Balancing Overhearing Energy and Latency in Wireless Sensor Networks

Byoungyong Lee, Kyungseo Park, Ramez Elmasri

Computer Science & Engineering Department
University of Texas at Arlington
Arlington, TX 76019, USA
{bylee,kpark,elmasri}@uta.edu

Abstract. A WSN (Wireless Sensor Networks) consists of a large number of sensor nodes. Each sensor node has limited battery, small storage, and short radio range. Many researchers have proposed various methods to reduce energy consumption in sensor nodes, since it is difficult to replace sensor node power sources. Generally, a sensor node consumes its energy during processing, receiving, transmitting and overhearing of messages that are directed to other nodes. Among those, overhearing is not necessary for correct operation of sensor networks. In this paper we propose a new synchronized wakeup scheme to reduce the overhearing energy consumption using different wakeup time scheduling for extending sensor network lifetime. The results of our simulation show that there is a trade-off between reducing overhearing energy and delay time. Therefore we propose Double Trees Structure, called DTS, having two routing trees, one based on Short Rings Topology and the other on Long Rings Topology. DTS has multi routing paths from base station to children nodes. If a node which is on the next routing path does not wakeup in time to receive the data, the sender node selects another path to connect to the destination. We can save the wait time until the next destination node wakes up. In the simulation result, our wakeup scheduling reduces overhearing energy consumption more than the S-MAC protocol. Using the double trees structure reduces the delay time.

Keywords: Sensor Network, Wakeup Scheduling, Overhearing Energy.

1 Introduction

Wireless sensor networks are increasingly applied to various physical worlds for surveillance applications. Because large numbers of sensors are typically deployed, the trend has been to decrease the cost of each sensor node. As a result, a sensor node has smaller size than before. Therefore there are various capacity limitations such as the small amount of battery, limited storage, and short radio range [2][4]. Even if each sensor node has small capacity, the large number of sensors can cover a large area by cooperating with each other to form a multi-hop wireless network. Nevertheless low battery power is one of the most crucial problems because it is hard to replace or recharge the battery in each sensor [3]. The life time of a sensor network depends on the energy in each senor node. To increase the life time of sensor networks, we need to reduce the energy consumption.

Generally, a sensor node consumes its energy during processing, receiving, transmitting, and overhearing of parts of messages that are not directed to the node. Among those, the energy wasted by overhearing energy consumption is not necessary for correct working of the wireless sensor networks. A characteristic of wireless networks is that some nodes that are not a destination have to receive unnecessary messages because they are within the radio range. This is called overhearing. As node density increases and radio range grows, energy consumed by overhearing also will increase. In order to achieve the purpose of reducing the energy consumption, synchronized wakeup scheduling is used to make a node stay in sleep mode when messages are not directed to the node. We focus on reducing overhearing energy consumption with wakeup scheduling.

For reducing overhearing energy consumption we propose a new wakeup scheme using different wakeup times between neighbor nodes. We call the new wakeup scheme Odd and Even Wakeup Scheduling (OEWS). We compare OEWS with the S-MAC protocol which is one of the popular MAC protocols for sensor networks [1]. In simulation, OEWS shows good results to reduce overhearing energy. This method improves the energy efficiency and increases the sensor network lifetime. OEWS adjusts different wakeup times for sibling nodes. A node in sleep mode will turn off its radio and will not overhear messages. There is a trade-off between energy efficiency and delay time because the node which intends to send the data has to wait until its next destination node wakes up according to a pre-defined synchronized schedule. For reducing delay time, we propose another new tree structure called Double Trees Structure (DTS).

For reducing the data delay time, there are many methods using various wakeup scheduling patterns [14]. However, even if those wakeup patterns are efficient to reduce data delay time, it is hard to adjust those wakeup patterns to OEWS because they didn’t consider overhearing energy consumption. Hence, we propose a new routing tree structure called DTS for reducing data delay time on OEWS. DTS has two tree structures called Short Rings Topology (SRT) and Long Rings Topology (LRT). SRT and LRT have the same number of hops from the base station to children nodes. There is no different delay time between SRT and LRT because they have the same number of hops. Therefore, by using multiple paths, we can save the waiting time for children nodes to wake up.

