# Design Optimization of Energy and Delay Efficient Wireless Sensor Network with the help of Particle Swarm Optimization and Cluster based Least Spanning Tree Algorithm

Satyesh Sharan Singh[1], Mukesh Kumar[2], A.K. Jaiswal[3], Rohini Saxena[4], Priya[5]

*[1,5]PG Student ECE, SHIATS (Deemed-to-be-university), U.P., India*
*[3] Professor & HOD, ECE, SHIATS (Deemed-to-be-university), U.P., India*
*[2,4]Assistant professor, ECE, SHIATS (Deemed-to-be-university), U.P., India*

***Abstract:*** *In WSN sensors are randomly deployed in the sensor field which brings the coverage problem and limited energy resources. Hence energy and coverage problem are very scarce resources for such sensor systems and has to be managed wisely in order to extend the life of the sensors and maximizing coverage for the duration of a particular mission. In past a lot of cluster based algorithm and techniques were used. In this paper we propose combination of PSO based algorithm and cluster based Least Spanning Tree algorithm, which are very effective alone for WSN, and we also obtain life of sensor node and data transmission by LST based PSO algorithm. These techniques effectively overcome the problems of low energy and coverage of sensor range.*
***Keywords***: *Energy efficient clustering, Least Spanning Tree algorithm, PSO algorithm, Wireless Sensor Networks.*

## I. Introduction

Advances in sensor technology, low-power electronics, and low-power radio frequency (RF) design have enabled the development of small, relatively inexpensive and low-power sensors, called micro sensors [4] which can be connected via a wireless network. These wireless micro sensor networks represent a new paradigm for extracting data from the environment and enable the reliable monitoring of a variety of environments for applications that include surveillance, machine failure diagnosis, and chemical/biological detection. An important challenge in the design of these networks is that two key resources (communication bandwidth and energy) are significantly more limited than in a tethered network environment. These constraints require innovative design techniques to use the available bandwidth and energy efficiently. Energy usage is an important issue in the design of WSNs which typically depends on portable energy sources like batteries for power .WSNs is large scale networks of small embedded devices, each with sensing, computation and communication capabilities. They have been widely discussed in recent years. Micro-Electro-Mechanical System [MEMS][3] sensor technology has facilitated the development of smart sensors, these smart sensors nodes are small devices with limited power, processing and computation resources. Smart sensors are power constrained devices that have one or more sensors, memory unit, processor, power supply and actuator. In WSNs, sensor nodes have constrained in term of processing power, communication bandwidth, and storage space which required very efficient resource utilization. In WSNs the sensor nodes are often grouped into individual disjoint sets called a cluster, clustering is used in WSNs, as it provides network scalability, resource sharing and efficient use of constrained resources that gives network topology stability and energy saving attributes. Clustering schemes offer reduced communication overheads, and efficient resource allocations thus decreasing the overall energy consumption and reducing the interferences among sensor nodes. A large number of clusters will congest the area with small size clusters and a very small number of clusters will exhaust the cluster head with large amount of messages transmitted from cluster members.

## II. Background

There are various algorithms which are used for the energy efficiency of wireless sensor network which are improved time to time by different authors; some important and useful algorithms are shown in figure 1.1.
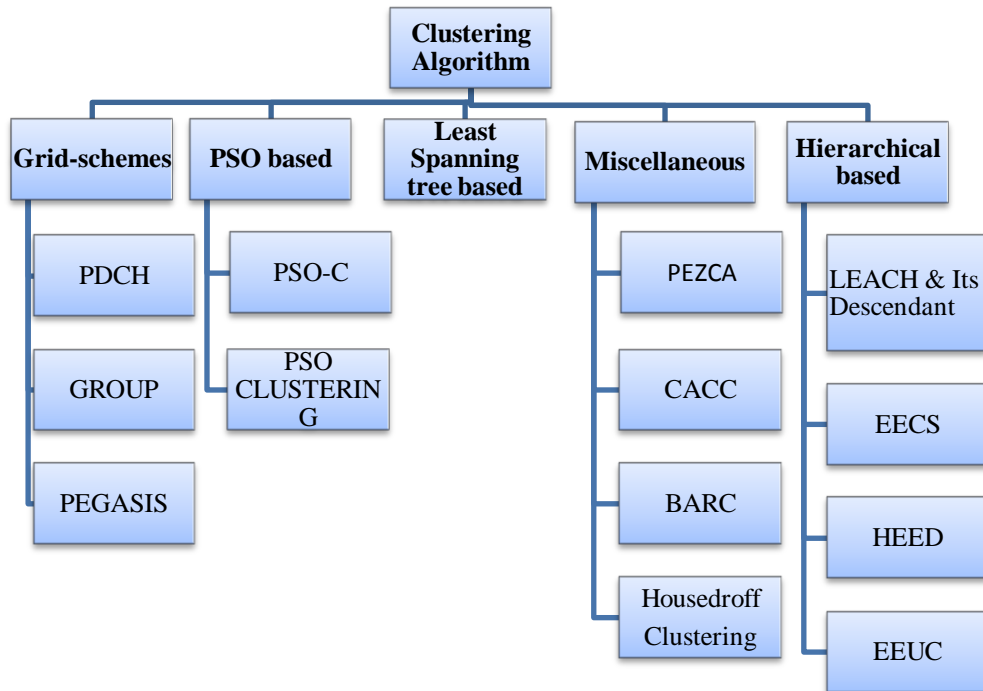
**Figure.1.1** Different types of algorithm

