ADVANCED LEACH PROTOCOL FOR WIRELESS SENSOR NETWORKS

by

DEVIREDDY BRUNDA

February 2021

List of figures

List of Tables

Abbreviations

BS	Base Station
CH	Cluster Head
LEACH	Lower Energy Adaptive Clustering Hierarchy
TDMA	Time Division Multiple Access
WSN	Wireless Sensor Network

Notations

English Symbols

E_{DA}	Data Aggregation Energy
E_{O}	Initial Energy per node
E_{RX}	Receiver Energy per node
E_{TX}	Transmitter energy per node

Greek Symbols

Σ	Summation

Miscellaneous Symbols

ϵ	Epsilon
$\int$	Integral

ABSTRACT

Wireless Sensor Networks (WSNs) provide several types of applications to lead a comfortable and smart-economic life. WSNs are of utmost importance as they are responsible for maintaining network paths, data forwarding, and ensuring reliable multi-hop communication. WSNs face problems in two aspects, one is energy and other is lifetime of the network. In order to overcome these limitations, an advanced LEACH protocol, the protocol of cluster routing has been developed in the proposed work. LEACH is extended and enhanced in the following ways. Firstly, base station selects cluster heads for the network with a restriction on the energy of the node. Secondly, selection of a cluster head by a node is based on the lowest degree of distance from the sink or base station to cluster head and from cluster head to the node to decrease power consumption in the whole network. Simulation has been done in MATLAB to show the comparison between LEACH protocol and advanced LEACH protocol.

Keywords: Routing, LEACH Protocol, Wireless Sensor Networks, Cluster Head, Base Station.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

WSNs are self-configured and Infrastructure less, wireless networks. They monitor physical or environmental conditions, such as temperature, pressure, sound, or pollutants etc. and pass their data through the network to a main location called BS or Sink. The data is then observed and analysed at the BS.

Routing is one of the important technologies for wireless sensor Networks. It has to send collected data to original nodes and then to the destination node with one hop by hop. This involves two aspects: one is to find the optimal path between two nodes and second is to send the collected data in the optimal path. As there is no communication path between source and destination nodes, neighbouring nodes have to send the collected data to the destination node in the form of a multi-hop. Also, the sensor node is powered by a limited battery. So, Energy Consumption of WSN Nodes and balance of the network energy is an important factor. Network lifetime is one of the major factors to be considered for adopting any protocol.

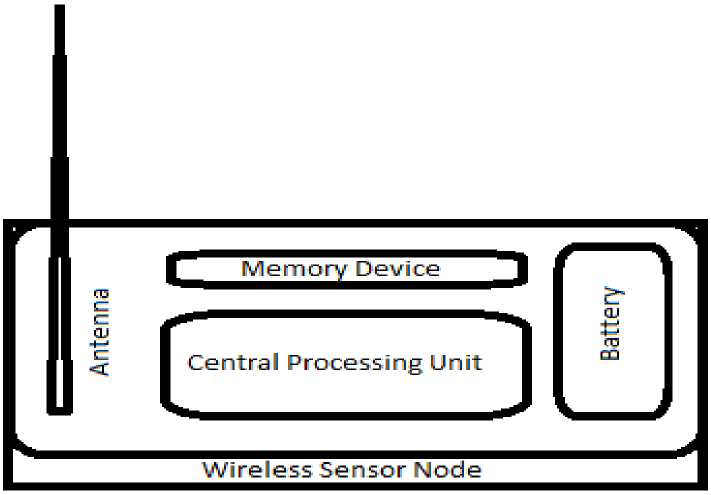

Fig. 1.1 Wireless Sensor Node architecture.

1.1 EXISTING SYSTEM

LEACH protocol is a TDMA based hierarchical Routing protocol. The main objective of this protocol is to increase the lifespan of wireless sensor networks by lowering the energy consumption required to create and maintain Cluster Heads (CH). LEACH Protocol partitions the nodes into clusters and for each cluster, CH is elected by BS. Only CH is allowed to gather the data, accumulate, aggregate and forward the data to base station.
Therefore, Cluster Heads consume more power. Figure1 shows an example of CH.

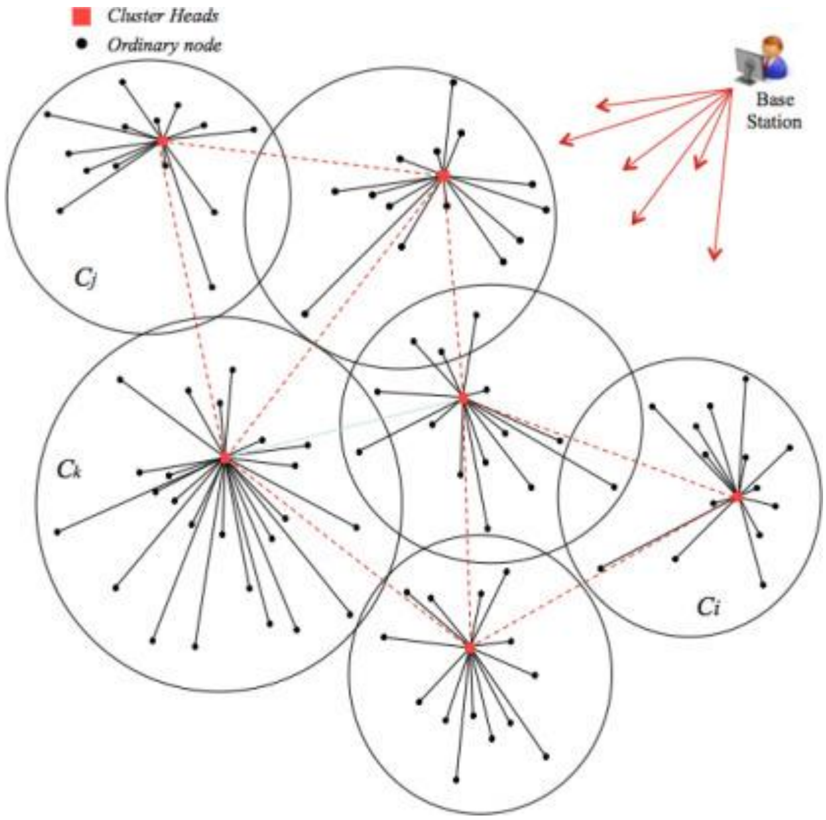

Fig. 1.2 Cluster Head – overview

The LEACH protocol divides the lifetime of the network into a number of rounds as shown in Figure2. Every round consists of two phases: the setup phase and the steady phase.

