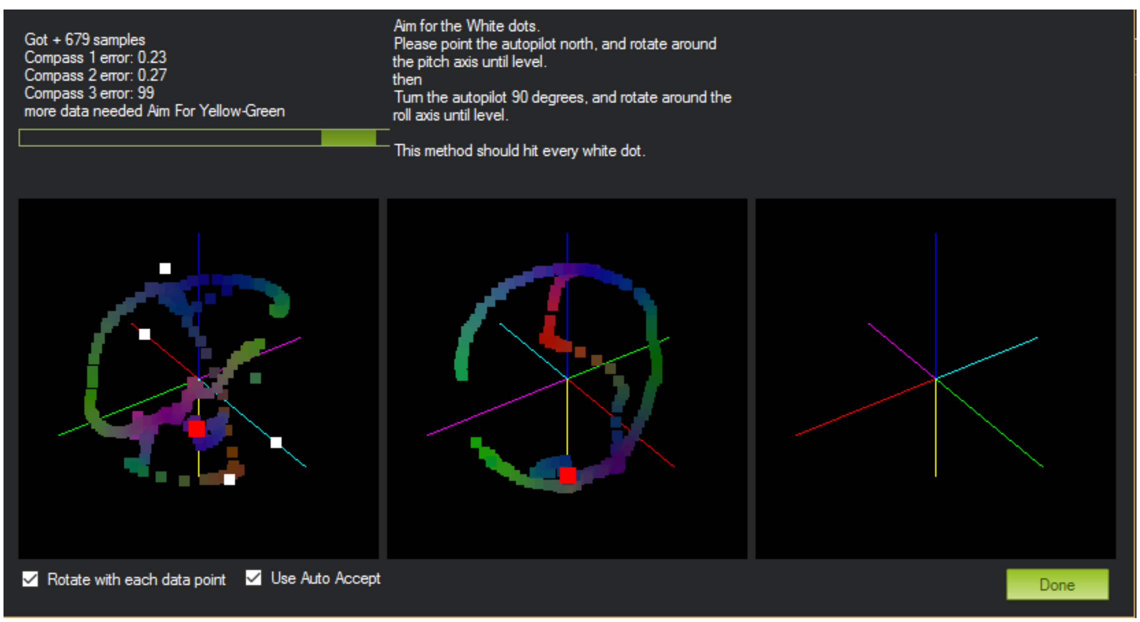


Figure 5.8: Compass calibration.

Calibration windows show three views of 3D excess. In first view (left to right), some white squares are visible. One should rotate the UAV in 3D space in circular form to catch these boxes. In second view, the observed and saved points are shown and third view shows the rotation of excess as UAV rotates. In top panel, progress bar and error messages are shown. To complete this process, UAV needs to rotate in 360 degrees in all directions like head up, head down, left up, right up, etc.
UAV remote control transmitter and receiver also need calibration. All the control sticks on transmitter need to move in all positions, while radio calibration is switched on in the mission planner, APM is connected with computer and transmitter itself is power on (see Figure 5.9).

Figure 5.9: Transmitter calibration.

Each bar shown in Figure 5.9, represents one joystick/switch on the remote control (Transmitter). Green color represents the current state of responding control. Calibration is done by moving all transmitter joysticks and switches in all directions one by one. High and low point references corresponding to each switch will be set as shown in red lines in figure.

5.2.6 ESC calibration

Next step is ESC calibration. ESC can be calibrated as follows:

1) Push ESC calibration button in the mission planner,
2) Disconnect USB cable,
3) Connect Battery with UAV,
4) Let the ESC complete beeping,
5) Restart APM manually by pressing restart button on it.

5.2.7 Flight mode setting

Flight modes can be declared in this section. Transmitter mostly has two switches SW1 and SW2 on it for said purpose. UAV flight modes can be seen in Figure 5.10 and should be set as:

00 = flight mode 1 = stable

01 = flight mode 2 = altitude hold

10 = flight mode 3 = auto flight

11 = flight mode 4 = return home

20 = flight mode 5 =

By using these settings, we can control and change the flight mode of UAV during mission. For example, by pressing 11, UAV will leave the mission and proceeds to return back.

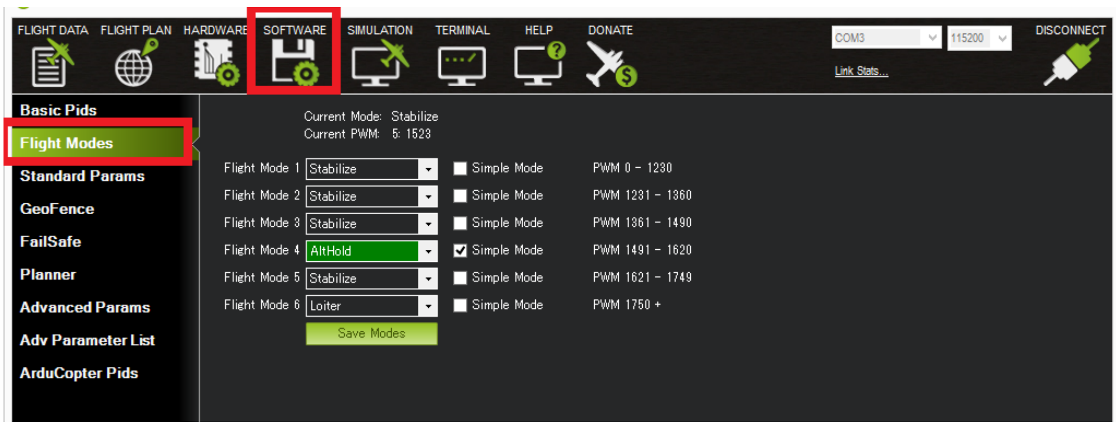


Figure 5.10: Flight mode settings

5.2.8 Mission planning

Mission can also be planned in mission planner. As shown in Figure 5.11, a mission may compose of waypoint coordinates, height at each waypoint and hovering time.

Figure 5.11: UAV mission.

In Figure 5.11 the dark green circles of different sizes are Tabuk farm field areas for mission. Small light green are the waypoints connected by a yellow solid line. Thin sold yellow line between node 1, 2 and 5 represents tentative path as the starting point is not yet finalized. It will lock point 1 as soon as we put it on ground to go for mission. Other thick lines are representing sure path.

Once the mission is designed and verified in the mission planner, it is uploaded in UAV APM module memory. To start mission, UAV should be placed on a relatively flat surface on return home position and press 10 buttons as mentioned in section 5.2.7. Our designed UAV can complete its mission automatically and scan the whole area as per its loaded mission information. Mission should be designed carefully so that UAV can complete it by using battery onboard. In case, UAV remaining power level approaches a threshold level[21] , it will interrupt all pending tasks and trigger return home function to come back.

5.2.9 Trouble shooting

The problems faced when flying this UAV on test basis are as under:

Field calibration: It's always good practice to calibrate UAV compass once before mission started otherwise mission can be failed or UAV may crash because of any compass malfunction happened due to

[21] Threshold depends on its distance from home location and remaining battery level.

previously hard landing or any magnetic interference. In-field, calibration can be conducted by using either method given in section 5.24 or press the right joystick of transmitter to the right most corner for a while and UAV will enter in calibration mode, now rotate the UAV in all directions until hear a sound in each direction.

Motor calibration: If four motors are not spinning with the same speed or any one is giving beeping sound, it means that it needs to be calibrated again.

Transmitter calibration: If transmitter is unable to arm the UAV, then it should be calibrate. Transmitter can be calibrated even without connecting the UAV with computer, just hold the SWB 1 switch and switch on the transmitter, it will go in calibration mode. Spin all the control sticks in the extreme position and turn off the transmitter. When it turns on again, it will already be calibrated.

5.3 UAV Connection with Developed Routing Protocol

Constructed UAV will be connected with the developed routing protocol, using designed scheme shown in Figure 5.12:

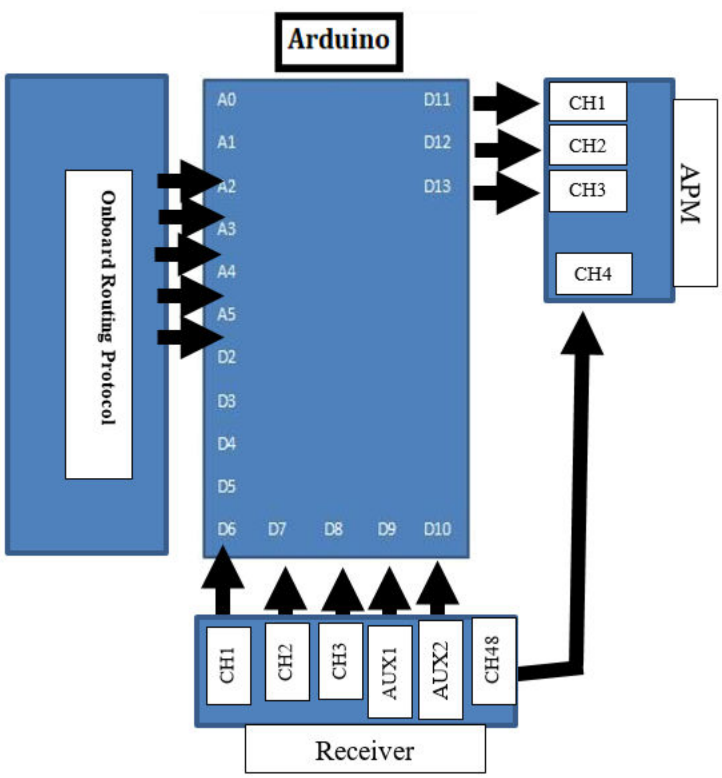


Figure 5.12: Complete sketch of proposed UAV.

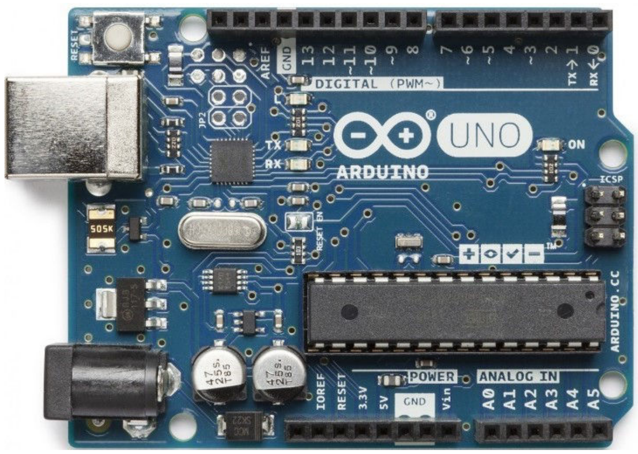


Figure 5.13: Arduino UNO micro controller.

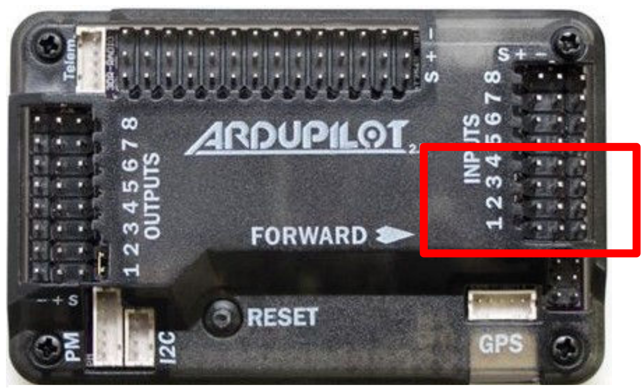


Figure 5.15: APM 2.6 autopilot.

Figure 5.14: Radio receiver for UAV.

UAV is designed such that onboard routing protocol can control it to scan given area and in the mean while it can also be controlled by remote control transmitter, if need to change the mission during execution (see Figure 5.12). We introduced Arduino (Figure 5.13) in between UAV receiver (Figure 5.14) and APM autopilot (5.15). This Arduino has also a routing protocol and related components on it. The Arduino takes all the inputs from remote-control transmitter and ground sensor nodes then drives the UAV as per developed routing protocol.

5.4 UAV Flight Data Monitoring and Feed Back

To monitor the flight data from the ground station, we use 3DR Radio Telemetry Kit 433MHZ (see Figure 5.16). This module has receiver sensitivity to -121 dBm, a transmission power up to 20 dBm (100mW) and can transmit 250 kbps up to 5 Km. This radio telemetry comes in pair, one is required to installed onboard APM autopilot and the other can be installed on the controlling computer as shown in Figure 5.17.

Figure 5.16: UHF long range telemetry kit.

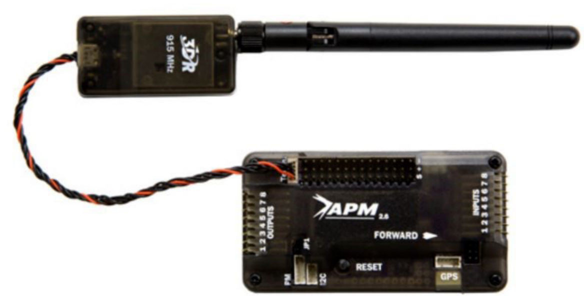


Figure 5.17: Telemetry installation guide.

5.5 Conclusion

In this chapter, we have given the details to develop a quadcopter UAV as per requirements. The most important factor to develop a UAV is the thrust calculation which helps us to estimate the specifications of the components and the performance of machine when completed. In this chapter, we also derived equation to compute this thrust value before starting to make UAV. At the end of this chapter, we have given a scheme to control the developed UAV by using an Arduino UNO microcontroller. This Arduino board takes control of UAV to drive it as per given protocol.

6

Conclusion & Future Work

6.1 Conclusion

Saudi Arabia is a very large country, characterized by a rich and fertile soil. It has no permanent rivers or lakes, and the rain is limited to few bursts per year. The ever-increasing demand for water to satisfy the population of a typical fast-developing country is a real challenge. Saudi Arabia recognizes and endeavors to meet the challenges with both continuous informed planning and development, and efficient use of water. With the focus on agriculture, Wireless Sensor Networks (WSNs) are used to ensure that crops are provided with the needed amount of water that allows them to grow and yield the intended quality and quantity. Care must be taken to minimize or eliminate an excess of water as it is both harmful to crops and translates into financial loss for farmers. Since, farming lands in KSA are mainly circular or rectangular parcels, then data collection from the WSNs planted on the parcels can be challenging as fixed communication infrastructures are either nonexistent or costly and unjustifiable. WSNs and UAVs become very good solutions to farming in KSA.

Our focus of in study is harnessing IoT [22] and UAVs in Saudi agriculture to produce quality and quantity of crops with optimum resource utilization. The challenge ahead was to develop an IoT system which can resist harsh weather, transfer data while there is no/very limited infrastructure, integrate heterogeneous agriculture sensors, locate low-cost sensors and finally collect need base data collections. To coup with all mentioned constraints, we developed a UAV Routing Protocol (URP) for data collection.

Formation of dynamic clusters of heterogeneous sensor nodes and Bayesian based cluster head selection are the key features of this system. One of the beauty of this scheme related to the fact that, UAV can move independently to scan a desire area and can harvest only selected IoT nodes. Another goodness of

[22] Internet of things (IoT) is the network connectivity of sensor nodes, actuators, embedded systems, electronics and software installed in daily life application (vehicles, agriculture, smart cities, home appliance, security, etc.) which enable these objects to collect and exchange data.

this protocol is that it preserves the maximum node energy by shifting most of the working load on UAV, using very low power frequency for localization and collection of data at reasonable height. UAV is doing many tasks throughout the data gathering process including beaconing, shunting of connected nodes, localization of cluster head candidates and data collection. Furthermore, it also plays a vital role in cluster formation and cluster head selection. Localization of field sensors by UAV is another very important part of this developed system. We designed and developed a light weight energy efficient virtual phase array antenna system to locate field sensors. The proposed system is evaluated in sequences of simulation models, developed in OMNeT++, MATLAB and STK. The results taken from different simulation cases reflects our success to preserve the maximum node energy on the expense of UAV energy and working load. The overall theme of this research is to monitor the crop related parameters, process collected data on a central point and takes appropriate action to save the resources (mainly water). Sensors coupled with actuators as a closed loop system can reduce or even eliminate water waste. UAVs address the tedious communication and data collection problem.

6.1.1 Concluding remarks

Our project designed a complete system starting from sensor nodes activation, gone through cluster formation, localization of sensor nodes, cluster head selection and finally ends with data collection. It has many characteristics such as:

1. No pre- instead infrastructure is required
2. Fast deployed and handy in use
3. Instant network formation, no special cluster head nodes, no predefined routes, no periodic updates
4. Selects only required nodes
5. Scans only selected area
6. Clusters are formed as per UAV trajectory
7. The best node is selected as CH, by evaluating all its parameters using Bayesian formula
8. Sensor nodes are located using developed light weight energy efficient virtual antenna
9. Developed antenna can switch operating frequency, steering precision and localization capacity during mission
10. Very low power UHF frequency is used for localization
11. Data is collected from CH at low height to save its energy

6.2 Future Work

This developed system is span in many directions and each dimension need more in-depth analysis and advancement that we have left for future work because of time restrictions. A paradigm of each dimension in the prospective of future needs and extensions is further provided here.

6.2.1 Clustering

In this thesis, we have developed a dynamic clustering scheme, highlights are as:

a. All sensor nodes will be location unaware and cheap in cost. These heterogeneous nodes will make a cluster as per UAV's request. Cluster will be formed in the way of UAV. One node will be considered as the best node (having the highest Bayesian value) and selected as CH.
b. In this developed system, UAV is taking the liberty to move freely in the field to harvest data but paying more in terms of energy and hard work to locate the sensor nodes evaluate them and nominate one of them as CH.
c. UAV collects data from a CH at reasonable height suitable for CH. It will result in saving nodes energy prominently.

The developed clustering scheme can be extended and modified in many ways as:

1. So far, clustering is restricted up to single hop and direct neighbors are only taking part in a cluster formation that can be extended up to n-hop and then multi-hop.
2. In our developed system, a single UAV is used to harvest data from a desire area. In future, Swarm of UAVs can be utilized in a self-coordinated way to share the load and harvest data. It will help us to improve the throughput.
3. In the developed system, all nodes except UAV (data mole) are considered as static. In future, an extra layer for mobile agents can be added between UAV and field nodes. These mobile agents will help to expedite the data collection process and optimize the node energy utilization. In this case, UAV will strictly follow its path, height and speed. These chargeable mobile agents will collect data for CH and deliver it to UAV. The agriculture machines performing daily routine work can also be used as mobile data collection centers.

6.2.2 Localization

We have designed a virtual antenna system for sensor node localization, where movement of UAV is exploited to take a snapshot of data after fixed interval of time to replicate the single antenna at different places. This phenomenon provided us the ground to initiate a virtual phase array antenna. All the work in this dimension is just a beginning and needs a lot of improvement and accuracy in future. Some suggestions are given below:

1. In this thesis, a single UAV is used to make a virtual linear array. In future multiple UAVs flying in a swarm can make a 3D virtual antenna to provide more accuracy and more identifiable targets.
2. Multiple UAVs making virtual antenna and trigonometry with each and other, can use the both effects to estimate the location of dispersed sensor nodes and then scan them in coordinated way. This concept is also needed to be developed.
3. UAV can also make other formations like virtual circular array and virtual rectangular array. In future, all these combinations can be evaluated to test which one is more feasible and provides more accuracy.

6.2.3 Mission optimization

In this research, the path of UAV is taken as fixed and given to the UAV in advance before starting of the mission. Some suggestions for future advancement are:

1. Path of UAV can be considered as adaptable and self-managed. For example, if UAV analyzes some patterns of disaster or disease spreading in the area, it can adapt its path to track this pattern or can track any kind of abnormality.
2. If swarm of UAVs is taken in consideration in future, then collision avoidance in its mission is also needed to be taken in account.
3. Obstacle avoidance in the mission should also be included in top layer of the proposed system, that can be a next version of the developed system.

6.2.4 Hardware development

1. The hardware for virtual antenna needs to be designed and developed as future work of this study.
2. The proof of concept devices developed in this thesis need to build as professional version.
3. UAV needs to equip with more peripherals including, collision avoidance and virtual antenna system.

Appendix A

Simulation Algorithms

In this appendix, different data gathering algorithms related to our developed UAV based Routing Protocol (URP) are discussed. We simulate these algorithms in two different simulation soft wares (OMNeT ++ and MATLAB) to evaluate their performances. In this appendix, we are providing the pseudocodes of these algorithms.

A.1 LEACH

Leach algorithm has described in section 3.2, where all the static nodes transmit their data to a fixed base station by making a cluster formation. Cluster head is selected on random bases in each round to preserve the node energy. Threshold calculation, energy consumption and other details are already provided in above mentioned section. The LEACH algorithm that we developed in our system is given below.

BEGIN

$Specify\ the\ probability\ (N_n), number\ of\ nodes\ (n)$

$E_{init}(s) = E_0, \quad \{s = 1,2, \dots, n\}$

PREPARATION PHASE

$if\ (E_{init}(s) > 0\ \&\ r\ mod\ \lfloor n/N_c \rfloor \neq 0)\ then\ //pset\ can\ set \geq 0.5$

$\quad R_n \leftarrow random(0,1)\ and\ compute\ CH_{prob}(s);\ //given\ by\ (3.1)$

$\quad if\ (R_n\ <\ CH_{prob}(s))\ then$

$\quad\quad CCH\{s\} = TRUE;\ //node\ s\ be\ a\ candidate\ CH$

$\quad else$

$\quad\quad CCH\{s\} = FALSE;\ //node\ s\ not\ be\ a\ candidate\ CH$

$\quad end\ if$

$end\ if$

$SendToBS(IDu, (xu, yu), CCH(u))\ \leftarrow\ All\ nodes\ send\ messages\ to\ BS;$

$GAinBS(popt)\ \leftarrow\ Optimal\ probability\ is\ determined;$

$BC\ (popt)\ \leftarrow\ BS\ broadcasts\ a\ message\ back\ to\ all\ nodes;\ (II)$

SET − UP PHASE

$do\ \{\ //repeat\ for\ r\ rounds$

$r \leftarrow random(0,1);$

$if\ (E_{init}(s) > 0\ \&r\ mod\ \lfloor n/N_c \rfloor \neq 0)\ then$

$\quad R_n \leftarrow random(0,1)\ and\ compute\ CH_{prob}(s);$

$\quad if\ (R_n\ <\ CH_{prob}(s))\ then$

$\quad\quad CH\{s\} = TRUE;\ //node\ s\ be\ a\ CH$

$\quad else$

$\quad\quad CH\{s\} = FALSE;\ //node\ s\ not\ be\ a\ CH$

$\quad end\ if$

$end\ if$

$if\ (CH\{s\} = TRUE)\ then$

$BC\ (ADV)\ \leftarrow\ broadcast\ an\ advertisement\ message;$
$Join(IDi);\ //non-cluster\ head\ node\ i\ join\ into\ the\ closest\ CH$
$Cluster(c);\ //form\ a\ cluster\ c;$
$end\ if$
STEADY − STATE PHASE
$If\ (CH(s) = TRUE)\ then$
$Receive(IDi, DataPCK)\ //receive\ data\ from\ members;$
$Aggregate(IDi, DataPCK)\ //aggregate\ received\ data;$
$TansToBS(IDi, DataPCK);\ //transmit\ received\ data;$
$else$
$If\ (MyTimeSlot = TRUE)\ then$
$TansToCH(IDi, DataPCK);\ //transmit\ sensed\ data;$
$else$
$SleepMode(i) = TRUE;\ //node\ i\ at\ a\ sleep\ state$
$end\ if$
$end\ if$
$\}\ //\ one\ round\ is\ completed\ END$

A.2 HEED

HEED is an extension of LEEACH algorithm, where CH is selected by evaluating the residual energy of all nodes rather than randomly selected. Working of HEED is already expressed in section 3.3, here we are giving the pseudocode that was used to simulate this scheme.

Initialize
$S_{nbr} \leftarrow \{v: v\ lies\ within\ my\ cluster\ range\}$
$Complete\ and\ boradcast\ cost\ to\ S_{nbr}$
$CH_{prob} \leftarrow \max(C_{prob} \times \frac{E_{current}}{E_{max}}, p_{min})$ // as given in (3.6)
$is\_final\_CH \leftarrow FALSE$
Repeat
$If\big((S_{CH} \leftarrow \{v: vis\ a\ cluster\ head\}) \neq 0\big)$
$my\_cluster\_head \leftarrow least\_cost(S_{CH})$
$if(my\_cluster\_head = NodeID)$
$if(CH_{prob} = 1)$
$Cluster\_head\_mes(NodeID.final_{CH}\ , CH.cost)$
$is\_final\_CH \leftarrow TRUE$
$Else$
$Cluster\_head)meg(NodeID\ , tentative\_CH, cost)$
$Elseif\ (CH_{prob} = 1)$
$Cluster\_head\_msg(NodeID.Final\_CH\ , cost)$
$is\_final\_CH \leftarrow TRUE$
$Else\ Random(0,1) \leq CH_{prob}$
$Cluster\_head\_msg(NodID\ , tentative\_CH\ , cost)$
$CH_{previous} \leftarrow CH_{prob}$
$CH_{prob} \leftarrow \min(CH_{prob} \times 2\ , 1)$
$\boldsymbol{Until\ CH_{previous} = 1}$
Finalize

$If\,(is\_final\_CH = FALSE)$

$\quad If\,((S_{CH} \leftarrow \{v: v\ is\ a\ final\ cluster\ head\}) \neq 0$

$\quad\quad my\_cluster\_head \leftarrow least.cost(S_{CH})$

$\quad\quad join\_cluster(cluster\_head\_ID\ , NodID)$

$\quad Else\ Cluster\_head\_msg(NodID\ , final.CH\ , cost)$

$Else\ Cluster\_head\_msg(NodID\ , final.CH\ , cost)$

A.3 Network Assisted Data Collection

As described in section 3.4 a network assisted data collection algorithm was implemented in our project, where data is collected by a mobile sink by scanning a predefined network. Network is defined as a connected graph of root node, Navigation Agents (NA), Intermediate Navigation Agents (INA) and designated Getaways (DG). All the nodes send periodic updates to keep network alive. Random root nodes are selected to create a network. After network discovery, every node knows its distance to the root and to neighbors. Every node that is k hop distance with root node will declare itself NA by sending a '$Declare - NA$' message with TTL (Time to Live) value. All nodes receiving this declaration message accept the Navigation Agent as their parent NA, decrement the value of the TTL and rebroadcast the '$Declare - NA$' message. If a node finds the received TTL value to be zero, the receiving node sends out an '$Accept - NA$' message, otherwise, they decrement TTL and re-broadcast the message. A node receiving an '$Accept - NA$' message with $TTL = O$ from all its neighbors that are closer to the root (have lower hop count to the root), declares itself as a 'Navigation Agent'. This process continues till all nodes are marked either as a Navigation Agent (NA) or as Covered. Once NAs are declared and all network is informed by broadcast messages. All NAs will find shortest distance with each and other, all the nodes in between two NAs will declare Intermediate Navigation Agents. The nodes neither NA nor INA and are directly connected with NA will declare itself Designated Getaway (*DGs*) nodes. Mobile agent starts scanning the area from any point in the network; when it comes in contact with any nearest *NA* or *INA*, it will guide him to next of the *DG* and all other *DGs*. The pseudocode of algorithm is given as.