## III.       The System Model
For making of an effective system we make a model which contains all prospects of the system and also it executable. So the system model of an energy and delay efficient wireless sensor network is given bellow:

### A.   Network Model
We adopt a wireless sensor architecture or model which is similar to use in the LEACH protocol with the following properties:
➢     Each node performs sensing tasks periodically and always has tendency to send data to the base station.
➢     A fixed base station can be located inside or outside the network sensor fields.
➢     All nodes are stationary and energy constrained.
➢     The nodes have power control capabilities to vary their transmitted power.
➢     All nodes are capable of operating in cluster head mode and sensing mode.
➢     Data fusion is used to reduce the total data message sent.

### B.   Radio Power Model and Characteristic Distance
For a simplified power model of radio communication, the energy consumed per second in Transmission is[6]:
$$E_t = (e_t + e_d r^n)B \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots.\ldots\ldots.(1)$$
Where $e_t$ is the energy/bit consumed by the transmitter electronics (including energy costs of imperfect duty cycling due to finite startup time), and $e_d$ accounts for energy dissipated in the transmit op-amp (including op-amp inefficiencies). Both $e_t$ and $e_d$ are properties of the transceiver used by the nodes, $r$ is the transmission range used. The parameter $n$ is the power index for the channel path loss of the antenna. This factor depends on the RF environment and is generally between 2 and 4. $B$ is the bit rate of the radio and is a fixed parameter in our study.
On the receiving side, a fixed amount of power is required to capture the incoming radio signal[6]:
$$E_r = e_r B \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots.(2)$$
For currently available radio transceivers are $e_t$ =50x10-9 J/bit, $e_r$ =50x10-9 J/bit, $e_d$ =100x10-12 J/bit/m2 (for n=2) and $B$=1Mbit/s. Since the path loss of radio transmission scales with distance in a greater-than-linear fashion, the transmission energy can be reduced by dividing a long path into several shorter ones. However, if the number of intermediate nodes is very large then the energy consumption per node is dominated by the term $e_t$ in equation (1) and the receiving energy consumption hence an optimum exists. Intermediate nodes between a data source and destination can serve as relays that receive and rebroadcast data. Let us consider multi-hop communication in a finite one dimensional network from the source to the base station

across a distance $d$ using $k$ hops. The source at $x=d$ will generate traffic of $A$ Erlang, so that each intermediate node receives and transmits the same traffic, $A$. The routing nodes are assumed to be regularly spaced and to consume no energy while idle. The power consumed by this communication is then simply the sum of the transmit and receive energies multiplied by the effective bit rate, $BA$, and is given by

$$P = \sum_{i=1}^{k} (e_t + e_d r_i^n + e_r) BA \dots \dots \dots \dots \dots \dots \dots \dots \dots (3)$$

$$d = \sum_{i=1}^{k} r_i \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots (4)$$

In order to minimize $P$ we note that it is strictly convex and use Jensen's inequality. Given $d$ and $k$ then $P$ is minimized when all the hop distances $r_i$ are made equal to $d/k$. The minimum energy consumption for a given distance $d$ has either no intervening hops or $k_{opt}$ equidistant hops where $k_{opt}$ is always of,

$$k_{opt} = \left( \frac{d}{d_{char}} \right) \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots (5)$$

