Efficient Scheduling of Sensor Activity for Information Coverage in Wireless Sensor Networks

S. Vashistha, A. P. Azad, A. Chockalingam
Department of ECE, Indian Institute of Science, Bangalore-560012

Abstract: **In this paper, we are concerned with algorithms for scheduling the sensing activity of sensor nodes that are deployed to sense/measure point-targets in wireless sensor networks using *information coverage*. Defining a set of sensors which collectively can sense a target accurately as an *information cover*, we propose an algorithm to obtain Disjoint Set of Information Covers (DSIC), which achieves longer network life compared to the set of covers obtained using an Exhaustive-Greedy-Equalized Heuristic (EGEH) algorithm proposed recently in the literature. We also present a detailed complexity comparison between the DSIC and EGEH algorithms.**

Keywords – *Sensor activity scheduling, physical coverage, information coverage, network lifetime.*

I. Introduction

Recent advances in the area of embedded systems and wireless communications have enabled the development of small-sized, low-cost, low-power sensor nodes that can communicate over short distances wirelessly. In addition to their traditional sensing function, these sensor nodes can perform processing and communications functions. The processing and communication functions embedded in these sensor nodes essentially allow networking of these nodes, which in turn can facilitate sensing function to be carried out in remote/hostile areas. A network of sensor nodes (often referred to as a wireless sensor network) can be formed by densely deploying a large number of sensor nodes in a given sensing area, from where the sensed data from the various sensor nodes are transported to stations which are often located far away from the sensing area [1],[2]. Energy is consumed in the sensor nodes for the purpose of sensing as well as communication. Several studies in the literature have addressed the issue of minimizing the energy spent for the purpose of communication (e.g., energy efficient routing [3]). In this paper, we address the energy spent for the purpose of sensing, focusing on energy efficient algorithms for scheduling the sensing activity of sensor nodes *using Information Coverage* [4].

Several interesting applications of wireless sensor networks involve certain physical quantity or phenomenon at a given location (referred to as a *target*) to be monitored using the deployed sensor nodes over a long period of time (e.g., monitoring the radiation level or temperature of a source at a given location in a remote/inaccessible area). The intensity of such a physical quantity to monitor typically decays with distance. A target is said to be 'covered' if a sensor individually (or a set of sensors collectively) can measure (i.e., sense) the target with certain acceptable accuracy. A set of nodes which collectively can sense a target in the network with acceptable accuracy is referred to as a *cover for that target*. In the case of multiple targets in the network, a set of nodes which together can sense all the targets in the network with acceptable accuracy is referred to as a *cover for the network*. There can be several such network covers feasible for a given number of targets and sensor nodes in the network (referred to as *set of feasible covers*). The network ceases to be of any use if there are no feasible covers left in the network, indicating end of network life. With the availability of only a finite amount of battery power in sensor nodes (replacing or recharging of batteries in sensor nodes are often not possible in such applications), it is of significant interest to retain sensor coverage in the network for the longest possible time.

The sensor coverage problem has been investigated by many in the literature [4]-[12]. In most of the above studies, the coverage model used is *physical coverage*, where a point-target is said to be covered by a sensor if the target is located within the *physical coverage radius* (PCR) of that sensor. PCR of a sensor is defined as the maximum distance between the sensor and the target up to which the sensor can sense the target with acceptable accuracy. More recently, Wang *et al*, in [4], have proposed a new coverage model, termed as *information coverage*, which is based on estimation theory to exploit collaboration among multiple sensors to accurately sense a target even if that target falls outside the PCR of all the sensors. That is, even if a target is not physically covered, it can be information covered through multiple sensors collaborating to make an accurate estimation of the target. Further extending their work on information coverage, they proposed a three-step heuristic in [12], referred to as Exhaustive-Greedy-Equalization Heuristic (EGEH), to obtain information covers for scheduling the sensing activity of sensor nodes, allowing up to K_{max} sensor nodes to collaborate for achieving information coverage. Through simulation results, they showed that the network lifetime achieved using their information coverage

This work was supported in part by the Indo-French Centre for Promotion of Advanced Research, New Delhi, under Project 2900-IT.

1-4244-0614-5/07/$20.00 ©2007 IEEE.

based EGEH algorithm is better than the physical coverage based heuristic proposed by Slijepcevic and Potkonjak in [8].

In this paper, we take the information coverage approach in [4] and propose a heuristic for obtaining disjoint information covers, which we refer to as Disjoint Set Information Covers (DSIC) algorithm, that results in longer network lifetime than the EGEH algorithm in [12]. We point out that the information covers obtained by the EGEH algorithm are not disjoint, i.e., a sensor can participate in more than one cover. Because of this, the scheduling of sensors becomes more involved than a simple round-robin schedule. In addition, the sensors participating in more than one cover can end up depleting their battery power sooner than others, leading to an early end of network life although other sensor nodes may be left with battery power.

Our proposed DSIC algorithm, on the other hand, generates disjoint information covers (i.e., a sensor node can participate in at most one cover). The resulting disjoint information covers allow a simple round-robin schedule of sensor activation (i.e., activate the covers sequentially). Further, since the sensors are uniformly drained, the network lifetime achieved by our DSIC algorithm is more than that achieved by the EGEH algorithm. We illustrate the above performance improvement through examples and detailed simulations. We also present a detailed complexity comparison between the DSIC and EGEH algorithms. We show that there is a cross-over between the complexities of these two algorithms, depending on the number of targets, the number of sensor nodes, and the maximum number of nodes allowed to collaborate for sensing.

The rest of this paper is organized as follows. Section II presents some preliminaries on information coverage. In section III, we present the proposed DSIC algorithm. Simulation results on performance and complexity comparison are presented in section IV. Conclusions are presented in section V.

