Volume 4, Issue 3, March 2014 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Routing Protocols for Wireless Sensor Networks: A Study

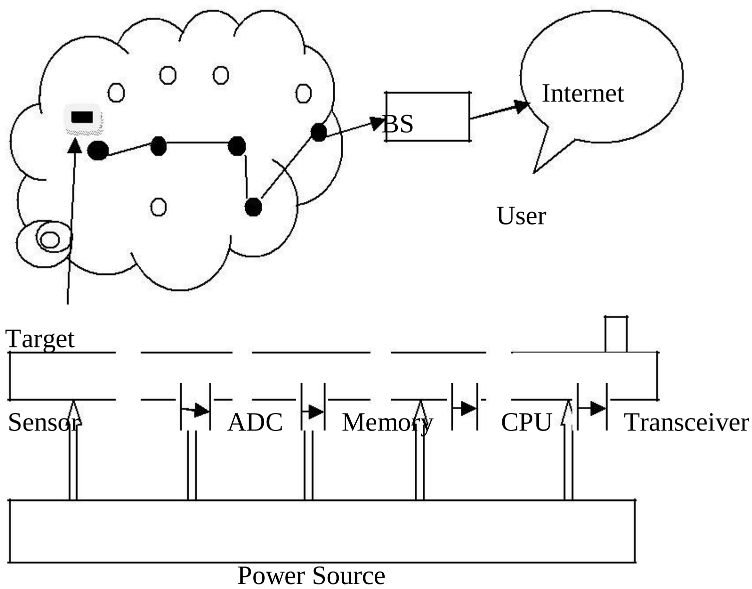
Sangita Khatkar
Computer science & Engineering,
Kurukshetra University (Kurukshetra), India

Abstract: Wireless Sensor network (WSN) is a fast growing and existing research that has attracted considerable research attention in the recent past. WSN consist a large number of sensor nodes that can be deployed inside or near to region that is to be observed. The performance of the network is greatly influenced by the routing techniques. Routing is to find out the path to route the sensed data to the base station. In this paper we are presenting a survey of routing protocols in WSN. Broadly, Routing protocols are classified in to three categories namely: Data Centric protocol, Hierarchal Routing protocol and Location Based routing protocol. We also study the trade-off between energy and communication overhead savings along with the advantages, disadvantages and performance issues in comparison to each other. In this paper, we first discuss the system architecture of the sensor network, then describe the some routing protocols for the sensor networks and finally give the comparison between them.

Keywords- WSN, Sensor, ADC, Memory, CPU, Transceiver

I. INTRODUCTION

Wireless sensor networks are one of the most important technologies that will change the world [10] in that such networks can provide us with fine-granular observations about the physical world where we are living. A wireless Sensor network is composed of large number of a low power electronics device called sensors nodes which are densely deployed in the wide area. Each node has the capability to sense the data from the environment perform some computation and communicate with the other nodes in the network. These sensor nodes have the various limitations such as low battery power, minimum computation capability. Once the sensor node is deployed, the network the network can keep operating only until the battery power is sufficient. as sink. All the other nodes in the network send the sense data to the sink ether directly or through the various nodes present in the network. Fig. 2 shows architecture of the sensor network in which the small circle shows the sensor nodes and a fill circle shows the gateway through which data is send to the main system. Fig. 3 shows that each node typically consists of five components: sensor unit (SU), analog digital convertor (ADC), central processing unit (CPU), power unit (PU), and Transceiver. They are



assigned with different tasks.

Fig. 1: Wireless sensor network

The sensor unit is responsible for collecting information as the ADC requests, and returning the analog data it sensed.

II. SYSTEM ARCHITECTURE FOR THE SENSOR NETWORKS AND ITS DESIGN ISSUE

A sensor network is composed of ten to thousand of the sensor nodes which are separate in the wide area. In the sensor network all the nodes are communicating to each other either directly or through the other nodes. In the sensor network one or more nodes among them are treated Transceiver is tasked to receive command or query from, and transmit the data from CPU to the outside world. CPU is the most complex unit. It interprets the command or query to ADC, monitors and controls power if necessary, processes received data, computes the next hop to the sink,

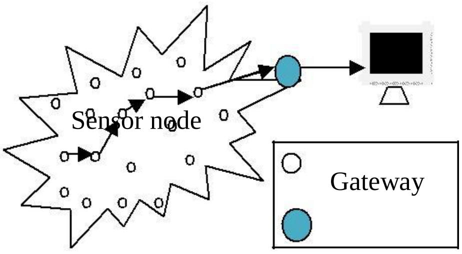


Fig. 2 Architecture of sensor network

etc. [2]. In case of wireless sensor network, the routing is slightly different form the other networks. The reason behind this is describe here. Firstly, there is no global addressing scheme to deploy the sensor node in the network so we can not apply any IP-based protocol here. Secondly, in order to communication, all the nodes present in the network do not communicate In this section we first discuss the classification of routing protocol for sensor network, then we discuss about the existing routing protocol.

