SELF HEALING CLUSTER-BASED TOPOLOGY CREATION & MANAGEMENT IN SECURE WIRELESS SENSOR NETWORK

By

Yasir Fayyaz

2008-NUST-MS PhD-CSE (E)-18

MS-08 (SE)

Submitted to the Department of Computer Engineering in fulfillment of the requirement for the degree of

MASTER OF SCIENCE
In
SOFTWARE ENGINEERING

Thesis Supervisor

Prof Dr Muhammad Younus Javed

College of Electrical & Mechanical Engineering

National University of Sciences & Technology

2011

بِسْمِ ٱللَّهِ ٱلرَّحْمَٰنِ ٱلرَّحِيمِ

In the name of Allah, the Most Beneficent, the Most Merciful

DECLARATION

I hereby declare that I have developed this thesis entirely on the basis of my personal efforts under the sincere guidance of my supervisor Prof Dr Muhammad Younus Javed. All the sources used in this thesis have been cited and the contents of this thesis have not been plagiarized. No portion of the work presented in this thesis has been submitted in support of any application for any other degree of qualification to this or any other university or institute of learning.

Yasir Fayyaz

ACKNOWLEDGEMENTS

I would like to thank Allah Almighty for giving me the courage, energy, wisdom and knowledge to complete my Master's Thesis.

I thank my parents for their prayers, encouragement and full support in the completion of this work.

I would also like to express my earnest gratitude to my advisor, Dr. Mohammad Younis Javed for his invaluable guidance, and making constructive comments during the development of this project. I would remain thankful to him forever.

I gratefully acknowledge the help and guidance provided by Guidance and Examination Committee members Dr. Khalid Iqbal, Dr. Aasia Khanum & Dr. Saad Rehman. Their valuable suggestions and comments were a great source to improve the research work presented in this thesis.

Dedicated to my parents and my supervisor
Who stood by me and remained a source of inspiration which inexorably helped me in completing my Masters degree.

ABSTRACT

Tiny devices known as sensor nodes capable of sensing computing and communicating simultaneously to form a wireless sensor network. WSN has emerged as a revolutionary technology which has brought with it numerous applications. Sensor nodes are placed in the hostile environment to monitor different type activities or changes. Different topologies to create a WSN from these sensor nodes have been proposed. Clustering is a form of topology in which sensor nodes organize themselves in the form of cluster which are common function to all the sensor nodes. It is important to describe an efficient topology discovery algorithm to find a set of distinguished nodes. The main objective of a clustering algorithm is to make the whole network connected in the form of efficient topology. A clustering algorithm organizes the nodes in two phases: set up is the first phase and maintenance of the cluster is the second phase. To optimally choose self healing cluster heads is an NP-hard problem. The research discusses and compares various strategies of self healing cluster formation in wireless sensor networks. Three popular techniques of clustering, namely highest degree, topology discovery and weighted clustering have been implemented. Two new algorithms, named maximal weight and bounded degree have been proposed for the design and implementation of self healing cluster based topology creation and management system. The detailed design and working of maximal weight and bounded degree algorithms are presented with the help of examples. The new proposed algorithms generate self healing clusters based, on minimizing reconfiguration, thus saving energy and optimizing the communication. The algorithms perform better than in comparison with the popular techniques in terms of number of rearrangements i.e. reaffiliations and dominant set updates, number of clusters, stability of clusters, and ratio of clusterheads to number of nodes. A detailed comparitive analysis of the proposed algorithms against the topology discovery and weighted clustering algorithm has been provided. The research work also highlights some open research areas for securing wireless sensor networks.

TABLE OF CONTENTS

DECLARATION III

ACKNOWLEDGEMENTS IV

ABSTRACT VI

CHAPTER 1 - INTRODUCTION 1

1.1. Issues and Challenges in Wireless Sensor networks 4

1.2. Definition of Clustering 6

1.3. The Clustering Process 7

1.4. Motivation for Using a Clustering Scheme 8

1.5. Challenges in Clustering 9

CHAPTER 2 - BACKGROUND AND LITERATURE SURVEY 10

2.1. Review of Various Clustering Techniques 10

2.1.1. Linked Cluster Algorithm (LCA) 10

2.1.2. Linked Cluster Algorithm 2 (LCA2) 11

2.1.3. Highest Connectivity Cluster Algorithm 11

2.1.4. Max Min D-Cluster Algorithm 11

2.1.5. Low-Energy Adaptive Clustering Hierarchy (LEACH) 12

2.1.6. Energy Efficient Clustering Scheme (EECS) 13

2.1.7. Hybrid Energy Efficient Distributed Clustering (HEED) 13

2.1.8. Topology Discovery Algorithm 14

2.1.9. Weighted Clustering Algorithm (WCA) 16

2.2. Review of Security for Wireless Sensor Networks 17

2.2.1. Security Challenges 17

2.2.2. Attacks 17

2.2.3. Security Requirements 18

2.2.4. Key Management 19

2.2.5. Authentication and Integrity 24

2.2.6. Secure Routing.. 26

2.2.6.1. Problems.. 26

2.2.7. Intrusion Detection and Counter Measures.. 28

2.2.8. Node Compromise.. 28

2.2.9. Active Attacks 28

CHAPTER 3 - DESIGN & IMPLEMENTATION OF SELF HEALING CLUSTER BASED TOPOLOGY CREATION & MANAGEMENT SYSTEM 30

3.1. Maximal Weighted Topology Discovery Clustering Algorithm for Sensor Networks..... 30

3.1.1. Devising a Weighted Function.. 31

3.1.2. Working of Algorithm 32

3.1.3. Flow Chart Diagram 34

3.1.4. Color Assignment Function.. 35

3.1.5. Example 36

3.1.6. Maximal Weight Cluster Update Function 40

3.2. Bonded Degree Clustering Algorithm.. 43

3.2.1. Basis of Algorithm.. 43

3.2.2. Working of Algorithm 44

3.2.3. Example 45

3.2.4. Bounded Degree Cluster Update Function.. 46

3.3. Graphical User interface of the System.. 47

CHAPTER 4 - SIMULATED RESULTS & DISCUSSION........................... 51

3.4. Network Model.. 51

3.5. Performance Metrics.. 51

3.6. Performance Comparison Reports 52

3.6.1. Average Number of Clusterheads 52

3.6.2. Number of Reaffiliations 53

3.6.3. Average Duration of a Clusterhead 55

3.6.4. Network Lifetime 56

3.6.5. Ratio of Black nodes to total number of nodes 58

3.7. Algorithms Analysis and Comparison with existing algorithms 59

3.7.1. Maximal Weight Topology Discovery Algorithm Comparison Table 59

3.7.2. Bounded Degree Algorithm Comparison Table 61

CHAPTER 5 - CONCLUSION & FUTURE WORK 63

REFERENCES 66

CHAPTER 1 - INTRODUCTION

A large number of resource constrained sensors deployed over a geographical region form a network, called wireless sensor network. A sensor node is designed to have limited resources and have preset functions. A sensor node is usually static but can be mobile as well. There can be one or more base station, which can either be static or mobile. Sensor node senses the environment and whenever a desirable event occurs they notify the base station using multihop wireless links. The WSN received command and queries using the base station which acts as a gateway between WSN and external Network and WSN keep the base station updated from time to time. Sensor Nodes are small tiny resource constrained devices which have very limited processing power, storage and battery power.

Why need ad hoc networks, the answer lies in the facts that they provide ease, speed of deployment and are infrastructure free. Adhoc Network provides flexible, self configuring and robust mechanism against disasters. They are particularly useful in minimizing wiring difficulties in historic buildings, wide natural and geographical areas, at conferences, trade shows etc.

A Wireless sensor network consists of a large number of tiny devices, called sensor nodes which are deployed in the hostile environment, across a geographical region. Each sensor has at least three capabilities. A sensor node using the sensing attribute can sense the environment changes, compute and has wireless communication capability to report the data to the base station. WSN has different application and some examples of wireless sensor networks are described below:

- Military uses sensor networks to keep the information about enemy movements, the existence of explosive materials, and other sensitive events.

- Different kinds of Chemical, Biological, Radiological, Nuclear, Explosive attacks and presence of radioactive materials can be detected with the help of Sensor networks deployed over a hostile region.

- Sensor networks are deployed in the forest to detect and monitor environmental changes such as fires, rain and other natural phenomena etc.
- Wireless surveillance under ocean water to monitor the movement of different fish or monitor temperature changes.
- Sensor nodes are being currently deployed in most of the public places like hospital, shopping malls, airports for the purpose of security.
- Automated parking spaces or slots are monitored with the help of smart sensor nodes.
- Monitoring seismic activity in areas with higher chance of an earthquake. Sensor nodes can detect this activity and report this to base station.
- Observing ground movements of Vehicles and Rivers and Canals and sensor node can report the occurrence of flood with increase of water levels.
- Buildings and high rises
- Monitor traffic for congested highway and roads in the city
- Environmental monitoring in buildings
- The natural phenomenon like Temperature changes, Lighting, Humidity can be monitored by deploying a WSN.
- Even sensors are deployed on other planets such as Sensors on Mars for research purposes.

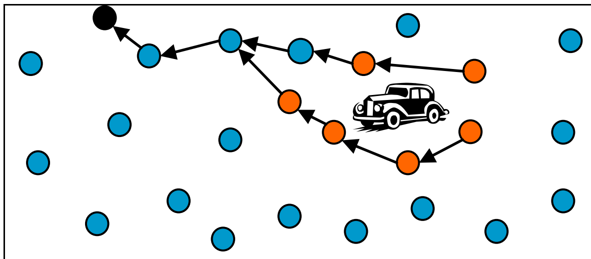
Sensor networks are special class of ad hoc network in which each device senses temperature, gases, pressure, movement, or water quality etc. This sensed data is collated, analyzed at a central location and a suitable response initiated within seconds.

Wireless Sensor Networks is also known as ubiquitous computing. These are tiny devices embedded in the physical world. These networks combine sensing, computation and communication [3]. Monitor the environment for critical events and their characteristics of sensor networks include

- Distributed sensing with multiple sensors
- Wireless communication
- Distributed information processing

- Ability to detect different kind of events happening in environments
- The Latency when an event is detected and the time it takes to report the event to the base station
- Accuracy
- Will depend strongly on errors and noise

Example of this Vehicle Tracking

	Low on power and resources Non replaceable battery Low memory Ad hoc deployment How do we configure and maintain such systems? Very large scale (1000s) High probability of individual failures

In above example, wireless sensor nodes are deployed in the hard to access area and a vehicle such as jeep movement is monitored. The application deployed at base station provided the results which concluded that sensor nodes can be effectively used for vehicle tracking.

1.1. Issues and Challenges in Wireless Sensor networks

Following are the common design issues [1] in wireless sensor networks.

- ***Fault Tolerance* –** When sensor nodes are deployed in the hostile environment wear and tear of sensor node take place. Few nodes get destroyed, some incur battery power failure. Few of them incur physical damage and some receive environmental interference. The overall task and working of the sensor network is not affected due to failure of some of the sensor nodes.

- ***Scalability* –** Hundred or thousand number of sensor nodes are deployed in a geographical region for studying different phenomena. With each day the number of applications for sensor network has been increasing. New proposed schemes must be able to work with large number of sensor nodes.

- ***Power Consumption* --** The wireless sensor node is a tiny device, comparable with the size of a coin, can only be equipped with a small battery power source. The limited power is consumed in both Useful or wasteful ways. There is a requirement for having energy efficient scheme which uses the limited battery power of a sensor efficiently and prolongs the network life time. Existing protocols have energy consumption used in many unneeded task such as transmitting or receiving data, processing query requests and forwarding queries etc. A security scheme can be evaluated based on its energy consumption.

- ***Stateless Architecture* –** There exist many physical limitations of wireless sensor network. WSN have high failure rates, and sensor nodes have very limited memory. Long routing tables cannot be stored in the memory and their architecture favors for a stateless approach in which routers do not maintain much information about network topology and flow state. In a traditional network or in wired network, a node can have thousands of routing entries that would be needed in state-based approaches.

- ***QoS Routing and Congestion Management*** – The routing protocol for wired networks have the QOS routing functionality available and protocol have mechanism to route the packet from congested areas to other available links for packet delivery. To design a routing scheme for wireless sensor network, the QoS and congestion management should be kept in mind. To design an efficient QoS based routing scheme for resource limited sensor network is a challenging task.

- ***Service Differentiation --*** In sensor networks, different types of packets flow between different sensor nodes, in which each type having its own importance. The packets carrying sensing event information have higher importance than to the packets having topology updates. The scheme should be designed so that the network should make more effort in delivering tasks with higher importance.

Topology Discovery and Dynamic Reconfiguration – Wireless sensor nodes are deployed in a random manner over a geographical region. The efficient topology scheme will have the following characteristics. The sensor nodes will be performing the sensing, computing and communicating functions. The scheme should select a set of distinguished nodes which are selected on certain criteria. These distinguished nodes are called clusterheads selected after communication has taken place with the local nodes, around them. The set of distinguished nodes organized themselves in the form of a cluster and they become cluster heads. These cluster heads will carry the task of all the common function from the nodes which are part of the cluster. Organized the network in the form of clusters, uses the limited resources of sensor network efficiently and reduce the communication overhead.

1.2. Definition of Clustering

After the sensor nodes are deployed in the geographical region, the nodes organized themselves in the form of clusters. Every node will be part of at least one cluster. A node will be part of more than one cluster if it acts as a forwarding node between two clusters. Every cluster has a leader, called the cluster head. Cluster head is selected based on the criteria proposed in the topology scheme used for clustering of the sensor network. All the nodes that are part of a cluster, communicate with the cluster head. For the notification of any sensing event, the node replies or deliver the information to the cluster head which it is associated. It is the responsibility of the cluster head to deliver the information to higher level cluster head which it is connected in the form a chain. The chain ultimately delivers the message to base station. The cluster head can directly send the message to the base station as displayed in Figure 1.1 but in this way more battery power is consumed. The main objective of a clustering algorithm is to form a connected topology and use the limited resources optimally.

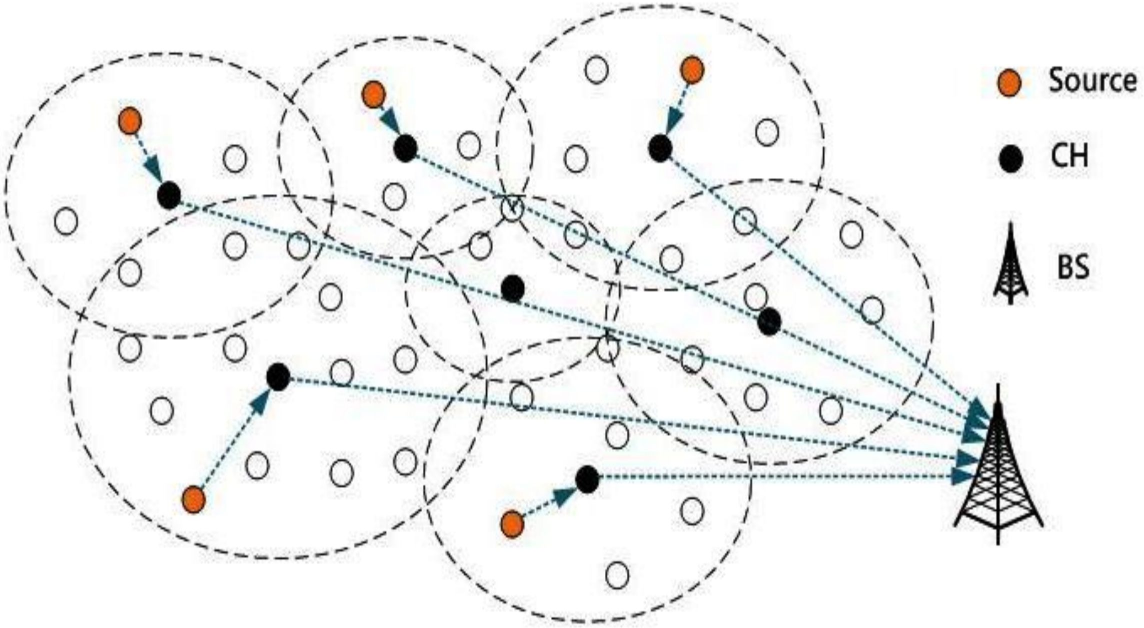


Figure 1.1 Wireless Sensor Network

1.3. The Clustering Process

In the first phase of clustering process a hostile environment or hard to access area is selected for the deployment of sensor network. The field is either selected by geological experts or select based on the selection of interest the application carries for the desired field. In case of military sensors, the field can be the enemy territory, where sensor nodes are dropped, using plane, and later enemy movement is tracked. The sensors are randomly deployed over the region.

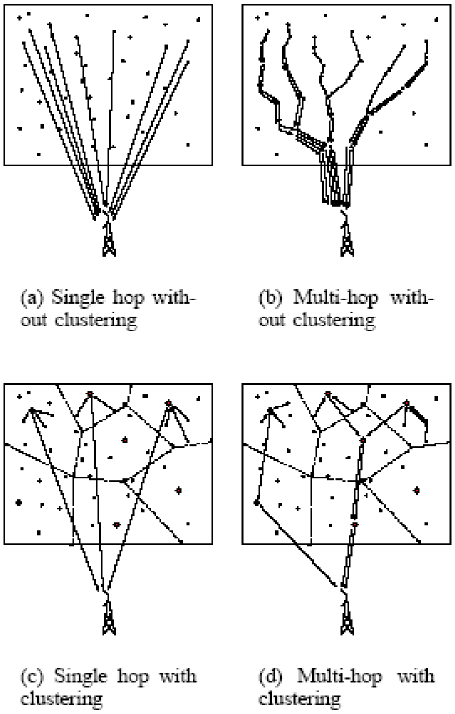
After the deployment phase, the clustering is performed. The based station sends a message over the region, that clustering should be performed and sensor nodes should organize themselves in the form of a clustered network. Cluster head are selected in the network based on the clustering algorithm. Cluster head are selected based on neighboring node information exchange and organized themselves in a hierarchical form to deliver message to base station.

After the clustering phase is complete, the sensor nodes part of cluster, reports to the respective cluster heads. The cluster head aggregate the information and report to higher level cluster in case of any sensing event. The clustering update message are exchange in case of mobile sensor nodes which leave one clusterhead and joins other upon entering in new cluster region. Clustering updates are communicated between clusterhead about node memberships.

The Cluster head is responsible for all the common functions performed by the sensors and responsible for the management of the cluster. After a certain time interval has been passed, the base station broadcast the message again to perform re-clustering.

