

Coverage and Detection of a Randomized Scheduling Algorithm in Wireless Sensor Networks

Yang Xiao, *Senior Member, IEEE*, Hui Chen, *Member, IEEE*, Kui Wu, *Senior Member, IEEE*, Bo Sun, *Member, IEEE*, Ying Zhang, Xinyu Sun, and Chong Liu, *Student Member, IEEE*

Abstract—In wireless sensor networks, some sensor nodes are put in sleep mode while other sensor nodes are in active mode for sensing and communication tasks in order to reduce energy consumption and extend network lifetime. This approach is a special case ($k = 2$) of a randomized scheduling algorithm, in which k subsets of sensors work alternatively. In this paper, we first study the randomized scheduling algorithm via both analysis and simulations in terms of network coverage intensity, detection delay, and detection probability. We further study asymptotic coverage and other properties. Finally, we analyze a problem of maximizing network lifetime under Quality of Service constraints such as bounded detection delay, detection probability, and network coverage intensity. We prove that the optimal solution exists, and provide conditions of the existence of the optimal solutions.

Index Terms—Wireless sensor network, quality of service, network lifetime, coverage, optimization.

1 INTRODUCTION

WIRELESS sensor networks (WSNs) have a wide variety of military and civil applications. We consider a WSN consisting of a great number of sensor nodes. The sensor nodes are powered by batteries with limited energy. Hostile or hazardous environments where the sensor nodes are deployed or the sheer number of the sensors prevents replacement or recharge of the batteries. The number of sensors in the WSN is abundant to provide sufficient sensing coverage and network connectivity. Thus, it is possible that redundant sensor nodes can be turned off or enter sleep mode to save their battery power. A sensor node is called a redundant node if its sensing range is fully covered by other sensor nodes. Thus, the WSN remains functional after a redundant node is turned off or enters the sleep mode. When a sensor node is in the sleep mode or turned off, it consumes only a tiny fraction of the energy consumed in active mode. A turned-off or sleeping sensor node can be

waken up by a low power consuming timer at a later time or the network component upon request from its neighboring nodes.

Many research efforts have been devoted to sensor scheduling algorithms that turn off redundant sensors for energy saving [1], [2], [3], [4], [5], [7], [9], [16], [19]. Some of them do not require location information and precise time synchronization [1], [7], [9], [19]. Recently, the joint problem of coverage and connectivity is considered [9], [15], [18], [21], [22]. In those studies, sensor nodes are deployed either in grids or randomly. There are many research efforts on coverage-preserving scheduling schemes to extend network lifetime for WSNs [1], [2], [3], [4], [5], [6], [7], [8]. Many such research works are surveyed in [8].

Unlike previous work, this paper focuses on performance modeling and mathematical properties of a random coverage algorithm (also called k -set randomized scheduling algorithm) for WSNs. The algorithm is designed as follows [9]: Let S denote the set including all the sensor nodes in a WSN. Each sensor node is randomly assigned to one of k disjoint subsets ($S_j, j = 1, 2, \dots, k$), which work alternatively. In other words, at any time, only one set of sensor nodes are working, and the rest of sensor nodes sleep. Network lifetime is the elapsed time during which the network functions well, and the formal definition is given in (21) in a later section. In case that there is an intrusion such as an enemy tank invading a field covered with sensor nodes, detection delay is the average delay in terms of scheduling rounds to detect such an event, and detection probability is the probability of detecting the intrusion event. In addition, network coverage intensity is the ratio of the time when a point in the field of the sensor network is covered by at least one active sensor node to the total time. We denote them as D , P_d , and C_n , respectively.

A related scheme is called pure randomized schedule, in which each node wakes up $1/k$ of time. We provide a simple example to illustrate that this pure randomized scheme is

- Y. Xiao is with the Department of Computer Science, University of Alabama, Box 870290, Tuscaloosa, AL 35487. E-mail: yangxiao@ieee.org.
- H. Chen is with the Department of Mathematics and Computer Science, Virginia State University, Petersburg, VA 23806. E-mail: huichen@ieee.org.
- K. Wu is with the Department of Computer Science, University of Victoria, Canada. E-mail: wkui@cs.uvic.ca.
- B. Sun is with the Department of Computer Science, Lamar University, Beaumont, TX. E-mail: bsun@my.lamar.edu.
- Y. Zhang is with the School of Mathematical Sciences, Soochow University, Suzhou, Jiangsu 215006, China. E-mail: yingzhang@alumni.nus.edu.sg.
- X. Sun is with the Department of Mathematics, Tulane University, New Orleans, LA 70118. E-mail: xsun1@tulane.edu.
- C. Liu is with Research In Motion Limited, ON L4W 0B5, Canada. E-mail: cliu@rim.com.

Manuscript received 6 Dec. 2007; revised 28 Feb. 2009; accepted 20 July 2009; published online 29 Oct. 2009.

Recommended for acceptance by J.C.S. Lui.