III. CLASSIFICATION OF ROUTING PROTOCOLS

According to participation of the sensor node in the network, the routing protocols are mainly classified into three categories: *Data-centric or negotiation based protocol* [2, 3], these protocols are basically queried based and depend on the desired data (name of the data), which help us to remove or eliminate the redundancy of the data. *Hierarchical or cluster based protocol*, in this protocol, the group of some nodes in the network makes one or more clusters (depend on the size of the networks). In a cluster one node works as a cluster head. All nodes in a cluster first send the data to the cluster head; the cluster head perform some aggregation function upon this data then send to the sink or base station. And the last one is *location-based protocol*, these protocol utilize the position information of the desired data in the desired region than rather considering the whole network [2, 3]. Depending on the protocol operation, routing in WSNs can be divided into: 1. Multipath-based routing 2.Query-based routing 3.Negotiation-based routing 4.QoS-based routing.

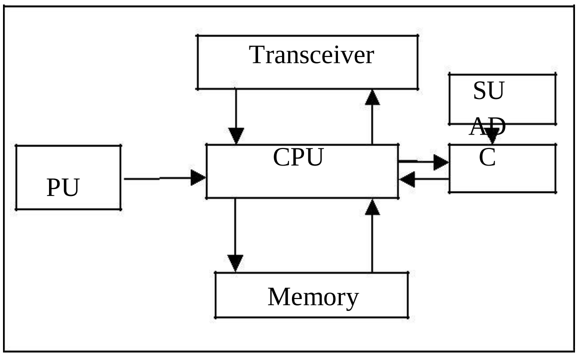


Fig. 3 Sensor node

A. *DATA-CENTRIC PROTOCOL*

In many application of sensor network it is not feasible to have unique global identifier due to numerous nodes deployment. Due to this problem it is hard to select a particular set of nodes to be queried. Therefore data is usually sent from each sensor node within the deployment region with lot of redundancy. This is very wasteful in terms energy consumption. Routing protocols that will be able to select a set sensor nodes and use data aggregation during the relaying of data have been considered. This consideration has led to data centric protocols. Based on the data centric algorithm SPIN is the first protocol.

a) SPIN (Sensor protocol for information exchange)

The first category of data centric protocol is SPIN (sensor protocol for information exchange).In SPIN, before the actual data transmission takes place, Meta data are exchanged among the sensor node via a advertisement mechanism which is the main feature of this protocol. When a node receives a new data, it advertises about this data to its neighbor and then interested neighbors (those who do not have this data) demand for data by sending a request message. There is no standard adoption for the Meta data, it is application dependent. SPIN is the 3-stage protocol since there are three messages in order to have communication between nodes. ADV (Advertisement): To advertise new data, REQ (Request):

To Request for data & DATA: Carry the actual data. SPIN protocol starts when a node has new data to send. It does so by sending an ADV message to its neighbor's node. The neighbors node which are interested to retrieve data send request by sending REQ message to the node from where ADV has come. Upon receiving the REQ message node sends actual data by DATA message. This process continues for the neighbor nodes also. As a result whole sensor area receives a copy of data. Fig. 4 has shown the various operation of SPIN.

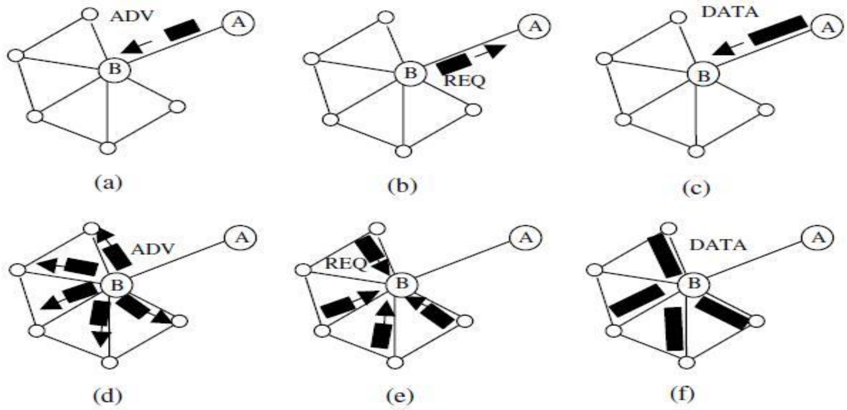


Fig. 4: SPIN Protocol (a) Node A starts by advertising its data to node B. (b)Node B responds by sending a request to node A.