1.4. Motivation for Using a Clustering Scheme

Figure 1.2 Sensor Communications


(a) Single hop without clustering

(b) Multi-hop without clustering

(c) Single hop with clustering

(d) Multi-hop with clustering

The basic reason of using a clustering scheme is the reduced communication and optimal use of the battery. As common functions are carried out by cluster head to provide communication to the base node. The Figure 1.2 demonstrates clearly that without clustering both in single hop and in multihop the communication overhead is large. The Figure 1.2 illustrates clearly that after doing clustering we can reduce the communication overhead for both single-hop and multi-hop networks. With clustering, the nodes are part of cluster and they transmit the information only to their cluster heads. The cluster head collects the information from the nodes part of the cluster, aggregates the received information and forwards it over to the base station.

In some clustering schemes the criteria are that, select nodes with higher residual energy to act as cluster heads are picked up or nominated among ordinary sensor nodes.

The lifetime of the Network is prolonged when there the numbers of nodes contending for channel access are reduced. By aggregating all the responses of nodes at the cluster head reduces and provides opportunity to summarize network state information. These aggregated response update increases network lifetime as well by saving limited energy.

The motivations behind the use of clustering schemes [5] in wireless Sensor Networks are, Reduce communication information, Favor spatial reuse, Minimize control information, Inter-cluster communication which allows the creation of backbone-based architectures ("virtual architectures," "private networks," etc.)

1.5. Challenges in Clustering

To design a clustering scheme which optimally uses the resources such as battery power and provides an efficient topology, for the wireless sensor network, different challenges are involved. The few parameters which we need to focus upon in the performance evaluation of the cluster based schemes for wireless sensor networks are the following

- Scalability
- Local self-healing
- Similar size clusters
- Optimum number of overlaps between clusters
- Minimum number of non-clustered nodes
- Optimum number of clusters
- Minimum communication overhead

CHAPTER 2 - BACKGROUND AND LITERATURE SURVEY

Many different clustering techniques have been researched and published for the domain of wireless sensor network. The following provides a review of some of the popular clustering schemes.

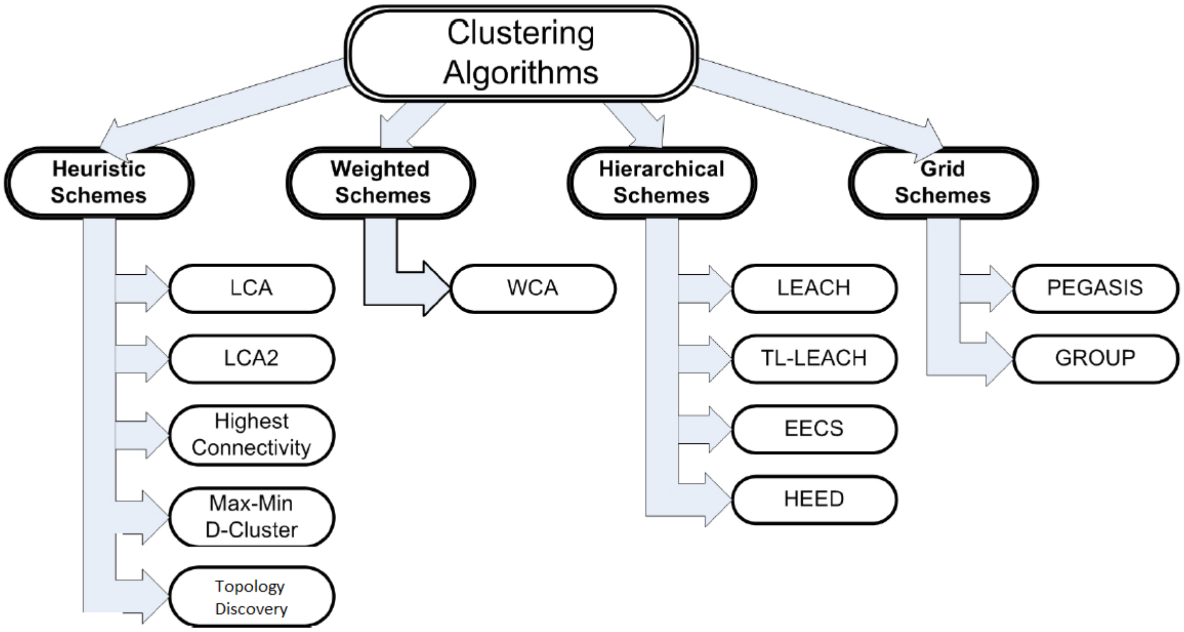


Figure 2.1 Clustering Scheme Hierarchy

2.1. Review of Various Clustering Techniques

Each of the clustering schemes is briefly explained.

2.1.1. Linked Cluster Algorithm (LCA)

This algorithm was developed as one of the early work done on the clustering for wireless sensor network. Previously it was developed for wired sensor network than modified to work for wireless domain. In this algorithm a unique ID is assigned to every sensor node and the scheme proposes two criteria to pass to become clusterhead [9].

i) Among all the neighbor of the node, it must have highest ID number
ii) No node is cluster head among all the neighbors of the node.

The result of the algorithm yields that it was determining a large number of clusters.

2.1.2. Linked Cluster Algorithm 2 (LCA2)

This algorithm was developed as modified version of linked cluster algorithm and the main focus was to eliminate the unnecessary election of number of clusterheads [10]. A new concept of node covered and node non-covered was introduced. If a node is the immediate neighbor of clusterhead it is considered covered. Among the non covered neighbors, the node with the lowest ID will be the candidate for becoming the clusterhead [11].

2.1.3. Highest Connectivity Cluster Algorithm

Many similarities can be found between this algorithm and LCA. In the algorithm every node broadcast to its neighbors the total number of nodes it is connected [2]. For the selection of clusterhead rather than looking at the ID, the connectivity of a node is taken into consideration. The node with the highest connectivity is elected clusterhead. If two neighboring nodes have same connectivity number, then the node with the lowest ID will be given the preference to nominate itself for the candidate of clusterhead [10].

2.1.4. Max Min D-Cluster Algorithm

This approach implements d-hop clusters. Each node is at most d hops away from the cluster head. Previous heuristics restricted themselves to 1-hop clusters. The value d is a parameter of the heuristic. In this paper they have first proved the generating d-hop clusters, in Ad Hoc networks, is an NP-Hard problem. In this paper they have proposed an efficient and a stable heuristic, which runs at regular intervals or when network configuration changes. It uses d round flooding twice to elect Cluster heads [11].

This heuristic is particularly good for large networks as it creates lesser number clusters for large network. The value of d can be adjusted with respect to the network size, hence it is Scalable. Nodes asynchronously runs the heuristic, and Limits the number of messages sent between the nodes to O(d). It minimizes the size of the Data Structure to O (d). This heuristic performs well in dynamic network as the cluster heads in their respective clusters tend to get reelected, hence it is highly stable. After electing cluster heads it further normalizes the Clusters to get further refined. It forms a back bone of clusters using multiple gateways. Cluster formation responsibility is distributed equally among all nodes.

The weakness for this heuristic is that it does not take into account the connectivity of the nodes and does not perform well, if nodes with higher connectivity located close to one another. If the nodes are linearly organized in ascending or descending order this heuristic fails to provide a good solution. Two clusters have multiple gateways between them and cluster head has to decide upon which path to reach the other cluster.

2.1.5. Low-Energy Adaptive Clustering Hierarchy (LEACH)

Consider as the first major improvement among the clustering approaches for wireless sensor network. Selection of the clusterhead is based on the criteria of received signal strength. The clusterhead then communicated with each other similar to a router, to deliver the information to base station. A cluster is locally responsible for the data aggregation and data fusion. LEACH is a distributive algorithm in which selected nodes propagate their request for becoming a cluster head with probability p [18]. A non cluster head node receiving this message determines its cluster by selecting cluster head that it can reach with minimum energy.

$$T(n) = \begin{cases} \dfrac{P}{1 - P * (r \bmod \frac{1}{P})} & \text{if } n \in G \\ 0 & \text{otherwise} \end{cases}$$

P = Desired cluster head percentage r = Current Round

G = Set of nodes which have not been cluster heads in 1/P rounds

In order to balance the load, nodes are periodically given the role of cluster head. A random number T between 0 and 1 is chosen by each node and node becomes cluster head if this number is less than the threshold.

2.1.6. Energy Efficient Clustering Scheme (EECS)

An Energy Efficient Clustering Scheme (or EECS) is a clustering algorithm in which clusterhead candidates compete for the ability to elevate to clusterhead for a given round.

All the nodes participating in the scheme, which desire to organize themselves in cluster form topology broadcasting their residual energy to neighboring sensor nodes. A node having more or greater residual energy than all its neighbors becomes a clusterhead. Cluster formation is different than that of LEACH. LEACH forms clusters based on the minimum distance of nodes to their corresponding clusterhead. EECS extends this algorithm by dynamic sizing of clusters, based on calculated distance from the base station [21]. This criteria outcomes an algorithm that addresses the problem that clusters at a greater range from the base station, requires more energy for transmission than those that are closer. Ultimately, this improves the distribution of energy throughout the network, resulting in better resource usage and extended network lifetime.

2.1.7. Hybrid Energy Efficient Distributed Clustering (HEED)

HEEDs provide a hybrid distributed clustering approach, which focus mainly on energy-efficiency. Based on the residual energy, the cluster heads are randomly selected and node becomes part of cluster such that the communication cost is minimized. The algorithm of the protocol called the HEED protocol is independent of the network diameter and terminates in constant number of iterations. HEED assumes that nodes are location-unaware and the network is quasi-stationary. HEED will take care of the topology, no matter how the density of the nodes is in the field. In the simulated result presented in the paper shows the network lifetime has been prolonged. The cluster produced by HEED shows many appealing characteristics. HEED is also suitable for multi-hop networks if necessary conditions for connectivity are provided. HEED can be pruned to different environment by setting different parameters and selection of cluster head can be controlled by these parameters.

2.1.8. Topology Discovery Algorithm

The main objective of topology discovery algorithm is to create the topology of whole network from the perspective of single node. We take one node as the starting reference point and create the whole topology from its perspectives.

A node starts the algorithm by becoming or declaring itself as cluster head. Two color schemes have been proposed. Initially all nodes are colored white. The topology discovery algorithm grows in the form of a rooted tree where the initiating node which starts the algorithm acts as the root node. The node receiving the root node message becomes grey means that they cannot become cluster head. Then grey nodes propagate the discovery message. All the white nodes receiving the message will become cluster head after the inverse of the distance between the grey node and new white nodes. The greater the distance the least amount of time the node will wait to become cluster head. As soon as it becomes cluster head it propagates the cluster message and all node receiving the message that are white or in process of waiting to become cluster head becomes grey and associate themselves with this cluster head. A node which is in locality of two cluster head becomes the forwarding node between these clusters. The Figure 2.2 illustrates a network topology upon which topology discovery algorithm is applied.

A rooted tree can be seen in the Figure 2.2. The algorithm spread similar to breadth first search technique. Two variation of color algorithm has been presented in the paper. In three color scheme there is no concept of forwarding node whereas in four color scheme to minimize the number of cluster head, the concept of forwarding nodes has been introduced. The topology discovery algorithm is rerun of the network after certain time interval has passed.

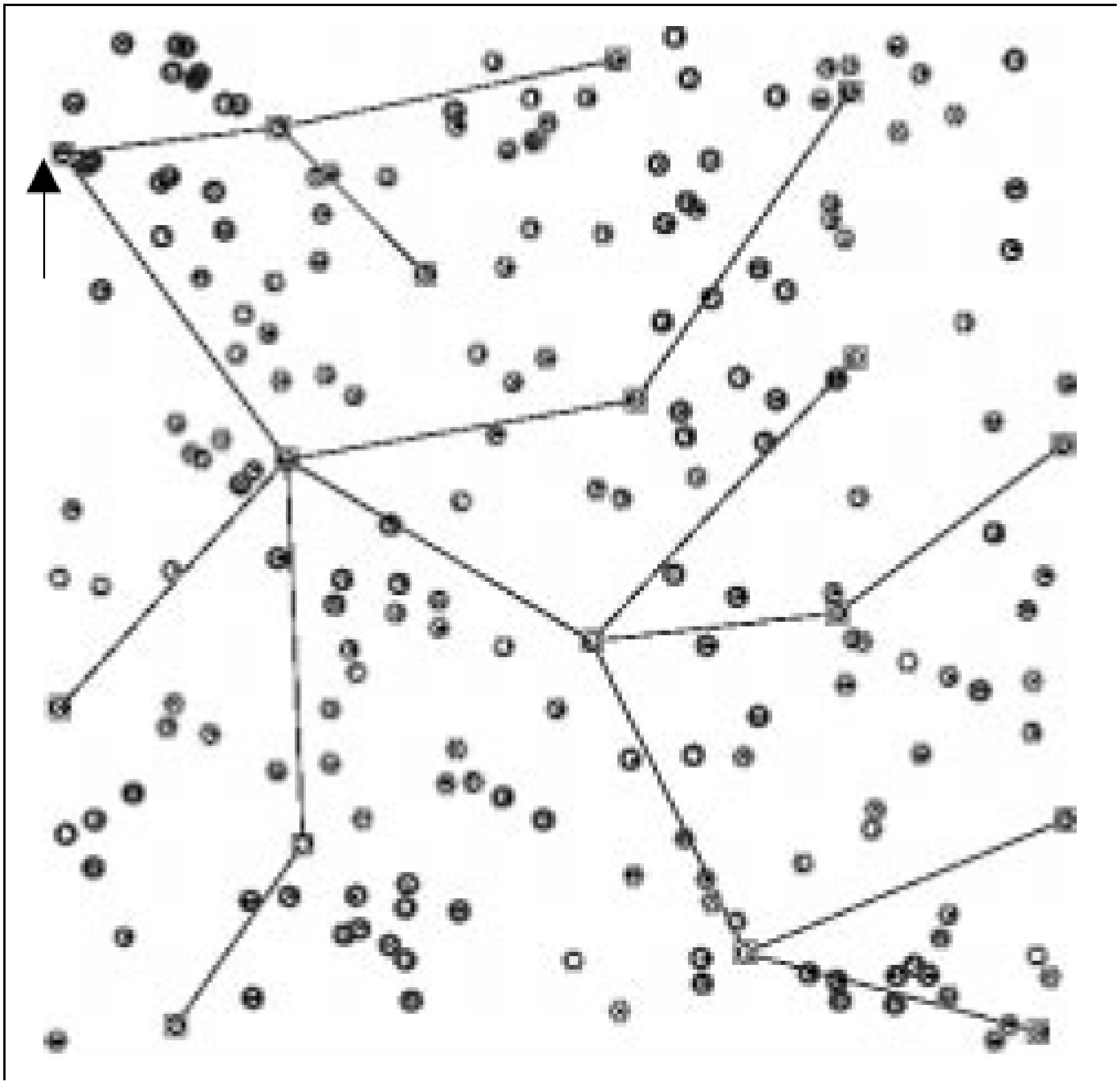


Figure 2.2 Topology Discovery Graph

The main weakness of this approach is that no logical criteria have been presented for the selection of cluster head. The algorithm depicts the picture of the network from single node perspective. Changing the initiating root node will evolve in different topology. Apart from the distance no specific criteria is provided for the selection of cluster head.

2.1.9. Weighted Clustering Algorithm (WCA)

A Clustering algorithm which selects a cluster head based upon a weighted function is computed with the help of information exchanges with neighboring sensor nodes. A cluster head is selected if it can support neighboring node functions. It has less mobility than neighboring nodes. Higher preference should be given to a node having more battery power and node locality is also given the importance. A weighted equation written below is calculated [6].

Calculate the combined weight W_v for each node

$$W_v = w_1\Delta_v + w_2D_v + w_3M_v + w_4P_v$$

for each node where $w_1 + w_2 + w_3 + w_4 = 1$

The Figure 2.3 demonstrates a network topology upon which weighted cluster algorithm has been applied. All the nodes which are selected as clusterheads have higher weighted function value than their neighboring nodes. Cluster heads communicate with each other to find a path toward the base station.

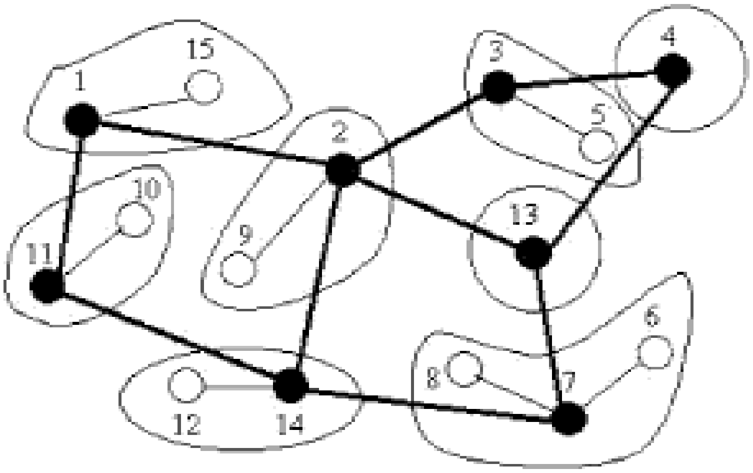


Figure 2.3 Weighted Clustering Graph

The weakness of WCA algorithm that it computes the clustering solution globally in the resource limited sensor network. The Algorithm assumes that sensor nodes have at least two power modes. Cluster nodes need a higher transmission range to have inter cluster communication. Cluster Heads Communication can be, between two nodes farthest located from each other. Calculating the solution globally consumes significant amount of energy and the cost of re-clustering is quite high in this algorithm.

2.2. Review of Security for Wireless Sensor Networks

To add security in WSN, a user will come across different challenges, which have been described below.

2.2.1. Security Challenges

To provide security in WSN is one of major concerns in today's world. But due to the characteristic of WSN, they are vulnerable to different type of attacks [14].

a. Everyone can listen to wireless channel. Anyone with wireless interface configured at the subscribed frequency can listen and monitor the traffic going through.
b. As for communication and collaboration, most of the public protocols are used, and attacker can penetrate using the loopholes in those protocols and exploit the network with many malicious attacks.
c. In WSN Network, sensor nodes have limited resources which provide restraint in supporting strong security algorithms. Symmetry key cryptography becomes the first choice for secure communication. Asymmetric key algorithm can be used as they utilize too many resources of the network which ultimately depletes the battery power of sensor nodes. As WSN comprises of thousands of nodes, there is a high demand for simple, efficient and scalable security solution and protocols which are designed by keeping in mind the resource limitation and constraint of WSNs [22].

2.2.2. Attacks

- Attack Techniques: - A normal working of WSN can be disrupted, using different attack techniques e.g. exploiting the security loop holes of publically known protocols or different attacks can be launched after ease dropping and analyzing the network traffic. Packets can be intercepted and modified during transmission.
- Node Compromise: - As sensor nodes are deployed in hostile environment and an attacker can capture the node and extract all the secrets which can lead the network to

most hazardous situation, where the node gets compromised. More severe attacks can be launched by attacker.

