



## Adaptive Scheme to Control Power Aware for PDR in Wireless Sensor Networks

<sup>1</sup>R.Sivaranjini, <sup>2</sup>S.Palanivel Rajan,

<sup>1</sup>Assistant Professor, Department of Electronics and Communication Engineering,  
M.Kumarasamy College of Engineering, Karur, Tamilnadu, India.

Email ID : [srsivaranjinir670@gmail.com](mailto:srsivaranjinir670@gmail.com)

<sup>2</sup>Assistant Professor, Department of Electronics and Communication Engineering,  
M.Kumarasamy College of Engineering, Karur, Tamilnadu, India.

Email ID : [palanivelrajanme@gmail.com](mailto:palanivelrajanme@gmail.com)

### ABSTRACT

Nowadays Wireless sensor networks playing vital role in all area. Which is used to sense the environmental monitoring, Temperature, Soil erosion etc. Low data delivery efficiency and high energy consumption are the inherent problems in Wireless Sensor Networks. Finding accurate data is more difficult and also it will lead to more expensive to collect all sensor readings. Clustering and prediction techniques, which exploit spatial and temporal correlation among the sensor data, provide opportunities for reducing the energy consumption of continuous sensor data collection and to achieve network energy efficiency and stability. So as we propose Dynamic scheme for energy consumption and data collection in wireless sensor networks by integrating adaptively enabling/disabling prediction scheme, sleep/awake method with dynamic scheme. Our framework is clustering based. A cluster head represents all sensor nodes within the region and collects data values from them. Our framework is general enough to incorporate many advanced features and we show how sleep/awake scheduling can be applied, which takes our framework approach to designing a practical dynamic algorithm for data aggregation, it avoids the need for rampant node-to-node propagation of aggregates, but rather it uses faster and more efficient cluster-to-cluster propagation. To the best of our knowledge, this is the first work adaptively enabling/disabling prediction scheme with dynamic scheme for clustering-based continuous data collection in sensor networks. When a cluster node fails because of energy depletion we need to choose alternative cluster head for that particular region. It will help to achieve less energy consumption. Our proposed models, analysis, and framework are validated via simulation and comparison with Static Cluster method in order to achieve better energy efficiency and PDR.

### Index Terms

Sensor nodes, Cluster Head, Failure node, algorithm/protocol design, clustering, adaptive scheme.

### 1. INTRODUCTION

Wireless sensor networks (WSNs) have a broad range of applications, such as battlefield surveillance, environmental monitoring, and disaster relief. A sensor network consists of a set of autonomous sensor nodes which spontaneously create spontaneous communication links, and then, collectively perform tasks without help from any central servers [13]. Without error and drop, sending packets from sender to receiver is major role in networks. In sensor networks, accurate data collection is difficult it is often too costly to obtain all sensor readings, as well as not necessary in the sense that the readings themselves only represent samples of the true state of the world [7]. As such, one technique so-called prediction emerges to exploit the temporal correlation of sensor data. Technology trends in recent years have resulted in sensors' increasing processing power and capacity [3]. Implementing more sophisticated distributed algorithms in a sensor network becomes possible. One important class of such algorithms is predictors, which use past input values from the sensors to perform prediction operations. The existence of such prediction capability implies that the sensors do not need to transmit the data values if they differ from a predicted value by less than a certain prespecified threshold, or error bound [1]. A simple approach to developing a predictor in sensor networks is simply to transmit the data from all sensors to the base station (i.e., the sink), which has been realized in many previous studies [16], [24], [5]. Predictor training and prediction operations are carried out by the base station only, but not the sensor nodes, despite their increasing computing capacity. This solution while practical has many disadvantages, such as a high energy consumption incurred by transmitting the raw data to the base station, the need for wireless link bandwidth, and potential high latency.

One solution is clustering-based localized prediction [28], where a cluster head also a sensor node maintains a set of history data of each sensor node within a cluster. We expect the use of localized prediction techniques is highly energy efficient due to the reduced length of routing path for transmitting sensor data. Another method is watch dog method is used to find the selfish node [10-12]. Clustering based local prediction in sensor networks faces a couple of new challenges. First, since the cost of training a predictor is nontrivial, we should carefully investigate the trade-off between communication and computation. To support prediction techniques, energy is consumed on communication and computation (e.g., processing sensor data and calculating a predicted value). Motivated by this observation, we analytically study how to determine whether a prediction technique is beneficial in this paper. We qualitatively derive sufficient conditions for this and reveal that the decision is a function of both the desired error bound and the correlation among the sensor data values. For instance, when the error bound is very tight or the correlation is not significant, a sensor node always has to send its data to the cluster head. The second challenge is due to the characteristics and inherent dynamics of the sensor data. When the data distribution, in particular the data locality, evolves over time, prediction techniques may not work well for a set of less predictable data. Global reclustering is costly if it is initiated periodically. We propose an algorithm for dynamic updates of clustering and the algorithm requires mostly local operations and very low communication cost. This adaptive update of clustering is expected to facilitate clustering-based localized



prediction by maintaining the similarity within clusters at low communication cost. The rest of the paper is organized as follows: In Section 2, we describe related work on prediction and clustering techniques in sensor networks. In Section 3, we describe the models, analysis, and algorithms our framework. Section 4 discusses the implementation issues and describes the application of our framework to the design of more efficient and scalable data aggregation algorithms and sleep/awake scheduling. Section 5 provides a performance comparison of different techniques. Finally, we conclude the paper in Section 6.

