CHAPTER 1

INTRODUCTION

Recent advances in sensing, computing and communication technologies coupled with the need to continuously monitor physical phenomena have led to the development of Wireless Sensor Networks (WSNs). WSN is defined as the collection of spatially distributed autonomous nodes that can communicate wirelessly. Each node consists of four main components: sensor, processor, radio, and battery. These nodes are commonly known as “Sensor Nodes”. A sensor is a device to interpret some characteristic of its environment. It detects events or changes in quantities and provides a corresponding output, generally as an electrical or optical signal. A processor is used for computation, a radio is for wireless communication to transfer the data to a base station and a Battery is the main power source in a sensor node. Secondary power supply that harvests power from the environment such as solar panels may be added to the node depending on the appropriateness of the environment where the sensor will be deployed.

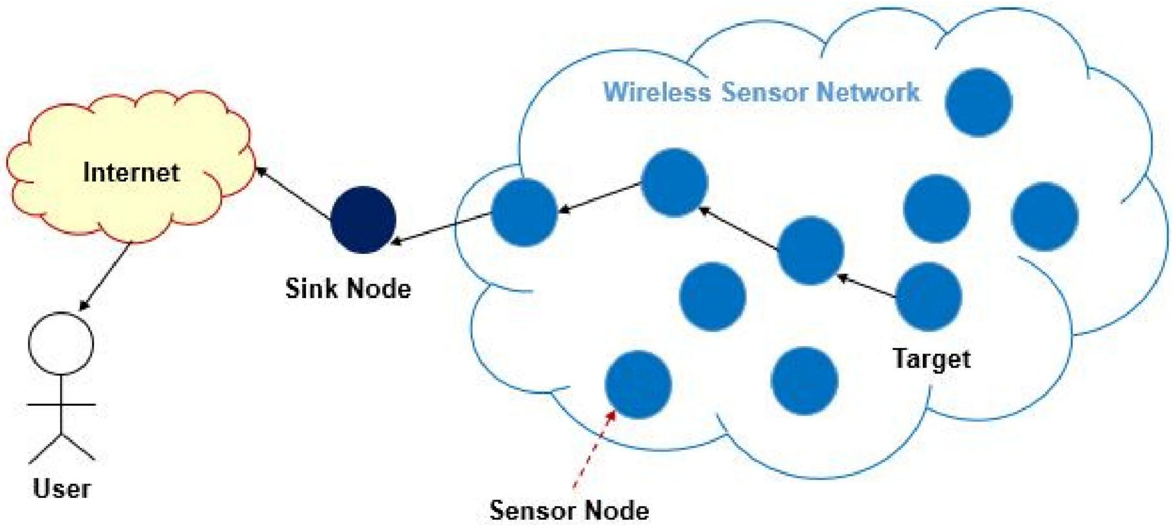

Figure 1.1: Wireless Sensor Network

Figure 1.1 presents the basic structure of a WSN. The sensor nodes are deployed densely in the desired environment. Depending upon the routing protocol of the WSN the sensors senses the environment continuously or after every fixed quantum. Then the sensor node transmits the sensed information either periodically or on the occurrence of some event to the Sink Node. The transmission of information to the sensor nodes to the Sink Node can be one in a variety of ways as proposed by many routing protocols to increase the efficiency of the network. Traditional way of transmitting the information directly from the sensor node to Sink Node is very energy consuming and inefficient.

Thus many protocols are proposed which decreases the transmission from the sensor nodes as one depicted in Figure 1.1. It uses multi-hop transmissions to transmit the information to the Sink Node which distributes the energy expended amongst many nodes and hence prolongs the lifetime of whole of the network. Use of such mechanisms and development of energy efficient routing protocol is a need for the proficient use of WSNs.

There are two types of WSNs: structured and unstructured. An unstructured WSN is one that contains a dense collection of sensor nodes. Sensor nodes may be deployed in an ad hoc manner into the field. Once deployed, the network is left unattended to perform monitoring and reporting functions. In an unstructured WSN, net-work maintenance such as managing connectivity and detecting failures is difficult since there are so many nodes. In a structured WSN, all or some of the sensor nodes are deployed in a preplanned manner. The advantage of a structured network is that fewer nodes can be deployed with lower network maintenance and management cost. Fewer nodes can be deployed now since nodes are placed at specific locations to provide coverage while ad hoc deployment can have uncovered regions. sensor nodes have self-organizing capabilities, to form an appropriate structure in order to collaboratively perform a particular task. Wireless Sensor Networks are found suitable for applications such as surveillance, precision agriculture, smart homes, automation, vehicular traffic management, habitat monitoring, and disaster detection.

1.1. Overview of key issues in WSN

Current state-of-the-art sensor technology provides a solution to design and develop many types of wireless sensor applications. Available sensors in the market include generic (multi-purpose) nodes and gate-way (bridge) nodes. A generic (multi-purpose) sensor node’s task is to take measurements from the monitored environment. It may be equipped with a variety of devices which can measure various physical attributes such as light, temperature, humidity, barometric pressure, velocity, acceleration, acoustics, magnetic field, etc. Gateway (bridge) nodes gather data from generic sensors and relay them to the base station. Gateway nodes have higher processing capability, battery power, and transmission (radio) range. A combination of generic and gateway nodes is typically deployed to form a WSN.

To enable wireless sensor applications using sensor technologies, the range of tasks can be broadly classified into three groups as shown in Figure 1.2. The first group is the system. Each sensor node is an individual system. In order to support different application software on a sensor system, development of new platforms, operating systems, and storage schemes are needed. The second group is communication protocols, which enable communication between the application and sensors. They also enable communication be-tween the sensor nodes. The last group is services which are developed to enhance the application and to improve system performance and network efficiency. From application requirements and network management perspectives, it is important that sensor nodes are capable of self-organizing themselves. That is, the sensor nodes can organize themselves into a network and subsequently are able to control and manage themselves efficiently. As sensor nodes are limited in power, processing capacity, and storage, new communication protocols and management services are needed to fulfil these requirements. The communication protocol consists of five standard protocol layers for packet switching: application layer, transport layer, network layer, data-link layer, and physical layer. In this survey, we study how protocols at different layers address network dynamics and energy efficiency. Functions such as localization, coverage, storage, synchronization, security, and data aggregation and compression are explored as sensor network services. Implementation of protocols at different layers in the protocol stack can significantly affect energy consumption, end-to-end delay, and system efficiency. It is important to optimize communication and minimize energy usage. Traditional networking protocols do not work well in a WSN since they are not designed to meet these requirements. Hence, new energy-efficient protocols have been proposed for all layers of the protocol stack. These protocols employ cross-layer optimization by supporting interactions across the protocol layers. Specifically, protocol state information at a particular layer is shared across all the layers to meet the specific requirements of the WSN. As sensor nodes operate on limited battery power, energy usage is a very important concern in a WSN; and there has been significant research focus that revolves around harvesting and minimizing energy. When a sensor node is depleted of energy, it will die and disconnect from the network which can significantly impact the performance of the application. Sensor network lifetime depends on the number of active nodes and connectivity of the network, so energy must be used efficiently in order to maximize the network lifetime. More details can be found in **[1]**.

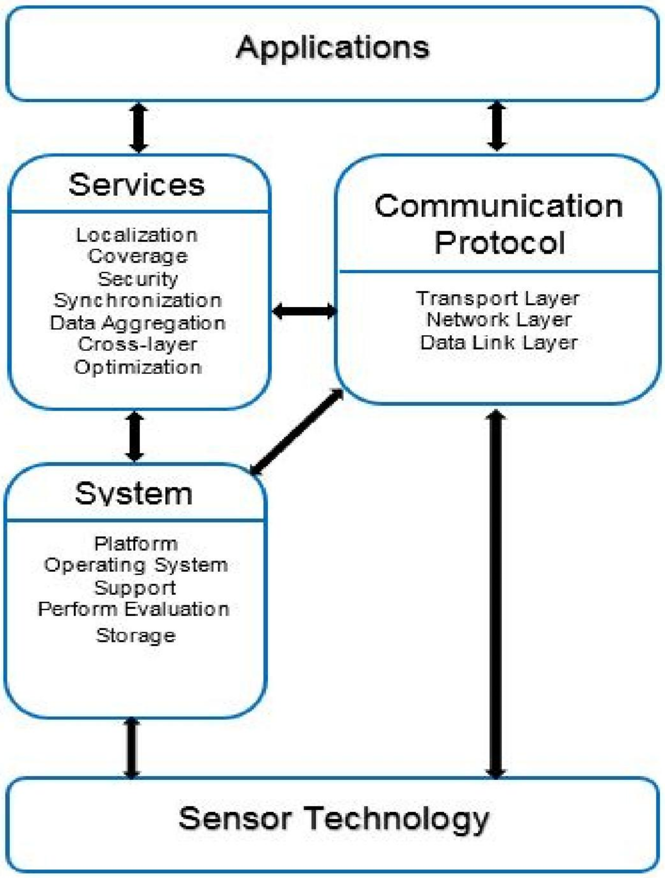

Figure 1.2: Broad classification of various issues in a WSN

Energy conservation in a WSN maximizes network life-time and is addressed through efficient reliable wireless communication, intelligent sensor placement to achieve adequate coverage, security and efficient storage management, and through data aggregation and data compression. The above approaches aim to satisfy both the energy constraint and provide quality of service (QoS) for the application. For reliable communication, services such as congestion control, active buffer monitoring, acknowledgements, and packet-loss recovery are necessary to guarantee reliable packet delivery. Communication strength is dependent on the placement of sensor nodes. Sparse sensor placement may result in long-range transmission and higher energy usage while dense sensor placement may result in short-range transmission and less energy consumption. Coverage is interrelated to sensor placement. The total number of sensors in the network and their placement determine the degree of network coverage. Depending on the application, a higher degree of coverage may be required to increase the accuracy of the sensed data. In this survey, we review new protocols and algorithms developed in these areas. The various applications are shown in figure 1.3.

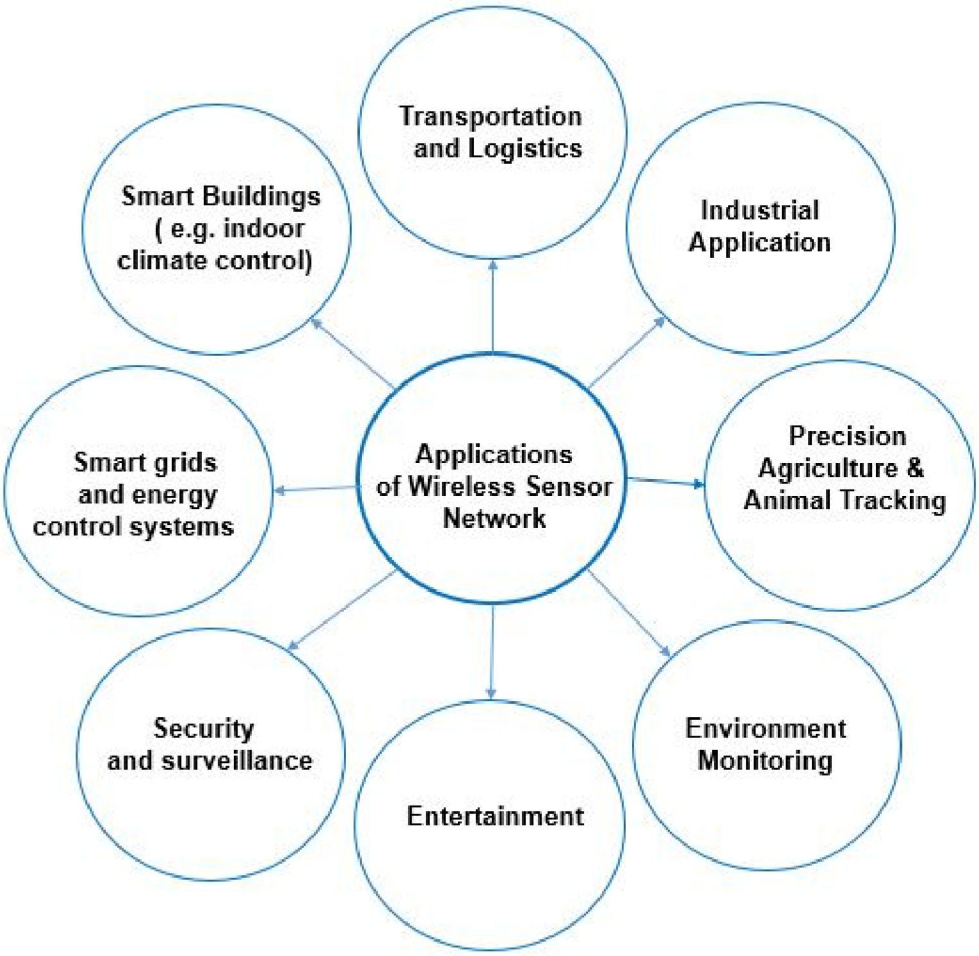

Figure 1.3: Applications of WSNs

1.2. Sensor Node Architecture

Advances in Micro-Electro Mechanical System (MEMS) based sensor technology has led to the development of miniaturized and cheap sensor nodes, capable of communicating wirelessly, sensing and performing computations. The sensing circuitry is capable of measuring ambient conditions related to surroundings of the sensor which is then transformed into an electric signal.

