ENERGY EFFICIENT ROUTING

IN

WIRELESS SENSOR NETWORKS

Thesis submitted in partial fulfillment of the requirements

for the award of degree of

Masters in Engineering

In

Computer Science and Engineering

Submitted By

Sukhchandan Randhawa

(Roll No.: 801032026)

Under the supervision of:

Dr. Anil Kumar Verma

Associate Professor, CSED Thapar University

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT

THAPAR UNIVERSITY

PATIALA -147004

MAY 2012

Certificate

I hereby certify that the work which is being presented in the thesis entitled, **"Energy Efficient Routing in Wireless Sensor Networks"**, in partial fulfillment of the requirements for the award of degree of Master of Engineering in Computer Science and Engineering submitted in Computer Science and Engineering Department of Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of *Dr. A. K. Verma* refers other researcher's works which are duly listed in the reference section.

The matter presented in this thesis has not been submitted for the award of any other degree of this or any other university.

(Sukhchandan Randhawa)

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

(Dr. A. K. Verma)
Associate Professor
Computer Science & Engineering Department
Thapar University
Patiala.

Countersigned by:

(Dr.Maninder Singh)
Assoc. Professor & Head
Computer Science & Engineering Department,
Thapar University,
Patiala.

(Dr. S. K. Mohapatra)
Dean (Academic Affairs)
Thapar University,
Patiala.

Acknowledgement

No volume of words is enough to express my gratitude towards my guide, ***Dr. A.K. Verma****, Assistant. Professor, Computer Science and Engineering Department, Thapar University, who has been very concerned and has aided for all the material essential for the preparation of this thesis report. He has helped me to explore this vast topic in an organized manner and provided me with all the ideas on how to work towards a research-oriented venture.*

I am also thankful to ***Dr. Maninder Singh****, Head of Department, CSED and* ***Mr. Karun Verma****, P.G. Coordinator, for the motivation and inspiration that triggered me for the thesis work.*

I would also like to thank the staff members and my colleagues who were always there in the need of the hour and provided with all the help and facilities, which I required, for the completion of my thesis.

Most importantly, I would like to thank my parents and the Almighty for showing me the right direction out of the blue, to help me stay calm in the oddest of the times and keep moving even at times when there was no hope.

Sukhchandan Randhawa

801032026

Abstract

Wireless Sensor Networks is an important research area as, WSNs have number of applications such as battlefield surveillance; today such networks are used in many industrial and consumer applications, such as industrial process monitoring and control, machine health monitoring, and so on. Now one of the biggest constraints is the energy consumed by nodes of WSNs, which requires frequent recharging and thus influences the mobility. So the energy consumption of nodes in WSNs assumes relatively high attention as compare to other parameters.

In the proposed work, energy comparison is made on the basis of various parameters such as residual energy Vs Time, Packet Delivery Fraction, Dropped packet Vs Time due decrease in the energy. This comparison is made between various routing protocols such as Flat routing Protocol(DSR,DSDV, Flooding, AODV,MAODV ,Directed Diffusion), Hierarchical Routing Protocols(LEACH),and Location Based Routing Protocols(GAF) and our results indicate that Location Based Routing Protocols performs better than Flat Routing Protocols as well as Hierarchical Routing Protocols(LEACH)

Keywords—Wireless Sensor networks, Energy Efficient Routing.

Table of Contents

List of Figures and Tables

List of Abbreviations

AODV	Ad-Hoc on-Demand Distance Vector
APTEEN	Adaptive Threshold sensitive Energy Efficient Network protocol
API	Application Programming Interface
CADR	Constrained Anisotropic Diffusion Routing
CDMA	Code division multiple access
DD	Directed Diffusion
DSDV	Destination-Sequenced Distance-Vector
DSR	Dynamic Source Routing
FEC	Forward Error Correction
GAF	Geographic Adaptive Fidelity
GBR	Gradient-based Routing
GEAR	Geographic and energy-aware routing
GPS	Global Positioning System
LEACH	Low Energy Adaptive Cluster Hierarchy
MAC	Medium Access Control
MECN	Minimum energy communication network
MEMS	MicroElectonicsMechanical Systems
NAM	Network Animator
PA	Power Available
PDF	Packet Delivery Fraction
PEGASIS	Power- Efficient Gathering in Sensor Information Systems
QoS	Quality of Service
RF	Radio Frequency
TDMA	Time division multiple access
TEEN	Threshold sensitive Energy Efficient sensor Network protocol

VGA	Virtual Grid Architecture
WSN	Wireless Sensor Networks

Chapter 1

Introduction

1.1 Wireless Sensor Networks

A WSN [1][2] consists of a large number of low-cost, low–power sensor nodes that are deployed in a area of interest .Sensors have computation, communication, sensing capabilities. Sensor communicates via a short range radio signals and collaborate to accomplish the common tasks as shown in figure 1.1 [3] and having limited bandwidth, power, memory, processing resources and limited lifetime [2].

Fig1.1: WSN [3]

There are many applications of the WSN, such as - Environment Monitoring, Military Applications, and Health care applications, Industrial Process Control, Home Intelligence, Security and Surveillance etc.

Also, following objectives are to be kept in mind when proposing any WSN solution - Small size, Power efficient, Scalability, Secure network design,Adaptability,Self-Configurability,Reliability,Fault-Tolerance,Bandwidth utilization, Low cost, QoS Support.

1.2 Sensor Node Structure:

A sensor node usually consist of 4 components - Sensing Unit, Processing Unit, Communication Unit and a Power Unit [4].

It can also be equipped with other units, depending upon specific applications. For example a Global Positioning System may be needed that provides global location information. A motor may be needed to move sensor nodes in some sensing tasks, as shown in figure 1.2, below.

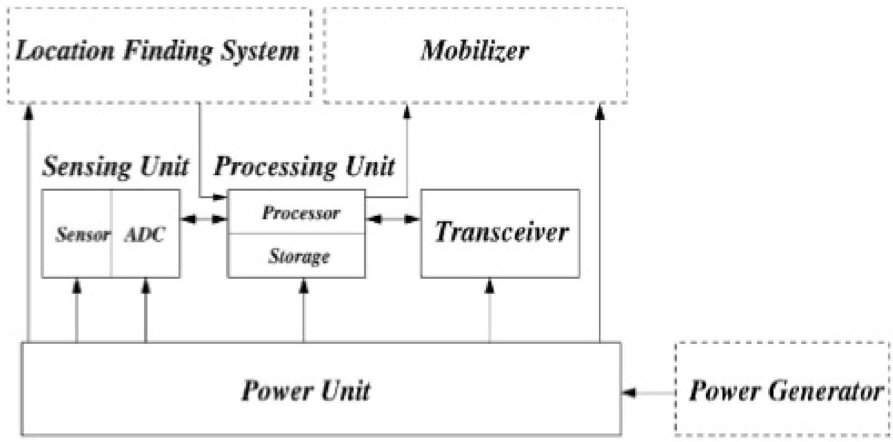


Fig 1.2: Components of Sensor Node [4]

1.3 Protocol Stack for WSN

It consists of 5 layers [5][6]: the physical layer, data link layer, network layer, transport layer and application layer as shown in the figure 1.3.

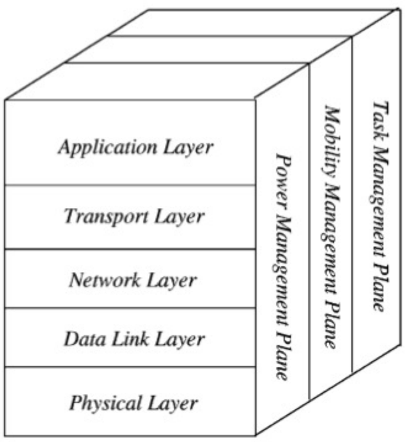


Fig 1.3. Sensor network Protocol Stack

- **Application Layer**: It performs various application oriented functions such as query dissemination, node localization, network security. For example, SMP (Sensor Management Protocol) is an application layer protocol whose main purpose is to perform a variety of task .For example: Exchanging location related data, moving sensor nodes, querying the sensor nodes.
- **Transport Layer**: It is responsible for reliable end–to-end delivery of data between the nodes and sink. There are 2 type of data delivery in WSN: Upstream and Downstream. In upstream, sensor nodes send data to sink node while in downstream, data is originated at sink node. The reliability of both upstream and downstream data delivery are different. For example in upstream, data delivery can be fault tolerant but this is not the case in case of downstream.
- **Network Layer**: This layer is responsible for routing the data from the transport layer. It is important to take into account the energy constraint as well as unique traffic pattern in the design of network layer and routing protocols(for multi-hop, single-hop)[5].
- **Data Link Layer**: It is responsible for data multiplexing, data frame creation, medium access and error control in order to provide reliable point-to-point and point to multipoint transmissions. The primary objective of MAC is to share the resources among multiple sensor nodes in term of throughput, latency, energy consumption. In case of error control ARQ (automatic repeat request) cannot be used, even FEC (Forward Error Correction) is also little bit complex but it is still used in sensor networks.
- **Physical Layer**: It deals with the issues like conversion of bit stream in to signals, frequency selection, data encryption, carrier frequency generation, signal modulation, design of underlying hardware and various electrical and mechanical interfaces.

Management Planes [7]: The protocol stack s also divided into management planes including power, connection and task management planes.

- **Power Management Plane**: It is responsible for managing the power level of node for various tasks such as sensing, computation and communication. For example, sensor can be turned off when there is no data to transmit and receive.
- **Connection Management Plane:** It is responsible for configuration or reconfiguration of sensor nodes in order to establish and maintain the connectivity of a network in case of node failure, node movement, addition of node etc.
- **Task Management Plane**: It is responsible for task distribution among various nodes to improve energy efficiency and prolong the network lifetime. It is possible in case of densely deployed sensor network that redundant nodes may perform the similar task that is of no use and leads to wastage of energy resources, so tasks should be properly assigned.

1.4 Technological Background:

The concept of WSN was introduced near about three decades ago. In initial stages this technology was not as much explored because of the state-of-art in sensor, wireless technology, and computers [8][9][10]. So it had limited military applications. But as there are technological advancements in MEMS, it is possible to have numerous applications of this technology.

- **MEMS Technology**: This is based on micromachining techniques, which have been developed to fabricate micron-scale mechanical components that are controlled electrically resulting in MEMS. It is technology for manufacturing tiny, low-cost and low-power sensor nodes. By using MEMS technology, many components of sensor nodes can be miniaturized, for example sensors, communication blocks, power supply units which can significantly reduce the cost. There can be many micromachining techniques like planar, bulk, surface micromachining.
- **Wireless Communication Technology**: In most of the wireless networks these days, Radio Frequency(RF) is used including Microwave and millimeter wave, because these doesn’t need line of sight and are Omni directional.

But these are also having some disadvantages, for example large radiators and low transmission efficiencies is not suitable for energy constrained environment's, so the remedy to this problem is usage of the optical communication.

Another issue is this, that the existing networking protocol cannot be applied directly to the WSNs, so these protocols require modifications.

1.4.1 Software Subsystems

WSN have following 5 software subsystems [8]:

1. **Operating System(OS) Microcode or Middleware**:

   The general purpose of operating system is to shield the software from lower level details of the machine. There should be open system operating system designed for WSNs that enables the rapid development while minimizing code size. For Example: TinyOS.

2. **Communication Drivers** (encoding and the physical layer): This code typically deals with functions like clocking and synchronization, signal encoding, bit recovery, bit counting, signal levels and modulation.
3. **Communication Processors:** This code deals with the communication functions like routing, packet buffering, topology maintenance, encryption and FEC.
4. **Sensor Drivers:** Depending upon the type and configuration of the sensors appropriate setting and functions must be loaded so that they can work properly, so all these tasks are performed by the modules of sensor drivers.
5. **Data processing Mini apps:** These are data processing, signal value storage and manipulation or other applications that are supported at node–level for in–network processing.

1.4.2 Hardware Platforms:

These can be of 3 types [9]:

- **Augmented General Purpose PCs**: it includes various low-power PCs and PDAs which run off-the-shelf Operating System, for example Linux or real time Operating System. These systems have higher processing capabilities and have networking protocols, middleware and Application Programming Interfaces. But the problem is this, that these require more power supply.

- **System-on-chip Sensor Nodes**: These are based on CMOS,MEMS and RF technologies .These consume very less power and having very less computation, sensing and communication capabilities. For example BWRC PicoNode.
- **Dedicated Sensor Nodes**: These are having very simple low-power processing and communication, interfaces. For example Medusa family, Berkeley Mote family.

1.5 Wireless Sensor Network Standards: It is important to establish standards so that products developed by different vendors may interoperate. A huge effort had been made to make standardization in order to unify the market, leading to low cost and interoperable devices and available wireless protocols [9] has been shown in Figure 1.4

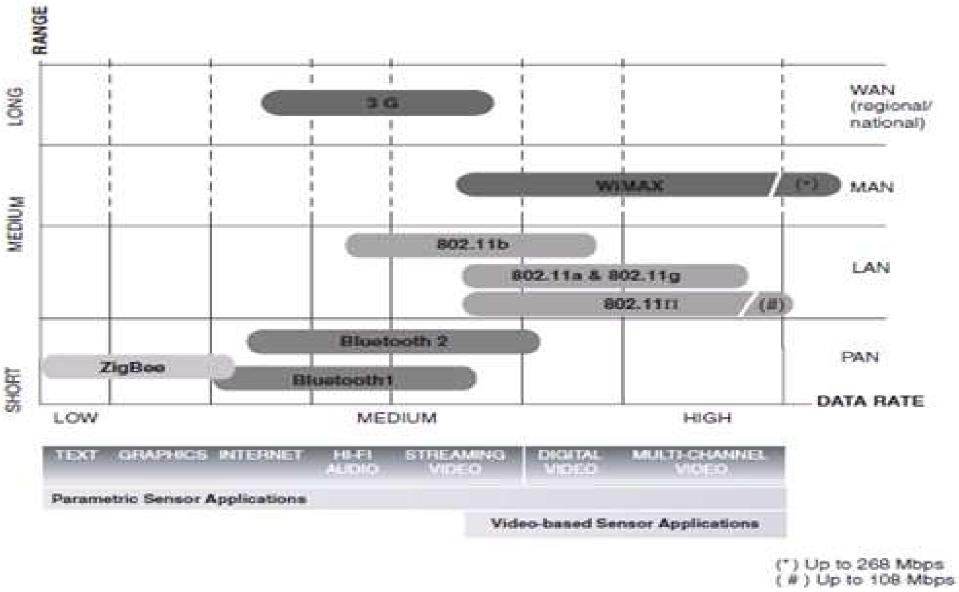


Fig 1.4: Available Wireless Protocols [8]

Table 1: Wireless Protocol Comparison [8]

Property	802.11	802.15.1/Bluetooth	802.15.4/ZigBee
Range (m)	100	10 to 100	10
Data throughput (Mbps)	2 to 54	1 to 3	0.25
Power consumption	Medium	Low	Ultralow
Battery life measured in:	Minutes to hours	Hours to days	Days to years
Size relationship	Large	Smaller	Smallest

The IEEE 802.15.4 standard supports a maximum data rate of 250 kbps, with rates as low as 20 kbps (slower than most telephone modems); however, it has the lowest power requirement of the group. ZigBee devices are designed to run on a single set of batteries, making them ideal candidates for unattended or difficult-to reach locations. Bluetooth is a short-range communication protocol widely used in cellular-type phones and PDAs (has a range of about 10 m, or a maximum of 100 m with power boost); it operates in the 2.4-GHz ISM band and has a bandwidth of approximately 1 to 3 Mbps. IEEE 802.11a/b/g/n is a collection of related technologies that operate in the 2.4-GHz ISM band, the 5-GHz ISM band, and the 5-GHz UNII bands; it provides the highest power and longest range of the common unlicensed wireless technologies. Transmission data rates can reach 54 Mbps (twice as much with the latest IEEE 802.11n protocol). Typically, hardware implementation of some or all of 802.11 protocols comes preinstalled on most new laptop computers; the technology is often also available for PDAs and cellular phones.

