International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Impact Factor (2012): 3.358

# Analysis of Clustering Algorithms

**Navneet Kaur[1], Dinesh Kumar[2]**

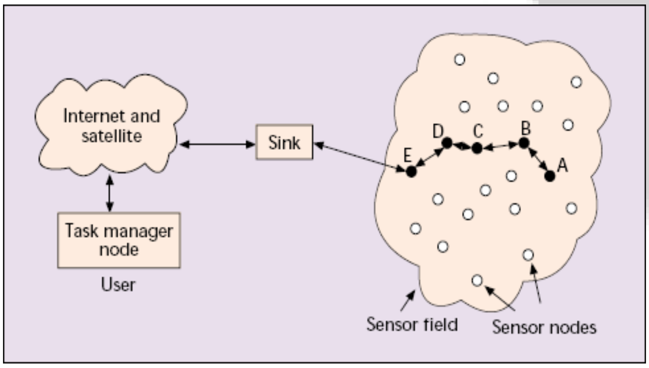
[1, 2] Department of Computer Science, PTU GZS Campus, Bathinda-151001, India

**Abstract:** ***Wireless sensor networks are infrastructure less. It monitors physical or environmental conditions. The network and users interact through base station.WSN consists of thousands of sensor nodes. Sensor nodes communicate among themselves through radio signals. Various routing protocols are used in WSN. Mainly LEACH, HEED, PEGASIS are discussed in this paper .HEED is hybrid energy efficient distributed protocol..It is a clustering protocol. It is used as a primary parameter. Network topology features are used in it .LEACH is low energy adaptive clustering hierarchy. It is the first network protocol that use protocol hierarchical routing. PEGASIS is power efficient gathering in sensor information system. The key idea in PEGASIS is to form and transmit to a closer neighbour .It is sensor information system .Comparison will be done between the algorithmic schemes by considering a particular parameter that is network lifetime.***

**Keywords:** wireless sensor network, clustering, routing protocols, network lifetime

## 1. Introduction

Wireless sensor networks are networks that consists of sensors which are distributed and are intended to monitor and record conditions at diverse location. These sensors work with each other to sense some physical phenomena then the information gather is processed to get result. Wireless sensor networks consist of protocols and algorithms with self organizing capabilities.




WSN is a very large array of diverse sensor nodes that are interconnected by a communication network. The elementary components of a sensor node are sensing unit, a processing unit, a transceiver unit and a power unit. The sensor node senses the physical quantity being measured and coverts it into an electrical signal. Then, the signal is fed to an A/D converter and is ready to be used by the processor. The processor will convert the signal into data depending on how it is programmed and it sends the information to the network by using a transceiver. The sensing data are shared between the sensor nodes and are used as input for a distributed estimation system.

The fundamental objectives for WSN are reliability, accuracy, flexibility, cost effectiveness, and ease of deployment. WSN is made up of individual multifunctional sensor nodes.

As we know that wireless sensor network mainly consists of tiny sensor node which is equipped with a limited power source. The lifespan of an energy-constrained sensor is determined by how fast the sensor consumes energy. A node in the network is no longer useful when its battery dies. Researchers are now developing new routing mechanisms for sensor networks to save energy and pro-long the sensor lifespan. The dynamic clustering protocol allows us to space out the lifespan of the nodes, allowing it to do only the minimum work it needs to transmit data. The WSN can be applied to a wide range of applications, such as environment management, environmental monitoring, industrial sensing, infrastructure protection, battlefield awareness and temperature sensing. So, it is essential to improve the energy efficiency to enhance the quality of application service.

### Applications of WSN

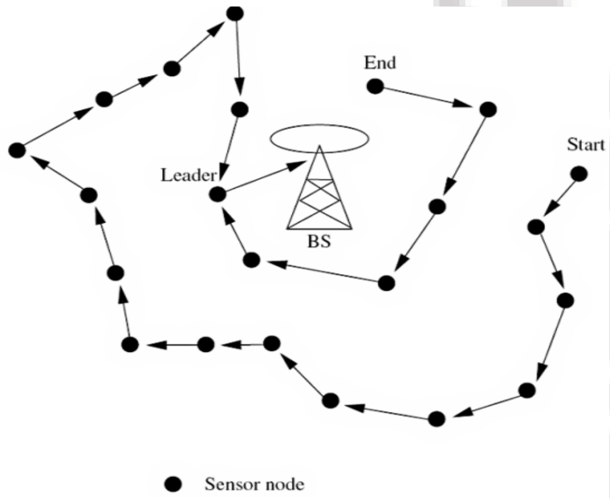
- Military applications
- Area monitoring
- Transportation
- Health applications
- Environmental sensing
- Structural monitoring
- Industrial monitoring
- Agricultural sector