The distance $d_{char}$, called the characteristic distance, is independent of $d$ and is given by,

$$d_{char} = \sqrt[n]{\frac{e_t + e_r}{e_d (n-1)}} \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots (6)$$

The characteristic distance depends only on the energy consumption of the hardware and the path loss coefficient (i.e. it is independent of the traffic); $d_{char}$ alone determines the optimal number of hops. For typical COTS (commercial, off-the-shelf)-based sensor nodes, $d_{char}$ is about 35 meters. The introductions of relay nodes are clearly a balancing act between reduced transmission energy and increased receive energy. Hops that are too short lead to excessive receive energy. Hops that are too long lead to excessive path loss. In between these extremes is an optimum transmission distance that is the characteristic distance.

In the above section I have introduced a simple energy model in which no energy was consumed while the node was idle. This led to a characteristic distance that was independent of traffic. We now include the idle state energy and show how the characteristic distance is modified. On one hand a short range is preferred for energy efficient data transmission as a result of the nonlinear path loss ratio. On the other hand more redundant nodes can be put into the sleep state to prolong the network lifetime if a long range is used in the topology management of sensor networks. If the transmission route is divided into $k$ grids and only one node wakes up in each grid as relay node, as in the GAF protocol, the total energy consumption per second by $k$ hops is[7]

$$P = k\left(e_r BA + e_t BA + e_d \left(2\frac{d}{k}\right)^n BA + ce_r (1 - 2A)B \dots \dots \dots (7)\right.$$

The last term $ce_r (1 - 2A)B$ in the equation (7) represents the energy consumption when the radio neither receives nor transmits, i.e. it is in the idle state. The energy consumption in the idle state is approximately equal to that in the receiving state, so that the parameter $c$ is close to 1. Note that we are currently assuming that nodes in the sleep state consume no energy. Also, we assume that the routing node in each grid can be located anywhere within that section and so the radio range is now twice the grid size. The energy efficient optimum size of the virtual grid can now be derived from equation (1.7) and is given by

$$R_{opt} = {r_{opt}}/2 = \sqrt[n]{\frac{(e_t + e_r)A + ce_r(1 - 2A)}{2^n A(n-1)e_d}} \dots \dots \dots (8)$$

We know the energy consumed at the cluster head is much larger than that at individual sensing node. The reason is as follows: (1) the cluster-head needs to relay all the traffic of the cluster; (2) for each data unit, the cluster-head needs to transmit longer distance due to transmission between clusters, while the sensing nodes just transmit data inside the cluster. In view of this, let $E_p$ and $E_c$ be, respectively, the current energy and clustering energy ($E_c$ is fixed), after a period of time, the $i^{th}$ cluster-head has transmits information $n_1$ times and has receives information $n2$ times before $T_1$(suppose the energy of the $i^{th}$ cluster-head is not lower than the threshold and the information unit is A Erlang ) .the remainder energy of the $i^{th}$ cluster-head at $T_1$ is then simply as the equation (9).

$$E_p(i) = E_p(i) - \sum_{k=1}^{n_1} E_r(i,k) - \sum_{i=1}^{n_2} E_r(i,l) - E_c \times n_i \dots (9)$$

Where i, j, k respectively denote cluster-head, $n_i$ is the clustering time in cluster i before $T_1$.

## IV. Methodology

### A. Particle Swarm Optimization[PSO]

PSO is originally attributed to Kennedy, Eberhart and Shi[1] and was first intended for simulating social behavior, as a stylized representation of the movement of organisms in a bird flock or fish school. The algorithm was simplified and it was observed to be performing optimization. The book by Kennedy and Eberhart describes many philosophical aspects of PSO and swarm intelligence. An extensive survey of PSO applications is made by Poli. In computer science, particle swarm optimization (PSO)[6] is a computational method that optimizes a problem by iteratively trying to improve a candidate solution with regard to a given measure of quality. PSO optimizes a problem by having a population of candidate solutions, here dubbed particles, and moving these particles around in the search-space according to simple mathematical formulae over the particle's position and velocity. Each particle's movement is influenced by its local best known position and is also guided toward the best known positions in the search-space, which are updated as better positions are found by other particles. This is expected to move the swarm toward the best solutions.

