IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

# Power Efficient On-Data Collection Using Mobile Robot in WSN

**Shiju Samuel[*1], S.Arulselvi[2], Dr Kirankumar[3]**

[*1,2,3] Bharath University, Chennai, India

shijuvector@gmail.com

## Abstract

The power efficient on data collection is to increase the energy efficiency condition even when there is problem in the failure of the main parent node. The on data collection is the concept, the power efficient data transfer in a network of nodes. The main concept that is used here is data gathering which is a method where the data from the child nodes are collected by the parent node first and then these child nodes sends the data to the base station periodically, if there exists a condition like failure of the parent node which collects the data, then the data will not be send to the base station. This paper propose a mobile robot which acts as a replacement for the failure and collects the data and forwards it to the base station. So that data loose is highly reduced. Thus the energy efficiency of the network is maintained.

**Keywords**: On-data Collection, Mobile robot, WSN.

## Introduction

### Overview

The design of wireless sensor networks depends on the application requirements. Environmental monitoring is an application where a region is sensed by numerous sensor nodes and the sensed data are gathered at a base station (a sink) where further processing can be performed. The sensor nodes for such applications are usually designed to work in conditions where it may not be possible to recharge or replace the batteries of the nodes. This means that energy is a very precious resource for sensor nodes, and communication overhead is to be minimized. These constraints make the design of data communication protocols a challenging task.

A common scenario of sensor networks involves deployment of hundreds or thousands of low-cost, low-power sensor nodes to a region from where information will be collected periodically. Hence, sensor nodes will periodically sense their nearby environment and send the information to a sink which is not energy limited. The collected information can be further processed at the sink for end-user queries. In order to reduce the communication overhead and energy consumption of sensors while gathering, the received data can be combined to reduce message size. A simple way of doing that is aggregating the data. A different way is data fusion (aggregation) which can be defined as producing a more accurate signal by combining several unreliable data measurements. In this paper, we focus on scenarios where perfect aggregation is used while gathering data, meaning that all forwarded messages are of the same size. An important problem studied here is finding an energy efficient routing scheme for gathering all data at the sink periodically so that the lifetime of the network is prolonged as much as possible. The lifetime can be expressed in terms of rounds where a round is the time period between two sensing activities of sensor nodes. There are several requirements for a routing scheme to be designed for this scenario. First, the algorithm should be distributed since it is extremely energy consuming to calculate the optimum paths in a dynamic network and inform others about the computed paths in a centralized manner. The algorithm must also be scalable. The message and time complexity of computing the routing paths must scale well with increasing number of nodes. Another desirable property is robustness, which means that the routing scheme should be resilient to node and link failures. The scheme should also support new node additions to the network, since not all nodes fail at the same time, and some nodes may need to be replaced. In other words, the routing scheme should be self-healing. The final and possibly the most important requirement for a routing scheme for wireless sensor networks is energy efficiency.

A previous study showed that the minimum spanning tree (MST)-based routing provides good performance in terms of lifetime when the data are gathered using aggregation. In that work, the authors proposed a new centralized protocol called PEDAP. The idea in PEDAP is to use the minimum energy cost tree for data gathering. This tree can be efficiently computed in centralized manner using Prim's minimum spanning tree algorithm. In PEDAP-PA, the authors changed the cost of the links so that the remaining energy of the sender is also taken into consideration. Since the link costs vary over time, the authors proposed re-computing the routing tree from time to time using a power-aware cost function. By changing the routing tree over time, the load on the nodes is balanced and a longer lifetime compared to the static version is achieved. In this way, the lifetime in terms of the first node failure is almost doubled.

The most important disadvantage of these two protocols, however, is their centralized nature. In this paper, we propose a localized version of PEDAP, which tries to combine the desired features of MST and shortest weighted path-based gathering algorithms. We also expand the idea and propose a new family of localized protocols for the power-efficient data aggregation problem. Our main concern in this work is the lifetime of the network. We name our new approach localized power-efficient data aggregation protocol (L-PEDAP).Our proposed routing approach consists of two phases and satisfies the requirements stated above. In the first phase, it computes a sparse topology over the visibility graph of the network in a localized manner. In the second phase, it computes a data gathering tree over the edges of the computed sparse topology. The topology needs to be efficiently computed by using only the one-hop neighborhood information.