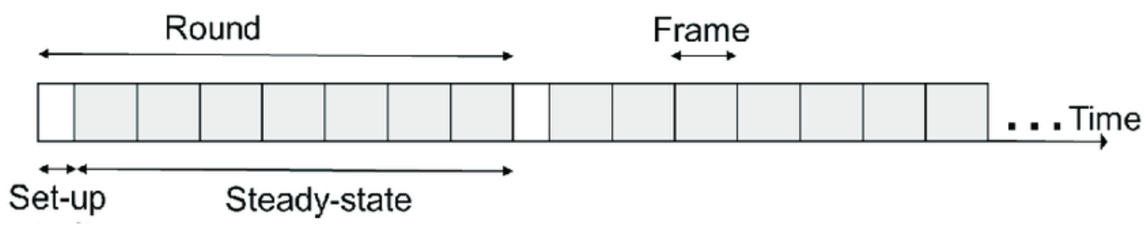

Fig. 1.3 LEACH Protocol Process

Setup phase: This phase consists of three steps as follows
1) BS selects CH and CH advertises that it is a CH now.
2) Non-cluster nodes select nearest CH.
3) CH creates a TDMA schedule for the nodes of its cluster to transmit data and sends the schedule to its nodes.

The selection of the cluster heads is based on the threshold T(n). The node will become a CH for the current round if the number is less than that threshold. The threshold value is based upon the chosen fraction to become a cluster-head (p), the present round (r), and the nodes that have not become the CH in the last 1/p rounds are expressed by G. The formulae are as follows:

$$T(n) = \begin{cases} \dfrac{T}{1 - p\left(r \bmod \dfrac{1}{p}\right)} & \text{if } n \in G \\ 0 & \text{Otherwise} \end{cases} \tag{1.1}$$

Steady phase:

In steady phase, nodes send information to the respective CH in the allotted timeslot which is called a *frame*. The CH then accumulates the received data from nodes and sends it to BS.

The CH which is once chosen for the present round cannot again become the CH until all the other cluster members become CH for once. Flowcharts for setup phase and leach protocol are show in fig 3 and fig 4.

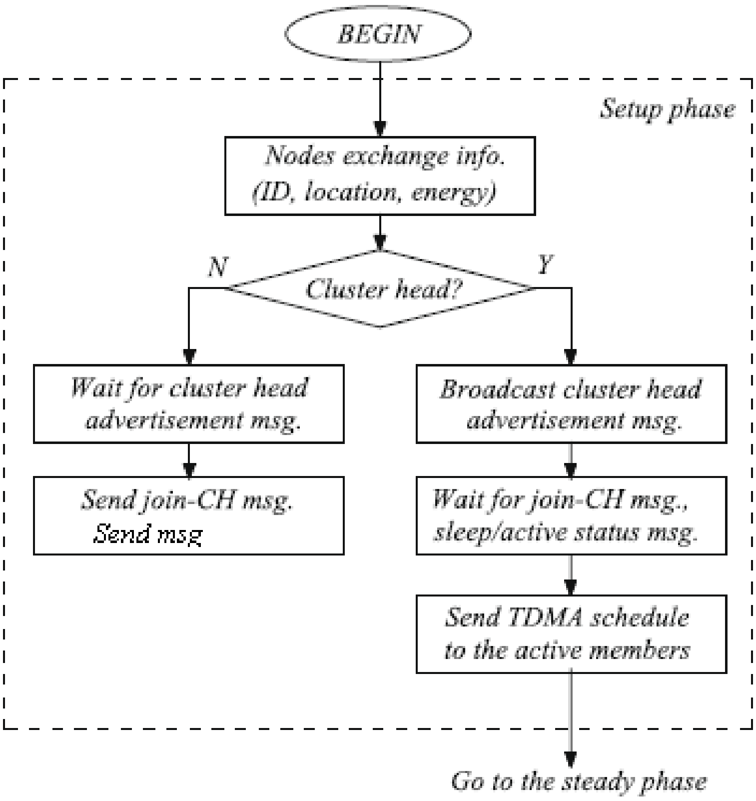

Fig. 1.4 Flowchart for setup phase

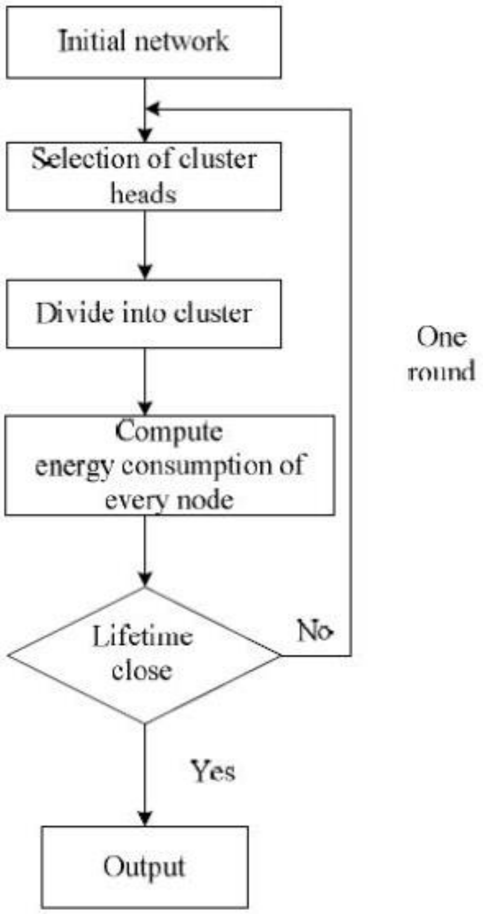

Fig. 1.5 Flowchart of Leach Protocol

1.1.1 LIMITATIONS OF EXISTING SYSTEM

1. Position of Base Station: In general, the Base Station is located at the end due to which some clusters that are at the other end of the network require more energy to send the data and may become dead nodes in less time.

2. Total distance from node to CH to BS is not considered while a node selects a CH. This leads to faster decrement of Average residual energy.