Passive Attacks VS Active Attacks: - In passive mode, attacker can eavesdrop the packets in promiscuous mode, without being detected. Attacker gathers the traffic or communication packets for later analysis. In Active attacks the attacker actively becomes part of network and exploits the network by many possible ways e.g. packets interception, modification, replaying packets onto the network, exploiting security holes in protocols, injection packets etc

- External Vs Internal Attacks: - Attacker launching attacks from outside the scope of network are considered as external attackers. While internal attacker gains legitimate access of WSN, either by node compromise or other way, can be more detrimental to the WSN.
- Jamming Attacks: - Sometimes attackers are interested in disrupting the normal flow of communication and can jam the communication channel. Permanent jamming or intermittent jamming can be introduced by attackers based upon their resource availability.

2.2.3. Security Requirements

Due to hostile environment and limited resources, wireless sensor network are pruned to more security threats than other networks. Certain security requirements are desired while working with sensor Network.

- Confidentiality: Only the intended sender and receiver understand the message contents. Critical parts of the packet are encrypted by sender node and decrypted at the receiver.
- Authentication: Both sender receiver are assured about the identities of each other.
- Integrity: Both sender and receiver should have guarantee that the communication taking place between them has not been modified. CRC and MAC are possible ways.
- Availability: Capability of Network, providing the necessary services, whenever they are required.

2.2.4. Key Management

Key Establishment is one of the most important steps in defining a security infrastructure. All the security requirements depend upon this step. To support the necessary security cryptography functions two types of keys are used.

2.2.4.1. Symmetric Key

Same secret key is shared between the sender and receiver for secure communication. The sender sends an encrypted message C by converting the plaintext M by using the Key K. The receiver receives the message C decrypts and converts in plaintext M using the decrypting key K.

Encryption Process Cipher Text C = Encrypt (M,K)
Decryption Process Plain Text M = Decryption (C,K)

2.2.4.2. Asymmetric Key

A key pair (K_s, K_p) is used to represent a key. Both sender and receiver have their own keys with them. K_s is called the private key which every node keeps it hidden or secret from the network. Every node shares its public key K_p with other nodes and it is known publically. A plain Text M can be converted into encrypted Text C using public key of receiver. Any node receiving the encrypted Text C can retrieve the plaintext M using its private key k_s.

Encryption Process Cipher Text C = Encrypt (M, K_p)
Decryption Process Plain Text M = Decryption (C , K_s)

2.2.4.3. Symmetric key Management

A very basic approach can be to distribute a global key among all the nodes to communicate which keeps the communication secure from the external attackers but this approach is prune to internal attackers.

Another approach is, considered as BS as a distribution centre (KDC). Every node shares a unique secret key with the KDC. Two nodes can communicate using a shared key, which can be unicast to both of them by BS in a secure manner but the approach incurs a lot of communication overhead and a consumes a lot of limited resources of WSN. Also in this approach, key server becomes a single point of failure and whole network is exposed if key server is compromised.

In Pre Distribution approach with is considered new a solution, nodes are preloaded with the key material for establishing keys among nodes, who wants to communicate. The other main thing is how to distribute the keys among nodes. Some design considerations are kept in mind while using this approach.

Memory Cost: Sensor nodes have very limited memory which can't hold large number of keys.

Node compromise Resilience: When a node gets compromised, an attacker gets to know all its secret keys, but the attacker must not be able to get to know keys of other nodes, so other node communication remains safe.

Local Secure Connectivity: Sensor node can store limited key material and it is much desirable that node establishes secure connectivity with neighboring nodes at one hop and rather avoids establishing indirect keys using multihop routing.

Key Agreement Models

Simply approach is preloading every node with N-1 keys. The overall number of key in WSN will be

N(N-1)/2

As WSN has thousands of sensor nodes this approach lacks scalability as the key number becomes exceptionally large.

Blom proposes a (t + 1) X N matrix approach but the memory cost of every node remains N-1 keys. In Blundo et al approach a t-degree bivariate systematic polynomial approach is used for key agreement.

Random Key Material Distribution

In this style of distribution, we preload every node with a subset of keys, called key ring. Select from a global pool, an idea that every node can atleast share a key with p probability.

Random Graph theory provides the theoretical foundation by considering WSN a graph G(n,p) of n sensor nodes with probability of link, exists between two nodes is p. p=0 mean no edge in network and p =1fully connected graph.

In Random Key Material Distribution Node compromise becomes a major concern because when the node gets compromised the key ring is exposed and some of the keys existing in the key ring are used by other nodes to communicate between each other . Non compromised nodes communication is exposed as well to the attacker.

Few techniques are proposed, which keep the node compromise impact to minimum. In Q composite a node has atleast q keys with certain probability. Approach fails if number of compromise node is large. In Spatial diversity approach, every node creates a derived key ring based on initial pre loaded key ring and by receiving the replies of nonce message among neighbors. So keeping the derived keyring the impact of compromise is limited to local area. Approach assume in initialize phase, the node can’t be compromised which is not generally true.

The possibility of same key by multiple nodes to communicate creates the problem of authentication and proving the identities of nodes. Random pair wise key approaches and challenge response approaches can be handy to prove the identity of suspected nodes.

Deterministic Key Material Distribution

Strongly regular graph or complete graph

Location Based Key Material Distribution

To establish indirect keys between two nodes over multihop is highly inefficient and consumes a lot of resources. The use of location information in key establishment improves the local secured connectivity. In Location based key pre distribution (LBKP) scheme we divide the network in square cells and assign each cell a unique t degree bivariate polynomial. Preload sensor nodes with its home cell polynomial and four neighboring cell polynomials.

Other Schemes (w.r.t Assumptions)

The Mentioned schemes above are developed by laying down some strong assumptions. E.g. all nodes have equal capability and any node can be compromised at any time. In Localized Encryption and Authentication protocol (LEAP) nodes can’t be compromised in initialization phase. Based on this assumption a global key is distributed for secure distribution of any shared key between two nodes.

In some schemes the assumption that every node can communicate with every other node securely is highly unnecessary. In some casess network is organized in the form of Tree where nodes send messages to higher level nodes which keeps on till message is reached to the root node or BS.

In weaker attack model making assumption that attacker is not listening to communication all the time and key can be shared in plaintext format between nodes. Second attacker can listen to one channel at a time and keys can be distributed using a randomly selected channel and sent in plain text format.

In heterogeneous network of nodes, powerful nodes are prune to attacks and are tamper resistant, are unnecessary.

2.2.4.4. Asymmetric Key Management

By keeping secret private key and publishing public key, these are easier to manage, more resilient to node compromise but are computationally expensive.

Computational Efficiency

In Asymmetric key technology computation efficiency is considered a highly concerned matter. Till today it is considered as one of critical issues how to implement asymmetric key algorithm efficiently on sensor platform. High level languages, compilers does not generate optimized code for specialized hardwire platforms. Research is still going on how to reduce the computation time.

Authenticate Public Keys

How to authenticate the public keys, still remain the most important concerns for applying asymmetric key technology. We need to verify that public key belongs to the node claiming to be. If this step is not done then any node can impersonate by claiming any node identity and man in the middle attack can be launched.

For example, two nodes A and B want to communicate with each other. A node C can impersonate to node A claiming to be B and to B that it is A, and it is possible if public keys are not verified. Then any communication taking place between A and B will be visible to C.

Merkley Tree, identity based cryptography and location based keys by binding private keys are few significant approaches suggested.

Group Key Management

Two types of broadcast/multicast communication takes place in WSN. One is Network Broadcast/multicast which is usually performed by BS as it incurs a large communication cost. The second is the local broadcast where sensor node collaborates with neighboring node to perform specified functions.

LEAP identifies two types of keys further. One the group key, which handles the network broadcast and cluster keys for local broadcast. The problem with LEAP approach is that in case of Node compromise attacker gets to know the group key. To improve this approach node must obtain some extra secret information from its neighbors, combining with its own secrets preloaded at the time of deployment to calculate the group key. Other node will not collaborate if the node turns out to be malicious node.

2.2.5. Authentication and Integrity

Authentication and integrity can be achieved by attaching the MAC with each of the transmitted packet. In Symmetric key technology a MAC C can be generated by taking the Hash with the help of collision resistant hash function of the data containing the message M along with the shared key K.

MAC C = HASH (M ||K)

Packet is transmitted along with message and MAC C. Receiver recalculate the MAC C' and checks C = C'. If matched the sender is authenticated and this guarantees message has not been tampered with, else message is rejected.

In Asymmetric key, the sender can encrypt a message with its private key K_s.

MAC C = Encrypt (M, K_s)

The operation is called digitally signing and result of operation is called signature. Packet is transmitted along with Message M and MAC C. Receiver decrypts the MAC C with sender public key and retrieves the message M' and checks M = M'. If matched the sender is authenticated and this guarantees that message has not been tampered with, else message is rejected.

In Hostile environment and in wireless medium authentication is very important as unauthorized attackers can tamper the ongoing transmission of packets.

Authentication can be performed as onehop unicast, multihop unicast and broadcast. In one hop unicast performing authentication at link layer is most efficient and performing on higher layers put extra overhead due to fragmentation. While performing authentication on multihop unicast link layer, authentication can be good choice as intermediate nodes are not trustworthy and performed at higher level usually at transport layer.

2.2.5.1. One Hop Authentication

One hop authentication can be supported by a shared link layer key between two communication nodes using any of the schemes, whether symmetric or asymmetric key management schemes. TinySec is first fully implemented infrastructure for link layer security for WSN [15].

2.2.5.2. Multi Hop Authentication

Multihop link layer security does not prove to be much feasible if intermediate nodes are not trustworthy or are compromised. Multipath enhancement and combining secret sharing techniques can be addressed here.

If Asymmetric key infrastructure is available, we can establish a multihop key between end node more securely and public key certificates can also support multihop authentication but consumes more resources in comparison to share multihop key approach.

2.2.5.3. Broadcast Authentication

Most of the schemes currently existing are based on symmetric key, where same key is used among a group of nodes. A node encrypting the message with group shared key and locally broadcast the information. All nodes receiving the message can decrypt the message using the shared key. Constant updation of the group key when nodes join or leave a group is required to protect pervious and future conversations. Group shared key can be managed by centralized as well as distributed way. Many approaches have been proposed.

2.2.6. Secure Routing

Routing lays an infrastructure for delivering data from source node to destination node. Routing also deals with finding a path from source to destination and choosing optimal paths, finding new path in case of link failures and make their best effort for data delivery between nodes. If routing protocols or information becomes disrupted due to malicious attack the high level application will fail and whole Network becomes useless. For smooth running of network topology secured routing is very important.

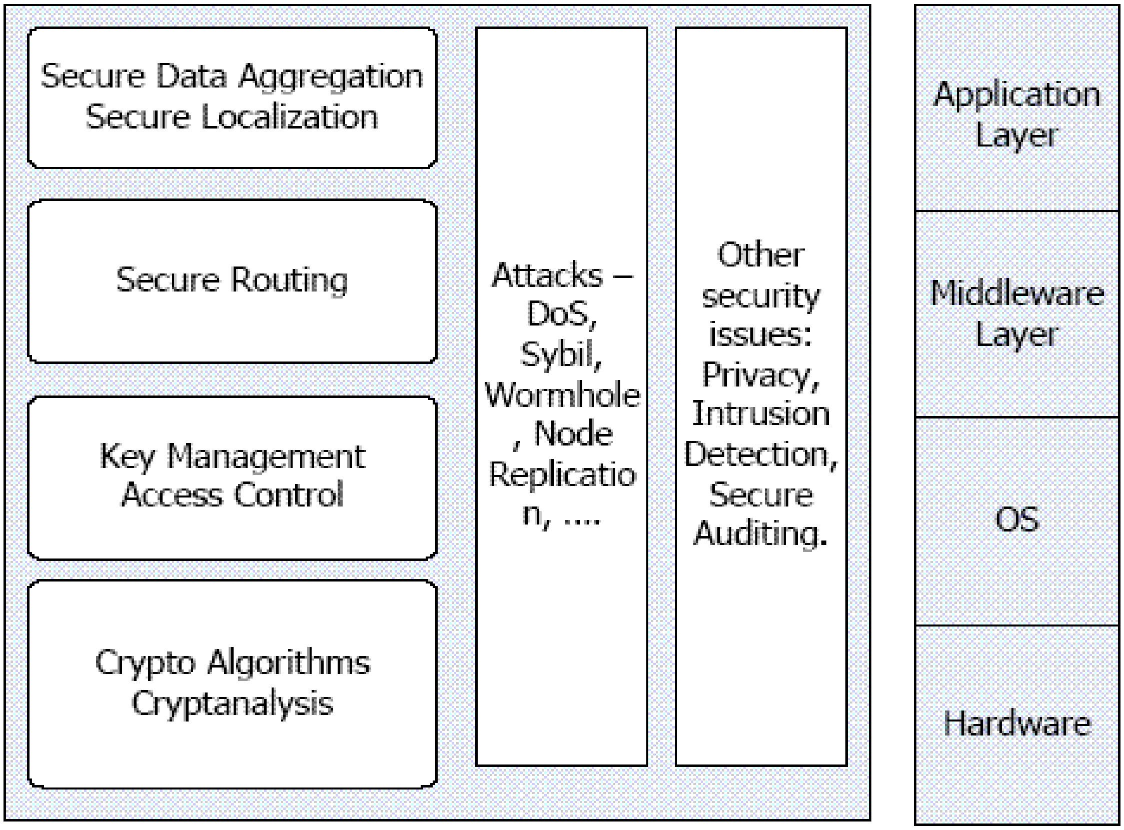
2.2.6.1. Problems

- Attackers can eavesdrop all the packets containing unencrypted routing information and discover the network topology.
- Attack can inject false routing information to launch many attacks and even disrupt the network topology. Traffic pattern of network can be changed, by which malicious nodes can receive most of the traffic before arrival to the BS. As Data is delivered to BS using aggregated approach and in case of malicious node receiving whole network traffic can lead to false report to BS which can lead to severe consequences [26].
- Several proposals to secure Adhoc routing protocols hardly can be applied to WSN for three reasons.
    - WSN differs from Adhoc network resource and communication, pattern wise.
    - Existing extension such as dynamic source routing (DSR), Adhoc on demand distance vector (AODV) or destination sequenced distance vector (DSDV) are not suitable for WSNs.
    - Due to limited processing power, currently routing protocols cannot be supported using complicated symmetric key operations or asymmetric key operations.

2.2.6.2. Available Solutions

- A global key for link layer authentication and encryption can protect WSNs against external attackers.
- After adding security to link layer, think of adding security to every layer of wireless sensor network also called the holistic view of security [28].
- Energy efficient symmetric key encryption algorithm should be used for wireless sensor network [29].
- Public key cryptography should be customized to work for WSNs.
- Use of Light weight security protocols for authentication, key distribution & key management.

A Security Map



2.2.7. Intrusion Detection and Counter Measures

As there is no absolute security, attacker will find ways to penetrate into secured WSN and launch many attacks. They will remain hidden in case of passive attack silently monitoring traffic while in case of active attacks some intrusion detection measures can detect those attacks; anomalies in the network and some counter measure can be taken.

2.2.8. Node Compromise

At the start no secret information is available to the attacker. An attacker can only eavesdrop the traffic and analysis of traffic might not help him much as most of the communication is encrypted. Due to deployment in hostile network there is possibility of network, coming in possession of the attacker and attacker can compromise the node and get the access to secret information. From there an external attacker can become an internal attacker and can launch many attacks. In the presence of compromise network remains secure.

In Location based key distribution schemes, nodes close to each other have more correlation in key material. Attacker cannot deduct the keys distant from the nodes, so node compromise impact remains local. Public key technique further limits the impact.

2.2.9. Active Attacks

Selective Forwarding: A malicious node drops certain packet and forwards other intentionally. This attack can be observed & further action can be taken if failure detection framework is introduced.

The Sybil Attack: A malicious or compromised node illegitimately spoofs or fakes itself to multiple identities. This attack poses serious threat to distributed storage and routing protocols. This attack can be detected using many ways. One method is radio resource testing in which a node is assigned a unique channel to each of its neighbors and test whether they can communicate using those channels. As radio is incapable of sending and receiving at same time present sign of Sybil attack. Another way is using ID based symmetric keys. Upon challenging the identity against ID, this attack can be detected. Registration and position verification are other possible ways to detect this attack.

The Node Replication Attack: Attacker places replicas of compromised node in many places of the network. This attack poses threat to Routing protocols specifically and used to change network topology. Attack can be detected using witness node which verifies location information or Asymmetric key techniques to authenticate location claims.

The Wormhole Attack: In this attack we make two distant nodes believing that they are neighbors. Attack tunnel packet through secret low latency broadband channel between two distant places and replay them. This poses serious threat to routing protocols. Attack can be detected by location and timing information in the packet. Directional antennas can defend against wormhole attack.

The Rushing Attack: Some routing protocol reply on broadcast ROUTE-REQUEST to find routes. Attacker forwarding requests quickly rather than forwarding only legitimate requests which make adversary path to be chosen. Embedding the node list can be used to detect rushing attack.

CHAPTER 3 - DESIGN & IMPLEMENTATION OF SELF HEALING CLUSTER BASED TOPOLOGY CREATION & MANAGEMENT SYSTEM

3.1. Maximal Weighted Topology Discovery Clustering Algorithm for Sensor Networks

Consider the network as an undirected graph G = (V, E) where V represents the vertices of the graph and E represents the links, in sensor network. Note that the cardinality of V remains the same but the cardinality of E always changes with the creation and deletion of links. Clustering can be considered as a graph partitioning problem with some constraints. The graph is clustered in such a way that the data disseminated from one part of the network to another is only through the cluster heads. Whereas all the other nodes which are the ordinary nodes are used to respond to cluster heads or act as forwarding nodes to deliver data between two cluster heads [8]. There exist many resource constraints on clustered networks and some of the clustering challenges like:

- Stability of clusters
- Optimum number of clusters
- Local self-healing
- Similar size clusters (Load Balancing)
- Optimum number of overlaps between clusters
- Minimum communication overhead

The point of emphasis is that we cannot base our clustering scheme solely on one parameter. Various parameters are considered to partition the network into clusters. In order to reach an optimal solution, we need to have a distributed topology discovery algorithm. Finding a global solution from the nodes that only have the local information is highly costly and energy consuming [3]. We want a solution that is locally computed and there is much less energy consumption.

3.1.1. Devising a Weighted Function

Node Energy or Battery power: A node which carries more energy is a better candidate for the cluster head job because a cluster head has more duties to perform than an ordinary node and does more information processing.

Mobility: Speed or mobility is an important factor to evaluate the stability of a cluster. Nodes which are relatively more mobile have a higher probability of going out of their clusters. For a mobile node the energy consumption is higher due to a higher number of reaffiliation messages.

