

A Decentralized Method for Maximizing k -coverage Lifetime in WSNs

Ryo Katsuma[†], Yoshihiro Murata[‡], Naoki Shibata^{*}, Keiichi Yasumoto^{**}, and Minoru Ito^{**}

[†]Osaka Prefecture University, Sakai, Osaka 599-8531, Japan, ryo-k@mi.s.osakafu-u.ac.jp

[‡]Hiroshima City University, Hiroshima 731-3194, Japan, yosihi-m@hiroshima-cu.ac.jp

^{*} Shiga University, Hikone, Shiga 522-8522, Japan, shibata@biwako.shiga-u.ac.jp

^{**} Nara Institute of Science and Technology

Ikoma, Nara 630-0192, Japan, {yasumoto,ito}@is.naist.jp

Abstract

In this paper, we propose a decentralized method for maximizing lifetime of data collection wireless sensor networks (WSNs) by making minimal number of nodes operate and putting other nodes in sleep. We divide a target field into multiple grids and make nodes in each grid locally achieve k -coverage. We can reduce energy consumption of WSN by minimizing the number of active nodes required for k -coverage. However, coverage degree is likely to go to excess beyond k near border between grids when deciding active nodes in each grid independently. To solve this problem, our method decides the minimal set of active nodes for adjacent grids at different times so that k -coverage of a grid is achieved taking into account the coverage in its neighboring grids. Through computer simulations, we confirmed that the proposed method achieved distribution of WSN processing with a small decrease of k -coverage lifetime compared to the centralized algorithm.

Keywords: wireless sensor network, distributed algorithm, k -coverage, lifetime maximization

1 Introduction

In recent years, wireless sensor networks (WSNs) consisting of many small sensor nodes have come up to realize applications such as environmental monitoring and object tracking. In a typical WSN, sensor nodes deployed over the target sensing field periodically senses environmental data and the data are collected at a sink through wireless multi hop communication among the nodes. An example application of such a WSN is temperature management in agricultural field where WSN is required to operate for a long time[1]. Many studies have been devoted to prolong WSN lifetime[2][3][4].

Some WSN applications like border guard require accuracy in sensed data as well as robustness of the system. Some studies [5][6][7] introduced the constraint called k -coverage of the target sensing field. k -coverage means that any point in the target field is covered by the sensing range of at least k sensors. There is a WSN deploying method [8] that scatters sensor nodes from the air for an environment where it is dangerous or hazardous for people to approach the target field. In this case, it is difficult to accurately deploy each sensor node in the intended position. Thus, to achieve k -coverage in such an environment, we need to scatter much more (surplus)

nodes than the case to deploy nodes in intended positions. We proposed a centralized method that prolongs k -coverage lifetime of a WSN with surplus nodes by activating minimal number of sensor nodes for k -coverage and letting the remaining nodes sleep[9]. Since it is a centralized method, it does not scale as the WSN size grows, due to computation overhead for deciding active and sleep nodes and communication overhead for notifying the computed sleep scheduling.

In this paper, we propose a decentralized method that prolongs k -coverage lifetime of a WSN with surplus nodes based on sleep scheduling. In the proposed method, we assume that much more sensor nodes than the minimal required for k -coverage are deployed over the target sensing field. We also assume that each node has three modes: sensing, relaying, and sleeping, and it can change its mode anytime. Each sensor node with sensing mode covers the circular region with a certain radius centered at its position. Each sensor node with other modes can send and receive data with other node. Each node with sleep mode can only change its mode to sensing or relaying, but it consumes much less power.

Our target problem is to decide a schedule of switching operating mode of each sensor node that maximizes the k -coverage time of the target field by the given WSN. This is a typical combinatorial optimization problem and for large size WSN, it is expected to be very hard to obtain the optimal solution in a short time. In the proposed method, we divide the target field into grid sub-fields (called *grids*, hereafter) so that the node with the highest battery amount in each grid becomes a leader and decides the schedule of all nodes in the grid. By the above process, we obtain the schedule of all nodes in the target field. Here, if we allow a leader of each grid to complete k -coverage of the grid independently of other grids, coverage degree near grid border is likely to be larger than k , uselessly. To cope with this problem, the proposed algorithm paints all the grids with black and white so that neighboring grids do not have the same color. Then, the algorithm lets white and black grids complete k -coverage alternately in this order so that the black grid nodes can efficiently complete k -coverage by considering the area already covered by the white grid nodes.

For evaluation of the proposed method, we conducted computer simulations and measured k -coverage lifetime. We compared the proposed method with the centralized method proposed in [9]. As a result, we confirmed that the proposed method achieved a certain level of WSN processing distri-

bution with about 14% decrease of k -coverage lifetime compared to the centralized method.

