Coverage Maximization with Sleep Scheduling for Wireless Sensor Networks

Chanrit Danratchadakorn[1] and **Chotipat Pornavalai**[2], Non-members

ABSTRACT

Sleep scheduling mechanisms have been widely used in wireless sensor networks so as to extend the lifetime of networks. Sensors are able to decide to be either in active or sleep mode to save the energy. Sensing coverage area is an important factor for some applications such as Intrusion Detection. It is necessary to have the full-sensing-covering set of active nodes on these applications. In this paper, we propose the Coverage Maximization with Sleep Scheduling protocol (CMSS) which is a decentralized protocol and maximize sensing coverage of the network. In our proposed solution, the area of network is divided into grid cells. Each sensor creates a neighbor table and transforms into cell-value table. These tables are used to make decision which mode it should be on each sensor. Simulation results show that CMSS not only consumes less overhead energy than Maximum Sensing Coverage Region (MSCR), but also has a lower number of selected active nodes. Besides, communication range of sensors does not affect to the efficiency of networks like the Layered Diffusion-based Coverage Control protocol (LDCC) which exploits hop count information.

Keywords: Sensing Coverage, Sleep Scheduling Mechanism, Wireless Sensor Networks

1. INTRODUCTION

Wireless sensor networks contain many sensors of which the batteries, sensing power and transmission power are limited. There are so many applications of wireless sensor networks such as Environment Monitoring, Battlefield Surveillance, Target Tracking and Medical Analysis [1][2]. Each of sensors detects data within its sensing range and sends them to the base station so that the users can transform these data into useful information. Due to limited power of sensors, energy conservation is the biggest problem in wireless sensor networks. Hence, there are many researches proposing energy-efficient mechanisms in order to prolong the lifetime of sensors such as clustering networks [3-5], appropriate sink placement [6-8] and sleep scheduling & sensor state planning [9-11] which is considered in this paper. In sleep scheduling mechanisms, each node can switch between the two major modes which is either on-duty mode (also called as active mode) or off-duty (also called as sleep mode).

Quality of Service (QoS) is a set of standards and processes for ensuring quality performance for applications such as delay, reliability, and integrity. In wireless sensor networks, sensing coverage area is also a part of QoS [12-14]. Some applications such as target tracking and movement detection need the sensors to cover all areas in the network. If the network has some coverage holes, it will affect to the quality of service of networks.

In this paper, we propose Coverage Maximization with Sleep Scheduling protocol which is called as CMSS. Obviously, the network with sensors randomly deployed must have many overlapped sensing areas. These areas have the same data such as temperature and humidity in environment monitoring applications. Hence, a sensor with its own sensing area fully overlapped by others is able to operate in sleep mode. The basic idea of the protocol is to find sensor nodes which do not reduce the coverage area when it enters sleep mode. In the other words, sensing coverage area of network is still fully covered by only selected active nodes. Fig. 1 shows an example of full-covering active sensors while all other sensors are sleeping.

The rest of the paper is organized as follows. Section 2 provides overview of related works on coverage and sleep scheduling protocols. Section 3 describes the system models and our problem statements. Section 4 presents the details of proposed method. Performance evaluation and conclusion are in the section 5 and 6 respectively.

2. RELATED WORKS

C.Zhu et al. [15] proposed the basic knowledge of coverage and connectivity issues in wireless sensor networks. They described 2 sensing models i.e., the binary disc sensing model and the probabilistic sensing model. The simplest model was the binary disc model because the cover point could be either *covered* (*Binary:1*) if it is within sensing radius of sensors or uncovered (*Binary:0*) if it is ‘not’ within sensing radius of sensors. The Probabilistic Sensing model was more actual perception which could be taken as an ex-

Manuscript received on August 16, 2015.
Final manuscript received on December 9, 2015.
[1,2] The authors are with Faculty of Information Technology, King Mongkut’s Institute of Technology Ladkrabang, Thailand, E-mail: d.chanrit@gmail.com and chotipat@it.kmitl.ac.th

tension of the previous sensing model. The quality of sensing gradually attenuates with increasing distance. Hence, the coverage value of point is in between 0 and 1.

TABU-RCC [9] is a centralized algorithm. TABU search technique is used to find a near-optimal state of sensors which is divided into 3 modes i.e. active, sleep and cluster head. Each node has to periodically send out a control message containing its own information, such as residual energy and ID to the base station so that the base station can use these data to make a decision which state that sensor deserved to be. TABU search movement will be computed until a predefined number of iterations in order to get a lower score from the minimized score function. The base station then broadcasts a state message which is routed to all the sensors via cluster head nodes. However, being a centralized algorithm, TABU-RCC comes with a large amount of communication overhead. Hence, a network using this protocol is limited to extend a number of sensors and size of network-field.