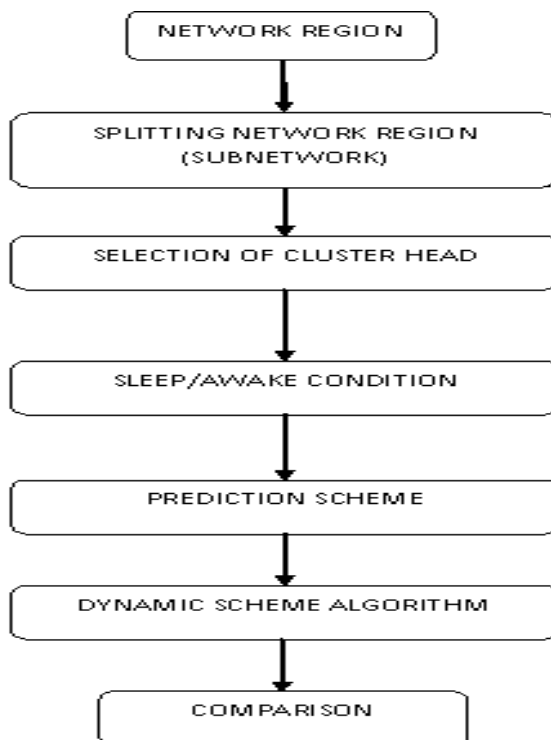


Fig.1. General Block diagram

## 2. RELATED WORK

Energy conservation is crucial to the prolonged lifetime of a sensor network. Many approaches for energy-efficient monitoring have been explored to minimize energy consumption. One class of techniques prediction-based algorithms is based on the observation that the sensors capable of local computation create the possibility of training and using predictors in a distributed way [26], [29], [15], [2], [21].

Taking lessons from MPEG encoding process, Goel and Imielinski [8] proposed a prediction-based monitoring mechanism in sensor network. McConnell and Skillicorn [19] proposed that each sensor transmits to the base station the predicted target class rather than the entire raw data. Chu et al. [25] proposed a robust approximate technique that uses prediction models to minimize communication from sensor nodes to the base station. Likewise, Silberstein et al. [27] proposed the data-driven processing to provide continuous data without continuous reporting. To do this, they developed a suppression strategy that adopts models for optimization of data collection. One alternative approach to selecting representative sensor data is using clustering technique [23]. With clustering, only cluster heads need to communicate with the base station via multihop communication. Several clustering algorithms have been designed with particular attention to energy-efficient query processing. The LEACH protocol [6] is an application-specific clustering protocol, which has been shown to significantly improve the network lifetime. Hussain and Matin [18] extended LEACH to a hierarchical clustering-based routing (HCR) technique. Kuhn et al. [14] proposed a probabilistic technique to select cluster heads in which the probability is dependent on the node degree.

One work that is close to our work is ASAP [4]. They also consider clustering such that nodes with similar sensor data values are assigned to the same clusters. In addition, adaptive data collection and model-based prediction are used to minimize the number of messages used to extract data from the network [20]. However, we emphasize the differences from their work. First, and most importantly, we provide an error-bounded data collection scheme. We are uniquely interested in when the prediction scheme really benefits data collection and have derived solutions after careful energy-aware analysis. Second, [34] focused on clustering and cluster head selection. We present a scheme for dynamic update of clustering. Third, we developed our framework integrating data aggregation to demonstrate the usefulness of our framework. Finally, we must note that the approach is not exclusive. Instead, it can use in conjunction with other techniques, for example, the clustering and cluster head selection algorithms in [9].

### 3. PROPOSED FRAMEWORK

Our framework consists of four main functional components: 1) data processing and intracluster prediction. It is noted that unlike previous dual-prediction techniques, our prediction operation can be enabled/disabled to achieve energy efficiency 2) adaptive cluster split/merge, 3) Sleep/awake scheme and 4) Dynamic scheme for energy efficient. Table 1 lists symbols used in this paper. Fig.1 shows General block diagram of this paper.