The following Section 2 outlines related work. We formulate our target problem in Section 3. Section 4 presents the proposed algorithm to compute operating mode of each sensor node and the data collection paths that maximize k -coverage lifetime in a distributed manner. We evaluate our method in Section 5. Finally, Section 6 concludes the paper.

2 Related Work

There are three types of WSN applications; (1) event detection type, (2) query sending type, and (3) data collection type. The event detection type WSNs let sensor nodes that detect events send their sensing data to the sink. The query sending type WSNs let sensor nodes that receive the query send their sensing data to the sink. The data collection type WSNs let all active sensor nodes send their sensing data to the sink. We target the data collection type WSNs. For example, a system called *field server* was proposed in [1]. Many illuminance and temperature sensors are deployed in agricultural sites, and the data sensed by these sensors are collected. Users can get illuminance and temperature data at each spot in the target area with field servers and improve the crop environment based on the obtained information. Field servers are expected to operate without maintenance for two years or more. Similarly to field server, most WSNs are expected to operate for a long time. However, since each sensor node used for a WSN has a limited amount of battery, WSN lifetime is also limited. Thus, many research efforts have been made to reduce power consumption in order to extend the lifetime of the whole WSN.

Heinzelman et al. proposed a method for reducing total data transmission amounts by merging the data received from multiple sensor nodes[10]. Since this approach may deteriorate sensing quality with respect to spacial and temporal density of sensing data, some applications that always need sufficient sensing quality may not accept such a quality deterioration. There are many studies using sleep mode for reducing energy consumption. The sleeping sensor nodes consume small power, but do not communicate with other nodes, and become active after specified time interval. In [2], Cao, et al. proposed a sleep scheduling method which lets nodes sleep when they need not communicate, in order to save the overall power consumption in WSN. Keshavarzian et al. proposed a method to minimize the number of active nodes and guarantee that the event information sensed by sensor nodes is delivered to the sink in a specified time[3]. Ma et al. proposed a distributed sleep scheduling algorithm based on TDMA for query sending type WSNs[4]. This method considers the energy consumption for changing sleeping mode to sensing mode and does not frequently change the mode of sensor nodes. However, these studies target event detection type WSNs or query sending type WSNs. Data collection type WSNs requiring k -coverage are not considered.

The sensor node deployment problem for efficiently covering the target sensing field is another big topic. Here, a given

geographical field is *k-covered* if any point in the field is included in the sensing ranges of at least k sensor nodes. In WSNs, information for the same place obtained by multiple sensor nodes is more accurate than that from only one sensor node.

Bai et al. discussed about the optimal locations of sensor nodes for k -covering the field[5]. However, it takes high costs to deploy a large number of nodes on the expected position accurately. There are also some studies to k -cover the field by mobile nodes. Wang et al. proposed a method to guarantee k -coverage [6] of the target field for an environment where mobile and static sensor nodes co-exist. In this method, sensor deployment must k -cover all points in the field. This method conserves the moving power of mobile sensor nodes and deploys those mobile nodes to guarantee the k -coverage of the whole field. Katsuma et al. proposed the method that prolongs k -coverage lifetime of a WSN using mobile nodes and static nodes[11]. In this method, mobile nodes move to cover a field for k -coverage, and data collection tree is constructed to distribute energy consumption among all nodes. However, mobile nodes are more expensive than static ones, and they cannot smoothly move on rough ground. Then, mobile nodes cannot always be applied to all WSN applications.

To overcome above difficulties, we proposed a sleep scheduling method that prolongs k -coverage lifetime using only static nodes[9]. In this method, minimal number of nodes are activated to satisfy k -coverage, and other nodes sleep. If battery of a node runs down, one or more sleep nodes are activated to complement it. However, it is a centralized method, and the sink calculates and sends results to all nodes. Then the computation time and the communication overhead significantly increase according to the number of nodes.

In this paper, we formulate the k -coverage lifetime maximization problem with surplus nodes, and propose a distributed sleep scheduling algorithm to solve it.

3 Problem of WSN Lifetime Maximization

In this section, we present the WSN model and formulate the problem of maximizing WSN lifetime while k -covering the target field by deciding schedules for operation modes of sensor nodes and constructing a data collection tree. This problem formulation is the same as in [9]. The notations used in this paper are summarized in Table 1.

3.1 Model, Assumptions, and Definitions

(1) Assumptions on Target WSN

We suppose a WSN in which many small battery-driven sensor nodes are deployed in a *target field*. Sensor nodes periodically sense such environmental information as temperature, humidity, sunlight, or moving object, and send it by multi-hop communication to a base station called *sink*. We denote the set of points in the target field, a sink, and sensing frequency as *Field*, *Bs*, and *I*, respectively. We denote the set of sensor nodes by $S = \{s_1, \dots, s_l\}$.