1.6 Network Architectures [9]:

There can be two types of architectures which are further classified as shown in figure 1.5.First of all there can be a single sink or multiple sink depending upon the area .The sink act as a gateway to outside world. Its main purpose is to collect the data from the sensor nodes, aggregates data and sends the result back to the user as a response to his query.

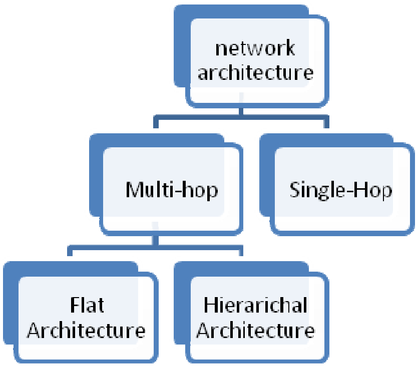


Fig 1.5:Classification of Network Architecture[9]

- **Single Hop Architecture**:

  In this architecture, the sensor node directly sends the data to the sink in single hop without forwarding it to any intermediate node.

  Disadvantage: As communication cost is more than sensing and computation cost in WSN. So it is very costly in terms of its energy consumption and moreover transmission power grows exponentially with the increase in distance.

- **Multi-Hop Architecture**: In densely deployed networks ,all the nodes are so close that the data can travel from one hop to other in by covering short distance. It is preferred over the single-hop arch. As it saves more energy. It can be further classified as

  - **Flat Architecture:**

    In this architecture, all nodes act as peers rather than acting as master slaves. Due to large number of nodes address centric routing is not possible (in this case we have to maintain global identification info), so data centric routing is more suitable. In this type of routing sink sends a query to all nodes, and the node which has the required data, only will respond via multi-hop path using peers just as relays.

  - **Hierarchical Architecture:**

    In this, nodes are divided into clusters and each cluster is having a cluster head which act as relays when all the nodes in the cluster send their data to them. Cluster head forwards that data to the sink as a response to the query. In this nodes having high energy act as a cluster head and nodes having low energy simply perform the task of sensing. In this way traffic, load is balanced.

But the problem with this architecture is this that how we can choose the cluster head and organize them. There are many strategies for e.g. Single-tier clustering architecture or multi-tier architecture.

1.7 Key Research Areas In Wireless sensor Networks

There are number of key research areas in wireless sensor networks in which a lot of efforts are being made [8][9]:

a. **Modulation schemes:** Simple and low-power modulation schemes need to be developed for sensor networks. The modulation scheme can be either baseband, as in UWB, or pass band.

b. **Strategies to overcome signal propagation effects:** There are many problems with the signal transmission like signal fading- selective fading, absorptive fading. a lot of research has been done in this area but still it is not completely successful.

c. **Hardware design**: Tiny, low-power, low-cost transceiver, sensing, and processing units need to be designed. Power-efficient hardware management strategies are also essential. Some strategies are managing frequencies of operation, reducing switching power, and predicting work load in processors.

d. **Determination of lower bounds on the energy required for sensor network self-organization:** Depending upon the power available or energy available several methods has been used to determine the residual energy of the network for their self-organization.

e. **Error control coding schemes**: Error control is extremely important in some sensor network applications like mobile tracking and machine monitoring. The feasibility of other error control schemes in sensor networks needs to be explored.

f. **Power-saving modes of operation:** To prolong network lifetime, a sensor node must enter into periods of reduced activity when running low on battery power. The transition management for these nodes is open to research.

g. **Sensor Query and data dissemination protocols:** SQTL language has been widely used for this purpose but because long delays its performance is not so good.

Chapter 2

Literature Survey

2.1 Power in Wireless Sensor Networks

Due to the large number of sensor nodes that may be deployed and because of the harsh hostile environment in which these are deployed, replacing the battery is not an option. Sensor systems must utilize the minimal possible energy while operating over a wide range of operating scenarios [15] [16]. So there is need of key technologies required for low-energy distributed sensors. These include power aware computation/communication component technology, low-energy signaling and networking, system partitioning considering computation and communication trade-offs, and a power aware software infrastructure.

2.1.1 Power Aware Design

Sensor network technology has faced several challenges in hardware, communication protocol and application design in order to be a realized in practical applications [12]. Advances in MEMS (Micro Electro Mechanical Systems) technology, radio circuitry and DSPs, have help to overdraw these challenges in a more or less successful way. Some of those challenges are [13][14]:

• Low cost (hundreds to thousands nodes per network).

• Low power consumption (battery supplied lifetime of 3– 5 years).

• Network intelligence for data gathering (networks without human management).

• Resource limited nodes (energy power supply, computational capacity memory).

• Dynamical reaction to changing network conditions and topology modifications.

• Minimum impact (nodes have to be small and go unnoticed).

• Reliability (Fault tolerance thanks to high level redundancy).

• Measurement accuracy improvement (processed information from distributed sensors).

Unlike in other wireless networks, such as MANET, Bluetooth sensor networks can have around hundreds or even thousands of nodes, generally stationary, where QoS (Quality of Service) is not the first objective, but making sensors' life as longer as possible. In fact, in these kind of networks, network connectivity degradation is not dependent on the mobility of the nodes but on their battery lives.

2.1.2 Power aware sensor modes:

A power-aware sensor node [17] model essentially describes the power consumption in different levels of node sleep state. Every component in the node can have different power modes. It can be in active, idle, or sleep mode; the radio can be in transmit; receive, standby, or off mode. Each node sleep state corresponds to a particular combination of component power modes. In general, if there are N components labeled (1, 2, …, N) each with ki sleep states, the total number of node sleep states is Πki. Every component power mode has a latency overhead associated with switching to that mode. Therefore each node sleep mode is characterized by power consumption and latency overhead.

Table 2: Sleep states for the sensor node [17]

Sleep state	**Processor**	**Memory**	**Sensor, analog-digital converter**	**Radio**
$s0$	Active	Active	On	Tx, Rx
$s1$	Idle	Sleep	On	Rx
$s2$	Sleep	Sleep	On	Rx
$s3$	Sleep	Sleep	On	Off
s4	Sleep	Sleep	Off	Off

Tx=transmit, Rx=receive.

However, not all sleep states are useful. Table 2 enumerates the component power modes corresponding to five different useful sleep states for the sensor node. Each of these node sleep modes corresponds to an increasingly deeper sleep state and is therefore characterized by an increasing latency and decreasing power consumption. These sleep states are chosen based on actual working conditions of the sensor node; for example, it does not make sense to have memory active and everything else completely off. The design problem is to formulate a policy for transitioning between states based on observed events so as to maximize energy efficiency.

Sensor networks must support the operation of power saving modes for the sensor node. For example means of power conservation is to turn the transceiver off when it is not required. Although this power saving method seemingly provides significant energy gains, an important point that must not be overlooked is that sensor nodes communicate using short data packets. The shorter the packets, the more the dominance of startup energy. In fact, if we blindly turn the radio off during each idling slot, over a period of time it might end up expending more energy than if the radio had been left on. As a result, operation in a power-saving mode is energy-efficient only if the time spent in that mode is greater than a certain threshold. There can be a number of such useful modes of operation for the wireless sensor node, depending on the number of states of the microprocessor, memory, A/D converter, and transceiver. Each of these modes can be characterized by its power consumption and latency overhead, which is the transition power to and from that mode. A dynamic power management scheme for wireless sensor networks is used where five power-saving modes are proposed .The threshold time is found to depend on the transition times and the individual power consumption of the modes.

2.2 Routing in WSNs:

Routing[18][19] is an important aspect in any type of network in order to get data delivered to the right destination depending upon the various parameters in the network like end-to-end delay, throughput, density, state of the network(errors, link failures) etc. Routing in WSNs has much distinguishable features as compared to the other networks like mobile ad-hoc network or vehicular ad hoc network because of the following reasons:

- Transmission power, receiving power, processing capacity and storage are constraint factors to be considered when dealing with WSNs.

- There is a huge redundancy of data because most of the sensors capture same data, but in order to increase bandwidth utilization, power consumption, this redundancy needs to be removed. In case of large number of nodes, global identifier cannot be used.

  The various protocols can be classified as location-based, flat and hierarchical as shown in figure 2.1.

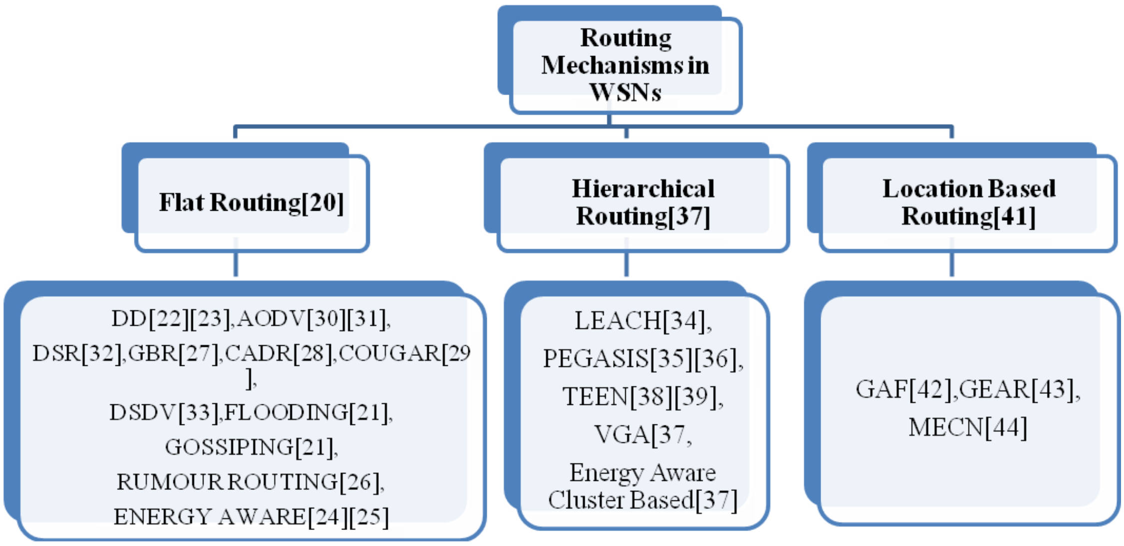


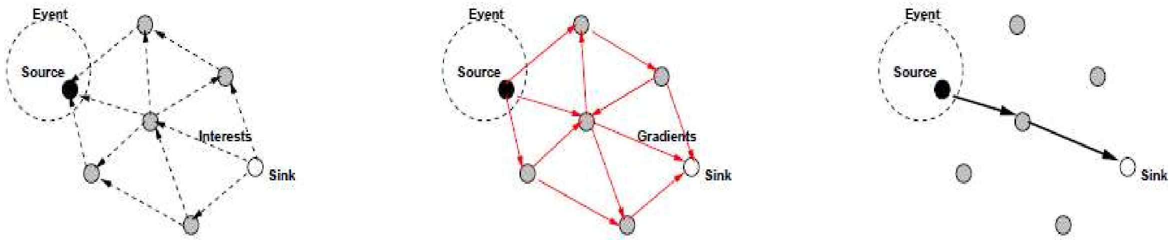
Fig 2.1: Classification of Routing Algorithms in WSNs

2.2.1 Flat Routing Protocols:

In flat routing protocols [20], sink is used to send queries to some selected regions and waits for data from sensors that are located in those selected regions. As queries are used for the requested data, attribute-based naming in order to specify the properties of data is necessary. There are some routing protocols also like Directed Diffusion, Flooding, MAODV, AODV, DSR, DSDV, Gradient Based Routing etc.

a. **Flooding and Gossiping:** In order to relay data in sensor networks without the need for any routing algorithms and topology maintenance there are two methods – Flooding and Gossiping [21]. A sensor node broadcast a data packet to all its neighbors and this process continues until destination is found and this technique is known as flooding. In gossiping packet is not sent to all neighboring nodes but only to selected random neighbors which selects another random neighbor and so on until this packet arrives at the destination.

b. Directed Diffusion: In this protocol, the idea is to diffuse data by using naming scheme for the data through sensor nodes. Directed diffusion [22] consists of several elements: interests, data messages, gradients, and reinforcements. An interest message is a query which specifies what a user wants to know. Each interest contains a description of a sensing task that is supported by a sensor network for acquiring data. Typically, data in sensor networks is the collected or processed information of a physical phenomenon. Such data can be an event which is a short description of the sensed phenomenon. In directed diffusion, data is named using attribute-value pairs. A sensing task is disseminated throughout the sensor network as an interest for named data. This dissemination sets up gradients within the network designed to "draw" events (i.e., data matching the interest). Specifically, a gradient is direction state created in each node that receives an interest. The gradient direction is set toward the neighboring node from which the interest is received [23]. Events start flowing towards the originators of interests along multiple gradient paths. The sensor network reinforces one, or a small number of these paths. Figure 2.2 illustrates these elements.