LDCC [10] is a decentralized algorithm used to control the coverage of networks by using the sleep scheduling mechanisms. The basic ideas of LDCC are to use a triangular tessellation as a coverage controlling and to utilize the hop count information as a part of decision whether it should be either an active node or sleep node. The operation of LDCC is divided into rounds. The base station first broadcasts an active message. An active message indicates that the sender has become an active state; inside the message contains information such as sender's hop count (H_s), transmission power. A sensor node Ni which received an active message compares its own hop count (H_i) which is initially set to a high number, with the sender's hop count (H_s). If $H_s < H_i$, then N_i randomly sets back-off time T_i so that this node will wait for the expiration of timer T_i before sending out an active message. If $H_s = H_i, N_i$ adds the number of times that received active messages from nodes with the same hop count. When this number of N_i equals to 2, Node N_i goes into sleep state. Otherwise, if $H_s > H_i, N_i$ will ignore that message. Although location information of sensors is not needed to be used in LDCC, the set of selected active sensors cannot be guaranteed to be fully covered the whole area of the network.

MSCR [11] is a distributed gossip-based sensing-coverage-aware algorithm used to solve the sensing coverage problem. This protocol requires each node to know its location information so that the network which uses MSCR can guarantee 100% coverage area. Each of sensors (called as N_i) first creates the overlapped neighbor set, $O(S_i)$. After that, each sensor node starts sending a gossip message to all of the members in its own $O(S_i)$. Then it waits for the reply message from these active members. When the sensor N_i receives the reply message from N_j, it

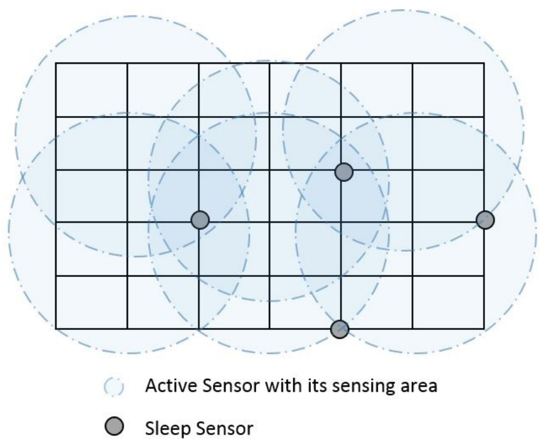

Fig.1: *A Network with full-covering active sensors.*

calculates the boundary arcs between itself and N_j and transform the arcs into the angle [0; 2Pi]. After receiving reply gossip message from all of active members, the sensor will consider the conditions by creating the union of boundary arcs. If neighbor nodes in $O(S_i)$ cover all its sensing range k times, N_i will enter sleep state. Due to sufficient condition in MSCR, each of the sensors in overlapped neighbor set has to be closed within sensing range. Otherwise, the area near to the sensor N_i may not be fully covered. Therefore, the number of active nodes after processing is higher than necessary. The energy consumption during data communication phase is also high as well.

C3 [16] is an integrated protocol for coverage, connectivity and communication (C3) in wireless sensor networks. Its function is similar to LDCC using triangular tessellation but the network is formed as connection of virtual rings and clusters. Each node can be either in fully active or sensing only or sleep mode. Similar to LDCC, C3 also cannot guarantee to be fully covered the whole area of the network.

In this paper, we propose the Coverage Maximization with Sleep Scheduling protocol (CMSS) to maximize sensing coverage of the network. Similar to LDCC, MSCR and C3, the proposed protocol is distributed but unlike the other protocols, the main advantages of CMSS is that it has low overhead energy consumption than and less number of active nodes, while still be able to guarantee the coverage area in the network.

3. SYSTEM MODELS & PROBLEM STATEMENTS

Before explaining the proposed protocol, we first describe the network models, energy consumption model and problem statements.

3.1 Network Model

We assume the following properties about the wireless sensor network.

- Sensor nodes are randomly deployed into the network at the intersection of grid cells and have the same initial energy.
- Sensor nodes are fixed and know their own locations.
- All sensors have the same sensing range R_s and communication range R_c. while R_c is equal to or more than $2R_s$ because of connectivity issues [17].
- Each node is assigned with a unique identifier (ID).
- We assume ideal MAC layer conditions, i.e., perfect transmission of data on a node-to-node wireless link.

3.2 Energy Consumption

We use energy model proposed in [18] to measure energy consumption for proposed and related protocols. There are two propagation models (1): free space model (d^2 power loss) and two-ray ground propagation model (d^4 power loss).

$$E_{TX}(l,d) = \begin{cases} lE_{elec} + l\varepsilon_{fs}d^2, & d < d_0 \\ lE_{elec} + l\varepsilon_{mp}d^4, & d \geq d_0 \end{cases} \tag{1}$$

Eq. (1) shows $ETX(l,d)$ which is the energy consumption in the transmitter of sender node that sends l bits of data with distance d to the receiver. E_{elec} depends on many factors such as digital coding, modulation. The $\varepsilon_{fs}d^2$ and $\varepsilon_{mp}d^4$ depend on the distance (d) between transmitter and receiver where ε_{fs} and ε_{mp} are the amplifier energy factors for free space and multi-path fading channel models respectively. d_0 is the threshold distance depending on the environment. Energy consumption for receiving an l-bits message at the receiver is shown in Eq. (2).

$$E_{RX}(l) = lE_{elec} \tag{2}$$

3.3 Sensing Model

We use a binary probability function (also called Binary Disc Sensing Model) [15]. An active sensor n_i is able to detect any event occurring within its sensing range R_s with the detection probability 1. In contrast, if an event occurring outside R_s, it would not be able to detect that event.

