ROAL: A Randomly Ordered Activation and Layering Protocol for Ensuring K-Coverage in Wireless Sensor Networks

Hogil Kim, Eun Jung Kim
Department of Computer Science, Texas A&M University, College Station, TX 77843-3112, USA
Email: {hogil, ejkim}@cs.tamu.edu
Ki Hwan Yum
Department of Computer Science
University of Texas at San Antonio
San Antonio, TX 78249, USA
yum@cs.utsa.edu

***Abstract*— In this paper, we propose a Randomly Ordered Activation and Layering (ROAL) protocol. Each node under the ROAL protocol can decide its eligibility regarding a given coverage degree K at randomly generated activation time using only the coverage status informed from its neighbor nodes located within its sensing region. A new concept of layer coverage also provides a simple and effective reconfiguration method for energy balancing. Using the layer concept, we also propose a circulation scheme to reconfigure the set of working nodes in an autonomous way, where the reconfiguration can be performed with a small and almost constant energy consumption. We also provide the model of the expected coverage and connectivity for the layer coverage and show a proper range in which only one node can be activated with regard to a node density and the sensing radius of a node.**

The simulation results show that the ROAL protocol can guarantee K-coverage with more than 95% coverage ratio, which almost closes to the coverage ratio that is achieved using the geographic coordinate. A significantly extended network lifetime is also observed against the original topology of a given network. Meanwhile, the experimental results on the circulation scheme show that the fraction of total reconfiguration energy becomes less than 1% of the energy consumed for the reconfiguration. Also, we obtain a greatly reduced packet latency, which corresponds to only 5% of the delay that occurred in the ROAL protocol.

***Index Terms*— wireless sensor networks, K-coverage, layer coverage, circulation scheme**

I. INTRODUCTION

A wireless sensor network (WSN) is composed of many low-cost but small sensor nodes, each of which is equipped with a radio and a processor module. Since the sensor nodes are spread randomly over a vast region for the purpose of detecting or monitoring some special events, many studies on the sensor networks have focused on how to effectively cover the region while conserving low energy budget. Coverage problem has been studied as a solution to achieve both energy efficiency and quality of detecting by maintaining a sufficient coverage degree that is enough for a certain application using minimal active nodes.

Previous coverage algorithms can be classified into two different categories, deterministic and probabilistic algorithms. The algorithms in the first category [1]–[3] aim to monitor as many points as possible with K different active nodes persistently. The other approaches [4], [5] provide the required K-coverage based on the frequency of detection of an event, or monitoring each point during a predefined time interval. The common goal of both the approaches is to maintain high detection probability with maximum energy conservation. However, there are several limitations with the approaches for the following reasons. First, algorithms with the deterministic approach usually take a long time to decide the degree of coverage because they need to calculate the physical distance between neighbors. They incur significant overhead due to exchange of a large number of messages for distance information, which increases with the number of nodes. On the other hand, the algorithms with the probabilistic approach suffer from the limitation of sensing ability. Due to their periodic behavior, some events that occur between the actual sensing activities would not be detected promptly. Furthermore, some algorithms [1], [3], [4] require the geographic information of each node to determine its relative location. This geographic information can be obtained using a global positioning system (GPS) or other location retrieval algorithms. However, the cost of a GPS and the running time of such an algorithm prohibit those approaches from being used widely.

In this paper, we propose Randomly Ordered Activation and Layering (ROAL) protocol to solve the K-coverage problem without using the GPS information while minimizing critical problems that reside in those approaches. The main idea is to construct K layers by selecting K disjoint subsets from the original set of sensor nodes such that each layer can provide 1-coverage and these K layers can provide K-coverage together. The ROAL protocol selects sensor nodes in a distributed and stochastic way for each layer within a given time interval and the connectivity between neighboring nodes is maintained if the density of the sensor network is high enough. We also propose a circulation scheme that can improve the performance of fault-tolerance and energy-balance for K-coverage based on our layer coverage. With our proposed circulation method, we recursively activate a different set of nodes for the fault-tolerance and the energy-balance among all deployed nodes without the repetitive layering procedure of the ROAL protocol. Also, we present the analytical model of the coverage and the connectivity of our layer coverage to verify the effect of the unit size of layering radius to the resultant coverage and connectivity.

Simulation results show that the ROAL protocol can provide K-coverage if the node density is higher than or equal to 0.025 (250 nodes/10,000 m^2) while providing a good approximation on the required K-coverage. The ROAL protocol increases the network lifetime more than four times while achieving almost no data packet loss. The experimental results on the circulation show that the reconfiguration can operate with only less than 1% of of the energy consumed in the original ROAL protocol. Also, we obtain a greatly reduced packet latency, which corresponds to only 5% of the delay that occurred in the ROAL protocol.

The rest of the paper is organized as follows. In Section II, we discuss the related work and we provide the details of the ROAL protocol in Section III. We propose a circulation-based reconfiguration scheme in Section **??**. In Sections **??** and VI, we analyze the coverage and connectivity of the proposed algorithms, and present simulation results. We conclude our work in Section VII.

II. Background and Related Work

The K-coverage problem is the study on the decision for selecting a set of working nodes such that, with K-covered sensor network, any point in an interesting area is monitored by at least K different sensor nodes. The final goal of the K-coverage is, hence, to prolong the network lifetime using a limited energy budget of sensor nodes without losing the sensing quality.

Previous K-coverage algorithms can be classified into two different categories, deterministic and probabilistic algorithms. The algorithms in the first category [1]–[3] aim to monitor as many points as possible with K different active nodes simultaneously. The other approaches [4]–[7] provide the required K-coverage based on the expected number of observations for each point, moving target during a given time interval, or the whole duration of movements.

