PROJECT REPORT ON

Energy Efficient Routing Protocols for Heterogeneous WSNs

Submitted in partial fulfilment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY

Submitted by

123003036 – BHAVANI OMKARIN M.J – CSE

Under the Guidance of

Prof. Sasikala Devi. N

School of Computing
SASTRA DEEMED TO BE UNIVERSITY
(A University established under section 3 of the UGC Act, 1956)
Tirumalaisamudram
Thanjavur - 613401
December (2021)

SHANMUGHA ARTS, SCIENCE, TECHNOLOGY & RESEARCH ACADEMY
(SASTRA DEEMED TO BE UNIVERSITY**)**
(A University Established under section 3 of the UGC Act, 1956)
TIRUMALAISAMUDRAM, THANJAVUR – 613401

BONAFIDE CERTIFICATE

Certified that this project work entitled **“Energy Efficient Routing Protocols for heterogeneous WSNs”** submitted to the Shanmugha Arts, Science, Technology & Research Academy (SASTRA Deemed to be University), Tirumalaisamudram - 613401 by *Bhavani Omkarini M.J (123003036)*, CSE in partial fulfillment of the requirements for the award of the degree of **BACHELOR OF TECHNOLOGY** in their respective programme. This work is an original and independent work carried out under my guidance, during the period August 2021 - December 2021.

Prof. Sasikala Devi. N

ASSOCIATE DEAN
SCHOOL OF COMPUTING

Submitted for Project Viva Voce held on____________________________

Examiner – I

Examiner – II

TABLE OF CONTENTS

ACKNOWLEDGEMENTS

First of all, I would like to thank God Almighty for his endless blessings.

I would like to express my sincere gratitude to **Dr S. Vaidyasubramaniam, Vice-Chancellor** for his encouragement during the span of my academic life at SASTRA Deemed University.

I would forever remain grateful and I would like to thank **Dr A.Umamakeswri, Dean, School of Computing and R. Chandramouli, Registrar** for their overwhelming support provided during my course span in SASTRA Deemed University.

I am extremely grateful to **Dr. Shankar Sriram, Associate Dean, School of Computing** for her constant support, motivation and academic help extended for the past three years of my life in School of Computing.

I would specially thank and express my gratitude to N. **Sasikala Devi, Associate Professor, School of Computing** for providing me an opportunity to do this project and for her guidance and support to successfully complete the project.

I also thank all the Teaching and Non-teaching faculty, and all other people who have directly or indirectly help me through their support, encouragement and all other assistance extended for completion of my project and for successful completion of all courses during my academic life at SASTRA Deemed University.

Finally, I thank my parents and all others who help me acquire this interest in project and aided me in completing it within the deadline without much struggle.

ABSTRACT

WSNs (Wireless Sensor Networks) are majorly used to monitor the system, physical or environmental conditions and they analyze the real-world details. A WSN is a collection of nodes where each and every node is connected to the sensors which have a radio transceiver, internal antenna, microcontroller, electronic circuit, etc... as its parts. Those sensor nodes in the WSNs are used to manage the particular environmental condition. The Base Station present in the WSNs is connected to the internet which helps it to share the data. In WSN, the nodes send the data from sink to destination but the efficiency of the entire network depends on various parameters such as energy consumption, throughput, packet-delivery-ratio, delay, network area, protocol used etc... based on the requirement. An appropriate protocol must be chosen based on the requirement to run the network efficiently. Here we analyze the performance measure of three important WSN protocols and we test the node density effects on dead nodes, alive nodes and packet to base station node density effect.

KEY WORDS

- Wireless Sensor Network
- SEP Protocol
- Z-SEP Protocol
- LEACH Protocol
- Dead Nodes
- Alive Nodes
- Throughput
- Overhead
- End to End Delay
- Packet
- Energy Consumption
- Network Area
- Node Density
- Base Station

CHAPTER 1

LIST OF NOTATIONS

Notation	Description
Eo	Initial Energy
p	Probability of a node
ETX	Expected Transmission Count of a link
EDA	Energy for Data Aggregation
ERX	Transmitting and Receiving Energy
Emp	Amplification Energy for long distance
Efs	Amplification Energy for short distance
m	Percentage of advanced nodes
a	The energy percentage of advanced nodes that is greater than the normal nodes
rmax	Maximum number of rounds
CHs	Cluster Heads
BS	Base Station
pnrm	Election Probability of Normal nodes
padv	Election Probability of Advanced nodes
dead	Number of dead nodes
dead_a	Number of advanced dead nodes
dead_n	Number of normal dead nodes

xm	X-maximum
ym	Y-maximum
n	Number of nodes in the field
(x_d, y_d)	(x, y) co-ordinates of destination node.

LIST OF FIGURES

ABBREIVIATION

WSN	Wireless Sensor Network
SEP	Stable Election Protocol
Z-SEP	Zonal Stable Election Protocol
LEACH	Low-energy adaptive clustering hierarchy protocol

CHAPTER 2

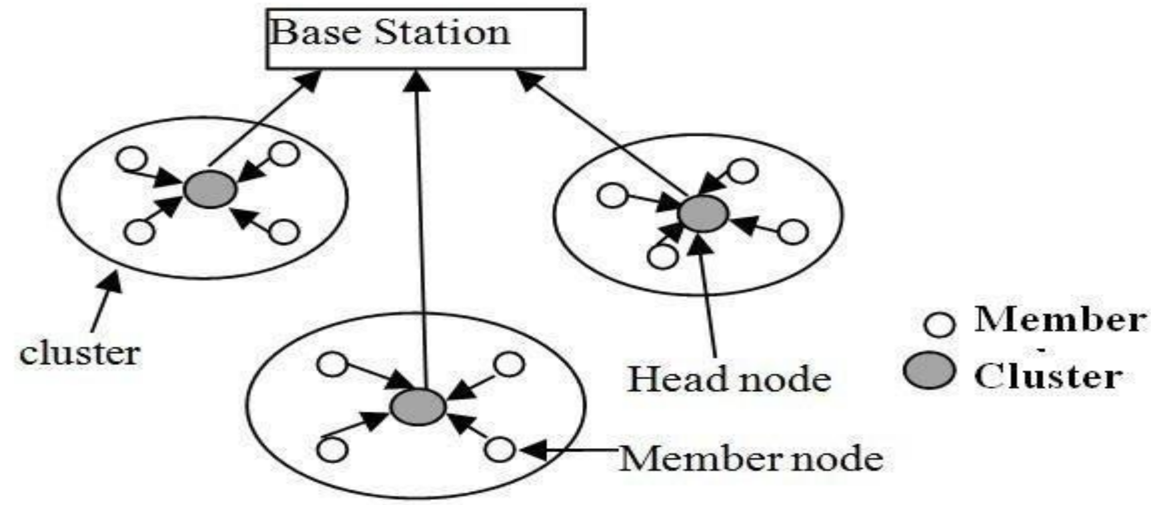
INTRODUCTION

WSNs (Wireless Sensor Networks) are majorly used to monitor the system, physical or environmental conditions and they analyze the real-world details. A WSN is a collection of nodes where each and every node is connected to the sensors which have a radio transceiver, internal antenna, microcontroller, electronic circuit, etc... as its parts. Those sensor nodes in the WSNs are used to manage the particular environmental condition. The Base Station present in the WSNs is connected to the internet which helps it to share the data. In WSN, the nodes send the data from sink to destination but the efficiency of the entire network depends on various parameters such as energy consumption, throughput, packet-delivery-ratio, delay, network area, protocol used etc... based on the requirement. An appropriate protocol must be chosen based on the requirement to run the network efficiently. Here we analyze the performance measure of three important WSN protocols and we test the node density effects on dead nodes, alive nodes and packet to base station node density effect.