3.4 Problem Statements

Consider a WSN consisting of N homogeneous sensor nodes $n_1, n_2, \ldots, n_N$, we divide the area of network into grid cells $Cell_{x,y}$ where x and y are the coordinate at the intersection of the grid cell as shown in Fig. 2. Each sensor has two states s_i , either in active mode (1) or sleep mode (0)

$$s_i = \begin{cases} 1 \text{ if sensor } i \text{ is active} \\ 0 \text{ otherwise (sleep)} \end{cases} \tag{3}$$

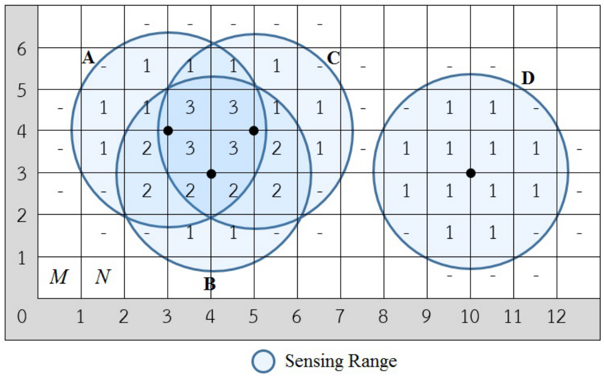

Fig.2: *Example of Covered-Cell in the network size 12x6.*

Definition 1. The reference of cell ($Cell_{x,y}$) or $p_{x,y}$ is at the left-bottom coordinate (x, y) of the cell. For example, the index of cell 'M' in Fig. 2 is $Cell_{0,0}$, and the index of cell 'N' is $Cell_{1,0}$.

Definition 2. $Cell_{x,y}$ is covered by sensor node n_i when the following conditions are satisfied:

$$d(p_{x,y}, n_i) \leq R_s \tag{4a}$$
$$d(p_{x+1,y}, n_i) \leq R_s \tag{4b}$$
$$d(p_{x,y+1}, n_i) \leq R_s \tag{4c}$$
$$d(p_{x+1,y+1}, n_i) \leq R_s \tag{4d}$$

where $p_{x,y}$ is the reference of $Cell_{x,y}$ as previously explained, $d(p_{x,y}, n_i)$ is the Euclidean distance between coordinate x, y and n_i's coordinate, and Rs is sensing radius of sensors.

In (4a-4d), it can be summarized that all corners of the cell must be within sensor's sensing radius.

Let A_i is the set of cells that are covered by sensor i (satisfied conditions 4a-4d). The problem is to maximize the coverage area

Find state of s_i in each round of operation such that

$$\text{Maximize} \quad : \bigcup_i (A_i.s_i), \forall i = 1 \ldots N \tag{5}$$

We need to find the optimal state of sensors that maximize coverage area of full-covering set of active sensors. Besides, the network lifetime must be maximized; we need not only low overhead energy dissipation as much as possible but also low data communication energy dissipation which depends on a number of selected active sensors. Therefore all the solutions that maximize by Eq. (5), we select the solution that minimize number of active sensors

$$\text{Minimize} \quad : \sum_{i=1}^{N} s_i \qquad (6)$$

If there are several solutions that satisfied Eq.5 and Eq.6, then we would select one with higher residual energy to balance their energy usages and maximize sensor's life time, where E_i is the residual energy of sensor i.

$$\text{Maximize} \quad : \sum_{i=1}^{N} (s_i.E_i) \qquad (7)$$

4. PROPOSED CMSS PROTOCOL

4.1 Coverage and Tables Setup

Each sensors n_i has two tables, i.e. neighbor table and cell-value table. Neighbor table is used to record the overlapping neighbor nodes. It contains three information.

- Node ID (N_{id}) : node id of sensors within distance $2R_s$ (including itself ID)
- Cells' coordinates (reference) covered by N_{id} (satisfied by conditions 4a-4d)
- Boolean variable ($N_{id}.active$): indicates whether node N_{id} is active or not.

Cell-value table also contains following three information.

- $Cell_{x,y}$: the list of cell covered by node n_i (table owner)
- Covered Node ID : ID of n_i's active neighbor nodes which covers $Cell_{x,y}$
- Cell Value : the total number of node covering $Cell_{x,y}$

For example, in Fig. 2, the network sized is 12x6 and containing 4 nodes, 'A' 'B' 'C' and 'D' locating on [3,4], [4,3], [5,4] and [10,3], respectively.

In this figure, if sensor covers full areas of cell, the cell value will be counted as 1 and more if there are other nodes also covered that cell. Otherwise, the cell will be called as uncovered-cell ('-').

Table 1-2 shows the Neighbor Table and Cell Value Table for node 'A' after finishing information exchanging process among neighboring sensors. The process of the proposed protocol will be explained in the next sub-section.

4.2 Coverage Maximization with Sleep Scheduling (CMSS)

Only the first round, all sensors have to exchange their own coordinates by sending out a coordinate message with transmission power $2R_s$. This message contains sender id and sender's coordinate. Nodes receiving this message calculate and update their own neighbor table as described in previous subsection.

For each round, each sensor n_i first clear its cell-value table and set $N_i.active$ in neighbor table to $FALSE$. After that, n_i sets a hello-exchange timer to $t_{helloExchange}$ and broadcasts a hello message with the transmission power for distance $2R_s$.

