UC Irvine
UC Irvine Previously Published Works

Title

Scalable Coverage Maintenance for Dense Wireless Sensor Networks

Permalink

Journal

ISSN

Authors

Lu, Jun
Wang, Jinsu
Suda, Tatsuya

Publication Date

2007

DOI

Peer reviewed

Research Article

Scalable Coverage Maintenance for Dense Wireless Sensor Networks

Jun Lu, Jinsu Wang, and Tatsuya Suda

Bren School of Information and Computer Sciences, University of California, Irvine, CA 92697, USA

Owing to numerous potential applications, wireless sensor networks have been attracting significant research effort recently. The critical challenge that wireless sensor networks often face is to sustain long-term operation on limited battery energy. Coverage maintenance schemes can effectively prolong network lifetime by selecting and employing a subset of sensors in the network to provide sufficient sensing coverage over a target region. We envision future wireless sensor networks composed of a vast number of miniaturized sensors in exceedingly high density. Therefore, the key issue of coverage maintenance for future sensor networks is the scalability to sensor deployment density. In this paper, we propose a novel coverage maintenance scheme, scalable coverage maintenance (SCOM), which is scalable to sensor deployment density in terms of communication overhead (i.e., number of transmitted and received beacons) and computational complexity (i.e., time and space complexity). In addition, SCOM achieves high energy efficiency and load balancing over different sensors. We have validated our claims through both analysis and simulations.

1. INTRODUCTION

The recent advances in microsensor and communication technologies have increased the possibility of manufacturing inexpensive small wireless sensors with simple sensing, processing, and wireless communication capabilities. Limited by their size, small wireless sensors are equipped with a restricted power source and storage capacity. For example, the typical Crossbow MICA2 mote MPR400CB [1] has a low-speed 16 MHz microcontroller equipped with only 128 KB flash and 4 KB EEPROM. Powered by two AA batteries, it has the maximal data rate of 38.4 KBaud and a transmission range of about 150 m. Such small wireless sensors are usually deployed in an ad hoc manner to monitor a specified region of interest for various applications such as environment monitoring, target tracking, and distributed data storage.

One fundamental problem faced by current sensor network deployment is efficient provision of the required coverage. Specifically, given a target region, how can it guarantee that every point in the region is covered by the required number of sensors, with the object of maximizing the lifetime of the whole network? This problem is challenging due to the limitation of wireless sensor capabilities as well as the ad-hoc deployment properties of wireless sensor networks. One ef-

fective approach to extend sensor network lifetime is to have sensors autonomously schedule their duty cycles according to local information while satisfying global coverage requirements, which is referred to as coverage maintenance in the literature. We envision future wireless sensor networks composed of a vast number of small wireless sensors with very limited processing capability and storage capacity in exceedingly high density [2, 3]. Therefore, coverage maintenance for future wireless sensor networks must be highly scalable to sensor deployment density in terms of communication overhead as well as computational complexity.

In this paper, we propose a novel coverage maintenance scheme, scalable coverage maintenance (SCOM), in which sensors decide their sensing states in a distributive manner. SCOM works in two phases—the decision phase and the optimization phase. In the decision phase, sensors start in BOOTSTRAP state, and gradually make their decisions to enter the ACTIVE or INACTIVE state according to local information on coverage and energy. In the optimization phase, redundant active sensors turn off while still guaranteeing the required coverage. The main contributions of SCOM are (1) high scalability to sensor deployment density in terms of communication overhead and computational complexity, (2) a simple algorithm for a sensor to decide coverage

redundancy by checking only a small number of locations, (3) high energy efficiency to maintain the required coverage, and (4) load balancing over sensors.

The rest of this paper is organized as follows. Section 2 specifies SCOM in detail. Theoretical analysis and simulation results are presented in Sections 3 and 4, respectively. Section 5 examines the existing work on sensor network coverage. Section 6 concludes the paper.

2. SCALABLE COVERAGE MAINTENANCE (SCOM)

2.1. Assumptions

We assume that sensors are static and each sensor knows its own location. Sensors can acquire the location of neighbors through one-hop communication. Such assumptions are reasonably taken by other researches (e.g., [4–6]) and supported by the existing workes (e.g., [7–10]). We also assume that sensors have synchronized timers (e.g., [11, 12]) and are aware of the amount of their own residual energy. We further assume that communication range of sensors, denoted by CR, is at least twice the maximal sensing range, denoted by SR. This assumption is usually true for real sensors. For example, HMC1002 magnetometer sensors have an SR of approximately 5 m [13] while MICA2 MPR400CB motes can transmit about 150 m [1]. Where CR is less than twice SR, SCOM can work by propagating control beacons through multiple hops.

2.2. Problem statement

Definition 1. A location is covered by a sensor if it is within the SR of the sensor. A location is said to be K-covered if it is within the SR of at least K sensors. A region is K-covered if every location within the region is K-covered.

Note that according to Definition 1 the sensing perimeter of a sensor is *not* covered by the sensor.

The number of sensors covering a location is regarded as the coverage degree at that location. The problem is to select a small number of sensors to maintain K-coverage of the target region while scheduling others to sleep, which is referred to as coverage maintenance [14].

Definition 2. Coverage maintenance: given a set of sensors S deployed in target region $\mathcal{A}$ and a natural number K, select a subset S' of S such that

$$\forall v \in \mathcal{A} \mid \begin{cases} C_{S'}(v) \geq K, & C_S(v) \geq K, \\ C_{S'}(v) = C_S(v), & C_S(v) < K, \end{cases} \quad (1)$$

where $C_S(v)$ and $C_{S'}(v)$ denote the coverage of location v provided by S and S', respectively.

From Definition 2, we can see that the subset S' should provide at least K-coverage to a location if the location is K-covered by the full set of sensors S and should maintain the original coverage otherwise.

2.3. Scheme description

In SCOM, time is slotted into rounds. At the beginning of each round, each sensor goes through the following two phases.

(1) *Decision phase*: sensors start in BOOTSTRAP state, and make the decisions to enter the ACTIVE or INACTIVE state according to local information on coverage and energy.
(2) *Optimization phase*: redundant active sensors turn off while still guaranteeing the required coverage.

In the decision phase, each sensor is initially in BOOTSTRAP state and has an empty active neighbor list. Before making its decision, each sensor sets a back-off timer T_{decision} according to its residual energy,

$$T_{\text{decision}} = \alpha \cdot (1 - p) + \epsilon, \quad (2)$$

where p is the residual energy percentage level, α is a positive real number, and ϵ is a small random number uniformly distributed within $(0, \chi]$. α and χ decide the sensitivity of T_{decision} to the percentage level of residual energy, that is, larger α accentuates while larger χ de-emphasizes the difference of residual energy among sensors. How to set the values of α and χ is beyond the scope of this paper, and will be part of our future work. When its timer expires, a sensor decides its redundancy by checking whether its sensing region is K-covered by the sensors in the active neighbor list, and switches to ACTIVE or INACTIVE state accordingly. Detailed description of the redundancy checking algorithm is presented in Section 2.4. If a sensor decides to switch to ACTIVE state, it broadcasts a TURNON beacon including its ID, coordinates, and SR to the neighbors whose sensing regions overlap with the sensor. Upon receiving the TURNON beacon, a neighbor in BOOTSTRAP or ACTIVE state adds the sender ID to the active neighbor list and stores the coordinates and the SR of the sender. The decision phase lasts for $(\alpha + \chi)$ time units.

After the decision phase, there may exist redundant active sensors because the sensors turning on later may cover the sensing regions of the sensors that had already turned on and create redundancy. To eliminate the redundancy, each active sensor starts the optimization phase right after the decision phase by setting a back-off timer T_{opt} according to its residual energy,

$$T_{\text{opt}} = \alpha \cdot p + \epsilon, \quad (3)$$

where α, p, and ϵ have the same meaning as in (2). When a sensor times out, it checks for redundancy based on its active neighbor list and if redundant, switches to INACTIVE state and broadcasts a TURNOFF beacon to its active neighbors. Upon receiving the TURNOFF beacon, an active neighbor removes the sender ID from its active neighbor list. The optimization phase also lasts for $(\alpha + \chi)$ time units.