The PSO [6] algorithm is an evolutionary computing technique, modeled after the social behaviour of a flock of birds. In the context of PSO, a swarm refers to a number of potential solutions to the optimization problem, where each potential solution is referred to as a particle. The aim of the PSO is to find the particle position that results in the best evaluation of a given fitness function. In the initialization process of PSO, each particle is given initial parameters randomly and is 'flown' through the multi-dimensional search space. During each generation, each particle uses the information about its previous best individual position and global best position to maximize the probability of moving towards a better solution space that will result in a better fitness. When a fitness better than the individual best fitness is found, it will be used to replace the individual best fitness and update its candidate solution according to the following equations [1]:

$$v_{id}(t) = w \times v_{id}(t-1) + c_1 \emptyset_1 (p_{id} - x_{id}(t-1)) + c_2 \emptyset_2 (p_{gd} - x_{gd}(t-1)) \dots\dots\dots (10)$$

Table 1 summarizes the variables in (1).

Table 1: List of variables used in PSO equations:-

| | |
|---|---|
| v | The particle velocity |
| x | The particle position |
| t | Time |
| C1,C2 | Learning factor |
| $\Phi1,\Phi2$ | Random numbers between 0 and 1 |
| $p$id | Particle's best position |
| $p$gd | Global best position |
| w | Inertia weight |

### B. Particle Swarm Optimization Algorithm for Cluster Setup

The operation of our protocol is based on a centralized control algorithm that is implemented at the base station, which is a node with a large amount of energy supply. The proposed protocol operates in rounds, where each round begins with a setup phase at which clusters formed. This is followed by a steady state phase in which we used a similar approach as in. At the starting of each setup phase, all nodes send information about their current energy status and locations to the base station. Based on this information, the base station computes the average energy level of all nodes. To ensure that only nodes with a sufficient energy are selected as cluster heads, the nodes with an energy level above the average are eligible to be a cluster head candidate for this round. Next, the base station runs the PSO algorithm to determine the best $K$ cluster heads that can minimize the cost function, as defined by:

$$\cos t = \beta \times f_1 + (1 - \beta) \times f_2 \dots\dots\dots\dots\dots\dots (11)$$

$$f_1 = max_{k=1,2,..,K} \left\{ \sum_{\forall n_i \in C_{p,k}} d(n_i, CH_{p,k}) / |C_{p,k}| \right\} \dots\dots\dots\dots\dots\dots (12)$$
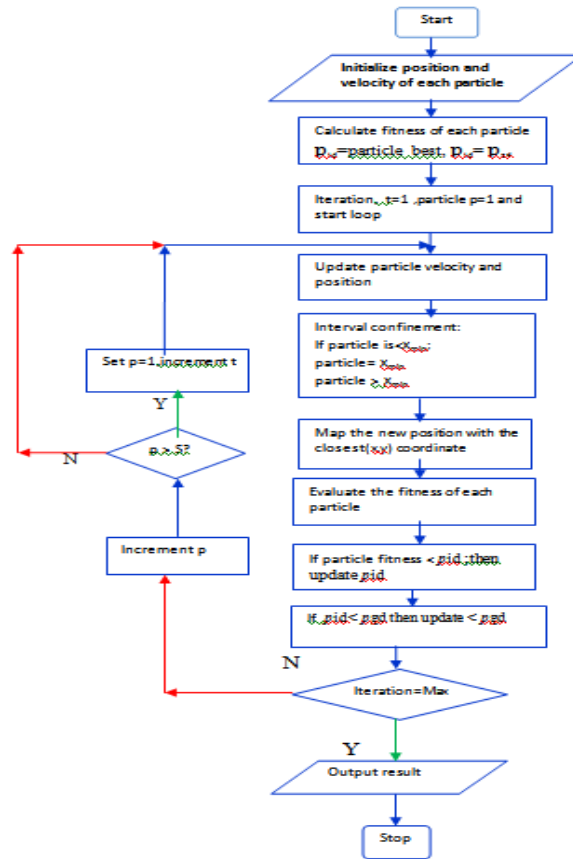
**Figure 1.2** Flowchart of the PSO algorithm for cluster setup

$$f_2 = \sum_{i=1}^{N} E(n_i) \ / \sum_{k=1}^{K} (CH_{p,k}) \ \dots\dots\dots\dots\dots\dots\dots (13)$$