For information on obtaining reprints of this article, please send e-mail to: tc@computer.org, and reference IEEECS Log Number TC-2007-12-0624. Digital Object Identifier no. 10.1109/TC.2009.170.

worse than the studied k -set randomized scheduling scheme in terms of coverage as follows: Assume that we have three sensors in a field and each, once working, will be able to cover the whole field. Assume that each sensor works $1/3$ of a unit time. In the studied k -set method, $k = 3$. The probability that the field can be fully covered within a unit time is equal to the probability that each set includes a sensor, which is $\frac{3 \times 2 \times 1}{3 \times 3 \times 3} = \frac{2}{9}$. With the pure randomized method, the probability that the field can be fully covered within a unit time is 0 since it equals to the probability that three random needles in a unit length, each with length $1/3$, fully cover the unit length. This is a typical Buffon's needle problem and has a probability of 0. Of course, the pure randomized algorithm is very simple and does not require time synchronization. Nevertheless, the benefit of the studied k -set randomized scheduling algorithm comes at a very trivial cost since it requires only loose time synchronization. In [9], the analysis on the impact of time asynchrony is provided. In [23], the analysis on coverage intensity of sensor networks where sensor nodes are deployed either on two-dimensional plane or in three-dimensional space and intrusion objects occupy either areas in two-dimensional plane or volumes in three-dimensional space is presented.

In this paper, we extend the study in [9]. The contribution can be summarized as follows: First, the paper provides a rigorous analysis for the randomized scheduling algorithm in terms of D , P_d , and C_n . The analysis is verified by computer simulations. Second, this paper analyzes the problem of maximizing network lifetime under Quality of Service (QoS) constraints such as bounded detection delay, detection probability, and network coverage intensity. Many works, such as [1], [2], [3], [4], [5], [6], [7], [8], [9], use only network sensing coverage as the QoS constraint. In addition to the coverage intensity, detection delay and detection probability are also very important measures. For example, since only one set of sensors are turned on, there is a chance that an intrusion event, in particular, a transient intrusion event, may not be detected. In some sensor networks, for example, actuator sensor networks, it is important to take an action based on the detection of an event. A too large detection delay may be disastrous. Thus, we believe that the optimization problem on the network lifetime with QoS constraints on coverage intensity, detection delay, and detection probability, is worth studying. We prove that the optimal solution exists, and provide the conditions of the existence of the optimal solutions. Third, based on the properties of the performance metrics discovered in the rigorous analysis, an efficient search algorithm similar to binary search for obtaining the optimal solution is discovered.

The rest of the paper is organized as follows: Since we often compare and verify the analytical results with computer simulations, we introduce our computer simulation program and the setup in Section 2. In Section 3, we study network coverage intensity and asymptotic coverage. In Section 4, we study intrusion period. In Section 5, we study detection probability and its properties, and in Section 6, we study detection delay and its properties. Section 7 analyzes the problem of maximizing network lifetime under QoS constraints. The duration that the simulations run affects the results. We explain the effects

of simulation duration on simulation results in Section 8. Finally, we conclude the paper in Section 9.

2 SIMULATION PROGRAM AND PARAMETER

We use computer simulations to verify the analytical model throughout the paper. This section presents the computer simulation program and the default parameters used in the paper. These parameters are applied only when simulations are used unless stated otherwise. In other words, these parameters may not be applied to analytical/mathematical models/derivations/theorems/lemmas.

We developed our own simulation program in C++. The program is an implementation of discrete event simulation. The locations of sensors and intrusions derived from uniform distributions. There are three types of events, intrusion events, detection events, and intrusion departure events. An intrusion event is generated randomly. A detection event occurs when the associated intrusion event is detected by at least one sensor node. The departure event is generated whenever the lifetime of the intrusion event expires.

By default, the sensing field is $a = 10,000$, the sensing area of a sensor is $r = 30$, the lifetime of an intrusion event is 2, the number of sensors deployed is $n = 10,000$ and all the sensors are divided into four disjoint sets of equal size. Note that the case that all subsets are of the same size can be regarded as an "average" case since each sensor node is randomly assigned to one of the four disjoint subsets as required by the random scheduling algorithm. Experiments indicate that this average case needs much less number of repeated simulations for a parameter setting to obtain a stable average of a performance metric. The above parameters are used in the simulations and the analytical analysis unless stated otherwise. As indicated in this study and [9], these parameters provide sufficient redundancy, which is required for the scheduling algorithm to maintain connectivity and network coverage.

3 NETWORK COVERAGE INTENSITY

In this section, we provide a derivation for network coverage intensity, and obtain the required number of sensors or the required number of subsets to achieve certain degree of network coverage intensity, which can be useful for sensor network deployments. The derivation is a simplified version of that presented in [9]. Furthermore, we derive and study asymptotic coverage, which is useful for better understanding the network coverage intensity.

3.1 Network Coverage Intensity

Let r , a , and k denote the size of sensing area of each sensor, the size of the whole sensing field, and the number of disjointed subsets, respectively. Then, r/a is the probability that each sensor covers a given point. Since any sensor is scheduled in one round among continuous k rounds, $r/(ak)$ is the probability that the sensor is active and covers a given point in any round. Therefore, for any given point and any given time, the probability that the point is not covered by any active sensor is $[1 - r/(ak)]^n$. Then, we have

$$C_n = 1 - [1 - r/(ak)]^n. \quad (1)$$

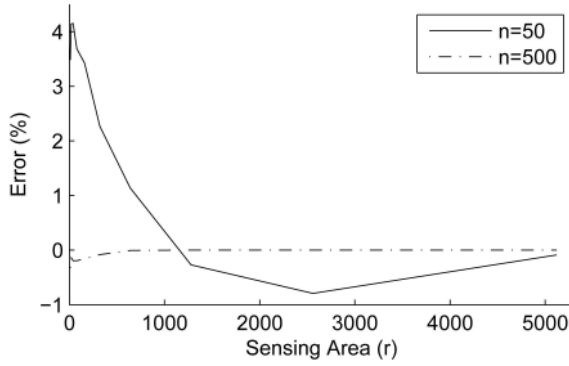


Fig. 1. Error of coverage intensity between analytical and simulation results.