As a highly related approach, we show three studies here. Ye et al. [2] proposed the probing environment and adaptive sleeping (PEAS) protocol that can cover and connect a sensor network by activating only one node within a probing radius of a node. They provided an heuristic way to provide a certain degree of coverage, decided by the number of distributed sensors. An integrated analytical model for multi-coverage and connectivity was suggested by Xing et al. [3], where a sensor network is K-covered if and only if all the points within the intersection area formed by all neighboring nodes are covered by K nodes. The main problem with this approach is the time complexity of $O(N^3)$, where N is the number of neighboring nodes. On the other hand, Set K-Cover problem [5] uses a similar concept as our layering algorithm. However, in these studies, the focus is to make K subsets using all deployed nodes such that each subset covers all area or can take a K-coverage effect by the iterative activation of each subset in a round-robin fashion. In this scheme, each node belongs to one subset, and then each subset is activated one by one. To select nodes efficiently in terms of accuracy, they also use the geographic information. The ROAL protocol suggested here selects only K subsets and the purpose is to guarantee 1-coverage for each layer without using the geographic information.

We apply the probing scheme proposed in PEAS [2] and sentry selection protocol [8], where each node sends a hello message to check out any other active nodes within its sensing area. If there is a reply from an active node, the probing node will sleep until the next probing time arrives. We enhance this approach to validate if any K different nodes are working within the sensing range using one message per each node. The running time of our algorithm is bounded by a small constant value of t, a given interval of a phase. The main idea is to make K layers such that each layer is composed of a set of nodes to provide 1-coverage and the K different layers provide K-coverage together.

III. The ROAL Protocol

A. Basic Idea

The basic idea is to build K logical layers[1] for requested K-coverage, where each layer consists of a disjoint set of working nodes that provide 1-coverage for the whole target sensing region, as shown in Fig. 1. In addition, we assume that the nodes remain in their original position in this work. From the set S of all sensor

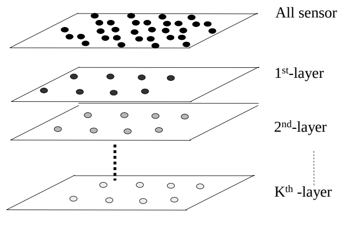


Figure 1. K-Layer Coverage

nodes, we select only a small number of nodes to form 1-coverage and repeat this process K times to form K-coverage. A set S_i, which is i^{th} subset (or layer) of S, is composed of selected nodes, and $S_i \cap S_j = \phi$, if $i \neq j$ and $1 \leq i, j \leq K$. Also, $\cup_{i=1}^{K} S_i \subseteq S$ and $\sum_{i=1}^{K} |S_i| \leq |S|$.

All these selected nodes remain working to provide K-coverage for a predetermined period, while the other

[1]A layer represents a virtual plane that includes a subset of working nodes.

nodes go to sleep to save their residual energy. After the period, this process can be repeated to evenly distribute the energy consumption among the sensor nodes in the WSN.

Using this idea, we can easily change the degree of coverage during the network running time if a user wants to increase the degree of coverage for more accurate data or to reduce the degree for energy conservation. Unlike all the previous studies that did not consider dynamic real-time reconfiguration on the degree of coverage seriously, our approach can easily cope with such demands.

B. Randomly Ordered Activation (ROA) Algorithm

Randomly Ordered Activation is a stochastic and greedy algorithm that selects K sets of working nodes for K-coverage at a randomly generated activation time. Before its activation time expires, a node running the ROA algorithm maintains a list of layer numbers (LIDs) sent by its neighbor nodes within its sensing circle area. The eligibility as a working node is decided when the activation time expires.

Pseudo Code of ROA Algorithm

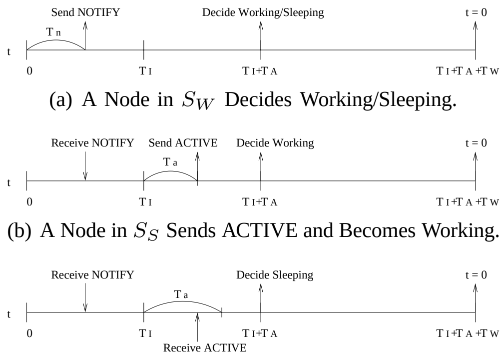
Algorithm ROA(K, T_A)

$t \leftarrow 0$
$LID \leftarrow 0$
$H \leftarrow \emptyset$
$T_a \leftarrow rand(0, T_A)$
while $t < T_A$
    if ACTIVE message arrives from neighbor node
        $H(ACTIVE.LID) \leftarrow true$
    if $t = T_a$
        $i \leftarrow 1$
        while $i \leq K$
            if $H(i) = false$
                $LID \leftarrow i$
                send ACTIVE.LID
                i++
if $LID = 0$
    sleep
else
    set active

A field of boolean array H indexed by the LID that is carried in the ACTIVE message of a neighbor node is set to true. A node will work if it finds its LID less than K or it will go to a sleep mode otherwise. The ROA algorithm, therefore, can run in $O(K)$ time with $O(K)$ number of message exchanges at each node.

C. Detailed ROAL Protocol

In this section, we complete the design of the ROAL protocol that can maintain the K-coverage in a round-robin fashion for the purpose of energy balancing among all distributed nodes. Each round consists of three phases: Initialization Phase (IP), Activation Phase (AP), and Working Phase (WP). The duration of each phase is determined by the condition of the network such as the density of sensor nodes or the tasks of the applications. For simplicity, let three parameters, T_I, T_A, and T_W, be the durations of the IP, the AP, and the WP, respectively. In addition, let T_a and T_n be randomly generated activation and notification times, respectively, and they are used to avoid collisions in the wireless channel. Note that $0 < T_n < T_I$ since T_n is used during the IP, and $0 < T_a < T_A$ since T_a is used during the AP.



(a) A Node in S_W Decides Working/Sleeping.

(b) A Node in S_S Sends ACTIVE and Becomes Working.

(c) A Node in S_S Receives ACTIVE and Becomes Sleeping.