In the decision phase, according to (2), sensors with a higher percentage level of residual energy have a shorter
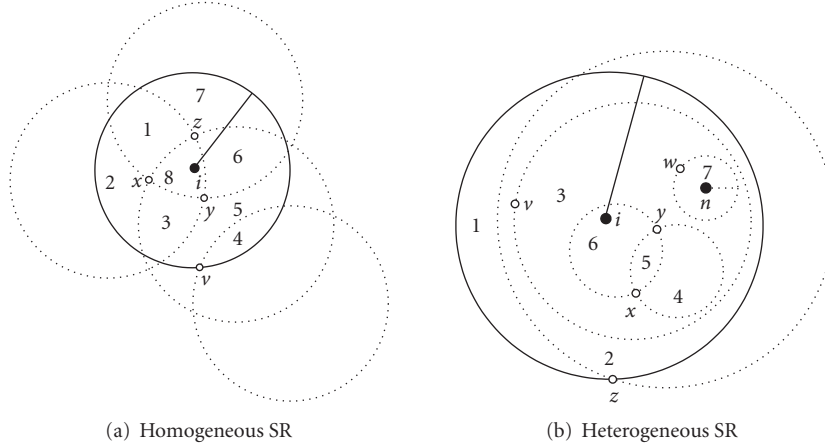
(a) Homogeneous SR

(b) Heterogeneous SR

FIGURE 1: SCOM-redundancy eligibility rule.

back-off period T_{decision} and thus time out earlier. Therefore, sensors with a higher percentage level of residual energy have more chance to switch to ACTIVE state. On the other hand, in the optimization phase, according to (3), sensors with a higher percentage level of residual energy have a longer back-off period T_{opt} and thus time out later. As a result, active sensors with a higher percentage level of residual energy have less chance to turn off. In this way, SCOM balances workload over sensors by employing sensors with more residual energy to provide coverage. It is clear that the precision of time synchronization and residual energy estimation may impact the performance of load balancing, but has no effect on guaranteeing required coverage.

2.4. Redundancy eligibility rule

The key operation of SCOM is to decide a sensor's redundancy given the location of the neighbors in the active neighbor list. Obviously, a sensor is redundant if its sensing region is K-covered by its neighbors. Here we propose a redundancy eligibility rule, by which a sensor is able to decide whether its sensing region is K-covered by its neighbors simply by checking the coverage at a few locations within its sensing region.

We first assume that *no* two sensors are at the same location, and later extend the proposed eligibility rule to handle multiple sensors at the same location. We describe redundancy eligibility rules for two cases: homogeneous SR (i.e., sensors have the same SR) and heterogeneous SR (i.e., sensors may have different SRs).

2.4.1. Sensors with homogeneous SR

For clarity, we have defined a sensor's critical point set.

Definition 3. Critical point set-sensor i's critical point set S_i contains, for each neighbor n, (1) the intersection points between the sensing perimeters of n and other neighbors within

the sensing region of sensor i, or if such intersection points do *not* exist, (2) one intersection point between the sensing perimeters of n and sensor i.

For example, in Figure 1(a), S_i contains three intersection points between sensor i's neighbors (i.e., x, y and z) and one intersection point between a neighbor and sensor i (i.e., v). Note that two tangent sensing perimeters are regarded to intersect each other at the point of contact.

Theorem 1. *In a homogeneous sensor network, given a natural number K, (1) if S_i is not empty, the sensing region of sensor i is K-covered by its neighbors if and only if each critical point in S_i is K-covered by its neighbors; (2) if S_i is empty, the sensing region of sensor i is not K-covered by its neighbors.*

Proof. (1) When S_i is not empty, the sensing region of sensor i is divided into subregions by the sensing perimeters of neighbors. For example, in Figure 1(a), sensor i's sensing region is divided into eight sub-regions. Since a sensor's sensing perimeter is not covered by the sensor itself according to Definition 1, the coverage of a sub-region is always higher than or equal to the coverage of adjacent critical points. For example, in Figure 1(a), the coverage of sub-region 8 is higher than or equal to the coverage of adjacent critical point x, y and z. Thus, the minimal coverage of sub-regions is *no less than* the minimal coverage of critical points. On the other hand, for each critical point, we can always find an adjacent sub-region with the same coverage. For example, in Figure 1(a), critical points x, y, and z have the same coverage as subregions 2, 5, and 7, respectively. Thus, the minimal coverage of critical points is *no less than* the minimal coverage of sub-regions. Therefore, the minimal coverage of critical points equals the minimal coverage of sub-regions, which means that if each critical point in S_i is K-covered by sensor i's neighbors, the sensing region of sensor i is K-covered by its neighbors, and vice versa. (2) An empty S_i implies that the sensing regions of sensor i and its neighbors do not overlap. Thus, the sensing region of sensor i is not K-covered by its neighbors. □
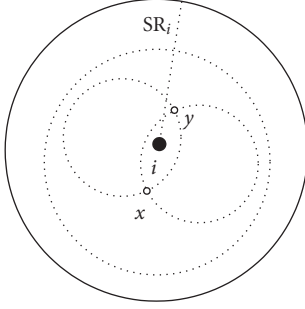
FIGURE 2: Theorem 1 cannot be applied for heterogeneous sensors.

FIGURE 3: Critical point set versus the existing algorithms.

2.4.2. Sensors with heterogeneous SR

When sensors have different SRs, Theorem 1 may not hold. For example, in Figure 2, S_i contains critical points x and y, both of which are 1-covered by sensor i's neighbors. However, the sensing region of sensor i is *not* 1-covered by its neighbors. To accommodate heterogeneous sensors, we have defined extended critical point set.

Definition 4. Extended critical point set-sensor i's extended critical point set ES_i contains (1) the critical points in critical point set S_i, and (2) a sampling point on each sensing perimeter that is within sensor i's sensing region and does *not* intersect with any other sensing perimeter.

For example, in Figure 1(b), S_i contains three critical points, x, y and z. There are two sensing perimeters that are contained in sensor i's sensing region and that do not intersect with other sensing perimeters. Thus, ES_i also contains v and w as the sampling points on the two sensing perimeters. Therefore, ES_i contains five critical points, x, y, z, w, and v.

Theorem 2. *In a heterogeneous sensor network, given a natural number K, (1) if ES_i is not empty, the sensing region of sensor i is K-covered by its neighbors if and only if each critical point in ES_i is K-covered by its neighbors; (2) if ES_i is empty, the sensing region of sensor i is K-covered by neighboring sensors if and only if a sampling point within the sensing region of sensor i is K-covered by its neighbors.*

Proof. (1) The proof is similar to Theorem 1. We can prove that the minimal coverage of the critical points in ES_i is equal to the minimal coverage of the sub-regions, which means that if each critical point in ES_i is K-covered by sensor i's neighbors, the sensing region of sensor i is also K-covered by its neighbors, and vice versa. (2) When sensors have heterogeneous SR, an empty extended critical point set does not necessarily mean that the sensing region has no overlap with others. For example, in Figure 1(b), ES_n contains no critical point, but sensor n's sensing region is contained in the sensing regions of its neighbors. In this case, the sensing region of n is *not* divided into sub-regions. Thus, sensor n can decide whether its sensing region is K-covered by checking the coverage of any sampling point within its sensing region.   □
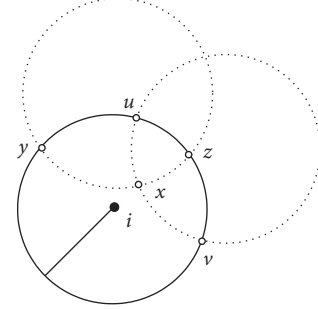
For the description above, we assume that no two sensors are at the same location. The redundancy eligibility rules described in Theorems 1 and 2 can be easily extended to accommodate the special case of multiple sensors at the same location. For sensors with homogeneous SR, if S_i is not empty, the coverage of the critical points *on sensor i's sensing perimeter* (e.g., v in Figure 1(a)) is increased by the number of sensors at the same location as sensor i; if S_i is empty, the sensing area of sensor i is covered by the number of sensors at the same location as sensor i. In the case of heterogeneous SR, if ES_i is not empty, the coverage of the critical points *on sensor i's sensing perimeter* (e.g., z in Figure 1(b)) is increased by the number of sensors at the same location and with the same SR as sensor i; in the case of an empty ES_i, we can still decide the redundancy of sensor i by checking a sampling point within its sensing region.