The above derivation does not consider edge effect. Since the entire sensing field must have boundaries, a coverage area of a sensor node may not be completely inside the entire sensing field, which we refer to as the edge effect. Fig. 1 shows that the error rate between the simulation results and the analytical results is very small. Error rate is defined as $(C_n^a - C_n^s)/C_n^s$ where C_n^a and C_n^s stands for the coverage intensity obtained from (1) and simulations, respectively. The parameters used in the simulation are $a = 10,000$, $r = 30$, and $k = 4$.

From (1), we also know that the network coverage intensity is the probability that a given point at a given time is covered by at least one active sensor. Readers are directed to [9] for more discussion on the network coverage intensity.

3.2 Sensor Network Deployment

We now study the required number of sensors or the required number of subsets to achieve certain degree of network coverage intensity. We will answer the following two questions:

- *Question A:* Given a network coverage intensity and $r/(ak)$, what is the minimum number of sensors to achieve the network coverage intensity?
- *Question B:* Given a network coverage intensity and r/a , what is the maximum k value to achieve the network coverage intensity?

From (1), we can easily have the following results, which were also obtained in [9]:

- Given a required network coverage intensity C_{n-req} , the minimum number of sensors to achieve C_{n-req} is at least $n \geq \ln(1 - C_{n-req}) / \ln(1 - r/(ak))$. This result answers Question A in the above.
- Given a required network coverage intensity C_{n-req} , the maximum number of subsets to achieve C_{n-req} is

$$k \leq \frac{r}{a(1 - (1 - C_{n-req})^{1/n})}.$$

This result answers Question B in the above.

Fig. 2a shows the required minimum number of sensor nodes for a given coverage intensity versus $r/(ak)$. As illustrated in the figure, the required minimum number of sensor nodes decreases as the value of $r/(ak)$ increases. A larger coverage intensity needs more sensor nodes. The figure answers Question A in the above.

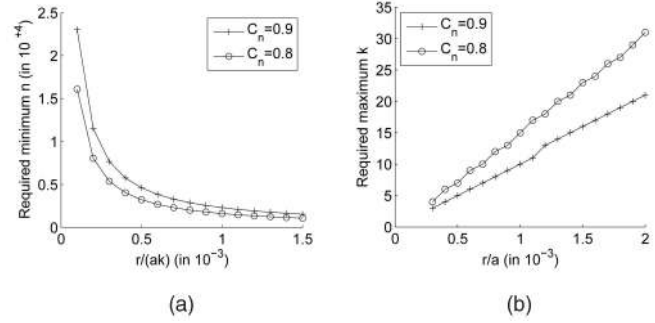


Fig. 2. The required minimum n and maximum k . (a) Required minimum n versus $r/(ak)$. (b) Required maximum k versus r/a .

Fig. 2b shows the required maximum k value for a given coverage intensity versus r/a , where $a = 25,000$. The figure answers Question B in the above. As illustrated in the figure, the required k value increases as the value of r/a increases. A larger coverage intensity needs a smaller k value.

3.3 Asymptotic Coverage and Other Properties

From (1), we can easily get the following lemma:

Lemma 1. *Network coverage intensity is an increasing function of n and $\lim_{n \rightarrow \infty} C_n = 1$ holds; Network coverage intensity is a decreasing function of k , and $\lim_{k \rightarrow \infty} C_n = 0$ holds.*

Lemma 1 implies that 1) given a fixed k , any network coverage intensity can be achieved by increasing the number of sensors deployed; 2) given a fixed number of sensors deployed, increasing k decreases network coverage intensity. These are consistent with our intuition.

Assuming that k and n are proportional such that $n = km$, where m is the number of sensors per subset/shift and is fixed, we have

$$\lim_{\substack{k=n/m \\ n \rightarrow \infty}} C_n = 1 - \lim_{n \rightarrow \infty} \left(1 - \frac{rm}{an}\right)^n = 1 - e^{-\frac{rm}{a}} \triangleq C(m), \quad (2)$$

where $C(m)$ is a function of the number of sensors per shift (m), which is an interesting feature of network coverage intensity.

Lemma 2.

1. $C(m) \triangleq \lim_{\substack{k=n/m \\ n \rightarrow \infty}} C_n = 1 - e^{-\frac{rm}{a}}$;
2. $\lim_{m \rightarrow \infty} C(m) = 1$; and
3. $C(m)$ monotonically increases with r/a .

4 INTRUSION PERIOD

In this section, we derive and evaluate, a nontrivial metric, intrusion period, which is important in deriving detection probability and detection delay in later sections.

Let L denote a duration when an intrusion event lasts. Let T denote the length of a scheduling round/cycle. Assume that an intrusion event happens randomly.

Let us study the number of cycles in which an intrusion overlaps. Let Z denote the average number of overlapping cycles of the intrusion period. Let Y denote a random variable representing the beginning of the intrusion event, and it is in the range of $[t_0, t_0 + T)$. Let us define $s = (\frac{L}{T} + 1 - \lceil \frac{L}{T} \rceil)$ and $Q = \lceil \frac{L}{T} \rceil$.