### WSN design factors

- Fault tolerance
- Scalability
- Production cost
- Hardware constraints
- Sensor network topology
- Environment
- Transmission media
- Power consumption

### 1) PEGASIS

PEGASIS is power efficient gathering in sensor information system. **The key idea in PEGASIS is to form a chain among the sensor nodes so that each node will receive from and transmit to a close neighbor.** The protocol, called Power-Efficient Gathering in Sensor Information Systems (PEGASIS), is a near optimal chain-based protocol. This algorithm decreases the energy consumption by creation of a chain structure comprised of all nodes and continually data aggregation across the chain. The algorithm presents the idea that if nodes form a chain from source to sink, only one node in any given transmission time-frame will be transmitting to the base station. Data-fusion occurs at every node in the sensor network allowing for all relevant information to permeate across the network. PEGASIS avoids cluster formation and uses only one node in a chain to transmit to the BS instead of using multiple nodes. In order to increase network life time, nodes need only to communicate with their closest neighbours and they take turns in communicating with the BS. When the round of all nodes communicating with the base-station ends, a new round will start and so on.




### 2) PEGASIS CHAIN

The nodes will be organized to form a chain, which can either be accomplished by the sensor nodes themselves using a greedy algorithm starting from some node. When a node dies, the chain is reconstructed in the same manner to bypass the dead node. The main idea in PEGASIS is for each node to receive from and transmit to close neighbours and take turns being the leader for transmission to the BS. Nodes take turns transmitting to the BS, and we will use node number i mod N (N represents the number of nodes) to transmit to the BS in round i.

The main idea in PEGASIS is for each node to receive from and transmit to close neighbours and take turns being the leader for transmission to the BS. This approach will distribute the energy load evenly among the sensor nodes in the network. We initially place the nodes randomly in the play field, and therefore, the *i* –th node is at a random location. The nodes will be organized to form a chain, which can either be accomplished by the sensor nodes themselves using a greedy algorithm starting from some node. Alternatively, the BS can compute this chain and broadcast it to all the sensor nodes. We placed the BS at a far distance from all other nodes. For a 50m x 50m plot, our BS is located at (25, 150) so that the BS is at least 100m from the closest sensor node. For constructing the chain, we assume that all nodes have global knowledge of the network and employ the greedy algorithm. We could have constructed a loop, however, to ensure that all nodes have close neighbours is difficult as this problem is similar to the travelling salesman problem. The greedy approach to constructing the chain works well and this is done before the first round of communication. To construct the chain, we start with the furthest node from the BS. We begin with this node in order to make sure that nodes farther from the BS have close neighbours, as in the greedy algorithm the neighbour distances will increase gradually since nodes already on the chain cannot be revisited. Figure 2 shows node 0 connecting to node 3, node 3 connecting to node 1, and node 1 connecting to node 2 in that order. When a node dies, the chain is reconstructed in the same manner to bypass the dead node.

Chain construction using the greedy algorithm. For gathering data in each round, each node receives data from one neighbour, fuses with its own data, and transmits to the other neighbour on the chain. Note that node *i* will be in some random position *j* on the chain. Nodes take turns transmitting to the BS, and we will use node number *i* mod N (N represents the number of nodes) to transmit to the BS in round *i*. Thus, the leader in each round of communication will be at a random position on the chain, which is important for nodes to die at random locations. The idea in nodes dieing at random places is to make the sensor network robust to failures. In a given round, we can use a simple control token passing approach initiated by the leader to start the data transmission from the ends of the chain. The cost is very small since the token size is very small. Node c2 is the leader, and it will pass the token along the chain to node c0.Node c0 will pass its data towards node c2. After node c2 receives data from node c1, it will pass the token to node c4, and node c4 will pass its data towards node c2.