The contribution of this paper is that we explore reducing both overhearing energy and latency together. We propose the OEWS to reduce overhearing energy and DTS for decreasing the latency. Overhearing energy is not necessary for operating the sensor network. Therefore it is important to reduce the overhearing energy for extending the lifetime of sensor networks. In the simulation, OEWS reduces the overhearing energy up to 43% compared to S-MAC protocol. Also, DTS helps OEWS reduce latency up to 30.4% than OEWS without DTS.

The remainder of this paper is organized as follows. In section 2, we provide an overview of our network model. In section 3, we introduce the OEWS for reducing overhearing energy consumption and DTS for reducing delay time. We show the simulation results in section 4. Finally, section 5 presents concluding remarks

2 Network Model

2.1 Radio Model.

For measuring energy consumption in a sensor network, we need an energy model. A sensor node consumes energy by transmitting, receiving and overhearing. E_{tx} is the energy used for transmitting and E_{rx} is the energy used for receiving. We assume the energy model including overhearing energy based on [6]. This radio model calculates the energy spent for one bit to send over a distance ‘d’ as

$$E_{tx} = E_{txelec} + \epsilon d^2 \qquad (1)$$

$$E_{rx} = E_{rxelec} \qquad (2)$$

E_{txelec}, the energy consumption by transmitter electronics, dissipates *50nJ/bit*. We assume that E_{txelec} is the same as E_{rxelec} (receiving energy) based on [7]. We suppose that ϵ, which is an amplifier characteristic constant, is *100pJ/bit*. The model assumes the radio channel to be symmetric, which means the cost of transmitting a message from A to B is the same as the cost for transmitting from B to A [7]. Overhearing energy consumption is defined by E_{oh} as

$$E_{oh} = E_{rxelec} \qquad (3)$$

In S-MAC protocol [1], when sending data from one sensor node to others, RTS(ready to send), CTS(clear to send), and ACK packets are necessary. Therefore, based on (1), (2), and (3), total energy consumption from node *i* to node *j* is represented by the following:

$$Eij = |RTS+DATA| \times (\epsilon di2+ Etxelec) + |CTS+ACK| \times (\epsilon dj2+ Etxelec) + RTS+CTS+DATA+ACK| \times Erxelec + NRTS \times |RTS| \times Erxelec + NCTS \times |CTS| \times Erxelec \qquad (4)$$

Here, d_i is the radio range of node *i* and d_j is the radio range of node *j*. N_{RTS} and N_{CTS} is the number of neighbors which overhear the RTS and CTS packets respectively. |RTS,CTS, ACK, or DATA,| is the size of the packet in bits.

In OEWS, total energy consumption is different from S-MAC. OEWS has two kinds of energy models. One is for nodes which have an odd id number and the other is for nodes which have an even id number. Therefore, total energy consumption from node *i* to node *j* which both have odd id number (or even id number) is represented by the following:

$$E_{ij} = |RTS+DATA| \times (\epsilon d_i^2+ E_{txelec}) + |CTS+ACK| \times (\epsilon d_j^2+ E_{txelec}) + |RTS+CTS+DATA+ACK| \times E_{rxele}c + N_{odd\_RTS} (or\ N_{even\_RTS}) \times |RTS| \times E_{rxelec} + N_{odd\_CTS} (or\ N_{even\_CTS}) \times |\ CTS| \times E_{rxelec} \qquad (5)$$

Here, N_{odd_CTS} (N_{even_CTS}) is the number of odd (Even) id neighbors which overhear CTS and N_{odd_RTS}(N_{even_RTS}) is the number of odd (even) id neighbors which overhear RTS.

