INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH & DEVELOPMENT

ISSN 2278 – 0211 (Online)

Duty Cycle and Link Life Time Prediction Routing In Wireless Sensor Network

N. Lavanya
Department of ECE, Prist University, Puducherry, India
V. Saranya
Assistant Professor, Department of ECE, Prist University, Puducherry, India

Abstract:
The nodes in sensor networks have limited battery power and it is not feasible or possible to recharge or replace the Batteries, therefore power consumption should be minimized so that overall Network lifetime will be increased. In order to minimize power consumed during idle listening, some nodes which can be considered redundant can be put to sleep. This paper presents a life time prediction routing protocol for WSN that maximizes the network life time. Link Life Time Prediction Algorithm (LLP) is proposed and Compared to Three energy-efficient online traffic scheduling algorithms in terms of sensor ratio and time averaged coverage. Offline scheduling results are presented that provide lower bounds on the energy needed to satisfy the communication requirements.

***Key words**: duty cycle control, energy efficiency, Link Lifetime prediction, Wireless Sensor Networks*

1. Introduction

Low-cost, low power, multifunctional sensor nodes that are small in size and communicate in short distances have been developed due to the recent advances in micro-electro-mechanical systems and wireless communication [1]. These tiny sensors have the ability of sensing, data processing, and communicating with each other. Wireless Sensor Networks (WSN) which rely on collaborative work of large number of sensors are realized. Sensor nodes can be used within many deployment scenarios such as continuous sensing, event detection, event identification, location sensing, and local control of actuators for a wide range of applications such as military, environment, health, space exploration, and disaster relief. The main sources of power dissipation are used during data processing, data transmission, data reception and idle listening. The power consumed during transmission is the greatest portion of energy consumption of any node.

Considering the limited capabilities and vulnerable nature of an individual sensor, a wireless sensor network has a large number of sensors deployed in high density (high up to 20nodes/m3. Since the nodes are deployed densely and

In an ad hoc fashion, many nodes stay inactive for long periods and idle listening power dissipation becomes significant. Therefore, these nodes can be considered as redundant and can be put to sleep. The main idea will be scheduling sensors to work alternatively and the system lifetime will be prolonged correspondingly. The algorithm that we proposed is self-configured, fully distributed. Since the working environments for WSNs are hostile and remote working environments, it would not be convenient or possible to configure the network manually after deployment. For this reason self-configuration is necessary. In order to erase the need for a global synchronization overhead, the proposed algorithm has to be distributed and localized. This favours also scalability of the network.

2. Previous Work

The main focus of research on routing protocols in WSN has been network performance.

2.1. Power–aware Routing

$$Min \sum_{i \in \pi} Ti, i+1$$

Where $Ti, i+1$ denote the power expended for transmitting between two consecutive nodes, *i* and *i+1*, in route π. This link cost can be defined for two cases:

- When the transmit power is fixed
- When the transmit power is varied dynamically

As a function of the distance between the transmitter and intended receiver [9]. The key requirement of this technique is that the relative positions of nodes are known to all nodes in the MANET.

2.2. Battery-cost Lifetime-aware Routing

Nodes along the least-power cost routes tend to "die" soon because of the battery energy exhaustion. This is doubly harmful since the nodes that die early are precisely the ones that are needed most to maintain the network connectivity (and hence useful service life). Therefore, it will be better to use a higher power cost route if this routing solution avoids using nodes that have a small amount of remaining battery energy [7]. This observation has given rise to a number of "*battery-cost lifetime-aware routing*"

2.3. Broadcasting in WSN

Broadcasting is an important communication paradigm in all networks including wireless sensor networks. The simplest way to broadcast a packet is *flooding*. In this technique, every node retransmits a packet once when it receives the packet for the first time. It is a very simple technique and ensures that every node receives the packet. The disadvantage of flooding is that it generates abundant retransmissions causing the wastage of battery energy and bandwidth. Retransmissions by geographically close nodes result in message collisions and channel contentions. This scenario is known as the broadcast storm problem.