WSNs are used in various fields such as Area monitoring, Healthcare monitoring, Habitat monitoring, Industrial monitoring, Threat detection, Environmental and Earth sensing, and many more fields. The huge number of sensor nodes in WSNs monitor all these fields and conditions like vibration, pressure, temperature, sound, motion, etc… Because of their low cost and variety of applications, the development of WSN makes an evaluation of wireless communications, electronics, and technological evolution. The protocols of WSN must be accurate, and efficient so that the network will work efficiently.

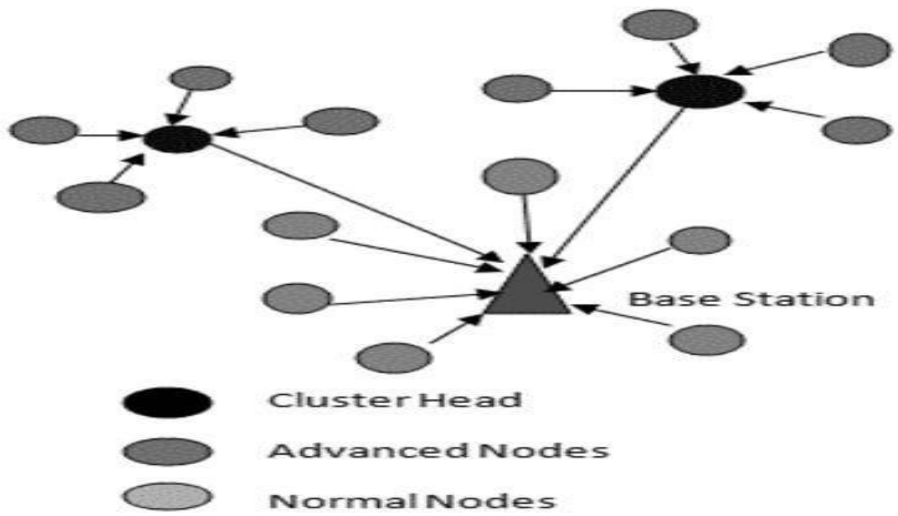
As we have seen that the WSN protocols should be accurate, efficient, and also self-organizing in order to work the network efficiently. The most efficient protocols of this network are hierarchical routing protocols. In these, we have homogeneous protocols and heterogeneous protocols like Leach (Low Energy adaptive clustering hierarchy) protocol, SEP (Stable Election Protocol), Z-SEP (Zonal Stable Election Protocol). Leach doesn’t work efficiently in heterogeneous environments and then SEP was proposed. Later Z-SEP was proposed which is more efficient than LEACH and SEP.

LEACH stands for Low-energy adaptive clustering hierarchy protocol. It is one of the MAC protocols which is based on TDMA in WSNs. It is a hierarchical protocol that works efficiently in a homogenous environment. It is not suited for heterogeneous environments. It is a clustering protocol that uses cluster heads to manage and distribute energy load between sensors and networks. Leach works on a distributed algorithm that is capable to select a cluster by itself by choosing a random number. It operates in two phases, the setup phase, and the state phase. In the setup phase, the network is organized into clusters, and the state phase data is aggregated, compressed, and transmitted to the destination. To determine whether a particular node will become a cluster head in a particular round, Leach uses a stochastic algorithm.



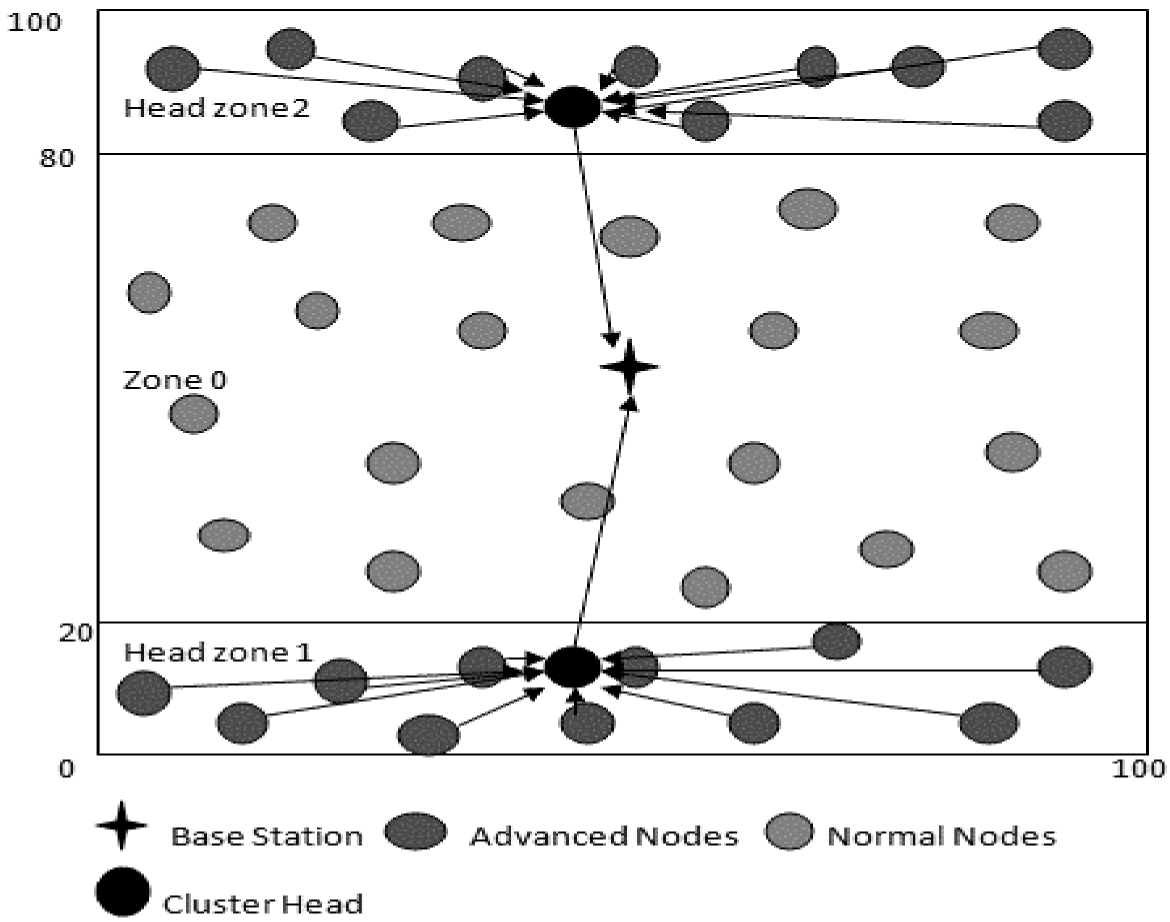
The diagram above represents the leach protocol which has Base Station also called as Sink, The cluster of head nodes and member nodes.