(c) After receiving the requested data, (d) node B then sends out advertisements to its neighbors, (e-f) who in turn send requests back to B.

In this way SPIN's Meta data negotiation solves the classic problems of flooding such as duplicate data passing, overlapping such as sending data by two nodes covering the

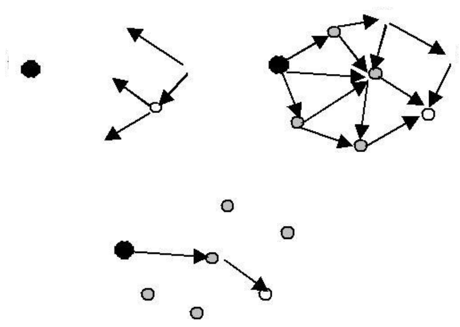


Fig. 5: Directed Diffusion protocol phases. (a) Interest Propagation, (b) Initial Gradient setup and, (c) Data delivery along reinforcement.

same area, and resource blindness hence, achieving energy efficiency. One of the most important advantages of the SPIN is that topology changes are localized since each node has needed to know about its single hop neighbor. On the other hand this protocol does not ensure the delivery of data. For this consider the case (like intrusion detection) which need data should be reported over periodic intervals and assume that if the nodes that are interested in the data are far away from the source node and the nodes between source and destination are not interested in that data, such data will not be delivered to the destination at all.

b) Directed Diffusion

Directed diffusion is an important data centric routing protocol. The idea behind at diffusing data through sensor nodes by using a naming scheme for the data. The main reason behind using such a scheme is to get rid of unnecessary operations of network layer routing in order to save energy. Direct Diffusion suggests the use of attribute-value pairs for the data and queries the sensors in an on demand basis by using those pairs. In order to create a query, an interest is defined using a list of attribute-value pairs such as name of objects, interval, duration, geographical area, etc. A request for data is broadcasted from an initiator node (Sink) in the network, and the positive answers from the sensors are forwarded back to the initiator (following a Shortest-Path-Tree (SPT) construction rooted at the initiator). Each node receiving the interest can do caching for later use. The nodes also have the ability to do in-network data aggregation, which is modeled as a minimum Steiner tree problem. The interests in the caches are then used to compare the received data with the values in the interests. The interest entry also contains several gradient fields. A gradient is a reply link to a neighbour from which the interest was received. It is characterized by the data rate, duration and expiration time derived from the received interest's fields.

Table 1: Comparison among LEACH, SPIN and Directed Diffusion

	SPIN	LEACH	Directed Diffusion
Optimal Route	No	No	Yes
Network Lifetime	Good	Very Good	Good
Use of meta data	Yes	No	Yes
Resources awareness	Yes	Yes	Yes

Hence, by utilizing interest and gradients, paths are established between sink and sources. Several paths can be established so that one of them is selected by reinforcement. The sink resends the original interest message through the selected path with a smaller interval hence reinforces the source node on that path to send data more frequently. Directed Diffusion differs from SPIN in terms of the on demand data querying mechanism it has. In Directed Diffusion the sink queries the sensor nodes if a specific data is available by flooding some tasks. In SPIN, sensors advertise the availability of data allowing interested nodes to query that data. Directed Diffusion has many advantages. Since it is data centric, all communication is neighbor-to-neighbor with no need for a node addressing mechanism. Each node can do aggregation and caching, in addition to sensing. Caching is a big advantage in terms of energy efficiency and delay. In addition, Direct Diffusion is highly energy efficient since it is on demand and there is no need for maintaining global network topology. However, Directed Diffusion cannot be applied to all sensor network applications since it is based on a query-driven data delivery model. The applications that require continuous data delivery to the sink will not work efficiently with a query-driven on demand data model. Therefore, Directed Diffusion is not a good choice as a routing protocol for the applications such as environmental monitoring. The comparison among LEACH,SPIN and Directed Diffusion is described in table 1.

B. *HIERACHICAL PROTOCOL*

Scalability is the one of major design issue in the all network. A single-tier network can cause the gateway to overload with the increase in sensors density. Such overload might cause latency in communication and inadequate tracking of events. In addition, the single-gateway architecture is not scalable for a larger set of sensors covering a wider area of interest since the sensors are typically not capable of long-haul communication. As we discuss the previous protocol (Data-centric), which is not so scalable. If the number of the node (overload) in the network is increases, the performance is degrade (data-centric protocol is not suitable for the wide area). To increase the scalability of the system and cover a wide area without degrading the performance clustering is the best.