Extensive research has been conducted to reduce the number of retransmissions during the broadcast operation. This optimization leads to the design of energy efficient broadcast protocols that are a necessity for energy-constrained wireless networks [14]. Research is also conducted to build up protocols that will achieve reachability as well as latency-optimized operation. To organize the discussion of protocols, we divide them into a number of groups depending on a number of aspects. Algorithms belonging to the same group have some common characteristics. In the following subsections we will describe the algorithms from various categories.

2.4. Adaptive Broadcasting

To alleviate the broadcast storm problem of simple flooding, several threshold-based broadcasting techniques are proposed. In the counter-based scheme every host maintains a counter c for each packet. This counter c is used to keep a record of the number of times a host has received a broadcast packet. When c reaches a predefined threshold value C, the host refrains from rebroadcasting the packet as the additional coverage achieved through this transmission is very low [3]. In the location-based scheme, each host is assumed to be equipped with a positioning device such as GPS. A receiver can accurately calculate the additional coverage that can be achieved from the location of the source from which it heard the broadcast packet. The receiving host uses a predefined threshold A to determine whether it should rebroadcast or not. The location based scheme achieved better performance in terms of both reachability and the amount of savings as compared to the counter-based scheme as more accurate information is used. Adaptive Counter-Based scheme, dynamically adjusts the threshold value $C(n)$ based on local neighbor information and introduces a time delay before broadcasting a packet to reduce the number of redundant transmissions further. A small value of $C(n)$ can significantly reduce the number of redundancies in a dense network while achieving a better reachability. For sparse networks, greater values of $C(n)$ should be used to achieve reachability, which will increase the number of rebroadcasts. As shown in Figure 1, when $n < n1$, $A(n)$ should be 0 to enforce a host to rebroadcast. Between $n1$ and $n2$, $A(n)$ gradually increases to balance savings and reachability. After $n > n2$, $A(n) = 0.187$ is used which is the expected additional coverage achieved after a host receives same broadcast packet twice.

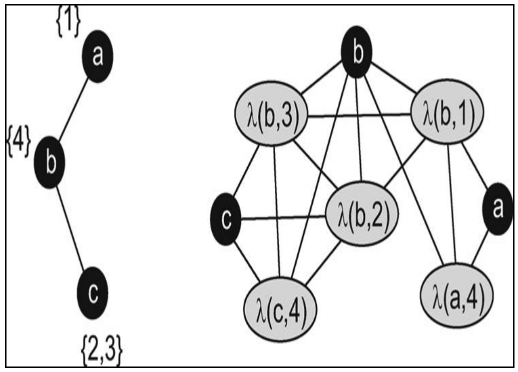

Figure 1: Duty cycled WSN graph

3. Routing Protocols

The routing protocols proposed for WSNs are classified considering several architectural factors. Taxonomy of routing protocols is helpful while designing the network protocol.

Advantages of proposed Systems

A efficient collision-free scheduling scheme on top of a multicast tree to avoid packet loss can be developed. Total energy cost is reduced to a large extent.

A distributed implementation of the proposed algorithms is presented and extensive simulations are conducted to evaluate the performance of proposed algorithms.

3.1. Data Centric Protocols

It is not appropriate to use global identifiers for this huge number of randomly deployed nodes, in most of the WSN applications [7]. However this introduces complexity to query data from a specific set of nodes. Therefore the data is collected from the deployed region.

3.2. Greedy Minimum Cost Flow (GMCF) Scheduler

Unlike the bound that incorporates all communication requirements at once, GMCF is executed upon arrival of a new node *v'* into the multicating unit range at time *t'*. The capacity between nodes *S* and *i* € *N'* is given by H_i*'*, which represents node *i*'s unsatisfied communications requirements at time *t' [4]*.

