B.Sc. in Computer Science and Engineering Thesis

# Study and Analysis of Protocols of Wireless Sensor Network

Submitted by

Nasrin Hakim Mithila
ID # 200914034

Bushra Rahman
ID # 200914038

Alif Bin Taher
ID # 200814011

Supervised by

Dr. Md Shamsul Alam
Professor
Department of Computer Science & Engineering, MIST

**Department of Computer Science and Engineering**
**Military Institute of Science and Technology, Dhaka.**
**December 2012**

# CERTIFICATION

This thesis paper titled “Study and Analysis of Protocols of Wireless Sensor Network”is submitted by the group as mentioned below has been accepted as satisfactory in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science and Engineering on December 2012.

**Group Members:**

**Nasrin Hakim Mithila**  
**Bushra Rahman**  
**Alif Bin Taher**

**Supervisor:**

________________________________________

Dr. Md Shamsul Alam  
Professor  
Department of Computer Science & Engineering, MIST.  
Dhaka-1216, Bangladesh.

# CANDIDATES' DECLARATION

This is to certify that the work presented in this thesis paper is the outcome of the investigation and innovation carried out by the following students under the supervision of Dr. Md. Shamsul Alam, Professor, Computer Science and Engineering Department, Military Institute of Science and Technology (MIST), Mirpur Cantonment, Dhaka 1216, Bangladesh.

It is also declared that neither of this thesis paper nor any part thereof has been submitted anywhere else for the award of any degree, diploma or other qualification.

———————————-

Nasrin Hakim Mithila
200914034

———————————-

Bushra Rahman
200914038

———————————-

Alif Bin Taher
200814011

# ACKNOWLEDGEMENT

We are thankful to Almighty Allah for his blessings for the successful completion of our thesis. Our heartiest gratitude, profound indebtedness and deep respect go to our supervisor Dr. Md. Shamsul Alam, Professor, Computer Science and Engineering Department, Military Institute of Science and Technology (MIST), Mirpur Cantonment, Dhaka 1216, Bangladesh, for his constant supervision, affectionate guidance and great encouragement and motivation. His keen interest on the topic and valuable advices throughout the study was great help in completing thesis.

We are especially grateful to the Department of computer Science and Engineering (CSE) of Military Institute of Science and Technology (MIST) for providing all out academic support during the thesis work.

Finally, we would like to thank our families and our course mates for their appreciable assistance, patience and suggestions during the course of our thesis.

Dhaka
December 2012

Nasrin Hakim Mithila
Bushra Rahman
Alif Bin Taher

# ABSTRACT

Wireless sensor networks (WSNs) consist of a large number of sensor nodes that are densely deployed in a region of interest to collect data about a target or event, and to provide a variety of sensing and monitoring applications. Efficient design and implementation of wireless sensor networks has become a hot area of research in recent years, due to the vast potential of sensor networks to enable applications that connect the physical world to the virtual world.

There are many characteristics which differ wireless network from wired network and these distinctions introduce the need for many different protocols of WSN at different layers i.e. Physical layer, Data Link layer, Network layer, Transport layer, Application layer. Many research works have been done on the design of low power electronic devices in order to reduce energy consumption of the sensor nodes of Wireless network. MAC protocol of Data Link layer; LEACH, DSDV, AODV, DSR of Network Layer and the standard IEEE 802.15.4 contribute to this factor in various ways. The uses, drawbacks, performance analysis and also comparison between separate MAC protocols as well as comparison between DSDV, AODV and DSR protocols are discussed in detail throughout this paper.

*Keywords:* ***WSN, NS-2, MAC protocol, IEEE802.11, SMAC, TMAC, BMAC, Routing protocol, LEACH, DSDV, AODV, DSR, IEEE802.15.4*** **.**

# TABLE OF CONTENT

# LIST OF FIGURES

# LIST OF TABLES

# LIST OF ABBREVIATION

| | |
|---|---|
| **COTS** | Commercial Off-The-Shelf |
| **PDA** | Personal Digital Assistant |
| **GPS** | Global Positioning System |
| **QoS** | Quality Of Service |
| **ADC** | Analog to Digital Converter |
| **STCP** | Sensor Transmission Control Protocol |
| **PSFQ** | Pump Slowly, Fetch Quickly |
| **IEEE** | Institute of Electrical and Electronics Engineers |
| **RF** | Radio Signal |
| **NAK** | Negative Acknowledgement |
| **ACK** | Acknowledgement |
| **MAC** | Medium Access Control |
| **FEC** | Forward Error Correction |
| **ARQ** | Automatic Repeat Request |
| **CSMA/CA** | Carrier sense multiple access with collision avoidance |
| **NS** | Network Simulator |
| **LEACH** | Low-Energy Adaptive Clustering Hierarchy |
| **TDMA** | Time Division Multiple Access |
| **DSDV** | Destination-Sequenced Distance-Vector Routing |
| **PEGASIS** | Power-Efficient Gathering in Sensor Information Systems |
| **sLEACH** | Solar-aware Low Energy Adaptive Clustering Hierarchy |
| **MLEACH** | Mobile-LEACH |
| **AODV** | Ad-hoc On-Demand Distance Vector |

| | |
|---|---|
| **DSR** | Dynamic Source Routing |
| **RREQ** | Route Request |
| **RRT** | Route Reply |
| **PDF** | Packet delivery fractions |
| **CTS** | Clear To Send |
| **RTS** | Request To Send |
| **FDMA** | Frequency Division Multiple Access |
| **CBP** | Contention-based protocol |
| **DCF** | Distributed Coordination Function |
| **PCF** | Point Coordination Function |
| **NAV** | Network Allocation Vector |
| **WM** | Wireless medium |
| **SYNC** | Synchronize |
| **TMAC** | Timeout MAC |
| **FRTS** | Future Ready To Send |
| **BMAC** | Berkeley Media Access Control for Low-Power Sensor Networks |
| **TRAMA** | Traffic-adaptive medium access protocol |
| **LPL** | Low power listening |
| **CCA** | Clear Channel Assessment |
| **DIFS** | Distributed Inter-frame Space |
| **SIFS** | Short Inter-frame Space |
| **CS** | Carrier sense |
| **LRWPAN** | Low rate wireless personnel area network |
| **CID** | Cluster identifier |
| **SYNC_CW** | Synchronized Contention Window |
| **DATA_CW** | Data Contention Window |

| | |
|---|---|
| **WPAN** | Wireless Personal Area Network |
| **CLH** | Cluster head |
| **LQI** | Link quality indication |
| **PLME** | Physical layer management entity |
| **PPDU** | PHY protocol data units |
| **DSSS** | Direct sequence spread spectrum |
| **CFP** | Contention free period |
| **CAP** | Contention access period |
| **GTS** | Guaranteed time slot |

# LIST OF SYMBOLS

| | |
|---|---|
| $\Delta$ | : Delta, difference between time |
| $\Sigma$ | : Summation of energy |
| $\infty$ | : Infinite |
| $\epsilon$ | : Belongs to |

# CHAPTER 1
# INTRODUCTION

## 1.1 Generel

A wireless sensor network (WSN) is a wireless network consisting of spatially distributed autonomous devices that use sensors to monitor physical or environmental conditions. These autonomous devices, or nodes, combine with routers and gateway to create a typical WSN system. With the popularity of laptops, cell phones, PDAs, GPS devices, and intelligent electronics in the post-PC era, computing devices have become cheaper, more mobile, more distributed, and more pervasive in daily life. It is now possible to construct, from commercial off-the-shelf (COTS) components, a wallet size embedded system with the equivalent capability of a 90's PC. Such embedded systems can be supported with scaled down Windows or Linux operating systems. From this perspective, the emergence of wireless sensor networks (WSNs) is essentially the latest trend of Moore's Law toward the miniaturization and ubiquity of computing devices. Potential applications for such large-scale wireless sensor networks exist in a variety of fields, including medical monitoring, environmental monitoring, surveillance, home security, military operations, and industrial machine monitoring.

## 1.2 Applications of Sensor Network

- **Surveillance:** A surveillance application can be designed on top of a sensor network where multiple networked sensors (e.g., acoustic, seismic, video) are distributed throughout an area such as a battlefield to provide information to an end-user about the environment. In such a sensor network, the traffic can range from raw sensor data to a high level description of what is occurring in the environment. The application will have some quality of service (QoS) requirements from the sensor network, such as the network is expected to provide service for a long time (months or even years) using the limited resources of the network (e.g., sensor energy and channel bandwidth) while requiring little to no outside intervention. Meeting these goals requires careful design of both the sensor hardware and the network protocols.

- **Medical Monitoring:** This field ranges from monitoring patients in the hospital using wireless sensors to remove the constraints of tethering patients to big, bulky, wired monitoring devices, to monitoring patients in mass casualty situations, to monitoring people in their everyday lives to provide early detection and intervention for various types of disease. Consider a personal health monitor application running on a PDA that receives and analyzes data from a number of sensors (e.g., ECG, EMG, blood pressure, blood flow, pulse oxymeter). The monitor reacts to potential health risks and records health information in a local database. Considering that most sensors used by the personal health monitor will be battery-operated and use wireless communication, it is clear that this application requires networking protocols that are efficient, reliable, scalable and secure[1].

- **Habitat Monitoring:** In habitat monitoring applications, it is necessary to monitor a variety of environmental characteristics, such as temperature, humidity, barometric pressure, and other physical parameters, over significantly long periods of time and or significantly large geographical areas. An example of a mobile sensor network for habitat monitoring is the ZebraNet project [Ju02], where a sensor network was deployed to study the behavior of zebras in the Mpala Research Centre in Kenya. The idea in this project was that sensors equipped with GPS and peer-to-peer ad hoc networking can yield more information about animal behavior than simple radio collars that are sampled during the day by researchers driving around the natural area [2].

- **Structural Health Monitoring:** Structural health monitoring refers to the continual or periodic monitoring of the health of large structures such as bridges, buildings, or ships. The vibration data from bridges can be used to detect the health of bridges (e.g. whether it is ambient vibrations or some other serious condition). Examples of projects that have looked at monitoring bridges include those of Xu et al. [Xu04] and Kim et al. [Kim06]. Kim et al. [Kim06] monitored a 260 foot ( 80 m) long suspension footbridge using 13 sensors that measured the vibration of the bridge using accelerometers and proposed extending the work to the Golden Gate Bridge in San Francisco[2].

## 1.3 Sensor Network Architecture

A typical sensor network architecture consists of a sensor field, which is the physical environment where the sensor nodes or devices are deployed. Sensor nodes can possibly be deployed in extremely large numbers, on the order of thousands of sensor nodes in the field.The main task of a sensor node in a sensor field is to detect events, perform local data processing and then transmit the data. In Sensor field 1, the sensors communicate directly

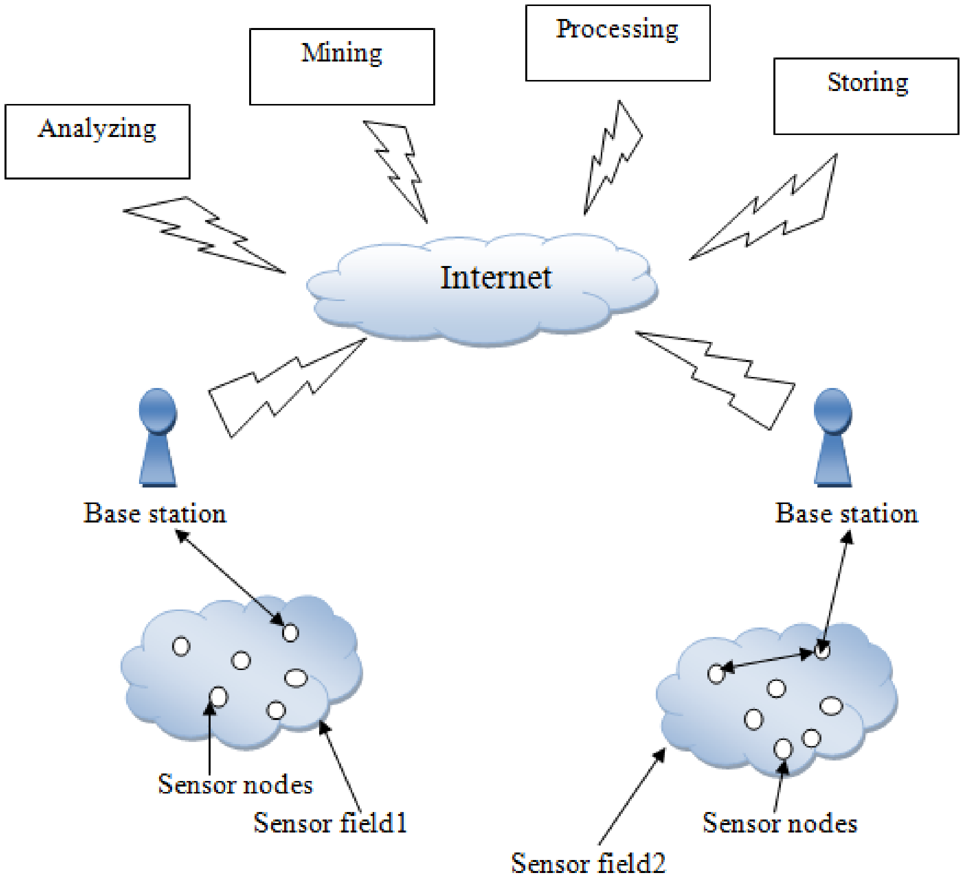


Figure 1.1: Wireless Sensor Network.

with the base station using a single hop. A multi-hop communication happens in Sensor field 2, where sensors collaborate to propagate aggregated sensor data towards the base station. Apart from sensing, a sensor node is responsible for receiving the data sent by its neighbors and forwarding these data to one of its neighbors according to the routing decisions. Typically, a sensor node is a tiny device that includes four basic components:

- **Sensing subsystem:** It includes several sensing units, which provide information gathering capabilities from the physical world. Each sensor unit is responsible for gathering information of certain type, such as temperature, humidity, or light and is usually composed of two sub units: a sensor and an analog-to-digital converter (ADC). The analog signals produced by the sensor are converted to digital signals by the ADC and fed into the processing unit.

- **Processing subsystem:** The processing unit is the main controller of the wireless sensor node, through which every other component is managed. The processing unit may consist of an on-board memory or may be associated with a small storage unit integrated into the embedded board. The processing unit manages the procedures that enable the sensor node to perform sensing operations, run associated algorithms, and collaborate with other nodes through wireless communication. IRIS and Mica mote family of nodes are equipped with 8-bit Atmel AVR micro-controllers with a speed of 4-16MHz and 128-256 kB of programmable flash.

- **Communication Unit:** Communication between any two nodes is performed by radio (transceiver) units. A communication unit implements the necessary procedures to convert bits to be transmitted into radio signal (RF) and recovers them at the other end.

- **Power Unit:** One of the most important components of a wireless sensor node is the power unit. Usually battery power is used, but other energy sources are also possible. Each component in the wireless sensor node is powered through the power unit and the limited capacity of this unit requires energy-efficient operation of the tasks performed by each component.

- **Location finding system:** Most of the sensor network applications, sensing tasks, and routing techniques need knowledge of the physical location of a node. This system may consist of a GPS (Global Positioning System) module or a software module that implements localization algorithms.

- **Mobilizer:** A mobilizer may sometimes be needed to move sensor nodes when it is necessary to carry out the assigned tasks. Mobility support requires extensive energy resources and should be provided efficiently. The mobilizer controls the movement of the sensor node.

- **Power Generator:** While battery power is mostly used in sensor nodes, an additional power generator can be used for applications where longer network lifetime is essential.

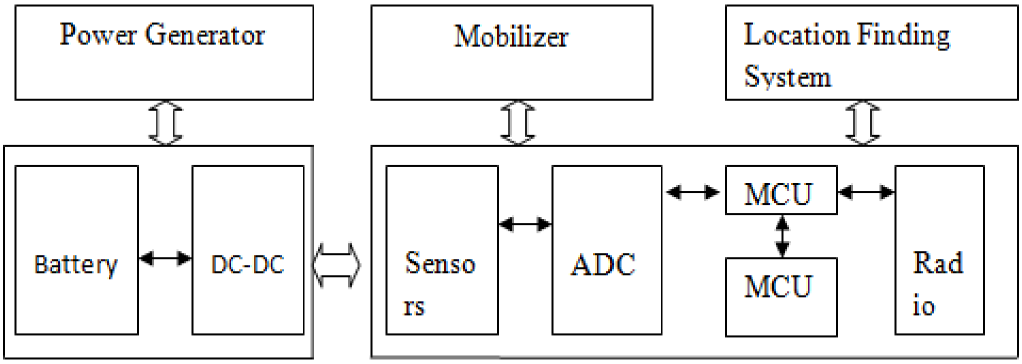


Figure 1.2: Typical Architeceture of Sensor Network

.

## 1.4 Taxonomy of Sensor Networks

There are many ways to classify different sensor network architectures, the following list highlights some fundamental differences in sensor networks that affect protocol design.

- **Data sink:** One of the most important aspects of a sensor network is the nature of the data sink. In some situations, the end user may be embedded within the sensor network or may be less accessible mobile access points that collect data once in a while.

- **Sensor mobility:** Another classification of sensor networks may be made based on the nature of the sensors being deployed. Typically Sensors are assumed immobile but mobile sensors are used in some applications. The mobility of sensors can influence protocols at the networking layer as well as those for localization services.

- **Sensor resources:** Sensor nodes may vary greatly in the computing resources available. It is obvious that memory and processing constraints should influence protocol design at nearly every level.
- **Traffic patterns:** Another important aspect to consider is the traffic generated on the network. In many event-driven applications, sensors may operate in a sentry state for the majority of time, only generating data traffic when an event of interest is detected. In other applications such as environmental monitoring, data should be continuously generated.

As can be seen by the above discussion, there are many features of the sensors, the network and the application that should influence protocol design. Accordingly, much research has gone into designing protocols for these different scenarios.

## 1.5 Unique Features of Sensor Networks

It should be noted that sensor networks do share some commonalities with general ad hoc networks. Thus, protocol design for sensor networks must account for the properties of ad hoc networks, including the following.

- Lifetime constraints imposed by the limited energy supplies of the nodes in the network.
- Unreliable communication due to the wireless medium.
- Need for self-configuration, requiring little or no human intervention.

## 1.6 WSN OSI layers

### 1.6.1 Transport layer:

The function of this layer is to provide reliability and congestion avoidance where a lot of protocols designed to provide this function are either applied on the upstream (user to sink, ex: STCP), or downstream (sink to user, ex: PSFQ) [3]. These protocols use different mechanisms for loss detection (ACK, NACK, and Sequence number) and loss recovery (End to End or Hop by Hop) [4]. This layer is specifically needed when a system is organized to access other networks.

### 1.6.2 Network layer:

The major function of this layer is routing. This layer has a lot of challenges depending on the application but apparently, the major challenges are in the power saving, limited memory and buffers, sensor does not have a global ID and have to be self organized. The basic idea of the routing protocol is to define a reliable path and redundant paths according to a certain scale called metric, which differs from protocol to protocol. There is a lot of routing protocols available for this layer; among of all these, some are thoroughly discussed in Chapter 3.

### 1.6.3 Data link layer:

Responsible for multiplexing data streams, data frame detection, MAC, and error control, ensure reliability of pointpoint or point multipoint. Errors or unreliability comes from:

- Co- channel interference at the MAC layer and this problem is solved by MAC protocols.
- Multipath fading and shadowing at the physical layer and this problem is solved by forward error correction (FEC) and automatic repeat request (ARQ).

**ARQ:** not popular in WSN because of additional re-transmission cost and overhead. ARQ is not efficient to frame error detection so all the frame has to be retransmitted if there is a single bit error.

**FEC:** decreases the number of retransmission by adding redundant data on each message so the receiver can detect and correct errors. By that we can avoid re-transmission and wait for ACK.

**MAC layer:** Responsible for Channel access policies, scheduling, buffer management and error control. In WSN we need a MAC protocol to consider energy efficiency, reliability, low access delay and high throughput as a major priority. The MAC layer is discussed in detail in Chapter 2.

### 1.6.4 Physical Layer:

Can provide an interface to transmit a stream of bits over physical medium. Responsible for frequency selection, carrier frequency generation, signal detection, Modulation and data encryption.

**IEEE 802.15.4:** proposed as standard for low rate personal area and WSN with low cost, complexity, power consumption, range of communication to maximize battery life. Use CSMA/CA, support star and peer to peer topology. There are many versions of IEEE 802.15.4.

### 1.6.5 Application layer:

Responsible for traffic management and provide software for different applications that translate the data in an understandable form or send queries to obtain certain information. Sensor networks deployed in various applications in different fields, for example; military, medical, environment, agriculture fields.

## 1.7 Simulator for performance evaluation of WSN

There are many simulators such as Network Simulator 2 (NS-2), OPNET Modeler, GloMoSim, OMNeT++ and etc. , we choices Network Simulation Tool (NS-2) version 2.35 which was released in November 4, 2011.

NS (version 2) is an object-oriented, discrete event driven network simulator developed at UC Berkely written in C++ and OTcl. NS2 has a virtual clock inside, and all simulations which are run by NS2 are driven by discrete events. It can be used to simulate wired, wireless, and wired-cum-wireless scenarios. NS2 is able to simulate a variety of networking protocols and functions, such as network transmission protocols, traffic generators, routing queue mechanism, routing algorithms, multicast, MAC protocols, and so on.NS2 has a module called network animator and has a function being able to record and store link information into trace files. So, the users can watch animation how the networks simulated works and analyze trace files after the simulations to figure out what happens in the networks. NS2 is public network simulation software which is free of charge. It can be downloaded from official website.

Usually, NS2 runs in Linux environment. Fortunately, there is software called Cygwin which offers a Unix-like environment on Microsoft Windows platform.

Although NS is fairly easy to use once you get to know the simulator, it is quite difficult for a first time user, because there are few user-friendly manuals. Even though there is a lot of documentation written by the developers which has in depth explanation of the simulator, it is written with the depth of a skilled NS user.

Currently, NS-2 is actively maintained and used in academic research since it is easily extendable and based on open source. For wireless scenarios, mobile nodes have been incorporated through a patch develop by MIT.

### 1.7.1 NS2 Architecture

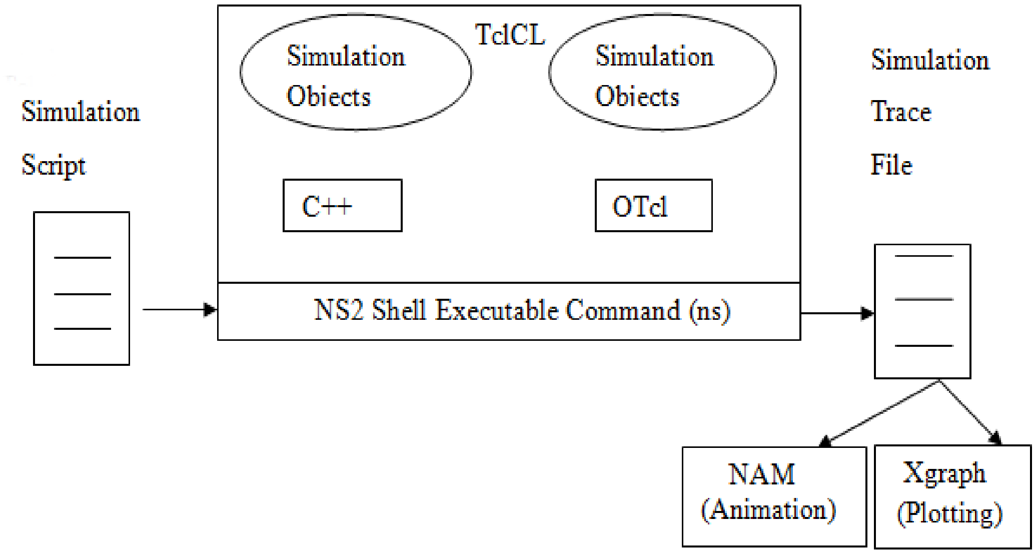


Figure 1.3: NS2 Architechture.

Figure 1.3 shows the basic architecture of NS2. NS2 provides users with an executable command ns which takes on input argument, the name of a Tcl simulation scripting file. Users are feeding the name of a Tcl simulation script(which sets up a simulation) as an

input argument of an NS2 executable command ns.
In most cases, a simulation trace file is created, and is used to plot graph and/or to create animation. Trace file analysis is done using awk scripting language. Graphs are plotted using gnuplot and animation is shown using network animator (NAM) both of which are included in the all-in-one package.

NS2 consists of two key languages: C++ and Object-oriented Tool Command Language (OTcl). While the C++ defines the internal mechanism (i.e., a backend) of the simulation objects, the OTcl sets up simulation by assembling and configuring the objects as well as scheduling discrete events (i.e., a frontend). The C++ and the OTcl are linked together using TclCL [5].

### 1.7.2 Modules and simulation process of NS2

In order to simulate all kinds of networks, many function modules are encapsulated inside NS2. The typical modules include nodes, links, agents, data packets, and more. The nodes can represent source nodes, routers, or other nodes. The links connect nodes, deliver data packets, and manage transmission. The agents are attached on the nodes and are given port numbers. They generate or receive traffic. The data packets carry information transmitted on links.

Generally, the progress of a NS2 simulation is as follows:

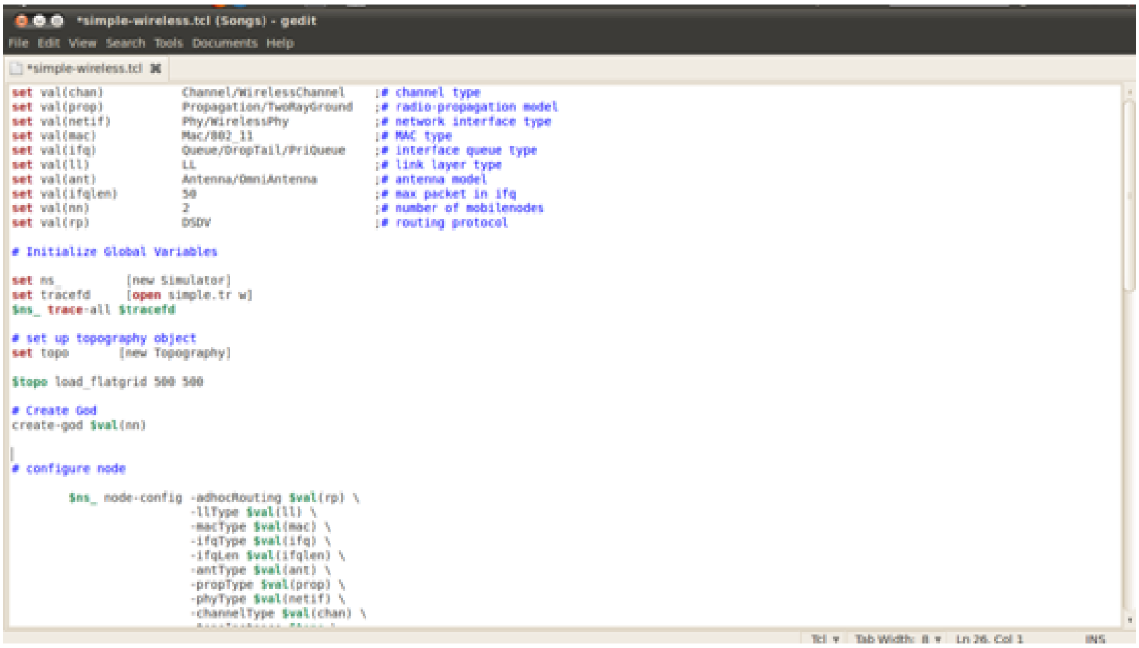
- Find out what scenario is going to be simulated.
- Start to write TCL scripts, setup parameters, and build topologies.
- Run the scripts.
- Get simulation results and analyze the trace files.
- If new data are needed, repeat the process.

### 1.7.3 A Simulation Example

#### 1.7.3.1 Simulation Design

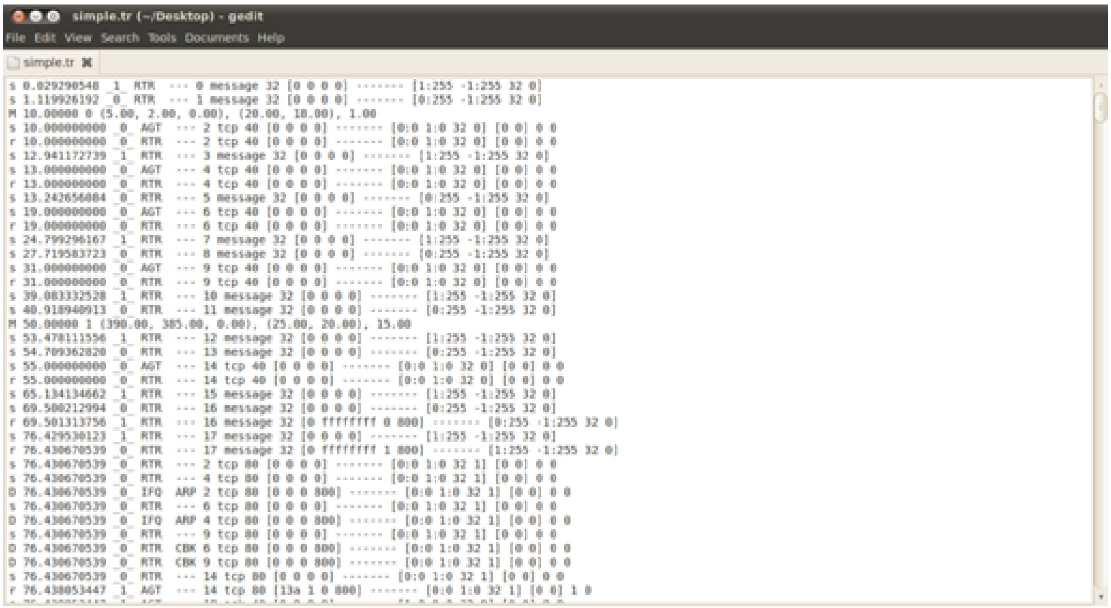
We are going to simulate a very simple 2-node wireless scenario. The topology consists of two mobile nodes, node_(0) and node_(1). The mobile nodes move about within an area whose boundary is defined in this example as 500mX500m. The nodes start out initially at two opposite ends of the boundary. Then they move towards each other in the first half of the simulation and again move away for the second half. A TCP connection is setup between the two mobile nodes. Packets are exchanged between the nodes as they come within hearing range of one another. As they move away, packets start getting dropped [6].

**Writing Tcl Script:** Just as with any other ns simulation, we begin by creating a tcl script for the wireless simulation. We will call this file simple-wireless.tcl. The tcl Script is look like:


*simple-wireless.tcl (Songs) - gedit
File Edit View Search Tools Documents Help
*simple-wireless.tcl
Tcl ▾ Tab Width: 8 ▾ Ln 26, Col 1 INS

```
set val(chan)           Channel/WirelessChannel    ;# channel type
set val(prop)           Propagation/TwoRayGround   ;# radio-propagation model
set val(netif)          Phy/WirelessPhy            ;# network interface type
set val(mac)            Mac/802_11                 ;# MAC type
set val(ifq)            Queue/DropTail/PriQueue    ;# interface queue type
set val(ll)             LL                         ;# link layer type
set val(ant)            Antenna/OmniAntenna        ;# antenna model
set val(ifqlen)         50                         ;# max packet in ifq
set val(nn)             2                          ;# number of mobilenodes
set val(rp)             DSDV                       ;# routing protocol

# Initialize Global Variables

set ns_         [new Simulator]
set tracefd     [open simple.tr w]
$ns_ trace-all $tracefd

# set up topography object
set topo        [new Topography]

$topo load_flatgrid 500 500

# Create God
create-god $val(nn)


# configure node

        $ns_ node-config -adhocRouting $val(rp) \
                         -llType $val(ll) \
                         -macType $val(mac) \
                         -ifqType $val(ifq) \
                         -ifqLen $val(ifqlen) \
                         -antType $val(ant) \
                         -propType $val(prop) \
                         -phyType $val(netif) \
                         -channelType $val(chan) \
```

Figure 1.4: Tcl Script of simple-wireless.tcl.


simple.tr (~/Desktop) - gedit
File Edit View Search Tools Documents Help
simple.tr

```
s 0.029290548 _1_ RTR  --- 0 message 32 [0 0 0 0] ------- [1:255 -1:255 32 0]
s 1.119926192 _0_ RTR  --- 1 message 32 [0 0 0 0] ------- [0:255 -1:255 32 0]
M 10.00000 0 (5.00, 2.00, 0.00), (20.00, 18.00), 1.00
s 10.000000000 _0_ AGT  --- 2 tcp 40 [0 0 0 0] ------- [0:0 1:0 32 0] [0 0] 0 0
r 10.000000000 _0_ RTR  --- 2 tcp 40 [0 0 0 0] ------- [0:0 1:0 32 0] [0 0] 0 0
s 12.941172739 _1_ RTR  --- 3 message 32 [0 0 0 0] ------- [1:255 -1:255 32 0]
s 13.000000000 _0_ AGT  --- 4 tcp 40 [0 0 0 0] ------- [0:0 1:0 32 0] [0 0] 0 0
r 13.000000000 _0_ RTR  --- 4 tcp 40 [0 0 0 0] ------- [0:0 1:0 32 0] [0 0] 0 0
s 13.242656084 _0_ RTR  --- 5 message 32 [0 0 0 0] ------- [0:255 -1:255 32 0]
s 19.000000000 _0_ AGT  --- 6 tcp 40 [0 0 0 0] ------- [0:0 1:0 32 0] [0 0] 0 0
r 19.000000000 _0_ RTR  --- 6 tcp 40 [0 0 0 0] ------- [0:0 1:0 32 0] [0 0] 0 0
s 24.799296167 _1_ RTR  --- 7 message 32 [0 0 0 0] ------- [1:255 -1:255 32 0]
s 27.719583723 _0_ RTR  --- 8 message 32 [0 0 0 0] ------- [0:255 -1:255 32 0]
s 31.000000000 _0_ AGT  --- 9 tcp 40 [0 0 0 0] ------- [0:0 1:0 32 0] [0 0] 0 0
r 31.000000000 _0_ RTR  --- 9 tcp 40 [0 0 0 0] ------- [0:0 1:0 32 0] [0 0] 0 0
s 39.083332528 _1_ RTR  --- 10 message 32 [0 0 0 0] ------- [1:255 -1:255 32 0]
s 40.918940913 _0_ RTR  --- 11 message 32 [0 0 0 0] ------- [0:255 -1:255 32 0]
M 50.00000 1 (390.00, 385.00, 0.00), (25.00, 20.00), 15.00
s 53.478111556 _1_ RTR  --- 12 message 32 [0 0 0 0] ------- [1:255 -1:255 32 0]
s 54.709362820 _0_ RTR  --- 13 message 32 [0 0 0 0] ------- [0:255 -1:255 32 0]
s 55.000000000 _0_ AGT  --- 14 tcp 40 [0 0 0 0] ------- [0:0 1:0 32 0] [0 0] 0 0
r 55.000000000 _0_ RTR  --- 14 tcp 40 [0 0 0 0] ------- [0:0 1:0 32 0] [0 0] 0 0
s 65.134134662 _1_ RTR  --- 15 message 32 [0 0 0 0] ------- [1:255 -1:255 32 0]
s 69.500212994 _0_ RTR  --- 16 message 32 [0 0 0 0] ------- [0:255 -1:255 32 0]
r 69.501313756 _1_ RTR  --- 16 message 32 [0 ffffffff 0 800] ------- [0:255 -1:255 32 0]
s 76.429530123 _1_ RTR  --- 17 message 32 [0 0 0 0] ------- [1:255 -1:255 32 0]
r 76.430670539 _0_ RTR  --- 17 message 32 [0 ffffffff 1 800] ------- [1:255 -1:255 32 0]
s 76.430670539 _0_ RTR  --- 2 tcp 80 [0 0 0 0] ------- [0:0 1:0 32 1] [0 0] 0 0
s 76.430670539 _0_ RTR  --- 4 tcp 80 [0 0 0 0] ------- [0:0 1:0 32 1] [0 0] 0 0
D 76.430670539 _0_ IFQ  ARP 2 tcp 80 [0 0 0 800] ------- [0:0 1:0 32 1] [0 0] 0 0
s 76.430670539 _0_ RTR  --- 6 tcp 80 [0 0 0 0] ------- [0:0 1:0 32 1] [0 0] 0 0
D 76.430670539 _0_ IFQ  ARP 4 tcp 80 [0 0 0 800] ------- [0:0 1:0 32 1] [0 0] 0 0
s 76.430670539 _0_ RTR  --- 9 tcp 80 [0 0 0 0] ------- [0:0 1:0 32 1] [0 0] 0 0
D 76.430670539 _0_ RTR  CBK 6 tcp 80 [0 0 0 800] ------- [0:0 1:0 32 1] [0 0] 0 0
D 76.430670539 _0_ RTR  CBK 9 tcp 80 [0 0 0 800] ------- [0:0 1:0 32 1] [0 0] 0 0
s 76.430670539 _0_ RTR  --- 14 tcp 80 [0 0 0 0] ------- [0:0 1:0 32 1] [0 0] 0 0
r 76.438053447 _1_ AGT  --- 14 tcp 80 [13a 1 0 800] ------- [0:0 1:0 32 1] [0 0] 1 0
```

Figure 1.5: Trace file of simple-wireless.tcl.