The main aim of the hierarchical based routing is to efficiently maintain the energy consumption of the sensor nodes by involving them in multi hop communication within a particular cluster and performing the data aggregation and fusion in order to decrease the no of transmitted messages to the sink. LEACH [5] is the first cluster based routing protocol for the sensor network.

a) Low Energy Adaptive Clustering Hierarchy (LEACH)

LEACH [5,6] is a most popular clustering-based protocol for the sensor network. The idea is to form clusters of the sensor nodes based on the received signal strength and use local cluster heads as routers to the sink. This will save energy since the transmissions will only be done by such cluster heads rather than all sensor nodes. It utilizes randomized rotation of the cluster-heads to evenly distribute the energy load among the sensor nodes in the network. All nodes in the network are homogeneous and energy-constrained. The main energy saving of *LEACH* protocol comes from the combination of data compression and routing.

LEACH is a dynamic clustering method. In this method, time is partitioned into fixed intervals with equal length. At the beginning of each interval, each sensor becomes a cluster head with some predefined probability. The cluster heads then broadcast messages to their neighbors. Other sensors receive messages and join a cluster by choosing the cluster head with the strongest signal. During the interval, cluster members send information to their cluster head. The cluster heads aggregate the information, compress the information, and route the information to the remote access point. Once the

interval ends, the whole clustering process restarts. Hence, the clusters and cluster heads are not fixed.
The data aggregation and data fusion function is performing local to the cluster. The cluster heads are changing over the time to balance the energy of the sensor node. The decision for choosing the cluster head is made by the node choosing a random number between 0 and 1. The node becomes a cluster head for the current round if the number is less than the following threshold:

$$T(n) = \begin{cases} \dfrac{p}{1 - p * (r \bmod \frac{1}{p})} & \text{if } n \in G, \\ 0 & \text{otherwise} \end{cases} \qquad (1)$$

Where p is the desired percentage of cluster heads (e.g. 0.05), r is the current round, and G is the set of nodes that have not been cluster heads in the last 1/p rounds. LEACH includes distributed cluster formation, local processing to reduce global communication, and randomized rotation of the cluster-heads. Together, these features allow LEACH to achieve the desired properties. Initial simulations show that LEACH [5.6] is an energy-efficient protocol that extends system lifetime.
The drawback of LEACH is: First, it didn't consider the distribution of nodes. Fig. 6 is an example of the clusters constructed by LEACH. Black nodes represent cluster-heads. All the cluster-heads are distributed on the upper right corner. Secondly, for cluster-heads that are far away from the base station, clearly direct transmission is bad for energy conservation.

b) Maximum energy cluster-head (MECH)

Maximum Energy Cluster-head (MECH) is based on LEACH. Our approach is divided into rounds and each round consists of three phases: set-up phase, steady phase

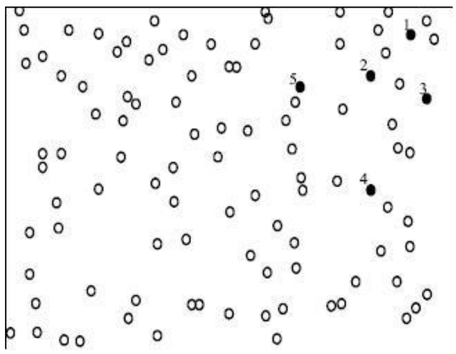


Fig. 6. Random nodes in LEACH

and forwarding phase. We construct clusters before each round. MECH consider the drawback of the LEACH [5, 6] protocol.
i) Cluster Initialization:

Each node broadcast a hello message with a TTL, which is set at once to gather the one hop neighbor. Each node maintains the record of number of neighbor. Define a system parameter called cluster number (*CN*). When the number of neighbors of a node reaches *CN*, the node will broadcast an advertisement to its one hop neighbors. The advertisement is to inform every node that "I am a cluster-head". All the nodes who receive th is advertisement record it and start a back-off timer. Furthermore, such nodes never broadcast the advertisement even if their number of neighbors reaches *CN*. The steady phase will be in direct proportion to the number of cluster members. In order to synchronize, each cluster-head counts the total number of its cluster members and broadcasts the information to the base station. The base station computes the maximum number of time slots needed and broadcasts this information back to each cluster head.
ii) *Set-up phase***:**

Because the maximum number of cluster members is bounded by *CN*, the cluster-head can schedule the TDMA time slot for each cluster member at each round. In the set-up phase, every node will turn on the receiver such as in LEACH. Then, the cluster-head will broadcast an advertisement that contains the TDMA time slot information. Each cluster member will know the time slot which belongs to it. Thus, the cluster member will keep the transceiver off until its time slot. It transmits the sensing data to the cluster-head during its time slot. The cluster-head maintains a table which records the node with maximum power at current round. After it forwards the data to the base station, it selects the node that has the maximum remaining power as the cluster-head for the next round.

iii) Steady phase:

Once the clusters are created and the TDMA schedule is fixed, data transmission can begin. Assuming nodes always have data to transmit, they send their sensed data and energy during the allocated time slot to the cluster-head. The cluster nodes adjust their transmission energy dynamically depending on the signal strength of cluster-head advertisements. In the steady phase, only the cluster-head always turns on the transceiver. The cluster members only turn on the receiver during its allocated time slot.