In GMCF, the number of nodes *n* is *v* + *t*, where *v* is the number of nodes inside the coverage range, and *t* is the number of timeslots needed to exit the multicating unit coverage range.

$$I^T = -O^T =_{i \in N-}^{1/2} H_{i-}^T$$

Let this graph be denoted $G^T(V^T, ET)$, where $V^T = \{S\} \cup N^T \cup T^T \cup \{D\}$, and E^T is the set of edges between *S*, *N*, *T*, and *D*. In the GMCF algorithm, both nodes representing the flow graph will be connected to the same timeslot nodes with the same cost over the corresponding edges. This leads the solver to treat them equally, and the assigned timeslots between the two nodes will be in arbitrary order, as one would expect since there is no preferential schedule from an energy viewpoint. To further clarify the different behaviors of the online algorithms, we discuss a simple example of two nodes arriving at the same time, from either the same direction or opposite directions. The two nodes have the same communication demand and travel at the same speed.

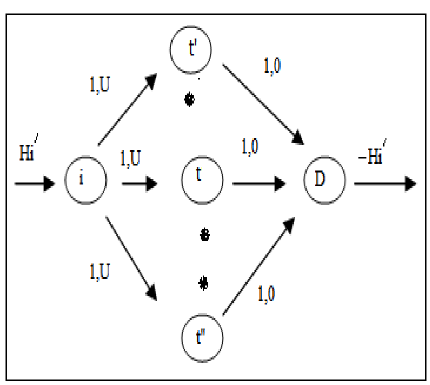

Figure 2: Example flow graph M for the SS algorithm. This is used in the scheduling phase.

3.3. Static Scheduler (SS)

The basic idea in SS is to sort nodes according to the energy they would use if they were served at energy optimal positions. The algorithm is static in the sense that these weights do not change as the node propagates through the multicasting unit coverage range. SS serves nodes with high energy costs first to reduce the total energy required. SS is executed upon the arrival of a new node *v'* into multicasting unit range at time *t'*. The algorithm consists of two phases, namely, weight computation and scheduling. In the weight computation phase, the weight Wi' for each node i € *N'* is computed by finding its optimal energy cost. In determining the complexity of SS, we will use the same notation as in the complexity analysis of GMCF, where *v* is the number of nodes inside the range, t is the number of time slots needed for them to exit the multicasting unit range, and we add to them *Hm* as the maximum number of slots a node can demand. SS is invoked upon the arrival of each new sensor node. In the weight computation phase, the process of finding the weight is executed for each sensor [4]. Finding the weight is equivalent in complexity to finding the minimum of an array of length *t*.SS is executed upon the arrival of a new node v^T into RSU range at time t. The algorithm consists of two phases, namely, weight computation and scheduling. In the weight computation phase, the weight W^T for each node $i \in N^T$ is computed by finding its optimal energy cost. This can be best described by using a minimum cost flow graph as in Fig. 2 but restricted solely to the node in question. We denote that flow graph as G_i for node i. An example of this is shown in Fig. 3 and contains one source node i, which generates a flow equal to node i's remaining demand H^T at time t.

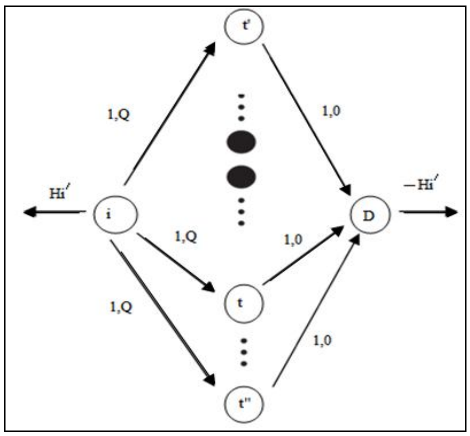

Figure 3: Example for SS Algorithm

This is used in the calculation phase

The intermediate nodes represent timeslots of the set T_i, starting at t^T and ending with t^T. The edge capacity between the node i node and time node $t \in T_i$ is set to 1. The capacity and cost for edges between time node $t \in T_i$ and dummy destination D are 1 and 0, respectively. The minimum cost flow for graph G_i is computed, and the cost associated with this flow is the weight W_i for node i.