Each sensor node has three operation modes: *sensing*, *relaying*, and *sleeping*. A node whose operation mode is sensing, relaying, or sleeping is called *sensing node*, *relaying node*, or *sleeping node*. We denote the sets of sensing, relaying, sleeping nodes by $U = \{u_1, u_2, \dots\}$, $V = \{v_1, v_2, \dots\}$, $W = \{w_1, w_2, \dots\}$, respectively, where $U \cup V \cup W = S$. Each sensing/relaying node can change its mode instantly. Each sleeping node can change to another mode when it wakes up after a specified sleeping time elapses.

Table 1: Notations

Notation	Meaning
$Field$	the target sensing field
Bs and $Bs.pos$	sink and its position
I	sensing frequency in Hz
D	size of data obtained by node
r_s	sensing radius of each sensor
r_c	maximum communicable distance of each sensor node
n	power attenuation co-efficient for antenna
$Trans(x, d)$	power required for a sensor node to transmit x bits for d meters
$Recep(x)$	power required for a sensor node to receipt x bits
$Sens()$	power required for a sensor node to sense
$Listen(y)$	power required for a sensor node to listen for y seconds
$Sleep(y)$	power required for a sensor node to sleep for y seconds
$C(s)$	energy consumption of sensor node s per seconds
S	set of sensor nodes
U	set of sensing nodes
V	set of relaying nodes
W	set of sleeping nodes
$s.pos, s.energy[t], s.range, s.desc,$ and $s.send$	position, remaining battery amount at time t , sensing range, number of descendant nodes, and next hop node of a sensor node s
k	number of sensors to cover each point in target field.

Each sensing node covers a disk with radius r_s centered at the node. We denote the covered range of sensing node $s \in U$ by $s.range$. Each sensing node obtains data by sensing. We assume that the data size is fixed and the data are sent to the sink without compression or aggregation along a multi-hop path (consisting of only sensing and relaying nodes) to the sink. We denote the data size by D .

Each sensor node has a wireless communication capability and its radio transmission range is a disk with a certain radius centered on it. Maximum communicable distance of each

sensor node is denoted by r_c . Each sensing/relaying node can change its transmission power so that the radio transmission radius can be adjusted depending on distance to a specific peer node ¹. Since there is little influence on radio interference when sensing frequency I is small enough, we assume that packet collision between nodes is negligible. A transmitted packet is always successfully received if the recipient node (sensing/relaying node) is within the radio transmission range, and always fails if outside of the range. We assume that each node uses only one-hop unicast communication by designating a recipient node.

(2) Assumptions for Power Consumption

Each sensor node s has a battery, where the initial energy amount is denoted by e_{init} and the remaining energy amount at time t is denoted by $s.energy[t]$. Each node consumes energy for data transmission, data reception, and sensing data, and even during idle time and sleeping time.

Powers $Trans(x, d)$ and $Recep(x)$ required to transmit x [bit] for d [m] and receive x [bit] conform to Formulas (1) and (2), respectively[10].

$$Trans(x, d) = E_{elec} \times x + \epsilon_{amp} \times x \times d^n \quad (1)$$

$$Recep(x) = E_{elec} \times x \quad (2)$$

Here, E_{elec} and ϵ_{amp} are constants representing the power required by information processing and the power for amplification, respectively.

Powers $Sens()$, $Listen(y)$, and $Sleep(y)$ required to sense D [bit] data, listen to whether radio messages come or not for y [sec.], and sleep for y [sec.] conform to the following formulas (3), (4), and (5), respectively.

$$Sens() = E_{elec} \times D + E_{sens} \quad (3)$$

$$Listen(y) = E_{listen} \times y \quad (4)$$

$$Sleep(y) = E_{sleep} \times y \quad (5)$$

Here, E_{sens} , E_{listen} , and E_{sleep} are constants representing the powers required for sensing data, listening for 1 second, and sleeping for 1 second, respectively.

3.2 Problem Definition

If a particular set of sensing nodes are used for a long time, their batteries will be quickly exhausted. Then, it is necessary to dynamically change the set of sensing nodes. So, we formulate a problem to derive schedules of when and to which mode each sensor node should change at each time during WSN operation time. Fig. 1 shows an example of schedules.

Let t_0 and t_{end} denote the initial WSN deployment time and the earliest time when the k -coverage of the WSN is no

¹IRIS mote [12], for example, is capable of changing its transmission power from -17.2[dBm] to 3[dBm].

longer maintained due to battery exhaustion of some nodes. For each $s \in S$ and each $t \in [t_0, t_{end}]$, let $Mode(s, t)$ denote the operation mode of s at time t . Then, for each $s \in S$, we denote a schedule $modeschedule$ to switch the operation mode of s during time interval $[t_0, t_{end}]$ by the following formula.

$$modeschedule(s, [t_{start}, t_{end}]) = \bigcup_{t \in [t_{start}, t_{end}]} \{Mode(s, t)\} \quad (6)$$