3. Energy of CH is not considered while selecting a node as CH.

1.2 PROPOSED SYSTEM

As number of dead nodes increase the number of existing nodes decreases which in turn decreases the system lifetime and reduces the efficiency of the system. Some modifications have been done in the existing protocol to improve the lifetime of the network.

1.2.1 Placement of Sink:

Placement of BS plays a crucial role in WSNs. It is found that cluster heads spend more energy when the sink is at a long distance. The optimum position of the BS can be calculated mathematically and then the results are simulated based on that position. Consider N as total number of nodes and node i is established of each other at (Pi, Qi) in the concerned region R. Now consider the below function

$$F(x,y) = E\left[\sum_{i=1}^{N} E_{TX}(D_i)\right] \tag{1.2}$$

Where, ETX=Energy of the transmitting

Di= Distance between sink (x, y) and node (Pi, Qi)

$$Di = \sqrt{(x - Pi)2 + (y - Qi)2} \tag{1.3}$$

$$F(x, y) = E\left[\sum_{i=1}^{N} E_{TX}\left(\sqrt{(x - Pi)2 + (y - Qi)2}\right)\right] \tag{1.4}$$

$$= N \int\int_{p,q \in R} E_{tx}\left(\sqrt{(x - Pi)2 + (y - Qi)2}\right) f(p, q)\, dpdq \tag{1.5}$$

Now, to minimize the function we differentiate. Final result is,

$$x = \iint p.\, f(p, q)\, dpdq \tag{1.6}$$

$$y = \iint q.\, f(p, q)\, dpdq \tag{1.7}$$

This shows that the center part of the region under consideration is the optimal position for the sink node. Hence, in our proposed version of the LEACH protocol, we have changed the position of the sink node in the middle corner and compared it with the general condition i.e. where the sink is at the end. Moreover, we see that the outcome is more efficient than the former case. We have computed the number of dead nodes in different cases and then compared it with the original model.

1.2.2 CALCULATION OF RESIDUE ENERGY:

The energy consumed in a round for the above model is given by

$$S(i).E = \begin{cases} S(i).E - (AE + A \in as\ d2, & d < d0 \\ S(i).E - (AE + A \in as\ d4, & d > d0 \end{cases} \quad (1.8)$$

where $d0 = \sqrt{(Efs/Emp)}$

E=Energy consumed in transmission of 1 bit
A= number of bits sent by the transmitter.
d=distance between CH and sink node.
Efs =Represent the amplification in free space
Emp=Represent the multipath fading coefficient

1.2.3 CALCULATION OF AVERAGE ENERGY

Average energy of the entire network is calculated as

$$En = \sum_{i=1}^{N} Ei \quad (1.9)$$

$$Eavg = En / n \quad (1.10)$$

where Ei = Energy of i^{th} node.

En = Total Energy of the network.

$Eavg$ = Average energy of the network.

Including average energy as a condition (i.e., for a node to become CH its energy should be greater than the average energy of the entire network) number of dead nodes will be decreased because of the fact that if a node has lesser energy and if it becomes CH then its energy will be drained out faster and it becomes a dead node. As, the nodes with lesser energy will not be selected as CH's number of dead nodes decreases which in turn increases the lifetime of the network.

1.2.4 IMPROVED VERSION OF LEACH PROTOCOL:

The Proposed Leach Protocol is similar to LEACH Protocol. It has two stages: setup phase and steady phase. The main difference occurs in the setup phase which leads to great improvement in terms of lifetime of WSNs.

In the setup phase, all the nodes select a random number from 0 to 1, then if the number is less than or equal to threshold (T(N), as calculated previously) and if the node energy is greater than the average energy (*Eavg*), it becomes a cluster head (CH) else the node remains ordinary. The process continues until all cluster heads are chosen. After all the CHs are chosen, they broadcast advertisement messages to the overall network informing that it became a CH.

Here comes another enhancement where a node must select a CH according to total distance to reach the base station (i.e., distance from node to CH + distance from CH to BS), not a CH that is closest to it. Distances are calculated according to the equation 1.11 and the minimum of the total distance is chosen.

$$D\ (x, y) = \sqrt{}\ (x_1 - x2)^2 + (y_1 - y_2)^2 \tag{1.11}$$

After the ordinary node selects a cluster head that has the minimum distance, it sends a message to inform a cluster head that it will be a member of its cluster. After a request message is received from ordinary nodes, the cluster heads make a time-division multiple access (TDMA) schedule for each member in its cluster. The TDMA schedule assigns a timescale for every ordinary member node in it; it means that every ordinary member node is only allowed to send in its specific timescale, or else it has to wait (go in sleep mode). After the setup stage is completed, a steady stage starts same as in the LEACH protocol.

1.3 ASSUMPTIONS OF NETWORK MODEL:

1.There exists only one base station and it is fixed.
2.The sensor nodes are homogeneous and energy constrained with uniform energy.
3.Clusters and nodes are static. Normal nodes transmit directly to their respective cluster heads within a particular cluster.
4.The base station is situated at the centre of the area space (optimal placement of sink).

1.4 MERITS AND DEMERITS

Merits

1. Proposed protocol is Energy efficient protocol
2. The life time of the network is increased in the Advanced LEACH protocol.

Demerits

1.BS must be located at center for the proposed protocol which may not be possible for some circumstances.
2. Small amount of time will be taken for the proposed protocol.