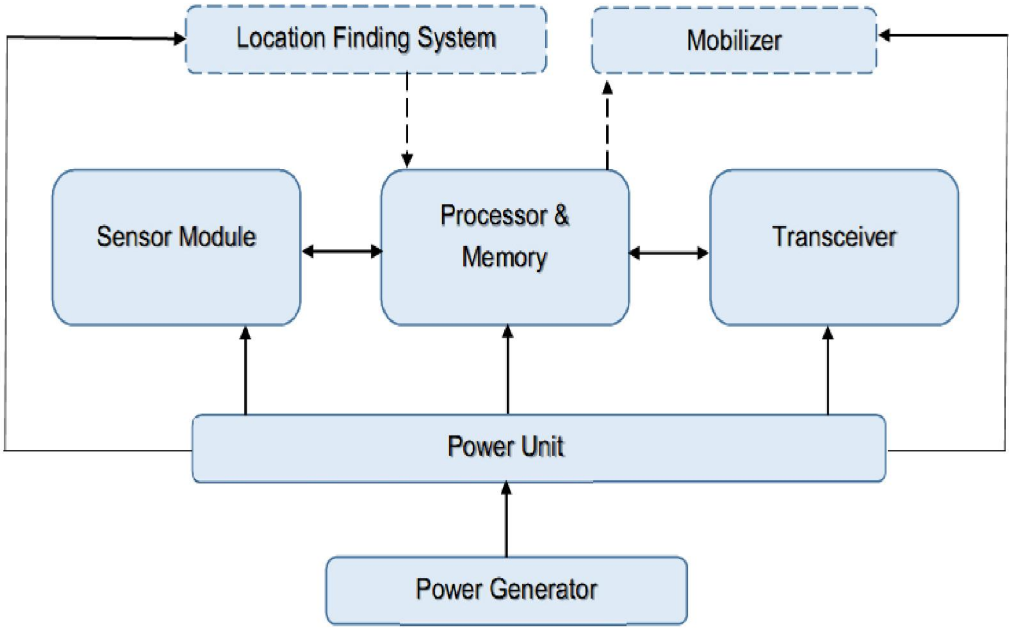

Figure 1.4: Components of a sensor node

Figure 1.3 represents the basic structure of a sensor node which is made up of several components. Sensors are generally composed of four basic components; a sensing unit, processing unit, transceiver unit and power unit. Some of the additional components which can be found in the sensor depending on the applications are mobilizer, location finding system and power generator. The sensors and the ADCs (Analog to Digital Convertors) form the sensing unit. Sensors produces the analog signals based on the desired phenomenon and then those sensed analog signals are converted in to digital signals by ADC. The core of the wireless sensor node is the processing unit, usually a microprocessor with a limited amount of memory. The converted digital signals are given as an input to the processing unit which is associated with a small storage unit and handles the collaboration of the sensor nodes with other nodes to accomplish desired sensing tasks. Transceiver is used to connect the node to the network and is usually capable of bidirectional communications. The most essential unit of a sensor node is the power unit as it derives all the other units to accomplish their respective tasks. The more details of sensor nodes can be found in **[2].**

Although in WSNs nodes have limited power and are not rechargeable there are some application dependent sources which can assist the power unit like solar cells. Specific nodes may integrate a location finding system that helps the node to discover its position, relative to its neighbors or global. Location finding system is another optional but essential unit in sensor nodes as location of the respective node is generally required time to time to accomplish the routing efficiently. Thus because of limited transmission and power capabilities major research is done to reduce the number of transmission without sacrificing the essential information and efficient use of power unit to prolong the lifetime of the nodes and hence the network. Dense deployment of sensor nodes in the sensing field and distributed processing through multi-hop communication among sensor nodes is required to achieve high quality and fault tolerance in WSNs.

1.3. Sensor Network Protocol Stack

The algorithms developed for wireless ad-hoc networks cannot be used for sensor networks, for several reasons. One is that the number of sensor nodes is typically much more than in a typical ad-hoc network, and sensor nodes, unlike ad-hoc nodes, are prone to permanent failure. In addition, sensor nodes normally use broadcast rather than point-to-point communication with its limited power and memory. Unlike computer networks, sensor nodes do not have global ID, since a typical packet overhead can be too large for them.

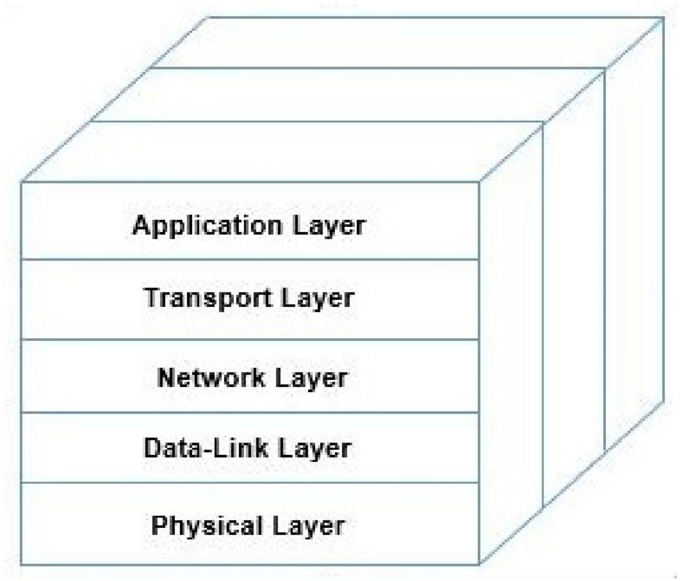

Figure 1.5: Sensor Network Protocol Stack

The sensor network protocol stack is different from the standard TCP/IP and along with the layers used in traditional protocols it has additional planes to handle the issues in the sensor nodes. Figure 1.4 shows a protocol architecture for sensor networks. The protocol stack combines power efficiency and least-cost-path routing. This protocol architecture integrates networking protocols and power through the wireless medium and promotes cooperative efforts of sensor nodes. The protocol stack consists of the physical layer, data-link layer, network layer, transport layer, and application layer, backed by a power-management plane, mobility-management plane, and task-management plane. The power management plane (PMP) has the responsibility to decide how a node will use its power like when it has to switch on their sensors and when they have to switch off them. Power management plane also takes care of when to transmit the information and when not to. Mobility management plane (MMP) is responsible to know the route back to the BS in case of movement of the sensor nodes from their initial positions and inform neighbors about the newly moved sensor node. With the updated knowledge of neighbors now they can contribute and use the power of the nodes efficiently. Task management plane (TMP) performs the balancing of sensing tasks amongst the sensor nodes. It is not necessary that all the sensor nodes in the vicinity should sense and transmit the information. Thus TMP takes such decisions as per the application used in WSNs. So the additional planes are very essential for the objective of WSNs as without them a node will have no capability and will act as an individual transmitting the unwanted information and wasting the constrained power of the sensor node.

The physical layer is responsible for robust modulation, transmission, and receiving signals. Media access control (MAC) at the data-link layer must minimize packet collision with neighboring nodes, as power is a restricted factor. The network layer routes packets provided by the transport layer. The application layer uses software for preparation of data on an event. The power-management plane monitors the sensor's power level among the sensor nodes and manages the amount of power a sensor node has used. Most of the sensor network routing techniques and sensing tasks require an accurate knowledge of location. Thus, a sensor node commonly has a location-finding system. A mobilizer may sometimes be needed to move sensor nodes to carry out assigned tasks. Sensor network routing protocols must be capable of self-organizing. For these purposes, a series of energy-aware MAC, routing, and clustering protocols have been developed for wireless sensor networks. Most of the energy-aware MAC protocols aim to either adjust the transmission power or keep transceivers off as long as possible.

1.4. WSN Characteristics

In a WSN, sensor nodes are scattered randomly or with a static strategy in the network field varying from hundreds to thousands in number. There are many operational characteristics of WSNs, few can be listed as:

- **Fault tolerant**

  The system should be robust against node failure (running out of energy, physical destruction, H/W, S/W issues etc.). Some beep mechanism should be incorporated to indicate that the node is not functioning properly.

- **Scalable**

  The system should support large number of sensor nodes to cater for different applications.

- **Long life**

  The node’s life-time entirely defines the network’s life-time and it should be high enough. The sensor node should be power efficient against the limited power resource that it have since it is difficult to replace or recharge thousands of nodes. The node’s communication, computing, sensing and actuating operations should be energy efficient too.

- **Programmable:**

  The reprogramming of sensor nodes in the field might be necessary to improve flexibility.

- **Secure:** the node should support the following
    - **Access Control:** Prevents unauthorized attempts to access the node.
    - **Message Integrity:** Detects and prevent unauthorized changes to the message.
    - **Confidentiality:** Assures that sensor node should encrypt messages so only those nodes would listen who have the secret key.
    - **Replay Protection:** Assures that sensor node should provide protection against adversary reusing an authentic packet for gaining confidence/network access, man in the middle attack can be prevented by time stamped data packets.
    - **Affordable:** The system should use low cost devices since the network comprises of thousands of sensor nodes, tags and apparatus. Installation and maintenance of system elements should also be significantly low to make its deployment realistic.
- **Self-Configuration**

  The topology of WSNs is not static and is supposedly changed with no traceable patterns. So the sensor nodes of the network have to be adaptable to such changes while keeping the power efficiency intact. Self-configuration also has to handle the situations like node failures and node additions to the network or any other obstacles.
- **Single-Hop Communication**

  Traditional WSN protocol used single hop mechanism to send the information to the BS. In this mode of communication all the sensor senses the information and then transmits all the information it has sensed to the BS directly without the involvement of other sensor nodes on its own. This mode of communication is very inefficient and power consuming with lots of redundant information to the BS. Thus modern protocols rely on the multi hop mode of communication for better efficiency.
- **Multi-Hop Communication**

  In case of larger networks where the distance of the BS from the node is greater than the transmission range of the sensor node. The single hop communication fails. Multi hop communication uses packet forwarding to increase the efficiency of the network. The nodes send the information to the sink with the help of other intermediate sensor nodes which receives the information from the node and transfers the information to other node along the path of the BS or to BS itself. Multi hop communication saves transmission energy and is proved to be useful for energy efficiency and network lifetime.
- **Automatic Load Balancing**

Nodes in the network must decide who will be the parent node to transmit the information based on the hop count to the respective node, signal strength, link quality and present load quantity of the parent node. Automatic load balancing is dynamic in nature as the number of nodes in the network can run out of power anytime in between the network lifetime.

- **Efficient Energy Usage**

  Power is very critical issue in increasing the performance of the network. So energy in a WSN should be expended in optimal manner like sensor nodes can switch off their sensors for a particular period of time and can switch on their transmitters only when any event occurs or after every frame time.

The architecture of single hop and multi-hop wireless sensor network is shown in Figure 1.5 in single hop communication all nodes are directly connected to the sink node or base station while in multi-hop communication some nodes are not directly connected to the sink node rather they are connected to some intermediate nodes.

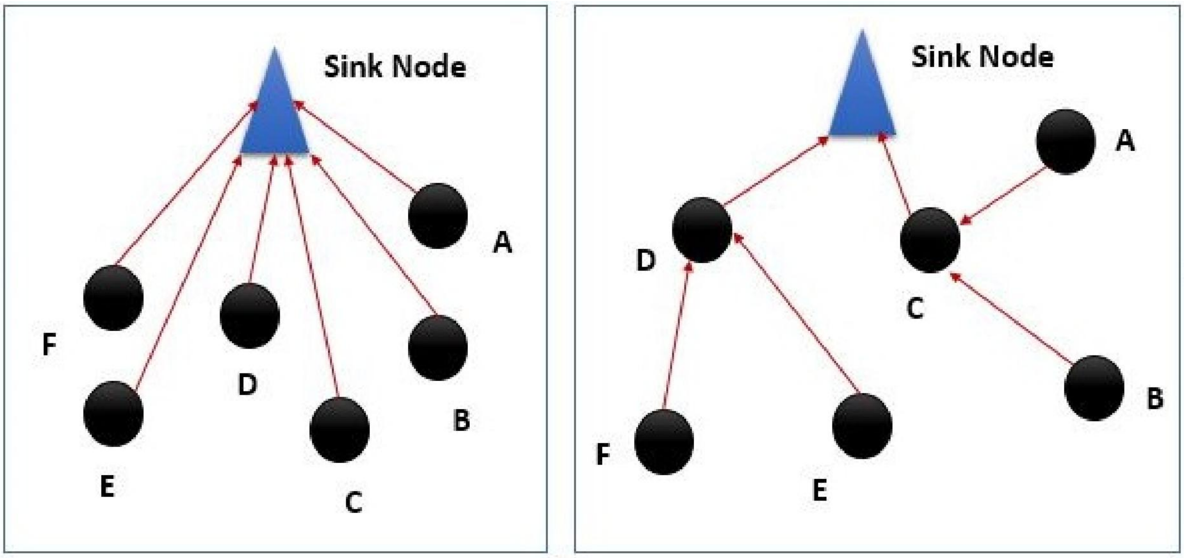

Figure 1.6: Single and multi-hop networks

1.5. Motivation

Wireless sensor network (WSN) is a special type of ad-hoc network, where nodes form the network dynamically without help of any infrastructure. Normally the nodes are deployed randomly and are supposed to sense a phenomenon, process the collected sensing data in a collaborative manner, and route the results to an end user. For this, the active nodes of the network have to maintain both network connectivity and coverage. The network cannot guarantee the quality of surveillance without sufficient coverage. Besides, data routing cannot be achieved without proper connectivity. In wireless sensor networks, sensors are often intended to work in remote or hostile environments, such as a battlefield or desert to complete the mission. In wireless sensor network, nodes can be classified into static and mobile nodes.

Current research in wireless sensor networks has focused on fixed sensor networks in which nodes are static. Static sensor nodes cannot change position by themselves after their deployment. On the other hand, mobile sensors can change their position autonomously, depending on the mission requirements and are able to dynamically adjust network topology to improve the performance of sensor networks. Recent advances in wireless technology with demands for greater user mobility have provided a major impetus toward the development of a mobile network architecture. Unlike the existing schemes in which sensors are stationary, if we deploy mobile sensors in the network such as a battlefield or environmental monitoring region and let few nodes move, the degree of coverage and connectivity can be improved. Besides, it is more economical and versa-tile than the existing fixed stationary sensor networks and redeployment. Due to mobility of sensor nodes, mobile sensors can change their position depending on the requirement of the missions.

Dynamic adjustment of the sensor node's position would change the topology of the nodes and hence promote the performance of sensor networks. When sensors are deployed in a disaster environment, where human interference is not possible, we need mobile sensors to accomplish the tasks such as cover-age and connectivity compensation, location assignment and node replacement. In a post deployment scenario, it is possible that some nodes over certain region are destroyed due to intrusion, explosion or due to environmental factors like heat, vibration and failure of electronic components or software bugs in the network. In another scenario, power sources of the nodes may lead death of the nodes, thus affecting the coverage and connectivity of the original network. Hence, it is essential to reconfigure the network by mobile sensors to maintain the connectivity and coverage, and thereby avoiding the network partitions.

1.6. Research Objective

Keeping in mind the motivation defined in the last section the objective of this research is explained as follows:

- To develop a hybrid scheme to deploy sensor nodes in a specified target field while considering the moving nodes.
- To compare the developed approach with standard available approaches in the literature.
- To understand the various issues in WSN and formulating the exact problem under consideration
- To modify the proposed hybrid approach to handle the connectivity problem in the WSN effectively.
- To perform simulation to assess the performance of the developed hybrid approach

1.7. Thesis Organization

The thesis starts with comprehensive introduction in chapter 1. Chapter 2 is dedicated for literature review, explained in detail in bottom up fashion keeping in mind the problem under consideration. The review is based on current research for the problem under consideration. Chapter 3 is dedicated for the related work in the selected area, this part is application oriented and describes the current available algorithms for the problem. Chapter 4 is dedicated for explaining and formulating the problem and then followed by the proposed work, the problem is explained in different parts for the sake of easiness. Chapter 5 is dedicated for the simulation and analysis of the results obtained. Chapter 6 concludes the thesis work with future work.