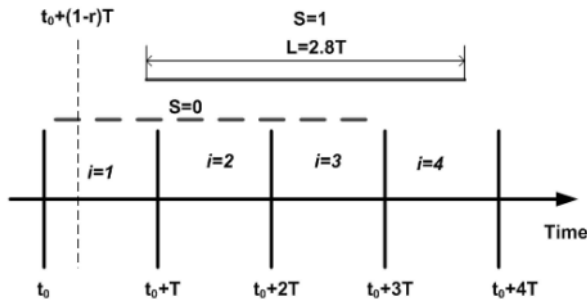


Fig. 3. Intrusion period.

Here, s is the remainder of the intrusion period in terms of the number of cycles when $L \neq iT$, where $(i = 1, 2, 3, \dots)$. In other words, when $L \neq iT$, we have $s = (\frac{L}{T} + 1 - \lfloor \frac{L}{T} \rfloor) = (\frac{L}{T} - \lfloor \frac{L}{T} \rfloor)$; however, when $L = iT$, where $(i = 1, 2, 3, \dots)$, $s = (\frac{L}{T} + 1 - \lfloor \frac{L}{T} \rfloor) \neq (\frac{L}{T} - \lfloor \frac{L}{T} \rfloor)$ because $(\frac{L}{T} + 1 - \lfloor \frac{L}{T} \rfloor)|_{L=iT} = \frac{iT}{T} + 1 - \lfloor \frac{iT}{T} \rfloor = i + 1 - i = 1$ and $(\frac{L}{T} - \lfloor \frac{L}{T} \rfloor)|_{L=iT} = \frac{iT}{T} - \lfloor \frac{iT}{T} \rfloor = i - i = 0$.

The interval $[t_0, t_0 + T)$ is cut into two regions/intervals, as shown in Fig. 3, $[t_0, t_0 + (1 - s)T)$ and $(t_0 + (1 - s)T, t_0 + T)$. If $Y \in [t_0, t_0 + (1 - s)T)$, intrusion duration L may overlap $\lfloor \frac{L}{T} \rfloor$ cycles. If $Y \in (t_0 + (1 - s)T, t_0 + T)$, intrusion duration L may overlap $\lfloor \frac{L}{T} \rfloor + 1$ cycles.

Since the intrusion duration L may overlap either $\lfloor \frac{L}{T} \rfloor$ or $\lfloor \frac{L}{T} \rfloor + 1$ cycles. Let us define a random variable $S \in \{0, 1\}$ such that if $S = 0$, L overlaps $\lfloor \frac{L}{T} \rfloor$ cycles, and if $S = 1$, L overlaps $\lfloor \frac{L}{T} \rfloor + 1$ cycles. We have

$$\Pr(S = 0) = \Pr(Y \in [t_0, t_0 + (1 - s)T]) = 1 - s, \quad (3)$$

$$\Pr(S = 1) = \Pr(Y \in (t_0 + (1 - s)T, t_0 + T)) = s. \quad (4)$$

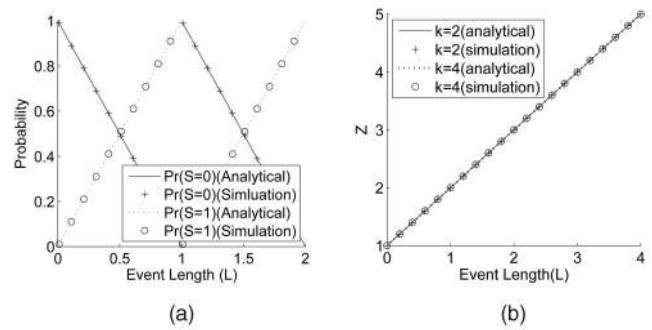
The reason that we use $(\frac{L}{T} + 1 - \lfloor \frac{L}{T} \rfloor)$ instead of $(\frac{L}{T} - \lfloor \frac{L}{T} \rfloor)$ is that if we use $s = (\frac{L}{T} - \lfloor \frac{L}{T} \rfloor)$, (3) and (4) will not be correct in some special cases when $L = iT$, where $(i = 1, 2, 3, \dots)$, i.e., $\Pr(S = 1) = (\frac{L}{T} - \lfloor \frac{L}{T} \rfloor)|_{L=iT} = 0$ will be incorrect. For example, assume that $L = 3.0T$ in Fig. 3. The intrusion period is of either three or four cycles, but the probability of three cycles is zero since it happens only in a very special case when $Y = t_0$ so that $\Pr(S = 0) = 0$ holds. The probability of four cycles is 1 so that $\Pr(S = 1) = 1 \neq 0$ holds. This proved that using $s = (\frac{L}{T} - \lfloor \frac{L}{T} \rfloor)$ causes incorrectness.

The average number of overlapping cycles of the intrusion period, Z , can be calculated as:

$$Z = \left\lfloor \frac{L}{T} \right\rfloor \Pr(S = 0) + \left(\left\lfloor \frac{L}{T} \right\rfloor + 1 \right) \Pr(S = 1) = \frac{L}{T} + 1. \quad (5)$$

For example, assume that $L = 2.8T$, shown in Fig. 3. Since $\lfloor \frac{2.8T}{T} \rfloor = 2$, the intrusion period overlaps either three cycles ($S = 0$) or four cycles ($S = 1$). The probability of three cycles is $\Pr(S = 0) = (1 - s) = 0.2$, where $s = \frac{2.8T}{T} + 1 - \lfloor \frac{2.8T}{T} \rfloor = 2.8 + 1 - 2 = 1.8$; the probability of four cycles is $\Pr(S = 1) = s = 0.8$. $Z = (\frac{2.8T}{T} + 1)T = 3.8T$.