Figure 2. Three Possible Scenarios in the ROAL Protocol

1) Initialization Phase (IP): Each round starts with setting the local timer to 0, and then the IP begins. At the beginning of the IP, all sleeping sensor nodes wake up and participate in the decision (for working or sleeping) process with the working nodes in the previous round. Let S_W and S_S be the sets of working nodes and sleeping nodes in the previous round, respectively. Also, R_n indicates the n^{th} round. Then there are two cases depending on the round number.

Case 1: The first round (R_1)

When sensor nodes are initially deployed over an area, all K layers should be constructed. In this case, all nodes generate the activation time T_a and wait for the starting of the AP.

Case 2: The second round or later ($R_n, n \geq 2$)

When the second round starts, we have a set of working nodes and a set of sleeping nodes. Each working node that belongs to S_W has to increase its LID and all sleeping nodes wake up. Depending on the new K and the previous K values, there are three cases for this increase.

- Option 1: If there is no request to change the degree of coverage, the LID of each working node is increased by one. After increasing its LID, each working node will decide the next state for itself by comparing its LID with K. If the increased LID is greater than K, the working node will sleep for the next round.
- Option 2: If there is a request for a new increased K, each working node needs to increase its LID by the difference between the new increased K and the previous K. This process will make more layers than one.

- Option 3: If there is a request for a decreased K, each working node increases its LID by one, like in Option 1, and if its LID is greater than the new K, the node goes to the sleep state for the next round.

After the increment of LID, each working node generates T_n to decide the time when it sends a NOTIFY message to its neighbors. When T_n expires, it broadcasts a NOTIFY message containing its LID and the new K, as shown in Fig. 2 (a). By receiving the message, newly awakened nodes can determine which layers have already been formed by currently working nodes and how many new layers should be built by themselves. In addition, each awakened node generates its Random Activation Time (T_a).

2) Activation Phase (AP): All newly awakened nodes try to be working during the AP by sending out ACTIVE messages to their neighbors. While waiting for the Random Activation Time (T_a), each awakened node maintains a list of layers already composed by its neighboring nodes using the LIDs, which are included in the NOTIFY messages from working nodes in the previous round or in the ACTIVE messages from other awakened nodes. When its T_a expires, the node checks the list of layers that are already constructed. If it finds out a layer that is not made yet, the node sets its LID as the layer number and sends out its ACTIVE message with the LID as shown in Fig. 2 (b). After a node broadcasts its ACTIVE message, it will work as a working node during the WP. A node will go to sleep during the WP if all layers are already constructed before its T_a expires, as shown in Fig. 2 (c).

The decision on the coverage is made by the reception of an ACTIVE message within the distance of sensing radius r_s at each node. Through this approach, we can obtain a good approximation on 1-coverage for each layer. A more accurate analytical model will be studied further as a future work.

3) Working Phase (WP): A node with its LID between 1 and K works as a working node during the WP. All the other nodes go to sleep during the WP in the current round. During the WP, if a request for a new K that is less than the current K is received, each working node compares the new K with its LID and goes to the sleeping state instantly if its LID is greater than the new K, as shown in Fig. 3, which is the state transition diagram for each node during one round. This is another benefit of our protocol for dynamic reconfiguration. If the new K is larger than the current K, reconfiguration occurs in the next round. In Fig. 3, a transition occurs when the local timer (t) of a node indicates the start of the next phase and/or a certain condition is met.

IV. Circulation-based Reconfiguration (CBR)

In the previous section, we introduced a primitive reconfiguration scheme where all nodes wake up and generate a random and uniformly distributed activation time T_a. When its T_a expires, a node looks up any available layer that has not yet been selected by its neighbors, and will be activated by sending an ACTIVE message if one of K layers is missing. Since the ROAL protocol repeats the layering procedure at every working period, the potential cost of energy will be increased as the frequency of layering is raised for an increased fault ratio. The reason is because the ROA algorithm needs to build K layers at every round. We resolve this problem by modifying the layering procedure such that all possible layers are built at the first period and we circulate them at every period, removing the repetitive layering procedure.

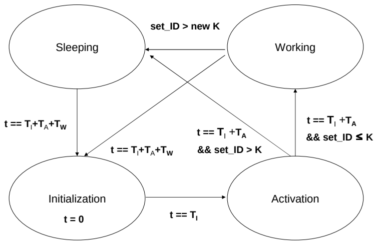


Figure 3. State Transition Diagram for Each Node During One Round

A. Model of CBR

In this section, we explain the details on the CBR scheme that circulates the layers without repeating the layering procedure at every working round T_w as the ROAL protocol needs. First, we define by $L(l)$ a set of nodes that have a layer_ID (LID) l.

Algorithm C-ROA (K, T_A)

$t \leftarrow 0$
$LID \leftarrow 0$
$H \leftarrow \emptyset$
$T_a \leftarrow rand(0, T_A)$
while $t < T_A$
  if $t = T_a$
    $LID \leftarrow min l | H(l) = false$
    send ACTIVE.LID
  else if ACTIVE.LID arrives
    $H(LID) \leftarrow true$
if $h \leftarrow max[H]$ is less than or equal to K
  $T_w = \infty$
else if my l is less than or equal to K
  calculate T_w and go to work
else
  calculate T_s and go to sleep

Every node obtains its LID l when its T_a expires. In the original ROAL protocol, only K layers are formed by the ROA algorithm. However, the C-ROA algorithm assigns layer id to all nodes. Then, each node will act following the circulation rule either setting a sleeping period (T_s) or a working period (T_w) based on its LID. The calculation of T_w and T_s is explained in the next section. Based on its LID, every node calculates its working or sleeping period.

B. Circulation Rule

A circulation rule is developed for the purpose of a low-power reconfiguration of working nodes while providing a robust and energy-balanced K-coverage. Usually, a sleeping node needs to wake up with a certain frequency to monitor if some healthy nodes are working within its sensing range for assuring a reliable quality of K-coverage. Hence, the frequency of the fault detection messages will be increased as a fault ratio increases, which deteriorates the energy credit in WSNs. A different way is implemented in this study using the circulation scheme that substitutes a set of working nodes with a different set of sleeping nodes, meanwhile guaranteeing a constant K-coverage. This property differentiates our approach from the Set K-cover study in that the degree of coverage can be changed whenever a different set is activated [5], [9].