BEGIN

$\quad number\ of\ nodes\ (n)$

$\quad E_{init}(s) = E_0, \qquad \{s = 1,2, \dots, n\}$

Periodic Updates

$\quad node\ (s)\ Boradcast\ \ routing\ table\ every\ time\ (T)$

$\quad node\ (s) =\ Root\ route\ \{s1, s2, .., sn\}$

$\quad node\ (s) = neibour\ table\ \{n1, n2, \dots, nn\}$

$\quad node\ (s) = routing\ table\ \{r1, r2, \dots, rn\}$

Navigation Agent identification

$\quad Set\ number\ of Accept - NA\ messages\ received\ to\ zero$

Process Declare − NA messagefrom a neighbor:

$\quad Mark\ neighbor\ as\ a\ Navigation\ Agent\ neighbor\ table$

$\quad if\ \ (\ TTL\ value\ in\ Declare - NA\ message\ \ ==\ 0)$

$Broadcast\ an\ Accept - NA\ message\ to\ all\ neighbors\ with\ TTL\ =\ k - l$
$else$
$Decrement\ TTL\ value\ in\ Declare - NA\ message$
$Re - broadcast\ Declare - NA\ message$
$end\ if$

Process Accept − NA messagefrom a neighbor:

$lower\_neighbor\_count\ number\ of\ neighbors\ with\ lower\ hop\ count\ to\ root$
$If\ (\ TTL\ value\ in\ Accept - NA\ message\ ==\ 0)$
$If(\ number\ of Accept - NA\ messages\ received\ lower\_neighbor\_count)$
$Mark\ myself\ as\ a\ Navigation\ Agent$
$Broadcast\ a\ Declare - NA\ message\ to\ all\ neighbors\ with\ TTL\ =\ k - l$
$else$
$track\ how\ many\ Accept - NA\ messages\ were\ received$
$Increment\ Accept - NA\ messages\ received$
$end\ if$
$ttl\ value\ is\ greater\ than\ 0$
$else$
$Decrement\ TTL\ value\ in\ Accept - NA\ message$
$Re - broadcast\ Declare - NA\ message$
end

Intermediate Nodes identification

$All\ NA\ calculates\ shortest\ distance\ with\ other\ NAs$
$All\ NA\ broadcasts\ shortest\ distance\ with\ other\ NAs$
$Node\ (s)\ exists\ in\ shortest\ distance\ declare\ itself\ intermediate\ navigation\ agents\ INA$
$DG\ identification$
$If\ (Node(s)\ not\ NA\ and\ not\ INA\ and\ direct\ connected\ with\ NA)$
$Declare\ itself\ getaway\ Node\ GA$

Appendix B

Agi STK Simulation for Clustering

AGI-STK is a Systems Tool Kit designed by Analytical Graphics. It is a licensed software, provides real time simulation modeling and analysis system that can be deployed in land, air, sea and space. The Picture B.1 is the IDE and first startup screen.

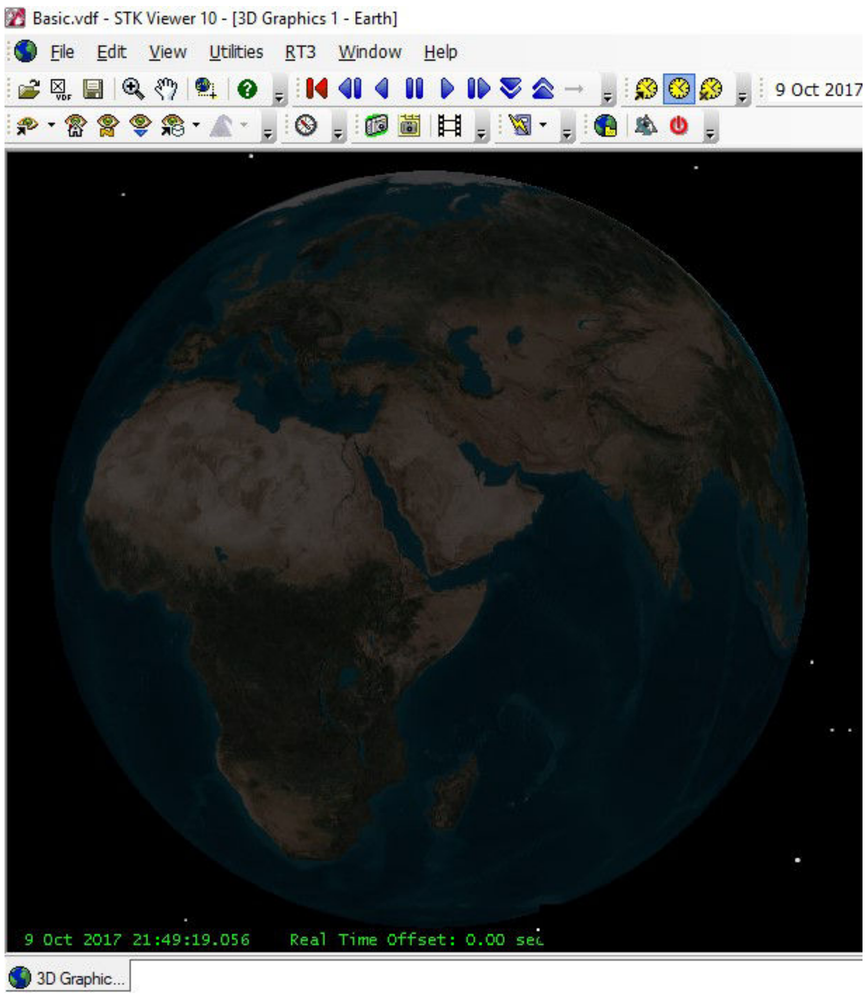


Figure B.1: AGI-STK stats up screen.

The main screen showing the world globe, it is STK imagery or maps module. STK has the high-resolution images of whole the work and this module runs in live mode. It is better to have good internet connection to design and monitor the desire mission progressing in real environment. From this globe, we can select any area to work, then we can put all the equipment and facilities on the selected area. The available objects in STK are shown in Figure B.2, these objects can be used to create out simulation module.

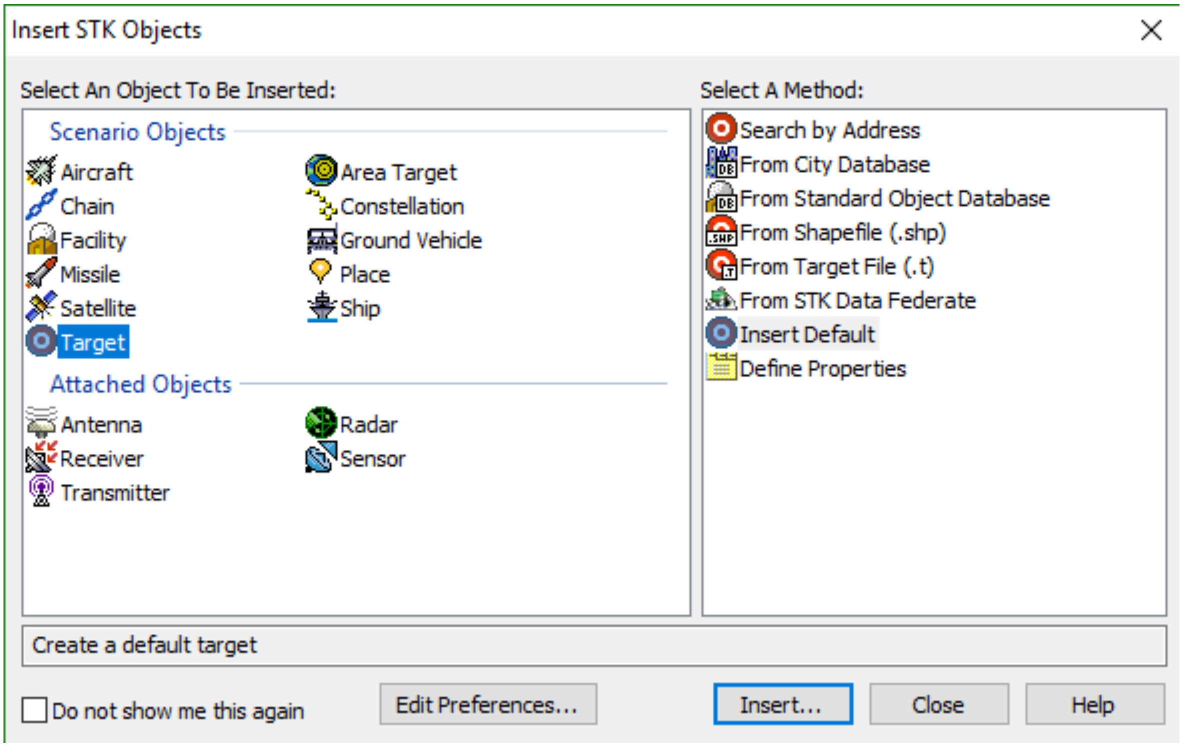


Figure B.2: STK objects.

From this list of objects, we may use

- Area target: to select our area of simulation,
- Aircraft: as UAV sink node,
- Place: where we will put sensors,
- Sensors: to monitor different parameters
- Antenna: one at each sensor and one on UAV for transmitting information,
- Receiver: one each for sensor node and UAV to receive information.

We selected Tabuk as target area and then we focused on the farm field as shown in Figure B.3. The right window is showing the 2D view of the target area while left is a 3D view. Bottom portion of the IDE is time line showing today's date and time right now. After runing the simulation, activities with respect to time will be shown here.

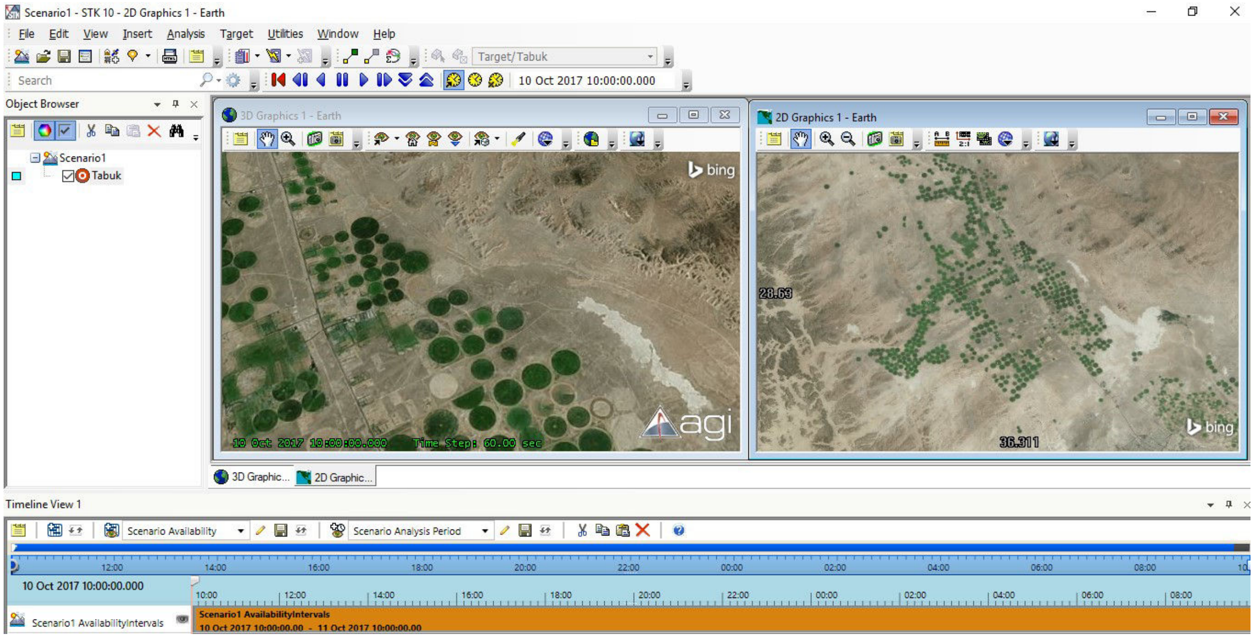
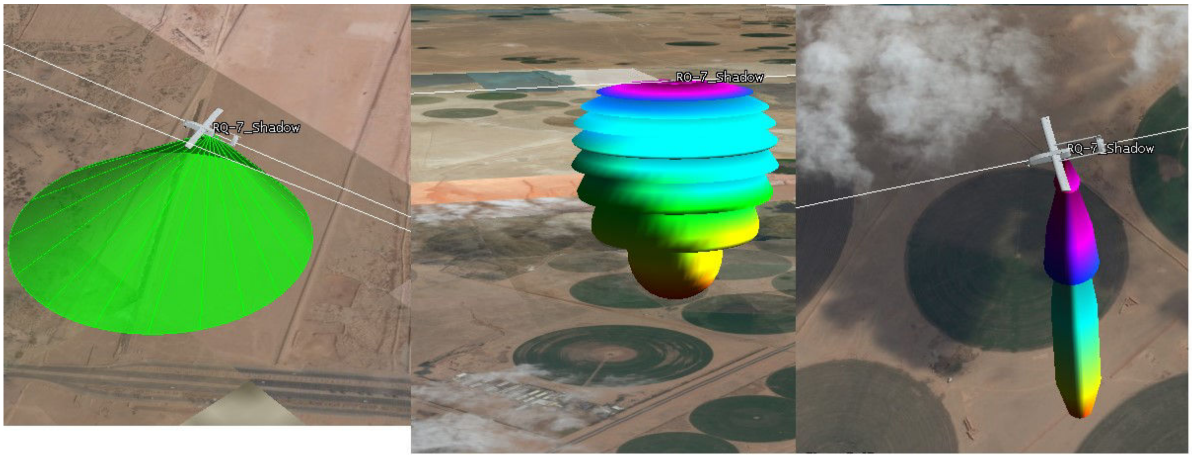


Figure B.3: Tabuk targeted area.

We took RQ-7 UAV from object list shown in Figure B.2. As it is a wings airplane, we need a landing strip to operate it. If there is any road or airport nearby, we can use it as landing strip; otherwise, we can built it by using STK objects. In our case, we are using a nearby road as landing strip. 2D and 3D views of UAV and its path are shown in Figure B.4.

Figure B.4: UAV landing strip and path in 2D and 3D.

The next step is to equipped UAV with required antenna and receiver. Some examples of antennas and radars that can be made and installed over UAV are shown in Figure B.5.



A) Simple comic with 45 deg.

B) Synthetic aperture radar Receiver, with Helix shape, frequency 2.9 GHz, diameter 1 meter, LNA gain 10dB, reflection celling 0.25km

C) Synthetic aperture Transmitter with frequency 2.9 GHz, Beam width 5 deg, diameter 1-meter, Main lobe gain 27.35 dB, Power 40 dBm

Figure B.5: Some examples of antenna and radars mounted over UAV.

Once UAV is ready, equipped with required antennas and receivers, and path is given, we tested it in real field and checked its take off/ landing. The next step is to install sensors in crop fields. To install a sensor, first of all, there is a need to mark some places in the field. We selected place object from the list and put it on the map at various required places. On each place we can put a sensor. Once sensor is installed in the field, it requires some settings and can be furnished with any kind of extra devices like transmitter or receiver. An example is give in Figure B.6.

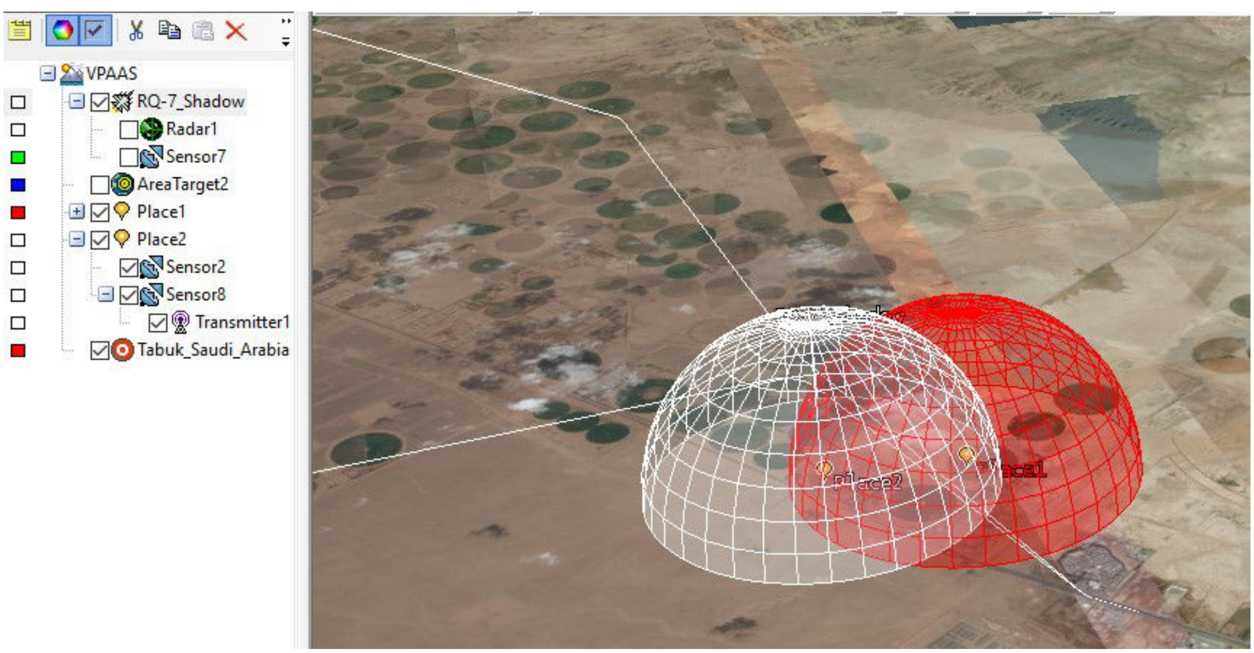


Figure B.6: Sensor nodes and their ranges.

In Figure B.6, left panel shows the list of objects used in our scenario. This panel is made in hierarchical way at the top it is RQ-7 Shadow UAV, which is equipped with radar and one sensor. All the installed sensor are having a transmitter and receiver pair for communication. Two different colors of cones representing the transmission ranges of these sensors.

Now we are going to develop our proposed data gathering scheme. We installed some heterogenous sensors with different sizes and communication ranges as shown in Figure B.7 and a closer look in B.8.

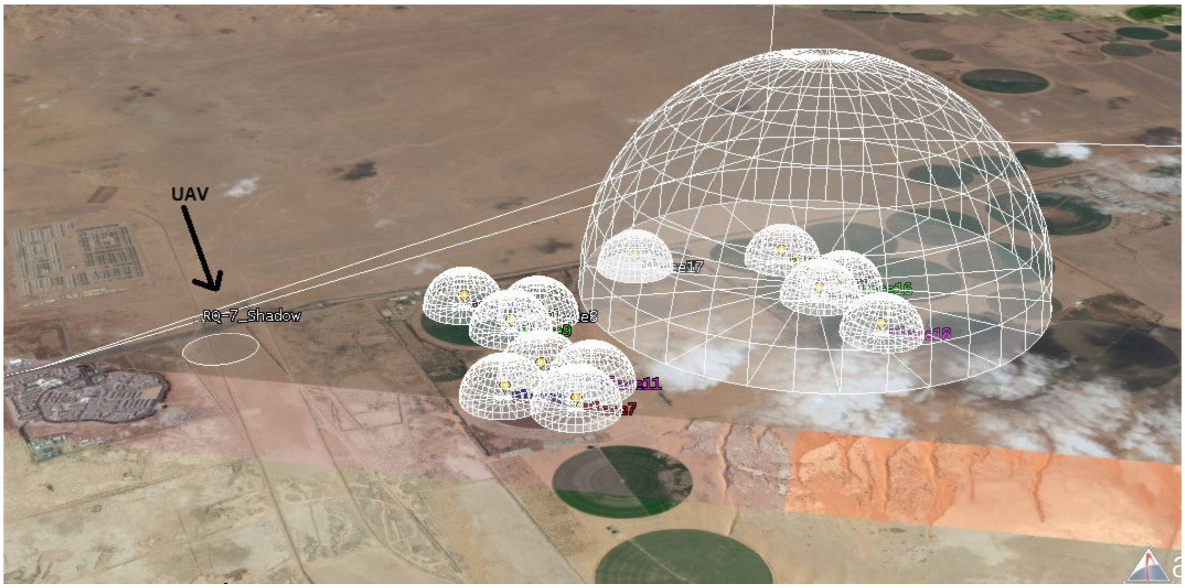


Figure B.7: UAV path and sensor nodes.

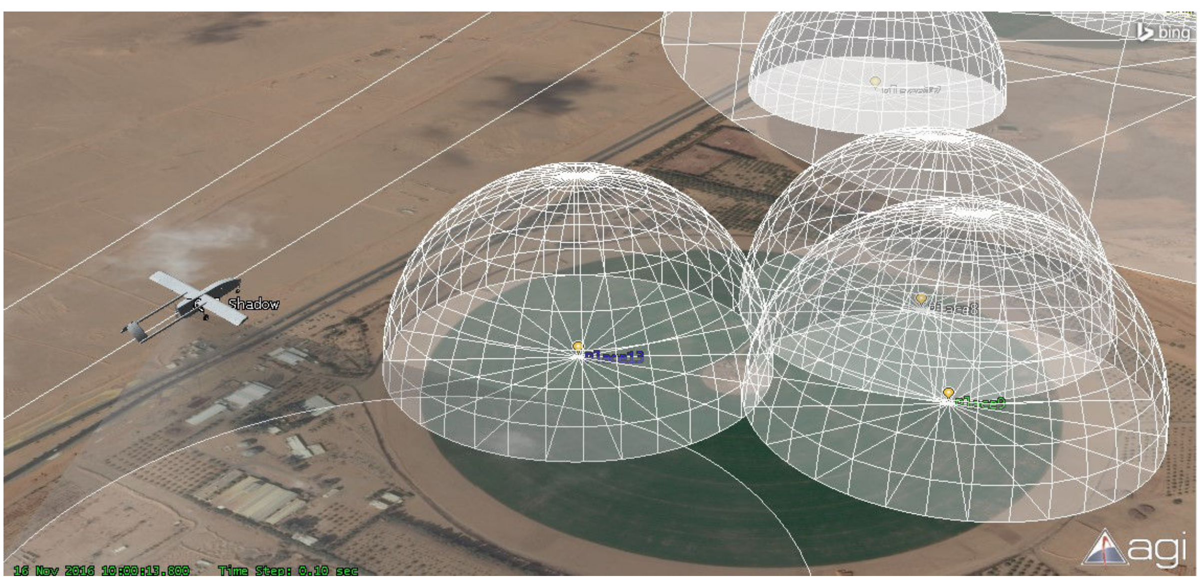


Figure B.8: Close up view of sensor nodes and UAV path.

B.1 AGI STK MATLAB Integration

AGI STK can be integrated with many other languages and applications like C++, visual studio, MATLAB. To install STK drivers in MATLAB, one should open MATLAB, browse to the MATLAB_Connectors or MATLAB_Connectors_X64 folder on the STK DVD, click on setup.exe and follow the install instructions.

Once STK is installed, STK and MATLAB can work together for better understanding.

Now we can open a MATLAB empty file or existing programing file and can write STK code also. We can use the MATLAB script file to build a simple STK scenario, from which MATLAB processed data can be pass on to STK for simulation.

Create a new instance of STK11 in MATLAB.

1. In MATLAB, type the following code into the Command window:
   app = actxserver('STK11.application')
   app.UserControl = 1
2. Grab a handle on the STK application root.
   root = app.Personality2

Now STK is launched, we are giving only some instructions to write a new scenario as example:

scenario = root.Children.New('eScenario','MATLAB_Starter')
Set the analytical time period:
scenario.SetTimePeriod('Today','+24hr')
Reset the Animation Time:
root.ExecuteCommand('Animate * Reset')

With a newly created scenario, it's time to populate it with objects. Take a moment to create a target and a RQ-7 Shadow UAV using MATLAB.

Add a target object to the scenario.
target = scenario.Children.New('eTarget','GroundTarget');
Move the target object to a desired location.
target.Position.AssignGeodetic(50,-100,0)
Add a UAV.
UAV = scenario.Children.New('eAircraft','RQ-7 Shadow')

In the same way, we can develop the whole simulation model in MATLAB and can use the commands of both (MATLAB and STK). In this way, simulation model can be created in both programs at a time where, we can use the power of MATLAB to do complex and heavy computation while STK can be used to visualize the scenario in real grounds.

Appendix C

OMNeT++ Simulation for Clustering

C.1 OMNet++ Installation

System Requirements

The latest available version of OMNeT++ 5.1 released on April 2017 can be downloaded from https://omnetpp.org/omnetpp. however, we used OMNet $^{++}$ 4.6 to develop simulation model, installed under windows environment and the system has following specifications:

- Processor Intel core i7,
- Operating system Windows 10 64 bit,
- Ram 8 GB,
- HDD 1TB.

License

OMNeT++ is an open source, free ware software for academic and educational use only. Commercial license is required for Commercial use that can be obtained from SimulCraft Inc, www.omnest.com.

OMNeT++ Windows Installation

OMNeT++ installation guide is given at https://omnetpp.org/doc/omnetpp/InstallGuide.pdf but step by step procedure for beginners is not given. We faced some difficulties and problems while installing it first time (especially pre-installation preparation), sequence of installation and post installation setup are important and given below:

Precautions:

1. Windows 32 bit support is no longer available in OMNeT++ version 5.1 and above.
2. OMNeT++ installation directory should not contain any space in name.
3. Download correct version of OMNeT++ according to operating system installed on the machine. Windows 32bit, Windows 64 bit, Linux and MacOS all versions are available on the omnetpp.org website.
4. Installation path and configure.user file should be check before installation.
5. Windows environment variable should be properly link to include, lib and bin directories after installation.

Installation steps

Download the required software from above mentioned webpage and unzip the package into the directory of your choice and follow the steps as under:

1. Set the path in configure.user file,
2. Run mingwenv.cmd file. mingwenv is,
3. Once Linux prompt $ is appear in mingwnv type the following,

   $ ./configure

   $ make
4. Configure and make file will take a long time (20 to 30 min depends upon processing speed) be patient.

Post installation setup

If OMNet is not working properly, path and environment variable should be rechecked and compare with given below list. If [C:\omnetpp-4.6] is the directory where it is installed, then path and environment variables will be as:

Path=;

C:\omnetpp-4.6;
C:\omnetpp-4.6\bin;
C:\omnetpp-4.6\ide\jre\bin\client;
C:\omnetpp-4.6\ide\jre\bin;
C:\omnetpp-4.6\ide\jre\lib\i386;
C:\omnetpp-4.6\tools\win32\usr\bin;
C:\omnetpp-4.6\tools\win32\mingw32\bin
C:\omnetpp-4.6\tools\win32\usr\local\bin;
C:\omnetpp-4.6\tools\win32\usr\bin\site_perl;
C:\omnetpp-4.6\tools\win32\usr\bin\vendor_perl;
C:\omnetpp-4.6\tools\win32\usr\bin\core_perl;
C:\omnetpp-4.6\include\omnetpp;
OMNETPP_ROOT=C:\omnetpp-4.6/;
OMNETPP_IMAGE_PATH=C:\omnetpp-4.6\images.

Running OMNet ++

First of all double click "Mingwenv" icon to start Linux environment in windows.

Once Linux environment prompt appears, type OmnetPP and press enter then OMNeT++ 4 will start loading as Figure C.1.

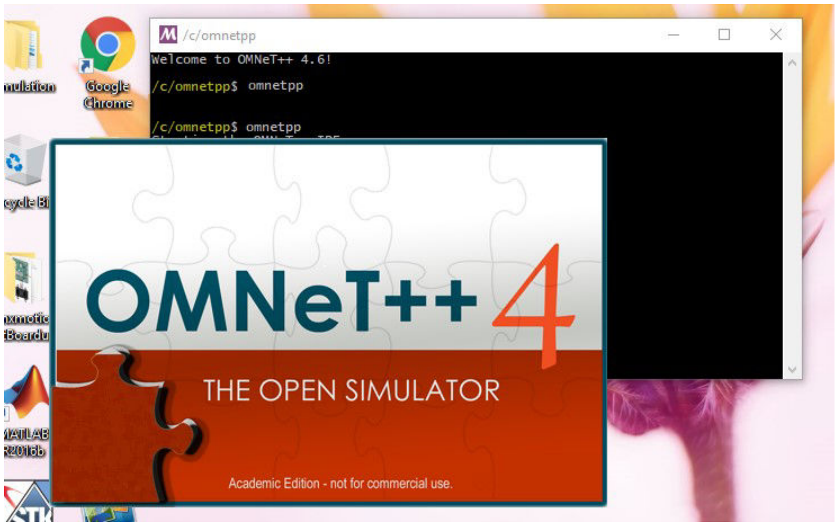


Figure C.1: OMNeT ++ starting screen.