For the first phase, we test two different sparse topologies in a distributed manner, namely, local minimum spanning tree (LMST) and relative neighborhood graph (RNG). These structures are supersets of MST and can be efficiently computed in a localized manner. For the second phase, we propose three different methods and provide performance results of them. All of the methods are based on flooding a special packet using only the edges of the computed structure. According to the decisions made during this flooding process, the tree is yielded. These three methods that can be executed at a node for choosing the parent node toward the sink are to choose:

- The first node from which the special packet is received,
- The node that minimizes the number of hops to the sink, and
- The node that minimizes the total energy consumed over the path to the sink.

Our solution can also handle new node arrivals and departures of existing nodes. Hence, it is adaptive. The routing path is maintained when those dynamic conditions occur. We also propose power-aware versions of our protocols that consider the dynamic changes in the remaining energy levels of nodes while constructing the sparse topologies and routing trees. We also derive a new theoretical upper bound for the lifetime in terms of the first node failure. We used this upper bound to see how close our protocols are to the theoretical limit. The simulation results showed that our protocols can achieve up to 90 percent of the upper bound. We also include a enhancement of using a mobile robot which will act as a node for transferring when the main parent node gathering the data becomes the failed one. So by using this we are able to make the network a more efficient one and also the packet drop is minimized 100 percent.

**Background**

In the paper *"Power efficient on-data collection using mobile robot in wsn"*, the author describes about making the network more efficient. They describes the condition that when a certain datum is transferred to a sink or base station. In these wireless sensor network the travelling never stops till it reaches the destination. So the link is to be used for all time when we consider about the entire data transfer. The point is that while the data transfer is going on the average energy of the node get decreased. So at final they will become dead node. This cause a reduce in efficient use of energy. Here the authors used a technique called the data gathering /data aggregation which uses PEDAP protocol. In this method we can see that the data from the child node are being collected by the parent node and are transferred to the base station periodically. Here the usage of link is limited by data and are also send without fail. In these conditions the base station will be receiving data periodically from its child node. The collecting capacity of the node is limited. So for up to a capacity of node the data is collected and then it is transferred. These concept become bases for our projects. With along these the paper also deals with those problems that may appear like addition of node, node failure and node recompilation. Addition of node will make new route discoveries and match the new node to the network. Node failure concept gives the fact that how a network recomputed to avoid the loss of data when a node becomes failure.

In the paper *"On Data Collection Using Mobile Robot in Wireless Sensor Networks"* the author describes a novel data-collecting algorithm using a mobile robot to acquire sensed data from a wireless sensor network (WSN) that possesses partitioned/islanded WSNs is proposed in this paper. This algorithm permits the improvement of data collecting performance by the base station by identifying the locations of partitioned/islanded WSNs and navigating a mobile robot to the desired location. To identify the locations of the partitioned/islanded WSNs, two control approaches, a global- and local-based approach, are proposed. Accordingly, the navigation strategy of the robot can be scheduled based on time and location using three scheduling strategies: time based, location based, and dynamic moving based. With these strategies, the mobile robot can collect the sensed data from the partitioned/islanded WSNs. Therefore, the efficiency of sensed data collected by the base station in partitioned/islanded WSNs is improved. Through simulation under the environment of an ns-2 simulator, the results, from various aspects, show that the collecting strategies proposed can dramatically improve sensed data-collecting performance in partitioned or islanded WSNs.

The scope of this paper is to increase the energy efficiency condition even when there is problem in the failure of the main parent node. Since the process of data aggregation is being carried out here, there is few possibility of changing the main parent node to the nearby main parent node, since it may cause to drop of data. In these circumstances, we introduce the use of mobile robot to act as the failed main parent node .thus the energy efficiency of the network is maintained.

## Organisation of the Work

We have divided this paper in mainly into four phases or modules. They are

- Module 1 : Route Computation
- Module 2 : Data gathering
- Module 3 : Route Maintenance
- Module 4 : Mobile robot concept

**Route Computation**

This module describes the initial phase of our proposed L-PEDAP protocol. Initially each sensor broadcasts Hello packets to Neighbors such as nodes which are in range. So that each node will identify its Neighbors. After that the Sink (Base Station) will broadcasts Route-Discovery packets to all the sensor, which makes each sensor should identifies its Parent Child sensors. While choosing Parent Node each sensor should aware of Energy-Efficient factor.

**Data Gathering**

After the parent and children nodes for an individual sensor node are determined, the node can join the data gathering process. In data gathering phase, each sensor node periodically senses its nearby environment and generates the data to be sent to the sink. However, before sending it directly to the parent node, it will wait all the data from its child nodes and aggregate the data coming from them together with its own data, and then, send the aggregated data to the parent node (Base Station).

