International Journal of Engineering Science Invention
ISSN (Online): 2319 – 6734, ISSN (Print): 2319 – 6726
www.ijesi.org Volume 2 Issue 3 ǁ March. 2013 ǁ PP.50-55

An Adaptive scheme for energy consumption and data collection in Wireless Sensor Networks

R.Sivaranjini[1], E.Dinesh[2]

[1]*P.G Scholar, II-M.E (Communication Systems) Dept of ECE, M.Kumarasamy College Of Engineering, Karur.*
[2]*Dept of ECE, M.Kumarasamy College Of Engineering, Karur.*

I. INTRODUCTION

Nowadays Wireless sensor networks playing vital role in all area. Which is used to sense the environmental monitoring, Temperature, Soil erosin 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 leads 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.

Index Terms— *Sensor nodes, Failure node, algorithm/protocol design, clustering, adaptive scheme*.

II. PROPOSED ALGORITHM

The rest of the paper is organized as follows: We describe related work on prediction and clustering techniques in sensor networks. We describe the models, analysis, and algorithms our framework and 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. This system provides a performance comparison of different techniques. Finally, we conclude the paper. 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.

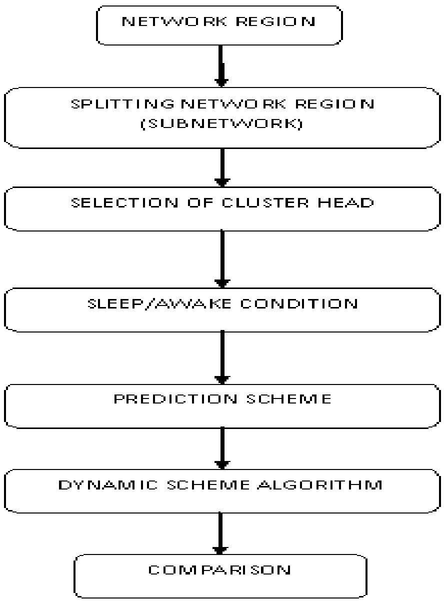

Fig.1. General Block diagram

A. 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 ε-loss approximation: Given an error bound ε > 0, a sensor node sends its value x_t to the cluster head if$|x_t - \hat{x_t}| > \varepsilon$. 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**.

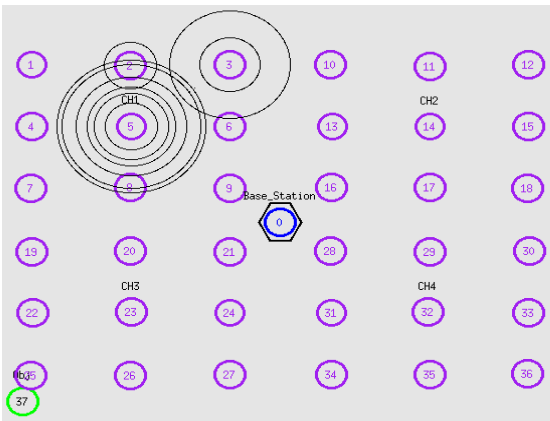

Fig.2. Node ganeration and cluster head selection.

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. 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.

B. Adaptive Update of Clustering

Although, we state that many clustering algorithms can be used in our framework, adaptive update of clustering is often required to capture the change in locality patterns. A complete reclustering is an option but can also be expensive. Not only it involves the establishment of clustering map for all sensors, the complete change in cluster membership also implies all history data and models must be constructed from scratch. In this section, we present algorithms for the dynamic split and merge of clusters, which require low communication cost.

……………………...

Process at the cluster head

01: **if** timeout after m*Δ seconds
02: **for** each member I 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

if prediction is disabled or $|x_t - \widehat{x_t}| > \varepsilon$
send the data value to the cluster head
Update the history data using the data value
else
perform prediction to update the history data

………………………………………………………………

Fig. 4. Operations at the cluster members.

