

Maximizing α -Lifetime for Wireless Sensor Networks

Honghai Zhang and Jennifer C. Hou
Department of Computer Science
University of Illinois at Urbana-Champaign
Email: {hzhang3,jhou}@cs.uiuc.edu

Abstract

In energy constrained wireless sensor networks, it is very important to conserve energy and prolong network lifetime while ensuring proper operations of the network. In this paper, we investigate how to maximize the α -lifetime of wireless sensor networks, where α -lifetime is defined as the time duration during which at least α portion of the surveillance region is covered. We first present, given node locations, two upper bounds of the α -lifetime. We then design, based on the derived upper bound, an algorithm that sub-optimally schedules node activities to maximize the α -lifetime of a sensor network. We carry out simulations to validate the derived results and evaluate the designed algorithm. Simulation results show that the proposed algorithm achieves around 90% of the derived upper bound. This implies that the derived upper bounds are rather tight and the proposed algorithm is close to optimal. Finally, we draw from our study several useful conclusions on sensor network deployment and design.

I. INTRODUCTION

Due to the advances in MEMS and Nanotechnology [1], and the wide applications in commercial and military applications [2], [7], [11], [12], wireless sensor networks have received tremendous attentions in recent years. They are composed of a large number of minuscule devices equipped with one or more sensors, a processing unit and a radio transceiver. Sensor nodes in such a network are often powered with onboard batteries with limited energy. It is impractical or infeasible to replenish energy via replacing batteries on these sensors in most applications. As a result, it is well perceived that a sensor network should be deployed with high density in order to prolong the network lifetime.

In a high-density network with energy-constrained sensors, it is important to have only a subset of nodes operate in the active mode at a time. To fulfill the objective of deploying sensor networks for monitoring/surveillance, it is often required that the subset of simultaneously active nodes completely cover the entire surveillance region. Slijepcevic and Potkonjak [13] studied the problem of finding the maximum number of (disjoint) subsets such that each subset can completely cover the whole region. They proved the NP-completeness of the problem and provided a heuristic algorithm.

There are, however, applications that do not require a 100% coverage of the surveillance region. For example, in the environmental science, it is often tolerable that only a small, confined area is polluted but not detected. In this paper, we study a problem that is similar to that in [13] but in a more general context, i.e., at any time, only α portion of the region is required to be covered, where $0 < \alpha < 1$. Specifically, we study the problem of finding a maximum number of disjoint subsets such that nodes in each subset can cover at least α portion of the region. In the case that the time is normalized to be the lifetime of each sensor node, the number of such subsets can be termed as the (α -)lifetime of the sensor network, as these disjoint subsets can take turn to monitor the surveillance area.

We first present two upper bounds of sensor network lifetime: one is for algorithms that aim to maintain as maximum coverage as possible until the coverage ratio drops below α , and the other holds universally for any possible algorithms. With availability of these two bounds, we are able to determine, given the lifetime T of a single sensor node, how many sensor nodes have to be deployed in a region, in order to continuously monitor the region for a period of $k \cdot T$, or conversely, what is the maximum achievable lifetime for a deployment of n nodes.

We then devise an algorithm to (sub)-optimally schedule the wakeup and sleep schedule of each node to achieve a lifetime that is close to the (universal) upper bound. The algorithm makes use of the derived lifetime upper bounds and chooses a subset of nodes to be active in each round, with the objective of maximizing the universal lifetime upper bound of the remaining nodes that have not been scheduled to be active. Finally, we carry out simulation to compare the derived lifetime upper bounds, the actual lifetime achieved using the devised algorithm, and the lifetime achieved using a baseline algorithm where in each round a minimum number of nodes are chosen to meet the coverage requirement. The simulation results show that the devised algorithm can achieve nearly (and sometimes over) 90% of the universal lifetime upper bound in most cases, and it outperforms the baseline algorithm by 15-20%. These results imply that i) the derived universal lifetime upper bound is tight; and ii) the

devised algorithm is effective in determining the achievable lifetime.

The significance of our study is that it shows that to improve the sensor network lifetime per unit of node density, one can either i) deploy sensor nodes with high density (as practically allowed by a distributed algorithm); or ii) allow a small portion of the area to be uncovered. For the second strategy, an algorithm should start with providing α -coverage from the beginning, where α is the percentage of the area that can be uncovered. Our study also suggests that choosing the minimum number of nodes to maintain coverage in each round is not always optimal with respect to maximizing the network lifetime. In general, it is better to choose a subset of nodes that maximizes the lifetime upper bound of the remaining nodes.

The rest of the paper is organized as follows. In the next section, we summarize related work. In Sections III–Section IV, we first present the lifetime upper bounds, and then devise a sub-optimal algorithm that schedules sensor activities to achieve a lifetime that is close to the derived (universal) upper bound. Following that, we carry out simulations in Section V to compare the derived lifetime upper bounds and the actual lifetime achieved by the proposed algorithm. Finally we conclude the paper in Section VI.