We note that a similar idea was proposed in Hall (1998) [15, page 56] to study the problem of covering a sphere with circular caps and later developed by [16, 17] for K-coverage maintenance in sensor networks. In their algorithms, however, the set of points to be checked by each sensor includes all the intersection points between the sensing perimeters of any two neighbors or between a neighbor and the sensor itself. Thus, their algorithms are required to check more points, and as a result, incur more computation overhead. For example, in Figure 3, critical point set S_i only contains point x, while the existing algorithms are required to compute coverage at all the intersection points, x, y, z, u, and v. Furthermore, the algorithms proposed in [15–17], assume homogeneous caps or sensors and cannot be applied to heterogeneous sensors.

3. SCHEME ANALYSIS

In this section, we analyze and compare the scalability of SCOM with the existing schemes proposed in [4, 6].

In the scheme proposed in [4] (hereinafter referred to as the *sponsored sector (SS) scheme*), every sensor calculates its eligibility to turn off. A sensor is eligible to turn off if its sensing region is contained by the union of the sponsored sectors offered by its active neighbors within SR. A back-off mechanism is used to avoid blind points caused by simultaneous decisions of multiple sensors. After the back-off period, a sensor eligible to turn off broadcasts a TURNOFF beacon

to the neighbors within SR. Upon receiving the TURNOFF beacon, every neighbor removes the sensor from the neighbor list so that the sensor will not be counted to decide the eligibility of other sensors.

In the scheme proposed in [6] (referred to as the *basic differentiated surveillance (DS)*), each sensor randomly generates a time-reference point and broadcasts it to the neighbors within twice SR. The target region is covered with a virtual square grid. A sensor decides the working schedule for each grid point within the SR based on its own time-reference point and the time-reference points of the neighbors covering the grid point. The final schedule of the sensor is the union of the working schedules for all the grid points. The final schedule can be optimized through exchanging schedule information among neighboring sensors (referred to as 2nd *pass differentiated surveillance (DS)*).

Let us investigate a sensor network composed of N homogeneous sensors with sensing range R uniformly deployed in a square area of $\ell \times \ell$ ($R \ll \ell$). For each scheme, we examine the growth of communication overhead (i.e., the number of transmitted and received beacons) and computational complexity (i.e., space and time complexity) as $N \to \infty$.

3.1. SCOM

Assume that there are M sensors turning on in the decision phase and N' ($N' \leqq M$) active sensors in the final network.

Theorem 3. *Given a limited required degree of coverage K, one has*

$$\lim_{N \to \infty} E(M) = O(1), \tag{4}$$

where $E(M)$ is the expected number of sensors that turn on in the decision phase of SCOM.

Proof. Without losing generality, we investigate a sensor network within a unit square area (i.e., ℓ is set to 1).

Let us first consider the independent turning on process, in which N sensors are uniformly deployed in BOOTSTRAP state initially and then randomly and independently turn on one by one until 1-coverage is fulfilled. It is clear that the location of the sensors turning on follows a stationary two-dimensional Poisson point process. Denote the density of the Poisson point process and the vacancy (i.e., the region not 1-covered) as λ and V_λ, respectively. It has been shown in Hall (1988) [15, Theorem 3.11, page 180] that

$$0.05\zeta_\lambda < P(V_\lambda > 0) < 3\zeta_\lambda, \tag{5}$$

where $\zeta_\lambda = \min\{1, (1 + \pi R^2 \lambda^2) e^{-\pi R^2 \lambda}\}$.

Obviously, the $(n+1)$th sensor turns on only when the n sensors that are already on cannot cover the area. Thus, the probability of requiring more than n active sensors can be calculated as the probability of vacancy larger than 0 with n active sensors, or

$$P(M > n) = P(V_{n/\ell^2} > 0) = P(V_n > 0). \tag{6}$$

Therefore, we have

$$\begin{aligned} E(M) &= \sum_{n=1}^{\infty} n \cdot P(M = n) \\ &< \sum_{n=1}^{\infty} n \cdot P(M > n - 1) \\ &= \sum_{n=1}^{\infty} n \cdot P(V_{n-1} > 0) \\ &< \sum_{n=1}^{\infty} n \cdot 3\zeta_{n-1} \\ &< \sum_{n=1}^{\infty} n \cdot 3(1 + \pi R^2 (n-1)^2) e^{-\pi R^2 (n-1)}. \end{aligned} \tag{7}$$

We can easily prove the convergence of the series in (7) with the ratio test

$$\lim_{n \to \infty} \frac{3(n+1)(1 + \pi R^2 n^2) e^{-\pi R^2 n}}{3n(1 + \pi R^2 (n-1)^2) e^{-\pi R^2 (n-1)}} = e^{-\pi R^2} < 1. \tag{8}$$

The convergence of the series indicates that $E(M)$ to provide 1-coverage is bounded by an upper limit, or $O(1)$ as $N \to \infty$.

In [18] (the proof of **Theorem 1**), Zhang and Huo presented an upper bound of the probability that a region is *not K-covered*. With the upper bound, we can prove that the expected number of sensors to provide K-coverage is also upper bounded by a limit, or $O(1)$ as $N \to \infty$. Since the proof is essentially the same as the 1-coverage case, we have omitted it here.

We have shown that $E(M)$ of the independent turning on process is $O(1)$ as $N \to \infty$. The difference between the decision phase of SCOM and the independent turning on process is that, in the decision phase of SCOM, a sensor turns on only when it is not redundant (instead of turning on independently). It is clear that the decision phase of SCOM yields fewer active sensors than the independent turning on process. Therefore, $E(M)$ of the decision phase of SCOM is also $O(1)$ as $N \to \infty$. □

3.1.1. Communication overhead

(a) Number of transmitted beacons

In SCOM, sensors transmit TURNON beacons and TURNOFF beacons in the decision phase and optimization phase, respectively. It is clear that sensors transmit M TURNON beacons in the decision phase and $(M - N')$ TURNOFF beacons in the optimization phase (note that N' is the number of active sensors after the optimization phase, which is no larger than M). The total number of transmitted beacons is $(2M - N')$, or $O(1)$ as $N \to \infty$.

(b) Number of received beacons

In the decision phase, only the sensors in BOOTSTRAP or ACTIVE state need to receive TURNON beacons. The average number of neighbors in BOOTSTRAP or ACTIVE

state of each sensor is upper bounded by $N\pi(2R)^2/\ell^2$ (in SCOM neighbor sensors are within the range of $2R$). Since there are in total M TURNON beacons transmitted, the total number of TURNON beacons received is upper bounded by $MN\pi(2R)^2/\ell^2$. In the optimization phase, only the sensors in ACTIVE state accept TURNOFF beacons. Since the average number of neighbors in ACTIVE state of each sensor is upper bounded by $M\pi(2R)^2/\ell^2$ and there are $(M - N')$ TURNOFF beacons transmitted, the total number of TURNOFF beacons received is upper bounded by $M\pi(2R)^2(M - N')/\ell^2$. The total number of TURNON and TURNOFF beacons received is upper bounded by $MN\pi(2R)^2/\ell^2 + M\pi(2R)^2(M - N')/\ell^2$, or $O(N)$ as $N \to \infty$.

3.1.2. Computational complexity

(a) Time complexity

In the decision phase, each sensor applies the redundancy eligibility rule to decide redundancy. The critical point set comprises the intersection points between the sensing perimeters of the sensor and its active neighbors. The average number of active neighbors of each sensor is upper bounded by $M\pi(2R)^2/\ell^2$. Thus, the number of critical points is upper bounded by $2 \cdot (M\pi(2R)^2/\ell^2)(1 + M\pi(2R)^2/\ell^2)$. For each critical point, a sensor needs to check for each active neighbor whether the critical point is covered. Thus, the number of basic computation steps to decide the redundancy is $2 \cdot (M\pi(2R)^2/\ell^2)^2(1 + M\pi(2R)^2/\ell^2)$, or $O(1)$ as $N \to \infty$. In the optimization phase, each active sensor checks its redundancy once, the number of basic computation steps of which is also $O(1)$. Thus, the time complexity is $O(1)$ as $N \to \infty$.