Once OMNeT++ loaded successfully, the first screen will appear like Figure C.2. It is a standard Eclipse based IDE, we are not giving its introduction.

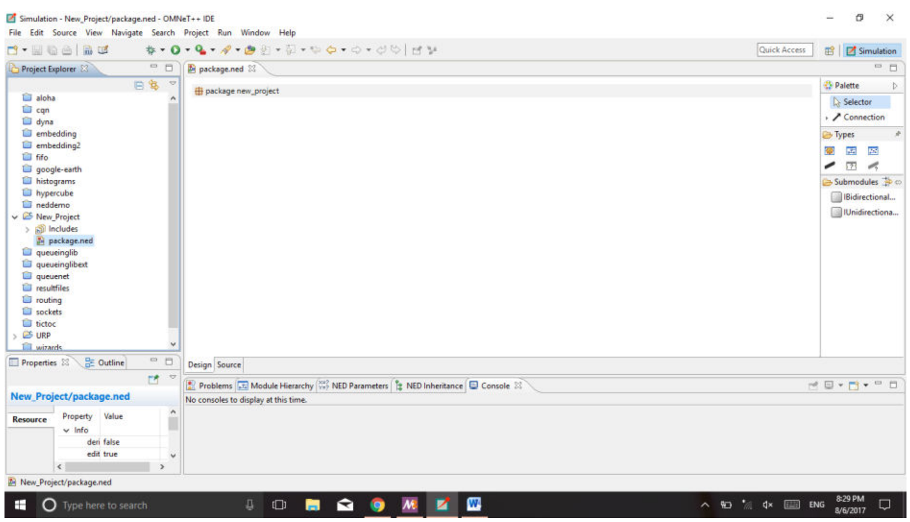


Figure C.2: OMNeT++ IDE interface.

C.2 First Practical Example in OMNeT++

Here, we are presenting first hands on example that can be built in OMNeT++. To build a wireless scenario, we can take one Base Station (BS) and connect 100 nodes to it, wirelessly. Initially, we assume all nodes are sending packets directly to BS. To develop a OMNeT++ simulation 3 files are very important and one should have clear understanding of working of these files.

1. NED file: It is a file like a canvas. We draw all our simulation modules in this file. In this example, we will take only two modules: BS and wireless nodes. Once a single node is developed we can replicate it in the network as many as we like.

2. CC file: Each module droped in NED file will have one CC file of same name. In this CC file implementation of this module is given.

3. INI file: In this file input parameters, network settings and different simulation cases are given. In every simulation, we can change the inputs and scenarios in this file to evaluate the system under different conditions rather than made changes in the whole source code.

Let's start simulation:

As first step, load OMNet++ and open a new empty project from file menu then insert new blank NED file.

Drag and drop two modules on canvas one BS and one sensor node as shown in Figure C.3.

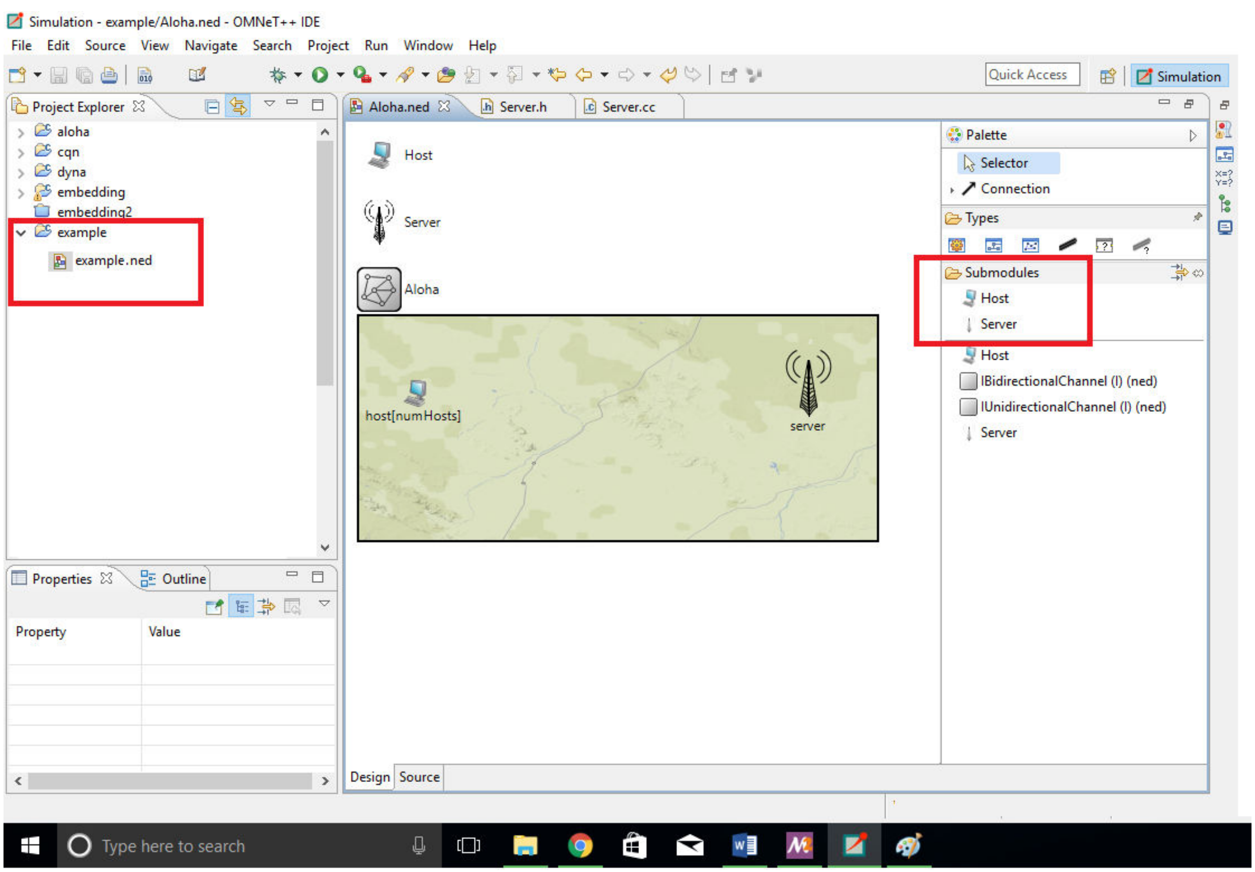


Figure C.3: Two modules are dropped on canvas from module list in NED file.

The next step is to give implementation to this NED file. See simple code of this NED file given below, afterward, we will describe the purpose of each instruction.

```
network Wireless_Example
{
    parameters:
        int numHosts;  // number of hosts
        double txRate @unit(bps);  // transmission rate
        @display("bgi=background/terrain,s;bgb=463,200");
    submodules:
        server: Server;
        host[numHosts]: Host
       {
            txRate = txRate;
            slotTime = slotTime;
            @display("is=s");
        }
}

simple Host
{
    parameters:
        double txRate @unit(bps);         // transmission rate
        double radioDelay @unit(s);       // propagation delay of radio link
        volatile int pkLenBits @unit(b);  // packet length in bits
        volatile double iaTime @unit(s);  // packet interarrival time
        double slotTime @unit(s);         // zero means no slots (pure Aloha)
        @display("i=device/pc_s");
       gates:
        output out @directOut;

}

simple Server
{
    parameters:
    gates:
        input in @directIn;
}
```

The above-mentioned code can be divided in three blocks network, host and server. The first block in network contains two portions parameter and submodule. Here, we defined two parameters "numHosts" is used to declare number of hosts wanted in the simulation and "txRate" is the transmission rate for communication. Both variables are not given any value, they will pick their values form INI file at the run time that can be changed easily for every simulation. This network has two submodules: one is server and other is host. We have taken server as single entity but host is written as "host[numHosts]" which it is an array of type host up to numHosts. numHost also does not have fixed value, and its value will be taken from INI file, so simulation can be run with different number of nodes. Each submodule should have to be given implementation separately after network block.

In the host submodule, parameters should be written as in network portion, but special things are gates. We declared only one output gate and did not connect it with any other module because it is a wireless node and all the packets will be sent to air through this gate.

Implementation of server is quite simple, since having only one input gate and all the packets will be received on this gate. It is a simple example, no transmission protocol is implemented so far, if two packets will be received at the same time, collision will occur otherwise attempt will be successful.

Then, we have to insert two more files in our example. There is a hierarchical list in the left panel of the windows showing all the simulation models and files under each category. Right click on example and insert two new and blank CC files, one named as “server” and the other “host” these names should be same as in the modules drawn on canvas. See the C++ code, we written in Host.cc as given bellow:

```
Host::Host(){endTxEvent = NULL;}
Host::~Host(){ cancelAndDelete(endTxEvent);}
void Host::initialize()
{
    txRate = par("txRate");
    radioDelay = par("radioDelay");
    iaTime = &par("iaTime");
    pkLenBits = &par("pkLenBits");
    slotTime = par("slotTime");

    endTxEvent = new cMessage("send/endTx");
    scheduleAt(getNextTransmissionTime(), endTxEvent);
}


void Host::handleMessage(cMessage *msg)
{

    if (state==IDLE)
    {
        // generate packet and schedule timer when it ends
        char pkname[40];
        state = TRANSMIT;
        cPacket *pk = new cPacket(pkname);
        pk->setBitLength(pkLenBits->longValue());
        simtime_t duration = pk->getBitLength() / txRate;
        sendDirect(pk, radioDelay, duration, server->gate("in"));
        scheduleAt(simTime()+duration, endTxEvent);
    }
    else if (state==TRANSMIT)
    {
        // endTxEvent indicates end of transmission
        state = IDLE;

        scheduleAt(getNextTransmissionTime(), endTxEvent);

        // update network graphic
```

```
    }
    else
    {
        error("invalid state");
    }
}
```

Above mentioned Host implementation have four functions contractor “Host ( )”, destructor “~Host()”, “initialize()” and “handleMassage()”. In initialize function, all variables with values are given in INI file then we created a message and selected a time, to sent on-air.

Once the scheduled time is over, host will generate a message and hander over to handlemessage function. If handlemessage function is in IDEL state then it will prepare the channel and will sent the message to air and if the function will be in transmission state then it will get next schedule for message transmission.

Functionality of server node is bit tricky and given below:

```
void Server::initialize(){----}
void Server::handleMessage(cMessage *msg)
{
    if (msg==endRxEvent)
    {
        EV << "reception finished\n";
        channelBusy = false;
        emit(channelStateSignal,IDLE);


        else
        {
            emit(collisionLengthSignal, dt);
        }

         emit(receiveBeginSignal, receiveCounter);

    }
    else
    {
        ASSERT(pkt->isReceptionStart());
        if (!channelBusy)
        {
            EV << "started receiving\n";
         }
        else
        {
            EV << "another frame arrived while receiving -- collision!\n";
            emit(channelStateSignal, COLLISION);

        }
        channelBusy = true;
        delete pkt;
    }
```

```
}
void Server::finish(){------}
```

Server implementation consists of three function initialize(), handleMessge() and finish(). We can declare and initialize our variable in initialize function. All the messages received at server in gate will be handed over to messagehandler. The code of handlemessage is written as if the server is in IDLE state then it will start receiving message and during receiving a message, if another message arrived then it will generate collision notification and count will number of collisions. The last function is finish(), in this function all the variables and watched performance are written in a log file before terminating the simulation run.

C.3 Extension of This Simulation

The server module written in this simulation can be extended to multithreaded. In this case, every time a server receive a message it will create its clone (thread) and starts communicating with this node independently then, server become free to serve any other wiling node.

C.4 Developed URP

As described above example, we developed our proposed system. The developed simulation is special in a way that data sink node as BS in above example is considered as mobile UAV. Another difference is, we used dual NIC cards to deal with dual frequencies. Furthermore, we implement data gathering scheme proposed in section 3.5. a snapshot of developed NED file is shown in Figure B.4 and later, other files (INI, CC) are shown.

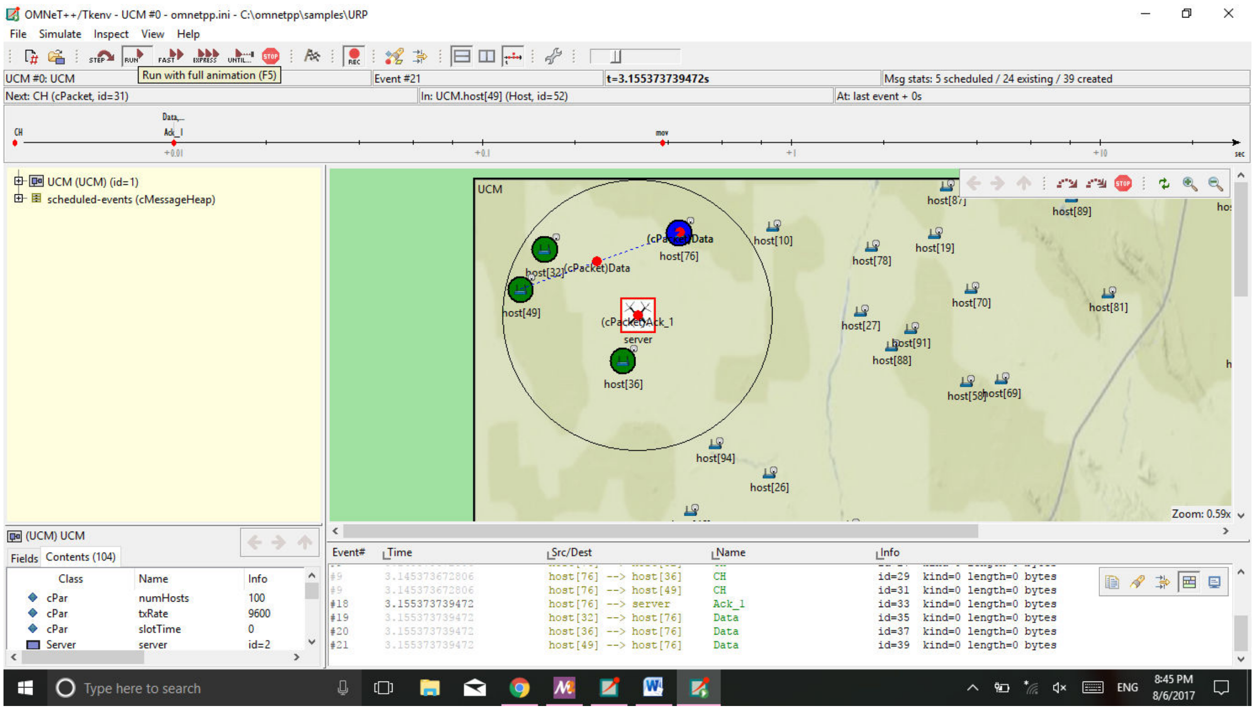


Figure C.4: First hand on example.

C.4.1 NED file

In this simulation model, we used two objects: sensor nodes and UAV. Connectivity of all these components and in/out gates of each component data movement are also developed in this file. The source code of this file is shown below.

```
simple Server
{
    parameters:
        @signal[state](type="long");
        @statistic[radioState](source="state"; title="Radio state";
        enum="IDLE=0,TRANSMIT=1"; record=vector);
        double txRate @unit(bps);         // transmission rate
        double radioDelay @unit(s);       // propagation delay of radio link
        volatile int pkLenBits @unit(b);  // packet length in bits
        volatile double movTime @unit(s);  // packet interarrival time
        double slotTime @unit(s);
        int rounds;  // number of rounds
        int in_x;
        int in_y;
        int fi_x;
        int fi_y;
        int numHosts;

        @display("i=device/coptor,white");
    gates:
        input in @directIn;}
```

```
simple Host
{
    parameters:
        double radioDelay @unit(s);       // propagation delay of radio link
        double Pa; // = 0.9              # // 1000 mJ
        double Pr;// = 0.000005          #// 50 nJ energy required per byte transfer
        double Pu;// = 0                 #; // UAV will transmit
        double Prr;// = 0.000001 #; // 10 nJ  energy required per byte receive
        double Pf;// = 0.0000005 #; // 5 nJ  energy required per byte fusion
        double Beacon_size;//= 1 #; //1 Byte
        double data_size;//=90   #;
        double pemp;//=0.0000008 #;
        int UAV_height;
        double WiFi_throughput @display("i=device/accesspoint;is=vl");
        @display("i=abstract/accesspoint");
    gates:
        input in @directIn;
}
network UCM
{
    parameters:
        int numHosts;  // number of hosts
        double txRate @unit(bps);  // transmission rate
        double slotTime @unit(ms);  // zero means no slots (pure Aloha)
        @display("bgi=background/terrain,s;bgb=2000,2000");
    submodules:
        server: Server {
            txRate = txRate;
            slotTime = slotTime;
            numHosts = numHosts;
            @display("r=250;p=325,88;is=vl");
        }
        host[numHosts]: Host {
            @display("is=s;p=46,51");
        }
}
```

C.4.2 INI file

In this file, input parameters are given, the source code of this file is given as under:

```
[General]
network = UCM
debug-on-errors = true
record-eventlog = true
UCM.numHosts = 100
UCM.server.rounds = 100
UCM.slotTime = 0
UCM.txRate = 9.6kbps
UCM.server.pkLenBits = 952b
UCM.server.radioDelay = 10ms
UCM.host[*].radioDelay = 10ms
UCM.host[*].Pa = 0.99           #  1000 mJ
UCM.host[*].Pr = 0.000005       #  50 nJ energy required per byte transfer
```

```
UCM.host[*].Pu = 0               #  UAV will transmit
UCM.host[*].Prr = 0.000001       #  10 nJ  energy required per byte receive
UCM.host[*].Pf = 0.0000005       #  5 nJ  energy required per byte fusion
UCM.host[*].Beacon_size= 1       #  1 byte
UCM.host[*].data_size=150        #  150 bytes
UCM.host[*].pemp=0.0000008
UCM.host[*].UAV_height=400
UCM.host[*].WiFi_throughput= 15000000

[Config UCM]
description = "UAV Crop Management"

UCM.server.movTime = exponential(0.8s)
UCM.server.in_x=0
UCM.server.in_y=250
UCM.server.fi_x=2000
UCM.server.fi_y=250
```

C.4.3 Server.cc file

Each component drawn on canvas (NED file) needs to be implemented. Server is developed in such a way that to start data processing by sending beacon message. Then it collects basic information from sensor nodes and nominates one of them as Cluster Head (CH). Then UAV moves toward selected CH and collects data by reducing its height up to some agreed point. Once data is collected from a cluster, UAV starts again the same process to make another cluster. The source code for server.cc is given below:

```
#ifndef __UCM_HOST_H_
#define __UCM_HOST_H_
#include <omnetpp.h>
#include <string>
#include <iostream>
#include <memory.h>
class Server : public cSimpleModule
{

public:
    int ah[500],dead_nodes[500]; //active connected hosts list
    int dh[500]; // desable hosts
    int ah_count; // active hosts count for currect round
    int round;
    int cround;
    int dead_count,no_heads;
    double t_data;
private:
    // parameters
    int numHosts;

    cPar *movTime;
    cPar *pkLenBits;
    SimTime radioDelay;
    // state variables, event pointers etc
    cModule  *host;
```

```
    cMessage *mov;
     enum {setup=0, receive=1} state;

    double txRate;
    int counter;
    char str1[500];
    int in_x,in_y,fi_x,fi_y,step_x,step_y;
    int nx,ny;
    char result[100];
    int reset;
  public:
    Server();
    virtual ~Server();
  protected:
    virtual void initialize();
    virtual void handleMessage(cMessage *msg);
    simtime_t getNextMovTime();
    void inRange(int nodeId);
    void Breadcast(int nodeId,int st);
    void deflt();
    virtual void finish();
    cOutVector dead_nodes_out;
    cOutVector total_dead_out;
    cOutVector no_heads_out;
    cOutVector total_data_out;

};

Define_Module(Server);
Server::Server()
{
    mov = NULL;
    for(int i=0; i<500; i++) ah[i]=dh[i]=0;
    strcpy(result,"");

}

Server::~Server()
{
    cancelAndDelete(mov);
}

void Server::initialize()
{
    reset=dead_count=no_heads=t_data=0;
    txRate = par("txRate");
    round= par("rounds");
    cround=1;
    radioDelay = par("radioDelay");
    movTime = &par("movTime");
    //pkLenBits = &par("pkLenBits");
    numHosts = par("numHosts");
    in_x = par("in_x");
    in_y = par("in_y");
    fi_x = par("fi_x");
```

```
    fi_y = par("fi_y");
    step_x=double(fi_x - in_x)/33;
    step_y=double(fi_y - in_y)/33;
    nx=in_x;
    ny=in_y;
    for(int i=0;i<500;i++)dead_nodes[i]=0;

    mov = new cMessage("mov");
    state = setup;
    counter=0; //pkCounter=0;

    if (ev.isGUI())
    {
        getDisplayString().setTagArg("p",0,in_x);
        getDisplayString().setTagArg("p",1,in_y);
    }


    scheduleAt(getNextMovTime(),mov);
}

void Server::handleMessage(cMessage *msg)
{

    if (msg==mov)
    {
      if (cround>=round)endSimulation();
      else if((in_x<=fi_x)&&(in_y<=fi_y)) { if (nx >=fi_x || ny> fi_y) {ev<<"called
      1"<<endl; deflt();}}
      else if((in_x<=fi_x)&&(in_y>=fi_y)) { if (nx >=fi_x || ny< fi_y) {ev<<"called
       2"<<endl; deflt();}}
      else if((in_x>=fi_x)&&(in_y<=fi_y)) { if (nx <=fi_x || ny> fi_y) {ev<<"called
       3"<<endl; deflt();}}
       else if((in_x>=fi_x)&&(in_y>=fi_y)) { if (nx <=fi_x || ny< fi_y) {ev<<"called
       4"<<endl; deflt();}}

      getDisplayString().setTagArg("p", 0, (nx+=step_x)); // movement in x direction
      getDisplayString().setTagArg("p", 1, (ny+=step_y));// movement in y direction

        if (state == setup)
        {

            if (counter>=4) // UAV will send 3 beacon without start transmission
            {
                ev<<"counter is greater then 2 ---> entered in comunication
                    phase"<<endl;
                state=receive;
                for(int i=0; i<numHosts; i++) { if (dh[i]==0 && dead_nodes[i]==0)
                    inRange(i); }
                for (int ii=0;ii<ah_count;ii++)  Breadcast(ah[ii],0);
                if(ah_count==0){counter=0;state=setup;}
            }
            counter++;

        }
```

```
        scheduleAt(getNextMovTime(), mov);
    }

    if (strcmp(msg->getName(),"Data_all")==0)
    {
        state = setup;
        ah_count=0; // active hosts count
        counter=0;
        no_heads++;
        t_data+=double(msg->par("data"));
        ev<<"data receive="<<double(msg->par("data"))<<" total data="<<t_data<<endl;
       // pkCounter=0;
        for(int i=0; i<500; i++) ah[i]=0;
        strcpy(result,"");
        strcpy(str1,"");
        delete msg;

    }
    else if (strcmp(msg->getName(),"dead")==0)
    {
        //ev<<"dead message is received"<<endl;
        int dd=int(msg->par("dead"));
        dead_nodes[dd]=1;
        dead_count++;
        //ev<<"Number of dead nodes are="<<dead_count<<endl;
        dead_nodes_out.record(dead_count);
        delete msg;

    }

}

simtime_t Server::getNextMovTime()
{
    simtime_t t = simTime()+movTime->doubleValue();
        return t;
}

void Server::inRange(int nodeId)
{
    double a= atoi(getDisplayString().getTagArg("p",0));
    double b= atoi (getDisplayString().getTagArg("p",1));
    double redious= atoi (getDisplayString().getTagArg("r",0));

    string str = "host[";
    stringstream ss;
    ss <<nodeId;
    str += ss.str();
    str+= "]";
    host = simulation.getModuleByPath(str.c_str());
    if (!host) error("host_5 not found");

    double c= atoi (host->getDisplayString().getTagArg("p",0));
    double d= atoi (host->getDisplayString().getTagArg("p",1));
    double distance_uav=sqrt(pow(a-c,2)+pow(b-d,2));
```

```
    if (distance_uav < redious)
    {
        int tag=0;
        for(int g=0; g<ah_count; g++) if (ah[g]== nodeId) tag=1;// if node is not
                                                          already in array add it

        if (tag==0)
        {
            dh[nodeId]=1; // activated will not be activated again
            ah[ah_count]=nodeId;
            itoa(ah[ah_count],str1,10);
            if(ah[ah_count]<10)strcat(result,"0");
            strcat(result,str1);
            strcat(result,"-");
            ah_count ++;
         }

     }

}
void Server::Breadcast(int nodeId, int st)
{
    string str = "host[";
    stringstream ss;
    ss <<nodeId;
    str += ss.str();
    str+= "]";
    host = simulation.getModuleByPath(str.c_str());
    if (!host) error("host_6 not found");
    double c= atoi (host->getDisplayString().getTagArg("p",0));
    double d= atoi (host->getDisplayString().getTagArg("p",1));
    double distance_end=sqrt(pow(c-fi_x,2)+pow(d-fi_y,2));
    //ev<<nodeId<<"distance="<<distance_end<<endl;
    //double slop_end = (double(d)-fi_y)/((c-fi_x)*10);
    //double cost = abs(distance_end * slop_end);//double(distance_end)/100;//
    //cost=double(cost)/1000;
    double cost=distance_end/10000;
    cPacket *pk = new cPacket("Beacon");
    pk->addPar("Pu")=cost;
    pk->addPar("t") =result;
    pk->addPar("st")=st;
    int g=ah_count;
    double duration=double (1)/15000000;
    ev<<"list sent="<<result<<"duration"<<duration<<endl;
    sendDirect(pk,radioDelay,duration, host->gate("in"));

}
void Server::deflt()
{
    state = setup;
    ah_count=0; // active hosts count
    counter=0; // how much beacon UAV will send before transmission starts
    //pkCounter=0;
    //total_dead_out.record(dead_count);
```

```
    for(int i=0; i<500; i++) ah[i]=dh[i]=0;
    strcpy(result,"");
    strcpy(str1,"");
    for(int i=0; i<numHosts; i++)
    {
        string str = "host[";
        stringstream ss;
        ss <<i;
        str += ss.str();
        str+= "]";
        host = simulation.getModuleByPath(str.c_str());
        if (!host) error("host_6 not found");
        host->getDisplayString().removeTag("t");

        if (strcmp(host->getDisplayString().getTagArg("b",3),"black")==1)
         host->getDisplayString().removeTag("b");

    }
    /*///////// diagonal ////////////
    in_x=0;
    in_y=0;
    fi_x=2000;
    fi_y=2000;
    step_x=(fi_x - in_x)/33;
    step_y=(fi_y - in_y)/33;
    nx=in_x;
    ny=in_y;
    ///////////////////////////////////////*/
   ////////////////// whole area /////////
    if (fi_y>=1750 ||in_y>=1750)
    {
        in_x=0;
        in_y=250;
        fi_x=2000;
        fi_y=250;
    }
    else
    {
        in_x=0;
        in_y+=500;
        fi_x= 2000;
        fi_y+=500;
    }
    step_x=(fi_x - in_x)/33;
    step_y=(fi_y - in_y)/33;
    nx=in_x;
    ny=in_y;
    ////////////////////////////////////////////
    //ev<<"reset"<<in_x<<" x "<<in_y<<"--"<<fi_x<<" x "<<fi_y<<endl;
    reset=1;
    cround++;
    ev<<"default called round="<<cround<<endl;

}
```

```
void Server::finish()
{
    ev<<"total data received="<<t_data<<endl;
    //total_data_out.record(t_data);
    EV << "duration: " << simTime() << endl;
    //no_heads_out.record(no_heads);
    //recordScalar("duration", simTime());
}

}; //namespace
```