II. RELATED WORK

Several researchers [13], [16], [17], [14], [21], [8], [15], [9], [10] have addressed various methods of minimizing energy consumption and prolonging network lifetime while maintaining coverage in wireless sensor networks. Slijepcevic and Potkonjak [13] devise a heuristic algorithm to find the maximum number of disjoint covers where a *cover* is defined as a subset of nodes which can completely cover the whole surveillance region. Their heuristic tries to cover fields that are covered by a small number of sensors and tries to avoid excessive use of those sensors which cover sparsely covered fields. Ye *et al.* [16], [17] present PEAS, a distributed, probing-based density control algorithm for robust sensing coverage. In this work, a subset of nodes operate in the active mode to maintain coverage while others are put into sleep. It ensures no two active nodes are in the proximity of each other but does not preserve complete coverage. Tian *et al.* [14] devise an algorithm that ensures complete coverage using the concept of “sponsored area.” Whenever a sensor node receives a packet from one of its working neighbors, it calculates its sponsored area (defined as the maximal sector covered by the neighbor). If the union of all the sponsored areas of a sensor node covers the coverage disk of the node, the node turns itself off. Zhang and Hou [19] analyze the relationship between complete coverage and connectivity (i.e., if the transmission radio range is at least twice of the sensing range, then coverage implies connectivity), develop several optimal conditions of maintaining coverage, and devise, based on the optimal conditions, a localized method to maintain coverage

and connectivity. Wang *et al.* [15] analyze the relationship between k -coverage and j -connectivity, prove a sufficient condition for satisfying k -coverage and propose, based on the sufficient condition, an algorithm (combined with SPAN [6]) to maintaining coverage and connectivity. Gupta *et al.* [8] devise both a centralized and a distributed algorithm to find a subset of nodes that ensure both coverage and connectivity. The centralized algorithm guarantees that the size of the formed subset is within $O(\log n)$ factor of the size of the optimal node set, where n is the network size. Huang and Tseng [9], [10] design efficient algorithms to determine whether a region is completely k -covered and suggest the algorithms can be used to maintain coverage.

Although all the above methods are targeted for prolonging the network lifetime, most of them do not perform serious analysis on the network lifetime and how close to the optimal operations those algorithms are. Recently, research efforts have also been made to analyze the upper bound of the lifetime for ad hoc/sensor networks [4], [3], [5], [18], [20]. Bhardwaj *et al.* [4], [3] study the upper bound of the lifetime of data gathering sensor networks with the assumption that sensor nodes only consume energy when they process, send or receive data. Blough and Santi [5] study the upper bound of the network lifetime for cell-based energy conservation techniques. Zhang and Hou [18] derive the lifetime upper bound for sensor networks that maintain coverage. However, it is not sufficient to derive the lifetime upper bound without studying whether, or to what extent, the derived lifetime upper bound can be achieved. The work reported in this paper is built upon [18], [20], and bridges this gap by devising an algorithm that (sub-)optimally schedules the wakeup and sleep schedule of each node to achieve a lifetime that is close to the (universal) upper bound.

III. UPPER BOUNDS OF THE LIFETIME

We denote the surveillance region as R and the area of R as $||R||$. We define the α -lifetime as the interval during which at least α portion of the region R is covered by at least one sensor node, where $0 < \alpha < 1$ is a tunable parameter and usually is close to 1. We assume each sensor has a maximum lifetime T and each sensor can cover a fixed region around itself. Without loss of generality, we assume $T = 1$.

A. Upper Bound of α -Lifetime for a Special Family of Algorithms

In this subsection we consider a special class of algorithms that aim to maintain as maximum coverage as possible until the coverage ratio drops below a certain threshold α . We present an upper bound of α -lifetime for these algorithms. The reason why this type of upper bounds is of interested is two fold. First, most existing distributed algorithms [14], [21], [8], [15] operate in this fashion. Second, results in this subsection serve a base for deriving the universal lifetime upper bound for all algorithms.

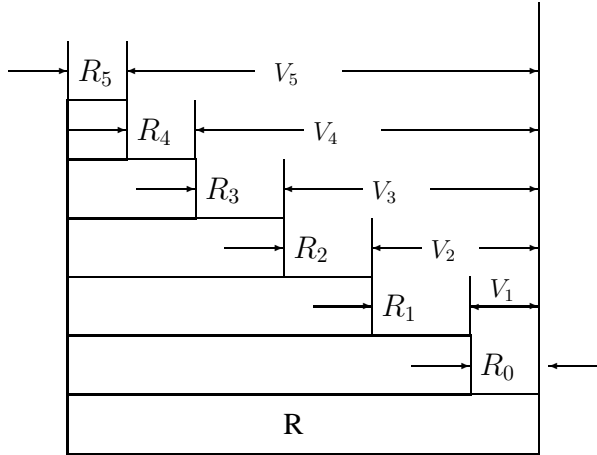


Fig. 1. The entire region R can be divided into different sub-regions: R_0, R_1, \dots, R_n , where all points in R_i are exactly covered by i nodes.