CHAPTER 2

LITERATURE REVIEW

A wireless sensor network (WSN) has many important applications in different areas such as security and surveillance, entertainment, remote environmental monitoring etc. The advances in the sensor technologies in recent years have made the availability of smaller, intelligent and cheaper sensors possible. The sensors are equipped with wireless interfaces so that they can communicate with each other to form a network. The design of a WSN is application oriented and may consider the factors such as the environment, system constraints, the application's design objectives cost and hardware.

2.1. Types of WSN

WSN can be classified in five types on the basis of the environment in which they have to operate and face different challenges and constraints.

- **Terrestrial WSNs**

  It typically consists of hundreds to thousands of low cost sensor nodes deployed in a given target area, this task is done either in pre-planned or an ad hoc in a fashion. In pre-planned deployment fashion, there are different placement models such as grid placement, optimal placement **[3]**, 2-d and 3-d placement **[4, 5]** models. In ad hoc deployment fashion, the sensor nodes can be dropped at a height from a plane and randomly deployed into the target area. In a terrestrial WSN, reliable communication in a dense environment is very important. Terrestrial sensor nodes must be able to communicate data effectively back to the sink node. The battery power of the sensors is very limited and may not be rechargeable, however a secondary power source such as solar cells can be used. In any case, it is important for sensor nodes to conserve energy. In a terrestrial WSN the conservation of energy is done with some factors, some of them are eliminating data-redundancy, short-transmission-range, minimizing-delays, low-duty-cycle operations and multi-hop optimal routing.

- **Underground WSNs**

  It consist of a number of sensor nodes buried underground or in a cave or mine used to monitor underground conditions. Additional sink nodes are located above ground to relay information from the sensor nodes to the base station. An underground WSN is more expensive than a terrestrial WSN in terms of equipment, deployment, and maintenance **[6, 7]**. Underground sensor nodes are expensive because appropriate equipment parts must be selected to ensure reliable communication through soil, rocks, water, and other mineral contents. The underground environment makes wireless communication a challenge due to signal losses and high levels of attenuation. Unlike terrestrial WSNs, the deployment of an underground WSN requires careful planning and energy and cost considerations. Energy is an important concern in underground WSNs. Like terrestrial WSN, underground sensor nodes are equipped with a limited battery power and once deployed into the ground, it is difficult to re-charge or replace a sensor node's battery. As before, a key objective is to conserve energy in order to increase the lifetime of network which can be achieved by implementing efficient communication protocol. Underwater WSNs **[8, 9]** consist of a number of sensor nodes and vehicles deployed underwater. As opposite to terrestrial WSNs, underwater sensor nodes are more expensive and fewer sensor nodes are deployed. Autonomous underwater vehicles are used for exploration or gathering data from sensor nodes. Compared to a dense deployment of sensor nodes in a terrestrial WSN, a sparse deployment of sensor nodes is placed underwater. Typical underwater wireless communications are established through transmission of acoustic waves. A challenge in underwater acoustic communication is the limited bandwidth, long propagation delay, and signal fading issue. Another challenge is sensor node failure due to environ-mental conditions. Underwater sensor nodes must be able to self-configure and adapt to harsh ocean environment. Underwater sensor nodes are equipped with a limited battery which cannot be replaced or recharged. The issue of energy conservation for underwater WSNs involves developing efficient underwater communication and net-working techniques.

- **Multi-media WSNs**

It have been proposed to enable monitoring and tracking of events in the form of multi-media such as video, audio, and imaging **[10]**. Multi-media WSNs consist of a number of low cost sensor nodes equipped with cameras and microphones. These sensor nodes interconnect with each other over a wireless connection for data retrieval, process, correlation, and compression. Multi-media sensor nodes are deployed in a pre-planned manner into the environment to guarantee coverage. Challenges in multi-media WSN include high bandwidth demand, high energy consumption, quality of service (QoS) provisioning, data processing and compressing techniques, and cross-layer design. Multi-media con-tent such as a video stream requires high bandwidth in order for the content to be delivered. As a result, high data rate leads to high energy consumption. Transmission techniques that support high bandwidth and low energy consumption have to be developed. QoS provisioning is a challenging task in a multi-media WSN due to the variable delay and variable channel capacity. It is important that a certain level of QoS must be achieved for reliable content delivery. In-network processing, filtering, and compression can significantly improve network performance in terms of filtering and extracting redundant information and merging contents. Similarly, cross-layer interaction among the layers can improve the processing and the delivery process.

- **Mobile WSNs**

  It consist of a collection of sensor nodes that can move on their own and interact with the physical environment. Mobile nodes have the ability sense, compute, and communicate like static nodes. A key difference is mobile nodes have the ability to reposition and organize itself in the network. A mobile WSN can start off with some initial deployment and nodes can then spread out to gather information. Information gathered by a mobile node can be communicated to another mobile node when they are within range of each other. Another key difference is data distribution. In a static WSN, data can be distributed using fixed routing or flooding while dynamic routing is used in a mobile WSN. Challenges in mobile WSN include deployment, localization, self-organization, navigation and control, coverage, energy, maintenance, and data process. Mobile WSN applications include but are not limited to environment monitoring, target tracking, search and res-cue, and real-time monitoring of hazardous material. For environmental monitoring in disaster areas, manual deployment might not be possible. With mobile sensor nodes, they can move to areas of events after deployment to provide the required coverage. In military surveillance and tracking, mobile sensor nodes can collaborate and make decisions based on the target. Mobile sensor nodes can achieve a higher degree of coverage and connectivity compared to static sensor nodes. In the presence of obstacles in the field, mobile sensor nodes can plan ahead and move appropriately to obstructed regions to increase tar-get exposure.

2.2. WSN Application Areas

WSN applications can be classified into two categories: monitoring and tracking (see Figure. 2). Monitoring applications include indoor/outdoor environmental monitoring, health and wellness monitoring, power monitoring, inventory location monitoring, factory and process automation, and seismic and structural monitoring. Tracking applications include tracking objects, animals, humans, and vehicles. While there are many different applications, below we describe a few example applications that have been deployed and tested in the real environment.

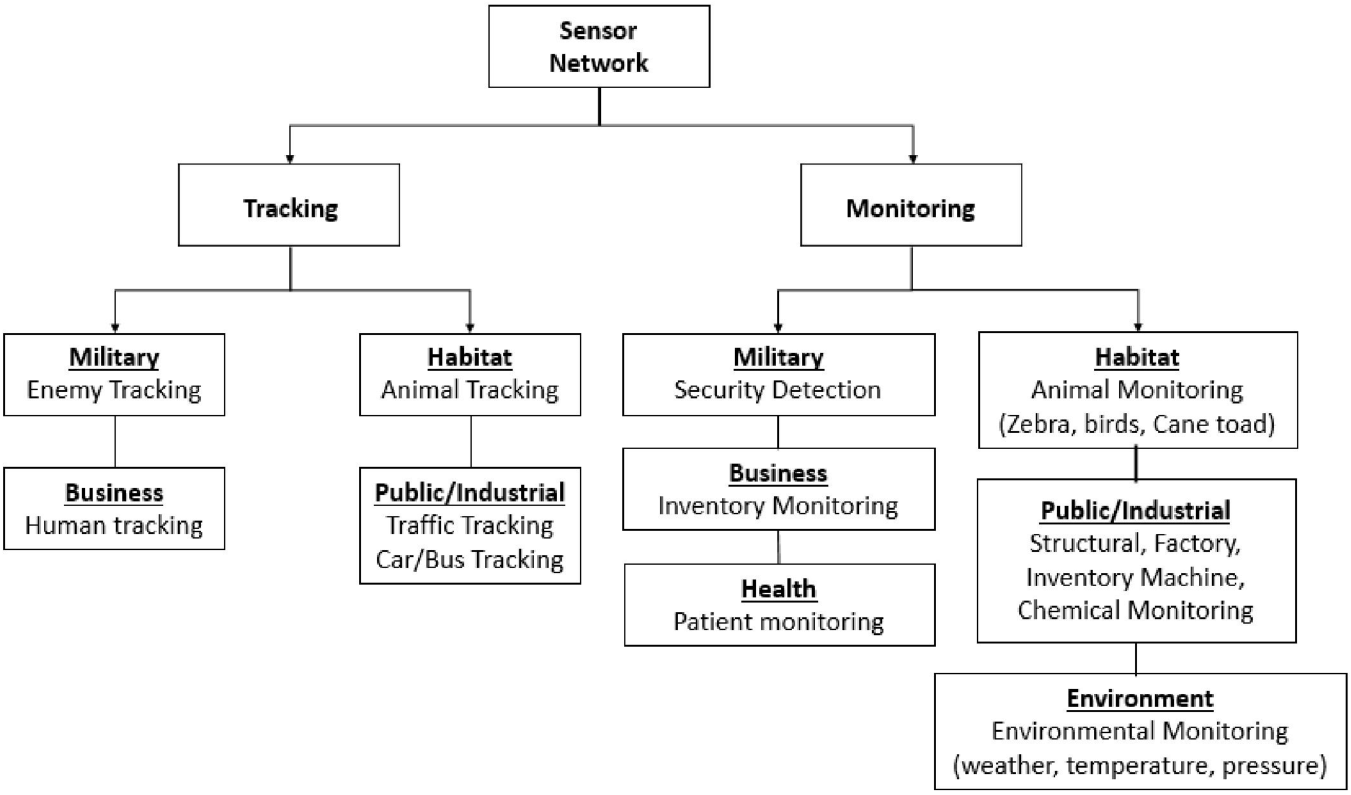

Figure 2.1: Overview of sensor applications

2.3. Application Characteristics

Most of the applications of WSNs share many common characteristics. Some of the similarities based on the interaction(s) between sources and BS are:

- **Periodic Monitoring**

  Sensor can be made to periodically measure the values in the environment and send the measured values to the user.

- **Event Detection**

  Specific events can be detected by the sensor nodes. Such sensor are also applied to addresses the event of natural calamities. Some of the applications with event detection are grass fires, forest fires, volcanic eruptions etc. Classification of events is necessary in case of a sensor network used for more than one event detection.

- **Location Driven**

  Most of the applications work on the changes in the position of the sensors. Applications theft detection, inventory management, detection of behavior of humans are monitored based upon the changes in the position of the sensors.

- **Target Tracking**

  Some applications sense the data even after continuous change in position. In cases of surveillance if any threat is on the move the sensor nodes communicate the source to the BS along with the speed and direction. This helps to estimate the present position of the targets.

2.4. Design Metrics

In the early days of WSNs, majority of its implementation was done for military applications. With the advances in technology and increase in WSNs scope now they are used in a variety of applications. To address these vivid classes of applications some of the essential design issues a sensor network must possess are [18, 20];

- **Scalability**

  As the numbers of nodes in a WSN are dense from hundreds to thousands, scalability is an important issue. Thus the schemes in the sensor nodes should be scalable enough to respond to certain events.

- **Fault Tolerance**

  There are many situations which can cause sensor nodes to be failed or blocked from the network. Situations like lack of power, environmental interference and physical damages can often arise of the sensor nodes. Thus the network should be fault tolerant to such scenarios and the working of network should not be affected even after reduction of the sensor nodes. Thus fault tolerance is the ability of the network to work as expected even after node failures.

- **Production Costs**

  The production costs of sensor nodes should be low as the numbers of sensor nodes in the network is very high and most of the network is defined by the sensor nodes only. Thus lesser prices of sensor nodes lead to feasible networks.

- **Power Consumption**

  The transmission of information in WSNs is very energy consuming, proportional to the square of the distance or even four times in some cases when the distance between the source and target nodes is greater than a particular distance. Multi-hoping is used to reduce the energy consumption. But multi-hops introduce complexity for the medium access control and topology management plane. Thus trade-off should be considered about which mode of communication should be used. If the nodes are in close vicinity of BS then direct communication should be preferred over multi-hop communication.

- **Operating Environment**

  Sensor networks can be setup almost anywhere any possible place in the environment like bottom of a ocean, around battle field beyond enemy lines, in a chemically contaminated field, inside large buildings or home, can be attached to humans or animals, can be installed in high speed vehicles, can spread out in forest to prevent disasters etc.

- **Sensor Network Topology**

  Must be maintained even with very high node densities.

- **Data Delivery Models**

  These models decide when the node has to transmit the information it has sensed to the BS. Some of the models which are used depending upon the applications are event-driven, continuous, query-driven and hybrid. Event-driven models transmit the information when any desired event has occurred. Transmission is done periodically in case of continuous data delivery model. A query-driven model transmits after receiving a particular query from the BS. Hybrid models are used by some of the applications which combine the approaches of continuous, event-driven and query-driven.

- **Environment**

  Nodes are operating in inaccessible locations either because of hostile environment or because they are embedded in a structure.

- **Data Fusion**

  As the sensor nodes are deployed with high density in the area of interest it is very likely that a group of these sensors will sense the same information. Such redundant information from the sensors can be aggregated to reduce the transmission. Data aggregation is the technique used for such purposes in which data from more than one sensor is combined with the help of functions like min, max, average or elimination of duplicates. For large networks, data aggregation introduces a little computation capability in the sensor nodes but still saves lot of energy by reducing the amount of transmissions. Thus varieties of routing protocols use data aggregation for traffic optimization and energy efficiency of the sensor networks.

- **Transmission Media**

  RF, Infrared and Optical.

- **Network Setup**

  Setting up the network is dependent on the application and also plays are role in the performance of the network. Deployment of nodes can be either deterministic or random. In deterministic scenarios the nodes are deployed to the pre-determined locations. Such situations are possible only in the areas with human involvements like inside a building. In random deployment of the sensor nodes distribution of the nodes is not uniform. In such cases when clustering is done position of CH is a critical issue to retain the energy efficiency of the network.

- **Overhead & Data Latency**

  Data latency can be introduced by data aggregation and/or multi-hop communication. Complex algorithms of some routing protocols can sometime create excess overheads which may not be suited for network energy high energy constraints.

- **Quality of Service**

  Quality of service is determined by the application. It may be data reliability, energy efficiency, location awareness, synchronized processing etc. These factors decide which routing protocol should be used for particular application. Some applications like military applications are focused on secure and periodic information from the sensor networks and hence the selection of routing protocol for such an application is continuous routing with cryptographic schemes.

2.5. Key issues of mobile sensor network deployment

This section throws light on the key issues for deploying the sensors in a given target areas, that has to be kept in mind while developing the sensor node deployment algorithms in WSN, details can be found in **[11]**.