C.4.4 Host.cc file

A sensor node is activated by receiving beacon message from UAV. The sensor nodes have sufficient energy and higher probability then threshold reply back to UAV beacon. All the nodes accept nominated node as CH and send join request. CH accepts join request and assigns time slot for communication. All the nodes send data to CH and CH relay it to UAV. The source code is as under:

```
#include <cdisplaystring.h>
#include <cenvir.h>
#include <cmessage.h>
#include <cmsgpar.h>
#include <cobjectfactory.h>
#include <coutvector.h>
#include <cpar.h>
#include <cregistrationlist.h>
#include <csimulation.h>
#include <distrib.h>
#include <onstartup.h>
#include <regmacros.h>
#include <simtime.h>
#include <simutil.h>
#include <cstdlib>
#include <cstring>

#ifndef __UCM_SERVER_H_
#define __UCM_SERVER_H_

#include <omnetpp.h>
#include <stdio.h>
#include <memory.h>
#include <string>
#include <iostream>
#include <memory.h>

namespace UCM {
class Host : public cSimpleModule
{
  private:
    // state variables, event pointers
    double
```

```
    // evaliable energy consumption rate UAV periority value
    Pa,Pr,Pu,Prr,Pf,Beacon_size,pemp,beacon_size,data_size,height,WiFi_throughput;

    int array[500];     // array of connected nodes id
    int array_length;   // array length of connected nodes
    double Pt_all[500]; // array of probabilities of  nodes
    int tt,pp;
    int dead_statuc;
    int reset_status;
    double total_con_energy;    //cLongHistogram dead_nodes_status;
    cOutVector nodes_energy;
    SimTime radioDelay;

    cMessage *endRxEvent;
    cPacket *Beacon;

    enum {IDLE=0, TRANSMISSION=1, COLLISION=2};
    cModule  *Server;
  public:
    Host();
    virtual ~Host();
  protected:
    virtual void initialize();
    virtual void handleMessage(cMessage *msg);
    virtual void finish();
    void connected(cMessage *msg);
    void broadcast();
    void def();
    void dead();
};
#endif
namespace UCM
{

    Define_Module(Host);

Host::Host() // constructor
{
    endRxEvent = NULL;
}

Host::~Host() // destructor
{
    cancelAndDelete(endRxEvent);
}

void Host::initialize()   // iniliazation function
{
    getDisplayString().setTagArg("p", 0,intuniform(0,1990) );
    getDisplayString().setTagArg("p", 1,intuniform(0,1990));

    array_length=tt=pp=Pu=total_con_energy=0;        // array of connected nodes,
activated by UAV
    for (int i=0;i<500;i++) array[i]=Pt_all[i]=0;
    Pa = par("Pa");             // = 0.9; // 1000 mJ
```

```
    Pr = par("Pr");             //0.000005; // 50 nJ energy required per byte transfer
    Prr = par("Prr");           //0.000001; // 10 nJ  energy required per byte receive
    Pf = par("Pf");             //0.0000005; // 5 nJ  energy required per byte fusion
    beacon_size= par("Beacon_size");//1; //1 byte
    data_size=par("data_size");     //90;
    pemp=par("pemp");               //0.0000008;
    height=par("UAV_height");
    radioDelay = par("radioDelay");
    WiFi_throughput=par("WiFi_throughput");
    dead_statuc=0;


if (ev.isGUI())
        getDisplayString().setTagArg("i2",0,"x_off");
}


void Host::handleMessage(cMessage *msg) // message handler
{
    if (strcmp(msg->getName(),"Beacon")==0)
    {
        def();
        Pa-=Prr;//nodes_energy.record(Pa);
        if(Pa<=0) dead();
        ev<<getIndex()<<"->New Energy="<<Pa<<endl;
        Pu= double(msg->par("Pu"));
        ev<<getIndex()<<"->Pu receive="<<Pu<<endl;
        double Paa=1-Pa;
        ev<<" Energy probability (1-energy)="<<Paa<<endl;
        double Pt = (double (Pu)*Paa)/((Pu*Paa)+((1-Pu)*(1-Paa)));
        ev<<getIndex()<<"Pt="<<Pt<<endl;
        Pt_all[getIndex()]=Pt;
        connected(msg);
        broadcast();
    }
   if (strcmp(msg->getName(),"CH")==0)
    {
       Pa-=Prr;//nodes_energy.record(Pa);
       if(Pa<=0) dead();
       ev<<getIndex()<<"->New Energy="<<Pa<<endl;
       Pt_all[int(msg->par("s_id"))] = double(msg->par("s_Pt"));
       if (tt>=array_length-2)
       {
           int smal=array[0];
           for(int x=0;x<array_length;x++)
           {
               if(Pt_all[array[x]]<Pt_all[smal]) smal=array[x];;
           }

           if(smal==getIndex())
           {
               cModule  *server;
               getDisplayString().setTagArg("b",0,50);
               getDisplayString().setTagArg("b",1,50);
               getDisplayString().setTagArg("b",2,"oval");
```

```
            getDisplayString().setTagArg("b",3,"blue");
            server = simulation.getModuleByPath("server");
            if (!server) error("host_1 not found");
            cPacket *pk = new cPacket("Ack_1");
            Pa-=Pr;
            if(Pa<=0) dead();
            ev<<getIndex()<<"->New Energy="<<Pa<<endl;
            double duration=beacon_size/WiFi_throughput;
            sendDirect(pk,radioDelay,duration, server->gate("in"));
        }
        else
        {
            getDisplayString().setTagArg("b",0,50);
            getDisplayString().setTagArg("b",1,50);
            getDisplayString().setTagArg("b",2,"oval");
            getDisplayString().setTagArg("b",3,"green");
            cModule  *host1;
            cPacket *pk = new cPacket("Data");
            std::string str2 = "host[";
            std::stringstream ss1;
            ss1 <<smal;
            str2 += ss1.str();
            str2+= "]";
            host1 = simulation.getModuleByPath(str2.c_str());
            if (!host1) error("host_2 not found");
            Pa-=(Pr*data_size);
            if(Pa<=0) dead();
            double duration=data_size/WiFi_throughput;
            sendDirect(pk,radioDelay,duration, host1->gate("in"));
        }
    }tt++;
  }
  if (strcmp(msg->getName(),"Data")==0)
  {
     Pa-=(Prr*data_size);//nodes_energy.record(Pa);
     if(Pa<=0) dead();
     if(pp<array_length-2) {pp++;ev<<"added"<<endl;}
     else
     {
         cModule  *server;
         int ds=array_length*data_size;// 10 Mb
         double duration= ds/WiFi_throughput; // 802.11 through put is 15 Mb/s
         server = simulation.getModuleByPath("server");
         if (!server) error("host_3 not found");
         cPacket *pk = new cPacket("Data_all");
         ev<<"i m "<<getIndex()<<"sending data "<<ds<<" to UAV"<<endl;
         pk->addPar("data")=ds;
         Pa-=(Pr*ds + pemp*ds*height); //nodes_energy.record(Pa);
         if(Pa<=0) dead();
         sendDirect(pk,radioDelay,duration, server->gate("in"));
     }

  }   delete msg;
}
```

```
void Host::connected(cMessage *msg)
{
    array_length=0; for (int i=0;i<500;i+=3) array[i]=0;
    std::string str (msg->par("t"));
    std::string str1;
    std::string::size_type sz;
    std::size_t t=0;
    t=3;//str.find("-");
    int l=abs(str.length());
    ev<<"list receive"<< str<<endl;
    for (int i=0;i<l;i+=3)
    {
         str1 = str.substr (i,t);
         array[array_length]= atoi(str1.c_str());
         t=str.find("-",t+1,str.length());
         ev<<"list to array"<<array[array_length]<<endl;
         array_length++;
    }


}
void Host::broadcast()
{
    int self_id = getIndex();
    ev<<"length for bradcast Ch="<<array_length<<endl;
    for (int i=0;i<array_length;i++)
    {
        if (array[i]==self_id && array_length==1)
        {
           getDisplayString().setTagArg("b",0,50);
           getDisplayString().setTagArg("b",1,50);
           getDisplayString().setTagArg("b",2,"oval");
           getDisplayString().setTagArg("b",3,"yellow");
            cModule  *server;
           server = simulation.getModuleByPath("server");
           if (!server) error("host_1 not found");
           cPacket *pk = new cPacket("Data_all");
           pk->addPar("data")=data_size;
           ev<<"i am the only one i am sending the data to uav"<<endl;
           Pa-=Pr;//nodes_energy.record(Pa);
           if(Pa<=0) dead();
           reset_status=1;
           double duration=data_size/WiFi_throughput;
           sendDirect(pk,radioDelay,duration,server->gate("in"));
        }
        else if (array[i]==self_id);
        else
        {
            cModule  *host1;
            cPacket *pk = new cPacket("CH");
            std::string str2 = "host[";
            std::stringstream ss1;
            ss1 <<array[i];
            str2 += ss1.str();
            str2+= "]";
```

```
            host1 = simulation.getModuleByPath(str2.c_str());
            if (!host1) error("host_4 not found");
            pk->addPar("s_id")=self_id;
            pk->addPar("s_Pt")=Pt_all[self_id];
            ev<<"sent Ch by "<<getIndex()<<" to "<<array[i]<<endl;
            Pa-=Pr;//nodes_energy.record(Pa);
            if(Pa<=0) dead();
            reset_status=1;
            double duration=beacon_size/WiFi_throughput;
            sendDirect(pk,radioDelay,duration,host1->gate("in"));
        }
    }

}
void Host::def()
{

    if(reset_status==1)
    {
        ev<<"reset for "<<getIndex()<<endl;
        reset_status=0;
        tt=pp=array_length=0;
        Pu = 0; // UAV will transmit
        for (int i=0;i<500; i++) array[i]=Pt_all[i]=0;
    }

 }
void Host::dead()
{
   ev<<"Dead node="<<getIndex()<<" dead message is sent"<<endl;
   getDisplayString().setTagArg("b",3,"black");
   cModule  * server;
   cPacket *pk = new cPacket("dead");
   server = simulation.getModuleByPath("server");
   pk->addPar("dead")=int(getIndex());
   sendDirect(pk,server->gate("in"));

}
void Host::finish()
{
    EV << "duration: " << simTime() << endl;
    if(Pa<0) Pa=0;
    nodes_energy.record(Pa);
    recordScalar("total energy consumption",Pa );
}

};
```

Appendix D

MATLAB Simulation for Clustering and Localization

We developed two different simulations under MATLAB to evaluate our proposed system for clustering and localization.

The simulation for clustering and data gathering scheme is created as algorithm given in section 3.7. Localization by virtual antenna as proposed in section 4.3 is also evaluated and provided later in this appendix.

D.1 Data Collection Simulation

Our developed system is composed of ground sensors and aerial UAV, where UAV is flying above the surface from 20 meters to 500 meters. To manage this scenario, we developed our simulation in 3D plane. Movement of UAV in the air and formation of cluster on ground has been developed and programmed. Snapshots of the developed system (Figure D.1) and source code are shown below. Each line of code is well intended and documented for better understanding.

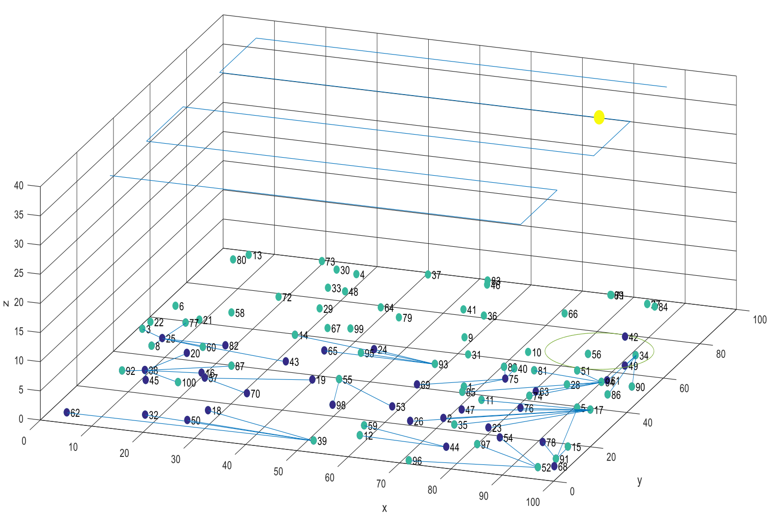


Figure D.1: MATLAB developed simulation.

```
%*****************************************************************
% MATLAB Source Code of URP Routing Protocol
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%*****************************************************************
% Mohammad Ammad uddin
% the source code can be used for learning purpose
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
close all;clear all;clc;
xm = 100; ym = 100;%Field Dimensions - x and y maximum (in meters)
n = 100;%Number of Nodes in the field
p=0.05;%Optimal Election Probability of a node to become cluster head
cov = 10; %10; % UAV coverage -- 10 meter radius on earth
height=40;
beacone=10;% every 25 meter
pkLen =6400;  %data packet length
ctrPkLen = 200;% control packet length
%Energy Model (all values in Joules)
Eo = 0.5;%Initial Energy
ETX=50*0.000000001;%Eelec=Etx=Erx
ERX=50*0.000000001;
Efs=10*0.000000000001;%Transmit Amplifier types 1
Emp=0.0013*0.000000000001;%Transmit Amplifier types 2
EDA=5*0.000000001;%Data Aggregation Energy
WP.x=[10 90 90 10 10 90 90 10 10 90];% way points x for 10 meter UAV coverage
WP.y=[10 10 30 30 50 50 70 70 90 90];% way points y for 10 meter UAV radius

rmax=3500;%maximum number of rounds
do=sqrt(Efs/Emp);
%%%%%%%%%%%%%%%%%%%%%%%%%% END OF PARAMETERS %%%%%%%%%%%%%%%%%%%%%%%%
et=0;
for i=1:1:n % all nodes
    S(i).xd=rand(1,1)*xm;%
    S(i).yd=rand(1,1)*ym;
    S(i).zd=0;
    S(i).type='N';  %initially there are no cluster heads only nodes
    S(i).E=Eo;%*(1+rand*1);
    S(i).ENERGY=0;
    et=et+S(i).E;

end
xx=[S.xd];yy=[S.yd];zz=[S.zd];
%Creation of the random Sensor Network
countCHs=0;
rcountCHs=0;
cluster=1;

packets_TO_UAV=0;
packets_TO_CH=0;

flag_first_dead=0;
distanceBroad = sqrt(cov*cov+cov*cov); % broadcast range of each node
%counter for bit transmitted to Bases Station and to Cluster Heads per round
Total_ED=0;
ED=0;
for r=0:1:rmax
```

```
r
if(r==0)
    c_bcon=beacone/2;
else
c_bcon=1; %current beacon
end
clstr=1; % currrent Cluster
m=1; %cluster member

%Number of dead nodes
dead=0;

%counter for bit transmitted to Bases Station and to Cluster Heads
figure(1);hold on;grid on;
xlabel('X');ylabel('Y');zlabel('Z');
scatter3(xx, yy, zz,50);



for i=1:1:n
    text((xx(i)+0.5), (yy(i)+0.5), num2str(i));
    %checking if there is a dead node
     if (S(i).E<=0)
       dead=dead+1;
       S(i).type='D';
     else S(i).type='N';
     end
end

%/// define path in air
for (i=1:1:length(WP.x)-1)
    lx= [WP.x(i) WP.x(i+1)];
    ly= [WP.y(i) WP.y(i+1)];
    lz= [height height];
    line(lx,ly,lz);
end

if (dead == n)
   break;
end
STATISTICS(r+1).DEAD=dead;
DEAD(r+1)=dead;
%When the first node dies
if (dead==1)
    if(flag_first_dead==0)
        first_dead=r
        flag_first_dead=1;
    end
end

countCHs=0;
cluster=1;
for (L=1:1:length(WP.x)-1)% visit all waypoints
  % number of steps between two waypoints
  nn=abs((WP.x(L)-WP.x(L+1)+WP.y(L)-WP.y(L+1)))+1;
  UAV.x=round(linspace(WP.x(L),WP.x(L+1),nn));%movement of UAV in x direction
```

```
  UAV.y=round(linspace(WP.y(L),WP.y(L+1),nn));%movement of UAV in y direction

for (i=1:1:nn)% visit next way point
  %//// move UAV //////
  h(i)=scatter3(UAV.x(i), UAV.y(i), height,150, 250, 'filled');% plot UAV
  x=(sin(0:(2*pi)/100:2*pi)*cov)+UAV.x(i);% UAV projection on ground x points
  y=(cos(0:(2*pi)/100:2*pi)*cov)+UAV.y(i);% UAV projection on ground y points
  k(i)=plot(x,y); % plot UAV projection circle on ground
     %//// discovery Phase---- activate nodes by beacon ///////
  for(j=1:1:n) % all nodes
  if (((UAV.x(i)-S(j).xd)^2+(UAV.y(i)-S(j).yd)^2)^0.5<=cov && S(j).type ==
  'N' && (S(j).E>0))% if node in the range and not already discovered and not dead
            AA(r+1)=j;
            S(j).type = 'C'; % change status to discovered
            scatter3(S(j).xd, S(j).yd, 0,50,50,'filled');% change color

            %//// node receives a beacon message /////////////
            ED=ETX*(ctrPkLen);
            S(j).E = S(j).E - ED; % energy dissipated to receive a packet from UAV
            Total_ED = Total_ED + ED;% total system energy used
            C(clstr).WP_dis(m)=((WP.x(L+1)-S(j).xd)^2+(WP.y(L+1)-
            S(j).yd)^2)^0.5; %UAV calculate distance of this node with WP
            C(clstr).id(m) = j;% add member in cluster
            m=m+1;%next member
        end
    end
            %//////Clustering Phase ////////////
    if(c_bcon>= beacone && length(C)==clstr )% if beacon is over and there
                                              %are some nodes in cluster
       mm = max(C(clstr).WP_dis);% UAV will find maximum distance node from WP
       % UAV calculate probabilities of nodes to be CH accorging to distance
       C(clstr).ProU=(1-C(clstr).WP_dis/mm);

       %// UAV will send Probability to each node to be CH
       %// All the nodes in cluster will receive
       for ii=1:1:(length(C(clstr).id))
            ED = ETX*(ctrPkLen);
            % energy dissipated to receive UAV probability
            S(C(clstr).id(ii)).E = S(C(clstr).id(ii)).E - ED;
            Total_ED = Total_ED + ED;% total system energy used
            C(clstr).ProE(ii)=S(C(clstr).id(ii)).E/0.99;
            ProE=S(C(clstr).id(ii)).E;   %probability according to Energy
            ProU=C(clstr).ProU(ii);      %probability received by UAV
            % node over all probability to become clusgter head
            C(clstr).Pro(ii)= ProE*ProU/((ProE*ProU)+(1-ProE)*(1-ProU));
            %enery used to broadcast CH probability message
            ED = (ETX*ctrPkLen + Efs*ctrPkLen*(distanceBroad));
            S(C(clstr).id(ii)).E = S(C(clstr).id(ii)).E - ED;
            Total_ED = Total_ED + ED;% total system energy used
            packets_TO_CH = packets_TO_CH +1;
         end
            % CH node  having maximum probability will announce final CH
            [val CH] = max(C(clstr).Pro);
            HEAD=C(clstr).id(CH)

            %enery used to broadcast final CH message
```

```
            ED = (ETX*ctrPkLen + Efs*ctrPkLen*(distanceBroad));
            % total system energy used
            S(C(clstr).id(ii)).E = S(C(clstr).id(ii)).E - ED;
            Total_ED = Total_ED + ED;
            packets_TO_CH = packets_TO_CH +1;

            countCHs=countCHs+1;
            for ii=1:1:(length(C(clstr).id)) % each node will link with CH
                  %link to CH x point
                  L_CH_X=[S(C(clstr).id(CH)).xd S(C(clstr).id(ii)).xd];
                  %link to CH y point
                  L_CH_Y=[S(C(clstr).id(CH)).yd S(C(clstr).id(ii)).yd];
                distance=((S(C(clstr).id(CH)).xd -
                S(C(clstr).id(ii)).xd)^2+(S(C(clstr).id(CH)).yd -
                S(C(clstr).id(ii)).yd)^2)^0.5;%distance of node and CH
                if (distance >=do)
                        ED =(ETX)*pkLen+
                        Emp*pkLen*(distance*distance*distance*distance);
                else
                    ED =(ETX)*pkLen+ Efs*pkLen*(distance*distance);
                 end
                  % energy used to send data to CH
                  S(C(clstr).id(ii)).E = S(C(clstr).id(ii)).E-ED;
                Total_ED = Total_ED + ED;
                line(L_CH_X,L_CH_Y);
            end
            distance_UAV=((S(C(clstr).id(CH)).xd -
            UAV.x(i))^2+(S(C(clstr).id(CH)).yd - UAV.x(i))^2 +(height-
            0)^2)^0.5;% distance of CH and UAV

            if (distance_UAV >=do)
            ED = (ETX+EDA)*pkLen+
            Emp*pkLen*(distance_UAV*distance_UAV*distance_UAV*distance_UAV);
            else
               ED = (ETX+EDA)*pkLen+ Efs*pkLen*(distance_UAV*distance_UAV);
            end
            S(C(clstr).id(CH)).E
            % energy used by CH to agrigate and send data to UAV
            S(C(clstr).id(CH)).E = S(C(clstr).id(CH)).E - ED;
            packets_TO_UAV = packets_TO_UAV + length(C(clstr).id);
            Total_ED = Total_ED + ED;
            clstr=clstr+1;%next cluster
            c_bcon=0;%restart beacon
            m=1;%first memeber of cluster
    end
    pause(0.025);
    delete (h(i));%move next UAV
    delete (k(i));%move next UAV circle on ground
    c_bcon=c_bcon+1;%next beacon
    %////clustering phase
end
end

C=[];
    STATISTICS(r+1).CH = countCHs;
    STATISTICS(r+1).PACKETS_TO_CH=packets_TO_CH;
```

```
    STATISTICS(r+1).PACKETS_TO_UAV=packets_TO_UAV;
    STATISTICS(r+1).T_ED = Total_ED;
    delete(figure(1));
end
for i=1:1:r;
    x(i)=i;
    y1(i) = STATISTICS(i).DEAD;
    y2(i) = STATISTICS(i).PACKETS_TO_CH;
    y3(i) =STATISTICS(i). PACKETS_TO_UAV;
    y4(i) =STATISTICS(i). T_ED;
end
plot(x,y1,'r');
title('Dead Nodes VS Rounds');
xlabel('Rounds') % x-axis label
ylabel('Dead Nodes') % y-axis label
%legend('y = sin(x)','y = cos(x)')
figure(2);
plot(x,y2,'r');
figure(3)
plot(x,y3,'r');

%write data in excel file
xlswrite('testdata.xlsx',y1',1,'F1');
xlswrite('testdata.xlsx',y2',1,'G1');
xlswrite('testdata.xlsx',y3',1,'H1');
xlswrite('testdata.xlsx',(et-y4)',1,'I1');
```

D.2 Localization Simulation

In section 4.2, we designed and developed a virtual phase array antenna system to estimate the location of GPS free, low cost and energy conscious sensor nodes. In this system, we proposed a single antenna mounted over an UAV, virtually making an array of antenna by taking snapshots after fixed intervals of time. To achieve this goal, we modified classical MUSIC algorithm and added some modules including calibration, rectification and theta adjustment. This proposed system is also evaluated in MATLAB simulation, the source code of this simulation is provided here.

```
% initiation block
clc;
clear all;
w=[pi/4 pi/4]';
P=length(w);
f=4.3e8;            %Hz radio
fADC= 2e4;          %Hz Analog To Digital converter frequency
foffset= 2e3;       %Hz Frequency Offset between Transmitter and Receiver
c=3e8;              % m/s celerity of wave
lambda=c/f;
v=7;                % speed of UAV
dt=lambda/(2*v);  % time difference between 2 snapshots (2 virtual antennas)
Rover= fix(fADC*dt); %oversampling of the virtual array= number of samples
between 2 snapshots of the virtual array
M=20;               %for 6 degree accuracy
```

```
delta_phi_offset= 2*pi /(fADC/foffset);%phase evolution between 2 samples at
fADC (due to frequency offset)

x1=3.4;           %x cooardinate -- initial position of sensor node 1
x2=-80;           % x cooardinate-- initial position of sensor node 2
y = 100;          % y cooardinate--initial position of sensor node
dt_over=1/fADC;   %time between 2 ADC samples
snr=20;
t=0;
phi_offset=0;
L=1;
for jj=1:M
   for k=1:Rover
        %/////geometrical parameters /////
        doa1=atan((x1-(v*t))/y);     % -20 degree for y=100m
        doa2=atan((x2-(v*t))/y);     %  60 degree
        D1=exp(-j*2*pi*v*dt*sin(doa1)/lambda*jj);
        D2=exp(-j*2*pi*v*dt*sin(doa2)/lambda*jj);
        dd1(L)= atand((x1-(v*t))/y);
        dd2(L)= atand((x2-(v*t))/y);L=L+1;
        %generation of temporal signal with phase offset Receiver/Transmitter
        xxx= 2*exp(j*(w))*exp(j*2*pi*foffset*t);
        xx= [D1,D2]*xxx;
        x(k,jj)= xx+awgn(xx,snr,'measured'); %noise in oversamples
        t= t+dt_over; %instantaneous time
       if(jj==1 && k~=1)
         %we calculate the phase difference on 2 successive samples
         phi= atan(imag(x(k,jj))/real(x(k,jj)))- atan(imag(x(k-
              1,jj))/real(x(k-1,jj)));
         if(k==2)
             sign_of_phi=sign(atan(imag(x(k,jj))/real(x(k,jj)))- atan
                         (imag(x(k-1,jj))/real(x(k-1,jj))));
         end

         if(sign(phi)~= sign_of_phi)
             if(sign_of_phi > 0) phi = phi + pi; else phi = phi - pi;end
         end
         pp(k)=phi;%for graph
         phi_offset = phi_offset +  phi;
       end
   end
   if(jj==1)
        %average
        phi_offset = phi_offset / (Rover-1);
   end
end
x_original=x; % for graph
%/we have to rectify the x time signal to compensate for this phase offset
x=  reshape(exp(-j*[0:phi_offset:(M*Rover-1)*phi_offset]),Rover,[]) .* x;
x_fixed=x; % for graph
R=(x'*x);
% Music on averaged covariance matrix
J=fliplr(eye(M));
R=R+J*conj(R)*J;
[N,V]=eig(R);
NN=N(:,1:M-P);
theta=-90:0.5:90;
```

```
for ii=1:length(theta)
    SS=zeros(1,length(M));
    for jj=0:M-1
        SS(1+jj)=exp(-j*2*jj*pi*v*dt*sin(theta(ii)/180*pi)/lambda);
    end
    PP= SS*NN*NN'*SS';
    Pmusic(ii)=abs(1/ PP);
end
Pmusic=10*log10(Pmusic/max(Pmusic));
% Theta adjustment
Dx=v*M*dt_over*Rover/y;
Cortheta=atand(tan(theta*pi/180)+Dx);
plot(Cortheta,Pmusic,'r')
xlabel('angle \theta/degree')
ylabel('spectrum function P(\theta) /dB')
title('DOA estimation with Phase Calibration')
grid on
```

Appendix E

Proof of Concept Design and Development

The developed system is also evaluated by creating proof of concept required to explain the system to different companies for implementation. All devices are made by using Arduino microcontroller is available in different sizes such as Uno, Mini, Micro, etc. We used Uno and mini to develop or proof of concept. Uno is used to develop sink node, while all sensor nodes are made by using mini, to make their size small.

Arduino Uno is a standard size kit having 5 analog pins to read analog values and 13 general purpose digital inputs. It also provides both +3 and +5 outputs to run other components. This board can be operated by computer output using standard USB cable and stand alone with 9 V battery. Figure E.1 is showing its pin detail.