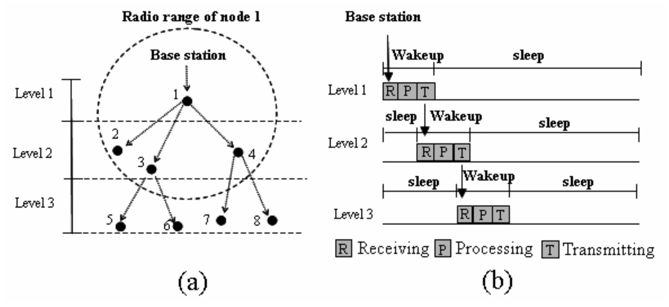


Fig. 1 Example of wakeup model

2.2 Wakeup Model

In wireless sensor networks, we can divide data flow into two directions. In the down direction, data flows from the base station to children nodes. In the up direction, data flows from children nodes to the base station. Our goal is to reduce the overhearing energy consumption when the base station transmits queries or data to children nodes. Hence in our wakeup model, we consider only down directional data flow.

Fig.1 shows our basic wakeup model based on [15]. In Fig.1 (a), the radio range of node 1 covers node 2, node 3 and node 4. Therefore if node 1 intends to send the data to node 2, node 3 and node 4, they could all receive the data from node 1. However, node 5, node 6, node 7 and node 8 could not receive the data from node 1 directly because they are not within its radio range. When node 1 intends to send data to nodes within its radio range, wakeup of nodes not in its radio range such as node 5, 6, 7 and 8 is wasteful of energy, because idle listening consumes energy between 50% and 100% of receiving energy consumption [12]. In [8], Stemm and Katez show that the ratios of idle:receive:send are 1:1.05:1.4 respectively. Also the Digitan 2 Mbps Wireless LAN module specification illustrates idle:receive:send ratios is 1:2:2.5 [9]. Therefore, when node 1 tries to send the data to nodes 2, 3, and 4, the nodes which are not within its radio range such as node 5, 6, 7, and 8 should be synchronized to enter sleep mode for saving idle listening energy. In sleep mode, sensor nodes turn off their power. At the next step, after node 3 or node 4 receives data from node 1, when node 3 or node 4 intend to send the data to nodes within the radio range of sender, the nodes in level 3 wake up and they are ready to receive data from a node in level 2. Nodes in level 1 such as node 1 go to sleep mode again after they send the data to nodes in level 2. All sensor nodes already know their level and their wakeup/sleep synchronized schedule through the setup of the initial tree structure. When the base station makes the initial tree structure, it sets the wakeup duration of each level in advance.

For example, in Fig.1 (b), if the base station decides that nodes wake up for 1 second for data transmission and wakeup for 1 second for data receiving, the nodes in level 1 wake up for 1 second for receiving data from the base station and also wakeup

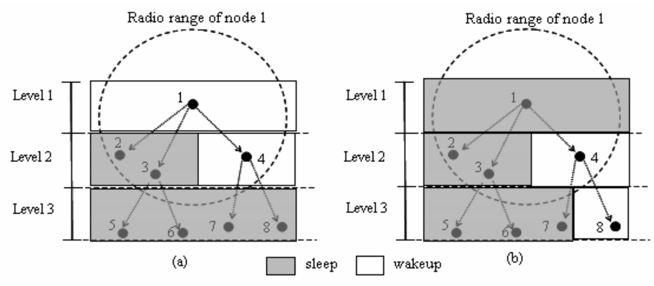


Fig. 2 Odd and even wakeup scheduling example

for another 1 second for transmission to the nodes of level 2. And then nodes of level 1 go to sleep mode. The time interval for data transmission between levels is allocated sufficient time to complete the process, otherwise failure of data transmission will be increased [15].

3 OEWS and DTS

In this section, first we describe the Odd and Even Wakeup Scheduling (OEWS) technique to reduce overhearing energy consumption and then, we describe the Double Tree Structure (DTS) for reducing the delay time caused by OEWS.