**Running the Script:** We run the by writing a command on the command prompt. We have to first go the directory where the tcl script is saved using the command prompt the have to write ns simple-wireless.tcl, then press enter. By doing this, a trace file named simple.tr will be created at the same directory.

Analyzing the trace file: Trace file looks like: The general format of each trace line is shown in Fig. 1.4, where 12 columns make up a complete trace line. The type identifier field corresponds to four possible event types that a packet has experienced: r (received), + (enqueued), - (dequeued), and d (dropped). The time field denotes the time at which such event occurs. Fields 3 and 4 are the starting and the terminating nodes, respectively, of the link at which a certain event takes place. Fields 5 and 6 are packet type and packet size, respectively. The next field is a series of flags, indicating any abnormal behavior. Note the output ”——-” denotes no flag. Following the flags is a packet flow ID. Fields 9 and 10 mark the source and the destination addresses, respectively, in the form of node port. For correct packet assembly at the destination node, NS also specifies a packet sequence number in the second last field. Finally, to keep track of all packets, a packet unique ID is recorded in the last field. Now, having this trace at hand would not be useful unless meaningful analysis is performed on the data. In post-simulation analysis, one usually extracts a subset of the data of interest and further analyzes it. For example, the average throughput associated with a specific link can be computed by extracting only the columns and fields associated to that

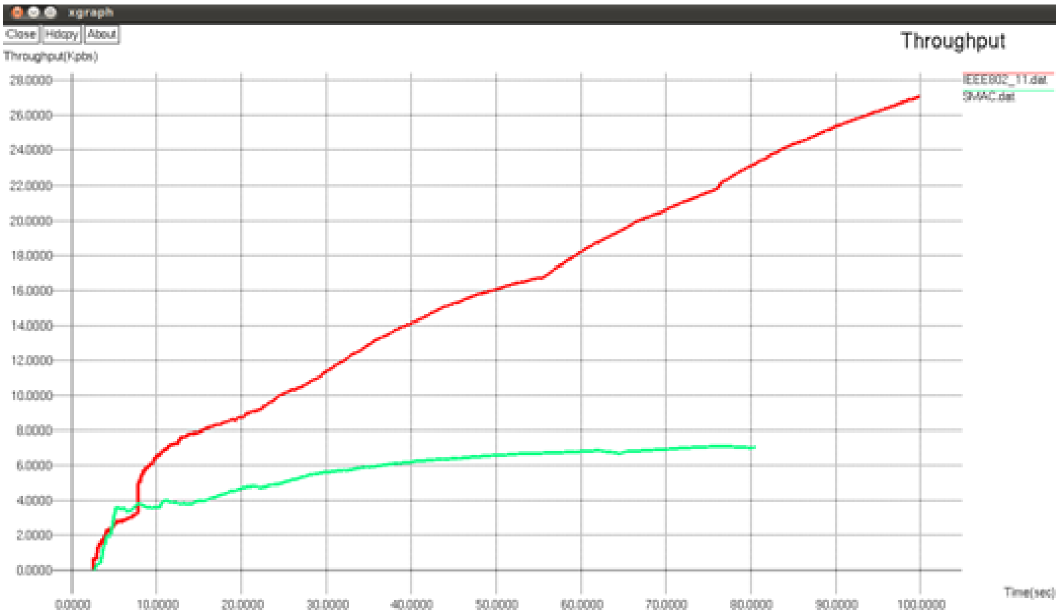


Figure 1.6: Graph output of NS2.

link from the trace file. In figure 1.6 an example of graph output of NS2 is shown. Two of the most popular languages that facilitate this process are AWK and Perl. In this paper, we have used AWK for analyzing our simulation output in Chapter 2 and Chapter 3.

# CHAPTER 2
# MEDIUM ACCESS CONTROL PROTOCOL

## 2.1 Preliminaries

Wireless Sensor Networks (WSN) consist of a large number of battery-powered sensors, embedded processor, moderate amount of memory and transmitter/receiver circuitry capable of communicating wireless transmission. They are distributed and arranged within a vicinity of interest in order to track, measure and monitors various events. This battery powered sensors consume a very large amount of energy. Many research works have been done on the design of low power electronic devices in order to reduce energy consumption of these sensor nodes. Medium Access control (MAC) protocols play a big role to reduce energy consumption in WSN. These protocols are designed in such a way that can consume little power, avoid collisions from interfering nodes, can be implemented with a small code size and memory requirements. It can also work with changing frequency. This technique ensures how nodes share the channel and do the successful operation of the network. The internal operations of the MAC layer are performed, while allowing the application program to run. This is achieved by implementing a scheduler which periodically checks if the MAC layer needs to perform any tasks. The timer starts the scheduler and allows the scheduler to execute any pending tasks. Designing power efficient MAC protocol is one of the ways to prolong the life time of the network. There are various types of MAC protocols .Such as: IEEE802.11,SMAC, TMAC,BMAC etc.

In section 2.1 we write about the objectives of MAC protocols. In section 2.2 designing a Well-defined MAC Protocol are written. As my thesis topic is related to contention based protocol so at section 2.3 some contention based protocols are described. At section 2.4 a detail description on SMACs working principle is described. At section 2.5 we give some simulation result based on IEEE802.11 and SMAC protocol by using NS-2.35. At section 2.6 a summary is given about this chapter.

## 2.2 Objectives of MAC protocol

There are two main objectives of MAC protocol. They are:

First objective of MAC protocol is the creation of the sensor network infrastructure. A wide range of sensor nodes are deployed and the MAC protocol has to establish the communication link among the sensor nodes of the network.

Second objective is to share the communication medium among the sensor nodes fairly and effectively.

## 2.3 Designing a well-defined MAC protocol

For designing a well-defined MAC protocol first we have to find the sources of energy wastages in MAC protocol then we have to know attributes of a good protocol for removing the sources of energy wastages. Then we have to know about the communication pattern in wireless sensor network.

### 2.3.1 Main reasons for energy waste

MAC sub-layers task is to provide fair access to channels by avoiding possible collisions because in WSN, nodes usually have to share a common channel. For this energy wastages happen in WSN. Reasons for energy wastage in WSN are described below:

1. **Collision of packets:** First reason is the collision of transmitting packets in the channel. It happens when nodes dont listen to the medium before transmission of a packet .When a packet is corrupted due to interference during its transmission from sender to receiver or vice versa, it has to be discarded and the follow on retransmissions increase energy consumption. It increases latency also. This causes unnecessary energy wastages.

2. **Overhearing:** Second one is overhearing the channel, meaning that a node picks up packets that are destined to other nodes.

3. **Control packet overhead:** Third source of energy waste is control packet overhead. There are two types of control packets.

    - **Vote packet:** Decision of a node is contained in it, which can be either positive vote or a negative vote. This packet is sent to nodes, which sent their energy values to this node

    - **Radio-power-mode packet:** Radio-power-mode of the sender is contained here, to indicate whether the sender is using one slot or two slots for transmitting its data. Minimum numbers of control packets are used to make a data transmission. Sending and receiving control packets consumes energy too much and contain less useful data packets to be transmitted. Though Control packets dont represent useful data they are necessary for successful data transmission. As an example we can say about 2 control packets: RTS (Request-to-send) and CTS (Clear-to-send) used in some protocols do not carry any useful data to applications although their transmission consumes high energy. The exchange of RTS/CTS induces high overheads in the range of 40 percent to 75 percent of the channel capacity, because data frames are typically very small in sensor networks [7].

4. **Idle listening:** One of the major sources of energy waste is idle listening. It is happened when the node keeps its radio on while listening to the channel waiting for potential data frames to receive that is not sent. When nothing is sensed at all, the sensor node will be in idle state for most of the time. Sensor network applications naturally generate low traffic load. So, the communication channel is expected to be idle most of the time. Nodes waste considerable amounts of energy as they keep their radios on for large time of intervals while listening to an idle channel. The amount of energy wasted whilst the radio is on in an idle state is shown in the following table 2.1:

Table 2.1: Current consumption of main status of a typical radio [8].

| | |
|---|---|
| Radio(transmit) | 22 mA |
| Radio(receive) | 14 mA |
| Radio(sleep) | 900 nA |
| Radio(idle) | 1.5 mA |
| Microcontroller(active) | 8 mA |
| Microcontroller(idle) | 2 mA |

Due to the importance of energy consumption of idle listening, in energy-efficient MACs nodes sleep for as long periods of time as possible instead of enabling for being permanently active. As an example, a sensor application that requires nodes to exchange messages with their neighbors an average rate of exchanging a message per second. Usually messages take less than 5 milliseconds to transmit. So, each node spends on average 5 ms per second on transmitting and 5 ms on receiving a message from another node. So, at last 990 ms is required on listening while nothing happens. We therefore see that radio is doing nothing for 99% of the total time.

5. **Over-emitting:** The last reason for energy waste is over-emitting. It is caused by the transmission of a message when the destination node is not ready. Nodes send data from sender when the recipient node is not ready to accept incoming transmission.

### 2.3.2 Attributes of a well-defined MAC protocol

1. **Efficiency of energy savings:** The main source of sensor node is battery. It is seen that it is cost-effective to replace the nodes rather than changing or replacing them. Energy efficiency of the sensor nodes can be defined as-

$$\text{Energy efficiency} = \frac{\text{Total energy consumed}}{\text{Total bytes in data packets received by sink node}}$$

Efficiency of a protocol in transmitting the information through the network depends on the above ratio. If the value of this ratio is lass then the nodes give better performance. Energy efficiency can be increased by minimizing the energy wastage like: collision, overhearing, idle listening and packet overhead.

2. **Latency:** Latency mainly depends on the sensor network applications. Sink nodes must be acknowledged before about the events to be occurred. So that nodes can take actions immediately.

$$\text{Average Packet Latency} = \frac{\text{Packets reach to sink node}}{\text{Total time}}$$

3. **Throughput of the network:** Better throughput of a network can be achieved if the sink nodes receive more data. Some sensor network applications sample the information with fine temporal resolution.

$$\text{Network Throughput} = \frac{\text{Total bytes in data packets received by sink node}}{\text{Time from first packet generated at source to last packet received by sink node}}$$

4. **Scalability and adaptability:** Due to the movement of sensor nodes and nature of wireless transmission, MAC protocol needs to be adaptable to changes in network topology. Changes in network by size of node, node density and topology should be handled quickly and perfectly for a successful adaptation.MAC protocol must be highly scalable to deal with a large number of nodes with different node density patterns in the controlled environment. Also, it must also be adaptable to dynamic topology changes due to node failure or mobility.

5. **Self-stabilization:** Protocol should be self-stabilized to changes in the network. If changes happen in the network such as arrival of a new node, it should affect only the nodes in the vicinity of the change.

6. **Avoiding collision:** When a receiver node receives more than one packet at the same time, these packets are called collided packets and this event is called collision. Reducing of collision can be achieved by listening to the channel (CSMA/SMAC) or by using time (TDMA), frequency (FDMA).

7. **Hidden and exposed terminal problems:** The reason for these two problems is the broadcast nature of the radio channel, namely, all the nodes within a nodes transmission range receive its transmission.

- **Hidden terminal problem:** Two nodes that are outside each-others range perform simultaneous transmission to a node that is within the range of each of them, hence, there is a packet collision.

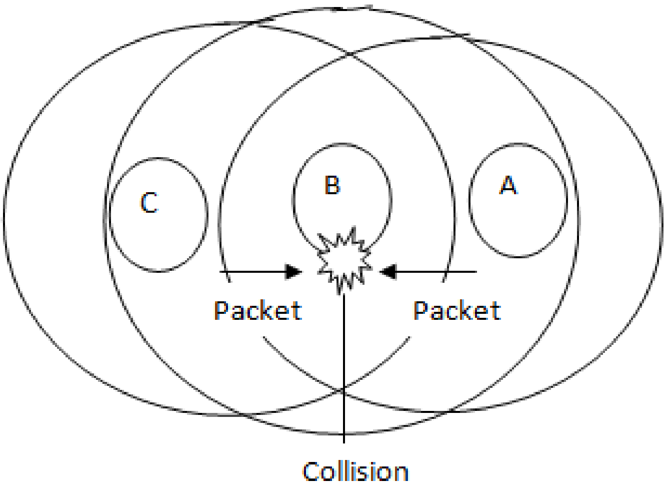


Figure 2.1: Hidden terminal problem for MAC protocol where A is sender B is receiver & C is hidden terminal.

For example, in Fig. 2.1 suppose node A is transmitting to node B and simultaneously C is trying to communicate with node B. According to the CSMA protocol, node C senses the medium, but since C is out of As transmission range, it fails to understand that A is transmitting to B and finds the medium free. As a result, C accesses the medium, causing collisions at B. This phenomenon is known as hidden terminal problem and C is called the hidden terminal.

- **Exposed terminal problem:** The node is within the range of a node that is transmitting, and it cannot transmit to any node. For example, in Fig. 2.2 suppose node A wants to transmit to node C. So, it first transmits the packet to node B and then B will transmit the packet to node C. But here, Node B is an exposed terminal. So, it doesnt forward the DATA packet to node C. Hidden nodes mean increased probability of collision at a receiver, whereas exposed nodes may be denied channel access unnecessarily, which means underutilization of the bandwidth resources. So, this problem must be removed.

8. **Delay:** Delay is the amount of time needed for the transmission to reach the receiver. The average delay for packet transmission should be as small as possible.

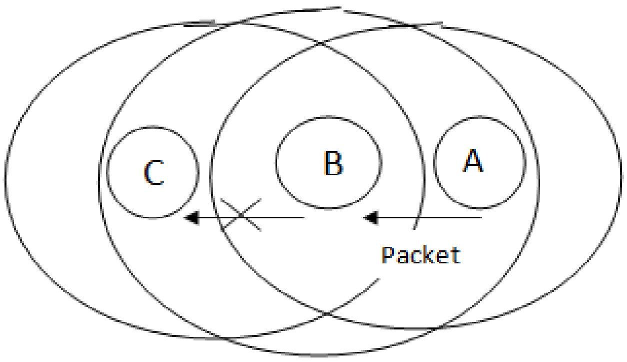


Figure 2.2: Exposed terminal problem for MAC protocol where A is sender, C is receiver & B is the exposed terminal.

$$\text{Mean end-to-end delay} = \frac{\text{Sum of end-to-end delays for all received packets}}{\text{Total number of packets received by sinknode}}$$

9. **Bandwidth efficiency:** The scarcity of bandwidth resources in these networks calls for its efficient usage. To quantify this, we could say that bandwidth efficiency is the ratio of the bandwidth utilized for data transmission to the total available bandwidth. In these terms, the target will be to maximize this value.

10. **Fairness:** Sink node receives information from all sensor nodes fairly, when there is limited bandwidth.

11. **Other issues:** Power control mechanisms are needed for efficient management of the energy consumption of the nodes. Protocol should provide QoS support for real-time traffic. Hardware constraints etc.

### 2.3.3 Communication pattern

It is important to design and test the behavior of MAC protocols based on the kind on the traffic they use for transmission data. They are:

1. **Broadcast communication:** A broadcast pattern is normally used by a base station to transmit some information to all the sensor nodes of the network. Broadcasted information contains queries of sensor query-processing architectures, program updates for sensor nodes, control packets of the system. One problem with broadcast transmission is that unless the protocol is sender scheduled, the transmission of broadcast

packets requires the repetition of the same packet several times. It is not very energy efficient.

2. **Local gossip:** Here, a sensor sends a message to its neighboring nodes within a range. Sensors that detect an event communicate with each other locally.

3. **Convergecast:** After detecting an event sensors need to send what they perceive to the information center. A group of sensors communicate to a specific sensor. Here, the destination node could be a cluster-head, a data fusion center, or a base station.

4. **Reporting of nodes to sink:** After processing a local event, nodes may want to report something. Designer expects messages to be directed to one or a few sink nodes, which are hooked up to a network. There should be some random variation in message paths- messages flow 'roughly' in the correct direction. In this communication pattern, we found unidirectional flow of messages through the network. Although this pattern is frequently used to study MAC protocols, we exclude routed, multi-hop communication between random nodes in the sensor network.

## 2.4 Some proposed MAC layer protocols

A wide range of MAC protocols defined for sensor networks are described below by stating the essential behavior of the protocols wherever possible. The MAC protocols for the wireless sensor networks can be classified into two categories: Schedule based and Contention based. The schedule based protocol maintains strict time synchronization and can avoid collisions, overhearing and idle listening by scheduling transmit and listen periods. Such as: TRAMA, DMAC etc. On the other hand, the contention based protocols relax time synchronization requirements. It can easily adjust to the topology changes as some new nodes may join and others may die few years after deployment. These protocols are based on Carrier Sense Multiple Access (CSMA) technique but have higher costs for message collisions, overhearing and idle listening. Such as: IEEE802.11,Sensor S-MAC, TMAC, BMAC etc. As my thesis work is related with contention based protocols, I study on contention based protocols. I study on SMAC in detail and then some updated versions of SMAC are explained in brief.

### 2.4.1 Contention based protocols

A contention-based protocol (CBP) is a communications protocol for operating wireless telecommunication equipment that allows many users to use the same radio channel without pre-coordination. Here, the channel access policy is based on competition. Whenever a node needs to send a packet, it tries to get access to the channel. These protocols cannot provide QoS, since access to the network cannot be guaranteed beforehand. This protocol allows multiple users to share the same bandwidth by defining the events that must occur when two or more transmitters attempt to simultaneously access the same channel and establishing rules by which a transmitter provides reasonable opportunities for other transmitters to operate. At first, working principle of IEEE802.11 MAC protocol is described. Then upgraded versions of MAC protocols are described below.

#### 2.4.1.1 IEEE802.11

The IEEE 802.11 is a contention based medium access control protocol which uses carrier sensing and randomized back-offs to avoid collisions of the data packets during transmission. Details of this protocol are described in Chapter 1. Here, working principle and limitation of this protocol are described.

**Working principle of IEEE 802.11:** IEEE 802.11 some controls frames are used. Controls frames facilitate in the exchange of data frames between stations. IEEE 802.11 controls frames are given below:

- **Acknowledgement (ACK) frame:** After receiving a data frame, the receiving station will send an ACK frame to the sending station if no errors are found. If the sending station doesn't receive an ACK frame within a predetermined period of time, the sending station will resend the frame.
- **Request to Send (RTS) frame:** The RTS and CTS frames provide an optional collision reduction scheme for access points with hidden stations. A station sends a RTS frame to as the first step in a two-way handshake required before sending data frames.
- **Clear to Send (CTS) frame:** A station responds to an RTS frame with a CTS frame. It provides clearance for the requesting station to send a data frame. The CTS provides

collision control management by including a time value for which all other stations are to hold off transmission while the requesting stations transmits.

The IEEE802.11 MAC protocol has two modes: DCF and PCF.

**a. DCF (Distributed Coordination Function):** In this mode no central device controlling mechanism is present for the communication. The DCF defines two access mechanisms for packet transmissions: basic access mechanism, and RTS/CTS access mechanism.

**a.1. Basic access mechanism:** Carrier sensing and Virtual carrier sensing are found here.

- **Carrier sensing:** Here, a node senses the medium and if it is idle, the node transmits the data frame. If the medium is busy, the node waits until it becomes idle again, waits for a random time and transmits. The receiver node answers with an ACK (acknowledgment) control frame, upon frame reception. If a collision occurs, transmitting nodes wait a random time and try again and again.

- **Physical carrier sensing:** Physical carrier sense is performed at physical layer by checking the current radio state. Every time when the radio starts receiving or transmitting, the PHY layer will inform the MAC thereof. It happens also when the receiving or transmitting is over. Obviously, the medium will be determined as busy when radio is in receiving or transmitting state.

- **Virtual carrier sensing:** Any station, before transmitting a DATA frame, senses the channel for duration of time equal to the Distributed Inter-frame Space(DIFS) to check if it is idle. If the channel is determined to be idle, the station starts the transmission of a DATA frame. All stations which hear the transmission of the DATA frame set their Network Allocation Vector (NAV)(An indicator, maintained by each station, of time periods when transmission onto the wireless medium (WM) will not be initiated by the station.) to the expected length of the transmission, as indicated in the Duration/ID field of the DATA frame. Upon successful reception of the DATA frame, the destination station waits for a SIFS interval following the DATA frame, and then sends an ACK frame back to the source station indicating successful reception of the DATA frame. The channel is considered to be busy if either the virtual carrier sensing indicates. In that case, the station enters into a wait period.

**a.2. RTS/CTS access mechanism:** The RTS/CTS access mechanism uses a four-way hand shake in order to reduce bandwidth loss due to the hidden terminal problem. The four way handshake prevents any DATA-DATA collisions that might occur due to the hidden terminal problem. A station that wishes to send a DATA frame first senses the channel for a DIFS duration If the medium is found idle, the node sends a control frame called RTS which contains the intended receiver address and the time required to send the information (transmission delay).Then if the destination node agrees to communicate, it will answer with a CTS control frame which also contains the delay. Upon successful transmission of the RTS frame, the destination waits for a SIFS interval, and then sends a CTS frame back to the source. The source can start sending the DATA frame a SIFS interval after the reception of the CTS frame. The node which first sends out the RTS packet wins the medium, and the receiver will reply with a CTS packet. It should be noted that, all nodes hearing RTS or CTS should refrain from transmission until the transmission delay has elapsed and the medium is idle again. The receiver must respond with an ACK for each data frame received. Since the RTS and CTS frames are very small in size, the RTS/CTS access scheme significantly reduces bandwidth loss due to collisions.

**b. PCF (Point Coordination Function):** Here a special node called the access point (AP), polls every node to control the communication process. Periodically, an AP broadcasts a beacon control frame with parameters and invitations to join the network.

**Limitations of IEEE802.11:**

- IEEE802.11includes the large overhead in control and data packets. 802.11 requires 34 bytes for the header and the checksum, TCP and IP require a minimum of 20 bytes for each header, so there is at least 74 bytes of overhead to send application information, which in WSNs may be only two bytes.

- The most important problem for using 802.11 in WSNs is energy consumption since it does not address the issue of avoiding overhearing and idle listening. Although this standard has power saving mechanisms, according to Ferrari et al. power consumption is rather high, and the short autonomy of a battery supply still remains the main disadvantage of the proposed IEEE802.11 sensor system.

- Again, for energy-constrained nodes, overhearing by all neighbors wastes a lot of energy. This causes a long delay if the receiver really needs the entire message to start processing.

- Transmitting a long message using a single data packet through a lossy channel is hazardous and risky. Even when a few bits in the packet are corrupted during the transmission, the whole packet must be re-transmitted. This will waste a lot of time and energy. Therefore, some MAC protocols like 802.11 support a fragmentation mechanism, which breaks a long message into some small fragments. All fragments are sent in a burst, and using one pair of RTS/CTS, if none of them is corrupted. If one of the fragments is corrupted, another pair of RTS/CTS is needed. When receiver has gotten all fragments, its MAC is responsible for assembling all the fragments into a whole and passing it upwards. Message passing extends the transmission time and re-transmits the current fragment. Thus it has fewer contentions and a small latency. There should be a limit on how many extensions can be made for each message in case that the receiver is really dead or lost in connection during the transmission of packets.

- Another problem that is associated with IEEE 802.11 is the memory overflow problem at the nodes because of using extra RTS/CTS control packets. After reaching the saturation point the number of successfully received packets is reduced rapidly because of the limited memory of the nodes.

- In most of the situations this technique works fine but under certain circumstances the IEEE 802.11 fails to solve this hidden terminal problem which is shown in the following Figure 2.3.

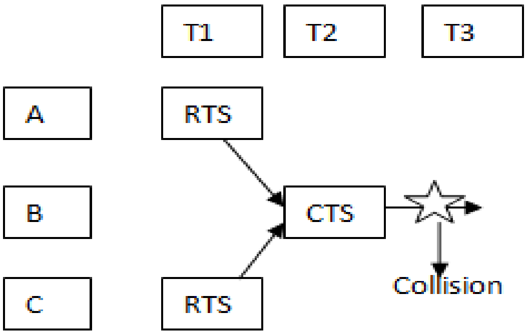


Figure 2.3: Hidden terminal problem after using RTS/CTS control packets.

The scenario in this figure 2.3 can be defined as; A wants to send data packet to B. So A sends RTS to B at time T1. These requests get by both B. Meanwhile, C sends a RTS to B at the same time T1 by sensing the medium free. In this position a collision is occurred at B and it cant understand which node to send CTS.

#### 2.4.1.2 Sensor SMAC

In the year of 2002, sensor S-MAC is designed for the wireless sensor network which is a contention based MAC protocol with integrated low-duty-cycle operation and it is a modification of IEEE 802.11 protocol. The main goal in MAC protocol design is to reduce energy consumption, while supporting good scalability and collision avoidance and also tries to reduce energy consumption from all the sources that we have identified to cause energy waste, i.e., idle listening, collision, overhearing and control overhead etc.

**Mechanisms and features:** Mechanisms and features of SMAC are: The energy saving with S-MAC is achieved by:

- Low operational duty cycle, varying from 1 to 10 percent.This means that a WSN node using S-MAC will cyclically alternate between Listen and Sleep states, where the listen state never exceeds 10 percent of the cycle duration [11].
- Going into the sleep state as often as possible, by turning off the radio transceiver during the transmission periods of other nodes to avoid overhearing unwanted packets [11]. Protocol overhead when streaming a sequence of message fragments is reduced by:
- S-MAC adopts a modified fragmentation mechanism for transmitting a long message, called message passing. Its basic idea is to fragment a long message into many small fragments and send them in a burst. SMAC tries to avoid overhearing by letting interfering nodes go to sleep after they hear an RTS or CTS packet. SMAC adopt the RTS/CTS mechanism to address the hidden terminal problem. It follows similar procedures, including virtual carrier sense and RTS/CTS exchange for collision avoidance.

**Advantages of SMAC :**

- S-MAC introduces is periodic sleep and listen which accordingly saves a lot of energy, especially when traffic load is low.

- To achieve maximum energy saving and improve latency, S-MAC defines a complete synchronization mechanism, including periodic SYNC packets broadcast, schedule table and neighbor list maintenance.

- For avoiding collision and solving hidden terminal S-MAC has adopted include physical and virtual carrier sense, RTS/CTS/DATA/ACK sequence.

- S-MAC tries to avoid overhearing by letting all interfering nodes, which are immediate neighbors of the both sender and receiver, go to sleep after they hear an RTS or CTS packet.

- To efficiently transmit long messages in both energy and latency respects, S-MAC supports message passing. In this way, S-MAC trades fairness on fragment level for fairness and latency on message level.

- Another important feature of SMAC is adaptive listening, which reduce latency. The basic idea is to give all nodes, which are involved in a transmission, an additional chance for transmitting their packets at the end of the transmission. These nodes include the sender, the receiver, and all their immediate neighbors that overhear the transmission. In this way, a data packet can be retransmitted immediately after its last transmission.

**Disadvantages of SMAC:**

- Broadcast data packets do not use RTS/CTS, which increases collision probability.
- Adaptive listening incurs overhearing or idle listening if the packet is not destined to the listening node.
- Sleep and listen periods are predefined and constant, which decreases the efficiency of the algorithm under variable traffic load.
- The period length is limited by delay and cache size; the active time depends on message transmission rate; the active time must adapt to highest traffic load to guarantee reliable and timely message transmission; the idle listening will relatively increase when traffic load is low.
- SMAC scheduling mechanism works when self-configuration is in set mode. In the listen period, a node senses its neighbor nodes and transmits SYNC packets that contain randomly generated schedule. Thus a long time is taken by each node to get synchronized. For instance, if 10 nodes are implemented in the network, they have to wait 100 seconds to setup the schedule and for 15 nodes the time rises to 150 seconds. Thus a longer time for stabilization takes place in proportion to the number of nodes in a network [11].

#### 2.4.1.3 Timeout MAC (TMAC)

Timeout MAC (TMAC)is proposed to enhance the poor results of the SMAC protocol under variable traffic loads which tries to remove high latency and lower throughput of SMAC. This problem of SMAC is caused by static sleeplisten periods of SMAC.

**Mechanisms and features:** Mechanisms and features of TMAC are:

- It introduces an adaptive duty cycle. All messages are transmitted in variable length bursts and the lengths of bursts are dynamically determined. Similar to S-MAC, there are active periods and sleep periods in a time-frame.
- An active period ends if there is no activity for a time period of TA. TA is the minimum listening time in the time-frame. The decision for TA is presented along with some

solutions to the early sleeping problem. The time TA is the minimal amount of idle listening per frame. The interval TA¿ TCI + TRT + TTA + TCT where TCI is the length of the contention interval, TRT is the length of an RTS packet, TTA is the time between the end of the RTS packet and the beginning of the CTS packet and TCT is the length of the CTS packet. Variable loads in sensor networks are expected, since the nodes that are closer to the sink must relay more traffic and traffic may change over time.

**Advantages of TMAC:**

- A major advantage of the TMAC over the SMAC protocol is in the adaptive frame time. In the SMAC protocol, as the traffic load changes the duty cycle need to be changed in order to operate efficiently.
- The TMAC protocol adapts to changes in network traffic by itself.
- TMAC also supports overhearing avoidance, full-buffer priority and Future Ready To Send (FRTS) packets.

**Disadvantages of TMAC:**

- Although T-MAC gives better results under these variable loads, the synchronization of the listen periods within virtual clusters is broken. This is one of the reasons for the early sleeping problem. Mechanisms introduced to signal to nodes there is traffic for them at the beginning of the active time to prevent them from going to sleep.

#### 2.4.1.4 Berkeley Media Access Control for Low-Power Sensor Networks

The Berkeley Media Access Control (BMAC) is a contention based MAC protocol which duty cycles the radio transceiver i.e. the sensor node turns ON/OFF repeatedly without missing the data packets. BMAC employs an adaptive preamble to reduce idle listening which is a major source of energy usage in many protocols. When a node has a packet to send, it waits during a fixed time called back-off time before checking the channel. If the channel is clear, the node transmits; otherwise it begins a second (congestion) back off. Each node must check the channel periodically using LPL (low-power listening) and if the

channel is idle and the node has no data to transmit, the node returns to sleep. The BMAC preamble sampling scheme adjusts the interval in which the channel is checked to equal the frame preamble size. As an example, if the medium is checked every 100ms, the preamble of the packet must last 100ms as a minimum, in order for the receiver to detect the packet. Upper layers may change the preamble duration, according to the application requirements.
**Mechanisms and features:** Mechanisms and features of BMAC are:

- One mechanism is the Clear Channel Assessment (CCA) for effective collision avoidance, which takes samples of the media to estimate the noise floor.
- BMAC also utilizes a preamble sampling like technique called Low Power Listening (LPL) to minimize the idle listening problem.
- BMAC includes the use of ACK frames for reliability purposes and throughput improvement.
- One of the most interesting features of BMAC not available in any other protocol thus far is the capability of tuning its operation and mechanisms. BMAC provides interfaces that can include/exclude BMAC mechanisms, such as the CCA, acknowledgments, and LPL to tradeoff power consumption, latency, throughput, fairness or reliability.