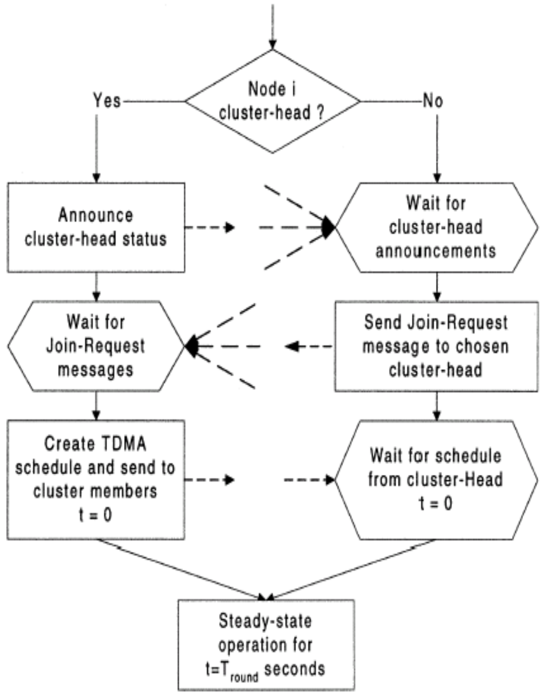
## 2. Token Passing Approach

PEGASIS performs data fusion at every node except the end nodes in the chain. Each node will fuse its neighbour's data with its own to generate a single packet of the same length and then transmit that to its other neighbour (if it has two neighbours). In the above example, node c0 will pass its data to node c1. Node c1 fuses node c0's data with its own and then transmits to the leader. After node c2 passes the token to node c4, node c4 transmits its data to node c3. Node c3 fuses node c4's data with its own and then transmits to the leader. Node c2 waits to receive data from both neighbours and then fuses its data with its neighbours' data. Finally, node c2 transmits one message to the BS. Thus, in PEGASIS each node will receive and transmit one packet in each round and be the leader once every 100 rounds.With our simulation experiments, we found that the greedy chain construction performs well with different size networks and random node placements. In constructing the chain, it is possible that some nodes may have relatively distant neighbours along the chain. Such nodes will dissipate more energy in each round compared to other sensors. We improved the performance of PEGASIS by not allowing such nodes to become leaders. We accomplished this by setting a threshold on neighbour distance to be leaders. Table 1 reflects this improvement. We may be able to slightly improve PEGASIS's performance further by applying a threshold adaptive to the remaining energy levels in nodes. Whenever a node dies, the chain will be reconstructed and the threshold can be changed to determine which nodes can be leaders. PEGASIS improves on LEACH by saving energy in several stages. First, in the local gathering, the distances that most of the nodes transmit are much less compared to transmitting to a cluster-head in LEACH. Second, the amount of data for the leader to receive is at most two messages instead of 20 (20 nodes per cluster in LEACH for a 100-node network).Finally, only one node transmits to the BS in each round of communication. In this paper, we describe PEGASIS, a greedy chain protocol that is near optimal for a data-gathering problem in sensor networks. PEGASIS outperforms LEACH by eliminating the overhead of dynamic cluster formation, minimizing the distance non leader-nodes must transmit, limiting the number of transmissions and receives among all nodes, and using only one transmission to the BS per round. Nodes take turns to transmit the fused data to the BS to balance the energy depletion in the network and preserves robustness of the sensor web as nodes die at random locations. Distributing the energy load among the nodes increases the lifetime and quality of the network. Our simulations show that PEGASIS performs better than LEACH by about 100 to 300% when 1%, 20%, 50%, and 100% of nodes die for different network sizes and topologies. PEGASIS shows an even further improvement as the size of the network increases. In order to verify our assumptions about PEGASIS, we will extend the network simulator ns-2 to simulate PEGASIS, LEACH, and direct transmission protocols. Based on our C simulations, we expect that PEGASIS will outperform the other two protocols in terms of system lifetime and the quality of the network.

As we all know that all the networks have a certain lifetime during which nodes have limited energy by using that, the nodes gather, process, and transmit information. This means that all aspects of the node, from the sensor module to the hardware and protocols, must be designed to be extremely energy-efficient. Decreasing energy usage by a factor of two can double system lifetime, resulting in a large increase in the overall usefulness of the system. In addition, to reduce energy dissipation, protocols should be robust to node failures, fault-tolerant and scalable in order to maximize system lifetime.

