Advanced Routing Algorithms for Partially Connected Cluster Based Wireless Sensor Networks

Need of specific protocol stack in wireless sensor networks (WSN):

WSN is a network formed by deploying sensor nodes in a particular area called as the area under surveillance. The aim of deploying sensor nodes is to collect the information, such as temperature, from the area and send that information to the base station. At base station required computation takes place and the output is presented to the user in an understandable manner. The base station can send commands, if necessary, to the actuators at sensor node. The actuator is used to turn on/off the connected application according to the commands it receives from the base station. So in WSN nodes can communicate with each and to the base station for sending information from the area under surveillance.

Unlike MANET the WSN is an optimized system with certain constraints which has to be followed in order to achieve efficiency. As the sensor nodes are battery powered and operate in the unattended area, the energy saving at the node is cardinal parameter **[1].** The small size of sensor node put a restriction on the battery size and in terms put a restriction on the amount of energy available at sensor node. The sensor nodes are expected to perform for a long time, for several months in some applications, with such limited power supply. Therefore, energy consumption is the main issue in WSN and it is considered as an umbrella term. For reducing the energy consumption it is very important to find where we are spending energy and how to control it. If we consider the architecture of sensor node it is consist of the communication device, controller, memory and power supply. According to literature, a large amount of energy is consumed in communication as the sensor nodes take advantage of broadcasting **[2].** However, broadcasting is an important feature of wireless sensor network which gives important network performance parameters such as scalability and reachability. Thus, it is not possible to eliminate broadcasting but it is possible to control when to broadcast and when to remain in the inactive state.

The communication devise at sensor node consists of only one trans-receiver which functions as transmitter and receiver. Sensor node will set up a timer for a particular time described by protocol stack and after the timer expires it will set itself in transmitting mode. The data packets will be broadcasted to all of the neighbors in communication range. After completion of transmission phase, the node will again set the timer for reception and will receive packets in reception mode. Considering two nodes sending data to each other at same time, it can be said that the data packets will collide and the information will get corrupted within the medium. Moreover, inability to send data effectively will result in retransmission and will increase communication overhead **[4]**. If it can be possible to control when to send data, when to receive data and when to remain in inactive phase then most of the energy at sensor node can be saved. So the first reason for defining specific protocol stack is to control the trans-receiver so that more energy can be saved at sensor node.

The second important aspect is to choose efficient topology for the network to increase efficiency. There are three topologies in WSN as star, mesh and cluster topology [fig1]. In a star topology, all the node are connected to the base station and each node is responsible for direct communication between node and base station. This topology increases load at the base station because each node is directly connected to the base station. As each node has to send its data packets to base station, the radio range of sensor node should be more which consumes more energy **[3]**. In a mesh topology, the burden at base station is reduced to some extent but the number of nodes responsible for data transmission increases. Although mesh topology reduces burden at base station, it fails to provide energy efficiency. The third topology is cluster-based topology in which the nodes are arranged in virtual clusters according to their radio range. This topology not only reduces the load at base station but also provides energy efficient operation by arranging nodes in the virtual cluster.

In cluster topology, the single node from the cluster will be responsible for sending data to base station so the energy consumption of other nodes can be reduced. However, if the same node is sending data to the base station for more time then it is possible that it will die soon and the cluster will get disconnected from the network. So it is important to distribute the load of sending data to base station among the nodes in the cluster. It can be done by selecting different cluster leader for different cycles of data transmission. Thus, we need a specific algorithm which can select cluster leader for different cycles. Considering the aspects of energy consumption, cluster topology and effective selection of cluster leader, it is obvious that we need specific protocol stack which can perform all three operations.

Partially Connected cluster based WSN:

In cluster topology, nodes are grouped in the virtual cluster according to their radio range. In completely connected topology each node can hear other node and can send data packets [fig1]. However, this is not the case in every cluster-based topology **[2]**. It is possible that some nodes are not in direct contact with each other but have some common node between them. Such topology where not every node in the cluster can hear each other is called s partially-connected cluster. So for such network topology, it is important that the partially connected nodes should communicate via the common node.