CHAPTER 2
SOURCE CODE

```
%%%%%%Advanced LEACH%%%%%%
clear;
xm=300;
ym=300;
sink.x=0.5*xm;
sink.y=0.5*ym;
n = 200;
p=0.1;
Eo=0.5;
ch=n/10;
ETX=50*0.000000001;
ERX=50*0.000000001;
Efs=10*0.000000000001;
Emp=0.0013*0.000000000001;
Efs1=Efs/10;
Emp1=Emp/10;
EDA=5*0.000000001;
a=Eo/2;
rmax=500;
h=100;
s=2;
do=sqrt(Efs/Emp);
do1=sqrt(Efs1/Emp1);
for h=1:1
    S(n+1).xd=sink.x;
    S(n+1).yd=sink.y;
    Et=0;
end
for i=1:1:n
    S(i).xd=rand(1,1)*xm;
    S(i).yd=rand(1,1)*ym;
    S(i).G=0;
    S(i).type='N';
    S(i).E=Eo;
    Et=Et+S(i).E;
    figure(h*1)
      plot(S(i).xd,S(i).yd,'bo');
      text(S(i).xd+1,S(i).yd-0.5,num2str(i));
      hold on;
end
plot(S(i).xd,S(i).yd,'b^','MarkerSize', 5, 'MarkerFaceColor', 'b');
text(S(n+1).xd+1,S(n+1).yd-0.5,num2str(n+1));
title("Nodes formation")
hold off ;
countCHs=0;
```

```
cluster=1;
flag_first_DEAD=0;
flag_teenth_DEAD=0;
flag_all_DEAD=0;
DEAD=0;
first_DEAD=0;
teenth_DEAD=0;
all_DEAD=0;
ALLIVE=n;
for r=0:1:rmax
  if(mod(r, round(1/p) )==0)
  for i=1:1:n
     S(i).G=0;
  end
  end
DEAD=0;
for i=1:1:n
  if (S(i).E<=0)
     DEAD=DEAD+1;
     if (DEAD==1)
       if(flag_first_DEAD==0)
         first_DEAD=r;
         flag_first_DEAD=1;
       end
     end
     if(DEAD==0.1*n)
       if(flag_teenth_DEAD==0)
         teenth_DEAD=r;
         flag_teenth_DEAD=1;
       end
     end
     if(DEAD==n)
       if(flag_all_DEAD==0)
         all_DEAD=r;
         flag_all_DEAD=1;
       end
     end
  end
  if S(i).E>0
     S(i).type='N';
  end
end
STATS.DEAD(r+1)=DEAD;
STATS.ALLIVE(r+1)=ALLIVE-DEAD;
countCHs=0;
cluster=1;
if   S(i).type=='C' && S(i).E>a
for j=1:1:ch
    countCHs=countCHs+1;
    S(i).type='C';
```

```
           S(i).G=round(1/p)-1;
           C(cluster).xd=S(i).xd;
           C(cluster).yd=S(i).yd;
    distance=sqrt( (S(i).xd-(S(n+1).xd) )^2 + (S(i).yd-(S(n+1).yd) )^2 );
           C(cluster).distance=distance;
           C(cluster).id=i;
           X(cluster)=S(i).xd;
           Y(cluster)=S(i).yd;
           cluster=cluster+1;
distance;
           if (distance>do)
               S(i).E=S(i).E- ( (ETX+EDA)*(4000) + 
Emp*4000*(distance*distance*distance*distance ));
           end
           if (distance<=do)
               S(i).E=S(i).E- ( (ETX+EDA)*(4000)  + Efs*4000*(distance * distance ));
           end
end
else
Etotal=0;
    for i=1:n
        if S(i).E>0
            Etotal=Etotal+S(i).E;
        end
    end
    Eavg=Etotal/n;
    STATS.TotalENERGY(r+1)=Etotal;
    STATS.AvgENERGY(r+1)=Eavg;
for i=1:1:n
  if(S(i).E>0)
  temp_rand=rand;
  if ( (S(i).G)<=0)
      if(temp_rand<= (p/(1-p*mod(r,round(1/p)))))
           if(S(i).E>Eavg)
           countCHs=countCHs+1;
            S(i).type='C';
           S(i).G=round(1/p)-1;
           C(cluster).xd=S(i).xd;
           C(cluster).yd=S(i).yd;
          distance=sqrt( (S(i).xd-(S(n+1).xd) )^2 + (S(i).yd-(S(n+1).yd) )^2 );
           C(cluster).distance=distance;
           C(cluster).id=i;
           X(cluster)=S(i).xd;
           Y(cluster)=S(i).yd;
           cluster=cluster+1;
           distance;
           if (distance>do)
               S(i).E=S(i).E- ( (ETX+EDA)*(4000) + 
Emp*4000*(distance*distance*distance*distance ));
           end
```

```
          if (distance<=do)
             S(i).E=S(i).E- ( (ETX+EDA)*(4000)  + Efs*4000*(distance * distance ));
          end
          end
       end
       end
  end
end
end
STATS.COUNTCHS(r+1)=countCHs;
for i=1:1:n
  if ( S(i).type=='N' && S(i).E>0 )
    if(cluster-1>=1)
      min_distance=Inf;
      min_distance_cluster=0;
      for c=1:1:cluster-1
          temp=min(min_distance,(sqrt( (S(i).xd-C(c).xd)^2 + (S(i).yd-C(c).yd)^2 )+sqrt(
(S(n+1).xd-C(c).xd)^2 + (S(n+1).yd-C(c).yd)^2 )));
          if ( temp<min_distance )
             min_distance=temp;
             min_distance_cluster=c;
          end
      end
          min_distance;
          if (min_distance>do1)
             S(i).E=S(i).E- ( ETX*(4000) + Emp1*4000*( min_distance *min_distance *
min_distance * min_distance));
          end
          if (min_distance<=do1)
             S(i).E=S(i).E- ( ETX*(4000) + Efs1*4000*( min_distance * min_distance));
          end
               S(C(min_distance_cluster).id).E =S(C(min_distance_cluster).id).E- ( (ERX +
EDA)*4000 );

       S(i).min_distance=min_distance;
       S(i).min_distance_cluster=min_distance_cluster;
   else
       min_distance=sqrt( (S(i).xd-S(n+1).xd)^2 + (S(i).yd-S(n+1).yd)^2 );
          if (min_distance>do)
             S(i).E=S(i).E- ( ETX*(4000) + Emp*4000*( min_distance *min_distance *
min_distance * min_distance));
          end
          if (min_distance<=do)
             S(i).E=S(i).E- ( ETX*(4000) + Efs*4000*( min_distance * min_distance));
          end
    end
 end
end
En=0;
for i=1:n
```

```
    if S(i).E<=0
        continue;
    end
    En=En+S(i).E;
end
ENERGY(r+1)=En;
STATS.ENERGY(h,r+1)=En;
end
for r=0:rmax
    STATS.DEAD(h+1,r+1)=sum(STATS.DEAD(:,r+1))/h;
    STATS.ALLIVE(h+1,r+1)=sum(STATS.ALLIVE(:,r+1))/h;
    STATS.ENERGY(h+1,r+1)=sum(STATS.ENERGY(:,r+1))/h;
    STATS.COUNTCHS(h+1,r+1)=sum(STATS.COUNTCHS(:,r+1))/h;
end
%%%%% ACTUAL LEACH %%%%%
xm=300;
ym=300;
sink.x=xm;
sink.y=ym;
n=200;
p=0.1;
Eo=0.5;
ETX=50*0.000000001;
ERX=50*0.000000001;
Efs=10e-12;
Emp=0.0013e-12;
EDA=5*0.000000001;
rmax=500;
do=sqrt(Efs/Emp);
Et=0;
for h=1:1
    S(n+1).xd=sink.x;
    S(n+1).yd=sink.y;
    Et=0;
for i=1:1:n
    S(i).xd=rand(1,1)*xm;
    XR(i)=S(i).xd;
    S(i).yd=rand(1,1)*ym;
    YR(i)=S(i).yd;
    distance=sqrt( (S(i).xd-(S(n+1).xd) )^2 + (S(i).yd-(S(n+1).yd) )^2 );