Fig. 4 shows both analytical results and simulation results for $\Pr(S = 0)$ and $\Pr(S = 1)$. As illustrated in the figure, the analytical results match the simulation results exactly. Both

Fig. 4. $\Pr(S = 0)$, $\Pr(S = 1)$, and Z . (a) Probabilities over L and (b) Z over L .

$\Pr(S = 0)$ and $\Pr(S = 1)$ are periodic functions for the event length L , and this can be easily proved using (3) and (4). Fig. 4 shows Z over L when $T = 1$, and as illustrated in the figure, 1) the simulation results match the analytical results exactly, and 2) Z is an increasing function of L .

5 DETECTION PROBABILITY

5.1 Detection Probability

Let X denote a random variable representing the number of sensor nodes covering a point where the intrusion event happens. Let $I(e)$ denote the indication function which returns 1 if the condition e is true, and returns 0 otherwise.

It is clear that P_d depends on L . If L is very large (i.e., $L \geq (k - 1)T$), we have $P_d = C_n|_{k=1} = 1 - [1 - r/a]^n$.

Let $B_{h,j}$ denote the event that the intrusion event cannot be detected in all of h rounds if $X = j$ ($j > 0, 1 \leq h \leq k$) and the intrusion period does not finish. We have

$$\Pr(B_{h,j}) = \prod_{i=1}^h \left(1 - \frac{1}{k+1-i} \right)^j = \left(\frac{k-h}{k} \right)^j. \quad (6)$$

Let $A_j \equiv \Pr(UD | X = j)$ denote the probability of being unable to detect the intrusion event when $X = j$. We have

$$A_0 = 1, \quad (7)$$

$$A_j = 0, \text{ if } L \geq (k - 1)T \text{ and } n \geq j > 0, \quad (8)$$

$$A_j = \Pr(S = 0) \Pr(B_{\lfloor \frac{L}{T} \rfloor, j}) + \Pr(S = 1) \times \Pr(B_{\lfloor \frac{L}{T} \rfloor + 1, j}), \{L < (k - 1)T\} \cap \{n \geq j > 0\}, \quad (9)$$

$$P_d = 1 - \sum_{j=0}^n A_j \Pr(X = j) = 1 - \left(1 - \frac{r}{a} \right)^n - I[L < (k - 1)T] \sum_{j=1}^n A_j \binom{n}{j} \left(\frac{r}{a} \right)^j \left(1 - \frac{r}{a} \right)^{n-j}. \quad (10)$$

Plugging (7)-(9) and (1) into (10), we can obtain (11).

$$P_d = 1 - \left(1 - \frac{r}{a} \right)^n - I[L < (k - 1)T] \times \sum_{j=1}^n G_j \binom{n}{j} \left(\frac{r}{a} \right)^j \left(1 - \frac{r}{a} \right)^{n-j}, \quad (11)$$

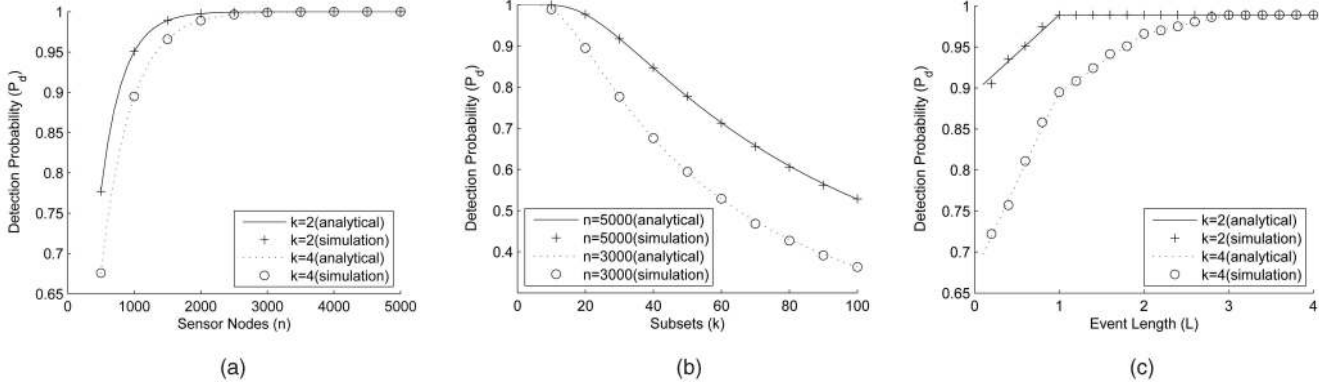


Fig. 5. Detection probability (P_d). (a) P_d versus n , (b) P_d versus k , and (c) P_d versus L .

where $G_j = (1-s)\left(\frac{k-\lceil \frac{L}{k} \rceil}{k}\right)^j + s\left(\frac{k-\lceil \frac{L}{k} \rceil - 1}{k}\right)^j$. We give proof of the following lemma in Appendix A.

Lemma 3. P_d can be simplified as follows:

$$P_d = \begin{cases} 1 - \left(1 - \frac{r}{a}\right)^n, & L \geq (k-1)T, \\ 1 - (1-s) \left(1 - \frac{\lceil \frac{L}{k} \rceil r}{a}\right)^n, & \\ -s \left(1 - \frac{\lceil \frac{L}{k} \rceil + 1}{k} \frac{r}{a}\right)^n, & L < (k-1)T. \end{cases} \quad (12)$$

5.2 Evaluation of Detection Probability

Fig. 5a shows the detection probability (P_d) versus the number of sensor nodes (n), where $T = 1$ and $L = 2$. As illustrated in the figure, the detection probability increases as n increases. A smaller k value causes a larger P_d . Furthermore, in both cases, P_d is very large in the figure. As illustrated in the figure, when n goes to infinity, P_d goes to 1, and this can be proved in Lemma 4 in the next section. Fig. 5a also shows that analytical results match the simulation results exactly.