Specification and constraints for routing algorithms:

As discussed in the previous section the routing algorithm should select a node from the cluster as a cluster leader and hold it responsible for transmitting data to base station. There are some algorithms defined for selecting node as cluster leader which considers knowledge about cluster as an important aspect. So the node with maximum knowledge of network will be selected as cluster leader. However, these algorithms are best suited for the completely connected cluster. The advancement over such algorithms gives new techniques which can be used in partially connected topologies as well. These advanced algorithms connect the partially connected nodes by the intermediate node. Intermediate node takes the responsibility to send data from one node to other.

The algorithm proposed by Dr.Muzumdar **[1]** gives the detailed structure of electing cluster leader in partially connected WSN by considering topology knowledge at the node as an important parameter. However, it is possible that the topology of network is static cluster topology in which every node in the cluster does not change its position. For such topology, it is obvious that every time the same node will have complete information about the network and because of this, the same node will be elected as cluster leader. If the same node is responsible for sending data packets to the base station every time then the node will die in less time. There are no specific provisions for fair distribution of selecting cluster leader in this algorithm. Some algorithms like LEACH-CC **[4]** consider energy at the node as an important aspect and divide the cluster network in TDMA and CDMA schedules. It considers the inter-cluster and intra-cluster communication and provides TDMA schedules for inter-cluster and CDMA schedules for intra-cluster communication. This protocol also considers the energy level of the node to be selected as cluster leader. However, this protocol is more suitable for completely connected cluster network.

The efficiency of wireless sensor network depends on the overall performance of network which includes a lifetime of network and energy efficiency. Thus to maximize the efficiency and to select cluster leader effectively we should consider two important parameters. Firstly the cluster leader should have information about all nodes in the cluster and secondly the cluster leader must have sufficient energy to send data packets to the base station. Considering these parameters the proposed algorithm is designed which gives robustness and scalability to the network.

Architecture of proposed algorithms:

The algorithm considers partially connected sensor network with nodes having some initial energy. This algorithm also takes advantage of in-network processing and data aggregation technique. In data aggregation, not every bit recorded at the sensor node is sent to the base station, instead, some data processing is done at the node and the aggregated data is ten sent to the base station. The data aggregation includes calculating some parameters such as mean, max or minimum value of data. Data aggregation reduces the load on the network by lowering the communication overhead. Considering the fig[2], the aim of the algorithm is to decide cluster leader from the scattered nodes considering maximum information and sufficient energy level. The algorithm consist of five phases:

1. Energy Level Computation
2. Initialization of cluster
3. Contention between nodes
4. Exchange of information
5. Termination & cluster leader confirmation

The first three phases give temporary cluster leader and in the fourth phase election of the cluster leader is confirmed by comparing information with other nodes. In the fourth and fifth phase the cluster leader confirms its dominance with energy constraints.

Energy level computation phase:

In this phase base station takes information about the energy levels of each node and computes a set of nodes with residual energy level. In the past research, the cluster leader selection was based on probabilistic models and random selection **[6].** The advancement in that gives selection criteria based on the energy level of individual nodes. If the energy level of the node is greater than the average energy of node then that node can be selected as cluster leader. However, this criterion is not sufficient if the energy of all the nodes is less than the average energy. Thus to overcome this limitation the algorithm is modified and new cluster leader selection criteria is given some flexibility. For example, if the energy of node to be selected as cluster leader is less than the average energy then the algorithm will find the node in close proximity of potential cluster leader with higher energy level. It is possible that none of the nodes in the cluster have energy greater than average energy level, in such condition the node with higher energy in the cluster will be selected as cluster leader.

The energy level computation in actual node environment is difficult as data from each node most be sent to the base station. However, at the starting phase of the algorithm selection of cluster leader is not decided and that leads to the confusion of sending data. Thus, either each node has to send data directly to the base station or any node in the cluster can be selected as a cluster leader and given responsibility to send data in the first round. Simulation result shows that random selection of cluster leader for sending data in the first round is more energy efficient. As all the nodes in the cluster are assumed to have an equal or sufficient level of energy while deploying, the initial cycle of sending data to the base station will not face the problem of the dead node.