(a) Interest propagation (b) Initial gradients set up (c) Data delivery along reinforced path

Fig 2.2. A simplified Schematic of Directed Diffusion [23]

c. Energy-Aware Routing:

Shah and Rabaey[24][25] proposed to use set of sub-optimal paths occasionally in order to increase the lifetime of the network. Depending on the energy consumption of the path, these paths are chosen by means of probability functions. The main metric in this routing is Network survivability. This protocol has the following phases [25]:

- Setup phase.
- Data communication and route maintenance phase

d. Rumor Routing:

Another variation of Directed Diffusion is the rumor routing [26] .It is proposed for contexts in which geographic routing criteria are not applicable. The query is flooded in the entire network in Directed Diffusion when there is no geographic criterion to diffuse tasks. Rather than using flooding, rumor touting can be used when a little amount of data is requested.

e. Gradient-based Routing:

Schurgers proposed a slightly changed version of Directed Diffusion called Gradient based routing (GBR) [27]. In this routing scheme the idea is to maintain number of hops when the interest is diffused through the network. Node's height is minimum numbers of hops are discovered by each hop to sink. The gradient is the difference between node's height and that of its neighbor on that link. With the largest gradient a packet is forwarded on the link.

f. Constrained Anisotropic Diffusion Routing (CADR):

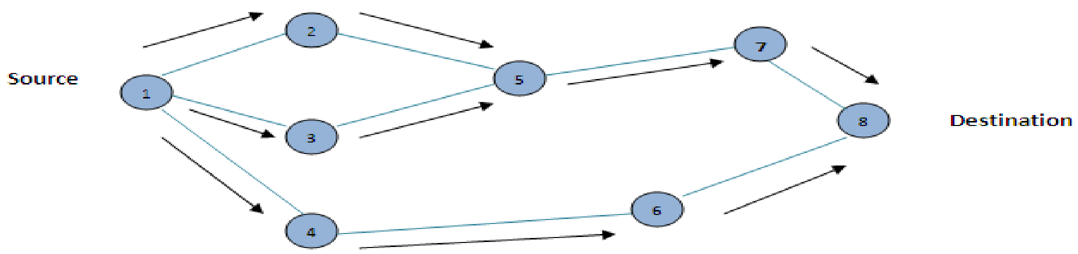
In order to maximize the energy gain and minimizing the bandwidth and latency, the idea is to query sensors and route data in network. Information-driven sensor querying (IDQS) and constrained anisotropic diffusion routing are the two proposed techniques [28]. The information/cost objective is evaluated by each node in CADR [28] and data based on local information/cost gradient is routed.

g. COUGAR:

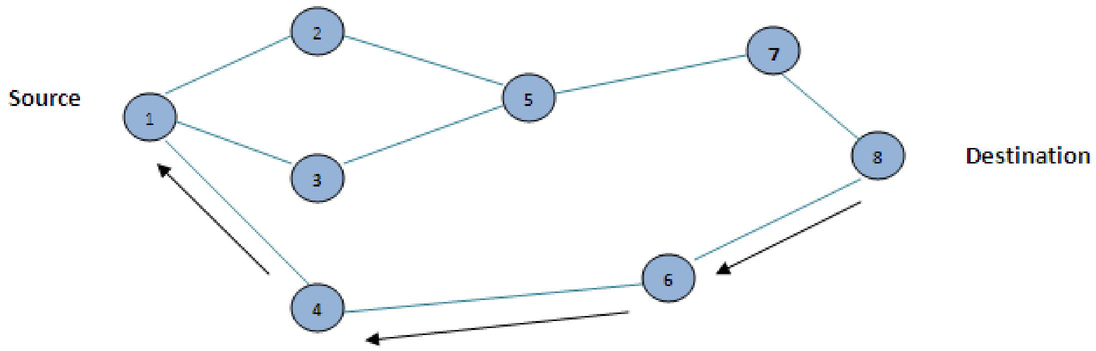
COUGAR [29] proposed architecture for sensor database systems in which a leader node is selected by other sensor nodes. Sensor nodes send their data to leader node which further sends data to sink and performs data aggregation. In this declarative queries are used in order to abstract query processing from the network layer functions in order to save energy by selection of relevant sensors etc.

h. Ad-Hoc On-Demand Distance Vector (AODV):

The Ad-Hoc On-Demand Distance Vector routing protocol is a reactive routing protocol. AODV protocol is a combination of Dynamic Source Routing (DSR) and DSDV protocol [30]. It is a distance vector routing protocol and is capable of both unicast and multicast routing [31]. It will maintain the routes only between the nodes which need to communicate. The routing information will be maintained as routing tables in each node. A routing table entry expires if it has not been used or reactivated for pre-specified expiration time. When a source node wants to send the packet to a destination node then the entries in the routing table will check whether there is a current route to the destination node or not, if there is a route then the packets will transmit to destination node in that path as shown in figure 2.3 [31]. If don't have any valid route, then the route discovery process will be initiated. For route discovery in AODV there are two types of packets - Routing Request (RREQ), Routing Reply (RREP) Packets. The RREQ packet containing the source node IP address, source node current sequence number, the destination node sequence number and broadcast ID. The whole process of RREP and RREQ is shown in figure2.3 a, b .The advantage of AODV is that it creates no extra traffic for communication along the existing link but requires more time to establish a connection. It is simple and doesn't require much memory or calculation



a) Distribution packet among neighbors



b) Reply to Demand

Fig 2.3 Route Discovery in AODV [31]

i. **Dynamic Source Routing protocol (DSR) :**

The Dynamic Source Routing protocol (DSR)[32] is a simple and efficient routing protocol designed specifically for use in multi-hop wireless ad hoc networks of mobile nodes. DSR, is a reactive routing protocol that uses source routing to send packets. It uses source routing which means that the source must know the complete hop sequence to the destination. DSR allows the network to be completely self-organizing and self-configuring, without the need for any existing network infrastructure or administration. Dynamic Source Routing, is composed of the two main mechanisms of "Route Discovery" and "Route Maintenance", which work together to allow nodes to discover and maintain routes to arbitrary destinations in the ad hoc network. The protocol allows multiple routes to any destination and allows each sender to select and control the routes used in routing its packets, for example, for use in load balancing or for increased robustness.

All aspects of the protocol operate entirely on demand, allowing the routing packet overhead of DSR to scale automatically to only what is needed to react to changes in the routes currently in use as shown in figure 2.4.

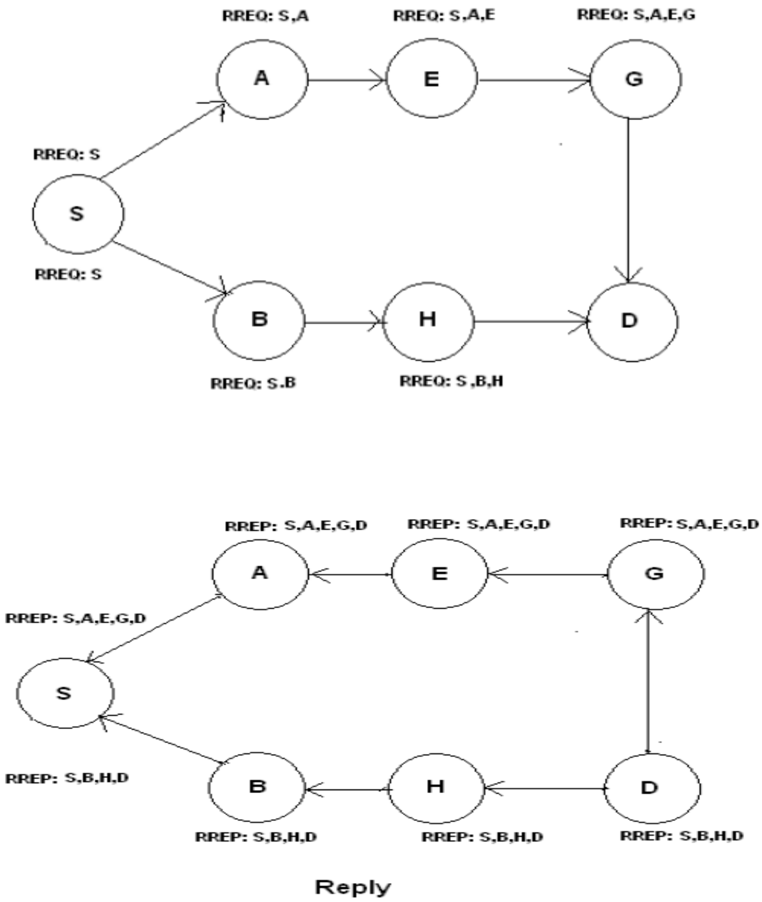


Fig 2.4 a) DSR Request, b) DSR Reply [32]

j. Destination-Sequenced Distance-Vector Routing (DSDV):

DSDV is a proactive routing protocol based on the classical distributed Bellman-Ford routing algorithm [33]. In DSDV messages are shared between the nodes (in the same transmission range. The nodes will continuously evaluate the routes to all reachable nodes and attempt to maintain consistent up-to date information. The routing table updates will sent periodically throughout the network thus the table will maintain its consistency but this will generate a lot of traffic in the network. Each node will maintain a routing table in which all of the possible destinations within the network and the number of hops to each destination are recorded [33]. Each entry in the routing table is marked with a sequence number that is assigned by the destination node; the sequence numbering system will avoid the formation of loops.

2.2.2 Hierarchical Routing Protocols:

Hierarchical routing performs energy-efficient routing in WSNs, and contributes to overall system scalability and lifetime. In a hierarchical architecture, sensors organize themselves into clusters and each cluster has a cluster head, i.e. sensor nodes form clusters where the low energy nodes are used to perform the sensing of the phenomenon. The less energy constrained nodes play the role of cluster-heads and process, aggregate and forward the information to a potential layer of clusters among themselves toward the base station.

a. **LEACH protocol**:

Heinzelman, introduced a hierarchical clustering algorithm for sensor networks, called Low Energy Adaptive Cluster Hierarchy – protocol (LEACH).It is an aggregation technique that combines the original data into a smaller size of data that carry only meaningful information of the individual sensors. In LEACH[34][7] the operation is divided into rounds, during each round a different set of nodes are cluster-heads (CH) as shown in figure 2.5. Nodes that have been cluster heads cannot become cluster heads again for P rounds. Thereafter, each node has a 1/p probability of becoming a cluster head in each round. At the end of each round, each node that is not a cluster head selects the closest cluster head and joins that cluster to transmit data. The cluster heads aggregate and compress the data and forward it to the base station, thus it extends the lifetime of major nodes. LEACH can be viewed as a hybrid approach using short and long range based data forwarding. The sensors within a cluster transmit their sensed data over short distances, whereas cluster heads communicate directly with sink. But this can be a problem so it is better to have multi-hop transmission. In this algorithm, the energy consumption will distribute almost uniformly among all nodes and the non-head nodes are turning off as much as possible. LEACH assumes that all nodes are in wireless transmission range of the base station which is not the case in many sensor deployments. In each round, LEACH has cluster heads comprising 5% of total nodes.. Figure 2.5 shows the communications in LEACH protocol

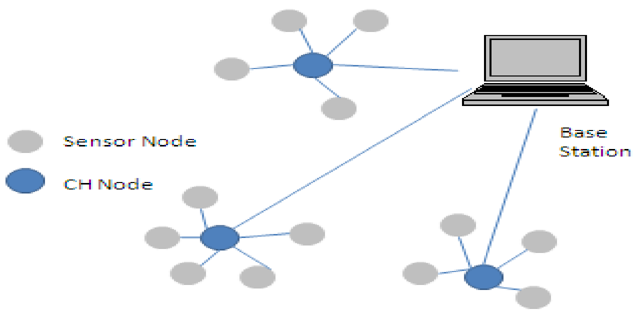


Fig 2.5 LEACH [34]

b. PEGASIS protocol:

Power- Efficient Gathering in Sensor Information Systems (PEGASIS)[35][36] is a near optimal chain-based protocol for extending the lifetime of network. It allows only cluster head to transmit their aggregated data to the sink in each round. A sensor has to transmit to its local neighbors in the data fusion phase instead of sending directly to its cluster head as in case of LEACH. it works by forming a chain first as shown in Figure 2.6.

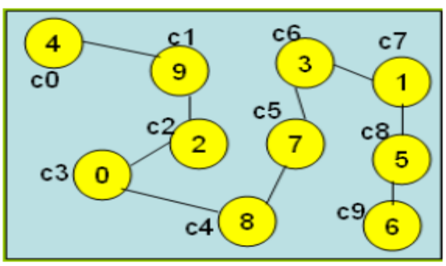


Fig 2.6: Chain formation in PEGASIS [35]

In PEGASIS, each node communicates only with the closest neighbor by adjusting its power signal to be only heard by this closest neighbor. Each Nodes uses signal strength to measure the distance to neighborhood nodes in order to locate the closest nodes. After chain Formation PEGASIS elects a leader from the chain in terms of residual energy every round to be the one who collects data from the neighbors to be transmitted to the base station. As a result, the average energy spent by each node per round is reduced. Unlike LEACH, PEGASIS avoids cluster formation and uses only one node in a chain to transmit to the Base station instead of multiple nodes. This approach reduces the overhead and lowers the bandwidth requirements from the BS. Figure 2.7 shows that only one cluster head leader node forward the data to the BS.

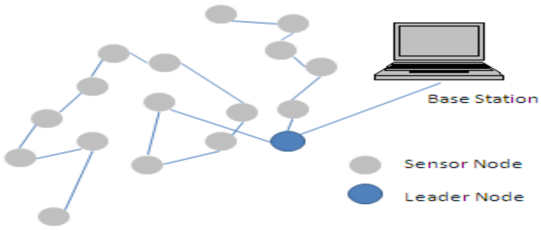


Fig 2.7 PEGASIS

A potential approach to reduce the delay required to deliver aggregated data to the sink is to use parallel data aggregation along the chain. A high degree of parallelism can be achieved if the sensor nodes are equipped with CDMA-capable transceivers. In this example it is assumed that all nodes have global knowledge of the network and employ a greedy algorithm to construct the chain. Furthermore, it is assumed that nodes take turns in transmitting to the base station such that node i mod N, where N represents the total number of nodes, is responsible for transmitting the aggregate data to the base station in round i. Based on this assignment in figure 2.8, node 3, in position 3 in the chain, is the leader in round 3. All nodes in an even position must send their data to their neighbor to the right. At the next level, node 3 remains in an odd position. Consequently, all nodes in an even position aggregate their data and transmit them to their right neighbors. At the third level, node 3 is no longer in an odd position. Node 7, the only node beside node 3 to rise to this level, aggregates its data and sends them to node 3. Node 3, in turn, aggregates the data received with its own data and sends them to the base station [36].

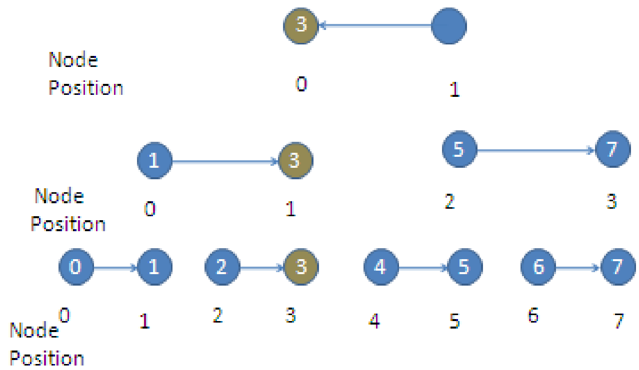


Fig 2.8: Chain based data gathering and aggregation scheme [35]

c. VGA Protocol:

Virtual Grid Architecture (VGA) is an energy-efficient routing paradigm proposed in [37]. The protocol utilizes data aggregation and in-network processing to maximize the network lifetime. Due to the node stationary and extremely low mobility in many applications in WSNs, a reasonable approach is to arrange nodes in a fixed topology. A GPS-free approach is used to build clusters that are fixed, equal, adjacent, and non-overlapping with symmetric shapes. In [37], square clusters were used to obtain a fixed rectilinear virtual topology. Inside each zone, a node is optimally selected to act as CH. Data aggregation is performed at two levels: local and then global. The set of CHs, also called Local Aggregators (LAs), perform local aggregation, while a subset of these LAs are used to perform global aggregation. However, the determination of an optimal selection of global aggregation points, called Master Aggregators (MAs), is NP-hard. Figure 2.9 illustrates an example of fixed zoning and the resulting virtual grid architecture (VGA) used to perform two level data aggregation. Note that the location of the base station is not necessarily at the extreme corner of the grid; rather it can be located at any arbitrary place.