3.1 Odd and Even Wakeup Scheduling

We propose the new wakeup schedule named Odd and Even Wakeup Scheduling (OEWS) whose purpose is to reduce the overhearing energy consumption as half the sensor nodes at the same level wake up alternately. In this scheme, sensor nodes having even id number and those having odd id number in the same level have different wakeup time schedule. Sensor nodes having even id number wake up at even time points and sensor nodes having odd id number wake up at odd time. For example, in Fig.2 (a), at a specific even time, sensor node 1 in level 1 can send the data to node 2 and 4 having even number id because node 3 having odd id number is in sleep mode at even time. But at a specific odd time, only node 3 can receive the data from node 1. If node 1 wants to send the data to node 8, node 1 sends the data at the specific even time for sending the data to node 4. Even if node 2 receives the data from node 1 at the same time with node 4, if the destination is not node 2, node 2 goes to the sleep mode. We still save the overhearing energy of node 3 and also we can reduce the wakeup time of node 2. In the next step as shown in Fig.2 (b), after node 4 receives data from node 1, nodes in level 1 fall in sleep mode again and nodes in level 3 wake up for receiving data from nodes in level 2 based on [15]. Nodes 5 and 6 in level 3, they go to sleep mode after they recognize that there is no data from node 3 directed to them. Node 7 and 8 which are children of node 4 wake up alternately in even time and odd time. If node 7 does not receive the data at odd time, node 7 also directly

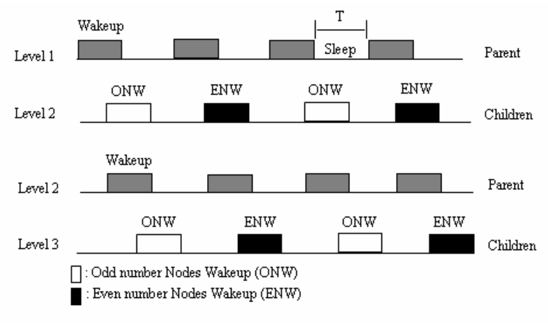


Fig. 3 Odd and even wakeup time scheduling

goes to the sleep mode. And then node 8 can receive the data from node 4 at even time. Hence we can save overhearing energy through the odd-even wakeup schedule. Also, we can decrease the duration of wakeup time for nodes which are not a destination, as they can go to sleep mode if no message is received.

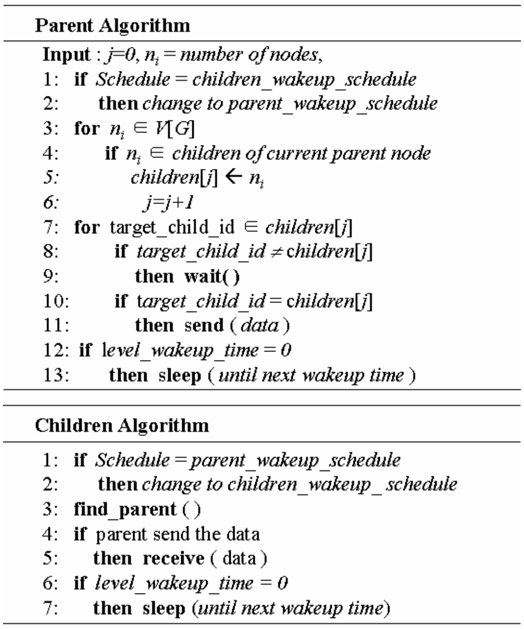
Fig.3 shows the Odd and Even wakeup Scheduling. We use a synchronous wakeup schedule. Level 1 and level 2 in Fig.3 correspond to node 1 and nodes 2, 3, 4 from Fig.2 respectively. First when level 1 transmits data to level 2, the nodes in other levels are in sleep mode. Level 1 follows the schedule for a parent and level 2 follows the schedule for children. A parent node wakes up at every time slot but children nodes wake up alternately. After a node in level 2 receives the data from a node in level 1, the node in level 2 switches from children wakeup time schedule to parent wakeup time schedule. And then the nodes in level 1 are in sleep mode and the nodes in level 3 wakeup and follow the children wakeup time schedule.