Where *f*1 is the maximum average Euclidean distance of nodes to their associated cluster heads and $C_{p,k}$ is the number of nodes that belong to cluster $C_k$ of particle p. Function $f_2$ is the ratio of total initial energy of all nodes $n_i$, *i*=1,2,…,*N* in the network with the total current energy of the cluster heads candidates in the current round. The constant β is a user defined constant used to weigh the contribution of each of the sub-objectives. The fitness function defined above has the objective of simultaneously minimizing the intra-cluster distance between nodes and their cluster heads, as quantified by $f_1$; and also of optimizing the energy efficiency of the network as quantified by $f_2$. According to the cost function defined above, a small value of $f_1$ and $f_2$ suggests compact clusters with the optimum set of nodes that have sufficient energy to perform the cluster head tasks. Fig. 1.2[1] shows the flowchart of PSO algorithm applied during the cluster setup phase. For a sensor network with *N* nodes and *K* predetermined number of clusters, the network can be clustered as follows:
1. Initialize *S* particles to contain *K* randomly selected cluster heads among the eligible cluster head candidates.
2. Evaluate the cost function of each particle:
 i. For each node *ni*, *i* = 1,2,…,*N*
   -Calculate distance *d(ni,CHp,k)* between node *ni* and all cluster heads *CHp,k*.
    - Assign node *ni* to cluster head *CHp,k* where;
 ii. Calculate the cost function using equations (2)to(3).
3. Find the personal and global best for each particle.
4. Update the particle's velocity and position using (1).
5. Limit the change in the particle's position value.
6. Map the new updated position with the closest (*x,y*) coordinates.
7. Repeat steps 2 to 6 until the maximum number of iterations is reached.

**C.     Least spanning tree[2]**
    The minimum spanning tree is a tree of a planar graph. Each edge is labelled with its weight, which here is roughly proportional to its length. Given a connected, undirected graph, a spanning tree of that graph is a sub graph that is a tree and connects all the vertices together. A single graph can have many different spanning

trees. We can also assign a weight to each edge, which is a number representing how unfavourable it is, and use this to assign a weight to a spanning tree by computing the sum of the weights of the edges in that spanning tree. A least spanning tree (LST) or minimum weight spanning tree is then a spanning tree with weight less than or equal to the weight of every other spanning tree. More generally, any undirected graph (not necessarily connected) has a minimum spanning forest, which is a union of minimum spanning trees for its connected components. A spanning tree for that graph would be a subset of those paths that has no cycles but still connects to every house. There might be several spanning trees possible. A minimum spanning tree would be one with the lowest total cost

**D.    Constructing least spanning tree[2]**

By Eq(9),we can compute the remainder energy, if $E_p(i)$ is lower than $E_v$, then modify the information table of the $i^{th}$ cluster-head: set flag=0; broadcast information to its children, also inform the neighbour $j^{th}$ cluster-head doesn't transmits information to it, and let $i^{th}$ cluster-head i is in the sleeping state; else the $i^{th}$ cluster-head may go along the next clustering or transmitting or receiving. According to Prim algorithm, suppose undirected graph G (V, E, D), where V is the set of cluster-heads and the number is N, E is the set of connections of cluster-head, and D is the distance of cluster-heads, the process of constructing least spanning tree as illustrated below:

• Initialization: $V_1$=Sink, E′=null, and $V_2$=V-$V_1$.

• Select a edge: which has minimum distance from Sink to one cluster-head (suppose is $V_i$), where $V_i$ is directly connected with Sink, then set, $V_1$= {Sink, $V_i$}, E ′= {(Sink, $V_i$)}, $V_2$=$V_2$-$V_1$.

• For each cluster-head $V_k$ in $V_1$ do :select a minimum distance d(k,j),which $V_k$ ∈$V_1$,$V_j$ ∈ $V_2$ and E′=($V_k$, $V_j$) ∈ E , but $V_k$ is not ∈ E′, then $V_1$=$V_1$ ∈ $V_j$, E′={($V_k$, $V_j$)} ∈ E′, $V_2$=$V_2$-$V_j$.

• If $V_2$ is empty then end, else go to above.