#### 3.1 Adaptive Scheme to Enable/Disable Prediction Operations

Consider a cluster of sensor nodes, which can be awake or sleeping. If the sensor nodes are sleeping, the prediction problem is reduced to estimating data distribution parameters using history data. In this case, it could well be the case that the estimates are already available. We can neglect this case. If the sensor nodes are awake, they continuously monitor an attribute  $x$  and generate a data value  $x_t$  at every time instance  $t$ . Without local prediction capability at the cluster head, a sensor node has to send all data values to the cluster head that estimates data distribution accordingly. With local prediction, however, a sensor node can selectively send its data values to the cluster head. One model for selective sending is  $\epsilon$ -loss approximation: Given an error bound  $\epsilon > 0$ , a sensor node sends its value  $x_t$  to the cluster head if  $|x_t - \hat{x}_t| > \epsilon$ . The intuition of this choice is that if a value is close to the predicted value there is not much benefit by reporting it. Node generation and cluster head selection is given in fig.2.

We first develop a localized prediction model. Very complex models are not practical in our application due to the limited computational capacity of sensor nodes. Fortunately, simple linear predictors are sufficient to capture the temporal correlation of realistic sensor data as shown by previous studies [22], [17]. A history based linear predictor is one of popular approaches to predicting the future based on past  $n$  measurements. After the prediction model is formulated, we look at the algorithm selection problem. Without local prediction all sensor nodes will send original data values to the cluster head. This scheme incurs significant communication cost. With local prediction the communication cost is reduced by selectively sending data to the cluster head, but the computation cost can be prohibitive. While a large value of  $m$  often leads to the condition of (1) hardly satisfied and thereby the prediction scheme at the node is prone to being disabled, it can reduce the energy consumption as a long term predictor when the condition is satisfied.

Fig. 3 shows the pseudocode description of the algorithms at the cluster head. The cluster head maintains a set (a circular array) of history data for each cluster member. Lines (08)-(12) show the cluster head will continuously receive data values from each cluster member to update the set of history data, or when no data values are received will use the predicted value instead for update. The cluster head also runs a periodic process, Lines (01)-(06) to determine algorithm selection, with or without local prediction. The decision is broadcast to all cluster members. The cluster head operation is given below with proper lines.

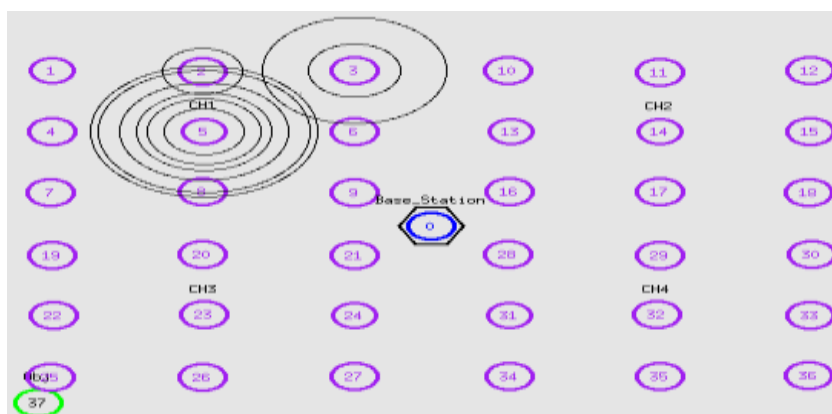


Fig.2. Node generation and cluster head selection.

Fig. 4 shows the pseudocode description of the algorithms at each cluster member. Each cluster member maintains a set of history data of its own. If the algorithm selection is "no local prediction," it simply transmits the data values. If local prediction is turned on, the cluster member will perform prediction on each data value. If the data value is not within the error bound, it will be sent to the cluster head too. Meanwhile, the local set of history data should be updated as well. In particular, if local prediction is enabled and the data value is within the error bound, the predicted value not the actual value will be included in the set of history data.

.....  
**Process at the cluster head**

01: if timeout after  $m \cdot \Delta$  seconds  
02:     for each member  $l$  in the cluster



```
03:     if condition (1) holds
04:     send message to member i to enable prediction
05:     else
06:     send message to member i to disable prediction
07: else
08:     for each member i in this cluster
09:         if receive a data value from member i
10:         update the history data for member i
11:     else
12:     perform prediction to update the history data
```

Fig. 3. Operations at the cluster head.

#### Process at the cluster members

```
01: if prediction is disabled or  $|x_t - \hat{x}_t| > \epsilon$ 
02:     send the data value to the cluster head
03:     Update the history data using the data value
04: else
05:     perform prediction to update the history data
```

Fig. 4. Operations at the cluster members.