*iv)Forwarding phase***:**

MECH uses two parameters to establish the hierarchical relation among clusters: *hop_count* and energy level. The following steps describe how the MECH implements the hierarchy.

Step 1: The Base station broadcasts a hello message periodically. The message contains two types of information: one is *hop_count* from base station to the cluster-head and the other is energy level. The *hop_count* is set to 0 and energy level is set to 0 initially. A non-cluster-head node ignores this message. When receiving this message, a cluster-head node adds one to the *hop_count* and compares its energy level with that in the message. If the current cluster-head energy level is lower, the energy level in the message is replaced with that of the cluster-head.

Step 2: After a cluster-head receives the hello message, the cluster-head will record the cluster-head *x* that forwards this hello message. Thus, *x* will become the forwarding cluster-head to the base station for this cluster-head. This mechanism is similar to that in a distance-vector routing.

Step 3: If a cluster-head receives another hello message, it will make decisions according to the following rules:

Case 1: $hop\_count_{old} < hop\_count_{new,}$ do nothing;

Case 2: $hop\_count_{old} > hop\_count_{new}$ replace the forwarding cluster-head;

Case 3: $hop\_count_{old} = hop\_count_{new}$ then If $energy_{old}$ is $energy_{new}$ replace the forwarding cluster-head; Else do nothing;

The MECH has the following features: Reducing energy dissipation, Self-configuration and localized coordination, Cluster head has the maximum energy and Load balanced.

c) *Power-efficient GAthering in Sensor Information Systems (PEGASIS):*

Power-efficient GAthering in Sensor Information Systems (PEGASIS) [9] is an improvement of the LEACH protocol. Rather than forming multiple clusters, PEGASIS forms chains from sensor nodes so that each node transmits and receives from a neighbor and only one node is selected from that chain to transmit to the base station (sink).

Gathered data moves from node to node, aggregated and eventually sent to the base station. The chain construction is performed in a greedy way. As shown in Fig. 6 node N0 passes its data to node N1. Node N1 aggregates node N0's data with its own and then transmits to the leader. After node N2 passes the token to node N4, node N4 transmits its data to node N3. Node N3 aggregates node N4's data with its own and then transmits to the leader. Node N2 waits to receive data from both neighbors and then aggregates its data with its neighbor's data. Finally, node N 2 transmits one message to the base station or SINK.

The difference from LEACH is to use multi-hop routing by forming chains and selecting only one node to transmit to the base station instead of using multiple nodes. However, PEGASIS introduces excessive delay for distant node on the chain. In addition the single leader can become a bottleneck.

d) Group-based Sensor Network (GSEN)

Routing protocols for sensor network are very much application dependent. It is not possible to design a routing protocol that could meet up all the requirements under every scenario and for all kind applications. The operation in the GSEN [8] can be divided into two phase: *Group formation phase and transmission phase.*

i) *Group formation phase*:

In this phase, the all network establishment begins with the group formation. There is several groups are formed with a group leader in each group. After the group formation a higher level group is formed by all group leaders in the network. Among this leader-to-leader route only one is randomly chosen as high-level leader. The group leader keeps on changing in every round in random order. In the formation of group leader this protocol uses the same algorithm as LEACH [5-6]. But unlike LEACH [5, 6] where cluster set-up takes place at every round, GSEN [8] [5-6] prefers to reconstruct groups after every R number of rounds. Therefore, once the groups are formed they remain fixed for the next (R-1) number of rounds but the responsibility of the group leader keeps on changing randomly among the other member nodes of the group at every round. Therefore we can alleviate the cost of dynamic cluster set-up in LEACH at every round. The groups are fixed but the leaders are chosen randomly at every round from different positions in the chain therefore the energy expenditure in the network is already in a random order and this is very important to let the sensors die in random positions. But still there is a slight improvement if we change the groups after several rounds. But changing groups requires extra negotiation therefore extra energy. Random selection of the higher-level leader facilitates energy dissipation in random positions of the network. Selection of the group leader can be done in the following way: while constructing the leader's chain, nodes temporally number themselves. The node that starts chain formation gives an id '0' to itself and transmits t hat id to its next neighbor. The next node then assigns itself the id= (previous node id+1) and transmits this id to its next. In this way the nodes in chain have a unique and sequential node id. Then node having id= round(r* N) will be selected to transmit to BS where r = 'a random number between 0 and 1' and N= 'total number of group-leaders' in the network. After that leaders are selected from random positions and a leader-to-leader route is formed. Finally, one higher-level leader among other leaders is selected in random to transmit to BS. It should be noted that groups are not reconstructed at every round but only after every *R* number of rounds but leaders are selected at every round as well as leader-to-leader route.