The deployment of node by least spanning tree is shown in the figure 1.3(a & b).In this deployment we take one cluster which is nearer to the sink. Now by MAC protocol we find out the weightege of different edge(path distance). And make an incident matrix which shows the direction of data flow from node to node.
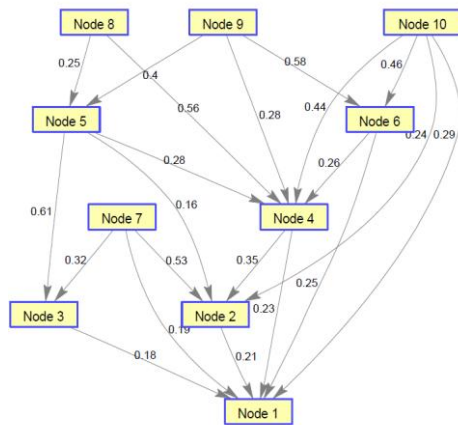


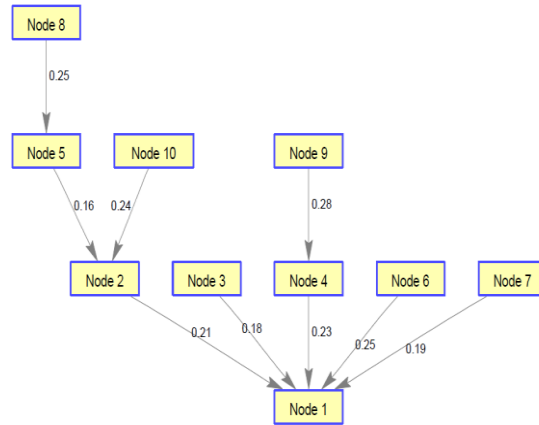**Figure 1.3(a)** deployment of sensor node                      **Figure 1.3 (b).** Least spanning tree graph.

Now we apply least spanning tree algorithm and we get shortest path from each node to the node 'A', which is the nearest cluster head of the network from the sink. The least spanning tree graph is shown in the figure 1.3 (b).

# V. Results

The performance of the proposed protocol is evaluated using Matlab. We ran the simulations for 100
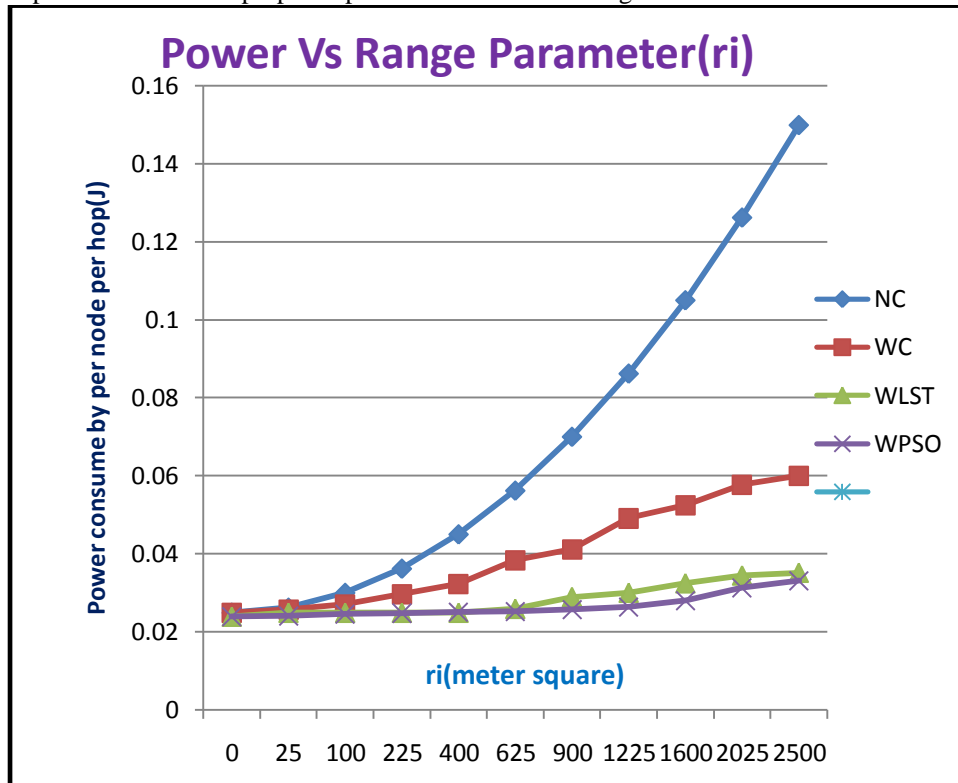


