An Energy Efficient Hybrid Node Scheduling Scheme in Cluster Based Wireless Sensor Networks

Sounak Paul, *Member, IAENG*, Naveen Kumar Sao

Abstract— **Cluster based architecture is an effective architecture for data collection in wireless sensor network. However if network consists of some mobile nodes, it becomes difficult to design an energy efficient routing protocol because of frequent changes in topology. In this paper we consider mobility of nodes while constructing clusters. We have used modulo based technique for electing cluster heads. We identify the nodes which give redundant coverage in a round and put them into sleep to conserve energy, without affecting network coverage and connectivity. This is followed by selecting node far away from cluster head to sleep with higher probability. We analyze the energy consumption and the functional lifetime of network of the proposed scheme and compared with some of the established sleep scheduling schemes. Simulation results show that our scheme yields better and balanced energy savings while maintaining an equivalent sensing coverage and connectivity.**

Index Terms— **Sensor Network, Cluster, Energy, Sleep Scheduling, coverage, distance.**

I. INTRODUCTION

Distributed wireless sensor networks (WSNs) have been increasing in popularity for a wide range of applications [1]. A sensor node is composed of typically four units. Sensing unit- sense the desired data from the interested region, the memory unit - store the data until it is sent for further use, computation unit- computes the aggregated data, and the power unit provides the power supply for entire process. Efficient use of energy in the wireless sensor network is one of the most challenging tasks for the user of the sensor networks. Usually sensor devices are small and inexpensive so they can be produced and deployed in large numbers, but their resources of energy, memory, computational speed and bandwidth has several constraints. Therefore, it is important to design sensor networks aiming to maximize their life expectancy. Energy of a sensor node is consumed for sensing coverage area of that node, to send data from one node to another node or to a cluster head, to receive data from other nodes.

In cluster based model cluster head uses some energy for data aggregation, computation and transmission to the base station.

Our proposed work is based on hierarchical cluster based homogeneous wireless sensor network model. The sensor nodes are virtually grouped into clusters and cluster head may be chosen according to some pre defined algorithm. Clustering architecture provides a convenient framework for resource management, such as channel access for cluster member, data aggregation, power control, routing, code separation, and local decision making [9]. The aim of our proposed work is to dynamically balancing energy consumption and to enhance the functional lifetime of the network by dynamically scheduling a percentage of nodes to go for a sleep in each round. Functional life time of the network normally refer to the duration when a percentage of sensors exceeding a threshold (e.g. 80%) have depleted their energy [10]. We have considered a network consist of large number nodes both static and mobile. The node scheduling leads to the following problem: which nodes are to be selected to put into sleep in a round, so that functional lifetime of the network extends, without affecting network coverage and connectivity.

Several researchers have proposed different sleep scheduling algorithms to determine which nodes are to be chosen in a round for sleep to enhance the overall efficiency and functional lifetime of the network. Lightweight deployment-aware scheduling (LDAS) [2] protocol does not require the GPS or other location finding systems to know the exact location of the neighboring nodes. It assumes that each working node has a mechanism to know the number of working nodes in its neighborhood. When the number of working neighbors is more than a threshold determined by the user on sensing coverage, the node randomly selects some of its neighbors to turn off and sends tickets to them. When a node has enough tickets from its neighbors, it may enter the off duty.

Probing Environment and Adaptive Sensing PEAS [3] conserve energy by separating all the working nodes by a minimum distance and check the status of neighboring nodes whether it is working. Each node broadcasts a message (probe) after sleeping for a random period. A node will enter the on-duty mode only if it receives no replies from the working neighbors; otherwise it will continue in the off duty mode.

Coverage Configuration Protocol (CCP) [4] increases the number of sleeping nodes while maintaining the quality of service as K-coverage and K -connectivity. To maintain K-

coverage, a node only checks whether the intersection points inside its sensing area are K-covered.

Adaptive self-configuring sensor network topologies (ASCENT) [5] maintain a certain data delivery ratio while allowing redundant sensors to stay asleep in order to conserve energy.