    S(i).distance=distance;
    S(i).G=0;
    S(i).type='N';
    S(i).E=Eo;
    Et=Et+S(i).E;
    figure(h*2)
      plot(S(i).xd,S(i).yd,'b^','MarkerSize', 5, 'MarkerFaceColor', 'b');
      text(S(i).xd+1,S(i).yd-0.5,num2str(i));
      hold on;
end
```

```
plot(S(n+1).xd,S(n+1).yd,'o', 'MarkerSize', 12, 'MarkerFaceColor', 'r');
text(S(n+1).xd+1,S(n+1).yd-0.5,num2str(n+1));
title("Nodes formation");
hold off ;
countCHs=0;
cluster=1;
flag_first_DEAD=0;
flag_half_DEAD=0;
flag_all_DEAD=0;
first_DEAD=0;
half_DEAD=0;
all_DEAD=0;
ALLIVE=n;
for r=0:1:rmax
    if(mod(r, round(1/p) )==0)
        for i=1:1:n
            S(i).G=0;
            S(i).cl=0;
        end
    end
    DEAD=0;
     for i=1:1:n
        if (S(i).E<=0)
            DEAD=DEAD+1;
            if (DEAD==1)
              if(flag_first_DEAD==0)
                 first_DEAD=r;
                 flag_first_DEAD=1;
              end
            end
            if(DEAD==0.5*n)
              if(flag_half_DEAD==0)
                  half_DEAD=r;
                  flag_half_DEAD=1;
              end
            end
            if(DEAD==n)
              if(flag_all_DEAD==0)
                  all_DEAD=r;
                  flag_all_DEAD=1;
              end
            end
        end
        if S(i).E>0
            S(i).type='N';
        end
     end
        STATISTICS(2).DEAD(h,r+1)=DEAD;
        STATISTICS(2).ALLIVE(h,r+1)=ALLIVE-DEAD;
   countCHs=0;
```

```
    cluster=1;
    for i=1:1:n
        if(S(i).E>0)
            temp_rand=rand;
            if ( (S(i).G)<=0)
                if ( temp_rand <= ( p/ ( 1 - p * mod(r,round(1/p)) )))
                    countCHs=countCHs+1;
                    S(i).type='C';
                    S(i).G=round(1/p)-1;
                    C(cluster).xd=S(i).xd;
                    C(cluster).yd=S(i).yd;
                    distance=sqrt( (S(i).xd-(S(n+1).xd) )^2 + (S(i).yd-(S(n+1).yd) )^2 );
                    C(cluster).distance=distance;
                    C(cluster).id=i;
                    X(cluster)=S(i).xd;
                    Y(cluster)=S(i).yd;
                    cluster=cluster+1;
                    distance;
                    if (distance>do)
                        S(i).E=S(i).E- ( (ETX+EDA)*(4000) + Emp*4000*(
distance*distance*distance*distance ));
                    end
                    if (distance<=do)
                        S(i).E=S(i).E- ( (ETX+EDA)*(4000)  + Efs*4000*( distance * distance ));
                    end
                end
            end
        end
    end

         STATISTICS(2).COUNTCHS(h,r+1)=countCHs;
     for i=1:1:n
       if ( S(i).type=='N' && S(i).E>0 )
        if(cluster-1>=1)
       min_distance=sqrt( (S(i).xd-S(n+1).xd)^2 + (S(i).yd-S(n+1).yd)^2 );
       min_distance_cluster=0;
         for c=1:1:cluster-1
           temp=min(min_distance,sqrt( (S(i).xd-C(c).xd)^2 + (S(i).yd-C(c).yd)^2 ) );
           if ( temp<min_distance )
               min_distance=temp;
               min_distance_cluster=c;
           end
         end
       if(min_distance_cluster~=0)
            min_distance;
            if (min_distance>do)
                S(i).E=S(i).E- ( ETX*(4000) + Emp*4000*( min_distance * min_distance *
min_distance * min_distance));
            end
            if (min_distance<=do)
```

```
            S(i).E=S(i).E- ( ETX*(4000) + Efs*4000*( min_distance * min_distance));
         end

         S(C(min_distance_cluster).id).E = S(C(min_distance_cluster).id).E- ( (ERX +
EDA)*4000 );
      else
         min_distance;
         if (min_distance>do)
            S(i).E=S(i).E- ( ETX*(4000) + Emp*4000*( min_distance * min_distance *
min_distance * min_distance));
         end
         if (min_distance<=do)
            S(i).E=S(i).E- ( ETX*(4000) + Efs*4000*( min_distance * min_distance));
         end
      end
       S(i).min_distance=min_distance;
      S(i).min_distance_cluster=min_distance_cluster;
   else
         min_distance=sqrt( (S(i).xd-S(n+1).xd)^2 + (S(i).yd-S(n+1).yd)^2 );
         if (min_distance>do)
            S(i).E=S(i).E- ( ETX*(4000) + Emp*4000*( min_distance * min_distance *
min_distance * min_distance));
         end
         if (min_distance<=do)
            S(i).E=S(i).E- ( ETX*(4000) + Efs*4000*( min_distance * min_distance));
         end
   end
  end
end
 En=0;
for i=1:n
   if S(i).E<=0
      continue;
   end
   En=En+S(i).E;
end
ENERGY(r+1)=En;
STATISTICS(2).ENERGY(h,r+1)=En;
end
first_DEAD_LEACH(h)=first_DEAD;
half_DEAD_LEACH(h)=half_DEAD;
all_DEAD_LEACH(h)=all_DEAD;
end
for r=0:rmax
   STATISTICS(2).DEAD(h+1,r+1)=sum(STATISTICS(2).DEAD(:,r+1))/h;
   STATISTICS(2).ALLIVE(h+1,r+1)=sum(STATISTICS(2).ALLIVE(:,r+1))/h;
   STATISTICS(2).ENERGY(h+1,r+1)=sum(STATISTICS(2).ENERGY(:,r+1))/h;
   STATISTICS(2).COUNTCHS(h+1,r+1)=sum(STATISTICS(2).COUNTCHS(:,r+1));
end
first_DEAD=sum(first_DEAD_LEACH)/h;
```

```
half_DEAD=sum(half_DEAD_LEACH)/h;
all_DEAD=sum(all_DEAD_LEACH)/h;
r=0:rmax;
figure(3)
plot(r,STATS.DEAD(h+1,r+1))
xlabel('Rounds');
ylabel('DEAD Nodes');
title('Rounds VS DEAD Nodes');
hold on
plot(r,STATISTICS(2).DEAD(h+1,r+1));
legend('Advanced LEACH','LEACH')
hold off
figure(4)
plot(r,STATS.ALLIVE(h+1,r+1))
xlabel('Rounds');
ylabel('Live Nodes')
title('Rounds VS Live Nodes');
hold on
plot(r,STATISTICS(2).ALLIVE(h+1,r+1))
legend('Advanced LEACH','LEACH')
hold off
figure(5)
plot(r,STATS.ENERGY(h+1,r+1))
xlabel('Rounds')
ylabel('Average ENERGY')
title('Average Residual ENERGY')
hold on
plot(r,STATISTICS(2).ENERGY(h+1,r+1))
legend('Advanced LEACH','LEACH')
hold off
```