II. Information Coverage

In this section, we present the necessary preliminaries on the estimation theory based [13] information coverage introduced in [4],[12]. Consider a sensing area with K spatially distributed sensor nodes s_i, $i = 1, 2, \cdots, K$. Consider a point-target at a known location in the sensing area. Let θ be the physical quantity (or parameter) denoting the target that needs to be measured/monitored. Let d_i, $i = 1, 2, \cdots, K$ denote the distance between the target and sensor node s_i. Assume that θ varies inversely with distance (which is true with many physical quantities of interest, e.g., radiation level), with a decay exponent α, $\alpha > 0$, such that the parameter observed at distance d is given by $\theta d^{-\alpha}$. In addition to this decay, additive noise at the sensor also corrupts the observation at the sensor node. Accordingly, the observation of the parameter at sensor node s_i is given by

$$y_i = \theta d_i^{-\alpha} + n_i, \quad i = 1, 2, \cdots, K. \tag{1}$$

where n_i denotes the additive noise at sensor node s_i. A linear unbiased estimator [13] can be employed to estimate θ using

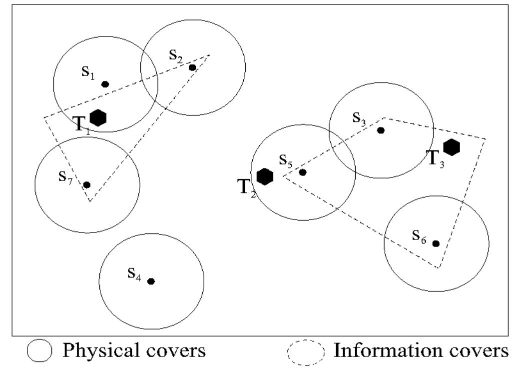


Fig. 1. Illustration of physical covers and information covers

the K different observations, y_i's. Let $\hat{\theta}_K$ denote such an estimate of θ using K observations. This estimate will be more accurate for large K and small noise variances. Let $\tilde{\theta}_K$ denote the error in the estimation, given by

$$\tilde{\theta}_K = \theta - \hat{\theta}_K. \tag{2}$$

For the estimate to be reliable, it is desired that the probability that the estimation error is less than a given value to be adequately large.

A. Definition:(K, ϵ) Information coverage

A target is said to be (K, ϵ) information covered, if K sensors collaborate to estimate the parameter θ at the target such that

$$\text{prob}\big(|\tilde{\theta}_K| \leq A\big) \geq \epsilon, \quad 0 \leq \epsilon \leq 1, \tag{3}$$

where $\tilde{\theta}_K$ is the estimation error given by (2), and A is threshold level below which estimation error is acceptable.

Note-1: The special case of $(1, \epsilon)$ information coverage (i.e., $K = 1$) corresponds to physical coverage. It can also be seen that if a target is (K, ϵ) covered, then that target is also $(K + 1, \epsilon)$ covered.

Illustration-1: Figure 1 illustrates the physical and information covers in a rectangular sensing area with 3 targets $\{T_1, T_2, T_3\}$ and 7 sensors $\{s_1, s_2, \cdots, s_7\}$. It can be seen that target T_2 is physically covered sensor s_5, and target T_1 is physically covered by sensor s_1. In addition, target T_1 is also $(2, \epsilon)$ information covered by sensors s_2 and s_7. Target T_3 is not physically covered, but it is $(3, \epsilon)$ information covered by sensors s_3, s_5 and s_6. Sensor s_4 contributes to neither physical nor information coverage of any of the targets. In summary, $i)$ $\{s_1, (s_2, s_7)\}$ is the set of feasible covers for target T_1, $ii)$ $\{s_5\}$ is the set of feasible covers for target T_2, and $iii)$ $\{(s_3, s_5, s_6)\}$ is the set of feasible covers for target T_3. Consequently, $\{(s_1, s_3, s_5, s_6), (s_2, s_7, s_3, s_5, s_6)\}$ is the set of feasible network information covers that can cover all the targets in this illustration.

B. Scheduling Sensing Activity using Covers

We note that the covers obtained are used to schedule the sensing activity of various sensors. Our aim is to cover *all* the targets in the network. Even if one target is not covered, then the network ceases to fulfill its sensing objective. *In other words, network lifetime is defined as the time up to which the deployed sensors can cover all the targets in the network.* For example, in Fig. 1, if physical coverage is employed then target T_3 is not covered at all, and so the network lifetime in this case is zero (i.e., the network can never sense all targets using physical coverage). On the other hand, if information coverage is used as in the above illustration in Fig. 1, then the information covers (s_1, s_3, s_5, s_6) and $(s_2, s_7, s_3, s_5, s_6)$ can be alternatively scheduled to sense all the targets. That is, in time slot 1, sensors s_1, s_3, s_5, s_6 are activated, and in time slot 2, sensors s_2, s_3, s_5, s_6, s_7 are activated, and so on as shown in Fig. 2, such that this activation cycle is continued until both the information covers become invalid (a cover is said to become invalid if any of the sensors in that cover is fully drained out of its battery power, and because of which that cover can not sense all the targets). Accordingly, the problem we address in this paper is to obtain information covers that will result in long network lifetimes.

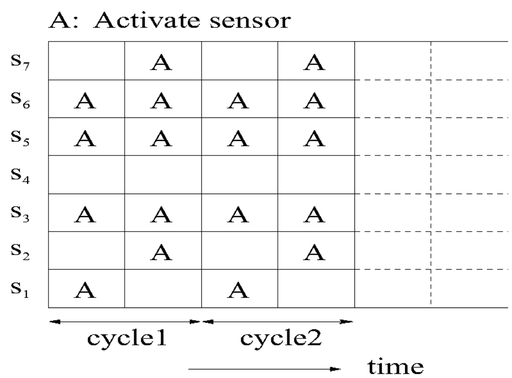


Fig. 2. An illustration of scheduling the sensing activity of sensors using information covers in Fig. 1.

C. Area Covered by Information Coverage

Note-2: An area is said to be completely (K, ϵ) covered if all the points in the area are (K, ϵ) covered.