**Figure1.4**: Power consume by per node per hop over range parameter

nodes in a 100m x 100m network area with unequal initial energy of nodes to show the effect of the different nodes' energy in the network. In figure 1.4 shows the calculated value of energy consumed by per node per hop over the distance from the node. So we find out that with PSO clustering the power consumption is very less. In figure 1.5 the lifetime of the node over high traffic is high in the PSO algorithm. in our proposed model it gives approximately 4000 seconds per node over full traffic
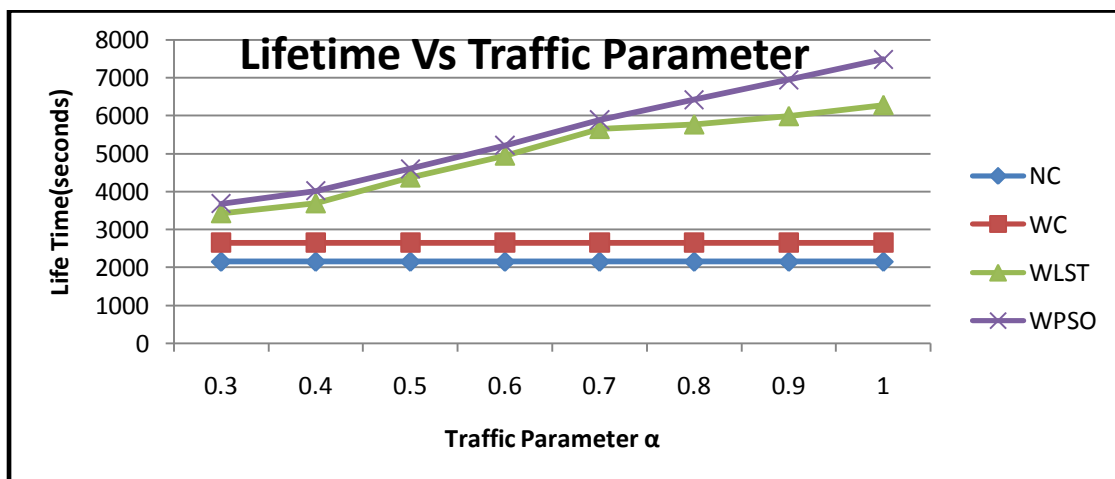


**Figure 1.5** Lifetime of node over traffic parameter

When we apply PSO algorithm for clustering and least spanning tree for data transmission then we get many nodes live for long time when full traffic apply as shown in the figure 1.6.
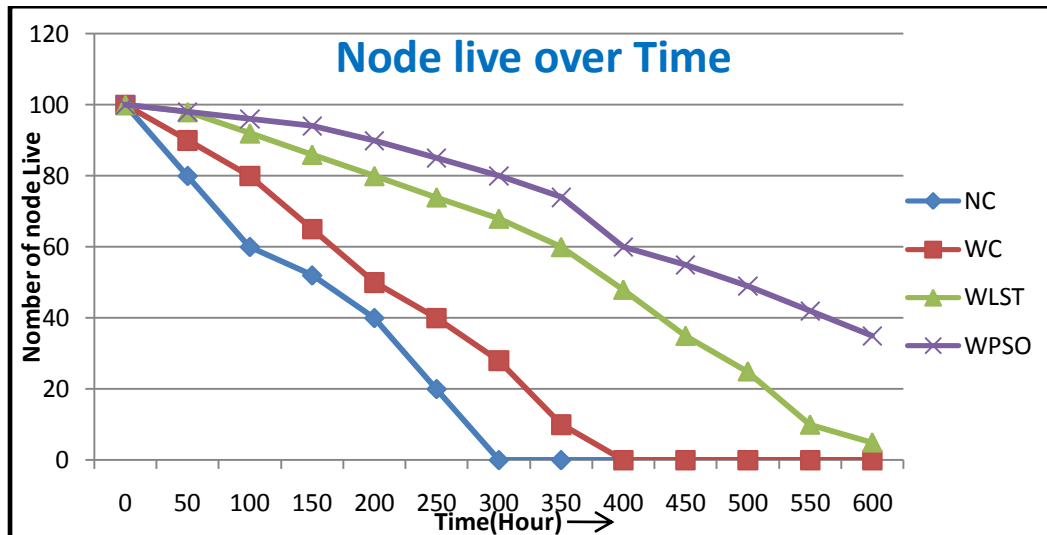
**Figure 1.6** Number of node over time