The base station will receive energy levels of all the nodes in the cluster and will compute sets of nodes which are called as potential nodes. Let E_a is available energy at the node, E_m is the average energy of the nodes in the cluster and E_{mr} is average energy after ‘r’ rounds. The base station will compute a set of potential nodes which satisfies the condition $E_a - E_m >= r * E_{mr}$.**[6].** The selection of the value of ‘r’ is crucial and can be decided by observing simulation results. If the value of ‘r’ is less then it will be of no effect as it will give same results as that of the previous protocols and efficiency will not improve much. However, if the value ‘r’ is large then we will have fewer numbers of nodes available which will satisfy the given condition. So the value of ‘r’ should be decided as to get maximum efficiency and a good number of cluster leaders.

It is obvious that as the number of rounds increases the energy levels of nodes will go on decreasing and we will get less value of average energy level, so the value of r is decided as (n-1)/m where ‘n’ is the number of rounds and ‘m’ is the number of nodes. This formula provides the effective scaling to the algorithm as it considers the number of nodes in the cluster and the exact value of current round. Thus, the nodes which satisfy the given condition are grouped in a set and that set is used to decide the cluster leader when node energy level is less than the required value. The information is then broadcasted to the cluster so that the nodes with required credential can be elected as cluster leader.

Initialization of cluster:

In this phase, the node can be in transmission mode, reception mode or in idle mode. Initially, each node will set up a timer for transmitting data and that timer value will be proportional to the node number. As the timer value will be proportional to the node number, the timer of nodes will expire at different times and each node will have some time to send data to its neighbors. Similarly, for receiving data each node will again set its timer and will wait for data in its reception phase. To separate between the transmitting and reception phase each node will be in an idle mode where no operation will be performed. Moreover, the node will sense the medium before transmitting data. If the medium is busy then it will set its transmission timer to a different value.

For example, if a node is in its reception mode but the data transfer is not completely done and the transmission timer expires. Then in this situation, the node will not abort the reception and start transmitting instead it will sense that reception is going on and will set the value of its timer for transmission. In transmission phase, each node will generate a packet called intro-packet and send it to its neighbors. The intro-packet will contain information like node number and energy level [fig4]. Each node will maintain a table called heard from the table which will contain the information for its neighbors. The table will have three entries for each node such as node id of neighbor, its energy level and information about connectivity. The connectivity information will include knowledge about direct or indirect connectivity. This connectivity field is used in exchange phase.

Contention between nodes:

This is a most important phase in the algorithm where the most dominant node will be selected as potential cluster leader. The important criterion for deciding potential cluster leader is the number of neighbors that is the number of nodes in its proximity.The node with a maximum number of neighbors will be elected as potential cluster leader.

Initially, each node will set a timer called as 'set-back' timer which will be proportional to the difference between fixed number 'X' and the number of neighbors'n'. The number 'N' will be same for every node and can be chosen arbitrarily. However, the number of neighboring nodes will be different for each node. So the setback timer will expire early for the node with maximum neighbors. The intention behind selecting the node with maximum neighbors is to select a node with maximum knowledge of the network. After the timer expires the node will send a packet called contention packet which will indicate that the node with a maximum number of nodes has been selected. After receiving the contention packet each node will turn off its setback timer. It is possible that two nodes can have the same number of neighbors, so theoretically their timers should expire at the same time. However, practically the difference between the clocks of the node is set in such a way that even after their timers expire simultaneously there will be some very small difference in actual timer implementation. So the packets will not be sent at the same time which will avoid a collision of packets.

After receiving the contention packet nodes will wait for some random time and will broadcast the packet in the network. The advantage of broadcasting is that the nodes which are not directly connected to the potential cluster leader will get to know that the cluster leader has been elected and they have to stop their setback timer. Let's consider the fig. in these figure node 5 is not directly connected to the potential cluster leader so it will not have the information about the election of cluster leader. When the intermediate node, in this case node 2, will broadcast the contention packet and it will reach to the node 5 and in this way node 5 will get an idea about the cluster leader selection. Thus, by broadcasting the contention packet in cluster the nodes will declare the dominance of potential cluster leader.

