International Journal of Future Generation Communication and Networking
Vol. 9, No. 2 (2016), pp. 155-162
http://dx.doi.org/10.14257/ijfgcn.2016.9.2.16

A Novel Sleeping Scheduling Method for Wireless Sensor Networks Based on Data Fusion

Feiyan Song[1], Bin Yang[2*] and Yang Zhao[3]

[1]*School of Automation, BaoTou Light Industry Vocaitonal Technical College, Baotou0 14035, China*
[2]*School of Electronic and Information Engineering, Shunde Polytechnic, Shunde 528300, China*
[3]*Department of Electronic and Information Technology, Jiangmen Polytechnic, Jiangmen 529090, China*
yang_bby@126.com

Abstract

In order to increase the node energy utilization and lengthen the lifetime of the wireless sensor network (WSN), a novel clustering node sleep scheduling algorithm based on data fusion (DFS) is proposed in this paper. DFS increases cluster head's sleep time by adapting the data fusion window, which can realize the goal of energy saving and increase the whole network's lifetime. Simulation results show that the DFS method improves the effectiveness of the network energy consumption and it has better performance on the network coverage maintenance with effectively extending the lifetime of the network.

Keywords*: WSN, data fusion, sleeping scheduling, Random Sleeping*

1. Introduction

With rapid advances development in sensor, network, wireless communication and embedded techniques in recent years, WSN (Wireless Sensor Network) finds wide uses in defense, environmental monitoring, structural health monitoring, transport management, industrial process control, anti-terrorism and disaster relief [1-4]. In WSN, the node is usually powered by the battery. Due to the limit of WSN nodes in power supply, how to lengthen the lifetime of the network by saving as many energies as possible becomes a major issue in WSN. How to reduce energy consumption of WSN nodes and increase the lifetime of the network has been an active topic in recent years [5-6].

The network nodes have three working modes: operation, idleness and sleep, each of which corresponds to an energy consumption. In the operation mode, the node consumes most energies; in the idle mode, the processor is allowed to stop but the network needs to continue to hear the interrupt servicing request, resulting in less energy consumption overall. In the sleep mode, most circuits are disconnected except the wake-up circuits, thus minimizing the power supplied. Hence, energy consumption of nodes can be decreased by scheduling node hibernation.

Many works have been done on the scheduling of node hibernation. Authors in [7] proposed a heuristic aggregation tree construction method by introducing the linear planning scheme, and also devised a heuristic clustering-based energy-efficient algorithm that can achieve maximum lifetime. A joint optimized data fusion and routing scheme was proposed in [8], which derived the optimized expression based on the maximum network lifetime from analysis. Finally, this scheme obtained the optimal solution using the distributed gradient algorithm. Authors in [9] combined the data fusion technique with the efficient routing scheme, analyzed the tradeoff of node energy consumption between

ISSN: 2233-7857 IJFGCN

MAC layer retransmission and data fusion routing, obtaining the general equation for energy consumption and the optimal solution. Most of the works above improved energy efficiency by optimizing the protocols and algorithms, but paid little attention to energy management.

Authors in [5] analyzed the clustering algorithm, and proposed a sleep scheduling algorithm based on the linear distance between the node and the cluster head, where the node distant from the head goes into the sleep state at a high probability. It was demonstrated in [10] that the distance-based sleep scheduling algorithm was very helpful in improving energy efficiency of the heterogeneous network. But because the energy consumption difference among nodes is at least second-order correlated with the transmission distance, some sensor nodes consume more energies than other nodes and thus fail quickly. An energy-based sleep scheduling algorithm was proposed in [11], where the sleep probability is in reverse proportion to the residual energy of communicating nodes, thereby improving energy balance. A node sleep scheduling algorithm was proposed in [12] to ensure network coverage, allowing nodes to sleep when they are covered by neighbors in their sensed areas. However, the coverage of neighbors in this algorithm was so redundant that some energies were wasted. Most of these works mainly consider the quality of network coverage to the neglect of node energy efficiency Therefore, some nodes in the high-density area may be always in the operation mode, resulting in energy imbalance among nodes.

