Link Quality Aware Sleep Scheduling for Wireless Sensor Networks

Akshata S Korishettar
M.Tech, Dept. of CSE
AMC Engineering College
Bangalore, India

Dr. Sivasankari G G
Professor and HOD, Dept. of CSE
AMC Engineering College
Bangalore, India

***Abstract*— Wireless Sensor Networks applications often need to be changed after deployment for a variety of reasons. Reconfiguring a set of parameters and patching security holes and modifying tasks of individual nodes. Wireless reprogramming is a crucial technique to address such challenges. Code Dissemination is a basic building block to enable wireless reprogramming. We present a link quality aware Sleep Scheduling Algorithm leveraging the 1-hop link quality information. It has following salient features compared to prior works. First it supports dynamically configurable packet sizes. Second it employs an accurate sender selection procedure to mitigate transmission collisions and transmission over poor links. Third it conserves energy and also takes short completion time.**

Key Words — Wireless Sensor Networks; Dissemination; Reprogramming

I. INTRODUCTION

Wireless Sensor Networks consist of spatially distributed autonomous sensor nodes placed over an area of interest, with high density to monitor physical or environmental conditions. Due to their low price and high deployment flexibility, they are more and more used in various applications such as improving public safety, environmental, disaster management, structural and traffic monitoring etc. A sensor node is composed of typically four units- a) sensing unit: - sense the desired data from the interested region. b) Memory unit: - store the data until it is sent for future use. c) Computation unit: - computes the aggregated data d) power unit: - provides power supply for entire process. Since sensor nodes are battery powered devices therefore they have limited energy. Recent advances in microelectronic mechanical systems (MEMS) and wireless communication technologies, and digital electronics have enabled the development of low-cost, low-power, multifunctional sensor nodes that are small in size and communicate undeterred in short distances.

WSN are usually deployed in inhospitable terrain such as mountainous region where it’s very difficult to recharge or replace batteries. Therefore the main aim of any energy efficient routing protocol is to prolong the network lifetime which is possible by minimizing energy consumption of individual nodes. In addition it is also necessary to ensure that the average rate of consumption of energy by each node is also same. So energy conservation is of main concern. In addition it is also necessary to ensure that the average rate of consumption of energy by each node is also same. Because this would ensure that the connectivity needed to transmit data from a source node to sink node is also maintained. Since lifetime of a network is defined as time in which a single node losses its energy and get exhausted. Moreover, the major unit of energy consumption in sensor node even when sensor nodes are in idle state is the transceiver. Therefore sensor nodes must be put to sleep (radio off) if they are not required to transmit or receive data.

Reprogramming is an important issue in wireless sensor networks. It is the capability to change software functionality of nodes within the network at run time. It enables users to extend or correct functionality of a sensor network after deployment, at a low cost. Changes come in the form of updates, consisting of new applications, bug fixes or modified parameters. Wireless sensor nodes are usually reprogrammed in two ways: either by flashing the node with a complete firmware image, or by loading a partial executable binary. Reprogramming is important both during development, for fast prototyping and debugging, and after deployment, for adapting functionality.

In summary, the contributions of this paper are as follows.

- To identify the inefficiency of previous sender selection algorithms in code dissemination and devise more accurate metric leveraging link quality information, which effectively reduces transmission collisions and transmissions over the poor links.
- The challenge is to integrate sleep scheduling scheme with routing protocols for WSNs with the link quality information. Results show that proposed approach effectively improves the energy consumption and performance in terms of completion time and throughput.

The rest of this paper is structured as follows. Section 2 introduces the literature review. Section 3 presents the mathematical model. Section 4 presents proposed system. Section 5 describes the evaluation results, and finally, Section 6 concludes the paper and gives future research directions.

II. LITERATURE REVIEW

Wei Dong et al., [1] Identified inefficiency in two main aspects in existing protocol designs for code dissemination such as Deluge and MNP. First, the data throughput inefficiency that is the ratio between the network throughput and PHY data rate degrades rapidly as the PHY rate increases. Second, the current sender selection algorithm in MNP [19] (for addressing the broadcast storm problem) does not consider link quality information and needs multiple rounds of message exchanges, resulting in transmission redundancy and long completion time. So proposed an efficient code dissemination (ECD) protocol.