## 3. LEACH

LEACH is the first network protocol that uses hierarchical routing for wireless sensor networks to increase the life time of network. All the nodes in a network organize themselves into local clusters, with one node acting as the cluster-head. All non-cluster-head nodes transmit their data to the cluster-head, while the cluster-head node receive data from all the cluster members, perform signal processing functions on the data (e.g., data aggregation), and transmit data to the remote base station. Therefore, being a cluster-head node is much more energy-intensive than being a non-cluster-head node. Thus, when a cluster-head node dies all the nodes that belong to the cluster lose communication ability .




LEACH incorporates randomized rotation of the high-energy cluster-head position such that it rotates among the sensors in order to avoid draining the battery of any one sensor in the network. In this way, the energy load associated with being a cluster-head is evenly distributed among the nodes. Since the cluster-head node knows all the cluster members, it can create a TDMA schedule that tells each node exactly when to transmit its data. In addition, using a TDMA schedule for data transfer prevents intra-cluster collisions.

The operation of LEACH is divided into rounds. Each round begins with a set-up phase when the clusters are organized, followed by a steady-state phase where several frames of data are transferred from the nodes to the cluster-head and onto the base station

### 3.1 Set-up phase

In LEACH, nodes take autonomous decisions to form clusters by using a distributed algorithm without any centralized control. Here no long-distance communication with the base station is required and distributed cluster formation can be done without knowing the exact location of any of the nodes in the network. In addition, no global communication is needed to set up the clusters. The cluster formation algorithm should be designed such that nodes are cluster-heads approximately the same number of time, assuming all the nodes start with the same amount of energy. Finally, the cluster-head nodes should be spread throughout the network, as this will minimize the distance the non-cluster-head nodes need to send their data. A sensor node chooses a random number, r, between 0 and 1. Let a threshold value be T(n) : T(n)= p/1-p ×( r mod p-1 ). If this random number is less than a threshold value, T(n), the node becomes a cluster-head for the current round. The threshold value is calculated based on the above given equation that incorporates the desired percentage to become a cluster-head, the current round, and the set of nodes that have not been selected as a cluster-head in the last (1/P) rounds, p is cluster head probability. After the nodes have elected themselves to be cluster-heads, it broadcasts an advertisement message (ADV). This message is a small message containing the node's ID and a header that distinguishes this message as an announcement message. Each non-cluster-head node determines to which cluster it belongs by choosing the cluster-head that requires the minimum communication energy, based on the received signal strength of the advertisement from each cluster-head. After each node has decided to which cluster it belongs, it must inform the cluster-head node that it will be a member of the cluster. Each node transmits a join-request message (Join-REQ) back to the chosen cluster-head. The cluster-heads in LEACH act as local control centres to co-ordinate the data transmissions in their cluster. The cluster-head node sets up a TDMA schedule and transmits this schedule to the nodes in the cluster. This ensures that there are no collisions among data messages and also allows the radio components of each non cluster-head node to be turned off at all times except during their transmit time, thus minimizing the energy dissipated by the individual.

### 3.2 Steady-State Phase

The steady-state operation is broken into frames where nodes send their data to the cluster-head at most once per frame during their allocated transmission slot. The set-up phase does not guarantee that nodes are evenly distributed among the cluster head nodes. Therefore, the number of nodes per cluster is highly variable in LEACH, and the amount of data each node can send to the cluster-head varies depending on the number of nodes in the cluster. To reduce energy dissipation, each non-cluster-head node uses power control to set the amount of transmits power based on the received strength of the cluster-head advertisement. The radio of each non-cluster-head node is turned off until its allocated transmission time. Since all the nodes have data to send to the cluster-head and the total bandwidth is fixed, using a TDMA schedule is efficient use of bandwidth and represents a low latency approach, in addition to being energy-efficient. The cluster-head must keep its receiver on to receive all the data from the nodes in the cluster. Once the cluster-head receives all the data, it can operate on the data and then the resultant data are sent from the cluster-head to the base station.