## VI.        Conclusion

In this paper we have presented energy and delay-aware cluster based protocol for wireless sensor networks using particle swarm optimization (PSO) algorithm and least spanning tree. Results from the simulations and calculation indicate that the proposed protocol using PSO and least spanning tree algorithm give a higher network lifetime and delivers more data to the base station compared to other protocol. Furthermore, the proposed protocol produces better clustering by evenly allocating the cluster heads and the shortest path to the communication throughout the sensor network area. Our future work includes the implementation of secure data transmission by distributed manner in ad- hock  WSN network with the help of PSO. Comparison with other evolutionary optimization algorithm, such as Genetic Algorithm will also be made.

## References

[1].    Priya, Satyesh Sharan Singh, Mukesh Kumar, Rohini Saxena "*Energy And Time Delay Efficient Wireless Sensor Network By Least Spanning Tree Algorithm: A Survey* International Journal Of Engineering Research And Applications (IJERA) ISSN: 2248-9622 Www.Ijera.Com Vol. 3, Issue 1, January -February 2013, Pp.712-719.

[2].    SATYESH SHARAN SINGH* Et Al. "*Application Of Particle Swarm Optimization For Energy Efficient Wireless Sensor Network: A Survey*".[**IJESAT**] International Journal Of Engineering Science & Advanced Technology ISSN: 2250–3676  Volume-2, Issue-5, 1246 – 1250

[3].    J. Kennedy And R. C. Eberhart, "Particle Swarm Optimization, (1995)," *IEEE International Conference On Neural Networks*, Vol.4, Pp. 1942-1948, Perth, Australia.

[4].    W. R Heinzelman, A. P Chandrakasan, And H. Balakrishnan,(2000), "Energyefficient Communication Protocol For Wireless Microsensor Networks," In *Proceedings Of The 33rd Hawaii International Conference On System Sciences*.

[5].    I. F. Akyilldiz, W. Su, Y. Sankarasubramaniam And E. Cayirci, (2002), "A Survey On Sensor Networks," *IEEE Communications Magazine*, Pp. 102−114.

[6].    W. B Heinzelman, A. P Chandrakasan, And H. Balakrishnan,(2002), "An Application-Specific Protocol Architecture For Wireless Microsensor Networks," *IEEE Transactions On Wireless Communications*, Vol. 1, No. 4, Pp. 660-70.

[7].    S. Lindsey, C. Raghavendra, And K. M Sivalingam,(2002), "Data Gathering Algorithms In Sensor Networks Using Energy Metrics," *IEEE Transactions On Parallel And Distributed System*, Vol. 13, No. 9, Pp. 924-935.

[8].    J. Tillet, R. Rao, And F. Sahin,(2002) "Cluster-Head Identification In Ad Hoc Sensor Networks Using Particle Swarm Optimization," *IEEE International Conference On Personal Wireless Communications*, Pp. 201-205.

[9].    S. D. Muruganathan, D. C. F. Ma, R. I. Bhasin, And A. O. Fapojuwo,(2005), "A Centralized Energy-Efficient Routing Protocol For Wireless Sensor Networks," *IEEE Radio Communications*, Pp. S8-S13.

[10].    S. Guru And S. Halgamuge And S. Fernando "Particle Swarm Optimizers For Cluster Formation In Wireless Sensor Networks". In *IEEE International Conference On Sensor*, 319–324.

[11].     N.M.A. Latiff, C.C. Tsimenidis, And B.S Sheriff(2008) "Energy-Aware Clustering For Wireless Sensor Networks Using Particle Swarm Optimization". In *IEEE International Conference On Mobile Radio Communication*, 1–5.

[12].     X. Co, H. Zhang, J. Shi, And G.Cui "Cluster Heads Election Analysis For Multi-Hop Wireless Sensor Networks Based On Weighted Graph And Particle Swarm Optimization". In *IEEE Fourth International Conference On Computing*, 7, 599–603.

[13].     Jason Tillett, Shanchieh Jay Yang, Raghuveer Rao And Ferat" Optimal Topologies For Wireless Sensor Networks"*IEEE International Conference On Personal Wireless Communication.*