ii) *Data collection and Transmission phase*:

Each group in GSEN covers a certain geographical region and each group leader acts as a representative of that region. At the beginning of the data collection and transmission phase each leader accumulates data from the member nodes within its group. For the initiation of data transmission we consider the similar token passing mechanism as in PEGASIS. As shown in Fig. 9, if node 'n3' is elec ted as a leader it sends a token toward node 'n1' which th en ends data to node 'n2'. Node 'n2' fuses the received dat a with its own data and transmits again to the group leader. After that 'n3' sends token toward 'n5', node 'n5' then sends its data toward the leader through node 'n4'. As the size of the token is small so cost associated to it is negligible. Each node except the end nodes of the chain fuses its own data with the neighbor's data to generate a single fixed length packet and transmits that to the other neighbor.

The process continues till data reach to the leader node. When all the leaders finish their data collection, only one randomly selected leader among all leaders collects other's data through previously described leader-to-leader chain and transmits the whole aggregated information to BS.

c) *CMST-DC: Cluster based Sensor Network*

CMST-DC (Cluster based Minimal Spanning Tree with Degree-Constrained)[10], a cluster tree based protocol. Sensors are grouped into several clusters. In every cluster, a routing tree is constructed for data transmission. One sensor node is elected as a cluster head in every cluster based on the residual energy and this node remains as a cluster head for an optimal number of rounds. Among all cluster heads, a routing tree is also constructed. One cluster head is selected to be the leader of all cluster heads based on some measures at every round. All nodes in a cluster send messages to the cluster head. Besides, all cluster heads send the information to leader of cluster heads. The leader is the node that transmits the information to the BS. After an optimal number of rounds, new group of cluster heads are selected. Due to the hierarchical tree structure, this protocol requires much lower time and energy as compared to other protocols of the wireless sensor networks for data collection [8-10]. The operation in cluster based Minimal Spanning Tree with Degree-Constrained (CMST-DC) routing protocol can be divided by two phases: cluster formation phase, tree formation phase and data transmission phase.

d) *Cluster formation phase:*

At the beginning of cluster formation only, here the authors adapt the same algorithm as LEACH for the selection of cluster head node. The idea of LEACH where each sensor chooses a number between 0 and 1. If the number is less than a threshold, the node broadcasts itself as the leader. Non-leader nodes receiving the broadcast decide by themselves to which leader it will join depending on the signal strength and inform the corresponding leader by sending an acknowledgement**.** After collecting all the acknowledgement signals, each leader node initiates tree formation starting from itself connecting all the nodes in the cluster. Once the tree is constructed, one node is selected by some criteria at every round. The main difference between this approach and LEACH is that, unlike LEACH where cluster set-up takes place at every round, this approach prefers to re-build clusters after a certain number of rounds. Thus, once the clusters are formed they remain fixed until next cluster formation phase is needed. When a cluster is formed, a tree has been constructed to connect all nodes within the cluster. Since all nodes in a cluster need to send data to the cluster head, the authors use the idea of minimum spanning tree (MST) to shorten the total transmission distance. This means a minimum spanning tree of the nodes in a cluster. However, it is possible that a node in the computed MST will be connected with many other nodes. In such case, this node needs to fuse more data collected from its neighbors than other nodes and consumes more energy. This may cause the node to die earlier than other nodes. In order to avoid the situation that a node will be connected with many other nodes, we introduce the degree constraint to each tree node.

Tree formation phase: The tree formation algorithm starts with the cluster head. This head node is treated both as a tree node and a starting node. A node which is a starting node will broadcast a find-nearest-neighbor (FNN) message with largest transmission range to find the nearest live node among all nodes in a cluster. Once a node receives the FNN message, the node first checks the message to see if it is sent by the node in the same cluster. It is not from the same cluster, the message is simply ignored. Otherwise, the node sets a backoff timer of t1 seconds, where t1 is distributed in some range and depends on the signal strength of the received message. The more signal strength of the received message, the less t1 will be. When the timer expires, the node sends back an acknowledgement (ACK) message with its node identification to the starting node. If a node hears other ACK messages before its timer expires, it cancels its timer. When all tree nodes receive the ACK message, they will set a backoff timer of t2 seconds. Again, t2 is relative to the signal strength of the received ACK message. When t2 expires, the node sends a confirmation (CFM) message with node ID to inform the node sent the ACK message to be the next starting node and the link between them can be constructed. The above process will be repeated for finding next nearest live neighbor node until no live neighbors exist.