Illustration-2: We present a comparison of the area covered by information coverage and physical coverage in Figs. 3 and 4. The physical coverage radius, R, is taken to be 1. This unit range corresponds to a A value equal to the noise standard deviation σ, and an ϵ value equal to 0.683 for $(1, \epsilon)$ coverage (i.e., physical coverage) in Eqn. (3) [12]. The area covered by one sensor using physical coverage is then a circular area of unit radius, i.e., $\mathcal{A}_{(1,\epsilon=0.683)} = \pi R^2 = \pi = 3.14$. The area covered by two sensors s_1 and s_2, separated by distance $d = 2$ is shown in Fig. 3. While physical coverage in this case results in a covered area of $2\pi R^2 = 2\pi = 6.28$ (denoted by two circles each of radius R, centered at s_1 and s_2), the $(2, \epsilon)$ information coverage satisfying Eqn. (3) results in a larger covered area of 7.9 (area of two unit-radius circles plus the black shaded area around the two circles in Fig. 3). Similarly, the area covered by $(3, \epsilon)$ information coverage using sensors s_3, s_4, s_5 each separated from the other by $d = 2$ as shown in Fig. 3 is 12.9 whereas the area covered by physical coverage with these three sensors is only $3\pi = 9.42$. Figure 4 shows the area covered as a function of the sensor separation distance d in Fig. 3 both for physical coverage (PC) as well as information coverage (IC). From Fig. 4, it is observed that information coverage covers a larger area compared to physical coverage, $\mathcal{A}_{(K,\epsilon)} \geq \mathcal{A}_{(1,\epsilon)}$ for $K \geq 1$.

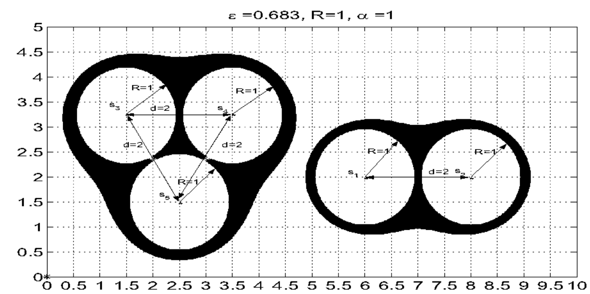


Fig. 3. Illustration of area covered by physical coverage and information coverage.

III. Proposed DSIC Algorithm

As stated earlier, our aim is to obtain information covers for scheduling sensor activity that can result in long network lifetime. In [12], Wang *et al* have proposed a Exhaustive-Greedy-Equalized Heuristic (EGEH) to obtain information covers. The information covers they obtained through EGEH are not disjoint. That is, a sensor can participate in more than one cover. Because of this, the resulting schedule of sensor activity is more involved than a simple round-robin schedule. Disjoint set covers, where a sensor can participate in at most one cover, on the other hand, are advantageous because a simple round-robin schedule is adequate.

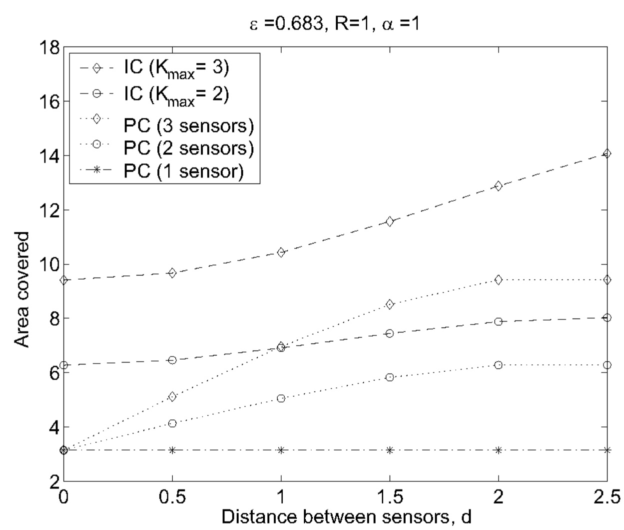


Fig. 4. Area covered by physical coverage (PC) and information coverage (IC). $\alpha = 1$, $R = 1$, $\epsilon = 0.683$.

The coverage problem using *disjoint set* of *physical* covers has been studied widely in the literature, where the disjoint set physical cover problem has been shown to be NP-complete [7], and so heuristics have been proposed to obtain disjoint set physical covers [7],[8],[9]. The problem of obtaining *disjoint set* of *information* covers is of interest, which has not been addressed so far to our knowledge. We, in this section, address this problem of obtaining disjoint set of information covers. We note that our problem being a (K, ϵ) coverage problem, in which $(1, \epsilon)$ coverage (i.e., physical coverage) is a special case, is also NP-complete. Hence, we propose a heuristic to obtain disjoint set information covers. We refer to the proposed heuristic as disjoint set information covers (DSIC) algorithm, which achieves longer lifetime than the EGEH algorithm in [12]. We explain the DSIC algorithm in the following.

A. Network

Consider a set of N homogeneous sensor nodes $\mathcal{S} = \{s_1, s_2, \cdots, s_N\}$ and a set of M targets $\mathcal{T} = \{T_1, T_2, \cdots, T_M\}$, which are distributed randomly in a sensing area. A target T_i can be sensed/monitored by a single sensor only if that target falls inside the the physical coverage range of that sensor. However, as described in the previous section, sensors can collaborate to sense the target T_i, even if T_i lies outside the physical coverage range of these sensors. Let K_{max} denote the maximum number of sensors allowed to collaborate in sensing a target. $K_{max} = 1$ corresponds to physical coverage. Let the operator $|\mathcal{B}|$ denote the cardinality of set $\mathcal{B}$ (i.e., number of elements in set $\mathcal{B}$). Let

$$\mathcal{C} = \left\{C_1, C_2, \cdots, C_{|\mathcal{C}|}\right\} \tag{4}$$

denote a disjoint set of information covers that cover all the targets, where the jth information cover C_j, $j = 1, 2, \cdots, |\mathcal{C}|$ denote a subset of $\mathcal{S}$ such that *all* targets can be information covered by using *all* sensors in C_j, and $C_j \bigcap C_k = \phi$ for $j \neq k$.

B. Algorithm

We propose to obtain $\mathcal{C}$, a disjoint set of information covers in (4), using the proposed heuristic. The operation of the proposed DSIC heuristic can be divided in two steps. In Step-I, disjoint set of information covers for each target in the network is obtained. In Step-II, using the outcome in Step-I, a disjoint set of information covers that can cover all the targets is obtained.