CHAPTER 3
SNAPSHOTS

3.1 EXPERIMENTAL ANALYSIS

MATLAB software is used for the simulation of the above code. In advanced LEACH protocol a homogenous sensor network with 200 sensors randomly distributed in the 300m*300m area as shown in fig 3.1. Red dot denotes the BS other triangles are nodes.

Advanced LEACH:

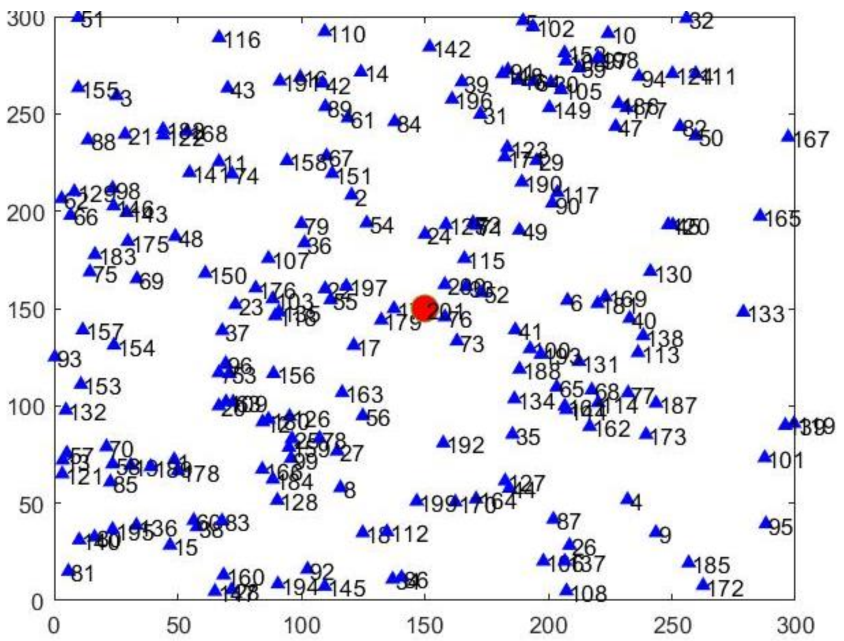

Fig. 3.1 Formation of nodes in advanced LEACH protocol

PARAMETERS	VALUES
Dimensions	300*300 m^2
BS Coordinates (X, Y)	(150, 150)
No. of Nodes	200
No. of rounds	500
Probability of a node to become CH(p)	0.1
Initial Energy(E_0)	0.5 J
E_{TX}	50 nJ/bit
E_{RX}	50 nJ/bit
E_{DA}	5 nJ/bit
Packet Size	4000 bits

Table 3.1 Simulation parameters for Advanced LEACH protocol

LEACH:

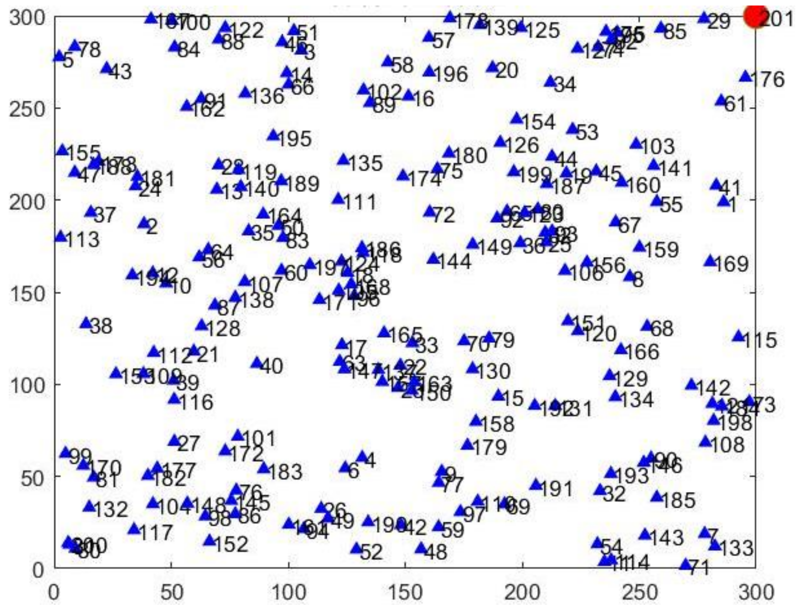

Fig. 3.2 Formation of nodes in LEACH protocol

PARAMETERS	VALUES
Dimensions	300*300 m^2
BS Coordinates (X, Y)	(300, 300)
No. of Nodes	200
No. of rounds	500
Probability of a node to become CH(p)	0.1
Initial Energy(E_0)	0.5 J
E_{TX}	50 nJ/bit
E_{RX}	50 nJ/bit
E_{DA}	5 nJ/bit
Packet Size	4000 bits