A recipient node to which a sensor node s sends a data at time t is denoted by $Send(s, t)$. We denote a schedule $sendschedule$ to switch the destination node of s during time interval $[t_0, t_{end}]$ by the following formula.

$$sendschedule(s, [t_{start}, t_{end}]) = \bigcup_{t \in [t_{start}, t_{end}]} \{Send(s, t)\} \quad (7)$$

Given the information on the target field $Field$, $s.pos$, $s.energy$, and $s.range$ for each sensor node $s \in S$, the position of a sink $Bs.pos$, and constants E_{elec} , E_{sens} , E_{listen} , E_{sleep} , ϵ_{amp} , n , D , and I , our target problem for maximizing the WSN lifetime denoted by t_{life} is to decide the schedule $schedule(s, [t_0, t_{end}])$ for each node $s \in S$ that satisfies Condition (8).

$$\forall t \in [t_{start}, t_{end}], \forall pos \in Field, |Cover(pos, t)| \geq k. \quad (8)$$

Here,

$$Cover(pos, t) \stackrel{def}{=} \{s | s \in S \wedge pos \in s.range \wedge Mode(s, t) = sensing \wedge s.energy[t] > 0\}. \quad (9)$$

Condition (8) guarantees k -coverage of the target field. In general, k -coverage can be achieved by a part of all sensor nodes ($U \subseteq S$) whose remaining energy amounts are not exhausted.

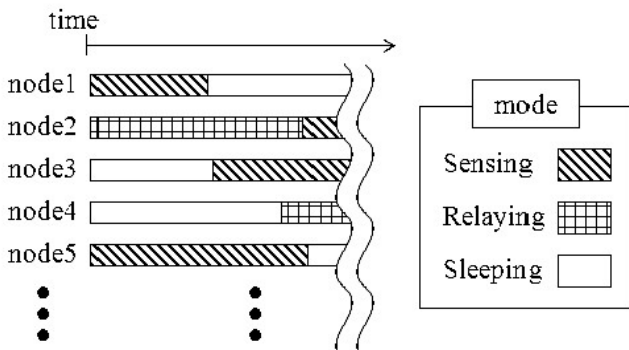


Figure 1: Schedule for Switching Operation Mode

We define the WSN lifetime t_{life} as the time from initial deployment to the time when the Condition (8) is no longer

satisfied for any set of sensing nodes. Then, we define the following objective function (10):

$$\text{maximize } (t_{life}) \text{ subject to (8)} \quad (10)$$

4 Decentralized Algorithm for Maximizing k -coverage Lifetime

4.1 Outline

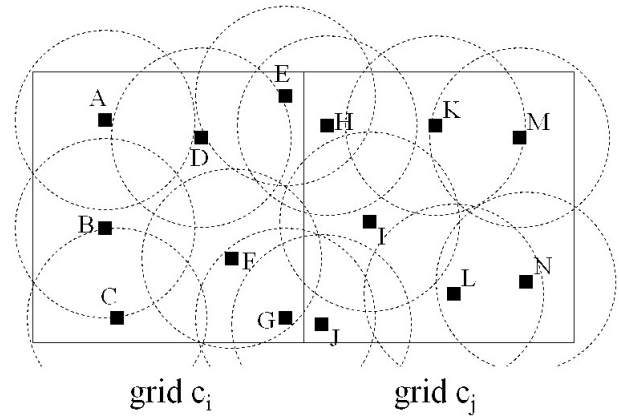


Figure 2: Example of excessive coverage

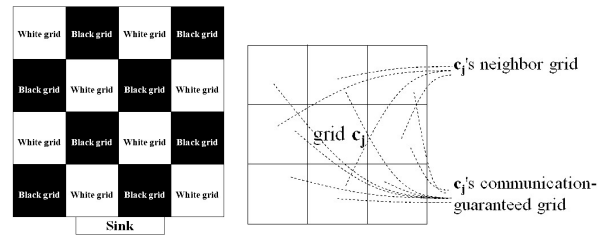


Figure 3: Dividing White and Black Grid Figure 4: Neighbor Grid and Communication-Guaranteed Grid

The problem explained in Section 3.2 is a combinatorial optimization problem, and it is difficult to find an algorithm to quickly derive the optimal solution. In [9], we proposed a centralized heuristic algorithm to derive a semi-optimal solution. In this paper, we propose a decentralized method for this problem.