1) Step-I: In this step, we obtain a disjoint set of information covers for each target. For a given target T_j, $j = 1, 2, \cdots, M$, let $\mathcal{U}_j$ denote the set of disjoint information covers for target T_j such that

$$\mathcal{U}_j = \left\{u_j^1, u_j^2, \cdots, u_j^{|\mathcal{U}_j|}\right\}, \tag{5}$$

where u_j^q, $q = 1, 2, \cdots, |\mathcal{U}_j|$, is the qth information cover of target T_j in $\mathcal{U}_j$, and

$$u_j^q \bigcap u_j^r = \phi, \quad \text{for}\, q \neq r. \tag{6}$$

	Step-I of the proposed DSIC algorithm
1	**Loop** $j : 1 \leq j \leq M$
2	***Initialize*** $\mathcal{U}_j = \phi$, $\mathcal{S}_t = \mathcal{S}$
3	**Loop** $K : 1 \leq K \leq K_{max}$
4	$\mathcal{W}_j(K, \mathcal{S}_t) = \Big\{(s_{i_1}, s_{i_2}, \cdots, s_{i_K}),$ $\forall \big\{1 \leq i_1 \leq \lvert\mathcal{S}_t\rvert - K + 1,$ $i_1 + 1 \leq i_2 \leq \lvert\mathcal{S}_t\rvert - K + 2,$ $\cdots, i_{K-1} + 1 \leq i_K \leq \lvert\mathcal{S}_t\rvert\big\} : \beta \geq \epsilon\Big\},$ where $\beta = \Pr\{\lvert\tilde{\theta}_{K,j}\rvert \leq A\} \geq \epsilon$
5	Sort($\mathcal{W}_j$) for $\uparrow \beta$
6	**Loop** $a : 1 \leq a \leq \lvert\mathcal{W}_j\rvert$
7	**if** $s_i \neq s_k$ for $s_i \in w_j^a$ and $s_k \in u_j^q$ for $i, k = 1, 2, \cdots, N$ and $q = 1, 2, \cdots, \lvert\mathcal{U}_j\rvert$
8	$\mathcal{U}_j = \mathcal{U}_j \biguplus w_j^a$
9	**end if**
10	**end Loop** a
11	$\mathcal{S}_t = \mathcal{S}_t - \Big\{s_i : s_i \in u_j^q, i = 1, 2, \cdots, N$ and $q = 1, 2, \cdots, \lvert\mathcal{U}_j\rvert\Big\}$
12	**end Loop** K
13	**end Loop** j

TABLE I
STEP-I OF THE PROPOSED DSIC ALGORITHM

Now, Step-I of the proposed DSIC algorithm, which is given in pseudo-code form in Table I, essentially obtains $\mathcal{U}_j$ for $j = 1, 2, \cdots, M$. Some key steps in Step-I of the algorithm are explained as follows.

Line 4: Line 4 in the algorithm generates all feasible information covers for a given target T_j and a given K, from a set of sensor nodes $\mathcal{S}_t$. The algorithm starts with $\mathcal{S}_t = \mathcal{S}$ for $K = 1$. $\mathcal{W}_j = \{w_j^1, w_j^2, \cdots, w_j^{|\mathcal{W}_j|}\}$ denotes the set of generated covers, where $0 \leq |\mathcal{W}_j| \leq \binom{|\mathcal{S}_t|}{K}$. For example, for the scenario in Fig. 1,

$$\begin{aligned} \mathcal{W}_1 &= \{(s_1)\} \text{ for } K = 1 \\ &= \{(s_2, s_7)\} \text{ for } K = 2 \\ \mathcal{W}_2 &= \{(s_5)\} \text{ for } K = 1 \\ \mathcal{W}_3 &= \{(s_3, s_5, s_6)\} \text{ for } K = 3. \end{aligned}$$

Line 5: For a given K, Line 5 arranges the elements of $\mathcal{W}_j$ for that K in the increasing order of their respective $\beta = \Pr\{|\tilde{\theta}_{K,j}| \leq A\}$ values. For illustration purposes, consider an example, where for $K = 2$

$$\mathcal{W}_j = \{(s_2 s_3), (s_2 s_4), (s_3 s_5)\}, \tag{7}$$

and let their corresponding β values be $\beta_{(s_2,s_3)} = 0.93$, $\beta_{(s_2,s_4)} = 0.81$, $\beta_{(s_3,s_5)} = 0.97$. The sorting operation in Line 5 in this case will result in

$$\mathcal{W}_{j_{sorted}} = \{(s_2 s_4), (s_2 s_3), (s_3 s_5)\}. \tag{8}$$

Lines 6 to 10: It is noted that the elements of $\mathcal{W}_j$ need not be disjoint (e.g., sensor s_3 is common to two information covers in the $\mathcal{W}_j$ in (8)). From these sorted set of non-disjoint information covers for a given K, Lines 6-10 pick disjoint information covers sequentially starting from the first element in the sorted $\mathcal{W}_j$. The relevance of picking disjoint information covers sequentially from the sorted $\mathcal{W}_j$ is that we give first preference to covers having smaller β values, which typically are from the covers involving nodes far from the target. This preference enables the effective use of energy in far-away nodes, which also results in improved network lifetime. In addition, it also results in a larger number of disjoint covers (we will illustrate this using an example in the next section).

It is noted that Line 7 in the algorithm checks for the mutual-exclusive condition. In Line 8, the $\biguplus$ operator is used to denote inclusion of a set (e.g., w_j^a) as an element into another set (e.g., $\mathcal{U}_j$). For example, if $\mathcal{U}_j = \{(s_1), (s_2, s_4), (s_3 s_7)\}$ and $w_j^a = (s_5, s_6)$, then $\mathcal{U}_j \biguplus w_j^a = \{(s_1), (s_2, s_4), (s_3, s_7), (s_5, s_6)\}$.