## 3.2 Adaptive Update of Clustering

In this section, we present algorithms for the dynamic split and merge of clusters, which require low communication cost. Let  $F_i$  be the feature of node  $i$  and  $F_{CH}$  be the feature of a cluster head. In our case, the feature is defined to be the coefficients of AR model. The similarity between two feature  $F_i$  and  $F_{CH}$  is defined by the distance  $D(F_i, F_{CH})$ . Given a real number  $\delta$ , a  $\delta$ -clustering means for any two nodes  $i$  and  $j$  in a cluster,  $d(F_i, F_j) \leq \delta$ . Since reclustering is expensive, local operations such as split/merge could be performed to avoid global computation. The problem is that the design of such local operations should guarantee that they will not lead to violations of the  $\delta$ -clustering condition. For the algorithm of splitting and merging clusters, we consider two cases. First, if the  $\delta$ -clustering condition is violated, reclustering within the cluster is necessary and the cluster will be split.

Local split requires less communication cost compared to global reclustering. In particular, since in our framework, each cluster head maintains sensor data for each of its cluster members, it has fresh knowledge on the distance  $d(.)$  to the cluster member. Thus, our framework does not incur much communication cost. When a split occurs, say a cluster with head  $CH_i$  is split into multiple clusters with heads  $CH'_1, CH'_2, \dots$ . The new cluster head nearest  $CH_i$  will inherit its position in the routing tree and all other new cluster heads will become its children. In the second case, clusters may be merged. Each cluster head will check whether a pair of children can be merged or whether itself should be merged with a child. Let us say, we need to merge two clusters with  $CH_i$  and  $CH_j$  as cluster heads and assume the first cluster is larger.  $CH_i$  will become the head of the merged cluster (this requires the update of all membership changes, e.g, via a broadcast). In addition, history data of the corresponding nodes will migrate from  $CH_j$  to  $CH_i$ .

## 4. RESULTS AND DISCUSSIONS

### 4.1 Implementation Details

#### 4.1.1 The Scenario with Packet Loss

Failure may not be rare in wireless sensor networks. Clearly, if an update message is lost, that is, the message from Line (04) or (06) in Fig. 2 is lost, the dual prediction (cluster head and cluster member) will not be correct. In that case, each node will perform a different prediction, therefore, leading to a possible misbehavior. This is a key issue that needs to be addressed in applications. One possible solution is by the cluster member replying a small ACK message to cluster head. As for the packets containing sensor readings, as long as the packet loss rate is not significant and approximation is acceptable, the impact of failure could not be crucial. As an example, we found in our previous aggregation work [9] that a small packet loss rate does not have significant impact on the final results. As a result, we focus on the discussion of adaptive scheme to control prediction and adaptive cluster provided justification for not considering packet loss.

#### 4.1.2 Adaptive Update for System Input Changes

We do not claim a set of fixed parameters for the linear predictor and for the value of error bound  $\epsilon$ . In practice, for many applications the model parameters and the error bound could be dynamics after setup. For instance, the system operator may not satisfy an initial error bound  $\epsilon$  and want to adjust it after the system has been set up for a long time. In that case, the cluster head after receiving the updated system input from the sink should re-estimate the model parameters and diffuse to the cluster members. Based on the time cluster head will be changed.

### 4.2 Accommodation with Sleep/Awake Scheduling



To allow sleep/wake scheduling for the cluster members, we replace Lines (01)-(06) in Fig. 3 by Lines (01')- (07') in Fig. 5, and by default, disable local prediction at cluster members. When a cluster member is awake, the cluster head checks if the member's data values are within the error bound with high probability. If yes, the cluster head will send a message to power off the member. The condition should be the confidence level  $\alpha_m$  is higher than the threshold  $\alpha_{\text{threshold}}$ . When the cluster members sleep, the cluster head will not receive any data values, and hence, it is impossible to perform accurate prediction. For this reason, periodic but infrequent collection of data from the cluster members is still necessary. The frequency of this infrequent data collection is due an optimization problem: if the frequency is high, the cost of collecting data can also be high; if the frequency is low, the prediction can be inaccurate and result in erroneous sleeping decisions. In this paper, we provide only a heuristic solution to the problem. Let  $\Delta$  be the time interval between two consecutive reporting by a member. We set the duration of a sleep period to  $m_f * \Delta$ , and when a cluster member wakes up, it will continuously perform data reading (and possibly reporting) for the next  $m * \Delta$  time. Initially,  $m_f$  is set to  $m$ . It can be increased if condition (2) consistently holds, or decreased if the condition does not hold.

```
.....  
Algorithm at the cluster head  
// ...  
// sleep scheduling for members, Lines (01')-(07')  
01':  while member I is awake  
02':      if timeout after  $m * \Delta$  seconds  
03':      if condition (2) holds  
04':          let member i power off for  $m_f * \Delta$  seconds  
05':      while member i is sleeping  
06':          if timeout after  $m_f * \Delta$  seconds  
07':          awake member i  
.....
```

Fig. 5. A variation with sleep/wake scheduling.

