HEAP: Hybrid Energy-efficient Aggregation Protocol for Large Scale Wireless Sensor Networks

Santhosh Simon, K Paulose Jacob

Department of Computer Science, Cochin University of Science & Technology, India

simon.santhosh@gmail.com

Department of Computer Science, Cochin University of Science & Technology, India

kpj0101@gmail.com

ABSTRACT

Wireless sensor networks (WSNs) can be meritoriously used in several application areas like agriculture, military surveillance, environmental monitoring, forest fire detection etc. Since they are used to monitor large geographic areas numerous sensor nodes are to be deployed and their radio range is also very short. Hence they depend on the cooperative effort of these densely deployed sensor nodes for reporting the sensed data. Any changes in environment or an event of interest may be initially observed in a particular area. In other words, they are correlated in space domain. Many nodes in that area may detect the event and report the same event. This redundant information is of no use to system and also depletes the precious energy of the intermediate sensor nodes. Sensor nodes are having very limited energy and needs to be conserved for attaining maximum network life time. Data aggregation is an effective technique for conserving energy by reducing the packet transmissions. Many aggregation systems are available, but when employed for large scale wireless sensor networks they are less effective. In this paper we propose a scheme for large scale WSNs which effectively uses the spatial correlation and temporal correlation of the data for effective aggregation and thereby preserving precious energy.

General Terms

Computer networks, wireless sensor networks.

Indexing terms

Wireless sensor networks, energy optimization, data aggregation, hybrid data aggregation

Academic Discipline And Sub-Disciplines

Computer Science, communication

SUBJECT CLASSIFICATION

Engineering, computer engineering

INTRODUCTION

Wireless sensor Network (WSN) is a major technology used for real time monitoring of environmental assets. WSN has the advantages of large scale deployment, low maintenance, scalability, adaptability, less power needs etc. with the disadvantages of low memory, low power, low bandwidth etc. They can be employed in hostile environments and the features like use of low power and low maintenance makes them the most suited technology for real-time environmental monitoring. Environmental monitoring requires monitoring of large geographic areas and numerous sensor nodes are to be deployed for this. One of the reasons for this is the sensor nodes are independent systems with a sensor module, less powerful processing module and a short range radio module. Hence one to one communication between the nodes and the coordinator in not possible, they depend on the cooperative effort of these densely deployed sensor nodes for reporting the sensed data.

Any changes in environment or an event of interest may be observed in a particular area initially. In other words, they are correlated in space domain. Many nodes in that area may detect the same event and report the same. This redundant information is of no use to system and also depletes the precious energy of the intermediate sensor nodes. The source node which detects the event may be far away from the root and the data packet needs to be forwarded through intermediate node by hopping. If ten nodes are participating in one packet transmission then for ten source nodes hundred nodes needs to be participated for packet transfer. If all these data are of the same event then huge amount of energy is wasted. Sensor nodes are having very limited energy and needs to be conserved for attaining maximum network life time. Data aggregation is an effective technique for conserving energy by reducing the packet transmissions. Through aggregation a node receiving a packet, instead of forwarding the packet immediately, waits for some time and checks for a similar packet from its child nodes. It then aggregates the data as per the defined function and forwards the data to its parent node on the way to the sink node. This waiting time should be set as per the requirements of the system. If the system needs packets without any delay then this waiting should be minimum. This waiting time is introduced for utilizing the temporal correlation of data.

In a large network nodes in the same geographic area may be part of many different clusters. Even if aggregation is employed, if source nodes are part of different clusters they may send the same data to their heads through different paths. Here spatial correlation of data is essential. If nodes in the same area can communicate each other about the event they have observed, without having the boundaries of clusters, aggregation shall be more effective.

Many aggregation systems are available, but when employed for large scale wireless sensor networks they are less effective. These aggregation systems follow either static aggregation or dynamic aggregation. When all the nodes are participating in the packet transfer, as initiated by the coordinator, static aggregation is more useful. But when the nodes report a special incident, which may be limited to small area, all nodes need not be part of the packet transfer. Here dynamic aggregation is more useful. For large scale sensor networks employed for monitoring purposes, a combination of static and dynamic aggregation scheme known as hybrid aggregation shall be effective.

Related Work

In many cases Wireless Sensor Networks are supposed to operate and fetch data from a relatively large geographic area. Scalability and large scale sensing coverage is a main issue to be addressed. Just by using WSN nodes and relaying data through multiple hops to the sink node cannot be visible solution when the network spans a very large area and with thousands of sensor nodes. The effective solution may be to combine the wireless sensor network with other network technologies. "Heterogeneous Sensor Networks" [4] is a similar project of Intel Corporation which uses a two-tier wireless network. They use fixed power supply devices such as 802.11 mesh nodes for improving the performance of the overall network. The project described in [6] uses a two-tier system with WLAN as the second tier. It was the 802.11 protocol, and affects a solid back bone to the entire network. They claim to have a reliable, scalable, energy efficient and cost efficient WSN. Our project aims to deploy large amount of WSN nodes as clusters that converge to a wireless mesh network. The management of multiple sensor clusters involves large data volume and requires complex data gathering and management strategies. The layout of the WSN architecture is high importance while developing an efficient data gathering and management strategy. Many of the research work for data aggregation have been done considering a flat sensor network. The research depicted in [7] uses a simple tree based topology. It does not require a complex protocol and is easy to implement and maintain. The drawback is high transmission delay while waiting for data aggregation and low aggregation efficiency.