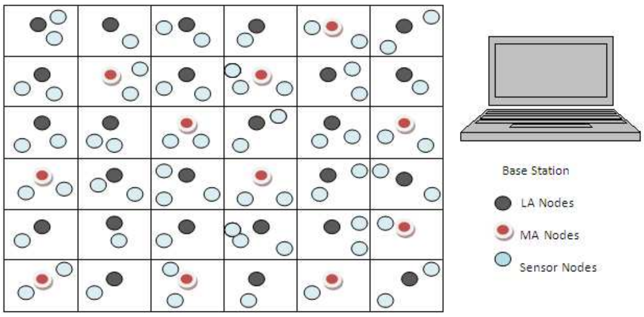


Fig 2.9: VGA [37]

d. Threshold sensitive Energy Efficient sensor Network protocol (TEEN):

TEEN [38] is a hierarchical protocol designed to be responsive to sudden changes in the sensed attributes such as temperature. Responsiveness is important for time-critical applications, in which the network operated in is active mode. TEEN pursues a hierarchical approach along with the use of a data-centric mechanism. The sensor network architecture is based on a hierarchical grouping where closer nodes form clusters and this process goes on the second level until base station (sink) is reached as depicted in Figure 2.10.After the clusters are formed, the cluster head broadcasts two thresholds to the nodes -Hard and Soft thresholds for sensed attributes. Hard threshold is the minimum possible value of an attribute to trigger a sensor node to switch on its transmitter and transmit to the cluster head. Thus, the hard threshold allows the nodes to transmit only when the sensed attribute is in the range of interest, thus reducing the number of transmissions significantly. Once a node senses a value at or beyond the hard threshold, it transmits data only when the values of that attribute changes by an amount equal to or greater than the soft threshold. As a consequence, soft threshold will further reduce the number of transmissions if there is little or no change in the value of sensed attribute. One can adjust both hard and soft threshold values in order to control the number of packet transmissions. However, TEEN is not good for applications where periodic reports are needed since the user may not get any data at all if the thresholds are not reached. The Adaptive Threshold sensitive Energy Efficient sensor Network protocol (APTEEN) [35] is an extension to TEEN and aims at both capturing periodic data collections and reacting to time critical events. When the base station forms the clusters, the cluster heads broadcast the attributes, the threshold values, and the transmission schedule to all nodes. Cluster heads also perform data aggregation in order to save energy. APTEEN supports three different query types: historical, to analyze past data values; one-time, to take a snapshot view of the network; and persistent to monitor an event for a period of time.

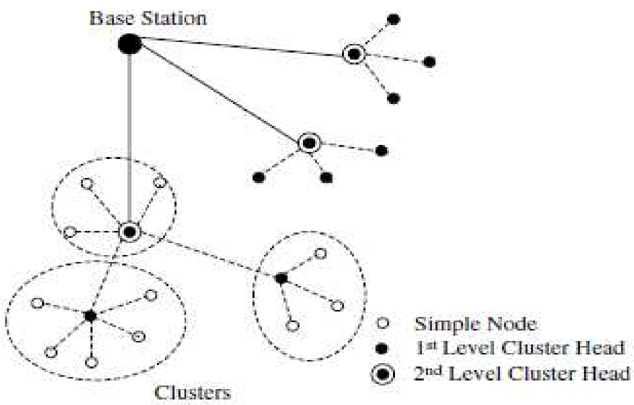


Fig 2.10: Hierarchical clustering in TEEN and APTEEN [38].

e. Energy-aware routing for cluster-based sensor networks[37] :

In this routing mechanism, clusters of sensor nodes are formed for network operations. Cluster heads act as gateways, whose main task is to maintain the states of sensors acts sets up multi-hop routes for collecting sensors data.

Gateways are less energy constrained than sensors and assumed to know the location of sensor nodes. A TDMA based MAC is used for nodes to send data to the gateway.

The gateway informs each node about slots in which it should listen to other nodes transmission and slots, which the node can use for its own transmission as shown in figure 2.11.

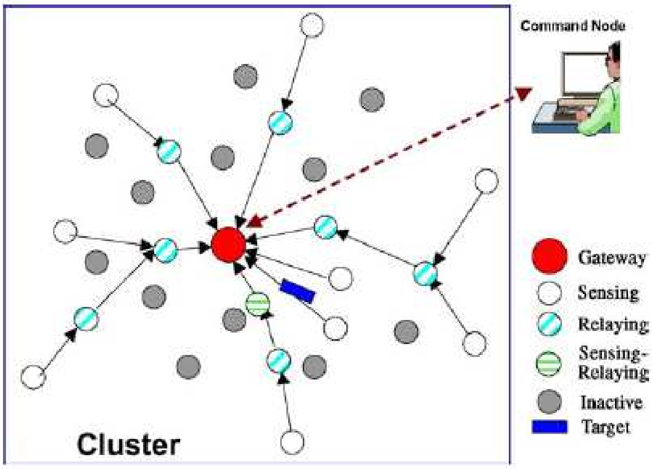


Fig 2.11 typical cluster in a sensor network [39]

The sink node communicates only with the gateways [39]. The sensor is assumed to be capable of operating in an active mode or a stand-by mode. The sensing and processing circuits can be powered on and off. In addition both the radio transmitter and receiver can be independently turned on and off and the transmission power can be programmed based on the required range. The sensor nodes in a cluster can be in one of four main states: sensing only, relaying only, sensing relaying, and inactive. In the sensing state, the node senses the environment and generates data at a constant rate. In the relaying state, communications circuitry is on to relay the data from other active nodes. When a node is both sensing and relaying messages from other nodes, it is considered in the sensing-relaying state. Otherwise, the node is considered inactive and can turn off its sensing and communication circuitry.

2.2.3 Comparison among Hierarchical routing and Flat routing:

In Table 3 [40],both Hierarchical routing and Flat routing has been compared on the basis of various parameters like data aggregation, scheduling, routing strategy, overhead, latency, energy dissipation etc.

Table 3: Comparison Among Hierarchical routing and flat routing [40]

Hierarchical routing	**Flat routing**
Reservation-based scheduling	Contention-based scheduling
Reduced duty cycle due to periodic sleeping	Variable duty cycle by controlling sleep time of nodes
Data aggregation by cluster head	Node on multi hop path aggregates incoming data from neighbors
Simple but non-optimal routing	Routing can be made optimal but with an added complexity.
Requires global and local synchronization	Links formed on the °y without synchronization
Overhead of cluster formation throughout the network	Routes formed only in regions that have data for transmission
Lower latency as multiple hops network formed by Cluster heads always available	Latency in waking up intermediate nodes and setting up the multipath
Energy dissipation is uniform	Energy dissipation depends on traffic patterns
Energy dissipation cannot be controlled	Energy dissipation adapts to traffic pattern
Fair channel allocation	Fairness not guaranteed

2.2.4 Location Based Routing Protocol:

In Location Based Routing Protocol [41], energy consumption is based on the distance between the nodes among which data has to be transmitted. In WSNs, there is no global addressing scheme, so in order to get response or to send request to a specific region, we need allocation information of the sensor nodes. Rather than sending the request to the whole network we can send the request to the specific region only in which we are interested. In this way energy consumption will be reduced, which is the main requirement of WSNs. Some protocols use Global Power Positioning System in order to reduce the energy consumption. For example: MECN (minimum energy communication network) uses this system in order to save the energy.

- **Geographic Adaptive Fidelity(GAF):**

  GAF [42] is a location-based protocol, it may also be considered as a hierarchical protocol, where the clusters are based on geographic location. GAF conserves energy by turning off unnecessary nodes in the network without affecting the level of routing fidelity. The whole area is divided into fixed zones and virtual grids. Nodes associated with the same point on the grid are considered equivalent in terms of the cost of packet routing .Inside each zone, nodes collaborate with other nodes and play different roles acc. to the conditions. For example, nodes will elect one sensor node to stay awake for a certain period of time and then they go to sleep. This node is responsible for monitoring and reporting data to the BS on behalf of the nodes in the zone. Nodes change states from sleeping to active in turn so that the load is balanced. There are three states defined in GAF. These states are discovery, for determining the neighbors in the grid, active reflecting participation in routing and sleep when the radio is turned off. The state transitions in GAF are depicted in Figure 2.12.

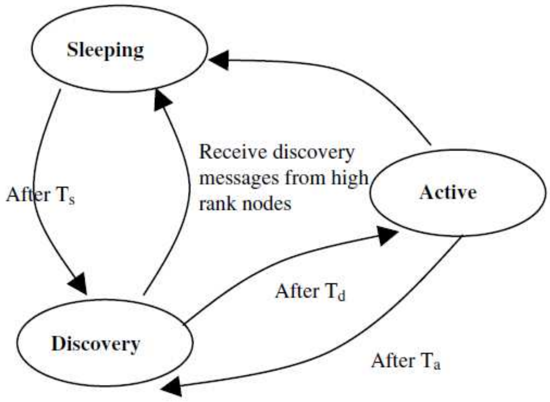


Fig 2.12: State transitions in GAF [42]

- **Geographic and energy-aware routing (GEAR):**

  In GEAR [43], there are two types of costs - an estimated cost and a learning cost of reaching the destination through its neighbors. The estimated cost is a combination of residual energy and distance to destination. The learned cost is a refinement of the estimated cost that accounts for routing around holes in the network. A hole occurs when a node does not have any closer neighbor to the target region than itself. If there are no holes, the estimated cost is equal to the learned cost. The learned cost is propagated one hop back every time a packet reaches the destination so that route setup for next packet will be adjust The idea is to restrict the number of interests in Directed Diffusion by only considering a certain region rather than sending the interests to the whole network. GEAR compliments Directed Diffusion in this way and thus conserves more energy as shown in figure 2.13.

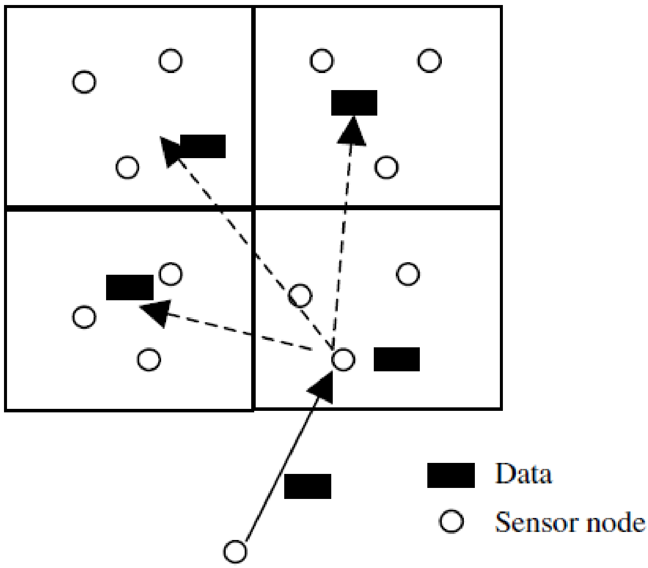


Fig 2.13: Recursive geographic forwarding in GEAR [43]

There are two phases in the algorithm:

1. **Forwarding packets towards the target region:** Upon receiving a packet, a node checks its neighbors to see if there is one neighbor, which is closer to the target region than itself. If there is more than one, the nearest neighbor to the target region is selected as the next hop. If they are all farther than the node itself, this means there is a hole. In this case, one of the neighbors is picked to forward the packet based on the learning cost function.

2. **Forwarding the packets within the region:** If the packet has reached the region, it can be diffused in that region by either recursive geographic forwarding or restricted flooding. Restricted flooding is good when the sensors are not densely deployed. In high-density networks, recursive geographic flooding is more energy efficient than restricted flooding. In that case, the region is divided into four sub regions and four copies of the packet are created. This splitting and forwarding process continues until the regions with only one node are left.

- **Minimum energy communication network (MECN):**

  MECN [44] sets up and maintains a minimum energy network for wireless networks by utilizing low power GPS. It is best applicable to sensor networks, which are not mobile. MECN assumes master sites as the information sink. MECN identifies a relay region for every node. The relay region consists of nodes in a surrounding area where transmitting through those nodes is more energy efficient than direct transmission. The enclosure of a node i is then created by taking the union of all relay regions that node i can reach. The main idea of MECN is to find a sub-network, which will have less number of nodes and require less power for transmission between any two particular nodes. In this way, global minimum power paths are found without considering all the nodes in the network. This is performed using a localized search for each node considering its relay region. The protocol has two phases:

  1. It takes the positions of a two-dimensional plane and constructs a enclosure graph, which consists of all the enclosures of each transmit node in the graph. This construction requires local computations in the nodes. The enclosed graph contains globally optimal links in terms of energy consumption.

  2. Finds optimal links on the enclosure graph. It uses distributed Bellman–Ford shortest path algorithm with power consumption as the cost metric. In case of mobility the position coordinates are updated using GPS. MECN is self-reconfiguring and thus can dynamically adapt to nodes failure or the deployment of new sensors. Between two successive wake-ups of the nodes, each node can execute the first phase of the algorithm and the minimum cost links are updated by considering leaving or newly joining nodes

2.2.5 Comparison among Flat, Hierarchical and location based Routing Protocols:

In Table 4,various routing mechanisms like flat routing, Hierarchical and Location Based routing has been compared on the basis of various parameters such as scalability, lifetime, data diffusion, power required.

Table 4: Comparison Among Flat, Hierarchical and location based Routing Protocols [45]

Parameters	**Flat Routing**	**Hierarchical Routing**	**Location-based Routing**
Scalability	Limited	Good	No
Lifetime	Long	Long	Long
Data Diffusion	No	Yes	No
Power Required	Limited	High	Limited

Chapter 3
Problem Statement

Due to the large number of sensor nodes that may be deployed and the due to harsh hostile environment in which they are employed, replacing the battery is not an option. Sensor systems must utilize the minimal possible energy while operating over a wide range of operating scenarios. So, there is need of key technologies required for low-energy distributed sensors.

Some of those challenges in area of energy Consumption are [13]:

- Low power consumption
- Network intelligence for data gathering (networks without human management)
- Resource limited nodes (energy power supply, computational capacity and memory).
- Dynamical reaction to changing network conditions and topology modifications.

Therefore, there is a need for evaluating various routing mechanism based on certain energy parameters and this task is very challenging as it includes energy parameters in the various routing protocols along with the following attributes:

1. Defining new parameters in the routing protocols so that various energy parameters can be traced.
2. Tracing transmitting energy, receiving energy, idle energy, Initial Energy parameters along with other parameters such as packet type, acknowledgment etc.
3. Extracting required fields from trace file for energy analysis as the format of energy trace files is different from general trace format.
4. To analyze that how the packets are lost when nodes are dead due to low energy in various routing mechanism.
5. To analyze the packet Delivery Ratio for various routing protocols as it is the measure of efficiency.

Chapter 4
Simulation Environment

Wireless sensor networks have the potential to become significant subsystems of engineering applications. Before relegating important and safety-critical tasks to such subsystems, it is necessary to understand the dynamic behavior of these subsystems in simulation environments. There is a need to develop simulation platforms that are useful to explore both the networking issues and the distributed computing aspects of wireless sensor networks. Current efforts to simulate wireless sensor networks largely focus on the networking issues. There are number of simulators available, such as-OMNET++, J-Sim, Glomosim, NS2, JiST/SWANS, but according to our needs and versatility available, we have chosen following platforms for our simulations:

4.1 NS2:

NS (version 2) [46] is an object-oriented, discrete event driven network simulator. It was developed at UC Berkeley & written in C++ and OTcl. NS is primarily useful for simulating local and wide area networks. It has an OTcl interpreter as a frontend. The simulator supports a class hierarchy in C++ (also called the compiled hierarchy), and a similar class hierarchy within the OTcl interpreter (also called the interpreted hierarchy). The root of this hierarchy is the class TclObject. The interpreted class hierarchy is automatically established through methods defined in the class TclClass.

4.1.1 Concept Overview

NS2 uses two languages because simulator has two different kinds of tasks it does. On one hand, detailed simulations of protocols require a systems programming language which can efficiently manipulate bytes, packet headers, and implement algorithms that run over large data sets. For these tasks run-time speed is important and turn-around time (run simulation, find bug, fix bug, recompile, re-run) is less important [47].

On the other hand, an iteration time (change the model and re-run) is more important. Since configuration runs once (at the beginning of the simulation), run-time of this part of the task is less important. NS2 meets both of these needs with two languages, C++ and OTcl. C++ is fast to run but slower to change, making it suitable for detailed protocol implementation. OTcl runs much slower but can be changed very quickly (and interactively), making it ideal for simulation configuration.