Fig. 4 shows the state transition diagram of circulation. Using a LID obtained by the C-ROA algorithm, each node calculates T_w or T_s. Whenever T_w of working nodes or T_s of sleeping nodes expires, each node transits its state, as shown in Fig. 4. Hence, the transition between the working and sleeping state happens autonomously, which can reduce the energy consumed for a repetitive layering at every round as in the ROAL protocol.

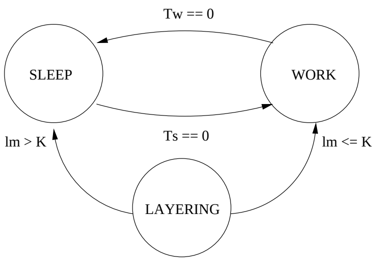


Figure 4. State Transition with Circulation

1) Deterministic Circulation: We implement a deterministic circulation, which uses a fixed and constant reconfiguration time T_r. Each node calculates its T_s or T_w using the predefined T_r. Each node knows the highest LID h by hearing the packets exchanged during the layering procedure. With a given degree of coverage K, its LID l, the highest layer id h and T_r, a node determines both T_s and T_w as follows.

A. Initial calculation for T_w and T_s

Initially, we need to activate all K layers for the first round. Each node that has a LID $\leq K$ decides to work and calculates its T_w. T_w will be different according to its LID l since only nodes in the bottom layer, i.e., layer 1, will sleep after this first round. For this reason, nodes that have LID l less than or equal to K will work during T_w that is calculated by:

$$T_w(l) = \begin{cases} T_r \times l, & \text{if } l \leq K \text{ and } h > K \\ \infty, & \text{if } l \leq K \text{ and } h \leq K. \end{cases}$$

If the highest LID h of a node v is smaller than K, v will set its $T_w = \infty$. If its LID is greater than K, v will go to sleep during T_s calculated as follows:

$$T_s(l) = (l - K) \times T_r, \text{ if } l \geq K.$$

We circulate one layer in a sequence of $C(1), C(2), C(3), \ldots, C(h)$ at every T_r, while sustaining K layers at every time instance. The term of circulation $C(l)$ defines that working nodes in layer l go to sleep and sleeping nodes in layer m wake up to work, where $m = (K + l) \bmod h$, if $m > 0$, or $m = h$ if not.

B. After initial calculation

Once the first period of working or sleeping mode finishes, the way of calculation changes as follows.

$$T_w(l) = T_r \times K$$
$$T_s(l) = (h - K) \times T_r.$$

Note that a current working node will have a new T_s and a newly wake-up node will calculate a new T_w. Because each node can decide its T_w and T_s using h and its LID with the given values of T_r and K, a local difference of density will not affect the the overall coverage.

2) Discussion on Fault Tolerance: Using the circulation scheme, we can provide an energy efficient fault tolerance while maintaining K-coverage. First of all, the K-coverage itself is a scheme to provide a high probability of detection unless K nodes within the unit sensing area have fault. In addition, the circulation scheme can provide a more robust environment by circulating a faulty layer with a new healthy layer. If we can decrease the interval of circulation, the recovery time for the faulty region can be minimized. We concern the cost of reconfiguration in terms of both the energy efficiency and the network performance. As shown in the experiment of an energy and delay in Section VI-C, the circulation proves itself as an energy-efficient fault-tolerant scheme for K-coverage in that the energy consumed for the reconfiguration remains almost constant and small even if the frequency of the reconfiguration increases dramatically. As well as the fault-tolerance, the circulation scheme is a good method for the energy-balance because a set of nodes belonging to a layer will be activated in a round-robin fashion. For the summary, we can say that the CBR scheme provides a robust and energy-efficient fault-tolerance with energy-balance.

V. Analysis on Layer Coverage

In this section, we analyze the probability of the coverage and connectivity of our layer coverage. The coverage and connectivity depends on the layering radius of r_l, the sensing radius of r_s, the communication range r_c, and a node density. We assume that the positions of nodes follow the Poisson point process of constant and finite density λ in area $\mathbb{R}^2$. Here, we assume that there will be no collision during the layering procedure for the simplicity of proof.

A random variable i that is the expected number of nodes in a circle area of πr^2 centered at a random point follows the Poisson distribution such as $P(i, r) =$

$e^{-\lambda\pi r^2}(\lambda\pi r^2)^i/i!$. At a random position, each node decides its layer by running ROA or C-ROA algorithm. Assume that the layering radius is r_l where a node sends an ACTIVE message to its neighbors. As shown in Fig. 5, the minimum distance between any nodes in the same layer will be $r_l/2$, and there is only one node within the area of $\pi(r_l/2)^2$ as depicted with the smallest ball in Fig. 5.

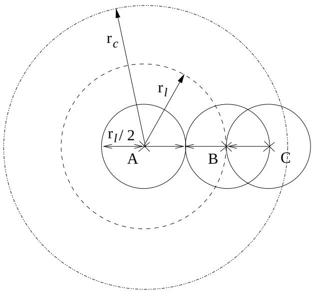


Figure 5. Minimum Distance between Two Working Nodes

A. Connectivity

The connectivity will be guaranteed in our layer coverage if any on-duty node working for a certain layer is connected. We mean by connected that any two nodes can reach to each other in multiple hops. We follow a similar procedure, as shown in the study [10] to prove the connectivity, but provide a different model using the Poisson point process model.

Lemma 1. Assume that n nodes are distributed in $\mathbb{R} = [0,l]^2$ according to the Poisson point process model, Then, there is almost surely at least 1 node per unit area of d^2 when l goes to infinity if the density of nodes satisfies $\lambda = \frac{k \ln l^2}{d^2}$ for $k > 1$.