In this paper, we proposed a novel sleeping scheduling method for WSN based on data fusion technology, which clusters the nodes and constructs the node tree using the LEACH- clustering method [13]. Considering that network nodes are likely to die or leave, the cluster head counts the number of neighbors. With decrease in the number of neighbors, the data fusion window is adjusted to increase the sleep time of neighbors and decrease unnecessary energy consumption among nodes, thereby improving the network lifetime overall.

2. Considerations in Sleep Scheduling

If nodes work alternately in the wireless sensor network, then the monitoring task of the entire network and the normal communications among nodes will be affected inevitably. Hence, the following considerations should be taken into account in the node sleeping scheduling scheme.

2.1. Coverage

Coverage refers to the ratio between the union of detection areas of all network nodes and the monitoring area of the whole network. The monitoring area may be completely or incompletely covered by the working nodes. How to turn off redundant nodes as many as possible without compromise of the coverage performance, is the chief issue that should be considered during node scheduling.

2.2. Connectivity

Connectivity means that the data collected by sensor nodes can be forwarded by other nodes and finally delivered to the destinations. If nodes are instructed to work alternately, then rerouting is necessary when the forwarding nodes happen to be sleeping. Hence, network connectivity must be taken into account in the sleep scheduling scheme

2.3. Sleep-Induced Delay

Sleep-induced delay refers to the transmission delay and the detection delay caused as the node goes into the sleep state. After nodes work alternately, if the event happens during the sleep, then it is not detected until the next slot when the node is awakened. This is the detection delay caused by sleep. Similarly, if the next hop of data transmission is sleeping, the data cannot be forwarded until the next slot when the node is awakened. Data interrupt is thus caused, and this is the sleep-induced transmission delay. The sleep-induced transmission delay cannot be neglected in delay-intensive applications.

3. Data Fusion-Based Sleep Scheduling Algorithm

The flowchart of DFS algorithm is shown as in Figure 1, which consists of four steps, initialization, updating of the table of neighbors information, adjustment of the data fusion window and adjustment of the head sleep time. The details of them are introduced as follows:

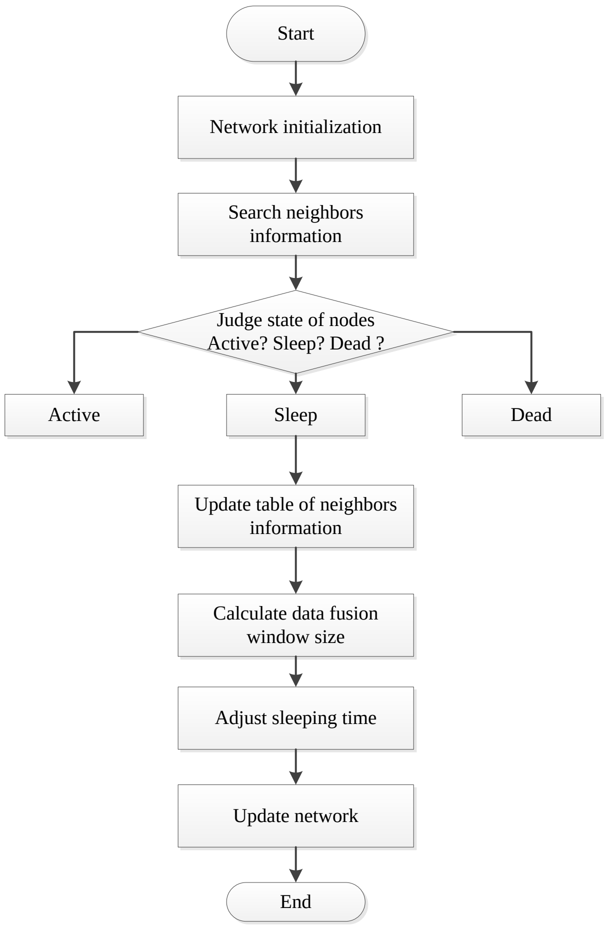

Figure 1. Flowchart of DFS Sleeping Scheduling Method

1. **Initialization**

The clustering algorithm from [13] is first used to cluster the network nodes. As in the LEACH protocol, a sensor node is randomly selected as the cluster head. Once the choice of the head is made, then the head begins to broadcast data to all nodes in the network. The nodes decide which cluster to join based on the strength of the broadcast signal, and tell this to the corresponding cluster heads via data transmission. At the readiness stage, the head begins to receive the data collected by each node in the cluster, aggregates the data using data fusion and compression techniques, and then sends the aggregated data to the base station or the sink. After the readiness stage lasts for a period, the network enters the next operation cycle to reselect the head. And the node tree is constructed by defining the head as the parent and the other nodes in the cluster as the children. Each node maintains a table of information about neighbors. When a node is chosen as the head, its table of neighbors information and the indexes are activated.

2. **Updating of the Table of Neighbors Information**

When the parent of the node tree is Active, the detection request is made periodically to the children. On reception of the requests, the children will send an acknowledgement. If the parent receives the returned data, the parent will update the state of this node in the table to Active; otherwise, the parent updates it to sleep tentatively, because it may be caused by time synchronization drift. If no message is received for three times in succession, the state of this node is updated to Dead.

3. Adjustment of the Data Fusion Window

We define an adaptive window for the head when it performs data fusion. This window is called the data fusion window and denoted by FW (fusion window). Its original value is the product of the number of child nodes in the cluster at first and the transmission time as well as the sensing delay:

$$FW = Size_{original} \times Time_{sum} \qquad (1)$$

The head periodically counts the number of children in the node tree and then compares with the number of children previously stored in the head. If the children are fewer than before, then the data fusion window will shrink, and the shrunk size is the product of the number of child nodes in the cluster currently and the transmission time as well as the sensing delay.

4. **Adjustment of the Head Sleep Time**

The sleep time of the head is denoted by SleepTime. It is adjusted based on NextFW. Variation of NextFW means that some nodes in this cluster die or leave. In this case, Sleeptime varies accordingly. The sleep time is the total time subtracted by the active time of the node; and the active time is the sum of the fusion window, data transmission time and the sensing delay.

4. Simulations and Result Analysis

4.1. Comparison of Results Achieved Before and After Algorithm Implementation

There are 100 nodes in the originally network. Four groups experiments are performed, keeping the parameters unchanged, include the collection time interval and the length of data packets. There are no nodes are removed manually in the first group of experiment, and remove 10, 20, 40 nodes in the following three experiments, respectively. If over half of the remaining nodes fail, the network is deemed failed, and the duration is defined as the lifetime of the network. Figure 2 shows the experimental result.

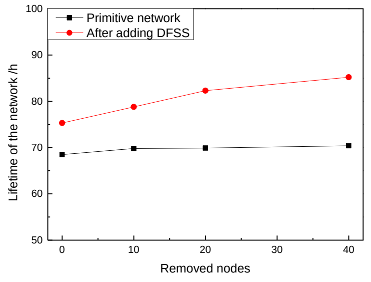

Figure 2. Lifetime Performance of the Network Before and After Adding DFS

It can be seen that the lifetime of network after adding DFS method is obviously longer than primitive network, increased by 15% on average. In DFS network, all nodes in the cluster is adjusted with the change of the head. Energy balance among nodes in the cluster is achieved and the sleep time is maximized in this way, thereby increasing the lifetime of the whole network.

4.2. Comparison with Other Sleeping Scheduling Methods

To demonstrate effectiveness of the proposed algorithm, simulations are carried out to analyze and compare the proposed algorithm with RS algorithm (Random Sleeping), and ENMS algorithm (Sleeping Based on Energy and Neighbor Message) in [6, 14]. These algorithms are evaluated in terms of the network's sleep ratio, coverage, redundant coverage area, and network lifetime and energy consumption.

4.2.1. Comparison of Performance in the Different Network Sizes

In this section, the coverage performance of the algorithms are tested in the network from 50 nodes to 800 nodes. The algorithm performance is measured with the sleep ratio, coverage and redundant coverage area. Simulation results are shown as in Figure 3-5.