SEP which stands for Stable Election Protocol is a heterogeneous protocol. It is based on weighted election probabilities for each node which become a cluster head according to their respective energy. Here in this protocol, the normal and the advanced nodes are deployed randomly and the network area is divided into regions, the far nodes away from the base station which requires more energy to transmit the data are termed as advanced nodes and the nodes which are near to the base station are termed as normal nodes. The reason for this division is that, if the nodes are deployed far away from the base station, it consumes more energy which affects the throughput and results in less stability. Hence in this protocol, we divide the network area into zones.



The above diagram represents SEP protocol with its Base Station (Sink), cluster head, Normal nodes and Advanced nodes.

Z-SEP which stands for Zonal Stable Election Protocol. It is also suitable for the heterogeneous environment. Z-SEP has two-level heterogeneity. Here, in this protocol, we divide the network area or field into three zones namely Zone 0, Head Zone 1, and Head Zone 2. In Zone 0, Normal nodes are deployed randomly, In Zone 1 both normal nodes and half of the advanced nodes are deployed and another half of the advanced nodes are deployed in Zone2. Z-SEP uses two different protocols for the data transmission to the base station unlike LEACH and SEP and hence it's a hybrid protocol. The two different techniques it uses are direct transmission as SEP and transmission using cluster as a LEACH.



CHAPTER 3

RELATED WORKS

In April 2014, Upasana Sharma and Sunil Tiwari proposed and wrote "Performance Analysis of SEP and LEACH for Heterogeneous Wireless Sensor Networks" in which they have explained the performance of SEP and LEACH protocols in detail and also talked about their efficiency, throughput, delay, and the better protocol that which is suitable for the particular environment.

After the proposal of LEACH and SEP, then DEEC which stands for Distributed Energy Efficient Clustering was designed for multilevel heterogeneous networks in which cluster heads are elected by a probability governed by the ratio of the average energy of the network and the nodes' residual energy.

Later on, they have proposed numerous protocols based on LEACH, SEEP, DEEC as per the requirements and the disadvantages we have with the existing ones.
RB Patel and Dilip Kumar proposed EECDA which stands for Energy Efficient Clustering and Data Aggregation Protocol. This combines the ideas of energy efficient cluster based routing and data aggregation to achieve a better performance in terms of lifetime and stability. This protocol includes a novel cluster head election technique and a path would be selected with maximum sum of energy residues for data transmission instead of the path with minimum energy consumption.

In 2015, MSEP-E, Enhanced Stable Election Protocol with multihop communication was proposed by Raju Pal which does multihop communication between cluster heads and sensor nodes along with the base station which is called a sink. It utilizes less power and attains a greater network lifetime compared to other network protocols.

In April 2017, the conference was held about E-SEP (Enhanced Stable Election Protocol) by N. Uniyal which is made is made up of a large number of sensor nodes with limited energy that cooperate to perform a sensing task. And it was added to the IEEE Xplore in April 2018.

Then in 2017, an Enhanced Stable Election Protocol was proposed by R Pal based on fuzzy logic called FSEP-E in order to prolong the lifespan of heterogenous WSNs, HWSNs in short. While designing this fuzzy logic, we focus on three different things which are node density, heterogeneity threshold, and proximity to the base station. In this, the stability period is increased by 16% and the overall lifetime of the network is increased by 30%.

In 2019, L Zhao proposed an improved energy-efficient routing protocol that performed better than SEP and then modified the SEP in which he introduced SEA (Special Energy Advanced) node which resulted in the reduced power consumption and longer stable period and network life to the data packets.

CHAPTER 4

PROPOSED FRAMEWORK

In the proposed algorithm and frame-work we analyze the performance measure of LEECH, SEP, and Z-SEP protocols. We test the node density, dead nodes, alive nodes, and packet to a base station or sink of these three protocols.

FOR SEP

We consider the initial energy as Eo and the initial optimal probability of the node as p which is equal to the value of 0.1. We consider the data aggregation energy as 5*0.000000001 and the percentage of advance nodes are considered as m which is assigned the value of 0.1. We also consider that alpha (a=2) times the energy of advanced nodes is greater than the normal nodes. We consider the maximum number of rounds (rmax) as 9000.

• Then we create the random sensor network in which we calculate the random election of normal nodes and random election of advanced nodes.

• Then we create cluster heads and find the election probability of normal nodes and advanced nodes separately.

• Then we initialize the number of dead nodes (dead=0) and number of dead advanced nodes (dead_a=0) and number of dead normal nodes (dead_n=0).

• Then we check the presence of dead nodes in the network. Similarly, we do it for alive nodes and packets to the base station.

• Later we perform the election of cluster heads for advanced nodes

• Then we calculate the energy dissipated by the cluster head along with the total energy that is dissipated in the network.

FOR Z-SEP

We consider the initial energy as Eo and the initial optimal probability of the node as p which is equal to the value of 0.1. We consider the data aggregation energy as 5*0.000000001 and the percentage of advance nodes are considered as m which is assigned the value of 0.1. We also consider that alpha (a=2) times the energy of advanced nodes is greater than the normal nodes. We consider the maximum number of rounds (rmax) as 9000.

• We initialize the sink, cluster nodes

• Then we transmit the counter to the sink (base station) and cluster heads.

• We calculate the election probability for the advanced nodes.

•Then we initialize the number of dead nodes (dead2=0), the number of dead advanced nodes(dead_a2=0), and the number of dead normal nodes (dead_n2=0).

• Then we check for the presence of dead nodes in the network.

• Later, we calculate the energy dissipated by the cluster member for transmission of the packet as well as receiving the packet.

FOR LEACH

We consider the initial energy as Eo and the initial optimal probability of the node as p which is equal to the value of 0.1. We consider the data aggregation energy as 5*0.000000001 and the percentage of advance nodes are considered as m which is assigned the value of 0.1. We also consider that alpha (a=2) times the energy of advanced nodes is greater than the normal nodes. We consider the maximum number of rounds (rmax) as 9000.

• Firstly, we initialize the nodes, sink (base station), and cluster heads.

• Then we perform random election of normal nodes

• Then we initialize the number of dead nodes (dead3=0), the number of advanced dead nodes (dead_a3=0), and the number of normal dead nodes (dead_n3=0).

• Then we check for the presence of the dead nodes in the network.

• Then we do the election of cluster heads along with calculating dissipated energy.

• Then we calculate the election of associated cluster heads for normal nodes.

• Finally, we calculate the dissipated energy of the entire network.

<u>Dead nodes</u>: The nodes which are out of energy are defined to be as dead nodes.

<u>Alive nodes</u>: The nodes which are not dead are defined as Alive nodes.

Then we analyze the performance of these three protocols based on the dead nodes, alive nodes and packet to base station and compare them by plotting the graphs.