3.4. Nearest Fastest Set (NFS) Scheduler

The NFS scheduler uses sensor inputs in a simpler and more dynamic way than SS. The motivation is to dynamically change the nodes according to the remaining demands. If a node is selected for communication from the multicasting unit at the current time slot, its cost is reduced while the cost of other nodes is increased. The notion of "fastest" comes from the role that node speed plays in weight computation. Consider the case where two sensors are together and moving away from the multicasting unit [4]. If they are moving at different speeds, then serving the faster one first will lead to lower overall energy consumption. This is clearly due to the fact that in the next time step the faster node will be farther away from the multicasting unit. nfs uses this by embedding the effect of sensor proximity and velocity in the weight calculation when considering which node to serve in a given timeslot.As the preparation phase is executed for every node *i € n'* and *t € ti* currently inside the multicasting unit range upon their respective arrivals, there is already a weight $w_{i,t'}$ and separate $z_{i,t}$ for each node *i € n'* identifying the timeslots each node would like to use.The execution phase happens every timeslot. Let the current time be *t'*. If there is no node *i € N'* that requires the current timeslot, then there is nothing to schedule, and the execution phase and update phase are terminated. If not, let the set of con-tending nodes be E. The weights of these nodes $W_{i,t\_}$ are compared, and the node with the highest weight is allocated the current timeslot t .The update phase is for the nodes that contended for times-lot t . A new set of candidate timeslots and weights W_{i,t_+1} for $i \in E$ is computed. The start time is $t^T + 1$ instead of t^Tbecause they will be contending for future timeslots following t.

4. Location Based Protocols

Most of the routing protocols for sensor nodes require location information for sensor nodes. Since addressing like IP-addressing is not employed in WSNs, and the nodes are spatially correlated, routing paths can be maintained easily and efficiently employing location information. MECN, Minimum Energy Communication Network. Each node is expected to know its location (using GPS, deterministic and anchored node deployment etc.). Multi-hop communication is employed in this algorithm without clustering. The node with a packet to send to the sink, decides whether or not to employ multi hop communication by calculating the approximate energy costs from the destinations. The scheme identifies a relay region for each node, to send any node from this region is more efficient in terms of power consumption, and a minimum energy path is formed using local information of each node. Since the protocol is self-reconfiguring, it can dynamically adapt to the node failures or topology changes.

- **SMECN,** Small Minimum Energy Communication Network is an extension to MECN. The minimum energy network is constructed like MECN, but the relay region is smaller (in terms of number of edges). SMECN uses less energy than MECN.
- **GAF,** Geographic Adaptive Fidelity: Although it is designed for mobile networks, it can be applied to stationary networks also. The algorithm based on turning of some nodes without affecting the routing fidelity. The network divided into grids, nodes in the same grid considered equivalent in energy cost. Some of the nodes in the same grid turn power off. Therefore an increase in network lifetime is observed, especially for higher densities. Nodes change their state from sleeping to active in turn so that load is balanced. There are three possible states for nodes: sleeping, active, and discovery (determining the neighbor in the grid).
- **GEAR,** Geographical and Energy-Aware Routing: The main idea is to use geographical information while diffusing the query. Each node keeps an estimated cost of transmissions to the destination through their neighbors. The transmission cost depends on residual energy and distance to destination. There is two phases of the algorithm: forwarding the packet to the target region, forwarding the packet within the target region.

  Maximum Lifetime Energy Routing: The main aim of the protocol is to maximize the network lifetime. There are two different algorithms defining the link costs differently.