Fig. 5b shows P_d versus k , where $T = 1$ and $L = 4$. As illustrated in the figure, P_d decreases as n increases. As illustrated in the figure, when k goes to infinity, P_d goes to 0, and this can be proved in Lemma 4 in the next section. A smaller n value causes a smaller P_d . Fig. 5b also shows that analytical results match the simulation results exactly.

Fig. 5c shows P_d versus L , where $T = 1$ and $n = 1,500$. As illustrated in the figure, P_d increases as the intrusion event length increases. A smaller k value causes a larger P_d . Furthermore, in both cases, P_d is very large in the figure. As illustrated in Fig. 5c, when L is large enough, P_d is close to 1. This is consistent with our intuition, and can be verified since when $L > (k-1)T$, $P_d = 1 - \left(1 - \frac{r}{a}\right)^n$ will be near to 1 for a large n value. Fig. 5c also shows that analytical results match the simulation results exactly.

5.3 Properties of P_d

Lemma 4.

1. P_d is an increasing function of n ;
2. P_d is a decreasing function of k ;
3. P_d is an increasing function of L ;
4. $\lim_{n \rightarrow \infty} P_d = 1$; and
5. $\lim_{k \rightarrow \infty} P_d = 0$.

Lemma 5. Let m be a fixed positive integer. Then, we have

$$\lim_{n=km, k \rightarrow \infty} P_d = 1 - (1-s)e^{-\lceil \frac{L}{k} \rceil \frac{r}{a} m} - se^{-\left(\lceil \frac{L}{k} \rceil + 1\right) \frac{r}{a} m}$$

for $L < (k-1)T$.

Let $W(\bullet)$ denote $\lim_{n=km, k \rightarrow \infty} P_d = 1 - (1-s)e^{-\lceil \frac{L}{k} \rceil \frac{r}{a} m} - se^{-\left(\lceil \frac{L}{k} \rceil + 1\right) \frac{r}{a} m}$. Fig. 6 shows $W(\bullet)$ over L , r/a , and m . As illustrated in the figure, $W(\bullet)$ is an increasing function of L , r/a , and m . When L , r/a , or m is large enough, $W(\bullet)$ is very close to 1, and becomes 1 when either L or r/a goes to infinity.

6 DETECTION DELAY AND ITS PROPERTIES

6.1 Detection Delay

It is clear that D also depends on the L value. We have either $D = \infty$ or $D < \infty$. If $D > L$ or $D \geq kT$, we have $D = \infty$. Since considering detection delay makes no sense if $D = \infty$, we only consider a finite value of detection delay for the rest of the paper, i.e., $D < \infty$.

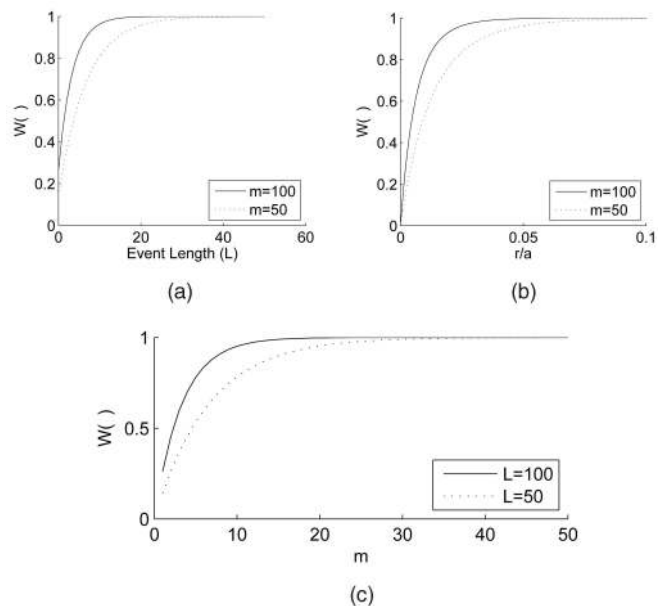


Fig. 6. Asymptotic detection probability. (a) $W(\bullet)$ versus L , (b) $W(\bullet)$ versus r/a , and (c) $W(\bullet)$ versus m .

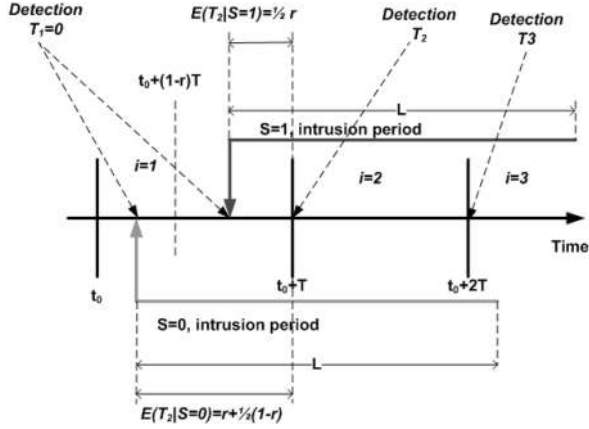


Fig. 7. Detection delay.