Exchange of Information

In exchange phase, the nodes which are not directly connected to the potential cluster leader will send their information to the cluster leader. However, for sending this information the intermediate nodes must have to participate in the exchange phase. If all nodes start involving in the exchange phase then unnecessary communication overhead will increase which will lead to increased convergence period. Thus, in exchange phase, we have to put a limitation on which node will send information and which node will not participate in this phase. The contention packet will contain the information about nodes which are directly connected to potential cluster leader. After receiving the contention packet, the node will check whether it has some extra information about the cluster**[1]**. If it has some information which is not present at cluster head then it will participate in exchange phase.Moreover, the nodes which are not directly connected to the cluster leader will also participate in the exchange phase. However, the potential cluster leader will not participate in this mode and will remain in the receiving mode for the entire phase.

Initially, after receiving the contention packet the node will check its heard from the table and will look for node numbers. It will compare the node numbers in contention packet to the node numbers in its heard from table. If it finds some extra nodes then it will conclude that it has some extra information and will decide to participate in exchange phase**[5]**. Firstly, the node will check whether the extra node number is directly connected to the given node. If it is connected directly then node will set the receiving timer for some specific amount of time which is constant for each and not dependent on node number or any other parameter. However, if the extra node is not directly connected to the given node then it will set the timer for more time and this timer period now will be dependent on the number of hops it requires to reach the extra node.

It can be observed that if the setback timer in this phase is dependent on the number of hops for indirectly connected node then the convergence time will increase drastically. To manage such scenario the node will find the shortest path to reach to the potential cluster leader and that will minimize the convergence time. In some algorithms which are defined for dynamic cluster topology, the cluster leader selection algorithm will run twice to confirm the dominance of potential cluster leader. Thus, in exchange phase every indirectly connected node will send its information to the potential cluster leader.

Termination & cluster leader confirmation

The termination phase will finalize the cluster leader selection and will make cluster leader as a representative of the cluster. As the cluster leader now has all the required information about the cluster and is ready to send the data packets to the base station, we need to check whether the selected node has sufficient energy to send this data packets. Termination phase is a very important phase in the algorithm as it decides the final decision for the cluster. There are different parameters given in termination phase which guarantees the selection of a robust node as cluster leader. In dynamic topology, the termination phase has special importance as it decides the parameters for the second round. As the cluster topology is expected to change for different time phases, it is reasonable to reconfirm that the cluster leader has sufficient knowledge about the cluster leader**[5]**. Running the algorithm will consume more time but this can be minimized by modifying the contention phase in algorithm. For dynamic topology, the contention phase for second round is designed to converge faster than first round so that the delay in the network can be minimized.

Initially, the selected cluster leader from the above phases will be checked for energy requirements in termination phase if the node meets the requirements then it will be declared as final cluster leader. However, if the node does not have sufficient energy that is its energy level is less than the expected level then another node in its close proximity will be selected. For the initial rounds, each node will have sufficient energy level and selection of cluster leader will not face any problem in termination phase. However, as the network time goes on increasing the energy of nodes will decrease and some nodes which have been selected as cluster leader will loose more energy. It is important to consider energy levels of nodes after some rounds so that the selected node will not die in the process of sending packets to the base station. If the Cluster leader does not have sufficient energy then the node in its proximity will be selected and checked for energy requirement. All the energy constraints will be applicable for the alternative node. Selecting node in close proximity will reduce the convergence time. As the node with sufficient energy level and information content is discovered it will be declared as cluster leader.

Worst Case Scenario and its Solution using Probabilistic Model:

In the above-discussed model, as the lifetime of the network increases the node will start loosing energy and specifically the nodes which reported as cluster leader will loose more energy. It is important to keep the nodes alive which are responsible for connecting more nodes to cluster leader so that the integrity of cluster last longer. In above-discussed model prevention is taken for not selecting a single node cluster leader multiple times instead the neighbor of node can get to be the cluster leader**[5]**. However the approach is useful till the energy levels of nodes are greater that the residual energy which is defined in the protocol as $E_a - E_m >= r * E_{mr}$. If the node energy is less than the residual energy then the approach needs to have some flexibility in order to produce more nodes for becoming cluster leaders. After the average energy level of nodes decreases the protocol has to find a node with higher energy in the cluster.