**Route Maintenance**

In this module we describe three events which cause changes in routes, such as route re-computation, node failure, and node addition. In route re-computation the sink node will broadcast a new ROUTE-DISCOVERY packet with a new sequence ID.

**Mobile node enhancement**

In future enhancement we address the problem of node failure, in case of no alternative path find to send sensed data to Base Station. In those cases we propose the data collection assisted by Mobile robot concept

.

## Materials and Methods

**Existing method and its Problem**

The most important requirement for a routing scheme for wireless sensor networks is energy efficiency. The existing system showed that the minimum spanning tree (MST)-based routing provides good performance in terms of lifetime when the data are gathered. In that work, a new centralized protocol called PEDAP. The idea in PEDAP is to use the minimum energy cost tree for data gathering. This tree can be efficiently computed in centralized manner using Prim's minimum spanning tree algorithm. In PEDAP-PA, the authors changed the cost of the links so that the remaining energy of the sender is also taken into consideration. Since the link costs vary over time, the authors proposed re-computing the routing tree from time to time using a power-aware cost function. The most important disadvantage of these two protocols, however, is their centralized nature.

**Proposed Method**

Our proposed approach based on a localized version of PEDAP, which tries to combine the desired features of MST and shortest weighted path-based gathering algorithms. We also expand the idea and propose a new family of localized protocols for the power-efficient data aggregation problem. Our main concern in this work is the lifetime of the network. We name our new approach localized power-efficient data aggregation protocol (L-PEDAP). Along with this the case in which the failure of the main parent node is also solved by using a mobile node.

## System Analysis and Design

**Power Efficient Topology**

The topologies that we focus in this work are supersets of euclidean MST. One of them is the RNG. An edge eij is included in the Euclidean RNG graph if there are no nodes closer to both nodes i and j than the distance between nodes i and j. That is, an edge eij remains in RNG if it does not have the largest cost in any triangle ikj4, for all common neighbors k. The euclidean MST of a graph is a sub graph of its RNG.

Another literature proposes a neighborhood structure called local MST (LMST) as an alternative topology. LMST is computed as follows: First, each node determines its one-hop neighbors and computes an MST for that set of nodes, based on the distance between nodes as the weight of the edges. After computing the MST of the neighbors, each node i selects the edges (eij) where node j is a direct neighbor of node i in its MST. The resulting structure is a directed graph. The structure can be converted to an undirected one in two ways. First way is to include edge (eij) only when both nodes i and j include that edge (LMST_). The second way is to include that edge when either node i or node j includes it (LMST þ). In this work, we choose to use LMST_ in our simulations, but our algorithm can support both.

There are some desirable properties of the LMST structure which make using the structure in the context of sensor networks advantageous. MST of a graph is a sub-graph of its LMST and the LMST is a sub-graph of its RNG. The maximum degree of a node is bounded by 6. In the authors compare their LMST structure with the enclosure graph and find out that the enclosure graph performs better in terms of energy consumption. However, the comparison did not consider the effect of data aggregation. Although the RNG and LMST structures are defined based on Euclidean distances, they can be used with other link cost functions as long as the functions are symmetric.

## System Implementation

**Our Approach**

Our approach for solving the aggregation and routing problem consists of two phases: topology construction and aggregation/routing tree computation.

**a. Topology Construction**

In this phase, we aim to construct a sparse and efficient topology over the visibility graph of the network in a distributed manner. We have different alternatives for sparse topologies that can be efficient for energy-aware routing. In this work, we choose to investigate the use of RNG and LMST and compare their relative performance. We expect that LMST performs better than RNG because it's sparser.

**b. Routing Tree Computation**

There are several methods for obtaining a tree structure we investigate the efficiency of three different methods:

- first parent path method (FP),
- nearest minimum hop path method (MH), and
- Shortest weighted path (i.e., least cost) method (SWP).

The FP method is the simplest among the three. In this method, a node will set its parent as the first neighboring node (among neighbors in selected sparse structure) from which the special route discovery packet was received. In the MH method, the node chooses its nearest neighbor among those with minimum hops to reach to the sink. So, the node updates its parent only if the sender node has a smaller hop count or has the same hop count as the current parent, but it is closer than the current parent (among neighbors in selected sparse structure). Otherwise, the packet is ignored. The SWP method tries to yield a tree has minimizes the cost of reaching the sink for each node.