In the $\mathcal{W}_j$ example in (8), the Lines 6-10 will pick disjoint covers $\{(s_2 s_4), (s_3 s_5)\}$.

Line 11: In Line 11, the set of sensor nodes $\mathcal{S}_t$ is updated by removing those nodes which have already been used in the disjoint covers in the previous iterations of K. This updated $\mathcal{S}_t$ is subsequently used in Line 4 in the iteration for $K+1$.

Thus, at the end of Step-I of the algorithm, we get the set of disjoint information covers for all the targets, i.e., $\mathcal{U}_j$'s $j = 1, 2, \cdots, M$.

2) Step-II: The Step-II of the algorithm in Table II takes the $\mathcal{U}_j$'s, $j = 1, 2, \cdots, M$ obtained from Step-I as the input, and generates a set of disjoint information covers $\mathcal{C} = \{C_1, C_2, \cdots, C_{|\mathcal{C}|}\}$, where C_i can cover all the targets in the network and $C_i \bigcap C_k = \phi$ for $i \neq k$.

We will explain the key steps in Step-II by taking the following example. Consider two targets T_1 and T_2, and let the Step-I give the following disjoint covers for targets T_1 and T_2:

$$\mathcal{U}_1 = \{(s_1), (s_2, s_4), (s_3, s_7), (s_5, s_6)\} \quad (9)$$

$$\mathcal{U}_2 = \{(s_3), (s_2, s_4), (s_5, s_7), (s_1, s_6, s_8)\} \quad (10)$$

Line 15-24: These lines generate all combinations of elements of $\mathcal{U}_j$'s (i.e., $\mathcal{M}_{a_1 a_2 \cdots a_M}$ in the algorithm) that result in information covers (not necessarily disjoint) that will cover all targets in the network.

For the example of $\mathcal{U}_1$ and $\mathcal{U}_2$ in (9) and (10), $\mathcal{V}_1 = \{v_1^1, v_1^2, \cdots, v_1^{|\mathcal{V}_1|}\} = \{(s_1), (s_2, s_4), (s_3, s_7), (s_5, s_6), \phi\}$, $\mathcal{V}_2 = \{v_2^1, v_2^2, \cdots, v_2^{|\mathcal{V}_2|}\} = \{(s_3), (s_2, s_4), (s_5, s_7), (s_1, s_6, s_8), \phi\}$. Several combinations from elements of $\mathcal{V}_1$ and $\mathcal{V}_2$ can result in information covers that can cover all targets. One such combination is $v_1^4 \bigoplus v_2^3 = (s_5, s_6) \bigoplus (s_5, s_7) = (s_5, s_6, s_7)$. Such covers are included in the set $\mathcal{C}_t = \{c_t^1, c_t^2, \cdots, c_t^{|\mathcal{C}_t|}\}$ (in Line 20), where $0 \leq |\mathcal{C}_t| \leq |\mathcal{V}_1||\mathcal{V}_2| \cdots |\mathcal{V}_M|$. In Line 25, elements of $\mathcal{C}_t$ are sorted in the increasing order their respective cardinality, i.e., $|c_t^m|, m = 1, 2, \cdots, |\mathcal{C}_t$. Finally, Lines 26 to 28 pick the disjoint covers starting sequentially from the sorted set $\mathcal{C}_t$. That is, we give first preference to those covers with less sensors participating.

For the example in (9) and (10), the final set of disjoint information covers that cover all targets $i.e., T_1$ and T_2 is obtained as $\mathcal{C} = \{(s_1, s_3), (s_2, s_4), (s_5, s_6, s_7)\}$.

	Step-II of the proposed DSIC algorithm
14	***Initialize*** $\mathcal{C} = \mathcal{C}_t = \{\phi\}$
15	**Loop** $a_1 : 1 \leq a_1 \leq |\mathcal{V}_1|,\ \mathcal{V}_1 = \mathcal{U}_1 \biguplus \{\phi\}$
16	**Loop** $a_2 : 1 \leq a_2 \leq |\mathcal{V}_2|,\ \mathcal{V}_2 = \mathcal{U}_2 \biguplus \{\phi\}$
	$\vdots$
17	**Loop** $a_M : 1 \leq a_M \leq |\mathcal{V}_M|,\ \mathcal{V}_M = \mathcal{U}_M \biguplus \{\phi\}$
18	$\mathcal{M}_{a_1 a_2 \cdots a_M} = \{v_1^{a_1} \bigotimes v_2^{a_2} \bigotimes \cdots \bigotimes v_M^{a_M}\}$
19	**if** $\mathcal{M}_{a_1 a_2 \cdots a_M}$ covers all targets in $\mathcal{T}$
20	$\mathcal{C}_t = \mathcal{C}_t \biguplus \mathcal{M}_{a_1 a_2 \cdots a_M}$
21	**end if**
22	**end Loop** a_M
	$\vdots$
23	**end Loop** a_2
24	**end Loop** a_1
25	Sort($\mathcal{C}_t$) for $\uparrow |c_t^m|,\ m = 1, 2, \cdots, |\mathcal{C}_t|$
26	**Loop** $m : 1 \leq m \leq |\mathcal{C}_t|$
27	**if** $s_i \neq s_k$ for $s_i \in c_t^m$ and $s_k \in C_r$, for $i, k = 1, 2, \cdots, N$ and $r = 1, 2, \cdots, |\mathcal{C}|$
28	$\mathcal{C} = \mathcal{C} \biguplus c_t^m$
29	**end if**
30	**end Loop** m

TABLE II
STEP-II OF THE PROPOSED DSIC ALGORITHM

IV. RESULTS AND DISCUSSION

We evaluated the performance of the proposed DSIC algorithm through simulations. Here, we present the simulation results on the network lifetime achieved by the DSIC algorithm. We also compare this lifetime performance with that achieved using EGEH algorithm.