In Fig.4, we present the algorithm for a parent node. In lines 1 - 2, a parent node checks its wakeup schedule. If it still follows the children schedule, it changes to parent schedule. In lines 3 – 6, we store the children nodes of the current parent into array c*hildren*[*j*]. We defined G as an undirected graph and V is a set of sensor nodes. In lines 7 – 11, if parent wakeup time matches with a wakeup time of the target child node, the parent node sends the data to the target child node. Otherwise the parent node waits until its wakeup time matches with the target child wakeup time. In lines 12 – 13, if parent level has time out, parent goes into the sleep mode.

In Fig.4, we also show the algorithm for children nodes. In lines 1 -2 they check their wakeup schedule. If children follow the parent schedule, it changes to children schedule. In line 3, each child tries to find the parent node. In lines 4 – 5, they wait for the data from the parent node. If a specific child node receives the data from the parent node, it becomes a new parent and other sibling nodes in same level go into sleep mode.

3.2 Trade-off between Energy Saving and Delay Time

When we try to send the data from a source to a destination in wireless sensor networks, there is a delay time. We assume the delay time based on [10]. Delay time is the time elapsed between the departure of a data packet from the source sensor and



```
Parent Algorithm
Input : $j$=0, $n_i$ = number of nodes,
if Schedule = children_wakeup_schedule
    then change to parent_wakeup_schedule
for $n_i \in V[G]$
    if $n_i \in$ children of current parent node
        children[$j$] ← $n_i$
        $j=j+1$
for target_child_id $\in$ children[$j$]
    if target_child_id $\neq$ children[$j$]
        then wait()
    if target_child_id = children[$j$]
        then send ( data )
if level_wakeup_time = 0
    then sleep ( until next wakeup time )
```

```
Children Algorithm
if Schedule = parent_wakeup_schedule
    then change to children_wakeup_ schedule
find_parent ( )
if parent send the data
    then receive ( data )
if level_wakeup_time = 0
    then sleep (until next wakeup time)
```

Fig. 4 Algorithm for the parent and children node

its arrival to destination [9]. Therefore we can denoted the delay by $DT(s,d) = (qd+td+pd+wd)\times Nd(s,d)$, where qd is queuing delay, td is transmission delay, pd is propagation delay and wd is waiting delay until the receiver node wakes up. $Nd(s,d)$ denotes total number of data disseminators on the routing path between the source node ‘s’ and the destination node ‘d’.

In OEWS, because we use the wakeup scheduling for the nodes to wake up alternately, it causes longer delay time. Therefore we suggest the Double Tree Structure called DTS to reduce the average delay time.

3.3 Double Tree Structure

For OEWS, we make two Rings topologies for the routing tree structure based on [11]. Two Rings topology consists of Short Rings topology called SRT and Long Rings topology called LRT. Rings topology makes a tree structure based on the radio range.

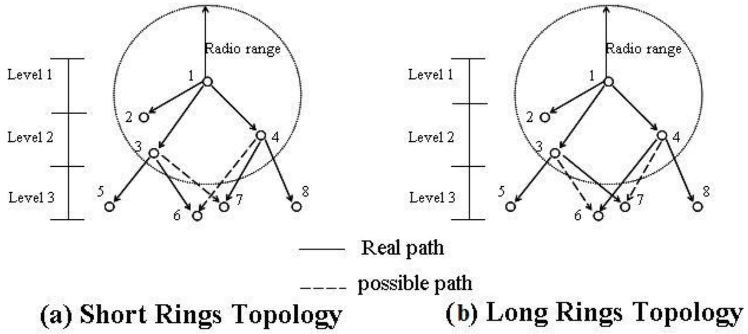


Fig. 5 Short rings and long rings topology

(1) Short Rings Topology and Long Rings Topology

Fig. 5 (a) shows Short Rings Topology (SRT). SRT starts from the base station. In the first step, all nodes within the radio range of the base station become children nodes of the base station. For example, in Fig. 5 (a), the only node within the radio range of the base station is node 1. Therefore node 1 becomes a child node of the base station. In the second step, all nodes within the radio range of node 1 become children nodes of node 1. There are nodes 2, 3, and 4. In the third step, nodes 5, 6 and 7 become children nodes of node 3 because these are within the radio range of node 3. And nodes 6, 7, and 8 become children node of node 4 because these are within the radio range of node 4. Notice that node 6 and node 7 are included as children of both node 3 and node 4. In this case, we can divide them into two Rings topology with distance from parent node to children node. In the view of node 6, node 3 is the closest parent node. Therefore, if node 6 becomes a child of node 3, this topology is Short Rings topology. Otherwise, if node 6 is connected with node 4 which is the most far away from node 6 within the radio range, this topology is Long Rings topology. Fig. 5(b) shows Long Rings Topology.