Let $E(D | X = j)$ denote the average detection delay under the condition of $(X = j)$. Let $A_{i,j}$ denote the event that the intrusion event is detected in the i th round if $X = j$ ($j > 0, 1 \leq i \leq k$) and the intrusion still exists in the i th round. Note that the first round is the 1st round instead of 0th round. We have

$$\begin{aligned} \Pr(A_{i,j}) &= \left[1 - \left(1 - \frac{1}{k+1-i} \right)^j \right] \prod_{h=1}^{i-1} \left(1 - \frac{1}{k+1-h} \right)^j \\ &= \left(\frac{k-i+1}{k} \right)^j - \left(\frac{k-i}{k} \right)^j. \end{aligned} \quad (13)$$

Let T_i denote the average time that the intrusion event is detected in the i th round. Let $(T_i | S = 0)$ and $(T_i | S = 1)$ denote T_i under conditions of $S = 0$ and $S = 1$, respectively.

Fig. 7 shows how to derive the mean values of $(T_i | S = 0)$ and $(T_i | S = 1)$, respectively. We have

$$E(T_i | S = 0) = \begin{cases} 0, & i = 1, \\ \left[i - 1 - \frac{1}{2}(1-s) \right] T, & \lceil \frac{L}{T} \rceil \geq i > 1, \end{cases} \quad (14)$$

$$E(T_i | S = 1) = \begin{cases} 0, & i = 1, \\ \left[i - \frac{1}{2}(4-s) \right] T, & \lceil \frac{L}{T} \rceil + 1 \geq i > 1. \end{cases} \quad (15)$$

If $X = 0$, the intrusion event cannot be detected so that $D = \infty$, which is not considered as stated before. In the

following derivations, a common technique is to use the conditional property, i.e., $\Pr(Y) = \sum \Pr(Y | X_i) \Pr(X_i)$, where $\sum \Pr(X_i) = 1$ and X_i is a division (without overlap) of the total set. Let $\phi_1 = \min(\lceil \frac{L}{T} \rceil, k)$ and $\phi_2 = \min(\lceil \frac{L}{T} \rceil + 1, k)$, we have

$$\begin{aligned} E(D | X = j \wedge D \neq \infty) &= \Pr(S = 0) \\ &\sum_{i=1}^{\phi_1} \frac{\Pr(A_{i,j})(T_i | S = 0)}{\sum_{i=1}^{\phi_1} \Pr(A_{i,j})} + \Pr(S = 1) \sum_{i=1}^{\phi_2} \frac{\Pr(A_{i,j})(T_i | S = 1)}{\sum_{i=1}^{\phi_2} \Pr(A_{i,j})}, \end{aligned} \quad (16)$$

$$\begin{aligned} E(D | D \neq \infty) &= \sum_{j=1}^n (D | X = j \wedge D \neq \infty) \Pr(X = j) \\ &= \sum_{j=1}^n (D | X = j \wedge D \neq \infty) \binom{n}{j} \left(\frac{r}{a} \right)^j \left(1 - \frac{r}{a} \right)^{n-j}. \end{aligned} \quad (17)$$

Plugging (13)-(16) into (17), we have (18)

$$E(D | D \neq \infty) = \sum_{j=1}^n M_j \binom{n}{j} \left(\frac{r}{a} \right)^j \left(1 - \frac{r}{a} \right)^{n-j}, \quad (18)$$

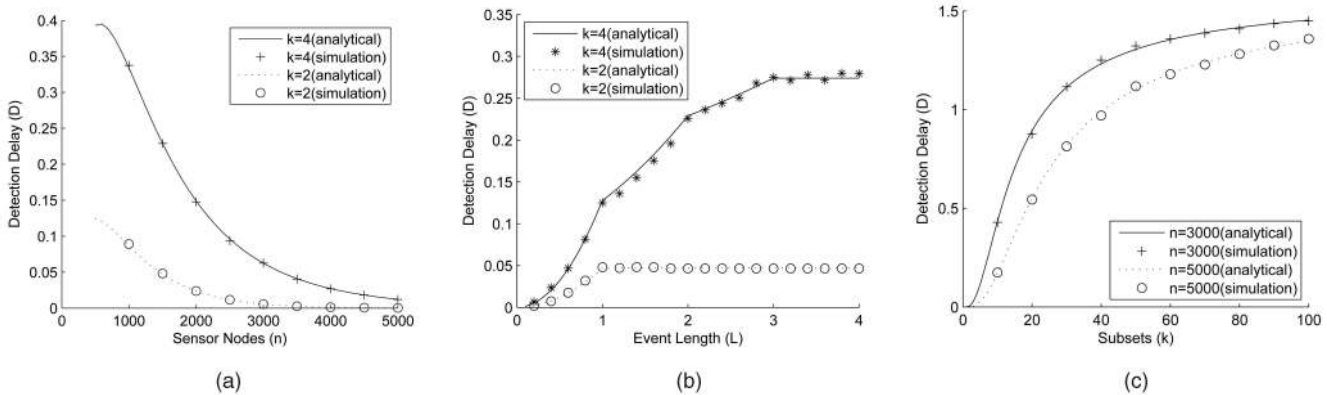
where

$$\begin{aligned} M_j &= \frac{(1-s) \sum_{i=2}^{\phi_1} \left[\left(\frac{k-i+1}{k} \right)^j - \left(\frac{k-i}{k} \right)^j \right] \left[i - \frac{3}{2} + \frac{s}{2} \right]}{\sum_{i=1}^{\phi_1} \left[\left(\frac{k-i+1}{k} \right)^j - \left(\frac{k-i}{k} \right)^j \right]} \\ &\quad + \frac{s \sum_{i=2}^{\phi_2} \left[\left(\frac{k-i+1}{k} \right)^j - \left(\frac{k-i}{k} \right)^j \right] \left[i - 2 + \frac{s}{2} \right]}{\sum_{i=1}^{\phi_2} \left[\left(\frac{k-i+1}{k} \right)^j - \left(\frac{k-i}{k} \right)^j \right]}. \end{aligned} \quad (19)$$

For the presentation purpose, in the rest of the paper, we simply use D to mean $E(D | D \neq \infty)$.