Our solution consists of four parts:

- Route Computation,
- Data Gathering,
- Route Maintenance.
- Mobile robot concept

**Topology and Route Computation**

The main goal in this phase is to find a sparse topology and set up the routes over it, which means determining the children and parent nodes for each node. At the end of this phase, a data aggregation tree rooted at sink is constructed.

The pseudo code for this phase is given in Algorithm 1.

*Algorithm 1. Topology and Route Computation*

1: Send HELLO message
2: Collect HELLO messages for thello
3: Reset Parent (_ null)
4: Compute neighbors on the sparse topology
5: while ROUTE-DISCOVERY packet RD received in tdiscovery do
6: if update required for RD then
7: Update parent (_ source RDÞ)
8: Broadcast ROUTE-DISCOVERY
9: end if
10: end while
11: Inform _ to construct its child-list

Initially, the nodes and the sink are not aware about the environment. In the setup phase, all nodes and the sink broadcast HELLO messages, which include their location and remaining energy, using their maximum allowed transmit Power. The route computation is done via a broadcasting process which starts at the sink node. The sink initiates a ROUTE-DISCOVERY packet in order to find and set up the routes from all sensor nodes toward itself. When a sensor node receives a ROUTE-DISCOVERY packet, it broadcasts the packet to all its neighbors on the computed topology if it updates its routing table. By this way, the routing tree rooted at the sink is established over the sparse topology.

Each ROUTE-DISCOVERY packet has three fields: a sequence ID which is increased when a new discovery is initiated by the sink, an optional distance field which shows the cost of reaching the sink, and an optional neighbor list field which is the list of the neighbors of the sending node in the chosen topology.

**Data Gathering**

After the parent and children nodes for an individual sensor node are determined, the node can join the data gathering process. In data gathering phase, each sensor node periodically senses its nearby environment and generates the data to be sent to the sink. However, before sending it directly to the parent node, it will wait all the data from its child nodes and aggregate the data coming from them together with its own data, and then, send the aggregated data to the parent node. Thus, at the beginning of data gathering step, only leaf nodes can transmit their data to their corresponding parent nodes. At each step, the data are gathered upward in the tree and reaches the sink after h steps, where h is the height of the aggregation tree. The reason for waiting to receive data from child nodes is to use the advantage of the aggregation.

**Route Maintenance**

After setting up the routes, three events can cause a change in the routing plan: route recomputation, node failure, and node addition. Recomputation of the aggregation tree is required when power-aware (dynamic) cost functions are used. In power aware methods, the tree must be recomputed at specified intervals. Since the computation depends on the remaining energy of nodes, each time the computation takes place and a different and more power-efficient plan is yielded. In our case, we handle this requirement by broadcasting a new ROUTE-DISCOVERY packet with a new sequence ID.

Node failures can be due to various reasons. However, the most critical reason is depletion of energy of a node. Failure of a node due to energy depletion can be handled gracefully. In our solution, when a node's energy reduces below a threshold value, the node broadcasts a BYE message using maximum allowed transmit power. The nodes receiving the BYE message will immediately update their local structure. The child nodes of the failed node that receive the BYE message reset their routing tables and enter the parent-discovery phase by broadcasting a special message PARENT-DISCOVERY to its neighbors on the structure. According to the receiver of that special message, if the sender is its own parent on the way to the sink, the receiver also resets its routing table and broadcasts the packet to its neighbors. In this way, all the nodes that should enter the parent-discovery phase will be reached. If the PARENT-DISCOVERY packet is received by a neighboring node of the sender and if it has a valid parent, the receiver constructs a new ROUTE-DISCOVERY packet as mentioned above and broadcasts it to the sender.

*Algorithm 2. Route Recovery*

1: _old _
2: if BYE message B received then
3: remove source BÞ from neighbor list
4: compute the sparse topology
5: if sourceðBÞ ¼ _ then
6: Reset parent (_ null)
7: Reset child list
8: Broadcast PARENT-DISCOVERY message
9: Enter route discovery phase
10: end if
11: end if
12: if PARENT-DISCOVERY message PD received then
13: if sourceðPDÞ ¼ _ then
14: Reset parent (_ null)
15: Reset child list
16: Broadcast PARENT-DISCOVERY message
17: Enter route discovery phase

```
else
if _ 6¼ null then
Send ROUTE-DISCOVERY
end if
end if
end if
if _ 6¼ _old then
Inform _old and _ to construct their child-list
end if
```

Consider now the case of node additions. When a new node is deployed, it broadcasts a HELLO message. Its neighbors update their local structure upon receiving this message and also inform the new node about their existence and locations by replying a HELLO message so that the newly deployed node can also determine its neighbors. Nodes that update their local structure send back a ROUTEDISCOVERY packet including their costs to the newly deployed node. The new node selects the most efficient node as its parent and broadcasts this information by a new ROUTE-DISCOVERY packet.