W. Dong et al., [2] proposed a way of analyzing mathematically the performance of bulk data dissemination protocols in WSNs. But the method failed to investigate rigorously the performance of these protocols because of evaluation limitations.

Milosh Stolikj et al., [3] here author investigated two problems they are improving the energy efficiency and improving the delay of reprogramming. And proposed method to solve them. But that method failed to integrate compression in existing or new protocols for distributing images and application updates in wireless sensor networks.

J.W. Hui et al., [4] proposed Deluge protocol which can reliably disseminate data to all nodes at a rate of 90 bytes/second in a real-world deployment, one-ninth the maximum transmission rate of the radio supported under TinyOS. A problem in Deluge is that when a sender receives requests from receivers, it will start transmitting data packets after a specified timeout. It is probable that multiple senders in a neighborhood start transmitting concurrently, causing serious collisions in the transmission.

Subhash Dhar et al., [5] proposed sleep scheduling algorithm. Main purpose of any sleep scheduling algorithm is to maintain network connectivity. It also reduces average energy consumption rate of each node as it is possible to put more number of nodes to sleep.

S. Kulkarni and L. Wang et al., [6] proposed MNP protocol which incorporates a sender selection algorithm which attempts to guarantee that in a neighborhood there is at most one source transmitting at a time. In MNP, source nodes compete with each other based on the number of distinct requests they have received. The sender selection is greedy in that it tries to select the sender that is expected to have the most impact. This effectively reduces the idle listening problem and avoids overhearing.

S. Murthy et al., [11] proposed cluster-based routing protocol, in which all nodes are organized into clusters with one node selected to be cluster-head for each cluster. This cluster-head receives data packets from its members, aggregates them and transmits to a data sink. In some cluster-based routing protocols, the cluster-head assigns TDMA slots to its members to schedule the communication and the sleep mode.

A. Manjeshwar et al., [14] proposed “Threshold sensitive Energy Efficient sensor Network protocol” (TEEN) that is designed for reactive networks, where the nodes react immediately to sudden changes in the environment. Nodes sense the environment continuously, but send the data to cluster heads only when some predefined thresholds are reached.

W. R. Heinzelman et al., [13] proposed “Low-Energy Adaptive Clustering Hierarchy (LEACH)” that was designed for proactive sensor networks, in which the nodes periodically switch on their sensors and transmitters, sense the environment and transmit the data. Nodes communicate with their cluster-heads directly and the randomized rotation of the cluster-heads is used to evenly distribute the energy load among the sensors.

A. Manjeshwar et al., [16] proposed “Adaptive Periodic Threshold sensitive Energy Efficient sensor Network protocol (APTEEN) protocol which combines the features of the above two protocols by modifying TEEN to make it send periodic data. The cluster-based routing protocols can arrange the sleep mode of each node to conserve energy. However, the high complexity and overhead are incurred.

X. Hou and D. Tipper et al., [20] have proposed flat structure based protocol called Gossip-based Sleep Protocol (GSP). That employs probabilistic based sleep modes. At the beginning of a gossip period, each node chooses either to sleep with probability p or to stay awake with probability 1 - p for the period, so that all the sleep nodes will not be able to transmit or receive any packet during the period. When an active node receives any packet, it must retransmit the same. All sleeping nodes wake up at the end of each period. All the nodes repeat the above process for every period.