Table 1: *An example of neighbour table for node 'A'.*

Neighbor Table for node A		
ID	***Covered Cell***	***Alive***
A	[1,3] , [1,4] , [2,2] , [2,3] , [2,4] , [2,5] , [3,2] , [3,3] , [3,4] , [3,5] , [4,3] , [4,4]	True
B	[2,2] , [2,3] , [3,1] , [3,2] , [3,3] , [3,4] , [4,1] , [4,2] , [4,3] , [4,4] , [5,2] , [5,3]	True
C	[3,3] , [3,4] , [4,2] , [4,3] , [4,4] , [4,5] , [5,2] , , [5,3] , [5,4] , [5,5] , [6,3] , [6,4]	True

Table 2: *An example of cell value table for node 'A'.*

Cell Value Table for node A		
Cell	***Covered Node ID***	***Cell Value***
[1,3]	A	1
[1,4]	A	1
[2,2]	A , B	2
[2,3]	A , B	2
[2,4]	A	1
[2,5]	A	1
[3,2]	A , B	2
[3,3]	A , B , C	3
[3,4]	A , B , C	3
[3,5]	A	1
[4,3]	A , B , C	3
[4,4]	A , B , C	3

A *hello message* contains sender *id*. It also indicates that the sender is currently active mode (also still alive). $t_{helloExchange}$ is a predefined time set which must be long enough for all nodes to exchange their information. When node n_i received a hello message from n_j, it updates its own neighbor table by setting n_j.active to TRUE. After $t_{helloExchange}$ waiting time expired, n_i updates its Cell-value table.

Sensor node having one or more of cell value with value 1, or min(*cell_value*) equals to 1, is going to be ACTIVE silently and finished its process for this round.

For a sensor node n_i, its waiting time W_i is expressed as Eq. (8), where W_{max} is the predefined maximum waiting time, E_i is the residual energy of the node n_i, and E_{max} is the maximum or initial energy of node. When n_i's waiting time is expired, n_i sends out a sleep message containing its own ID with transmission power for distance $2R_s$ and enters sleep mode. Any nodes that the waiting time has not expired yet (still working as active node), after receiving a sleep message from n_j, then update their own neighbor tables by setting $N_j.active = FALSE$ and update new