**Advantages of BMAC:**

- An advantage of using BMAC in wireless sensor network is that it does not use RTS, CTS, ACK, or any other control frame by default, but they can be added.
- Additionally, it is one of the few specialized MAC protocols whose implementation was tested in hardware.
- This provides optimal trade-off between energy savings and latency or throughput.
- No synchronization is required, and the protocol performance can be tuned by higher layers to meet the needs of various applications.

**Disadvantages of BMAC:**

- The main disadvantage is that the preamble creates large overhead. One example presents 271 bytes of preamble to send 36 bytes of data.
- Hidden terminal and multi-packet mechanisms not provided.
- Low duty cycle and has a longer preamble
- Little cost to receiver yet higher cost to sender and Longer delay.

Rest of the paper is concerned with SMAC on which we study thoroughly during our thesis work. At first we tell about the components of SMAC, and then we tell about the simulation work related with IEEE802.11 & SMAC.

## 2.5 SMAC design overview

SMAC consists of three major components: periodic listen and sleep, collision and overhearing avoidance and message passing. Their brief descriptions are given below:

### 2.5.1 Working principle of the Components of Sensor SMAC

#### 2.5.1.1 Periodic listen & sleep

In many sensor network applications, nodes are in an idle mode for a long time if no sensing event happens. During this period, data rate is very low and it is not necessary to keep nodes listening all the time. The main technique used to reduce energy consumption in S-MAC is to make each node in the network follow a listen and sleep cycle. A complete cycle of listen and sleep period is called a frame (figure 2.4).SMAC reduces the listen time by letting node go into periodic sleep mode. For example, if a node for each second sleeps for half second and listens for the other half, its duty cycle is reduced to 50%. So we can achieve close to 50% of energy savings. Each node goes to sleep for some time, and then wakes up for listening to see if any other node wants to talk to it. In this period, the node turns off its radio, and sets a timer to awake itself later. The duration of time for listening and sleeping can be selected according to different application scenarios. S-MAC provides a controllable parameter duty cycle, whose value is the ratio of the listen period to the frame length. In fact, the listen period is normally fixed according to some physical and MAC layer parameters. The user can adjust the duty cycle value from 1% to 100% to control the length of sleep period. Normally, the frame length is the same for all nodes in network.

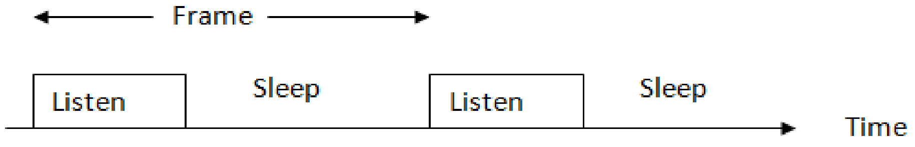


Figure 2.4: Frame = Time (Listen) + Time (Sleep)

For maintaining periodic listen & sleep, the activity can be divided into 2 parts. Initial schedule is established by

1. Choosing and Maintaining Schedules.

2. Maintaining Synchronization.

**1. Choosing and Maintaining Schedules:** For choosing and maintaining schedule, each node maintains a schedule table that stores the schedules of all its known neighbors. Before each node starts its periodic listen and sleep, it needs to choose a schedule and exchange it with its neighboring nodes. Here, we have to know about two terms: synchronizer& follower.

**a. Synchronizer:** Though all nodes are free to choose their own listen/sleep schedules for reducing control overhead, we prefer neighboring nodes to synchronize together. That is, during listening period one node follows another nodes scheduling time of listening and during sleeping period one node follows another nodes scheduling time of sleeping. It should be pointed that not all neighboring nodes can synchronize together in a multi-hop network. For example, two neighboring nodes 1 and 2may have different schedules but if they can synchronize with different nodes, 3 and 4, respectively, as shown in Figure 2.5.

Figure 2.5: Neighboring nodes 1 and 2 have different schedules. They synchronize with nodes 3 and 4 respectively.

In the figure 2.5, if node 1 want to synchronize with node 3 then it must broadcasts its schedule to node 3 for synchronization. Here, node 1 is called synchronizer.

When nodes exchange their schedules by broadcasting it to all its immediate neighbors it must be ensured that all neighboring nodes can talk to each other even if they have different schedules. As shown in Figure if node 1 wants to talk to node 2, it just waits until node 2 is listening.

**b. Follower:** If the node receives a schedule from a neighbor before choosing its own schedule, it follows that schedule by setting its schedule to be the same. We call such a node a follower.

In the figure 5, if node 1 selects a scheduling time for him then node 3 must follow the scheduling time of node 1 and choose its scheduling time to keep pace with the scheduling time of node 1. Here, node 3 is called follower.

By using synchronizer and follower, nodes follow the below steps to choose their schedule and establish its schedule table.

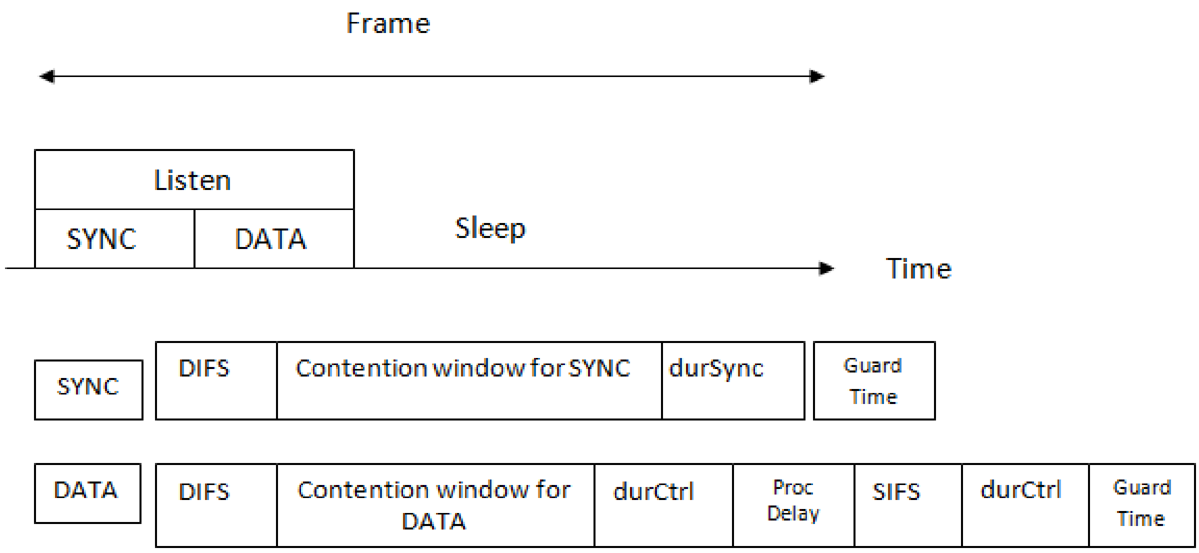
- At the first step, node listens for a certain amount of time. If it does not hear a schedule from another node, it randomly chooses a time to go to sleep and immediately broadcasts its schedule in a SYNC message which indicates that it will go to sleep after a time, t seconds. Here, synchronization between nodes is happened and such a node is called a synchronizer, since it chooses its schedule independently and other nodes will synchronize with it.

- After getting the schedule from the synchronizer, follower then waits for a random delay t and rebroadcasts this schedule, indicating that it will sleep in t+t seconds. The random delay is for collision avoidance, so that multiple followers triggered from the same synchronizer do not systematically collide when rebroadcasting the schedule.

- If a node receives a different schedule after selecting and broadcasting its own schedule, it adopts both schedules (i.e., it schedules itself to wake up at the times of both is neighbor and itself). It broadcasts it own schedule before going to sleep.

**2. Maintaining Synchronization:** The listen/sleep scheme requires maintaining synchronization among neighboring nodes. Although the long listen time can tolerate fairly large clock drift, neighboring nodes still need to periodically update each other their schedules to prevent long-time clock drift. Here, the updating period can be quite long. Updating schedules can be accomplished by sending a SYNC packet. The SYNC packet is very short which includes the address of the sender i.e. identification number of sender and the time of its next sleep. The next-sleep time is relative to the moment that the sender finishes transmitting the SYNC packet, which is approximately when receivers get the packet. Receivers will adjust their timers immediately after they receive the SYNC packet. A node will go to sleep when the timer fires. In the case that multiple neighbors want to talk to a node, they need to contend for the medium when the node is listening. The contention mechanism is the same as that in IEEE 802.11 which is stated earlier, i.e., using RTS (Request To Send) and CTS (Clear To Send) packets. The node that first sends out the RTS packet wins the medium, and the receiver will reply with a CTS packet and sender can sends DATA to receiver. When a node fails to win the contention or it encounters an RTS collision, it goes to sleep until the next active period and when anode sends out an RTS successfully, it does not go back to sleep until the transmitted DATA packet is acknowledged. In order for a node to receive both SYNC packets and data packets, we divide its listen interval into two parts. The first part is for receiving SYNC packets, and the second one is for DATA packets where RTS packets are send for DATA and if CTS is received then DATA is send to sender, as shown in Figure. Each part is further divided into many time slots for senders to perform carrier sense. For example, if a sender wants to send a SYNC packet, it starts carrier sense when the receiver begins listening and randomly selects a time slot to finish its CS (carrier sense). If it has not detected any transmission by the end of the time slot, it wins the medium and starts sending its SYNC packet at that time. The same procedure is followed when sending data. In figure 2.6, frame format of SMAC is shown.

Figure 2.7 shows the timing relationship of three possible situations that a sender transmits to a receiver. In the figure 7, two types of sender are shown. In fig. 2.7(b) & (c) Sender 1 only sends a SYNC packet and sender 2 sends a SYNC packet and a RTS packet. In the case of sender 2 DATA is send if only CTS is received. Each node periodically broadcasts SYNC packets to its neighbors even if it has no followers. This allows new nodes to join an existing neighborhood. The new node follows the same procedure in the above subsection to choose its schedule. In fig. 2.7(d) at the receiver side, receiver sends CTS after getting RTS from sender. In fig. 2.7(e) after getting DATA from sender receiver sends back ACK to



DIFS = Distributed Inter-frame Space.

SIFS= Short Inter-frame Space.

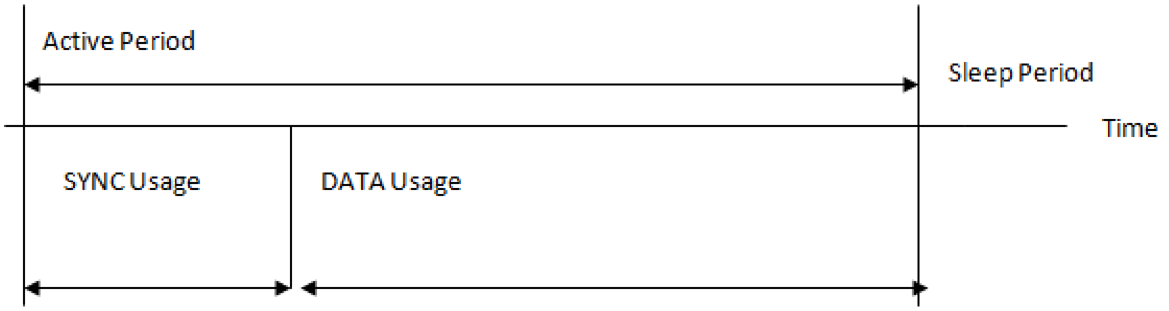
durSync =Time for transmitting a SYNC packet.

durCtrl = Time for transmitting a control packet.

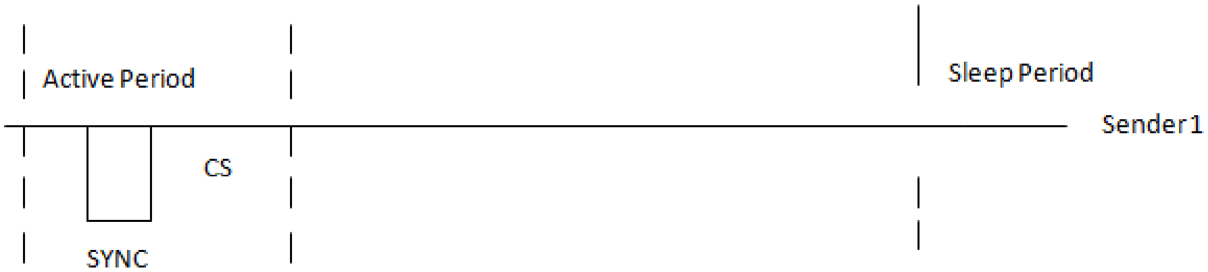
Guard Time= Packets waits for a guard time before attempting to transmit anything.

Figure 2.6: SMAC frame format.

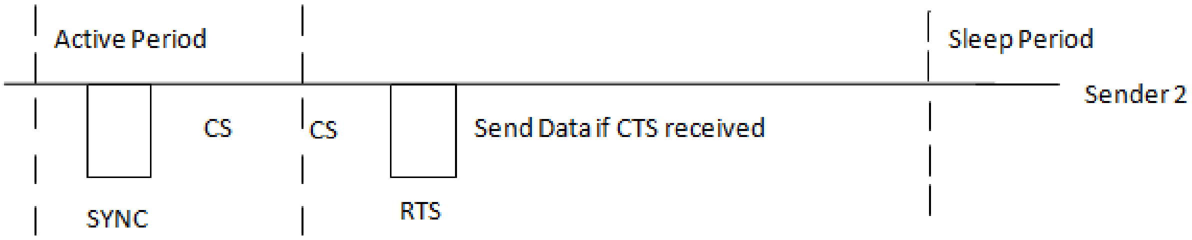
sender for ensuring of getting DATA. In fig. 2.7(f) after getting ACK from receiver, sender goes to sleep mode. So, packets follow the sequence of RTS/CTS/DATA/ACK between the sender and the receiver for transmitting DATA.



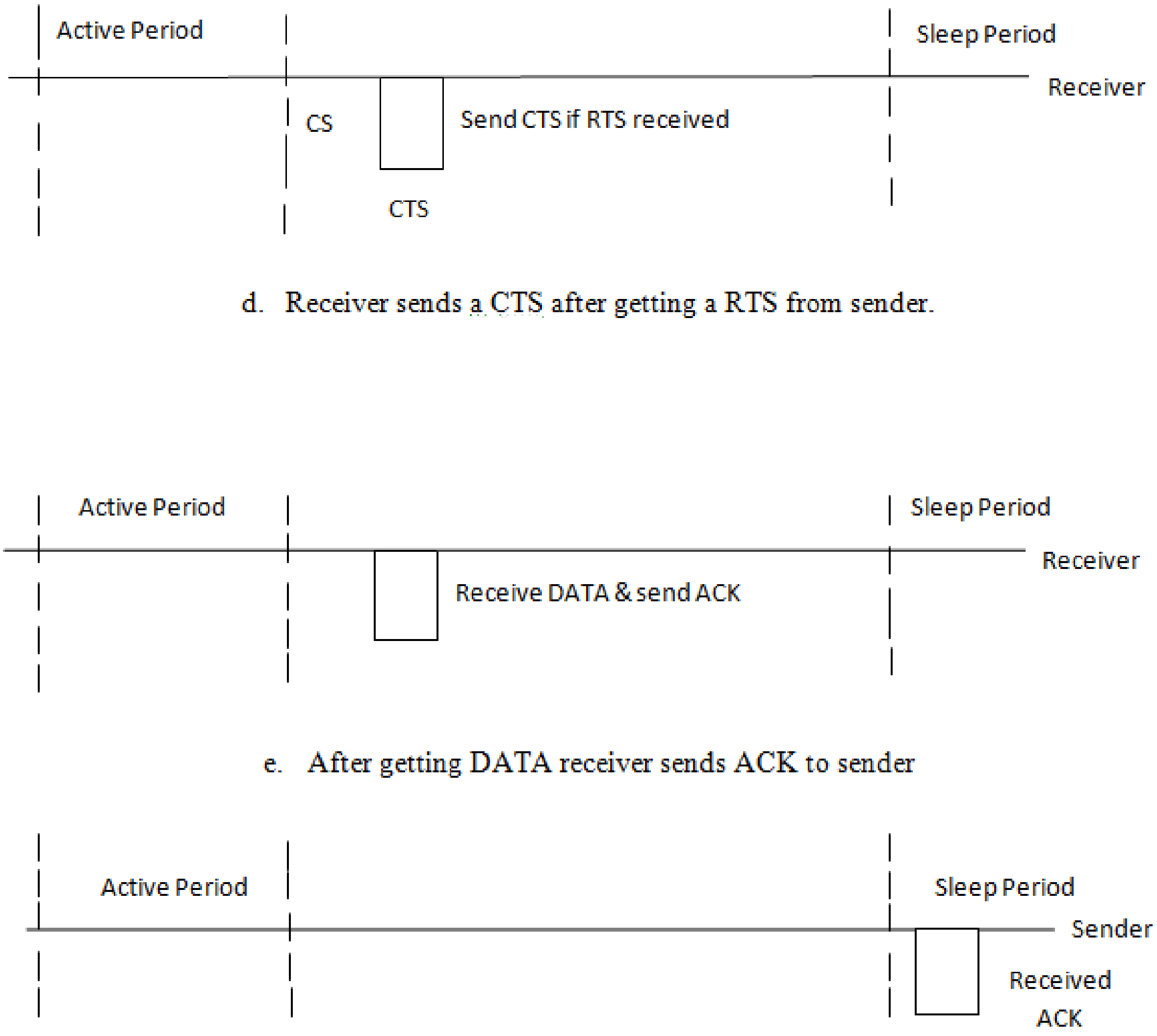
a. Contained packets in a SMAC frame.



b. Sender 1 sends only SYNC packet.



c. Sender 2 sends both SYNC & DATA packet.



d. Receiver sends a CTS after getting a RTS from sender.

e. After getting DATA receiver sends ACK to sender

f. After getting ACK from receiver sender goes to sleep mode.

CS = Carrier sense.

ACK = Acknowledgement.

RTS = Request To Send.

CTS = Clear To Send.

Figure 2.7: Timing relationship between a receiver and senders.

The initial listen period should be long enough so that it is able to learn and follow an existing schedule before choosing an independent one. Once transmission starts, it does not stop until completed. After the data transmission between nodes they simply follow a sleep schedule together. They do not follow their sleep schedules until they finish transmission. For this component, latency is increased due to the periodic sleep of each node and the delay can accumulate on each hop.

#### 2.5.1.2 Collision and overhearing avoidance

Multiple senders may want to send to a receiver at the same time and so that collision occurs among nodes. So, they need to contend for the medium to avoid collisions. Among contention based protocols, though 802.11 does a very good job of collision avoidance SMAC performs better than 802.11. SMAC adopt the RTS/CTS mechanism to address the hidden terminal problem. It follows similar procedures, including virtual carrier sense and RTS/CTS exchange for collision avoidance.

- **Physical carrier sense:** Physical carrier sense is performed at physical layer by checking the current radio state. Every time when the radio starts receiving or transmitting, the PHY layer will inform the MAC thereof. It happens also when the receiving or transmitting is over. Obviously, the medium will be determined as busy when radio is in receiving or transmitting state.

- **Virtual carrier sense:** There is a duration field in each transmitted packet that indicates how long the remaining transmission will be so that, if a node receives a packet destined to another node, it knows how long it has to keep silent. The node records this value in a variable called the network allocation vector (An indicator, maintained by each station, of time periods when transmission onto the wireless medium (WM) will not be initiated by the station.) and sets a timer for it. When the NAV timer fires, the node decrements the NAV value until it reaches zero. When a node has data to send, it first looks at the NAV then if its value is not zero; the node determines that the medium is busy. So, the node doesnt send DATA to another. All senders perform carrier sense before initiating a transmission and if a node fails to get the medium, it goes to sleep and wakes up when the receiver is free and listening again.

Here, NAV is used to indicate the activity in its neighborhood. When a node receives a packet destined to other nodes, it updates its NAV by the duration field in the packet and a non-zero NAV value indicates that there is an active transmission in its neighborhood. So, a node should sleep to avoid overhearing if its NAV is not zero. The NAV value decrements every time when the NAV timer fires and a node can wake up when its NAV becomes zero.

In 802.11 each node keeps listening to all transmissions from its neighbors. As a result, each node overhears a lot of packets that are not destined to it. This is a significant waste of energy, especially when node density is high and traffic load is heavy. SMAC tries to avoid overhearing by letting interfering nodes go to sleep after they hear an RTS or CTS packet. Since DATA packets are normally much longer than control packets, this approach prevents neighboring nodes from overhearing long DATA packets and the ACKs.

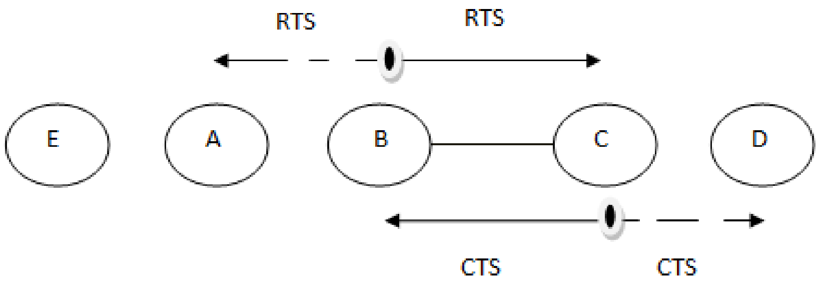


Figure 2.8: Nodes A and D overhear the transmission of Nodes B and C respectively.

In figure 2.8, let nodes A,B,C,D create a multi-hop network where each node can only hear the transmissions from its immediate neighbors. Collision only happens on the receiver side. Obviously, D is supposed to go to sleep, because its transmission interferes with Cs reception of the DATA packet. A is two hops away from C, so As transmission will not interfere with Cs reception. But if A talks to E while B is sending data to C, A will not receive any packet from E because collision happens on A. As transmission is a waste of energy and it also needs to go to sleep.

#### 2.5.1.3 Message Passing

S-MAC adopts a modified fragmentation mechanism for transmitting a long message, called message passing. Its basic idea is to fragment a long message into many small fragments and send them in a burst. This supports to reduce protocol overhead when streaming a sequence of message fragments. Through using this technique, one may achieve energy savings by minimizing communication overhead at the expense of unfairness in medium access. A message is the collection of meaningful, interrelated units of data which can be a long series of packets or short packets. Usually the receiver needs to obtain all the data units before it can perform in-network data processing or aggregation. Transmitting a long message as a single packet results high cost of re-transmitting the long packet if only a few bits have been corrupted in the first transmission. Moreover, if fragmentation of the long message into many independent small packets is made, we have to pay the penalty of large control overhead and longer delay. It is so because the RTS and CTS packets are used in contention for each independent packet in every transmission. Only one RTS packet and one CTS packet are used for a long message which is divided into many small fragments, and transmit them in burst. These small fragments reserve the medium for transmitting all the fragments. Every time a data fragment is transmitted, the sender waits for an ACK from the receiver. If it fails to receive the ACK, it will extend the reserved transmission time for one more fragment, and re-transmit the current fragment immediately. Switching the radio from sleep to active does not occur instantaneously. Therefore, it is desirable to reduce the frequency of switching modes. The message passing scheme tries to put nodes into sleep state as long as possible, and hence reduces switching overhead. As stated before, all packets have the duration field, which is now the time needed for transmitting all the remaining data fragments and ACK packets. If a neighboring node hears a RTS or CTS packet, it will go to sleep for the time that is needed to transmit all the fragments. The purpose of using ACK after each data fragment is to prevent the hidden terminal problem. It is possible that a neighboring node wakes up or a new node joins in the middle of a transmission. If the node is only the neighbor of the receiver but not the sender, it will not hear the data fragments being sent by the sender. If the receiver does not send ACK frequently, the new node may mistakenly infer from its carrier sense that the medium is clear. If it starts transmitting, the current transmission will be corrupted at the receiver. Each data fragment and ACK packet also has the duration field. In this way, if a node wakes up or a new node joins in the middle, it can properly go to sleep no matter if it is the neighbor of the sender or the receiver. For example, suppose a neighboring node receives an RTS from the sender or CTS from the receiver, it goes to sleep for the entire message time. If the sender extends the transmission time due to fragment losses or errors, the sleeping neighbor will not be aware of the extension immediately. However, the node will learn it from the extended fragments or ACKs when it wakes up.

## 2.6 Simulation and performance analysis

Here, we find the throughput between the MAC protocols IEEE802.11 & SMAC & compare their outputs. We compute the throughout, using the payloads received at the MAC layer.

**Experiment platform:** Network Simulator version 2 (NS-2) has been used as experiment platform. NS-2 provides extensive support for queuing algorithms, routing protocols, multicast protocols and IP protocols over both wired network and wireless network.

**Topology:** We have simulated wireless sensor networks with regular topology as well as randomly generated topology.

**Traffic pattern:** We attach a UDP agent and a CBR traffic source to the sink node. The CBR source generates 20 packets (each 80 Bytes, because it will be added with 20 bytes IP header at the routing layer, so the actual size at MAC layer is 100 Bytes).

**Routing Protocol:** DSR.

**Simulation Time:** 100 sec.

**Number of nodes:** 20 & 10.

**Simulation Parameters settings:** Some important parameters used in the steady-state simulations are listed in the Table 2.2 & Table 2.3.

Table 2.2: Default values of SMAC parameters [13].

| **Parameter name** | **Value** |
|---|---:|
| SMAC_DUTY_CYCLE | 10% |
| SMAC_MAX_NUM_NEIGHBORS | 20 |
| SMAC_MAX_NUM_SCHEDULES | 4 |
| SMAC_EXTEND_LIMIT | 5s |
| SYNC_CW | 31 |
| DATA_CW | 63 |
| SYNCPERIOD | 10s |
| SIZEOF_SMAC_DATAPKT | 512bytes |
| durDataPkt_ | 43ms |
| syncTime_ | 55.2ms |
| dataTime_ | 105ms |
| listenTime_ | 160.2ms |
| sleepTime_(10% duty cycle) | 1442.8ms |

Now, we measure the performance i.e. throughput along with bandwidth vs. time by applying simulation on wireless sensor network with MAC protocols IEEE802.11 and SMAC.

**Throughput:** Throughput or network throughput is the average rate of successful message delivery over a communication channel. The throughput is usually measured in bits per second (bit/s or bps). Bandwidth measures the maximum throughput of a computer network [15].

### 2.6.1 Simulation result

We know that, IEEE802.11 consumes much energy for idle listening, collision, overhearing and control overhead etc. In SMAC, these problems can be reduced by using sleep periods and consumes less energy than IEEE802.11. Though S-MAC reduces energy consumption, idle listening, overhead this savings may be offset by decreased throughput.

When we run simulation we see that, IEEE802.11 uses much bandwidth than SMAC but gives a higher throughput. As SMAC uses sleep period (when data rate is very low and it is

Table 2.3: PHY and other MAC layer parameters [13].

| **Parameter name** | **Value** |
|---|---:|
| BANDWIDTH | 20kbps |
| SIZEOF_SMAC_CTRLPKT | 10bytes |
| SIZEOF_SMAC_SYNCPKT | 9bytes |
| slotTime_ | 1ms |
| Difs | 10ms |
| Sifs | 5ms |
| Eifs | 50ms |
| durSyncPkt_ | 11ms |
| RTS/CTS/ACK Size | 10bytes |

not necessary to keep nodes listening all the time) for which duty cycle is reduced to 50% & we can achieve close to 50% of energy savings. But during this period bandwidth utilization is very low. So, throughput is also low. Moreover, SMAC follows synchronization among nodes for avoiding collision. For this reason when a synchronizer starts transmission, it will not stop until finishing transmission. So, synchronizer occupies the bandwidth for it during transmission. As, few nodes take part in transmission, throughput becomes low. In the figure 2.9, we see that, when we use IEEE802.11, at a certain time all nodes start to sense the medium and when find that the medium is not busy they start transmission. For this, there was a higher use of bandwidth and we got a higher throughput though there may be packets loss, higher energy consumption is occurred. When we use SMAC, we found in figure 2.9 that at a certain time nodes start to sense the medium and when one node does not hear a schedule from another node, it randomly chooses a time to go to sleep and immediately broadcasts its schedule to its neighboring nodes in a SYNC message which indicates that it (synchronizer) will go to sleep after a time, t seconds. Before that time it will transmit packets and occupies bandwidth. So, at this time only a few nodes (synchronizers) can utilize the bandwidth and give a lower throughput.

We run the simulation among randomly generated 20 nodes for both IEEE802.11 and SMAC. Here, we use cbrgen.tcl script built in NS-2.35 for creating maximum 8 random connections among nodes.

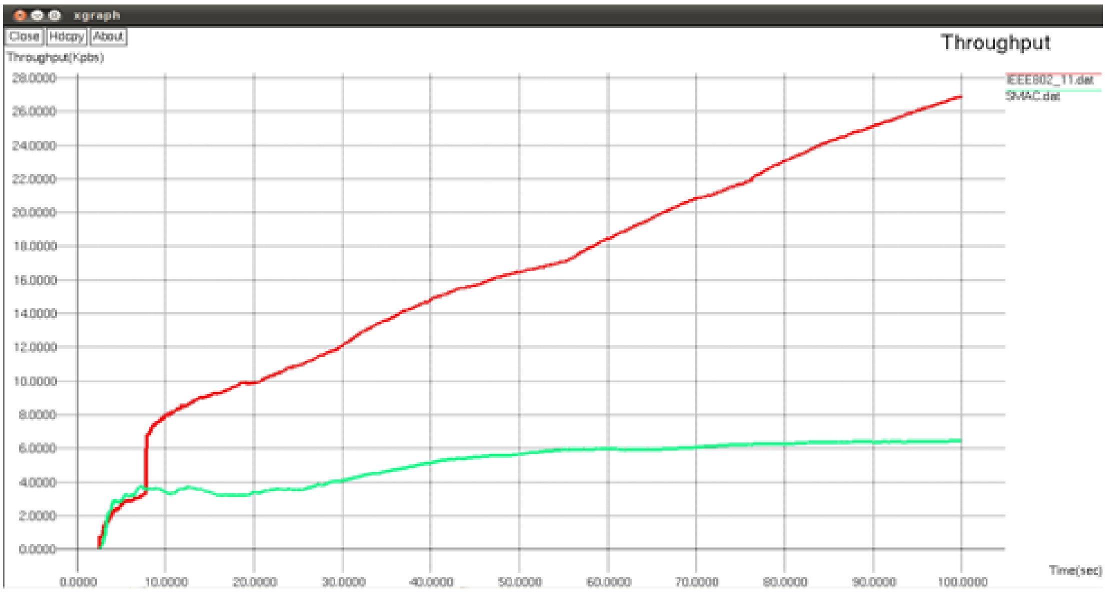


Figure 2.9: End-to-End throughput of IEEE802.11(red) and SMAC for 20 nodes(green)

By using a limited connection among a small no. of nodes we can get higher throughput in SMAC than IEEE802.11. By using the same random connection, used in above throughput measure and 10 randomly generated nodes we can measure the throughput shown in figure 2.10.

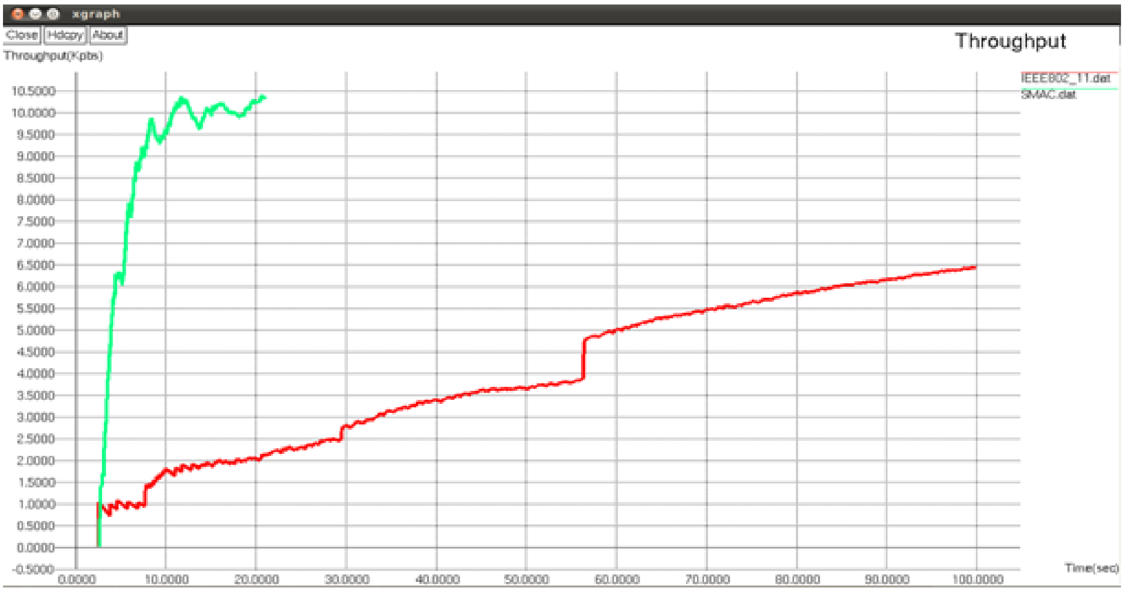


Figure 2.10: End-to-End throughput of IEEE802.11(red) and SMAC for 10 nodes(green)

From figure 2.10, we see that for a small no. of nodes (10) SMAC gives higher throughput than IEEE802.11 but we also see that approximately at time 20.3 sec simulation for SMAC becomes stop. This happens because nodes in SMAC update their scheduling time each time of sleeping & listening. During running simulation for SMAC an error message is shown that  Couldnt schedule with timer which means that scheduling time of nodes exceed the time of simulation time, so that it is not possible for further transmission between nodes as they cant schedule their synchronized period.