### 4.3 Dynamic scheme

We develop our aggregation algorithm based on the dynamic method. In this scheme various sensor node from each network will be monitored for particular time period. In that which sensor node is highly energized and capable to link with all other region will be selected as a cluster head. By using this scheme we can avoid high power consumption [35-39].

## 5. PERFORMANCE EVALUATION

### 5.1 Cluster Model

With our framework, a sensor network is partitioned into multiple sub networks, i.e., clusters [32]. A cluster can be formed by the set of sensor nodes in a geographical area, where data locality exists among the sensor nodes, and clusters are dynamically split/merged to maintain good locality within each cluster. All sensor nodes in a cluster are called cluster members, including one elected cluster head. Within each cluster, the cluster head receives data selectively reported by all cluster members, and performs local prediction on the data distribution of the sensor data. Cluster members also perform prediction, and data values are transmitted to the cluster head only if they are not within a specified error bound. In this way, a cluster head can perceive an accurate view of all sensor data across the cluster, while communication cost is drastically reduced.

### 5.2 Benefits of Adaptive Scheme

When we use dynamic scheme it will provide following advantages over static scheme [33]:

- Avoid high energy consumption
- High throughput
- Good packet delivery ratio
- Zero packet loss

Fig. 6 shows that energy consumption with dynamic scheme. First, it shows that energy consumption is a decreasing for without adaption. While going for with adaption by dynamic scheme, energy consumption reduced. Second, Packet delivery ratio analysis is more beneficial shown in fig.7. We emphasize that while communication is more expensive compared to prediction, the scheme is still applicable due to the existence of other computational operations (e.g., calculating coefficients, maintaining/updating history data, etc) not mentioned in this work. Finally, accommodating with sleep scheduling improves the performance by up to 10 percent, mainly because we set the confidence level threshold at 90 percent.

