Proper Deployment of Sensors in a Weather Station

List of Acronyms used in this paper

1. AWS, Automatic Weather Station
2. UNMA, Uganda National Meteorological Authority
3. WSN, Wireless Sensor Network
4. WSS-D, Wireless Sensor Smart Deployment Method.
5. WIMEA-ICT, Weather Information Management in East Africa

**Abstract**

# Uganda is experiencing an increase in the frequency and intensity of drought, floods and landslides which have significantly impacted the livelihoods of local communities especially those that largely depend on agriculture. Experts have blamed these extreme weather occurrences on the changing rainfall patterns. Farmers can no longer rely on the traditional seasonal predictions in order to make well informed decisions on planting and harvesting, therefore there is need for accurate and timely weather information. However, there is insufficient coverage of the country by weather stations as there are a few fully functional ones which leads to poor representation of some regions in the weather forecasts. WIMEA-ICT[14] in collaboration[15] with National Meteorological Authority of Uganda is a project whose aim is to improve the accuracy of and access to weather information by communities in the East African region through suitable Information and Communication Technologies (ICTs) for increased productivity and safety. The project is designing a robust and affordable Automated Weather Station (AWS) based on the Wireless Sensor Network (WSN) technology. These AWSs rely on Radio Sensor Version 2 (RSS2) nodes to collect, process, store and transmit weather parameters to a remote repository. To achieve the whole goal of AWS by WIMEA-ICT, WSN is pivotal in dispensation of data. This survey looks at the way these sensors have been deployed and suggests a much better deployment method of Wireless Sensor Smart Deployment in which sensors can be deployed taking into account the obstacles in the target area.

**1.0 Introduction**

Automatic Weather Stations (AWSs)[8] collect and transmit weather data without human intervention, enabling them to operate in remote areas. AWSs, which use Wireless Sensor Networks (WSNs) technology, in which distributed sensors collect varying parameters at predetermined intervals from various sensors deployed in a given area which is the focus of this paper. While in remote deployments, WSNs face challenges such as coverage [12], packet loss and limited energy among others, which lower their health and life time. These losses are partly attributed to how the sensors have been deployed in a target area.

The performance of a wireless sensor network is greatly influenced by the process of deploying the sensor nodes. The issue of deployment and positioning of sensor nodes in a WSN is a strategy which is used in defining the topology of the network, the number and the position of the sensor nodes. Quality monitoring, connectivity, and power consumption are also directly affected by the network topology. The problem of optimal placement of nodes is proven NP-hard for most deployment formulations[16]

[17] The WIMEA-ICT proposed the implementation of the of WSN in the AWS in an optimized way to deal with challenges that WSN face such as low power by using close-Layer cycling[18] which simply suggests how to optimize WSNs but not how these are deployed on the weather station yet their improper deployment causes another problem in relation to data collection.

In this survey we consider aspects of determining the best and suitable method of deploying sensors in a weather station in consideration of the obstacles. Specifically, We study and undertake a survey of how sensors in a WSN have been deployed in various weather stations and propose the best deployment method which can be used by WIMEA-ICT in their AWS project in collaboration with UNMA.

# **1.1 What sensors are in a weather station?**

AWS has several key components[13]. It has the sensor group, which measures the weather phenomena, the display, which is located nearby to show the weather data, the wireless transmitter for relaying data to the display (and possibly an internet bridge) and the power system, which often consists of rechargeable batteries and a small solar panel. These sensors do the rainfall, humidity, wind, temperature, pressure sensing among others



**1.2 The problem statement**

The proposed project of WIMEA-ICT includes the deployment of Sensors[18], they propose that the distance from the sensors to the measurements point determines the length of the conductor and they state that this clearly affects data integrity. They further propose that all sensors will be connected directly to a wireless-enabled microprocessor unit on a Printed Circuit Board (PCB), which will capture and transmit data to the gateway. For sensors at a distance from the PCB, such as the soil moisture sensor, the PCB tracks will be extended by flexible wire to the sensor location. And this was done to avoid contact of these wires with the ground as much as possible. The survey was majorly focused on deploying sensors in such a way that they didn’t have any constraints on their design that they were proposing.