  cij=1/(Ei-eij)

  cij= eij/Ei

  Ei is the residual energy at node i, eij energy consumed when a packet transferred over the link, cij link costs.
- **Maximum Lifetime Data Gathering:** The lifetime "T" of the system defined as the number of the rounds or periodic data readings from the sensors until the first sensor dies. There are many algorithms proposed based on maximum life time concept. An algorithm called Maximum Lifetime Data Aggregation (MLDA) is proposed. The algorithm considers data aggregation while setting up maximum lifetime routes. In this case, if a schedule "S" with "T" rounds is considered, it induces a flow network G. The flow network with maximum lifetime subject to the energy constraints of sensor nodes is called an optimal admissible flow network. Then, a schedule is constructed by using this admissible flow network.
- **Minimum Cost Forwarding**
- The aim is to find the path with minimum cost in the network. The cost function for the protocol captures the effect of delay, throughput and energy consumption from any node to the sink. There are two phases of the protocol. The set-up phase is every node calculates its cost of transmission to the sink by adding up cost of the link to the cost of the neighbor node (minimum of it). In the second phase, the source broadcasts the data to its neighbors. The nodes receiving the broadcast message, adds its transmission cost (to sink) to the cost of the packet. Then the node checks the remaining cost in the packet. If the remaining cost of the packet is not sufficient to reach the sink, the packet is dropped.
- **SPEED:** Each node maintains its neighbors' information, and routes the packets using geographical information. The protocol requires calculating the estimated speed of the links and end-to-end delays. The main consideration of the algorithm is the end-to-end delay (not the power consumed). Moreover, it provides congestion control.

  The key challenge in wireless sensor network protocol designs is to provide energy efficient communication, since most of the nodes in sensor networks have limited battery power and it is not feasible to recharge or replace the batteries. There are several levels of power consumption in sensor networks such as:
    - Idle Listening: The major power consumption source for WSNs,
    - Retransmissions resulting from collisions,
    - Control packet overhead,
    - Unnecessarily high transmitting power,
    - Sub-optimal utilization of the available resources

Sensor nodes are deployed in an ad hoc fashion, with individual nodes remaining largely inactive for long periods of time. In order to minimize power consumed during idle listening, some nodes, which can be considered redundant, can be put to sleep. Therefore the energy of the nodes and the energy of the network are conserved. The idea is sensor nodes dynamically create on-off schedules such that the nodes will be awake only when they are needed. This also limits the collisions, therefore the energy consumed during retransmissions. The key design considerations for WSN multicasting design are scheduling and routing.

5. Scheduling

Lifetime Prediction Routing (LPR) is an on demand source routing protocol that uses battery lifetime prediction. The objective of this routing protocol is to extend the service life of

WSN with dynamic topology. This protocol favors the path whose lifetime is maximum. We represent our objective function as follow:

$Max\ T\pi\ (t) = Min\ (Ti\ (t))$

T_π (t) : lifetime of path π

Ti (t) : predicted lifetime of node i in path π

The cross-layer scheduling algorithm for power efficiency [2] is proposed in order to conserve energy by turning off some sensor nodes. The idea is sensor nodes dynamically create on-off schedules such that the nodes will be awake only when they are needed. The main constraints considered are latency, capacity (i.e. ability to carry a certain load), employing no global time-slots or coordination with neighbor.

If the estimated activity is too low to satisfy the delay constraint, the node decides to wake up more often. Conversely, if the activity is higher than necessary, the node decides to sleep longer.

5.1. Lifetime Prediction

Each node tries to estimate its battery lifetime based on its past activity. This is achieved using a predictor by keeping track of the last *N* values of residual energy and the corresponding time instances for the last N packets received/relayed by each mobile node. This information is recorded and stored in each node. We have carefully compared the predicted lifetimes to the actual lifetimes for different values of *N* and found *N=10* to be a good value.

The evaluation of the proposed method is conducted by the use of stability-based greedy routing algorithm, which selects the next hop node having the highest link lifetime. In the prediction process, some link-state parameters should be defined: link state indicator, threshold of link connection, cost of selecting a link, cost of selecting a path and time needed for transmitting a packet. There are two types of lifetimes relevant to the lifetime of the network: node battery lifetime and link lifetime. The lifetime of a link is a result of a combination of (relative) velocity and intersections. Our motivation in using lifetime prediction is that mobility introduces different dynamics into the network. In the lifetime of a node is a function of residual energy in the node and energy to transmit a bit from the node to its neighbors. This metric works well for static networks for which it was proposed. However, it is very difficult to efficiently and reliably compute this metric when we have mobility since the location of the nodes and their neighbors constantly change. PSR does not use prediction and only uses the remaining battery capacity. We believe LPR is superior to PSR since LPR not only captures the remaining (residual) battery capacity but also accounts for the rate of energy discharge. This makes the cost function of LPR more accurate as opposed to just using battery capacity. Since mobility can change the traffic patterns through the node, which thereby affects the rate of depletion of its battery. Also, recent history is a good indicator of the traffic through the node and hence we chose to employ lifetime prediction.