**a) Advantages**

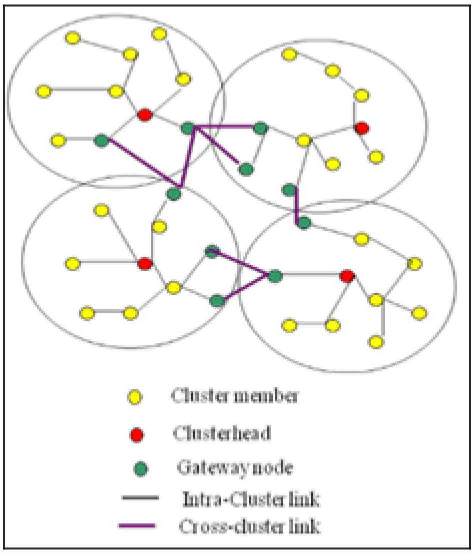
- Outperforms conventional routing protocols
- LEACH is completely distributed, requiring no control information from the base station
- Nodes do not need global topology information

**b) Disadvantages**

- Nodes must have data to send in the allotted time
- Perfect correlation is assumed, which might not be true always

## 4. HEED

HEED (Hybrid Energy Efficient Distributed) protocol is the clustering protocol. It uses using residual energy as primary parameter and network topology features (e.g. node degree, distances to neighbours) are only used as secondary parameters to break tie between candidate cluster heads, as a metric for cluster selection to achieve load balancing. In this all nodes are assumed to be homogenous i.e. all sensor nodes are equipped with same initial energy. But, in this paper we study the impact of heterogeneity in terms of node energy. We assume that a percentage of the node population is equipped with more energy than the rest of the nodes in the same network - this is the case of heterogeneous sensor networks. As the lifetime of sensor networks is limited there is a need to re energize the sensor network by adding more nodes. These nodes will be equipped with more energy than the nodes that are already in use, which creates heterogeneity in terms of node energy, leads to the introduction of H-HEED protocol.




### 4.1 Cluster Formation of Heed Protocol

In this section, we describe the network model. Assume that there are N sensor nodes, which are randomly dispersed within a 100m*100m square region. Following assumptions are made regarding the network model is:

i. Nodes in the network are quasi-stationary.
ii. Nodes locations are unaware i.e. it is not equipped by the GPS capable antenna.
iii. Nodes have similar processing and communication capabilities and equal significance.
iv. Nodes are left unattended after deployment.

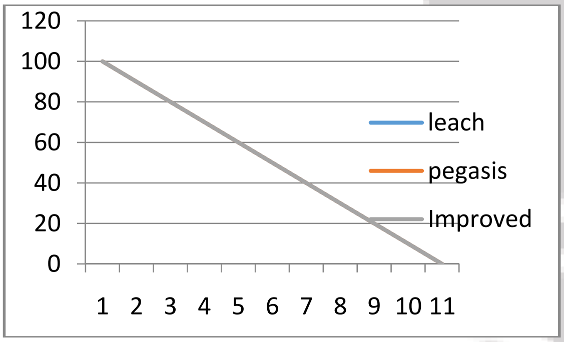
Cluster head selection is primarily based on the residual energy of each node. Since the energy consumed per bit for sensing, processing, and communication is typically known, and hence residual energy can be estimated. Intra cluster communication cost is considered as the secondary parameter to break the ties. A tie means that a node might fall within the range of more than one cluster head.