C. LOCATION-BASED PROTOCOL

When we are taking any sensor routing protocol into consideration they require location information for sensor nodes. since, there is no addressing schema like IP-addresses and they are spatially deployed on region, that's why we need location information in order to calculate distance between two particular nodes and on this calculation make estimation of energy consumption and make utilization of information in order to route data in energy efficient way. Location information can be utilized in routing data in an energy efficient way. For instance, if the region to be sensed is known, using the location of sensors, the query can be diffused only to that particular region which will eliminate the number of transmission significantly. Some of the protocols discussed here are designed primarily for mobile ad hoc networks and consider the mobility of nodes during the design [11] [12]. However, they are also well applicable to sensor networks where there is less or no mobility. It is worth noting that there are other location-based protocols designed for wireless ad hoc networks, such as Cartesian and trajectory-based routing [13][14]. However, many of these protocols are not applicable to sensor networks since they are not energy aware. Geographic Adaptive Fidelity (GAF) [11] is energy aware routing protocol designed for mobile adhoc network, but also applicable to sensor network. The basic idea of GAF for conserve energy is that it turn off nodes that are equivalent to some another nodes in same region in routing perspective and achieve same level of routing fidelity. Nodes are arranged in Virtual grid form and each GAF node have location information in order to associate itself to virtual grid. Nodes associated with the same point on the grid are considered equivalent in terms of the cost of packet routing. such type of arrangement make possible of some node in sleeping mode and some in active mode that do all routing work, nodes periodically wake up and do load balancing, nodes in same grid coordinate to each other in this way to save energy. To understand this scenario more clearly lets take an example that is redrawn from [11]. In this fig. 8, node 1 can reach any of 2, 3 and 4 and nodes 2, 3, and 4 can reach 5. Therefore nodes 2, 9 and 10 are equivalent and two of them can sleep. or how long is application dependent and according to r routing process parameters set. The sleeping neighbours adjust their sleeping time accordingly in order to keep the routing fidelity. Before the leaving time of the active node expires, sleeping nodes wake up and one of them becomes active. GAF is implemented both for non-mobility (GAF-basic) and mobility. This distance is approximately equal to Euclidean length of a sensor node's shortest path to base station.

Initially node starts with discovery node, in this state node exchange discovery message to determine the neighbor's node into grid. Which one node will active and To understand more clearly we take a sample fig 7. redrawn from [12]. When a node receive data it forward it to base station using greedy forwarding

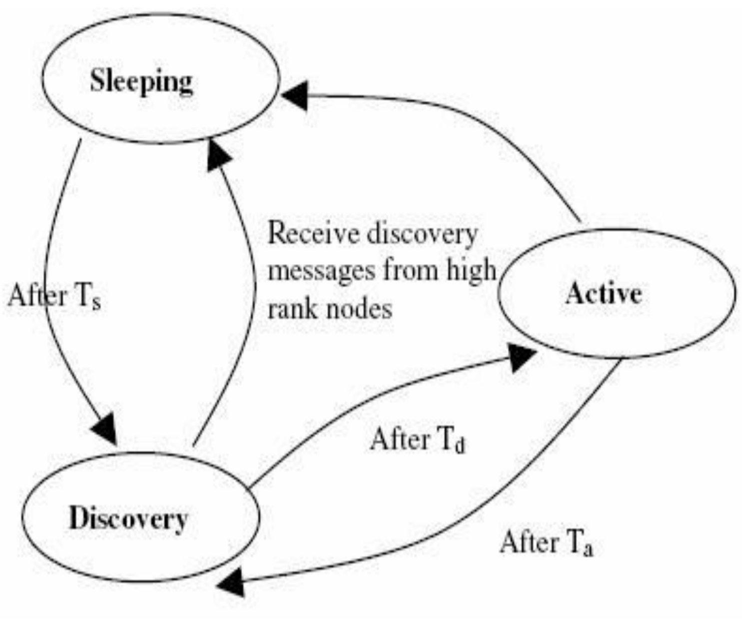


Fig.7. State transitions in GAF.

[15],but this approach may no longer work when node is concave node (minimum local) that has no more closer neighbors to base station, to avoid this situation PAGER-M uses high-cost-to-low-cost rule when packet reaches near to

concave node. Another features of PAGER-M is stateless routing protocol because it does not bother about to remember past traffic/path of node and it reduce transmission failure, due to mobility, by providing multiple forwarding choice to a single node.