Low Energy Adaptive Clustering Hierarchy (LEACH) [6] cluster-based protocol uses randomized rotation of the cluster heads, so that every node gets an opportunity to act as a cluster head at least once in its life time. LEACH divides the operations into two phases- set-up phase and steady phase. Set-up phase contains the process of electing the cluster head and each sensor joins a cluster by choosing the cluster head that requires the minimum communication energy. In the steady phase, all nodes send data to their respective cluster head and cluster head aggregates the data from the sensors in its cluster and then transmits compressed data to the base station. In the nth cycle, a sensor that has not become a cluster head during the previous $1/p$ cycles decides to become a cluster head with the probability

$$= \frac{p}{\left(1 - p \times \left(r \bmod \frac{1}{p}\right)\right)} \quad (1)$$

In Linear Distance-based Scheduling (LDS) [8] the authors observed that the nodes, far away from the cluster head needs more energy to send their data to their cluster head. So they deplete their energy before the nodes which are closer to the cluster head. To balance the energy consumption, LDS scheme selects sensors farther away from the cluster head to sleep with higher probabilities. The drawback of LDS is that, if sleep probability is not appropriately according to the distance then it happens that the nodes farther away from cluster head has better lifetime than the nodes which are situated nearer from the cluster head. The same authors proposed balanced scheduling [9] scheme that aims to evenly distribute the energy load of the sensing and communication tasks among all the nodes in the cluster.

In this paper we have proposed a node scheduling scheme to balance the energy consumption among different nodes, so that almost all the nodes die approximately at same time, thus enhancing the functional lifetime of the network. Our scheme is divided into 3 parts:

- Cluster Head election for clusters consist of both static and mobile nodes.
- Nodes which produce redundant coverage in a particular round are put into sleep for that round.
- To further conserve energy, nodes distant away from cluster head are put into sleep mode with higher probability.

We evaluate the efficiency of both our proposed schemes in terms of energy consumption, functional lifetime of the network, and compare with Linear Distance-based Scheduling (LDS) [8] and coverage based scheduling (CS) [13]. We also provide the estimate of optimal number of cluster by attempting to minimize the total amount of energy of the network required for communication.

II. CLUSTER HEAD SELECTION

Though our network model consists of both static and mobile nodes, we have assumed that only static nodes will participate in cluster head election. The logic behind this assumption is that, mobile nodes are already spending more energy than static nodes because of their mobility. Static nodes are numbered from 1 to N. Our algorithm elects the sensor nodes in Round Robin fashion. The approach is implemented in a distributed manner and allows each sensor node to check whether it is a cluster head for this round. Nodes are clock synchronized. If it is the node's turn then only it will claim to become cluster head otherwise, it will simply be a member of cluster. Each sensor node will copy its ID to a variable N_{id} and the value of another variable id will decide the cluster head of the cluster. There are two cases-

i) When (id=k) then it's that node's turn to become cluster head for this round.

ii) When (id $\neq$ k) then sensor node N_{id} will act as a member of the cluster for this round. A round is time interval of two phases in first phase t_1 the cluster head is elected and cluster formation takes place. In second phase t_2 the nodes senses data and transmit data to cluster head for further processing. t_1 is small interval comparer to t_2 transmission phase.

A. Cluster Head Election Algorithm

    N_C: The number of cluster heads in one round
    N: The total number of nodes in sensor field

1. k=0
2. id= N_{id} mod N_C
3. If (id=k) then
    i. N_{id} is a cluster head
    ii. Broadcast an advertisement message to all neighboring nodes
    iii. Wait for some time to receive reply messages from neighboring nodes
    iv. Perform cluster formation
    else
  a. wait a period of time to receive the advertisement messages from cluster head
  b. if (RecMessage =1) then
    Choose it as a cluster head.
  c. Else If (RecMessage>=2) then
    Calculate the shortest distance, choose shortest distance node as a cluster head
  d. Else N_{id} is a forced Cluster head
    End If
    End If
4. After a time interval when a round is completed
    k=k+1;
5. If (k=N_C+1) then
    k=0
6. Go to step 2.
EndIf
End

B. Sleep Scheduling Scheme

In this paper we have studied the performance of a hybrid sleep scheduling. We identify the nodes which give redundant coverage in a round and put them into sleep to

conserve energy, without affecting network coverage and connectivity. This is followed by selecting node far away from cluster head to sleep with higher probability. The probability value is calculated as per LDS [8].