**Mobile robot concept**

In this concept when a parent node nearby to the base station goes dead i.e. becomes low threshold level then the mobile robot , which is a node nearly placed will be acting as the failure one. The base station sends the data that the node is to be failed to the mobile node. Hence it act like a transporter by gathering the message from the failed node and sending to the base station.

## Results and Conclusion

**Stage Description**

A node has to find the neighbours in order to continue the path in the network. A node sends hello packets to its surroundings and certain node retransmits the message then those are the neighbouring nodes.

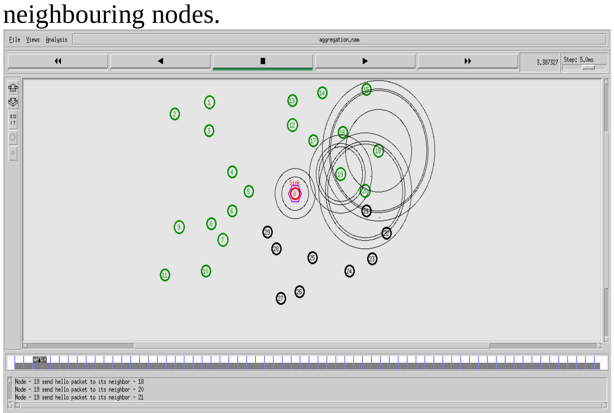

**Fig (a). Hello Packets For Finding Neighbours**

In order to transfer the packet the entire topology considered as the tree with base station as its starter from main parent till the child nodes. For finding those paths, route discovery program is done.

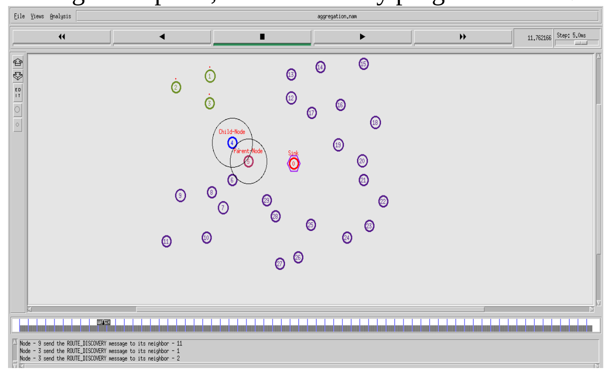

**Fig (b). Route Discovery To Find The Parent Child**

The packets send from each child node travels up to the main parent node. And the according to our concept the main parent node aggregate the datas and send these only periodically. For a certain period of time there will be no transmission to the sink and then a burst of data will be transmitted to the base station.

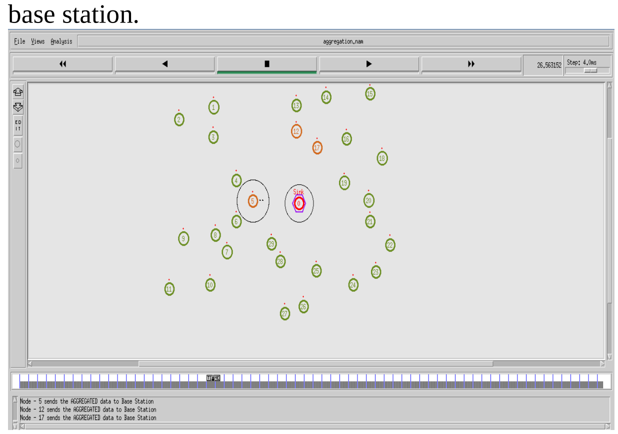

**Fig (c). Data Aggregation**

While transmission is going on, the energy of the node decreases. Finally it reaches below the threshold level which is the condition of failure.