We use OTcl:

• For configuration, setup, and “one-time” stuff

• You can do what you want by manipulating existing C++ objects and use C++:

• Processing each packet of a flow

• To change the behavior of an existing C++ class inmany ways

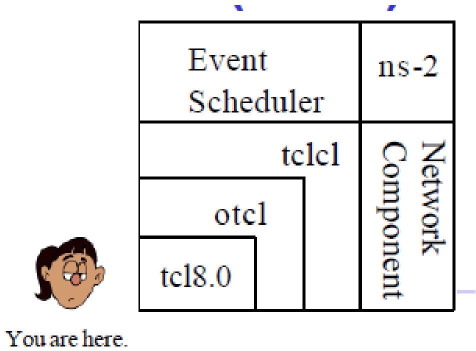


Fig 4.1: Structure of NS2 [23]

Figure 4.1 shows the general structure of NS. A general user can be thought of standing at the left bottom corner, designing and running simulations in Tcl using the simulator objects in the OTcl library. The event schedulers and most of the network components are implemented in C++ and available to OTcl through an OTcl linkage that is implemented using tclcl. The whole thing together makes NS, which is an OO extended Tcl interpreter with network simulator libraries.

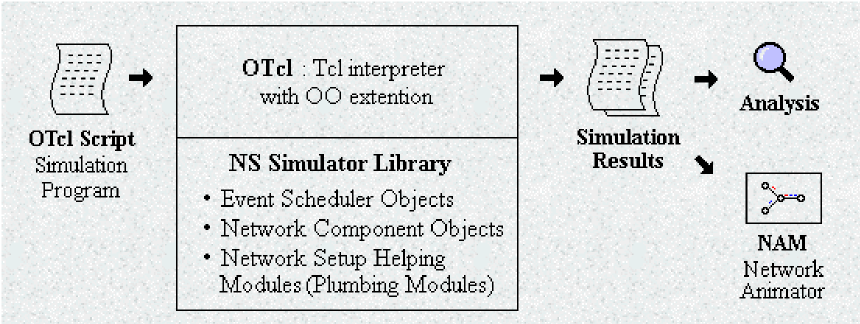


Fig 4.2: Simplified Users' View of NS [23]

As shown in figure 4.2, when a simulation is finished, NS produces one or more text-based output files that contain detailed simulation data, if specified to do so in the input Tcl (or more specifically, OTcl) script. The data can be used for simulation analysis or as an input to a graphical simulation display tool called Network Animator (NAM). NAM has a graphical user interface and also has a display speed controller. Furthermore, it can graphically present information such as throughput and number of packet drops at each link, although the graphical information cannot be used for accurate simulation analysis.

4.1.2 Installing ns2 in Ubuntu:

In this proposed work, ns-allinone-2.34.tar.gz (NS2 package) has been installed and all the simulations has been carried out on this platform.

Steps for installing ns2:

1. Install necessary tools for building:
   sudo apt-get install build-essential autoconf automake libxmu-dev gcc-4.3

2. Change ns-allinone-2.34/otcl-1.13/Makefile.in. Find the line that is:
   CC= @CC@
   and change it to:
   CC= gcc-4.3

3. Suppose that there is "ns-allinone-2.34.tar.gz" in current directory (e.g. "~/" "/home/srandhawa"). All steps are done in Terminal, so open a terminal in Accessories/Terminal.
   To extract the file:
   tar xvfz ./ns-allinone-2.34.tar.gz

4. Change the directory within terminal as following way:

   $ cd /home/randhawa $ cd /home/randhawa/ns-allinone-2.34

5. Download & install some packages from repository.

   $ sudo apt-get install build-essential autoconf automake libxmu-dev

6. Install the NS2

   $ cd ns-allinone-2.34

   $ ./install

7. Edit some paths

   $ gedit ~/.bashrc

8. Put these lines on that file in the last.

   %export LD-LIRARY-PATH=/home/randhawa/NS2/ ns-allinone-2.34/bin

   %export LD-LIRARY-PATH=/home/randhawa/NS2/ ns-allinone-2.34/lib

   %export TCL-LIRARY=/home/randhawa/NS2/ ns-allinone-2.34/tcl8.4.11/library

9. Validate it.

   $ cd ns-2.33$ ./validate

10. Let it take effect immediately

    $ source ~/.bashrc

11. Try to run it as following way, by just typing ns with terminal

    $ ns

12. If the installation is successful, then there will be % at the command prompt as shown in figure 4.3. Type following command to exit

    % exit

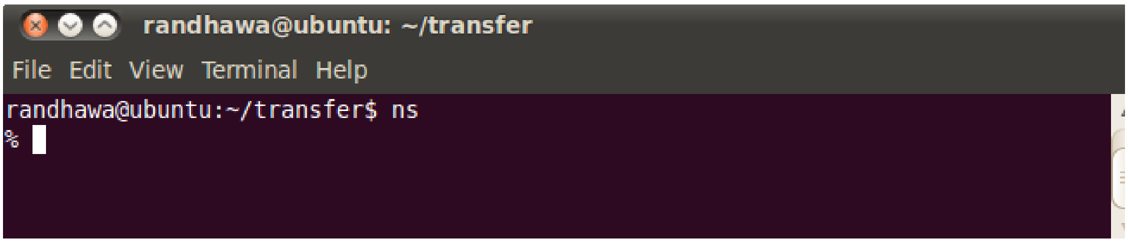


Fig 4.3 Checking installation of ns2

4.1.3 The Network Animation (NAM)

The network animator (NAM) [47] began in 1990 as a simple tool for animating packet trace data. This trace data is typically derived as output from a network simulator like ns or from real network measurements.

The Trace graph

Trace graph supports the following ns-2 trace file formats; wired, satellite, wireless (old and new trace), wired-cum-wireless. Trace file loading stage is divided into 4 stages; automatic trace file format recognition, trace file parsing to extract necessary simulation data which is saved to a temporary file, trace files can contain much more data than is needed by the system, so unnecessary information is omitted to speed up trace file loading, temporary file loading, constants calculations (packets types, packets sizes, flows IDs, trace levels, number of nodes, simulation time) – in order to speed up data processing. Wireless and wired-cum-wireless trace files are parsed and saved in Trace graph format and displayed as a graph as shown in figure 4.4.

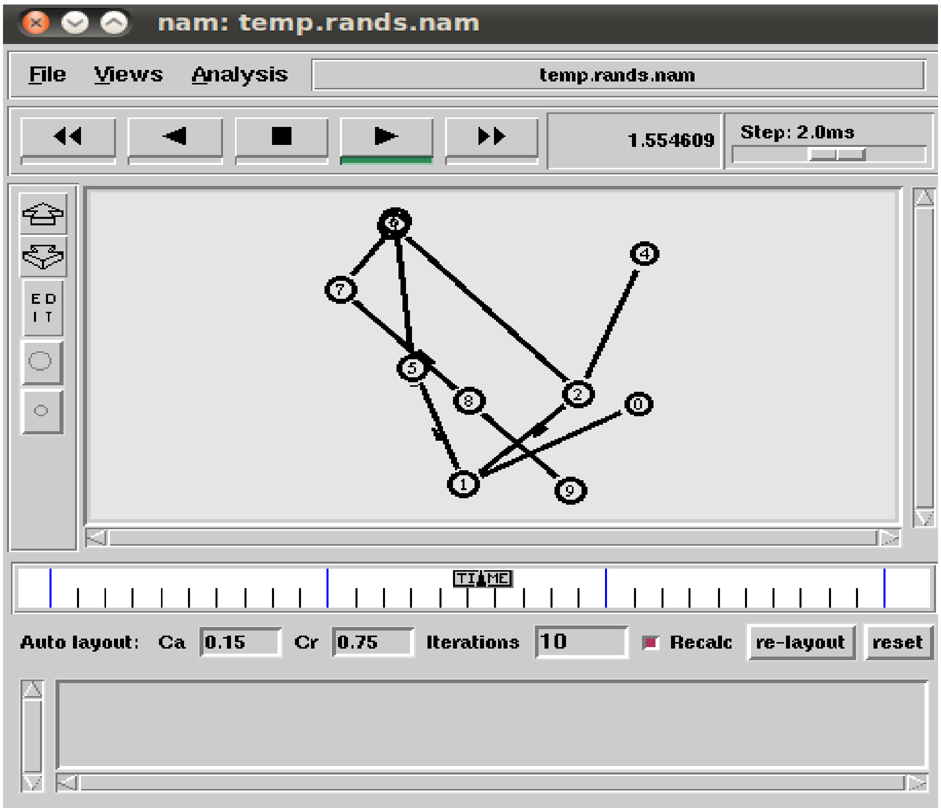


Fig 4.4 Network Animation

4.1.4 Tcl: The User Language:

Tool Command Language (Tcl) [48] is an interpreted script language developed by Dr. John Ousterout at the University of California, Berkeley, and now developed and maintained by Scripts. Tool Command Language is a reusable command language interpreter. It is implemented as an embeddable C library It provide all features such as Procedures, variables, associative arrays, lists, expressions, loops, exceptions, introspection, etc. Applications can add new commands to the interpreter.

4.1.5 Packet:

A NS packet is composed of a stack of headers, and an optional data space (see Figure 4.5).a\A packet header format is initialized when a Simulator object is created, where a stack of all registered (or possibly useable) headers, such as the common header that is commonly used by any objects as needed, IP header, TCP header, RTP header (UDP uses

RTP header) and trace header, is defined, and the offset of each header in the stack is recorded. What this means is that whether or not a specific header is used, a stack composed of all registered headers is created when a packet is allocated by an agent, and a network object can access any header in the stack of a packet it processes using the corresponding offset value [47].

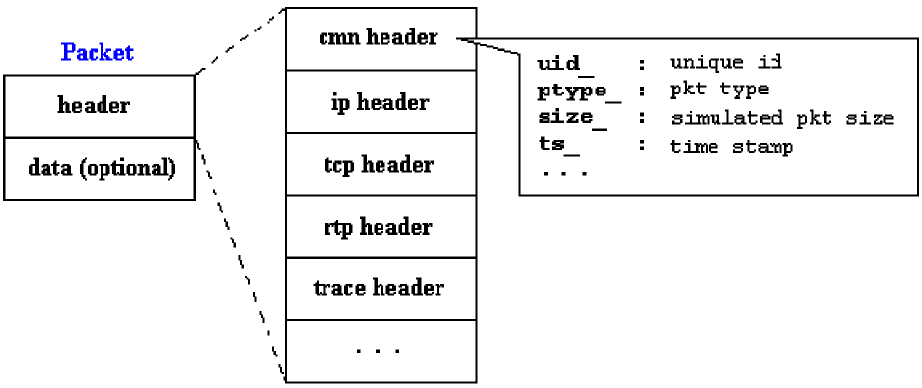


Fig 4.5 Packet Format [46]

4.1.6. Trace File Format:

The trace file is an ASCII code files and the trace is organized in fields as shown in figure 4.6

Fig 4.6: Trace File Format (.tr)

Event	Time	Node	Trace Generated By	Flag	Pkt ID	Pkt Type	Pkt Size	Pkt Type ID	Flag	Receiver	TTL	Ack

The first field is the event type and given by one of four available symbols r, +, - and d which correspond respectively to receive, enqueued, dequequed and dropped. The second field is telling the time which the event occurs. The third and fourth fields are the input and output node of the link at which the events takes place. The fifth field is for flag which is usually empty. The Sixth field is for Packet id which is being transmitted .The seventh is the packet type such as continuous bit rate (CBR) or transmission control protocol (TCP). The eighth is the size of the packet and the next field is some kind of flags. The ninth one is for packet type id which is different for different type of files. Then there are some other fields like time to live and acknowledgement number

Depending on the user's purpose for an OTcl simulation script, simulation results are stored as trace files, which can be loaded for analysis by an external application:

1. A NAM trace file (file.nam) for use with the Network Animator Tool
2. A Trace file (file.tr) for use with XGraph [46] or extract using AWK Scripts and Trace Graph can also be used

We have used NAM and Trace Graph is used to analyze the trace file.

4.1.7 Energy Model API

The energy model is used through the node-config API [49]. The following parameters are generally there.

```
$ns_ node-config -adhocRouting DumbAgent \
                 -llType $opt(ll) \
                 -macType Mac/SMAC \
                 -ifqType $opt(ifq) \
                 -ifqLen $opt(ifqlen) \
                 -antType $opt(ant) \
                 -propType $opt(prop) \
                 -phyType $opt(netif) \
                 -channelType $opt(chan) \
                 -topoInstance $topo_ \
                 -agentTrace ON \
                 -routerTrace ON \
                 -macTrace ON \
                 -energyModel $opt(energymodel) \
                 -idlePower 712e-6 \
                 -rxPower 35.28e-3 \
```

-txPower 31.32e-3 \

-sleepPower 144e-9\

-initialEnergy $opt(initialenergy)

The following parameters are newly added:

-sleepPower: power consumption (Watt) in sleep state

-energyModel $opt(energymodel):specify the energy model used

-idlePower

-rxPower

-txPower

-sleepPower

4.1.8 Energy Analysis through Trace Files

We have added energy breakdown in each state in the traces to support detailed energy analysis [49]. In addition to the total energy, now users will be able to see the energy consumption in different states at a given time. Following is an example from a trace file on energy as shown in figure 4.7.

[energy 979.917000 ei 20.074 es 0.000 et 0.003 er 0.006]

The meaning of each item is as follows:

energy: total remaining energy

ei: energy consumption in IDLE state

es: energy consumption in SLEEP state

et: energy consumed in transmitting packets

er: energy consumed in receiving

Fig 4.7 Contents of Trace File(.tr)

Using Trace Files:

NS2 simulation allows us to create trace files to log the results of our simulation. Depending on the size of simulation, these trace files can be very large, which makes it impossible to extract any meaningful information out of them manually. Hence we resort to using text processing tools.

4.2 AWK programming:

AWK programming [50] is like any other high level programming. Since there are different traces formats, the same AWK code will not work for all trace files, however the basic concept is the same. AWK identifies the strings separated by tabs and spaces on a single line in the text as a single unit and accordingly designates those numbers.

AWK code has main 3 parts- BEGIN, END and Central Processing part as shown below

```
BEGIN { #starting block to define the variables
……………………
……………………
}
{
#central processing block where all the calculations occur
…………………… ……………………
}
END { #finishing block to terminate the code and display results
……………………
……………………
}
```

4.2.1 Executing an AWK script:

We just want to display residual energy with time. Then this can be extracted from trace file(.tr) using an AWK script [50] as shown in the figure 4.8.

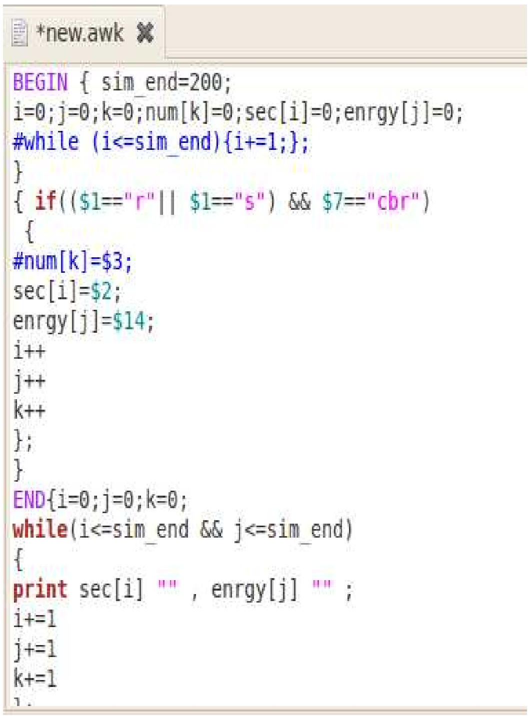


Fig 4.8 AWK Script for extracting parameters from Trace File

It is better to have the script and the trace files in the same subfolder. The script can be run as shown in Figure 4.9,where p.awk is the AWK script dsr.tr is the trace file of DSR protocol and all the required fields are stored in ds.xgr file.