In the proposed method, the field is divided into multiple sub-regions called *grids*. By k -covering each grid, the entire field is k -covered. However, if we determine the minimal set of sensing nodes for k -covering each grid independently, areas close to grid boundaries may be excessively covered by nodes in two neighboring grids. For example, Fig. 2 shows a case that either node E or H has to be activated for 1-coverage of the entire field. Nodes G and J are also such nodes that only one of them has to be activated at a time. However, if we do not synchronize node selection processes in two grids c_i

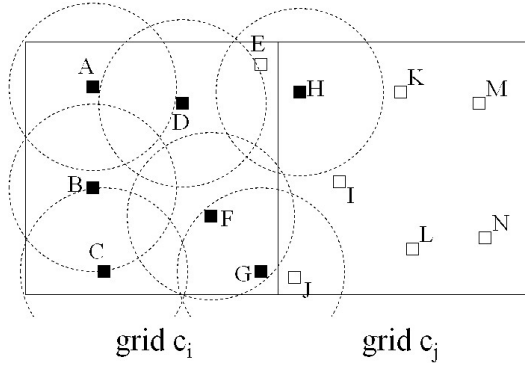


Figure 5: A white grid c_i is 1-covered

and c_j in Fig. 2 and coverage is computed independently, it is difficult to uniquely determine which nodes to activate, and it is possible that both nodes E and H are activated. Therefore, in the proposed method, we try to solve this problem by making neighboring grids to compute at different times. In order to complete the entire computation in two steps, we classify all grids into two groups: white and black, like a checkerboard as shown in Fig. 3. In the algorithm described later, after a set of active nodes for k -coverage is determined in each white grid, another set is determined in each black grid, considering the coverage of the white grids. The nodes to be activated are computed periodically. When the computation starts, the node with the largest remaining battery energy in each grid becomes the leader of the grid, and it determines a set of nodes to be activated in the grid.

Hereafter, we describe the initial configuration during node deployment, and the algorithm executed after deployment. In the initial configuration, division of the whole field into grids and classification of the grids into white and black groups are determined. In the proposed algorithm, a schedule of operating modes and routes for collecting data are determined.

4.2 Initial configuration during deployment

We first describe the initial configuration.

First, the entire field is divided into grids. The grids are denoted by c_1, c_2, \dots, c_m . The set of all nodes in grid c_j is called *belonging set* of c_j . The set of nodes that are subject to the computation of k -coverage in grid c_j consists of all nodes whose sensing range intersects c_j , and this set of nodes is called *set of nodes subject to computation* of c_j . In order to make the distance between any node in a grid and any node in its 8 neighboring grids less than the radio range, the shape of each grid is set to a square whose sides are shorter than $\frac{r_c}{2\sqrt{2}}$. As Fig. 4 shows, the surrounding 8 grids of c_j are called *communication-guaranteed grids* of c_j .

We paint the grids into black and white so that any two adjoining grids have different colors (Fig. 3). By the algorithm described later, computation is performed in all white grids

²Here, we assume that the radio range r_c is sufficiently larger than sensing range r_s .

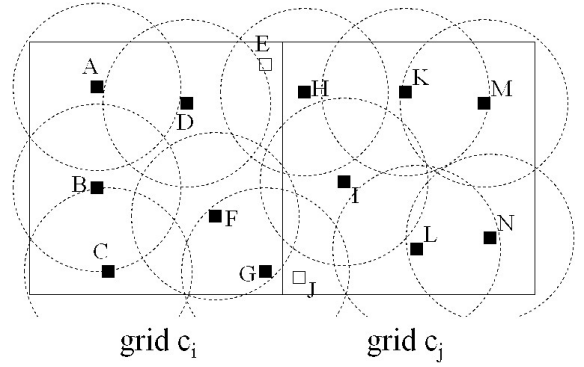


Figure 6: A black grid c_j is 1-covered taking into account the coverage of white grid c_i

at the same time in step 1. In step 2, after black grids confirm that computation is finished in neighboring white grids, computation in black grids is performed based on coverage information given as a result of computation in white grids.

Before the proposed algorithm is executed, all nodes subject to computation of the same grid communicate with each other, and obtain the residual energy, positions and unique IDs of all the other nodes. The sink sends the information of the sets of nodes belonging to and subject to each grid, respectively, by flooding.

4.3 Proposed Algorithm

In this subsection, we first describe an overview of the entire algorithm, and then details of the algorithm executed in each grid to k -cover it.

In the proposed algorithm, a leader of each grid is determined every period T called *duty cycle*, and the leader computes the schedule of operating modes for the all nodes. The algorithm is executed after the nodes are first deployed and configured.

We need to set the time to wake up for the nodes that are set to sleep. In order to allow those nodes to receive the computation result from the leader, they are woken up every period T .

Below, we describe the algorithm executed on each grid.

1. Each node collects battery information for nodes subject to the same computation by communicating with each other.
2. A leader node is selected at each grid (explained later).
3. Each leader node notifies the information of itself to the members of belonging sets of all the neighboring grids.
4. In each white grid, the leader node determines the set of sensing nodes that k -covers the grid (explained later).
5. If there is a grid that cannot be k -covered, then finish the operation of the WSN.
6. In each white grid, routes for collecting data from all sensing nodes to the leader node are constructed

7. The leader node of each white grid sends coverage information to leader nodes of neighboring black grids
8. In each black grid, the leader determines sensing nodes to k -cover the grid after the leader receives coverage information from all neighboring white grids
9. If there is a grid that cannot be k -covered, then finish the operation of the WSN.
10. In each black grid, routes for collecting data from all sensing nodes to the leader node are constructed
11. Leader nodes of all grids send operation mode change notice to the nodes subject to the same computation, and the nodes receiving the notice change their operation mode.
12. Operate the WSN for T , and then go to step 1.