A. Simulation Model

We consider the following simulation model. A network with 5×5 square sensing area is considered. The number of sensor nodes in the network considered include $N = 10, 20, 30$. These sensor nodes are uniformly distributed in the sensing area. The number of targets in the network considered include $M = 1$ to 5. Each sensor is provided with an initial battery energy of $E_0 = 2$ Joules. As in [14], we assume that each sensing operation when a sensor is activated to sense is $4\ nJ$. We also assume that no energy is consumed when the sensor is not activated (i.e., left idle). In all the simulations, we used the following parameter values - α=1, physical coverage range $R = 1$, and ϵ=0.683. The maximum number of nodes allowed to collaborate is taken to be $K_{max} = 1, 2, 3, 4$.

B. Performance Results

In Fig. 5, we illustrate the average network lifetime as a function of K_{max} (the maximum number of sensors allowed to collaborate) obtained from the simulations by averaging over 100 different realizations of the network with 95% confidence interval. The performance of both DSIC as well as EGEH algorithms are shown for $N = 10, 20, 30$. It is observed that both the algorithms perform better than physical coverage (i.e., $K_{max} = 1$) – that is, due to information coverage, lifetime increases as K_{max} is increased. However, this improvement saturates for large values of K_{max} which can be expected (diminishing returns for increased K_{max}). It is also observed that the proposed DSIC algorithm performs better than the EGEH algorithm. The reason for this better performance can be attributed to the fact that EGEH returns covers which are not disjoint whereas as DSIC returns disjoint covers. Since a sensor can participate in more than one cover in EGEH, it can drain the battery power of those sensors participating in multiple covers sooner. This results in the network life to end soon (i.e., covers become invalid due to lack of battery power in sensors participating in multiple covers) even if some other sensors may have power left in their batteries. Whereas in the proposed DSIC, since the covers are disjoint, the energy in the sensors are fully used, resulting in longer network lifetime. Also, a simple round-robin scheduling is adequate for DSIC whereas a more elaborate scheduling is required for EGEH [12].

We illustrate the above observations with an example using Figs. 6 and 7. Consider the network with nodes and a target as shown in Figs. 6 and 7. For this scenario, DSIC will return the following four information covers $\{s_1, s_2, (s_3, s_4), (s_5, s_6)\}$ (see Fig. 6(a)). Note that all the above covers are disjoint. Whereas EGEH will return the following five information covers $\{s_1, s_2, (s_3, s_4), (s_5, s_6), (s_4, s_6)\}$ (see Fig. 6(b)). Although EGEH 'greedily' returns more number of covers than DSIC, as can be seen, these covers are not disjoint (i.e., sensors s_4 and s_6 in EGEH are present in two covers). Further, the 'equalization' part in Step-3 of EGEH, in order to make the appearances (i.e., activation) of all sensors in the covers to be almost equal in a scheduling cycle, results in a more involved activity schedule as shown in Fig. 7(b). In the EGEH schedule in Fig. 7(b), sensors s_1 and s_2 are activated twice in a scheduling cycle in order to almost equalize the number of activations. Also, the network lifetime achieved in this EGEH schedule can be computed to be 3.5 (using the formulation in [12], which requires the LCM of certain weights of the covers which are proportional to the number of activations of the sensors in those covers). On the other hand, the covers returned by DSIC, being disjoint, results in a simple round-robin schedule as shown in Fig. 7(a) is adequate. In addition, the network lifetime achieved here is 4. Thus, by preferring to $i)$ use disjoint covers, and $ii)$ use far-away nodes (e.g., in Fig. 6(a), DSIC did not give preference to the nearby cover (s_4, s_6)), DSIC achieves better performance than EGEH.

In Fig. 8, we illustrate the average network lifetime as a function of M, the number of targets in the network, for $N = 20, 30$ and $K_{max} = 4$. As the number of targets is increased, lifetime decreases, which is expected. Here again, DSIC is seen to achieve longer network lifetime than EGEH.

C. Computational Complexity

In this subsection, we present a comparison of the computational complexity of DSIC and EGEH algorithms. A worst case analysis of the complexity for these two algorithms are presented in Appendix. Equations (16) and (21) in Appendix I give the worst case approximate complexity for DSIC and EGEH algorithms, respectively. From Eqns. (16) and (21), we

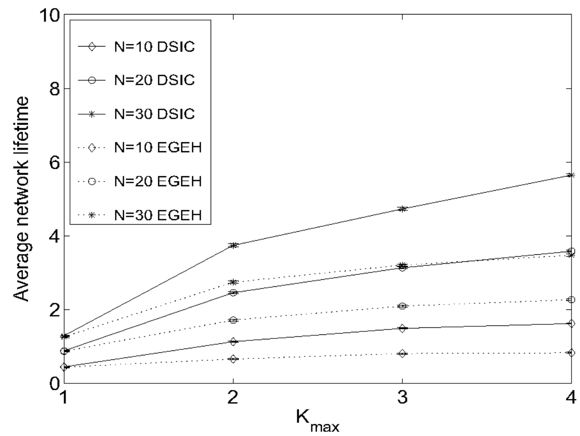
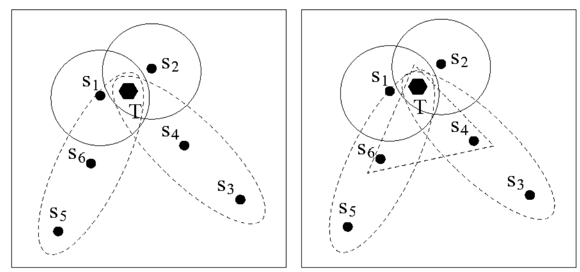


Fig. 5. Average network lifetime versus K_{max} for $N = 10, 20, 30$, $M = 1$, $\alpha = 1$, $\epsilon = 0.683$, $R = 1$, $E_0 = 2\ J$



(a) DSIC (b) EGEH

Physical covers

Information covers

Fig. 6. Illustration of information covers chosen by DSIC and EGEH.

can write

$$\frac{H_{DSIC}}{H_{EGEH}} \approx \frac{\left[2N^{K_{max}} M K_{max}^2 + N^{M+1}\right] \log N}{K_{max}(M+2) N^{2K_{max}}}. \quad (11)$$

From the above equation, it can be seen that

- for $M + 1 > 2K_{max}$
$$\frac{H_{DSIC}}{H_{EGEH}} > 1,$$
- for $M + 1 < 2K_{max}$
$$\frac{H_{DSIC}}{H_{EGEH}} < 1,$$
- and for $M + 1 = 2K_{max}$
$$\frac{H_{DSIC}}{H_{EGEH}} \approx \frac{\log N}{K_{max}(M+2)}.$$

From the above, it can be seen that the complexities of the DSIC algorithm can be less or more than the EGEH algorithm depending on the values of N, M, and K_{max}.