- **Localization**

  For putting the sensors we ought to get the objective position of the sensors in the target field. To discover sensor node position diverse systems are utilized effectively. We have to do some deployment work on paper when the target field is simple and plane. If the target area is not simple then we may require some technique to get precise location information of the sensors. Global Positioning System (GPS) is used for locating the sensor nodes at each point of time. GPS incorporation is excessively costly, making it impossible to empower in sensor node, therefore many other schemes must be used.

- **Coverage**

  Sensing range is a major factor to decide the coverage of sensor network in the target field. Coverage can be thought as one of the performance metrics of the services in WSN. An efficient node deployment algorithm leads very few coverage holes. In general the coverage problem is called k-coverage problem. When k =1 the problem is called single-coverage problem and when k>1 the problem is called multi-coverage problem.

- **Connectivity**

  To maintain the network topology node connectivity is an important factor to be considered. In WSN nodes can wake up to join or sleep to quite the network at any time which makes the topology dynamic in nature. Many network characteristics such as capacity, robustness, and latency are directly affected by the network topology that also affects the complexity of data routing and processing. In general a sensor can be connected to k number of sensors which is called as k-connectivity. To maintain the connectivity a sensor has to be connected with at least one sensor in its sensing range. If the connectivity is preserved (possibly over multiple hops in WSN) between any two sensor nodes, the network is said to be connected. Connectivity is a critical issue to be considered while sensor node deployment in the target field area.

- **Communication and Sensing Range**

In a sensor node the sensing unit and the communication unit are independent, therefore the sensing range (R_s) and the communication range (R_c) are not directly related to each other from a hardware point of view. When coverage and connectivity both are considered simultaneously both the units have to be integrated together. The performance of the network depends on R_c and R_s), therefore they have to be decided beforehand which in turn depends upon the type of sensor we use. The R_c/R_s is a critical ratio and thus has to be considered for node deployment. If $R_s < R_c$ then the problem type is coverage and if $R_s > R_c$ then the problem type is connectivity. We can take the assumption that R_c is at least twice R_s, only need to guarantee the coverage and for single coverage problem it will satisfy the connectivity as well.

- **Energy**

Sensors are battery backed up devices and work autonomously so any kind of human intervention is not required for recharging that is they fully dependent on the installed battery life. As the sensors are massively deployed in a hostile target area most of the times, that makes its almost infeasible to recharge all the batteries in the target area, therefore energy is the far most critical parameter in WSN. Energy has to be utilized effectively to maintain the WSN life time. The communication among the sensor noes consumes most of the energy. To consider this problem optimal number of sensors have to be deployed on the field. If nodes are mobile then it consumes extra energy for the mobility in WSN. Energy efficiency and power consumption may be considered while deploying the nodes.

- **Node Density**

Among several design issues scalability is an important one that should be considered during sensor node deployment that may eventually affect the coverage, cost and the performance of WSN. Node density is an important parameter that may increase computational overhead and cost of the network significantly. On the other hand if the density is low then it may lead the problem of network partition or coverage holes. If the density is uniform it will decrease the chances of coverage holes and clustering.

- **Obstacle Adaptability**

If we consider the general case when the target field is not a plane surface then in that case we may have to consider the obstacles during sensor node deployment. In real time scenario it’s very important issue. Sensor nodes may change their path or they can take proper action while encountering any obstacle in their paths.

- **Lifetime**

  The lifetime of WSN has a great influence on the required degree of robustness and energy efficiency of the nodes. Network lifetime is an important parameter and is application specific, so lifetime may range from few minutes to several hours or even to several years. Many approaches have been proposed to increase the lifetime, like incremental deployment. When a node is detected non-functional after adding a new node to the network, we have to consider several issues like load balancing, balancing energy consumption and sensor relocation.

- **Sensor Relocation**

  At any time if we detect the coverage hole then we have to relocate the redundant nodes to occupy that region, this process is called as sensor relocation. There may be some other reasons for sensor relocation like to balance energy consumption, message overhead during communication.

- **Movement of Sensors**

  Mobility of sensor nodes add extra overhead as it gives functionalities like relocation, locomotion and coverage optimization property to WSN. Mobility consumes power so resource management is very important in this regard. The idea is if we could anticipate whether a node will move or not we can distribute the energy accordingly but an efficient mobility policy will lead to a better coverage remains an unanswered question or an open problem so far.

- **Fault Tolerance**

  Sensor node loses its energy during its lifetime and fails eventually that may lead to degradation in network performance. The WSN has to be fault tolerant that is the proposed scheme must work even in the case of node failure. We may employ techniques like new node deployment, sensor relocation, load balancing, and incremental deployment. Sensor node mobility is highly considered for handling fault tolerance.

2.6. Deployment Algorithms

The purpose of good node deployment algorithm is to minimize the node redundancy and the network costs, but also can prolong the life time of WSN. All the deployment algorithms can be classified into Centralized or Distributed algorithms. When the node density is low we should prefer centralized deployment algorithm that runs on a single sensor node in WSN that may be a sink node or cluster head node. The sink node controls the functioning of the nodes in the network by communicating them over a period of time. When the node density is large we should prefer distributed deployment algorithm that runs on each of the sensor node in in WSN, performing their own task. They are more desirable because they are scalable and computationally more efficient. Most of the times central server architecture might not be possible, in those cases distributed algorithms are unavoidable.

2.6.1. Classification based on optimal deployment of sensor nodes in the target field

- **Computational Geometry Based**

  In this approach the target field is considered as a set of grids or polygons, these grids are either fixed or moving. The aim of these grids or polygons is to provide location indication for the moving sensor nodes. The computational geometry method when coverage problem is considered is similar to the deployment problem in WSN.

- **Potential Field Based**

  In this approach the sensor nodes in target field are modeled as points and are subjected to attractive or repulsive force based on the distance between each two sensors, the forces are of type Newton's Law. We can set a threshold of desired distances between the sensors in the field, each sensor then moves according to the summation of the force vectors and eventually we get a uniform deployment in the field. Virtual force can minimize sensing overlap by moving sensors from high to low density areas. This approach is generally used in self-deployment of sensor that ensure the mobility of sensors in the field.

- **Probabilistic Approach**

  In this approach, the initial positions of the sensors are chosen by random deployment method and deployment of sensor nodes on the target field will be obtained dynamically. The sensor field is assumed to be a two-dimensional grid. Each sensor node is supposed to knows its position. In this approach probabilistic sensing model is used in place of binary detection model.

- **Bio-Inspired Algorithms**

  In this approach the initial deployment is done using random deployment and then optimal position of the sensor node is obtained using bio inspired method like particle swarm optimization, ant bee colony, genetic algorithm, teaching learning based optimization etc.

  These algorithms are inspired from nature so they are called as nature inspired or bio inspired optimization techniques. The way in which bio-inspired computing differs from the traditional artificial intelligence (AI) is in how it takes a more evolutionary approach to learning, in traditional AI, intelligence is often programmed from above: the programmer is the creator, and makes something and imbues it with its intelligence. Bio-inspired computing, on the other hand, takes a more bottom-up, decentralized approach; bio-inspired techniques often involve the method of specifying a set of simple rules, a set of simple organisms which adhere to those rules, and a method of iteratively applying those rules. For example, training a virtual insect to navigate in an unknown terrain for finding food includes six simple rules

2.6.2. Classification based on the manner of node placement in the target field

- **Random Deployment**

  It's the most practical way when the target field is susceptible to frequent changes or when no knowledge is available beforehand about the field. Random deployment is often backed up by robots, they are dropped initially from a height in air. Random deployment is always served as the initial deployment strategy in movement assisted deployment. It's noteworthy that this strategy may not provide a uniform distribution which may be desirable for a longer lifetime of WSN over a particular area in the field. This strategy may cause clustering while leaving the concentration of other part low.

- **Incremental Deployment**

  This strategy is a centarlized approach, i.e nodes are deployed one at a time. The deployment of a node is based on the information obtained by previously deployed sensor node to determine the deployment position in the field. The calculation of new deployment position is done by the sink node. In this approch we don't require any extra localization method, when the environment model is not available then this approch considers the nodes as landmarks. This approach is a greedy in the sense at each step the location calculated for each node will produce maximum network coverage. This approch is commonly applied to static deployment.

- **Movement-Assisted Deployment**

  The deployment approaches may be inaccurate if the actual node deployment position can not be controlled due to wheather conditions like wind and obstacles. So its deirable to use mobile nodes which may be backed up by robots. The initial deployment is done by random deployment approach and then we use any optimal deployment algorithm to re-deploy these nodes in the field. These algorithms are inspired by multi-robot exploration problems.

2.6.3. Classification based on the mobility of sensor nodes in the target field

- **Static Deployment**

  The static deployment obtains the best location according to some optimization strategy, and the location of the sensor nodes don't change during lifetime of the network. In present scenario, the static deployment includes randomly deployment & deterministic deployment.

- **Dynamic Deployment**

  The dynamic deployment may be backed up by robots from a height in air. In his approach nodes are first deployed randomly then these nodes are supposed to acquire their best location based on some optimal deployment scheme. This approach considers moving nodes in the field.

CHAPTER 3

RELATED WORKS

Keeping in mind different issues in Wireless Sensor Networks, this section describes few research work in the area of coverage, connectivity, sensor mobility, sensor relocation, obstacle adaptability and WSN lifetime. Figure 3.1 shows the key issues in WSN.

Figure 3.1: WSN Key Issues

This chapter focuses the problem of finding an optimal coverage of sensor nodes in WSN while ensuring connectivity among sensors. This connectivity preservation is achieved without using centralized control and accurate location information. The optimal node coverage is done according to OTLBO (Orthogonal Teaching Learning Based Optimization) in order to improve network coverage. OTLBO is an improvement over TLBO (Teaching Learning Based Optimization) that makes TLBO fast to converge and more robust. OTLBO is a recent approach in the optimization field. The connectivity preservation algorithm is localized and is based on a subset of neighbors for taking motion decision. The connectivity preserving algorithm maintains a connected topology; the distance covered by the mobile nodes is constrained by the connectivity of the node to its neighbors in a connected sub-graph like the relative neighborhood graph. Finally the node coverage is based on OTLBO optimization technique.

Sensor Node deployment in WSN is mainly classified in two categories: static and dynamic deployment. The static node deployment chooses the best location according to some optimization strategy, and the location of the sensor nodes will not change in the lifetime of the WSN. The static deployment includes the deterministic deployment and the randomly deployment. Although deterministic deployments can provide optimal solutions, they are not always feasible since they require precise knowledge of the monitored area.

The dynamic deployment or on-demand deployments are only feasible when the sensors' positions are available, and when sensors have motion capabilities. The advantage of on-demand deployment is the possibility to obtain particular topologies which can reduce energy consumption, optimize routing scheme or flooding, etc.

The coverage schemes associated with on-demand deployment are full coverage (sensors try to cover the whole area of interest), barrier coverage (sensors try to form a barrier for intrusion detection) or sweep coverage (sensors try to cover only some specific points). To preserve connectivity, the distance covered by the mobile nodes is constrained by the connectivity of the node to its neighbors in a connected sub-graph like the relative neighborhood graph.

The chapter focuses on the related works done for the problem of network coverage while ensuring the connectivity among the sensor nodes all the time. The coverage problem is handled by Orthogonal Teaching Learning Based Optimization, based on TLBO (Teaching Learning Based Optimization). OTLBO makes the convergence rate of the algorithm faster. Teaching Learning Based Optimization is a recent nature inspired global optimization method. TLBO is a Population-based iterative learning method that uses a population of initial solutions to obtain the global solution. The technique is based on a simple fact that a teacher in a class can influence the performance of learners. TLBO works in two phases, the first phase is called "Teacher Phase" which means learning from the teacher and the second phase is called "Learner Phase" which means learning through the interaction between learners.

As teacher is considered most knowledgeable person in the society, so in each evolutionary step the best learner is considered as a teacher. The teacher tries to disseminate knowledge among learners, which will in turn increase the knowledge level of the whole class and help learners to get good marks or grades i.e. the presence of a teacher may increase the performance of all learners in the class. The teacher increases the mean of the performance of the class according to his or her capability i.e. the teacher T1 will try to move mean M1 towards her own level according to her capability, thereby increasing the learners' level to a new mean M2.

Learners can increase their knowledge by two different approaches: First, with the help of input from the teacher and Second, with the help of interaction between themselves. A learner interacts randomly with other learners with the help of group discussions, presentations, formal communications, etc. A learner adds something new to her knowledge if the other learner has more knowledge than her. Orthogonal Teaching Learning Based Optimization (OTLBO) is an improvement over TLBO which is based on orthogonal design concept that can be efficiently applied on multi-factor and multi-level problems. OTLBO makes TLBO faster and robust. It is known empirically that OTLBO has high performance in solving some benchmark functions comprising many parameters, as compared with some existing Evolutionary Algorithms. Here “Teaching Learning Based Optimization (TLBO)” is used as a function optimization technique and the “Orthogonal design (OD)” represents some properties of the orthogonal design method.

Wireless sensor networks (WSNs) have been used successfully in many important areas such as surveillance, target tracking and classification. There are the two most significant factors for the performance of WSNs namely coverage and target detection probability. Sensor nodes can be deployed according to some effective scheme to obtain optimized coverage in the monitoring field. Thus the performance of WSN is a function of sensor nodes deployment.

We can classify the node deployment in two categories: static deployment and dynamic deployment. The static node deployment selects the best location according to some optimization strategy, and the location of the sensor nodes will remain fixed during the lifetime of the WSN. At present, the static deployment is classified in two categories that includes the deterministic deployment and the random deployment. In the study of the deterministic sensor node deployment based on the target coverage, X. He et al. **[12]** have proposed a deployment approach of sensor nodes using the maximum multi-overlapping domains of target points and the genetic algorithm, which can reduce the network deployment cost and can realize the optimal allocation of space resources in wireless sensor networks. X. M. Guo, C. J. Zhao, X. T. Yang et al. **[13]** have proposed how to do target coverage based on grid scan. Moreover, Y. Z. Zhang, C. D. Wu, L. Cheng et al. **[14]** have proposed the deployment of wireless sensor network in deterministic space with obstacles.

Robots can be used for dynamic deployment of sensors in order to get the maximum performance of sensor networks, sensor nodes need automatically move to proper location, then start to work. Sensors can be randomly thrown, and then a variety of optimization algorithms for deployment optimization can be used; such as virtual force algorithm **[15]**, virtual force oriented particles algorithm **[16]**, simulated annealing algorithm **[17]**, particle swarm optimization algorithm **[18]** and simulated annealing genetic algorithm **[19]** more details can be found in Haitao Zhang & Cuiping Liu **[20]**. In mobile sensor networks it has been shown that mobility can make the design of higher layer algorithms complicated, but it can also improve the network performance. There is one recent approach given by Valeria Loscrí, Enrico Natalizio & Francesca Guerriero **[21]** “Particle Swarm Optimization Schemes Based on Consensus for Wireless Sensor Networks” that effectively handles this problem but the algorithm does not consider whether the problem is multi-factor, multi-level search space or not.