Below, we describe the details of each part of the algorithm and how to collect data when the WSN is operated.

Selecting leader node

All nodes in the belonging set of each grid communicate with each other, and they get to know the highest remaining battery energy of other nodes. Then, the node with the highest remaining battery is selected as the leader node for each grid. If there are more than one node that have the highest remaining battery, the node with the lowest ID is selected. Here, the leader node can be uniquely determined, and thus it does not need to send an acknowledgement.

Deciding sensing nodes

In order to extend the k -coverage lifetime, we save the battery energy by making the number of nodes to k -cover as small as possible, and making other nodes relay communication or sleep. The leader node of each grid determines the set of sensing nodes for efficiently k -covering the grid by the *sequential activation method* that we proposed in [9]. The sequential activation method is a greedy method that incrementally adds a node with the largest contribution area for k -coverage into the set of sensing nodes until k -coverage is achieved.

By using figures, we now explain how the proposed method eliminates excessive coverage when the entire field has to be 1-covered. In Fig. 2, we suppose that c_i is a white grid and c_j is a black grid. In the proposed method, the white grid c_i first determines the sensing nodes to 1-cover c_i using the sequential activation method. Here, suppose that nodes A, B, C, D, F, G and H become sensing nodes for c_i as shown in Fig. 5. The leader node of c_i sends the coverage information that consists of sensing nodes whose sensing regions intersect c_j to the leader of black grid c_j . Here, the information is that nodes H, F and G are selected as sensing nodes. Then, the set of sensing nodes that 1-covers c_j is determined. Here the set of sensing nodes that covers c_j except the area covered by nodes H, F and G is determined. By the sequential activation

method, nodes L, M, K, N and I are added to c_j in the order of how big area it contributes to the 1-cover the c_j as shown in Fig. 6. As shown above, we can avoid the case that both nodes E and H or both nodes G and J become sensing nodes.

Collecting data in each grid

Here, we describe how to determine routes to collect sensed data to the leader node. Each sensing node regards the leader node as a sink, and they send data using multi-hop communication along these routes for energy saving. We describe the way to collect data to the sink later.

By Equation (1), the communication load depends on the amount of data and the distance. WSN for data collection relays towards a sink, data received from its ancestor nodes in addition to data sensed by itself, and this shortens the battery life. On the other hand, if we send data to the sink by a single-hop communication, the communication distance from the nodes distant from the sink becomes large, and it consumes large amount of energy. Thus, we use the Balanced edge selection method [11] in order to equalize energy consumed for transmission among all nodes. Then, in order to shorten the transmission distance for each link in the tree, we modify the data collection tree by Relay selection method[11]. This method is to find the node that has the largest communication load according to Equation (1) and use a sleep node to relay the communication for this link.

Collecting data to sink

Here, we describe how to collect data from leader nodes to the sink. Balanced edge selection method can be used only if all positions of nodes and remaining battery energy are known. Thus, we propose a new data collection method that can be used with local information.

In this method, when the WSN is operated and each leader node retains the data to be sent to the sink, this leader node communicates with leader nodes that are closer to the sink than itself, and they get the information of residual energy for those nodes. By sending the data to the node with the largest residual energy, the energy consumption can be equalized while the data are collected to the sink.

The algorithm executed by each leader node is shown below.

1. Get the positions of leader nodes that are in the communication-guaranteed range.
2. Make all leader nodes that are in the communication-guaranteed range and closer to the sink be *transmission-destination candidate nodes*.
3. Receive sensed data from all nodes in the responsible grid, or wait until it receives sensed data from a leader node of another grid.
4. Communicate with all transmission-destination candidate nodes and get the residual energy information.

Table 2: Simulation Configurations

Parameter	Value
Initial energy amount of each node	$s.energy = 32400J$ (by referring to [10])
Power consumption coefficient for data processing	$E_{elec} = 50 \text{ nJ/bit}$ (by referring to [13])
Power consumption coefficient for signal amplification	$\epsilon_{amp} = 100\text{pJ/bit/m}^2$ (by referring to [13])
Power consumption coefficient for sensing	$E_{sens} = 0.018J$ (by referring to [14])
Power consumption coefficient for idle time	$E_{listen} = 0.043J/s$ (by referring to [14])
Power consumption coefficient for sleep time	$E_{sleep} = 0.000054J/s$ (by referring to [14])
Area of sensing disk of each sensor	$r_s = 20m^2$ (by referring to [15])
Size of data for sensed information	$D = 128\text{bit}$ (by referring to [16])
Sensing frequency	0.1Hz (by referring to [16])

5. Send the retaining data to the transmission-destination candidate node with the largest residual energy (the sink is assumed to have unlimited power supply).
6. Go to step 3.