We divide the entire region R into sub-regions R_0, R_1, \dots, R_n , where all points in R_i are exactly covered by i sensor nodes (Fig. 1). Thus $V_k = \sum_{i=0}^{k-1} \|R_i\|$ and $1 - V_k/\|R\|$ is the portion of the region in which each point is covered by at least k nodes. We summarize the following results we obtained in [20].

Theorem 1: The α -lifetime is upper bounded by

$$L(\lambda, \alpha) = \max\{k : \alpha \leq 1 - V_k/\|R\|\}. \quad (1)$$

As V_k depends on the node distribution, its value cannot be accurately calculated unless the locations of all the nodes are known. However, if we assume nodes are distributed as a Poisson point process with density λ , and each node can cover a unit-area disk, we can calculate the expectation of V_k as (without considering the boundary effects):

$$E[V_k] = \|R\| \cdot e^{-\lambda} \sum_{i=0}^{k-1} \frac{\lambda^i}{i!}. \quad (2)$$

In addition, the following theorem [20] shows that it suffices to approximate V_k with $E(V_k)$ in large regions.

Theorem 2: Assume nodes are distributed according to a Poisson point process with density λ and each node can cover a unit-area disk. Let R be a square with side length ℓ . As $\ell \rightarrow +\infty$, $V_k/\ell^2 \rightarrow e^{-\lambda} \sum_{i=0}^{k-1} \frac{\lambda^i}{i!}$ almost surely.

B. Upper Bound of α -Lifetime for All Algorithms

We now consider algorithms which maintain α -coverage from the very beginning and present the upper bound of the network lifetime in this case. Interested readers are referred to [20] for a detailed account of proof.

Theorem 3: Let $\gamma_i \triangleq 1 - V_i/\|R\|$ and $\beta_i \triangleq \gamma_i - \gamma_{i+1} = \|R_i\|/\|R\|$. Then the upper bound of α -lifetime for a sensor network with density λ is

$$\min_{k:\alpha>\gamma_k} H(k, \alpha) \triangleq \frac{\sum_{i=1}^{k-1} i\beta_i}{\alpha - \gamma_k}, \quad (3)$$

Moreover, the following lemma contains several nice properties of $H(k, \alpha)$.

Lemma 1: For all k such that $\alpha > \gamma_k$, $H(k, \alpha)$ given in Eq. (3) has the following properties:

- (i) If $H(k, \alpha) > k$, $H(k, \alpha)$ monotonically decreases as k increases;
- (ii) If $H(k, \alpha) < k$, $H(k, \alpha)$ monotonically increases as k increases;
- (iii) If $H(k, \alpha) = k$, then $H(k, \alpha) = H(k+1, \alpha)$;
- (iv) If $H(k, \alpha) > k$, then $H(k+1, \alpha) > k$;
- (v) If $H(k, \alpha) = k$, then $H(k+1, \alpha) = k$;
- (vi) If $H(k, \alpha) < k$, then $H(k+1, \alpha) < k$.

Numerical examples: Figure 2 gives the upper bound of the lifetime derived in Section III-A and that in this subsection (we have used $E[V_k]$ in Eq. (2) to approximate V_k), and their respective lifetime per unit of density. As compared with the upper bound of the lifetime derived in Section III-A, the “universal” upper bound of the lifetime increases by 15% for 99%-coverage and over 20% for 95%-coverage. The upper bound of the lifetime per unit density increases as the density increases in general, and slightly decreases at certain density values for the special class of algorithms. This is because the upper bound of the lifetime for the special class of algorithms does not change for the slight increase in the node density λ at some points. It is not surprising to observe that the universal lifetime upper bound per unit density can be more than 1 in some cases, because less than 100%-coverage is required in each round.

Remarks on fractional lifetime: Unless it is required the lifetime have to take integer values, it is possible to achieve fractional lifetime. As an example in Fig. 3, let the surveillance region be the square with unit area and there be three sensors covering the region. Let a, b, c be the area covered by any single node, any two nodes, and all three nodes respectively. If we let the coverage requirement $\alpha = 2a + b + c$, then $\beta_1 = 3a, \beta_2 = 3b, \gamma_3 = c$, and the lifetime is upper bounded by