Proof. We divide the region $\mathbb{R}$ into $N = \frac{l^2}{d^2}$ squares of size $d \times d$. Let $\mu_0(n, N)$ be the random variable denoting the number of empty squares of size $d \times d$, where n is the number of nodes in area $\mathbb{R}$, and p_0 is the probability of empty nodes in one square. Then, the expected number of empty cell $E[\mu_0(n, N)]$ will be:

$$E[\mu_0(n, N)] = N \cdot p_0 = N \cdot e^{-\lambda d^2} = Ne^{-nN^{-1}}.$$

Here, we want to find $E[\mu_0(n, N)]$ when $l \to \infty$. From the above equation, we obtain:

$$\ln E[\mu_0(n, N)] = \ln \frac{l^2}{d^2} - \frac{nd^2}{l^2}.$$

If we assume $nd^2 = kl^2 \ln l^2$, the above equation becomes:

$$\ln E[\mu_0(n, N)] = \ln \frac{1}{d^2 l^{2k-2}}.$$

If $k > 1$, then

$$\lim_{n,N\to\infty} lnE[\mu_0(n, N)] = -\infty.$$

Hence, $\lim_{n,N\to\infty} E[\mu_0(n, N)] = 0$ and there almost surely is at least 1 node in each square. The density of nodes will be:

$$\lambda = \frac{n}{l^2} = \frac{k \ln l^2}{d^2}, \text{ where } k > 1.$$

The connectivity will be satisfied if the communication range is greater than or equal to the expected maximum distance between two working nodes as proved in Theorem 1.

Theorem 1. A maximum distance between two active nodes that belong to the same layer is less than or equal to $(1 + \sqrt{5}) \times r_l$ if the density of nodes is satisfied as in Lemma 1.

Proof. Based on the condition of density derived in Lemma 1, we follow the complete proof of Lemma 3.1 and Theorem 3.1 in [2]. We assume that each square in Fig. 5 has only one node considering the worst case of Lemma 1 where $d = r_l$. Because there will be only one node per each square, node B is the furtheest node to be activated away from A in the gray square if A is activated in the dark square as shown in Fig. 6. However, if node C is activated earlier than B within the layering radius of r_l from B, node B will sleep. The distance between node A and node C is the maximum distance where any two working nodes can be apart from each other. Hence, the communication range $r_c = (1 + \sqrt{5})r_l$ is a minimum range to guarantee the connectivity. The details can be found in [2].

In addition, the connectivity for one layer of Theorem 1 is a sufficient condition for the case of K-coverage.

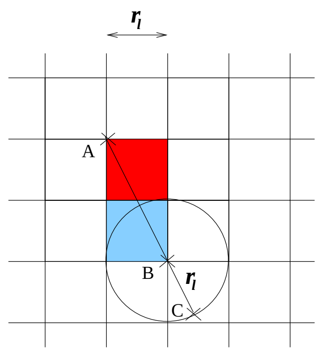


Figure 6. Maximum Distance of Two Closest Working Nodes

B. Coverage

If a given sensor network has the density λ, we can scale down the density to $\lambda_l(1)$ for 1-coverage obtained by the ROA or C-ROA algorithm. Because ROA and C-ROA selects one node per area of $\pi(r_l/2)^2$, the scaled-down density will be $\lambda_l(1) = \frac{1}{\pi(r_l/2)^2}\lambda = \frac{4\lambda}{\pi r_l^2}$. According to the Poisson point process model, the coverage is the probability of empty node within the sensing radius of r_s. Hence, the percentage of 1-coverage will be:

$$R_c = 1 - e^{-\lambda_l(1)\pi r_s^2} \quad (1)$$

$$= 1 - e^{-4\lambda\alpha^{-2}}, \alpha = r_l/r_s. \quad (2)$$

From (1), a minimum required density for R_c almost surely is calculated by

$$\lambda = \frac{-ln(1 - R_c)}{4} \cdot (\frac{r_l}{r_s})^2. \quad (3)$$

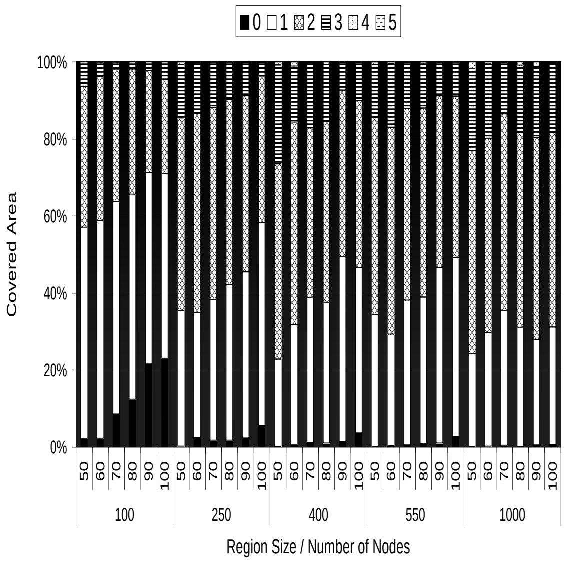
For K-coverage, one can easily calculate a required density by $\lambda_l(K) = \frac{K}{\pi(r_l/2)^2} \cdot \lambda$.

VI. Experimental Results

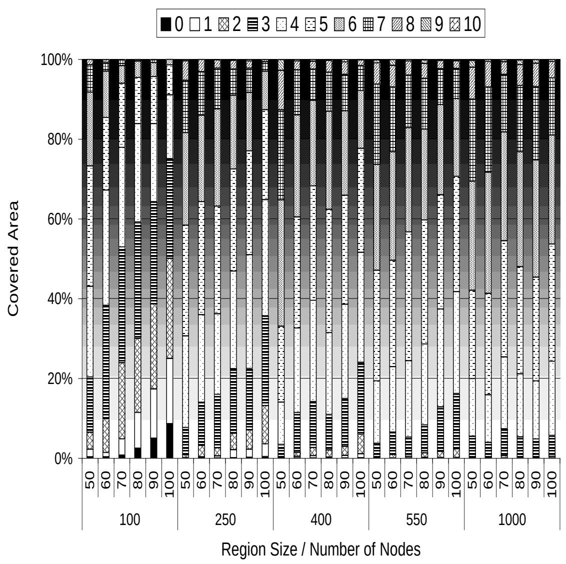
In this section, we show the results on coverage and network performance with the ROAL protocol.