(b) Space complexity

The memory size required for each sensor to execute SCOM is mainly composed of (1) $M\pi(2R)^2/\ell^2$ entries for neighbors in ACTIVE state and (2) $2 \cdot (M\pi(2R)^2/\ell^2)(1 + M\pi(2R)^2/\ell^2)$ entries for critical points. Thus, the space complexity is $O(1)$ as $N \to \infty$.

3.2. Sponsored sector scheme

Denote the number of active sensors in the resulting network of SS as N'. Similarly, we can derive that $E(N')$ of SS is $O(1)$ when $N \to \infty$.

3.2.1. Communication overhead

(a) Number of transmitted beacons

Each sensor to turn off sends a TURNOFF beacon to inform its neighbors. Obviously, the total number of TURNOFF beacons transmitted is $(N - N')$, or $O(N)$ as $N \to \infty$.

(b) Number of received beacons

Only sensors that have not made their decisions need to receive TURNOFF beacons. In the best case, all the N' active sensors make decisions before the other sensors, and

thus no beacon is received by these N' sensors. Therefore, TURNOFF beacons are only exchanged among the $(N - N')$ sensors. For the ith sensor to turn off, the average number of received TURNOFF beacons is $(i - 1)\pi R^2/\ell^2$ (in SS neighbor sensors are within the range of R). Thus, the total number of TURNOFF beacons received by all the sensors is $\sum_{i=1}^{N-N'}((i - 1)\pi R^2/\ell^2)$, or $O(N^2)$ as $N \to \infty$.

3.2.2. Computational complexity

(a) Time complexity

In SS, each sensor checks all the active neighbors to decide its redundancy. Thus, the computational complexity is in the order of the number of active neighbors. A lower bound of the computational complexity can be derived by merely counting the computation overhead of the $(N - N')$ inactive sensors in the resulting network. For the ith sensor to turn off, the average number of active neighbors is $(N - i + 1)(\pi R^2/\ell^2)$. Thus, the total computational complexity of all the sensors is $O(\sum_{i=1}^{N-N'}((N - i + 1)\pi R^2/\ell^2))$, or $O(N^2)$. Thus, the average time complexity *per sensor* is $O(N)$ as $N \to \infty$.

(b) Space complexity

The memory size required for each sensor is mainly composed of $N\pi R^2/\ell^2$ entries on average to store neighbor states. Thus, the space complexity is $O(N)$.

3.3. Basic differentiated surveillance

3.3.1. Communication overhead

It is noted in [6] that the time-reference point beacons can be combined with the beacons to exchange coordinates among neighbors. Thus, there is no extra communication overhead in Basic DS.

3.3.2. Computational complexity

(a) Time complexity

As described in [6], there are averagely $\pi R^2/d^2$ grid points within a sensor's sensing region, where d is the unit grid size. Each sensor decides the schedule for each grid point according to neighbors' time reference points. Given $N\pi R^2/\ell^2$ neighbors on average covering the same grid point, it takes $(N\pi R^2/\ell^2)\log(N\pi R^2/\ell^2)$ basic computation steps to sort time-reference points and another constant time C to decide the sensor's schedule for the grid point. Finally, the schedules for all the grid points are combined to generate the integrated schedule for the sensor, which costs $2\pi R^2/d^2$ basic computation steps. Thus, the overall computational complexity is $(\pi R^2/d^2)((N\pi R^2/\ell^2)\log(N\pi R^2/\ell^2) + C) + 2\pi R^2/d^2$, or $O(N\log N)$ as $N \to \infty$.

(b) Space complexity

As described in [6], the memory size required for each sensor is mainly composed of (1) $N\pi(2R)^2/\ell^2$ entries on average for a neighbor table, (2) $N\pi R^2/\ell^2$ memory units on average for sorting time reference points and (3) $2\pi R^2/d^2$ memory units for schedules of grid points. The total space complexity is $O(N)$.

3.4. 2nd pass differentiated surveillance

3.4.1. Communication overhead

(a) Number of transmitted beacons

In 2nd pass DS, each sensor sends two beacons to inform its original integrated schedule and optimized schedule to neighbors. Thus, the total number of beacons transmitted is $O(N)$.

(b) Number of received beacons

First, each sensor receives the beacons for original integrated schedules from its neighbors. The total number of received beacons is $N \cdot (N\pi(2R)^2/\ell^2)$, or $O(N^2)$. Second, only sensors that have not optimized need to receive the beacons for optimized schedules. For the ith sensor to optimize, the average number of the received beacons is $(i-1)\pi(2R)^2/\ell^2$. Thus, the total number of received beacons for the optimized schedule is $\sum_{i=1}^{N}((i-1)\pi(2R)^2/\ell^2)$, or $O(N^2)$. Therefore, the total number of received beacons is $O(N^2)$.

3.4.2. Computational complexity

(a) Time complexity

In 2nd pass DS, each sensor carries out the basic DS algorithm and optimizes its schedule according to the schedules of its neighbors, both of which can be done in $O(N\log N)$. Thus, the time complexity is $O(N\log N)$.

(b) Space complexity

The memory capacity required for each sensor is mainly composed of (1) $N\pi(2R)^2/\ell^2$ entries on average for a neighbor table, (2) $N\pi R^2/\ell^2$ memory units on average for sorting time reference points on average, (3) $2\pi R^2/d^2$ memory units on average for schedules for all the grid points and (4) $N\pi(2R)^2/\ell^2$ entries on average for integrated schedules of neighboring sensors. Thus, the space complexity is $O(N)$.

Table 1 summarizes the scalability of different coverage maintenance schemes to sensor deployment density (note that given a fixed ℓ, N actually represents sensor deployment density) in terms of total communication overhead (i.e., number of transmitted and received beacons) and computational complexity (i.e., time and space complexity). We can see that SCOM outperforms other schemes except for the communication overhead of basic DS. However, the achievement of Basic DS is at the cost of energy efficiency and adaptability to sensor network dynamics such as sensor failures.

An integrated schedule generated by basic DS is a super set of schedules for many grid points, and therefore may be more than sufficient to provide the coverage guarantee. Moreover, when executed in multiple rounds, basic DS is not able to restore coverage from sensor failure because sensors are unaware of the failure of neighboring sensors. Although it is possible to use heartbeat signals to check the state of neighbors as described in [6], the communication overhead to transmit and receive heartbeat signals is $O(N)$ and $O(N^2)$, respectively. In contrast, at the beginning of each round, since only working sensors turn on and transmit TURNON beacons, SCOM can easily restore the coverage by substituting failed sensors with working ones.

Note that we assumed sensors with homogeneous SR in the above analysis. The analysis results are also valid for heterogeneous sensor networks as long as the SR is within a limited range.

From the above analysis, we know that SCOM is scalable because it only turns on necessary sensors in the decision phase. We have shown that, given the required degree of coverage, the number of sensors turning on in the decision phase is a limited value as sensor deployment density approaches infinity. Since each sensor only communicates to its active neighbors and only considers the active neighbors to make its decision, the communication and computation overhead per sensor remains limited with the increase of sensor deployment density. A similar technique is adopted by [17, 19, 20], but they do not provide specific analysis and evaluation of scalability of their schemes.

In summary, communication overhead and computational complexity *per sensor* are limited as the sensor deployment density approaches infinity, which makes SCOM favorable for dense sensor networks composed of simple sensors equipped with a slow processor and small storage.

4. SIMULATION STUDY

In this section, we compare the performance of SCOM with SS, DS, and 2nd pass DS schemes through simulations.

4.1. Simulation setup

The simulations are carried out over a square region of $100\,\text{m} \times 100\,\text{m}$ with wrap around in both dimensions. Thus, the results are representative of an infinite system, and therefore apply to typical large-scale sensor networks. Sensors are uniformly deployed in the square region.