CHAPTER 5

SOURCE CODE

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
%Comparison of LEACH, SEP and Z-SEP routing protocols for WSNs
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%

clear;


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% PARAMETERS
%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Here we are considering the Field Dimensions - x and y maximum (in meters)
xm=100;
ym=100;

% We are considering x and y as the  Coordinates of the Sink
sink.x=50;
sink.y=50;

% Here the variable n represents Number of Nodes in the field
n=100;

% This code is to check Optimal Election Probability of a node
% The following code is to become cluster head
p=0.1;

% Here is the code for Energy Model (all values in Joules)
% Here Eo represents the Initial Energy
Eo=0.5;
% These are the same values Eelec=Etx=Erx
ETX=50*0.000000001;
ERX=50*0.000000001;
% Here are the types of Transmit Amplifiers
Efs=10*0.000000000001;
Emp=0.0013*0.000000000001;
% Given code is for Data Aggregation Energy
EDA=5*0.000000001;

yd1=33;

% These are the Values for Hetereogeneity
% m is the Percentage of nodes than are advanced
m=0.1;
%\alpha times advance nodes have energy greater than normal nodes
a=2;


%rmax is the maximum number of rounds
rmax=9000;
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% This code is for Computation of do
do=sqrt(Efs/Emp);
Et=0;
% This code is for Creation of the random Sensor Network
%figure(1);
for i=1:1:n
   S0(i).xd=rand(1,1)*xm;
   XR(i)=S0(i).xd;
   S0(i).yd=rand(1,1)*ym;
   YR(i)=S0(i).yd;
   S0(i).G=0;
   % Initially there are no cluster heads present, we have only nodes
   S0(i).type='N';

   temp_rnd0=i;
   % This is for Random Election of Normal Nodes
   if (temp_rnd0>=m*n+1)
      S0(i).E=Eo;
      E(i)=S0(i).E;
      S0(i).ENERGY=0;
      %  This is for plotting (S0(i).xd,S0(i).yd,'o');
      %   hold on;
   end
   % This is for Random Election of Advanced Nodes
   if (temp_rnd0<m*n+1)
      S0(i).E=Eo*(1+a);
      E(i)=S0(i).E;
      S0(i).ENERGY=1;
      % This is for plotting (S0(i).xd,S0(i).yd,'+');
      %  hold on;
   end
   Et=Et+S0(i).E;
end

S0(n+1).xd=sink.x;
S0(n+1).yd=sink.y;
% This is for plotting (S0(n+1).xd,S0(n+1).yd,'x');


% For First Iteration
%figure(1);

%counter for CHs
countCHs=0;
%counter for CHs per round
rcountCHs=0;
cluster=1;

countCHs;
rcountCHs=rcountCHs+countCHs;

flag_first_dead=0;
allive=n;

packets_TO_BS=0;
```

```
packets_TO_CH=0;

for r=0:1:rmax
  r

  % This code is for Election Probability for Normal Nodes
  pnrm=( p/ (1+a*m) );
  % This code is for Election Probability for Advanced Nodes
  padv= ( p*(1+a)/(1+a*m) );

  % This code is for Operation for heterogeneous epoch
  if(mod(r, round(1/pnrm) )==0)
    for i=1:1:n
      S0(i).G=0;
      S0(i).cl=0;
    end
  end

  % Here is the code for Operations for sub-epochs
  if(mod(r, round(1/padv) )==0)
    for i=1:1:n
      if(S0(i).ENERGY==1)
        S0(i).G=0;
        S0(i).cl=0;
      end
    end
  end


  %hold off;

  % dead represents the Number of dead nodes
  dead=0;
  % dead_a represents Number of dead Advanced Nodes
  dead_a=0;
  % dead_n represents Number of dead Normal Nodes
  dead_n=0;

  %figure(1);

  for i=1:1:n
    % The following code is for checking if there is a dead node
    if (S0(i).E<=0)
      %plot(S0(i).xd,S0(i).yd,'red .');

      dead=dead+1;

      if(S0(i).ENERGY==1)
        dead_a=dead_a+1;
      end
      if(S0(i).ENERGY==0)
        dead_n=dead_n+1;
      end
      %hold on;
    end
    if S0(i).E>0
      S0(i).type='N';
```

```
        if (S0(i).ENERGY==0)
            % This is for plotting (S0(i).xd,S0(i).yd,'o');
        end
        if (S0(i).ENERGY==1)
            % This is for plotting (S0(i).xd,S0(i).yd,'+');
        end
        %hold on;
    end
    STATISTICS.DEAD(r+1)=dead;
    STATISTICS.ALLIVE(r+1)=allive-dead;

end
% This is for plotting (S0(n+1).xd,S0(n+1).yd,'x');


% This code is for cnonsidering When the first node dies
if (dead==1)
    if(flag_first_dead==0)
        first_dead=r
        flag_first_dead=1;
    end
end

countCHs=0;
cluster=1;
for i=1:1:n
    if(S0(i).E>0)
        temp_rand=rand;
        if ( (S0(i).G)<=0)

            % This code is for Election of Cluster Heads for normal nodes
            if( ( S0(i).ENERGY==0 && ( temp_rand <= ( pnrm / ( 1 - pnrm * mod(r,round(1/pnrm)) )) ) )  )

                countCHs=countCHs+1;
                packets_TO_BS=packets_TO_BS+1;
                PACKETS_TO_BS(r+1)=packets_TO_BS;

                S0(i).type='C';
                S0(i).G=100;
                C(cluster).xd=S0(i).xd;
                C(cluster).yd=S0(i).yd;
                % This is for plotting (S0(i).xd,S0(i).yd,'k*');

                distance=sqrt( (S0(i).xd-(S0(n+1).xd) )^2 + (S0(i).yd-(S0(n+1).yd) )^2 );
                C(cluster).distance=distance;
                C(cluster).id=i;
                X(cluster)=S0(i).xd;
                Y(cluster)=S0(i).yd;
                cluster=cluster+1;

                % This code is for calculating the Energy dissipated
                distance;
                if (distance>do)
                    S0(i).E=S0(i).E- ( (ETX+EDA)*(4000) + Emp*4000*( distance*distance*distance*distance ));
                end
                if (distance<=do)
                    S0(i).E=S0(i).E- ( (ETX+EDA)*(4000)  + Efs*4000*( distance * distance ));
```

```
            end
        end



        % This code is for Election of Cluster Heads for Advanced nodes
        if( ( S0(i).ENERGY==1 && ( temp_rand <= ( padv / ( 1 - padv * mod(r,round(1/padv)) )) ) )  )

            countCHs=countCHs+1;
            packets_TO_BS=packets_TO_BS+1;
            % This code is for PACKETS_TO_BS(r+1)=packets_TO_BS;

            S0(i).type='C';
            S0(i).G=100;
            C(cluster).xd=S0(i).xd;
            C(cluster).yd=S0(i).yd;
            % This code is for plotting (S0(i).xd,S0(i).yd,'k*');

            distance=sqrt( (S0(i).xd-(S0(n+1).xd) )^2 + (S0(i).yd-(S0(n+1).yd) )^2 );
            C(cluster).distance=distance;
            C(cluster).id=i;
            X(cluster)=S0(i).xd;
            Y(cluster)=S0(i).yd;
            cluster=cluster+1;