Scalable and Unified Management And Control (SUMAC)[1] is a large scale wireless sensor network architecture that uses a mesh network as a bridge between geographically dispensed sensor cluster and Internet. The data aggregation policies of SUMAC is variable when an event of interest occurs sensors can automatically stick down the aggregation level. Low aggregation or nil aggregation means the raw data is send from the sensor nodes to server. In other situations the aggregation level can be set to medium or high for better energy efficiency. Nodes communicate their energy level, delay, distance and buffer size with their intermediate neighbors. This enables the nodes to consider the energy cost while path generation. It uses a graphical user interface to control the aggregation between static and dynamic. Through the GUI can control the threshold values of the sensor nodes. The addressing scheme employed in SUMAC enables the server to establish a one to one communication with any node in the network. This is advantageous in node reconfiguration. The feedback mechanism is also employed in the SUMAC for better fairness.

Directed diffusion [2] is a simple periodic data collection protocol for sensor networks. In this the sink node sends interest message for relevant data to all its sensor nodes. This interest propagation is through flooding and the data messages from source node reaches the sink through multiple path. Over a period of time the sink reinforces these paths and the total number of relaying nodes is reduced. A node that belongs to many data path is the best place for data aggregation.

Clustered diffusion with dynamic data aggregation (CLUDDA) [8] is a method which improves network efficiency by combing clustering with Directed Diffusion. In this the packets have not only the query but also the entire definition of the query. This allows the nodes to breakdown the query to fundamental parts and nodes can answer them separately. The aggregating nodes can combine the query responses from their child nodes can report to the sink node. Optimal Clustering Algorithm based on target Reconfiguration (OCABTR) [10] collects data periodically for reducing the transmission overload and energy consumption. The sensors that detect an event may reside in many different clusters and report separately. This increases the data redundancy and reduces the network efficiency. OCABTR uses generic algorithm to address this issue by separating these nodes into a different cluster. Data aggregation based on dynamic routing DABDR [15] is also a cluster based aggregation protocol. It creates a tree structure and considers the direction of data flow and prefers to pass through a path with nodes containing a long queue of similar data packets. Fault-tolerant Energy Data Aggregation (FEDA) [17] is an in-network data aggregation approach which achieves ideal energy consumption by limiting a number of redundant and unnecessary responses from the sensor nodes. It also gives importance to reliability of the network by ensuring higher chance of receiving data packets at the destination and cause more accurate results. FEDA uses a tree structure and during the initial phase of tree building each node select a node as its parent when it receives the control packet from it. When it receives a subsequent control packet it checks for the cost of the node. Even if it is not a low cost route it will not discard the message but it accepts the node as its backup parent. The backup parent plays a very important role when the packet is lost at the parent node. This is achieves through the over hearing of the packet by the backup parent.

Woo-Sung Jung et al [18] proposes a hybrid approach for clustering-based data aggregation in wireless sensor networks. The HEAP aggregation scheme can adaptively choose a suitable technique depending on the status of the network, increasing the data aggregation efficiency as well as decreasing the energy consumption and ensuring data transmission ratio. As stated by them neither static nor dynamic method is suitable for target tracking operations. They use

a combined method in which static clustering based aggregation is used when there are multiple targets and data traffic is high. It adaptively switches to dynamic clustering aggregation when the data traffic is low.

Tiny Aggregation Approach (TAA) [19] is considered as one of the most energy efficient data aggregation approach. It is also a dynamic data aggregation method where each nodes epoch or time duration is divided into time slots. Nodes at different levels are associated with different time slots and each level can send the packet only at the allocated time slot. If the data from a child node is not received in time the node cannot wait for the data and the entire data routed through that child node which arrives later is wasted. Lut ful karim et al [9] proposes an efficient data aggregation scheme for large scale WSNs that considers the tradeoff between energy efficiency and end-to-end delay. They claims to achieve better performance than the existing standard data aggregation approach SUMAC. The network architecture consist of three overlay networks namely Sensor network, Wi-Fi and Wi-Max. The sensor network is organized as a number of small zones. Each zone has one or more active and several alternative nodes. The Wi-Fi nodes acts as an interface between each sensor cluster and mesh network. The nodes of the infrastructure network plane, (Wi-Max or GPRS) are responsible for collecting and aggregating the data coming from the Wi-Fi nodes. The data aggregation scheme is a dynamic in nature. The queries from the sensor can be broken into fundamental units and aggregation rules can be remotely changed by the user. TDMA scheme is used for collecting data from each level.