## 2.7 Summary

In this chapter, we studied routing protocols and the classification of routing protocol. We have given the definition of different types routing protocols. Our main focus was on LEACH and Mobile Ad-hoc Network protocol DSDV, AODV, DSR. We run the simulation to do the comparison between DSDV, AODV and DSR on different matrices. We have plotted the results which we have got from the simulation and describe the interesting observations we have found.

# CHAPTER 3
# ROUTING PROTOCOL

## 3.1 General

A routing protocol is a protocol that specifies how routers communicate with each other to disseminate information that allows them to select routes between any two nodes on a network. Typically, each router has a prior knowledge only of its immediate neighbors. A routing protocol shares this information so that routers have knowledge of the network topology at large. Wireless sensor network is one of the most considered factors of todays world. Due to severe energy constraint of densely deployed network sensor nodes, to maintain network functions and management different routing techniques are needed. There are many techniques available for routing in wireless sensor network. Traditional techniques include- Flooding and Gossiping. In flooding a given node broadcasts data and control packets that it has received to the rest of the nodes in the network. This process repeats until the destination node is reached. Note that this technique does not take into account the energy constraint imposed by WSNs. As a result, when used for data routing in WSNs, it leads to the problems such as implosion which refers that-As flooding is a blind technique, duplicating packets may keep circulate in the network, and hence sensors will receive those duplicate packet. To overcome this problem gossiping was introduced where a sensor would select randomly one of its neighbors and send the received packet to it. The same process repeats until all sensors receive this packet. Using gossiping, a given sensor would receive only one copy of a packet being sent. While gossiping tackles the implosion problem, there is a significant delay for a packet to reach all sensors in a network. Furthermore, these inconveniences are highlighted when the number of nodes in the network increases [16]. In this chapter we have discussed about different routing protocols (LEACH, DSDV, AODV, DSR), their significance in wireless sensor network and analyze the performance of some routing protocol using NS-2.

## 3.2 Design issues for Routing Protocol

The design of routing protocols in WSNs is influenced by many challenging factors. These factors must be overcome before efficient communication can be achieved in WSNs. In the following, we summarize some of the routing challenges and design issues that affect routing process in WSNs.

### Node deployment

Node deployment in WSNs is application dependent and affects the performance of the routing protocol. The deployment can be either deterministic or randomized. The sensors are manually placed and data is routed through pre-determined paths in deterministic deployment. However, in random node deployment, the sensor nodes are scattered randomly creating an infrastructure in an ad hoc manner. If the resultant distribution of nodes is not uniform, optimal clustering becomes necessary to allow connectivity and enable energy efficient network operation. Inter-sensor communication is normally within short transmission ranges due to energy and bandwidth limitations. Therefore, it is most likely that a route will consist of multiple wireless hops.

### Power Consumption

Since the transmission power of a wireless radio is proportional to distance squared or even higher order in the presence of obstacles, multi-hop routing will consume less energy than direct communication. However, multi-hop routing introduces significant overhead for topology management and medium access control. Direct routing would perform well enough if all the nodes were very close to the sink. Sensor nodes are equipped with limited power source (¡0.5 Ah 1.2V).Node lifetime is strongly dependent on its battery lifetime.

### Data Delivery Models

Data delivery models determine when the data collected by the node has to be delivered. Depending on the application of the sensor network, the data delivery model to the sink can be Continuous, Event-driven, Query-driven and Hybrid [17]. In the continuous delivery model, each sensor sends data periodically. In event-driven models, the transmission of data is triggered when an event occurs. In query driven models, the transmission of data is triggered when query is generated by the sink. Some networks apply a hybrid model using a combination of continuous, event-driven and query-driven data delivery.

### Node/Link Heterogeneity

Although many applications of wireless sensor network rely on homogenous nodes, the introduction of different kinds of sensors could bring significant benefits. The use of nodes with different processors, transceivers, power units or sensing components may improve the characteristics of the network. Among other, the scalability of the network, the energy drainage or the bandwidths are potential candidates to benefit from the heterogeneity of nodes. For example, hierarchical protocols designate a cluster-head node different from the normal sensors. These cluster-heads can be chosen from the deployed sensors or can be more powerful than other sensor nodes in terms of energy, bandwidth, and memory. Hence, the burden of transmission to the BS is handled by the set of cluster-heads.

### Fault Tolerance

Some sensor nodes may fail or be blocked due to lack of power, physical damage, or environmental interference. The failure of sensor nodes should not affect the overall task of the sensor network. If many nodes fail, MAC and routing protocols must accommodate formation of new links and routes to the data collection base stations. This may require actively adjusting transmit powers and signaling rates on the existing links to reduce energy consumption, or rerouting packets through regions of the network where more energy is available. Therefore, multiple levels of redundancy maybe needed in a fault-tolerant sensor network.

### Scalability

The number of sensor nodes deployed in the sensing area may be in the order of hundreds or thousands, or more. Any routing scheme must be able to work with this huge number of sensor nodes. In addition, sensor network routing protocols should be scalable enough to respond to events in the environment. Until an event occurs, most of the sensors can remain in the sleep state, with data from the few remaining sensors providing a coarse quality.

### Network Dynamics

Most of the network architectures assume that sensor nodes are stationary. However, mobility of either BSs or sensor nodes is sometimes necessary in many applications. Routing messages from or to moving nodes is more challenging since route stability becomes an important issue, in addition to energy, bandwidth etc. Moreover, the sensed phenomenon can be either dynamic or static depending on the application, e.g., it is dynamic in a target detection/tracking application, while it is static in forest monitoring for early fire prevention. Monitoring static events allows the network to work in a reactive mode, simply generating traffic when reporting. Dynamic events in most applications require periodic reporting and consequently generate significant traffic to be routed to the BS.

### Transmission Media

In a multi-hop sensor network, communicating nodes are linked by a wireless medium. The traditional problems associated with a wireless channel (e.g., fading, high error rate) may also affect the operation of the sensor network. In general, the required bandwidth of sensor data will be low, on the order of 1-100 kb/s. Related to the transmission media is the design of medium access control (MAC). One approach of MAC design for sensor networks is to use TDMA based protocols that conserve more energy compared to contention based protocols like CSMA (e.g., IEEE 802.11). Bluetooth technology can also be used.

### Connectivity

High node density in sensor networks precludes them from being completely isolated from each other. Therefore, sensor nodes are expected to be highly connected. This, however, may not prevent the network topology from being variable and the network size from being shrinking due to sensor node failures. In addition, connectivity depends on the, possibly random, distribution of nodes.

### Operating Environment:

In WSNs, each sensor node obtains a certain view of the environment. A given sensor's view of the environment is limited both in range and in accuracy; it can only cover a limited physical area of the environment. Hence, area coverage is also an important design parameter in WSNs.

### Data Aggregation/Fusion

Since sensor nodes might generate significant redundant data, similar packets from multiple nodes can be aggregated so that the number of transmissions would be reduced. Data aggregation is the combination of data from different sources by using functions such as suppression (eliminating duplicates), min, max and average [18]. As computation would be less energy consuming than communication, substantial energy savings can be obtained through data aggregation. This technique has been used to achieve energy efficiency and traffic optimization in a number of routing protocols.

### Quality Of Service (Q o S)

The quality of service means the quality service required by the application, it could be the length of life time, the data reliable, energy efficiency, and location-awareness, collaborative-processing. These factors will affect the selection of routing protocols for a particular application. In some applications (e.g. some military applications) the data should be delivered within a certain period of time from the moment it is sensed.

### Production Costs

Since the sensor networks consist of a large number of sensor nodes, the cost of a single node is very important to justify the overall cost of the networks and hence the cost of each sensor node has to be kept low.

### Data Latency And Overhead

These are considered as the important factors that influence routing protocol design. Data aggregation and multi-hop relays cause data latency. In addition, some routing protocols create excessive overheads to implement their algorithms, which are not suitable for serious energy constrained networks.

### Autonomy

The assumption of a dedicated unit that controls the radio and routing resources does not stand in wireless sensor networks as it could be an easy point of attack. Since there will not be any centralized entity to make the routing decision, the routing procedures are transferred to the network nodes.

## 3.3 Classification of Routing Protocol

To classify Routing protocols a large number of characteristics, e.g. the routing technique,the route establishment procedure, and the protocol operation, and the generated network structure can be used. According to network structure, the routing protocols are categorized into flat, hierarchical, and location-based protocols. The protocol operation can be multi-path-based, query-based, negotiation-based, QoS-based, or coherent-based [19].The way the routing protocols establish routes in the network classifies them into three different categories. The first one is called proactive. A proactive protocol sets up routing paths and states before there is a demand for routing traffic. Paths are maintained even there is no traffic flow at that time. In reactive routing protocol, routing actions are triggered when there is data to be sent and disseminated to other nodes. Here paths are setup on demand when queries are initiated. The last group consists of hybrid protocols which combine the ideas of reactive and proactive route establishment. Routing protocols are also classified based on whether they are destination-initiated (Dst-initiated) or source-initiated (Src-initiated). Fig. 3.1 gives an overview of the routing taxonomy.

### 3.3.1 Route Establishment based routing protocol

Routing protocols can follow different strategies to enable connectivity between the nodes in the networks. Different strategies effects on the performance and lifetime of the network differently.

Three different route establishment strategies are available.

First one is proactive routing. Here, each node has one or more tables that contain the latest information of the routes to any node in the network. The proactive protocols are not suitable for larger networks, as they need to maintain node entries for each and every node in the routing table of every node. This causes more overhead in the routing table leading to consumption of more bandwidth. Examples of such schemes are the conventional routing schemes, Destination Sequenced Distance Vector (DSDV). Second routing protocol is Reactive protocol. These protocols do not maintain routing information or routing activity at the network nodes if there is no communication. If a node wants to send a packet to another node then this protocol searches for the route in an on-demand manner and establishes the connection in order to transmit and receive the packet. The route discovery usually occurs by flooding the route request packets throughout the network. Example of reactive routing protocol is ad hoc on-demand distance vector routing (AODV). Hybrid protocols use both reactive and proactive mechanisms to maintain existing routes or to establish new routes. Example of hybrid routing protocol is dynamic source routing (DSR). The majority of hybrid protocols can be divided into two groups. The first group does not transmit any routing information if no route is required. Only the source and the destination of an active route

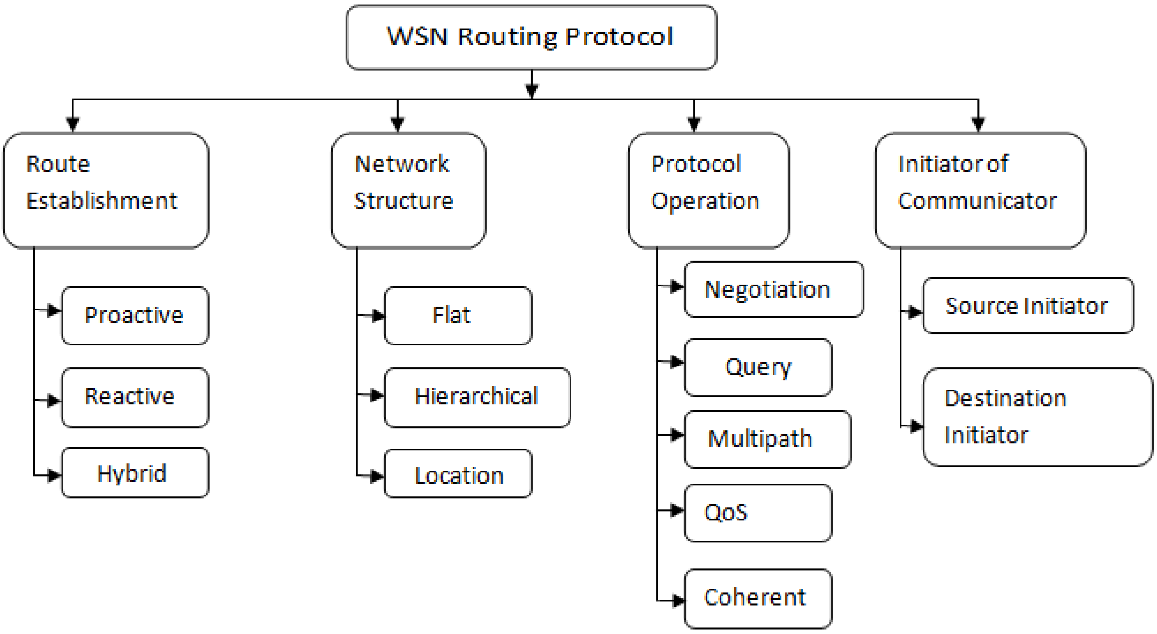


Figure 3.1: Taxonomy of Wireless Routing Protocol

periodically transmit routing information in order to keep the existing routes up-to-date. The advantage of this approach compared to reactive strategies is that the routing protocol is able to quickly detect link breaks in active routes. The second group of hybrid protocols uses proactive routing mechanisms for short range communication and reactive routing techniques for long range communication. Thus, they use periodic broadcast mechanisms to establish and maintain routes to nodes which are reachable within two or three hops. We will discuss elaborately about DSDV, AODV, DSR in section 3.4, 3.5 and 3.6 respectively.

### 3.3.2 Network Structure based routing protocol

To make efficient the network WSN contain characteristics and among them network structure is one of them. The most common network structures are flat, hierarchical, and location-based. In flat networks, each node typically plays the same role and sensor nodes collaborate together to perform the sensing task. Due to the large number of such nodes, it is not feasible to assign a global identifier to each node. This consideration has led to data centric routing, where the BS sends queries to certain regions and waits for data from the sensors located in the selected regions [20]. The main objective of hierarchical routing is to reduce energy consumption by classifying nodes into clusters. In each cluster, a node is selected as the leader or the cluster head. The different schemes for hierarchical routings mainly differ in how the cluster head is selected and how the nodes behave in the inter and intra-cluster domain [21].In section 3.3 we will discuss about LEACH elaborately which is a hierarchical routing protocol. Location-based protocols follow a different approach to structure the network. Nodes that are placed within a certain area are grouped instead of using a unique address for each node. Therefore, the networks scale with their size and not with the number of nodes.

### 3.3.3 Protocol Operation

Routing protocols can be also categorized with respect to their protocol operation. This kind of categorization has the advantage of being more application oriented compared to the two previously discussed taxonomies. In the following, the protocols are distinguished in negotiation-based and query-based protocols. Moreover, the protocols are grouped whether they offer multi-path or QoS support and depending on the used data processing technique. However, the protocol-based classification is not as strict as the other classifications. Thus, a routing protocol may fit into more than one category.

### 3.3.4 Initiator of Communicator

Routing protocols are also classified into destination-initiated (Dst-initiated)or source-initiated (Src-initiated). A source-initiated protocol sets up the routing paths upon the demand of the source node, and starting from the source node. Here source advertises the data when available and initiates the data delivery. A destination initiated protocol, on the other hand, initiates path setup from a destination node [22].

## 3.4 Low-energy adaptive clustering hierarchy(LEACH)

In WSN, LEACH is used for the following reasons-

- We need network protocol such as LEACH is due to the fact that a node in the network is no longer useful when its battery dies
- This protocol allows us to space out the lifespan of the nodes, allowing it to do only

the minimum work it needs to transmit data.

LEACH performs self-organizing and re-clustering functions for every round. In LEACH routing protocol, sensor nodes are formed into clusters, one of the node acts as a cluster-head and other act as member of the cluster. To communicate with the sink only cluster-head is involved and member nodes use cluster-head as intermediate router in case of communication to sink. Cluster-head collects the data from all the nodes, aggregate the data and route meaningful compressed information to sink. Because of these cluster-head dissipates more energy and if it remains cluster-head permanently it will die quickly as happened in case of static clustering. LEACH tackles this problem by randomized rotation of cluster-head to save the battery of individual node. In this ways LEACH maximize life time of network nodes and also reduce the energy dissipation by compressing the date before transmitting to cluster-head [23]. Optimal number of cluster heads is estimated to be 5% of the total number of nodes. LEACH routing protocol operations is based on rounds, where each round normally consists of two phases.

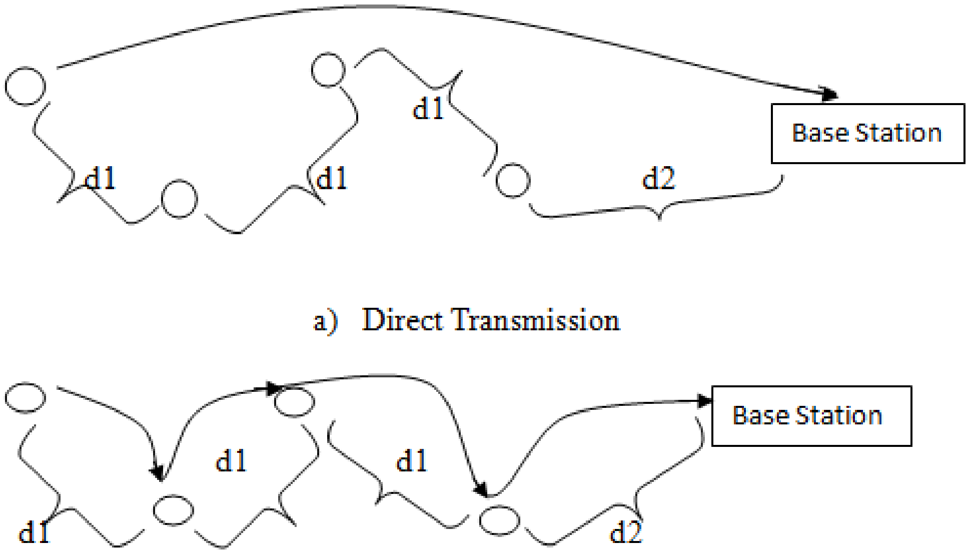
1. **Setup phase:** In setup phase cluster-head and cluster are created. Whole network nodes are divided into multiple clusters. Some nodes elect themselves as a cluster-head independently from other nodes. These nodes elect themselves on behalf Suggested percentage P and its previous record as cluster-head. Nodes which were not cluster-head in previous 1/p rounds generate a number between 0 to 1 and if it is less then threshold T(n) then nodes become cluster-head. Threshold value is set through this formula.

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

   Where G is set of nodes that have not been cluster-head in previous 1/p rounds, P= suggested percentage of cluster-head, r =is current round. The node becomes cluster-head in current round, it will be cluster-head after next 1/p rounds [23]. This indicates that every node will serve as a cluster-head equally and energy dissipation will be uniform throughout the network. Elected cluster-head broadcasts its status using CSMA MAC protocol.

2. **Steady state phase:** It starts when clusters have been created. In this phase nodes communicate to cluster-head during allocated time slots otherwise nodes keep sleeping. Due to this attribute LEACH minimize energy dissipation and extend battery life of all individual nodes. When data from all nodes of cluster have been received to cluster-head.it will aggregate, compress and transmit to sink. The steady state phase is longer than setup phase.

By Fig 3.2 we can see how LEACH minimizes the energy.



a) Direct Transmission

b) Minimum transmission energy

Figure 3.2: Energy Calculation for Direct transmission and Minimum Transmission

- The amount of energy used in figure (a) can be modeled by this formula

$$\sum k(3d1+d2)^2$$

- Whereas the amount of energy used in figure (b) uses this formula:

$$\sum k(3d1^2+d2^2)$$

Where k is the length of data and d1 is the distance between the sensor nodes, d2 is the distance between sensor node and Base station. LEACH uses fig-3.2(b) approach. Using the following feature LEACH reduces this energy dissipation [23]-

1. Reducing the number of transmission to sink using cluster-head.

2. Reducing the date to be transmitted through compression technique.
3. LEACH Increase the life time of all nodes through randomizing rotation being as cluster-head.
4. LEACH allows non-cluster-head nodes to keep sleeping except specific time duration. 5. In LEACH routing protocol nodes die randomly and dynamic clustering enhance network lifetime. 6. LEACH routing protocol makes wireless sensor network scalable and robust. Although LEACH is able to increase the network lifetime, there are still a number of issues about the assumptions used in this protocol [20].

1. LEACH assumes that all nodes can transmit with enough power to reach the BS if needed and that each node has computational power to support different MAC protocols. Therefore, it is not applicable to networks deployed in large regions.

2. It also assumes that nodes always have data to send and nodes located close to each other have correlated data. It is not obvious how the number of predetermined Cluster Heads CH (p) is going to be uniformly distributed throughout the network. Therefore, there is a possibility that the elected CHs will be concentrated in one part of the network. Hence, some nodes will not have any CHs in their vicinity. Furthermore, the idea of dynamic clustering brings extra overhead (head changes, advertisements, etc.), which may diminish the gain in energy consumption.

3. Finally, the protocol assumes that all nodes begin with the same amount of energy for each node. The protocol should be extended to account for non-uniform energy nodes, i.e., use energy-based threshold.

### 3.4.1 Power-Efficient Gathering in Sensor Information Systems (PEGASIS)

Lindsey & Ragavendra (2002) proposes an enhancement over the LEACH protocol called Power-Efficient Gathering in Sensor Information Systems (PEGASIS). It is chain based protocol. PEGASIS has two main objectives. They are-

1. To increase the lifetime of each node by using collaborative techniques.

2. To allow only local coordination between nodes that are close together so that the bandwidth consumed in communication is reduced (Lindsey et al., 2001).

Unlike LEACH, PEGASIS avoids cluster formation and uses only one node in a chain to transmit to the BS instead of multiple nodes. To locate the closest neighbor node in PEGASIS, each node uses the signal strength to measure the distance to all neighboring nodes and then adjusts the signal strength so that only one node can be heard [24]. Here each sensor node transmits and receives from a neighbor and only one node is selected from that chain to transmit to the base station (sink). The data is gathered and moves from node to node, aggregated and eventually sent to the base station. The chain construction is performed in a greedy fashion.

The structure of PEGASIS looks like the Fig 3.3 PEGASIS has an advantage over LEACH because it avoids clustering overhead, but PEGASIS still has some overheads [24]. They are-

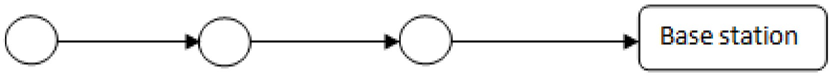


Figure 3.3: Chain construction using greedy algorithm

1. It requires dynamic topology adjustment as the energy status information of each node should be known to determine alternate routing path for data communication.

2. It assumes that each sensor node has the potential to directly communicate with the BS which conflicts the practical implementation.

3. It assumes that all nodes maintain a complete database of the location of all other nodes in the network. The method by which the node locations are obtained is not outlined.

4. It assumes that all sensor nodes have the same level of energy and are likely to die at the same time.

5. It introduces excessive delay for distant nodes in the chain to communicate to BS. The single leader in this protocol can become a bottleneck.

6. Finally, although in most scenarios sensors will be fixed or immobile as assumed in PEGASIS, some sensors may be allowed to move and hence affect the protocol functionality.

### 3.4.2 Variations of LEACH

Many protocols have been derived from LEACH with some modifications and applying advance routing techniques. A description about these protocols can be given shortly like below-

#### 3.4.2.1 Solar-aware Low Energy Adaptive Clustering Hierarchy (sLEACH)

In this protocol, cluster-heads are selected by their solar-status because some nodes are facilitated by solar-power. It is of two types. They are-

1. **Solar-aware Centralized LEACH:**In solar-aware Centralized LEACH cluster head are selected by Base station with help of improved Central control algorithm. As each node sends solar status along with the energy to the base station, it is then up to the base station to select the cluster-head with the highest energy. Performance of sensor network depends on number of solar aware nodes as well as sunDuration proportionally. In times of smaller sunDuration cluster-head handover is done.

2. **Solar-aware Distributed LEACH:**In Solar-aware Distributed LEACH choosing preference of cluster-head is given to solar-driven nodes.

#### 3.4.2.2 Multi-hop LEACH

When the distance between the cluster-head and base station increases, the energy dissipation of cluster-head is not affordable. Thats why to overcome this problem, multi-hop LEACH was introduced. Like LEACH, in Multi-Hop LEACH some nodes elect themselves as cluster-heads and other nodes associate themselves with elected cluster-head to complete cluster formation in setup phase. In steady state phase cluster-head collect data from all nodes of its cluster and transmit data directly or through other cluster-head to Base station after aggregation. Figure 3.4 (b) describes Multi-Hop LEACH communication architecture. Randomized rotation of cluster-head is similar to LEACH. Multi-Hop LEACH selects best path with minimum hop-count between first cluster-head and base station.

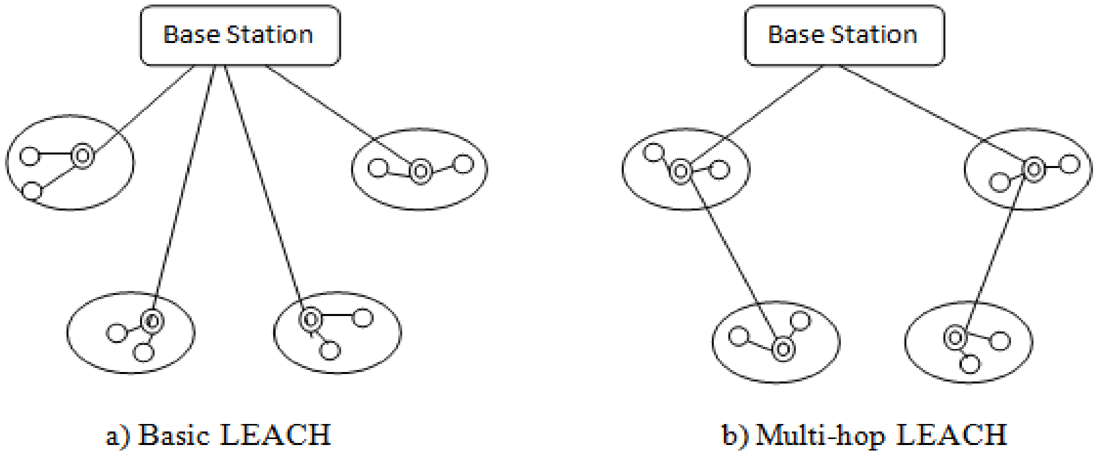


Figure 3.4: Difference between the Structure of Basic and Multi-hop LEACH

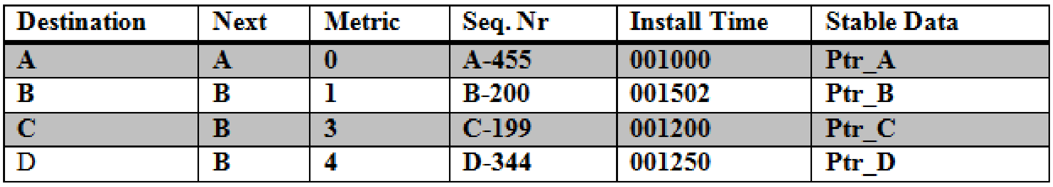
#### 3.4.2.3 Mobile-LEACH (M-LEACH)

M-LEACH allows mobility of non-cluster-head nodes and cluster-head during the setup and steady state phase. In M-LEACH cluster-heads are elected on the basis of attenuation model and mobility speed. Node with minimum mobility and lowest attenuation power is selected as cluster-head I M-LEACH. In steady state phase, if nodes move away from cluster-head or cluster-head moves away from its member nodes then other cluster-head becomes suitable for member nodes. It results into inefficient clustering formation. To deal this problem M-LEACH provides handover mechanism for nodes to switch on to new cluster-head.

## 3.5 Destination-Sequenced Distance-Vector Routing (DSDV)

The DSDV routing algorithm [Perkins 1994] is built on top of Bellman-Ford routing algorithm. There are two routing algorithms available. First one is Link-State algorithm each node maintains a view of the network topology and second one is Distance-Vector algorithm where every node maintains the distance of each destination. Distance-Vector algorithm is not suited for ad-hoc networks because it causes loops and count-to-infinity problem. The solution is to using DSDV protocol which is Destination Based and contains no global view of topology. In DSDV each node maintains routing information for all known destinations and routing information must be updated periodically. The route labeled with the highest sequence number is always used. This also helps in identifying the stale routes from the new ones, thereby avoiding the formation of loops. DSDV allows fast reaction to topology

Table 3.1: Table entries of DSDV routing protocol



| Destination | Next | Metric | Seq. Nr | Install Time | Stable Data |
|---|---|---|---|---|---|
| A | A | 0 | A-455 | 001000 | Ptr_A |
| B | B | 1 | B-200 | 001502 | Ptr_B |
| C | B | 3 | C-199 | 001200 | Ptr_C |
| D | B | 4 | D-344 | 001250 | Ptr_D |

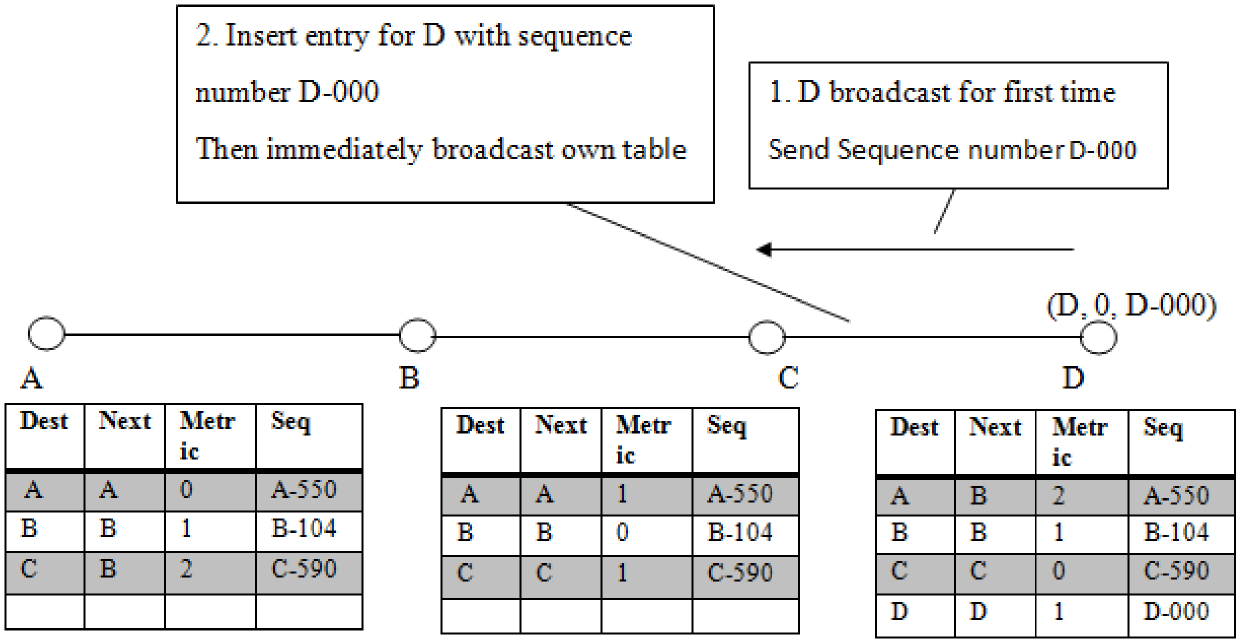
changes. It makes immediate route advertisement on significant changes in routing table but waits with advertising of unstable routes (damping fluctuations). DSDV routing table looks like below:

- Sequence number originated from destination. Ensures loop freeness.
- Install Time when entry was made (used to delete stale entries from table)
- Stable Data Pointer to a table holding information on how stable a route is. Used to damp fluctuations in network.

In DSDV each node advertises to each neighbor own routing information. These information are Destination Address, Metric (Number of Hops to Destination) and Destination Sequence Number. DSDV maintains some rules to set sequence number information. On each advertisement DSDV increases own destination sequence number (use only even numbers). If a node is no more reachable (timeout) then DSDV increases sequence number of this node by 1 (odd sequence number) and set metric = .To minimize the traffic generated, there are two types of packets in the system. One is known as full dump, which is a packet that carries all the information about a change. However, at the time of occasional movement, another type of packet called incremental will be used, which will carry just the changes, thereby, increasing the overall efficiency of the system.