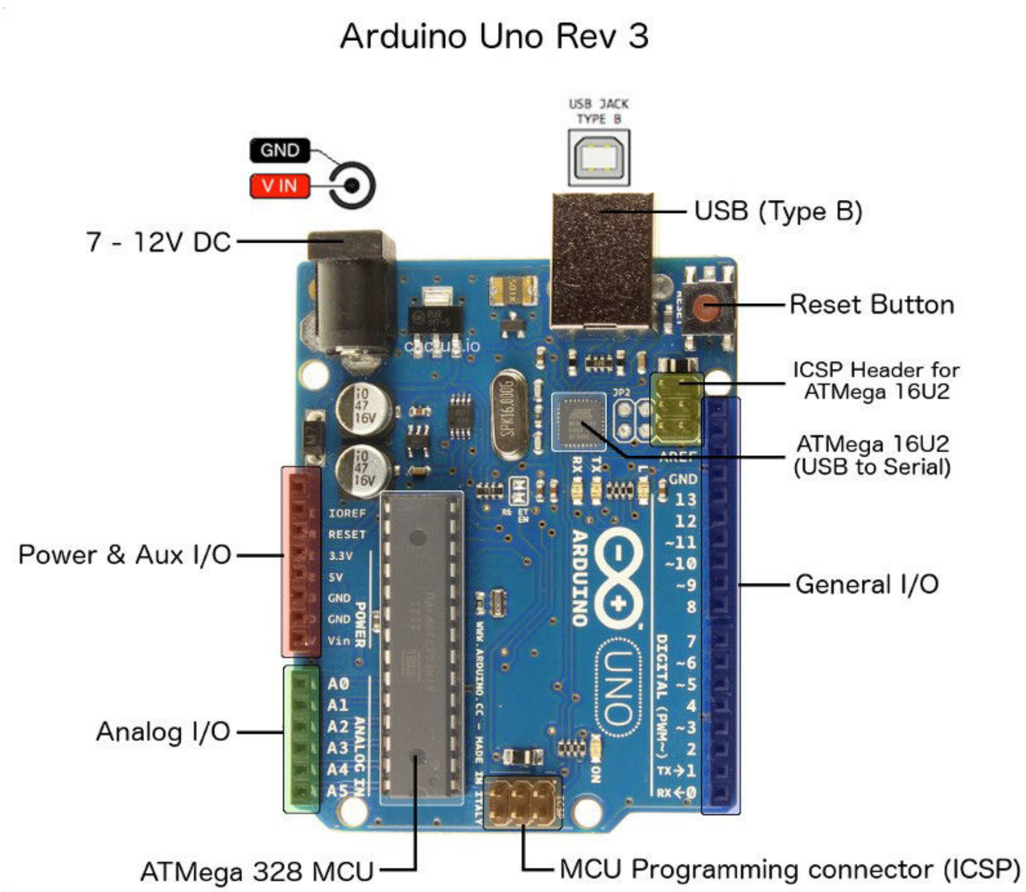


Figure E.1: Arduino Uno.

we used nRF24L01 transceiver (Figure E.2) to communicate between UAV and sensor nodes. The specifications are as under:

- On board support for power amplifier and low noise amplifier.
- Can communicate and configure over a 4-pin serial peripheral interface.
- Configuration registers are accessible through the SPI connection.
- Configurable parameters include frequency channel (125 selectable channels),
- Data rate 250kbps, 1Mbps and 2Mbps.
- Output power on chip voltage regulator accepts supply voltages from 1.9 to 3.6V.
- Radio uses Gaussian Frequency-Shift Keying (GFSK) modulation as well as fast Automatic Gain Control (AGC).
- Includes an Interrupt Request (IRQ) pin which can be used to wake the host microcontroller from sleep.
- Auto-acknowledge and auto-retransmit function.

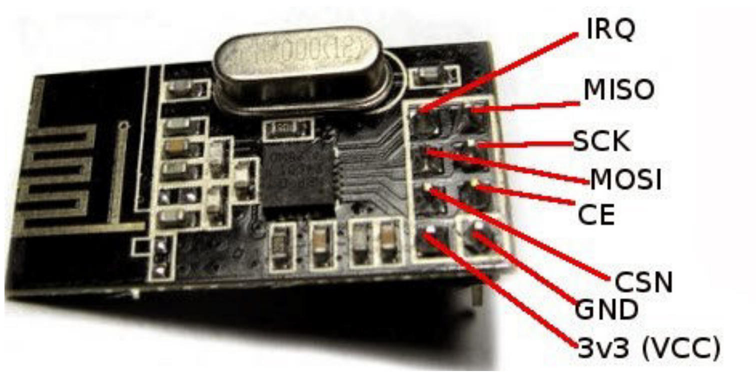


Figure E.2: nRF24L01 Transceiver

E.1 First Hands on Example with Arduino

Here, we are giving step by step procedure to conduct a first hand on experiment with Arduino to send some information wirelessly using nRF24L01 transceiver.

1. Install Arduino driver and IDE application for the operating system used for programming.
2. Connect two nRF24L01 transceiver with two Arduino as per wiring diagram given in Figure E.3. Label one board as server and one as client, but both will have same wiring scheme.
3. As nRF24L01 is a transceiver, we are creating a two-way communication example. We will send some information from client to server and will get acknowledgement message from server.
4. Connect Arduinos circuit labeled server with computer through USB cable.

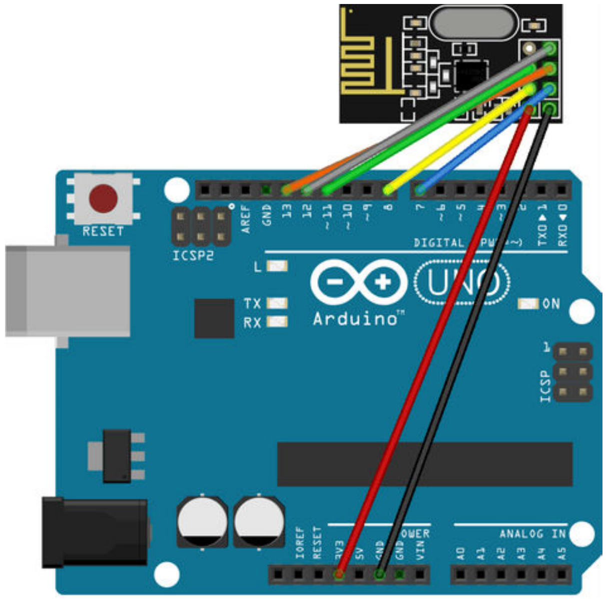


Figure E.3: Arduino nRF24L01 wiring connection details.

5. Write following code in Arduino IDE compile it, if no error, then send it to Arduino chip.

```
// nrf24_client
#include <SPI.h>
#include <RH_NRF24.h>
RH_NRF24 nrf24; // Singleton instance of the radio driver
void setup()
{
  Serial.begin(9600);
  if (!nrf24.init())    Serial.println("init failed");
  if (!nrf24.setChannel(1))    Serial.println("setChannel failed");
  if(!nrf24.setRF(RH_NRF24::DataRate2Mbps,RH_NRF24::TransmitPower0dBm))
    Serial.println("setRF failed");
 }


void loop()
{
  Serial.println("Sending to nrf24_server");// Send a message to nrf24_server
  uint8_t data[] = "Hello World!";
  nrf24.send(data, sizeof(data));
  nrf24.waitPacketSent();// Now wait for a reply
  uint8_t buf[RH_NRF24_MAX_MESSAGE_LEN];
  uint8_t len = sizeof(buf);
  if (nrf24.waitAvailableTimeout(500))
  {
    if (nrf24.recv(buf, &len)) // Should be a reply message for us now
    {
      Serial.print("got reply: ");
      Serial.println((char*)buf);
    }
```

```
    else      Serial.println("recv failed");   }
  else    Serial.println("No reply, is nrf24_server running?");
  delay(400);
  }
}
```

Arduino program is mostly containing two functions: setup and loop. In setup function, we initialized serial to see the out puts. Then we initialized the RF channel. The loop is the function, which continuously working until board has power. This loop function is written to send Hello packet to server wait for 500 ms and get reply. If client gets reply, it will generate a message to serial interface showing transmission is successful and after 400 ms it will transmit again. If clients do not get reply then it will generate an error message accordingly.

6. Now connect server circuit with computer USB and write code as given below:

```
#include <SPI.h>
#include <RH_NRF24.h>
RH_NRF24 nrf24; // Singleton instance of the radio driver
void setup()
{
  Serial.begin(9600);
   if (!nrf24.init())
  if (!nrf24.setChannel(1))   Serial.println("setChannel failed");
  if (!nrf24.setRF(RH_NRF24::DataRate2Mbps, RH_NRF24::TransmitPower0dBm))
    Serial.println("setRF failed");
}
void loop()
{
  if (nrf24.available())    // Should be a message for us now
  {
    uint8_t buf[RH_NRF24_MAX_MESSAGE_LEN];
    uint8_t len = sizeof(buf);
    if (nrf24.recv(buf, &len))
    {
      Serial.print("got request: ");
      Serial.println((char*)buf);
      ///// Send a reply  ////////
      uint8_t data[] = "And hello back to you";
      nrf24.send(data, sizeof(data));
      nrf24.waitPacketSent();
      Serial.println("Sent a reply");
     }
    else   Serial.println("recv failed");
  }
}
```

Server code is almost same as client, it receives a message first then prepares a reply and sends back to client.

We extended this simple example and wrote a program to implement designed routing and data collection algorithm. Development of proof of concept and achieved results already have been discussed in section 3.11.

Appendix F

Agriculture Supporting WSN Material

Different prototypes we studded to developed proposed system, Table F-1, contains some basic information about various prototypes proposed, developed and implemented in different areas of the world. While, Table F-2 prototypes other than agriculture. Table F-3, contains information regarding some of the leading sensor manufactures and provides a glimpse regarding their components and products. Use of wireless sensors to monitor these parameters becoming common hence a large range of sensors are available and few are listed in Table F-4. Specification of some famous agriculture sensors are available in Table F-5. More detail is given in our research paper [34].

TABLE F-1: PROTOTYPES FOR AGRICULTURE WSN AND APPLICATIONS

Prototype/Test-bed	Monitoring Parameters	Scale and Density	Data Amounts/ Frequency	Year	Country
Agrisensor [135]	Soil and air temp., Soil Moisture, Air humidity	Small number of nodes (5-8) in a Small-Plot	Every 15 mint, Frequent intervals, Over 8 days	2011	Czech Republic
Root Zone Sensors for Irrigation [136]	Irrigation, Moisture, Water Salinity	6 Number of nodes and 3 Repeaters	Depends on irrigation interval, 5 Months duration	2008	Italy
Reactive Soil Moisture Network [137]	Rain storms, Soil Moisture,	11 Sensors of different types, One hectare area, GSM Gateway	Frequent when raining (every 10 mints), Once a day without rain.	2005	Australia
Smart Irrigation System [138]	Irrigation, Moisture	2-Sensor motes, 1 EC-5 Soil humidity sensors, Tiny OS	Small amount of data. After every 4.40 hours, Continue for 2 days	2011	Greece
Sensors for Vineyard Monitoring [139]	Temperature, Frost damage, Grape variety, Slop of surrounding	Dense and deep deployed as 65 Nodes deployed in two acres, Maximum 8 hops	Every 5 mints, Frequent intervals, Deployed for 6 months but results shown for 1 month.	2003	USA
Web Based Precision Farming [140]	Weather and solar related parameters	Spars as nodes deployed upto 180 meters	Frequent sampling as after 6 mints	2014	Germany
Wireless Sensor for Greenhouse Parameter [141]	Inside and outside Temperature, Humidity, Light, CO2	Densely deployed as 40-50 sensors for 70*150 meter area	Small amount of data. Mostly infrequent as event based	2010	India
Precision Agriculture using WSN [142]	Soil moisture and condition	Laboratory based experiments only but not in field	Frequent reading, total 200 packets where each is 30 byte	2011	USA
Agro-Sense [143]	Humidity, Soil moisture and Conductivity	Sparse, only four nodes deployed in 200 meters.	Frequent, after every 3 hours for 3 days.	2008	India
Greenhouse Monitoring using WSN [144]	Temperature, light. Irradiance, Carbon dioxide	Lab setup, only 4 nodes in 18*80 meter area	Sleep and wakeup based periodic data gathering.	2008	Finland
APTEEN [145]	Light intensity, pH value, Soil moisture, Temp.	Densely, different number of nodes for different parameters.	Large data amounts, Monitoring time varies from half day to six weeks.	2013	Egypt
VineSense [146]	Temperature, Soil moisture, Humidity	Variable density, 255 nodes required, 50 nodes results are shown	Frequent but vary for different parameters, overall sense after each 10 minutes	2011	Italy

TABLE F-2: WSN PROTOTYPES FOR APPLICATIONS OTHER THAN AGRICULTURE

Applications	Prototype/Test-bed	Monitoring Parameters	Scale and Density	Data Amounts/ Frequency	Achievement	Country
Healthcare	Three Layer WBSN [146]	Different physical activities in elderly	Few Shimmer Nodes on Body, Smart Phone as Base Station	Minimal, Periodically Data Transmission	Wearable sensors are connected with smartphone instead of computer to provide the mobility.	Germany
	Smart Healthcare Clothing [147]	Body temperature, Heartbeat, Respiration,	Inner layer: Sensors and RFID tags, Outer layer: Cloths	Small amount of data, React only during emergency cases	Health condition of wearers is diagnosed and results are transferred to the computer to analyze them in real time.	South Korea
	Smart-Jacket for neonatal monitoring [148]	Infant ECG, Respiration, and Blood oxygen saturation	ECG amplifier, Communication module, A base station	Continuous monitoring for 1 hour, Total 6 data channels were possible with 3 nodes.	Wearable jacket enables ECG measurement simply by textile electrodes. Considering future, it is expandable to adapt new technologies.	Netherlands
Marine Environment	SEMAT [149]	Oceanographic features like Salinity, Temp, Light	Few sensor nodes, 3 Surface buoys, Base station	2 Bytes/Reading, 96 Readings/Day, Can continue for many months	Suggested for short-term deployments, pre-commercial staged system for shallow but calm water monitoring.	Australia
	Ocean-TUNE UCONN Testbed [150]	Acoustic Channel Behavior, Conductivity	Sensors are sparsely deployed, 3 Surface buoys	Frequently, Mostly with 15 mints sampling, Stayed for 7 Months.	A testbed for underwater monitoring. Can be technology incubator for comm. and networking, promote research prototypes	USA
	Monitoring Marin Env. at Mar Menor [151]	Temperature, Turbidity, O_2 , Salinity, Chlorophyll and Nitrates	Several sensor nodes and buoys depends on monitoring parameters	Frequent sampling as after 10 - 20 mints, Continue for 1 month.	Monitor coastal shallow environment. Used to get hydrodynamic performances of the lagoon and other oceanographic parameters.	Spain
Safety and Security	SASA [152]	Structure variations, Detect collapsed locations	27 Mica-2 Motes. Large number of sensors nodes	Large amount of data Packets are generated, Frequent transmission	Provide underground monitoring of coal mines. It detects structural variations due to underground collapse.	Hong Kong
	Gas Leakage Detection [153]	Monitoring of combustible gases, Gas leakage detection	9 nodes with catalytic sensor, STM32F102C6 microcontroller as coordinator	Frequent updates. Large amount of data packets but small in size.	Monitor real boiler for safety critical environment. Analyses of the catalytic sensor response under various conditions were performed.	Italy
	Quake Detection using Smartphones [154]	Early warning in disasters like earthquake& disruptive event	Inertial sensor as an accelero-graph, 40 Smartphones	2 months monitoring of mobile & repose position. Results stored in phone app.	A 3-layered architecture to satisfy the objectives of an early warning and help people to make timely decisions while using personal phones for this purpose.	Spain
Environment Monitoring/ Forecasting	Long-Term Environmental Monitoring [155]	Temperature, Humidity, Smoke Sampling	50 Sensors, 2 Gateways & a Server, Densely deployed as each node at 20 meters	Frequent messages as 50 within 15 mints. Monitoring continue for 2 and half months.	Long term wildfire monitoring, low power consumption as single 1 Ah lithium battery can work for 16 years. Faulty nodes can be detected easily.	Greece
	SenseWeather [156]	Rainfall, Wind speed, Wind direction, and Soil moisture	Multiple sensors and boards from Libelium are used. Cellular SIM cards used for SMS	Usually frequent on hourly bases. Small size data messages transferred.	Help to integrate weather readings of multiple parameters those received from parse network of weather stations. Calibration is done for accuracy purpose.	Kenya
	WSN Air Pollution Monitoring	Air Quality Index (AQI)	Large number of nodes are required as 50 to	Maximum within 20 mints data can be	Air Quality Index (AQI) is used to monitor air quality. Real time information about	Mauritius

	System (WAPM) [157]		200 nodes, depending on area size.	collected. Data collected only whenever required.	air pollution and issue alerts if drastic change occur in air quality.	
Transportation	Traffic Monitoring System Virginia [158]	Number of axles speed, Distance between Axles.	Several sensors as 2 for wheel path, 4 to collect stress, 5 for traces of horizontal strain	Periodic testing for one week to three months	An intelligent transport system to count traffic volume, tracking speed according to classification (light or heavy), and monitor pavement deformation.	USA
	Street Parking using WSN (SPS) [159]	Shadow and Object detection	82 Sensor nodes for 720 parking slots.	Frequent data forwarding, Sampling period of 8 days. Low data rate	Help to find a suitable parking with 98% accuracy. Energy efficient as sensor alive for 5 years with two 2500mAh batteries.	China
	ParkNet [160]	Collect parking space occupancy information	Three cars, each equipped with an ultrasonic sensor	Sound waves emitted after every 50 ms. Experiments continue for 2 months.	Provide road-side parking availability information with more than 90% accuracy. Small number of sensors are required.	USA
Livestock	Mad-Cow Monitoring a Tracking [161]	Monitor cow behavior, Track beef distribution.	Dozens of sensor, Data sent to gateways and base station	Generate Alerts only when required	Monitoring and tracking mad-cows from its birth to slaughtering house. Further, monitor beef process till the grocery shelf	USA
	Monitoring Wildlife Passages [162]	Safe animal crossing, Target identification	Variable node density in 40 m of radius (Detector, Cameras)	Whenever target detected, Real time action is not required	Track animals through wildlife passages under roads. Analyze their reactions according to different locations.	Spain
	Livestock Management system [163]	Pulse rate, Heart rate, Blood oxygen levels	Different number of sensors deployed on cattle body	Whenever alerts require, Cow escaping or Early sign of disease	Reliable cattle monitoring system help to understand their health and natural pattern New sensors can be integrated easily.	USA

TABLE F-3: LEADING SENSOR MANUFACTURERS AND PARAMETERS

Applications	Manufacturer	Components	Sensing Parameters/Applications	Reference
Agriculture	SensaTrack	Sensor, Adaptors, Gateways	Saving Water, Soil Moisture, Temperature, Humidity, light	[147]
	PYCNO	Sensors	Humidity, Temperature, Soil Moisture, Pesticides	[148]
	Stevens	Sensor, Data Loggers	Irrigation, Golf Courses/Sports Turf	[149]
	SOLCHIP	Sensors, RFIDs	Precision agriculture, Environmental monitoring, Traceability systems (RFID)	[150]
	Landscape Technologies	Sensors, Data Loggers, Wire Systems	Soil Moisture, Irrigation, Precision Temp.	[151]
	IRROMETER	Sensors, Lysimeters, Data Loggers	Irrigation, Landscape	[152]
	ICT International	Sensors, Meters, Probes, Gauges	Horticulture, Irrigation, Plant Physiology	[153]
Marine Environment	CAMPBELL SCIENTIFIC	Data Loggers, Sensors, Communication devices	Oceanography, Fisheries, Flood alert, Water level and flow	[154]
	Satlantic	Sensors, Radiometers, Observing systems	Depth, Density, Salinity, pH, Fluorescence, UV Absorbance	[155]
	Sea Bird Elec.	Sensors, buoys, AUVs, ROVs, Underway Systems	Water samples, Wave, Tide, pH and Oxygen sensors, Dredging, Oil Spills	[156]
	KONGSBERG	Cameras, Sensors, Info and Safety, Simulators	Navigation, Positioning, Underwater imaging and CCTV, Deck Control	[157]
Healthcare	CMOSIS	Imaging Sensors, Area scan sensors	Endoscopy, Dental X-ray, Mammography, FFOCT	[158]
	SOL-CHIP	Tracking Sensors, Medical	Tracking elders and children, Monitoring,	[150]

		wristband and bracelets	Hearing aids	
	MICRO-EPSILON	Measurement, Sampling and positioning tools	Dental samples, Tablet color, Surgical equipment position.	[159]
Environment Monitoring/ Forecasting	ACURITE	Weather sensor and station, Rain gauges	Environment, Weather forecasting, Rain, Humidity, Lunar cycle	[160]
	S+S REGELTECHNIK	Sensor, Meters, Transducers	Air quality and flow, Light intensity and motion, Temperature, Humidity	[161]
	Stevens	Weather stations, Gauges, Wind sensors	Rain gauges, Wind direction and speed, Solar radiation, Evaporation	[149]
	[154] SCIENTIFIC	Sensor, Data loggers, Towers and mounts	Lightening and disaster warning, Air quality, Micrometeorology.	[162]
Transportation	Custom Sensors & Techn. (CST)	Sensors, Pneumatics, Motors & Actuators	Automation, Cockpit control, Protection, Position and pressure sensing.	[162]
	Silicon Labs	Sensors, Microcontroller, Isolators, Interfaces	Anti-Pinch window, Tire pressure monitor, Entry security, HMI.	[163]
	Libelium	Sensors, Gateways, IoT Kits, Cloud Platforms	Smart Parking, Traffic Congestion, Smart Lighting and Roads	[164]
Gas and Petroleum	RigStat	Sensors, Cameras, GPS, Horns, Obstruction Lights	Rig and Met-Ocean monitoring, Inspection emergency operations, Tracking	[165]
	Gems Sensors & Controls	Sensor, Level-Indicators, Switches, Gauges. Valves	Flow and presser sensing, Level measuring, Boiler control, Indications	[166]
	Stellar Technology Crp.	Sensors, Transducers, AUVs, ROVs, Transmitters,	Drilling, Pressure, Leak and choke detection, Pump-Off control	[167]
Safety and Security	PURELINK	Sensors, Badges, Tags, Severs, Software	Anti-Aggression, Visit control, Zone access, Emergency, Safety, Prevention	[168]
	South-West Microwave	Sensor, Alarms, Detection Systems,	Fence and buried cable detection, Protect VIPs, Alarm control and monitoring	[169]
	NTM Sensors	Hydrogen Sensors, Alarms, Environment Monitoring	Hydrogen concentration, Safety alarms, Leak detections, Control ventilation	[170]
Live Stock	SOL-CHIP	Sensors, RFIDs, Tracking systems	Animal behavior, Movement, Health, Long range tracking. Fencing.	[150]
	Libelium	Sensors, Gateways, IoT Kits, Cloud Platforms	Animal Tracking, Toxic Gas Levels, Hydroponics	[164]
Home Automation	Smart Home	Sensors, Cameras, Switches, Technology bridges	Leakage, Motion, Smoke detection, Open/Close sensing, Security system	[171]
	SOL-CHIP	Sensors, Meters, Switches, Alarms	Remote home control, Smoke detection, Motion, Gas leakage	[150]

TABLE F-4: FAMOUS AGRICULTURE SENSORSRENT AND PROTOTYPES

Sensor/ Manufacturer	Target			Considered Purpose/Parameters							Ref.
	Weather	Soil	Plant	Water	Temp.	Moisture	Salinity	Humidity	Wind	Pollution /Co2	
PYCNO	✓	✓		✓	✓	✓		✓			[148]
MP406		✓			✓	✓					[153]
Sol Chip Com (SCC)	✓			✓						✓	[150]
SenseH2TM			✓						✓	✓	[170]
Met Station One	✓							✓	✓		[172]
OBS-3A	✓			✓	✓		✓				[154]
PASPORT Salinity Sensor		✓					✓				[173]
CI-340			✓			✓				✓	[174]
Wind Sentry 03002	✓								✓		[154]

Leaf-Wetness Sensor (LWS-L)			✓			✓					[154]
Si-7015	✓				✓			✓			[163]
AQM-65	✓									✓	[175]
SAL-BTA	✓						✓				[176]
POGO Portable		✓		✓	✓	✓					[149]
TPS-2 Portable Photosynthesis			✓		✓	✓				✓	[177]
EC-20		✓		✓		✓					[48]

TABLE F-5: MONITORING CROP PARAMETERS AND SUPPORTING SENSORS

Monitoring Parameter and Unit	Sensor	Supported Range	Accuracy	Power Supply	Product Reference
Photosynthesis (ppm)	CI-340	0 to 2000 ppm	< ± 2%	7.2 VDC	[153]
	CI-110 CI-202 CI-710	0-200,000 ppm	2%	NA	[174]
	PAR Sensor	0 to 2000 µmol	±5%	NA	[176]
	S-LIA-M003	0 to 2500 µmol	±5 µmol	0-5 VDC	[153]
Irrigation (centi bars)	Irrometer-SR	0-100 cb	± 3-2-3 %	NA	[152]
Soil Moisture (VSW %)	MP406	0-100 VSW%	± 5 VSW%	9-18 VDC	[153]
	Hydra Probe II	1 to 80	± 1.5%	30 mA active	[178]
Temperature (oC)	T/H Sensor	-50° to 140° F	± 1°F	NA	[179]
	pH100	-10 to +120°C	±0.3°C	30 VDC	[179]
Salinity	PS-2195	1 to 55 ppt	±1%	NA	[179]
	SAL-BTA	0 to 50 ppt	±1%	5 VDC	[176]
Humidity (RH)	T/H Sensor	0 to 100% RH	± 3%	NA	[176]
	HUM-M2	0 ~ 100 % RH	< 3% RH	4.5 ~ 5.5 V	[180]
	HMT330	0 to 100 %RH	±1 %RH	10 to 35 VDC	[181]
	WT Sensor	0 - 100%RH	±2%RH	5V USB Cable	[180]
Wind	WMT52	0 to 60 m/s	±3%	5 to 32 VDC	[181]
	OMC-160	0.3 to 75 m/s	2% FRO	8 TO 30 VDC	[182]
Fruit Size	FI-LM	30-160 mm		10 to 30 Vdc	[182]
Stem or trunk Size	SD-5M	4 to 70 mm		10 to 30 Vdc	[182]
Leaf Temperature	LT-2M	5-50 °C	± 0.2 °C	10 to 30 Vdc	[182]
Sap flow relative Rate	SFM1	-100 to +100 cm/hr	0.5 cm/ hr	5 Vdc	[153]
Root Image	CI-600	21.6 × 19.6 cm	Up to 600 DPI	5 Vdc	[174]

Appendix G

Evaluation and Approval of Project by AAAS and KACST

AAAS
ADVANCING SCIENCE, SERVING SOCIETY

KACST Proposal Review Spring 2015

Project Title:	Wireless Sensor Networks for Water and Crop Health Management
Principal Investigator:	Muhammad Ayaz Arshad
Institution:	Tabuk University
Technology Sector:	Agriculture Strategic Priorities
Project ID:	15-AGR5298-48

1. **The quality of the proposed research. Is the proposed research innovative? Can it contribute significantly to the state of knowledge in its field? How does the proposed research compare**

Is the proposed research innovative? The proposed research is somewhat innovative in that the investigators will seek a means to remotely acquire measured plant growth parameters and climatic data using wireless sensor networks (WSN). It should be noted however that measuring and remotely acquiring climatic conditions is not as novel as climatic data have been remotely monitored for many years. However, this project propose to use a novel application of “state of the art” technology for remote irrigated field plots. The pest sensor (bug detector) noted as having been developed by the investigators is innovative, and this could make a contribution to science. The overall project has some merit, but it lacks specificity on the use of many of the proposed sensors, including water and water management which is a key point throughout the proposal. This suggests that the investigators do not understand application detail use of many of these sensors.

Can it contribute significantly to the state of knowledge in its field? The proposed research has the potential to make a contribution to the current knowledge of monitoring crop growth and pest management but not as proposed. The pest monitoring aspect of the research apparently is already underway and if further developed it could make a contribution. However, there is limited innovation and originality in many other aspects of the proposed approach, especially given that no field implementation or proof-of-concept for the Water and Crop Health Management (WCHM) system is described. The main potential outcome of this research is pulling together numerous existing sensors to form a cluster of remote data collection and develop a system of remotely acquiring data with minimum effort from a cluster of instruments for input into a system for farm management decision making. What is lacking is an apparent understanding by these investigators of actual farming practices.