III. MATHEMATICAL MODEL

A. Determination of Link Quality

All sensor nodes are deployed with equal amount of energy. We assume WSN as a Graph (G) i.e., G = (V,E) where V is a set of vertices and E is a set of edges. Neighbor of a node is selected based on the range of transmission. In a graph, vi is the source node and vj the next neighbor node to receive data as the destination. The distance between source and destination is calculated based on Euclidean distance. Nodes are conditionally dependent when they are connected. Nodes are independent in the nature when they are not connected. Energy consumption of a node depends upon the distance and number of packets transmission.

$$Dij = \sqrt{(X1 - X2)^2 + (Y1 - Y2)2}$$

(1)

$$Ei = Et - (Ei) * N_p * Dij \qquad (2)$$

where Dij is the distance between two nodes i,j and Ei is the energy at node i. N_p is the total number of packets and Et is the total amount energy present in the node.

IV. PROPOSED SYSTEM

The proposed system is a link quality aware Sleep Scheduling Algorithm leveraging the 1-hop link quality information. It has following salient features. First it supports dynamically configurable packet sizes. Second it employs an accurate sender selection procedure to mitigate transmission collisions and transmission over poor links. Third it conserves energy and also takes short completion time.

A. Optimal Link State Routing Protocol

The Optimized Link State Routing (OLSR) is described in RFC3626. As the name suggests, it uses the link state scheme in an optimized manner to diffuse topology information. It is a table-driven pro-active protocol. In a classic link-state algorithm, link-state information is flooded throughout the network. The optimization is based on a technique called Multipoint Relaying. Being a table-driven protocol, OLSR operation mainly consists of updating and maintaining information in a variety of tables. The data in these tables is based on received control traffic, and control traffic is generated based on information retrieved from these tables. OLSR is an optimization of pure link state routing protocol like Open Shortest Path First (OSPF). OLSR provides two main functionalities: Neighbour Discovery and Topology Dissemination. With the help of these two functionalities, each node computes routes to all known destinations.

The two most important message types in OLSR are the HELLO and the TC (Topology Control) messages:

1. HELLO Messages: Every node broadcasts HELLO messages periodically, to support link sensing, detection of neighbours and signalling of MPR selection.
2. TC Messages: Based on the information collected through HELLO messages, link state (TC) messages are created and broadcasted throughout the network by each MPR.

B. System Architecture

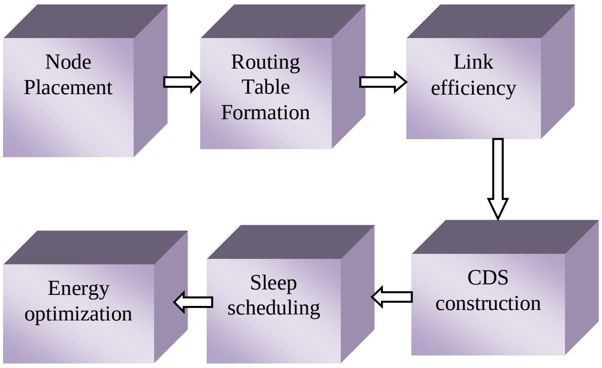


Fig. 1 Proposed System Architecture

As shown in the above figure the proposed system consists of five phases. They are as follows.

a. Node Placement

This module involves building Wireless Network topology, topology consisting of mobile nodes, each node working with multiple channels. This module consists of following steps such as setting up wireless network topology, specifying the simulation start time and end time, specifying the source, destination and data, identifying the neighbors, setting up OLSR parameters, setting the bandwidth and threshold.

b. Routing Table Formation

In this phase with the help of distance equation the distance between nodes is calculated and 1- hop distance routing table is formed for each node. This information is used by OLSR in the next phase.

c. Link Efficiency

In this phase OLSR uses the routing information and the link quality of each node and calculates the efficient routes between each nodes so as to reach the destination node as early as possible in order to take short completion time and if there is failure of any link it also calculates new route through another link and updates the corresponding routing table.

d. CDS Construction

It is Connected Dominating Set (CDS) construction phase. It is prerequisite step for sleep scheduling. It first constructs connected dominating set for parent selection and uses sleep scheduler scheme.

e. Sleep Scheduling

In this phase sleep scheduling algorithm is used. In sleep scheduling algorithms most of the nodes are put to sleep to conserve energy and increase network life time. Main purpose of any sleep scheduling algorithm is to maintain network connectivity. Here an energy aware routing protocol is integrated with the proposed sleep scheduling scheme.

f. Energy Conservation

The proposed approach also considerably reduces average energy consumption rate of each node as we are able to put more number of nodes to sleep in comparison to other approaches such as GSP, which incorporates sleep scheduling using random approach. In this phase we can see the amount of energy conserved.