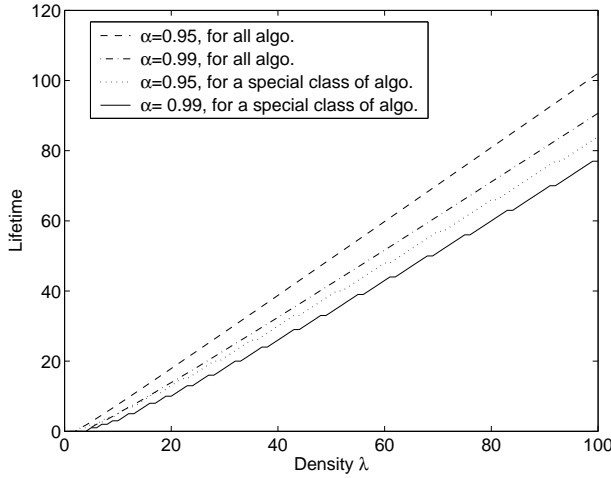
$$\frac{\beta_1 + 2\beta_2}{\alpha - \gamma_3} = \frac{3a + 6b}{2a + b}. \quad (4)$$

Fix any $c \geq 0$, let $b \rightarrow 0$, the lifetime upper bound can be made arbitrarily close to (but above) 1.5. Now if we schedule the node 1 and 2 working during the first half time unit, node 1, 3 working during the second half time unit, and node 2,3 working during the third half time unit, we can achieve an α -lifetime of 1.5.

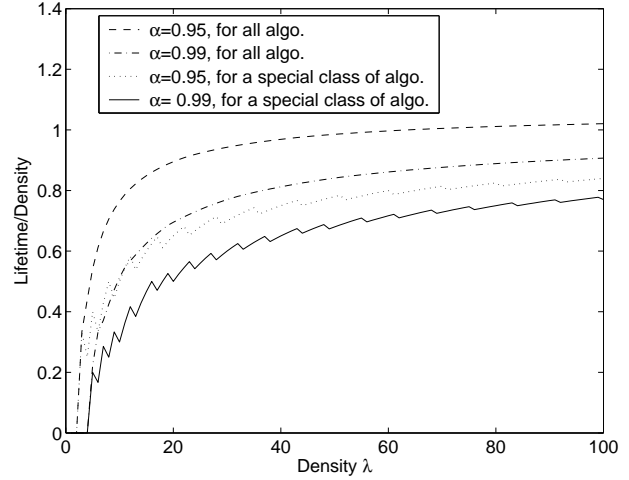
Although in general it is possible to obtain larger α -lifetime, the lifetime improvement is normally less than 1. In a high density network, the improvement is marginal.

IV. AN ALGORITHM FOR MAXIMIZING α -LIFETIME

In this section we devise a centralized algorithm that maximizes α -lifetime. The algorithm is designed to maintain



(a) Average α -lifetime v.s. density



(b) Average α -lifetime per unit of density v.s. the density

Fig. 2. Average α -lifetime and α -lifetime per unit of density derived in Section III-A and in Section III-B.

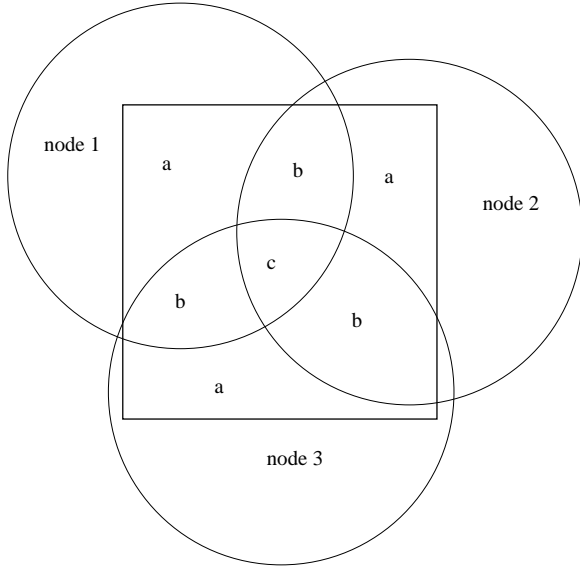


Fig. 3. Illustration of fractional lifetime

α -coverage from the very beginning. There are two purposes for devising a centralized algorithm. First, it can be used to investigate whether, or to what extent, the derived lifetime upper bound in Section III-B can be achieved under practical assumptions. Note that the tightness proof in Section III-B is based on the assumption that the coverage region of a node can be arbitrarily selected to provide α -coverage. In practice, at any time, a node's coverage region is either completely selected or completely unselected for providing α -coverage. Under this practical condition, in general the

derived lifetime upper bound cannot be achieved exactly. A centralized algorithm can be used to investigate to what extent the lifetime upper bound can be achieved. Second, a good centralized algorithm provides a practical upper bound of the lifetime that can be possibly achieved by any distributed/localized algorithm, thus serving as a baseline of performance comparison for distributed algorithms.

To facilitate the description of the algorithm, we first define several notations:

Definition 1: An α -cover is a set of nodes which can cover at least α portion of the monitored region.