Table 3.2 Simulation parameters for LEACH protocol

3.2 GRAPHICAL ANALYSIS:

Both the protocols are compared based on the following parameters

1. Average Energy.
2. Number of dead nodes.
3. Number of live nodes.

Average Energy:

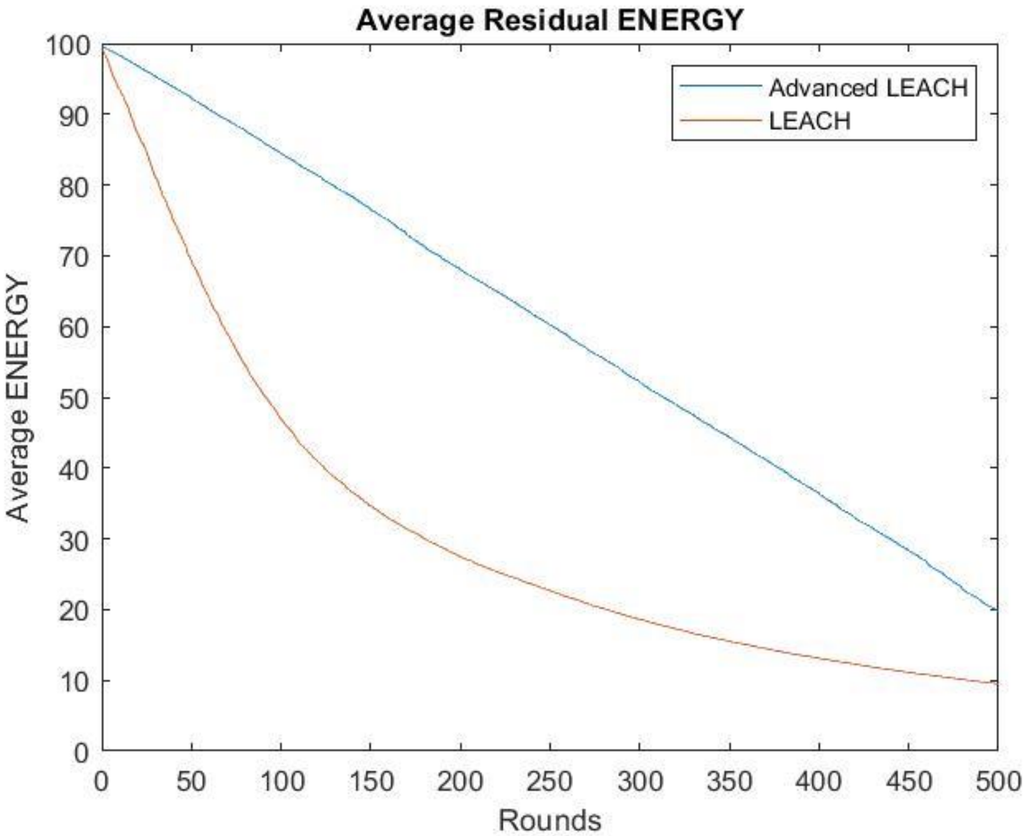

Fig. 3.3 Average Residual Energy comparison graph.

From the Above plots, it is visible that Average Energy in Advanced LEACH is more than that of LEACH. This proves that Advanced LEACH is an Energy-Efficient Protocol when compared to LEACH protocol.

Number of dead nodes:

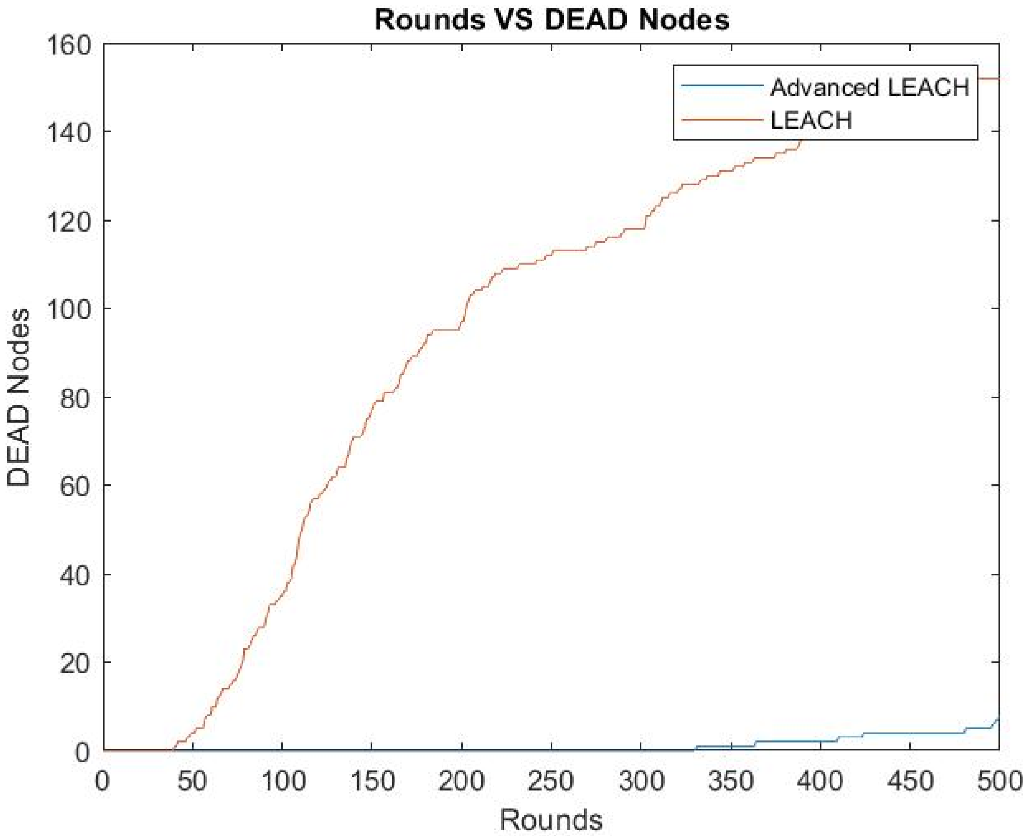

Fig. 3.4 Number of Dead nodes comparison graph.