C. Proposed Sleep Scheduling Algorithm

```
ALGORITHM 1. (* Run the following at each node su *)
1. Get the information of current remaining energy Ei and Eranku;
2. Broadcast Eranku and receive the energy ranks of its currently awake neighbors Nu. Let Ru be the set of these ranks.
3. Broadcast Ru and receive Rv from each sv ∈ Nu.
4. If |Nu| < k or |Nv| < k for any sv ∈ Nv, remain awake. Return.
5. Compute Eu = {sv|sv ∈ Nu and Erankv > Eranku};
6. Go to sleep if both the following conditions hold. Remain awake otherwise.
• Any two nodes in Eu are connected either directly themselves or indirectly through nodes which is in the su's 2-hop neighborhood that have Erankv larger than Eranku;
```

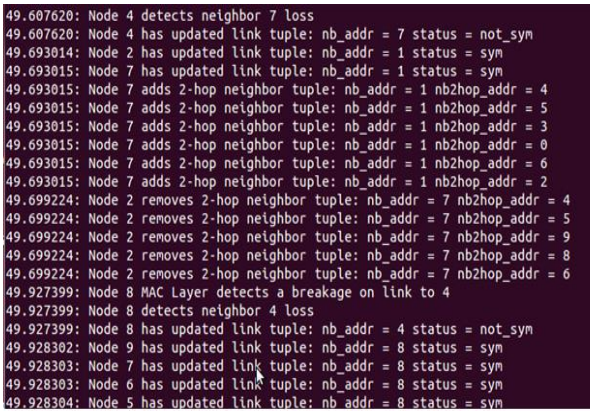
• Any node in Nu has at least k neighbors from Eu.
7. Return.

–

The algorithm takes an input parameter K, the required minimum number of awake neighbors per node. In algorithm, a node su broadcasts its current residual energy information Eranku (Step 1) and computes a subset Eu of neighbors having Erank > Eranku (Step 5). Before su can go to sleep it makes sure that all nodes in Eu are connected by nodes with Erank > Eranku and each of its neighbors has at least k neighbors from Eu (Step 6). These requirements ensure that if a node has less than k neighbors, none of its neighbors goes to sleep and if it has more than k neighbors, at least k neighbors of them decide to remain awake. Note that these requirements are easy to keep by computing locally with 2-hop neighborhood information. The current residual energy is exchanged in Steps 2 and 3.