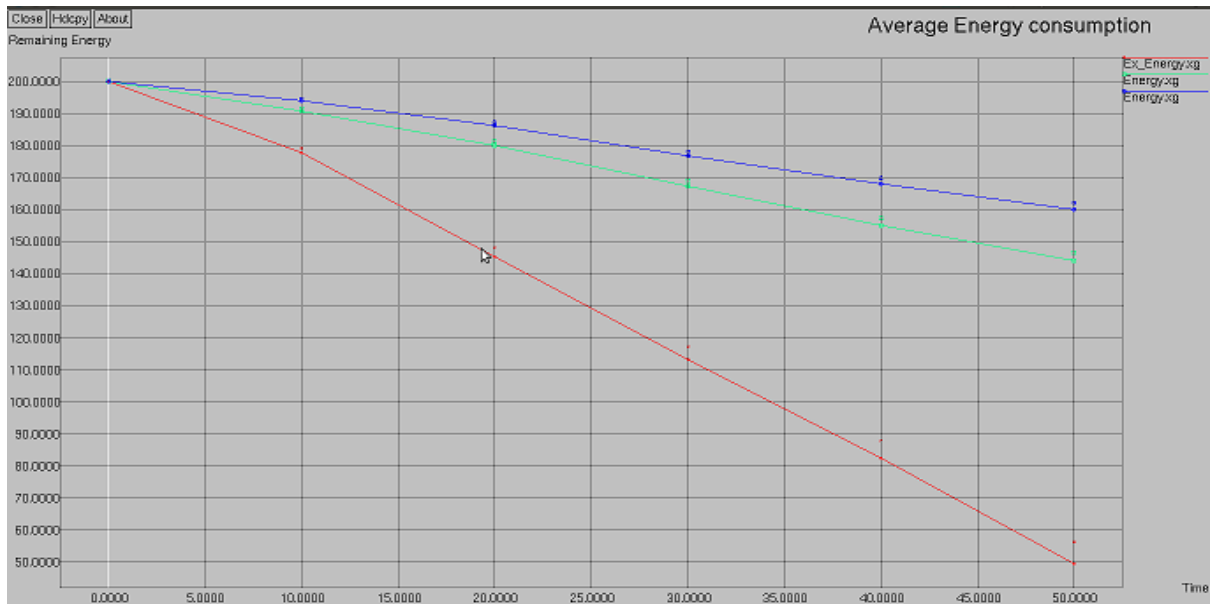


Fig.6. Energy consumption with dynamic scheme

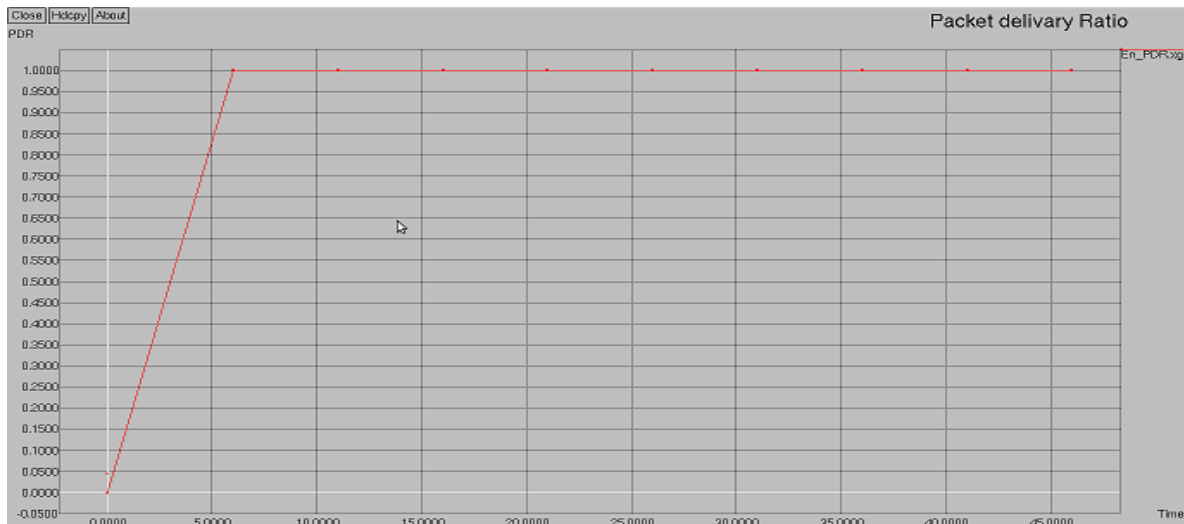


Fig.7. PDR analysis

Fig.8 shows sleep/awake sensor nodes. Here the object is sensed by the sensor which is nearest to the object. That sensor will be in awake condition remaining sensors in that region will be in sleep condition. When the object is moving towards sleep sensor node, it will be enabled and object will be sensed. Sensed data will be transfer to cluster head. It will forward the data to base station. Here we are using DSR protocol for routing purpose [30-31].

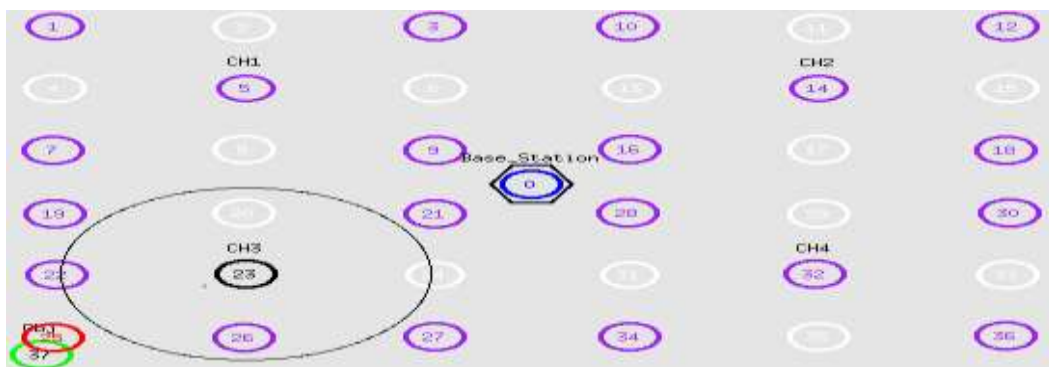


Fig.8. Sleep/Awake sensor nodes.



## 6. CONCLUSION

We have proposed and described our framework for dynamic method. Our framework is 1) clustering-based: sensor nodes form clusters and cluster heads collect and maintain data values 2) prediction based: energy-aware prediction is used to find the subtle trade-off between communication and prediction cost, and 3) Adaptive scheme for energy efficient. We have presented the detailed analysis and description of its two main components: adaptive scheme to enable/disable prediction operations and adaptive update of clustering. Via performance evaluation, we have shown that it achieves energy efficiency and throughput analysis. In order to improve throughput, PDR ratio and to save the sensor energy we can use adaptive scheme method.