Our approach is a dynamic distributed load balancing approach that avoids power-congested nodes and chooses paths that are lightly loaded. This helps LPR achieve minimum variance in energy levels of different nodes in the network. As an example, consider the scenario shown in figure. Here, node F has two flows going through it (D — F—, B — F — and C — F—). Now, if A wants to transmit data to E, the shortest path routing will use A— F — E. However, LPR will use A — B— C— D — E since E is very power-congested (as a result of relaying multiple flows) and the path passing through F will not be selected by LPR. Our duty cycle control algorithm brings in an overhead of time dissipation, and packet traffic, and power dissipation caused by scheduling the sensor on-duty time. Therefore the data transfer phase should be long enough compared with self scheduling phase in order to overcome this overhead caused by scheduling time. The actual threshold levels of data transfer time beyond which our scheduling becomes beneficial may be further studied for precise recommendations. The above scheduling schemes are compared with our proposed link life time algorithm and the simulation results are given below.

5.2. Energy Efficient Broadcasting

Given a sequence of broadcast operations, they tried to increase the number of successful communications before the first communication fails. For this purpose, they proposed an $O\ (m\log m)$ algorithm to construct a broadcast tree that maximizes the critical energy of the network following a broadcast operation, where m denotes the number of links in the network. The critical energy of a broadcast tree T is the minimum of the remaining battery power of all the nodes in T followed by a broadcast operation. In T, the residual energy of node i is $re(i, T) = ce(i) - max\{w(i, j)|j$ is a child of i in $T\}$, where $ce(i)$ is the current energy of i before sending a message and $w(i, j)$ is the energy expended for transmitting a message from i to j. The critical energy $CE\ (T)$ following a broadcast operation is $CE\ (T) = min\ \{re(i, T)|1 \leq i \leq n\}$. The *Maximum Critical Energy Problem (MCEP)* finds a broadcast tree T rooted at s such that $CE\ (T)$ is maximum. This maximum value of $CE\ (T)$ is called the maximum critical energy and is denoted $MCE\ (G,s)$. This algorithm first constructs a sorted list L of all possible residual energy values. For each node i of G, the set $a(i)$ of residual energy values is defined as $a(i) = \{ce(i) - w(i, j)|(i, j)$ is an edge of G and $ce(i) \geq\ w(i, j)$. Set $l(i)$ denotes the set of all possible values for residual energy of i following a broadcast operation.

$$l(i) = \begin{cases} a(i) & \text{if } i=8 \\ a(i) \cup \{c(i)\} & \text{otherwise} \end{cases}$$

When a node *u* receives the packet for the first time from a node *v*, two cases can occur:

- The packet contains some instructions for *u*. It starts constructing a BIP tree within its own 2-hop neighborhood. But instead of starting with an empty tree, it uses the information contained in the packet that is with the neighbors assigned to it by *v* and with its transmission range also fixed by *v*. In this way, nodes located exactly at 2 hops from *u* and 3 hops from *v* will be added to the tree.
- There is no instruction for *u*. In this case, *u* will not rebroadcast the packet.

5.3. Mobility-Based Schemes

In this, sensor nodes are made mobile in nature. But when mobility is added to the sensor nodes, the resulting energy consumption is much higher than the energy gain to mobility itself. There are two ways of making it convenient. One is, instead of making each sensor node mobile, mobility should be limited to some special nodes which consume less energy. Second is, instead of setting up the mobilizers, sensors can be placed on objects which are mobile at their own.