A. Coverage Evaluation

To measure the coverage, the entire sensing region is divided into $1m \times 1m$ grids. Each point is considered to be covered if the point is located within the sensing range of a working node. The sensing range is $10m$, while the communication range is $30m$. Fig. 7 (a) and (b) show that



(a) ROAL Protocol with 1-coverage



(b) ROAL Protocol with 3-coverage

Figure 7. Ratios of Covered Areas

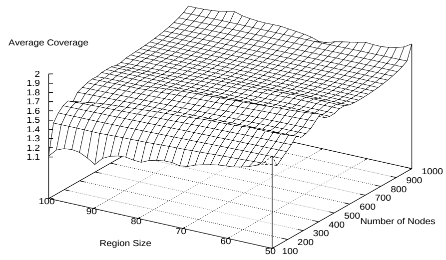
the percentages of the covered areas for 1- and 3-coverage networks with the ROAL protocol, respectively. Each bar represents the ratios of resulting coverages for a specific region size/number of nodes. The upper row of the X-axis indicates the size of the region where sensor nodes are deployed. For example, 50 implies a $50m \times 50m$ region. The lower row of the X-axis indicates the number of sensor nodes deployed in the region. The ratio of the uncovered area with 1-coverage in Fig. 7 (a) reaches up to 24% when the density is 0.01 (100 nodes/10,000m^2), which is the worst case. If the density exceeds 0.025 (250 nodes/10,000m^2), the ratio of the uncovered area decreases to below 5%. The ratio of the uncovered area with 3-coverage (0-, 1-, and 2-coverages) in Fig. 7 (b) reaches up to 50% when the density is 0.01. However, as the density increases, this ratio also becomes small. According to our observation, there still exists around 8% (for 1-coverage) and 23% (for 3-coverage) uncovered area with all sensor nodes working with the same number of nodes and the network size. Hence, the uncovered area incurred by the ROAL protocol is very small, less than 2% of the total region.

Fig. 8 (a) and (b) show the average degrees of 1- and 3-coverage networks with the ROAL protocol. The average degrees of 1-coverage range from 1.1 to 2 with different densities. This implies that the ROAL protocol can efficiently manage the quality of the required degree of coverage using a reasonable number of working nodes. The average degrees of the 3-coverage network also range from 2.5 to 6. Fig. 9 shows the number of working nodes with the region size of $50m \times 50m$ for 1-coverage and 3-coverage, respectively. The actual number of working nodes grows very slowly, while the number of the sensor nodes increases steeply. Compared to the results obtained using the geographic information in the CCP [3], the ROAL protocol can provide very competitive results without using any geographic information. The results on the average degree for 1-coverage and the number of working nodes for 1 and 3-coverage are close to each other. Moreover, since the ROAL protocol requires much lower running overhead compared to the approaches that use the geographic information, it really improves the energy performance of the sensor network. In addition, our protocol can support the desired degree of coverage, which is not provided in PEAS protocol.

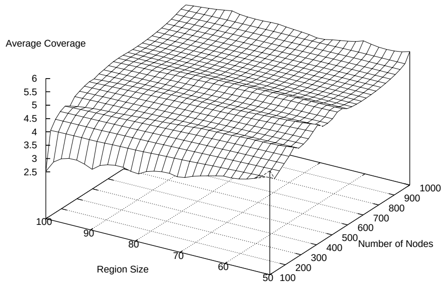
B. Network Performance

In this section, we evaluate the coverage lifetime and the packet delivery ratio, along with the residual energy of the network using the ns-2 simulator. We use the DSR routing protocol [11] to evaluate the ROAL protocol because it provides an on-demand source routing that does not need any location information and it is the basic routing scheme for other on-demand routing protocols.

For this simulation, $30m$ is set for the sensing radius and $75m$ for the communication radius of each node. We use $250m \times 250m$ 2-dimensional square for a target sensing region. In addition, there are 10 event points distributed randomly around the upper bound of the sensing area and each point generates 5 events per second. When working nodes around the event points sense the

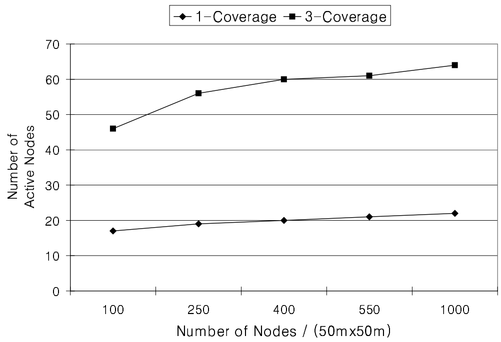


(a) Average Coverage Degrees with 1-coverage



(b) Average Coverage Degrees with 3-coverage

Figure 8. Average Coverage Degree



(a) Number of Working Nodes

Figure 9. Number of Working Nodes

generated events, they send a 512 byte packet per one event to the sink node that is located at the right bottom of the sensing region. The average coverage is measured by counting the number of neighboring nodes that detect the event. All results shown in this section are obtained using 1,000 second round time (5 seconds for T_I and T_A each, and 990 seconds for T_W), and simulation data are collected every 100 seconds. Also, each sensor node is given 100 Jules of initial energy.