Our wakeup scheme uses both SRT and LRT in the wireless sensor network. Therefore, it is possible that there are several routing paths. For example, if the base station intends to send the data to node 6, in the first step, the base station sends data to node 1. In the second step, node 1 could send the data to node 3 or node 4. Node 4 can connect to node 6 through the LRT, and node 3 can connect to node 6 through the SRT. Therefore it does not need to wait until odd nodes wake up or even nodes wake up. If there is only one routing path, node 1 has to wait for even nodes wake up or odd nodes wakeup. As a result, we can reduce the delay time.

Even though SRT and LRT are decided by distance, energy consumption for transmitting, receiving and overhearing is the same whether we use SRT or LRT. Because all sensor nodes have the same fixed radio range, radio range of SRT and

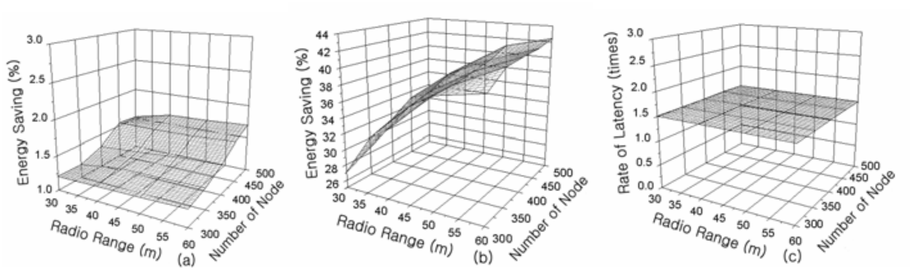


Fig. 6 OEWS vs S-MAC. (a) Energy saving in total energy consumption. (b) Energy saving in overhearing. (c) Latency

LRT is the same. Also, whether we use SRT or LRT, the number of hops from base station to destination is the same. Even if we change the routing path from SRT to LRT or from LRT to SRT, it does not change the number of hops or the radio transmission range.

4 Simulation

In this section, we present simulation results of OEWS and DTS. Our simulation results show that OEWS helps to reduce the overhearing energy consumption and DTS decreases the data delay time. Therefore, we evaluated the efficiency of energy consumption and latency comparing with S-MAC protocol [1]. In S-MAC one of the sources of wasted energy which they tried to reduce is the overhearing energy.

In the experiments, we randomly spread the homogeneous sensors in a 300 × 300m^2 sensor field area. All sensor nodes have the same fixed radio range and same energy. We use the DTS with both Short Rings topology and Long Rings topology for the initial routing tree structure. For measuring the energy consumption for transmitting, receiving, and overhearing data, we used the energy model based on [6].

4.1 Efficiency to Reduce Overhearing Energy

In one experiment, we measured the rate of energy saving comparing with S-MAC protocol. Even though S-MAC protocol already reduced the overhearing energy, the experiment result shows that OEWS reduces overhearing energy more than S-MAC. Fig.6 (a) and (b) show the energy saving results of OEWS comparing with S-MAC. We then increased the number of sensor nodes from 300 to 600 and the radio range from 30m to 60m. In Fig.6 (a), we compare the energy saving rate in total energy consumption including transmitting, receiving and overhearing. This result shows that OEWS reduces up to 1.7% more energy than S-MAC protocol. With the high density of sensor nodes, OEWS produces more saving of overhearing energy consumption. Fig.6 (b) shows the result where only compare the overhearing energy