HEAP DESIGN

In the HEAP aggregation scheme the WSN nodes are organized as a number of zones. Considering the large coverage area of our application each zone may contain hundreds of WSN nodes. These nodes communicate to cluster head. Each zone will be having a cluster head which is a resource rich node. These nodes will have more processing and storage capabilities and power. These cluster head nodes are GPRS enabled and can communicate to a remote server directly. Each sensor node senses the data and relays it to its cluster head through a hop by hop fashion. Since the nodes nearer to the cluster head needs to relay all the messages that are transmitted from far away nodes their energy sources gets depleted faster. This may cause failure of the network. In order to avoid this more number of WSN nodes working as relay nodes can be deployed near to the cluster heads.

In our application the WSN nodes are supposed to perform three types of operations.

1) *Periodic monitoring and reporting of sensor data.*

2) *Reporting data when demanded by server.*

3) *Intimating the server when detecting an event.*

The first one is a routine process repeated on a timely basis. The WSN nodes wakes up when the timer expires, fetches the data from the sensors and forward them to the cluster heads. It then activates the timer and enters into sleep mode which consumes very less energy. The wake up intervals is set as per need during the node configuration phase.

The second type of activity is answering the queries demanded by the server. On receiving the request from the server the server nodes even if in sleep node, wakes up fetches the data and forwarded them to the server.

In the third one, when any of the sensor node detects a value higher than the set threshold value in any of the sensor it sends the message to the server.

In the first two processes the data are to be collected from all the sensor nodes in the cluster and nodes to be routed to the cluster head. Since all nodes are participating in the activities data gathering and forwarding is static in nature. Hence in this protocol we are using static clustered data aggregation for these two modes. It can send data quickly and easily to the cluster head with relatively low overhead.

The third mode of operation is highly dynamic in nature. It is not a predefined activity. It is triggered only when any of the nodes senses an unusual event. All nodes detecting such an activity at the same time is a very rare chance. Such an increase in the threshold value will be mostly on a region wise. A small number of nodes in that region may detect such an event and all other nodes in the network may be unaware of this. In other words only a small cluster of nodes which are spatially closer may detect the event. Hence, only those nodes need to forward the data to the cluster head. The data from these nodes can be aggregated and forwarded to the cluster head in a single transmission, there by achieving high energy efficiency. Dynamic clustered data aggregation is best suited in this situation and is employed in the protocol.

Static Data Aggregation

Firstly, the WSN nodes and the cluster head are organized in a tree topology during the network initialization phase. This activity is initiated at the cluster head as it is being the root node. It broadcast a control message to its one hop neighbors. This control message will have the nodes ID and the *hop_count* variable with an initial value of 0. The neighboring nodes which receives this control message will chose the message sender as its parent and joins the network if it is not a member. It calculates its current distance to the cluster head using the *hop_count* variable. It will be one hop plus the values in the *hop_count* variable. It then increments the *hop_count* variable in the control packet, updates the node ID with its own ID and broadcast the message. Whenever it receives a different control packet its checks for the *hop_count* in the message and calculates the distance to the cluster head. If it senses to be less and the sender node possesses more energy resource then it breaks the link from the current parent and accepts the message sender as its parent node. Otherwise the message will be dropped. This procedure is continued until all the nodes in the network join the tree topology, with the cluster head being the root node.

After a defined period of time the cluster head re-broadcast the control message. All the above described processes shall be repeated since a node receives control messages from a number of nodes in the vincity, it has the option to select a node with more energy and nearer to the cluster head as its parent nodes. The main advantage here is the node selects its parent node based on the distance to cluster head and the current energy level of the parent node. Hence more energy depleted nodes, even if possessing a shorter distance to the cluster head will not be selected as a parent node. This strategy will ensure that energy level of the intermediate node will be depleted on a uniform basis. By deploying more number of WSN nodes in the neighborhood of the cluster head the bottleneck problem near the cluster head shall be avoided. In this way we can ensure that no node will prematurely die off due to energy depletion and thereby ensuring the dependability and robustness of the network.

Static data aggregation scheme is employed here. The leaf nodes sense the data and forward them to their parents. The parent nodes wait for the messages from its child nodes. Once it receives the messages from all its child nodes it aggregates the data based on the selected aggregation function such as MAX, MIN and AVERAGE and sends the data message to its parent node. Each parent node is a child node of its higher level node. This procedure is repeated until the data is received by the cluster head. The cluster head ensures that it has received the message from all its immediate child nodes. When we need the data from all the WSN nodes in the network, this static cluster data aggregation scheme seems to be best suited as it is scalable, quick, possesses relatively low head and robust.

Dynamic Data Aggregation

As discussed earlier, in our application the scheme of dynamic data aggregation can be employed in the third scenario where the nodes detects events and report to the cluster head. Only few nodes may detect the said event. So these event sensing nodes can be considered as a small cluster. The data from the nodes in the cluster can be aggregated and send to the cluster head. The problem we have to face here is that these nodes may belong to different zones and may have different cluster heads, for example, consider situation described in figure 1.