Literature surveys help us to conclude that the formulation of WSNs issues as multidimensional optimization problems and through soft-computing techniques is very promising research field.

3.1. PSO Scheme Based on Consensus for Wireless Sensor Networks

PSO is a very versatile population-based swarm intelligence technique **[21]**. The particles of PSO are located within a research space and evaluate a fitness function based on their position. They can move around the research space by combining their history with the information received by one or more neighbors in the swarm. If we assume that the particles move in a 2-D space, in PSO the new velocity of the particle i will be calculated as:

$$v_i(t+1) = \omega \,.\, v_i(t) + \emptyset_p \,.\, r_p \; o \left(p_i - x_i(t)\right) + \emptyset_g . r_g \; o \left(p_g - x_i(t)\right) \quad \textbf{(3.1)}$$

Where $x_i(t)$, $v_i(t)$, p_i, p_g , r_p and r_g are all R^2 vectors. Specifically x_i , v_i are the current position and velocity of the particle i, respectively; p_i is the particle's best known position, p_g is the swarm's best position; r_p , r_g are two random vectors in $U(0,1)$(the p and g in the subscript stand for personal and global respectively); $\omega, \emptyset_p , \emptyset_g$ are parameters selected in order to control the efficacy of the PSO technique, and o is the Hadamard matrix multiplication opera- tor. Therefore, the position of the particle at the next time instant is calculated as

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad \textbf{(3.2)}$$

The three terms of the velocity update equation (1) characterize the particles' behavior: the first term is called inertia (or moment) because it keeps track of the particle's previous velocity, the second term is the cognitive component and is used to make particles keep track of their best known positions, the third term is the social component be- cause it represents the best position achieved by the swarm.

3.1.1. Consensus Algorithm

The study of information flow and interaction among multiple agents in a group plays an important role in understanding the coordinated movements of these agents. As a result, a critical problem for coordinated control is to design appropriate protocols and algorithms such that the group of agents can reach consensus on the shared information in the presence of limited and unreliable exchange and dynamically changing interaction topologies.

The term "consensus" in multi-agent systems indicates the process of reaching an agreement on a certain quantity of interest that depends on the state of all agents. A consensus algorithm states the in- formation exchange between an agent and all of its neighbors in the network. A basic variant of the PSO algorithm works by having a population of candidate solutions called particles. In our scenario the particles are the sensor devices that exchange information and move to reach a common objective. Vicsek proposes a consensus algorithm based on a time discrete model of n autonomous agents that move in a research space. The state of each agent is updated by a rule based on its state and the state of its neighbors, where its neighbors are all the agents located in a limited transmitting/receiving range. The evolution of the system can be described as follows:

$$x_i(t+1) = x_i(t) + u_i(t) \quad \textbf{(3.3)}$$

$$\text{Where, } u_i = (\omega_{ii} - 1) \,.\, x_i(t) + \sum_{j \in N_t(t)} \omega_{ij}(t) \,.\, x_j(t) \quad \textbf{(3.4)}$$

In (3) and (4), x_i is the status of agent i , u_i is the control law on agent i , N_i is the set of neighbors of node i and ω_{ij} is the weight associated to the contribution of the j^{th} agent on the i^{th} . A neighbor is a 1 hop distance node. The strategy of consensus consists in choosing the right weights in order for all the agents to converge to the same state. It is worth to notice that a global consensus is reached thanks to the nodes at the boundaries that make consensus regions physically connected.

3.1.2. Modified PSO

It has been considered that the PSO scheme is a good solution for sensors network coverage problems, above all for the applications where the energy cost for moving nodes is not too high. Usually, in the PSO scheme, the social component is the best position achieved globally by the swarm in the research space. It is not useful to consider a global best position, because it implies a centralized scheme of control or, at least, the capacity of the nodes to communicate with every other node in the sensor field. In order to take into account the limited communication capabilities of sensors, it's stated that the social term involves the position that enjoys the maximum consensus within each node's neighborhood, where a neighborhood is composed only of the sensors within its transmitting/receiving range. Thus, it's assumed that at each iteration of the modified PSO algorithm, the sensors exchange information and determine the maximum consensus in their neighborhood. The velocity update equation is modified as follows:

$$v_i(t+1) = \omega \,.\, v_i(t) + \emptyset_p \,.\, r_p \; o \left(p_i - x_i(t)\right) + \emptyset_g \,.\, r_g \; o \; (l_i) \quad \textbf{(3.5)}$$

$$l_i = \frac{x_k - x_i}{||x_k - x_i||} \,.\, \frac{||x_k - x_i||}{d_{rep}} \quad \textbf{(3.6)}$$

In Eq. (3.6), x_k is the position of the particle in the set of neighbors of i that obtained the best value of objective function, and d_{rep} is a coefficient of repulsion that is meant to avoid sensors' coverage area overlapping. The computation of the objective function is based on the number of events per unit of time that occurred in the sensor field, by assuming that the nodes can have a global or a local knowledge of the sensor field. It is worthy to note that we used the maximum consensus because only the node that achieves the highest value of objective function is considered in the velocity update equation.

3.2. Dynamic Deployment of Wireless Sensor Networks by Artificial Bee Colony Algorithm

The ABC algorithm, a new swarm intelligence method inspired by the intelligent foraging behavior of honey bees, is used for the dynamic deployment problem of WSNs. The aim of the optimization technique is to maximize the coverage rate of the network, as given in Eq. (3.1). In the network's scenario, it is assumed that:

- The detection radii of the sensors are all the same(r).
- All of the sensors have the ability to communicate with the other sensors.
- All sensors are mobile.

In the ABC algorithm, the position of a food source represents a possible solution to the optimization problem and the nectar amount of a food source corresponds to the quality (fitness) of the associated solution. Therefore, the deployment of the sensors in the sensed area (each solution of the deployment problem) refers to a food source in the algorithm. The coverage rate of the network, i.e. the total coverage area, corresponds to the fitness value (nectar) of the solution. In the ABC model, artificial bee colonies, in which the goal of the bees is to find the best solution, comprise 3 groups of bees: employed bees, onlookers, and scouts. A bee waiting in the dance area to determine the choice of a food source is an onlooker, and when a bee goes to a previously visited food source, it is an employed bee. A bee that carries out random searches is called a scout.

The steps of the algorithm are as follows:

1. Initialize the parameters: detection radius r, size of area of interest A, number of mobile sensors m, colony size cs , maximum number of iterations Max Cycle, and limit for scout l

2. Deploy m sensors randomly for each food source x_i of employed bees using Eq. (3.7)

$$x_{ij} = min_j + rand(0,1)(max_j - min_j) \quad \textbf{(3.7)}$$

3. Evaluate the population.

4. c = 0.

5. **Repeat** steps 6 to 14

6. Produce new solutions $\boldsymbol{u_i}$ in the neighborhood of x_i for the employed bees using Eq. (3.8).

$$u_{ij} = x_{ij} + \emptyset_{ij}(x_{ij} - x_{kj}) \quad \textbf{(3.8)}$$

Here, k is a solution in the neighborhood of $i, \emptyset$ is a random number in the range [-1, 1], and j is the randomly selected mobile sensor's position.

7. Check $\boldsymbol{u_{ij}}$ for staying in the bounds of the area.

8. Apply the greedy selection process between $\boldsymbol{x_i}$ and $\boldsymbol{u_i}$.

9. Calculate probability values $\boldsymbol{p_i}$ for solutions $\boldsymbol{x_i}$ by means of their fitness values using Eq. (3.9).

$$p_i = \frac{0.9 \times fit_i}{fit_{best}} + 0.1 \quad \textbf{(3.9)}$$

10. Produce the new solutions, $\boldsymbol{u_i}$, for the onlooker bees from solutions $\boldsymbol{x_i}$, selected depending on Pi , and evaluate them.

11. Apply the greedy selection process for the onlookers between $\boldsymbol{x_i}$ and $\boldsymbol{u_i}$.

12. Memorize the best solution achieved thus far.

13. Determine the abandoned solution; if it exists, replace it with a new randomly produced solution using Eq. (3.7).

14. $\boldsymbol{c = c + 1}$.

15. Until $\boldsymbol{c} = MaxCycle$.

Each solution represents an array that has m items. Figure 3.1 shows the solution array. Items of the solution array are (x, y) positions of the mobile sensors in the network.

1	2	3	…	2 $\boldsymbol{m}$
$(\boldsymbol{x_1 - y_1})$	$(\boldsymbol{x_2 - y_2})$	$(\boldsymbol{x_3 - y_3})$		$(\boldsymbol{x_m - y_m})$

Figure 3.2: Solution Array

CHAPTER 4

PROPOSED WORK

4.1. Problem Definition

The problem under consideration is to develop an approach to find out optimal number of sensor nodes that can be deployed in a specified target field to get the maximum coverage area while ensuring the node connectivity during the lifetime of WSN.

4.1.1. Coverage Problem

The problem requires to obtain a minimum number of nodes to be deployed in the target field, such that every point in the target field is covered optimally. In general, given a set of sensor nodes to be deployed in a specified target field, the problem requires to determine if all points in the field is sufficiently k-covered, i.e. every point in the target field is covered by at least k sensor nodes, where k is defined as a given parameter.

In the proposed work area-coverage is taken into consideration whereby the main objective of the sensor network is to monitor a specified region (the collection of all points in the target field), and each point of the field has to be monitored.

4.1.2. Connectivity Problem

By connectivity we mean that the underlying graph of the network is connected in the sense that between any two nodes there is a path in the graph. The path in the graph represents single-hop or multi-hop communication between the nodes. In general the problem is considered as k-connectivity problem if removal of any (k-1) nodes does not leave the underlying graph disconnected. More formally for homogeneous network, the problem can be stated as follows: given a set of sensor nodes to be deployed in a specified target field, the problem requires to determine minimum value of communication range (R_c) that can be assigned to all the sensor nodes to ensure global connectivity. In the proposed work k is taken as 1.

In figure 4.1 the problem of coverage and connectivity has been shown, here sensor nodes are deployed in a specified target field. The purpose of sensor nodes are to gather the information about a phenomenon in the target field and disseminate this information to the data fusion center or the sink node ultimately. Here, nodes 1 and 2 are not covered and connected.

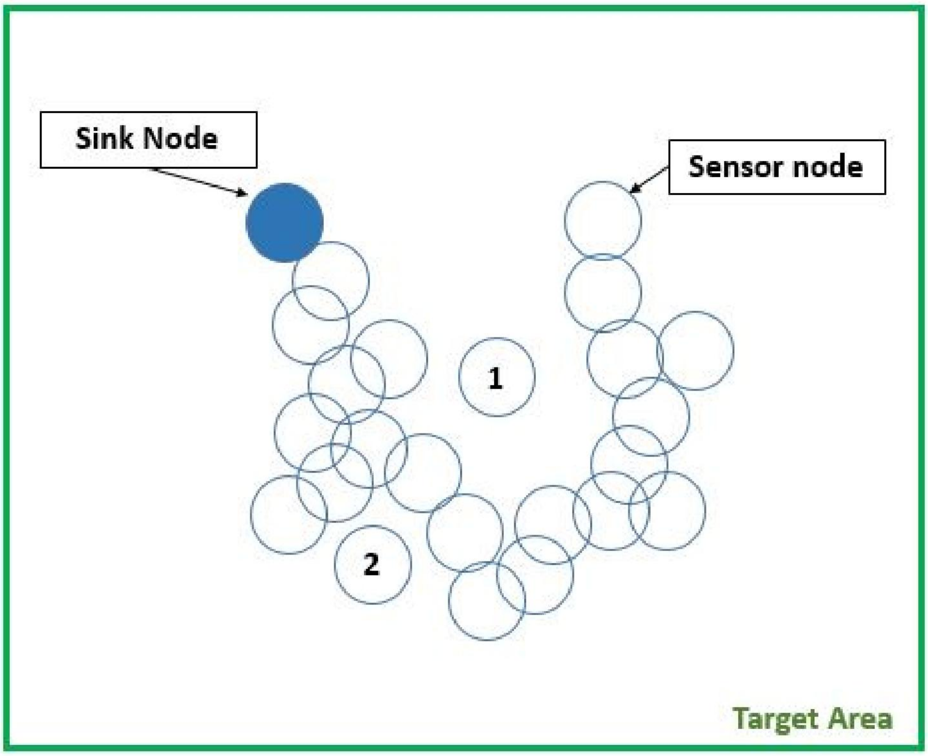

Figure 4.1: An illustration of coverage with connectivity. Sensor node 1 and 2 are not covered and connected in the target area

4.2. Proposed approach for coverage problem

In this section, first we give some relevant details about TLBO and OTLBO and then we introduce how this particular scheme can be utilized for our purpose, more details about TLBO and OTLBO are given by Chandra Satapathy, Anima Naik, & K. Parvathi **[23]**. The details of Connectivity Preservation and Coverage Schemes for Wireless Sensor Networks is given by Tahiry Razafindralambo & David Simplot-Ryl **[24]**.

4.2.1. Teaching Learning Based Optimization (TLBO)

The scheme TLBO is divided into two phases. The first part consists of the *Teacher Phase* and the second part consists of the *Learner Phase*.

4.2.2. Teacher Phase

As teacher is considered most knowledgeable person in the society, the best learner in the class is treated as a teacher. The teacher tries to disseminate its knowledge among learners that may in turn increase the knowledge level of the whole class and help learners to get better marks or grades. So a teacher increases the mean of the class according to its capability i.e. the teacher T_i will try to move mean M_i towards its own level according to its capability, thereby increasing the learners' level to a new mean M_j. Teacher T_i will put maximum effort into teaching its students, but students will gain knowledge according to the quality of knowledge delivered by a teacher and the quality of students present in the class. The quality of the students is judged from the mean value of the population. Teacher T_i puts effort in so as to increase the quality of the students from M_i to M_j, at which stage the students require a new teacher, of superior quality than themselves, and i.e. in this case the new teacher is T_j.