5.4. Multicasting Schemes

5.4.1. MEMTCS

In duty cycled WSN, the nodes switch between active and sleep states. Each node is capable of determining its active/sleep schedule independently. This complicates the Minimum-Energy Multicasting problem that was studied in [6]-[10]. Hence, MEMTCS problem was formulated. A WSN is shown by an undirected graph G= (V, E), [14] where V is set of nodes and E is set of links between the wireless nodes. An assumption is made that all the nodes have same fixed transmission power and a link exists between two nodes only if they are within the transmission range of one another. Another assumption is that every node has unique ID and it knows the IDs of its one-hop neighbors. Time is divided into equal-length slots. It is also assumed that a node can wake up its transceiver to transmit data packet at any time slot but it can receive any data packet only when it is in active state.

5.4.2. MEMBCS

Just as the MEMTCS problem, the MEMBCS problem is also NP-hard. The NP-hardness of the MEMBCS problem can be proved by using a reduction from the Maximum Leaf Spanning Tree (MLST) problem [14] .The MEMBCS problem is NP-hard. Next, we provide an approximation algorithm for the MEMBCS problem [10]. Note that the transmission schedule of any internal node in is actually obtained by using the mapping method.

6. Simulation and Numerical Results

With our duty cycle control algorithm, the sensor network will survive longer, in which power consumption is one of the main constraints. This, however, is achieved at the cost of reduced area coverage, as expected. Redundant nodes at the edges cannot be put to sleep with the duty cycle control algorithm. One of the shortcomings of our algorithm is that if the network is not densely deployed, our algorithm does not put any node to sleep. In essence, while the algorithm runs, some nodes do not have any node to select as next head node; this brings to an end to the algorithm. For small node densities, no node can be put to sleep. For this reason, for small node densities in which the network has large number of blind points, it is not healthy. Each node is connected if and only if the distance between them is equal or smaller than the sensing and communication range, R. Each node considers the connected nodes as its neighbor nodes. The network consists of a sink and randomly deployed nodes. The dimension of one edge is L. Since the sink aggregates data in order to send to a distant station, the events gathered by nodes will be sent to the sink which is at the one corner of the deployment area. The location of the sink is considered as known by deployed nodes. Each node knows its location coordinates and the dimensions of the deployment region using low-cost, low-power GPS or other localization algorithms. Therefore, each node is connected if and only if the distance between them is equal or smaller than the sensing and communication range, R. Each node considers the connected nodes as its neighbor nodes.

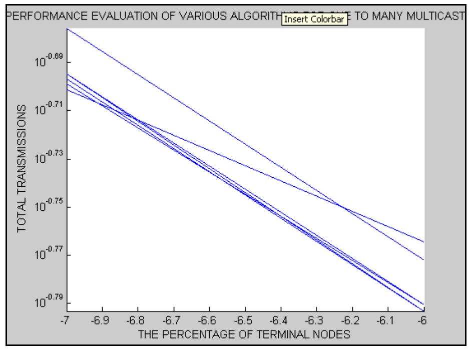

Figure 4: performance of various algorithm in a one to many routing

The link life time scheduling provides the higher throughput compared with the other three scheduling schemes.