Here the nodes detected the event belongs to different zones and hence aggregation is not possible. The same data reported by their nodes is received by many cluster heads. This redundant data is send to the server. The aggregation efficiency is very low and the wastage of precious energy is high. Hence static clustering process proves to be of less use in this scenario.

We propose a dynamic data aggregation protocol for this situation supporting efficient data aggregation. When a WSN node detects an event of interest, it packetizes the data and prepares to send the data to the cluster head. The same event may have detected by its neighboring nodes also. So instead of sending redundant data and wasting energy, it is better to aggregate and send as a single data packet. The more number of packets gets aggregated higher is the energy efficiency. Also the data needs to be aggregated as close to the source of data for higher efficiency. In our case the events are likely to occur in the nearby premises, so the possibilities of neighboring nodes detecting the same event is high. Hence data can be aggregated within a few numbers of hops. The neighbor nodes may not detect the event at that instant but later it detects the same event and reports it to the cluster head. This is also a form of redundant data. Therefore temporal convergence should also be employed for aggregation. In other words data must meet at the aggregating node at the same time for efficient aggregation. If this is not possible then the aggregation process may be forcefully delayed such that sufficient data for aggregation is received at the node. One of the key issues in the real time monitoring system is how long can a node waits for data arrival before aggregation.

Aggregation scheduling schemes are mostly classified into three categories [25]

1) Simple periodic: This is a simple technique where each node sends aggregated packets in a predefined interval.

2) Periodic Per hop: In this code node wait for packets from all its child nodes and then it aggregates and send the data as a single packet. We employed this method in static route aggregation.

3) Periodic per hop adjusted: It is similar to periodic per hop, but in each node the holding time is calculated based on the position of the node in the tree structure.

Even though these methods are applicable for static data aggregation and for structure oriented routes they cannot be directly employed in dynamic data aggregation. The periodic per hop scheme can be employed with slight modifications. The main criteria to decide is how long the intermediate node can wait for data before aggregation or how long the received data can be delayed for attaining maximum aggregation efficiency ensuring timely delivery of the data. Of course, the Time to Live (TTL) of the packet at any node should be greater than the time to cluster head, otherwise the packet may die off before reaching the destination. After sensing an event the node n formulates a packet with an event ID, node_dist, E_{rem}, Delay D and assign a Time to Live (TTL) for that packet. ID is the packets ID, node_dist is no.of hops to cluster head and E_{rem} is the remaining energy associated with the node. The delay D is the delay for the next wakeup time. It then broadcast the packet which is received by its one including its parent node hop neighbors. If no neighbors are responding for this message before the expiry of the safe time T_{safe}, it then requests its parent node to forward the data packet to the cluster head.

If any of the neighboring nodes had detected the event they also responds to node n with their data packet containing the same event ID, their node_dist, Delay (D) and E_{rem}. The node n then calculates the weight of each node as follows:

NODE_WT = α x(1-node_dist) + β x (E_{rem}) + γ (D)   **(eq 1)**

where **α, β & γ** are user configurable cost function weights. The node with the maximum node_wt will be selected as the superior node. Node n then aggregate the data as per the policy and forwards the data to the superior node with a new TTL after waiting for a period of time T_{seek}. T_{seek} is the time an intermediate node is to wait before aggregating the data. T_{seek} is a user configurable time delay for attaining efficient data aggregation. Tseek is employed so that the aggregating node waits for a possible event detection by a neighboring node. T_{seek} will be less than or equal to the safe time T_{safe}. T_{safe} is the maximum time an intermediate node can hold the data without affecting the on time delivery of the packet to the cluster head.

Once the aggregated packet is received by the superior node it checks the TTL. It then calculates the safe time (T_{safe}) period which it can hold the packet before forwarding.

$$T_{safe} = [TTL-[M \times node\_dist)+(D \times node\_dist)]] + μ$$

$$T_{safe} = [TTL-[(M+D)node\_dist]] + α \qquad \text{(eq 2)}$$

where M is the expected time consumed at each hop to the cluster head, D is the expected delays executed at hop and μ is a constant factor introduced to reserve some time so as to ensure that the end to end dead line would be met.

If T_{safe} is found to be greater than the seek time T_{seek} then there is an opportunity for the superior node to search for a similar event in its neighborhood. It then repeats all the activities performed by the previous node-node n and tries to find a new superior node. If it succeeds in finding a new superior node it aggregates the data and sends the packet with a new TTL. Else it requests its parent node to forward the data packet to the cluster head.