The observation made through an approximate worst case analysis in the above is also reflected in the complexities of the algorithms evaluated in the actual simulations. Figures 9 and 10 show the complexity of the algorithms obtained from the simulations by averaging over 100 different network realizations. Figure 9 shows the complexity as a function of K_{max} for 1 target (i.e., $M = 1$) and $N = 10, 20, 30$, whereas

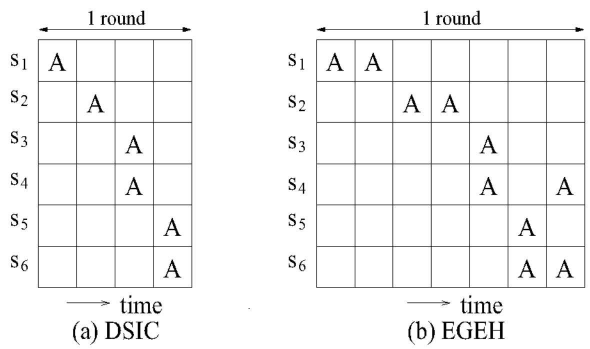


Fig. 7. Illustration of scheduling of sensor activity in DSIC and EGEH.

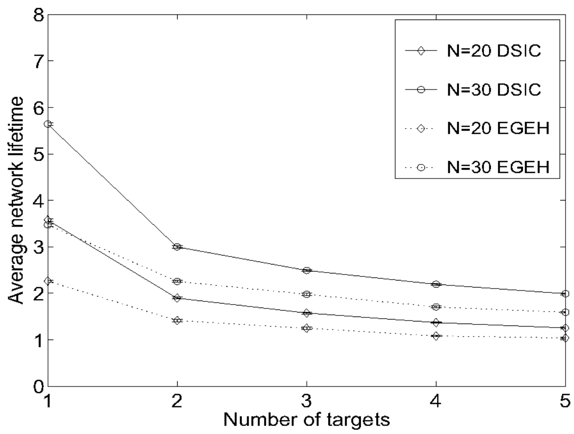


Fig. 8. Average network lifetime versus number of targets, M, to monitor. $N = 30, 20$, $K_{max} = 4$, $\alpha = 1$, $\epsilon = 0.683$, $R = 1$, $E_0 = 2$ J.

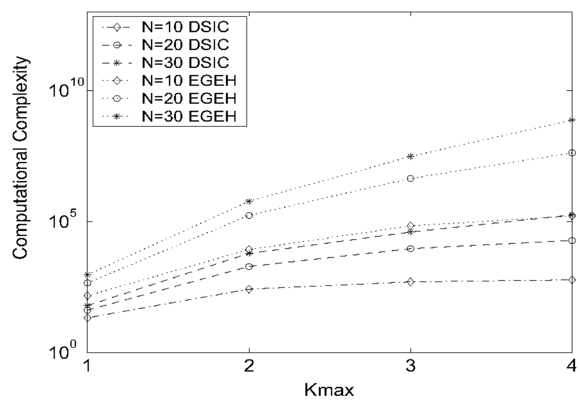


Fig. 9. Computational complexity versus K_{max} obtained from simulations. $N = 10, 20, 30$, number of targets, $M = 1$.

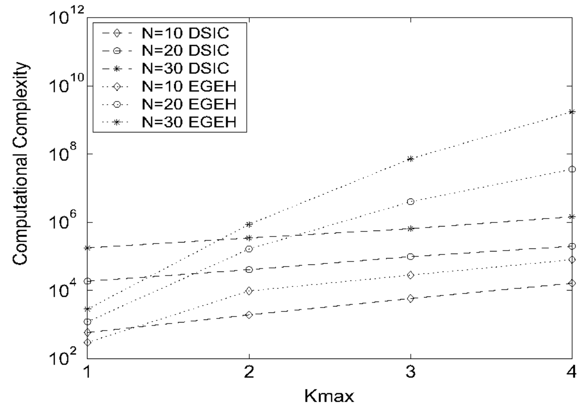


Fig. 10. Computational complexity versus K_{max} obtained from simulations. $N = 10, 20, 30$, number of targets, $M = 4$.

Fig. 10 is for 4 targets (i.e., $M = 4$). From these figures, it can be observed that the complexity of one algorithm is more or less than the other depending on the values of M (the number of targets), K_{max} (the maximum number of sensors allowed to collaborate), and N (the number of sensors). It is noted that the cross-overs between the resulting complexities of DSIC and EGEH algorithms seen in Figs. 9 and 10 essentially corroborate with a similar observation we made earlier based on the approximate worst case complexity analysis.

V. Conclusion

Disjoint set cover problem for scheduling the sensing activity of various sensors in wireless sensor networks have been studied by others in the literature for 'physical coverage'. But, to our knowledge, disjoint set covers have not been investigated for 'information coverage'. Our new contribution in this paper was that we proposed an algorithm to obtain disjoint set of information covers (DSIC) for energy efficient scheduling of sensors using information coverage. It was shown that the network lifetime is increased as the number of collaborating nodes for sensing is increased. We showed that the proposed DSIC algorithm achieved longer network lifetimes compared to an EGEH algorithm, that has been recently proposed in the literature. Presenting a detailed computational complexity comparison between the proposed DSIC and EGEH algorithms, we showed that, there is a cross-over between the complexities of these two algorithms depending on the number of targets to monitor, the maximum number of sensors allowed to collaborate, and the number of sensors in the network.

References

[1] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: A survey", *Computer Networks, Elsevier Publishers*, Vol. 39, no 4, pp. 393-422, 2002.