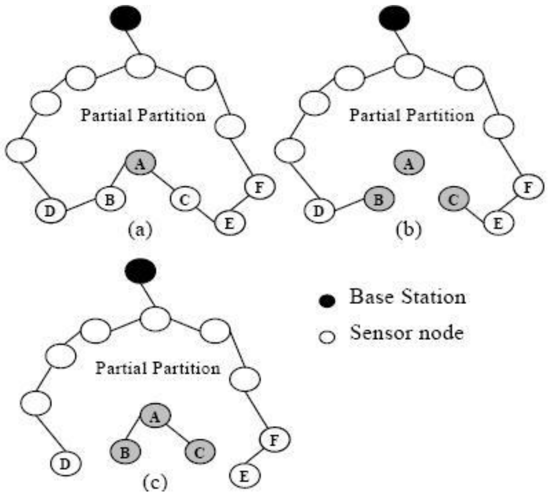


Fig.8: Various operation of PEGER-M

IV. CONCLUSION

The performance and longevity of the network greatly influenced by the routing techniques, adopted for route the data from sensor nodes to the base station. In this paper, we surveyed on different techniques for routing in WSN. Overall, depending upon the network structure, routing techniques are divided into three categories. Data centric or flat routing, hierarchical or cluster routing and location based routing. All these routing techniques have common goal to increase the longevity of the network.

REFERENCES

[1] F.L. LEWIS"Wireless sensor networks" appears in Sma rt Environments: Technologies, Protocols, and Applications ed. D.J. Cook and S.K. Das, John Wiley, New York, 2004.

[2] Qiangfeng Jiang, D. Manivannan "Routing Protoc ols for Sensor Networks" in IEEE trans. 2004.

[3] Kemal Akkaya , Mohamed Younis "A survey on routing protocols for wireless sensor networks" Ad Hoc Networks 3 (20 05) page no. 325–349.

[4] Peng Jiang, Yu Wen, Jianzhong Wang, Xingfa Shen, and Anke Xue "A Study of Routing Protocols in Wireless Sensor Ne tworks" IEEE Proceedings of the 6th World Congress on Intelligent Control and Automation, June 21 - 23, 2006, Dalian, China.

[5] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy efficient Communication Protocol for Wireless Micro Sensor Networks," in Proc. of the 33rd Annual Hawaii Inter national Conf. on System Sciences, 2000, pp. 3005–3014.

[6] Dezhen Song "Probabilistic Modeling of Leach Protoc ol and Computing Sensor Energy Consumption Rate in Sensor Networks" CS Department, Texas A&M University Technical Report: TR 2005-2-2, February 22, 2005

[7] Ruay-Shiung Chang and Chia-Jou Kuo" An Energy Effic ient Routing Mechanism for Wireless Sensor Networks" Pro ceedings of the 20th International Conference on Advanced Information Networking and Applications (AINA'06) IEEE, 2006

[8] Nahdia Tabassum, A K M Ahsanul Haque, Yoshiyori Urano "GSEN: An Efficient Energy Consumption Routing Scheme for Wireless Sensor Network" Proceedings of the International Co nference on Networking, International Conference on Systems and International Conference on Mobile Communications and Learning Technologies (ICNICONSMCL'06) IEEE trans, 2006.

[9] S. Lindsey, C.S. Raghavendra, "PEGASIS: power effic ient gathering in sensor information systems", in: Proceedings of the IEEE Aerospace Conference, Big Sky, Montana, March 2002

[10] Chiu-Kuo Liang, Yu-Jie Huang and Jian-Da Lin"An Ene rgy Efficient Routing Scheme in Wireless Sensor Networks" 22nd International Conference on Advanced Information Networking and Applications IEEE 2008.

[11] Y. Xu, J. Heidemann, and D. Estrin, "Geography-informed energy conservation for ad hoc routing," in the Proceedings of the 7th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'01), Rome, Italy, July 2001.

[12] Le Zou, Mi Lu, Zixiang Xiong "PAGER-M: A Novel Loca tion-based Routing Protocol for Mobile Sensor Networks"

[13] G. Finn, "Routing and addressing problems in large metropolitan-scale internetworks," University of Southern Califo rnia, Tech. Rep. ISI Research Report ISI/RR-87-180, 1987.

[14] B. Nath and D. Niculescu, "Routing on a curve," in HOTNETS 1, Princeton, NJ, October 2002.

[15] I. Stojmenovic and X. Lin, "Loop-free Hybrid Single -Path/Flooding Routing Algorithms with Guaranteed Delivery for Wireless Networks," IEEE Trans. Parallel Dist. Sys., vol. 12 , no. 10, 2001, pp. 1023–32.