The operations performed by the nodes in the network during dynamic data aggregation as per the HEAP aggregation system can be analyzed according to the following scenario. The network topology and connections shall remain same for the static aggregation and dynamic aggregation until it is re-setted. The following figure Fig 1, projects the working of a section of a large scale wireless sensor network.
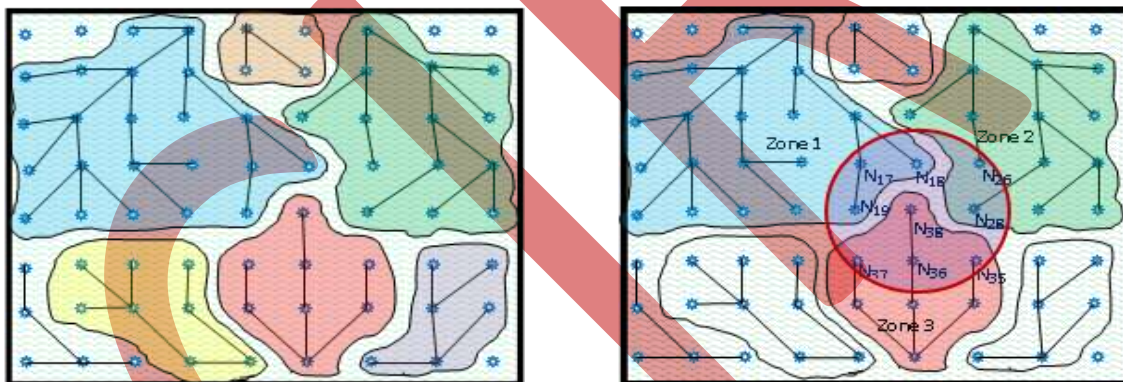

Fig 1(a)   **Fig 1(b)**

When the system gets booted up or restarted the sink nodes commands the cluster heads to establish the network. Each cluster head send the network initialization messages to its neighbors. All sensor nodes receiving this message, joins the network and rebroadcasts the message. Likewise all the nodes join the network through a cluster head which is reached through its neighbors. The network can be observed as a tree structure with the sink node as the root and cluster heads as its immediate branches. Each node is part of any one of the branches. These branches can be considered as zones with leaf nodes as its boundaries. These zones can be seen in the figure Fig 1(a). Nodes which are one hop away may be part of different zone.

In the Fig 1(a) many different zones can be seen. The data from a leaf node reaches the root by hopping through its intermediate nodes. Adjacent nodes may choose a different parent considering the distance and the remaining energy possessed by the parent nodes. Hence nearby nodes may be part of different zones or even children of a different cluster head. In the figure nodes N19 and N8 are one hop away from N38, but they are in three different zones. Sensor networks employed for monitoring purposes may be getting alerts from certain sections of the network. The change in threshold values may not be visible for all the sensors. It may happen only for a small area. Consider the example of the area shown in the above figure with in the red circle. When some criteria like the water level or temperature is increasing in that area the sensors may be triggered and the data needs to be send to the sink node. There are nine sensor nodes in this area, but three are from zone 1, two in zone 2 and four in zone 3. Here static aggregation produces three different messages and reaches its corresponding cluster heads. Through our dynamic aggregation process these messages can be unified and send as a single message. Since the aggregation is performed at the source location itself, it proves to be of much efficient. Suppose node N36 detected this event initially. It the processes the data and after waiting for a period of time it broadcasts the data to the nodes in its radio range, irrespective of the zone divisions. This particular instance is depicted in the figure Fig 1(b).
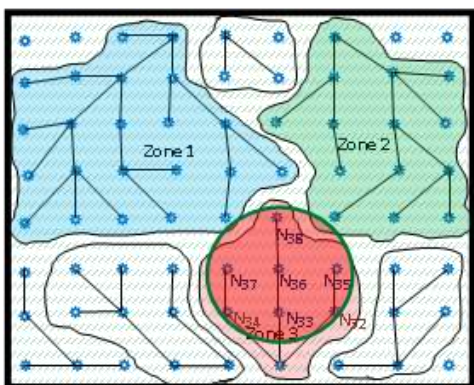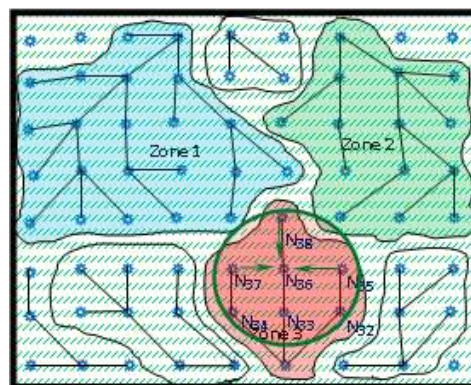
Fig 2(a)

Fig 2(b)