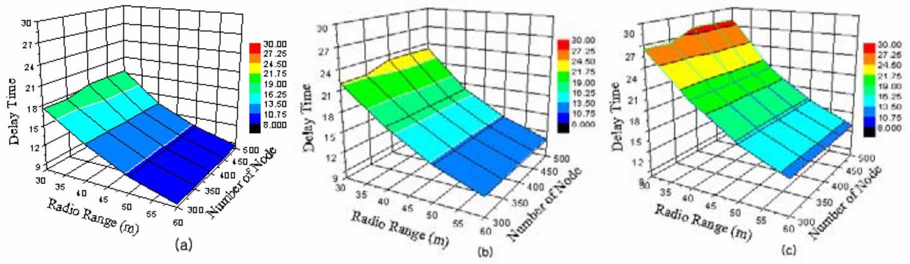


Fig. 7 (a) Delay time on S-MAC. (b) Delay time on DTS. (c) Delay time on OEWS

with S-MAC protocol. We see that OEWS can save the overhearing energy up to 43% more than S-MAC protocol. In this result, the more density of sensor nodes, the larger the decrease in overhearing energy consumption. Therefore, OEWS is more suitable for high density sensor networks than low density sensor networks.

4.2 Effect on Latency between OEWS and S-MAC

In this section, we analyze the data latency between OEWS and S-MAC protocol. In Fig.3, for example, when a node in level 1 detects some events happening between the T period which is the duration of sleep, the node in level 1 waits until its next wakeup time. The probability of occurring event between T periods is uniformly distributed. Therefore we represent the uniform distribution between A and B as X~U[A,B] based on [14]. X is random delay time. A and B are the smallest delay time and the largest delay time respectively. S-MAC has very similar performance to wakeup synchronized time where all the nodes in the sensor networks wake up and go to sleep mode at the same time with the same wakeup time. Hence delay time of S-MAC is represented by the following:

$$X \sim U[\,(h-1)T,\, hT\,] \qquad (6)$$

Therefore, average delay time is :

$$E(X) = (h - \tfrac{1}{2})T \qquad (7)$$

In formulas (6) and (7), h means the number of hops.

In OEWS, half of nodes of all sensor nodes wake up and then the other half wake up. Therefore a node in OEWS takes two times waiting time until the next wakeup time. Therefore, we can represent OEWS delay by the following:

$$X \sim U[\,(h-1)T,\, 2hT\,] \qquad (8)$$

The average delay time of OEWS is the following:

$$E(X) = (\tfrac{3}{2}h - \tfrac{1}{2})T \qquad (9)$$

Fig.6 (c) shows the result of a average latency in OEWS comparing with S-MAC. We simulated with number of nodes from 300 to 600 and radio range from 30 to 60m.

In this environment, latency of OEWS is 1.51 times latency of S-MAC protocol. Hence this simulation shows that there is a trade-off between overhearing energy consumption and latency. In the next section, we show that DTS which we proposed as a routing tree structure reduces the latency.

4.3 Effect on DTS in OEWS

We use the new routing tree structure called Double Tree Structures (DTS) to reduce the latency on OEWS. In this section, we compare latency between OEWS with DTS and OEWS without DTS. Fig. 7(b) shows the delay time of OEWS with DTS and Fig. 7(c) shows the delay time of OEWS without DTS. From these results, we know that DTS works to reduce the latency. DTS has Long Rings topology and Short Rings Topology. Therefore, there are two routing trees in the sensor network. If some nodes have alternative paths to the destination, we can reduce the waiting time. Hence average delay time of DTS with OEWS is represented by the following:

$$E(X) = \frac{(h-\frac{1}{2})T + (\frac{3}{2}h - \frac{1}{2})T}{2} \qquad (10)$$

where, h is number of hops and T is the time duration of sleep mode.

Also, we compare latency between OEWS with DTS and S-MAC. When we compare S-MAC delay time on Fig. 7(a) with OEWS without DTS on Fig. 7(c), even though OEWS reduce overhearing energy, OEWS has more delay time by about 50%. But when we used DTS on OEWS, OEWS with DTS has more delay time over S-MAC by about 13%. Therefore, we improved the trade-off between overhearing energy consumption and latency with proposed OEWS and DTS.