Therefore sensors in the project were deployed without considering any of the deployment methods which would make WSN difficult to achieve the goal of the project of transmitting the collected data. The survey paper proposes a best deployment strategy that takes in account the obstacles when deploying the sensors to foster integrity of data that is being transmitted from the weather stations.

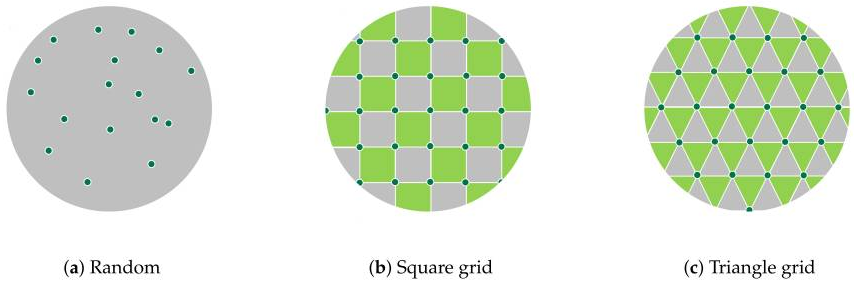
**2.0 Background and Related Work**

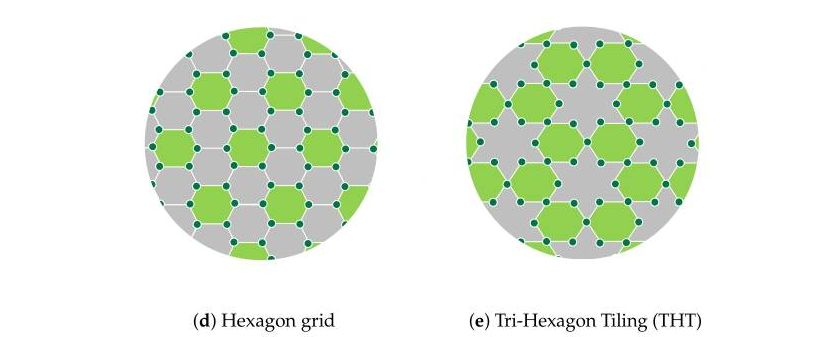
**2.1 Background**

Sensor deployment can be classified based on placement strategy into random and deterministic deployment [[28](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7765373/" \l "B28-sensors-20-07191)].

#### **2.1.1. Random Deployment**

In random deployment, sensors are randomly scattered over the region of interest to gather the target information. Figure 2a shows an example of random placement. It is suitable for regions where human existence is difficult (e.g., disaster areas, battlefields, air pollution, and forest fires) [[29](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7765373/" \l "B29-sensors-20-07191)]. Random sensor deployment is preferred in many WSN applications due to the simplicity of the sensor distribution [[15](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7765373/" \l "B15-sensors-20-07191)]. As a drawback, however, this method leads to uneven connectivity with critical sensors, which results in a network which is non-robust to sensor failure [[30](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7765373/" \l "B30-sensors-20-07191)].





#### **2.1.2. Deterministic (Pre-determined)Deployment**

In deterministic deployment, sensors are placed on the region of interest based on a certain geometrical structure [[31](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7765373/" \l "B31-sensors-20-07191)]. Examples of this type of deployment are square, triangle, and hexagon grids, and tri-hexagon tiling (THT), as shown in [Figure 2](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7765373/figure/sensors-20-07191-f002/)b–e, respectively [[15](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7765373/" \l "B15-sensors-20-07191)].

A theoretical analysis in [[32](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7765373/" \l "B32-sensors-20-07191)] proved that a hexagonal structure can provide a high coverage area with low energy consumption using a minimum number of sensors. Tri-hexagon tiling deployment was proposed to combine the advantages of the triangle and hexagon deployment methods. In terms of energy consumption, the THT deployment outperforms the square and hexagon deployments [[33](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7765373/" \l "B33-sensors-20-07191)].