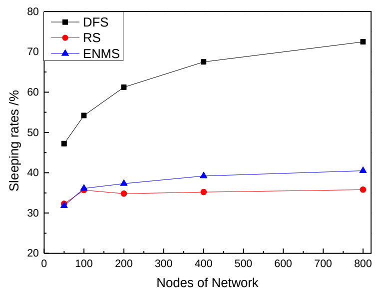

Figure 3. Comparison of Sleeping Rates of Different Algorithms under the Same Condition

From Figure 3, it can be observed that RS and ENS always provide a stably low sleep ratio and the ratio does not vary greatly with the size of the network. Compared with other two algorithms, the proposed algorithm DFS has a higher sleep ratio and the change of the sleep ratio better matches the reality.

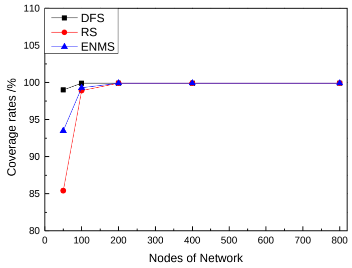

Figure 4. Coverage Rates of Different Algorithms under the Same Condition

From Figure 4, it can be seen that compared with other two methods, DFS provides a stably high network coverage when the scale of network is not very huge (less than 150 nodes). All of the three methods have a high coverage rates when the nodes of network more than 200, nearly 100%.

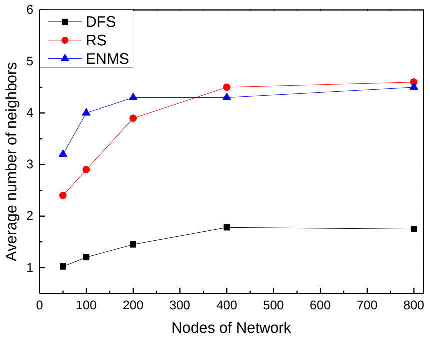

Figure 5. Overlapping Covering Area Ratios of Different Algorithms under the Same Condition

From Figure 5, it can be observed that the network redundant coverage area of DFS is lower than that of other two algorithms. This means that DFS is more capable of adapting to the variation of the network size, maintaining low coverage without compromise of the network coverage ratio, and of effectively reducing the network's redundant coverage area.

4.2.2. Comparison of Coverage in the Networks of the Same Size

In this experiment, the coverage performance of the algorithms is evaluated in the networks under the same network scale. The nodes of network in the simulation is set by 100 nodes. During the test, the network operates until 30% of the nodes fail. The experimental results are shown in Figure 6. It can be seen that there are more and more failed nodes in the network during network operation, and that DFS maintains good coverage, with the average 57.2% and 19.1% than RS and ENMS methods, respectively.

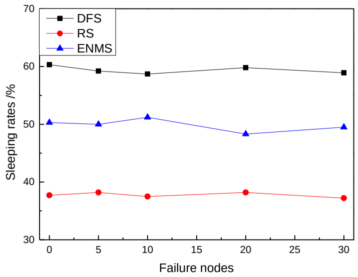

Figure 6. Comparison of Sleeping Rates of Different Algorithms with Failure Nodes Condition

4.2.3. Comparison of Algorithm Performance in the Networks of the Same Size

We define the time when 30% of nodes fail as the lifetime of the network. As in previous section, the energy consumption and lifetime of the network are evaluated in the network with 100 nodes. Simulation results are shown in Figure 7. It can be seen that compared with RS and ENMS algorithms, DFS effectively lengthens the network lifetime and reduces the network's energy consumption.