Definition 2: A minimal α -cover is an α -cover from which removing any node will lead to a coverage ratio less than α .

Definition 3: A trim operation is an operation of reducing a α -cover to a minimal α -cover by iteratively visiting each node in the α -cover and removing the node if after removing the node, the remaining nodes can still provide α -coverage.

The basic idea of the algorithm is that in each round, it tries to find a minimal α -cover which maximizes the α -lifetime upper bound of the remaining set of nodes (i.e., those which have not been selected in earlier rounds and the current round). The maximization is performed through a global, greedy search, while jumping out local minima.

The pseudo-code of the algorithm is listed in Figure 4-6. The algorithm continuously searches for minimal α -covers until the remaining set of nodes cannot provide α -coverage. In each round (line 4-6 in Figure 4), the algorithm seeks a minimal α -cover which maximizes the α -lifetime upper bound of the remaining set of nodes. The search in each round is composed of two steps. In the first step (Figure 5), it finds a minimal α -cover, which is obtained by first iteratively selecting the node that maximizes the coverage improvement

```

α-Lifetime Algorithm
1. the remaining set of nodes = the set of all nodes,
2. lifetime = 0,
3. while (the remaining set of nodes have α coverage)
4.   find an initial minimal α-cover C,
5.   search for a minimal α-cover which maximizes the
      lifetime upper bound of the remaining nodes,
6.   lifetime += 1,
7. endwhile

```

Fig. 4. α-Lifetime algorithm

of the current cover divided by the reduction of the lifetime upper bound of the remaining set of nodes (line 2-5 in Figure 5) and then performing a trim operation (line 6 in Figure 5).

In the second step (Figure 6), a global search is performed to further improve the lifetime upper bound of the remaining set of nodes. In each iteration of the search, a node (sequentially chosen) in the remaining set is added to the found minimal α-cover (line 5 in Figure 6) and is followed by a trim operation (line 6 in Figure 6). If the new minimal α-cover leads to a larger lifetime upper bound, the new minimal α-cover is kept as the starting point for the next iteration; otherwise the original minimal α-cover is kept (line 7-12 in Figure 6). The global search finishes when adding to the minimal α-cover any node in the remaining set of nodes followed with a trim operation does not lead to any improvement on the lifetime upper bound of the remaining nodes. The lifetime is then the number of α-covers constructed.

Baseline Algorithm: As a baseline algorithm for comparison, we have also implemented an algorithm that performs the same search for minimal α-covers except that the objective in each round of the algorithm is to find a minimal α-cover that minimizes the number of nodes in the α-cover.

V. SIMULATION STUDY

In this section, we carry out several sets of simulations to validate the theoretical lifetime upper bound and to investigate to what extent the upper bound can be achieved by the algorithm we have developed. In addition, we are

```

Finding an initial minimal α-cover C
1. C = ∅, coverage = 0,
2. while (coverage < α)
3.   Find the node in the remaining set of nodes that maximizes
      the coverage improvement of C divided by the reduction
      of the lifetime upper bound of the remaining nodes
      due to moving the node from the remaining set to C,
4.   move the node to C and update the coverage of C,
5. endwhile
6. trim the cover C,
7. return C.

```

Fig. 5. Procedure for finding an initial minimal α-cover

```

/* Search for a minimal α-cover which maximizes the
lifetime upper bound of the remaining nodes */
GlobalSearch(C)
1. improved = true,
2. while (improved)
3.   improved = false,
4.   for (each node in the remaining set)
5.     move the node to the α-cover C
6.     trim the cover C,
7.     if (there is improvement on the lifetime
          upper bound of the remaining set)
8.       keep the current α-cover C,
9.       improved = true,
10.    else
11.      revert to the original α-cover C.
12.    endif
13.  endfor
14. endwhile

```

Fig. 6. Global search for a minimal α-cover that maximizes lifetime upper bound of the remaining nodes.

most interested in the lifetime achievable using algorithms that maintain α-coverage from the very beginning, although we will also compare it with the lifetime upper bound for algorithms that maintain as large coverage as possible.

A. Simulation Methodology

We randomly place N nodes in a square region R with 100×100 pixels. The positions of the nodes are independently and uniformly distributed in the square region. We assume that each node has a sensing range of r . For each pixel we count the number of disks that cover it. For each randomly generated instance, we calculate the k -vacancy V_k by counting the number of pixels that are not covered by at least k nodes. With the numerical value of V_k , we compute the upper bound of the lifetime for the special class of algorithms using Eq. (1), and that for all algorithms using Eq. (3). The lifetime of a single sensor T is set to 1. To accommodate the effect of sensing radius, the (normalized) network density is evaluated as $\frac{N\pi r^2}{10000}$. All the results reported below are averages of 10 simulation runs.