In SCOM, α and χ of (2) and (3) are set to 10.0 and 1.0, respectively. We simulated both homogeneous and heterogeneous sensor networks. For homogeneous networks, SR is fixed at $10\,\text{m}$. For heterogeneous networks, a sensor's SR is uniformly chosen from three possible values: $5\,\text{m}$, $10\,\text{m}$, and $15\,\text{m}$.

4.2. Simulation results

The simulation results are shown for communication overhead, computational complexity, energy efficiency, and load balancing.

TABLE 1: Communication overhead and computational complexity.

Scheme	Total communication overhead		Computational complexity	
	Transmitted beacons	Received beacons	Time complexity	Space complexity
SCOM	$O(1)$	$O(N)$	$O(1)$	$O(1)$
SS	$O(N)$	$O(N^2)$	$O(N)$	$O(N)$
Basic DS	0	0	$O(N \log N)$	$O(N)$
2nd pass DS	$O(N)$	$O(N^2)$	$O(N \log N)$	$O(N)$

4.2.1. Communication overhead

Figures 4 and 5 show the communication overhead of different schemes to provide 1-coverage for sensor networks with homogeneous SR and heterogeneous SR, respectively. Figure 4 illustrates the total number of beacons transmitted and received in homogeneous sensor networks. Basic DS is not shown because it incurs no extra communication overhead by piggybacking the beacons to exchange time-reference points to the location exchanging beacons (as shown in Table 1, the number of transmitted and received beacons of basic DS is 0). Figure 4(a) depicts the total number of transmitted beacons with various sensor deployment densities. We can observe that the number of transmitted beacons of SCOM remains stable while that of the other two schemes grows linearly with the increase of sensor deployment density. We also see that the growth rate of SS is lower than that of 2nd pass DS because in SS only redundant sensors need to send beacons while every sensor transmits two beacons in 2nd pass DS. The simulation results confirm the analysis results in Table 1. Figure 4(b) shows that the number of received beacons of SCOM increases linearly with sensor deployment density, while that of 2nd pass DS grows quadratically. More detailed analysis reveals that the growth rate of SS is also quadratic, although much lower than 2nd pass DS. This observation also agrees with Table 1. Figure 5 describes the number of transmitted and received beacons in heterogeneous sensor networks. Our observations are similar to Figure 4; SCOM is more scalable than SS and 2nd pass DS in terms of communication overhead.

4.2.2. Computational complexity

The analysis in Section 3 reveals that the computational complexity is decided by the number of neighbors. Thus, we use the average number of neighbors of each sensor to measure the computational complexity. The results are shown in Figure 6 for 1-coverage. Since the average numbers of neighbors in the two phases (i.e., the decision phase and optimization phase) of SCOM are different, we show the average number of active neighbors in both phases. Because 2nd pass DS always has more computation overhead than Basic DS, we only show the results of basic DS. Figure 6(a) depicts the average number of active neighbors in homogeneous networks. We can see that the average number of active neighbors of both phases of SCOM remains constant, whereas that of SS and basic DS rises linearly with the growth

of sensor deployment density, which means that the computation overhead per sensor of SCOM remains stable (i.e., $O(1)$) while that of SS and basic DS increases with network deployment density. We also see that SS has fewer neighbors than basic DS because SS only considers neighbors within the range of SR. Again, this observation conforms to the analysis results in Table 1. As shown in Figure 6(b), we obtained similar results for heterogeneous sensor networks.

4.2.3. Energy efficiency

Figure 7 illustrates the energy consumption of monitoring to provide coverage for homogeneous sensor networks. The energy consumption is measured in units, which means the amount of energy consumed by an active sensor for a unit of time. In [21], a theoretical lower bound of the active sensor density to achieve 1-coverage is provided as $2/\sqrt{27}SR^2$, and is calculated in Figure 7(a) as a baseline for comparison. We can see that SCOM consumes less energy than the other three schemes. For example, the energy consumption of SCOM is about 16% less than that of 2nd Pass DS, which is the best among the other schemes. This is because SCOM uses actual SR while DS schemes use smaller conservative SR in order to avoid small sensing holes. From Figure 7(a), we also observe that SCOM consumes about 75% more energy than the theoretical lower bound. Figure 7(b) illustrates the energy consumption to provide differentiated degree of coverage (i.e., K-coverage), for which the sensor deployment density is fixed at 8 sensors/SR^2. Since [6] does not specify how to use 2nd pass DS to provide K-coverage, 2nd pass DS is not shown. We can see that SCOM significantly outperforms both basic DS and SS. The large discrepancy between SCOM and basic DS is due to the fact that a sensor's integrated schedule generated by basic DS is a super set of its schedules for many grid points, and therefore is more than sufficient to provide the coverage guarantee. Moreover, we notice that, with the increase of the required degree of coverage, the energy consumption of SCOM grows slower than that of basic DS and SS, and only slightly faster than the theoretical lower bound. The energy efficiency of different schemes in heterogeneous sensor networks is shown in Figure 8. Again, SCOM conserves more energy than other schemes.

4.2.4. Load balancing

As described in Section 2.3, by setting the back-off timers according to sensor residual energy, SCOM can achieve
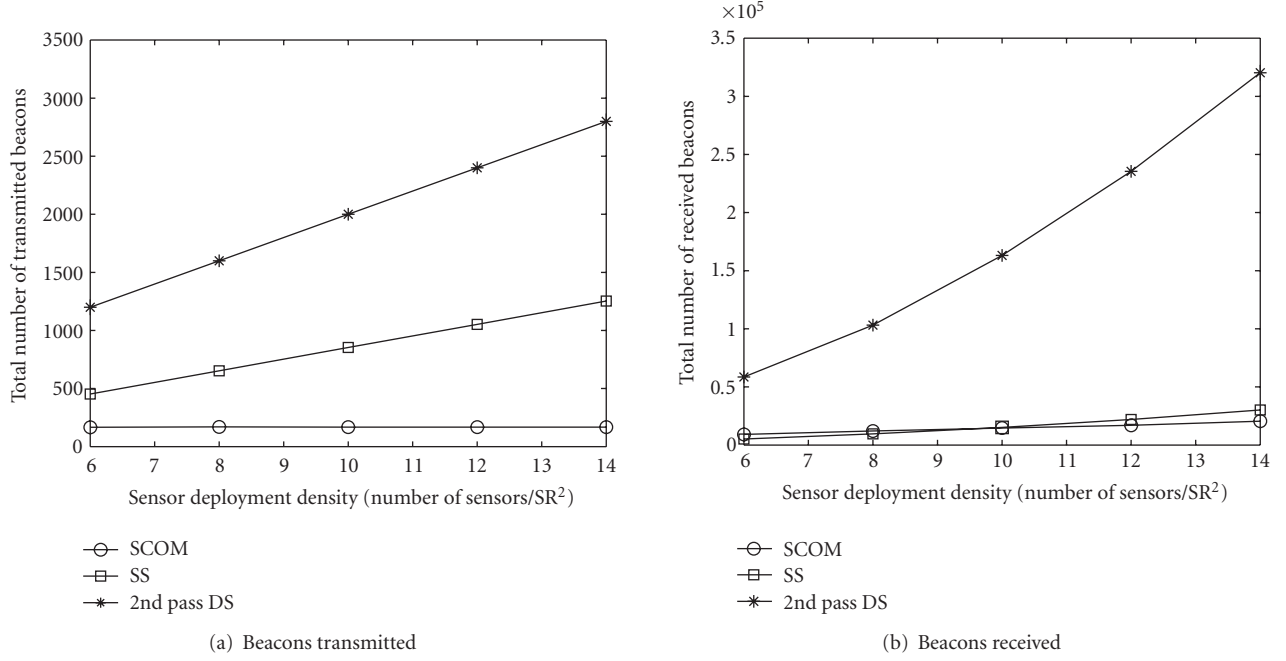
(a) Beacons transmitted

(b) Beacons received

FIGURE 4: Communication overhead (1-coverage, homogeneous SR = 10 m).