Fig. 10 shows the average residual energy with the DSR protocol only (i.e., all nodes are working) for 50 nodes, 200 nodes, and 250 nodes, respectively. It is clear that, without any energy-saving scheme, the network with a small number of nodes has more residual energy than the one with a larger number of nodes. This implies that excessively redundant nodes cause more energy consumption with the DSR routing protocol that uses a broadcasting scheme. Fig. 11 (a) and (b) show the average

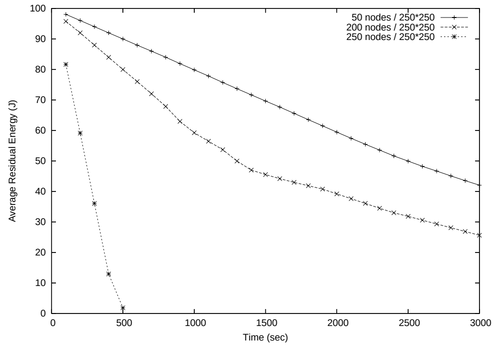


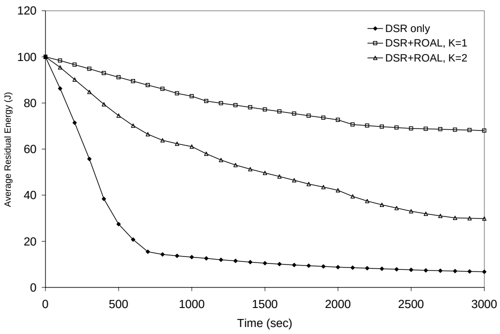
Figure 10. Average Residual Energy with DSR Only

residual energy and the minimum residual energy of the network for different coverage degrees (K = 1 and 2). With the ROAL protocol, the network can reserve more energy than with only the DSR protocol.

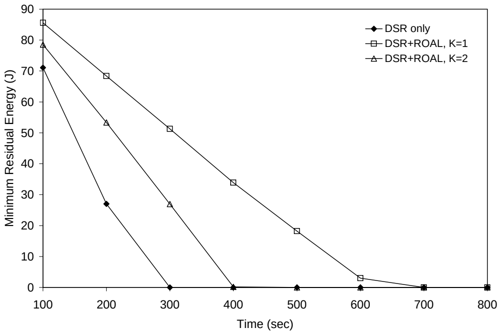
The average packet delivery ratio is shown in Fig. 12 (a). More than 95% of packets are dropped after 800 seconds with the DSR only. With the ROAL protocol, almost 100% packets are delivered up to 2,900 seconds when K is 2, and up to 3,000 seconds when K is 1. When K is 2, the delivery ratio drops to 0 after 2,900 seconds because some intermediate nodes between the sources and the sink node completely depletes their energy. Some temporal drops are caused by packet losses during the reconfiguration period. In Fig. 12 (b), the average degree of coverage is shown with 380 sensor nodes. The average degree of 2-coverage remains around 2.0, while 1-coverage shows the average degree over 1 during the whole simulation time. Without the ROAL protocol, the average degree of coverage is around 5 at the beginning of the simulation, but it rapidly drops to around 1 after 300 seconds, since sensor nodes around the event points have died together for energy depletion except about one working node. Therefore, the network with the ROAL protocol can capture the events for a longer time since it uses a small number of different working nodes in each round. In addition, the results also prove that the ROAL protocol can provide the required degree of coverage efficiently.

C. Simulation

Using 300 nodes deployed over a two-dimensional area of 500m$\times$500m, we obtain the total energy consumed for the layering procedure and the average delay incurred for data packet delivery using the ns-2 simulator. The other options are 25m for r_s, 70m for r_c, and $K = 3$. Dynamic source routing (DSR) is used as a routing protocol. In Fig. 13, we compare the energy consumption for the reconfiguration. The C-ROAL protocol consumes a constant small energy for the layering procedure even though the reconfiguration periods vary from 50 to 1,000 seconds. However, the total energy consumption increases

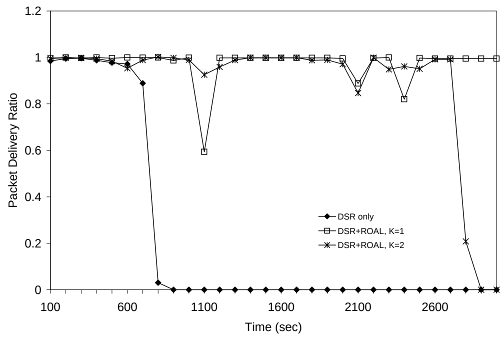


(a) Average Residual Energy

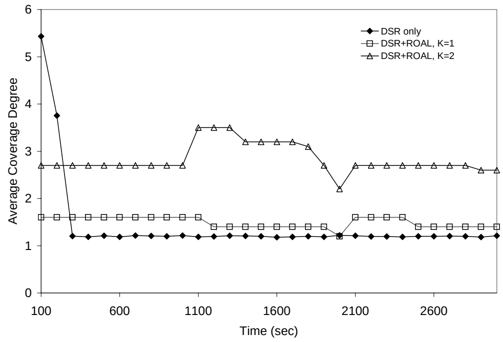


(b) Minimum Residual Energy

Figure 11. Residual Energy with the ROAL Protocol



(a) Packet Delivery Ratio



(b) Average Coverage Degree

Figure 12. Packet Delivery Ratio and Average Coverage Degree

greatly as the reconfiguration period becomes shorter in the ROAL protocol. We also show the average delay of a data packet incurred by the reconfiguration in Fig. 14. We can see that the packets are delivered much faster with the C-ROAL protocol, while the delay is increasing as the number of reconfiguration is increased in ROAL. The reason is because C-ROAL protocol never deters the packet delivery due to the autonomous circulation. We also expect that the difference of the total reconfiguration energy will be increased as the number of nodes increases. From the above results, we can say that the C-ROAL protocol can greatly improve the energy efficiency, while providing K-coverage with both the energy-balance and the fault-tolerance together for WSNs.