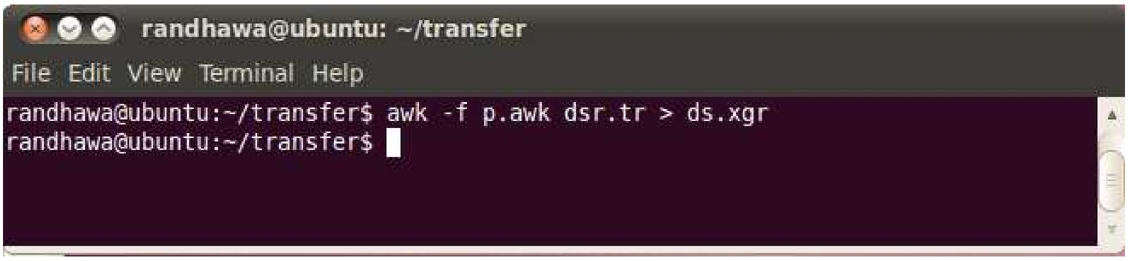


Fig 4.9 Executing AWK script on trace file

Now ds.xgr will contain only two fields a shown in figure 4.10:

- Time
- Residual energy

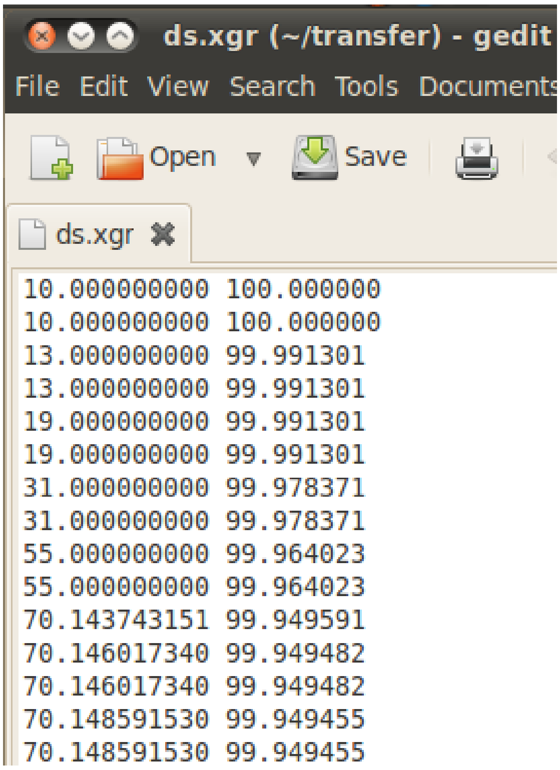


Fig 4.10 After Extracting Required fields from trace file

Once the required fields are extracted from the trace file using AWK script, then we can plot those values using xgraph utility as shown in figure 4.11

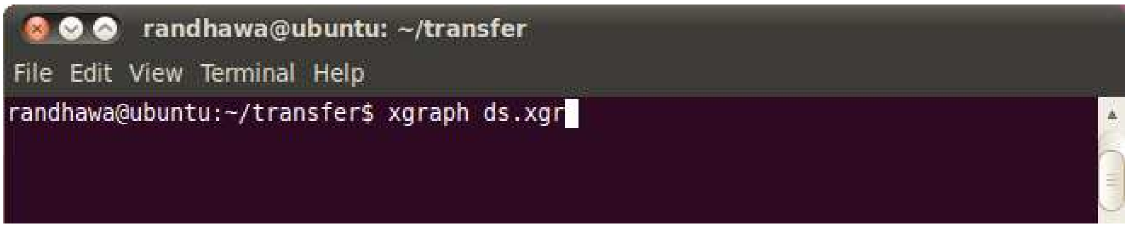


Fig4.11 Executing xgraph on extracted fields

4.3 XGRAPH[46]:

One part of the ns-allinone package is 'xgraph', a plotting program which can be used to create graphic representations of simulation results. The xgraph program draws a graph on an X display given data read from either data files or from standard input if no files are specified. It can display up to 64 independent data sets using different colors and/or line styles for each set. It annotates the graph with a title, axis labels, grid lines or tick marks, grid labels, and a legend. Once the window has been opened, all of the data sets will be displayed graphically with a legend in the upper right corner of the screen. Xgraph presents three control buttons in the upper left corner of each window: *Hardcopy, Close and About* as shown in the figure 4.12.

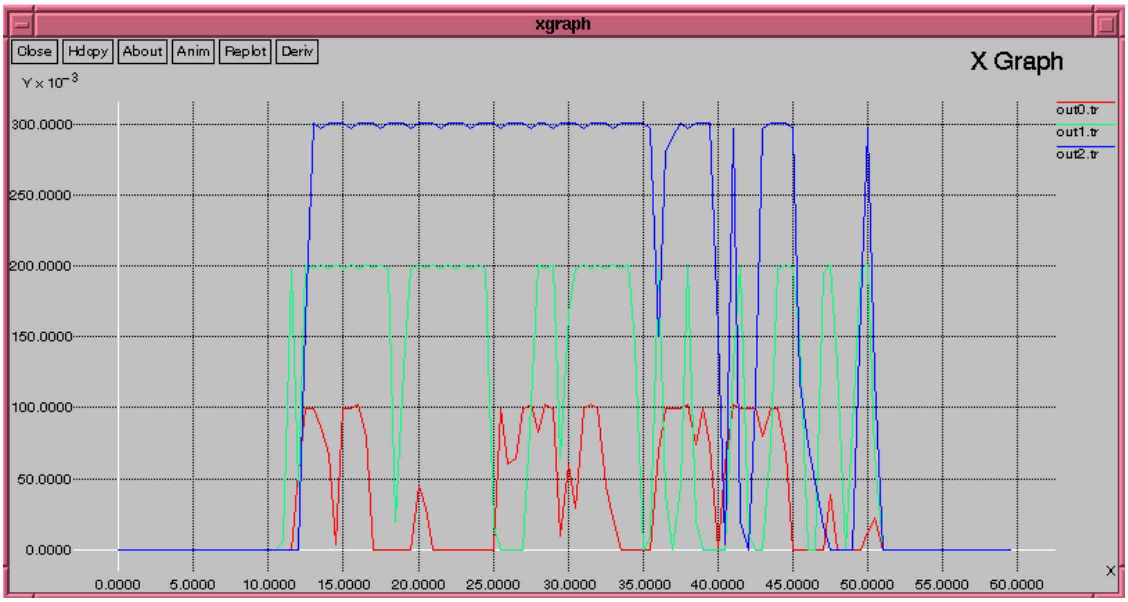


Fig 4.12: Plotting graph using xgraph [46]

Chapter 5

Results and Evaluation

Experiment has been carried out to analyze the energy efficiency among different classifications of the routing protocols:

- **All Flat routing Protocols** such as Ad-Hoc On-Demand Distance Vector (AODV), Multipath Ad-Hoc On-Demand Distance Vector (MAODV), Dynamic Source Routing (DSR), Directed Diffusion, Flooding, Destination-Sequenced Distance-Vector (DSDV).
- Different classes of routing protocols have also been compared such as, Flat RoutingProtocols(Flooding,MAODV,AODV,DirectedDiffusion,DSR,DSDV), Hierarchical Routing (LEACH, PEGASIS, VGA, and APTEEN/TEEN) and Location Based Routing Protocols (GAF, GEAR, and MECN).

The following parameters were made use of for the analysis of various routing protocols:

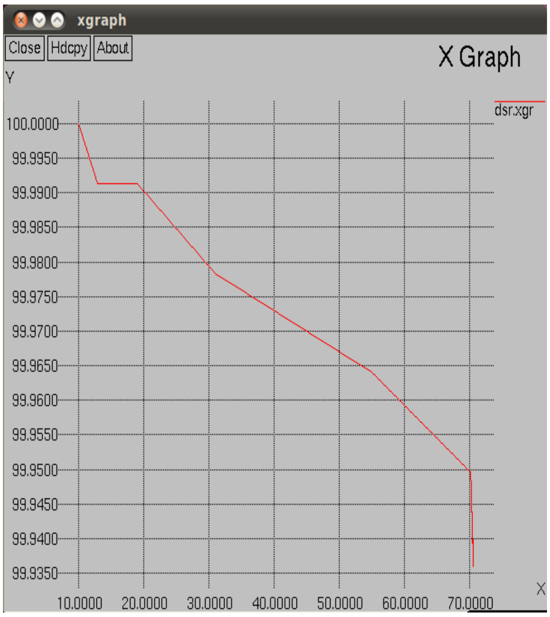
- Residual Energy Vs Time
- Dropped Packet Vs Time Due to decrease in energy
- Packet Deliver Fraction (ratio of packet sent/packet received)

5.1 Residual Energy Vs Time:

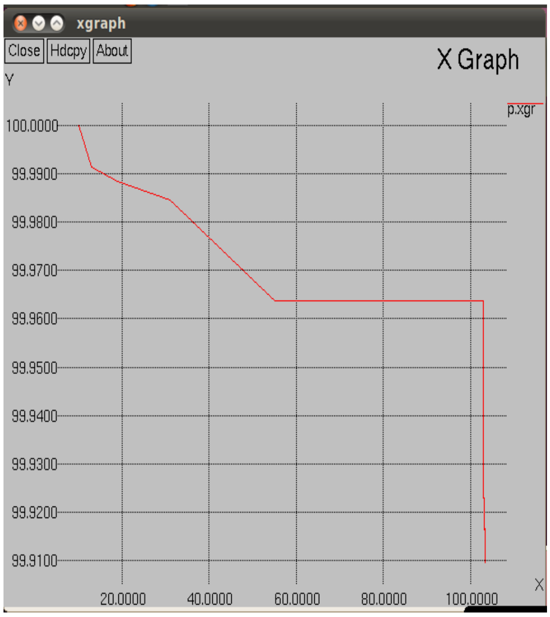
After transmitting or receiving data to /from neighbors the some energy of sensor nodes gets dissipated. So along with the passage of time the remaining energy or residual energy of sensor nodes decreases. So based on this parameter ,several routing protocols are being compared such as flat routing protocols(AODV, MAODV, DSR, DSDV, Directed Diffusion, Flooding), hierarchical routing protocols(LEACH,VGA,MECN, PEGASIS) and location based routing protocols(GAF,GEAR etc).

5.1.1 Flat Routing Protocols

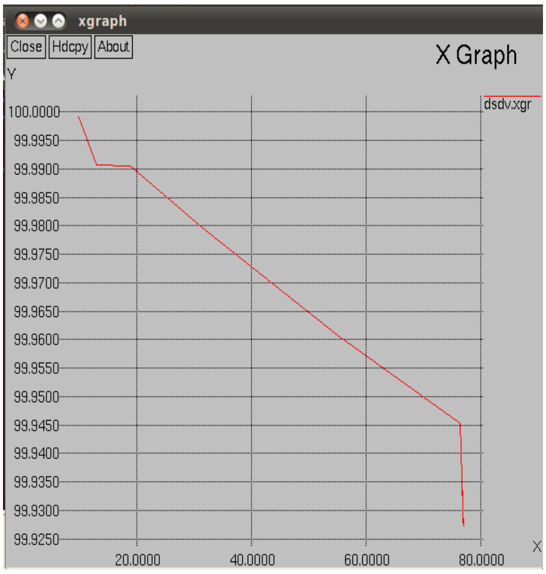
Various protocols such as Destination-Sequenced Distance-Vector, Dynamic Source Routing, Ad-Hoc On-Demand Distance Vector, Multipath Ad-Hoc On-Demand Distance Vector, Directed Diffusion and Flooding have been simulated and the results are being compared for Residual energy Vs time.



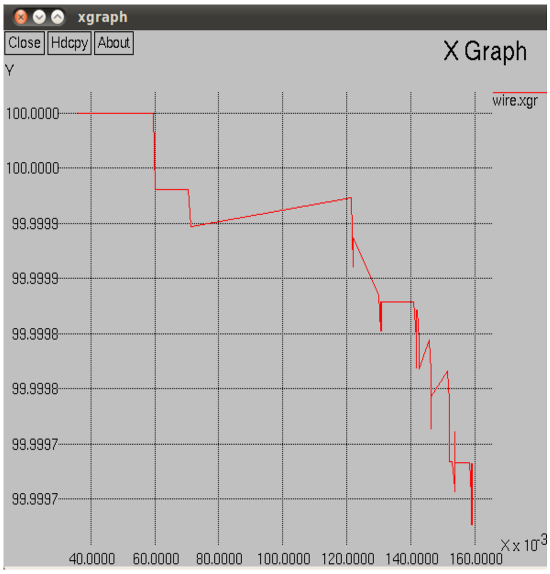
a) DSR



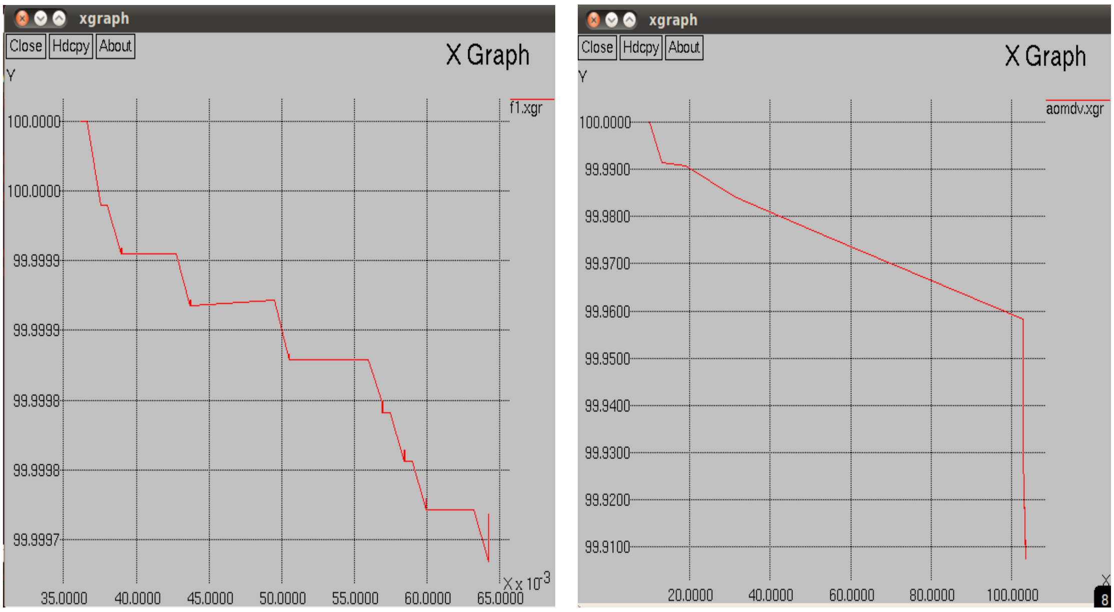
b) AODV



c) DSDV



d) Directed Diffusion



e) FLOODING f) MAODV

Fig 5.1: Residual Energy Vs Time in WSNs using various Flat Routing Protocols
a)DSR , b)AODV, c)DSDV, d)Directed Diffusion , e)Flooding, f)MAODV

Analysis:

As shown in the graphs figure 5.1 (a), (c) plotted, the energy curve of DSR, DSDV resembles. Both show a steady energy for a short period of time, but then energy level exhausted. This is because of the REQUEST/REPLY mechanism in both protocols which require a large no. of data packets to be transmitted.