How does the proposed research compare to established programs? Existing research on sensor clustering for data collection by unmanned remote aircrafts is limited. Thus, the data acquisition cluster approach for collecting data related to water conservation could aid established, and other proposed research on water conservation. Since water needs will only get worse as we face warmer climate conditions, with global climate change, this proposed monitoring system will help in water management. However, it should be noted that a number of other researchers have reported the potential benefits of wireless sensor network systems in agriculture and precision agriculture, but these do not use the cluster and Unmanned Aerial Vehicles (UAV) approach as proposed here.

Score (5.0 possible points)

3.0

2. **The impact of the proposed program. How well does the activity advance discovery and understanding while promoting teaching, training, and learning? To what extent will it enhance the infrastructure for research and education, such as facilities, instrumentation, networks, and partnerships? What are the potential impacts on the institution? Will the**

results be disseminated broadly to enhance scientific and technological understanding? What may be the benefits of the proposed activity to society?

How well does the activity advance discovery and understanding while promoting teaching, training, and learning? The proposed research will provide training for one part-time MS graduate student. There are two consultants, listed for 14 days each. Thus, training is at a minimum given the potential for significant education and training. The specific objectives of the proposed project include: (a) integrating research with education through the use of WCHM, (b) developing training programs, and (c) convening workshops involving local and international scholars. However, there is no description of how these education and outreach activities will be carried out and how the training would proceed. While it is clear that effective education and public awareness programs are needed to build capacity for adoption of precision agriculture, the proposed research offers little potential for successfully meeting this need especially given the poorly defined education and public outreach plan. Furthermore, the potential for realizing significant technology transfer, given the outcomes and deliverables of the project appears to be low.

To what extent will it enhance the infrastructure for research and education, such as facilities, instrumentation, networks, and partnerships? The proposed research will add to what is currently being done by way of remote sensor and pest observation research. However, the other sensors noted in the proposal, such as water, temperature, rain, wind speed etc., are not clearly defined as being part of an existing collection of equipment. It appears these will be a part of the proposed research to determine what existing sensors can be used. There is not sufficient networking. This research provides an opportunity for the researchers to partner with someone with expertise in agronomic science, and this is not noted in the proposal but should be. The impact on the institutional infrastructure at the University of Tabuk is relatively low given the research tasks described in the proposal, but the project holds some potential as some new equipment would be purchased that could be used in education and training programs.

Will the results be disseminated broadly to enhance scientific and technological understanding? Limited transfer of this technology is proposed. The researchers have proposed to organize training workshops, provide demonstrations at public forums: local, regional, and international conferences. This involves workshops and demonstrations of the technology as it is developed. So, the information is proposed to be disseminated to the broader agricultural community, but more detail is needed as to how this will be done.

What may be the benefits of the proposed activity to society? The proposed research has the potential to add significantly to water use reduction and reduced pesticide use if the end results are to be used to minimize pesticide use. That is if the research results are to be used for integrated pest management and water management. The potential societal benefits are improved agricultural production given limited resources such as water availability and a lower environmental impact of farming. However, in terms of the broader societal impacts, it is unclear that, if all the stated objectives were successfully realized, the benefits to society would be that significant.

Score (5.0 possible points)

3.0

3. **The path to success. Are the costs reasonable and the time and resources dedicated appropriate to the program? Is the management plan appropriate? Are there any issues that might prevent the successful execution of the program?**

Are the costs reasonable and the time and resources dedicated appropriate to the program? It is difficult for to determine if the cost are reasonable as it is not clear as to how many and the specific type of sensors that will be used for in-field monitoring. The type of equipment that has been listed could be very costly if a significant number of these are to be installed. The investigators are requesting a total of 1,839,330 in Saudi Riyals (SAR) or approximately $490,488 US dollars over a 24-month period to cover the primary tasks outlined in the proposal. Of this amount, about 650,000 SAR (or $173,333 US) is requested for equipment, while 697,000 SAR (or $185,867 US) is requested for salaries and wages of the project personnel. Considering the scientific scope, the general nature of the tasks described in the proposal, the expected deliverables, and the potential impacts of the project on training, education, and the broader agricultural sector, the level of funding requested under each budget category appear to be unreasonable. There is no detailed budget justification provided in the proposal. Furthermore, there is no information to justify the 112,000 SAR for the two Consultants whose roles in the project are unclear or undefined.

Is the management plan appropriate? The investigators appear to have given some thought to the management plan. The PI's appear to have extensive experience in the field of sensors and have already generated hardware and technology for the project. This is critical for potential success, as it appears that system implementation could occur relatively quickly. It is also important that the PI's will access system performance as part of the project. The PI's do not just assume that the system will provide benefits. On the other hand, the system could have limitations for use only with tree crops such as dates or fruit because of the need for plant-based sensors which might be difficult to utilize with smaller herbaceous plants. The adaptability of the system for herbaceous plants such as vegetables or wheat was unclear from the proposal. While there has clearly been some planning, there is not a clearly defined project management and organization plan that describes how the expertise of the PI, co-PIs and Consultants will be brought to bear on the project. For example, each of the two Consultants participating in the project will devote an equivalent of 14 days to the project; yet, their expected activities and accomplishments are not clearly stated. There is also no detailed and elaborated work plan for each of the six tasks defined in the proposal. The success and outcomes of the proposed project clearly hinge on a well-defined, logical, and technically feasible research approach, including the design of the sensor monitoring system and architecture, sensor deployment and routing strategy, and the types of environmental variables that would be monitored at the farm/field level. Although WSN applications in precision agriculture is widely recognized as an excitingly new area of research that greatly improves agricultural productivity, the proposed project will contribute little to new knowledge and scientific insights.

Are there any issues that might prevent the successful execution of the program? The greatest issue is reducing this to a manageable project and the need for agronomic expertise. That is the potential number of sensor to be installed could be daunting. Another potential problem is not conducting ground truth of installed sensors to know if data being collected is reasonable and if sensors are functioning properly. Another concern is will farmers adopt this technology and work with the researchers? However, it should be noted that farmers in other countries are already using UAV technology. As noted, one other primary area of weakness that could prevent the successful execution of the program is the lack of expertise of the project team in the field of production agriculture (an agronomist or production based soil scientist) and specifically precision agriculture. Knowledge of effective production agriculture practices is needed to direct not only the implementation of the WSN, but to guide the farm/field implementation of the system, including data collection and analysis.

Score (5.0 possible points)

3.0

. **Alignment with NSTIP. Does the proposed research topic align with the goals of Saudi Arabia's Strategic Technologies Program, established by the National Science, Technology and Innovation Plan (NSTIP)? Please refer to the appropriate strategic technologies program to answer this question, and include comments in the space provided below.**

yes

Does the proposed research topic align with the goals of Saudi Arabia's Strategic Technologies Program, established by the National Science, Technology and Innovation Plan (NSTIP)? Given the importance of water to the Saudi Arabia and the proposed researcher is focused on water, this proposal is within the scope of the NSTIP. One could make a stretch to say that if integrated pest management are followed there could be an economic and environmental benefit derived from this proposed research as well. That is the development of competitive and environmentally safe agricultural technologies to improve sustainable growth and biosafety is an important potential contribution. Another positive is development of modern technologies to promote competition in global markets under conditions of limited resources (ex. water).

As stated in the NSTP, the objectives of the Agricultural Strategic Priorities are to build a knowledge-based economy through the support of scientific research and innovation activities, the strengthening creativity, and the development of the country's human resource capabilities in areas related to science, technology and engineering. This project is intended to meet these objectives by focusing on the strategic area of production agricultural technology and an emphasis on precision agriculture. However, from the structure of the project and the six tasks identified by the PIs, it appears that the emphasis is on wireless sensor network design rather than on the demonstration of its application to agriculture. The entire research methodology focuses on details of the sensor network and no description of implementation in the TADCO farm is provided, which could align it with the goals of the Strategic Technology Program defined in the NSTP. Given that the objective is to collect data and information relative to the environment and crop health to help farmers make informed decision, one would expected that the major component of the research plan should focus on farm-level description of the WCHM system.

Please rate the proposal against the criteria:
Please use the following scale from one to five:

5.0 – Very impressive against this criterion, virtually flawless
4.0 – High quality proposal against this criterion, no serious concerns
3.0 – Adequate against this criterion, some doubts or concerns
2.0 – Some deficiencies against this criterion, needs improvement
1.0 – Serious flaws against this criterion

Criteria	**Maximum Score**	**Total Score**
1. Quality of proposed research	5.0	3.0
2. Impact of proposed program	5.0	3.0
3. Path to success	5.0	3.0
Total Score	15.0	9

Overall, how would you rate this proposal for funding?

Recommended

Please give your recommendations as to how this proposal might be improved:
Please indicate if these could be implemented in the course of making an award, and if so, whether your recommendation for funding would change were these to be implemented.

There seems to be great concern about sensor battery drain and nonfunctional sensors. This seems to be of sufficient concern by the investigators that it might be one of the areas of failure of the proposed research. A suggestion to solve this problem would be not to install more sensors than the researchers can maintain with timely battery and failed sensor replacement during the life of the project. This will require some advanced planning as to specific parameters to measure for optimal/quality data collection. This also requires one to know what the data will be used for before collecting it. This does not appear to be a well-planned proposal in this regard, as there are many UAVs to be used and the data collection is not very specific.

The process of making decisions based on government policies and other fuzzy data collection for farmers to take action on is not clear. Following are some questions for the investigators to consider: What action will farmers be expected to take? Have farmers agreed to participate in this project? If so, this should be noted in the proposal. How realistic are these Actuator Notes? How realistic is it to use fans/heater/desert cooler to control temperature at the field scale?

There is an opportunity for more student training than what has been proposed. Students could be trained on sensor installation and maintenance, as well as calibration, and data proofing, that is assessing if data values are reasonable. This could include other MS and/or undergraduate students. However, as noted on the last page of the proposal, it appears that there is some difficulty in finding a student for this project. There is also an opportunity for partnering with agronomists and/or soil

While this proposal has some potential, it is not as well defined as it could be. If the proposal was better defined as to the number of and specific sensors to be installed it would be much better. In addition, if student training, technology transfer, farmer involvement, agronomic scientists, and ground truth aspects of the proposed research were to be improved this would increase the strength of the proposal. Another point to consider is the adaptability of this technology for use with herbaceous crops such as wheat or vegetables.

As previously noted, one specific area of weakness is the lack of expertise of the project team in the area of production agriculture (agronomy or production soil science), specifically precision agriculture. Thus a recommendation to improve this proposal would be to recruit an expert to the project team with experience in agricultural science and production agriculture. More scientific emphasis should be placed on deployment, testing and evaluation of the WSN under field conditions (e.g., TADCO farm) rather than on the engineering design. As previously noted, the project education, training and outreach plan needs to be clearly detailed to include how the instrumentation and results would be used to improve learning. There is need for a description of not only how the project would be organized and managed, but also how communication and evaluation of progress would be achieved. There is need for clear identification of the specific contribution of the Consultants in the project beyond reading the project report. Finally, the proposal should include specific plans and a vehicle for disseminating the results of the project to the scientific community locally, nationally and internationally.

List of Publications

Book Chapter

[1] el-H. Aggoune, S. Alwakeel, M. Alwakeel, and **M. Ammad-uddin**, “Wireless Sensor Networks with Dynamic Nodes for Water and Crop Health Management,” in Sensor Networks for Sustainable Development, CRC Press, 2014, pp. 27–40.

Journal

[2] M. Ayaz, **M. Ammad-uddin**, I. Baig , el –H. Aggoune. “Wireless Sensor's Civil Applications, Prototypes and Integration Possibilities: A Review,” *IEEE J. Sensors*, vol. 18, no. 1, pp. 4–30, Nov. 2017.

[3] **M. Ammad-uddin**, A. Mansour, D. Le Jeune, el-H. Aggoune. “Total Solution for Smart Agriculture: KSA Case Study”, (Submitted to IEEE Sensors Journal).

[4] **M. Ammad uddin**, A. Mansour, D. Le Jeune, el-H. Aggoune. “Data Collection by Drones for Smart Agriculture”, (Submitted to International Journal of Information, Communication Technologies and Applications).

Conference

[5] **M. Ammad-uddin**, M. Ayaz, A. Mansour, and D. Le Jeune, “Wireless Sensors for Modern Agriculture in KSA: A survey,” in 7th International Conference on Computer Science and Information Technology (CSIT) Amman, Jordan, 13-15 July, 2016, pp.1-7.

[6] **M. Ammad-udin**, A. Mansour, D. Le Jeune, E. H. M. Aggoune, and M. Ayaz, “UAV Routing Protocol for Crop Health Management,” in 24th European Signal Processing Conference (EUSIPCO), Budapest, Hungary, 29 August-2 September, 2016, pp. 1818–1822.

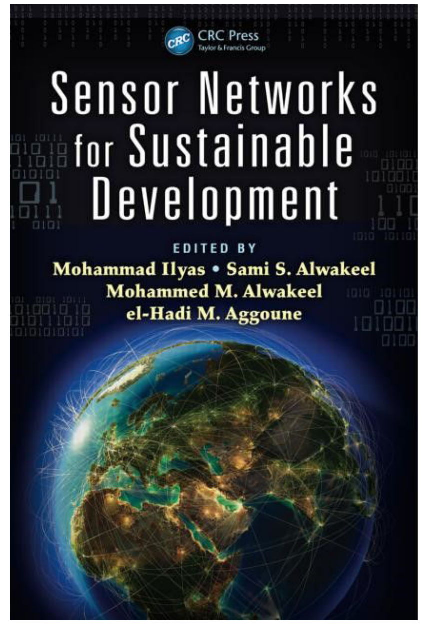
[7] **M. Ammad-uddin**, A. Mansour, D. Le Jeune, el-H. Aggoune. “Agriculture Internet of Things: AG-IOT”. 27th International Telecommunication Networks and Applications Conference (ITNAC), Melbourne, Australia, 22-24 November, 2017.

[8] **M. Ammad-uddin**, D. Le Jeune, A. Mansour, el-H. Aggoune, “Direction of Arrival of Narrowband Signals Based on Virtual Phased Antennas,” Asia-Pacific Conference on Communications, APCC, Perth, 11-13 December, 2017.

In book: Sensor Networks for Sustainable Development, Publisher: CRC Press

2 Wireless Sensor Networks with Dynamic Nodes for Water and Crop Health Management

el-Hadi M. Aggoune, Sami S. Alwakeel, Mohammed M. Alwakeel, and ***Mohammad Ammad-Uddin***



Contents

Abstract

Recent advances in microelectronic and microelectromechanical systems have produced new battery-powered sensor devices that have capabilities for detecting and processing physical information. These devices (nodes) can be connected to form a wireless sensor network (WSN) that performs a variety of operations. WSNs provide sensing accuracy and fault tolerance and can be deployed in harsh environments to provide continuous monitoring and processing capabilities. WSNs collect various types of data from a monitored area. Depending on the application, parameters sensed may include moisture, temperature, nutrients, and pollutants. Sensed information is carried over multi-hop from node to node to a base station (BS) for further processing and action taking. Given the numerous benefits WSNs offer, a case study was developed for their potential implementation in the farming sector in Saudi Arabia. Water utilization in Saudi is very critical as there is little permanent storage for it such as reservoirs or dams. At the same time, the Saudi land is fertile and has the potential to produce both quantity and quality crops such as wheat, dates, fruits, vegetables, flowers, and alfalfa. This case study focuses on WSNs to control water used for irrigation as well as for monitoring the quantity and quality of crops.

This study is motivated by both the lack of water and the premise that for the majority of crops, an excess of water may have an equally negative effect as does a deficit. Hence, the need for a technology as an aid to optimally dispense the appropriate amount of water for optimal crop quantity and quality. It is believed that WSNs provide an answer.

2017, IEEE Sensors Journal, (18446-2017.R1)

Wireless Sensor's Civil Applications, Prototypes and Future Integration Possibilities: A Review

Muhammad Ayaz, **Mohammad Ammad-uddin**, I. Baig, *Member IEEE,* el-Hadi M. Aggoune, *Senior Member IEEE*

***Abstract*— Advances in wireless communication are forging new possibilities for sensors. New sensors are equipping major systems around us with unparalleled intelligence as in the case of smart grids while the Internet of Things (IoT) is yet another major beneficiary of sensor technology. Considering the current developments in the field of sensor networks, one feels that it has reached on an interesting stage where sensor's role becoming crucial in numerous applications. This all speaks volumes of the fact that sensors are going to be at the front and center of most of the future technologies. Considering their vital role from futuristic perspective, this survey reports the characteristics of wireless sensors, their applications and prototypes that impact human life and wellbeing. In addition, what significant sensors are available for a particular application including their specifications and manufacturer whenever and wherever relevant and for which major projects are used over the last decade. Moreover, sensor's integration possibilities with other networks and major technologies are discussed and the possible challenges and key benefits are also highlighted. This research effort can be considered to accentuate the latest developments in the area of sensors and sensor networks as research gears up to meet the challenges of the emerging technologies and their applications particularly that are going to rely heavily on IoT and smart sensors.**

Index Terms*—** *Wireless Sensors, Sensor applications and classification, Prototypes, Communication, Integration possibilities, Future expectations.**

I. INTRODUCTION

Wth their reduced size, power consumption and cost, wireless sensors have gained access to almost every aspect of life. Vital among these reasons are ease of deployment, low maintenance requirement and ability to operate in harsh and hostile environment without requiring any supervision. In fact, wireless sensors are linking the physical world with digital computational systems where the sensed information is used to take some timely decision. The increased acceptance of wireless sensors across industry is well founded [1-3].

****

Muhammad Ayaz, is assistant professor and member of Sensor Networks and Cellular Systems (SNCS) Research Center, University of Tabuk, KSA (e-mail: ayazsharif@hotmail.com).

M. Ammad-uddin is final year PhD scholar at Lab STICC, ENSTA Bretagne, Brest, France. mohammad.ammad-uddin@ensta-bretagne.org

Imran Baig working as assistant professor at department of Electrical and Computer Engineering (ECE), Dhofar University, Oman (e-mail: ibaig@du.edu.om).

el-Hadi M. Aggoune is currently serving as professor and director of Sensor Networks and Cellular Systems(SNCS) Research Center, University of Tabuk, KSA (e-mail: haggoune@ut.edu.sa).

Sensors and sensor networks are now the part of our living environment even being integrated into our clothing with

Fig. 1: Wireless Sensors for Remote Sensing

minimal disruption and many other daily activities [4,5]. At one
side, sensors are embedded in our bodies and surrounding environment to monitor our health and safety [6,7], same time they are also integrated in our economic and social systems and thus becoming an important player in their sustainability. They are in the hundreds and thousands embedded in our transportation, safety and defense, business and industry, food supply chain, environmental systems and countless other applications. Without the use of sensors, there would hardly be automation, imagine a product assembly line of any kind without sensors or sensor driven actuator.

Despite their widespread use, sensors used individually or networked with other technologies still have limitations that this paper will discuss. In a sense, the parameters that make sensors attractive for wide range applications, limit their capacity and ability. For example, the smaller size and form of sensors allow them to use only tiny batteries that have a relatively short useful time span. To extend the life of the battery as much as possible, a low power transceiver must be used, thus limiting in both its signal strength and throughput data rate. To mitigate these two shortcomings, lately research has focused in augmenting the capabilities of sensors, for example, with unmanned vehicles that play the role of mobile relays which help to enhance the network life and reduce the network dependency. However, this has led to the emergence of other issues such as synchronization of signals, security and privacy compromises and need of continuous maintenance. Now, knowing the advantage gained from MIMO (multiple input, multiple output) technology, there are serious attempts.

2016, 24th European Signal Processing Conference, (EUSIPCO)

UAV Routing Protocol for Crop Health Management

Muhammad Ammad-udin[1,2], Ali Mansour[1], Denis Le Jeune[2], el. Hadi M. Aggoune[2], Muhammad Ayaz[2]
[1]Lab STICC, ENSTA Bretagne, Brest, France
[2]Sensor Network and Cellular Systems (SNCS) Research Center, University of Tabuk, Tabuk 71491, KSA
mashfaq@ut.edu.sa, mansour@ieee.org, denisle_jeune@ensta-bretagne.fr,
aggoune.sncs@ut.edu.sa, ayazsharif@hotmail.com

Abstract —**Wireless sensor networks are now a credible means for crop data collection. The installation of a fixed communication structure to relay the monitored data from the cluster head to its final destination can either be impractical because of land topology or prohibitive due to high initial cost. A plausible solution is to use Unmanned Aerial Vehicles (UAV) as an alternative means for both data collection and limited supervisory control of sensors status. In this paper, we consider the case of disjoint farming parcels each including clusters of sensors, organized in a predetermined way according to farming objectives. This research focuses to drive an optimal solution for UAV search and data gathering from all sensors installed in a crop field. Furthermore, the sensor routing protocol will take into account a tradeoff between energy management and data dissemination overhead. The proposed system is evaluated by using a simulated model and it should find out a class among all under consideration.**

Keywords- Smart farming; Routing protocol; Precision agriculture; Data gathering; Wireless sensor network, UAV

I. INTRODUCTION

The total area of the Kingdom of Saudi Arabia (KSA) is 2,149,690 km2. While, only 1.6% of it is urban area, and about 80% (1,736,250 km2) is desert of which only 1.6% is agriculture land [1]. The biggest hurdles for cultivation are the shortage of water, the spread of land, and adverse weather and atmospheric conditions. To cope with the scarcity of water, there is a need to equip the agricultural sector with modern tools and implement scientific approaches suggested by the fast-developing precision agriculture and smart agriculture relying on Wireless Sensor Networks (WSNs) to achieve sustainability. More recently, with the advent of Unmanned Aerial Vehicles (UAVs) and the accompanying progress in research and development in ad-hoc and vehicular communication, WSNs are positioned to gain further functionality. Some of the nodes can become dynamic (carried by UAVs) facilitating both data collection and wireless communication in areas that are not equipped with fixed communication infrastructures. Sensors are normally planted in strategic locations forming disjoint network and subnetworks in individual parcels. The data can be collected from the individual networks using UAVs that have the ability to loiter and hover at certain collection points. The objective is to collect and store vital data relative to the environment, soil, and crop and allowing farmers to make timely decisions or decision is taken by automatic Supervisory Control and Data Acquisition (SCADA) system. In this proposed system, all the field sensor nodes are taken static while, a mobile sink (UAV) is used to harvest the field data. The system is supposed to be dynamic enough to adapt network changes like nature of data, type of sensors, number of alive sensor node, field area of interest, and path of UAV in every mission, as well as robust in the sense that can sustain even in severe weather and geographical conditions.

Many routing and data gathering schemes are developed and proposed for wireless sensor networks, we classify existing schemes into four categories: 1) static sink routing, 2) mobile sink direct contact data collection and 3) rendezvous based data collection 4) Hash Table based routing. Static sink routing protocols like LEACH [2], HEED [3], Linked cluster [4], Adaptive clustering [5], random competition based clustering [6], and node hierarchical control clustering [7] are not suitable in our scenario because of their fix communication infrastructure and no compatibility with mobile sink (UAVs), therefore we are not discussing them. Other related three types of protocol are described below and comparison is given in TABLE 1.

A. Mobile Sink Direct Contact Data Collection

In this category of protocols, data is collected from the sensor network by using mobile sinks. But sink has to collect data by visiting each sensor node in the network one by one, therefore is not considered efficient, due of very low latency and small coverage area.

B. Rendezvous based Data Collection

In this type of data collection, sensor nodes are grouped in clusters and the mobile sink has to visit each cluster at predefined rendezvous (appointment points) which acts as CH and delivers the data to the mobile sink.

C. Hash Table

Data is hashed according to the geographical locations and collected by static or mobile sink [8].

D. Distinction of our Proposed System

In our proposed system, fixed (static) sensor nodes are deployed in a crop field and a mobile sink (UAV) is used to collect data. All the sensor nodes are heterogeneous in nature and deployed to monitor different parameters. Sensor nodes are unaware of their location, the location of UAV and its path. The first distinction of this research is that only specific data from selective sensors is collected from the field in order to investigate as per need basis. While, second distinction is the path of UAV, which is fixed but adaptive as well means combination of both. Fix in the sense that it needs to follow a predefined path to scan a particular area (suspected area or area of interest) and at the same time UAV can deviate from its path up to certain extent as per location of cluster and cluster head. Clusters are made dynamically with respect to the UAV path.

2016, 7th International Conference on Computer Science and Information Technology, (CSIT)

Wireless Sensors for Modern Agriculture in KSA: A Survey

Mohammad Ammad Uddin[1,2], Muhammad Ayaz[1], Ali Mansour[2], Denis Le Jeune[2], el. Hadi M. Aggoune[1]
[1]Sensor Network and Cellular Systems (SNCS) Research Center, University of Tabuk, Tabuk 71491, KSA
[2]Lab STICC, ENSTA Bretagne, Brest, France,

***Abstract*— Agriculture in Kingdom of Saudi Arabia (KSA) is not same as in other part of world where scarcity of water and very intense weather condition makes it more challenging. With the advancement in technology, sensors and other wireless devices are being integrated in different daily life applications. Agriculture is one of them, where sensor networks can be used to improve the quality and quantity of yield by utilizing precise amount of resources where water is the most critical. Wireless sensors for Smart Agriculture (SA) are being used from many years but facing different challenges like large and remote geographical areas, limited or unavailable communication infrastructure, reliability of un-attendant sensor nodes etc. Typical challenges faced by SA are even overstated in the considering case study by adding worst weather condition, and tight irrigation supervision. The purpose of this article is to survey the potential technologies and sensing devices that can be used in KSA environment to acquire data from field same time providing plausible mechanism to aggregate it. The most important objective of this survey is to identify such modifications, customizations or supplement parameters that are required to incorporate in SA system to adapt it for KSA environment.**

Keywords-- Wireless Sensors, Sensor Applications, Smart Farming, Irrigation, Crop Health.

I. INTRODUCTION

The total area of the Kingdom of Saudi Arabia (KSA) is 2,149,690 km2. While, about 1.6% of it is urban area, and about 80% (1,736,250 km2) is desert of which only 1.6% is arable land [17], [18]. The biggest hurdles for cultivation are shortage of water, large spread of land, and adverse weather and atmospheric conditions. KSA is a desert country with virtually no permanent rivers or lakes and with only limited bursts of rainfall during a short time span of year. Additionally, there is an ever-increasing demand for water to suit the population of a typical fast-developing country in terms of construction, industry, and lifestyle [20], [21], [32]. Crops are grown in dispersed circulator rectangular-shaped parcels of land having limited water resources and exposed to harsh environmental conditions including excessive heat or cold weather and sandstorms. Furthermore, the farming parcels have limited or no communication infrastructures. Most common crops include dates, seasonal fruits and vegetables, olives, wheat, and alfalfa. It is worth mentioning that wheat growing is receding because of its water requirements.

To produce quality crops in KSA, the following facts are need to focus 1) crop parameters (leaf wetness, leaf chlorophyll level, height of plant, water circulation, fruit size ect.) to monitor and maintain crop health 2) soil parameters as plant growth is also effected by soil quality as mentioned by International Center for Soil Fertility and Agricultural Development (IFDC) that owing to the limitations in farming practices such as fertilizer usage, the levels of soil nutrients are declining at an annual rate of 30 Kg /ha in 85 % of African farm land [183] and 3) Environmental factors like temperature, humidity, sunlight, presence of carbon dioxide and oxygen etc. Wireless Sensor Networks (WSNs) are considering as the enabling technology for smart agriculture as it can provide real time feed-back on a number of different crop, soil and site parameters. With the use of WSN, notable increase in yield amount is possible by utilizing precise amount of resources. Using WSN, crop health is being monitored as well as amount of water, fertilizer, and pesticides. This technology can isolate a single plant for monitoring and nurturing, or more typically an area in the tens or hundreds of square feet.