## REFERENCES

1. Dr.C.Vivek, S.Palanivel Rajan, "Z-TCAM: An Efficient Memory Architecture Based TCAM", Asian Journal of Information Technology, ISSN No.: 1682-3915, Vol. No.: 15, Issue : 3, pp. 448-454, 2016.
2. R.Sivaranjini "Dynamic scheme for energy efficient data collection in Wireless Sensor Networks", International Conference, 2013.
3. S. Goel and T. Imielinski, "Prediction-Based Monitoring in Sensor Networks: Taking Lessons from MPEG," ACM SIGCOMM Computer Comm. Rev., vol. 31, no. 5, pp. 82-98, 2001.
4. R.Sukanesh, S.Palanivel Rajan, "Cellular Phone based Biomedical System for Health Care", Proceedings of the IEEE, International Conference on Communication Control and Computing Technologies, INSPEC Accession Number: 11745436, ISBN No.: 978-1- 4244-7769-2, pp. 550-553, 2010.
5. A. Silberstein, R. Braynard, G. Filpus, G. Puggioni, A. Gelfand, K. Munagala, and J. Yang, "Data-Driven Processing in Sensor Networks," Proc. Conf. Innovative Data Systems Research (CIDR), 2007.
6. S.Palanivel Rajan, R.Sukanesh, "Viable Investigations and Real Time Recitation of Enhanced ECG Based Cardiac Tele-Monitoring System for Home-Care Applications: A Systematic Evaluation", Telemedicine and e-Health Journal, ISSN: 1530-5627, Vol. No.: 19, Issue No.: 4, pp. 278-286, 2013.
7. A. Meka and A.K. Singh, "Distributed Spatial Clustering in Sensor Networks," Proc. Int'l Conf. Extending Database Technology (EDBT), 2006.
8. S.Palanivel Rajan, R.Sukanesh, "Experimental Studies on Intelligent, Wearable and Automated Wireless Mobile Tele-Alert System for Continuous Cardiac Surveillance", Journal of Applied Research and Technology, ISSN No.: 1665-6423, Vol. No. 11, Issue No.: 1, pp.133 - 143, 2013.
9. S.Vijayprasath, S.Palanivel Rajan, R.Sukanesh, "Experimental Explorations on EOG Signal Processing for realtime Applications in LabVIEW", Proceedings of the IEEE, International Conference on Advanced Communication Control and Computing Technologies, INSPEC Accession Number: 13037966, ISBN No.: 978-1-4673-2045-0, pp. 67-70, 2012
10. S.R. Madden, M.J. Franklin, J.M. Hellerstein, and W. Hong, "Tinydb: An Acquisitional Query Processing System for Sensor Networks," ACM Trans. Database Systems, vol. 30, no. 1, pp. 122- 173, 2005.
11. S.Palanivel Rajan, "Review and Investigations on Future Research Directions of Mobile Based Telecare System for Cardiac Surveillance", Journal of Applied Research and Technology, ISSN No.: 1665-6423, Vol. 13, Issue 4, pp. 454-460, 2015.
12. J. Koshy, I. Wirjawan, R. Pandey, and Y. Ramin, "Balancing Computation and Communication Costs: The Case for Hybrid Execution in Sensor Networks," Ad Hoc Networks, vol. 6, pp. 1185- 1200, 2008.
13. S.Palanivel Rajan, R.Sukanesh, S.Vijayprasath, "Analysis and Effective Implementation of Mobile Based Tele-Alert System for Enhancing Remote Health-Care Scenario", HealthMED Journal, ISSN No. : 1840-2291, Vol. No. 6, Issue No. 7, pp. 2370-2377, 2012.
14. W.B. Heinzelman, A.P. Chandrakasan, and H. Balakrishnan, "An Application-Specific Protocol Architecture for Wireless Microsensor Networks," IEEE Trans. Wireless Comm., vol. 1, no. 4, pp. 660-670, Oct. 2002.
15. S.Palanivel Rajan, K.Sheik Davood, "Performance Evaluation on Automatic Follicles Detection in the Ovary", International Journal of Applied Engineering Research, ISSN No.: 0973-4562, Vol. 10, Issue 55, pp. 1-5, 2015.
16. C.M. Sadler and M. Martonosi, "Data Compression Algorithms for Energy-Constrained Devices in Delay Tolerant Networks," Proc. Int'l Conf. Embedded Networked Sensor Systems (SenSys), 2006.
17. S.Palanivel Rajan, R.Sukanesh, A.Ramprasad, M.Jeyakumar, G.ShanmugaRaja, "A Comprehensive Analysis of Wireless Mobile Based Tele-monitoring System for Myocardial Malfunctioning", International Journal of Computer Science Issues, ISSN No.: 1694-0814, Volume 9, Issue 2, No. 3, pp. 578-582, 2012.
18. A. Wang and A. Chandrakasan, "Energy-Efficient DSPs for Wireless Sensor Networks," IEEE Signal Processing Magazine, vol. 19, no. 4, pp. 68-78, July 2002.