**2.2 Related work**

This section presents previous work related to sensor deployment on wireless sensor networks. The work varies in terms of deploying objectives and strategies, based on the application of a wireless sensor network.

There are many types of research that have been done to build a weather station and weather forecasting. The paragraphs below explain some of these works.

In work [1] propose a weather station using sensors for measuring the pressure luminosity. Also the measure the temperature and the humidity and placing these sensors in a board with a micro-controller. The station can be controlled by mobile phones through the SMS service.

[2] presents an automated weather station for real-time and local measurements, based on an embedded system that continuously measures several weather factors such as temperature, humidity, barometric pressure, wind speed, wind direction, and rainfall. This weather station consists of two parts which are located indoor and outdoor and connected together wirelessly.

[3] aims to build a weather station as a second prototype for measurements of the conditions of the weather such as the temperature, pressure and the humidity. They used variance analysis to validate the weather station operation. Also, they used an experimental design which was called r&R. a number of sensors selected such as TMP36, RHT03, and BMP085 with Arduino UNO board in the weather station and they used a digital hygrometer.

[4] develop weather stations for monitoring weather in real-time. They used a mobile application depending on the Automatic Weather Station (AWS). The system contains several sensors connected to the AWS to collect data. They stored the data on the web server. The mobile application which uses the Android system is able to read the data and display it from the web server.

[5] shows a low-cost weather station capable of measuring the temperature of the air. Also it measures the humidity, pressure, and is able to detect the wind speed and in which direction and the amount of the rainfall. A LiPo battery and four solar panels were used to power up the complete system. The weather station communicates through Wi-Fi connection allowing the user to access the data remotely.

[10] Focusing on outdoor deployments , the authors evaluated the link quality in three different environments: two tunnels (one with vehicular traffic and another without vehicular traffic) and a vineyard. Their study shows that the link quality in the tunnels is significantly different from other classical WSNs. Indeed, the authors observed that, in tunnels, links are generally stable and long-range compared to the vineyard deployment, with the tunnel behaving as a waveguide. While the experiments described in [10,11] do not consider the scenario of ground-level deployment, an important conclusion

[9]This typically suggests the use of a tripod when deploying their weather stations. They mount their sensors either on the tripod or run cables to nearby locations. If using a tripod, it must be secured so that it will withstand the maximum possible winds for the site. When assembling the tripod, stay organised and build as much of the weather station indoors as possible. This cuts down on losing important small parts. At the site, lay down a tarp and place all tools and components on top. Some researchers mount sensors on the upper mast of the tripod before it is attached to the lower mast. This makes it easier to attach the sensors. To do this, stand the upper mast upright and use zip ties to attach it to one leg brace and one leg of the tripod. Once the cross arm and sensors are installed, remove the zip ties and place the upper mast on top of the lower mast. Then make final sensor height adjustments and levelling. To help prevent corrosion of sensors and sensor ports, it is good practice to spray WD40 or a similar lubricant on sensor ports prior to plugging in the sensors.

The limitations are summarised as follows, based on the work presented in this section.

* The focus is on deploying sensor nodes in a WSN. The aspect of sink distribution is neglected.
* The existence of obstacles in a sensing area is not considered when designing a sensor deployment algorithm.
* [1][2][3][4][5][6] The studies report on the deployment of sensors without explicitly describing the sensor deployment methodology they used. Well [1] looks at a small weather station that could deploy a few sensors but nonetheless the method of deployment is so important to avoid problems in case the project scales up in future.