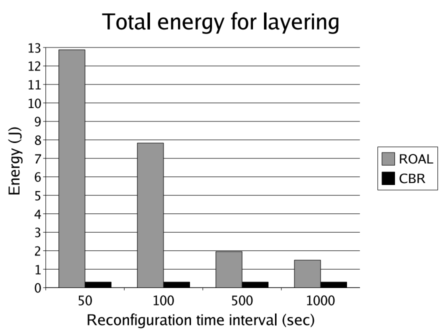


Figure 13. Comparison of Energy Consumption for Reconfiguration

VII. Conclusion

We proposed a fast and efficient K-coverage algorithm, called the ROAL protocol, to solve the problem of providing a certain degree of coverage in WSNs. The main idea of the ROAL protocol is to ensure K-coverage using K subsets of working nodes using the layering concept, where each subset guarantees 1-coverage. The ROAL protocol efficiently constructs K-coverage network with low message overheads and guaranteed packet delivery with the advantages of energy-savings in the network. We developed an energy efficient circulation scheme that can improve the fault-tolerance and energy-balance for K-coverage using a new circulation and C-ROA algorithm. The circulation scheme can reconfigure the sets of working nodes with a greatly decreased energy consumption for the ROAL protocol. This property enhances the energy balance and fault tolerance for WSNs.

We also proved that the ROAL protocol can guarantee the connectivity and coverage if a certain minimum density is satisfied regarding the sensing and the layering radius. Simulation results also support our claim.

In future work, we will suggest more useful schemes to select the working node sets regarding energy burdens in each node and may study on the measurement scheme for the duration of each phase regarding both maximal and a given desired network lifetime. We will implement a more concrete strategy that can replace a fault node in a certain layer with a healthy node in other layers to stabilize the QoS during the network lifetime.

References

[1] C. F. Huang and Y. C. Tseng, "The Coverage Problem in a Wireless Sensor Network," in *Proceedings of the 2nd ACM International Conference on Wireless Sensor Networks and Applications*, 2003.

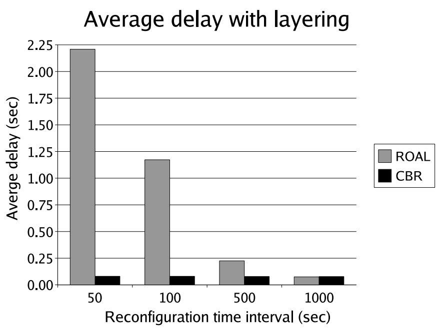


Figure 14. Average Delay Incurred by Reconfiguration

[2] F. Ye, G. Zhong, S. Lu, and L. Zhang, "PEAS: A Robust Energy Conserving Protocol for Long-lived Sensor Networks," in *Proceedings of ICDCS*, May 2003.
[3] G. Xing, X. Wang, Y. Zhang, C. Lu, R. Pless and C. Gill, "Integrated Coverage and Connectivity Configuration for Energy Conservation in Sensor networks," in *Processings of ACM Sensys*, 2003.
[4] S. Kumar, T. H. Lai, and A. Arora, "Barrier Coverage with Wireless Sensors," in *Proceedings of MobiCom*, 2005.
[5] Z. Abrams, A. Goel, and S. Plotkin, "Set K-Cover Algorithms for Energy Efficient Monitoring in Wireless Sensor Networks," in *Proceedings of IPSN'04*, April 2004.
[6] S. Meguerdichian, F. Koushanfar, G. Qu, and M. Potkonjak, "Exposure in wireless Ad-Hoc sensor networks," in *Proceedings of MOBICOM*, 2001.
[7] S. Kumar, T. Lai, and J. Balogh, "On k-coverage in a mostly sleeping sensor network," in *Proceedings of MOBICOM*, 2004.
[8] T. He, S. Krishnamurthy, J. A. Stankovic, T. Abdelzaher, L. Luo, R. Stoleru, T. Yan, and L. Gu, "Energy-efficient surveillance system using wireless sensor networks," in *Processings of ACM Mobisys*, 2004.
[9] S. Slijepcevic and M. Potkonjak, "Power Efficient Organization of Wireless Sensor Networks," in *Proceedings of IEEE International Conference on Communications (ICC)*, 2001.
[10] D. M. Blough and P. Santi, "Investigating upper bounds on network lifetime extension for cell-based energy conservation techniques in stationary ad hoc networks," in *Proceedings of MOBICOM*, 2002, pp. 183–192.
[11] D. B. Johnson, D. A. Maltz, and J. Broch, "DSR: The Dynamic Source Routing Protocol for MultiHop Wireless Ad Hoc Networks," *IEEE Transactions on Mobile Computing*, 1996.

Hogil Kim was born on June 19, 1969, in Andong, South Korea. After graduating in Korea Military Academy, he had served as an army officer during the next 2 years. He received his master degree on electrical engineering at Younsei university, South Korea, in 1996.

Currently, he is an assistant professor at Korea Army Academy at Yeong-cheon. His research topic is focused on an energy efficient topology management for WSN.

Eun Jung Kim received the BS degree in Computer Science from Korea Advanced Institute of Science and Technology, Korea, in 1989, the MS degree in Computer Science from Pohang University of Science and Technology, Korea, in 1994, and the PhD degree in Computer Science and Engineering from the Pennsylvania State University in 2003. From 1994 to 1997, she worked as a member of Technical Staff in Korea Telecom Research and Development Group. Dr. Kim is currently an assistant professor in the Department of Computer Science in Texas A&M University. Her research interests include Computer Architecture, Parallel/Distributed Systems, Computer Networks, Cluster Computing, QoS Support in Cluster Networks and Internet, Performance Evaluation, Wireless Sensor Netowrks, and Fault-Tolerant Computing. She is a member of the IEEE Computer Society and of the ACM.

Ki Hwan Yum received the BS degree in Mathematics from Seoul National University, Korea, in 1989, the MS degree in Computer Science and Engineering from Pohang University of Science and Technology, Korea, in 1994, and the Ph.D. degree in Computer Science and Engineering from the Pennsylvania State University in 2002. From 1994 to 1997, he was a member of Technical Staff in Korea Telecom Research and Development Group. Dr. Yum is currently an assistant professor in the Department of Computer Science in the University of Texas at San Antonio. His research interests include Computer Architecture, Parallel/Distributed Systems, Cluster Computing, and Performance Evaluation. He is a member of the IEEE Computer Society and of the ACM.