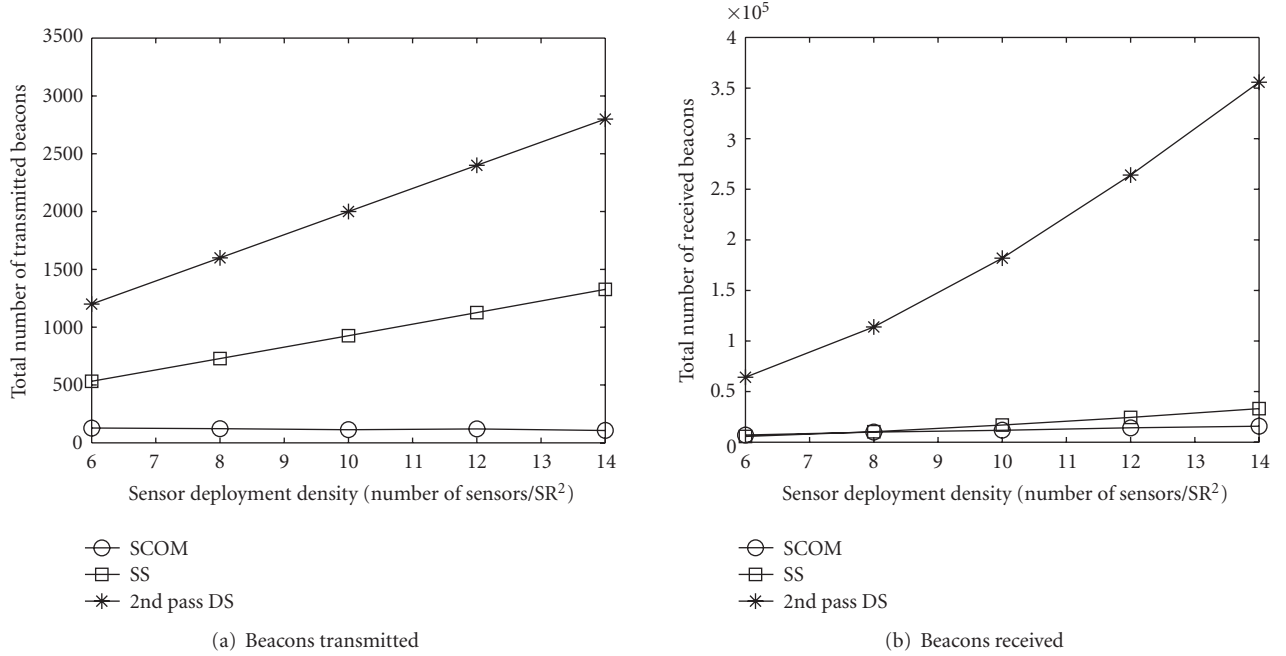(a) Beacons transmitted

(b) Beacons received

FIGURE 5: Communication overhead (1-coverage, heterogeneous SR = 5/10/15 m).

load balancing by employing sensors with more percentage of residual energy to provide network coverage. Here we compare SCOM with a modified version of SCOM (referred to as *SCOM without load balancing*). In SCOM without load balancing, instead of setting timers according to the amount of residual energy using (2) and (3), sensors simply

adopt random back-off timers. In the simulations, each sensor starts with 100% energy and the energy consumption rate is fixed at 10% per round. Figure 9(a) depicts the network lifetime of maintaining 1-coverage, which is measured as the time from the beginning of the deployment until the network loses 1-coverage of the target region. We can see that SCOM
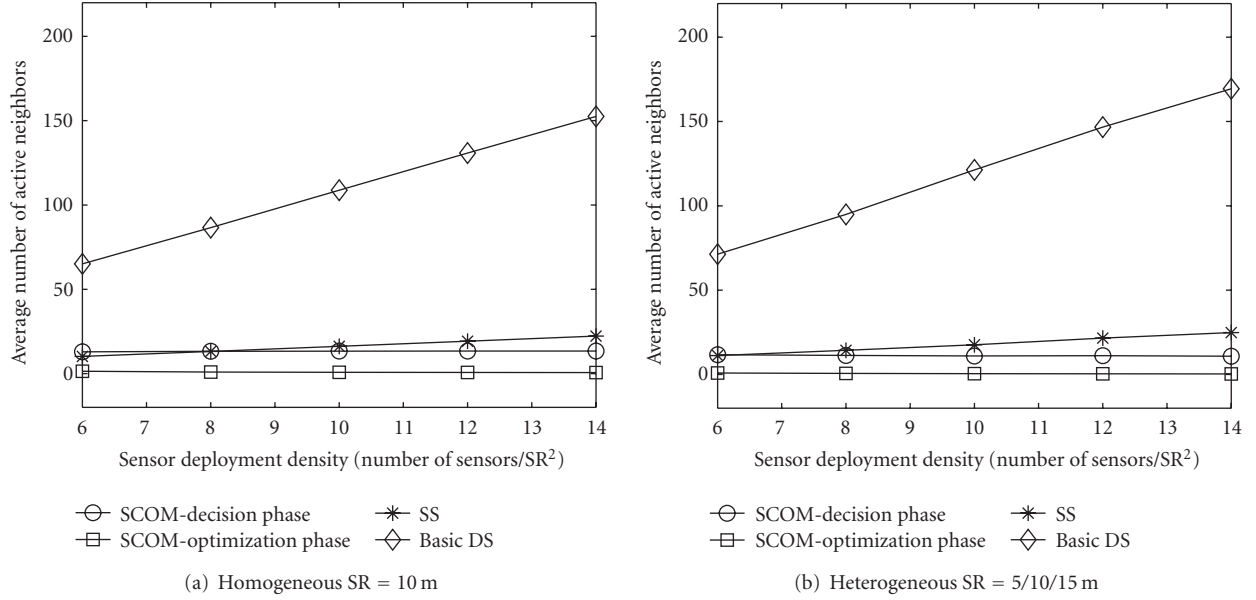
(a) Homogeneous SR = 10 m

(b) Heterogeneous SR = 5/10/15 m

FIGURE 6: Average number of active neighbors (1-coverage).

(a) 1-coverage

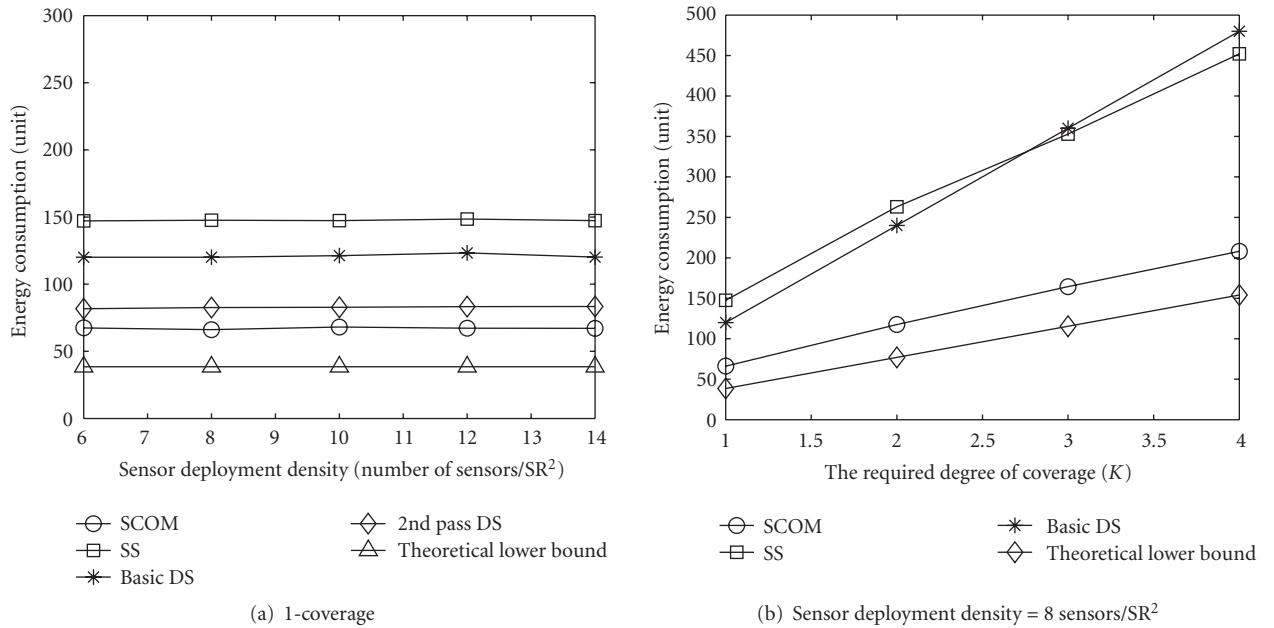(b) Sensor deployment density = 8 sensors/SR^2

FIGURE 7: Energy efficiency (homogeneous SR = 10 m).

considerably extends the lifetime of networks. Figure 9(b) provides a closer look at the load balancing of SCOM by showing how the standard deviation of residual energy in a network of 800 sensors evolves. We can see that SCOM lowers the residual energy deviation significantly, which means that SCOM better distributes workload among different sensors.