V. SIMULATION RESULTS

The below figures show the simulation results of network topology of 50 nodes. The proposed energy efficient algorithm is implemented in NS2. Proposed algorithm is compared on the basis of total number of packets transmitted, network lifetime and energy consumed by each node. After completion of simulation we can see the result which is shown in below figure.



```
49.607620: Node 4 detects neighbor 7 loss
49.607620: Node 4 has updated link tuple: nb_addr = 7 status = not_sym
49.693014: Node 2 has updated link tuple: nb_addr = 1 status = sym
49.693015: Node 7 has updated link tuple: nb_addr = 1 status = sym
49.693015: Node 7 adds 2-hop neighbor tuple: nb_addr = 1 nb2hop_addr = 4
49.693015: Node 7 adds 2-hop neighbor tuple: nb_addr = 1 nb2hop_addr = 5
49.693015: Node 7 adds 2-hop neighbor tuple: nb_addr = 1 nb2hop_addr = 3
49.693015: Node 7 adds 2-hop neighbor tuple: nb_addr = 1 nb2hop_addr = 0
49.693015: Node 7 adds 2-hop neighbor tuple: nb_addr = 1 nb2hop_addr = 6
49.693015: Node 7 adds 2-hop neighbor tuple: nb_addr = 1 nb2hop_addr = 2
49.699224: Node 2 removes 2-hop neighbor tuple: nb_addr = 7 nb2hop_addr = 4
49.699224: Node 2 removes 2-hop neighbor tuple: nb_addr = 7 nb2hop_addr = 5
49.699224: Node 2 removes 2-hop neighbor tuple: nb_addr = 7 nb2hop_addr = 9
49.699224: Node 2 removes 2-hop neighbor tuple: nb_addr = 7 nb2hop_addr = 8
49.699224: Node 2 removes 2-hop neighbor tuple: nb_addr = 7 nb2hop_addr = 6
49.927399: Node 8 MAC Layer detects a breakage on link to 4
49.927399: Node 8 detects neighbor 4 loss
49.927399: Node 8 has updated link tuple: nb_addr = 4 status = not_sym
49.928302: Node 9 has updated link tuple: nb_addr = 8 status = sym
49.928303: Node 7 has updated link tuple: nb_addr = 8 status = sym
49.928303: Node 6 has updated link tuple: nb_addr = 8 status = sym
49.928304: Node 5 has updated link tuple: nb_addr = 8 status = sym
```

Fig. 2 This Shows the Link Break and Updates to the Routing Table with Status.

Fig. 2 shows that for each node the status of links to other nodes and also links that are broken and the updated links based on the link quality and energy remaining in each node. So that packet will be sent over good links so as to mitigate transmission collisions and transmissions over poor links. By doing so completion time is also reduced. And at the same time the nodes that are not involved in the transmission are put to sleep so as to conserve energy remarkably.

Fig. 3 gives the link qualities. We can see that 36 percent links are good with link quality > 90 percent, 20 percent links are intermediate with link quality varied between 10 percent and 90 percent. Additionally, 44 percent of the links are poor (with link quality < 10 percent) or disconnected, which represents a proper multihop setting.

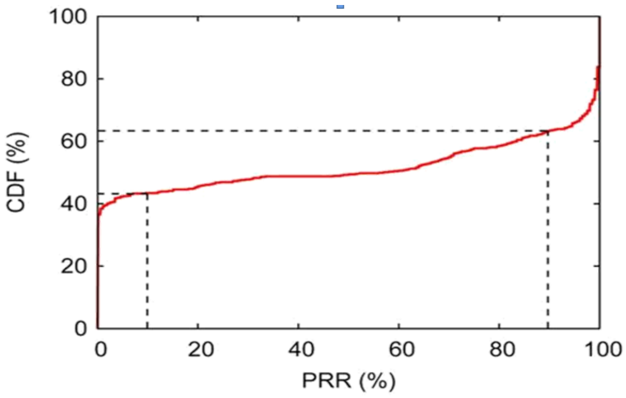


Fig. 3 Link Qualities

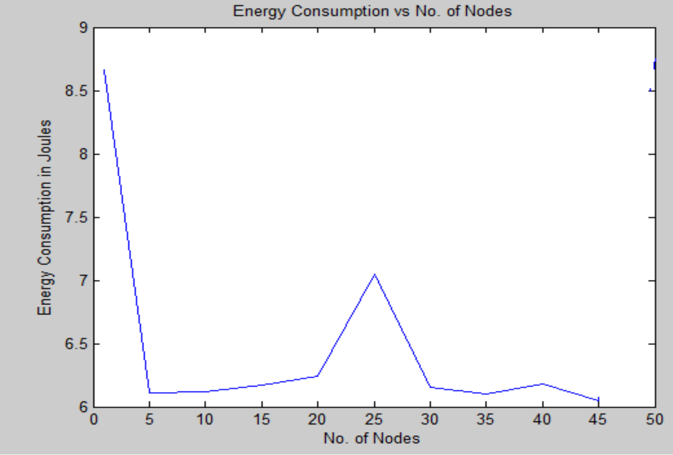


Fig.4 Energy Consumption vs Number of nodes

Fig. 4 shows the amount of energy consumed by different nodes during the simulation. In figure we can see that initially energy consumption is high because most of the links quality need to be analyzed and poor links need to be updated and packets need to be routed. Later due to sleep scheduling energy consumption is reduced. And Fig. 5 shows the number of packets delivered for unit time and it varies as link breaks are detected and updated.

