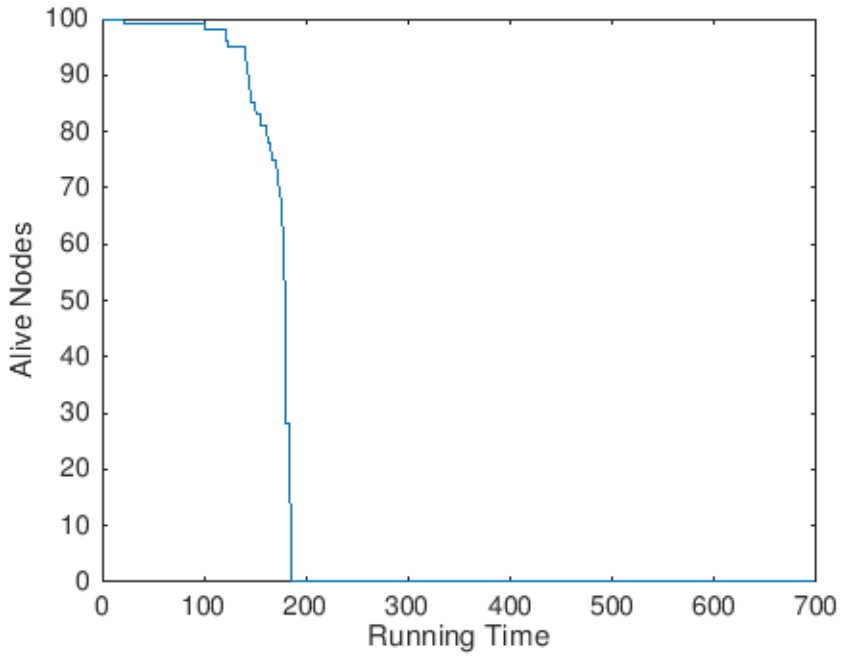
```
NodeNums = 100; % the num of node
AreaR = 100 ;   % the area of simulate
NodeTranR=10;   %  the transit Radius
Elec=50 * 10^(-9);  % Electric energy
Eamp=100*10^(-12);   %Transmit amplifier energy
Bx=50;  % The Postion of Base station
By=175;
MaxInteral =700; % the leach simulate time
Pch=0.05;  % the desired percentage of cluster heads
InitEn=0.5;  % the init energy of all node
Tr=30;
 TDMA=100;  %transmission schedule
Kbit=2000;   % the bits of a node transmiting a packet every time
%BandWitch = 1*10.^(6); %  Channel Bandwitch
TOS_LOCAL_ADDRESS = 0;
for i=1:(MaxInteral)
    AliveNode(i)=NodeNums;
    AmountData(i)=0;
end
sym alldata; %asaign alldata as symbolic variable
alldata=0;
LAECH = zeros(1,MaxInteral);
LAENO = zeros(1,MaxInteral);
for i=1:1:NodeNums
    EnNode(i)=InitEn;  % the init energy of all node
    StateNode(i)=1;    % the State of all node 1: alive 0:dead
    ClusterHeads(i)=0;  % the Set of Cluster Head ,1: cluster head
 0 :node

    Rounds=0;  % the round
end
Threshold=0;    % the threshold of node becoming a cluster-head

    Node.x=AreaR*rand(1,NodeNums);  % the position of node
    Node.y=AreaR*rand(1,NodeNums);
    Node.c=zeros(1,NodeNums);
    Node.d=zeros(1,NodeNums);
    Node.l=zeros(1,NodeNums);
    Node.csize=zeros(1,NodeNums);
     Node.initclEn=zeros(1,NodeNums);

  %phase 1 where condition for node to  become  a clusterhead%
for Rounds = 1:MaxInteral
   % the Setup phase of cluster
    Node.csize=Node.csize-Node.csize;
    Node.d=Node.d-Node.d;
    Node.c=Node.c-Node.c;
    for i =1:NodeNums
      Threshold=Pch/(1-Pch*(mod(Rounds-1,1/Pch)));
      if  StateNode(i)==1         % if  node is alive
          if ClusterHeads(i) ==1
              ClusterHeads(i)=0;
```

```
          elseif rand(1,1)<Threshold %node become cluster-head if the
    number is less than treshold

          ClusterHeads(i)=1;
          Node.c(i)=TOS_LOCAL_ADDRESS;
           Node.initclEn(i)=EnNode(i);
         else   ClusetHeads(i)=0;
                Node.initclEn(i)=EnNode(i);
         end
       end
     end
     if sum(ClusterHeads)==0
         continue;
     end
       %%cluster head sends the advertising paket that they will become
       %%a clusterhead
     EntranPCH =  Elec * Kbit+ Eamp*Kbit*((Tr.^2+Tr.^2));  % The
    expended engergy by new Cluster head advertising that it is new
    cluster head
     for i=1:NodeNums
         if ClusterHeads(i) ==1

           if  EnNode(i) >=  EntranPCH
                EnNode(i) = EnNode(i) - EntranPCH ;
               else
                StateNode(i)=0;
            end
        end
     end

     %2nd phase where the non clusterhead node receive the cluster head
     %advertisement and then sends join request to clusterhead that
    they are
     %the members of clusters....