### 3.5.1 DSDV (fluctuations and solution of fluctuations)

If in case of fig 3.9 we suppose that entry for D in A is [D, C, 14, D-100] and D makes Broadcast with Seq. Nr. D-102. If A receives from P Update (D, 15, D-102) then Entry for D in A, will be [D, B, 15, D-102], so A must propagate this route immediately. Again if A receives from B Update (D, 14, D-102) then Entry for D in A, will be [D, C, 14, D-102], so


2. Insert entry for D with sequence number D-000
Then immediately broadcast own table
1. D broadcast for first time
Send Sequence number D-000
(D, 0, D-000)
A
B
C
D

| Dest | Next | Metric | Seq |
|---|---|---|---|
| A | A | 0 | A-550 |
| B | B | 1 | B-104 |
| C | B | 2 | C-590 |
| | | | |

| Dest | Next | Metric | Seq |
|---|---|---|---|
| A | A | 1 | A-550 |
| B | B | 0 | B-104 |
| C | C | 1 | C-590 |
| | | | |

| Dest | Next | Metric | Seq |
|---|---|---|---|
| A | B | 2 | A-550 |
| B | B | 1 | B-104 |
| C | C | 0 | C-590 |
| D | D | 1 | D-000 |

Figure 3.5: DSDV (new node)

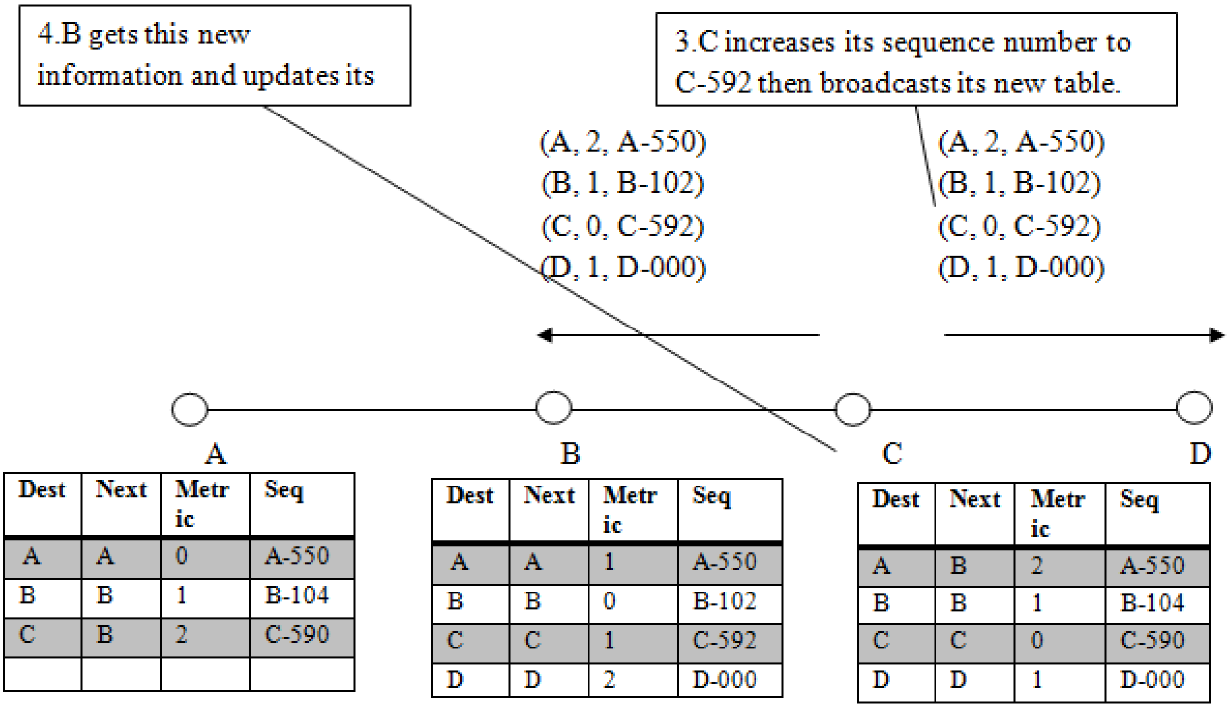
A must propagate this route immediately.

This can happen every time D or any other node does its broadcast and lead to unnecessary route advertisements in the network, so called fluctuations. To dump fluctuations some steps should be taken.

- Record last and avg. Settling Time of every Route in a separate table. (Stable Data)Settling Time = Time between arrival of first route and the best route with a given seq. nr.
- A still must update his routing table on the first arrival of a route with a newer seq. nr., but he can wait to advertising it. Time to wait is proposed to be 2*(avg. Settling Time).
- Like this, fluctuations in larger networks can be damped to avoid unnecessary advertisement, thus saving bandwidth.

**Advantages of DSDV protocol:**

- DSDV protocol guarantees loop free paths.
- Count to infinity problem is reduced in DSDV.


4.B gets this new information and updates its
3.C increases its sequence number to C-592 then broadcasts its new table.
(A, 2, A-550)
(B, 1, B-102)
(C, 0, C-592)
(D, 1, D-000)
(A, 2, A-550)
(B, 1, B-102)
(C, 0, C-592)
(D, 1, D-000)
A
B
C
D

| Dest | Next | Metric | Seq |
|---|---|---|---|
| A | A | 0 | A-550 |
| B | B | 1 | B-104 |
| C | B | 2 | C-590 |
| | | | |

| Dest | Next | Metric | Seq |
|---|---|---|---|
| A | A | 1 | A-550 |
| B | B | 0 | B-102 |
| C | C | 1 | C-592 |
| D | D | 2 | D-000 |

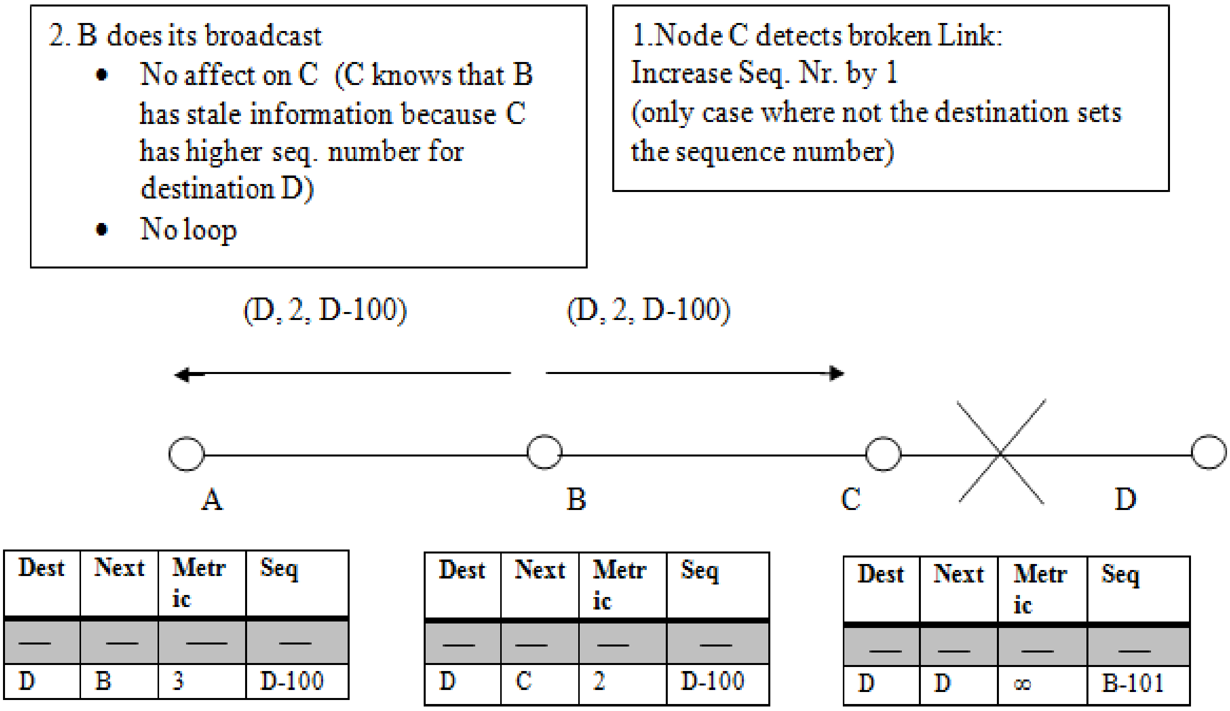
| Dest | Next | Metric | Seq |
|---|---|---|---|
| A | B | 2 | A-550 |
| B | B | 1 | B-104 |
| C | C | 0 | C-590 |
| D | D | 1 | D-000 |

Figure 3.6: DSDV (new node)contd.

- We can avoid extra traffic with incremental updates instead of full dump updates.
- Path Selection: DSDV maintains only the best path instead of maintaining multiple paths to every destination. With this, the amount of space in routing table is reduced.

**Limitations of DSDV protocol:**

- Wastage of bandwidth due to unnecessary advertising of routing information even if there is no change in the network topology.
- DSDV doesnt support Multi path Routing.
- It is difficult to determine a time delay for the advertisement of routes.
- It is difficult to maintain the routing tables advertisement for larger network. Each and every host in the network should maintain a routing table for advertising. But for larger network this would lead to overhead, which consumes more bandwidth.



2. B does its broadcast

- No affect on C (C knows that B has stale information because C has higher seq. number for destination D)
- No loop

1.Node C detects broken Link:
Increase Seq. Nr. by 1
(only case where not the destination sets the sequence number)

| Dest | Next | Metric | Seq |
|---|---|---|---|
| — | — | — | — |
| D | B | 3 | D-100 |

| Dest | Next | Metric | Seq |
|---|---|---|---|
| — | — | — | — |
| D | C | 2 | D-100 |

| Dest | Next | Metric | Seq |
|---|---|---|---|
| — | — | — | — |
| D | D | ∞ | B-101 |

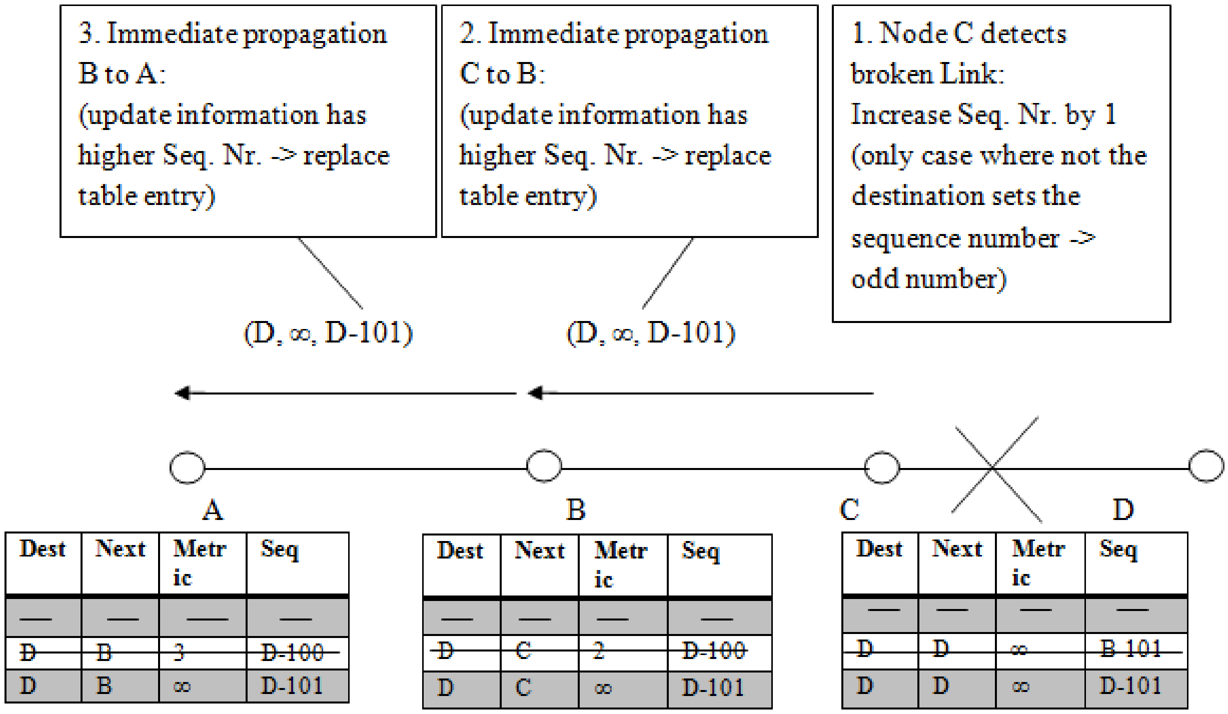
Figure 3.7: DSDV (no loops, no count to infinity)

## 3.6 Ad-hoc On-Demand Distance Vector (AODV) Routing

AODV is an on demand routing algorithm, meaning that it establishes paths only upon demand by source nodes. It maintains these paths as long as they are needed. Nodes that do not participate in active path neither maintain any routing information nor participate in any periodic routing table exchange. AODV established path based on route request- route reply mechanism.

### 3.6.1 Path Discovery Process

In order to discover the path, a route request message (RREQ) is broadcasted to all the neighbors which again continue to send the same to their neighbors, until the destination is reached. Every node maintains two counters: sequence number and broadcast-id in order to maintain loop-free and most recent route information. The Broadcast-id is incremented for every RREQ the source node initiates. If an intermediate node receives the same copy of request, it discards it without routing it further and if the neighboring nodes which receiving the RREQ has no route information about the destination then it will further broadcast RREQ packet in the network otherwise it will send answer by the route reply (RREP) packet



3. Immediate propagation B to A: (update information has higher Seq. Nr. -> replace table entry)

2. Immediate propagation C to B: (update information has higher Seq. Nr. -> replace table entry)

1. Node C detects broken Link: Increase Seq. Nr. by 1 (only case where not the destination sets the sequence number -> odd number)

A

| Dest | Next | Metric | Seq |
|---|---|---|---|
| — | — | — | — |
| ~~D~~ | ~~B~~ | ~~3~~ | ~~D-100~~ |
| D | B | ∞ | D-101 |

B

| Dest | Next | Metric | Seq |
|---|---|---|---|
| — | — | — | — |
| ~~D~~ | ~~C~~ | ~~2~~ | ~~D-100~~ |
| D | C | ∞ | D-101 |

C

| Dest | Next | Metric | Seq |
|---|---|---|---|
| — | — | — | — |
| ~~D~~ | ~~D~~ | ~~∞~~ | ~~B-101~~ |
| D | D | ∞ | D-101 |

Figure 3.8: DSDV (Immediate Advertisement)

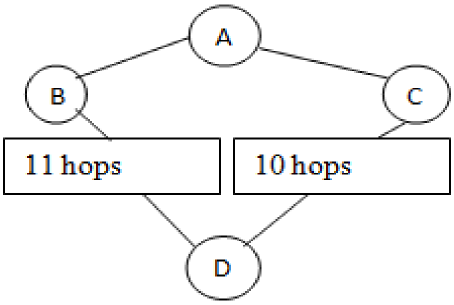
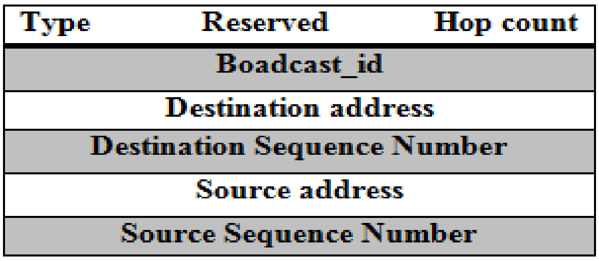


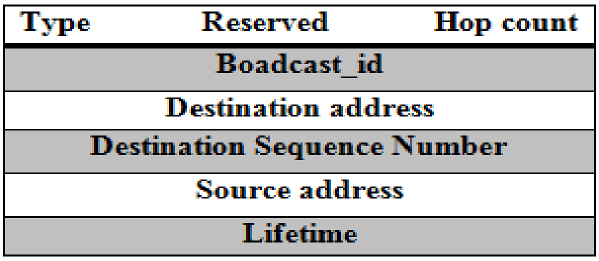
Figure 3.9: DSDV (Problem of Fluctuations)

to the sender from which RREQ is received. When a node forwards the RREQ message, it records the address of the neighbor from which it received the first copy of the broadcast packet, in order to maintain a reverse path to the source node. RREQ contains source address, source sequence number, broadcast_id, destination address, destination sequence number, and hop count as shown in Figure 3.10; broadcast_id uniquely identifies a RREQ, where broadcast_id is incremented when a new RREQ is issue by source. Similarly Figure 3.11 shows the structure of PREP. When the RREP reaches the source, the route is ready, and the initiator can use it. A neighbor that has communicated at least one packet during the past active timeout is considered active for this destination. An active entry in the routing



| Type | Reserved | Hop count |
|---|---|---|
| Boadcast_id | | |
| Destination address | | |
| Destination Sequence Number | | |
| Source address | | |
| Source Sequence Number | | |

Figure 3.10: Structure of an RREQ packet



| Type | Reserved | Hop count |
|---|---|---|
| Boadcast_id | | |
| Destination address | | |
| Destination Sequence Number | | |
| Source address | | |
| Lifetime | | |

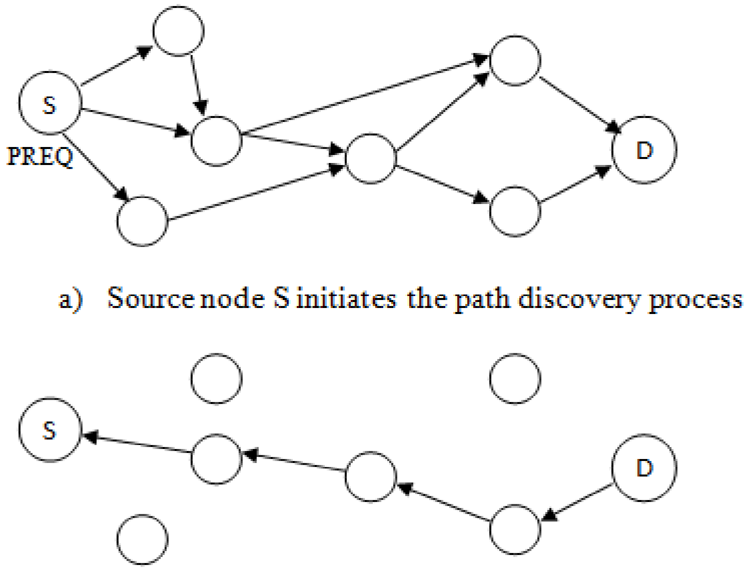
Figure 3.11: Structure of an RREP packet

table is an entry that uses an active neighbor. An active path is a path established with active routing table entries. A routing table entry expires if it has not been used recently. In this main content that AODV uses the route expiration technique, where a routing table entry expires within a specific period, after which a fresh route discovery must be initiated.

### 3.6.2 Maintaining Routes

The nodes participating in an active route are notified by RERR (Route ERROR) packets when the next-hop link is down. The Route error propagation in AODV is achieved by all nodes participating in the route. Each one of them forwards the RERR to its predecessors. Consequently, all routing tables must be updated after this process. Nodes launches the RERR message in three cases: 1- detection of a link break for the next hop of an active route in its own routing table, 2- getting a data packet intended to a node that does not have an active route, 3- receiving a RERR from a neighbor in relation to one or more active routes. AODV uses the lifetime field to determine the expiry time for an active route. It also is also used to define the deletion time for an invalid route.

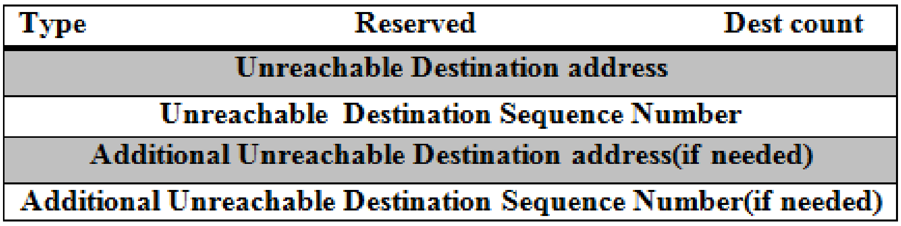
**Advantages of AODV protocol:**



a) Source node S initiates the path discovery process

b) A RREP packet is sent back to the source.

Figure 3.12: AODV Path Discovery Process



| Type | Reserved | Dest count |
|---|---|---|
| Unreachable Destination address | | |
| Unreachable Destination Sequence Number | | |
| Additional Unreachable Destination address(if needed) | | |
| Additional Unreachable Destination Sequence Number(if needed) | | |

Figure 3.13: Structure of an RERR packet

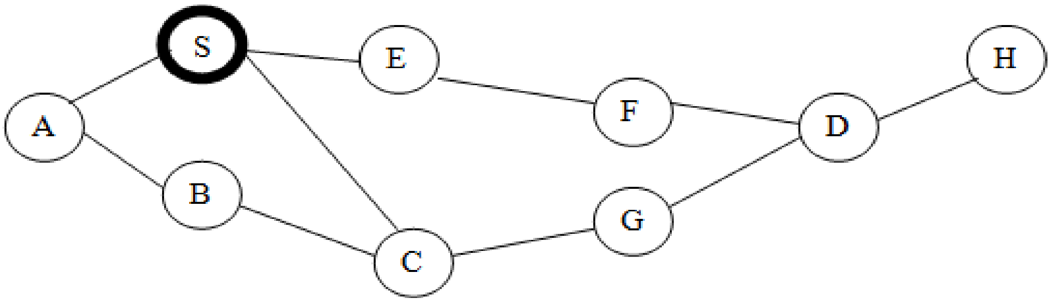
- Reduces memory requirements and needless duplications
- Responses quickly to link breakage in active routes
- Maintains loop-free routes by use of destination sequence numbers
- Minimizes need for broadcast
- Increases size of network depending on the number of nodes.
- AODV for IPv6 is specified, built, and works

**Limitations of AODV protocol:**

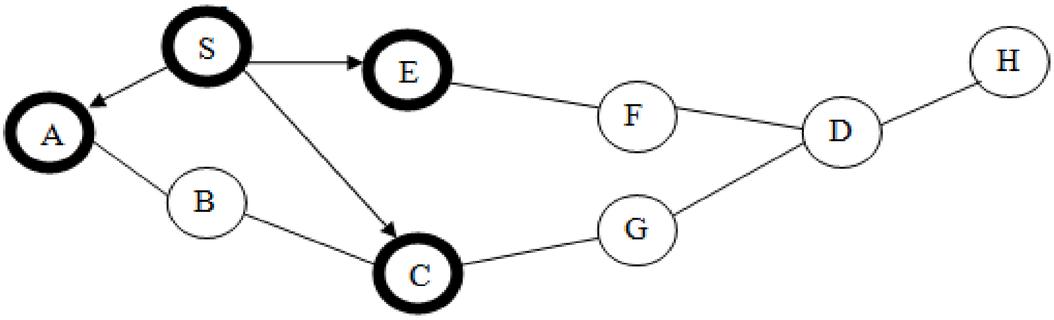
- The algorithm requires that the nodes in the broadcast medium can detect each others broadcasts.
- Overhead on bandwidth will be occurred when an RREQ travels from node to node in the process of discovering the route info on demand, it sets up the reverse path in itself with the addresses of all the nodes through which it is passing and it carries all this info all its way.
- AODV lacks an efficient route maintenance technique. The routing info is always obtained on demand, including for common case traffic.
- The messages can be misused for insider attacks including route disruption, route invasion, node isolation, and resource consumption.
- AODV is designed to support the shortest hop count metric. This metric favors long, low bandwidth links over short, high bandwidth links.
- AODV does not discover a route until a flow is initiated. This route discovery latency result can be high in large-scale mesh networks.

## 3.7 Dynamic Source Routing (DSR)

DSR maintains some mechanism for routing packets. Firstly, to send a packet to another host, the sender constructs a source route in the packets header, giving the address of each host in the network through which the packet should be forwarded in order to reach the destination host. The sender then transmits the packet according to the route. When a host receives a packet, if this host is not the final destination of the packet, it simply transmits the packet to the next hop identified in the source route in the packets header. Once the packet reaches its final destination, the packet is delivered to the network layer software on that host. Secondly, while sending packet from one host to another, if the route is not found, the sender may attempt to discover one using the route discovery mechanism. If host S is source host and D is destination host and S initiates a route discovery in absence of route. Source host S floods Route Request (RREQ). Each host appends own identifier when forwarding RREQ.

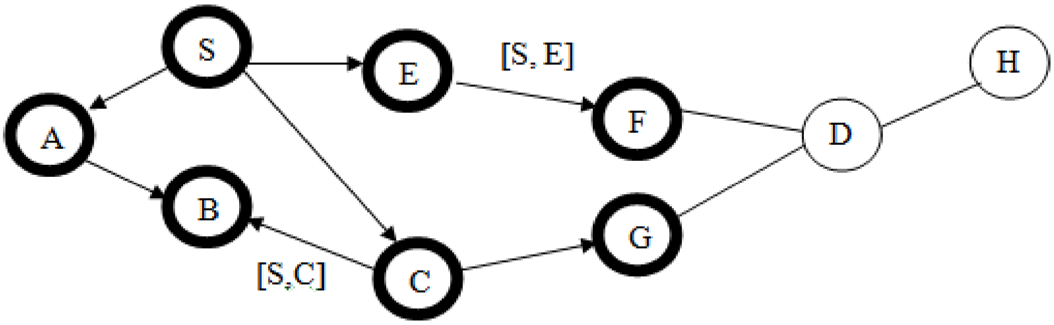


a) ◯ Represents a host that has received RREQ for D from S

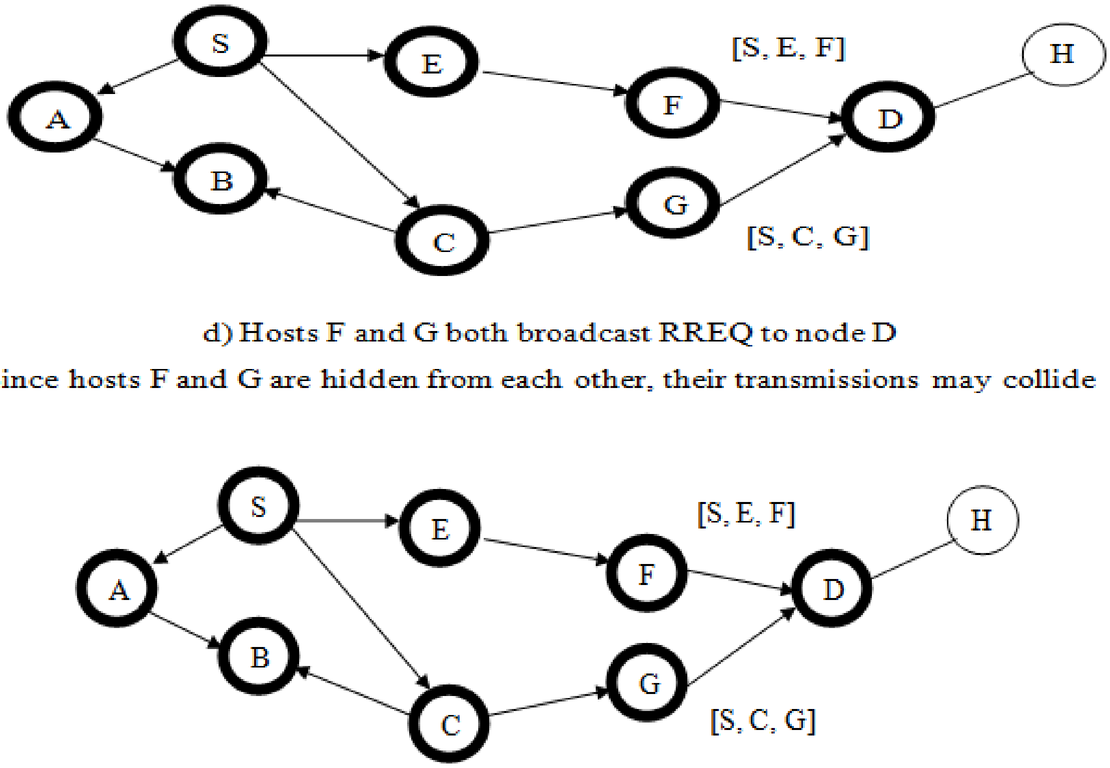


b) ← Represents transmission of RREQ

While waiting for the route discovery to complete, the host may continue normal processing and may send and receive packets with other hosts. The host may buffer the original packet in order to transmit it once the route is learned from route discovery, or it may discard the packet, relying on higher layer protocol software to retransmit the packet if needed. Destination D on receiving the first RREQ sends a Route Reply (RREP). RREP is sent on a route obtained by reversing the route appended to receive RREQ. RREP includes the route from S to D on which RREQ was received by host D. Thirdly, while a host is using any source route, it monitors the continued correct operation of that route. For example, if the sender, the destination, or any of the other hosts named hops along a route move out of



c) Host B receives packet RREQ from two neighbors: potential for collision



d) Hosts F and G both broadcast RREQ to node D
Since hosts F and G are hidden from each other, their transmissions may collide

e) Host D doesn't forward RREQ, because node D is the intended target of the rout discovery

Figure 3.14: Route discovery of DSR

wireless transmission range of the next or previous hop along the route, the route can no longer be used to reach the destination. A route will also no longer of work if any of the hosts along the route should fail or be powered off. This monitoring of the correct operation of a route in use we call route maintenance. When route maintenance detects a problem with a route in use, route discovery may be used again to discover a new, correct route to the destination.

Fourthly, each mobile host participating in the ad hoc network maintains a route cache in which it caches source routes that it has learned. When one host sends a packet to another host, the sender first checks its route cache for a source route to the destination. If a route is found, the sender uses this route to transmit the packet. When host S finds route [S, E, F,D] to host D, host S also learns route [S,E, F] to host F. When host G receives Route Request [S, C] destined for host, host G learns route [G, C, S] to host S. When host E forwards Route Reply RREP [S, E, F,D], host E learns route [E, F, D] to host D. But there is a problem of route cache and that is Stale caches may increase overheads.

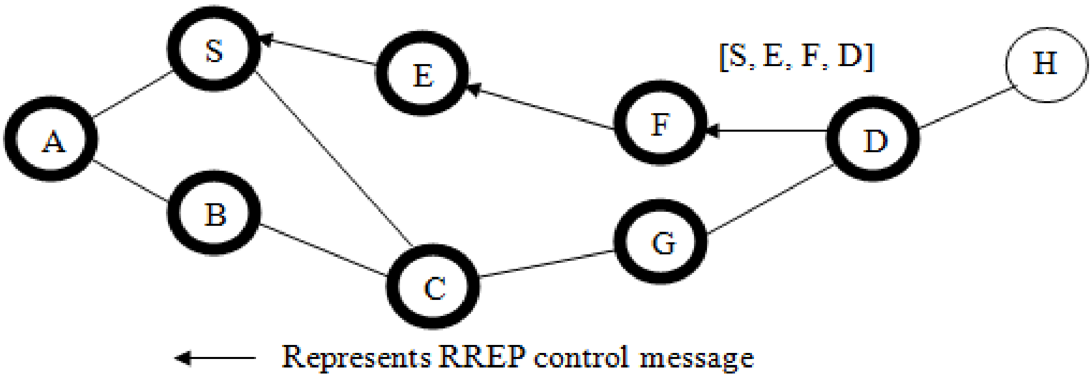


Figure 3.15: Route reply of DSR

**Advantages of DSR:**

- Routes maintained only between nodes who need to communicate
  - Reduces overhead of route maintenance
- Route caching can further reduce discovery overhead
- A single route discovery may yield many routes to the destination, due to intermediate nodes replying from caches

**Disadvantages of DSR:**

- Packet header size grows with route length due to source routing
- Flood of route requests may potentially reach all nodes in the network
- Potential collisions between route requests propagated by neighboring nodes
  - Insertion of random delays before forwarding RREQ
- Stale caches will lead to increased overhead
- Increased contention if too many route replies come back due to nodes replying using their local cache
  - Route Reply Storm problem

## 3.8 Comparison and Evaluation

In this section, we compare the three routing protocol: DSDV, AODV and DSR, discussed previously. We have used simulation results to compare the protocols.

### 3.8.1 Simulation Model

The experiments use a fixed number of packet sizes (512-bytes). The mobility model used is a radio propagation model. The field configurations used is 500m X 500mwith different number of nodes. Here, each packet starts its journey from a random location to a random destination with a randomly chosen speed. Simulations is run for 100 simulated seconds. Identical mobility and traffic scenarios are used across protocols to gather fair results.
The Wireless scenario for this experiment looks like:

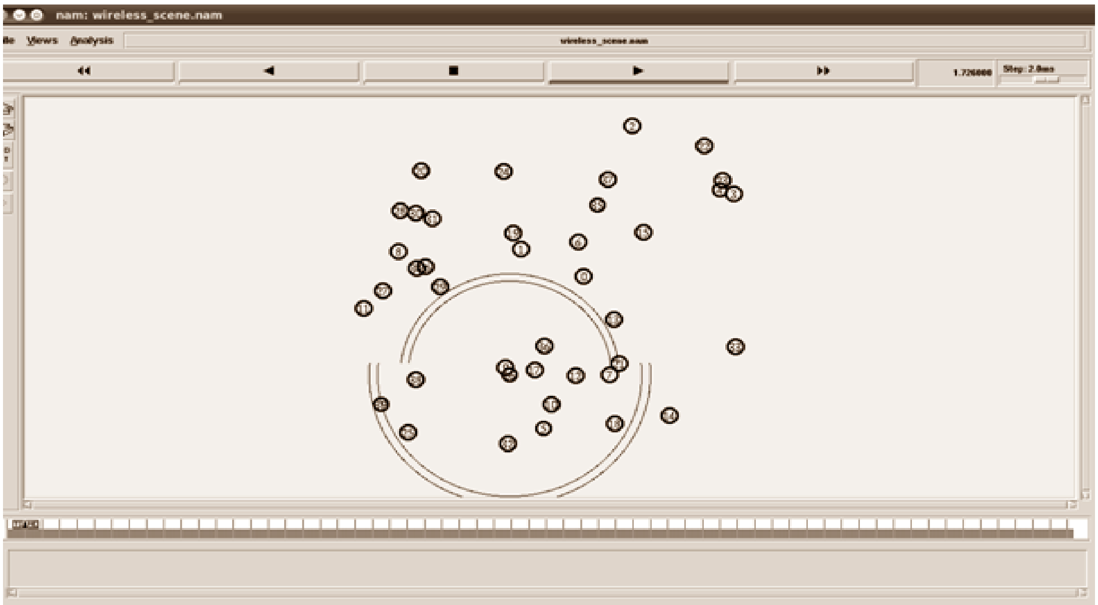


Figure 3.16: NAM output

### 3.8.2 Performance Metrics:

We focus on 3 performance metrics which are quantitatively measured. The performance metrics are important to measure the performance and activities that are running in NS-2 simulation. The performance metrics are:

1. **Packet delivery fractions (PDF):** the ratio of the data packets delivered to the destinations to those generated by the CBR sources. The PDF shows how successful a protocol performs delivering packets from source to destination. The higher for the value give use the better results. This metric characterizes both the completeness and correctness of the routing protocol also reliability of routing protocol by giving its effectiveness.

   Packet Delivery Ratio = (Packets received by the destination node)/(Packet sent)

2. **Throughput:** The ratio of the total amount of data that reaches a receiver from a sender to the time it takes for the receiver to get the last packet is referred to as throughput. It is expressed in bits per second or packets per second. Factors that affect throughput include frequent topology changes, unreliable communication, limited bandwidth and limited energy. A high throughput network is desirable.