It is assumed that each mobile sensor node is equipped with location finding system. Each mobile node can calculate its speed and direction. When the cluster head has been elected it broadcast a message to all the sensor nodes with their position. To determine which cluster head to join, each non-cluster head node considers the relative position and selects the nearest one. t1 is time interval when the cluster head election ends, t2 when the next round starts. Mobile node calculates distances to all cluster heads at time t1 and t2 respectively. If $DH_j(t)$ be set of relative positions of the mobile node to the cluster head c for time t. At time instance t, when nodes need to transmit data to base station it selects cluster head using distances from cluster heads. Node j simply makes a set of distance from all nodes.

$$DH_j(t) = \forall c \left\{ DH_j(c) \right\} \qquad \text{where c is cluster head} \qquad (2)$$

Nodes Select cluster head with minimum distance from itself for that time interval.

$$DH_j(t) = \min_{J=1}^{C} \left\{ DH_j(c) \right\} \qquad (3)$$

The time instance t is given to determine the cluster head to join for the sensor node. Each sensor S_i for i=1 to N is located on co-ordinate (x_i, y_i) inside the network area where they have same sensing range R_s and communication range R_c. The position of nodes can be found using "HELLO" message communication

Definition: **Sensing region** – A sensing region of the sensor S_i is a set of all points located inside its sensing range .A point is said to be K covered if it is within at least K sensor's sensing node.

Definition: **Redundant Sensor Nodes-** A sensor node S_i is called redundant, if its entire sensing range R_s is covered by at least one neighboring nodes.

Definition: **Overlapping area-** Set of all points which are covered by at least two nodes [13-14].

If a large area of node's sensing area is covered by its neighboring sensors then letting this sensor sleep with a higher probability in the next cycle will save energy without affecting coverage. In Each scheduling cycle a sensor k detects how many sensors are active in its neighborhood and calculates the total area A(k), in its sensing range that is covered by its active neighbors. The overlap area of two neighboring sensors S_1 and S_2 can be calculated by following equation. If two neighboring nodes having sensing range r and distance d then
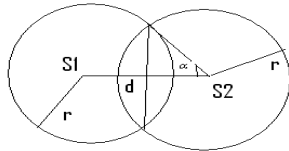
Overlapping area =

$$O(s_1, s_2) = 2r^2 arc \cos \frac{d}{2r} - d \sqrt{r^2 - \frac{d^2}{4}} \qquad (4)$$

If two neighboring nodes have a common overlap area then we subtract their common overlap area from total overlap, if neighboring nodes contains no overlap between

them than above equation can be directly applied to find overlap area [13]. It is complex to calculate overlap area if area is covered by many nodes. A node will be put on sleep depending on overlapping area by neighboring nodes.

$$O(s_1, s_n) \geq \theta(A) \qquad (5)$$

If overlapping area is greater than threshold area it is considered as a redundant node, the node turns itself off into power saving mode up to next round. Here A is sensing area of sensor node. θ is threshold overlap value required to sleep.

Sleeping of the redundant nodes followed by distant scheduling, where nodes far away from cluster head has more probability of sleeping. We have used the calculation of sleeping probabilities of nodes same as LDS [8] . A cluster member uses most of its energy in sensing and transmitting data to cluster head. Transmission energy is proportional to distance between nodes. Hence the nodes farther away from cluster head consume more energy than closer nodes. According to the LDS scheme, the probability with which a sensor node elects to sleep is linear in x. Let this probability be $p(x)$. To keep the analysis tractable, LDS assume $p(x) = Cf(x)$, where C is a constant and $f(x)$ is probability distribution function of the distance x, between sensor and the cluster head.

$p(x)$ has been defined as follows.

$$p(x) = \begin{cases} Cf(x) & \text{for } 0 \leq x \leq x_s \\ 1 & \text{for } x_s \leq x \leq R \end{cases} \qquad (6)$$

R is the maximum transmission range of the cluster. x_s and C has been calculated as per LDS.

Depending on the distance from cluster head to node, node's sleeping probability varies. The node further away from cluster head goes into sleep with higher probability.[8]

Henceforth we will refer our hybrid approach of sleeping Minimum energy scheme (MES).