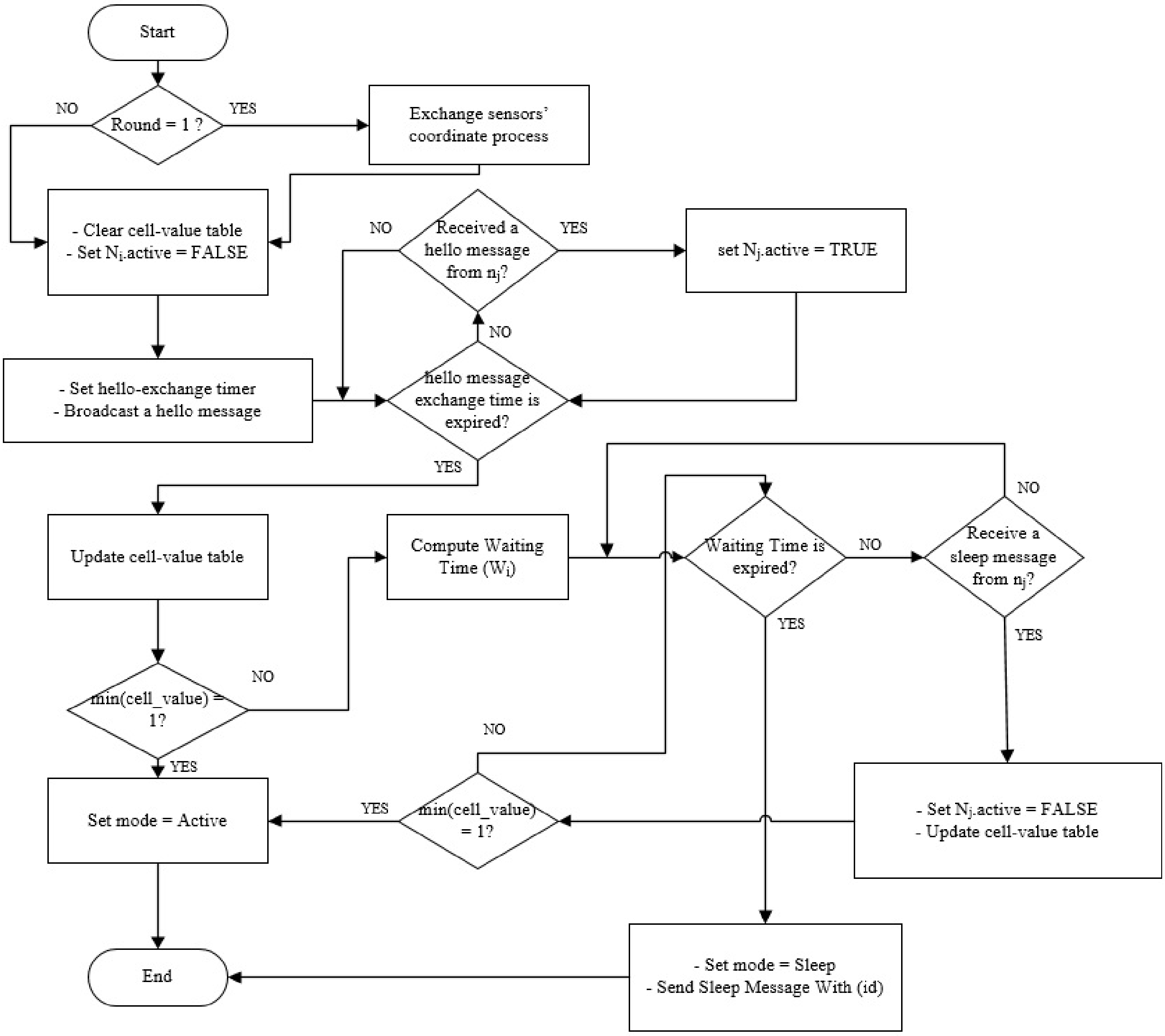

Fig.3: *Operation flowchart of CMSS Protocol.*

cell-value table again.

$$W_i = W_{max} \left(\frac{E_i}{E_{max}} \right) \tag{8}$$

After having new cell-value table, if the waiting node has a cell that Cell Value = 1, or $min(cell\_value) = 1$, the node is going to be ACTIVE silently. Otherwise, it would wait for the expiration of waiting time. The flowchart for CMSS is given in Fig. 3. This proposed flowchart ensures that the network achieves the optimization formulation as Eq. (5) because sensor node i can enter sleep mode only when there is at least one neighbor sensor node covering all the cells of sensor i. If $min(cell\_value)$ is greater than 1, then it sets another waiting time, W_i, depending on its own residual energy. Waiting for the expiration of waiting time then enter sleep mode leads to not having the number of active sensors more than necessary as required by Eq. (6).

This waiting time achieves the optimization formulation as Eq. (7) by making the sensors with $min(cell\_value)$ greater than 1 balancing energy as much as possible. Sensors with low residual energy would have less waiting time to enter sleep mode than sensors with high residual energy. However, we are unable to control the energy fairness of sensors of which $min(cell\_value)$ is equal to 1 because these sensors have to enter active mode every round in order to make the network fully covered.

5. PERFORMANCE EVALUATION

Our simulations are performed using MATLAB. We compare the CMSS with LDCC [10] and MSCR [11] with k equal to 1. Both of them have been briefly explained already in Section 2. The important parameters are shown in Table 3.

To evaluate the performance of these protocols, we vary the number of nodes, N, and R_c parameters. N

Table 3: *Simulation Parameters.*

Parameter	*Value*
Network size	100 x 100 m
Grid unit length	1 m
Number of sensor nodes (N)	100 - 1000
Node Deployment	Random
Initial Energy	1 J
E_{elec}	50 nJ/bit
ε_{fs}	10 pJ/bit/m^2
ε_{mp}	0.0013 pJ/bit/m^4
Overhead Packet Size	32 bytes
R_s	10 m
R_c	20 m for MSCR and CMSS 20, 30 m for LDCC

are 100, 200, 300, 400, 500, 1000. R_c is 20m and 30m. The experiments take 500 iterations and are averaged over 10 results. In CMSS and MSCR, the results show only for $R_c = 20$, longer R_c would not gain any performance advantages. On the other hands, LDCC, the value of R_c highly affects to the performance of network. Since we want the ensure connectivity among the sensors for all protocols (R_c must be at least $2R_s$), therefore Rc less than 20 m is not evaluated in the simulation. Please note that LDCC [10] and MSCR [11] as well as CMSS, the values of R_c and R_s are fixed. In contrast, protocol C3 [16] requires sensor nodes to be equipped with transmitter that can adjust the value R_c at any time. Therefore we neglect C3 in this simulation.

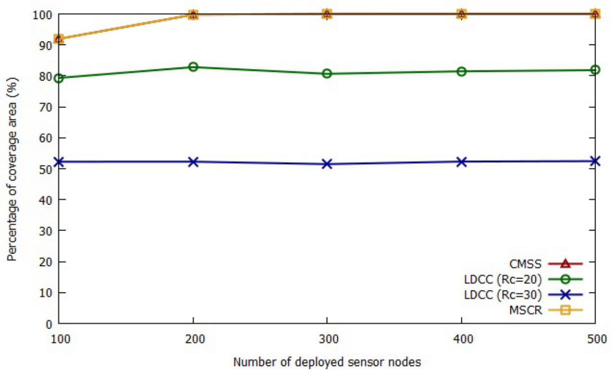

Fig.4: *Percentage of coverage area with different number of deployed sensor nodes.*

5.1 Sensing Coverage Area

In Fig. 4, selected active nodes of our protocol and MSCR has the same coverage ratio as when all nodes are active. Because of randomly deployment, when the number of sensors in the network is not dense enough, i.e., sensor = 100 nodes, the coverage area is not fully covered the whole areas.

Hence, the coverage ratio of our protocol and MSCR are 91.97 percent. For LDCC with $R_c = 20$, the area cannot be fully covered as some nodes are going to be sleep mode when it receives an active message from other nodes with the same hop count twice. The node does not care whether how much they overlap its sensing area. When $R_c = 30$ in LDCC, the coverage area is obviously worse than $R_c = 20$. It shows that with LDCC, using high communication range would reduce coverage area of the network.