- **Throughput Comparison**

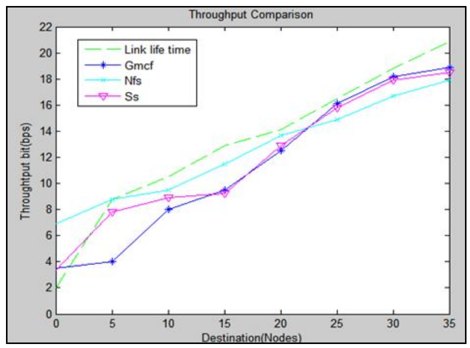

Figure 5: Through put of link life time with scheduling schemes- Gmcf,Nfs,Ss

- **Time Consumption of nodes**

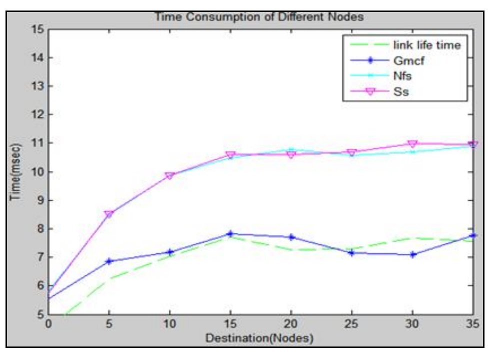

Figure 6: Time consumption of LLP scheduling scheme is low compared to Gmcf,Nfs,Ss.

- **Data transfer**

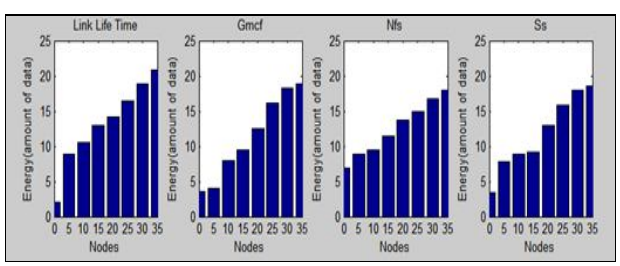

Figure 7: Data transfer rate is better for LLP

7. References

1. I.Akyildiz,W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," IEEE Communications Magazine, Volume: 40 Issue: 8, pp.102-114, August 2002.
2. ML Sichitiu, “Cross-layer scheduling for power efficiency in wireless sensor networks,” in INFOCOM, 2004, pp. 1740–. 1750.
3. A. Farbod and T. Todd, “Resource allocation and outage control for solar powered WLAN mesh networks,” IEEE Trans. Mobile Comput., vol. 6, no. 8, pp. 960–970, Aug. 2007.
4. F. Li and Y. Wang, “Routing in vehicular ad hoc networks: A survey,” IEEE Veh. Technol. Mag., vol. 2, no. 2, pp. 12–22, Jun. 2007.
5. D. Wang, M. Hu, and H. Zhi, “A survey of secure routing in ad hoc networks,” The Ninth International Conference on Web-Age Information, pp. 482–486, 2008.
6. J. Wieselthier, G. Nguyen, A. Ephremides, “On the construction of energy-efficient broadcast and multicast trees in wireless networks,” in Proceedings of the IEEE INFOCOM, 2000, pp. 585–594.
7. P.-J. Wan, G. Calinescu, C.-W. Yi, “Minimum-power multicast routing in static ad hoc wireless networks,” IEEE/ACM Trans. Netw., vol. 12, no. 3, pp. 507–514, 2004.
8. W. Liang, “Approximate minimum-energy multicasting in wireless ad hoc networks,” IEEE Trans. Mobile Comput., vol. 5, no. 4, pp. 377–387, 2006.
9. D. Li, Q. Liu, X. Hu, and X. Jia, “Energy efficient multicast routing in ad hoc wireless networks,” Computer Communications, vol. 30, no. 18, pp. 3746–3756, 2007.
10. W. Liang, R. Brent, Y. Xu, and Q. Wang, “Minimum-energy all-to all multicasting in wireless ad hoc networks,” IEEE Trans. Wireless Commun., vol. 8, no. 11, pp. 5490–5499, 2009.
11. T. Karygiannis and L. Owens, “Wireless network security 802.11, Bluetooth and handheld devices,” 2002.
12. S. Mueller, R. P. Tsang, and D. Ghosal, “Multipath routing in mobile ad hoc networks: Issues and challenges,” Performance Tools and Applications to Networked Systems, pp. 209–234, 2004.
13. Stojmenovic and X. Lin, “Power-aware localized routing in wireless networks.” Proc. IEEE IPDPS, Cancun, Mexico, May 2000.
14. Duty-Cycle-Aware Minimum-Energy Multicasting in Wireless Sensor Networks ,Kai Han, Yang Liu, and Jun Luo, Member, IEEE, ACM