III. SIMULATION AND ANALYSIS

We have used ns-2 [11] to evaluate our model and compare it with other scheduling techniques. When the cluster head node's energy is depleted, the nodes in the cluster lose communication ability with the Base Station and are essentially "dead" Once the cluster head is selected in a round it broadcasts a message that contains the cluster head ID to each node in that cluster. Cluster head uses CDMA to transmit its data to Base Station. Nodes within a cluster use TDMA to transmit data.

A. Experimental Setup

In our experiments, we assumed the same set of radio hardware energy dissipation model as used in [7]. The parameters used for simulation have been summarized in Table I. The energy required in power attenuation depends on the distance between the node and the receiving Cluster Head [6-10].

$$E_{Tx}(m, d_m) = m \left(eT_c + eT_A d_m^\alpha \right) \qquad (8)$$

The transmitter consumes

$$E_{Rc} m = m \left(eR_c \right)$$

TABLE I.   PARAMETERS USED

Items	Values
Network Area	200 m X 200 m
Base station location	100,100
Number of static sensor nodes	360
Number of mobile sensor nodes	40
Mobility model for mobile nodes	Simple Mobility
Speed of mobile nodes	Between 0 and 1 m/s
Size of packet (in a round)	4000 bits
Period of each round	10 ms
Transmission distance	>30
ID's of sensor nodes	0-399
Expected number of cluster heads	20
Network Simulator Version	Ns-2.34

B. Optimal Number of Cluster Heads

The cluster formation algorithm was created to ensure that the expected number of clusters per round is C (system parameter). We can analyze the optimal value of C in our technique, using the computation and communication energy models. If the network is divided into C clusters, there are on average K/C nodes per cluster. Each cluster head dissipates energy receiving signals from the nodes, aggregating the signals, and transmitting the aggregate signal to the BS. Since the BS is located on distant from the nodes, presumably the energy dissipation follows the multi path model (power loss). Assume that there are K nodes are distributed across M*M region. The energy consumption of a cluster head finishing data gathering and transmission in one round.

$$E_{CH} = \begin{bmatrix} m * \left(\dfrac{K}{C} - 1 \right) * (eR_c + E_{DA}) + m(eR_c) + \\ m(eT_c + eT_A d_m^4) + m(eT_c + eT_A d_m^4) + E_{sens} \end{bmatrix} \quad (9)$$

A common node which is only member of a cluster consumes energy by

$$E_{member} = m\left(eT_c + eT_A d_m^2\right) + E_{sens} \quad (10)$$

Energy consumption of one cluster in one data gathering round is

$$E_{Cluster} = E_{CH} + \left(\dfrac{K}{C} - 1 \right) * E_{member} \approx E_{CH} + \left(\dfrac{K}{C} \right) * E_{member}$$

It is assumed that clusters are formed in a circular form and if the density of nodes is uniform then $eT_A d_m^2 = \dfrac{1}{2\pi} \dfrac{M^2}{C}$ (11)

Energy consumption of network is

$$E_{Total} = \begin{bmatrix} m * \left(\dfrac{K}{C} \right) * (eR_c + E_{DA}) + m\left(eT_c + \dfrac{1}{2\pi} \dfrac{M^2}{C} \right) + m(eR_c) \\ + m\left(eT_c + eT_A d_m^4 \right) + \left(\dfrac{K}{C} \right) \left[m\left(eT_c + \dfrac{1}{2\pi} \dfrac{M^2}{C} \right) \right] + E_{sens} \end{bmatrix} \quad (12)$$

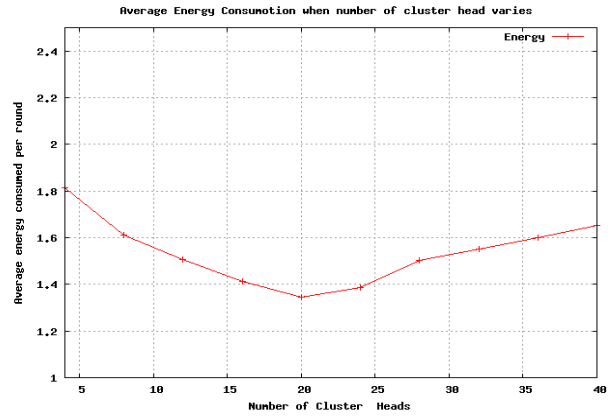We can find the optimum number of clusters by setting the derivative of E_{Total} with respect to C to zero