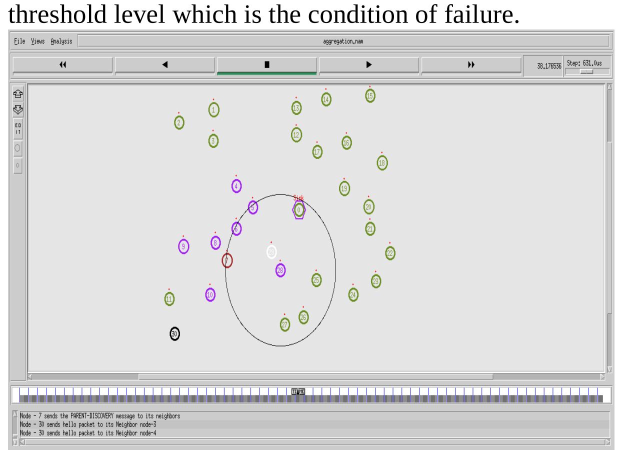

**Fig (d). Node Failure Condition**

Cases appear like failure of main parent node. In other cases it can be routed to neighbouring node. But the main parent will be having aggregated data which is high. So it is routed to nearby node ie.another main parent node then the case of packet drop occurs due to overcrowding.

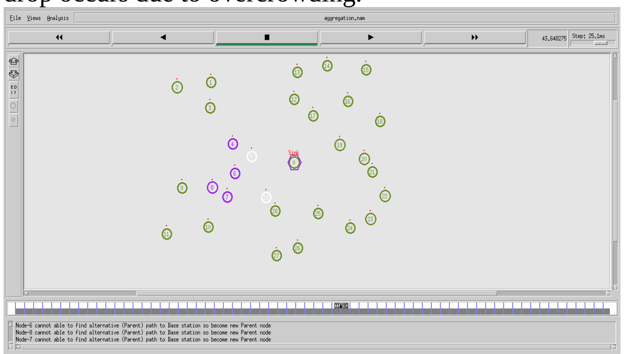

**Fig (e). Main Parent Node Failure**

After getting the information that there is a main parent node failure, the mobile robot node goes to base station and gather information about the failed node like its position ,number,data capacity etc

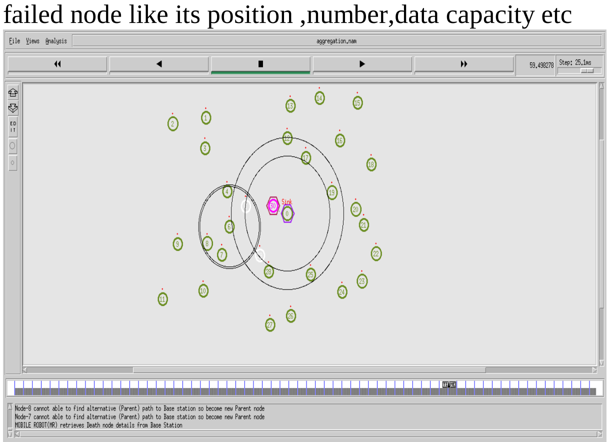

**Fig (f). Mobile Node Gathering Data From Base Station**

When main parent node fails the mobile robot concept appears. it, after getting information from the failed node goes to it and takes the data it has collected.

**Fig (g). Mobile Node Gathering Data From Failed One**

After collecting the data from the failed one, it again reverses to the base station and sends the gathered data. Thus the network transmission is completed.

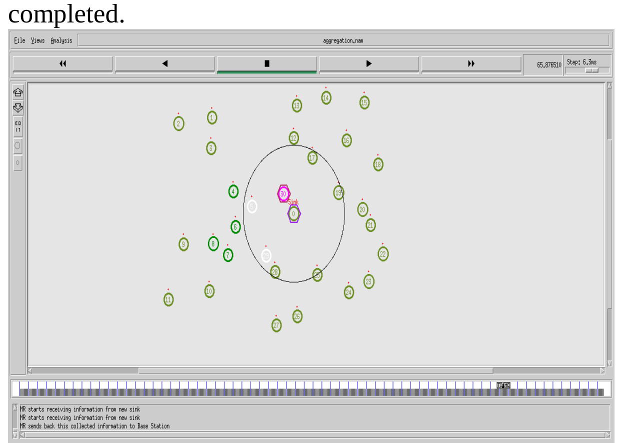

**Fig (h). Mobile Node Transferring Data To Base Station**

## Graph Analysis

Throughput is the number of successfully transmitted packet. Since packet drop is highly minimized here or can say that minimized to level zero, the throughput is maximum for that. All the packets ie.send is reached at the destination for a while there will be no transmission to base station because while using data gathering there will be no transmission to the base station as the data ie.gathered by the main parent node initially and after a period only, the parent node starts to transfer the data to base station. In the previous one as there are drops occurring ,the throughput is minimized .and also we can see that very soon after the node configuration the data starts to travel to base station .but in the enhanced one we can see high throughput and it is starts only after a certain period used for data gathering.