III. EXPERIMENT AND RESULT

A. 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. 3 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.

B. 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 ε. 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 ε 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.

C. 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 α_m is higher than the threshold $\alpha_{threshold}$.

When the cluster members sleep, the clusterhead 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 Δ 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.

D. 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 [30].

E. Benefits of Adaptive Scheme

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

- 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.

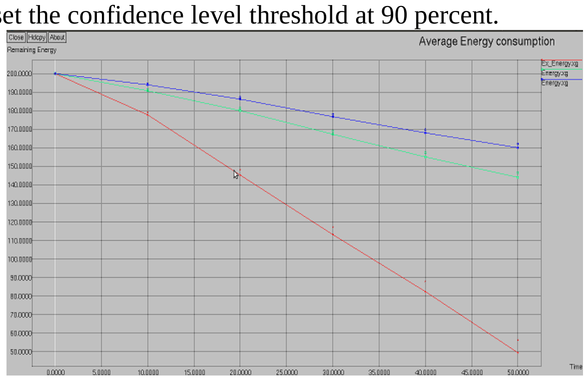

Fig.6. Energy consumption with dynamic scheme

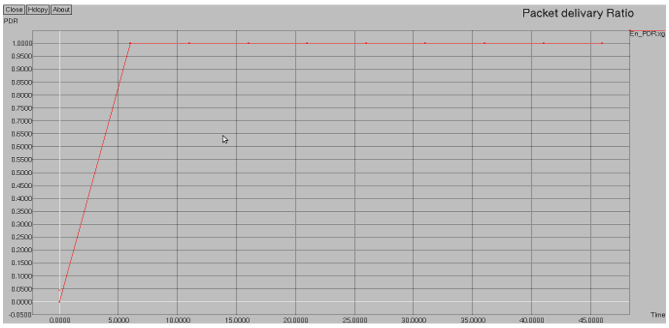

Fig.7. Packet delivery ratio analysis

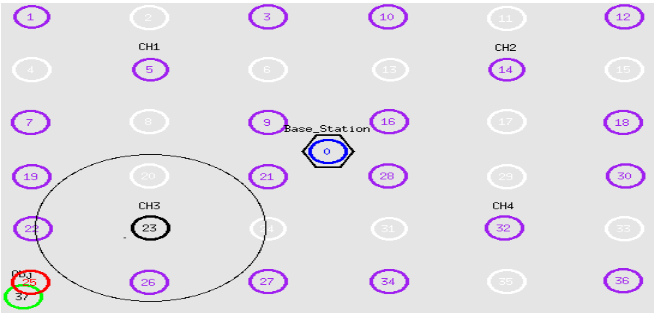

Fig.8. Sleep/Awake sensor nodes.

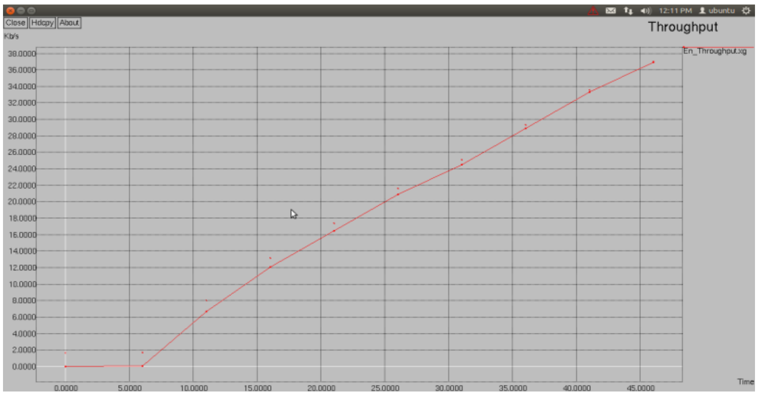

Fig.9.Throughput analysis of Dynamic scheme

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 [29]. Fig.9 shows the throughput analysis of proposed systems. It consists more efficient than the existing system.