Figure 1 Average energy dissipated per round for a network of size K and area A, as the number of cluster head varies

d_m^4 Is the maximum distance between Cluster head and Base Station so it can be written as d_{toBS}^2

$$C_{opt} = \sqrt{\dfrac{K}{2\pi}} \sqrt{\dfrac{eT_A}{eR_c}} \dfrac{M}{d_{toBS}^2} \quad (13)$$

For our experiments M=200 meters, K=400 nodes eT_A=10 pJ , eR_c =0.0018 pJ similar values from leach.
And $50 < d_{toBS}^2 < 70$, so we expect the optimal number of cluster heads $16 < C_{opt} < 23$.

C. Energy Consumption

In our experiments, each node begins with 5J of energy and able to send an unlimited amount of data to the BS. Each round lasts for 10 seconds. For these simulations, energy is consumed whenever a node transmits or receives data or performs data aggregation. A node puts itself into sleep mode when its more than 90% coverage area is covered by neighboring nodes. In our simulation we have put θ value to 0.9.
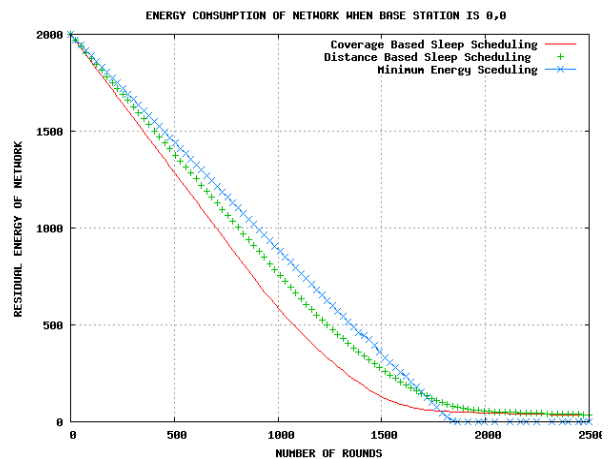

Figure 2 Energy consumption of network, comparison with other energy saving mechanism

Figure 2 presents the total amount of energy consumed in the experimented wireless sensor network, where each sensor network node is assumed to detect simultaneously the same number of events means each node is sensing and

sending equal amount of energy to base station through Cluster heads. We observed that the energy consumption is affected by the location of base station if we put base station on distant from the network field than energy consumption is more than the near base station.

A functional lifetime of a wireless sensor network can be defined as effective sensing coverage and an event can be sensed and outcome of network represents the whole network. From Figure 3 we can see that our scheme outperform other two sleep scheduling techniques in term of functional lifetime
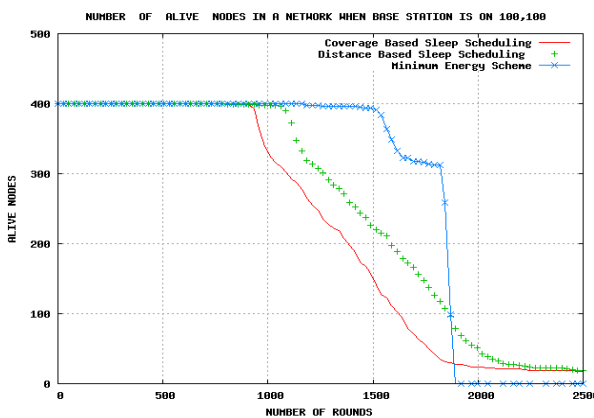

Figure 3 Number of Alive nodes with each node's initial battery power is 5J

In our scheme almost all the node die at same time, increasing the active lifetime of the network. Our scheme achieves complete use of wireless sensor network and it causes balanced energy dissipation. There are some rare cases in which CS and LDS performs slightly better than our scheme. This may happen due to the deployment density of nodes. If network does not contain any redundant region than LDS and minimum energy scheme are equivalent.