The simulation results presented above confirm that SCOM is highly scalable in terms of communication over-head and computational complexity, while remaining effective to conserve energy and balance load among sensors.

5. RELATED WORK

Sensing coverage reflecting the quality of monitoring provided by a sensor network has been the focus of intense studies recently.
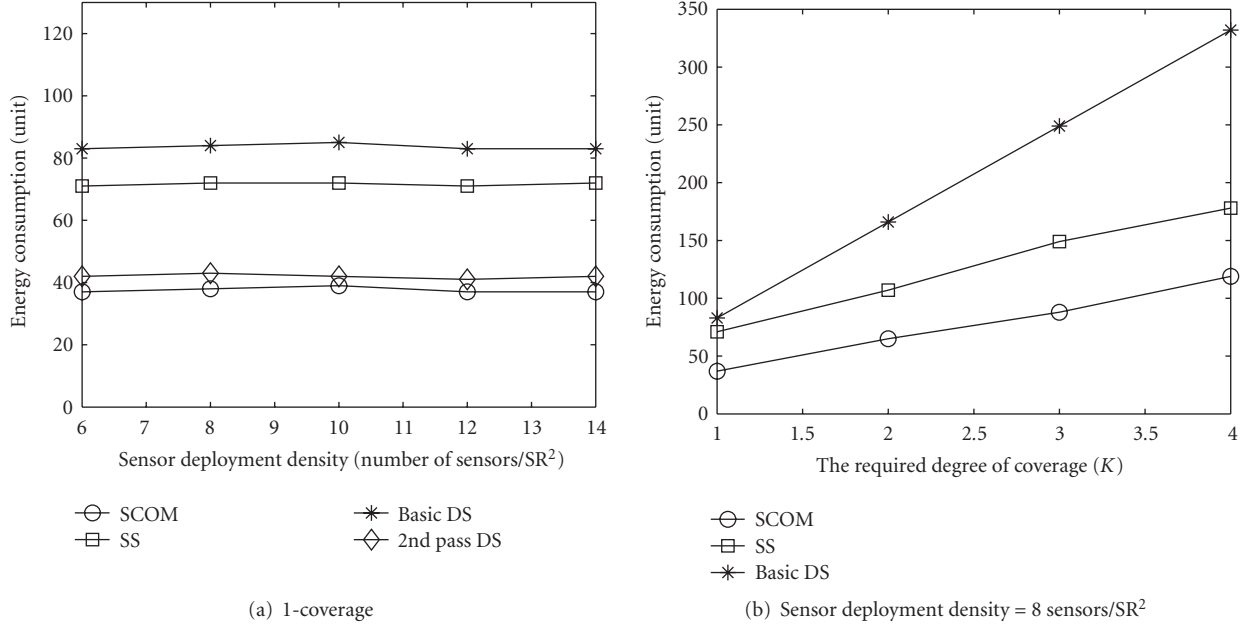
(a) 1-coverage

(b) Sensor deployment density = 8 sensors/SR^2

FIGURE 8: Energy efficiency (heterogeneous SR = 5/10/15 m).
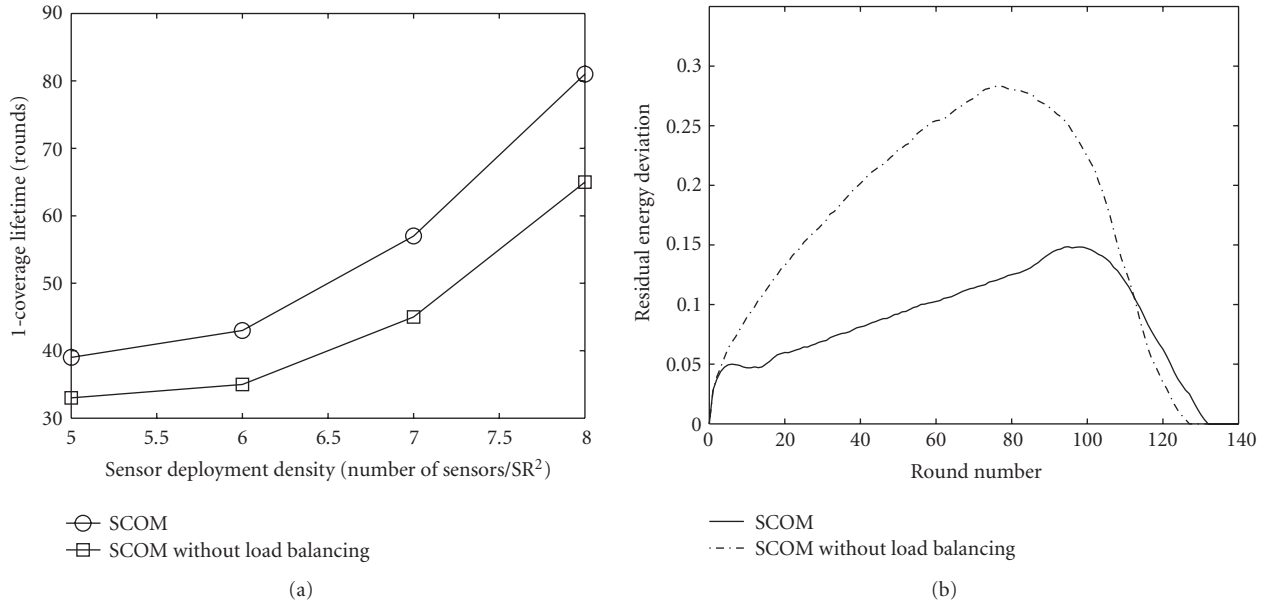

(a)

(b)

FIGURE 9: SCOM load balancing (1-coverage).

Some of the research studies investigate sensor network coverage from a theoretical perspective. For example, Zhang and Huo [18] derived the asymptotic upper bound of the 1-lifetime (i.e., the network lifetime to provide full coverage) for an infinite monitored area and the upper bound of the α-lifetime (i.e., the network lifetime to provide α-portion coverage) for a finite monitoring area. The authors of [22] analyzed sensor network coverage of wireless sensor networks by studying the relation between the number of neighbors and the coverage redundancy. Liu and Towsley [23] investigated the limits of sensor network coverage using different coverage measures, that is, area coverage, node coverage fraction and detectability. The critical conditions of sensor network configuration for asymptotic coverage are investigated in [24].

There are many coverage maintenance schemes proposed. For example, Tian and Georganas [4] presented a node scheduling algorithm to turn off sensors whose sensing areas are fully covered by the neighbors within sensing

range. Randomized as well as coordinated sleep algorithms were proposed in [25] to maintain network coverage using low duty-cycle sensors. The randomized algorithm enables each sensor to independently sleep under a certain probability, while the coordinated sleep algorithm allows a sensor to enter sleep state if its sensing area is fully contained by the union set of its neighbors. An algorithm was proposed in [5] to decide the coverage of a target area by merely checking the coverage state of sensing perimeters. Yan et al. [6] proposed an adaptable energy-efficient sensing coverage protocol, in which each sensor broadcasts a random time-reference point, and decides its duty schedule based on neighbors' time-reference points. Co-Grid proposed in [14] schedules sensors by adopting a distributed detection model based on data fusion. Abrams et al. studied a variant of the NP-hard SET K-COVER problem in [26], partitioning the sensors into K covers such that as many areas are monitored as frequently as possible. Xing et al. [16] studied the relationship between coverage and connectivity, and proposed a coverage maintenance scheme, coverage configuration protocol (CCP), which, when integrated with an existing connectivity maintenance scheme, is able to provide both coverage and connectivity guarantees. In [27], Kumar et al. proposed algorithms to decide quickly whether a deployed region is K-barrier covered. Two notions of probabilistic barrier coverage, the weak and strong barrier coverage, are introduced and studied. Cardei et al. [28] proposed an efficient scheme to cover a set of targets with known locations in a randomly and densely deployed sensor network. The target coverage problem is modeled as the maximal set cover problem and two heuristics are proposed and evaluated. Zhang and Huo [19] presented a scheme to optimize coverage maintenance while providing global connectivity by keeping a minimum number of active sensors to minimize coverage redundancy.