From the Above plots, it is clearly visible that Number of dead nodes in Advanced LEACH has drastically decreased compared to LEACH Protocol. In Advanced LEACH protocol there are no dead nodes till round 325 where as in leach nodes started to become dead before round 50. Also, the total number of dead nodes at end in Advanced LEACH protocol are very very less compared to LEACH protocol.

Number of live nodes:

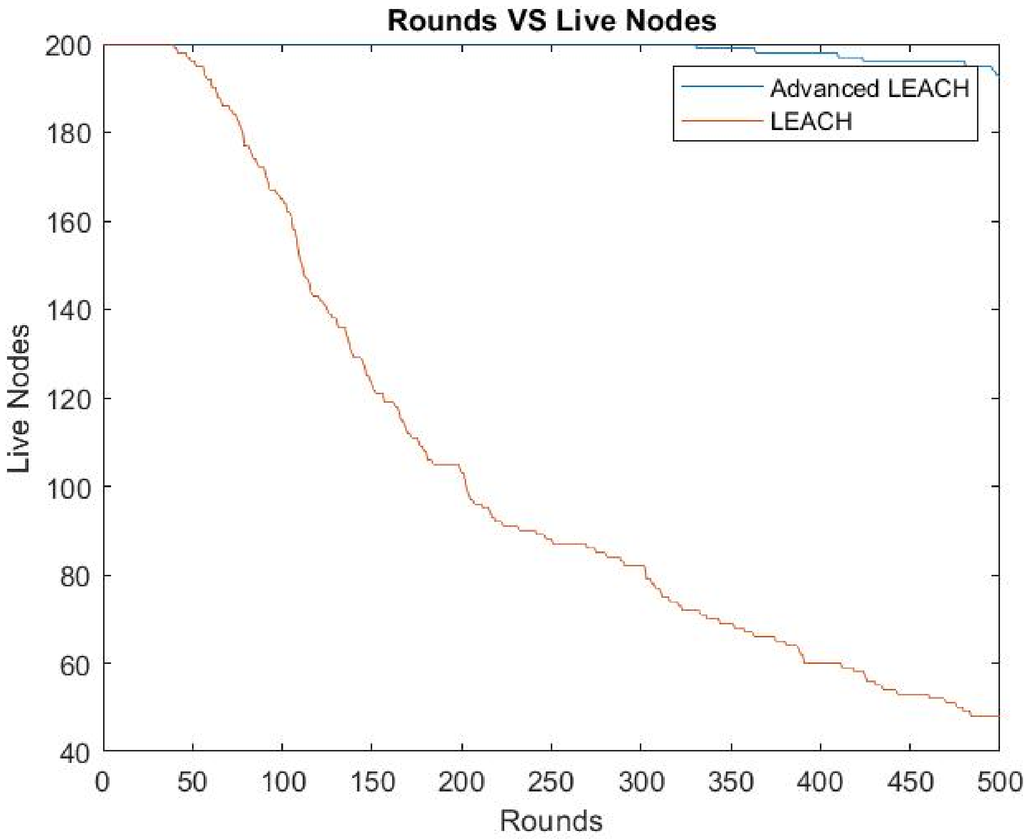

Fig. 3.5 Number of Live nodes comparison graph.

From the Above plots, it is clearly visible that number of live nodes has increased intensively in Advanced LEACH protocol compared to LEACH protocol, which means that the lifetime of the Network has increased.

CHAPTER 4
CONCLUSION AND FUTURE PLANS

4.1 CONCLUSION

In this project, an improved version of LEACH Protocol (Advanced LEACH) has been developed in order to improve efficiency and lifetime of the network. Both the protocols have been compared in terms of Average Energy and dead nodes, and found that Advanced LEACH protocol outstrips the LEACH Protocol. MATLAB has been used for analysing the performance of the protocols. Based on the simulation results, it is proved that Advanced LEACH performs better than LEACH in terms of energy efficiency and lifetime of the network. The simulation results validate that our proposed approach could extend the network lifetime for WSNs applications.

4.2 FUTURE PLANS

Further, parameters like lower threshold and higher threshold can be introduced and conditions in selecting CH can be introduced such as centralising the CH which results in energy efficiency of the network.

CHAPTER 5

REFERENCES

1. Nandi, A., Sonowal, B., Rabha, D., & Vaibhav, A. (2019, April). Centered Sink LEACH Protocol for Enhanced Performance of Wireless Sensor Network. In 2019 International Conference on Automation, Computational and Technology Management (ICACTM) (pp. 436-440). IEEE.

2.Kumar, K., & Khera, S. (2019, February). Optimization of Transceiver Energy with LEACH Protocol for Wireless Sensor Ad-hoc Networks. In 2019 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT) (pp. 1-4). IEEE.

3. Alnawafa, E., & Marghescu, I. (2017, April). IMHT: Improved MHT-LEACH protocol for wireless sensor networks. In 2017 8th international conference on information and communication systems (ICICS) (pp. 246-251). IEEE.

4.N. Mazumder, H. Om, Distribute fuzzy approach to unequal clustering and routing algorithm for wireless sensor networks. International Journal of Communication Systems 31(12) (2018).

5. O. Younis, S. Fahmy, Heed: a hybrid energy-efficient distributed clustering approach for ad hoc sensor networks. IEEE Trans On Mobile Computing 3(4), 660–669 (2004)

6. X.X. Ding, M. Ling, Z.J. Wang, et al., DK-LEACH: an optimized cluster structure routing method based on LEACH in wireless sensor networks. Wireless Personal Communications 96(4), 6369–6379 (2017).

7. M. Ye, C. Li, G. Chen, et al., EECS: an energy efficient clustering scheme in wireless sensor networks. Journal of Frontiers of Computer Science & Technology 3(2-3), 99–119 (2007).

8. M. Bsoul, Al-Khasawneh, et al., An energy-efficient threshold-based clustering protocol for wireless sensor networks. Wireless Personal Communications 70(1), 99–112 (2013).

9. T. Samant, P. Mukherjee, et al., TEEN-V: a solution for intra-cluster cooperative communication in wireless sensor networks. International Conference on I-SMAC 2017, 209– 213 (2017).

10. S. AS, O. Ridha, Reliable and energy-efficient multi-hop LEACH-based clustering protocol for wireless sensor networks. Sustainable Computing-Informatics and Systems 20, 1– 13 (2018).