Node Degree: To achieve load balancing the node degree for all cluster heads needs to be relatively equal.

Neighboring Node Position: If a node uses the basic energy level for communication its broadcast message will reach a particular range. In this range some nodes are relatively closer to this node and other are relatively farther. Selection of node located in centre of all nodes will provide clusters of greater size in term of distance covered and more regions will be covered and the number of cluster heads is reduced.

Data Rate: It is the rate at which a given target can be sent the sample data in a particular time. Pottie and Kaiser say that the cost of transmitting 1Kb a distance of 100 meters is approximately equal to the cost of executing three million CPU instructions [7]. But in many sensor applications the WSNs are heterogeneous with nodes having varying amounts of resources. Some applications require a high data rate from the clusters. In such a case a node having maximum energy but a low date rate is not a better candidate for a clusterhead as compared to a node which has a relatively low energy but a high data rate.

Target Revisit Rate: It is the rate at which a given sensory input is revisited by the sensor to perform sample measurement. In other words it is the response time of a sensor node. As mentioned earlier that timely response is also a required characteristic of a sensor network. A node exhibiting a high target revisit rate is a better candidate for becoming a clusterhead.

A weighted average function calculates and evaluates a value which is Weight of a Node Ω_{ϖ}. All the nodes calculate their weights by following equation:

$$\Omega_{\varpi} = \omega_{1E}+\omega_{2M} +\omega_{3\Delta}+\omega_{4\Pi\varpi}+\omega_{5\Delta\rho}+\omega_{6T} \qquad (1)$$

i.e. Ω(node) = Node Energy + Node Mobility + Node Degree + Neighboring Nodes positions + Data Rate + Target Revisit Rate.

The weight factor can be adjusted. The node with the maximum weight is considered as the best candidate for becoming a clusterhead. If the residual energy of a node v is less than a pre-set threshold, then that node cannot be declared a clusterhead. When the nodes communicate their weights they tend to lose energy. Similarly, on transmission and reception of Cluster discovery messages, the energy of the node is decremented. Energy computations are out of the scope of this research report.

3.1.2. Working of Algorithm

The algorithm works in two phases.

1) *Information Exchange*

   All nodes broadcast a hello message with their Mac ID and their position. After transferring this information all the nodes calculate their weights and again a broadcast is generated. Now all the nodes know the weights of their neighbors.

2) *Cluster Discovery*

   Now all those nodes that have their weights better than all their neighbors propagate their discovery request called "CLUSTERFINDER" and change their color to black means they have become the cluster heads. Ties are broken on the bases of node ID.

   All nodes which receive this message become gray. After this they will propagate this request to their neighbors after a certain delay that equals to inverse of their weights. Now those farther nodes that receive the message become Dark Gray (showing that they have not become the cluster head but they are possible candidates).

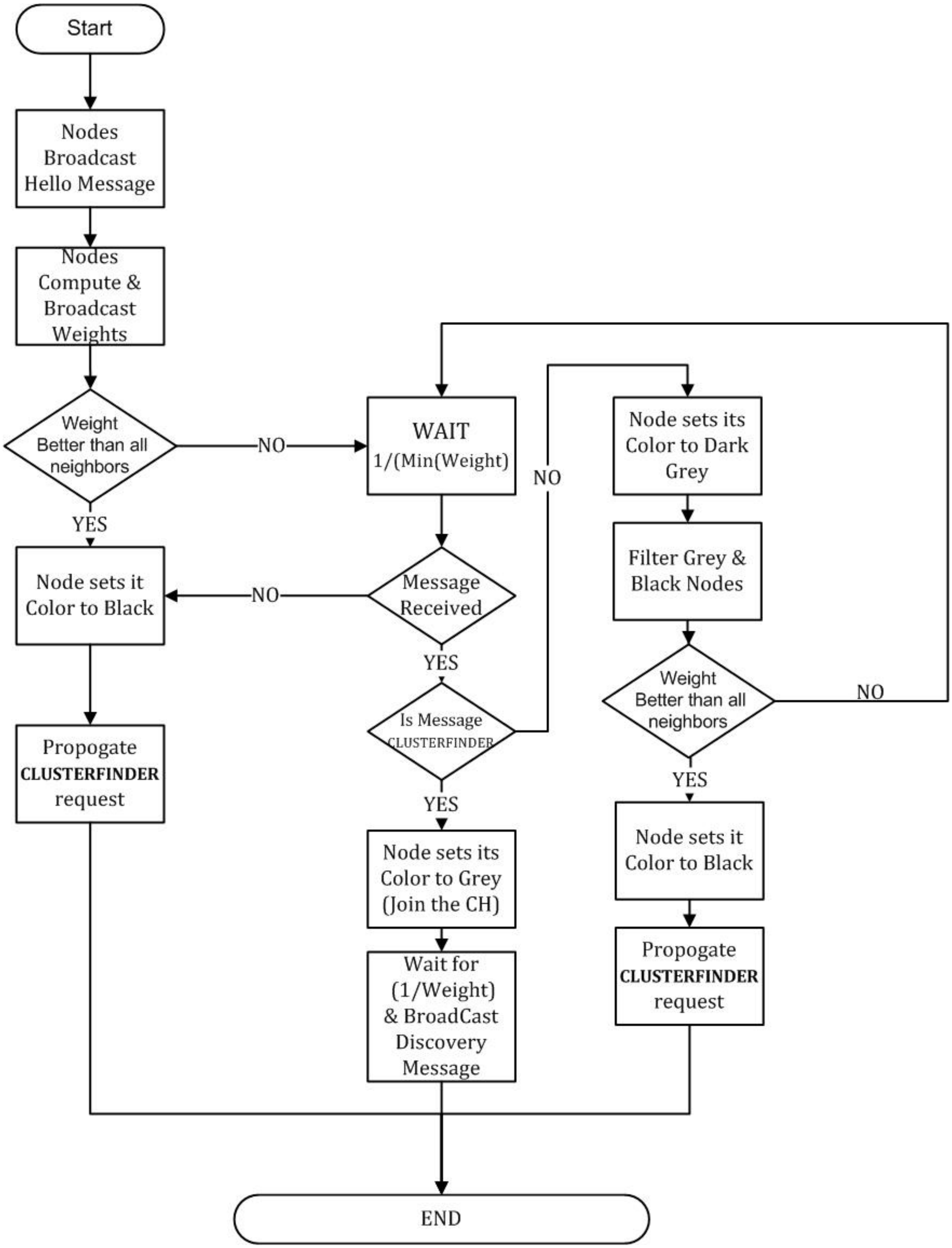
Weights of the nodes decide that which node should become the Clusterhead. The Dark Gray nodes compare their weights with their neighbors. The comparison does not include those nodes whose status has become gray or black. The Dark Gray Node with the best weight will become the cluster head and turns in to black.

The newly formed cluster head informs neighbors for becoming the cluster head. All Dark Gray and white nodes that receive this message turn gray.

All white nodes that have become gray will propagate the Discovery request and algorithm moves to the start of Cluster Discovery Phase. The algorithm does not finish until all the nodes become a part of at least one cluster.

All the nodes that receive the cluster head message information from two Cluster head nodes become forwarding nodes for information transfer between the cluster heads. A weighted centric distance strategy is adopted to select only one node as a forwarding node between two cluster heads. Nodes are generally used for routing between clusters. Only gateway nodes can listen to the different nodes of the overlapping clusters that they lie in. This overcomes the limitation of WCA [6]. The Clusterheads in our algorithm do not require using a higher transmission range to talk to other Clusterheads. This saves energy.

3.1.3. Flow Chart Diagram


Start
Nodes Broadcast Hello Message
Nodes Compute & Broadcast Weights
Weight Better than all neighbors
NO
YES
WAIT 1/(Min(Weight)
Node sets it Color to Black
NO
Message Received
YES
Is Message CLUSTERFINDER
NO
YES
Propogate CLUSTERFINDER request
Node sets its Color to Grey (Join the CH)
Wait for (1/Weight) & BroadCast Discovery Message
Node sets its Color to Dark Grey
Filter Grey & Black Nodes
Weight Better than all neighbors
NO
YES
Node sets it Color to Black
Propogate CLUSTERFINDER request
END

3.1.4. Color Assignment Function

The following pseudo code represents color assignment process. Let u be the node sending the cluster discovery request and v be the node receiving it.

```
AssignColors(u,v)
{
   if u.color equals black
   if v.color equals white
   {
    v.color=LightGray
   }
   if u.color equals LightGray
   if v.color equals white
   {
    v.color=DarkGray
   }
   if v.color equals LightGray
   {
    v.color=DarkGray
   }
   if u.color equals DarkGray
   if v.color equals white && Timer is
   expired && u.clusterhead is true
      {
       v.color=LightGray
      }
   if v.color equals DarkGray && Timer is
   expired
   if v. weight is best in neighbourhood
      {
       v.color=black
      }
  else
     {
         v.color=LightGray
     }
   }
```

3.1.5. Example

The systems demonstrates the new proposed algorithms with the help of an example in which the total nodes is eleven which are numbered through 0 to 10. The speed of the node varies between 0 and 10. The Maximum transmission range of all the nodes is fixed that is 30m. All the nodes are given a random energy at the start. Table 1 shows the weights of nodes and their degrees. The weights have been calculated with the help of equation 1.

TABLE 1: PROPERTIES OF SENSOR NODES

Node ID	Node Degree	Node Weight
0	4	11.423
1	1	13.661
2	4	11.388
3	3	12.228
4	3	12.074
5	3	12.071
6	1	13.424
7	4	11.382
8	4	11.347
9	4	11.417
10	3	12.199

At the start all the nodes are white as shown in Figure 3.1. This means the network is not partitioned into clusters. All the nodes have variable energy and velocity. At any given time the nodes exchange their information.

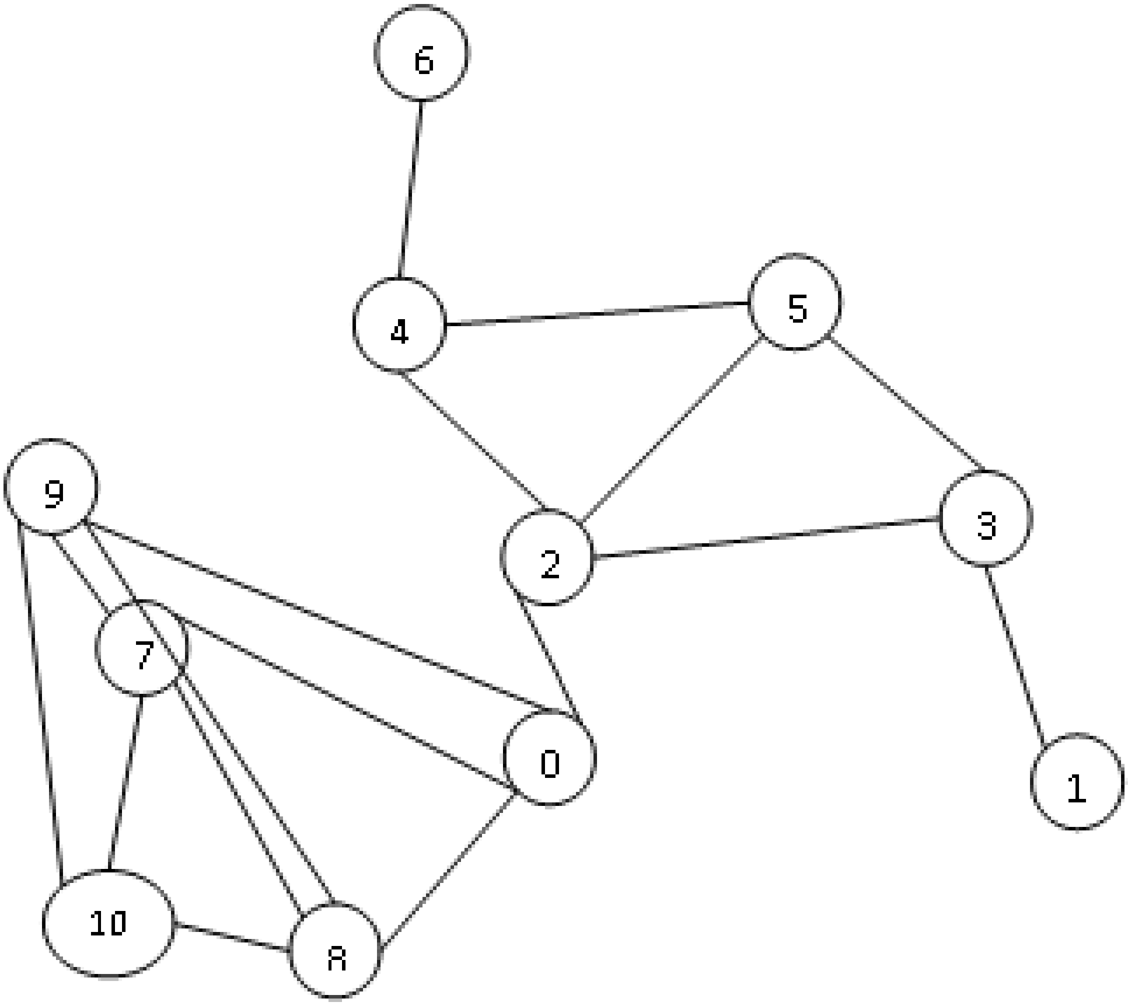


Figure 3.1: Nodes Randomly Placed

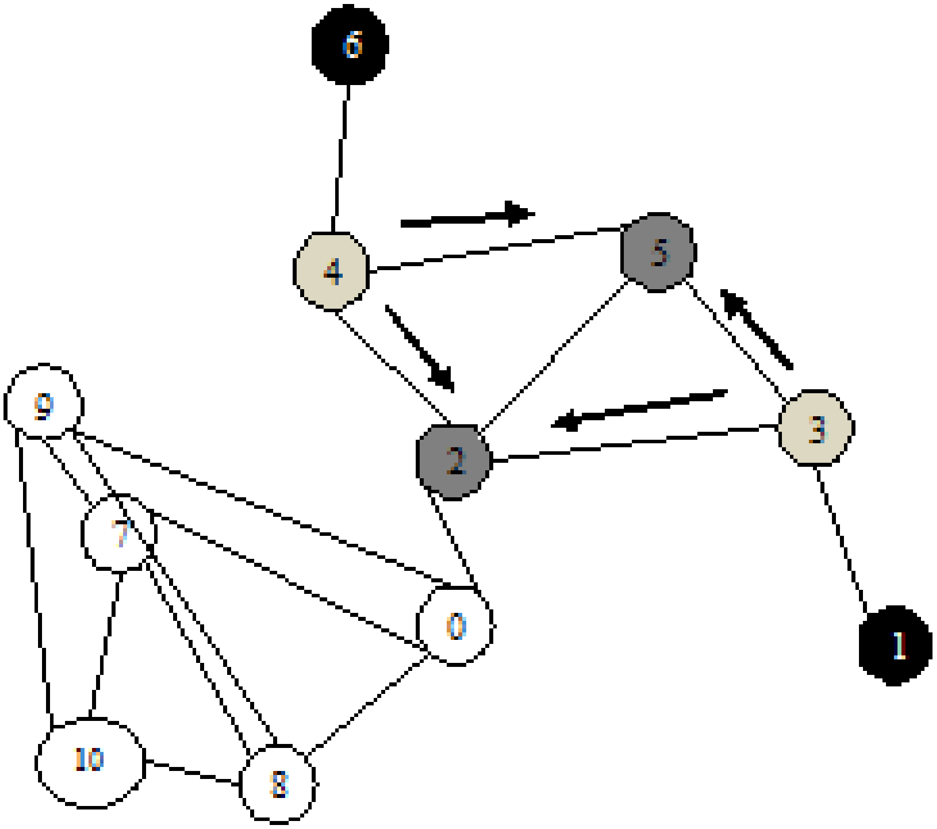


Figure 3.2: Cluster Discovery Starts

We suppose that node 1 and 6 start the algorithm. In Figure 3.2, node 1 and 6 declare themselves as clusterheads. Node 1 and Node 6 also declare themselves as clusterhead because their weights are better than neighboring nodes. Node 1 and node 6 send "CLUSTERFINDER" request to their neighbors which turn them from white to Light gray.

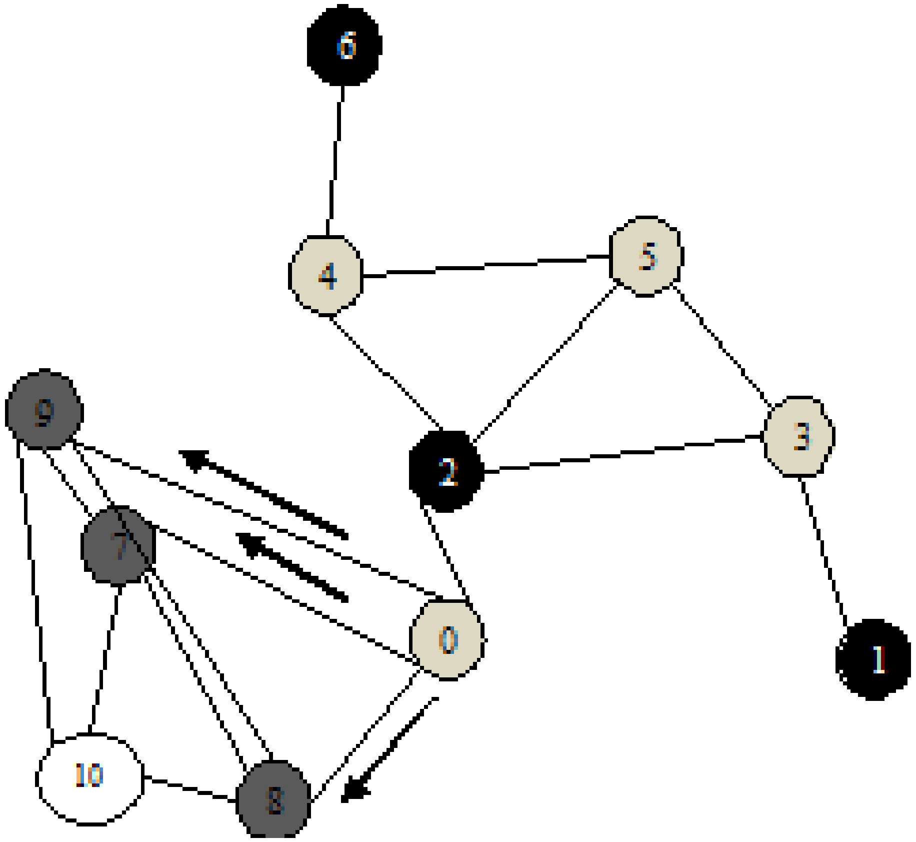


Figure 3.3: Cluster Discovery Continues

After a delay which is equal to the inverse of the weight of a Light gray node, the Light gray nodes send the "CLUSTERFINDER" request to their neighbors. The neighbors upon receiving the message turn Dark gray. The Dark gray nodes weight for a random time to see if they receive a node weight better than their own weight. Upon the timer expiration the nodes become the clusterhead if they are the best in their neighborhood or join an existing cluster. Figure 3.3 shows how Node 2 has become a clusterhead but node 5 has joined an existing cluster. The cluster discovery request further propagates in the network and Node 7 becomes the clusterhead in Figure 3.4. Figure 3.5 shows the cluster view of the given network. Node 0, 3 and 4 are border nodes between the clusters.