Note that decreasing the disk radius with the side length of the square area fixed has the same effect of increasing the side length of the square area with the disk radius fixed. For each value of α we vary the disk radii over different runs (but keep the radii of all disks fixed in each run) to investigate how the area size of the region affects the upper bound of the lifetime. For each value of α and disk radius, we vary the number of sensors to change the node density. The performance metrics we consider are the actual lifetime achieved and the ratio of the achieve lifetime to its corresponding upper bound.

B. Simulation Results

Figure 7 compares various lifetime upper bounds and the actual lifetime achieved using our algorithm and that using

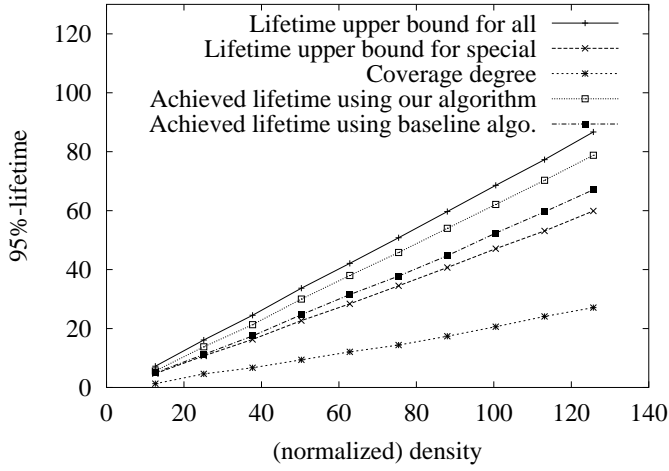


Fig. 7. Average 95%-lifetime upper bounds and the achieved 95%-lifetime.

the baseline algorithm. In this simulation, we set $\alpha = 0.95$ and the radius $r=20$. We define the *critical point* as the point in the surveillance region that is covered by the least number of sensors and the *coverage degree* as the number of sensors which cover the critical point. The coverage degree is an upper bound of lifetime for complete coverage.

Several observations are in order. First, allowing a small portion of the region to be uncovered can lead to significant improvement on the system lifetime upper bound and the actual lifetime achieved. Even for the special class of algorithms that provide as much coverage as possible, the 95%-lifetime upper bound can improve by more than 100%. Second, the α -lifetime upper bound is significantly larger in the case that α -coverage is maintained from the very beginning (than in the case that as maximum coverage as possible is maintained until the coverage ratio drops below a certain threshold α). For $\alpha=95\%$, the improvement is usually more than 50%. Third, the actual lifetime achieved by the proposed algorithm is much larger than that by an algorithm that maintain as maximum coverage as possible at the beginning. Although we have not devised the latter algorithm, we can make this conclusion by observing that the former is much larger (usually by 30-40%) than the upper bound of the latter. This suggests that it is desirable to maintain α -coverage from the beginning of system operations, as long as this is acceptable to applications. Fourth, although it is in general a good choice to select a minimum subset of nodes to maintain α -coverage, this does not always lead to maximum lifetime. As a matter of fact, the proposed algorithm (which maximizes the lifetime upper bound of the remaining nodes) achieves over 20% longer lifetime than the baseline algorithm which minimizes the number of nodes used in each round. All these performance trends have been observed for different combinations of sensing radii and α values.

Note that the lifetime upper bound depicted in Figure 7

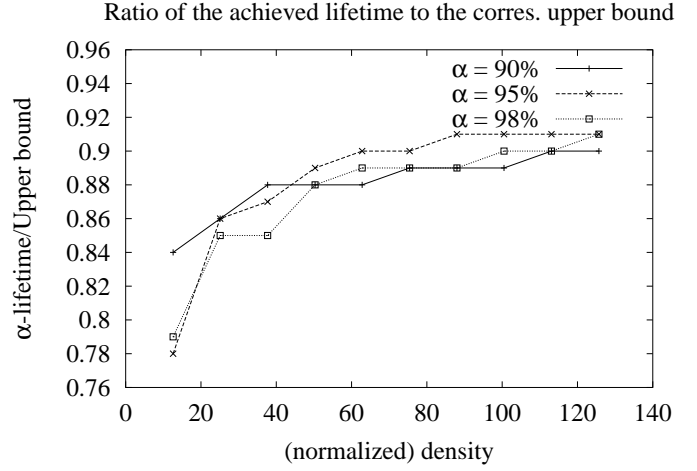


Fig. 8. Average ratios of the achieved α -lifetime to the corresponding upper bound.

is smaller than that in Figure 2. This is because in Figure 2, we used $E(V_k)$ to replace V_k , which usually increases the lifetime upper bound. Also, we take into consideration of the boundary effect in Figure 7 — if some portion of the coverage area of a node falls outside of the monitored region, it does not contribute to the lifetime upper bound.