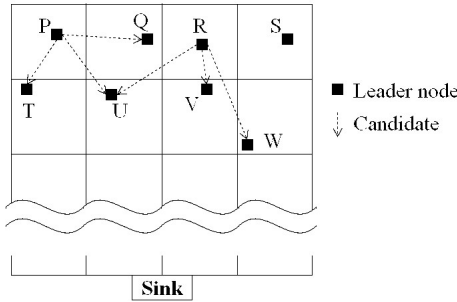


Figure 7: Example of Routing Between Grids

The example of this algorithm is shown in Fig. 7. The transmission-destination candidate nodes of node P are decided to Q, T, and U by Step 2. The node with the largest residual energy of these three nodes is selected to the recipient node of P by Step 5. The transmission-destination candidate nodes of node R are also decided to U, V, and W by Step 2. Similarly, R sends the data to the node with the largest residual energy.

5 Experimental Validation

We have conducted computer simulations for measuring the k -coverage lifetime by the proposed method, and compared the k -coverage lifetime with the centralized method [9].

As a common configuration among the experiments, we used the parameter values shown in Table 2. For all the experiments, we used a WSN simulator which we implemented in Java and executed the simulator on a PC with Intel Corei7-2600 (3.40GHz), 8GB memory, Windows7 Home Premium, and Sun Java Runtime Environment 6.0.240.7.

Centralized Method periodically decides a set of sensing nodes that k -covers the entire field by the sequential activation method and constructs a data collection tree connecting all the sensing nodes by Balanced edge selection method.

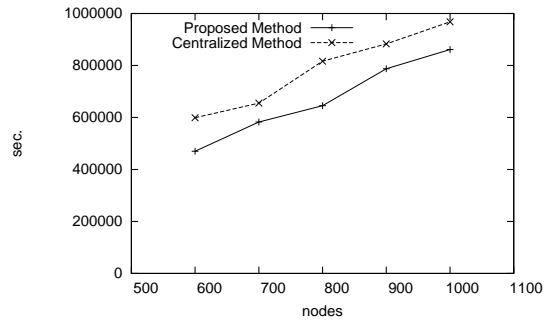


Figure 8: 1-Coverage Lifetime

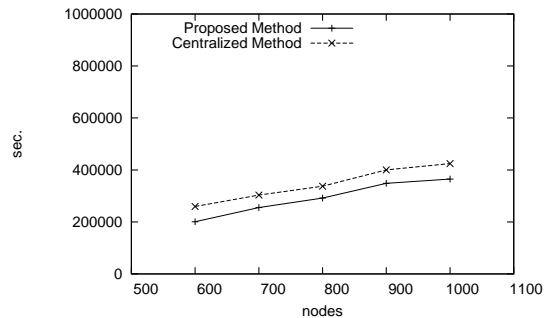


Figure 9: 3-Coverage Lifetime

The configuration of this experiment other than Table 2 is provided as follows.

- Field size: 50m × 50m
- Grid size: 10m × 10m
- Position of the sink: around the south (bottom) end in the field
- Number of sensor nodes: 600, 700, 800, 900, and 1000
- Required coverage: $k=1$ and 3

Note that the size of the target field and grid size should be appropriately decided so that each grid can be sufficiently k -covered for a given number of nodes and coverage degree k . Thus, we used field size 50m × 50m and grid size 10m ×

10m, that is, when 600 sensing nodes are randomly deployed in the target field, there will be extremely surplus nodes for $k=1, 2,$ and 3 . In the experiment, the initial positions of nodes are given in the target field by uniform random values.

We show experimental results obtained through computer simulations in Fig. 8 for 1-coverage and Fig. 9 for 3-coverage. These results are average of 30 runs.

Both Figs. 8 and 9 show that k -coverage lifetime by Proposed Method was only about 14% less than Centralized Method. The reason of reduction is that the union of optimal solutions for grids is not always the optimal solution for the whole field. Therefore, Proposed Method needed more sensing nodes for k -covering the field than Centralized Method. Proposed Method made less nodes keep their battery by sleeping until sensing nodes exhaust their battery than Centralized Method. However, we confirmed that our proposed algorithm takes reasonably short calculation time. In these experiments, maximum calculation time of the proposed algorithm for k -covering each grid was always less than 0.1 second. On the other hand, Centralized Method took 1.2 seconds when the number of nodes is 1000.

The above results support that our proposed method achieves a certain level of processing distribution with a reasonably small fraction of WSN lifetime reduction.