For solving this issue probabilistic model can be used to determine the probability that a node will have sufficient energy after ‘r’ rounds **[3]**. In the beginning, the base station will have information about the energy level of each node but as the lifetime goes on increasing the chance of getting updated energy level will depend on the received information. That is if the node dies while sending information to the base station then it cannot compute the residual energies and the algorithm will fail to give exact results. Instead, if the base station knows the probability of node having sufficient energy level it can give a more specific allocation of potential cluster leader members. The probability that a node has the energy level to transfer the packets can be calculated on the basis of its previous chances of getting cluster leader and the participation in the exchange phase. The probability that node will have sufficient energy after ‘r’ rounds is calculated by- $P_{suffE} = P_{CH}*(1- P_{EP})* r$ **[4]**. Where ‘P_{CH}’ is the probability of getting cluster leader and ‘P_{EP}’ is the probability of participation in the exchange phase of the algorithm. So when the energy levels are below residual energy level then the probabilistic approach is used to get potential cluster leader.

Dead Node Condition and Communication Overhead Tradeoff

The probabilistic approach is very useful in finding potential cluster leader in low energy constraints. However, this has some tradeoff in terms of communication overhead and increased latency. In low energy constraints when the actual cluster leader will not have sufficient energy to transfer data to the base station, a secondary cluster in the near proximity of potential cluster leader will be chosen to transmit data. However, it is possible that the secondary node will also not have sufficient energy. So the iteration for next cluster leader will go on until the node with sufficient energy is found. After the node is found the primary cluster leader has to send all the information to secondary cluster leader. Thus, the process of selecting secondary cluster leader and sending data to that node will result in communication overhead and in terms will also increase latency in the network. Consider fig[3], we can say that if the node e does not have sufficient energy then node f can be chosen as cluster leader if it satisfies the energy constraints. Thus, we can say that the robustness in protocol come with the cost of latency in the network.

Simulation Structure and Requirements

Deployment time can be reduced by implementing the algorithm on simulation tool. The above-discussed algorithm types are implemented in a simulation environment such as Mathworks **[1]** using the block simulation tool Simulink. Two important events in the algorithm are sensor node and the communication channel. The nodes are constructed as a block from Simulink Stateflow and the connectivity between the nodes is defined by channel. The algorithm is implemented in the network as well at the base station so a block for the base station is also specified. The implementation of most of the discussed algorithms is carried out by using the ‘C’ programming language so that it can be modified to reuse using different blocks. The sensor node block is consist of sensor nodes which user can define while programming. The channel block consists of properties of the medium and is responsible for connecting different nodes by using block matrix logic. The different phases in algorithm and the timers are defined by inbuilt functions in Simulink.

Implementation of algorithms on complex network:

The algorithm can be implemented on complex cluster formed by deploying nodes on an area called as the area under surveillance. Considering the fig.[3] we can say that the fewer nodes are deployed in a complex manner in order to prove the effectiveness of the algorithm.The algorithm can be considered as robust if it can connect all the nodes to cluster leader and the cluster leader to the base station. Initially, the nodes are spread at random and the first phase will start. Here the first phase will not be energy computation as no node in the network is directly connected to the base station and the energy computation takes place at the base station. So here the first phase will be initialization phase in which all the nodes will send the information they have to each other. The information packet will contain node id and energy level. The second phase here will be contention phase in which the node with a maximum number of neighbors will be selected as potential cluster leader. From the fig [3] we can say that the node 'e' has maximum neighbors that is node a,b,c and d. So node 'e' will be selected as potential cluster leader.

Practically the node e will set a timer called as set back timer which will be proportional to (X-4). Here 'X' is constant number fixed while programming and 4 is the number of neighbors of node e. The same timer will be set at each node but the number of neighbors for others will be different so different timers values will be formed. It can be observed that as node e has maximum nodes connected so its timer value will expire prior to other nodes. When the time value will expire the node will send a packet called as contention packet which will contain the information of all the connected nodes to node e. After the contention phase we can observe that though the node e has a maximum number of nodes connected, it does not have the knowledge of entire topology. For example node e does not know about the node k which is also the part of the topology. Thus, to connect these remote nodes to cluster leader the third phase that is the exchange phase is used. In this phase, all the nodes which received contention packet in the last phase will broadcast the contention packet. In the fig [3] nodes a,b,c and d will broadcast the contention packet to their neighbors. Here node d has neighbor nodes h and g which will receive the contention packet.