Figure 8 shows the average ratio of the actual lifetime achieved using our algorithm to the corresponding upper bound for three different values of α : 90%, 95% and 98%. The ratio increases as the node density increases, but levels off when the node density becomes sufficiently large. This is because when the density is too small, there is little room for optimizing the coverage usage. When the density is very high, the lifetime is limited by the upper bound and the practical physical constraint (i.e. a node's coverage area has to be either completely used or completely not used). This is another argument for deploying sensor nodes with a relatively high density. On the other hand, little difference has been observed in the average ratio with different values of α .

Figure 9 shows the actual 95%-lifetime achieved and the corresponding upper bound for three different values of sensing radii: 10, 20, and 40. As the radii increase, both the actual lifetime achieved and its corresponding upper bound decrease. This is because as the radii increase, there is a higher chance that some portion of the coverage area of a node falls out of the monitored region, and thus the effective coverage areas given by all the nodes decrease. Figure 10 shows the average ratio of the achieved lifetime using the proposed algorithm to the corresponding upper bound for different values of sensing radii. It is a little surprising that the ratio decreases if the sensing radius is very large or very small. When the sensing radius is very large, the room for optimizing the coverage usage is small, leading to a low ratio of the actual lifetime achieved to its corresponding upper

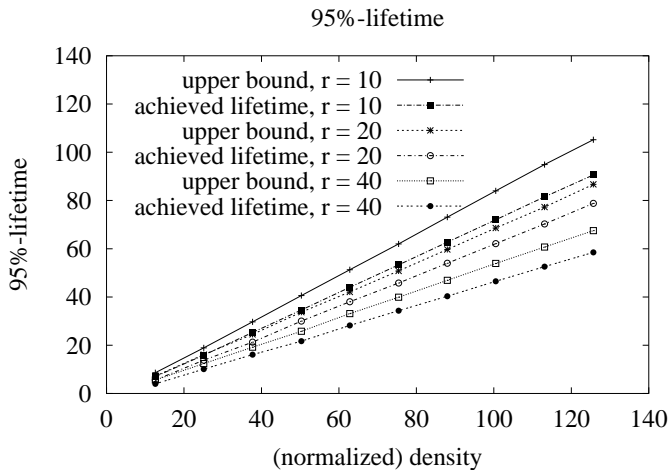


Fig. 9. Achieved 95%-lifetime and corresponding upper bound for different values of sensing radii: 10, 20, 40.

bound. When the sensing radius is very small, to maintain the same normalized density as in the case of large sensing radii, one has to deploy much more sensor nodes. When the number of nodes becomes very large, the optimization problem becomes more challenging. The search algorithm may easily get stuck at some local minima and fails to find the minimal α -cover that maximizes the lifetime upper bound of the remaining nodes. This is partially confirmed by the fact that in the low density case, the ratio with smaller radii is larger than that with larger radii in Figure 10.

VI. CONCLUSIONS

In this paper we have investigated the upper bound of α -lifetime for large scale sensor networks and studied whether,

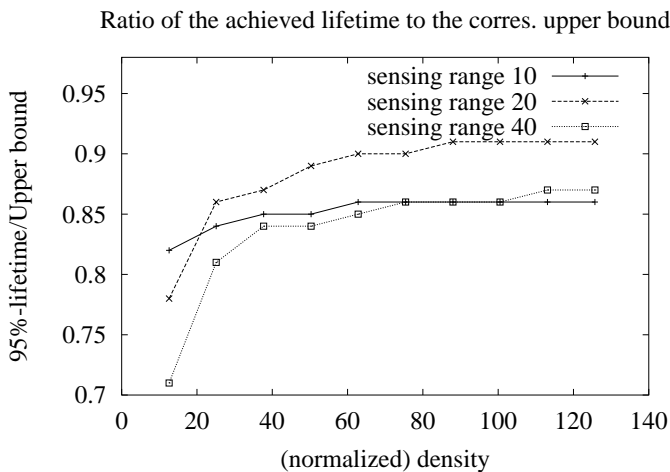


Fig. 10. Average ratios of the achieved α -lifetime to the corresponding upper bound for different values of sensing radii.

and to what extent the derived lifetime upper bound can be achieved. We first present two upper bounds of α -lifetime, one for a special family of algorithms that aim to maintain as maximum coverage as possible until the coverage ratio drops below α , and the other applies to all algorithms that maintain the coverage ratio of α from the beginning of network deployment. We also design an algorithm that schedules node activities (wake up and sleep) to maximize the α -lifetime. We carry out several sets of simulations to validate the derived bounds and evaluate the proposed algorithm. Simulation results show that the proposed algorithm achieves around 90% of the derived upper bound. This implies that the derived upper bound is rather tight and the solution determined by the proposed algorithm is close to the optimal one.