     for i=1:NodeNums
       if  StateNode(i)==1         % if  node is alive
         if ClusterHeads(i) ~=1   % the node  is not cluster head
           for j=1:NodeNums
              if ClusterHeads(j) ==1    %the node is cluster head
                dist = ((Node.x(i)-Node.x(j)).^2)+((Node.y(i)-
    Node.y(j)).^2);  % the distance.^2
    %               if dist < (Tr.^2+Tr.^2)                % blong to the
     transmit radius
                  EnRecP =  Elec * Kbit ;               %the receiving
     energy
                   if  EnNode(i) >=  EnRecP            % the energy
     reciving a boardcast packet can expend
                     EnNode(i) = EnNode(i) - EnRecP ;
                   else
                     StateNode(i)=0;
                   end
                   if  Node.d(i) ==0                 % choose the
     cluster head
```

```
                                                        %assign cluster
 head on
                                                        %the basis of
 distance
                                                        %from base station
                    Node.d(i)=dist ;
                    Node.c(i)=j;
                   else
                    if  Node.d(i) > dist
                        Node.d(i)=dist ;
                        Node.c(i)=j;
                    end
                   end
%                 end
              end
          end

            if StateNode(i)==1
             Node.csize(Node.c(i))= Node.csize(Node.c(i))+1;
            end
         else      % the node  is  cluster head
          Node.d(i)=((Node.x(i)-Bx).^2)+((Node.y(i)-By).^2) ;
          Node.c(i)=TOS_LOCAL_ADDRESS;
         end
      end
     end

  %3rd phase each of the choosen cluster head creates a transmission
  %schedule......

    % the TDMA Phase
    alldata=0;
      for i=1:NodeNums
        if StateNode(i)==1  %if node is alive
         if ClusterHeads(i)==1 %if it is cluster head

             TolLengthPacket = Kbit.*Node.csize(i); % length of packet
 is defined as  kbit*Node.csize%
             alldata=alldata+TolLengthPacket;
             EntranPCH =  Elec * TolLengthPacket+
 Eamp*TolLengthPacket*(Node.d(i));
             EntranPCH.*TDMA;
              if  EnNode(i) >=  EntranPCH
                   EnNode(i) = EnNode(i) - EntranPCH ;
               else
                   StateNode(i)=0;
              end
          else
               EntranP = Elec * Node.l(i)+ Eamp*Node.l(i)*(Node.d(i));
               EntranP=EntranP.*TDMA;
               if  EnNode(i) >=  EntranP
                   EnNode(i) = EnNode(j)-EntranP;
               else
                   StateNode(i)=0;    % the node dead
```

```
                end
                 EnRecP =  Elec * Node.l(i) ;
                 EnRecP=EnRecP.*TDMA;
                if  EnNode(Node.c(i)) >=  EnRecP
                    EnNode(Node.c(i)) = EnNode(i) - EnRecP ;
                else
                    StateNode(Node.c(i))=0;
                end
            end
          end
       end
    if Rounds==1
        AmountData(Rounds)=alldata;
    else
       AmountData(Rounds)=alldata+AmountData(Rounds-1);
  end
    for i=1:NodeNums
         if StateNode(i)==0
             AliveNode(Rounds)= AliveNode(Rounds)-1;
         end
     end


end
  xtime= 1:1:MaxInteral;
  figure(1);
  plot(xtime,AliveNode),xlabel('Running Time'), ylabel('Alive
 Nodes'),;
```



Published with MATLAB® R2019a