There is a difference between MAODV (multipath AODV) and AODV (single path).as depicted in figure 5.1 (b), (f) in case of MAODV, energy exhausted rapidly as compare to the AODV. In case of AODV, energy is stable for some period of time (as shown by steady line in graph b), but this is not the case with MAODV. Based on energy consumption DSDV performs low in state compared with AODV, Since AODV requires less energy for transmission of packets

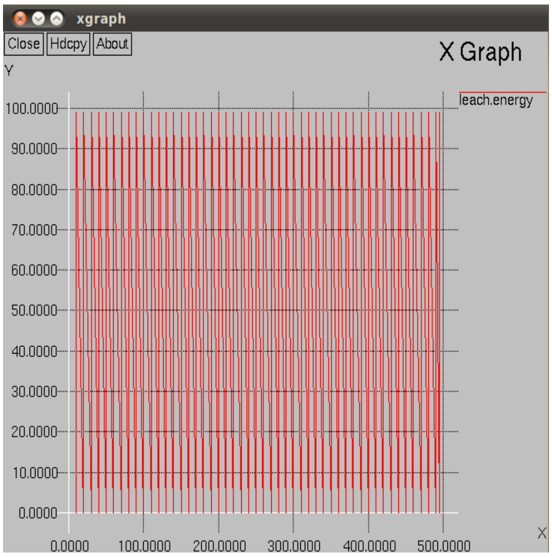
In case of directed diffusion as shown in figure 5.1 (d), there is a zigzag curve of energy. In this, there is also a loss of energy with time, but not continuously as with other routing protocols

5.1.2 Hierarchical Routing

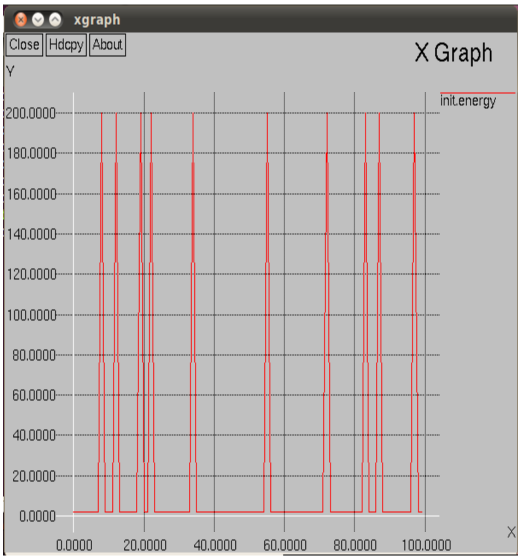
There are number of hierarchical routing protocols like Low Energy Adaptive Cluster Hierarchy (LEACH), Power- Efficient Gathering in Sensor Information Systems (PEGASIS), Threshold sensitive Energy Efficient sensor Network protocol (TEEN), Virtual Grid Architecture (VGA), Energy Aware Routing Protocol, but we have simulated LEACH protocol.

Analysis:

As we can see from figure 5.2 (a) initial energy in LEACH is high for all the nodes in the network. But LEACH works on the principle of clustering. So after some time the energy of cluster heads will be very low because all data has to be transferred via Cluster heads .Therefore energy level is down in figure 5.2 (b)



a) Initial Energy In LEACH



b) Energy of Cluster Heads

Fig 5.2: Residual Energy Vs Time in wireless Sensor Networks using Hierarchical Routing Protocols a) Initial Energy In LEACH, b) Energy of Cluster Heads

5.1.3 Location Based Routing:

In location based routing protocols there are number of routing protocol like Geographic Adaptive Fidelity (GAF), Geographic and energy-aware routing (GEAR), Minimum energy communication network (MECN). But here we are doing analysis of GAF. Other routing protocols can't be implemented because these require special Power GPS in order to simulate which is not supported by software emulation part of the available simulators

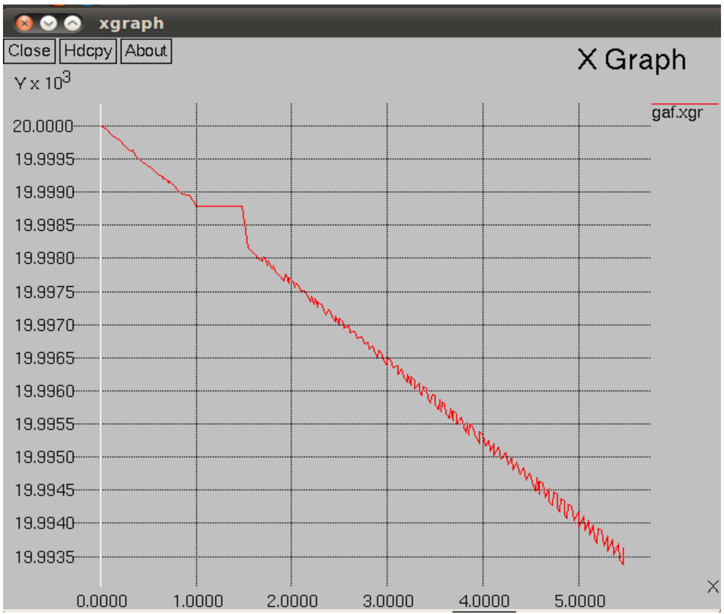


Fig 5.3: Residual Energy Vs Time in WSNs using GAF (Location Based Routing)

Analysis:

As we can analyze from the figure 5.3, in GAF there is stability in the energy for a very short time period, otherwise the residual energy get exhausted constantly. There are some spikes in the curve because of nodes which fall asleep for some duration. Still there is a huge loss of energy because more energy will be wasted when transceiver is switched rapidly from sleep to active state.

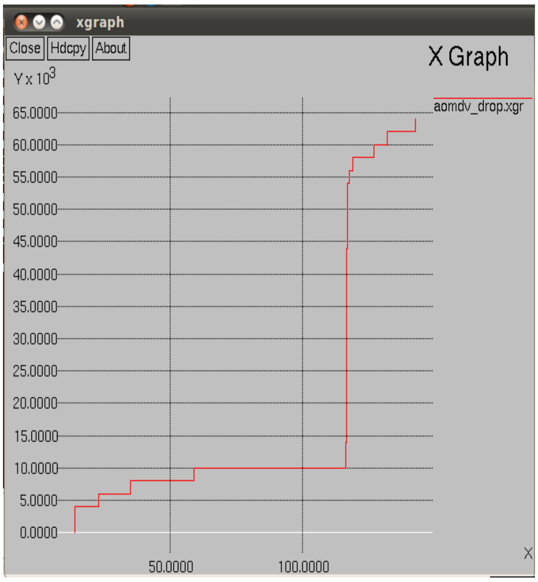
5.2 Dropped Packet Vs Time Due to decrease in energy:

As energy level decreases, destination nodes become unreachable or die, due to which some packets will be dropped. As the time passes, energy decreases and no. of dropped packets will be increased depending upon the routing algorithms such as flat routing-

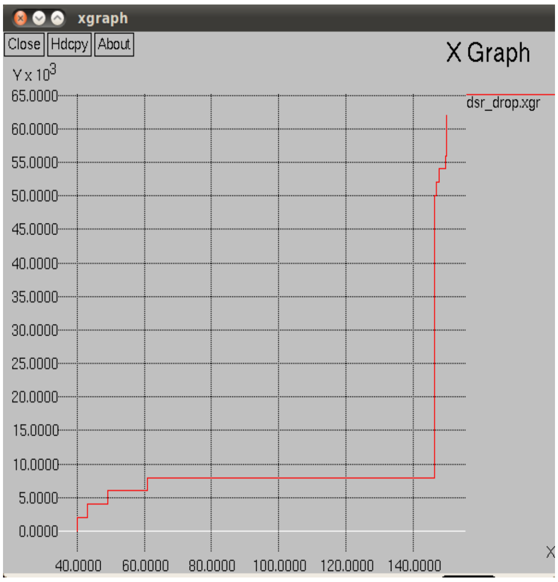
AODV,MAODV, Flooding, DSR, DSDV, Directed Diffusion and Hierarchical Routing-LEACH and Location Based Routing - GAF .

5.2.1 Flat Routing:

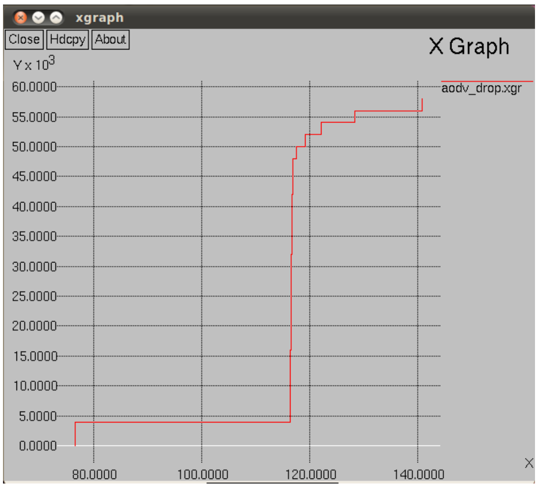
Various protocols such as Destination-Sequenced Distance-Vector, Dynamic Source Routing, Ad-Hoc On-Demand Distance Vector, Multipath Ad-Hoc On-Demand Distance Vector, Directed Diffusion and Flooding have been simulated and the results are being compared for Dropped Packet Vs time.



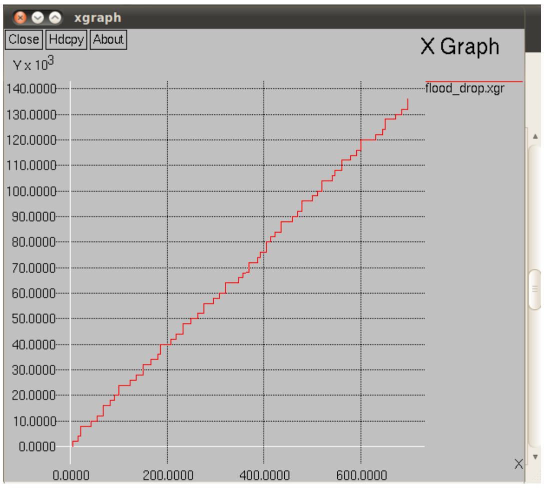
a) MAODV



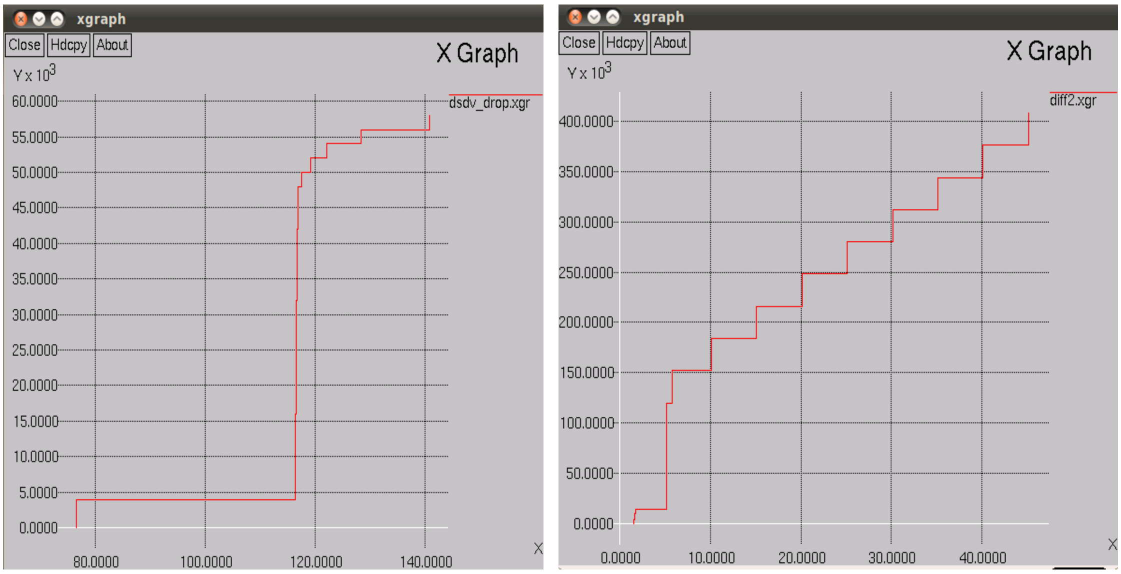
b) DSR



c) AODV



d) Flooding



e) DSDV f) Directed Diffusion

Fig 5.4 : Dropped Vs Time in WSNs using various Flat Routing Protocols a)MAODV b)DSR , c) AODV, d)Flooding , e) DSDV, f) Directed Diffusion

Analysis:

As depicted from fig 5.4 (a) and fig 5.4 (e), no. of dropped packet of MAODV and DSDV is same .in the initial phase some packets are dropped. Then after some time, no. of packets dropped increases due to lack of energy.DSR performs well as compare to other flat routing protocols as in DSR, packets are dropped only in the beginning and then this quantity remains same for most of the time as in figure 5.4(b). And then there is increase in number of packets dropped.

In Flooding, number of packets dropped with respect to time increases very rapidly ,because in flooding packet is broadcasted to all neighbors as shown in figure 5.4 (d) ,so energy diminishes very rapidly and some nodes become dead very soon. Directed Diffusion, there are phases internally, in one phase the no. of packets dropped are constant and in second phase, this number increases refer figure 5.4 (f), but not rapidly as in Flooding.

5.2.2 Hierarchical Routing:

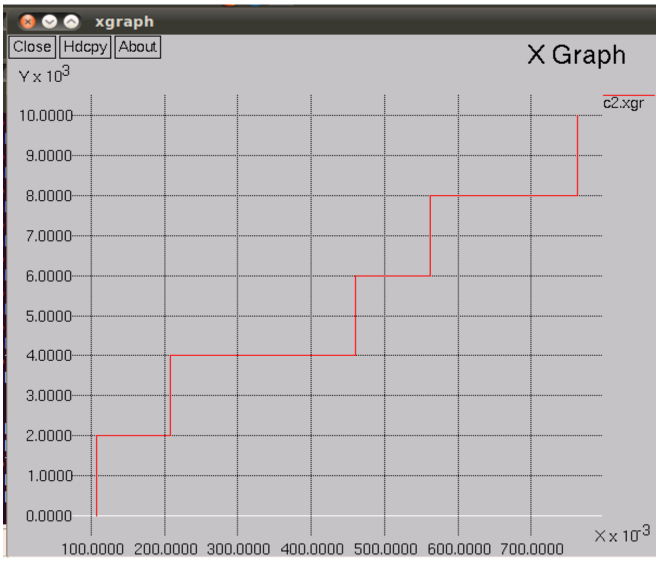


Fig 5.5 Dropped Vs Time in WSNs using Hierarchical Routing Protocol LEACH

Analysis: In Hierarchical Routing, clusters are formed and via data aggregation, data is send to the cluster heads. After some time, energy of cluster heads exhausted. In figure 5.5 there is steady state in which no. of packets dropped remains same, after some time this quantity increases because of cluster heads decreasing energy level.