There are six nodes in its radio range, but only three of them namely N35, N37 and N38 have detected similar event and is awake. The remaining nodes are in sleep state and will not receive the message. Active nodes process their data and create a message after appending the remaining energy of that node and the next wakeup time. Node N36 receives these messages and acknowledges for the same. It then aggregates the data and elect the most eligible node to forward the data based on the remaining energy and time to wake up. This data along with the hop count is send to the most eligible node. The figure Fig 2(a) shows the above mentioned process. In this example node N38 is considered as the superior node in this round.Similarly a neighboring node in this region may have detected the same event and broadcast the message to its neighbors in its radio range. Node N18 is such a node and it broadcast the message. The following figure Fig 3(a) depicts the process.
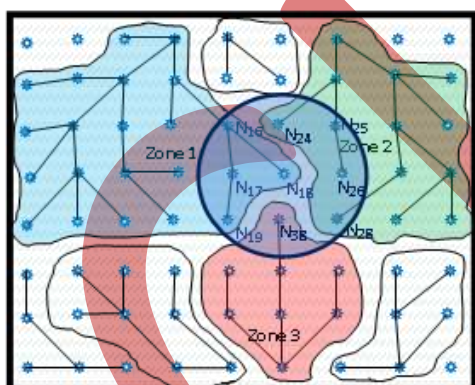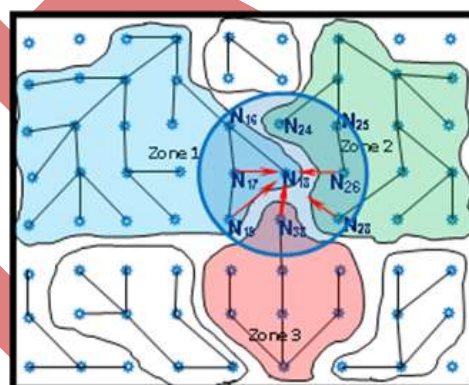

Fig 3(a)

Fig 3(b)

Nodes N17, N19, N26, N28 and N38 receive this message. All other node haven't sensed such an event and is asleep. Node 18 receives the messages from these nodes which also has the aggregated message from node N38. The figure Fig 3 (b) represents the message transfer process.

Node N18 then aggregates the data from all these nodes and repeats the process for finding the eligible node to forward the data packet, if sufficient time is remaining. Here it is holding the data aggregated and forwarded by node N38 which may need to reach the sink node immediately. On evaluation, if there is no sufficient time for more aggregations then the packet will be forwarded to the sink node through the intermediate nodes. If node N18 is finding unavailability of time then it forward the packet to the sink node through its parent node. The below given figure Fig 4. illustrates the message transfer.
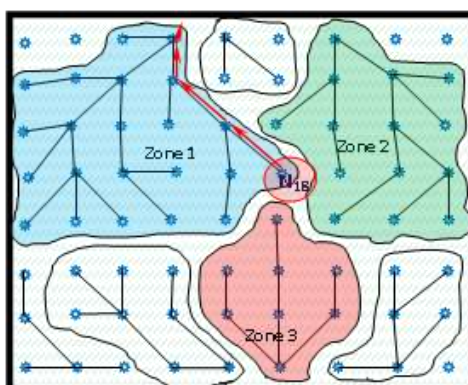

Fig 4.

The main advantage of this protocol is that only necessary nodes will participate in the data aggregation process. The control packet overhead is also relatively low because frequent clustering and election is not necessary furthermore, since all the data aggregation is made near the vincity of the event, aggregation is very close to the source. It also considers the energy level, delay and distance for selecting a node for forwarding the data. So this data aggregation method is scalable and dependable. The pseudo-code for the HEAP aggregation dynamic aggregation is shown below.

Algorithm 1 : Dynamic data aggregation in *HEAP*

1: **repeat**

2:    **if** *Physical sensors in* **node n** *detects an event of interest* **then**

3:    *wakeup the node, switches ON the radio and listen for incoming packet*

4:    **node n** *computes remaining energy E_{rem}, Distance to sink and delay for next wakeup **D***

5:    *after waiting for a time* **t***, and no response from neighbors, it broadcast the packet.*

6:    *wait for response from neighbors*

7:    **if** *any node responds with data* **then**

8:      **If** *not aggregated data* **then**

9:        *compute* **NODE_WT** *for each responded node using* **eq 1**

10:        **SUPERIOR_NODE** = { *n | n is possessing MAX* **NODE_WT** }

11:        *Perform aggregation (MAX, MIN, AVG) and packetize data*

12:        *forward packet to* **SUPERIOR_NODE**

13:        *switch OFF radio and go to sleep mode*

14:      **else** *compute T_{safe} using* **eq2**

15:        **if** T_{safe} > T_{seek} **then** goto step 5

16:          **else** *route packet to sink through parent node*

17:          *switch OFF radio and go to sleep mode*

18: ***until*** (**DYN_AGG** = 1)

PERFORMANCE EVALUATION

The performance of HEAP dynamic aggrigation system was simulated using network simulator-2(NS-2). For comparison purposes, we implemented SUMAC also. The main criteria for evaluation were the energy consumption of the nodes in the network. SUMAC is considered to be a potential system for efficient data aggregation. It was implemented for deploying in large scale networks.