This survey presents: 1) survey of different type of technologies and sensors available for agriculture, and how we can use these technologies to improve quality and quantity of crops in KSA. 2) Short survey of potential data gathering schemes that can be used to collect data from different field sensors. 3) Identify short comings in existing crop monitoring systems and required modifications or improvements in these systems to cope with KSA agriculture needs.

The rest of the paper is organized as follows. Section 2 provides an overview on sensors, dividing them in different categories according to monitoring parameters. Section 3 discusses existing routing and data gathering schemes proposed for agriculture applications. Section 4 includes different comparison tables based on sensor types, manufactures and some of the famous test beds. Further, some suggested alterations or additional features that need to be considered to build in smart agriculture to make it compliance with KSA agriculture environment are provided in section 5, while section 6, briefly concludes this article including some future issues.

II. SENSOR CATEGORIES BASED ON MONITORING PARAMETERS

This section presents a survey of some renowned technologies that are being used to monitor crop parameters, so that resources like water, pesticide, fertilizer etc., can be used in more precise way. We can divide crop monitoring in following categories.

2017, 23rd Asia-Pacific Conference on Communications, (APCC)

Direction of Arrival of Narrowband Signals Based on Virtual Phased Antennas

M. Ammad Uddin[1,2], D. Le Jeune[2], A. Mansour[2], el-H.M. Aggoune[1]

[1]Sensor Networks and Cellular Systems Research Center, University of Tabuk, Saudi Arabia
[2]Lab STICC, ENSTA Bretagne, Brest, France. mohammad.ammad-uddin@ensta-bretagne

***Abstract*— Data collection from field sensors by using Unmanned Aerial Vehicle (UAV) is the application taken in consideration in this paper. All the sensor nodes are kept location unaware to reduce their cost and energy utilization. The issue addressed in this paper is localization of sensor nodes by UAV to collect data in efficient way. ULA of multiple antennas are used to measure the Angle of Arrival (AoA) of incoming signals. However, the drawbacks of mounting such multiple antennas on an Unmanned Aerial Vehicle (UAV) outweigh the benefits. The challenge is to affix multiple antennas and receivers on an UAV, increase its weight which ultimately decrease its payload capacity, flight time, speed and agility. In this paper, we are proposing a new method to estimate the AoA, called Virtual Phase Array (VPA) antenna system. A single moving antenna installed over an UAV taking snapshots every fixed time periods forms an antenna array virtually. This VPA has enable us to introduce two new concepts of adaptive steering precision and multiple frequency use. All these became reality only because number and spacing between antenna elements can be adjusted, which is not easy to implement in physical antenna array especially when antenna is onboard. The proposed system is evaluated by simulation model. Suggested modifications and additions in classical MUSIC algorithm make it possible to operate the virtual antenna system with the same precision as the physical antenna may have, but adding more flexibility, ease of use, cost economy, more reliability and better throughput.**

***Index Terms*— Array antenna system, direction of arrival, narrowband signal, virtual phased array antenna**

I. INTRODUCTION

Agriculture field is taken as a case study, in this research paper. Large number of heterogeneous sensors are installed in a crop field to monitor various parameters related to the plant health, soil and atmospheric conditions. All these sensors are location unaware and left unattended in the field. As farm field is in very remote area and no fixed infrastructure is available for communication, so an Unmanned Aerial Vehicle (UAV) is used as a mean of communication and to harvest desired data from these field sensors. Adverse environmental factors are another challenge where rain, dust storm, growing follicles of plants, may cause hurdles in data communication. Considering all these challenges, we proposed an UAV data routing protocol for crop health monitoring in EUSIPCO 2016 [128]. Dynamic clustering of heterogeneous field sensor nodes and runtime Cluster Head (CH) selection criteria was introduced in that article. As mentioned before, all field sensors are location unaware. In

this case, it is the responsibility of UAV to estimate the location of multiple sensors deployed on the ground to collect data from all of them efficiently. The success and efficiency of above mentioned dynamic data collection scheme lies in accurate estimation of CHs locations. We have taken this particular example in consideration, and proposing a Virtual Phase Array (VPA) antenna system which will mitigate many challenges, a typical physical antenna system is facing, which restrict it to be installed in small size UAV such as: heavy weight, big size, energy demanding, etc.

The accurate estimation of Angle of Arrival (AoA) of a signal is very important in many applications such as: those involving Radar [92], Sonar [93], Emergencies and surveillance [94], and cellular systems [95]. To find AoA, one should use a set of multiple antennas which is either fixed to form an array, or rotating in case of radar, except some exception like multi cell static radar in aircrafts. An array of antenna system can be used to detect many parameters of the incoming signal including range, frequency, polarization and the most important is AoA. Array antenna system not only improves the resolution of AoA of incoming signal, but also makes it possible to identify multiple sources that are emitting these signals. AoA can be described as the direction in terms of angle (azimuth θ and elevation Φ), created by multiple plane wave signals (narrowband or wideband) incident on a single or array of antennas.

Array geometry is another important factor in AoA estimation accuracy (resolution), which may composed of a set of antennas organized in a particular formation such as: Uniform Linear Array (ULA), Uniform Circular Array (UCA), Concentric Circular Arrays (CCAs), Uniform Rectangular Arrays (URAs), L-shaped array, V-shaped array, Displaced Sensor Arrays (DSAs), Parallel linear arrays and Y-shaped arrays for details see [96][97][98]. Uniform Circular Array (UCA) is proposed in [96] to provide two-dimensional coverage and uniform performance in all azimuth directions at the cost of adding complexity. AoA is estimated using a rectangle geometry with 8 elements is developed in [97]. Despite all these, ULA is the simplest possible array geometry working on narrowband signals, delivers acceptable resolution and accuracy during beamforming and AoA estimation. The strength of ULA is its simple structure, less computational / processing requirements and a good resolution for the AoA estimation.

2017, 27th International Telecommunication Networks and Applications Conference, (ITNAC)

Agriculture Internet of Things: AG-IOT

M. Ammad Uddin[1,2], A.Mansour[1], D. Le Jeune[1]
[1]Sensor Networks and Cellular Systems Research Center, University of Tabuk, Saudi Arabia
[2]Lab STICC, ENSTA Bretagne, Brest, France. mohammad.ammad-uddin@ensta-bretagne.org

***Abstract*— The Internet of Things (IoT) for agriculture is a rapidly emerging technology where seamless connected sensors device make it possible to monitor and control crop parameters to get quality and quantity of food. This research proposes a new dynamic clustering and data gathering scheme for harnessing the IoT in agriculture. In this paper, an Unmanned Aerial Vehicle (UAV) is used to locate and assists ground IoT devices to form themselves in cluster formation then establishes a reliable uplink communications backbone for data transmission. Use of multi-frequency, multi power transmission, and mobile sink make it possible to reduce power utilization of IoT devices as much as possible. The proposed scheme is evaluated by using simulation models and practical experiments. It is found working outclass as compare to all existing systems.**

Keywords*— *IoT; Smart agriculture; Dynamic routing; WSN; Clustering

I. INTRODUCTION

In this era, Smart Agriculture (SA)[1]–[4] is not only a technology to ease the human life but it has rather become a necessity or even a compulsion to cope with rapidly increasing food demand of the world population which is multiplying itself every second. With the passage of time, the agriculture sector is facing more problems and greater challenges such as falling land fertility and dwindling water reservoirs. Some of the wildlife is losing their habitat and thus being pushed to the verge of extinction. Furthermore, arable lands are being replaced by urban population and industrial units at an alarming rate. Environmental pollution, excessive use of pesticides and contaminated water are some additional factors, which are compounding the problems in agriculture further.

Smart agriculture in the Kingdom of Saudi Arabia (KSA) is considered as case study in this research. The agriculture sector in the KSA faces even greater challenges because of scarcity of water, very extreme climatic conditions characterized by high temperatures, dry air, dust / sand storms, vast expanse of desert and lack of communication infrastructure in very remote geographical locations. The only plausible solution to overcome the above-mentioned challenges lies in making an effective use of the most modern tools and technologies in classical agriculture such as Unmanned Aerial Vehicles (UAVs) and Internet of Things (IoT) field devices to ensure optimum usage of the available resources in achieving better quality and higher yield of crops. In this study, UAV based Routing Protocol (URP) is proposed for Agriculture-IoT we name it (AG-IoT). The life cycle of proposed AG-IoT is consists of six steps (Figure-1). Wide range of heterogynous IoT devices are used to monitor different parameters related to crop, soil and environment. UAVs are utilized to build cheap, handy and instant communication infrastructure for AG-IoT devices that is considered as reliable backbone uplink.

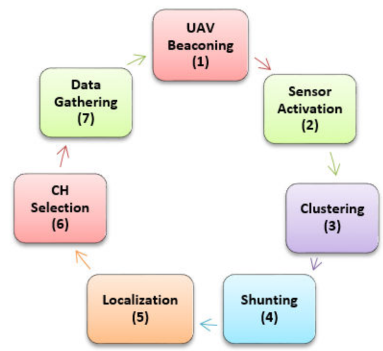


Figure 1: Life Cycle of the proposed system

The process starts when UAV send beacon message to activate ground sensors. The entire activated sensors make a cluster on UAV call. Once cluster is formed, all the nodes that are capable to communicate with UAV starts replying a narrow band signal and begin contesting to be selected as Cluster head (CH). Shunting is the next step to keep the contestants in a reasonable range (more than one and less then UAV antenna capacity). UAV has limited functionality to locate sensor nodes that is mostly (M-1) where, M is the number of antenna elements mounted over UAV. Shunting is introduced to limit number of replying sensors between 1 and M. UAV locates and evaluate all the candidate cluster heads and select one of them as CH then establish a link with it to collect data.

A. Localization

In proposed AG-IoT, many sensor nodes are installed in crop field to monitor crop, soil and environmental parameters. An UAV is used to harvest data from ground sensors. UAV should know the exact number and location of sensor nodes, not only to collect data but also to assist them in cluster formation and CH selection. Many schemes are proposed for localization like [13], [14] but in all proposed schemes multiple antennas are used to measure Angle of Arrival (AOA) for incoming signal to estimate location of field sensor. However, the drawbacks of mounting such multiple antennas on UAV outweigh the benefits. The challenge is that adding multiple antennas and receivers on an UAV, increases its weight which ultimately decreases its payload capacity, flight time, speed and agility. Design and development of light weight energy efficient antenna that can be installed on small sized UAV was a challenging task. It was the first phase of this research study, that is already done and under publication process. In this paper, we are considering that the used UAV is equipped with light weight energy efficient localization antenna and always have up to date list of connected nodes and their locations in the 3D plane.

Bibliography

[1] X. Zhang, J. Zhang, L. Li, Y. Zhang, and G. Yang, “Monitoring Citrus Soil Moisture and Nutrients Using an IoT Based System,” *IEEE J. Sensors*, vol. 17, no. 3, pp. 430–447, Feb. 2017.

[2] M. H. Kabir, K. Ahmed, and H. Furukawa, “A Low Cost Sensor Based Agriculture Monitoring System Using Polymeric Hydrogel,” *J. Electrochem. Soc.*, vol. 164, no. 5, pp. 107–112, Mar. 2017.

[3] T. Wark *et al.*, “Transforming Agriculture through Pervasive Wireless Sensor Networks,” *IEEE J. Pervasive Comput.*, vol. 6, no. 2, pp. 50–57, Apr. 2007.

[4] P. Tokekar, J. Vander Hook, D. Mulla, and V. Isler, “Sensor Planning for a Symbiotic UAV and UGV System for Precision Agriculture,” *IEEE Trans. Robot.*, vol. 32, no. 6, pp. 1498–1511, Dec. 2016.

[5] C. Fu, Z. Jiang, W. Wei, and A. Wei, “An Energy Balanced Algorithm of LEACH Protocol in WSN,” *Int. J. Comput. Sci.*, vol. 10, no. 1, pp. 354–359, 2013.

[6] X. Kaixin and M. Gerla, “A Heterogeneous Routing Protocol Based on a New Stable Clustering Scheme,” in *Military Communication Conference MILCOM*, 2010, vol. 2, pp. 838–843.

[7] L. Shangguan, L. Mai, J. Du, W. He, and H. Liu, “Energy-efficient Heterogeneous Data Collection in Mobile Wireless Sensor Networks,” in *20th International Conference on Computer Communication Networks (ICCCN’2011)*, 2011.

[8] A. W. Khan, A. H. Abdullah, M. A. Razzaque, and J. I. Bangash, “VGDRA: A Virtual Grid-Based Dynamic Routes Adjustment Scheme for Mobile Sink-Based Wireless Sensor Networks,” *IEEE J. Sensors*, vol. 15, no. 1, pp. 526–534, Jan. 2015.

[9] H. Okcu and M. Soyturk, “Distributed Clustering Approach for UAV Integrated Wireless Sensor Networks,” *Int. J. Ad Hoc Ubiquitous Comput.*, vol. 15, no. 3, pp. 106–115, 2014.

[10] A. Khan, A. Abdullah, M. Anisi, and J. Bangash, “A Comprehensive Study of Data Collection Schemes Using Mobile Sinks in Wireless Sensor Networks,” *IEEE J. Sensors*, vol. 14, no. 2, pp. 2510–2548, Feb. 2014.

[11] J.-Y. Chang and J. Pei-Hao, “An Efficient Cluster-based Power Saving Scheme for Wireless Sensor Networks,” *EURASIP J. Wirel. Commun. Netw.*, vol. 2012, no. 1, pp. 172–179, Dec. 2012.

[12] B. T. Sharef, R. A. Alsaqour, and M. Ismail, “Vehicular Communication Ad-hoc Routing Protocols: A Survey,” *J. Netw. Comput. Appl.*, vol. 40, pp. 363–396, Apr. 2014.

[13] J. A. Srar, A. Mansour, and K.-S. Chung, “Adaptive Array Beamforming Using a Combined LMS-LMS Algorithm,” *IEEE Trans. Antennas Propag.*, vol. 58, no. 11, pp. 3545–3557, Nov. 2010.

[14] J. A. Srar and K.-S. Chung, “Performance of RLMS Algorithm in Adaptive Array Beam Forming,” in *11th IEEE Singapore International Conference on Communication Systems*, 2008, pp. 493–498.

[15] “Googel Maps,” *Google*, 2005. [Online]. Available: https://www.google.com/maps. [Accessed: 15-Sep-2017].

[16] F. Karam, “Saudi Arabia Food Imports Worth $24 Billion Annually,” *Middle East, News, Saudi Arabia*, 2017. [Online]. Available: https://www.middleeastmonitor.com/20140827-saudi-arabia-food-imports-worth-24-billion-annually/. [Accessed: 15-Sep-2017].

[17] Web, “Agricultural Land in Saudi Arabia,” *Trading Economics*. [Online]. Available: http://www.tradingeconomics.com. [Accessed: 01-Jun-2016].

[18] Web, “Saudi Arabia,” *Worldstat Info*, 2013. [Online]. Available: http://en.worldstat.info/Asia/Saudi_Arabia/Land. [Accessed: 01-Jun-2016].

[19] M. Makkawi, “Water Resouces of Saudi Arabia,” *King Fahd University of Petroleum and Minerals*, 2010. [Online]. Available: http://faculty.kfupm.edu.sa/ES/makkawi/ENVS524/module3.pdf. [Accessed: 15-Sep-2017].

[20] S. Abdulaziz, “Water Resources And Water Consumption Pattern In Saudi Arabia,” *Water Research Group, King Fahd University of Petroleum and Minerals*, 2012. .

[21] A. A.-I. Abdulla, “Water Use in Saudi Arabia: Problems and Policy Implications,” *J. Water Resour. Plan. Manag.*, vol. 116, no. 3, pp. 375–388, May 1990.

[22] MOFA, “Dams in the Kingdom,” *Ministry of Foreign Affairs*, 2012. [Online]. Available: http://www.mofa.gov.sa/sites/mofaen/ServicesAndInformation/news/GovernmentNews/Pages/NewsArticleID72349.aspx. [Accessed: 15-Sep-2017].

[23] WikiPedia, “Dams in Saudi Arabia,” *WikiPedia*, 2014. [Online]. Available: http://en.wikipedia.org/wiki/Category:Dams_in_Saudi_Arabia. [Accessed: 15-Sep-2017].

[24] WikiPedia, “List of Dams and Reservoirs,” *Wikipedia*, 2014. .

[25] Web, “Desalination Plants,” *Corporation, Saline Water Conversion*, 2017. [Online]. Available: http://www.swcc.gov.sa/english/Projects/DesalinationPlants/Pages/default.aspx. [Accessed: 15-Sep-2017].

[26] M. Picow, “Saudi Arabia Opens World’s Largest Desalination Plant,” *Green Prophet*, 14-May-2009.

[27] W. Alzuhair, “Azzizia Desalination Plant,” *Flickr*. [Online]. Available: https://www.flickr.com/photos/waleedalzuhair/2175773780/. [Accessed: 15-Sep-2017].

[28] K. H. Al-Zahrani, “Water Demand Management in the Kingdom of Saudi Arabia,” *Int. J. Arts Sci.*, vol. 2, no. 3, pp. 68–76, 2010.

[29] O. K. M. Ouda, “Review of Saudi Arabia Municipal Water Tariff,” *J. World Environ.*, vol. 3, no. 2, pp. 66–70, 2013.

[30] O. K. M. Ouda, “Towards Assessment of Saudi Arabia Public Awareness of Water Shortage Problem,” *J. Resour. Environ.*, vol. 3, no. 1, pp. 10–13, 2013.

[31] M. Makkawi, “Resources and Environment,” 2015. [Online]. Available: http://faculty.kfupm.edu.sa/ES/makkawi/ENVS524/module3.pdf. [Accessed: 15-Sep-2017].

[32] M. Makkawi, “Water Resources of Saudi Arabia,” *King Fahd University of Petroleum and Minerals*, 2013. [Online]. Available: http://faculty.kfupm.edu.sa/ES/makkawi/ENVS524/module3.pdf. [Accessed: 15-Sep-2017].

[33] NASA, “NASA Sees Fields of Green Spring up in Saudi Arabia.” [Online]. Available: http://www.nasa.gov/topics/earth/features/saudi-green.html. [Accessed: 15-Sep-2017].

[34] M. Ayaz, M. Ammad-Uddin, I. Baig, and el-H. M. Aggoune, “Wireless Sensor’s Civil Applications, Prototypes and Integration Possibilities: A Review,” *IEEE J. Sensors*, vol. 18, no. 1, pp. 4–30, Nov. 2017.

[35] F. Schmidt, “Agricultural Sensors: Improving Crop Farming to Help Us Feed the World.” [Online]. Available: http://www.dw.com/en/agricultural-sensors-improving-crop-farming-to-help-us-feed-the-world/a-17733350. [Accessed: 28-Feb-2016].

[36] O. López, M. Rach, H. Migallon, M. Malumbres, A. Bonastre, and J. Serrano, “Monitoring Pest Insect Traps by Means of Low-Power Image Sensor Technologies,” *IEEE J. Sensors*, vol. 12, no. 12, pp. 15801–15819, Nov. 2012.

[37] M. Rach, H. Gomis, O. Granado, M. Malumbres, A. Campoy, and J. Martín, “On the Design of a Bioacoustic Sensor for the Early Detection of the Red Palm Weevil,” *IEEE J. Sensors*, vol. 13, no. 2, pp. 1706–1729, 2013.

[38] L. Bedord, “Sensors Protect Crops from Insect Damage,” *Meredith Corporation*, 2015. [Online]. Available: http://www.agriculture.com/technology/crop-management/fieldwork/senss-protect-crops-from-insect-damage_590-ar47778. [Accessed: 28-Feb-2016].

[39] R. Stoner, “The Rev 3 Leaf Sensor,” *Word Press*, 2014. [Online]. Available: https://leafsensor.wordpress.com/. [Accessed: 15-Sep-2017].

[40] Web, “LT-1M Leaf Temperature Sensor,” *Phyto Sensor Group*, 2010. [Online]. Available: http://phyto-sensor.com/LT-1M. [Accessed: 15-Sep-2017].

[41] E. V. Thomsen, “Greenhouse Sensor,” *DTU Nanotech*. [Online]. Available: http://www.nanotech.dtu.dk/Research-mega/Forskningsgrupper/MEMS-AppliedSensors/NewResearch/Single-Chip-Multi-Sensor/Greenhouse-Sensor. [Accessed: 17-Jul-2017].

[42] S. L. Kedar, "SD-5M Stem Microvariation Sensor," *Agri Support Online*. [Online]. Available: http://www.store.agrisupportonline.com/Phytech/technical_data/sd-5m.htm. [Accessed: 17-Jul-2017].

[43] Web, "DE-1M Dendrometer," *Phyto-Sensor Group*. [Online]. Available: http://phyto-sensor.com/DE-1M. [Accessed: 17-Jul-2017].

[44] Web, "SF-4M, SF-5M Sap Flow Sensors," *Phyto Sensor Group*. [Online]. Available: http://phyto-sensor.com/SF-4M-SF-5M. [Accessed: 17-Jul-2017].

[45] Web, "Fruit Growth Sensors," *Phyto Sensor Group*. [Online]. Available: http://phyto-sensor.com/FI-LM-FI-MM-FI-SM. [Accessed: 28-Feb-2016].

[46] K. Lawton, "Wireless Crop Sensing Technology Keeps Improving," *Precision AgWired*, 2010. [Online]. Available: http://precision.agwired.com/2010/06/16/wireless-crop-sensing-technology-keeps-improving/. [Accessed: 15-Sep-2017].

[47] D. L. Karlen, M. J. Mausbach, J. W. Doran, R. G. Cline, R. F. Harris, and G. E. Schuman, "Soil Quality: A Concept, Definition, and Framework for Evaluation (A Guest Editorial)," *Am. J. Soil Sci. Soc.*, vol. 61, no. 1, p. 4, 1997.

[48] Web, "EC-5 Small Soil Moisture Sensor," *Decagon Devices*. [Online]. Available: http://www.decagon.com/en/soils/volumetric-water-content-sensors/ec-5-lowest-cost-vwc/. [Accessed: 01-Jun-2016].

[49] "EO-1 Earth Observing-1," *U.S. Geological Survey's*, 2017. [Online]. Available: https://lta.cr.usgs.gov/ALI. [Accessed: 15-Sep-2017].

[50] "SPOT Satellite Pour l'Observation de la Terre," *EOEdu observing our planet*, 2017. [Online]. Available: http://eoedu.belspo.be. [Accessed: 15-Sep-2017].

[51] Web, "Multispectral VS Hyperspectral in Agriculture," *GAMAYA*, 2016. [Online]. Available: http://gamaya.com/blog-post/multispectral-vs-hyperspectral/. [Accessed: 15-Sep-2017].

[52] Web, "Vegetation," *SpecTIR Remote Sensing Division (SRS)*. [Online]. Available: http://www.spectir.com/applications/vegetation-airborne-hyperspectral/. [Accessed: 17-Jul-2017].

[53] A.-K. Mahlein, "Plant Disease Detection by Imaging Sensors – Parallels and Specific Demands for Precision Agriculture and Plant Phenotyping," *J. Plant Dis.*, vol. 100, no. 2, pp. 241–251, Feb. 2016.

[54] A.-K. Mahlein, U. Steiner, C. Hillnhütter, H.-W. Dehne, and E.-C. Oerke, "Hyperspectral Imaging for Small-Scale Analysis of Symptoms Caused by Different Sugar Beet Diseases," *J. Plant Methods*, vol. 8, no. 1, pp. 3–13, 2012.

[55] S. C. Dunagan, M. S. Gilmore, and J. C. Varekamp, "Effects of Mercury on Visible/Near-Infrared Reflectance Spectra of Mustard Spinach Plants," *J. Environ. Pollut.*, vol. 148, no. 1, pp. 301–311,

Jul. 2007.

[56] J. A. Gamon, L. Serrano, and J. S. Surfus, "The Photochemical Reflectance Index: an Optical Indicator of Photosynthetic Radiation use Efficiency Across Species, Functional Types, and Nutrient Levels," *J. Oecologia*, vol. 112, no. 4, pp. 492–501, Nov. 1997.

[57] S. L. Osborne, J. S. Schepers, D. D. Francis, and M. R. Schlemmer, "Detection of Phosphorus and Nitrogen Deficiencies in Corn Using Spectral Radiance Measurements," *J. Agron.*, vol. 94, no. 6, p. 1215, 2002.

[58] E. Ben-Dor, K. Patkin, A. Banin, and A. Karnieli, "Mapping of Several Soil Properties Using DAIS-7915 Hyperspectral Scanner Data - a Case Study over Clayey Soils in Israel," *Int. J. Remote Sens.*, vol. 23, no. 6, pp. 1043–1062, 2002.

[59] R. A. Viscarra Rossel, R. Webster, E. N. Bui, and J. A. Baldock, "Baseline Map of Organic Carbon in Australian Soil to Support National Carbon Accounting and Monitoring under Climate Change," *J. Glob. Chang. Biol.*, vol. 20, no. 9, pp. 2953–2970, Sep. 2014.

[60] J. Farifteh, F. Van der Meer, C. Atzberger, and E. J. M. Carranza, "Quantitative Analysis of Salt-Affected Soil Reflectance Spectra: A Comparison of Two Adaptive Methods (PLSR and ANN)," *J. Remote Sens. Environ.*, vol. 110, no. 1, pp. 59–78, Sep. 2007.

[61] "Leaf Insect," *Keeping Insects*, 2017. [Online]. Available: https://www.keepinginsects.com/stick-insect/species/leaf-insect-psg-278/. [Accessed: 15-Sep-2017].

[62] Web, "Keenlon Sound Sensor," *Keenlon*, 2014. [Online]. Available: http://en.gbot.cn/cp/html/?40.html. [Accessed: 20-Oct-2015].

[63] Web, "1/2.5-Inch 5Mp CMOS Digital Image Sensor," *Aptina*. [Online]. Available: https://s1.dl.theimagingsource.com/api/2.0/packages/publications/sensors-cmos/mt9p031/67c1f32125/mt9p031_1.2.en_US.pdf. [Accessed: 15-Sep-2017].

[64] M. Sanchez, D. Martinez de, and A. Ollero, "Cooperative Localization and Tracking With a Camera-Based WSN," in *2009 IEEE International Conference on Mechatronics*, 2009, pp. 1–6.

[65] V. Pande, W. Elmannai, and K. Elleithy, "Classification and Detection of Fire on WSN using IMB400 Multimedia Sensor Board," in *2013 IEEE Long Island Systems, Applications and Technology Conference (LISAT)*, 2013, pp. 1–6.

[66] Web, "2.1 Mp/Full HD Digital Image Sensor," *ON Semiconductor*. [Online]. Available: http://www.onsemi.com/PowerSolutions/product.do?id=AR0238. [Accessed: 15-Sep-2017].

[67] "Index DataBase," *IDB Project*, 2017. [Online]. Available: http://www.indexdatabase.de/db/i.php. [Accessed: 18-Jul-2017].

[68] Web, "OCI UAV Airborne Hyperspectral Camera," *BaySpec, Inc.* [Online]. Available: http://www.bayspec.com/spectroscopy/oci-uav-hyperspectral-camera/. [Accessed: 18-Jul-2017].

[69] Sense Fly SA, "eBee SQ The Advanced Aricultural Drone," *eBee*. [Online]. Available: https://www.sensefly.com/drones/ebee-sq.html. [Accessed: 08-Aug-2017].

[70] Web, "End-to-End Aerial Data Platform," *Precision Hawk*. [Online]. Available: http://www.precisionhawk.com/. [Accessed: 08-Aug-2017].

[71] "DJI Phantom 4 Specs," *DJI Phantom*. [Online]. Available: http://www.drone-world.com/dji-phantom-4-specs/. [Accessed: 08-Aug-2017].

[72] Web, "Electric UAV Flight Time Calculator," *UAV RF Dbershad sky*, 2015. [Online]. Available: http://controls.ae.gatech.edu/dbershad/EMSTAirTimeCalculator.html. [Accessed: 20-Oct-2015].