Let M_k be the mean and T_k be the teacher at any iteration *k*. T_k will try to move mean M_k towards its own level, so now the new mean will be M_{new}. The solution is updated according to the difference between the existing and the new mean given by

$$X_{new,k} = X_{old,k} + r_k(M_{new} - T_F M_k) \quad \textbf{(4.1)}$$

Where T_F is a teaching factor that decides the value of mean to be changed, $X_{new,k}$ and $X_{old,k}$ are new and existing solution. In this equation T_F can be calculated as follows:

$$T_F = round[1 + rand(0,1) \times \{2 - 1\}] \quad \textbf{(4.2)}$$

4.2.3. Learner Phase

A learner learns something new if the other learner has more knowledge than her. Learner modification is expressed as follows:

$\boldsymbol{for}\ i = 1\ \boldsymbol{to}\ P_n\ \boldsymbol{do}$

$\quad Randomly\ Select\ two\ learners\ X_i\ and\ X_j,\ where\ i \neq j$

$\quad \boldsymbol{if}\ f(X_i) < f(X_j)$

$\qquad X_{new,i} = X_{old,i} + r_i(X_i - X_j)$

$\quad \boldsymbol{else}$

$\qquad X_{new,i} = X_{old,i} + r_i(X_j - X_i)$

$\boldsymbol{end}$

Accept X_{new} if it gives a better function value.

4.3. Orthogonal Design

To understand Orthogonal Design let us consider a problem having some factors, each of which has several possible values called levels. Suppose that there are F factors and each factor has L levels, then total no of combination is L^F, we can see here that for large F and L it is not practical to evaluate all combinations.

Orthogonal design has been developed as a mathematical tool to study multi-factor, multi-level problems. It aims to extract an orthogonal array A of R rows, where each row represents a combination to be evaluated. The array has three key properties:

- During the experiment, the array represents a subset of R combinations, from all possible L^F combinations. Computation is reduced considerably because R << LF.
- Each column represents a factor. If some columns are deleted from the array, it means a smaller number of factors are considered.
- The columns of the array are orthogonal to each other. The selected subset is scattered uniformly over the search space to ensure its diversity.

A simple but efficient method is proposed in the work (Wing-Leung & Yuping **[22]**) to generate an orthogonal array A where, $R = L \times L\ and\ F = L + 1$. The steps of this method are shown in the algorithm described as follows:

Algorithm-1: Procedure for generating an Orthogonal array A.

$\boldsymbol{input}: The\ number\ of\ levels\ L$

$\boldsymbol{output}: An\ orthogonal\ array\ A$

$Calculate\ R = L \times L\ and\ F = L + 1$

$Initialize\ a\ zero\ matrix\ A\ with\ R\ rows\ and\ F\ columns.$

$\boldsymbol{for}\ i = 1\ \boldsymbol{to}\ R\ \boldsymbol{do}$

$A_{i,1} = mod(\lfloor (i-1)/L \rfloor, L)$

$A_{i,2} = mod(i-1, L)$

$\boldsymbol{for}\ j = 1\ \boldsymbol{to}\ F-2\ \boldsymbol{do}$

$A_{i,2+j} = mod(A_{i,1} \times j + A_{i,2}, L)$

$\boldsymbol{end}$

$\boldsymbol{end}$

4.3.1. Optimizer (OTLBO)

In this approach each learner in the class of learners can be divided into several partial vectors where each of them acts as a factor in the orthogonal design. Orthogonal design is then employed to search the best scales among all the various combinations.

4.3.2. OD-based operator and updating strategy

If there are m learners, we can execute the multi-parent problem efficiently using orthogonal design. Since each learner consists of n factors, there are m^n combinations. Hence orthogonal design is employed to select m (here A_m (L^F) is the orthogonal array, where F = n and L = m) representative sets of combinations to reduce the computational time. The procedure of OD-based multi-parent is given in Algorithm 2.

Algoritm-2: OD-based operator for m learners.

$\boldsymbol{input}: m\, particles\ X_{i,j},\ \ i\ \epsilon\ [1, m]\ and\ j\ \epsilon\ [1, n]$

$\boldsymbol{output}: A\ new\ set\ of\ m\ learners\ \ p_{i,j}$

$Construct\ the\ orthogonal\ array\ A_{m\times m}\ (m^m + 1)$ using Algorithm1

$Delete\ the\ last\ (m + 1 - n)\ columns\ of\ A_{m\times m}(m^m + 1)$ to get

$B = A_{m\times m}(m^n)$

$Generate\ m\ new\ learners$:

$\boldsymbol{for}\ i = 1\ \boldsymbol{to}\ m\ \boldsymbol{do}$

$\quad \boldsymbol{for}\ j = 1\ \boldsymbol{to}\ n\ \boldsymbol{do}$

$\quad\quad index = B_{i,j}$

$\quad\quad p_{i,j} = X_{index,j}$

$\quad \boldsymbol{end}$

$\boldsymbol{end}$

4.3.3. Steps of OD-based TLBO

To obtain a more precise solution compared to the standard TLBO, the OD-based operator is employed. The elitism preservation strategy for upgrading the current population is proposed, in which the learner is updated only if its fitness is improved. The procedure for the OD-based TLBO is shown in Algorithm 3. A convergence criterion or the maximum run can be used as the termination condition.

Algorithm-3: OD-based TLBO

$\boldsymbol{input}: Population\ Size\ S, Maximum\ iterations\ K$

$\boldsymbol{output}: Best\ Learner\ g$

$Construct\ a\ random\ initial\ population\ of\ learners\ X_i,\ i\ \epsilon\ [1, s]$

$\boldsymbol{for}\ t = 1\ \boldsymbol{to}\ K\ \boldsymbol{do}$

$\quad \boldsymbol{for}\ i = 1\ \boldsymbol{to}\ S$

$\quad\quad \boldsymbol{do}\ Update\ learners\ through\ teacher\ phase\ as\ in\ original$

$\quad\quad\quad TLBO\ applying\ elitism\ preservation\ strategy$:

$\quad\quad\quad Select\ m\ particles\ randomly\ and\ execute\ algorithm\ 2.$

$\quad\quad\quad Set\ p_i = X_i^{t+1}\ if\ f(X_i^{t+1}) < f(p_i)$

$\quad\quad\quad Set\ g = arg\ min\ f(p_i)$

$\quad \boldsymbol{end}$

$Check\ the\ termination\ condition$

$\boldsymbol{end}$

4.4. Connectivity Preservation Algorithm

The deployment algorithm is localized and is based on potential field theory. Each node is considered as a particle and its movements are governed by the interactions with some of its neighboring nodes. The subset of neighbors with which a node interacts together with the direction has to choose only rely on the kind of coverage required while the distance to be covered by is constraint by the preservation of its connection to nodes in Relative Neighborhood Graph (RNG). The RNG is a suitable solution since its computation only requires local information. Moreover, the use of the Euclidean distance for computing the RNG strongly reduces the mean degree of the graph. The mean degree of an RNG is 3. In addition, one of the most interesting feature of an RNG is that, while removing some edges from the initial graph, the graph also preserves the connectivity, provided that the initial graph is connected.

Algorithm-4: MSD (Mobile Sensor Deployment) protocol

Part 1:- Direction computation on node u:

1: $\underset{\Delta}{\rightarrow}(x(\Delta),(y\ \Delta))$ Is the direction vector of u

2: $\underset{\Delta}{\rightarrow} = \sum_{\substack{v \in RNG(u) \\ d(u,v)<R}} \big(R - d(u,v)\big) \times \left\|\underset{vu}{\rightarrow}\right\|$

3: $x(\Delta) = \sum_{\substack{v \in RNG(u) \\ d(u,v)<R}} \big(x(u) - x(v)\big) \times \big(R - d(u,v)\big)/R$

4: $y(\Delta) = \sum_{\substack{v \in RNG(u) \\ d(u,v)<R}} \big(y(u) - y(v)\big) \times \big(R - d(u,v)\big)/R$

5: $P(x(u) + x(\Delta), y(u) + y(\Delta))$

6: $\underset{\Delta}{\rightarrow} = \underset{\Delta}{\rightarrow} / \left\|\underset{\Delta}{\rightarrow}\right\|$

7: return $\underset{\Delta}{\rightarrow}$; // unit vector that gives the direction of the node

Part 2.a:- Speed computation on node u:

1: $v = (R - d^{+}(u))/\delta \times 2);$

2: $v = \min(v,\ v_{max})\,;$

Part 2.b:- Distance computation for node u:

1: $d_{max} = \max\big(\epsilon, R - d^{+}(u)\big)\,;$

2: $d_{opt} = \{d \in [d_{max}] |\ \forall\ v \in RNG(u), d(u_{new}, v) + v_{oth} \times\ \delta < R\}$

3: where $v_{oth}(v) = (R - d(u,v))/(\delta \times\ 2)$

4: and u_{new} is the new position of node u based on:

5: Direction $\underset{\Delta}{\rightarrow}$ from Part 1.

6: Speed v from Part 2.a.

7: Distance d.

Part 2.c:- Node u destination and movement:

1: Move to u_{new} using:

2: Direction $\underset{\Delta}{\rightarrow}$ from part 1.

3: Speed v from Part 2.a.

4: Distance d_{opt} from Part 2.b.

5: Take field border into account

4.4.1. Analysis of Algorithm 4

The step by step analysis of algorithm 4 is given as follows, this analysis is divided into two parts, and these two parts are independent to each other for choosing any deployment scheme in part 1.

Part 1: Direction Computation

The direction to be chosen by node u depends on the deployment requirements. Part 1 provides a normalized vector $\underset{\Delta}{\rightarrow}$ giving the direction of node u.

Part 2: Connectivity preservation

This part is divided in several steps: (a) Speed computation, (b) Distance computation and (c) Destination and movement.

Part 2.a: Speed computation

Based on the information gathered from its neighborhood, a node u computes its movement speed. This speed has to be as fast as possible (to allow a fast deployment of nodes) while preserving connectivity. This maximum speed is thus divided by two to consider the worst case movement of $RNG^{+}(u)$. It can't be null to still allow some small movements of node u. This also allows nodes that are at distance R to move toward each other. At the end of Part 2.a, node u knows its movement speed v.

Part 2.b: Distance computation

In this part, node u computes the distance it has to cover before running Algorithm 1 again. This distance is chosen in a given range $[\epsilon, d_{max}]$(Line 1 of Part 2.b) where ϵ is a parameter of the algorithm and is used to allow some small movements when needed. To do so, it computes the maximal speed of its RNG neighbors V_{oth} in such a way that even in the worst case, i.e. if its further RNG neighbor $RNG^{+}(u)$ goes in the opposite direction, they both remain connected. This means that in the worst case, they can both travel a distance equals to $d = (R - d^+(u))/2$ before being disconnected. Indeed, $RNG^+(u)$ can't chose a higher distance than u since, as links are bidirectional $d^+(RNG^+(u)) \geq d^+(u)$. This distance has to be covered between two checking, i.e. in δ. Based on these information, node u computes the maximum possible distance d_{opt} it can travel given that in its new position, it still remains connected to its $RNG_{(u)}$ nodes.

Part 2.c: Destination and movement

In this part, the path planning of the sensor is considered. Given a direction, a speed and a distance, any path planning algorithm can be used for sensor movement. The distinction between Part 3 and Part 2 allows us to use a path planning and obstacle avoidance algorithm from robotics such as the one described in [29]. This allows un-modified motion planning algorithms to control the trajectories of each mobile sensors. It is important to note here that Part 1 of Algorithm 1 is completely independent from the other parts. This is an important property since the direction of nodes can be easily modified to fit some other.

4.5. Proposed Hybrid Approach

Algorithm-5: Sensor node deployment while preserving connectivity

1. CALL Algorithm 3 to find optimal no. of nodes, N in the sensor field, F.

2. for each node u in the sensor field, F

3. CALL Algorithm 4

4.5.1. Analysis of Hybrid Approach

The algorithm is based on two algorithms, the first algorithm is for optimal deployment of sensor nodes based on the Orthogonal Teaching Learning Based Optimization (OTLBO) which has been proved to be an improvement over Teaching Learning Based Optimization (OTLBO) in terms of convergence rate and is based on orthogonal design technique for multi-factored and multi-leveled problems.

The Algorithm 3 returns the optimal number of nodes that can be deployed on the specified field size, this process is again done in two steps, the first step is to deploy these nodes randomly on the field and then these nodes will move to their optimal position according to a bio-inspired optimization technique to get the optimal positions for the deployment. The second step is to use Algorithm 4 to maintain the connectivity of these nodes during the lifetime of WSN. This connectivity preservation algorithm is taken from "Connectivity Preservation and Coverage Schemes for Wireless Sensor Networks" by Tahiry Razafindralambo, and David Simplot-Ryl, Member, IEEE.

This hybrid approach is designed to get the better approach for the coverage problem and the connectivity problem for mobile nodes in WSN. The connectivity graph considered in Algorithm 4 is Unit Disk Graph (UDG) model, this algorithm also valid for realistic graph (graph maintained by real nodes in the network).

CHAPTER 5

SIMULATION RESULTS AND ANALYSIS

Simulation has been defined as computer based statistical sampling experiment on the digital computer, it’s the process of mimicking the system or problem under consideration for the purpose of analysis. Simulation is considered as very efficient and suitable tool to analyze the performance of the system under consideration.

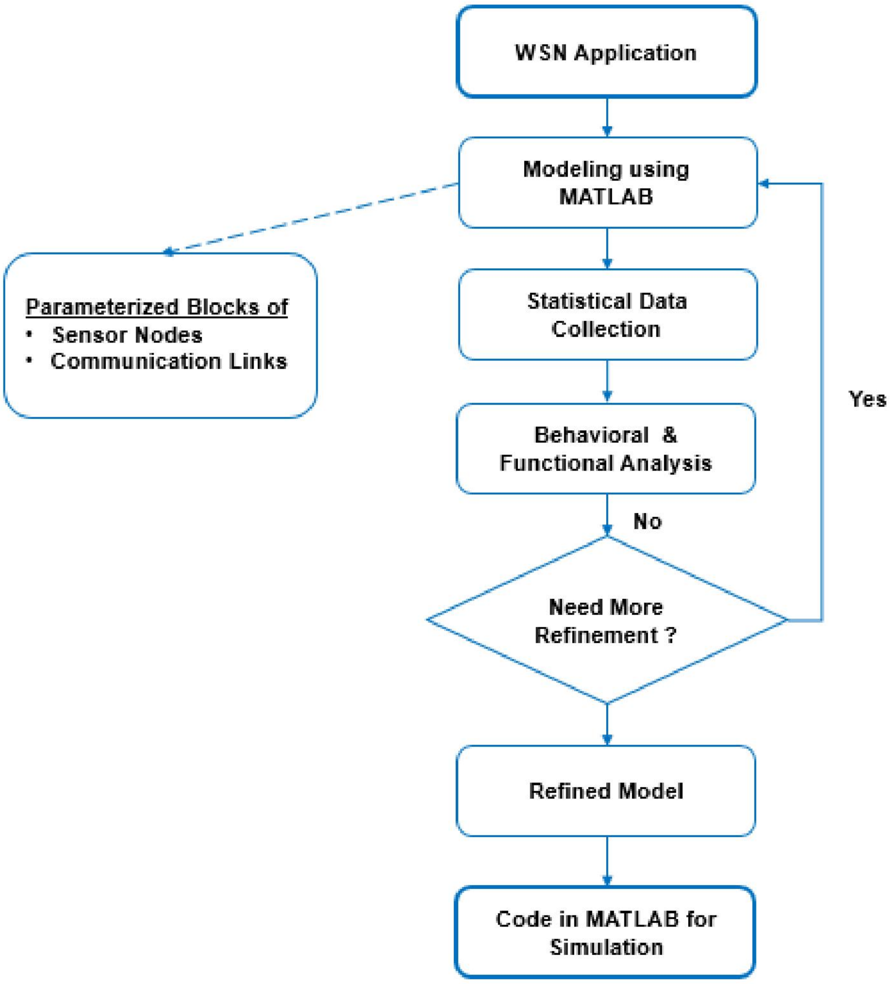

Figure 5.1: Process of simulation

The overall simulation process is described in the figure 5.1, Wireless Sensor Model is represented by a system model with simplified assumptions. Following section describes the system model and the assumptions taken under consideration.