**a) Packet Delivery Fraction** Figure3.17, 3.18, 3.19, 3.20 shows the Xgraph for AODV, DSDV and DSR with nodes 20, 30, 40 and 50 respectively where red curve is for DSDV, green one is for AODV and blue is for DSR. The X-axis of the graph indicates the time and the Y-axis shows the Packet Delivery Fraction. Based on these Figures it is shown than AODV perform better as we know when the number of nodes increases, nodes become more

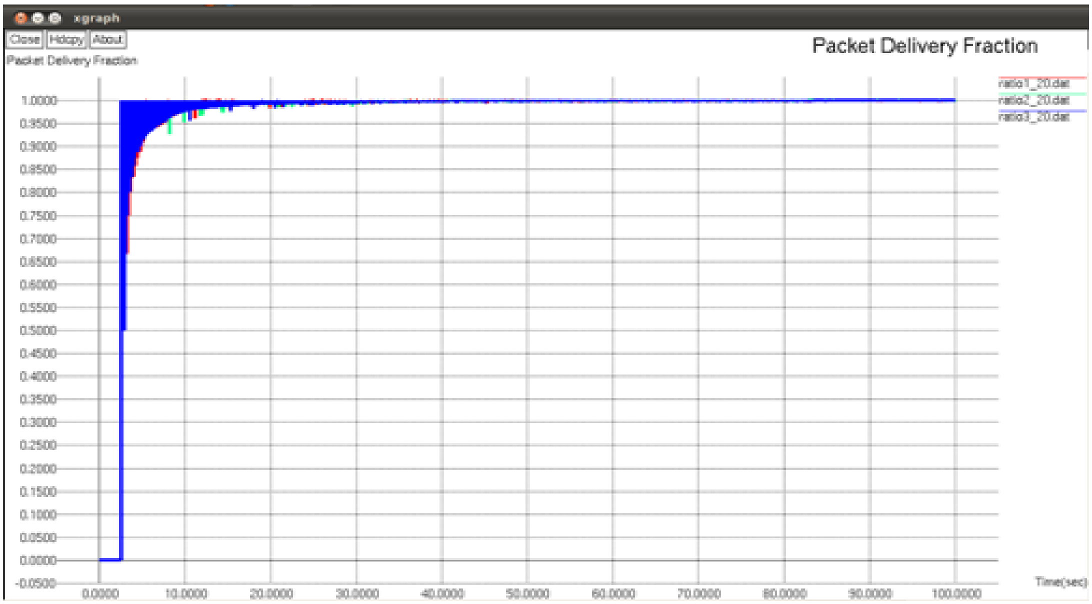


Figure 3.17: Packet Delivery Fraction of DSDV, AODV, DSR with 20 nodes

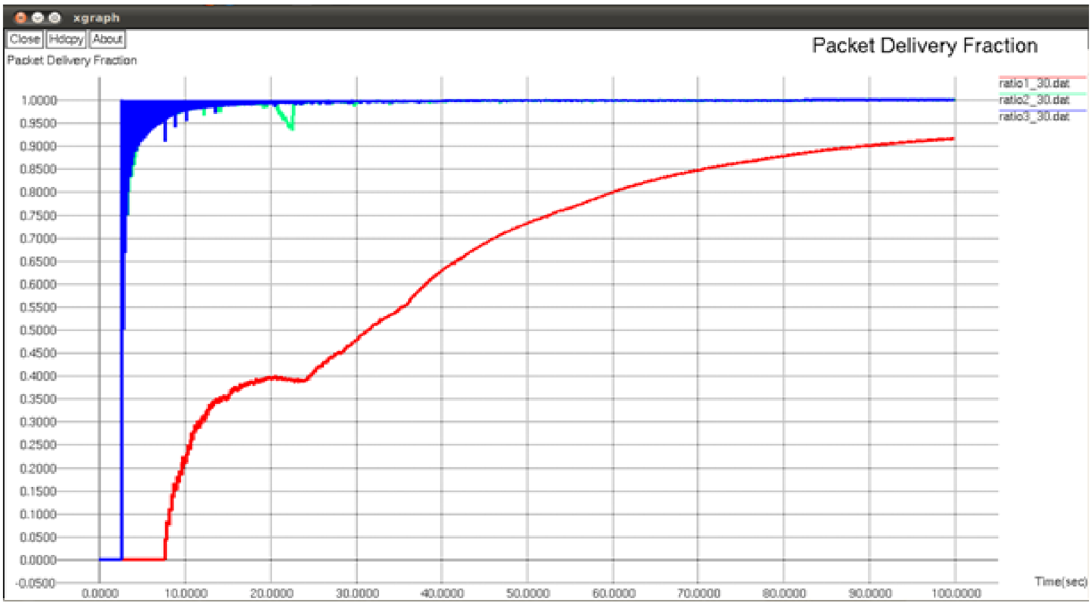


Figure 3.18: Packet Delivery Fraction of DSDV, AODV, DSR with 30 nodes

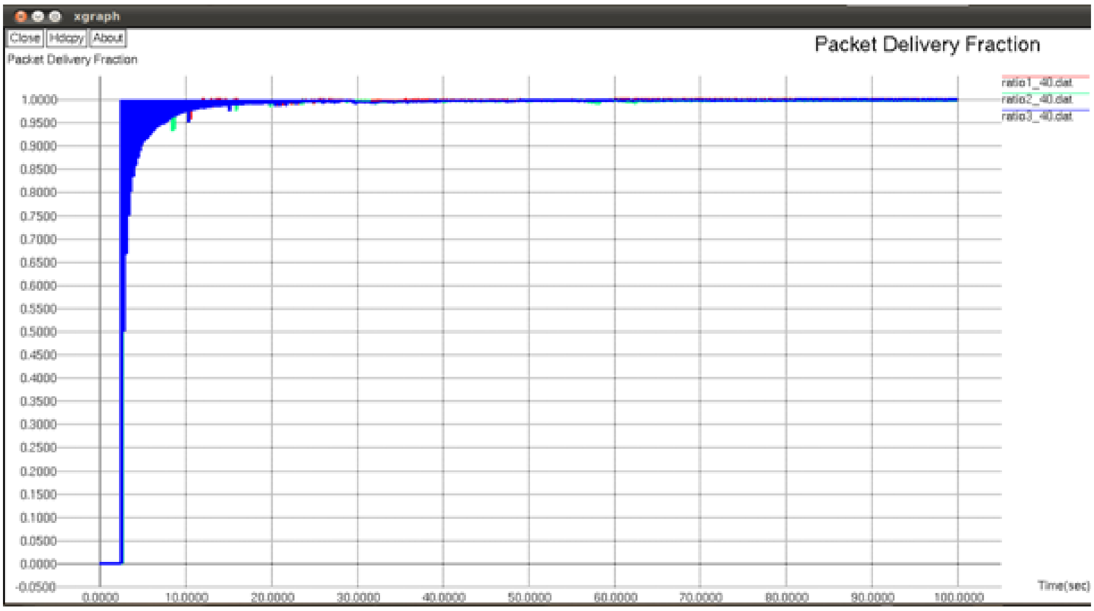


Figure 3.19: Packet Delivery Fraction of DSDV, AODV, DSR with 40 nodes

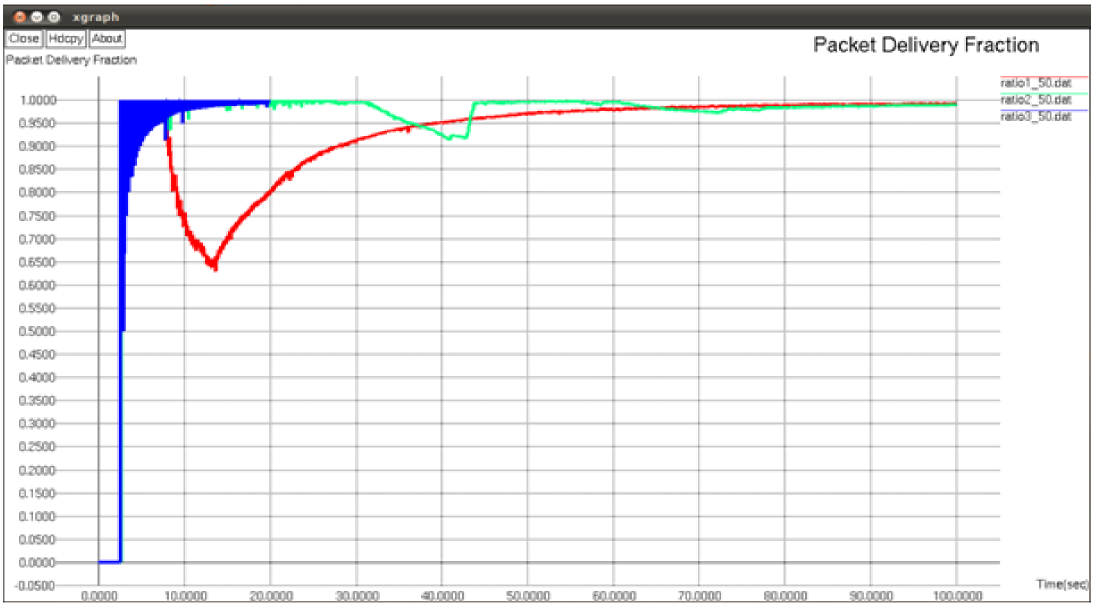


Figure 3.20: Packet Delivery Fraction of DSDV, AODV, DSR with 50 nodes

stationary which will lead to more stable path from source to destination in case of AODV. Thats why the output curve for AODV is high having slight change in all the plotting above. DSDV performance dropped as number of nodes increase because more packets dropped due to link breaks. Thats why the curve for DSDV fluctuates so much. In figure 3.17 it is high but in figure 3.18 it suddenly drops and increases with time, then again for node 40 it becomes high and for node 50 it drops slightly and starts to rise with the advancement of time. DSR curve in all the figures acts same as AODV.

**b) Throughput** The figure 3.21 shows the Xgraph for AODV, DSDV and DSR with connections 20 where red one is DSDV, green one is AODV and blue is for DSR. The X-axis of the graph indicates the time and the Y-axis shows the throughput. As we can clearly observe from the graph, the throughput of AODV and DSR is better than DSDV. There is a steady increase of throughput in case of AODV and DSR whereas in case of DSDV it is not the same. Thus in case of Throughput, AODV performs well when compared to DSDV. The figure 3.22 shows the Xgraph for AODV, DSDV and DSR with 30 connections. The X-axis of the graph indicates the time and the Y-axis shows the throughput. From this figure also we can see that throughput for AODV and DSR is better than DSDV. The reason for this difference is the wastage of bandwidth in case of DSDV due to unnecessary advertising of routing information even if the network is in idle mode. Thats why throughput for DSDV rises at time 0.0000 then gradually decreases with the advancement of time.

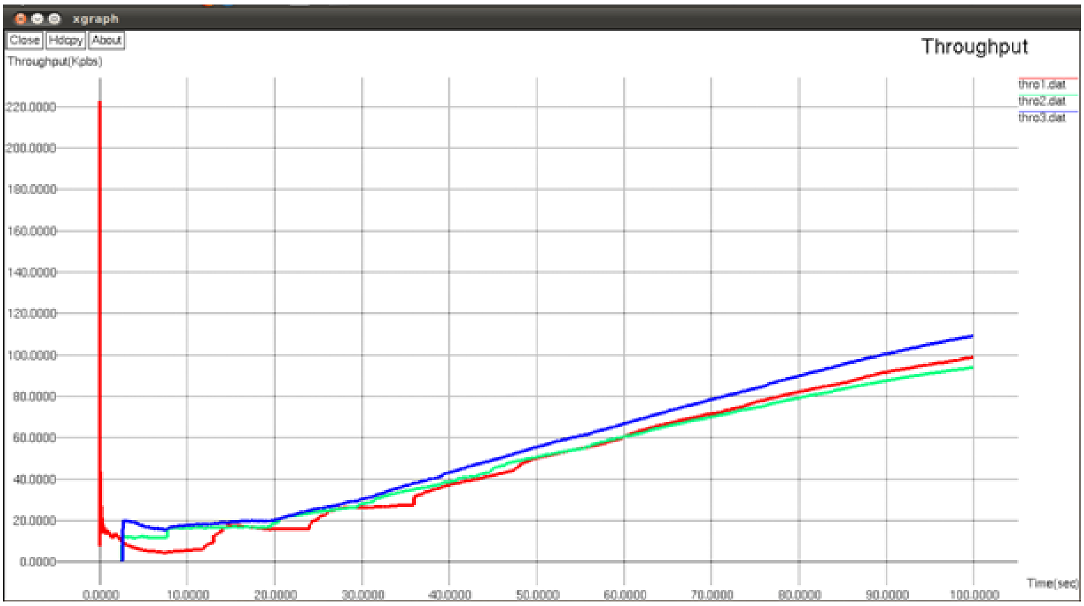


Figure 3.21: Throughput comparison with 20 connections

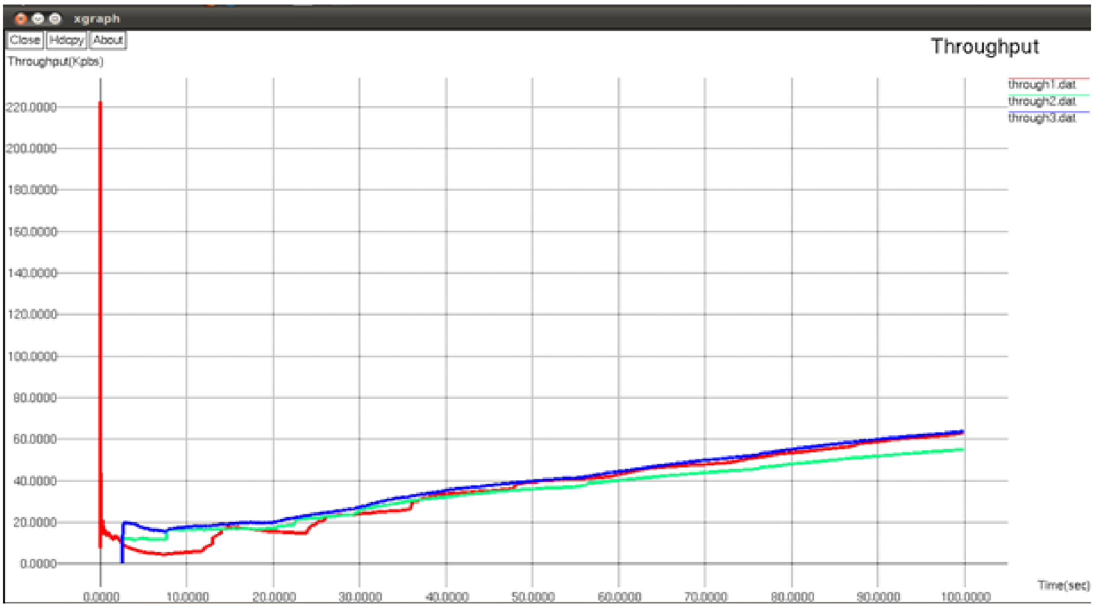


Figure 3.22: Throughput comparison with 30 connections

## 3.9 Summary

In this chapter, we studied routing protocols and the classification of routing protocol. We have given the definition of different types routing protocols. Our main focus was on LEACH and Mobile Ad-hoc Network protocol DSDV, AODV, DSR. We run the simulation to do the comparison between DSDV, AODV and DSR on different matrices. We have plotted the results which we have got from the simulation and describe the interesting observations we have found.

# CHAPTER 4
# IEEE 802.15.4 LOW RATE WIRELESS PERSONAL AREA NETWORK

## 4.1 Introduction

Due to the vast success of wireless local area networks,the wireless networking community has been focused on increasing WLAN capabilities and developing new techniques to meet the needs of the growing applications requiring wireless facilities. With the new world, there is a movement towards standardized protocols and away from applications requiring inflexible wireless connectivity often based on proprietary technologies. IEEE 802.15 is a working group of the Institute of Electrical and Electronics Engineers (IEEE) IEEE 802 standards committee which specifies Wireless Personal Area Network (WPAN) standards.

**Why 802.15?**

As the world is gradually being dependent on wireless technology which otherwise provides way better flexibility than the wired one, developing a flexible protocol for WPAN had become a need of time. IEEE standard 802.15.4 intends to offer the fundamental lower network layers of a type of wireless personal area network (WPAN) which focuses on low-cost, low-speed ubiquitous communication between devices (in contrast with other, more end-user oriented approaches, such as Wi-Fi). The emphasis is on very low cost communication of nearby devices with little to no underlying infrastructure, intending to exploit this to lower power consumption even more.

There are different types of Task Groups of 802.15. They are:

### 4.1.1 Task Group 1 : WPAN/Bluetooth

Task group one is based on Bluetooth technology. It consists of physical layer (PHY) and Media Access Control (MAC) specification for wireless connectivity with fixed, portable and moving devices. Standards were issued in 2002 and 2005. Task Group 2 : Coexistence Task group two addresses the coexistence of wireless personal area networks (WPAN) with other wireless devices operating in unlicensed frequency bands such as wireless local area networks (WLAN). It has gone obsolete due to its drawbacks.

### 4.1.2 Task Group 3 :High Rate WPAN

IEEE 802.15.3 has been evolved from previous technology (e.g. Bluetooth ,Coexistence ) to support higher rate of data transfer ranging from 11 to 55 Mbit/s in WPAN.

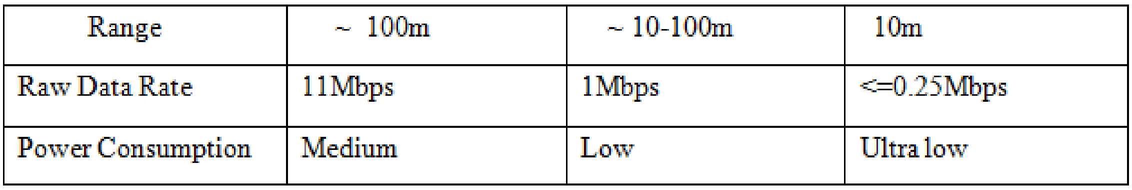
## 4.2 Motivation for evolution of 802.15.4

Due to the vast success of wireless local area networks,the wireless networking community has been focused on increasing WLAN capabilities and developing new techniques to meet the needs of the growing applications requiring wireless facilities. With the new world, there is a movement towards standardized protocols and away from applications requiring inflexible wireless connectivity often based on proprietary technologies. Recently, the concept of a standardized low rate wireless personnel area network (LRWPANs) has emerged. Fuelled by the need to enable inexpensive wireless sensor network applications, in December 2000 Task Group 4, under the IEEE 802 Working Group 15, was formed to begin the development of a LRWPAN standard IEEE 802.15.4. The goal of Task Group 4 is to provide a standard which has the characteristics of ultra-low complexity, low-cost and extremely low-power for wireless connectivity among inexpensive, fixed, portable and moving devices [29].

## 4.3 Coexistence with Bluetooth and other technologies

The IEEE 802.15.4 devices are proposed to work in the 2.4 GHz (ISM) band. Other IEEE 802 wireless devices, such as IEEE 802.11b (WLAN) and IEEE 802.15.1(Bluetooth) use

Table 4.1: Comparison between WLAN, WPAN and LRWPAN.



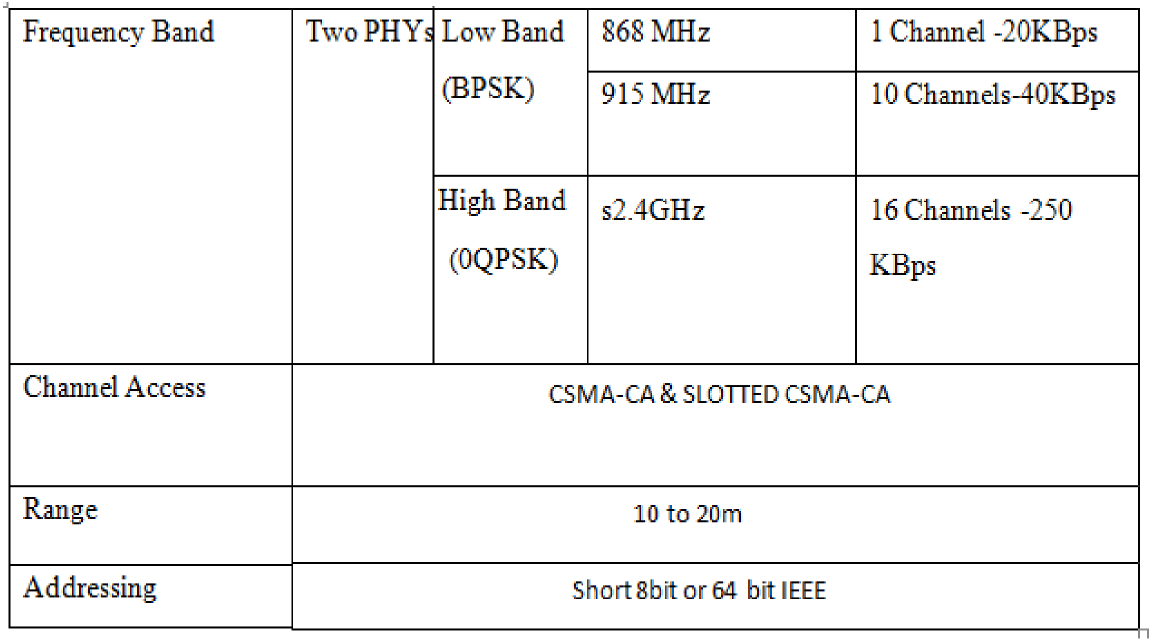
| | | | |
|---|---|---|---|
| Range | ~ 100m | ~ 10-100m | 10m |
| Raw Data Rate | 11Mbps | 1Mbps | <=0.25Mbps |
| Power Consumption | Medium | Low | Ultra low |

the same band. IEEE 802.15.4 and IEEE 802.11b standards support complimentary applications; e.g., IEEE 802.15.4 devices used to support a wireless sensor array within a home or industrial complex could be collocated with IEEE 802.11b in order to provide WLAN support. Wireless devices based on these two standards are likely to be collocated and therefore their ability to coexist needs to be evaluated. Central to the coexistence issue between wireless devices is the ability to differentiate between operational conditions which will and will not result in the communication devices failing to meet the requirements of an application.

## 4.4 Features

Keeping an eye on low-cost and low-power, IEEE 802.15.4 is implementing applications in the fields of industrial, agricultural, vehicular, residential and medical sensors and actuators. The goal of IEEE 802.15.4 is to address applications where existing WPAN solutions are too expensive and the performance of a technology such as BluetoothTM is not required. IEEE 802.15.4 LR-WPANs complement other WPAN technologies by providing very low power consumption capabilities at very low cost, thus enabling applications that were previously impractical. Table 4.2 illustrates a basic comparison between IEEE 802.15.4 and other IEEE 802 wireless networking standards. The IEEE 802.15.4 standard is being designed to be

Table 4.2: IEEE 802.15.4 High Level Characteristics



| | | | | |
|---|---|---|---|---|
| Frequency Band | Two PHYs | Low Band (BPSK) | 868 MHz | 1 Channel -20KBps |
| | | | 915 MHz | 10 Channels-40KBps |
| | | High Band (0QPSK) | s2.4GHz | 16 Channels -250 KBps |
| Channel Access | CSMA-CA & SLOTTED CSMA-CA | | | |
| Range | 10 to 20m | | | |
| Addressing | Short 8bit or 64 bit IEEE | | | |

used in a wide variety of applications which require simple wireless communications over short-range distances with limited power and relaxed throughput needs. IEEE 802.15.4 facilitates Wireless Sensor Networks (WSNs) with the goal of reducing the installation cost of sensors and actuators while enabling sensor-rich environments.

## 4.5 LR-WPAN Design Issues

The obvious goal of designing LR-WPANs is low power consumption, thereby maximizing battery life. To achieve low average power consumption, IEEE 802.15.4 assumes that the amount of data transmitted is short and that it is not transmitted frequently for keeping a low duty-cycle. Besides that, the packet structure was designed to add minimal overhead over the transported payload.

### 4.5.1 Topologies

Three network toplogies are allowed in this standard.

Star topology Mesh topology Cluster-tree Topology

Figure 4.1. In the star topology, the communication is performed between network devices

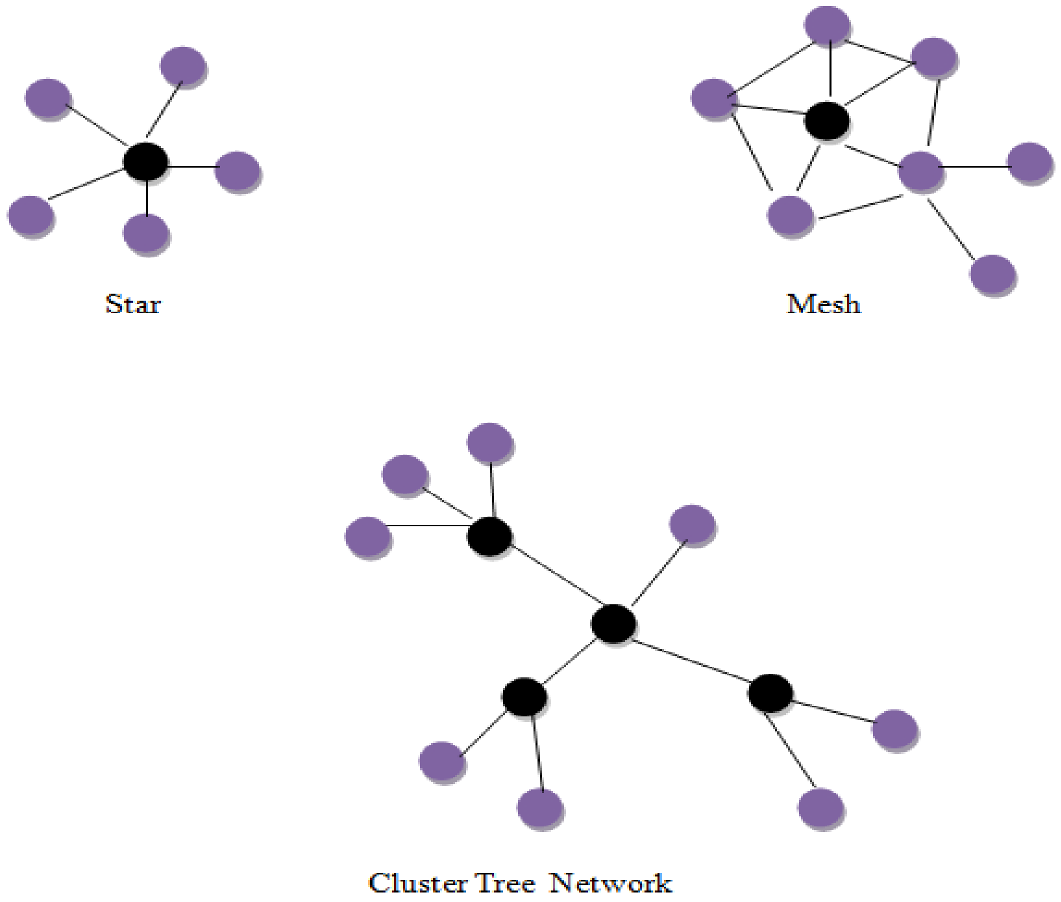


Figure 4.1: Star , Mesh and Cluster Tree Networks

and a single central controller, called the PAN coordinator. A network device is either the initiatorr point or the terminatoro of network communications. The PAN coordinator is in charge of managing all the star PAN functionality. In the peer-to-peer topology, every network device can communicate with any other within its range. This topology also contains a PAN coordinator, which acts as the root of the network. Peer-to-peer topology allows more complex network formations to be implemented; e.g. ad hoc and self-configuring networks. The routing mechanisms required for multi-hopping are part of the network layer and are therefore, not in the scope of IEEE 802.15.4. Cluster-tree network is a special case of a peer-to-peer network in which most devices are FFDs and an RFD may connect to a cluster-tree network as a leave node at the end of a branch. Any of the FFD can act as a coordinator and provide synchronization services to other devices and coordinators. Only one of these coordinators however is the PAN coordinator. The PAN coordinator forms the first cluster by establishing itself as the cluster head (CLH) with a cluster identifier (CID) of zero, choosing an unused PAN identifier, and broadcasting beacon frames to neighboring devices. A can-

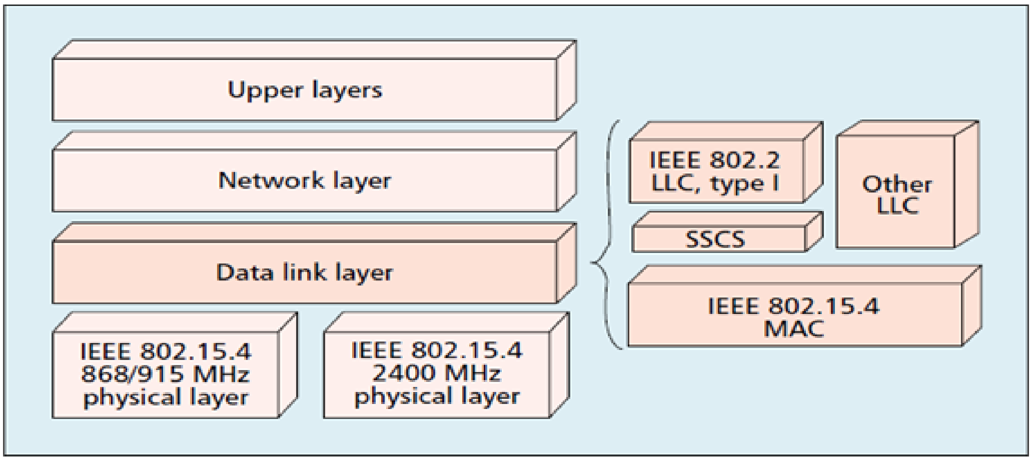


Figure 4.2: ISO-OSI layered network model view of 802.15.4

didate device receiving a beacon frame may request to join the network at the CLH. If the PAN coordinator permits the device to join, it will add this new device as a child device in its neighbor list. The newly joined device will add the CLH as its parent in its neighbor list and begin transmitting periodic beacons such that other candidate devices may then join the network at that device. Once application or network requirements are met, the PAN coordinator may instruct a device to become the CLH of a new cluster adjacent to the rst one. The advantage of this clustered structure is the increased coverage area at the cost of increased message latency.

### 4.5.2 Ins and Outs

An IEEE 802.15.4 LR-WPAN device is composed of a physical (PHY) layer and a medium access control (MAC) sub layer that provides access to the physical channel for all types of transfer and ensures the reliable transfer of frames.

## 4.6 IEEE 802.15.4 PHY

The PHY provides two services: the PHY data service and PHY management service interfacing to the physical layer management entity (PLME). The PHY data service enables the transmission and reception of PHY protocol data units (PPDU) across the physical radio channel. The features of the PHY are activation and deactivation of the radio transceiver,

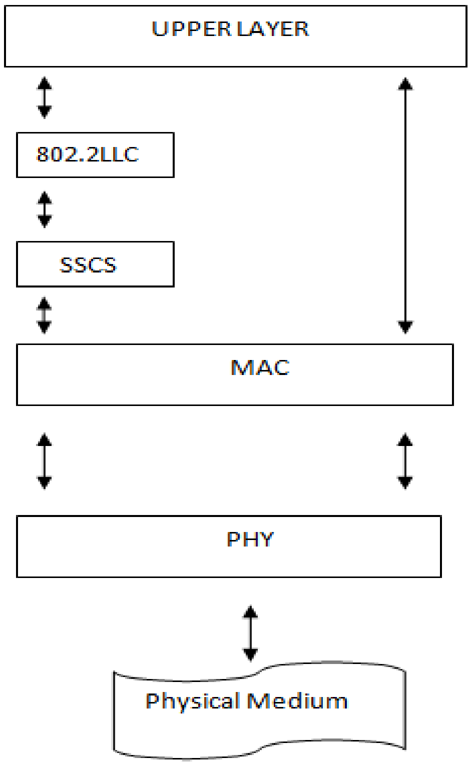
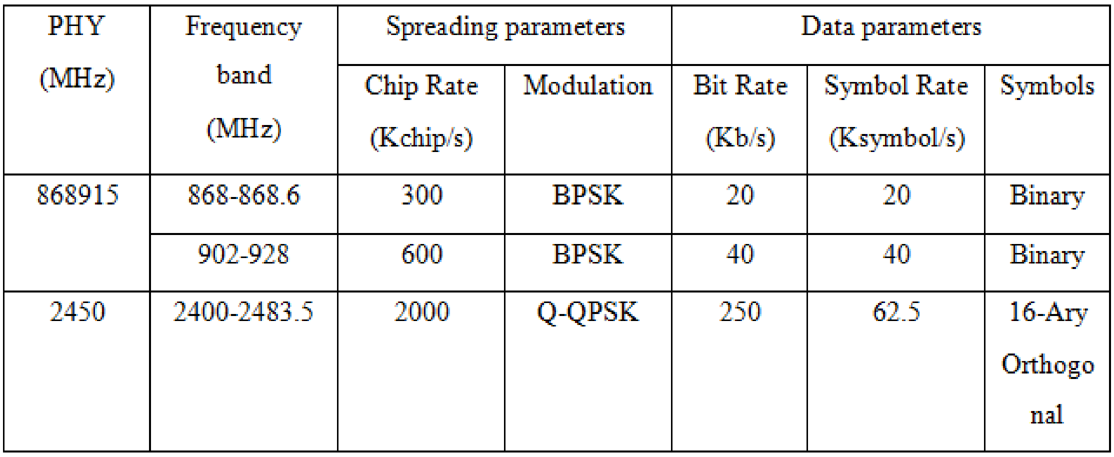


Figure 4.3: LR-WPAN Device Architecture.

energy detection (ED), link quality indication (LQI), channel selection, clear channel assessment (CCA) and transmitting as well as receiving packets across the physical medium. The standard offers two PHY options based on the frequency band. Both are based on direct sequence spread spectrum (DSSS). The data rate is 250kbps at 2.4GHz, 40kbps at 915MHz and 20kbps at 868MHz. The higher data rate at 2.4GHz is attributed to a higher-order modulation scheme. Lower frequency provide longer range due to lower propagation losses. Low rate can be translated into better sensitivity and larger coverage area. Higher rate means higher throughput, lower latency or lower duty cycle. This information is summarized in Table 4.3.