            % This is for calculating the Dissipation Energy
            distance;
            if (distance>do)
                S0(i).E=S0(i).E- ( (ETX+EDA)*(4000) + Emp*4000*( distance*distance*distance*distance ));
            end
            if (distance<=do)
                S0(i).E=S0(i).E- ( (ETX+EDA)*(4000)  + Efs*4000*( distance * distance ));
            end
        end

    end
  end
end

packets_TO_BS=packets_TO_BS+1;
STATISTICS.packets_TO_BS(r+1)=packets_TO_BS;



% This code is for Election of Associated Cluster Head for Normal Nodes
for i=1:1:n
   if ( S0(i).type=='N' && S0(i).E>0 )
      if(cluster-1>=1)
         min_dis=sqrt( (S0(i).xd-S0(n+1).xd)^2 + (S0(i).yd-S0(n+1).yd)^2 );
         min_dis_cluster=1;
         for c=1:1:cluster-1
            temp=min(min_dis,sqrt( (S0(i).xd-C(c).xd)^2 + (S0(i).yd-C(c).yd)^2 ) );
            if ( temp<min_dis )
               min_dis=temp;
               min_dis_cluster=c;
            end
         end
```

```
            % This code is for Energy dissipated by associated Cluster Head
            min_dis;
            if (min_dis>do)
               S0(i).E=S0(i).E- ( ETX*(4000) + Emp*4000*( min_dis * min_dis * min_dis * min_dis));
            end
            if (min_dis<=do)
               S0(i).E=S0(i).E- ( ETX*(4000) + Efs*4000*( min_dis * min_dis));
            end
            % Here is Energy dissipated calculation
            if(min_dis>0)
               S0(C(min_dis_cluster).id).E = S0(C(min_dis_cluster).id).E- ( (ERX + EDA)*4000 );
               PACKETS_TO_CH(r+1)=n-dead-cluster+1;
            end

            S0(i).min_dis=min_dis;
            S0(i).min_dis_cluster=min_dis_cluster;

         end
      end
   end
   %hold on;

   countCHs;
   rcountCHs=rcountCHs+countCHs;


end


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%             Z sep                          %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%figure(1);
for i=1:1:n
   if (i<=5)

      S1(i).xd=rand(1,1)*xm;
      S1(i).yd=rand(1,1)*yd1;
      S1(i).E=Eo*(1+a);
      S1(i).ENERGY=1;
      S1(i).type='A';
      % Here we are plotting (S1(i).xd,S1(i).yd,'og');
      %hold on
   end
   if (i>10 &&  i<=100)
      S1(i).xd=rand(1,1)*xm;
      S1(i).yd=33+rand(1,1)*yd1;
      S1(i).E=Eo;
      S1(i).ENERGY=0;
      S1(i).type='N';
      % Here we are plotting (S1(i).xd,S1(i).yd,'ob');
      %hold on
   end
   if(i<=10 && i>5)
      S1(i).xd=rand(1,1)*xm;
```

```
        S1(i).yd=66+rand(1,1)*yd1;
        S1(i).E=Eo*(1+a);
        S1(i).ENERGY=1;
        S1(i).type='A';
        % Here we are plotting (S1(i).xd,S1(i).yd,'og');
        %hold on
    end
end
S1(n+1).xd=sink.x;
S1(n+1).yd=sink.y;
%% Here we are plotting (S1(n+1).xd,S1(n+1).yd,'green x');

%figure(1);

flag_first_dead2=0;
allive2=n;

% Here we are transmitting counter to Base Station and also to the Cluster Heads
packets_TO_BS12=0;
packets_TO_BS22=0;
packets_TO_BS2=0;
packets_TO_CH2=0;

for r=0:1:rmax
    r
    % This code is for calculating Election Probability for Advanced Nodes
    padv= ( p*(1+a)/(1+(a*m)) );

    % This code is for Operations for sub-epochs
    if(mod(r, round(1/padv) )==0)
        for i=1:1:n
            if(S1(i).ENERGY==1)
                S1(i).G=0;
                %S1(i).cl=0;
            end

        end
    end

    %hold off;

    %figure(1);

    dead2=0;

    % Here the dead_a2 represents the Number of dead Advanced Nodes
    dead_a2=0;
    % Here the dead_n2 represents the Number of dead Normal Nodes
    dead_n2=0;

    for i=1:1:n
        % This is the code for checking if there is a dead node
        if (S1(i).E<=0)
            %% Here we are plotting (S1(i).xd,S1(i).yd,'red .');

            dead2=dead2+1;
```

```
        if(S1(i).ENERGY==1)
            dead_a2=dead_a2+1;
        end
        if(S1(i).ENERGY==0)
            dead_n2=dead_n2+1;
        end
        %hold on;
    end
    if (S1(i).E>0)
        S1(i).type='N';
        if (S1(i).ENERGY==0)
            %plot(S1(i).xd,S1(i).yd,'ob');
        end
        if (S1(i).ENERGY==1)
            %plot(S1(i).xd,S1(i).yd,'og');
        end
        %hold on;
    end
    %plot(S1(n+1).xd,S1(n+1).yd,'green x');

    STATISTICS.DEAD2(r+1)=dead2;
    STATISTICS.ALLIVE2(r+1)=allive2-dead2;

end

% Here we are considering the case When the first node dies
if (dead2==1)
    if(flag_first_dead2==0)
        first_dead2=r
        flag_first_dead2=1;
    end
end
for(i=1:1:n)
    if(S1(i).E>=0)
        if(S1(i).type=='N')
            distance=sqrt( (S1(i).xd-(S1(n+1).xd) )^2 + (S1(i).yd-(S1(n+1).yd) )^2 );
            if (distance>do)
                S1(i).E=S1(i).E- ( (ETX+EDA)*(4000) + Emp*4000*( distance*distance*distance*distance ));
            end
            if (distance<=do)
                S1(i).E=S1(i).E- ( (ETX+EDA)*(4000)  + Efs*4000*( distance * distance ));
            end
            packets_TO_BS12=packets_TO_BS12+1;
        end
    end

end



countCHs2=0;
cluster2=1;

for i=1:1:n
    if(S1(i).E>=0)
        if(S1(i).G<=0)
            if(S1(i).type=='A')
```

```
        temp_rand=rand;
        if( ( S1(i).ENERGY==1 && ( temp_rand <= ( padv / ( 1 - padv * mod(r,round(1/padv)) )) ) )  )

          countCHs2=countCHs2+1;
          packets_TO_BS22=packets_TO_BS22+1;


          S1(i).type='C';
          S1(i).G=(1/padv)-1;
          C(cluster2).xd=S1(i).xd;
          C(cluster2).yd=S1(i).yd;
          %plot(S2(i).xd,S1(i).yd,'k*');

          distance=sqrt( (S1(i).xd-(S1(n+1).xd) )^2 + (S1(i).yd-(S1(n+1).yd) )^2 );
          C(cluster2).distance=distance;
          C(cluster2).id=i;
          X(cluster2)=S1(i).xd;
          Y(cluster2)=S1(i).yd;
          cluster2=cluster2+1;