5.1. System Model

A point in the target field is said to be covered if that point can be detected by one or more sensor nodes. Let R_s be taken as the *sensing range* of a sensor node then we say a location is covered if its distance to at least one sensor node is within R_s.

Two sensor nodes are said to be directly connected if the distance between the nodes is less than the *communication range,* R_c. In WSN, coverage and connectivity both are important sometimes, because the ultimate aim of wireless sensor network is to gather and disseminate the information to the sink node. This is why, the connectivity, which is defined as the ability to disseminate the information to the sink node, is also considered to be critical. In this regard the sensor nodes are randomly deployed in the target field first then these nodes are considered to be moving nodes and find out their optimal positions in the field according to *Orthogonal Teaching Learning Based Optimization (OTLBO)* technique which is a *bio-inspired optimization* approach.

For the analysis of OTLBO algorithm on the given problem, we consider the monitoring field as a square sensor field (Z), where a certain number of mobile sensors (N) is deployed randomly or uniformly. Events happen in the limited and measurable space according to a probability density function F(z), where z is the surface unit over Z. The idea is to distribute the sensors present in the sensor field according to the events probability function. The fraction of nodes n that should cover a region R of Z can be computed by the following ratio of surface integrals:

$$n = \frac{\int_R \mathrm{F(z)\,dz}}{\int_Z \mathrm{F(z)\,dz}} \quad \textbf{(5.1)}$$

The objective function consider the position of the node in respect of the events that occurred in the whole sensor field. Hence, the objective function needs global information. To evaluate the scheme, we consider the difference between the ideal numbers of sensors in each area ($n_i.N$) and the number of sensors located in the same area as the performance parameter at the time of convergence. The Energy for the movement has been set to a proportional cost model that depends on traveled distance d, as follows:

$$E_m(d) = kd \quad \textbf{(5.2)}$$

Where the constant k can take value between 0.1 and $1\frac{J}{m}$. This system model considers two objective functions Global and Local Objective Functions calculated at each iteration by each node. GOF takes into consideration the position of the node in respect of the events that occurred in the whole sensor field. Hence, this version needs global information. LOF takes into consideration only the events that occurred within the node's sensing area. In this case, each node can calculate the objective function depending only on local information and it is called Local Objective Function.

For the purpose of simulation we draw concentric circles of inner radius R_s and outer radius R_c to represent sensor nodes. If two circles overlap, then we say, they are connected nodes and thus can communicate with each other. We assume that the target area to be considered is sufficiently larger than the sensing and communication ranges of the sensor nodes, and thus we assume there is no *boundary effect*.

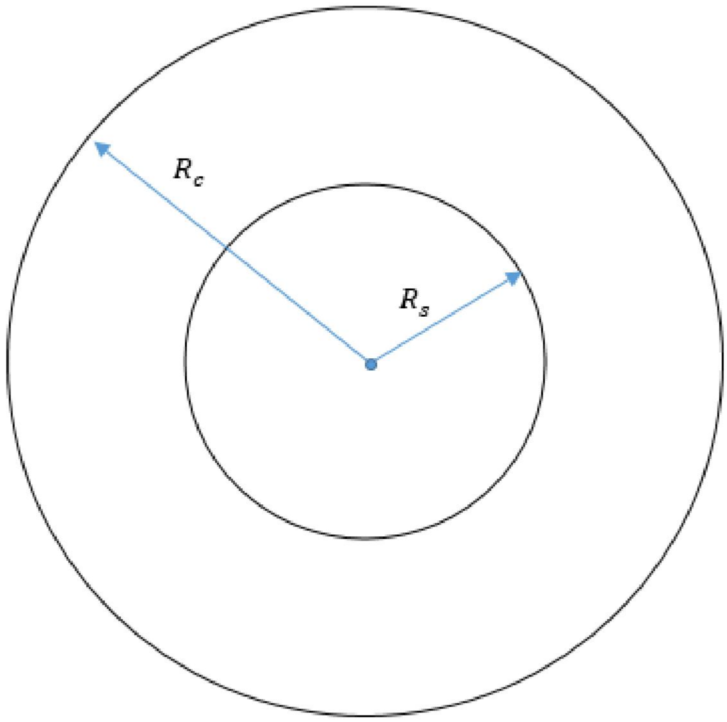

Figure 5.2: A model of a sensor node

We have considered a *homogeneous network* i.e. all the sensors are alike when sensing and communication capabilities are considered in the network, we have considered *individual detection model* in which each node detects an event independently, furthermore a range-based *disk model* is considered for sensing and communication in which each sensor is capable of detecting points only within distance R_s for sensing purpose and within distance R_c for communication purpose, no node failure is assumed in the network. The initial deployment of sensor nodes in the target field is done using *random deployment scheme* with uniform distribution.

For the purpose of connectivity it is assumed that all the communicational link are bidirectional, and because all the nodes are alike, they have same communication capacities, thus the communication graph can be modeled as Unit Disk Graph (UDG).

Let k is the node density, N nodes are deployed randomly with uniform distribution in the target field of size A, and then the deployment model is in accordance with equation 5.3:

$$N = kA, \text{ where } A \text{ goes to infinity} \quad \textbf{(5.3)}$$

5.2. Simulation Setup

Simulation is done in MATLAB, the purpose of the simulation is to compare the performance of the proposed algorithm i.e. node deployment using orthogonal teaching learning based optimization which is a bio inspired optimization method to obtain the best location of nodes for the coverage problem, while ensuring the connectivity of the nodes all the times in the network.

For the purpose of simulation we have taken isotropic sensors in which sensing area is considered as a circle in 2D or a disc in 3D, binary detection sensing model is taken as a detection model that assumes perfect detection i.e. probability of detection is 1.0 if the point under consideration is within the circle otherwise its treated as 0. The target field is a square field with dimensions 100m X 100m, number of sensor nodes to be deployed is 50, sensing range is 12.5m, communication range is twice of the sensing range, number of maximum runs is 100, maximum velocity of nodes is 10 m/s,

Table 5.1: Simulation Parameters

Sensor field size (L x L)	100m x 100m
Number of sensors	50
Sensing range (R_s)	12.5 m
Communication range (R_c)	2 R_s m
Number of runs	100
v_{max}	10 m/s
Time interval (δ)	5s
Movement step (ϵ)	0.1
Simulation time	5000s
Confidence Interval	95%

5.3. Performance Evaluation

For the purpose of result evaluation we need to consider the difference between the ideal number of sensors in the specified target field that is equal to $(n.N)$ and the number of sensor nods located in the same specified area, when the convergence condition of the Algorithm 3 is satisfied, as the performance parameter. Here noteworthy point is sensor nodes can have overlapping coverage areas, moreover we are using the mobile sensor nodes so it's important to quantify the energy spent on movement. Energy for the movement is considered to proportional to the energy cost model described in equation 5.2.

The performance evaluation is based on the objective function defined in equation 5.1 that gives the optimal number of fractional nodes to be deployed in the specified target field. The result is based on the initial random deployment in the target field then OTLBO is used to get optimal deployment in the field and finally the connectivity preservation algorithm is run to ensure the connectivity during the lifetime of the sensor nodes in WSN. The connectivity preserving algorithm uses different regular node patterns to compare the reachability of nodes to the sink node. The performance is checked assuming that the nodes in the initial phase are disconnected entirely and when the simulation is carried on for different time duration, produces a connected graph. During simulation presence of obstacles in the field is not considered for the connectivity point of view.

5.3.1 Performance Metrics

Two performance metrics, total area coverage and reachability evaluation are taken under consideration for analysis of the protocols are as follows:

- **Area Coverage:** Area covered by sensor nodes divided by total area of the target field.
- **Reachability Evaluation:** Connectivity graph is connected or not during lifetime of WSN.

5.3.2 Simulation Results

The simulation comparison is done with standard swarm based optimization technique known as Particle Swarm Optimization (PSO) technique, the simulation is written in MATLAB, the program runs for 1500 iterations.

In the PSO algorithm, velocity and position of the particles are updated by Equation (5.4) and Equation (5.5) as described in chapter 3.

$$v_{ij}(c+1) = \omega\,(c) \times v_{ij}(c) + c_1 r_{1i}(c)\left(y_{ij}(c) - x_{ij}(c)\right) + c_2 r_{2i}(c)\left(\hat{y}(c) - x_{ij}(c)\right) \quad \textbf{5.4}$$

$$x_{ij}(c+1) = x_{ij}(c) + v_{ij}(c+1) \quad \textbf{(5.5)}$$

Where c_1 and c_2 are acceleration constants, $r_{1i}(c)$ and $r_{2i}(c)$ are random numbers in range [0, 1]. $x_{ij}(c)$ and $v_{ij}(c)$ represents the position and velocity of i_{th} particle in j_{th} dimensions at time c, $y_i(c)$ is the local best position of i_{th} particle and $\hat{y}(c)$ is the global best position. The inertia weight $\omega\,(c)$ at time c is set by using equation (11):

$$\omega\,(c) = 0.9 - c/MaxNumber \times 0.5 \quad \textbf{(5.6)}$$

where $MaxNumber$ is the maximum number of cycles. PSO algorithms' swarm size is 20 and the acceleration constants are set c1 = c2 = 1.

The results are stored in table 5.2, for the performance measures mean, standard deviation (std) Best and the worst cases are listed for direct comparison. The results show the performance of OTLBO is better than PSO for all 30 independent runs each having 1500 iterations. Initial deployment is taken same for the fair comparison and to avoid the biasness in the results.

Table 5.2: Node Deployment Results

	Initial coverage of stationary sensors	**PSO**	**OTLBO**
Mean	0.7436	0.9368	0.9601
Std	0.0224	0.0128	0.0078
Best	0.7888	0.9581	0.9752
Worst	0.6975	0.9094	0.9365

5.3.3. Analysis

The final deployment of the sensor nodes is shown in figure 5.2, each node is given a unique identity to locate on the graph. The network topology is decided by the communication range. The communication graph is connected to ensure the connectivity of the nodes duration the lifetime of WSN.

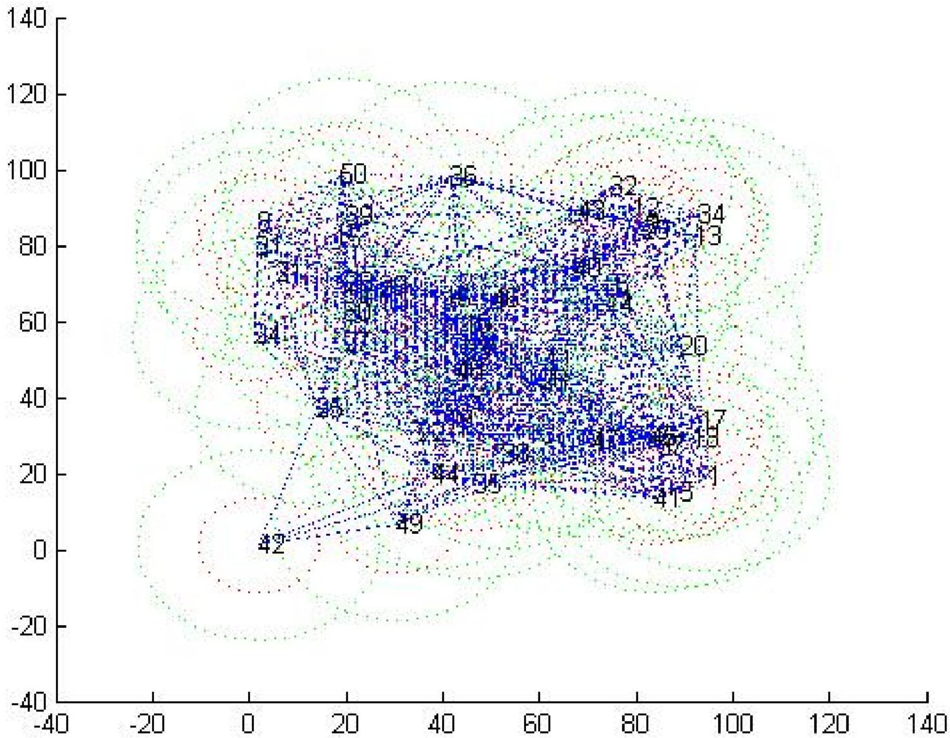

Figure 5.3: Final deployment of sensors

To compare the performance of the proposed approach, two graphs are plotted to demonstrate the best and worst case scenario. Mean of sample run is taken for 30 runs each one of them consists of 1500 rounds.

In figure 5.4 the graph shows that in the case of dynamic deployment of sensor nodes in a specified field Orthogonal Teaching Learning Based Optimization (OTLBO) outperforms Particle Swarm Optimization (PSO) approach, the figure shows the best case where the deployment of the node gives the maximum coverage on the field, given the initial deployment is done randomly and then the nodes they move to obtain their optimal position, then to maintain the connectivity as well the connectivity preservation algorithm is applied on the nodes that causes these nodes to move further at their best location. We have assumed here that no node can be entangled by any of the obstacles that may present in the ground. In figure 5.5, the worst case is shown whereby two above said approaches differ the most possible manner. The graph shows that the performance of the Orthogonal Teaching Learning Based Optimization (OTLBO) still performs better than Particle Swarm Optimization (PSO) in the same scenario but for different sets of runs.