5.2.3 Location Based Routing

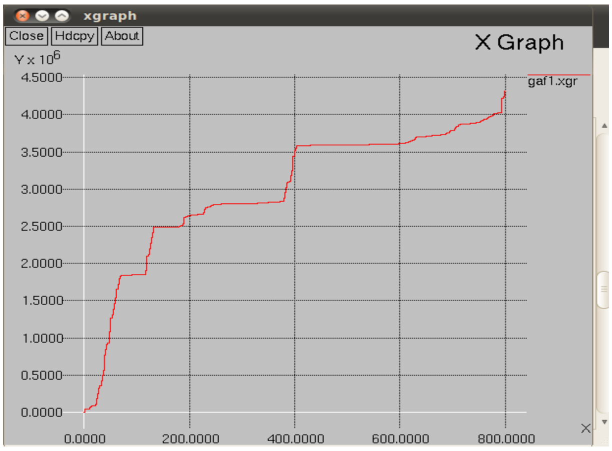


Fig 5.6 Dropped Packet Vs Time as the energy Diminishes in GAF (Location Based Routing)

Analysis:

In Location Based Routing, no. of packet dropped increases at irregular time intervals but not instantly as in case of flat routing protocols like Flooding as shown in figure5.6. There are little variations in packets dropped with respect to time and performs better than Hierarchical Routing.

5.3 Packet Delivery Fraction (ratio of packet sent/packet received):

This is the ratio of the data packets delivered to the destination to those generated by the traffic source as depicted in table 5.

Table 5: Packet Delivery Fraction of various routing protocols in WSNs

Type of Routing	Routing Protocol	No. of packets send	No. of packets Received	PDF
Flat Routing	AOMDV	2180	2149	0.9858
Flat Routing	DSR	7505	7475	0.9960
Flat Routing	DSDV	6474	6444	0.9954
Flat Routing	AODV	6474	6444	0.9954
Flat Routing	Directed Diffusion	5800	5001	0.8621
Flat Routing	Flooding	6800	6292	1.0001
Location Based Routing	GAF	2518	1800	1.3990
Hierarchical Routing	LEACH	6600	5911	1.1106

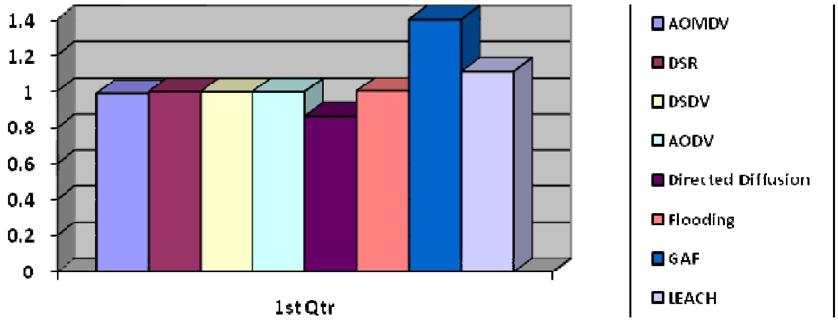


Fig 5.7: Packet delivery Fraction of various routing Protocols (MA0DV, DSR, DSDV, AODV, Directed Diffusion, Flooding, GAF, LEACH)

Analysis:

As depicted in the bar chart in figure 5.7 in terms of Packet Delivery Fraction, Location Based Routing performs better as compare to hierarchical and flat routing and also hierarchical routing performs better than flat routing protocols such as MAODV, AODV, DSR, DSDV, Directed Diffusion, and Flooding as shown in Figure 5.7.

In Flat Routing Protocols, Directed Diffusion lacks in terms of Packet Delivery Fraction as compare to the others.

Chapter 6

Conclusion and Future Scope

WSNs are different kind of networks having their importance in certain areas such as Environment Monitoring, Military Applications, and Health care applications, Industrial Process Control, Home Intelligence, Security and Surveillance etc. A routing Protocol is used to decide on the best suitable route to be considered for sending data to the sink from a sensor node. One of the major concerns is to send this data on a route which consumes less power. The power is a scarce commodity in WSNs because when we deploy them in a hostile environment, it is not possible to give them power supply or to get them recharge. So there is need of key technologies required for low-energy distributed sensors.

An analysis about the different flat routing Protocols such as AODV, MAODV, DSDV, Flooding, Directed Diffusion is presented and a comparison has also been carried out among the various routing mechanism such as flat routing (AODV,DSR,DD,FLOODING,DSDV,MAODV) and also with other routing mechanism such as Hierarchal routing(LEACH,PEGASIS,VGA) and Location based Routing mechanism such as (GAF,GEAR,MECN) etc.

In terms of packet delivery fraction, Location Based Routing Protocols performs better than hierarchical Routing Protocols and Flat Routing. In terms of residual energy and dropped packet parameters Hierarchical Routing performs better than flat routing and Location Based Routing.

In future, it would be interesting to analyze the behavior of these routes for greater number of nodes as the present study is limited to 10 or 15 number of nodes. Further we can also study the impact of change in the simulation area for these protocols. The work presented is based upon simulation; we can analyze the performance of these WSN protocols on a test bed.

References

[1] D.Culler ,D. Estrin and M.Srivastava, “Overview of Sensor Networks”, IEEE Computer Society , vol. 37 ,Issue No. 8, pp 41-49, August 2004.

[2] I.F.Akyilidiz, W.Su, Y.Sankarsubramaniam and E.Cayirci. “A survey on Sensor Computer Networks’, IEEE Communication magazine pp 102-114,August 2002.

[3] Zone.ni.com Assessed on 1.11.2011.

[4] Intel Motes and Wireless Sensor Networks, http://www.techresearch.intel.com/Downloads/SNOverviewCD.pdf Accessed on 10.18.2011.

[5] J.Hill, R.Szewcky, A.Woo, D.Culler, S.Hollar, and K.Pister, “System Architecture directions for networked sensors”, in Proceedings of 9th International Conference on Architectural Support for Programming Languages and Operating Systems ,Cambridge ,MA ,Nov. 2000 ,pp 93-104

[6] J.N. Al-Karaki and A.E. Kamal, “Routing techniques in Wireless Sensor Networks :A Survey”,IEEE Wireless Communication, vol .11 no.6 Dec.2004 ,pp 6-28.

[7]L.B.Ruiz,J.M.Nogueria and A.A.F.Loureira, “Sensor network management”,SMART DUST:Sensor Network Applications ,Architecture and Design, CRC Press ,Boca Raton ,FL,2006.

[8] http://alkautsarpens.wordpress.com/wsn/ accessed on 1.10.2011

[9] J. M. Kahn, R. H. Katz, and K. S. J. Pister, “Next Century Challenges: Mobile Networking for Smart Dust,” Proc. ACM MobiCom ’99, Washington, DC, 1999.

[10] B. Peters, ‘‘Sensing Without wires: Wireless Sensing Solves Many Problems, But Introduces a Few of Its Own,’’ Machine Design, Penton Media, Cleveland,OH,http://www.machinedesign.com/ASP/viewSelectedArticle.asp?strArticleId ¼57795&str- &strSite¼MDSite&Screen ¼ & CURRENTISSUE&CatID¼3.

[11] A.A. Abbasi and M.Younis , “A Survey on clustering Algorithms for Wireless Sensor Networks”, Computer Communications, vol. 30 ,no.s 14-15,Oct. 2007,pp. 2826-2841.

[12] J.N Al-karaki et al., “Data Aggregation in Wireless Sensor Networks – Exact and Approximate Algorithms,” Proc. IEEE Wks. High Perf. Switching and Routing 2004, Phoenix, AZ, Apr. 18-21, 2004.

13]J. M. Rabaey et al., “PicoRadio Supports Ad Hoc Ultra- Low Power Wireless Networking,” IEEE Comp. Mag., 2000.

[14] V. Rodoplu and T. H. Meng, “Minimum Energy Mobile Wireless Networks,” IEEE JSAC, vol. 17, no. 8, Aug.1999.

[15] C. Shen, C. Srisathapornphat, and C. Jaikaeo, “Sensor Information Networking Architecture and Applications,” IEEE Pers. Commun., Aug. 2001, pp. 52–59.

[16]Ying Liang and Yongxin Feng, "An Energy-Aware Routing Algorithm for Heterogeneous Wireless Sensor Networks", Ninth International Conference on Hybrid Intelligent Systems, 2009, vol. 2, 2010.

[17]Sinha and A. Chandrakasan, “Dynamic Power Management in Wireless Sensor Networks,” IEEE Design Test Comp., Mar./Apr. 2001.

[18]Parul Kansal. P, Kansal. D and Balodi, A.(August 2010). Comparison of Various Routing Protocol in Wireless Sensor Network. International Journal of Computer Applications (0975 – 8887) Volume 5– No.11

[19] J. N. Al-Karaki and A. E. Kamal, “Routing Techniques in Wireless Sensor Networks-A Survey,” IEEE Wireless Communication, Vol: 11, Dec. 2004, pp. 6-28.

[20] J. Kulik, W. R. Heinzelman, and H. Balakrishnan, "Negotiation-based protocols for disseminating information in wireless sensor networks," Wireless Networks, Volume: 8, pp. 169-185, 2002.

[21] S. Hedetniemi, A. Liestman, A survey of gossiping and broadcasting in communication networks, Networks 18 (4) (1988) 319–349

[22] B. Krishnamachari, D. Estrin, S. Wicker, Modeling data centric routing in wireless sensor networks, in: Proceedings of IEEE INFOCOM, New York, June 2002.

[23] C. Intanagonwiwat, R. Govindan, D. Estrin, Directed diffusion: a scalable and robust communication paradigm for sensor networks, in: Proceedings of the 6th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom_00), Boston, MA, August 2000

[24] M. Youssef, M. Younis, K. Arisha, A constrained shortestpath energy-aware routing algorithm for wireless sensor networks, in: Proceedings of the IEEE Wireless Commu-348 K. Akkaya, M. Younis / Ad Hoc Networks 3 (2005) 325–349 nication and Networks Conference (WCNC2002), Orlando,FL, March 2002.

[25] M. Younis, M. Youssef, K. Arisha, Energy-aware routing in cluster-based sensor networks, in: Proceedings of the10th IEEE/ACM International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS2002), Fort Worth, TX,October 2002.

[26] D. Braginsky, D. Estrin, Rumor routing algorithm for sensor networks, in: Proceedings of the First Workshop on Sensor Networks and Applications (WSNA), Atlanta, GA, October 2002.

[27] C. Schurgers, M.B. Srivastava, Energy efficient routing in wireless sensor networks, in: The MILCOM Proceedings on Communications for Network-Centric Operations: Creating the Information Force, McLean, VA, 2001.

[28] M. Chu, H. Haussecker, F. Zhao, Scalable information driven sensor querying and routing for ad hoc heterogeneous sensor networks, The International Journal of High Performance Computing Applications 16 (3) (2002) 293–299.

[29] Y. Yao, J. Gehrke, The cougar approach to in-network query processing in sensor networks, in: SIGMOD Record, September 2002.

[30] Charles E. Perkings, Elizabeth M. Belding-Royer, Samir R.Das, Ad Hoc On-Demand Distance Vector (AODV) Routing, http://www.ietf.org/internet-drafts/draft-ietfmanet-aodv-13.txt, IETF Internet draft, Feb 2003.

[31] C.Perkins and E.Royer, "Ad Hoc On-Demand Distance Vector Routing", Internet Draft, MANET working group, draft-left-manetaodv-05.txt, March 2000

[32]David B. Johnson and David A. Maltz. "Dynamic source routing in ad hoc wireless networks", Mobile Computing, Kluwer Academic Publishers. 1996 pp.153–181, 1996.

[33] C.E. Perkins & P. Bhagwat, "Highly Dynamic Destination Sequence-Vector Routing (DSDV) for Mobile Computers", Computer Communication Review, vol. 24, no.4, 1994, pp. 234-244.

[34] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan,“Energy-Efficient Communication Protocol for Wireless Microsensor Networks,” IEEE Proc. Hawaii Int’l. Conf. Sys. Sci., Jan. 2000, pp. 1–10.

[35]S. Lindsay and C. Raghavendra, “PEGASIS: Power-Efficient Gathering in Sensor Information Systems”, international Conf. on Communications, 2001.

[36]Indu Shukla, Natarajan Meghanathan, Jackson State University, Jackson MS, USA “Impact of leader selection strategies on the PEGASIS Data Gathering Protocol for WSN”.

[37] Laiali Almazaydeh, Eman Abdelfattah , Manal Al- Bzoor, and Amer Al- Rahayfeh Computer Science and Engineering Department “Performance Evaluation of Routing Protocols in WSN”.

[38] Manjeshwar, E.; Agrawal, D.P. TEEN: A Routing Protocol for Enhanced Efficiency in Wireless Sensor Networks. In Proceedings of the 15th International Parallel and Distributed Processing Symposium (IPDPS), San Francisco, CA, USA, April, 2001; pp. 2009–2015

[39] C. Mallanda, A. Suri, V. Kunchakarra, S.S. Iyengar*, R. Kannan* and A. Durresi , Sensor Network Research Group, Department of Computer Science, Louisiana State University, Baton Rouge, LA “ Simulating Wireless Sensor Networks”

[40] Jamal N. Al-Karaki Ahmed E. Kamal ,Iowa State University, Ames, Iowa “Routing techniques in wireless sensor networks :A Survey”, vol no 7 ,pp-24,25,2007

[41]Kemal Akkaya and Mohammed Younis (May,2005.). “A survey on routing protocols for wireless sensor networks and Ad hoc networks” [online], pp325 -349.

[42] Y. Xu, J. Heidemann, D. Estrin, Geography-informed energy conservation for ad hoc routing, in: Proceedings of the 7th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom_01), Rome, Italy, July 2001.

[43]Y. Yu, D. Estrin, R. Govindan, Geographical and energy aware routing: a recursive data dissemination protocol for wireless sensor networks, UCLA Computer Science Department Technical Report, UCLA-CSD TR-01-0023, May 2001

[44] V. Rodoplu, T.H. Ming, Minimum energy mobile wireless networks, IEEE Journal of Selected Areas in Communications 17 (8) (1999) 1333–1344.

[45] S. Dai, X. Jing and L. Li "Research and analysis on routing protocols for wireless sensor networks," in Proceeding IEEE, pp 407-41, 2005

[46]Ian Downard Naval Research Laboratory Code 5523 "Simulating Sensor networks in NS2".

[47] Eitan Altman and Tania Jimenez "NS Simulator for Beginners", December 4, 2003 Uni. De Los Andes, Merida ,Venzuela, and ESSI France.

[48] "TCL Scripts"http://cnds.eecs.jacobs-university.de/courses/anl-2008/tcl.pdf accessed on 11.04.2011

[49] http://www.isi.edu/ilense/software/smac/ns2_energy.html accessed on 2.05.2012

[50] http://www.computerworld.com.au/index.php/id;1726534212;pp;2 The A-Z of Programming Languages: AWK accessed on 12.05.2012

List of Publications

Published

- Sukhchandan Randhawa, Anil Kumar Verma, *"A Review of Power Aware Routing in Wireless Sensor Networks"* National Conference of Wireless Sensor Networks and its Applications(NCWSNA),Tamilnadu,March22, 2012

Communicated

- Sukhchandan Randhawa and A. K. Verma, *"Energy Efficient Routing in Wireless Sensor Networks"*, International Journal of Computer Science, Engineering and Applications(IJCSEA), Volume2, number 3, june 2012.