5.2 Overhead Energy Consumption

Fig. 5 shows the overhead energy consumption in each round. Obviously, energy dissipation of MSCR is high because every node uses the large energy in order to get information for decision which mode it should be. It has to send out a gossip message to all of the neighbors and wait for a reply message. In CMSS, each node just sends out a hello message. Then it uses the back-off time technique and the node's tables to make a decision. In LDCC, the major problem is also about the value of sensor's communication range (R_c). The overhead energy consumption when $R_c = 30$ is more than $R_c = 20$ approximately 90 percent.

To see that the overhead energy consumption results significantly affect to the network lifetime, in Fig 6, we show the network lifetime (Round) of each protocol. A number of round is calculated by *Initial Energy / Overhead Energy Consumption per Round.*

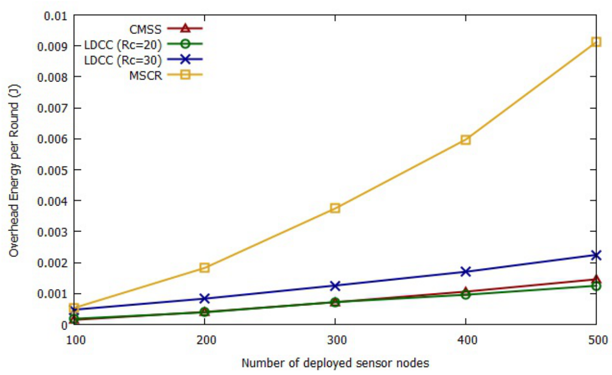

Fig.5: *Overhead energy consumption per round with different number of deployed sensor nodes.*

In fig 6, all sensors initially set their energy to 0.01 J. It shows that CMSS and MSCR which have the same coverage radio, CMSS has network lifetime longer than MSCR approximately 3.6, 4.5, 5.2, 5.6, 6.2, 7.6 times when N = 100, 200, 300, 400, 500 and 1000, respectively.

Although some cases for LDCC when $R_c = 20$ have the network lifetime longer than CMSS, the network area is not be fully covered like CMSS.

5.3 Number of active nodes

This factor also affects to network lifetime. Network with high number of selected active nodes would also have high dissipated energy for transmitting data

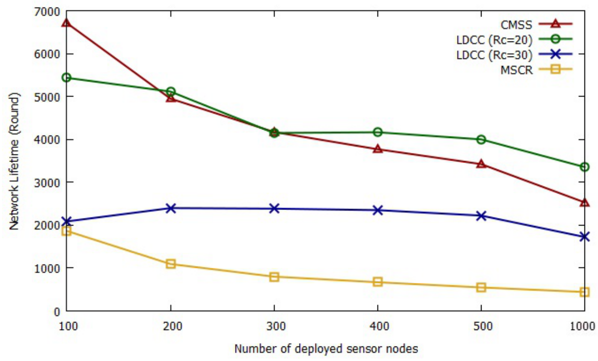

Fig.6: *Network Lifetime (Round) with different number of deployed sensor nodes.*

to the base station. In Fig. 7, between our protocol and MSCR which get the full coverage areas, the number of selected active nodes of our protocol CMSS is less than MSCR because of the overlapped-calculating algorithm. In case of MSCR, they define that a node which is calculated the overlapping area must be within its Rs not '$2R_s$'. This means some nodes will lose opportunities to enter sleep node. In contrast, our protocol defines that a node which is calculated must be within its $2R_s$. It leads to having a less number of active node than MSCR.

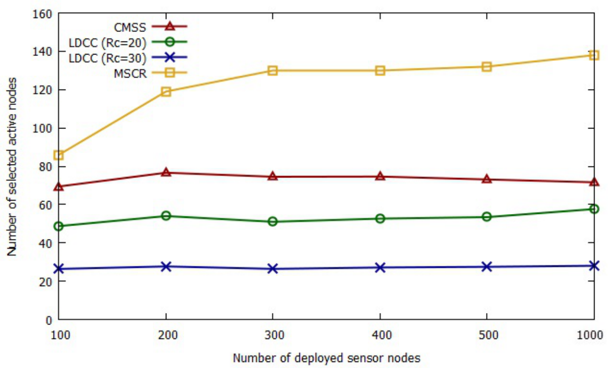

Fig.7: *Number of selected active nodes with different number of deployed sensor nodes.*

Although LDCC has the number of active nodes less than our protocol, the area is not be fully covered like CMSS. Besides, the same weakness of LDCC (value of communication range) still occurs.

5.4 Energy Balancing

In this paper, we use standard deviation (S.D.) to indicate energy balancing among sensors. In Fig 8, CMSS has S.D. less than MSCR because of waiting time calculation. For CMSS, a node with high residual energy will have the waiting time longer than a node with low residual energy. It makes the selected active node have their own residual energy as much as possible.