6 Conclusion

In this paper, we target a problem that maximizes the k -coverage lifetime of the data gathering WSN using surplus static sensor nodes with sleeping mode. We proposed a decentralized method to solve this problem by dividing the field into grids and decide the minimal number of sensing active nodes for k -covering each grid.

As a simulation result, we confirmed that our method achieved 1- and 3-coverage of the target field with only 14% lifetime reduction compared to the centralized method.

As part of future work, we will consider the more effective method to construct a data collection tree. Through the preliminary experiment, the number of activated (sensing) nodes are almost identical in both the centralized algorithm and the proposed algorithm. So, we think that the reason of k -coverage lifetime reduction is due to the inefficiency of the data collection tree. The reduction of k -coverage lifetime will be mitigated by improving the data collection tree.

REFERENCES

- [1] Fukatsu, T. and Hirafuji, M.: "Long-Term Monitoring System using Field Monitoring Servers," *Proc. of AFITA/WCCA Joint Congress on It In Agriculture*, pp. 685–691, 2004.
- [2] Cao, Q., Abdelzaher, T., He, T., Stankovic, J.: "Towards Optimal Sleep Scheduling in Sensor Networks for Rare-Event Detection," *Proc. of Information Processing in Sensor Networks (IPSN2005)*, pp. 20–27, 2005.
- [3] Keshavarzian, A., Lee, H., and Venkatraman, L.: "Wake-up scheduling in wireless sensor networks," *Proc. of 7th ACM Int'l Symp. on Mobile ad hoc networking and computing (MobiHoc2006)*, 2006.
- [4] Ma, J., Lou, W., Wu, Y., Li, X. Y., and Chen, G.: "Energy Efficient TDMA Sleep Scheduling in Wireless Sensor Networks," *Proc. of 28th IEEE Int'l Conf. on Computer Communications (Infocom2009)*, 2009.
- [5] Bai, X., Yun, Z., Xuan, D., and Zhao, W.: "Optimal Multiple-Coverage of Sensor Networks," *Proc. of Int'l Conf. on Computer Communications (INFOCOM2011)*, 2011.
- [6] Wang, W., Srinivasan, V., and Chua, K. C.: "Trade-offs Between Mobility and Density for Coverage in Wireless Sensor Networks," *Proc. of 13th Int'l. Conf. on Mobile Computing and Networking (MobiCom 2007)*, pp. 39–50, 2007.
- [7] Abrams, Z., Goel, A. and Plotkin, S.: "Set K -Cover Algorithms for Energy Efficient Monitoring in Wireless Sensor Networks," *Proc. of 3rd Int'l. Symp. on Information Processing in Sensor Networks*, pp. 424–432, 2004.
- [8] Frank Y. S. Lin and P. L. Chiu: "Energy-Efficient Sensor Network Design Subject to Complete Coverage and Discrimination Constraints," *Proc. of 2nd Int'l. Conf. on Sensor and Ad Hoc Communications and Networks (SECON 2005)*, 2005.
- [9] Katsuma, R., Murata, Y., Shibata, N., Yasumoto, K., and Ito, M.: "Extending k -Coverage Lifetime of Wireless Sensor Networks with Surplus Nodes," *Proc. of the 5th Int'l. Conf. on Mobile Computing and Ubiquitous Networking (ICMU 2010)*, pp. 9–16, 2010.
- [10] Heinzelman, W.R., Chandrakasan, A., and Balakrishnan, H.: "Energy-Efficient Communication Protocol for Wireless Microsensor Networks," *Proc. of the 33rd Hawaii Int'l. Conf. on System Sciences (HICSS 2000)*, 2000.
- [11] Katsuma, R., Murata, Y., Shibata, N., Yasumoto, K., and Ito, M.: "Extending k -Coverage Lifetime of Wireless Sensor Networks Using Mobile Sensor Nodes," *Proc. of 5th Int'l. Conf. on Wireless and Mobile Computing, Networking and Communications (WiMob'2009)*, 2009.
- [12] Crossbow Technology, Inc. 2008: "IRIS mote," <http://www.xbow.jp/mprmb.pdf>.
- [13] Wang, G., Cao, G., La Porta, T., and Zhang, W.: "Sensor Relocation in Mobile Sensor Networks," *Proc. of Infocom 2005*, Vol. 4, pp. 2302–2312, 2005.
- [14] Crossbow Technology, Inc. 2003: "MICA2: Wireless Measurement System," http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/MICA.pdf.
- [15] Ganeriwal, S., Kansal, A., and Srivastava, M. B.: "Self aware actuation for fault repair in sensor networks," *Proc. of Int'l Conf. on Robotics and Automation (ICRA2004)*, 2004.
- [16] Kamimura, J., Wakamiya, N., and Murata, M.: "Energy-Efficient Clustering Method for Data Gathering in Sensor Networks," *Proc. of First Workshop on Broadband Advanced Sensor Networks (BaseNets2004)* (2004).