          % This is the code for calculating the Energy that is dissipated
          distance;
          if (distance>do)
            S1(i).E=S1(i).E- ( (ETX+EDA)*(4000) + Emp*4000*( distance*distance*distance*distance ));
          end
          if (distance<=do)
            S1(i).E=S1(i).E- ( (ETX+EDA)*(4000)  + Efs*4000*( distance * distance ));
          end

        end
      end

    end
    for i=1:1:n
      if ( S1(i).type=='A' && S1(i).E>0 )
        if(cluster2-1>=1)
          min_dis=inf;
          min_dis_clustera=1;

          for c=1:1:cluster2-1
            temp=min(min_dis,sqrt( (S1(i).xd-C(c).xd)^2 + (S1(i).yd-C(c).yd)^2 ) );

            if ( temp<min_dis )
              min_dis=temp;
              min_dis_cluster2=c;
            end

          end

          % This is the code for calculating the Energy dissipated by  Cluster menmber for transmission
of packet
          %  min_dis;
          if (min_dis>do)
            S1(i).E=S1(i).E- ( ETX*(4000) + Emp*4000*( min_dis * min_dis * min_dis * min_dis));
          end
          if (min_dis<=do)
            S1(i).E=S1(i).E- ( ETX*(4000) + Efs*4000*( min_dis * min_dis));
```

```
            end
            % This is the code for calculating the Energy dissipated by clustre head in receving
            if(min_dis>0)
              S1(C(min_dis_cluster2).id).E = S1(C(min_dis_cluster2).id).E- ( (ERX + EDA)*4000 );
              PACKETS_TO_CH(r+1)=n-dead-cluster2+1;
            end

            S1(i).min_dis=min_dis;
            S1(i).min_dis_cluster2=min_dis_cluster2;

          end
        end
      end
    end
    CLUSTERHS(r+1)=cluster2+1;

    STATISTICS.CLUSTERHEADS2(r+1)=cluster2+1;


  end


  packets_TO_BS2=packets_TO_BS12+packets_TO_BS22;
  STATISTICS.PACKETS_TO_BS2(r+1)=packets_TO_BS2;

end





%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%   LEACH                         %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%



for i=1:1:n
  S2(i).xd=rand(1,1)*xm;
  XR(i)=S2(i).xd;
  S2(i).yd=rand(1,1)*ym;
  YR(i)=S2(i).yd;
  S2(i).G=0;
  % Here we consider there are no cluster heads and there are only nodes initially
  S2(i).type='N';

  temp_rnd0=i;
  % This is for Random Election of Normal Nodes
  if (temp_rnd0>=m*n+1)
    S2(i).E=Eo;
    S2(i).ENERGY=0;
    % Here we are plotting (S2(i).xd,S2(i).yd,'o');
    %hold on;
  end
  % This code is for Random Election of Advanced Nodes
  if (temp_rnd0<m*n+1)
    S2(i).E=Eo*(1+a)
```

```
        S2(i).ENERGY=1;
        % Here we are plotting (S2(i).xd,S2(i).yd,'+');
        %hold on;
    end
end

S2(n+1).xd=sink.x;
S2(n+1).yd=sink.y;
%plot(S2(n+1).xd,S2(n+1).yd,'x');


%First Iteration
%figure(1);

%counter for CHs
countCHs3=0;
%counter for CHs per round
rcountCHs3=0;
cluster3=1;

countCHs3;
rcountCHs3=rcountCHs3+countCHs3;
flag_first_dead3=0;
allive3=n;
packets_TO_BS3=0;
packets_TO_CH3=0;
for r=0:1:rmax
    r

    %Operation for epoch
    if(mod(r, round(1/p) )==0)
        for i=1:1:n
            S2(i).G=0;
            S2(i).cl=0;
        end
    end

    %hold off;

    % Here dead3 represents the Number of dead nodes
    dead3=0;
    % Here dead+a3 represents the Number of dead3 Advanced Nodes
    dead_a3=0;
    % Here dead_n3 represents the Number of dead Normal Nodes
    dead_n3=0;

    %counter for bit transmitted to Bases Station and to cluster3 Heads

    %counter for bit transmitted to Bases Station and to cluster3 Heads
    %per round


    %figure(1);

    for i=1:1:n
        % Here is the code for checking if there is a dead node
        if (S2(i).E<=0)
```

```
        % Here we are plotting (S3(i).xd,S2(i).yd,'red .');
        dead3=dead3+1;
        if(S2(i).ENERGY==1)
            dead_a3=dead_a3+1;
        end
        if(S2(i).ENERGY==0)
            dead_n3=dead_n3+1;
        end
        %hold on;
    end
    if S2(i).E>0
        S2(i).type='N';
        if (S2(i).ENERGY==0)
            % Here we are plotting (S2(i).xd,S2(i).yd,'o');
        end
        if (S2(i).ENERGY==1)
            % Here we are plotting (S2(i).xd,S2(i).yd,'+');
        end
        %hold on;
    end


    STATISTICS.DEAD3(r+1)=dead3;
    STATISTICS.ALLIVE3(r+1)=allive3-dead3;
end
%plot(S0(n+1).xd,S0(n+1).yd,'x');




% This is the code When the first node dies
if (dead3==1)
    if(flag_first_dead3==0)
        first_dead3=r
        flag_first_dead3=1;
    end
end

countCHs3=0;
cluster3=1;
for i=1:1:n
    if(S2(i).E>0)
        temp_rand=rand;
        if ( (S2(i).G)<=0)

            % This is the code for Election of cluster3 Heads
            if(temp_rand<= (p/(1-p*mod(r,round(1/p)))))
                countCHs3=countCHs3+1;
                packets_TO_BS3=packets_TO_BS3+2;
                % PACKETS_TO_BS3(r+1)=packets_TO_BS3;

                S2(i).type='C';
                S2(i).G=round(1/p)-1;
                C(cluster3).xd=S2(i).xd;
                C(cluster3).yd=S2(i).yd;
                %plot(S2(i).xd,S2(i).yd,'k*');
```

```
        distance=sqrt( (S2(i).xd-(S2(n+1).xd) )^2 + (S2(i).yd-(S2(n+1).yd) )^2 );
        C(cluster3).distance=distance;
        C(cluster3).id=i;
        X(cluster3)=S2(i).xd;
        Y(cluster3)=S2(i).yd;
        cluster3=cluster3+1;

        % This is the code for calculating the dissipating energy
        distance;
        if (distance>do)
          S2(i).E=S2(i).E- ( (ETX+EDA)*(4000) + Emp*4000*( distance*distance*distance*distance ));
        end
        if (distance<=do)
          S2(i).E=S2(i).E- ( (ETX+EDA)*(4000)  + Efs*4000*( distance * distance ));
        end
      end

    end
  end
end

STATISTICS.CLUSTERHEADS3(r+1)=cluster3-1;
CLUSTERHS(r+1)=cluster3-1;