[73] A. A. Abbasi and M. Younis, "A Survey on Clustering Algorithms for Wireless Sensor Networks," *J. Comput. Commun.*, vol. 30, no. 14–15, pp. 2826–2841, Oct. 2007.

[74] X. Li, A. Nayak, and I. Stojmenovic, "Exploiting Actuator Mobility for Energy-Efficient Data Collection in Delay-Tolerant Wireless Sensor Networks," in *2009 Fifth International Conference on Networking and Services*, 2009, pp. 216–221.

[75] P. L. Cowen, "The Traveling-Salesman Problem," *J. Oper. Res.*, vol. 7, no. 1, pp. 1–11, 1956.

[76] R. Tarjan, "Depth-First Search and Linear Graph Algorithms," *SIAM J. Comput.*, vol. 1, no. 2, pp. 146–160, Jun. 1972.

[77] G. Xing, T. Wang, W. Jia, and M. Li, "Rendezvous Design Algorithms for Wireless Sensor Networks With a Mobile Base Station," in *9th ACM iIternational Symposium on Mobile Ad-hoc Networking and Computing - MobiHoc '08*, 2008, p. 231.

[78] G. Xing, T. Wang, Z. Xie, and W. Jia, "Rendezvous Planning in Mobility-Assisted Wireless Sensor Networks," in *28th IEEE International Real-Time Systems Symposium (RTSS 2007)*, 2007, pp. 311–320.

[79] C. Konstantopoulos, G. Pantziou, D. Gavalas, A. Mpitziopoulos, and B. Mamalis, "A Rendezvous-Based Approach Enabling Energy-Efficient Sensory Data Collection with Mobile Sinks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 23, no. 5, pp. 809–817, May 2012.

[80] G. Pantziou, A. Mpitziopoulos, D. Gavalas, C. Konstantopoulos, and B. Mamalis, "Mobile Sinks for Information Retrieval from Cluster-Based WSN Islands," in *8th International Conference, ADHOC-NOW*, 2009, pp. 213–226.

[81] R. Sugihara and R. K. Gupta, "Improving the Data Delivery Latency in Sensor Networks with Controlled Mobility," in *4th IEEE International Conference, DCOSS*, 2008, pp. 386–399.

[82] S. Ratnasamy *et al.*, "GHT: A Geographic Hash Table for Data-Centric Storage," in *1st ACM iIternational Workshop on Wireless Sensor Networks and Applications - WSNA '02*, 2002, p. 78.

[83] J. Rao and S. Biswas, "Joint Routing and Navigation Protocols for Data Harvesting in Sensor Networks," in *5th IEEE International Conference on Mobile Ad Hoc and Sensor Systems*, 2008,

pp. 143–152.

[84] S. Say, H. Inata, J. Liu, and S. Shimamoto, "Priority-Based Data Gathering Framework in UAV-Assisted Wireless Sensor Networks," *IEEE J. Sensors*, vol. 16, no. 14, pp. 5785–5794, Jul. 2016.

[85] Web, "PmodGPS Reference Manual," *Digilent*. [Online]. Available: https://reference.digilentinc.com/_media/pmod:pmod:pmodgps-gms-u1lp_rm.pdf. [Accessed: 08-Sep-2017].

[86] H. A. Abdulqader, T. Rahman, and C. Leow, "Performance Evaluation of Localization Accuracy for a Log-Normal Shadow Fading Wireless Sensor Network under Physical Barrier Attacks," *IEEE J. Sensors*, vol. 15, no. 12, pp. 30545–30570, Dec. 2015.

[87] E. Damosso, "Digital Mobile Radio Towards Future Generation Systems–Cost," 1999.

[88] N. Priyantha, A. Chakraborty, and H. Balakrishnan, "The Cricket Location-Support System," in *6th Annual International Conference on Mobile Computing and Networking - MobiCom*, 2000, pp. 32–43.

[89] Priya and A. Malik, "A Review on Localization Algorithms in Wireless Sensor Networks," *Int. J. Comput. Sci. Eng. Technol.*, vol. 5, no. 6, pp. 677–682, 2014.

[90] C.-W. Fan, Y.-H. Wu, and W.-M. Chen, "RSSI-Based Localization for Wireless Sensor Networks with a Mobile Beacon," *IEEE J. Sensors*, pp. 1–4, Oct. 2012.

[91] S. Ravindra and S. N. Jagadeesha, "Time of Arrival Based Localization in Wireless Sensor Networks: A Linear Approach," *Int. J. Signal Image Process.*, vol. 4, no. 4, pp. 13–30, Aug. 2013.

[92] F. Liu and W. Jinkuan, "AD-MUSIC for Jointly DOA and DOD Dstimation in Bistatic MIMO Radar System," in *International Conference On Computer Design and Applications*, 2010, pp. 455–458.

[93] H. Jianguo, Z. Lijie, Z. Qunfei, J. Yong, and J. Min, "Performance Analysis of DOA Estimation for MIMO Sonar Based on Experiments," in *IEEE/SP 15th Workshop on Statistical Signal Processing*, 2009, pp. 269–272.

[94] M. Grice, J. Rodenkirch, A. Yakovlev, H. K. Hwang, Z. Aliyazicioglu, and A. Lee, "Direction of Arrival Estimation using Advanced Signal Processing," in *3rd International Conference on Recent Advances in Space Technologies*, 2007, pp. 515–522.

[95] S. Bellofiore, J. Foutz, C. . Balanis, and A. . Spanias, "Smart Antenna System for Mobile Communication Networks Beamforming and Network Throughput," *IEEE Mag. Antennas Propag.*, vol. 44, no. 4, pp. 106–114, Aug. 2002.

[96] P. Ioannides and C. A. Balanis, "Uniform Circular Arrays for Smart Antennas," *IEEE Mag. Antennas Propag.*, vol. 47, no. 4, pp. 192–206, Aug. 2005.

[97] J. Liu, li Li, and W. Huazhi, "Investigation of Different Types of Array Structures for Smart

Antennas," in *International Conference on Microwave and Millimeter Wave Technology*, 2008, pp. 1160–1163.

[98] M. Anwar, "Direction of Arrival in Smart Antenna using different Types of Uniform Array Configuration," *Int. J. Comput. Appl.*, vol. 1, no. 1, pp. 2–6, 2014.

[99] J. A. Srar, K.-S. Chung, and A. Mansour, "Analysis of the RLMS Adaptive Beamforming Algorithm Implemented with Finite Precision," in *2010 16th Asia-Pacific Conference on Communications (APCC)*, 2010, pp. 231–236.

[100] J. A. Srar, K.-S. Chung, and A. Mansour, "LLMS Adaptive Beamforming Algorithm Implemented with Finite Precision," in *in 20th Telecommunications Forum (TELFOR)*, 2012, pp. 303–306.

[101] J. A. Srar, K.-S. Chung, and A. Mansour, "Performance of an LLMS Beamformer in the Presence of Element Gain and Spacing Variations," in *The 17th Asia Pacific Conference on Communications*, 2011, pp. 593–598.

[102] J. A. Srar, K.-S. Chung, and A. Mansour, "LLMS Adaptive Array Beamforming Algorithm for Concentric Circular Arrays," in *Advanced Science and Technology Letters*, 2013, pp. 65–77.

[103] M. A. Yaqoob, F. Tufvesson, A. Mannesson, and B. Bernhardsson, "Direction of Arrival Estimation With Arbitrary Virtual Antenna Arrays using Low Cost Inertial Measurement Units," in *IEEE International Conference on Communications Workshops (ICC)*, 2013, pp. 79–83.

[104] A. Broumandan, T. Lin, A. Moghaddam, D. Lu, J. Nielsen, and G. Lachapelle, "Direction of Arrival Estimation of GNSS Signals Based on Synthetic Antenna Array," in *20th International Technical Meeting of the Satellite Division of the Institute of Navigation ION GNSS*, 2007, pp. 25–28.

[105] W. M. Thomas, R. L. Haupt, and R. A. Monzingo, *Introduction to Adaptive Arrays*, 2nd ed. Institution of Engineering and Technology, 2011.

[106] S. Bellofiore, J. Foutz, C. A. Balanis, and A. Spanias, "Smart Antennas for Wireless Communications," in *IEEE Antennas and Propagation Society International Symposium. 2001 Digest. Held in conjunction with: USNC/URSI National Radio Science Meeting (Cat. No.01CH37229)*, vol. 4, pp. 26–29.

[107] R. D. De Groat, E. M. Dowling, and D. A. Linebarger, "The Constrained MUSIC Problem," *IEEE Trans. Signal Process.*, vol. 41, no. 3, pp. 1445–1449, Mar. 1993.

[108] R. Roy and T. Kailath, "ESPRIT-Estimation of Signal parameters via Rotational Invariance Techniques," *IEEE Trans. Acoust.*, vol. 37, no. 7, pp. 984–995, Jul. 1989.

[109] "Kingdom Vision 2030." [Online]. Available: http://vision2030.gov.sa/en. [Accessed: 22-Aug-2017].

[110] Web, "VISION 2030," *The Saudi-US Trade Group, Inc.*, 2017. [Online]. Available:

http://sustg.com/saudi-arabia-approves-24-5b-for-water-agriculture-and-environment-spending-to-meet-vision-2030-goals/. [Accessed: 14-Sep-2017].

[111] E.-H. Aggoune, S. Alwakeel, M. Alwakeel, and M. Ammad-Uddin, "Wireless Sensor Networks with Dynamic Nodes for Water and Crop Health Management," in *Sensor Networks for Sustainable Development*, CRC Press, 2014, pp. 27–40.

[112] M. Ammad-Uddin, M. Ayaz, A. Mansour, D. Le Jeune, and el. H. M. Aggoune, "Wireless Sensors for Modern Agriculture in KSA: A Survey," in *IEEE 7th International Conference on Computer Science and Information Technology (CSIT)*, 2016, pp. 1–7.

[113] Z. J. Haas, J. Y. Halpern, and Li Li, "Gossip-Based Ad-hoc Routing," *IEEE/ACM Trans. Netw.*, vol. 14, no. 3, pp. 479–491, Jun. 2006.

[114] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks," in *33rd Annual Hawaii International Conference on System Sciences*, 2000, vol. vol.1, p. 10.

[115] D. Mahmood, N. Javaid, S. Mahmood, S. Qureshi, A. M. Memon, and T. Zaman, "MODLEACH: A Variant of LEACH for WSNs," in *2013 8th International Conference on Broadband and Wireless Computing, Communication and Applications*, 2013, pp. 158–163.

[116] Y. Wang, I. Henning, X. Li, and D. Hunter, "SOTP: A Self-Organized TDMA Protocol for Wireless Sensor Networks," in *2006 Canadian Conference on Electrical and Computer Engineering*, 2006, pp. 1108–1111.

[117] C. Yun and Y. Lim, "ASO-TDMA: Ad-hoc Self-Organizing TDMA Protocol for Shipborne Ad-hoc Networks," *EURASIP J. Wirel. Commun. Netw.*, vol. 12, no. 1, pp. 320–328, Dec. 2015.

[118] T. Gaugel, J. Mittag, Hartenstein, H. Papanastasiou, S. Strom, and E. G., "In-depth Analysis and Evaluation of Self-Organizing TDMA," in *IEEE Vehicular Networking Conference*, 2013.

[119] M. Manzano, F. Espinosa, Á. M. Bravo-Santos, E. Santiso, I. Bravo, and D. Garcia, "Dynamic Cognitive Self-Organized TDMA for Medium Access Control in Real-Time Vehicle to Vehicle Communications," *J. Math. Probl. Eng.*, vol. 2013, pp. 1–13.

[120] I.-K. Rhee, J. Lee, J. Kim, E. Serpedin, and Y.-C. Wu, "Clock Synchronization in Wireless Sensor Networks: An Overview," *IEEE J. Sensors*, vol. 9, no. 1, pp. 56–85, Jan. 2009.

[121] B. Sundararaman, U. Buy, and A. D. Kshemkalyani, "Clock Synchronization for Wireless Sensor Networks: a Survey," *J. Ad Hoc Networks*, vol. 3, no. 3, pp. 281–323, May 2005.

[122] P. Ferrari, A. Flammini, D. Marioli, and A. Taroni, "IEEE 1588-Based Synchronization System for a Displacement Sensor Network," *IEEE Trans. Instrum. Meas.*, vol. 57, no. 2, pp. 254–260, 2008.

[123] C. Tian, H. Jiang, X. Liu, X. Wang, W. Liu, and Y. Wang, "Tri-Message: A Lightweight Time

Synchronization Protocol for High Latency and Resource-Constrained Networks," in *2009 IEEE International Conference on Communications*, 2009, pp. 1–5.

[124] WikiPedia, "Free-space path loss," *WikiPedia*. [Online]. Available: https://en.wikipedia.org/wiki/Free-space_path_loss. [Accessed: 21-Sep-2017].

[125] WEB, "2.4 GHz IEEE 802.15.4 / ZigBee-ready RF Transceiver," *Texas Instruments Incorporated*. [Online]. Available: http://www.ti.com/lit/ds/symlink/cc2420.pdf. [Accessed: 02-May-2017].

[126] Web, "CC1021 Single Chip Low Power RF Transceiver for Narrowband Systems," *Texas Instruments Incorporated*. [Online]. Available: http://www.ti.com/lit/ds/swrs045e/swrs045e.pdf. [Accessed: 06-Oct-2017].

[127] Web, "RF24L01 2.4GHz Radio/Wireless Transceivers How-To," *arduino info.wikispaces.com*. [Online]. Available: https://arduino-info.wikispaces.com/Nrf24L01-2.4GHz-HowTo. [Accessed: 06-Oct-2017].

[128] M. Ammad-Uddin, A. Mansour, D. Le Jeune, E. H. M. Aggoune, and M. Ayaz, "UAV Routing Protocol for Crop Health Management," in *24th European Signal Processing Conference (EUSIPCO)*, 2016, pp. 1818–1822.

[129] M. Ammad-Uddin, A. Mansour, and D. Le Jeune, "Agriculture Internet of Things: AG-IOT," in *27th International Telecommunication Networks and Applications Conference (ITNAC)*, 2017.

[130] R. Schmidt, "Multiple Emitter Location and Signal Parameter Estimation," *IEEE Trans. Antennas Propag.*, vol. 34, no. 3, pp. 276–280, Mar. 1986.

[131] "Advanced Radar Systems," *COURSE SYLLABUS (Introduction to Naval Weapons Engineering)*. [Online]. Available: https://fas.org/man/dod-101/navy/docs/es310/advradrsys/AdvRadr.htm. [Accessed: 15-Sep-2017].

[132] Web, "CX-20 Path Finder," *Cheerson*. [Online]. Available: http://www.cheersonhobby.com/en-US/Home/ProductDetail/108. [Accessed: 07-Oct-2017].

[133] E. Geck and P. Connolly, "Air Craft DataSheet," *Air Craft*. [Online]. Available: https://www.aircraft-world.com/Datasheet/en/hp/emeter/hp-propconstants.htm. [Accessed: 15-Sep-2017].

[134] Web, "A2212/13T Technical Data," *BatteryH eated Clothing*. [Online]. Available: http://www.batteryheatedclothing.com/pages/a2212-13t-technical-data.html. [Accessed: 15-Sep-2017].

[135] P. Kubicek, J. Kozel, R. Stampach, and V. Lukas, "Prototyping the Visualization of Geographic and Sensor Data for Agriculture," *J. Comput. Electron. Agric.*, vol. 97, pp. 83–91, Sep. 2013.

[136] A. Pardossi *et al.*, "Root Zone Sensors for Irrigation Management in Intensive Agriculture," *IEEE J. Sensors*, vol. 9, no. 4, pp. 2809–2835, Apr. 2009.

[137] R. Cardell-Oliver, M. Kranz, K. Smettem, and K. Mayer, "A Reactive Soil Moisture Sensor Network: Design and Field Evaluation," *Int. J. Distrib. Sens. Networks*, vol. 1, no. 2, pp. 149–162, 2005.

[138] C. M. Angelopoulos, S. Nikoletseas, and G. C. Theofanopoulos, "A Smart System for Garden Watering Using Wireless Sensor Networks," in *9th ACM International Symposium on Mobility Management and Wireless Access - MobiWac '11*, 2011, p. 167.

[139] R. Beckwith, D. Teibel, and P. Bowen, "Report from the Field: Results from an Agricultural Wireless Sensor Network," in *29th Annual IEEE International Conference on Local Computer Networks*, 2004, pp. 471–478.

[140] J. Geipel, M. Jackenkroll, M. Weis, and W. Claupein, "A Sensor Web-Enabled Infrastructure for Precision Farming," *ISPRS Int. J. Geo-Information*, vol. 4, no. 1, pp. 385–399, Mar. 2015.

[141] D. D. Chaudhary, S. P. Nayse, and L. M. Waghmare, "Application of Wireless Sensor Networks for Greenhouse Parameter Control in Precision Agriculture," *Int. J. Wirel. Mob. Networks*, vol. 3, no. 1, pp. 140–149, Feb. 2011.

[142] X. Dong, M. C. Vuran, and S. Irmak, "Autonomous Precision Agriculture Through Integration of Wireless Underground Sensor Networks with Center Pivot Irrigation Systems," *J. Ad Hoc Networks J.*, vol. 11, no. 7, pp. 1975–1987, 2013.

[143] S. Roy and S. Bandyopadhyay, "A Test-bed on Real-time Monitoring of Agricultural Parameters using Wireless Sensor Networks for Precision Agriculture," in *First International Conference on Intelligent Infrastructure the 47th Annual National Convention at Computer Society of India*, 2013.

[144] T. Ahonen, R. Virrankoski, and M. Elmusrati, "Greenhouse Monitoring with Wireless Sensor Network," in *IEEE/ASME International Conference on Mechtronic and Embedded Systems and Applications*, 2008, pp. 403–408.

[145] S. M. Abd El-kader and B. M. Mohammad El-Basioni, "Precision Farming Solution in Egypt using the Wireless Sensor Network Technology," *Egypt. Informatics J.*, vol. 14, no. 3, pp. 221–233, Nov. 2013.

[146] J. Lloret, I. Bosch, S. Sendra, and A. Serrano, "A Wireless Sensor Network for Vineyard Monitoring That Uses Image Processing," *IEEE J. Sensors*, vol. 11, no. 12, pp. 6165–6196, Jun. 2011.

[147] Web, "Sensatrack," *Cannon Water Technology*. [Online]. Available: sensatrack.com. [Accessed: 13-Oct-2017].

[148] Web, "pycno.co.uk," *PYCNO*. [Online]. Available: pycno.co.uk. [Accessed: 13-Oct-2017].

[149] Web, "New! Stevens Smart sensors.," *Remote Communications Data Collection Platforms*.

[Online]. Available: stevenswater.com. [Accessed: 13-Oct-2017].

[150] Web, “Sol Chip,” *Sol Chip Ltd*. [Online]. Available: http://sol-chip.com/. [Accessed: 13-Oct-2017].

[151] Web, “Landscape Technologie,” *Landscape Technologies*. [Online]. Available: http://www.landscapetechnologies.net/. [Accessed: 13-Oct-2017].

[152] Web, “IRRO Mesh Solar Powered Wireless Data Logging System,” *Irrometer*. [Online]. Available: www.irrometer.com. [Accessed: 15-Sep-2017].

[153] Web, “Hydraulic Conductivity in Plant Stems,” *ICT International*. [Online]. Available: http://www.ictinternational.com/casestudies/hydraulic-conductivity-in-plant-stems/. [Accessed: 28-Feb-2016].

[154] Web, “world class dataloggers,” *Campbell Scientific*. [Online]. Available: www.campbellsci.com. [Accessed: 18-Oct-2017].

[155] Web, “Welcome to Satlantic,” *Satlantic*. [Online]. Available: www.satlantic.com. [Accessed: 18-Oct-2017].

[156] Web, “Welcome to Sea-Bird Scientific,” *Sea-Bird Scientific*. [Online]. Available: https://sea-birdscientific.co. [Accessed: 18-Oct-2017].

[157] Web, “Ocean Space,” *Kongsberg Gruppen*. [Online]. Available: www.kongsberg.com. [Accessed: 18-Oct-2017].

[158] Web, “CMOSIS, Member of the AMS Group,” *ams AG*. [Online]. Available: http://www.cmosis.com/. [Accessed: 18-Oct-2017].

[159] Web, “High precision sensors, measurement devices and systems,” *Micro-Epsilon*. [Online]. Available: http://www.micro-epsilon.com/. [Accessed: 18-Oct-2017].

[160] Web, “Shop the #1 Weather Station Brand in North America,” *AcuRite*. [Online]. Available: www.acurite.com. [Accessed: 18-Oct-2017].

[161] Web, “Neues von S+S Regeltechnik,” *Regeltechnik GmbH*. [Online]. Available: www.spluss.eu. [Accessed: 18-Oct-2017].

[162] Web, “Mission-critical sensors & electrical protection,” *Sensata Technologies*. [Online]. Available: http://www.sensata.com/. [Accessed: 18-Oct-2017].

[163] Web, “Explore Featured Products,” *Silicon Laboratories*. [Online]. Available: https://www.silabs.com/. [Accessed: 18-Oct-2017].

[164] Web, “libelium,” *Libelium Comunicaciones*. [Online]. Available: www.libelium.com. [Accessed: 18-Oct-2017].

[165] Web, “RigStat Environmental Monitoring System,” *RigStat, LP*. [Online]. Available: http://www.rigstat.com/. [Accessed: 18-Oct-2017].

[166] Web, "GEMS Sensors Controls," *Gems Sensors, Inc*. [Online]. Available: https://www.gemssensors.com/. [Accessed: 18-Oct-2017].

[167] Web, "https://www.stellartech.com/," *LORD Corporation*. [Online]. Available: https://www.stellartech.com/. [Accessed: 18-Oct-2017].

[168] Web, "estion Assistée des équipements par RTLS," *Localiser Pour Simplifier*. [Online]. Available: http://www.purelink.ca. [Accessed: 18-Oct-2017].

[169] Web, "Security Systems," *Southwest Microwave, Inc*. [Online]. Available: http://www.southwestmicrowave.com/. [Accessed: 18-Oct-2017].

[170] Web, "MTN Sensors We Sense a Green Future," *Nexceris*. [Online]. Available: https://www.ntmsensors.com/. [Accessed: 18-Oct-2017].

[171] Web, "Smart Home Automation Superstore," *SMARTHOME*. [Online]. Available: www.smarthome.com. [Accessed: 18-Oct-2017].

[172] Web, "Ecotech," *Ecotech groups*. [Online]. Available: https://www.ecotech.com. [Accessed: 18-Oct-2017].

[173] Web, "https://www.pasco.com/prodCatalog/PS/PS-2195_pasport-salinity-sensor/index.cfm," *Foothills Blvd*. [Online]. Available: https://www.pasco.com/prodCatalog/PS/PS-2195_pasport-salinity-sensor/index.cfm. [Accessed: 18-Oct-2017].

[174] Web, "Plant Science," *Bio-Science, Inc.* [Online]. Available: https://cid-inc.com. [Accessed: 20-Jul-2017].

[175] Web, "AQM 65, new compact air monitoring station," *Aeroqual*. [Online]. Available: https://www.aeroqual.com/aqm-65-air-monitoring-station. [Accessed: 18-Oct-2017].

[176] Web, "Vernier Software & Technology," *Vernier Technology*. [Online]. Available: https://www.vernier.com/. [Accessed: 20-Jul-2017].

[177] Web, "TPS-2 Portable Photosynthesis System," *PP Systems*. [Online]. Available: http://ppsystems.com/tps2-portable-photosynthesis-system/. [Accessed: 18-Oct-2017].

[178] Web, "Stevenswater," *Systems, Stevens Water Monitoring*. [Online]. Available: www.stevenswater.com. [Accessed: 20-Jul-2017].

[179] Web, "Agriculture," *Davis Instruments*. [Online]. Available: www.davisnet.com. [Accessed: 20-Jul-2017].

[180] Web, "Introducing The Smart Outlet," *Grid Connect*, 2017. [Online]. Available: https://www.connectsense.com/. [Accessed: 20-Jul-2017].

[181] Web, "Vaisala Environment," *Vaisala*, 2007. [Online]. Available: http://www.vaisala.com. [Accessed: 15-Sep-2017].

[182] "Agristore." [Online]. Available:

http://www.agrisupportonline.com/store/Phytech/technical_data/de-1m.htm. [Accessed: 28-Feb-2016].

[183] J. Henao and C. Baanante, “Agricultural production and soil nutrient mining in africa: implications for resource conservation and policy development,” 2006. [Online]. Available: http://agris.fao.org/agris-search/search.do?recordID=GB2013202609. [Accessed: 01-Jun-2016].

[184] Web, “What is a Linear Variable Displacement Transducer,” *OMEGA*. [Online]. Available: https://www.omega.com/technical-learning/linear-variable-displacement-transducers.html. [Accessed: 02-May-2017].

[185] C.-C. Weng, C.-W. Chen, and C.-J. Ku, “A Minimum Transmission Energy Consumption Routing Protocol for User-Centric Wireless Networks,” in *EEE International Symposium on System Integration (SII)*, 2011, pp. 1143–1148.

[186] A. Harchowdhury, B. K. Sarkar, K. Bandyopadhyay, and A. Bhattacharya, “Generalized Mechanism of SOTDMA and Probability of Reception for Satellite-Based AIS,” in *5th International Conference on Computers and Devices for Communication (CODEC)*, 2012, pp. 1–4.

[187] Web, “All about AIS,” *Joomla Templates*. [Online]. Available: http://www.allaboutais.com/index.php/en/technical-info/transmission-types/105-ais-technicals/technical-lfundamentals/92-cstdma. [Accessed: 01-Jan-2017].

[188] Web, “Mission Planner Home,” *ArduPilot Dev*, 2016. [Online]. Available: http://ardupilot.org/planner/. [Accessed: 08-Oct-2017].

UAV Routing Protocol (URP) pour la gestion de la santé des cultures

Résumé :
Les réseaux de capteurs sans fil sont maintenant un moyen crédible de collecte de données sur les cultures. L'installation d'une structure de communication fixe pour relayer les données surveillées depuis la tête de grappe jusqu'à sa destination finale peut être soit impraticable en raison de la topologie du terrain, soit prohibitive en raison du coût initial élevé. Une solution plausible consiste à utiliser des véhicules aériens sans pilote (UAV) comme moyen alternatif de collecte de données et de contrôle de supervision limité de l'état des détecteurs. Dans cet article, nous considérons le cas des parcelles agricoles disjointes comprenant chacune des grappes de capteurs, organisées de manière prédéterminée en fonction des objectifs d'élevage. Cette recherche vise à trouver une solution optimale pour la recherche de UAV et la collecte de données à partir de tous les capteurs installés dans un champ de culture. En outre, le protocole de routage des capteurs tiendra compte d'un compromis entre la gestion de l'énergie et les frais généraux de diffusion des données.
Le système proposé est évalué en utilisant un modèle simulé et il devrait trouver une classe parmi toutes les sous-considération.

Mots-clés : Agriculture intelligente ; Protocole de routage ; Agriculture de précision ; Collecte de données ; Réseau de capteurs sans fil, UAV.

UAV Routing Protocol (URP) for Crop Health Management

Abstract :
Wireless sensor networks are now a credible means for crop data collection. The installation of a fixed communication structure to relay the monitored data from the cluster head to its final destination can either be impractical because of land topology or prohibitive due to high initial cost. A plausible solution is to use Unmanned Aerial Vehicles (UAV) as an alternative means for both data collection and limited supervisory control of sensors status. In this paper, we consider the case of disjoint farming parcels each including clusters of sensors, organized in a predetermined way according to farming objectives. This research focuses to drive an optimal solution for UAV search and data gathering from all sensors installed in a crop field. Furthermore, the sensor routing protocol will take into account a tradeoff between energy management and data dissemination overhead.
The proposed system is evaluated by using a simulated model and it should find out a class among all under consideration.

Keywords : Smart farming ; Routing protocol ; Precision agriculture ; Data gathering ; Wireless sensor network, UAV.