6. CONCLUSION

In this paper, we introduced SCOM that conserves energy while preserving the required sensing coverage by allowing sensors to autonomously decide their active/inactive states. An important property of SCOM is the high scalability to sensor deployment density in terms of communication overhead and computational complexity, which makes SCOM suitable for densely deployed sensor networks composed of simple sensors. We showed that the scalability of SCOM is better than the earlier works through both theoretical analysis and simulation study. Moreover, we demonstrated through simulation study that SCOM outperforms several existing competitors on energy efficiency and effectively balances workload among sensors.

ACKNOWLEDGMENTS

REFERENCES

[1] Crossbow Technology Inc., *MICA2 Data Sheet*, http://www.xbow.com/products/Product_pdf_files/Wireless_pdf/MICA2_Datasheet.pdf.

[2] K. S. J. Pister, J. M. Kahn, and B. E. Boser, "Smart Dust: Wireless Networks of Millimeter-Scale Sensor Nodes," Highlight Article in 1999 Electronics Research Laboratory Research Summary, 1999.

[3] B. Warneke, M. Last, B. Liebowitz, and K. S. J. Pister, "Smart dust: communicating with a cubic-millimeter computer," *Computer*, vol. 34, no. 1, pp. 44–51, 2001.

[4] D. Tian and N. D. Georganas, "A coverage-preserving node scheduling scheme for large wireless sensor networks," in *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications (WSNA '02)*, pp. 32–41, Atlanta, Ga, USA, September 2002.

[5] C.-F. Huang and Y.-C. Tseng, "The coverage problem in a wireless sensor network," in *Proceedings of the 2nd ACM International Workshop on Wireless Sensor Networks and Applications (WSNA '03)*, pp. 115–121, San Diego, Calif, USA, September 2003.

[6] T. Yan, T. He, and J. A. Stankovic, "Differentiated surveillance for sensor networks," in *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems (SenSys '03)*, pp. 51–62, Los Angeles, Calif, USA, November 2003.

[7] J. Albowicz, A. Chen, and L. Zhang, "Recursive position estimation in sensor networks," in *Proceedings of International Conference on Network Protocols (ICNP '01)*, pp. 35–41, Riverside, Calif, USA, November 2001.

[8] N. Bulusu, J. Heidemann, and D. Estrin, "GPS-less low-cost outdoor localization for very small devices," *IEEE Personal Communications*, vol. 7, no. 5, pp. 28–34, 2000.

[9] P. Bahl and V. N. Padmanabhan, "RADAR: an in-building RF-based user location and tracking system," in *Proceedings of the 19th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '00)*, vol. 2, pp. 775–784, Tel Aviv, Israel, March 2000.

[10] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan, "The Cricket location-support system," in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MOBICOM '00)*, pp. 32–43, Boston, Mass, USA, August 2000.

[11] H. Dai and R. Han, "TSync: a lightweight bidirectional time synchronization service for wireless sensor networks," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 8, no. 1, pp. 125–139, 2004.

[12] J. Elson, L. Girod, and D. Estrin, "Fine-grained network time synchronization using reference broadcasts," in *Proceedings of the 5th Symposium on Operating System Design and Implementation (OSDI '02)*, pp. 147–163, Boston, Mass, USA, December 2002.

[13] Crossbow Technology Inc., *MTS/MDA Sensor and Data Acquisition Boards Users Manual*, Crossbow Technology Inc, San Jose, Calif, USA, 2003.

[14] G. Xing, C. Lu, R. Pless, and J. A. O'Sullivan, "Co-Grid: an efficient coverage maintenance protocol for distributed sensor networks," in *Proceedings of the 3rd International Symposium*

on Information Processing in Sensor Networks (IPSN '04), pp. 414–423, Berkeley, Calif, USA, April 2004.

[15] P. Hall, *Introduction to the Theory of Coverage Processes*, John Wiley & Sons, New York, NY, USA, 1988.

[16] G. Xing, X. Wang, Y. Zhang, C. Lu, R. Pless, and C. Gill, "Integrated coverage and connectivity configuration for energy conservation in sensor networks," *ACM Transactions on Sensor Networks*, vol. 1, no. 1, pp. 36–72, 2005.

[17] A. Gallais, J. Carle, D. Simplot-Ryl, and I. Stojmenović, "Localized sensor area coverage with low communication overhead," in *Proceedings of the 4th Annual IEEE International Conference on Pervasive Computing and Communications (PerCom '06)*, pp. 328–337, Pisa, Italy, March 2006.

[18] H. Zhang and J. C. Hou, "On deriving the upper bound of α-lifetime for large sensor networks," in *Proceedings of the 5th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '04)*, pp. 121–132, Tokyo, Japan, May 2004.

[19] H. Zhang and J. C. Huo, "Maintaining sensing coverage and connectivity in large sensor networks," *Ad Hoc & Sensor Wireless Networks*, vol. 1, no. 1-2, pp. 89–123, 2005.

[20] J. Lu and T. Suda, "Coverage-aware self-coordination in sensor networks," Tech. Rep. 04-19, School of Information and Computer Sciences, UC Irvine, Irvine, Calif, USA, 2004.

[21] F. Ye, G. Zhong, S. Lu, and L. Zhang, "Energy efficient robust sensing coverage in large sensor networks," Tech. Rep., UC Los Angeles, Los Angeles, Calif, USA, 2002.

[22] Y. Gao, K. Wu, and F. Li, "Analysis on the redundancy of wireless sensor networks," in *Proceedings of the 2nd ACM International Workshop on Wireless Sensor Networks and Applications (WSNA '03)*, pp. 108–114, San Diego, Calif, USA, September 2003.

[23] B. Liu and D. Towsley, "A study of the coverage of large-scale sensor networks," in *Proceedings of IEEE International Conference on Mobile Ad-Hoc and Sensor Systems (MASS '04)*, pp. 475–483, Fort Lauderdale, Fla, USA, October 2004.

[24] S. Kumar, T. H. Lai, and J. Balogh, "On K-coverage in a mostly sleeping sensor network," in *Proceedings of the 10th Annual International Conference on Mobile Computing and Networking (MOBICOM '04)*, pp. 144–158, Philadelphia, Pa, USA, September-October 2004.

[25] C.-F. Hsin and M. Liu, "Network coverage using low duty-cycled sensors: random & coordinated sleep algorithms," in *Proceedings of the 3rd International Symposium on Information Processing in Sensor Networks (IPSN '04)*, pp. 433–442, Berkeley, Calif, USA, April 2004.

[26] Z. Abrams, A. Goel, and S. Plotkin, "Set K-cover algorithms for energy efficient monitoring in wireless sensor networks," in *Proceedings of the 3rd International Symposium on Information Processing in Sensor Networks (IPSN '04)*, pp. 424–432, Berkeley, Calif, USA, April 2004.

[27] S. Kumar, T. H. Lai, and A. Arora, "Barrier coverage with wireless sensors," in *Proceedings of the 11th Annual International Conference on Mobile Computing and Networking (MOBICOM '05)*, pp. 284–298, Cologne, Germany, August-September 2005.

[28] M. Cardei, M. T. Thai, Y. Li, and W. Wu, "Energy-efficient target coverage in wireless sensor networks," in *Proceedings of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '05)*, vol. 3, pp. 1976–1984, Miami, Fla, USA, March 2005.