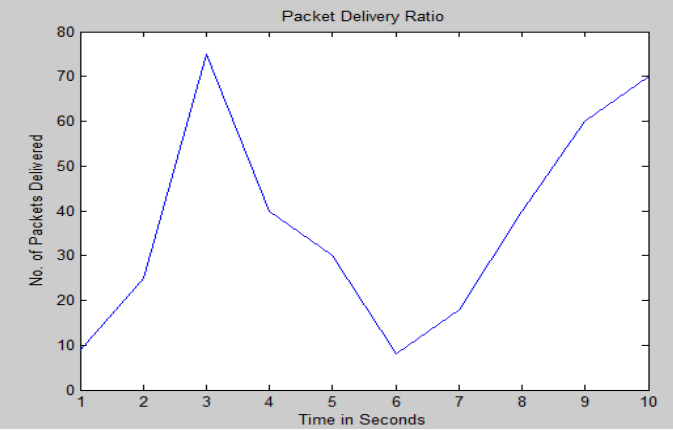


Fig. 5 Number of packets delivered for unit time

TABLE-1: PARAMETERS FOR SIMULATION

Parameters	Values
Initial Energy	100 J
Simulation area	1000m x 1000m
Routing Protocol	OLSR
Bandwidth	11Mb
Willingness	3 sec
hello_ival	2 sec
tc_ival	5 sec
Tx Power	5
Rx Power	1
CBR start	5
CBR stop	45
Listen Time	0.5 sec
Simulation Time	100 sec

VI. CONCLUSION AND FUTURE WORK

In this paper, we present link quality aware sleep scheduling algorithm which is effectively implemented. It has three salient features compared to prior works. First it supports dynamically configurable packet sizes. Second it employs an accurate sender selection procedure based on link quality to mitigate transmission collisions and transmission over poor links. Third it conserves energy and also takes short completion time. This protocol puts more number of nodes to sleep, and therefore provides longer network lifetime. So it conserves more energy. In future we would like to examine effectiveness of our design in large scale Wireless Sensor Networks systems.

REFERENCES

[1] Wei Dong, Member, IEEE, Yunhao Liu, Senior Member, IEEE, Zhiwei Zhao, Student Member, IEEE, Xue Liu, Member, IEEE, Chun Chen, Member, IEEE, and Jiajun Bu, Member, IEEE, “Link Quality Aware Code Dissemination in Wireless Sensor Networks” in IEEE Transactions On Parallel And Distributed Systems, VOL. 25, NO. 7, 1776.

[2] W. Dong, C. Chen, X. Liu, J. Bu, and Y. Liu, “Performance of Bulk Data Dissemination in Wireless Sensor Networks,’’ in Proc. IEEE/ ACM Int’l. Conf. DCOSS, 2009, pp. 356-369.

[3] Milosh Stolikj, Pieter J. L. Cuijpers, Johan J. Lukkien “Efficient reprogramming of wireless sensor networks using incremental updates and data compression,” in May 2012.

[4] J.W. Hui and D. Culler, ‘‘The Dynamic Behavior of a Data Dissemination Protocol for Network Programming at Scale,’’ in Proc. ACM SenSys, 2004, pp. 81-94.

[5] Subhash Dhar Dwivedi, Praveen Kaushik “Energy Efficient Routing Algorithm with sleep scheduling in Wireless Sensor Network” (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 3 (3) , 2012,4350 – 4353.

[6] S. Kulkarni and L. Wang, ‘‘Energy-Efficient Multihop Reprogramming for Sensor Networks,’’ ACM Trans. Sens. Netw., vol. 5, no. 2, p. 16, Mar. 2009.