3.1.6. Maximal Weight Cluster Update Function

There are four different types of updates possible in a cluster structure. These updates include when a node comes alive, or when a node dies, when a node creates a link with other node or when the link is broken.

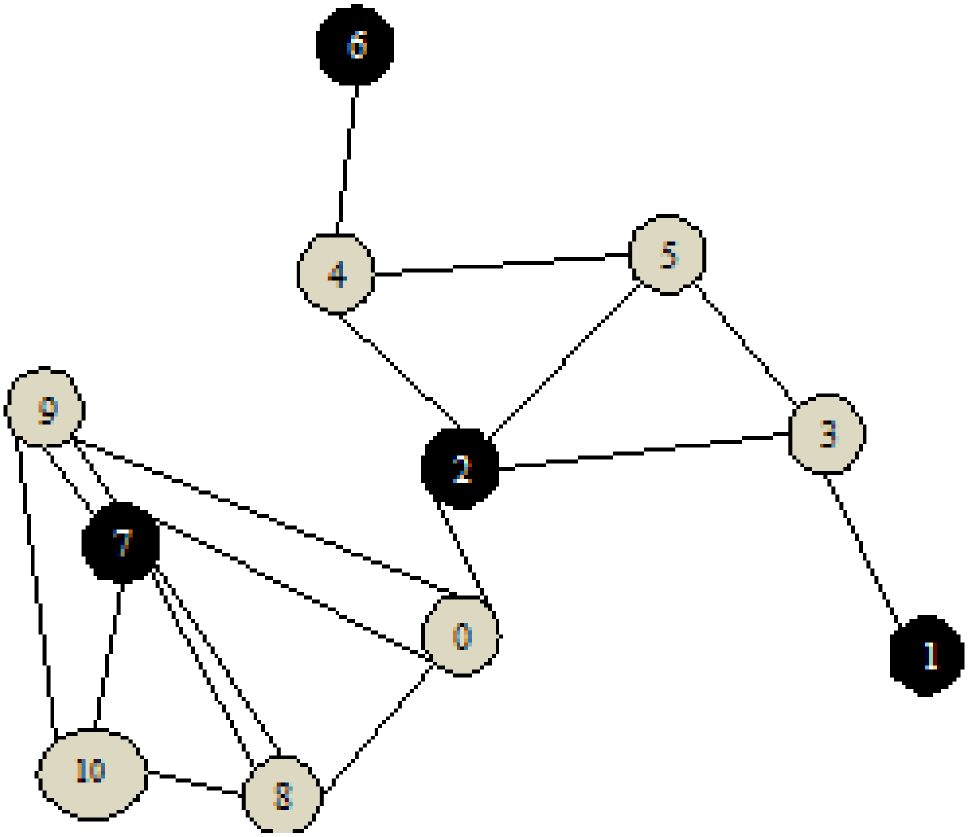


Figure 3.4: All nodes have become a part of some cluster

1. When a node v other than the displayed nodes in Figure 3.4 switches on, it checks if there is a Clusterhead in its neighborhood. If a clusterhead is found, then it joins that clusterhead. Otherwise, it determines the best weight in its neighborhood to become the clusterhead and sends the CLUSTERFINDER request to its neighbors.

2. When a node is dead, no change is made if it is an ordinary node. If it was a forwarding node or a Clusterhead, the nodes in that cluster decide a new Clusterhead which has the best weight in the neighborhood and which is not a neighbor to another Clusterhead. All nodes in the neighborhood will retrigger clustering process for local self healing.

3. When a link is created between two nodes there are three cases:

   (a) If two nodes are ordinary nodes then both nodes recalculate their weights and retrigger clustering algorithm.

   (b) If one of the nodes is a Clusterhead then the other node simply joins that Clusterhead.

   (c) If both nodes are Clusterheads then both the nodes will have to retrigger the clustering algorithm again and only one of them will be able to retain its position of Clusterhead.

4. When an existing link is broken there are again three cases to consider:

   (a) If one node is a Clusterhead and other an ordinary node then both will retrigger local clustering again.

   (b) If both the nodes are ordinary nodes and if they belong to separate clusters then no changes are to be made.

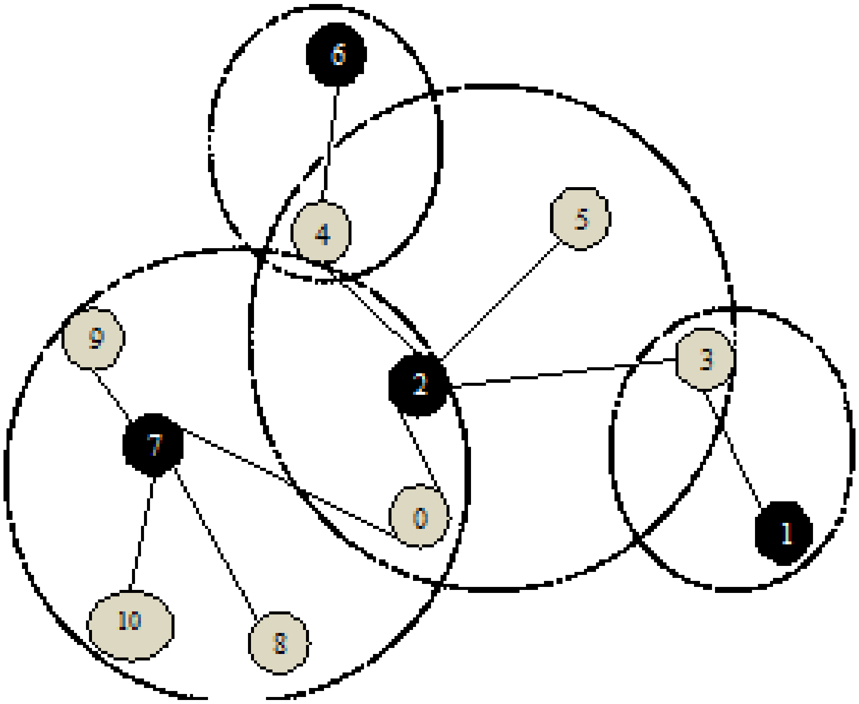


Figure 3.5: Cluster in the Network

Otherwise if both the nodes are ordinary nodes and if they belong to same Clusterhead then both the nodes broadcast their new weight to the Clusterhead and all the neighbors.

3.2. Bonded Degree Clustering Algorithm

The heuristic that we have devised is a completely distributed asynchronous in nature like k-DegreeID [12] and HEED [5], unlike Topology Discovery [14] or Weekly Connected Dominating Set [21] heuristics. In which the algorithm starts by one monitoring node (as in case of TopDisc [14]) or by few nodes that are the best in there own neighborhood (as in Weekly Connected Dominating Set [21]).

3.2.1. Basis of Algorithm

In this heuristic every node makes its own decision itself or on the call of its neighbors. So a node decides by looking at its own weight and the weight of its neighboring nodes to decide whether it will become a cluster head or an ordinary node. The algorithm is basically targeted at basically minimizing the number of cluster heads and to achieve load balancing. Load balancing is achieved, setting bounds on the degree of a node for becoming a cluster head. The bounds are the Lower Bound and Upper Bound which are an input to my algorithm. The lower bound is a strict one that is if there is node in the neighborhood greater than the lower then nodes whose degree is less than the lower bound will not become cluster head. The upper bound is not a strict bound if a node degree exceeds the upper bound its weight are made to deteriorate so as to discourage nodes from becoming cluster heads whose degree in greater than the upper bound. It is not a strict bound because a node might still become a cluster head even if its degree is greater than the upper bound. Upper bound will be associated to nodes weight which can we calculated using the Wv weight equation in which the Degree parameter is the modulus of difference between the node degree and the Upper bound. Parameter D is calculated as Calculated in WCA heuristic [16].

$$D = |\,Degree - Upper\ Bound\,|$$

So if the Degree is close to the Upper Bound assigns the best value to the Nodes weight.

3.2.2. Working of Algorithm

On start every node calculates its own weight on the bases of local information. For the sake of simplicity, let us assume that a node's weight is equal to its own degree. Every node after calculating its weight broadcasts its weight to all its neighbors. Thus every node gets the weights of all its neighbors. Now if a node's own weight is better than all its neighbors then node declares itself a cluster head. Ties are broken on the bases of node ID [30]. If the node is not itself the best node then it sends a "Cluster Head Message" to the node with best weight to become a cluster head, if that node has not already declared itsself a cluster head. After sending the "Cluster Head Message" to the best node the node weights for the best node to reply back. If that best node did not declare itself a cluster head and nor any other node in the nodes neighborhood declared itself a cluster head. Then the node compares its own degree with the "Lower Bound" (an input for the heuristic). If its own degree is greater then the "Lower Bound" then it declares itself to be a cluster head. Otherwise becomes an ordinary node.

When a node receives a "Cluster Head Message" it checks if any node in its neighborhood has declared itself a cluster head or not. If no node in its neighborhood declared itself a cluster head then it becomes a cluster head. Otherwise it becomes a Forwarding node for the node that sent it the "Cluster Head Message".

Now what is a Forwarding node? Forwarding node is a node that could not become a cluster head because it has a better node in its neighborhood which had already declared itself a cluster head. Unlike a cluster head it does not receive messages from all its neighbors, but only from nodes which elected it a cluster head. Since it did not became a cluster head so it became a Forwarding node.

NOTE if node is a neighbor of both a cluster head and forwarding node, then it will route the messages only to the cluster head not to the forwarding node. Only those nodes will route the messages to the forwarding nodes which are not directly connected to any cluster head. Other forwarding nodes only act a border or gateway nodes between two cluster heads.

Lower Bound input controls the number of forwarding nodes in the network. If you increase Lower Bound the number of Forwarding Nodes increases and vice versa. And if you bring down the Lower Bound to be equal to one then no Forwarding nodes are created. Because then every node will be directly connected to a Cluster Head. And there will be no node dependent on a Forwarding node for routing of messages. The Lower Bound when is equal to one means that a node with only one neighbor (a leaf node) can also become a cluster head.

3.2.3. Example

In the Figure 3.6 below the Lower Bound is set to two, node 6 becomes a forwarding node (navy blue) which was elected by node 4 as its cluster head. But since node 6 best neighbor node 3 has declared itself a cluster head so it will not become a cluster head and will become a forwarding node for node 4. In this case Lower Bound was set to two. In Figure 3.7 below the Lower Bound is set to one, the result is that no node becomes a forwarding node and node 4 becomes a cluster head. And node 6 becomes a border node.

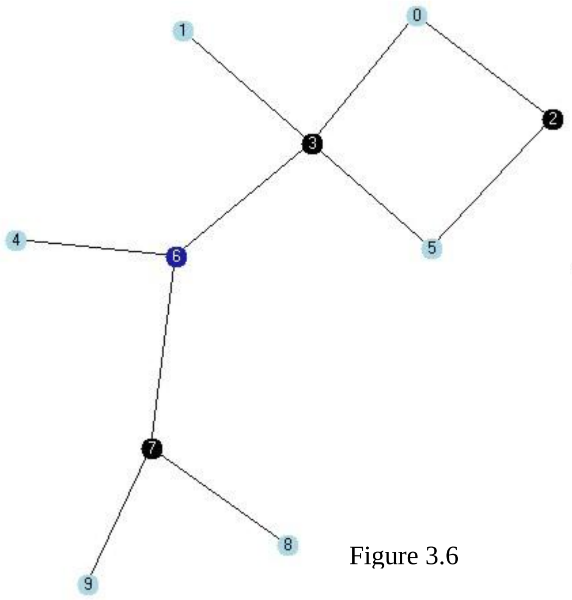


Figure 3.6

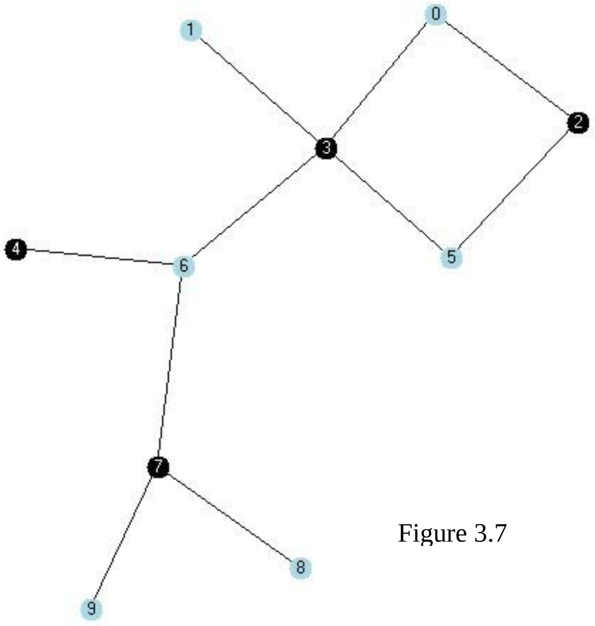


Figure 3.7

3.2.4. Bounded Degree Cluster Update Function

There are four different types of Updates possible in cluster structure. These events include when a node switches on: when a node switches off, when a link is created and when a link is broken.

When a node switches on, it checks to see if there is a cluster head in its neighborhood, and if so it joins the cluster head. Otherwise it checks to see, if it has the best weight in its neighborhood, if so it becomes a cluster head, or else asks the best neighbor to become cluster head. If the best neighbor does not become a cluster head then if nodes own degree is greater than the Lower Bound it declares itself a cluster head.

When a node switches off, no change is made if it is an ordinary node. If it was a forwarding node or a cluster head the nodes in that cluster decide a new cluster head which has the best weight in the neighborhood. And which is not a neighbor to another cluster head. Basically all nodes in neighborhood will retrigger clustering process for local self healing.

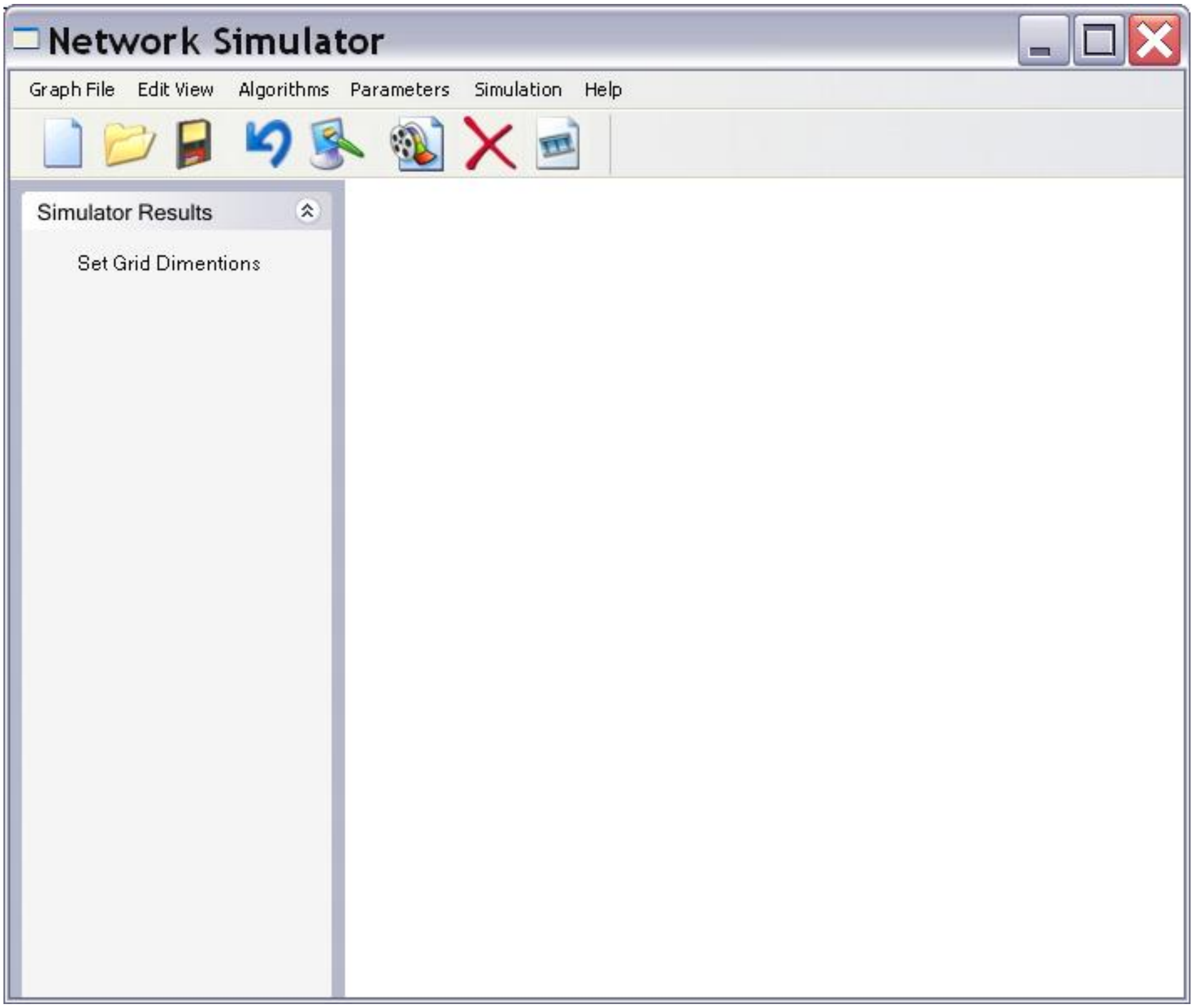
When a link is created between two nodes there are three cases here if two nodes are ordinary nodes then both nodes recalculate their weights and retrigger clustering algorithm. If one of the nodes is a cluster head then the other node simply joins that cluster head. And if both the nodes are cluster heads, both the nodes will have to retrigger the clustering algorithm again and only one of them will be able to retain its position of cluster head.

When an existing link is broken there are again three cases to consider if one node is a cluster head and other an ordinary node then both will retrigger local clustering again.