There is a single channel between 868 and 868.6MHz, 10 channels between 902.0 and 928.0MHz, and 16 channels between 2.4 and 2.4835GHz as shown in Table-3. Several channels in different frequency bands enables the ability to relocate within spectrum. The standard also allows dynamic channel selection, a scan function that steps through a list of supported channels in search of beacon, receiver energy detection, link quality indication, channel switching. Receiver sensitivities are -85dBm for 2.4GHz and -92dBm for

Table 4.3: Frequency bands and data rates.



| PHY (MHz) | Frequency band (MHz) | Spreading parameters | | Data parameters | | |
|---|---|---|---|---|---|---|
| | | Chip Rate (Kchip/s) | Modulation | Bit Rate (Kb/s) | Symbol Rate (Ksymbol/s) | Symbols |
| 868915 | 868-868.6 | 300 | BPSK | 20 | 20 | Binary |
| | 902-928 | 600 | BPSK | 40 | 40 | Binary |
| 2450 | 2400-2483.5 | 2000 | Q-QPSK | 250 | 62.5 | 16-Ary Orthogonal |

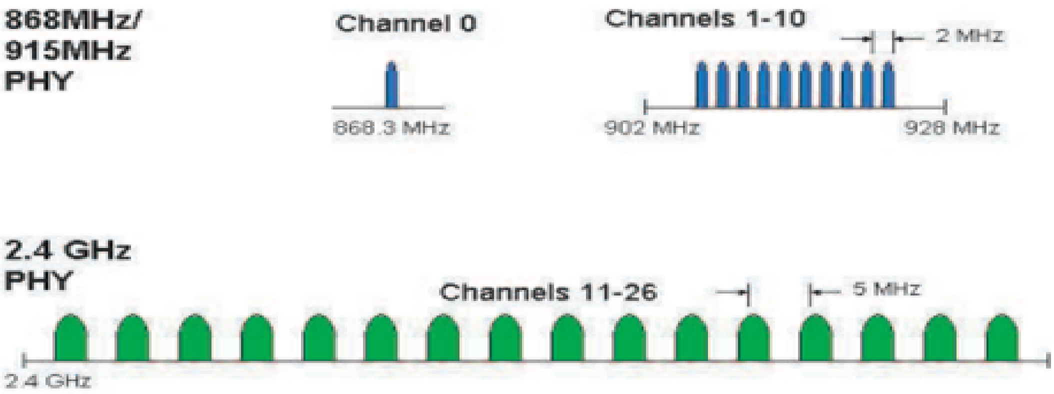


Figure 4.4: Operating frequency bands.

868/915MHz. The advantage of 6-8dB comes from the advantage of lower rate. The achievable range is a function of receiver sensitivity and transmit power. The maximum transmit power shall conform with local regulations. A compliant device shall have its nominal transmit power level indicated by the PHY parameter, phyTransmitPower.

### 4.6.1 Receiver Energy Detection (ED)

The receiver energy detection (ED) measurement is intended for use by a network layers part of channel selection algorithm. It is an estimate of the received signal power within the bandwidth of an IEEE 802.15.4 channel. No attempt is made to identify or decode signals on the channel. The ED time should be equal to 8 symbol periods. The ED result shall be reported as an 8-bit integer ranging from 0x00 to 0xff. The minimum ED value (0) shall indicate received power less than 10dB above the specied receiver sensitivity.

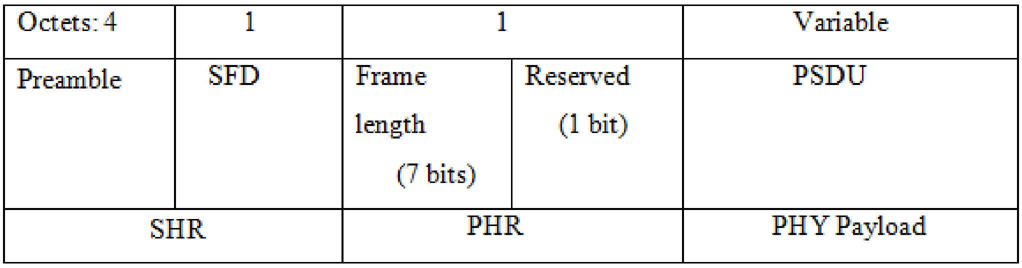
The range of received power spanned by the ED values shall be at least 40dB. Within this range, the mapping from the received power in decibels to ED values shall be linear with an accuracy of + 6dB.

### 4.6.2 Link Quality Indication (LQI)

Upon reception of a packet, the PHY sends the PSDU length, PSDU itself and link quality (LQ) in the PD-DATA.indication primitive. The LQI measurement is a characterization of the strength and/or quality of a received packet. The measurement may be implemented using receiver ED, a signal-to-noise estimation or a combination of these methods. The use of LQI result is up to the network or application layers. The LQI result should be reported as an integer ranging from 0x00 to 0xff. The minimum and maximum LQI values should be associated with the lowest and highest quality IEEE 802.15.4 signals detectable by the receiver and LQ values should be uniformly distributed between these two limits.

### 4.6.3 Clear Channel Assessment (CCA)

The clear channel assessment (CCA) is performed according to at least one of the following three methods:



| Octets: 4 | 1 | 1 | | Variable |
|---|---|---|---|---|
| Preamble | SFD | Frame length (7 bits) | Reserved (1 bit) | PSDU |
| SHR | | PHR | | PHY Payload |

Energy above threshold. CCA shall report a busy medium upon detecting any energy above the ED threshold.

Carrier sense only. CCA shall report a busy medium only upon the detection of a signal with the modulation and spreading characteristics of IEEE 802.15.4. This signal may be above or below the ED threshold.

Carrier sense with energy above threshold. CCA shall report a busy medium only upon the detection of a signal with the modulation and spreading characteristics of IEEE 802.15.4 with energy above the ED threshold.

## 4.7 Layers

### 4.7.1 The Network Layer

Resembling all IEEE 802 standards, the IEEE 802.15.4 draft standard covers only those layers up to and including portions of the data link layer(DLL). Higher-layer protocols are at the discretion of the individual applications utilized in an in-home network environment. In particular, this section considers the issues and obstacles surrounding the network layer. In traditional wired networks, the network layer is responsible for topology construction and maintenance, as well as naming and binding services, which incorporate the necessary tasks of addressing, routing, and security . The same services exist for wireless in-home networks, but are far more challenging to implement because of the premium placed on energy conservation. In fact, it is important for any network layer implementation built on the already energy conscious IEEE 802.15.4 draft standard to be equally conservative. Network layers built on the standard are expected to be self-organizing and self-maintaining, to minimize total cost to the consumer user.

### 4.7.2 The Data Link Layer

The IEEE 802 stadard proposes to split the DLL into two sub layers, the MAC and logical link control (LLC) sub layers. The LLC is standardized in 802.2 and is common among the 802 standards such as 802.3, 802.11, and 802.15.1.On the other hand, MAC sub layer is closer to the hardware and may vary with the physical layer implementation. Figure 2 shows how IEEE 802.15.4 fits into the International Organization for Standardization (ISO) open systems interconnection (OSI) reference model . The IEEE 802.15.4 MAC provides services to an IEEE 802.2 type I LLC through the service-specific convergence sub layer (SSCS), or a proprietary LLC can access the MAC services directly without going through the SSCS. The SSCS ensures compatibility between different LLC sub layers and allows the MAC to be accessed through a single set of access points. Using this model, the 802.15.4 MAC provides features not utilized by 802.2, and therefore allows the more complex network topologies mentioned above. The features of the IEEE 802.15.4 MAC are association and disassociation, acknowledged frame delivery, channel access mechanism, frame validation, guaranteed time slot management, and beacon management. These items will be introduced in the following subsections. The MAC sub layer provides two services to higher layers that can be accessed through two service access points (SAPs). The MAC data service is accessed through the MAC common part sub layer (MCPS-SAP), and the MAC management services are accessed through the MAC layer management entity (MLME-SAP). These two services provide an interface between the SSCS or another LLC and the PHY layer. The MAC management service has 26 primitives. Compared to 802.15.1 (Bluetooth), which has about 131 primitives and 32 events, the 802.15.4 MAC is of very low complexity, making it very suitable for its intended low-end applications, albeit at the cost of a smaller feature set than 802.15.1 (e.g., 802.15.4 does not support synchronous voice links).

## 4.8 The general MAC Frame format

MAC frame structure is kept very feasible to accommodate the needs of different applications and network topologies while maintaining a simple protocol. The general format of a MAC frame is shown in Fig. 4.5. The MAC frame is called the MAC protocol data unit (MPDU) and is composed of the MAC header (MHR), MAC service data unit (MSDU), and MAC footer (MFR). The first field of the MAC header is the frame control field. It indicates the type of MAC frame being transmitted, specifies the format of the address field, and controls the acknowledgment. In short, the frame control field specifies how the rest of the frame looks and what it contains. The size of the address field may vary between 0 and 20 bytes. For instance, a data frame may contain both source and destination information, while the return acknowledgment frame does not contain any address information at all. On the other hand, a beacon frame may only contain source address information.

In addition, short 8-bit device addresses or 64-bit IEEE device addresses may be used. This flexible structure helps increase the efficiency of the protocol by keeping the packets short. The payload field is variable in length; however, the complete MAC frame may not exceed 127 bytes in length. The data contained in the payload is dependent on the frame type. The

IEEE 802.15.4 MAC has four different frame types. These are the beacon frame, data frame, acknowledgment frame, and MAC command frame. Only the data and beacon frames actually contain information sent by higher layers; the acknowledgment and MAC command frames originate in the MAC and are used for MAC peer-to-peer communication. Other fields in a MAC frame are the sequence number and frame check sequence (FCS). The sequence number in the MAC header matches the acknowledgment frame with the previous transmission. The transaction is considered successful only when the acknowledgment frame contains the same sequence number as the previously transmitted frame. The FCS helps verify the integrity of the MAC frame. The FCS in an IEEE 802.15.4 MAC frame is a 16-bit International Telecommunication Union  Telecommunication Standardization Sector (ITU-T) cyclic redundancy check (CRC).

### 4.8.1 IEEE 802.15.4 MAC

The MAC sublayer provides two services: the MAC data service and the MAC management service interfacing to the MAC sublayer management entity (MLME) service access point (SAP) (MLMESAP). The MAC data service enables the transmission and reception of MAC protocol data units (MPDU) across the PHY data service. The features of MAC sublayer are beacon management, channel access, GTS management, frame validation, acknowledged frame delivery, association and disassociation.

### 4.8.2 Superframe Structure

LR-WPAN allows the optional use of a superframe structure. The format of the superframe is defined by the coordinator. The superframe is bounded by network beacons and is divided into 16 equally sized slots. The beacon frame is sent in the first slot of each superframe. If a coordinator does not want to use the superframe structure, it may turn off the beacon transmissions. The beacons are used to synchronize the attached devices, to identify the PAN and to describe the structure of superframes. The superframe can have an active and an inactive portion. During the inactive portion, the coordinator shall not interact with its PAN and may enter a low-power mode. The active portion consists of contention access period (CAP) and contention free period (CFP). Any device wishing to communicate during the CAP shall compete with other devices using a slotted CSMACA mechanism. On the

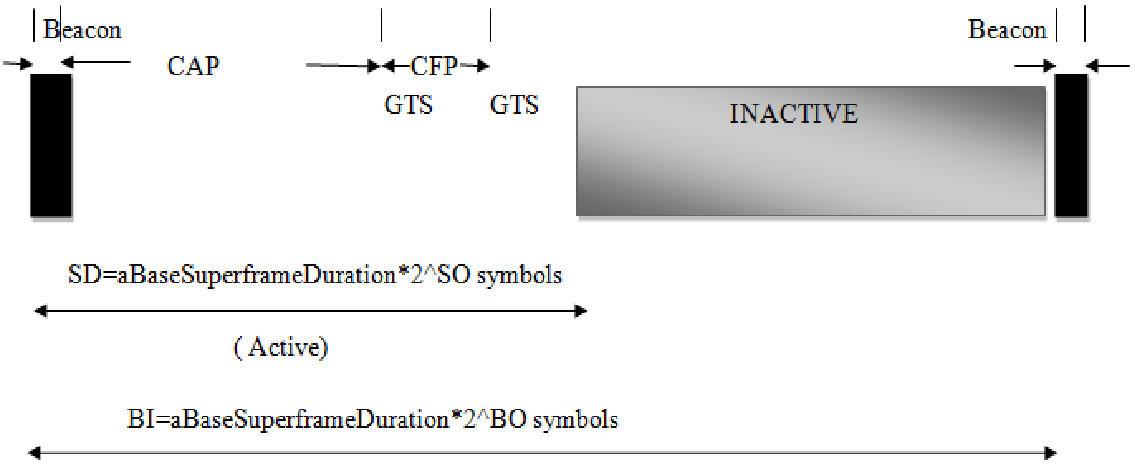

Beacon
CAP
CFP
GTS
GTS
INACTIVE
Beacon
SD=aBaseSuperframeDuration*2^SO symbols
( Active)
BI=aBaseSuperframeDuration*2^BO symbols

Figure 4.5: An example superframe structure.

other hand, the CFP contains guaranteed time slots (GTSs). The GTSs always appear at the end of the active superframe starting at a slot boundary immediately following the CAP. The PAN coordinator may allocate up to seven of these GTSs and a GTS can occupy more than one slot period. The duration of different portions of the superframe are described by the values of macBeaconOrder and macSuperFrameOrder. macBeaconOrder describes the interval at which the coordinator shall transmit its beacon frames. The beacon interval, BI, is related to the macBeaconOrder, BO, as follows: BI = aBaseSuperFrameDuration2 BO, 0 BO 14. The superframe is ignored if BO = 15. The value of macSuperFrameOrder describes the length of the active portion of the superframe. The superframe duration, SD, is related to macSuperFrameOrder, SO, as follows: SD = aBaseSuperFrameDuration2 SO , 0 SO 14. If SO = 15, the superframe should not remain active after the beacon. The active portion of each superframe is divided into a aNumSuperFrameSlots equally spaced slots of duration 2 SO aBaseSlotDuration and is composed of three parts: a beacon, a CAP and CFP. The beacon is transmitted at the start of slot 0 without the use of CSMA. The CAP starts immediately after the beacon. The CAP shall be at least aMinCAPLength symbols unless additional space is needed to temporarily accommodate the increase in the beacon frame length to perform GTS maintenance. All frames except acknowledgement frames or any data frame that immediately follows the acknowledgement of a data request command that are transmitted in the CAP shall use slotted CSMA-CA to access the channel. A transmission in the CAP shall be complete one IFS period before the end of the CAP. If this is not possible, it defers its transmission until the CAP of the following superframe. An example superframe structure is shown in Figure 4.6. The CFP, if present, shall start on a slot boundary immediately following the CAP and extends to the end of the active portion of the superframe. The length of the CFP is determined by the total length of all of the combined GTSs. No transmissions within the CFP shall use a CSMA-CA mechanism. A device transmitting in the CFP shall ensure that its transmissions are complete one IFS period before the end of its GTS. IFS time is the amount of time necessary to process the received packet by the PHY. Transmitted frames shall be followed by an IFS period. The length of IFS depends on the size of the frame that has just been transmitted. Frames of up to aMaxSIFSFrameSize in length shall be followed by a SIFS whereas frames of greater length shall be followed by a LIFS. The PANs that do not wish to use the superframe in a nonbeacon-enabled shall set both macBeaconOrder and macSuperFrameOrder to 15. In this kind of network, a coordinator shall not transmit any beacons, all transmissions except the acknowledgement frame shall use unslotted CSMA-CA to access channel, GTSs shall not be permitted.

### 4.8.3 CSMA-CA Algorithm

If superframe structure is used in the PAN, then slotted CSMA-CA shall be used. If beacons are not being used in the PAN or a beacon cannot be located in a beacon-enabled network, unslotted CSMA-CA algorithm is used. In both cases, the algorithm is implemented using units of time called backoff periods, which is equal to aUnitBackoffPeriod symbols.

In slotted CSMA-CA channel access mechanism, the backoff period boundaries of every device in the PAN are aligned with the superframe slot boundaries of the PAN coordinator. In slotted CSMA-CA, each time a device wishes to transmit data frames during the CAP, it shall locate the 11boundary of the next backoff period. In unslotted CSMA-CA, the backoff periods of one device do not need to be synchronized to the backoff periods of another device. Each device has 3 variables: NB, CW and BE. NB is the number of times the CSMA-CA algorithm was required to backoff while attempting the current transmission. It is initialized to 0 before every new transmission. CW is the contention window length, which denes the number of backoff periods that need to be clear of activity before the transmission can start. It is initialized to 2 before each transmission attempt and reset to 2 each time the channel is assessed to be busy.CW is only used for slotted CSMA-CA. BE is the backoff exponent, which is related to how many backoff periods a device shall wait before attempting to assess the channel. Although the receiver of the device is enabled during the channel assessment portion of this algorithm, the device shall discard any frames received during this time. In slotted CSMA-CA, NB, CW and BE are initialized and the boundary of the next backoff period is located. In unslotted CSMA-CA, NB and BE are initialized (step1). The MAC layer shall delay for a random number of complete backoff periods in the range 0 to 2 BE 1 (step 2) then request that PHY performs a CCA (clear channel assessment) (step 3). The MAC sublayer shall then proceed if the remaining CSMA-CA algorithm steps, the frame transmission, and any acknowledgement can be completed before the end of the CAP. If the MAC sublayer cannot proceed, it shall wait until the start of the CAP in the next superframe and repeat the evaluation. If the channel is assessed to be busy (step 4), the MAC sublayer shall increment both NB and BE by one, ensuring that BE shall be no more than aMaxBE. In slotted CSMA-CA, CW can also be reset to 2. If the value of NB is less than or equal to macMaxCSMABackoffs, the CSMA-CA shall return to step 2, else the CSMA-CA shall terminate with a Channel Access Failure status. If the channel is assessed to be idle (step 5), in a slotted CSMA-CA, the MAC sublayer shall ensure that contention window is expired before starting transmission. For this, the MAC sublayer first decrements CW by one. If CW is not equal to 0, go to step 3 else start transmission on the boundary of the next backoff period. In the unslotted CSMA-CA, the MAC sublayer start transmission immediately if the channel is assessed to be idle. The whole CSMA-CA algorithm is illustrated in Figure 4.7 [26].

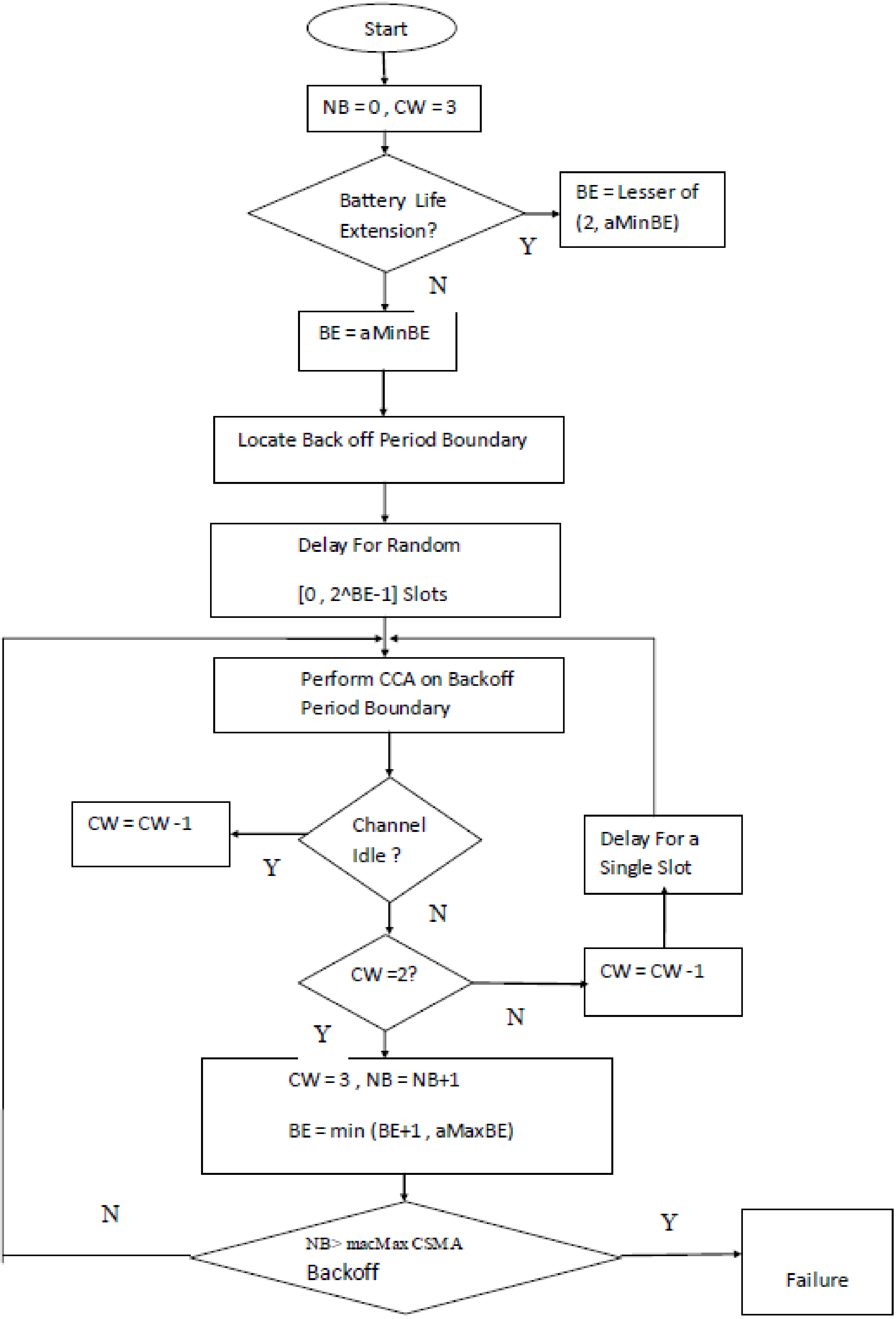


Figure 4.6: Slotted The CSMA-CA algorithm.

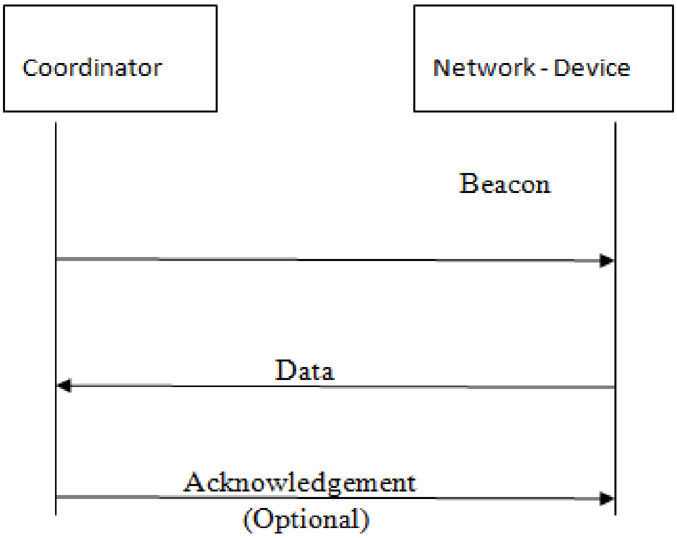


Figure 4.7: Communication to a coordinator in a beacon-enabled network.

### 4.8.4 Data Transfer model

Three types of data transfer transactions exist: from a coordinator to a device, from a device to a coordinator and between two peer devices. The mechanism for each of these transfers depend on whether the network supports the transmission of beacons. When a device wishes to transfer data in a nonbeacon-enabled network, it simply transmits its data frame, using the unslotted CSMA- CA, to the coordinator. There is also an optional acknowledgement at the end as shown in Figure 9. When a device wishes to transfer data to a coordinator in a beacon-enabled network, it rst listens for the network beacon. When the beacon is found, it synchronizes to the superframe structure. At the right time, it transmits its data frame, using slotted CSMA-CA, to the coordinator. There is an optional acknowledgement at the end as shown in Figure 4.8.

The applications transfers are completely controlled by the devices on a PAN rather than by the coordinator. This provides the energy-conservation feature of the ZigBee network. When a coordinator wishes to transfer data to a device in a beacon-enabled network, it indicates in the network beacon that the data message is pending. The device periodically listens to the network beacon, and if a message is pending, transmits a MAC command requesting this data, using slotted CSMA-CA.The coordinator optionally acknowledges the successful transmission of this packet. The pending data frame is then sent using slotted CSMA-CA.

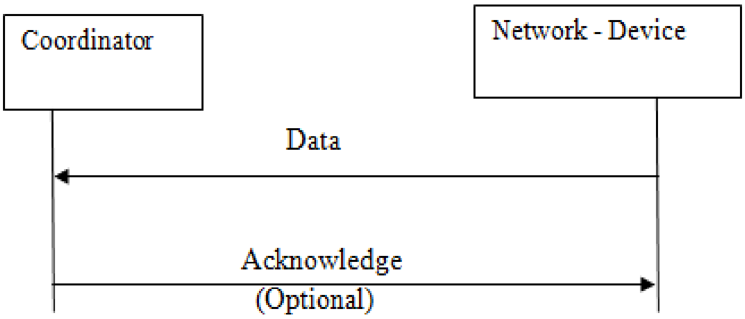


Figure 4.8: Communication to a coordinator in a nonbeacon-enabled network.

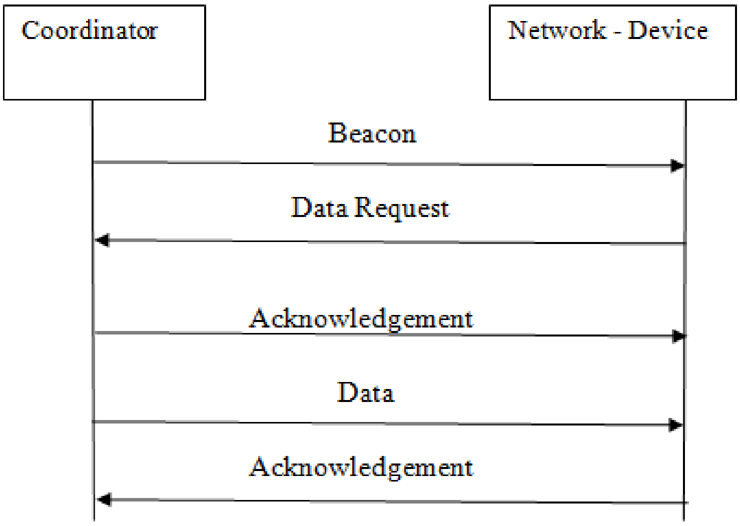


Figure 4.9: Communication from a coordinator in a nonbeacon-enabled network.

The device acknowledged the successful reception of the data by transmitting an acknowledgement frame. Upon receiving the acknowledgement, the message is removed from the list of pending messages in the beacon as shown in Figure 4.9. When a coordinator wishes to transfer data to a device in a nonbeacon-enabled network, it stores the data for the appropriate device to make contact and request data. A device may make contact by transmitting a MAC command requesting the data, using unslotted CSMA-CA, to its coordinator at an application-dened rate. The coordinator acknowledges this packet. If data are pending, the coordinator transmits the data frame using unslotted CSMA-CA. If data are not pending, the coordinator transmits a data frame with a zero-length payload to indicate that no data were pending. The device acknowledges this packet as shown in Figure 4.10.

In a peer-to-peer network, every device can communicate with any other device in its transmission radius. There are two options for this. In the first case, the node will listen constantly and transmit its data using unslotted CSMA-CA. In the second case, the nodes synchronize with each other so that they can save power.

### 4.8.5 Starting and Maintaining PANs

A PAN shall be started by an FFD only after an active channel or ED channel scan has been performed and a suitable PAN identifier selection has been made as shown in Figure 4.7. The active scan allows the FFD to locate any coordinator transmitting beacon frames within its POS (personal operating space). An active channel scan is requested over a specified set of logical channels. For each logical channel, the device shall first switch to the channel and send a beacon request command. The device shall then enable its receiver for at most aBaseSuperframeDuration (2n + 1) symbols, where n is between 0 and 14. During this time, the device shall reject all nonbeacon frames and record the information contained in all unique beacons in a PAN descriptor structure. If the coordinator of a beacon-enabled PAN receives the beacon request command, it shall ignore the command and continue transmitting its beacons as usual. If the coordinator of a nonbeaconenabled PAN receives this command, it shall transmit a single beacon frame using unslotted CSMACA. The active scan on a particular channel terminates when the number of PAN descriptors stored equals this implementation-specified maximum or aBaseSuperframeDuration (2n + 1) symbols, where n is between 0 and 14, have elapsed. The entire scan shall terminate when the number of PAN descriptors stored equals the implementation-specified maximum or every channel in the set of available channels has been scanned. Then selecting a suitable PAN identifier BY prospective PAN coordinator from the list of PAN descriptors returned from the active channel scan IS UP TO application. An ED scan allows the FFD obtain a measure of the peak energy in each requested channel. During the ED scan, the MAC sublayer shall discard all frames received over the PHY data service. An ED scan is performed over a set of logical channels. For each logical channel, repeatedly perform an ED measurement for aBaseSuperframeDuration (2n + 1) where n is the value of the scanDuration. The maximum ED measurement obtained during this period shall be noted before moving onto the next channel in the channel list. The ED scan shall terminate when either the number of channel ED measurements stored equals the implementation-specified maximum or energy has been measured on each of the specified logical channels. In some instances, a situation could occur in which two PANs exist in the same POS with the same PAN identifier. If this conflict happens, the coordinator and its devices shall perform PAN identifier conflict resolution procedure. The PAN coordinator shall conclude that a PAN identifier conflict is present if either a beacon frame is received by the PAN coordinator with the PAN coordinator subfield set to 1, i.e. transmitted by the PAN coordinator, and the PAN identifier is equal to macPANId or a PAN ID conflict notification command is received by the PAN coordinator from a device on its PAN. A device shall conclude that a PAN identifier conflict is present if a beacon frame is received by the device with the PAN coordinator subfield set to 1, the PAN identifier equal to macPANId, an address that is not equal to both macCoordShortAddress and macCoordExtendedAddress. On the detection of the PAN identifier conflict by a device, it shall generate the PAN ID con-flict notification command and send it to the PAN coordinator. If the PAN ID conflict notification command is received correctly, the PAN coordinator shall send an ack and resolve the conflict. On the detection of the PAN identifier conflict by a coordinator, the coordinator shall first perform an active scan and then select a new PAN identifier based on the information from the scan. The coordinator shall then broadcast the coordinator realignment command containing the new PAN identifier with the source PAN identifier field equal to the value in macPANId. Once the coordinator realignment field has been sent, the coordinator shall set macPANId to the new PAN identifier.

### 4.8.6 Beacon Generation

Depending on the parameters of the MLME-START.request primitive, the FFD may either operate in a beaconless mode or may begin beacon transmissions either as the PAN coordinator or as a device on a previously established PAN. An FFD that is not the PAN coordinator shall begin transmitting beacon frames only when it has successfully associated with a PAN. This primitive also includes macBeaconOrder and macSuperFrameOrder parameters that determine the duration of the beacon interval and the duration of the active and inactive portions. The time of the transmission of the most recent beacon shall be recorded in macBeaconTxTime and shall be computed so that its value is taken at the same symbol boundary in each beacon frame the location of which is implementation specific.