%Election of Associated Cluster Head for Normal Nodes
for i=1:1:n
  if ( S2(i).type=='N' && S2(i).E>0 )
    if(cluster3-1>=1)
      min_dis=sqrt( (S2(i).xd-S2(n+1).xd)^2 + (S2(i).yd-S2(n+1).yd)^2 );
      min_dis_cluster=1;
      for c=1:1:cluster3-1
        temp=min(min_dis,sqrt( (S2(i).xd-C(c).xd)^2 + (S2(i).yd-C(c).yd)^2 ) );
        if ( temp<min_dis )
          min_dis=temp;
          min_dis_cluster=c;
        end
      end

      %Energy dissipated by associated Cluster Head
      min_dis;
      if (min_dis>do)
        S2(i).E=S2(i).E- ( ETX*(4000) + Emp*4000*( min_dis * min_dis * min_dis * min_dis));
      end
      if (min_dis<=do)
        S2(i).E=S2(i).E- ( ETX*(4000) + Efs*4000*( min_dis * min_dis));
      end
      %Energy dissipated
      if(min_dis>0)
        S2(C(min_dis_cluster).id).E = S2(C(min_dis_cluster).id).E- ( (ERX + EDA)*4000 );
        PACKETS_TO_CH3(r+1)=n-dead3-cluster3+1;
      end

      S2(i).min_dis=min_dis;
      S2(i).min_dis_cluster=min_dis_cluster;

    end
  end
```

```
    end
    %hold on;

    packets_TO_BS3=packets_TO_BS3+1;
    STATISTICS.packets_TO_BS3(r+1)=packets_TO_BS3;

    countCHs3;
    rcountCHs3=rcountCHs3+countCHs3;

end
figure(2);
r=0:rmax;
plot(r,STATISTICS.ALLIVE3(r+1),'-b',r,STATISTICS.ALLIVE(r+1),'-r',r,STATISTICS.ALLIVE2(r+1),'-g')
legend('LEACH   m=0.1,a=2','SEP   m=0.1,a=2','Z-SEP  m=0.1,a=2');
xlabel('Rounds');
ylabel('Alive Nodes');

figure(3);
plot(r,STATISTICS.DEAD3(r+1),'-b',r,STATISTICS.DEAD(r+1),'-r',r,STATISTICS.DEAD2(r+1),'-g')
legend('LEACH  m=0.1,a=2','SEP    m=0.1,a=2','Z-SEP  m=0.1,a=2');
xlabel('Rounds');
ylabel('Dead Nodes');