V. CONCLUSIONS

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 and to save the sensor energy we can use dynamic scheme method.

REFERENCE

[1]. http://db.lcs.mit.edu/labdata/labdata.html, 2010.
[2]. D. Chu, A. Deshpande, J.M. Hellerstein, and W. Hong, "Approximate Data Collection in Sensor Networks
[3]. Using Probabilistic Models," Proc. IEEE Int'l Conf. Data Eng. (ICDE), 2006.
[4]. Crossbow, "MICA2 Wireless Measurement System Datasheet," 2003.

[5]. B. Gedik, L. Liu, and P.S. Yu, "ASAP: An Adaptive Sampling Approach to Data Collection in Sensor
[6]. Computing, vol. 18, no. 12, pp. 1766-1783, Dec. 2007.
[7]. S. Goel and T. Imielinski, "Prediction-Based Monitoring in Sensor Networks: Taking Lessons from MPEG,"
[8]. ACM SIGCOMM Computer Comm. Rev., vol. 31, no. 5, pp. 82-98, 2001.
[9]. W.B. Heinzelman, A.P. Chandrakasan, and H. Balakrishnan, "An Application-Specific Protocol Architecture
[10]. for Wireless Microsensor Networks, " IEEE Trans. Wireless Comm., vol. 1, no. 4, pp. 660-670, Oct. 2002.
[11]. W.R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Communication Protocol for
[12]. Wireless Microsensor Networks," Proc. Hawaii Int'l Conf. System Sciences, 2000.
[13]. S. Hussain and A.W. Matin, "Hierarchical Cluster-Based Routing in Wireless Sensor Networks," Proc. Int'l
[14]. Conf. Information Processing in Sensor Networks (IPSN), 2006.
[15]. H. Jiang and S. Jin, "Scalable and Robust Aggregation Techniques for Extracting Statistical Information in
[16]. Sensor Networks," Proc. IEEE Int'l Conf. Distributed Computing Systems (ICDCS), 2006.
[17]. H. Jiang and S. Jin, "Leap: Localized Energy-Aware Prediction for Data Collection in Wireless Sensor
[18]. Networks," Proc. IEEE Int'l Conf. Mobile Ad Hoc and Sensor Systems (MASS), 2008.
[19]. H. Jiang, W. Liu, D. Wang, C. Tian, X. Bai, X. Liu, Y. Wu, and W. Liu, "Case: Connectivity-Based Skeleton
[20]. Extraction in Wireless Sensor Networks," Proc. IEEE INFOCOM, 2009.
[21]. H. Jiang, W. Liu, D. Wang, C. Tian, X. Bai, X. Liu, Y. Wu, and W. Liu, "Connectivity-Based Skeleton
[22]. Extraction in Wireless Sensor Networks," IEEE Trans. Parallel and Distributed Systems, vol. 21, no. 5, pp. 710-721, May 2010.
[23]. J. Koshy, I. Wirjawan, R. Pandey, and Y. Ramin, "Balancing Computation and Communication Costs: The
[24]. Case for Hybrid Execution in Sensor Networks," Ad Hoc Networks, vol. 6, pp. 1185- 1200, 2008.
[25]. F. Kuhn, T. Moscibroda, and R. Wattenhofer, "Initializing Newly Deployed Ad Hoc and Sensor Networks,"
[26]. Proc. ACM MOBICOM, 2004.
[27]. S.R. Madden, M.J. Franklin, J.M. Hellerstein, and W. Hong, "Tinydb: An Acquisitional Query Processing
[28]. System for Sensor Networks," ACM Trans. Database Systems, vol. 30, no. 1, pp. 122- 173, 2005.
[29]. S.M. McConnell and D.B. Skillicorn, "A Distributed Approach for Prediction in Sensor Networks," Proc.
[30]. SIAM Int'l Conf. Data Mining Workshop Sensor Networks, 2005.
A.Meka and A.K. Singh, "Distributed Spatial Clustering in Sensor Networks," Proc. Int'l Conf. Extending
[31]. Database Technology (EDBT), 2006.