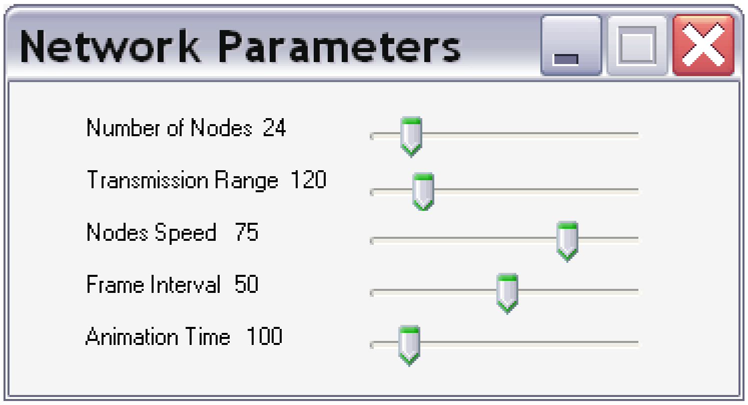
If both the nodes are ordinary nodes and if they belong to separate clusters then no changes are to be made. Otherwise if both the nodes are ordinary nodes and if they belong to same cluster head then the both the nodes broad cast their new weight to the cluster head and all the neighbors.

3.3. Graphical User interface of the System

A custom simulator is designed not only to test the performance of existing topologies but the new proposed algorithms have been incorporated to be tested in term of performance. The simulator provides the functionality of designing any custom or generic topology of wireless sensor network over a field of 600*600. The simulator can save the topology in the form of bitmaps and can retrieve any previously saved topology. A user can custom can draw can own topology of sensor nodes or the system can generate the random topology for the user of the system.



The simulator can run any algorithm on the topology and can deliver simulated results by providing the simulated input.



Number of Nodes 24

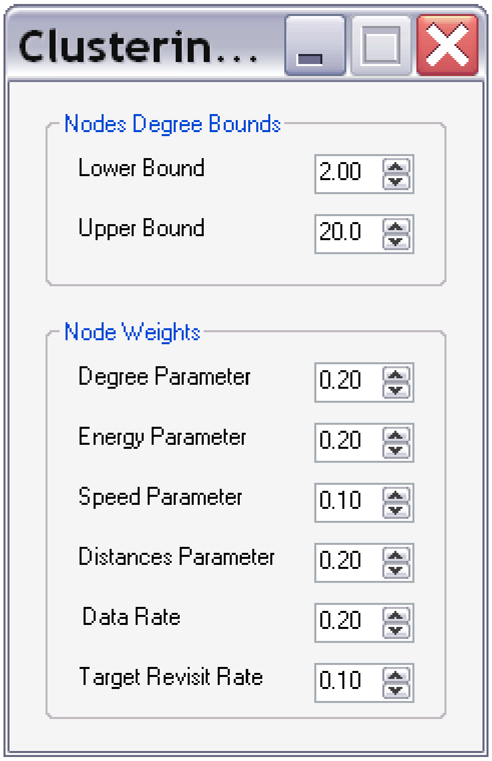
Transmission Range 120

Nodes Speed 75

Frame Interval 50

Animation Time 100

If the user would like to generate a random topology he can control the number of nodes the system should generate with the number of nodes field. A user can increase or decrease the transmission range of a sensor. The node speed can also be controlled. The frame interval control after how long the animation should update the screen. The Animation time can control the total animation time for a topology.



Nodes Degree Bounds

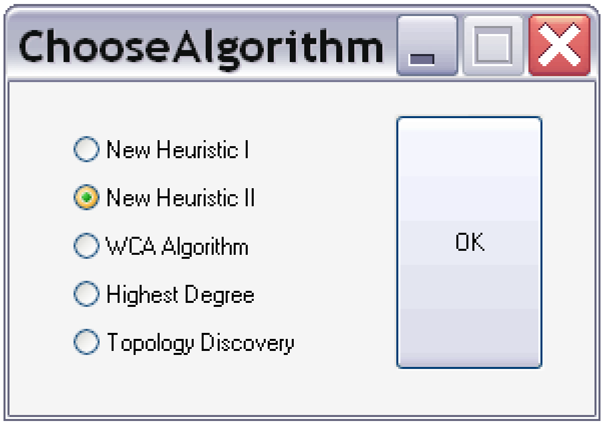
Lower Bound	2.00
Upper Bound	20.0

Node Weights

Degree Parameter	0.20
Energy Parameter	0.20
Speed Parameter	0.10
Distances Parameter	0.20
Data Rate	0.20
Target Revisit Rate	0.10

The clustering parameters can be controlled using the above displayed picture of the dialog. For WCA or Maximal topology Algorithm, a user is required to provide how much weight he wants to assign to each sensor node. The total weight should not go beyond 1.0. A user is required to provide lower and upper bound in a Node degree Algorithm

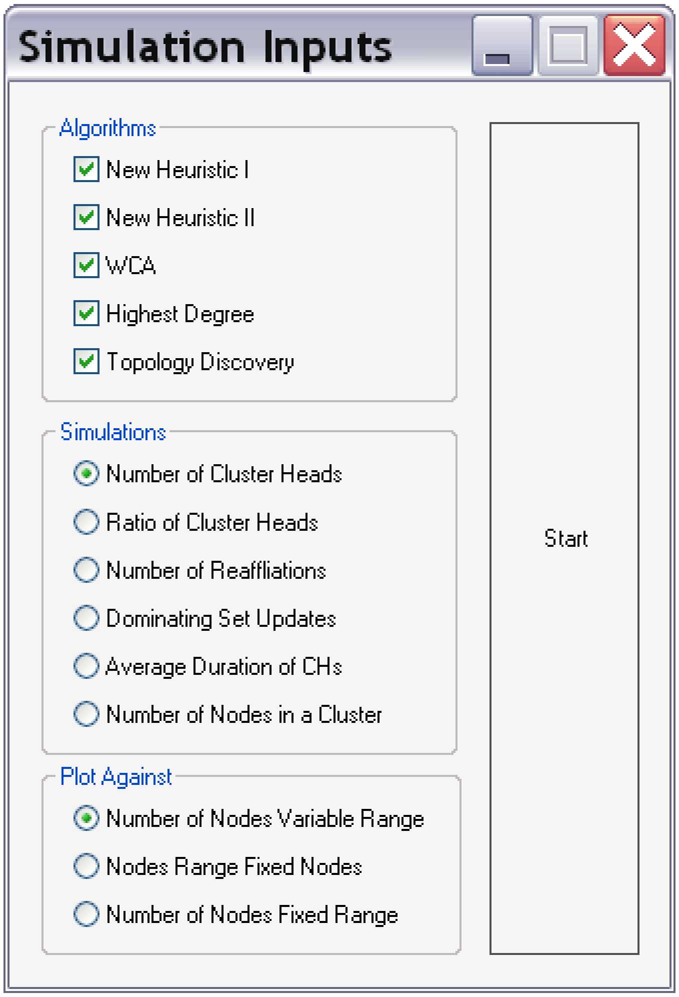
The Simulator provides the user option to select any algorithm from the available list of five implemented algorithms. The New Heuristic I is the Maximal weight topology algorithm and New Heuristic II is the Bound Degree Algorithm. The other three are shown in the picture.



ChooseAlgorithm

○ New Heuristic I
● New Heuristic II
○ WCA Algorithm
○ Highest Degree
○ Topology Discovery

A user is required to provide the following inputs to the system to generate the reports.



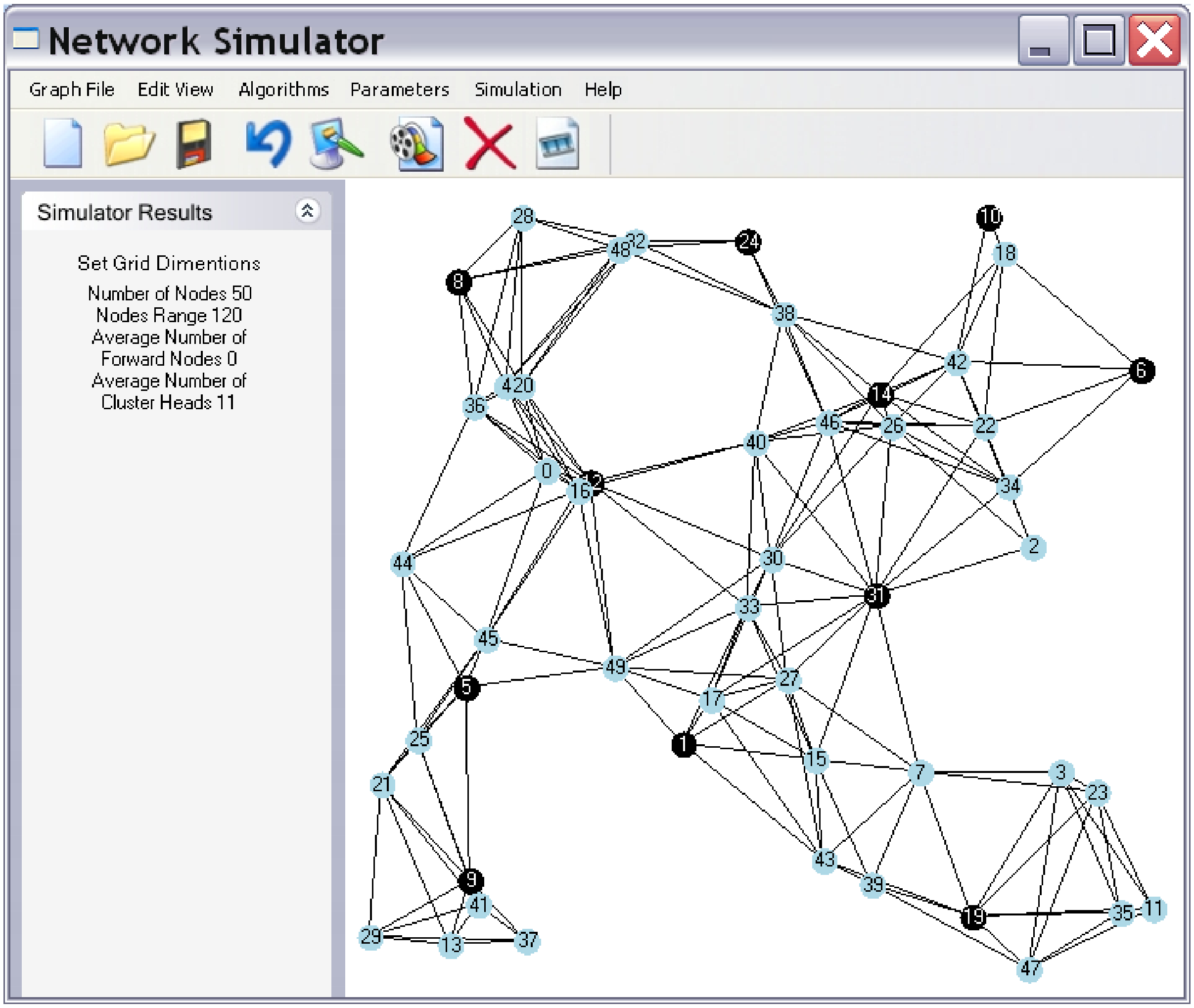
Simulation Inputs

Algorithms
☑ New Heuristic I
☑ New Heuristic II
☑ WCA
☑ Highest Degree
☑ Topology Discovery

Simulations
● Number of Cluster Heads
○ Ratio of Cluster Heads
○ Number of Reaffliations
○ Dominating Set Updates
○ Average Duration of CHs
○ Number of Nodes in a Cluster

Plot Against
● Number of Nodes Variable Range
○ Nodes Range Fixed Nodes
○ Number of Nodes Fixed Range

The following picture shows a topology having 50 nodes with the node range of 120. The picture shows there is a total number of 11 cluster heads. Black nodes on system screen represent cluster head and nodes with light blue color represent nodes which are associated with their neighboring cluster heads. The Dark blue nodes are the forwarding nodes. The algorithm select for the topology is Heuristic one which is also known as Maximal weight topology discovery Algorithm.



Simulator Results

Set Grid Dimentions

Number of Nodes 50
Nodes Range 120
Average Number of
Forward Nodes 0
Average Number of
Cluster Heads 11

CHAPTER 4 - SIMULATED RESULTS & DISCUSSION

3.4. Network Model

A custom simulator is designed, developed and used for comparison. The simulator presents a square grid of variable range from SIZE (100X100 to 600X600). A system of N nodes can be randomly deployed or can be placed on defined location. Node can be static or can move in all possible directions with random speed ranging from 0 to maximum defined per unit time. Transmission range of all the nodes can be defined before the simulation starts. Nodes are called neighbors if they exist within each other transmission range. To identify every node a unique ID is assigned to it. A node keeps records of the its location (x,y) and all the above mentioned necessary parameters. At the beginning all nodes have same Energy level. The simulator used to compare the performance of new proposed algorithm against the existing algorithms. Among the existing approaches are tree based clustering algorithm called Topology Discovery [4], Highest Degree and WCA a weighted clustering Algorithm [1].

3.5. Performance Metrics

We have evaluated the performance of new algorithms on the basis of following metrics:

- Average number of Clusterheads
- Number of Reaffiliations per unit time
- Network lifetime
- Stability of Clusters.
- Ratio of Cluster heads to total number of nodes

A node leaving a cluster and joining another cluster is known as reaffiliation. In Simulation the reaffiliation count provides the information [1]. The number of dead nodes vs. time determines the network life time [4]. Stability of a clusterhead is the duration for which a node remained a clusterhead [6]. The optimization of ratio of clusterheads to the total number of nodes is one of the goals of our algorithms.

3.6. Performance Comparison Reports

In Figure 4.1 to Figure 4.5, maximum displacement of the nodes is 10. The transmission range varies between 10 and 70. The sum of weight factors Wv for node v has to be 1. In our simulations, we assigned the weights as: Node Energy =0.05, Node Mobility =0.10, Node Degree=0.60, Neighboring Nodes positions=0.10, Data Rate=0.05 and Target Revisit Rate=0.10.

3.6.1. Average Number of Clusterheads

Average number of Clusterheads is considered very important in selection of an efficient clustering algorithm.

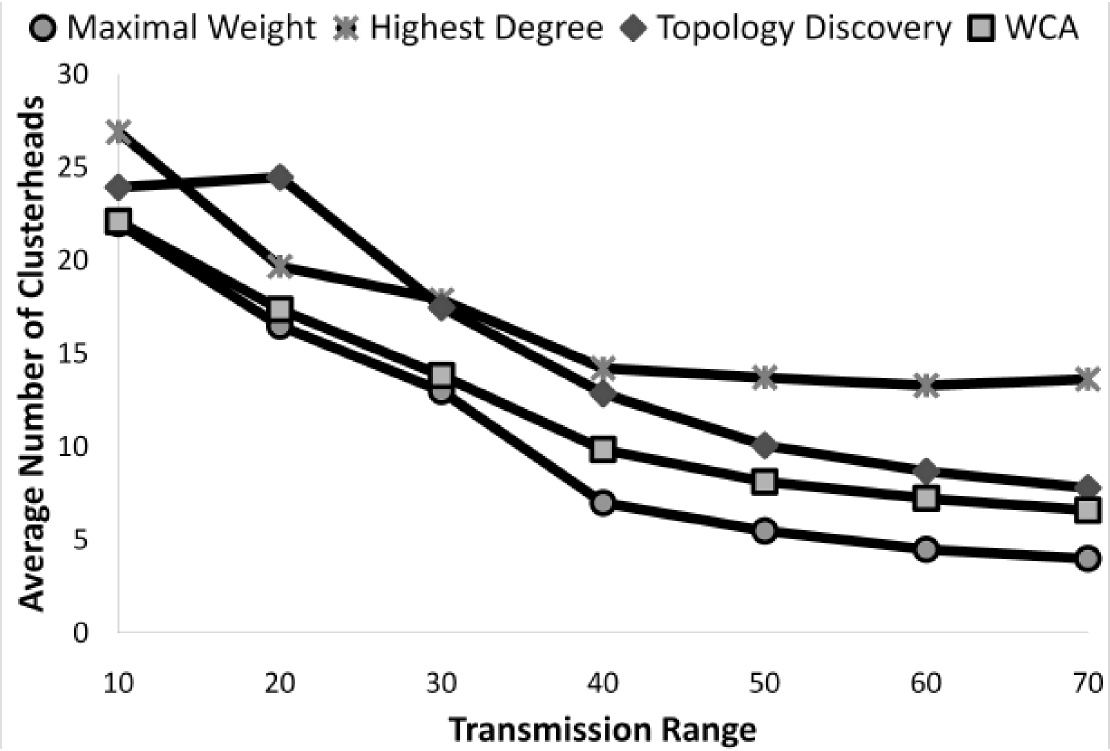
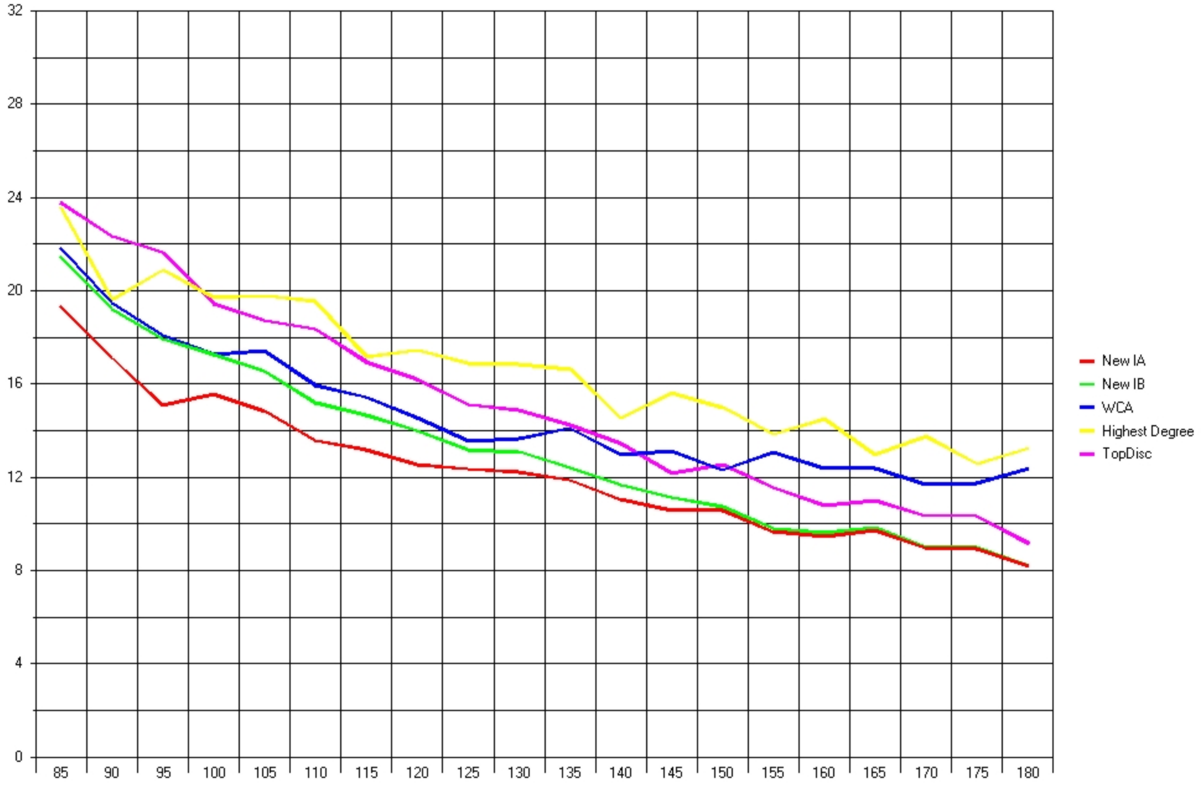


Figure 4.1: Average number of Clusters

The Figure 4.1 shows the Average number of cluster-heads for Maximal Weight, WCA, Highest Degree and Topology Discovery. The following simulated chart Average number of cluster-heads plotted against the transmission range is displayed below where New 1A is the maximal topology Algorithm and New 1B is the Bounded algorithm.