IV. CONCLUSION

We have devised an energy saving scheme for homogeneous wireless sensor networks. We studied the problem of sleep scheduling power nodes in a hierarchical network to improve the overall efficiency and network lifetime. We proposed a novel energy saving scheme that enhances the basic schemes used in hierarchical sensor networks. We compared our scheme with two well known sleep scheduling schemes coverage based [13] and linear distance based sleep scheduling schemes [8]. Analytical and simulation results show that our method outperforms other two methods in terms of network residual energy, energy efficiency, network lifetime while maintaining the same degree of network coverage. Our results show that MES extends lifetime of nodes by 1.3 times compared with coverage and 1.2 times better than linear distance based schemes.
In our future work we will explore how the sleeping probabilities should change if nodes have different initial energy. We will also explore how mobility of nodes affects the performance of network. In addition, we plan to explore our cluster formation technique on heterogeneous network.

REFERENCES

[1] Akiyldiz I., Su W., Sankarasubramaniam Y., and Cayirci E., "A survey on sensor Networks," IEEE Commun. Mag., vol. 40, no. 8, pp. 102–114, Aug. 2002

[2] Wu, K., Gao, Y., Li, F., & Xiao, Y, "Lightweight deployment-aware scheduling for wireless sensor networks". ACM/Kluwer Mobile Networks and applications December 2005.

[3] Ye, F., Zhong, G., Cheng, J., Lu, S., & Zhang, L, "PEAS: A robust energy conserving protocol for long-lived sensor networks ". In Proceedings of the 23rd International Conference on Distributed Computing Systems (ICDCS '03) pp. 28–37, 2003.

[4] Wang, X., Xing, G., Zhang, Y., Lu, C., Pless, R., & Gill, C, "Integrated coverage and connectivity configuration in wireless sensor networks". In Proceedings of the 1st International Conference on Embedded Networked Sensor Systems (Sensys '03), pp. 28–39, Los Angeles, California, 2003.

[5] Cerpa, A., & Estrin, D., "ASCENT: Adaptive Self-Configuring sEnsor Networks Topologies". In Proceedings of IEEE INFOCOM 2002, New York, New York June, 2002.

[6] Heinzelman, W. R., Chandrakasan, A., & Balakrishnan, H, "Energy-efficient communication protocols for wireless microsensor networks", In Proceedings of Hawaaian International Conference on Systems Science January, 2000.

[7] Heinzelman, W., Chandrakasan, A., & Balakrishnan, H, "An application-speciffic protocol architecture for wireless microsensor networks", IEEE Transactions on Wireless Communications,1(4),660–670, October 2002.

[8] Deng, J., Han, Y. S., Heinzelman, W. B., & Varshney, P. K. ,"Scheduling sleeping nodes in high density cluster-based sensor networks".In ACM/Kluwer Mobile Networks and Applications, Issue: Volume 10, Number 6 (pp. 825–835), Springer Netherlands December, 2005

[9] Deng, J., Han, Y. S., Heinzelman, W. B., & Varshney, P. K. ,Balanced-energy sleep scheduling scheme for high density cluster-based sensor networks. In Proc. of the 4th workshop on Applications and Services in Wireless Networks(ASWN '04), Boston, Massachusetts, 2004

[10] Paul Sounak, N. Sukumar, Singh I. "A Dynamic Balanced-energy Sleep Scheduling Scheme in Heterogeneous Wireless Sensor Network", 16th IEEE International Conference on Networks New-Delhi, India, December, 2008.

[11] Network Simulator[ns2]: www.isi.edu/nsnam/ns

[12] Network Simulator[ns2]: www.cse.msu.edu/~wangbo1/ns2

[13] Shen F., Sun M., Liu C., and Salazar A., "Coverage-aware Sleep Scheduling for Cluster-based Sensor Networks," Proc. IEEE Wireless Communications and Networking Conf., April 2009.

[14] Tian D. and Georganas N.D., "A Coverage-preserved Node Scheduling scheme for Large Wireless Sensor Networks", Proceedings of First International Workshop on Wireless Sensor Networks and Applications (WSNA'02), Atlanta, USA, September 2002.