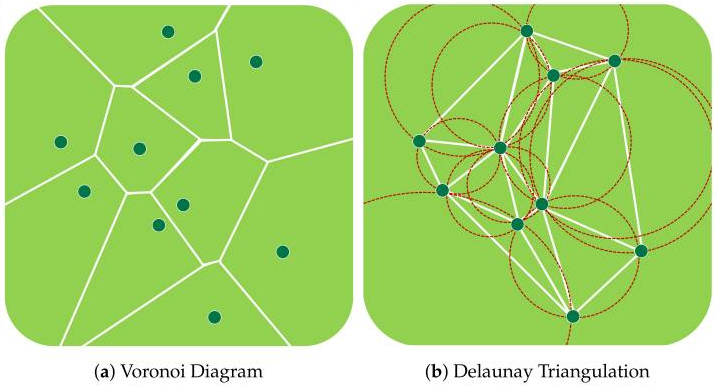
**3.0. Proposed Deployment Strategy**

### **3.1 Preliminaries**

In the preliminary section, the Voronoi diagram, Delaunay triangulation, and *k*-means clustering algorithm are introduced to convey a good understanding of our WSS-D approach.

#### **3.1.1. Voronoi Diagram**

The Voronoi diagram was proposed by Rene Descartes in 1644. For a set of random points, P, a Voronoi diagram VD(P) is set in Voronoi regions, called Voronoi cells (i.e., one for each point in P) [[41](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7765373/" \l "B41-sensors-20-07191)]. The locations inside a Voronoi cell are closer to their corresponding point than the other points [[42](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7765373/" \l "B42-sensors-20-07191)] (Figure 3a).

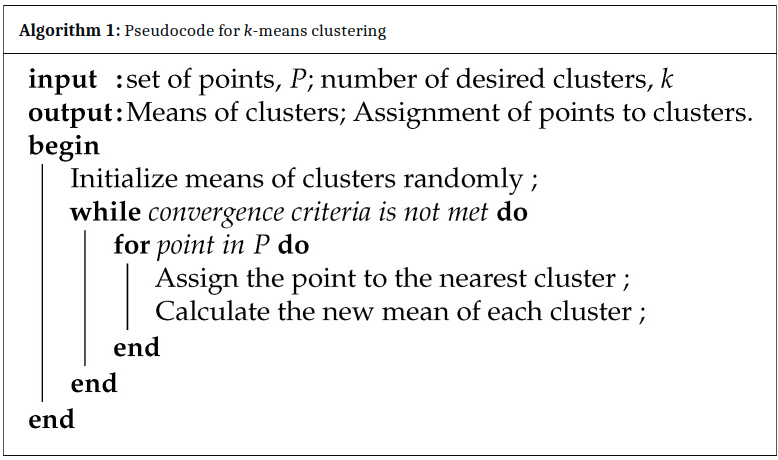


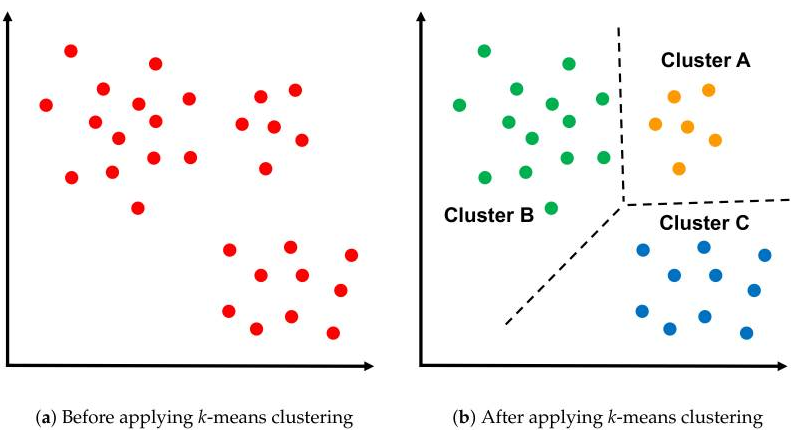
#### **3.1.2. Delaunay Triangulation**

Delaunay triangulation (DT) is a triangular mesh that links a group of points in a plane [[43](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7765373/" \l "B43-sensors-20-07191)]. DT was proposed by Boris Delaunay in 1934. For a set of random points, P, the Delaunay triangulation DT(P) is dual to its Voronoi diagram (Figure 3b). The Delaunay triangulation of P is a triangulation such that no point in P exists inside the circumcircle of any DT(P) triangle [[41](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7765373/" \l "B41-sensors-20-07191)].