3.6.2. Number of Reaffiliations

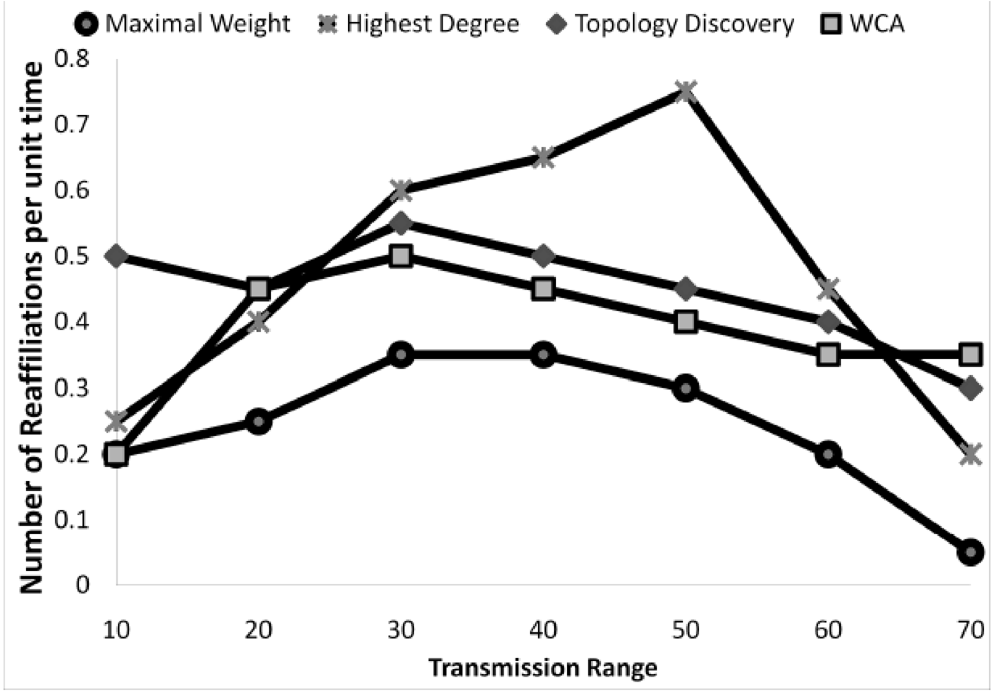
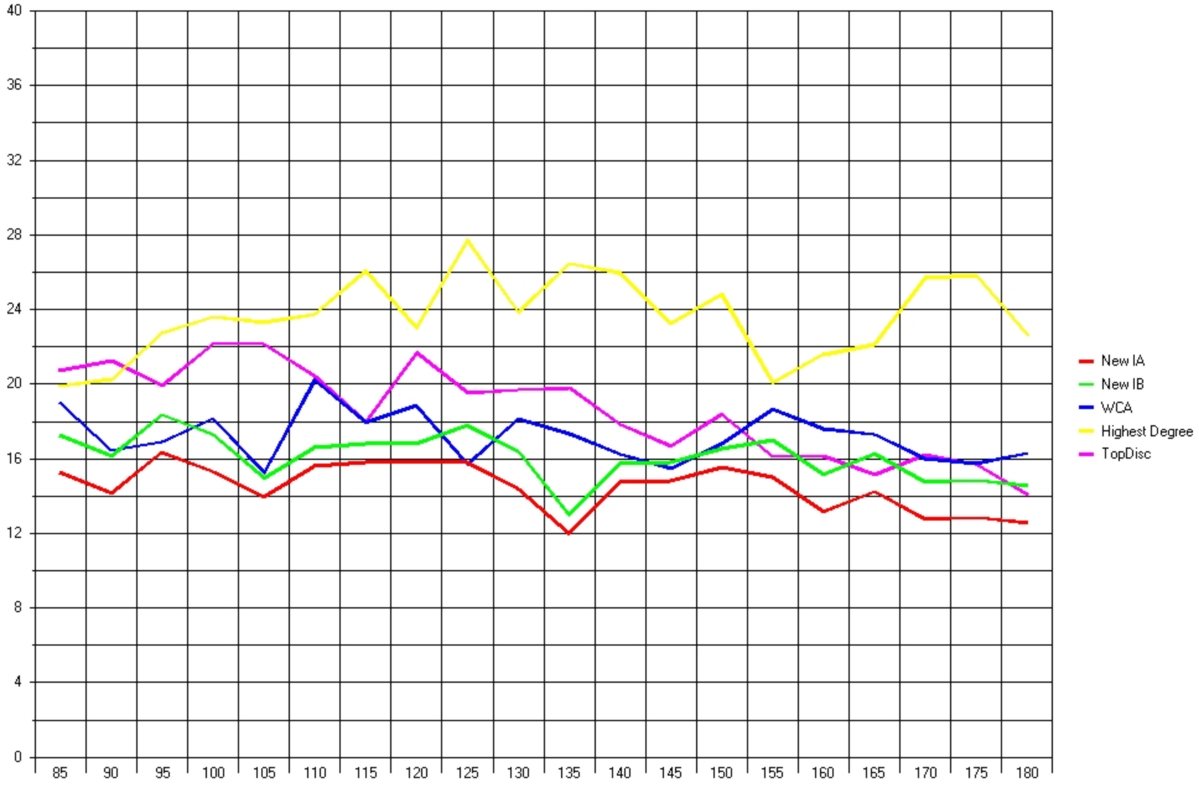


Figure 4.2: Number of Re-affiliations

The Figure 4.2 shows number of Re-affiliations. In Maximal Weight the number of reaffiliations has gone down significantly at higher transmission ranges in mobile environments where sensor nodes tend to move, the nodes often change their clusters. When a node leaves its cluster, the clustering algorithm is triggered again to reconfigure the network and affiliate the node with another cluster and update records. The process of reconfiguration has both computation and communication cost. In Fig. 7, Highest Degree and Topology discovery show much higher rate of reaffiliations. The number of reaffiliations of Maximal weight nearly becomes quarter of WCA as the transmission range is increasing. Maximal weight shows relatively more stability than the other three algorithms.

The following simulated chart generated by simulator Re-affiliations plotted against the transmission range is displayed below where New 1A is the maximal topology Algorithm and New 1B is the Bounded algorithm. The Algorithm clearly performs better than the existing proposed approaches.



3.6.3. Average Duration of a Clusterhead

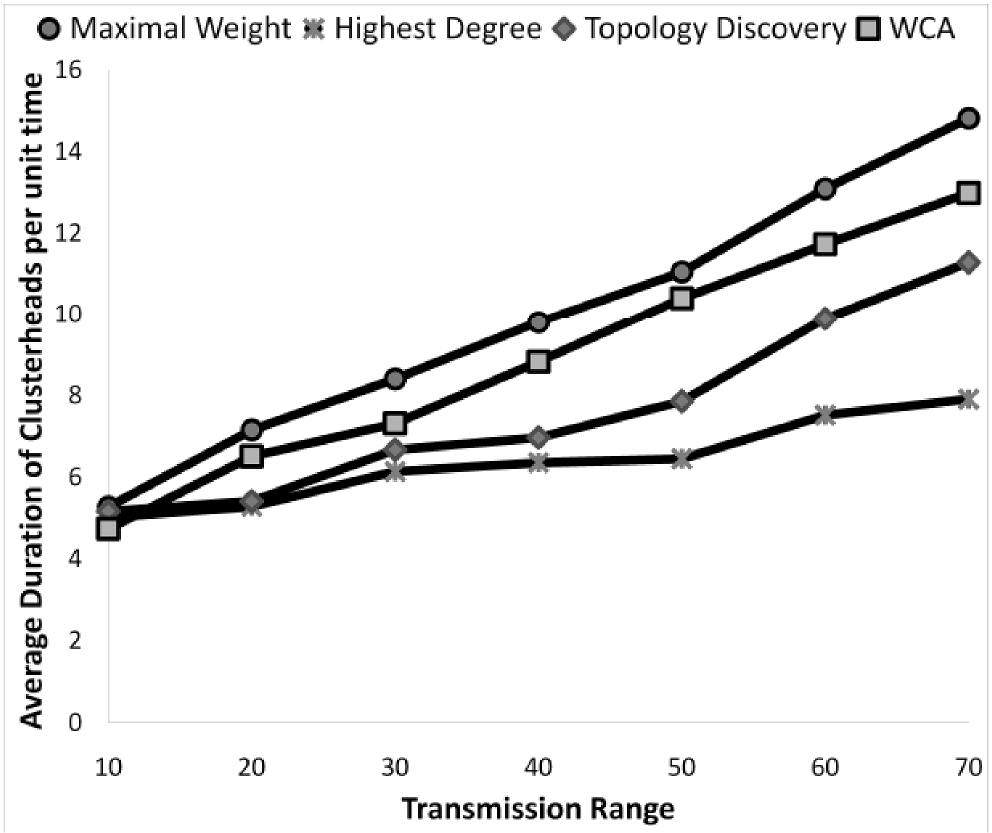


Figure 4.3: Average duration of a Cluster head

In the above Figure 4.3, Average duration of clusterheads in Maximal Weight, Topology Discovery, Highest Degree and WCA is shown. The stability is also determined by Average Duration of Clusterheads per unit time. The graph shows that Maximal Weight is the most stable algorithm over varying transmission ranges. A lower number of reaffiliations and a higher duration of a node being a clusterhead indicate the stability of a topology discovery algorithm.

3.6.4. Network Lifetime

The following Figure 4.4 will show the network life time of different clustering protocols in comparison to the newly proposed & implemented clustering protocols.

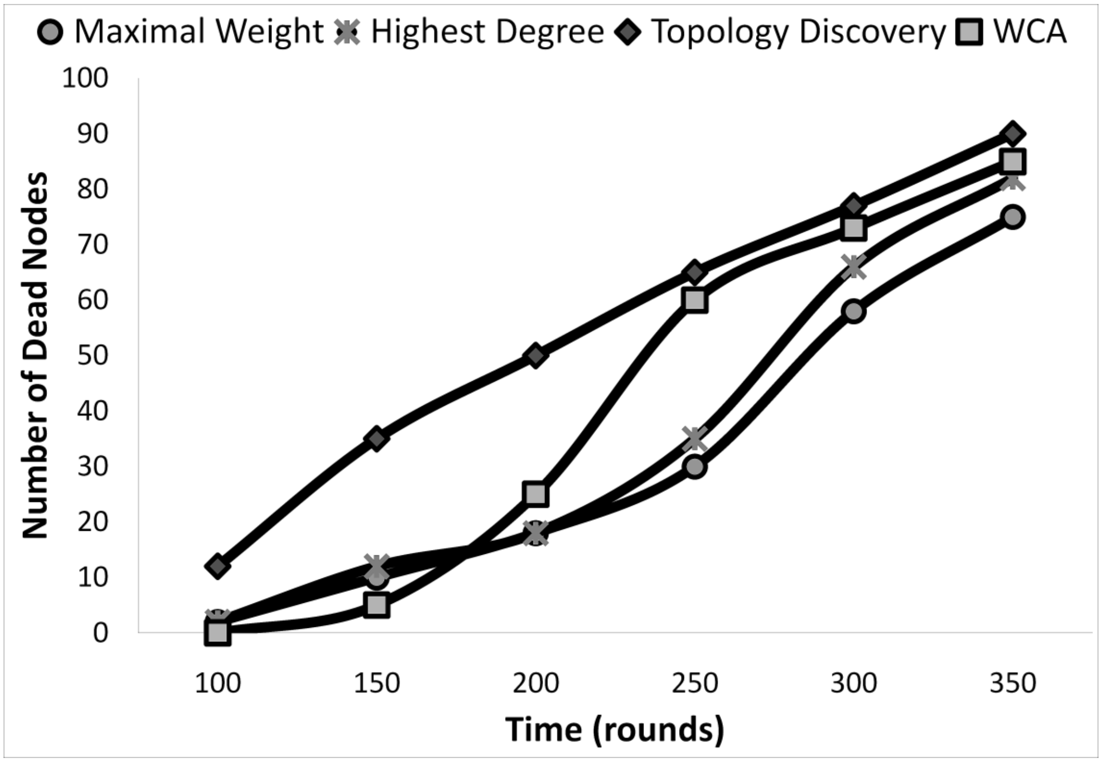
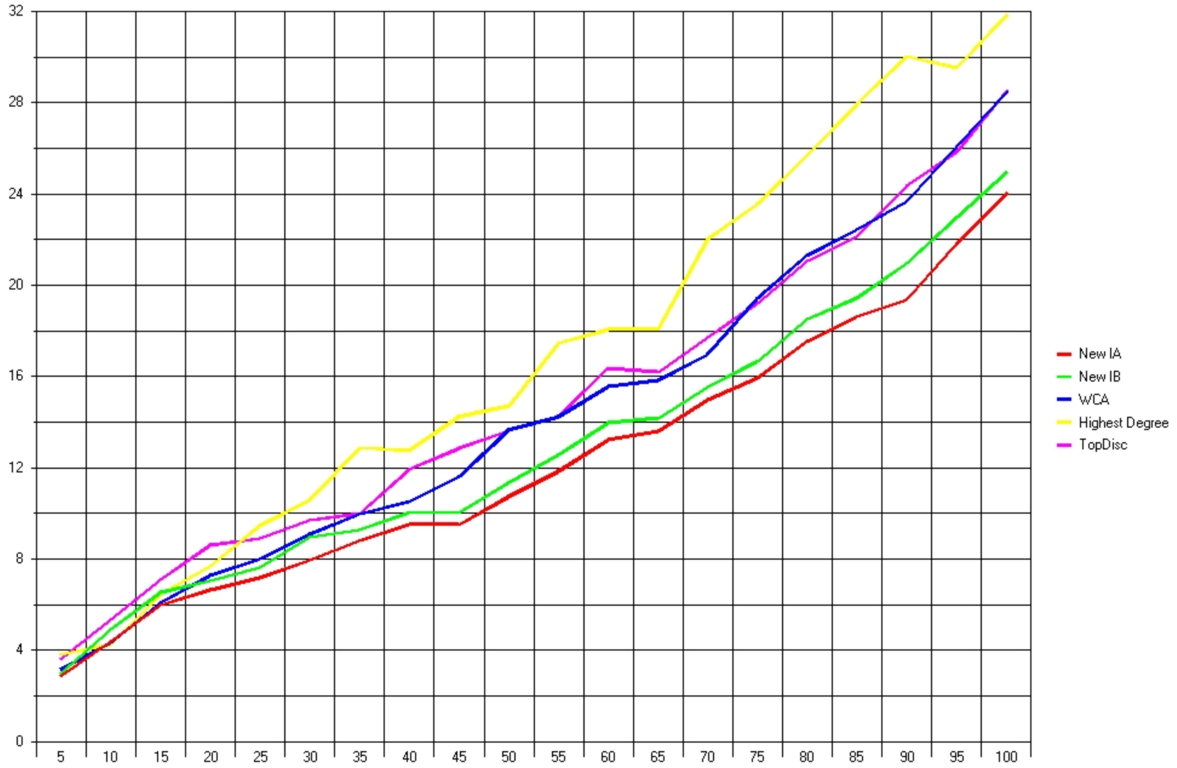


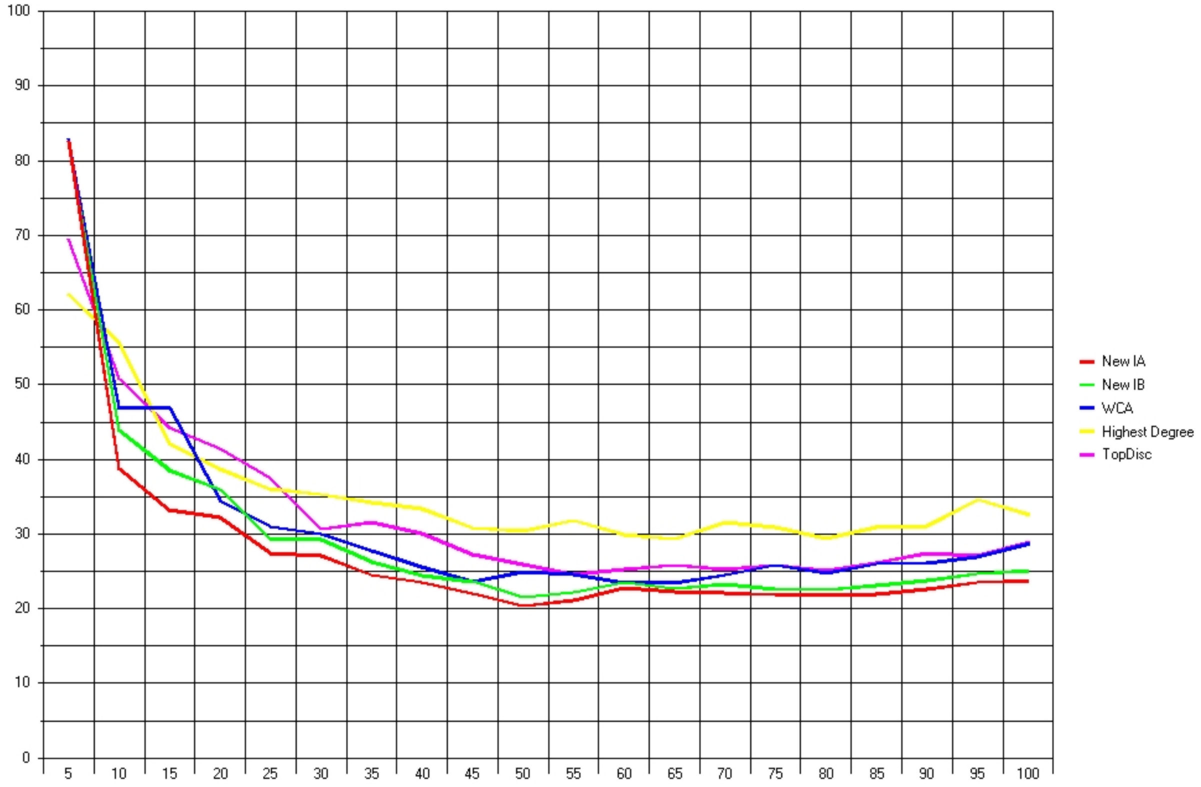
Figure 4.4: Network Lifetime

Network lifetime is the number of alive nodes over a period of time. In Fig. 4.4, as the time proceeds it can be seen that Maximal weight has lowest number of dead nodes. Number of nodes N is 100 in this simulation.