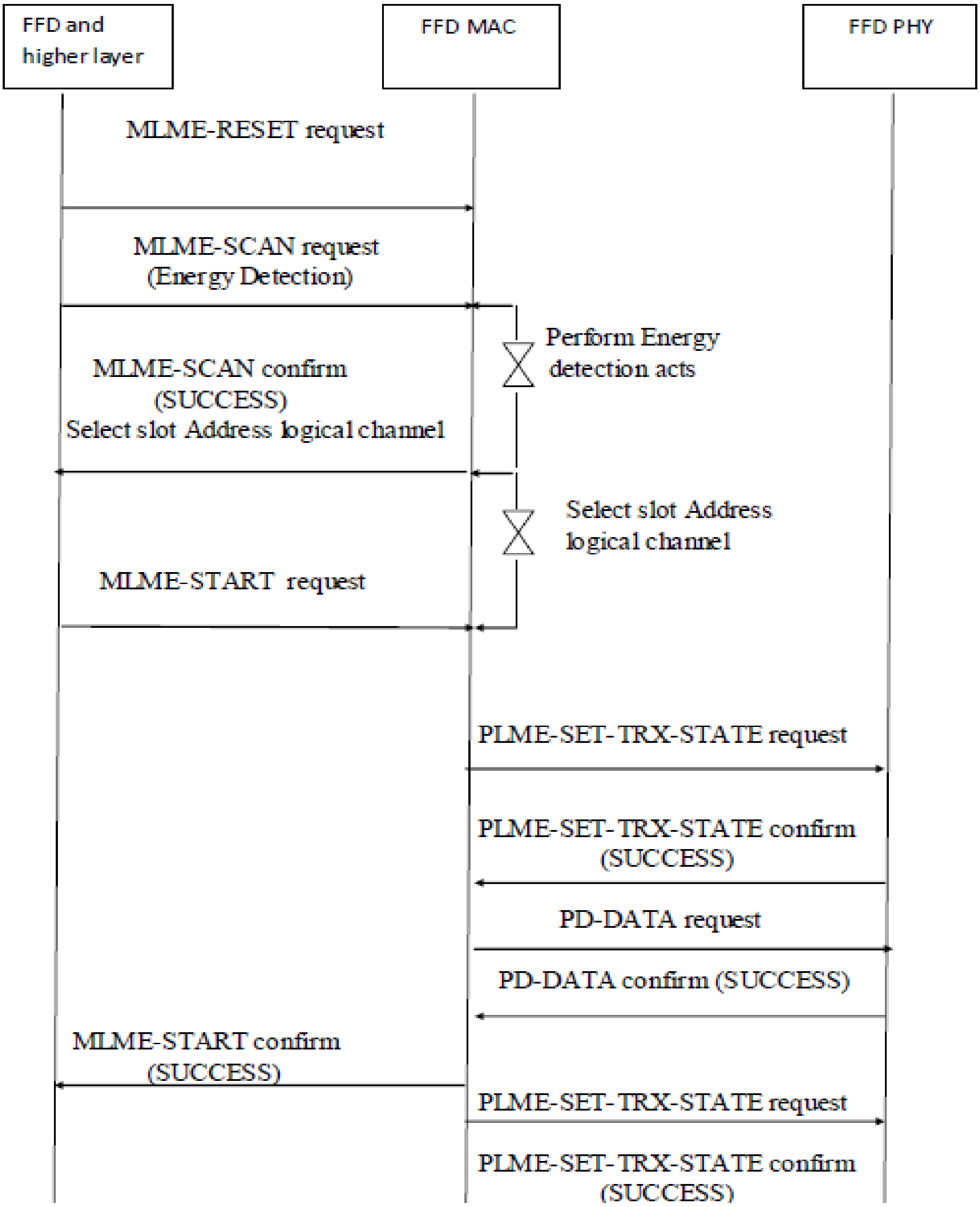


Figure 4.10: PAN start message sequence chart - PAN coordinator

### 4.8.7 Association and Disassociation

An FFD may indicate its presence on a PAN to other devices by transmitting beacon frames. This allows other devices to perform device discovery. An FFD that is not a PAN coordinator shall begin transmitting beacon frames only when it has successfully associated with a PAN. Association of a device starts after having completed either an active channel scan or a passive channel scan. The passive scan, like an active scan, allows a device to locate any coordinator transmitting beacon frames within its POS whereas there beacon request command is not required for passive scan. The results of the channel scan are then used to choose a suitable PAN. A device shall attempt to associate only with a PAN that is currently allowing association. Now, how to choose a suitable pan with which to associate from the list of pan descriptors returned from the channel scan is up to application. Following the selection of a PAN with which to associate, the next higher layers request that MLME congure the phyCurrentChannel to the appropriate logical channel on which to associate, macPANId to the identier of the PAN with which to associate and macCoordExtendedAddress or macCoordShortAddress to the address of the coordinator with which it associates. An unassociated device shall initiate the association procedure by sending an associate request command to the coordinator of an existing PAN. If the association request command is received correctly, the coordinator shall send an acknowledgement. This acknowledgement however does not mean that the device has associated. The coordinator needs time to determine whether the current sources available on a PAN are sufficient to allow another device to associate. This decision should be made within aResponseWaitTime symbols. If already associated, remove all information. If sufficient resources are available, the coordinator shall allocate a short address to the device and generate an association response command containing the new address and a status indicating the successful association. If there are not enough resources, the coordinator shall generate an association response command containing a status indicating failure. This response is sent to the device using indirect transmission (pending, request,...). On the other side, the device, after getting the acknowledgement frame, waits for the response for aResponseWaitTime symbols. It either checks the beacons in the beacon-enabled network or extracts the association response command from the coordinator after aResponseWaitTime symbols. On reception of association response command, the device shall send an acknowledgement. If the association is successful, store the addressed of the coordinator with which it has associated. The association procedure is shown in Figure 4.8 on the coordinator side and in Figure 4.9 on the device side. When a coordinator wants one of its associated devices to leave the PAN, it shall send the disassociation notification command to the device using indirect transmission. Upon reception of the packet, the device should send the acknowledgement frame. Even if the ack is not received, the coordinator shall consider the device disassociated. If an associated device wants to leave the PAN, it shall send a disassociation notification command to the coordinator. Upon reception, the coordinator sends ack. Even if the ack is not received, the device shall consider itself disassociated. An associated device shall disassociate itself by removing all references to the PAN. A coordinator shall disassociate a device by removing all references to that device.

### 4.8.8 Synchronization

For PANs supporting beacons, synchronization is performed by receiving and decoding beacon frames. For PANs not supporting beacons, the synchronization is performed by polling the coordinator for data. In a beacon enabled network, devices shall be permitted to acquire synchronization only with beacons containing the PAN identifier specified in macPANId. If tracking is specified in the MLMESYNC.request primitive, the device shall attempt to acquire the beacon and keep track of it by regular and timely activation of its receiver. It shall enable its receiver at a time prior to the next expected beacon frame transmission, i.e. just before the known start of the next superframe. If tracking is not specified, the device shall attempt to acquire the beacon only once. To acquire beacon synchronization, a device shall enable its receiver and search for at most aBaseSuperframeDuration (2n + 1) symbols, where n is the macBeaconOrder. If a beacon frame containing the current PAN identifier of the device is not received, the MLME shall repeat the search. Once the number of missed beacons reached aMaxLostBeacons, the MLME notifies the next upper layer by issuing MLME-SYNC-LOSS.indication with a reason BEACON-LOSS. The MLME shall timestamp each received beacon frame at the same symbol boundary within each frame, the location of which is implementation specific. In a nonbeacon-enabled network, the devices shall be able to poll the coordinator for data at the discretion of the next higher layer. On receipt of MLME-POLL.request primitive, the MLME follow the procedure for extracting pending data from the coordinator. Another problem with synchronization is orphaned device. If the next higher layer receives repeated communication failures following its requests to transmit data, it may conclude that it has been orphaned. A single communications failure occurs when a device transaction fails to reach the coordinator, i.e. an acknowledgement is not received after aMaxFrameRetries attempts at sending data. If the next higher layer concluded that the device has been orphaned, it may either reset the MAC sublayer and perform the association procedure or perform the orphaned device realignment procedure. If the decision is for orphaned device alignment, orphan scan is performed. During the orphan scan, the MAC sublayer shall discard all frames received over the PHY data service that are not coordinator realignment MAC command frames. For each logical channel over a specified set of logical channels, the device sends an orphan notification command. The device shall then enable its receiver for at most aResponseWaitTime symbols. If the device successfully receives a coordinator realignment command within this time, the device shall disable its receiver. If a coordinator receives the orphan notification command, it searches its device list for the device sending the command. If the coordinator finds a record of the device, it shall send a coordinator realignment command to the orphaned device. Otherwise, it shall ignore the packet. The orphan scan terminates when the device receives a coordinator realignment command or the specified set of logical channels has been scanned [28].

### 4.8.9 Transmission, Reception and Acknowledgement

In order to transmit a data or a beacon or a MAC command frame, the MAC sublayer shall copy the value of masDSN into the sequence number field of the MHR of the outgoing frame and then increment it by one. The source address field shall contain the address of the device sending the frame. If the device has been allocated a short address, it shall use that address in preference to its 64 bit extended address. If the source address field is not present, the originator of the frame shall be assumed to be a PAN coordinator and the destination address shall contain the address of the recipient. The destination address shall contain the intended recipient of the frame, which may be either a 16 bit short address or a 64 bit extended address. If the destination address field is not present, the recipient of the frame shall be assumed to be the PAN coordinator. The destination and source address may be in different PANs, which is identified by the PAN identifier fields. In beacon-enabled PAN, the transmitting device shall attempt to find the beacon before transmitting. If it cannot find the beacon, it shall use unslotted CSMA-CA. Once the beacon is found, it transmits in the appropriate portion of the superframe. Transmission in the CAP shall use slotted CSMA-CA and those in GTS shall not use CSMA-CA. In a nonbeacon-enabled network, the frames are transmitted using unslotted CSMA-CA. Upon reception of packets, the MAC sublayer shall discard all its received frames that do not contain a correct value in their FCS field in the MFR. Receiver is important in terms of energy consumption. Each device may choose whether the MAC sublayer is to enable its receiver during idle periods. During these idle periods, the MAC sublayer shall still service transceiver task requests from the next higher layer. On completion of each transceiver task, the MAC sublayer shall request that the PHY enables or disables its receiver, depending on whether macRxOnWhenIdle is set to TRUE or FALSE, respectively. If beacon is enabled, the value of macRxOnWhenIdle shall be considered only during idle periods of the CAP. Another energy conserving feature of the standard is the indirect transmission feature. The transactions start by the devices themselves rather than the coordinator. In other words, either the coordinator needs to indicate in its beacon when messages are pending for devices or the devices themselves need to poll the coordinator to determine whether they have any messages pending. A device on a beacon-enabled PAN can determine whether any frames are pending for it by examining the contents of the received beacon frame. If the address of the device is contained in the address list field of the beacon frame, the MLME of the device shall send a data request command to the coordinator during the CAP. Upon reception of this command, the coordinator shall send an ack. It indicates whether any data is pending for that device in the ack frame. On receipt of the ack, the device shall enable its receiver for at most aMaxFrameResponseTime CAP symbols in a beacon-enabled PAN or symbols in a nonbeacon-enabled PAN to receive the corresponding frame from the coordinator. If there is data pending, the coordinator should send the frame else send a frame containing zero length payload, indicating that no data is present. The data frame is transmitted without using CSMA-CA if the MAC sublayer can commence transmission of the data frame between aTurnaroundTime and aTurnaroundTime+aUnitBackoffPeriod symbols and there is time remaining in the CAP for the message, appropriate IFS and acknowledgement and using CSMA-CA otherwise. A frame transmitted with the acknowledgement request field set to 1 shall be acknowledged by the recipient. If the intended recipient correctly receives the frame, it shall generate and send an acknowledgement frame containing the same DSN from the data or MAC command frame that is being acknowledged. The transmission of the ack shall commence between aTurnaroundTime and aTurnaroundTime + aUnitBackoffPeriod symbols after the reception of the last symbol of the data or MAC command frame.

#### 4.8.10 GTS Allocation and Management

A GTS allows a device to operate on the channel within a portion of the superframe that is dedicated exclusively to that device. A device shall attempt to allocate and use a GTS only if it is currently tracking the beacons. A GTS shall be allocated only by the PAN coordinator and it shall be used only for communications between the PAN coordinator and a device. A single GTS can extend over one or more superframe slots. The PAN coordinator may allocate up to seven GTSs at the same time, provided there is sufficient capacity in the superframe. A GTS shall be allocated before use, with the PAN coordinator deciding whether to allocate a GTS based on the requirements of the GTS request and the current available capacity in the superframe. GTS shall be allocated on a first-come-first-serve basis and all GTSs shall be placed contiguously at the end of the superframe and after the CAP. Each GTS shall be deallocated when the GTS is no longer required, and a GTS can be deallocated at any time at the discretion of the PAN coordinator or by the device that originally requested the GTSs. A device that has been allocated GTS may also operate in the CAP. The management of the GTSs shall be undertaken by the PAN coordinator only. For each GTS, the PAN coordinator shall be able to store its starting slot, length, direction and associated device address. The GTS direction is specified as either transmit or receive. Each device may request one transmit GTS and/or one receive GTS. For each allocated GTS, the device shall be able to store its starting slot, length and direction. If a device has been allocated a receive GTS, it shall enable its receiver for the entirety of the GTS. In the same way, a PAN coordinator shall enable its receiver for the entirety of the GTS if a device has been allocated a transmit GTS. A device is instructed to request the allocation of a new GTS through the GTS request command, with GTS characteristics (direction, length,...) set according to the requirements of the intended application. On receipt of this command, the PAN coordinator shall send an acknowledgement frame. Following the ack transmission, the PAN coordinator shall first check if there is available capacity in the current superframe based on the remaining length of the CAP and the desired length of the requested GTS.

The superframe shall have available capacity if the maximum number of GTSs has not been reached and allocating a GTS of the desired length would not reduce the length of the CAP to less than aMinCAPLength. The PAN coordinator shall make its decision within aGTSDescPersistenceTime superframes. On receipt of the ack from the coordinator, the device shall continue to track the beacons and wait for at most aGTSDescPersistenceTime superframes. If no GTS decsriptor in the superframe, notify the next upper layer of failure. When the coordinator determines whether capacity is available for the requested GTS, it shall generate a GTS descriptor with the requested specifications and the short address of the requested device. It indicates the length and the start of the GTS in the superframe and notifies the next upper layer of the new GTS allocation. If there was not sufficient capacity to allocate the requested GTS, the start slot shall be set to 0 and the length to the largest GTS length that can currently be supported. This GTS descriptor shall remain in the beacon frame for aGTSPersistenceTime superframes. On receipt of the beacon frame, the device shall process the descriptor and notify the next upper layer of the success. In the same way, a device is instructed to request the deallocation of an existing GTS through the GTS request command using the characteristics of the GTS it wishes to deallocate. From this point on, the GTS to be deallocated shall not be used by the device. Then an ack from the PAN coordinator to the device. The PAN coordinator then deallocates the request of the GTS characteristics in the packet matches those in its allocation. The PAN coordinator shall also ensure that any gaps occurring in the CFP, appearing due to the deallocation of a GTS, are removed to maximize the length of the CAP. The MLME of the PAN coordinator shall also attempt to detect when a device has stopped using a GTS using the following rules: For a transmit frame GTS, the MLME of the PAN coordinator shall assume that the device is no longer using the GTS if a data frame is not received for at least 2  n superframes. For receive GTSs, the MLME of the PAN coordinator shall assume that the device is no longer using its GTS if an acknowledgement frame is not received for at least 2  n superframes. The value of n is equal to 28macBeaconOrder if 0  macBeaconOrder  8 and 1 if 9  macBeaconOrder  14.

### 4.8.11 MAC Frame Formats

The general MAC frame format is given in Figure 4.10.Each MAC frame consists of the following basic components:

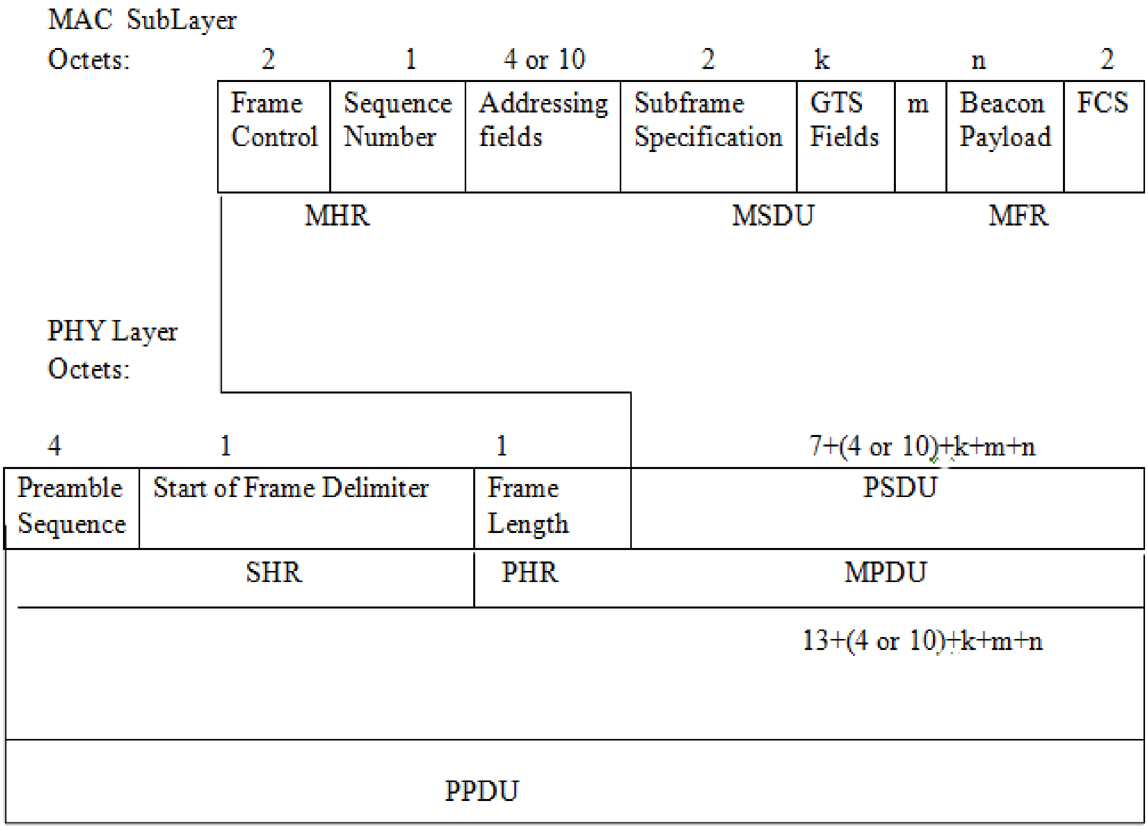


Figure 4.11: Schematic view of the beacon frame.

MHR, which comprises frame control, sequence number, and address information

A MAC payload of variable length, which contains information specific to the frame type. Acknowledgement frames do not contain a payload.

A MFR, which contains FCS. LR-WPAN defines 4 frame structures: beacon frame (Figure 4.10), data frame (Figure 4.11), acknowledgement frame (Figure 4.13), MAC command frame (Figure 4.15).MAC SubLayer The general MAC frame format is given in Figure 4.10.Each MAC frame consists of the following basic components:

MHR, which comprises frame control, sequence number, and address information

A MAC payload of variable length, which contains information specific to the frame type. Acknowledgement frames do not contain a payload.

A MFR, which contains FCS. LR-WPAN defines 4 frame structures: beacon frame (Figure 4.12), data frame (Figure 4.13), acknowledgement frame (Figure 4.13), MAC command frame (Figure 4.14).MAC SubLayer

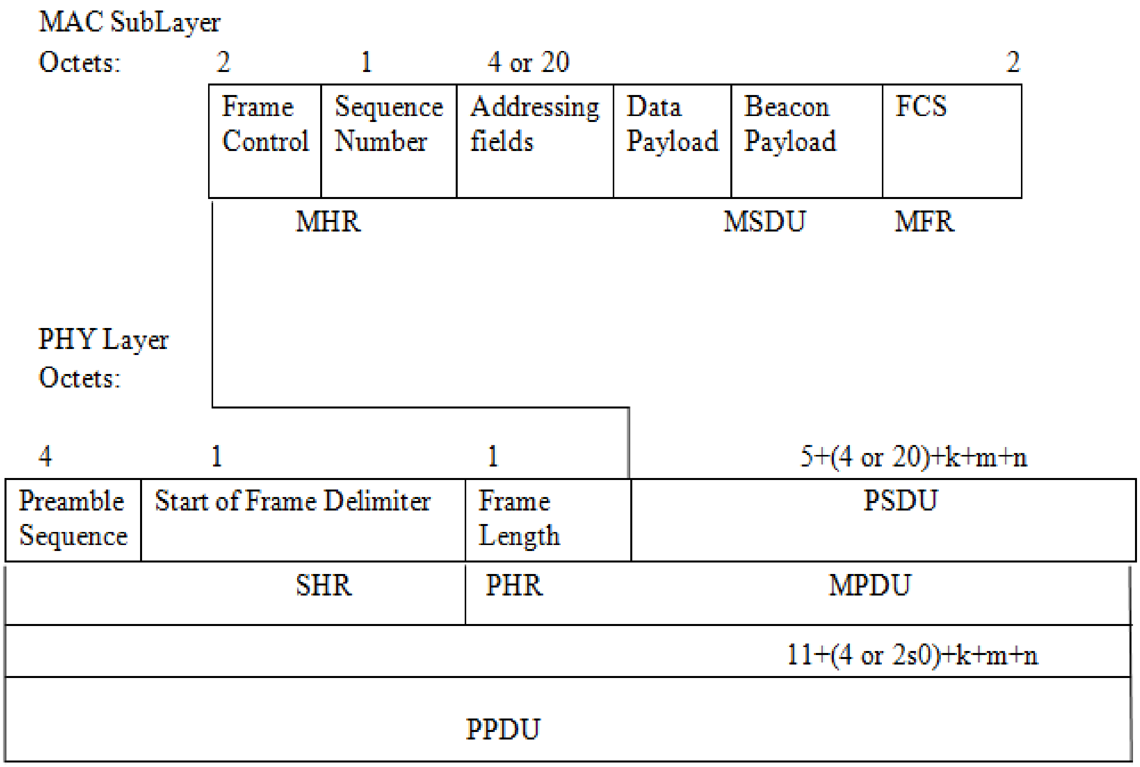


Figure 4.12: Schematic view of data frame.

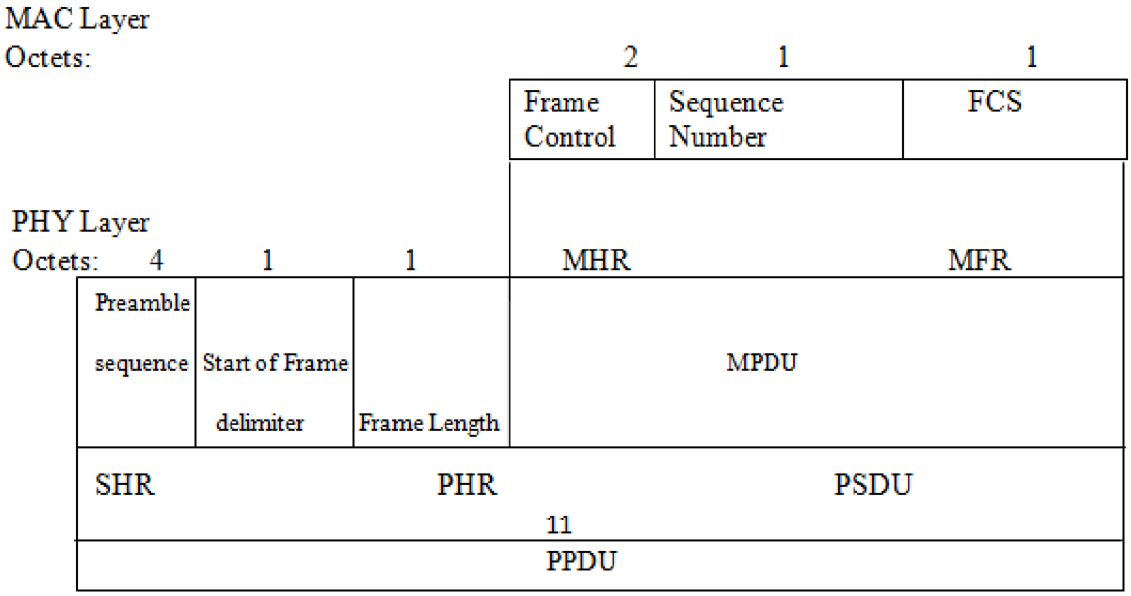


Figure 4.13: Schematic view of the acknowledgement frame.

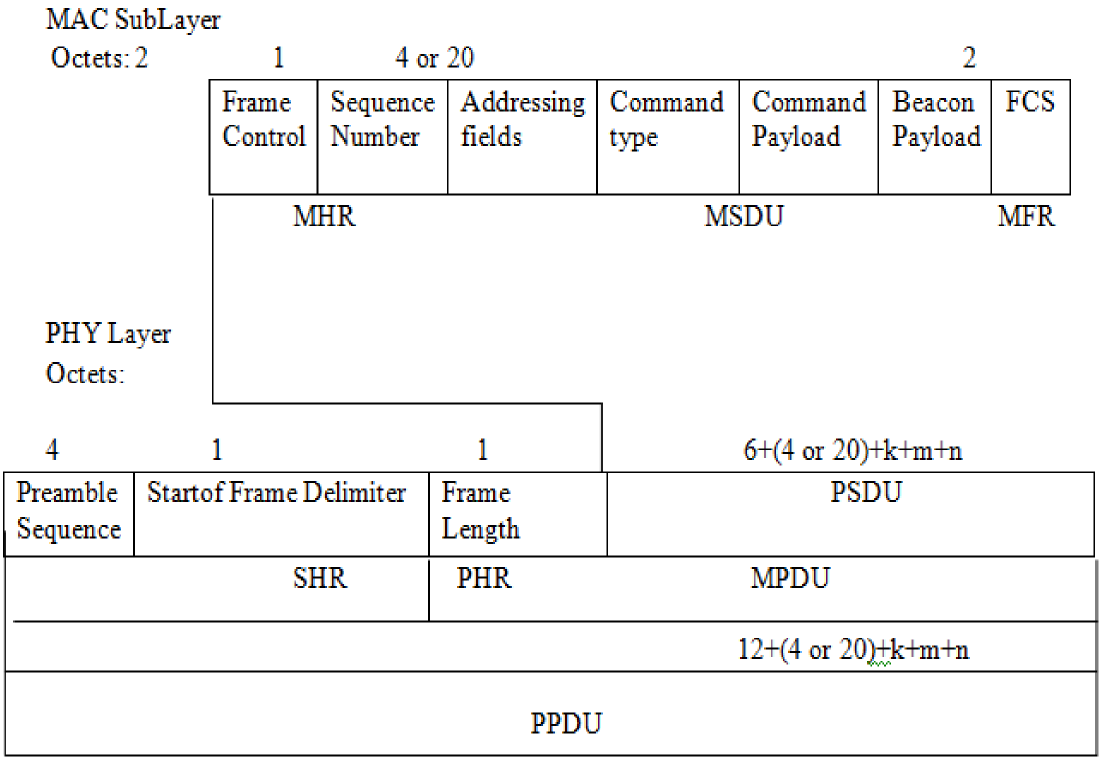


Figure 4.14: Schematic view of the MAC command frame.

## 4.9 Conclusion

With the standardization of the MAC and PHY almost complete, the focus is now on the upper protocol layers and application profiles. The ZigBee Alliance is taking the lead in this effort, and a first generation of results is expected by late 2002. In parallel, several leading semiconductor manufacturers are expected to announce the first generation of ICs. The development of this wireless solution within the standards organization has the advantage of bringing several views together to define a better solution. The quick development of the standard was due to the proactive participation of several developers and users of the technology. The focus of 802.15.4 development was on maintaining simplicity by concentrating on the essential requirements that will leverage a successful standard. The standard is targeting the residential and industrial market. It is expected that the industrial market will be the first to enable new products with focus on adding value through installation ease. The residential market will follow, taking advantage of lower cost due to the volume already driven by the industrial segment. IEEE 802.15.4 has already caught the attention of other communities, such as IEEE 1451 with a focus in sensor networking. It is expected that several users of proprietary wireless technologies will shift toward the standard solution due to the expected lower cost and performance improvement. The main goal of this effort has been to address applications that could benefit from wireless connectivity and enable new ones that cannot use higher-end wireless technologies. The value will be in the application, not in the technology. LR-WPAN is thus designed to be an enabler technology. The IEEE 802.15.4 complements other wireless networking technologies by occupying the lower end of the power consumption and data throughput space.

# CHAPTER 5
# CONCLUSION

In computer networking there is a great value of wireless sensor network which consists of power components, spatially distributed autonomous devices that use sensors which are interlinked or connected with each other, for performing the same functions collectively or cooperatively for the sake of checking and balancing the environmental factors. There may be thousands of sensor nodes in the field. For collecting data, the data traffic flow is from sensor to another sensor node or from sensor nodes to an access point (AP). By using various protocols this transmission of traffic is maintained at different layers. There are many issues related with designing of protocols which must be considered for saving energy, getting good throughput, minimum delay of transmitting data. Through this paper we discussed about various types of protocols of network at different layers by describing their components, design issues, advantages, disadvantages etc.

Then we measure the performance of some of these protocols by using the network simulator NS2 and describing the reasons of those outputs which we are getting at the graph. In this chapter we told about various types of MAC protocols, their working principle and usage of them. But we mainly concerned on the SMAC and a detail descriptio about it is given. Then we apply simulation to create a environment of wireless sensor network by both using IEEE802.11 and SMAC and fint out the throughput given by them. After comarising the output we can say that, though as a power saving protocol SMAC plays a vital role on MAC layer by reducing idle listening problem, overhean, control packet overhead, hidden terminal problem but by using this protocol we get lower throughput if no. of nodes is large and having a huge connection.

# REFERENCES

[1] Mark A. Perillo and Wendi B. Heinzelman, *Wireless Sensor Network Protocols*, IEEE 2004, ISBN 0-7803-8344-3.

[2] Kaveh Pahlavan and Prashant Krishnamurthy, *Networking Fundamentals: Wide, Local and Personal Area Communications*, John Wiley & Sons, Jul 7, 2009.

[3] Ahmad Abed Alhameed Alkhatib, Gurvinder Singh Baicher, *Wireless Sensor Network Architecture*, 2012 International Conference on Computer Networks and Communication Systems (CNCS 2012), IPCSIT vol.35(2012) IACSIT Press, Singapore

[4] Pereira, P., Grielo, A., Rocha,F., Nunes, M., Casaca, C., Almsrtrom, P., Johansson, M., *End-To-End Reliability In Wireless Sensor Networks: Survey and Research Challenges*, in EuroFGI Workshop on IP QoS and Traffic Control, Lisbon. 2007.

[5] Teerawat Issariyakul, Ekram Hossain, *Introduction to Network Simulator NS2*, Springer, 2nd ed. 2012.

[6] http://www.isi.edu/nsnam/ns/tutorial/

[7] A. Woo and D. Culler. *A Transmission Control Scheme for Media Access in Sensor Networks*. In Proceedings of ACM Mobicom, Rome, Italy, July 2001.

[8] Chipcon Corporation. *CC2500 Single Chip Low Cost Low Power RF Transceiver*, Data Sheet. 2005.

[9] P. Ferrari, A. Flammini, D. Marioli, and A. Taroni, *IEEE802.11 sensor networking,* IEEE Transactions on Instrumentation and Measurement, vol. 55, no. 2, pp. 615–619, 2006.

[10] Joseph Kabara1 and Maria Calle. *MAC Protocols Used by Wireless Sensor Networks and a General Method of Performance Evaluation.* Accepted 16 September 2011.

[11] T. V. Dam and K. Langendoen, *An Adaptive Energy- Efficient MAC Protocol for Wireless Sensor Networks*, 1st ACM Conf. Embedded Networked Sensor Sys., Los Angeles, CA, Nov. 2003.

[12] IlkerDemirkol, CemErsoy, and FatihAlagöz; Bogazici University. *MAC Protocols for Wireless Sensor Networks: A Survey*. IEEE Communications Magazine • April 2006

[13] Header file of SMAC (smac.h). (url: ~/ns-allion-2.35/ns-2.35/mac)

[14] Wei YE, John Heidemann, Deborah Estrin; Information Science Institute, University of Southern California, Los Angeles, USA, Computer Science Department, University of California, Los Angeles, USA. *An Energy-Efficient MAC Protocol for Wireless Sensor Networks*. Published Online June 2008 in SciRes .

[15] Comparing Energy-Saving MAC Protocols for Wireless Sensor Networks. G.P. Halkes ,T. van Dam and K.G. Langendoen ; Faculty of Electrical Engineering, Mathematics, and Computer Science Delft University of Technology, The Netherlands. *Comparing Energy-Saving MAC Protocols for Wireless Sensor Networks*. 2004 Kluwer Academic Publishers.

[16] Shio Kumar Singh, M P Singh, and D K Singh, *Routing Protocols in Wireless Sensor Networks –A Survey*, International Journal of Computer Science & Engineering Survey (IJCSES) Vol.1, No.2, November 2010.

[17] Kemal Akkaya , Mohamed Younis, *A survey on routing protocols for wireless sensor networks*, Ad Hoc Networks 3 (2005) 325–349

[18] S. Tilak et al., *A taxonomy of wireless microsensor network models, Mobile Computing and Communications Review* 6 (2) (2002) 28–36.

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

[20] Jamal N. Al-Karaki, Ahmed E. Kamal, *Routing Techniques in Wireless Sensor Networks: A Survey,* Year-2004, IEEE Wireless Communications, Pages: 6-28, volume-11.

[21] Luis Javier García Villalba, Ana Lucila Sandoval Orozco, Alicia Triviño Cabrera and Cláudia Jacy Barenco Abbas, *Routing Protocols in Wireless Sensor Networks*, Journal: Sensors, Year: 2009, Vol: 9, Issue: 11, Pages/record No.: 8399-8421.

[22] Rajashree.V.Biradar, V.C .Patil, Dr. S. R. Sawant, Dr. R. R. Mudholkar, *Classification and comparison of routing protocols in wireless sensor network*, Ubiquitous Computing and Communication Journal, Special Issue on Ubiquitous Computing Security Systems, Volume: Ubiquitous Computing Security Systems Publishing Date: 8/20/2009.

[23] M. Aslam, N. Javaid, A. Rahim, U. Nazir, A. Bibi, Z. A. Khan, *Survey of Extended LEACH-Based Clustering Routing Protocols for Wireless Sensor Networks*, Journal-ref: 5th AHPCN in conjunction with 14th HPCC-2012, Liverpool, UK Subjects: Networking and Internet Architecture (cs.NI) , 11 July 2012.

[24] P.T.V.Bhuvaneswari and V.Vaidehi, *Enhancement techniques incorporated in LEACH- a survey*, Indian Journal of Science and Technology, Vol.2 No 5 (May 2009) ISSN: 0974-6846.

[25] José A. Gutiérrez, Edgar H. Callaway, Raymond L. Barrett, Chapter-1,2,3.
*Low-Rate Wireless Personal Area Networks: Enabling Wireless Sensors with IEEE 802.15.4.*

[26] Improving the IEEE 802.15.4 Slotted CSMA/CA MAC for Time-Critical Events in Wireless Sensor Networks. Writer-Anis Koubaa, Mário Alves , Bilel Nefzi ,Ye-Qiong Song. TR-060704, Version: 1.0 Date: Jul 2006.

Department of Computer Science, National Pingtung University of Education, No. 4-18, Ming Shen Rd., Pingtung 90003,Taiwan. *The development of enhancing mechanisms for improving the performance of IEEE 802.15.4* Accepted 27 May, 2010.

[27] Luca De Nardis and Maria-Gabriella Di Benedetto Infocom Department School of Engineering University of Rome La Sapienza Rome, Italy 001844th workshop on positioning, navigation and communication 2007 (wpnc'07), hannover, germany. Overview of the *IEEE 802.15.4/4a standards for low data rate Wireless Personal Data Networks*.

[28] Ed Callaway, Paul Gorday, and Lance Hester, Motorola Laboratories, Jose A. Gutierrez and Marco Naeve, Eaton Corporation, Bob Heile, Appairent Technologies, Venkat Bahl, Philips Semiconductors.*Home Networking with IEEE 802.15.4:A Developing Standard for Low-Rate Wireless Personal Area Networks*.

[29] Dusan Stevanovic. *Zigbee / IEEE 802.15.4 Standard*. June 20, 2007.