[7] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, ‘‘Wireless Sensor Networks: A Survey,’’ Comput. Netw., vol. 38, no. 4, pp. 393-422, Mar. 2002

[8] Q. Wang, Y. Zhu, and L. Cheng, ‘‘Reprogramming Wireless Sensor Networks: Challenges and Approaches,’’ IEEE Netw. Mag., vol. 20, no. 3, pp. 48-55, May/June 2006.

[9] W. Dong, Y. Liu, X. Wu, L. Gu, and C. Chen, ‘‘Elon: Enabling Efficient and Long-Term Reprogramming for Wireless Sensor Networks,’’ in Proc. ACM SIGMETRICS, 2010, pp. 49-60.

[10] W. Dong, X. Liu, C. Chen, Y. He, G. Chen, Y. Liu, and J. Bu, ‘‘DPLC: Dynamic Packet Length Control in Wireless Sensor Networks,’’ in Proc. IEEE INFOCOM, 2010, pp. 1-9.

[11] S. Murthy and J. J. Garcia-Luna-Aceves, (1996) “An efficient routing protocol for wireless networks,” ACM Balzer Mobile Networks and Applications Journal.

[12] E. M. Royer and C.-K. Toh,(1999) “A review of current routing protocols for ad hoc mobile wireless networks,” IEEE Personal Communications.

[13] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan,(2000) “Energyefficient communication protocol for wireless microsensor networks,” in IEEE Hawaii International Conference on System Sciences.

[14] A. Manjeshwar and D. P. Agrawal,(2001) “TEEN: A routing protocol for enhanced efficiency in wireless sensor networks,” in IEEE International Parallel Distributed Processing Symposium.

[15] F. Ye, A. Chen, S. Lu, and L. Zhang,(2001) “A scalable solution to minimum cost forwarding in large sensor networks,” in Computer Communications and Networks, Proceedings.Tenth International Conference on, 2001, pp. 304–309.

[16] A. Manjeshwar and D. P. Agrawal,(2002) “APTEEN: A hybrid protocol for efficient routing and comprehensive information retrieval in wireless sensor networks,” in IEEE International Parallel Distributed Processing Symposium.

[17] R. Shah and J. Rabaey,(2002) “Energy aware routing for low energy ad hoc sensor networks,” in Wireless Communications and Networking Conference, 2002. WCNC2002. 2002 IEEE,vol. 1 pp. 350–355 vol.1.

[18] J. Al-Karaki and A. Kamal,(2004) “Routing techniques inwireless sensor networks: a survey,” Wireless Communications, IEEE, vol. 11, no. 6, pp. 6–28.

[19] M.-L. Pham, D. Kim, Y. Doh, and S. eun Yoo,(2004) “Power aware chain routing protocol for data gathering in sensor networks,” in Intelligent Sensors, Sensor Networks and Information Processing Conference, 2004. Proceedings of the 2004, pp. 107–112.

[20] X. Hou and D. Tipper,(2004) “Gossip-based sleep protocol (gsp) for energy efficient routing in wireless ad networks,” in Wireless Communications and Networking Conference, 2004. WCNC. 2004 IEEE, vol. 3, pp. 1305–1310 Vol.3.

[21] E. Bulut and I. Korpeoglu,(2007) “Dssp: A dynamic sleep scheduling protocol for prolonging the lifetime of wireless sensor networks,” in Advanced Information Networking and Applications Workshops, 2007, AINAW ’07. 21st International Conference on, vol. 2,pp. 725– 730.

[22] L. Hong and J. Yang,(2009) “An energy-balance multipath routing based on rumor routing for wireless sensor networks,” in Natural Computation, 2009. ICNC ’09. Fifth International Conference on, vol. 3, pp. 87–91.

[23] Martin Heusse, Franck Rousseau, Romaric Guillier, Andrzej Duda “Idle Sense: An Optimal Access Method for High Throughput and Fairness in Rate Diverse Wireless LANs” SIGCOMM’05, August 22–26, 2005, Philadelphia, Pennsylvania, USA.