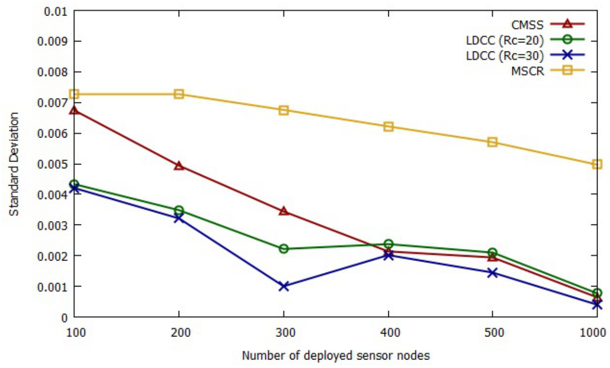

Fig.8: *Standard deviation of nodes' residual energy with different number of deployed sensor nodes.*

However, the reason that LDCC has S.D. less than CMSS and MSCR because the networks using LDCC don't guarantee full-covering network. It means some nodes don't have to become active mode in every single round to save the coverage area of network. For LDCC, the waiting time is randomly set (not dependent on residual energy of a node).

5.5 Data Transmission Error

One of the assumptions made in this paper is perfect transmission of data on a node-to-node wireless link.

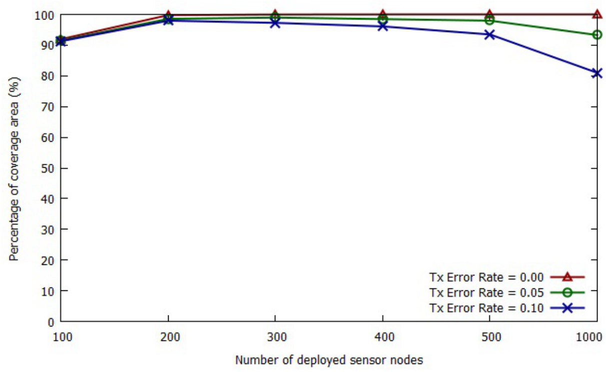

Fig.9: *Percentage of coverage area by CMSS with transmission errors.*

However, in the real implementation, there are some interferences and refraction of electromagnetic. These cause data which are sent from sender not be able to reach to receivers. Hence, we consider a simple uniform transmission error model with fixed transmission error rates. We make experiments by applying the error rates during sensors receiving a sleep message from the other sensor.

In Fig 9, the transmission error causes the network lose some coverage area. When a sensor doesn't receive a sleep message from neighbor node, it always understands that its *cell_value* is more than 1 and it is still waiting for the expiration of waiting time and

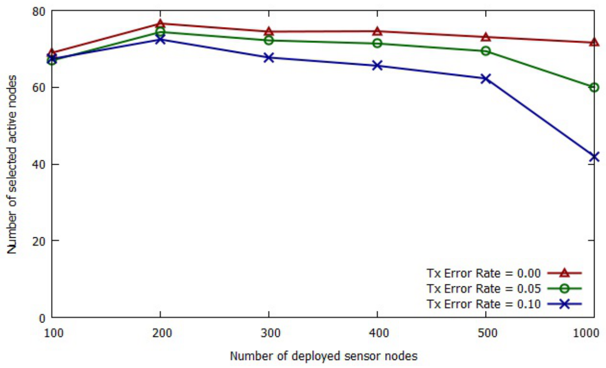

Fig.10: *Number of selected active nodes by CMSS with transmission errors.*

falsely enter sleep mode. Actually, if this sensor received a sleep message, its *cell_value* might be equal to 1 after updating and this sensor should enter active mode.

In Fig 10, there is a trade-off between coverage area ratio and the number of selected active nodes as we explain in 5.3. When there are few sensors in the network such as 100 sensor, the data transmission errors scarcely affect to the network performance. In contrast, when there are many sensors such as 1000 sensors, these errors significantly affect to the network performance. This subject is going to be focused in future works.

6. CONCLUSION & DISCUSSION

6.1 Conclusion

In this paper, we have introduced a sleep-scheduling protocol for wireless sensor networks (CMSS) which can guarantee whether the area of network will be fully covered by selected active sensors. Each sensor exchanges information with its neighboring sensors and set the waiting time. During sensor's waiting time, a sensor can receive a sleep message from neighbor nodes. When a sensor received the messages, it updates its own neighbor and cell value table. If sensor's $min(cell\_value)$ after updating is equal to 1, it becomes active mode silently. Otherwise, it will wait for the expiration of waiting time and then switches its mode to sleep. The results show that networks using CMSS protocol are fully covered by sensors as much as possible, the same as MSCR. Whereas, CMSS can extend the network lifetime more than MSCR significantly. Sensors in the network using CMSS make their own decision by applying back-off timer technique so that they can reduce their overhead energy consumption. Moreover, CMSS can reduce the number of selected active nodes which leads to having a less number of messages used for sending data to the base station.

6.2 Discussion

Using computer simulation may not show some problems occurring in real implementation. Hence, the protocol's procedures may not be able to avoid and fix those problems. The following subjects need to be researched in deep to improve the efficiency of CMSS in the future.

A. Localization Errors

A sensor needs some hardware, such as GPS module, to indicate the location. In the real world, these hardware may have errors. A result of location from hardware doesn't always be equal to real location. In the future, we are going to study about effects to the performance of CMSS protocol.

B. Synchronizing

In this paper, the operation of CMSS is divided into rounds. The operation begins next round when all sensors in current round finish deciding which mode it should be.