Simulation Environment

For the simulation of the HEAP aggregation system, we used version 2.29 of the network simulator-2(NS-2). For an initial set up a static network with 49 sensor nodes was used. Three nodes were deployed as cluster heads and five different zones were created. The transmission power receiving power, transmission range, carrier sensing range etc. was set as per the default parameters in NS-2. The HEAP dynamic aggregation system was set to operate on dynamic data aggregation phase and the results were evaluated. Constant bit rate packets were used for simulating the data traffic and the packet size was 100 bytes.

Result Analysis

The HEAP aggregation system and SUMAC networks were implemented in the simulation environment as per the network parameter specifications. The HEAP aggregation system was designed to increase the energy efficiency along with satisfying reliable communication without much delay. The energy consumption of the network is the total energy spent by the nodes for transmitting, receiving and aggregating data. An initial energy of 50 Joules was assigned to each node. After each round of operation the remaining energy in each node was assessed to calculate the average remaining energy of the network. The graph showing the energy consumption of the network is shown in figure 5.
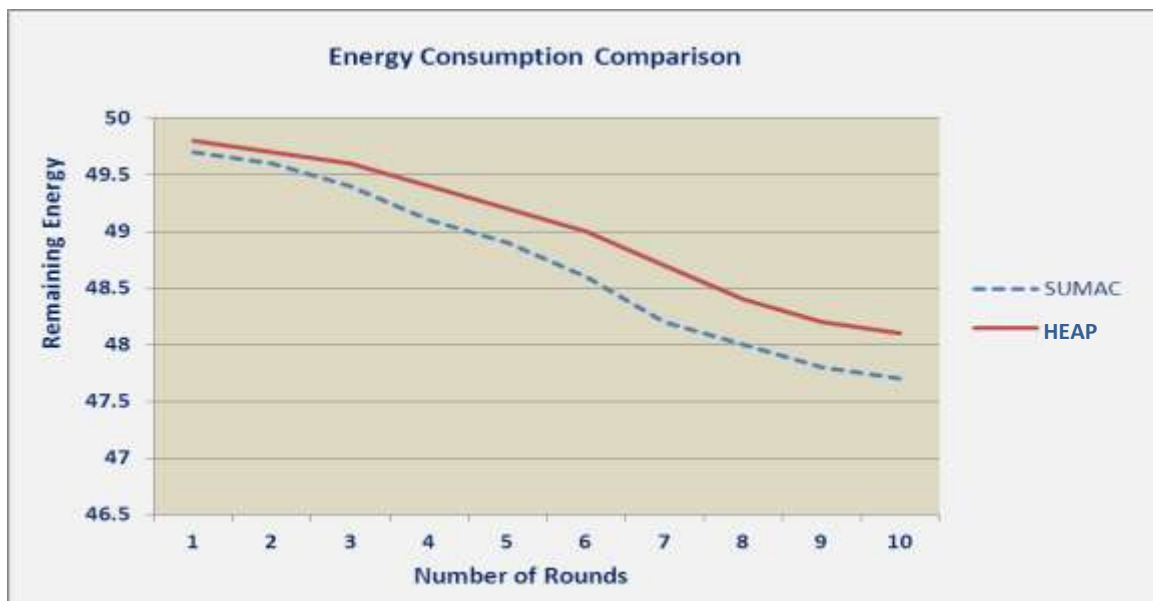
Fig 5 : Comparison of network life time

Since the HEAP aggregation protocol is aggregating the data as close to the source and aggregating data from maximum nodes the energy consumption is found to be low in comparison with SUMAC. The system is designed for large scale sensor networks and will be more effective in conserving energy as the network becomes large.

CONCLUSION

In this paper a new reliable energy efficient duty cycle protocol called HEAP aggregation is presented. The protocol is designed to be employed in large sensor networks employed for monitoring purposes. These networks are mainly used for detecting and informing any environmental or other similar changes. Since the network is very large, many nodes may participate in transferring a message from source to the sink node. Without aggregation more energy needs to be spent for message transfer. Also, these networks may have number of head nodes and clusters. A particular region where the data of interest is triggered may be part of different clusters. In such cases many messages with same data may be sent to the sink node through different paths, thereby dissipating the energy of many intermediate nodes. The major energy draining system in a sensor node is its radio module; hence restricting the radio transfer to the maximum will provide maximum energy efficiency. The HEAP aggregation system seems to be promising in minimizing the radio operations and thereby attaining better energy efficient without compromising on delay and throughput.

REFERENCES

[1] R. Jurdak, A. Nafaa and A. Barbirator, "Large Scale Environmental Monitoring through Integration of Sensor and Mesh Networks", MDPI Sensors Journal, 2008, Basel, Switzerland

[2] Intanagonwiwat, C.; Govindan, R.; Estrin, D. Directed diffusion: A scalable and robust communication paradigm for sensor networks. In Proceedings of the International Conference on Mobile Computing and Networking, 2000; pp. 56–67.