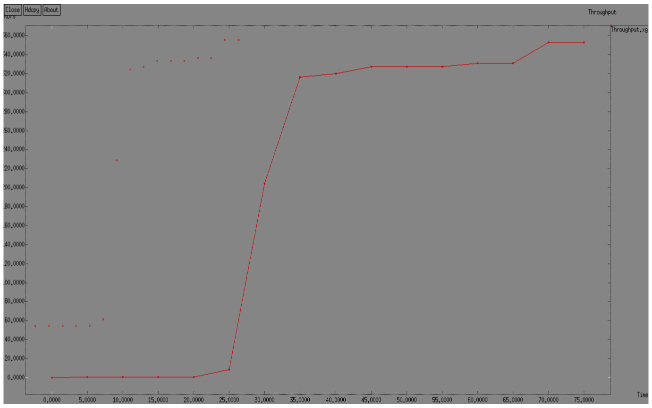

**Fig (i). Throughput**

In the first part itself by using the energy efficiency the packet drop is minimized to a level. The only case of drop is when main parents failed it shares the aggregated data of its child node to the nearby main parent node. So then by more data than the expected one, it starts to drop packet. This is avoided in our project by using mobile node in place of failed main parent node. Thus packet drop is minimized to zero. Even though ,initially energy efficiency is practised drops due to main parent node failure occurs and the node is not able to withstand as much data gives to it .but in second one by the use of mobile node we can see that packet drop is nullified as the previous s a problem gets a solution.

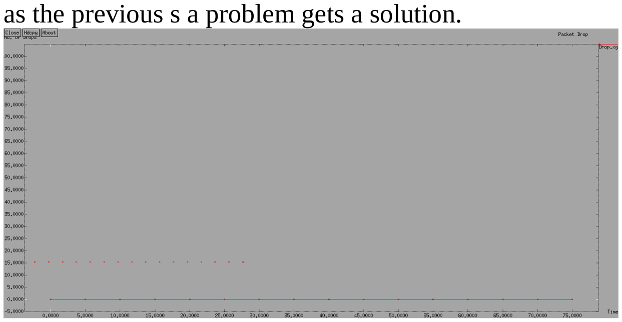

**Fig (j). Packet Drop**

In the first part itself by using the energy efficiency the packet drop is minimized to a level. The only case of drop is when main parents failed it shares the aggregated data of its child node to the nearby main parent node. So then by more data than the expected one, it starts to drop packet. This is avoided in our project by using mobile node in place of failed main parent node. Thus packet drop is minimized to zero. Even though ,initially energy efficiency is practised drops due to main parent node failure occurs and the node is not able to withstand as much data gives to it .but in second one by the use of mobile node we can see that packet drop is nullified as the previous s a problem gets a solution.

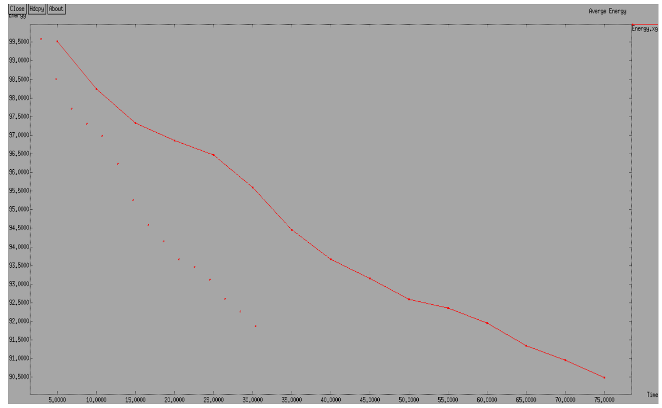

**Fig (k). Average Energy**

Energy is more extended for each node. Also it is able to withstand to a much lower level. Since the data gathering is used have the node is not functioning ie. transferring of data from the nodes to the base station is not continuously. Data gathering uses nodes only for creation period, so that the energy is also used efficiently. The average energy of nodes decreases very soon in first case as the nodes are used continuously for transmission .but in better one as the data gathering occurs the efficient use of the nodes is done .so that the energy usage is prolonged and get a more life time.