[32]. M. Pourahmadi, Foundations of Time Series Analysis and Prediction Theory. John Wiley and Sons, 2001.
[33]. C.M. Sadler and M. Martonosi, "Data Compression Algorithms for Energy-Constrained Devices in Delay
[34]. Tolerant Networks," Proc. Int'l Conf. Embedded Networked Sensor Systems (SenSys), 2006.
[35]. B. Sheng, Q. Li, W. Mao, and W. Jin, "Outlier Detection in Sensor Networks," Proc. ACM MOBIHOC, 2007.
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.
[36]. Silberstein, K. Munagala, and J. Yang, "Energy-Efficient Monitoring of Extreme Values in Sensor
[37]. Networks," Proc. ACM SIGMOD, 2006.
[38]. M. Tahir and R. Farrell, "Optimal Communication-Computation Tradeoff for Wireless Multimedia Sensor
[39]. Network Lifetime Maximization," Proc. IEEE Wireless Comm. and Network Conf. (WCNC), 2009.
[40]. D. Tulone and S. Madden, "An Energy-Efficient Querying Framework in Sensor Networks for Detecting Node
[41]. Similarities," Proc. IEEE/ACM Int'l Conf. Modeling, Analysis, and Simulation of Wireless and Mobile
[42]. Systems (MSWiM), 2006.
A.Wang and A. Chandrakasan, "Energy-Efficient DSPs for Wireless Sensor Networks," IEEE Signal
[43]. Processing Magazine, vol. 19, no. 4, pp. 68-78, July 2002.
[44]. Y. Xu and W.-C. Lee, "On Localized Prediction for Power Efficient Object Tracking in Sensor Networks,"
[45]. Proc. IEEE Int'l Conf. Distributed Computing Systems (ICDCS), 2003.
[46]. O. Younis and S. Fahmy, "HEED: A Hybrid Energy-Efficient Distributed Clustering for Adhoc Sensor
[47]. Networks," Proc. IEEE INFOCOM, 2004.
[48]. O. Younis, M. Krunz, and S. Ramasubramanian, "Node Clustering in Wireless Sensor Networks: Recent
[49]. Developments and Deployment Challenges," IEEE Network, vol. 20, no. 3, pp. 20-25, May/ June 2006.
[50]. G.Sriram and D.Srinivasa Rao "Performance Analysis of WRP, DSR and AODV in Wireless Sensor Networks"
[51]. International Journal of Advanced Research in Computer Science and Software Engineering Volume 2, Issue 6,
[52]. pp. 155-159, June 2012.
[53]. In-Sook Lee, Zhen Fu, WenCheng Yang, Myong-Soon Park " An Efficient Dynamic Clustering Algorithm for
[54]. Object Tracking in Wireless Sensor Networks" Complex systems and applications- Modeling, control and
[55]. Simulations, 2007.

R.Sivaranjini received the Bachelor's Degree in Electronics and Communication Engineering, from SSM College of Engineering, Namakkal, India in 2008 and the Pursuing Master's Degree in Communication systems from M.Kumarasamy College of Engineering, Karur, India. Her fields of interest are DIGITAL ELECTRONICS, VLSI, TELECOMMUNICATION, COMPUTER NETWORKS, and WSNs, She is author of more than 4 conference papers.

E.Dinesh received the Bachelor's Degree in Electronics and Communication Engineering, from Adhiyamaan College of engineering, Hosur, India in 2009 and the Master's Degree in Communication systems from Bannari Amman institute of technology, Sathy, India in 2011. His fields of interest are WIRELESS NETWORKS, COMPUTER NETWORKS, and WSNs. He is author of more than 3 conference papers and one journal.