[3] Intanagonwiwat, C.; Estrin, D.; Govindan, R.; Heidenmann, J. Impact of network density on data aggregation in wireless sensor networks. In Proceedings of the International Conference on Distributed Computing Systems, 2002.

[4] Intel Research. Heterogeneous Sensor Networks: Exploring Way to improve Network Performance.Available from http://www.intel.com/research/exploratory/heterogeneous.html.

[5] Leal, J.; Cunha, A.; Alves, M.; Koubaa, A. On a IEEE 802.15.4/ZigBee to IEEE 802.11 Gateway for the ART-WiSe Architecture. In Proceedings of the IEEE Conference on Emerging Technologies & Factory Automation, 2007.

[6] Koubaa, A.; Alves, M. A two-tiered architecture for Real-Time Communications in Large Scale Wireless Sensor Networks: Research Challenges. In Proceedings of the 17th Euromicro Conference on Real-Time Systems, 2005.

[7] Zhang Jiao, Ren Fengyuan, He Tao and Lin Chuang, "Data Aggregation Protocol Based on Dynamic Routing in Wireless Sensor Networks," Communications and Mobile Computing, 2009. vol. 1, pp. 501-505

[8] S. Chatterjea and P. Havinga, "A Dynamic Data Aggregation Scheme for Wireless Sensor Networks". In ProRISC 2003, 14th Workshop on Circuits, Systems and Signal Processing, 26-27 Nov 2003, Veldhoven, Netherlands.

[9] Lutful Karim, Nidal Nasser, Hanady Abdulsalam and Imad Moukadem : "An Efficient Data Aggregation Approach for Large Scale Wireless Sensor Networks", IEEE Globecom 2010 proceedings

[10] Ying Liang and Hongwei Gao, "An energy-efficient clustering algorithm for data gathering and aggregation in sensor networks," Industrial Electronics and Applications. ICIEA 2009. 4th IEEE Conference on, pp. 3935-3939, 2009.

[11] Liang, C.K.; Lin, J.D.; Li, C.S. Steiner Points Routing Protocol for Wireless Sensor Networks. In 5th International Conference on Future Information Technology (FutureTech), Busan, Korea, 2010.

[12] Narang, S.K.; Shen, G.; Ortega, A. Unidirectional graph-based wavelet transforms for efficient data gathering in sensor networks. In IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP), USA, March, 2010 .

[13] Bian, Q,; Zhang, Y.; Zhao, Y. Research on Clustering Routing Algorithms in Wireless Sensor Networks. In International Conference on Intelligent Computation Technology and Automation, Changsha, China, 2010

[14] Motegi, S.; Yoshihara, K.; Horiuchi, H. DAG Based In-Network Aggregation for Sensor Network Monitoring. In IEEE SAINT 2006. Phoenix, AZ, US, Jan, 2006.

[15] Zhang Jiao, Ren Fengyuan, He Tao and Lin Chuang, "Data Aggregation Protocol Based on Dynamic Routing in Wireless Sensor Networks," Communications and Mobile Computing, 2009. vol. 1, pp. 501-505

[16] Asim, M.; Mokhtar, H.; Merabti, M. A Cellular Approach to Fault Detection and Recovery in Wireless Sensor Networks. Third international conference on sensor technologies andapplications, Greece, , Jun, 2009.

[17] M. H. Anisi, J. Rezazadeh and M. Dehghan, "FEDA: Fault-tolerant Energy-Efficient Data Aggregation in wireless sensor networks," Software, Telecommunications and Computer Networks, 2008. SoftCOM 2008. 16th International Conference on, pp. 188-192, 2008.

[18] Woo-Sung Jung, Keun-Woo Lim, Young-Bae Ko and Sang-Joon Park, "A Hybrid Approach for Clustering-Based Data Aggregation in Wireless Sensor Networks," Digital Society, 2009. ICDS '09. Third International Conference on, pp. 112-117, 2009.

[19] Hassanein, H., Luo, J.: "Reliable Energy Aware Routing in Wireless Sensor Networks," In Second IEEE Workshop on Dependability and Security in Sensor Networks and Systems, pp. 54–64, 2006.

Author' biography with Photo

Santhosh Simon has been working as a Lecturer of Department of Computer Science at St.Thomas College of Engineering and Technology, Kerala, India and a research student of Cochin University of Science and Technology. He has acquired BTech and MTech degree in Computer Science. He has presented research papers in several National and International conferences. His research interests are in Artificilal Intelligence, Sensor Networks and Robotics. Santhosh Simon is a Professional Member of ISTE and IEEE.

K Poulose Jacob Professor of Computer Science at Cochin University of Science and Technology since 1994, is currently Director of the School of Computer Science Studies there. A National Merit Scholar all through, Dr. Jacob has been teaching at the Cochin University since 1980. He has presented research papers in several International Conferences in Europe, USA, and other countries. Dr. K.Poulose Jacob is a Permanent Professional Member of the ACM and a Life Member of the CSI. His research interests are in Information Systems Engineering, Intelligent Architectures and Networks