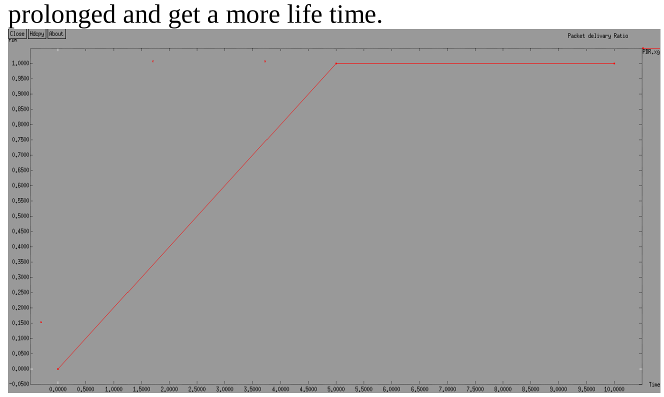

**Fig (l). Packet Delivery Ratio**

Packet delivery ratio is the ratio of received one to transmitted data. Since in this works there is no packet drop and also has high throughput. The packet delivery ratio is not equal to one as packet drop occurs. So the number of transmitted packet and reviewed packets are not same .so packet drop is less than one in first case. But in enhanced we can see that as much packets are sent the same number is reached at the output .so packet delivery ratio is reached one.

## Conclusion

In this paper, we enhanced the process of energy efficiency concept while in the case of failure of main parent node nearby to the base station. The energy-efficient routing approach is a combination of the desired properties of minimum spanning tree and shortest path tree-based routing schemes. The proposed scheme uses the advantages of the powerful localized structures such as RNG and LMST and provides simple solutions to the known problems in route setup and maintenance because of its distributed nature. The proposed algorithm is robust, scalable, and self-organizing. The algorithm is appropriate for systems where all the nodes are not in direct communication range of each other. We show through simulations that our algorithm will give a zero packet drop condition and satisfies a complete energy efficient need for the network

## Acknowledgement

Thanks to Dr. Kiran Kumar, who comforted me in various nodal issues simulating this paper and also Mrs. s. Arulselvi for her help in seeding me regarding the practical implementation and its pros and cons. The insightful and detailed feedback and suggestion by them which improved content and presentation of this paper.

## References

[1] Hu¨ seyin ozgu¨r tan, member, ieee, Ibrahim korpeoglu, member, ieee, andivan stojmenovic, fellow, ieee," computing localized power-efficient data aggregation trees for sensor networks" ieee transactions on parallel and distributed systems, vol. 22, no. 3, march 2011

[2] Tzung-cheng chen, tzung-shi chen, *member, ieee*, and ping-wen wu," on data collection using mobile robot in wireless sensor networks", ieee transactions on systems, man, and cybernetics—part a: systems and humans,vol.30,no.3,april 2011

[3] H.o. tan, i. Korpeoglu, and i. Stojmenovic, "a distributed and dynamic data gathering protocol for sensor networks," proc. 21st int'l conf. Advanced networking and applications, pp. 220-227, 2007.

[4] A. Ababnah and B. Natarajan, "Optimal control-based strategy for sensordeployment," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 41, no. 1, pp. 97–104, Jan. 2011.

[5] W.r. heinzelman, a. Chandrakasan, and h. Balakrishnan,"energy-efficient communication protocol for wireless microsensor networks," proc. 33rd ann. Hawaii int'l conf. System sciences, pp. 3005-3014, 2009.

[6] V. Rodoplu and t. Meng, "minimum energy mobile wireless networks," ieee j. Selected areas in comm., vol. 17, no. 8, pp. 1333-1344, aug. 2007.

[7] S. Singh, m. Woo, and c.s. raghavendra, "power-aware routing in mobile ad hoc networks," proc. Int'l conf. Mobile computing and networking, pp. 181-190, 2008.

[8] I. Stojmenovic and x. Lin, "power-aware localized routing in wireless networks," ieee trans. Parallel and distributed systems, vol. 12, no. 11, pp. 1122-1133, nov. 2008.

[9] J.-h. Chang and l. Tassiulas, "energy conserving routing in wireless ad-hoc networks," proc. Ieee infocom '09, pp. 22-31,mar. 2009.

[10] J. Chang and l. Tassiulas, "maximum lifetime routing in wireless sensor networks," proc. Advanced telecomm. And information Distribution research program conf., 2010.