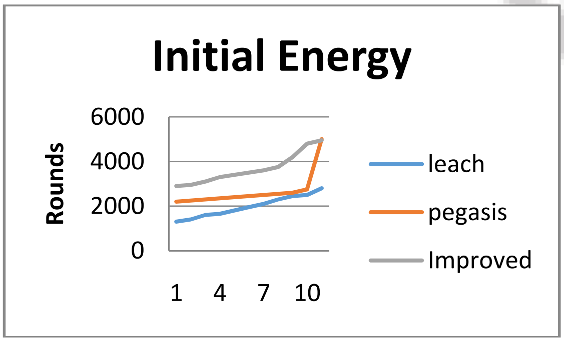
## References

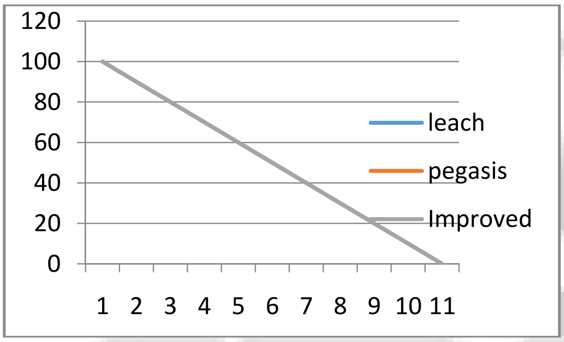
[1] NAVNEET KAUR ,AP.Dinesh kumar,"ANALYSIS OF CLUSTERING ALGORITHMS" IOSRJEN,VOL. 4,5 pages,MAY,2014.
[2] F. Akyildiz,W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A Survey on Sensor Networks," IEEE Communications Magazine, vol. 40, no. 8,pp. 102–114, Aug 2002.

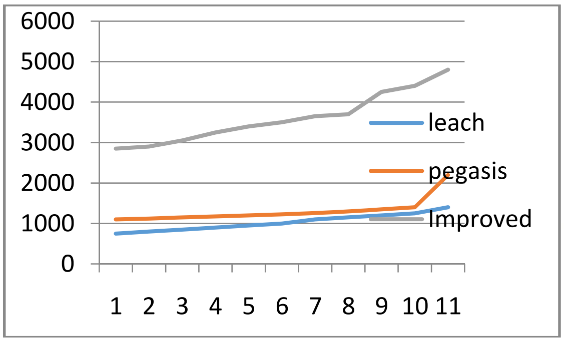
[3] S. Meyer and A. Rakotonirainy, "A Survey of Research on Context-Aware Homes," Workshop on Wearable, Invisible, Context-Aware, Ambient, Pervasive and Ubiquitous Computing, Adelaide Australia, 2003.
[4] B. Warn eke, M. Last, B. Liebowitz, Kristofer, and S. Pister, "SmartDust: Communicating with a Cubic-Millimeter Computer," ComputerMagazine, vol. 34, no. 1, pp. 44–51, Jan 2001.
[5] J. M. Kahn, R. H. Katz, and K. Pister, "Next Century Challenges Mobile Networking for Smart Dust," 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking, Aug 1999.
[6] V. Hsu, M. Kahn, and K. S. J.Pister, "Wireless Communication for Smart Dust," Electronic Research Laboratory Technical Memorandum, Feb 1998.

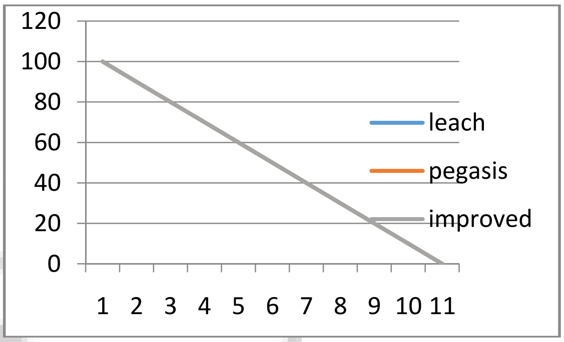
## Result Graphs

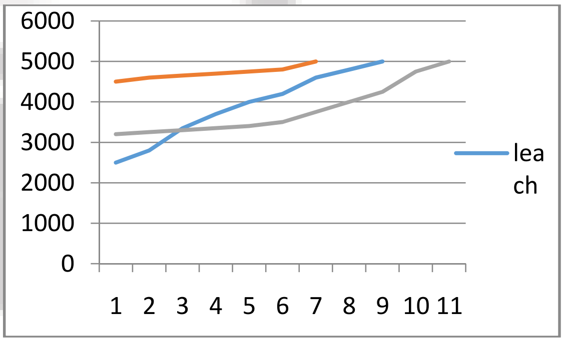



## Author Profile

**Navneet Kaur Sidhu** is M. Tech student. She has completed graduation and post graduation from PTU GZS campus. She is working as a supervisor in an international organization (IDP IELTS TEST CENTRE).

**Mr. Dinesh Kumar** is M. Tech qualified and pursuing Ph.D. Currently he is working as an assistant professor at PTU GZS campus. His work experience is about 10 to 12 years.