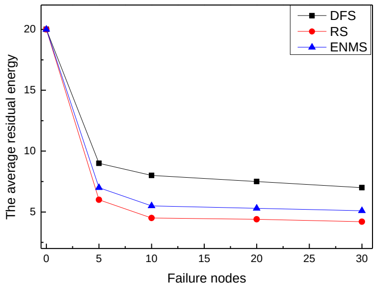

Figure 7. Average Residual Energy Changing of With Failure Nodes Condition

5. Conclusion

This paper proposed a novel sleep scheduling schemes for WSN based on data fusion, which constructs the node tree in the cluster and adapt the data fusion window to the number of child nodes. Simulation results demonstrate that DSF can adapt to the changes of network size, while maintaining low coverage while ensuring the network coverage, but also can effectively reduce the redundancy of the network coverage, compared with RS and ENMS methods. DFS greatly saves network energies while maintaining the quality of network coverage, as well as lengthening the network lifetime. In the future work, we will study how to further reduce the data transmission delay between nodes in the sleep scheduling scheme.

References

[1] T. Harms, S. Sedigh and F. Bastianini, "Structural health monitoring of bridges using wireless sensor networks", IEEE Instrumentation and Measurement Magazine, vol. 13, no. 6, **(2010)**, pp. 14-18

[2] H. Ramamurthy, B.S. Prabhu, R. Gadh and A. Madni, "Wireless industrial monitoring and control using a smart sensor platform", IEEE Sensors Journal, vol. 7, no. 5, **(2007)**, pp. 611-618.

[3] D. Ghataoura, J. Mitchell and G. Matich, "Vigilant+: mission objective interest groups for wireless sensor network surveillance applications", IET Wireless Sensor Systems, vol. 1, no. 4, **(2011),** pp. 229-240.

[4] G. Tuna, V.C. Gungor and K. Gulez, "An autonomous wireless sensor network deployment system using mobile robots for human existence detection in case of disasters", Ad Hoc Networks, vol. 13, **(2014)**, pp. 54-68.

[5] J. Deng, Y.S. Han and W.B. Heinzelman, "Scheduling Sleeping Nodes in High Density Cluster-Based Sensor Networks", Mobile Networks and Applications, vol. 10, no. 6, **(2005)**, pp. 825-835.

[6] G.T. Shi and M.H. Liao, "Stochastic Sleeping for Energy-Conserving in Large Wireless Sensor Networks, Journal of Computer Research and Development, vol. 43, no. 4, **(2006)**, pp. 579-585, (in Chinese).

[7] K. Dasgupta and P. Namjoshi, "Efficient algorithms for maximum lifetime data gathering and aggregation in wireless sensor networks", Computer Networks, vol. 42, no. 6, **(2003)**, pp. 697-716.

[8] C.Q. Hua and T.S. Yum, "Optimal routing and data aggregation for maximizing lifetime of wireless sensor networks, IEEE /ACM Trans on Networking, vol. 16, no. 4, **(2008)**, pp. 892-903.

[9] F. Lin, S. Yong and H.H. Yen, "Delay QoS and MAC aware energy-efficient data aggregation routing in wireless sensor networks", Sensors, vol. 9, no. 10, **(2009)**, pp. 7711-7732.

[10] E.J. Duarte and M.Y. Liu, "Analysis of Energy Consumption and Lifetime of Heterogeneous Wireless Sensor Networks", Proc of the IEEE Global Telecommunications Conference, **(2002)**, pp. 21-25, Taipei, China.

[11] M.R. Pearlman, J. Deng and B. Liang, "Elective Participation in Ad Hoc Networks Based on Energy Consumption, Proceedings of the IEEE Global Telecommunications Conference, **(2002),** pp. 026-31, Taipei, China.
[12] D. Tian and N.D. Georganas, "A Node Scheduling Scheme for Energy Conservation in Large Wireless Sensor Networks", Wireless Communications and Mobile Computing, vol. 3**,** no. 2, **(2003)**, pp. 271-290.
[13] W.R. Heinzelman, A. Chandrakasan and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks", International Conference on System Sciences, 2000. Proceedings of the 33rd Annual Hawaii, **(2000)**, pp. 3005-3014.
[14] K.S. He, J.S. Huang and Z.S. Yao, "Stochastic Sleeping Protocol Based on Energy and Neighbor Message in Wireless Sensor Network", Journal of Computer Applications, vol. 28, no. 6, **(2008)**, pp. 1475-1477, (Chinese).