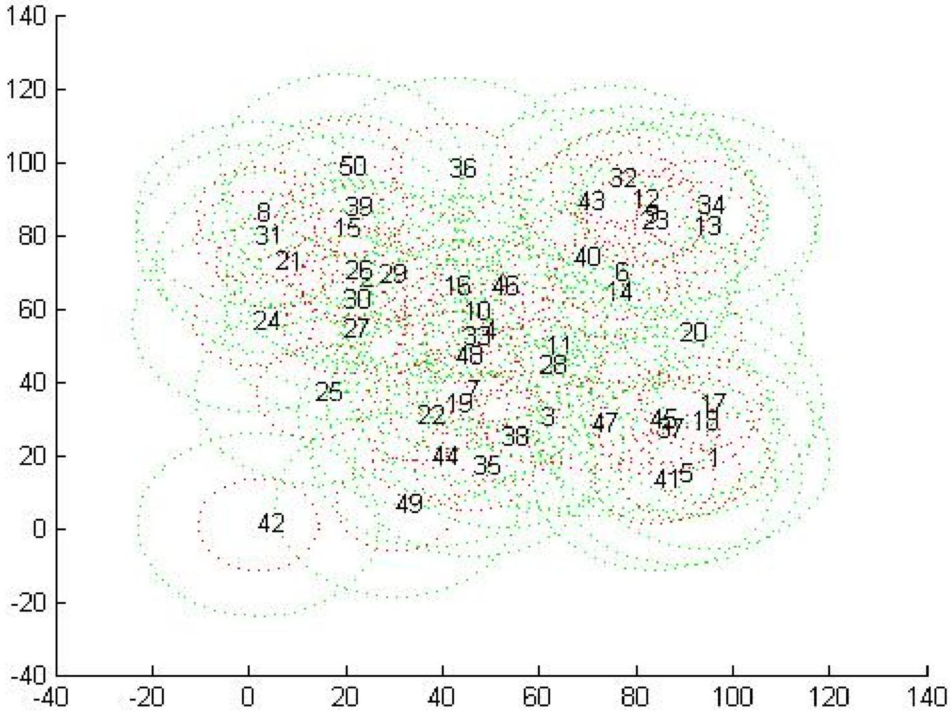

Figure 5.4: Final deployment of sensors without topology

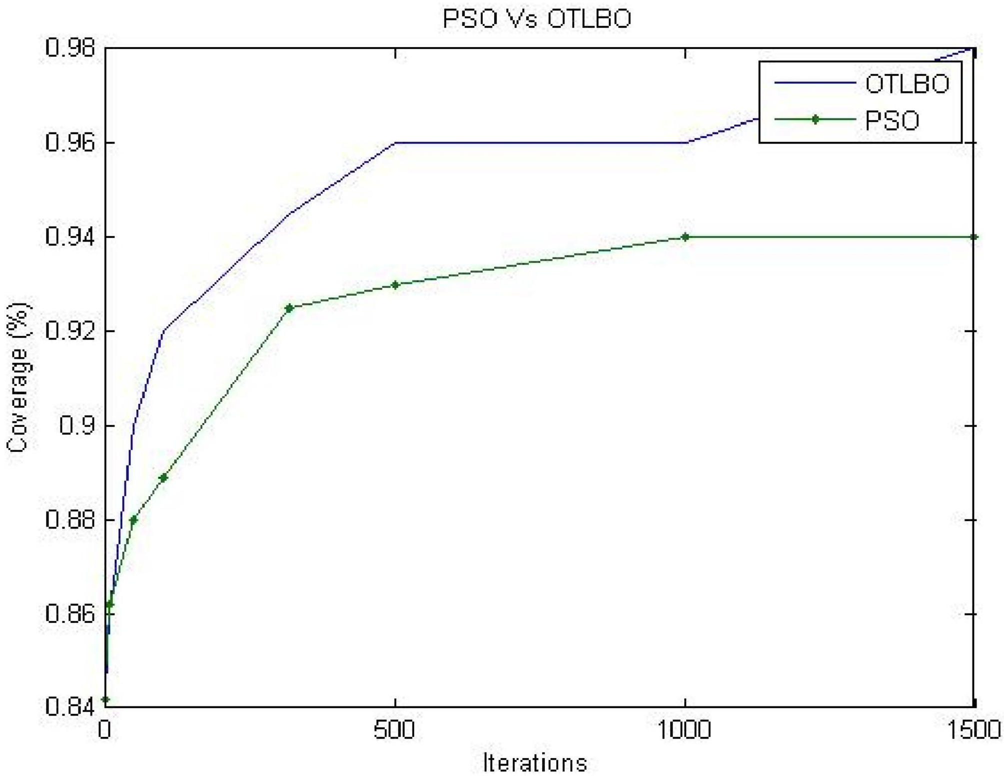

Figure 5.5: Average of 30 runs

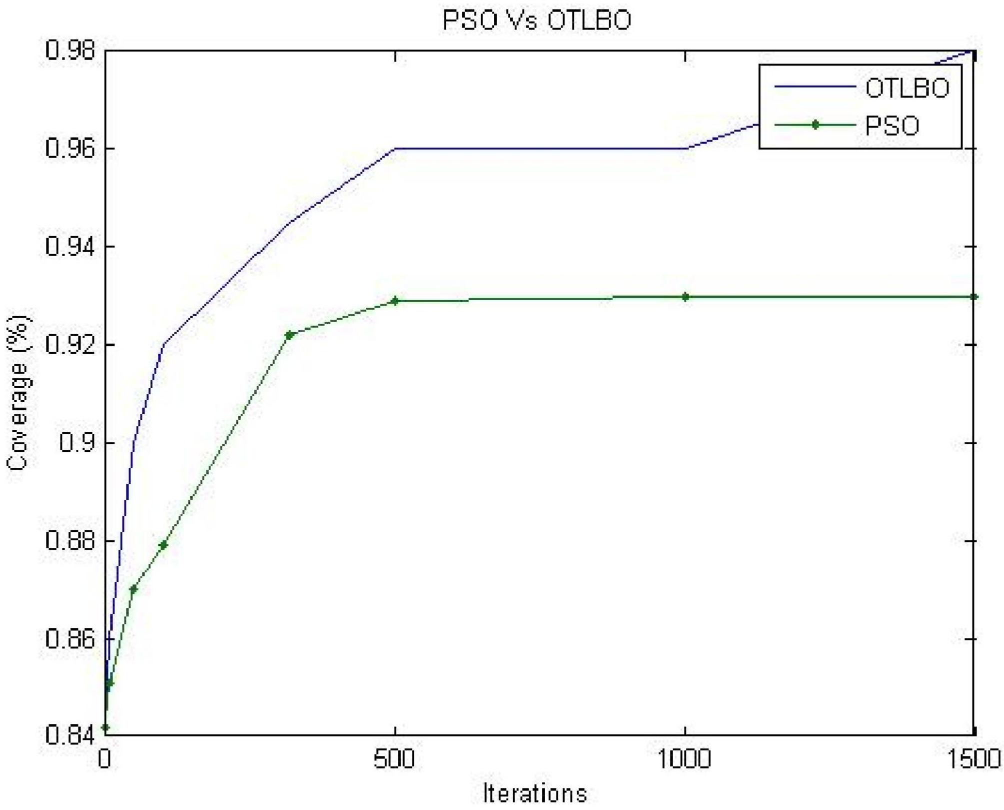

Figure 5.6: The most difference in a run

CHAPTER 6

CONCLUSION AND FUTURE WORK

Coverage is an important property in WSN, it's a performance metric of the sensor node deployment scheme. We can hereby conclude by saying that OTLBO is a very powerful approach of optimizing different types of problems which are separable, non-separable, unimodal and multimodal in providing quality optimum results in faster convergence time compared to very popular evolutionary techniques like PSO, GA and its variants. As a further research it remains to be seen how this adapts to multi-objective optimization problems and also some engineering applications from mechanical, chemical or data mining may be investigated. When we apply this technique on the sensor nodes deployment in WSN we can get better result, the convergence rate is faster because we are using orthogonal design on TLBO to obtain OTLBO which is an improvement over TLBO.

Connectivity is another important property in wireless networks and especially in wireless sensor networks. In this thesis a hybrid approach is developed for mobile sensor deployment with connectivity guarantee. The Algorithm 4 is divided into two independent parts. 1) Direction computation. In this part, the direction of the mobile sensor is computed depending on the application requirements. In this regard, we considered area coverage problem only in which the sensor node deployment approach tries to maximize the area coverage. We showed that our deployment scheme provides a coverage close to the regular pattern coverage with squares. In the connectivity preservation part, we have used a connectivity preservation scheme to avoid nodes to be disconnected during their deployment over a specified time duration in WSN. To pre-serve connectivity, nodes only maintain the connections with a sub-part of its neighbors during the deployment. The algorithm chooses the Relative Neighborhood Graph (RNG) since it can be computed locally and it can maintain the required global connectivity also. It's shown by analysis that if we have a perfect physical channel in the network then the connectivity is always guaranteed. It's noteworthy that we have considered Unit Disk Graph (UDG) model for description and evaluation of the algorithm but the model can also be replaced by any realistic models. The only thing that is required is that each node must be able to discover its neighborhood node and to must be able to compute a subset of its neighbors which guarantees the connectivity preservation.

The proposed work can be extended for k-coverage and k-connectivity problems whereby we can consider node failure case also. The proposed work can also be extended for taking the energy as a parameter, thus developing an approach for energy efficient approach for node deployment while ensuring the connectivity in the network.

References

[1] Jennifer Yick, Biswanath Mukherjee, Dipak Ghosal, “Wireless sensor network survey”, Elsevier, Computer Networks 52 (2008) 2292–2330.

[2] Jiming Chen, EntongShen and YouxianSun “The deployment algorithm in Wireless Sensor Network: A Survey” Information Technology Journal 2009.

[3] S. Toumpis, T. Tassiulas, Optimal deployment of large wireless sensor networks, IEEE Transactions on Information Theory 52 (2006) 2935–2953.

[4] J. Yick, G. Pasternack, B. Mukherjee, D. Ghosal, Placement of network services in sensor networks, Self-Organization Routing and Information, Integration in Wireless Sensor Networks (Special Issue) in International Journal of Wireless and Mobile Computing (IJWMC) 1 (2006) 101–112.

[5] D. Pompili, T. Melodia, I.F. Akyildiz, Deployment analysis in underwater acoustic wireless sensor networks, in: WUWNet, Los Angeles, CA, 2006.

[6] I.F. Akyildiz, E.P. Stuntebeck, Wireless underground sensor networks: research challenges, Ad-Hoc Networks 4 (2006) 669–686.

[7] M. Li, Y. Liu, Underground structure monitoring with wireless sensor networks, in: Proceedings of the IPSN, Cambridge, MA, 2007

[8] I.F. Akyildiz, D. Pompili, T. Melodia, Challenges for efficient communication in underwater acoustic sensor networks, ACM Sigbed Review 1 (2) (2004) 3–8.

[9] J. Heidemann, Y. Li, A. Syed, J. Wills, W. Ye, Underwater sensor networking: research challenges and potential applications, in: Proceedings of the Technical Report ISI-TR-2005-603, USC/Information Sciences Institute, 2005.

[10] I.F. Akyildiz, T. Melodia, K.R. Chowdhury, A survey on wireless multimedia sensor networks, Computer Networks Elsevier 51 (2007) 921–960.

[11] Mr. Mayur C. Akewar, Dr. Nileshsingh V. Thakur, “A Study of Wireless Mobile Sensor Network Deployment”, IRACST – International Journal of Computer Networks and Wireless Communications (IJCNWC), ISSN: 2250-3501 Vol.2, No4, August 2012

[12] X. He, X. L. Gui, and J. An, “A Deterministic Deployment Approach of Nodes in Wireless Sensor Networks for Target Coverage”, Journal of XiAn Jiaotong University Vol. 44, No.6, 2010, pp.6-10.

[13] X. M. Guo, C. J. Zhao, X. T. Yang et al, “A Deterministic Sensor Node Deployment Method with Target Coverage Based on Grid Scan”, Chinese Journal of Sensors and Actuators, Vol.25, No.1, 2012, pp.104-109.

[14] Y. Z. Zhang, C. D. Wu, L. Cheng et al, “Research of node deployment strategy for wireless sensor in deterministic space”, Control and Decision, Vol. 25, No. 11, 2010, pp. 1625-1629.

[15] Q. Luo, and Z. M. PAN, “An Algorithm of Deployment in Small-Scale Underwater Wireless Sensor Networks”, Chinese Journal of Sensors and Actuators, Vo. 24, No. 7, 2011, pp. 1043-1047.

[16] X. Wang, S. Wang, and J. J. MA, “Dynamic Sensor Deployment Strategy Based on Virtuai Force-Directed Particle Swarm Optimization in Wireless Sensor Networks”, Acta Electronica Sinica, Vol. 35, No. 11, 2007, pp. 2038-2042.

[17] Y. S. Lin, and P. L. Chiu, “A Near-optimal Sensor Placement Algorithm to Achieve Complete Coverage Discrimination in Sensor Network,” IEEE Communications Letters, Vol. 9, No. 1, 2005, pp43-45.

[18] Y. Liu, “Wireless Sensor Network Deployment Based on Genetic Algorithm and Simulated Annealing Algorithm”, Computer Simulation, Vol. 28, No.5, 2011, pp. 171-174.

[19] W. H. Liao, Y. C. Kao, and Y. S. Li, “A Sensor Deployment Approach Using Glowworm Swarm Optimization Algorithm in Wireless Sensor Networks”, Expert Systems with Applications, Vol. 38, No. 2011, pp. 12180–12188.

[20] A Review on Node Deployment of Wireless Sensor Network Haitao Zhang and Cuiping Liu IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 6, No 3, November 2012 ISSN (Online): 1694-0814

[21] Valeria Loscrí, Enrico Natalizio and Francesca Guerriero, “Particle Swarm Optimization Schemes Based on Consensus for Wireless Sensor Networks”.

[22] Wing-Leung Y, Yuping W. “An orthogonal genetic algorithm with quantization for global numerical Optimizationg” IEEE Trans Evol Comput. 2001; 5(1):40–53.

[23] Chandra Satapathy, Anima Naik, and K. Parvathi, “A teaching learning based optimization based on orthogonal design” *SpringerPlus* 2013, **2**:130 doi:10.1186/2193-1801-2-130.

[24] Tahiry Razafindralambo , Member, IEEE , and David Simplot-Ryl , Member, IEEE “Connectivity Preservation and Coverage Schemes for Wireless Sensor Networks” in IEEE TRANSACTIONS ON AUTOMATIC CONTROL, VOL. 56, NO. 10, OCTOBER 2011.