19. S.Palanivel Rajan, R.Sukanesh, S.Vijayprasath, "Design and Development of Mobile Based Smart Tele-Health Care System for Remote Patients", *European Journal of Scientific Research*, ISSN No.: 1450-216X/1450-202X, Vol. No. 70, Issue 1, pp. 148-158, 2012.
20. D. Chu, A. Deshpande, J.M. Hellerstein, and W. Hong, "Approximate Data Collection in Sensor Networks Using Probabilistic Models," *Proc. IEEE Int'l Conf. Data Eng. (ICDE)*, 2006.
21. S.Palanivel Rajan, T.Dinesh, "Statistical Investigation of EEG Based Abnormal Fatigue Detection Using LabVIEW", *International Journal of Applied Engineering Research*, ISSN: 0973-4562, Vol. 10, Issue 43, pp.30426-30431, 2015.
22. H. Jiang, W. Liu, D. Wang, C. Tian, X. Bai, X. Liu, Y. Wu, and W. Liu, "Case: Connectivity-Based Skeleton Extraction in Wireless Sensor Networks," *Proc. IEEE INFOCOM*, 2009
23. S.Palanivel Rajan, T.Dinesh, "Analysis of Human Brain Disorders for Effectual Hippocampus Surveillance", *International Journal of Modern Sciences and Engineering Technology*, ISSN 2349-3755, Vol. No.: 2, Issue No.: 2, pages: 38-45, 2015.
24. B. Gedik, L. Liu, and P.S. Yu, "ASAP: An Adaptive Sampling Approach to Data Collection in Sensor Computing, vol. 18, no. 12, pp. 1766-1783, Dec. 2007.
25. S.Palanivel Rajan, T.Dinesh, "A Systematic Review on Wearable Driver Vigilance System with Future Research Directions", *International Journal of Applied Engineering Research*, ISSN No.: 0973-4562, Vol. 10, Issue No.1, pp. 627-632, 2015.
26. Crossbow, "MICA2 Wireless Measurement System Datasheet," 2003.
27. S.Palanivel Rajan, S.Vijayprasath, "Design of a Simple Graphical User Interface to the Relational Database Management System", *International Journal of Computer Science and Mobile Computing*, ISSN: 2320-088X, Volume 4, Issue 1, January 2015.
28. S. Hussain and A.W. Matin, "Hierarchical Cluster-Based Routing in Wireless Sensor Networks," *Proc. Int'l Conf. Information Processing in Sensor Networks (IPSN)*, 2006
29. S.Palanivel Rajan, S.Vijayprasath, "Performance Analysis on Web based Traffic Control for DDoS Attacks", *International Journal of Engineering Research and General Sciences*, ISSN: 2091 - 2730, Volume 3, Issue 1, 2015.
30. S.Vijayprasath, S.Palanivel Rajan, "Performance Investigation of an Implicit Instrumentation Tool for Deadened Patients Using Common Eye Developments as a Paradigm", *International Journal of Applied Engineering Research*, ISSN No.: 0973-4562, Vol. 10, Issue No.1, pp. 925-929, 2015.
31. S.Palanivel Rajan, "A Survey on Detecting Methods of Driver Drowsiness", *International Journal of Innovations in Engineering and Management*, ISSN: 2319-3344, Vol. 4, Issue No. 1, pp. 1-5, March 2015.
32. W.R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks," *Proc. Hawaii Int'l Conf. System Sciences*, 2000
33. S.Palanivel Rajan, "A Significant and Vital Glance on "Stress and Fitness Monitoring Embedded on a Modern Telematics Platform", *Telemedicine and e-Health Journal*, ISSN: 1530-5627 (Online ISSN: 1556-3669), Vol. No.: 20, Issue No.: 8, pages: 757-758, 2014.
34. S.Vijayprasath, R.Sukanesh, S.Palanivel Rajan, "Assessment of relationship between heart rate variability and drowsiness of post operative patients in driving conditions", *JoKULL Journal*, ISSN No.: 0449-0576, Vol. 63, Issue 11, pp. 107 – 121, 2013.
35. H. Jiang and S. Jin, "Scalable and Robust Aggregation Techniques for Extracting Statistical Information in Sensor Networks," *Proc. IEEE Int'l Conf. Distributed Computing Systems (ICDCS)*, 2006.
36. S.Palanivel Rajan, R.Sukanesh, S.Vijayprasath, "Design and Development of Mobile Based Smart Tele-Health Care System for Remote Patients", *European Journal of Scientific Research*, ISSN No.: 1450-216X/1450-202X, Vol. No. 70, Issue 1, pp. 148-158, 2012.
37. Hemalatha.G and Sivaranjini.R. "Improving Selfish Node Detection In ZigBee Wireless Network With Shortcut Tree Routing", *International Journal of Applied Engineering Research*, ISSN 0973-4562 Vol. 10 No.1 (2015) pp. 850-854.
38. S.Vijayprasath, R.Sukanesh, S.Palanivel Rajan, "Certain Investigations on Eye State Analysis for the Development of a Non-Intrusive Fatigue Detection System", *European Journal of Scientific Research*, ISSN No.: 1450-216X/1450-202X, Vol. 93, No 3, pp.342-350, 2012.
39. M. Tahir and R. Farrell, "Optimal Communication-Computation Tradeoff for Wireless Multimedia Sensor Network Lifetime Maximization," *Proc. IEEE Wireless Comm. and Network Conf. (WCNC)*, 2009.