[2] I. F. Akyildiz and I. H. Kasimoglu, "Wireless sensor and actor networks: Research challenges," *Ad-hoc Networks Journal (Elsevier)*, vol. 2, pp. 351-367, October 2004.

[3] R. C. Shah and J. M. Rabaey, "Energy aware routing for low energy ad-hoc sensor networks" *Proc. IEEE WCNC'2002*, pp. 350-355, March 2002.

[4] B. Wang, W. Wang, V. Srinivasan, and K. C. Chua, "Information coverage for wireless sensor networks," *IEEE Commun. Letters*, vol. 9, no.11, pp. 967-969, November 2005.

[5] M. T. Thai, F. Wang, and D.-Z. Du, "Coverage problems in wireless sensor networks: Designs and analysis," *Intl. Jl. on Sensor Networks, Special Issue on Coverage Problems in sensor networks*, 2005.

[6] D. Tian and N. Georganas, "A coverage-preserving node scheduling scheme for large wireless sensor networks", *Proc. of the 1st ACM Workshop on Wireless Sensor Networks and Applications*, pp. 32-41, 2002.

[7] M. Cardei and D.-Z. Du, "Improving wireless sensor network lifetime through power aware organization," *Wireless Networks*, vol. 11, no. 3, pp. 333-340, May 2005.
[8] S. Slijepcevic and M. Potkonjak, "Power efficient organization of wireless sensor networks," *Proc. IEEE ICC'2001*, vol. 2, pp. 472-476, June 2001.
[9] Z. Abrams, A. Goel, and S. Plotkin, "Set K-cover algorithms for energy efficiency monitoring in wireless sensor networks," *Proc. 3rd Intl. Symp. on Information Processing in Sensor Networks (IPSN'04)*, pp. 424-432, 2004.
[10] M. Cardei, M. T. Thai, Y. Li, Y. Li, and W. Wu, "Energy-efficient target coverage in wireless sensor networks," *Proc. IEEE INFOCOM'2005*, pp. 1976-1984, 2005.
[11] S. Yang, F. Dai, M. Cardei, and J. Wu, "On multiple point coverage in wireless sensor networks," *Proc. 2nd IEEE Intl. Conf. on Mobile Ad-Hoc and Sensor Systems (MASS'05)*, 2005.
[12] B. Wang, W. Wang, V. Srinivasan, and K. C. Chua, "Scheduling sensor activity for point information coverage in wireless sensor networks," Proc. WiOpt'2006, April 2006.
[13] J. M. Mandel, *Lessons in Estimation Theory for Signal Processing, Communications and Control*, Prentice Hall, 1995.
[14] L. Doherty, B. A. Warneke, B. E. Boser, and K. S. J. Pister, "Energy and performance considerations for smart dust," *Intl. Jl. of Parallel and Distributed Systems and Networks*, vol. 4, no. 3, pp. 121-133, 2001.

VI. APPENDIX

A. Complexity of the Proposed DSIC Algorithm

From the operations involved in Step-I of the DSIC algorithm given in Table I, the worst case computational complexity in Step-I can be obtained as

$$H_{S1,dsic} \leq M \sum_{K=1}^{K_{max}} \left[\binom{N}{K} + \binom{N}{K} \log \binom{N}{K} + \binom{N}{K} K \log N \right]. \quad (12)$$

Likewise, from the operations in Table II, the worst case complexity of in Step-II can be obtained as

$$H_{S2,dsic} \leq (N+1)^M + (N+1)^M \log(N+1)^M + (N+1)^M N \log N. \quad (13)$$

Combining (12) and (13), the worst case complexity of the proposed DSIC algorithm can be written as

$$H_{DSIC} \leq H_{S1,dsic} + H_{S2,dsic}. \quad (14)$$

By bounding $\sum_{K=1}^{K_{max}} \binom{N}{K}$ in (12) by $K_{max}\binom{N}{K}$, and using a simplifying approximation of $\binom{N}{K} \approx N^K$, which is a good approximation for $N > K/2$, Eqn. (12) and hence Eqn. (14) can be further simplified as

$$H_{DSIC} \approx N^{K_{max}} M K_{max} \Big[1 + 2K_{max} \log N \Big] + (N+1)^M \Big[1 + M \log(N+1) + N \log N \Big]. \quad (15)$$

For $M << N$ (i.e., number of targets much smaller than number of sensors), the above equation can be further approximated as

$$H_{DSIC} \approx 2N^{K_{max}} M K_{max}^2 \log N + N^M (M+N) \log N \approx \Big[2N^{K_{max}} M K_{max}^2 + N^{(M+1)} \Big] \log N. \quad (16)$$

B. Complexity of the EGEH Algorithm in [12]

Based on the operations in Steps 1, 2, and 3 in the EGEH algorithm (refer [12] for these steps), the worst case complexities in these three steps can be obtained as

$$H_{S1,egeh} \leq M \sum_{K=1}^{K_{max}} \left[\binom{N}{K} + \binom{N}{K} K \log \binom{N}{K} \right] + M \binom{N}{K_{max}} K_{max} \log \binom{N}{K_{max}}, \quad (17)$$

$$H_{S2,egeh} \leq \binom{N}{K_{max}} \left[M \binom{N}{K_{max}} + N \log \binom{N}{K_{max}} + \binom{N}{K_{max}} \right], \quad (18)$$

and

$$H_{S3,egeh} \leq \binom{N}{K_{max}}^2. \quad (19)$$

Combining Eqns. (17), (18), and (19), we can write the worst case complexity of the EGEH algorithm as

$$H_{EGEH} \leq H_{S1,egeh} + H_{S2,egeh} + H_{S3,egeh}. \quad (20)$$

Using the approximations used to simplify the DSIC complexity expression, we can simplify Eqns. (17), (18) and (19) to obtain

$$H_{EGEH} \approx N^{K_{max}} K_{max} \Big[(K_{max}^2 M + M K_{max} + N K_{max}) \log N + (M+2) N^{K_{max}} \Big] \approx K_{max} (M+2) N^{2K_{max}}. \quad (21)$$