C. Hidden Terminal Problem

For hello message exchanging process operating only the first round, if this problem occurs, the network still continues to operate but the neighbor table of some nodes will have incomplete information. Because of this process operate only the first round, those incomplete neighbor table will always be the same. Besides, a sensor having incomplete neighbor table will lose an opportunity to become sleep mode as there are less sensors calculated overlapped area

For a sensor broadcasting a sleep message, a receiver sensor will become sleep mode if it update their own cell-value table after receiving a sleep message and has min($cell\_value$) is equal to 1, in fact. However, if hidden terminal problem occurs, receiver node will understand that its own *cell_value* is still more than 1. Then the node decides not to become sleep mode. This situation leads the network lose the coverage area ratio immediately.

References

[1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless Sensor Networks: A survey," *Elsevier Sci. Comput. Netw.*, vol. 38, pp. 393-422, 2002.

[2] G. Fan, R. Wang, H. Huang, L. Sun and C. Sha, "Coverage-guaranteed Sensor Node Deployment Strategies for Wireless Sensor Networks," *Sensors*, vol. 10, pp. 2064-2087, March 2010.

[3] W.B. Heinzelman, "*Application-Specific Protocol Architectures for Wireless Networks*," Ph.D. Thesis of Massachusetts Institute of Technology, May 2000.

[4] O. Younis and S. Fahmy, "HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks," *IEEE Transactions*

on Mobile Computing, vol. 3, no. 4, pp. 366-376, 2004.

[5] G. Chen, C. Li, M. Ye and J. Wu, “An Unequal Cluster-based Routing Protocol in Wireless Sensor Networks,” *Wireless Networks*, vol. 15, issue 2, pp. 193-207, February 2009.

[6] D.R. Dandekar and P.R. Deshmukh, “Energy Balancing Multiple Sink Optimal Deployment in Muli-hop Wireless Sensor Networks,” *Advance Computing Conference (IACC)*, pp. 408-412, February 2013.

[7] D. Das, Z. Rehena, S. Roy and N. Mukherjee, “Multiple-Sink Placement Strategies in Wireless Sensor Networks,” *Communication Systems and Networks (COMSNETS)*, pp. 1-7, January 2013.

[8] F. Chen and R. Li, “Single sink node placement strategy in Wireless Sensor Networks,” *Electric Information and Control Engineering (ICEICE)*, pp. 1700-1703, April 2011.

[9] A. Chamam and S. Pierre, “On the Planning of Wireless Sensor Networks: Energy-Efficient Clustering under the Joint Routing and Coverage Constraint,” *IEEE Transactions on Mobile Computing*, vol. 8, no. 8, pp. 1077-1086, August 2009.

[10] B. Wang, C. Fu and H.B. Lim, “Layered Diffusion-based Coverage Control in Wireless Sensor Networks,” *Computer Networks*, vol. 53, issue 7, pp. 1114-1124, May 2009.

[11] V.T. Quang and T. Miyashi, “A novel gossip-based sensing coverage algorithm for dense wireless sensor networks,” *Computer Networks*, vol. 53, issue 13, pp. 2275-2287, August 2009.

[12] S. Meguerdichian, F. Koushanfar, M. Potkonjak and M.B. Srivastave, “Coverage problems in Wireless Ad-hoc Sensor Networks,” *INFOCOM*, vol. 3, pp. 1380-1387, April 2001.

[13] K.R. Karkvandi, E. Pecht and Y. Pecht, “Effective Lifetime-Aware Routing in Wireless Sensor Networks,” *IEEE Sensors Journal*, vol. 11, no. 12, pp. 3359-3367, December 2011.

[14] P.J. Wan and C.W. Yi, “Coverage by randomly deployed Wireless Sensor Networks,” *IEEE Transactions on Networking*, vol. 14, pp. 2658-2669, June 2006.

[15] C. Zhu, C. Zheng, L. Shi and G. Han, “A survey on coverage and connectivity issues in wireless sensor networks,” *Journal of Network and Computer Applications*, vol. 35, issue 2, pp. 619-632, March 2012.

[16] M. Akhlaq, T. R. Sheltami and E. M. Shakshuki, “An integrated protocol for Coverage, Connectivity and Communication (C3) in Wireless Sensor Networs,” *the 27th International Conference on Advanced Information Networking and Applications (AINA)*, pp. 606-612, March 2013.

[17] A. Ghosh and S.K. Das, “Coverage and connectivity issues in wireless sensor networks: A Survey,” *Pervasive and Mobile Computing*, vol. 4, issue 3, pp. 303-334, June 2008.

[18] W.B. Heinzelman, A.P. Chandrakasan and H. Balakrishnan, “An application specific protocol architecture for wireless microsensor networks,” *IEEE Transaction on Wireless Communications*, vol. 1, issue 4, pp. 660-670, December 2002.

Chanrit Danratchadakorn received the B.E. and M.S. degrees in information technology from King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand, in 2013 and 2014, respectively. His research interest includes wireless sensor networks.

Chotipat Pornavalai received the B.E. degree in electronic engineering from King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand, in 1992, and the M.S. and Ph.D. degrees in Information Sciences from Tohoku University, Japan, in 1995 and 1998, respectively. In 1999, he joined the Faculty of Information Technology, King Mongkut's Institute of Technology Ladkrabang, where he is currently an Associate Professor. His research interests include wireless sensor networks and delay-tolerant networks.