#### **3.1.3. *k*-Means Clustering Algorithm**

The *k*-means clustering algorithm is an unsupervised machine learning algorithm that aims to divide n observations into clusters (Figure 4). An observation is assigned to the cluster with the closest mean. The *k*-means algorithm has three stages: initialization, computation, and convergence (Algorithm 1) [[44](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7765373/" \l "B44-sensors-20-07191)].





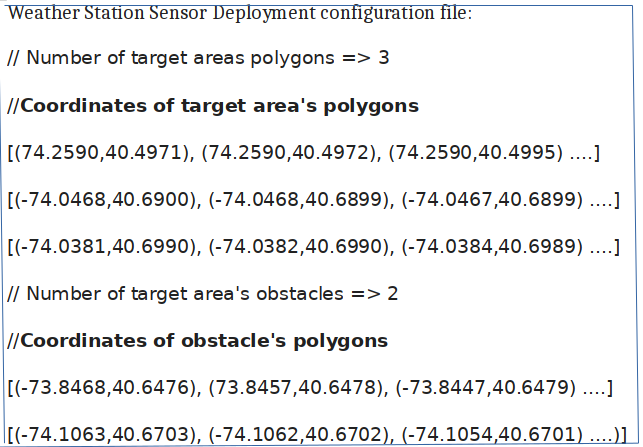
*Figure 4, Applying k-means Algorithm on the cluster*

### **3.2. WSS-D Algorithm**

This section explains the proposed sensor deployment scheme (i.e., WSS-D) in detail. The WSS-D algorithm was designed for IoT smart cities with obstacles. It focuses on determining the sensor locations and identifying the suitable sink positions based on the *k*-means clustering algorithm, because a proper selection of sink locations positively affects the WSN performance. Accordingly, the WSS-D strategy consists of three main phases: configuration , sensor deployment and sinks deployment

#### **3.2.1. Configuration Phase**

A weather station sensor configuration file is read in the configuration phase. This file contains information about the polygons that make up the sensor target deployment area map and the obstacles within the range. Figure 5 shows an example of a weather station sensor configuration file.

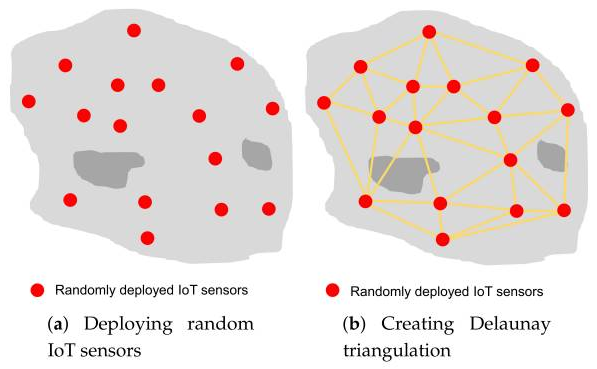


*Figure 5; Example of weather station sensor deployment configuration file*

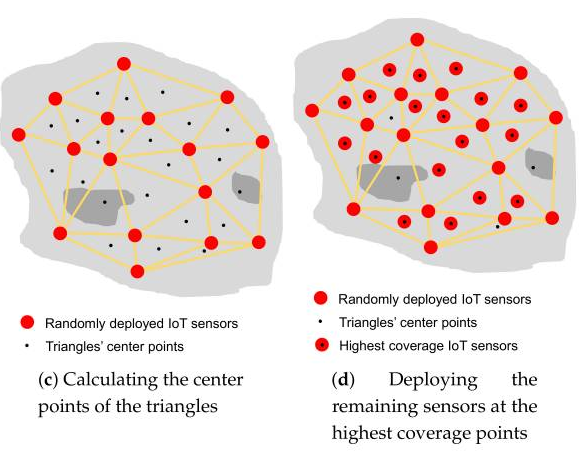
#### **3.2.2. Sensors Deployment Phase**