figure(4);
plot(r,STATISTICS.packets_TO_BS(r+1),'-r',r,STATISTICS.packets_TO_BS3(r+1),'-
b',r,STATISTICS.PACKETS_TO_BS2(r+1),'-g')
legend('LEACH   m=0.2,a=1','SEP    m=0.2,a=1','Z-SEP m=0.2,a=1');
xlabel('Rounds');
ylabel('Packets to Base Station');
```

CHAPTER 6

RESULTS

After executing the code, i.e.., after simulating the LEACH, SEP and Z-SEP protocols in MATLAB R2021b, we got the following results.

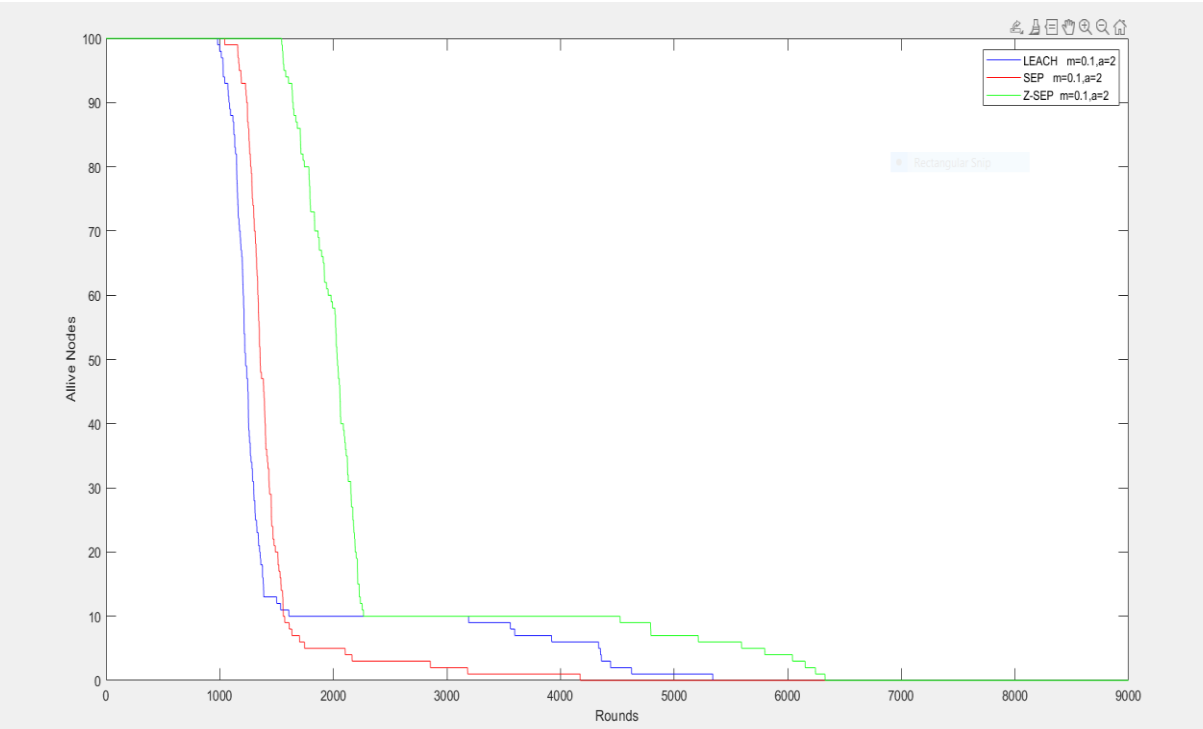


FIGURE 4

Figure 4: This represents the comparison of Alive nodes in LEACH, SEP, and ZSEP protocols. Here m represents the advance nodes. We know that the advanced nodes are present only in the SEP, unlike Z-SEP and LEACH, hence the value of m is 0 for LEACH and Z-SEP and 1 for ZEP protocol.

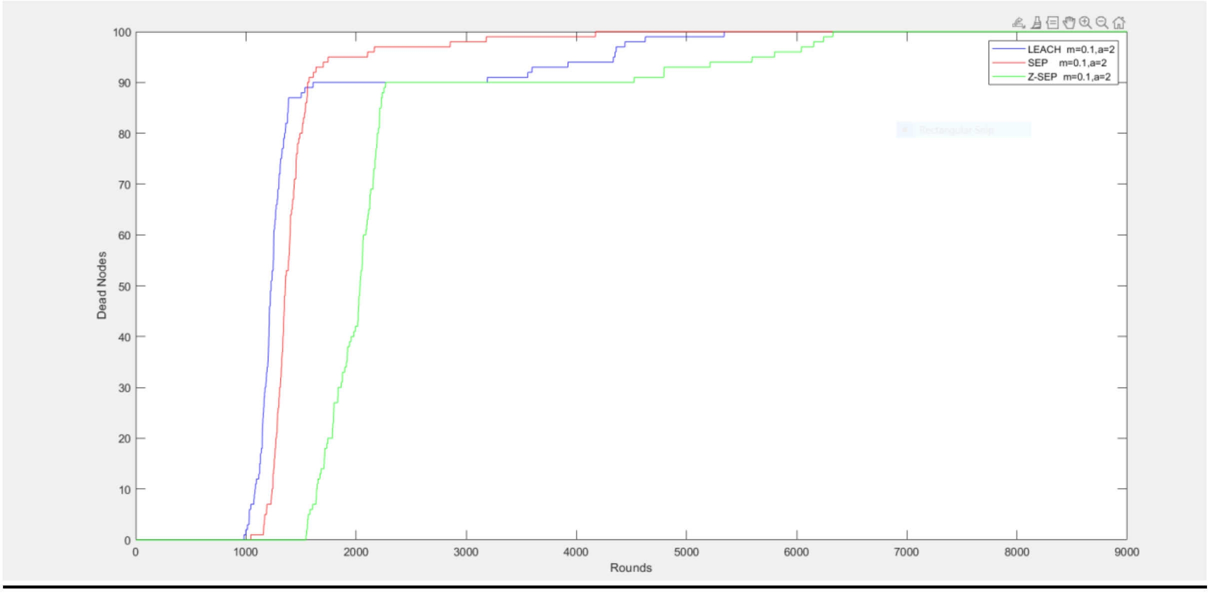


FIGURE 5

Figure 5: This represents the comparison of Dead nodes in LEACH, SEP, and ZSEP protocols. Here m represents the advance nodes. We know that the advanced nodes are present only in the SEP, unlike Z-SEP and LEACH, hence the value of m is 0 for LEACH and Z-SEP and 1 for ZEP protocol.

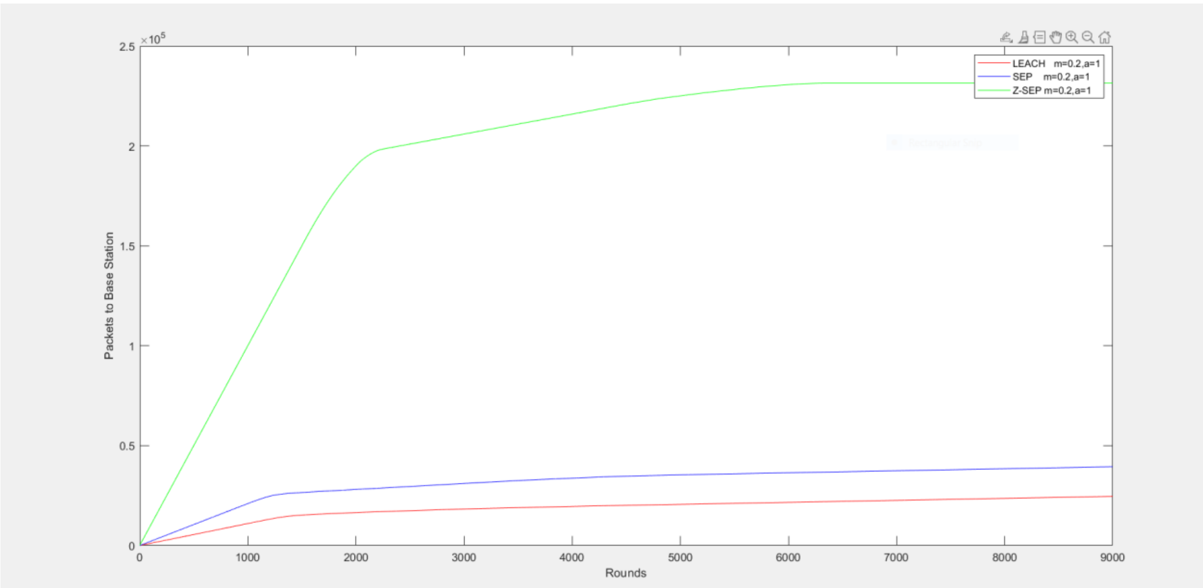


FIGURE 6

Figure 6: This represents the comparison Packet to base station in LEACH, SEP, and ZSEP protocols. Here m represents the advance nodes. We know that the advanced nodes are present only in the SEP, unlike Z-SEP and LEACH, hence the value of m is 0 for LEACH and Z-SEP and 1 for ZEP protocol.

CHAPTER 7

CONCLUSION

We have compared the three important WSN protocols namely LEACH, SEP, and Z-SEP in heterogeneous environments in this project.

Looking at the results, we can infer that the stability period of LEACH is low when compared to SEP and Z-SEP protocols.

In SEP and Z-SEP protocols, the lifetime of the network is increased due to the presence of advanced nodes which die slower when compared to the normal nodes.

Z-SEP takes less time when compared to SEP to send the packet. Packet to the base station is decreased by increasing the number of nodes for the three protocols but in the Z-SEP protocol, it is decreased a lot than others. It reaches the stable stage quickly.

CHAPTER 8

REFERENCES

Heinzelman, W. R., Chandrakasan, A., & Balakrishnan, H:"Energy-efficient communication protocol for wireless microsensor networks", In System Sciences, Proceedings of the 33rd Annual Hawaii International Conference on (pp. 10-pp). IEEE, January, 2000.

Smaragdakis, G., Matta, I., & Bestavros, A. SEP: A stable election protocol for clustered heterogeneous wireless sensor networks. Boston University Computer Science Department, 2004.

S. Faisal , N. Javaid, A. Javaid, M. A. Khan, S. H. Bouk and Z. A. Khan "Z-SEP: Zonal-Stable Election Protocol for Wireless Sensor Networks", Journal of Basic and Applied Scientific Research (JBASR), 2013.

Heinzelman, A. P. Chandrakasan, and H. Balakrishnan "An application-specific protocol architecture for wireless microsensor networks," IEEE Transactions on Wireless Communications, vol. 1, no. 4, pp. 660–670, October 2002.

G. Smaragdakis, I. Matta, A. Bestavros, "SEP: A Stable Election Protocol for clustered heterogeneous wireless sensor networks", in: Second International Workshop on Sensor and Actor Network Protocols and Applications (SANPA 2004), 2004.

Aderohunmu, F. A., & Deng, J. D. An Enhanced Stable Election Protocol (SEP) for Clustered Heterogeneous WSN (No. 2009/07), Discussion Paper Series, 2009.

Banoth Thirumala, B. Vijay Kumar and Aloth Sowjanya: "Zonal-Stable Election Protocol for Wireless Sensor", International Journal of Ethics in Engineering & Management Education, Volume 3, Issue 10, October 2016.

Upasana Sharma, Sunil Tiwari, "Performance Analysis of SEP and LEACH for Heterogeneous Wireless Sensor Networks", International Journal of Computer Trends and Technology (IJCTT) – volume 10 number 4 – Apr 2014.

Hamdy H. El-Sayed :" Data Aggregation Energy and Probability Effects on the Performance of EDEEC and MODLEACH Protocol in WSN", Appl. Math. Inf. Sci. 12, No. 1, 1-7 (2018).