We can observe that the nodes h and g were not directly connected to the potential cluster leader but now they will receive the contention packet and will get to know that the selection of cluster leader is decided. After receiving the contention packet this node will also broadcast the packet so that it can reach to other nodes which are in proximity of node h and g. Therefore, the nodes k and f will receive contention packets from one h and g respectively. Node k will send its information to the node h and node h will send the combined information to node d. The heard from table fig[3] for the node d will get updated as the node entries for node k will be added. The heard from entry will be updated to node h for node k so that the node d will know that it can reach node k via node h. The function of heard from the table is to tunnel the direction from one node to other. Similarly, the heard from the entry for node g will be created. Thus, node d will know how to reach to the distant nodes k and g. As the node d is directly connected to cluster leader, it can directly send the information of distant nodes to cluster leader.

In exchange phase every node will get connected to the potential cluster leader via some intermediate nodes. In fig **[3]** node k will connect to cluster leader via node h and node d and node k will send its information to the potential cluster leader. Node e will remain in the reception phase for the entire exchange phase and the remaining nodes will send information to node e. After the exchange phase, node e will have information about each node in the cluster. Node e will take the responsibility to send the data to base station. Base station now will get to know about the energy levels of each node and the connectivity of the network. The base station will take part in the algorithm after the first round of data transfer. The base station will calculate different connectivity parameters such as shortest path and the energy level constraints. In the next round also, the base station will receive updated information about the cluster. After each round base station will broadcast a message called as energy state which will contain updated information about the energy levels of different nodes in the cluster. After receiving the message nodes will consider the energy criteria for selecting cluster leader. So in fig **[3]** it may be possible that node d will become cluster leader after the energy level of node e decreases.

Future Improvements

Flexible election- The algorithm decides cluster leader node based on the number of nodes in its proximity and the energy level of the node. The number of nodes in proximity are the nodes which are directly connected to the cluster leader node. However,in some specific node deployment cases fig**[5],** we can observe that indirect connectivity also plays an important role as the intermediate nodes sends data to the potential cluster leader. Therefore, for such cases, it is possible that the node with maximum neighbors should not be the node with maximum indirect connectivity. Thus, the algorithms have some scope of improvement considering indirect connectivity criterion for cluster leader selection.

Security: Sensor nodes are generally placed in the unattended environment and can be compromised by intruders. Security provisions for completely connected networks are solved with techniques such as symmetric key exchange **[3].** However, in partially connected WSN the security provision is yet to be implemented . The security provisions can be aded in the aggregation algorithm in order to make it robust and secure.

Conclusion:

The algorithm can effectively find robust cluster leader considering energy and connectivity parameters. The low energy states in the cluster are solved by applying probabilistic models for finding a node with available energy. The algorithm can also withstand the extra low energy condition in the cluster but that comes at the cost of latency in the network. However, latency can be tolerated for having an efficient network structure. Thus, the algorithm is best suitable for partially connected wireless sensor networks where not every node can hear each other.

Bibliography

1. Mozumdar Mohammad and Guofang Gregoretti. " **An Efficient Data Aggregation Algorithm for Cluster-Based Sensor Network** " 4, no. 7 (2009): 598-606.

2. Karl, willing. "Protocol and Architecture for Wireless Sensor Networks". (John Willi & sons, 2005,) 109-189.

3.Sandhya, M, K Murugan, and P Devaraj. 2015. Selection of Aggregator Nodes and Elimination of False Data in Wireless Sensor Networks. *Wireless Networks*. 21, no. 4: 1327-1341.

4. K. Singh. "WSN LEACH Based Protocols: A Structural Analysis." Computing and Communication (IEMCON), 2015 International Conference. IEEE 2015.

5.Tawfik Hassan and Rowayda Sadek "A novel energy efficient vice Cluster Head routing protocol in Wireless Sensor Networks" 2015 IEEE Seventh International Conference on Intelligent Computing and Information Systems (ICICIS),IEEE 2015, 313 – 320.

6. Fengling Zhang and Xuandong Li. "Numerical Analysis of WSN Protocol Using Probabilistic Timed Automata". Cyber-Physical Systems (ICCPS), 2012 IEEE/ACM Third International Conference. IEE 2009, 237-245.