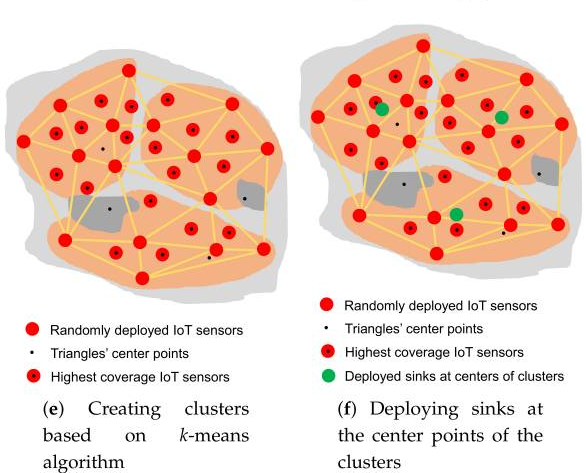
In this phase, IoT sensors are deployed throughout a smart city. The sensor deployment phase includes the three following steps:

1. Random Location Generation: In this step, a set of random places is generated based on the assumed percentage of random points (i.e., 50%). Subsequently, IoT sensors are placed on random points, provided that they are within the boundaries of a smart city and not within any obstacle. Figure 6a illustrates the random location generation step in a simple example.



1. Delaunay Triangulation Creation: In this step, depending on the locations of the sensors deployed in the previous step, triangles are created based on Delaunay triangulation (Figure 6b). A center point is then calculated for each triangle (Figure 6c).



1. Coverage Evaluation Step: In this step, the triangle center points are evaluated based on the coverage percentage. Sensors will be deployed in the center points with the highest coverage ratio, depending on the available number of IoT sensors (Figure 6d). Therefore, deploying sensors in areas that have a small number of sensors (including areas around obstacles) has a higher priority than other points.

#### **3.2.3. Sinks Deployment Phase**

This phase aims to deploy sinks to be close to all sensor nodes to achieve a high WSN performance. To achieve this goal, the *k*-means clustering algorithm is applied to sensors deployed in the previous phase. The sensors are divided into clusters (i.e., three clusters in this example) equal to the desired number of sinks (Figure 6e). The sinks are then placed on the center points of the clusters (Figure 6f). If the center point is located on an obstacle, it will be moved to the nearest point on the obstacle boundary. In this way, our WSS-D algorithm ensures that areas around obstacles are covered and there are no disconnected sensors. Algorithm 2 shows the pseudocode of the WSS-D algorithm. The code illustrates the three main phases of the WSS-D deployment scheme.

Algorithm-2: Pseudocode for WSS-D

**Input**:

n:= Number of sensors;

m := Number of sinks;

p := Percentage of random sensors;

**Output**:

Sloc := Optimal location of sensors;

skloc := Optimal location of sinks;

**begin**

/\* **Configuration Phase** \*/:

Read a weather station sensor deployment - WSSD configuration file

Assign values to WSSD polygon WSSDpol and target area obstacles TAobs

/\* **Senors Deployment Phase** \*/:

/\* Random Locations Generation Step\* /

Num Rand = p\*n

**while** Num Rand !=0 do

Generate random point r

**if** r within WSSDpol and not within TAobs then

Add r to TAobs

Decrease n by 1

**end**

**end**

/\* Delaunay Triangulation Creation Step\*/

Generate Delaunay triangulation Dtri based on Sloc

**for** each Dtri do

Calculate center point tricen

**end**

/\* Coverage Evaluation Step\*/

Evaluate each tricen based on coverage percentage

Sort tricen in descending order based on corresponding evaluation values

i=0

**while** n !=0 **do**

**if** tricen (i) within sopol and not within scobs then

Add tricen (i) to Sloc

Increase i by 1

Decrease n by 1

**end**

**end**

/\* Sinks Deployment Phase \*/:

NumClusters = m

Calculate k-means clusters of Sloc

**for** each cluster do

**if** clustercen within scpol and not within scobs then

Add clustercen to skloc

**else**

Move clustercen to nearest point on obstacle boundary

Add the new location of clustercen to skloc

**end**

**end**

return Sloc, skloc

**end**

**6.0 Conclusion and Future work**

Sensor deployment in WSNs is an important issue that needs to be addressed. This study proposed a three-phase deployment strategy, called WSS-D, for WSNs in AWS. The WSS-D is a deployment approach suitable for determining the locations of sensors and sinks in a sensing area with obstacles. The first phase involves reading a configuration file including information about the polygons that make up a weather station map and the obstacles within the station. The second phase involves sensor deployment, comprising three steps: random location generation, Delaunay triangulation creation, and coverage evaluation. The last phase is sink distribution, based on the *k*-means clustering machine learning technique. As a result, the proposed scheme outperformed the random and four regular deployment strategies in terms of the end-to-end delay and the area coverage, which are the most important performance metrics in WSN applications.

**7.0 References**

[1] Mircea Popa Embedded Weather Station with Remote Wireless Control 19th Telecommunications forum TELFOR 2011.

[2] Mohamad Farhat, Mohammed Abdul-Niby, Mohammed Abdullah, and Abdelbaset Nazzal

“A Low Cost Automated Weather Station for Real Time Local Measurements” , Engineering Technology & Applied Science Research, 2017.

[3] Gabriel Piñeres-Espitia, Alejandro Cama-Pinto, Daniel De La Rosa Morrón, Francisco Estevez, And Dora Cama-Pinto, “Design of a low cost weather station for detecting environmental changes ” , Revista Espacios, 2017.

[4] Aris Munandar, Hanif Fakhrurroja, Muhammad Ilham Rizqyawan, Rian Putra Pratama, Jony Winaryo Wibowo, and Irfan Asfy FakhryAnto, “Design of Real-time Weather Monitoring System Based on Mobile Application using Automatic Weather Station International Conference on Automation”, Cognitive Science, Optics, Micro Electro Mechanical System, and Information Technology (ICACOMIT),2017.

[5] Diego A. Aponte-Roa, Luis Benitez Montalvan, Christopher Velazquez, Albert A. Espinoza, Luis Feliciano Velazquez, and RicardoSerrano, “ Evaluation of a Low-Cost, Solar-Powered Weather Station for Small-Scale Wind Farm Site Selection”, IEEE, 2018.

[6]https://medium.com/deep-math-machine-learning-ai/chapter-4-decision-trees-algorithms-b93975f7a1f1

*[7]. Deploying Weather Stations: A Best Practices Guide*. (n.d.). CiK Solutions. Retrieved May 4, 2022, from https://www.cik-solutions.com/content/files/en\_cik\_weather\_stations\_guide.pdf

*[8] THE WIMEA-ICT AUTOMATIC WEATHER STATION (AWS) March, 2021*. (n.d.). WIMEA-ICT.

[9]. Mottola L., Picco G.P., Ceriotti M., Gună Ş., Murphy A.L. Not All Wireless Sensor Networks Are Created Equal: A Comparative Study on Tunnels. *ACM Trans. Sens. Netw.* 2010;7:15. doi: 10.1145/1824766.1824771.

[10] Srinivasan K., Dutta P., Tavakoli A., Levis P. An Empirical Study of Low-Power Wireless. *ACM Trans. Sens. Netw.* 2010;6:16. doi: 10.1145/1689239.1689246.

[11]. *(PDF) Condition Monitoring for Wireless Sensor Network-Based Automatic Weather*

[13] Price, S. (2022, April 15). *What sensors are in a weather station? | Electronics360*. Electronics360.

*[14] REPORT ON INSTALLATION OF GENERATION (Gen) 3 WIMEA-ICT AUTOMATIC WEATHER STATIONS IN UGANDA*. (n.d.). WIMEA-ICT.

*[15] Network Densification Strategies for Automatic Weather Stations: Challenges and Opportunities for Uganda*. (n.d.). NRU.

*[16]The Deployment in the Wireless Sensor Networks: Methodologies, Recent Works and Applications*.

[17] The-real-survey-paper

[18] L. D. Mendes and J. J. Rodrigues, A survey on cross-layer solutions for wireless sensor networks. Journal of Network and Computer Applications (2011), Vol. 34, pp. 523–534.