With the results derived in this paper, we suggest the following sensor network deployment and design strategies. First, a sensor network should be deployed with a reasonably high density in order to achieve a large lifetime per unit of nodal density and to optimize the coverage usage of sensor nodes. Second, as far as the lifetime is concerned, a sensor network should not cover the entire monitored region, but merely maintain α -lifetime from the beginning of systems operations, where α is the minimum percentage required by applications. Third, the criteria for choosing a working set of nodes may not necessarily be the number of nodes. Instead, it is desirable to choose a set of working nodes that maximizes the lifetime upper bound of the remaining set of nodes.

We have identified several research avenues. In particular, we are in the stage of devising, based on the proposed centralized algorithm, a light-weight distributed/localized algorithm that maximizes α -lifetime. The localized algorithm can be practically deployed in wireless sensor nodes to determine their wakeup and sleep schedule, with the objective of maintaining α -coverage while maximizing the network lifetime.

REFERENCES

- [1] Mems and nanotechnology. <http://www.memsnet.org/>.
- [2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: A survey. *Computer Networks*, 38(4):393–422, March 2002.
- [3] M. Bhardwaj and A. P. Chandrakasan. Upper bounds on the lifetime of wireless sensor networks. In *Proc. of IEEE International Conference on Communications (ICC) 01*, 2001.
- [4] M. Bhardwaj and A. P. Chandrakasan. Bounding the lifetime of sensor network via optimal role assignments. In *Proc. of IEEE Infocom*, 2002.
- [5] Douglas M. Blough and P. Santi. Investigating upper bounds on network lifetime extension for cell-based energy conservation techniques in stationary ad hoc networks. In *Proc. of ACM Mobicom 2002*, 2002.
- [6] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris. Span: An energy-efficient operation in multihop wireless ad hoc networks. In *Proc. of ACM MobiCom'01*, 2001.
- [7] D. Estrin, R. Govindan, J. S. Heidemann, and S. Kumar. Next century challenges: Scalable coordination in sensor networks. In *Proc. of ACM MobiCom'99*, Washington, August 1999.
- [8] H. Gupta, S. Das, and Q. Gu. Connected sensor cover: Self-organization of sensor networks for efficient query execution. In *Proc. of Mobicom 2003*, 2003.

- [9] C. F. Huang and Y. C. Tseng. The coverage problem in a wireless sensor network. In *ACM International Workshop on Wireless Sensor Networks and Applications (WSNA)*, 2003.
- [10] C. F. Huang and Y. C. Tseng. The coverage problem in a wireless sensor network. *ACM Mobile Networks and Applications (MONET) special issue on Wireless Sensor Networks (to appear)*, 2004.
- [11] J. M. Kahn, R. H. Katz, and K. S. J. Pister. Next century challenges: Mobile networking for "smart dust". In *Proc. of ACM MobiCom'99*, August 1999.
- [12] A. Mainwaring, J. Polastre, R. Szewczyk, and D. Culler. Wireless sensor networks for habitat monitoring. In *First ACM International Workshop on Wireless Workshop in Wireless Sensor Networks and Applications (WSNA 2002)*, August 2002.
- [13] S. Slijepcevic and M. Potkonjak. Power efficient organization of wireless sensor networks. In *ICC 2001*, Helsinki, Finland, June 2001.
- [14] D. Tian and N. D. Georganas. A coverage-preserving node scheduling scheme for large wireless sensor networks. In *First ACM International Workshop on Wireless Sensor Networks and Applications*, Georgia, GA, 2002.
- [15] X. Wang, G. Xing, Y. Zhang, C. Lu, R. Pless, and C. Gill. Integrated coverage and connectivity configuration in wireless sensor networks. In *ACM Sensys'03*, Nov. 2003.
- [16] F. Ye, G. Zhong, S. Lu, and L. Zhang. Energy efficient robust sensing coverage in large sensor networks. Technical report, UCLA, 2002.
- [17] F. Ye, G. Zhong, S. Lu, and L. Zhang. Peas: A robust energy conserving protocol for long-lived sensor networks. In *The 23rd International Conference on Distributed Computing Systems (ICDCS)*, 2003.
- [18] H. Zhang and J. Hou. On deriving the upper bound of α -lifetime for large sensor networks. In *Proc. of ACM MobiHoc 2004*, May 2004.
- [19] H. Zhang and J. Hou. Maintaining sensing coverage and connectivity in large sensor networks. *Wireless Ad Hoc and Sensor Networks: An International Journal*, 1(1-2):89–123, 2005.
- [20] H. Zhang and J. Hou. On the upper bound of α -lifetime for large sensor networks. *ACM Transaction on Sensor Networks*, To appear.
- [21] H. Zhang and J. C. Hou. Maintaining sensing coverage and connectivity in large sensor networks. Technical Report UIUCDCS-R-2003-2351, Department of Computer Science, University of Illinois at Urbana-Champaign, June 2003.