Figures:

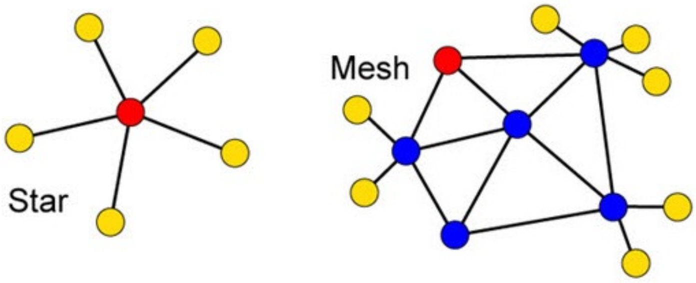


Fig (1)

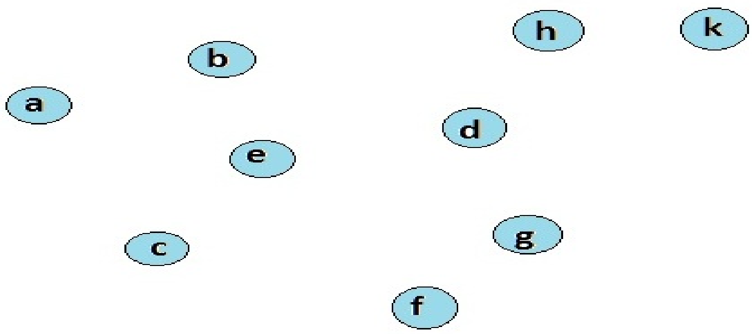


Fig (2)

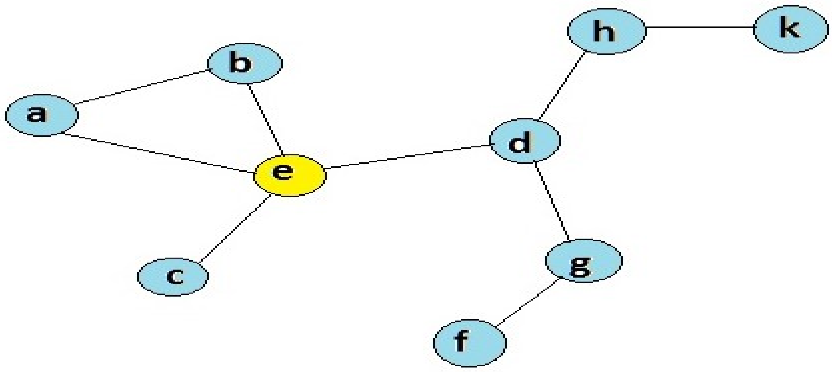
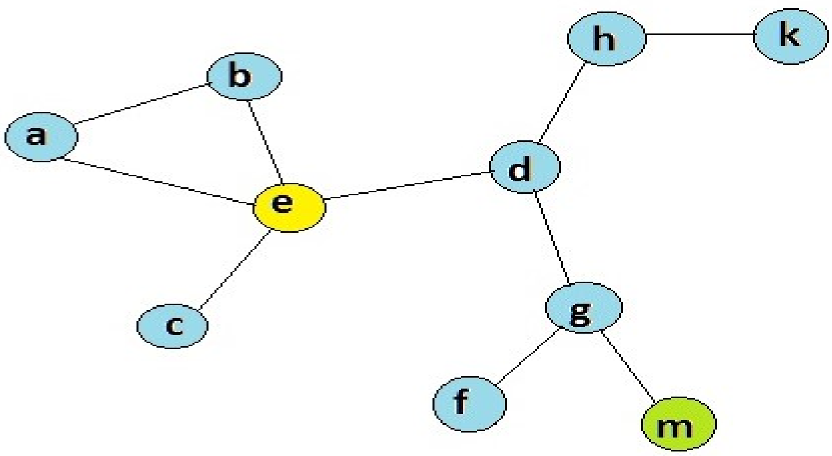


Fig (3)

Heard from	Routed Through	Energy Level
1	empty	12 J
3	empty	12J
4	empty	12J

Fig (4)



Fig(5)