5 Conclusion

In this paper, we have proposed OEWS for reducing the overhearing energy consumption with different wakeup times and DTS for decreasing the latency of OEWS with double tree structure. Even if OEWS advantage is to reduce the overhearing energy consumption, there is a delay time because of a trade-of between energy saving and delay time. But DTS is useful to reduce the delay time. In DTS, Long Rings topology and Short Rings topology have the same number of hops from any node to the base station. Therefore using either Long Rings topology or Short Rings topology for routing path, the number of hop does not have effects on delay time.

Our simulation results also show OEWS and DTS have good performance. OEWS and DTS are more suitable for high density sensor network. Overhearing energy consumption is high when nodes are having many neighborhood sensor nodes.

References

1. W. Ye, J. Heidemann, D. Estrin, “An energy-efficient MAC protocol for wireless sensor netowkrs”in INFOCOM 2002: Proceedings of the Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies, Vol. 3, IEEE, June 2002 .
2. Mohamed A. Sharaf, Jonathan Beaver, Alexandros Labrinidis, Panos K. Chrysanthis, “Balancing energy efficiency and quality of aggregate data in sensor networks” VLDB journal, 2004 13:384-403.
3. Ramanan Subramanian, Faramarz Fekri, “Sleep scheduling and lifetime maximization in sensor networks: fundamental limits and optimal solutions” Proceedings of the fifth international conference on Information processing in sensor networks, IPSN 06.
4. Ossama Younis, Sonia Fahmy, “Distributed clustering in ad-hoc sensor networks:a hybrid, energy-efficient approach,” IEEE Transaction on Mobile Computing Vol.3, No.4, 2004.
5. Byoungyong Lee, Kyungseo Park, Ramez Elmasri, “Energy Balanced In-Network Aggregation Using Multiple Trees in Wireless Sensor Networks”, Consumer Communications and Networking Conference, Jan. 2007.
6. Prithwish Basu, Jason Redi, “Effect of overhearing transmissions on energy efficiency in dense sensor networks”, Proceedings of the third international symposium on Information processing in sensor networks, IPSN 04.
7. W.R. Heinzelman, A. Chandrakasan, H.Balakrishnan, “Energy-efficient communication protocol for wireless microsensor networks”, Proceedings of the 33rd Hawaii International Conference on System Sciences 2000.
8. Mark Stemm, Randy H Katz, “Measuring and reducing energy consumption of network interfaces in hand-held devices,” IEICE Transactions on Communications, vol. E80-B, no.8, pp1125-1131, Aug.1997.
9. Oliver Kasten, Energy Consumption, http://www.inf.ethz.ch/~kasten/ research/bathtub/ energy_consumption.html, Eldgenossische Technische Hochschule Zurich.
10. Habib M. Ammari and Sajal K. Das, “Trade-off between energy savings and source-to-sink delay in data dissemination for wireless sensor networks,” Proceedings of the 8th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems, MSWiM '05.
11. S. Nath, P. Gibbons, S. Seshan, Z. Anderson, “Synopsis Diffusion for Robust Aggregation in Sensor Networks,” In Proc. 2nd ACM SenSys, pp 250-262,2004.
12. Qin Wang, Zygmunt J. Haas, “BASS: an adaptive sleeping scheme for wireless sensor network with bursty arrival,” Proceedings of the 2006 international conference on wireless communications and mobile computing, IWCMC '06.
13. Wei Ye, John Heidemann, Deborah Estrin, “Medium access control with coordinated adaptive sleeping for wireless sensor networks,” IEEE/ACM Transaction on Networking, vol. 12, No.3, June 2004.
14. A. Keshavarzian, H. Lee, L. Venkatranman, “Wakeup Scheduling in Wireless Sensor Networks,” MobiHoc ’06, 2006, Florence, Italy.
15. S. Madden, M. Franklin, J. Hellerstein, W. Hong “TAG: a Tiny Aggregation Service for Ad-Hoc Sensor Networks,” In OSDI, 2002.