The following simulated chart generated by simulator number of cluster head plotted against the number of nodes is the network displayed below where New 1A is the maximal topology Algorithm and New 1B is the Bounded algorithm. The Algorithm clearly performs better than the existing proposed approaches. As we increase the number of nodes there is a linear increase in the number of cluster heads.



The following simulated chart generated by simulator have Ratio of cluster heads plotted against the number of nodes is the network. Maximal algorithm performs best.



3.6.5. Ratio of Black nodes to total number of nodes

A higher ratio of Clusterheads to ordinary nodes indicates that the network has high energy consumption; therefore it is desired to minimize the number of Clusterheads.

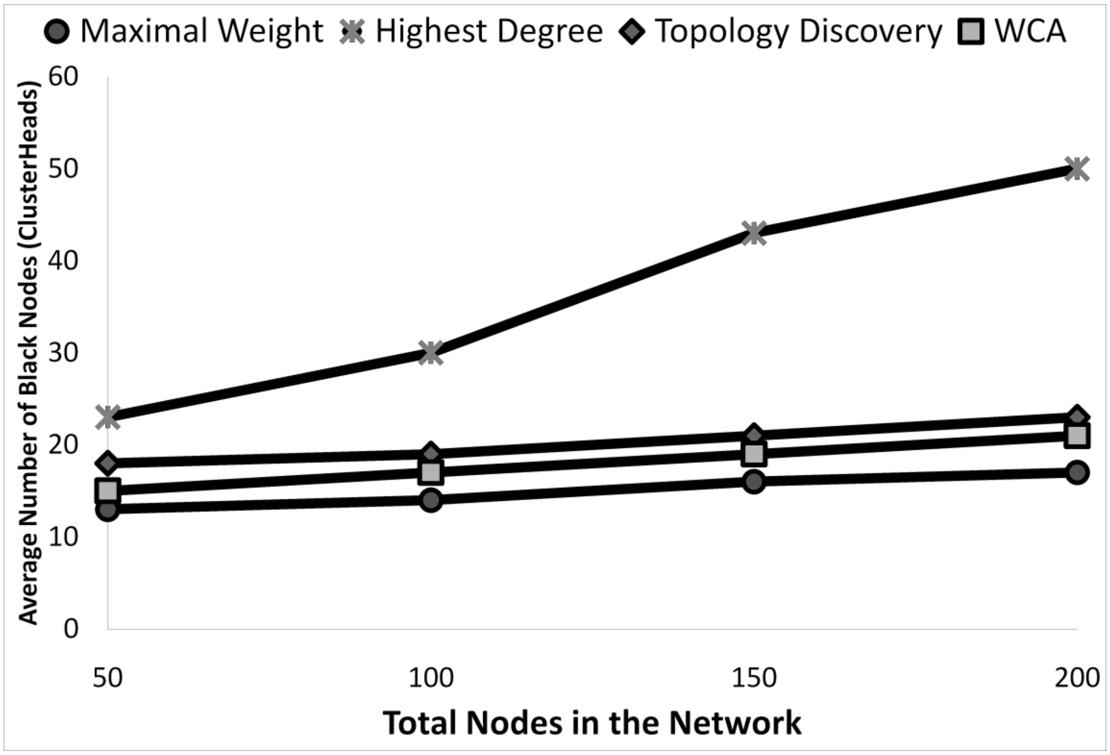


Figure 4.5: Number of Cluster heads for varying number of nodes.

Above Figure 4.5 shows the behavior of all the four algorithms. Maximal weight has relatively a lower ratio of black nodes to ordinary nodes. The simulation is performed from Number of nodes N ranging from 50 to 200.

3.7. Algorithms Analysis and Comparison with existing algorithms

In this report two algorithms have been proposed & provided with their implementation. Each Algorithm is separately compared with the existing popular algorithms.

3.7.1. Maximal Weight Topology Discovery Algorithm Comparison Table

Similar cluster heads are produced just as weighted clustering but we have used only the local information. More stable cluster than topology discovery.

PROPERTY	**Topology Discovery**	**WCA**	**Algorithms Presented** **Maximal Weight Topology Discovery**
Fault Tolerance	The approach is somewhat fault tolerant. This approach does not explain which particular node will start the topology discovery, in case the cluster head is dead.	Whenever a cluster head dies, among its neighbors the node with best weight will become the new clusterhead. But if the topology is disturbed, the network may not remain in the form of a connected graph.	Whenever a cluster head dies, among its neighbors the node with best weight will become the new clusterhead. The algorithm also caters if any other node beside cluster head dies. If the dead node is a forwarding node, then neighboring clusters propagate request to find the new forwarding node and update their routes.
Scalability	Highly Scalable	Not Scalable	Highly Scalable

Power Consumption	Low	Quite high	Average
Stateless Architecture	Supported	Supported	Supported
QoS Routing and Congestion Management	Not Supported (Only one route exists between two nodes.)	Supported (Multiple routes exist between two nodes)	Supported (Multiple routes exist between two nodes)
Service Differentiation	Very easy	Difficult	Very easy (do not forward packets to other clusters or forwarding nodes)
Topology Discovery	Very fast	Slow	Fast
Dynamic Reconfiguration	A bit tough	Tough	Easy (contains weights and requests)
Stability of clusters	Not stable	Very stable	Stable
Local self-healing	Possible	costly	Possible
Similar size clusters	No	Yes	Yes
Number of non-clustered nodes	Very less	Nil	Nil
Optimum number of clusters	Somewhat higher	Almost equal to	Almost equal to
Communication Cost	minimum	Huge	Minimum

3.7.2. Bounded Degree Algorithm Comparison Table

Similar cluster heads produced just as weighted clustering but we have bounded the cluster size. More stable cluster than topology discovery.

PROPERTY	**Topology Discovery**	**WCA**	**Algorithms Presented** **Bounded Degree**
Fault Tolerance	The approach is somewhat fault tolerant. This approach does not explain which particular node will start the topology discovery, in case the cluster head is dead.	Whenever a cluster head dies, among its neighbors the node with best weight will become the new clusterhead. But if the topology is disturbed, the network may not remain in the form of a connected graph.	Whenever a cluster head dies, among its neighbors the node with best weight will become the new clusterhead. The algorithm also caters if any other node beside cluster head dies. If the dead node is a forwarding node, then neighboring clusters propagate request to find the new forwarding node and update their routes.
Scalability	Highly Scalable	Not Scalable	Highly Scalable
Power Consumption	Low	Quite high	Average
Stateless Architecture	Supported	Supported	Supported
QoS Routing and Congestion	Not Supported	Supported	Supported

Management	(Only one route exists between two nodes.)	(Multiple routes exist between two nodes)	(Multiple routes exist between two nodes)
Service Differentiation	Very easy	Difficult	Very easy (do not forward packets to other clusters or forwarding nodes)
Topology Discovery	Very fast	Slow	Fast
Dynamic Reconfiguration	A bit tough	Tough	Easy (contains weights and requests)
Stability of clusters	Not stable	Very stable	Stable
Local self-healing	Possible	costly	Possible
Similar size clusters	No	Yes	Yes
Minimum number of non-clustered nodes	Very less	Nil	Nil
Optimum number of clusters	Somewhat higher	Almost equal to	Almost equal to
Communication Cost	minimum	Huge	Minimum

CHAPTER 5 - CONCLUSION & FUTURE WORK

This research report explains the concept of wireless sensor network which comes into existence with the help of sensor nodes performing the function of sensing, computing and communication with each other to deliver timely information to the base station. Sensor nodes are tiny resource constrained devices having limited battery power which introduced many new challenges for wireless sensor networks. Various applications were discussed in the report for wireless sensor network. There are various topologies proposed for the wireless sensor network. Issues such as Fault tolerance, scalability, power consumption, stateless architecture were discussed and light is also shed upon many challenges such as Adding QOS routing, congestion management support, service differentiation and devising efficient topology discovery and dynamic reconfiguration algorithms for wireless sensor networks.

After the domain study it is concluded that clustering is the most important and efficient way the sensor network can be organized. After defining the clustering form of topology in the report it is explained how the clustering process is performed on a region where sensor nodes are deployed. After deployment of the sensor nodes, they organize themselves in the form of a cluster and later a hierarchy structure is made by the clusterheads to deliver information regarding any sensitive event to the base station. The motivations behind the use of clustering schemes in wireless Sensor Network are, Reduce communication information, Favor spatial reuse, Minimize control information, Inter-cluster communication which allows the creation of backbone-based architectures and prolong network life time. To devise an efficient topology algorithm scalability, local self healing, similar size clusters are some of the challenges faced by the designers of the network.

After providing a review of popular clustering techniques for wireless sensor network, a security requirement is evaluated. Later, explained a review for the sensor network for various threats and highlighted numerous problems and available solutions.

Two Clustering Algorithms have been proposed by keeping in mind various challenges for wireless sensor network. The basics for the new proposed algorithms have been provided.

The proposed algorithms which are compared with the available algorithms and their results predicted that proposed clustering algorithms perform better in different conditions. Selection of cluster head is not dependent on only one or two factors but many factors are involved in deciding about a node that becomes cluster head. The new proposed Clustering can be scalable to cater hundreds or thousands of nodes deployed over hostile environment. Results show that proposed algorithms have efficient resource utilization. The number of cluster heads is an important factor to be considered because as the number of cluster heads increases, the number of hops for information flow from one end to the other end will also increase.

Both Topology Discovery algorithm and Weighted Average Clustering Algorithm are explained and implemented for simulator. Both of these algorithms are studied critically and the report pointed out the weaknesses in these algorithms. To prove new proposed algorithms present better solution than existing algorithms, all algorithms are simulated and compared with each other. The simulations show that new proposed algorithms behave better for varying number of nodes at variable transmission ranges as well as speed. Proposed algorithms are better in terms of rearrangements, number of clusterheads and number of clusters when compared with existing popular techniques.

The Average number of cluster-heads for Maximal Weight algorithm comes out to be less than all the heuristics, which concludes significant saving in energy as increasing the number of cluster heads increases the communication overhead. In Maximal Weight the number of reaffiliations has gone down significantly at higher transmission ranges in mobile environments, where sensor nodes tend to move or often change their clusters. The stability is determined by Average Duration of Clusterheads per unit time which comes out to be highest for both proposed algorithms in comparison with existing algorithms. With the passage of time less nodes tend to die for a network running the Maximal weight algorithm. A complete and comprehensive algorithm analysis is provided for both of the new proposed algorithms. They are compared with the existing algorithms for various parameters which are considered important in the domain of wireless sensor network. By providing the complete performance analysis it is concluded that the proposed system provides stable and satisfactory results.

In the research report efficient self healing clustering solution has been proposed but there is a great concern to add security to the proposed system developed for wireless sensor network. Following are the open research areas in the domain of security for wireless sensor networks.

- In WSN symmetric key approaches aim at link layer security for one hop communication. They do not aim for transport layer for multihop communication.
- Asymmetric key technology is computationally expensive but provide easily manageable infrastructure.
- Computation efficiency and authentication of public key are considered critical issues in Asymmetric key Technology for WSN.
- There is still demand for more efficient symmetric key algorithms and system can be enhanced to have the capability of secure communication among sensor nodes.
- Key revocation is another problem which is currently not been addressed, in which key of compromised or malicious node must be revoked from neighboring nodes memory as it cannot participate in communication.

As WSN is comprised of thousand of nodes so there is a high demand for simple, efficient and scalable security solution and protocols which are designed by keeping in mind the resource limitation and constraint of WSNs.

REFERENCES

[1] P. Krishna, N.N. Vaidya, M. Chatterjee, D.K. Pradhan, *"A Cluster-Based Approach for Routing in Dynamic Networks,"* ACM SIGCOMM Computer Communication Review, 49, 1997.

[2] F. G. Nocetti, J. S. Gonzalez, I. Stojmenovic, "*Connectivity based k-hop clustering in wireless networks,"* Telecommunication Systems 22 (2003), 1-4, 205-220.

[3] W. Heinzelman, A.P. Chandrakasan, and H. Balakrishnan, *"Energy Efficient Communication Protocols for Wireless Microsensor Networks,"* In Proceedings of Hawaiian International Conference on Systems Science, 2000.

[4] B. Deb, S. Bhatnagar, B. Nath, *"A Topology Discovery Algorithm for Sensor Networks with Applications to Network Management,"* In IEEE CAS workshop, Sept. 2002.

[5] S. Sivavakeesar, G. Pavlou*, "A Prediction Based Clustering Algorithm to Achieve Quality of Service in Multihop Ad Hoc Networks,"* CCSR, Proceedings of the London Communications Symposium (LCS), London, UK, pp. 17-20, September 2002.

[6] M. Chatterjee, S. K. Das, D. Turget, *"A Weighted Clustering Algorithm for Mobile Ad Hoc Networks,"* Cluster Computing, 193-204, 2002.

[7] Begumhan Turgut, Ramez Elmasri and Than V. Le *"Optimizing Clustering Algorithm in Mobile Ad-Hoc Networks Using Simulated Annealing"*

[8] Yasir Fayyaz, Mehwish Nasim and Muhammad Younus Javed, "*Maximal Weight Topology Discovery in Ad-hoc Wireless Sensor Networks*", Proceedings of the 10th IEEE International Conference on Computer and Information Technology (ICIT 2010), 29 Jun – 1 Jul, 2010 (ISBN: 978-0-7695-4108-2). Bradford, **UK**, pp.715-722.

[9] D. J. Baker and A. Epheremides, "*The Architectural Organization of a Moblie Radio Network via a Distributed Algorithm*," IEEE Transactions on Communications, vol. Com-29, no. 11, November 1981.

[10] P. Tsigas, "Project on Moblie Ad Hoc Networking and Clustering for the Course EDA390 Computer Communcation and Distributed Systems," Manual for University Course.

[11] A. Amis, R. Prakash, T. Vuong, and D. Huynh, "*Max-Min D-Cluster Formation in Wireless Ad Hoc Networks*," IEEE INFOCOM, March 2000.

[12] C. Johnen and L. Nguyen. "*Self-stabilizing clustering algorithm for ad hoc networks,*" Technical Report no. 1357, L.R.I, Universit´e de Paris Sud, 1429, 2006.

[13] *Congfeng Jiang, Daomin Yuan, Yinghui Zhao,"Towards clustering algorithms in wireless sensor networks: a survey," Wireless Communications and Networking Conference, 2009. WCNC 2009. IEEE*

[14] Kavitha, T., & Sridharan, D. (2010). Security Vulnerabilities in Wireless Sensor Networks: A survey. *Journal of Information Assurance and Security, 5,* 31-44.

[15] Chris karlof, Naveen Sastry, David Wagner,"TinySec: A Link Layer Security Architecture for Wireless Sensor Networks," Proceedings of the 2nd international conference on Embedded networked sensor networks, p-162-175, 2004

[16] M. Qin and R. Zimmermann. VCA: An energy-efficient Voting-based Clustering Algorithm for sensor networks. *Journal of Universal Computer Science*, 13(1):87--109, 2007.

[17] Maan Younis Abdullah, Gui Wei Hua, "Cluster-Based Security for Wireless Sensor Networks," cmc, vol. 3, pp.555-559, 2009 WRI International Conference on Communications and Mobile Computing, 2009

[18] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Sensor Networks," Proceedings of the 33th Hawaii International Conference on System Sciences, 2000.

[19] Falko Dressler, Yong Guan, Zhen Jiang, "Wireless and Sensor Networks Security (WSNS) A Retrospection," mahss, pp.1-6, 2007 IEEE Internatonal Conference on Mobile Adhoc and Sensor Systems, 2007

[20] Akbar Abbasi, "Better Security for Wireless Sensor Networks," p-100-103 ICFN '09 Proceedings of the 2009 International Conference on Future Networks

[21] L. Liestman *"Approximating Minimum Size Weakly-Connected Dominating Sets for Clustering Mobile Ad Hoc Networks"*, 2002.

[22] Al-Sakib Khan Pathan, Hyung-Woo Lee, Choong Seon Hong: Security in Wireless Sensor Networks: Issues and Challenges CoRR abs/0712.4169: (2007)

[23] Hongfa Wang, "A Robust Mechanism for Wireless Sensor Network Security" 4th International Conference on Wireless Communications Networking Mobile Computing (WiCOM '08), PP 1-4, Oct 2008.

[24] Xiaojiang Du Hsiao-Hwa Chen, "Security in wireless sensor networks". Wireless Communications, IEEE Publication Date: Aug. 2008 Volume: 15 , Issue: 4 On page(s): 60 – 66.

[25] Fei Hu, Jason Tillet, Jim Ziobro, and Neeraj K. Sharma, "Secure Wireless Sensor Networks: Problems and Solutions", Journal on Systemics, Cybernetics and Informatics (Best Paper Award), Vol.1, No.9, 2004.

[26] Vidyasagar Potdar, Atif Sharif, Elizabeth Chang, "Wireless Sensor Networks: A Survey," waina, pp.636-641, 2009 International Conference on Advanced Information Networking and Applications Workshops, 2009

[27] Ameer Ahmed Abbasi, Mohamed F. Younis: A survey on clustering algorithms for wireless sensor networks. Computer Communications 30(14-15): 2826-2841 (2007)

[28] Luigi Alfredo Grieco, Gennaro Boggia, Sabrina Sicari, Pietro Colombo, "Secure Wireless Multimedia Sensor Networks: A Survey," ubicomm, pp.194-201, 2009 Third International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies, 2009

[29] David Boyle, Thomas Newe, “Securing Wireless Sensor Networks: Security Architectures” in Journal of Networks, Vol. 3, No. 1, January 2008.

[29] Chen, X., Makki, K., Yen, K., & Pissinou, N. (2009). Sensor network security: a survey. *IEEE Communications Surveys Tutorials*, *11*(2), 52-73

[30] Mehwish Nasim, Yasir Fayyaz and Muhammad Younus Javed, “*Bounded Degree Energy Aware Topology Discovery in Ad-hoc Wireless Sensor Networks*”, Proceedings of the Fifth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP 2009), December 7-10, 2009, Melbourne, Australia, pp. 13-18.