6.2 Evaluation of Detection Delay

Fig. 8a shows D versus n , where $T = 1$ and $L = 2$. As illustrated in the figure, D decreases as n increases. A smaller k value results in a smaller D . As illustrated in the figure, when n goes to infinity, D goes to 0, and this is consistent with our intuition. This figure also shows that analytical results almost match the simulation results, but not exactly. This is mainly because in the simulations, those sensors in the boundary of the field have the edge effect, which is not considered in the analytical model.

Fig. 8. Detection delay (D) (in units of rounds/cycles). (a) D versus n , (b) D versus L , and (c) D versus k .

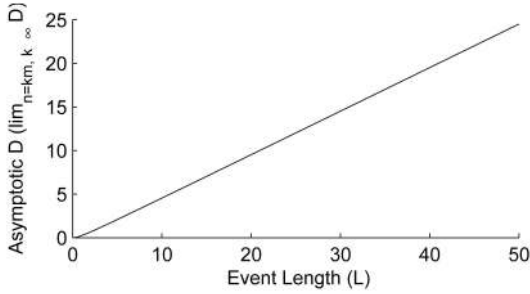


Fig. 9. $\lim_{n=km, k \rightarrow \infty} D$ versus L .

Fig. 8b shows D versus L , where $T = 1$ and $n = 1,500$. As illustrated in the figure, D increases as L increases. A smaller k value results in a smaller D . Fig. 8b also shows that the analytical results roughly match the simulation results, but not exactly. This is mainly because in the simulations, those sensors in the boundary of the field have the edge effect, which is not considered in the analytical model. The figure also indicates that as L goes to infinity, D goes to a positive fixed value, and this can be proved by Lemma 6 in the next section. It appears that the analytical results and simulation results of D with a small k (e.g., $k = 2$) have a better match than those with a large k (e.g., $k = 4$).

Fig. 8c shows D versus k , where $T = 1$ and $L = 4$. As illustrated in the figure, D increases as k increases. As k increases, the number of active sensor nodes per round/cycle is smaller so that it is more likely that the intrusion is not detected and, therefore, D increases. This figure also shows that the analytical results roughly match the simulation results, but not exactly. The figure also indicates that as k goes to infinity, D goes to a positive fixed value, and this is proved inside the proof to Lemma 7 in Appendix A.

6.3 Properties of D

We give proofs of lemma 7 in Appendix A.

Lemma 6. *If $L > (k - 1)T$, then D is a function independent of L and T .*

Lemma 7. *Let m be a fixed positive integer. Then, we have*

$$\lim_{n=km, k \rightarrow \infty} D = \frac{\frac{L}{T} \left(\left(\frac{L}{T} \right)^2 - \left\lceil \frac{L}{T} \right\rceil + \frac{L}{T} \right)}{2 \left\lceil \frac{L}{T} \right\rceil \left(\left\lceil \frac{L}{T} \right\rceil + 1 \right)}.$$

Fig. 9 show that $\lim_{n=km, k \rightarrow \infty} D$ is an increasing function of L .

Lemma 8. *D is a decreasing function of n when n is large enough.*

Lemma 9. $\lim_{n \rightarrow \infty} D = 0$.

Lemma 10. *D is an increasing function of k .*

7 MAXIMIZATION UNDER QoS

We studied the required number of sensors or the required number of subsets to achieve certain degree of network coverage intensity in Section 3, but detection delay and probability are not guaranteed. In this section, we study an optimization problem, i.e., to maximize network lifetime under Quality of Service constraints such as bounded detection delay, detection probability, and network coverage intensity.

Let T_{Slife} denote the average lifetime of a typical sensor. We provide the following definition (denoted as T_{Nlife}) for the network lifetime as follows:

$$T_{Nlife} = kT_{Slife}. \quad (20)$$

Note that the above definition assumes that the overhead of context-switches of the sleeping mode and the waking mode is omitted.

Optimization Problem 1. *To maximize T_{Nlife} under the following conditions:*

1. $D \leq QoS_{DD}$,
2. $P_d \geq QoS_{DP}$,
3. $C_n \geq QoS_{C_n}$, and
4. $n = c$,

where QoS_{DD} , QoS_{DP} , and QoS_{C_n} are pre-defined QoS constraints, and c is a constant value.

Since we have $T_{Nlife} = kT_{Slife}$, to maximize T_{Nlife} is to search the maximum k value to satisfy the QoS constraints. When k is very large, D must be large. Thus, a very large k value is not the best solution. In other words, there is an upper bound on k values with a relative small D . Since $C_n \geq QoS_{C_n} > 0$ can be rewritten

$$1 \leq k \leq \frac{r}{a(1 - (1 - QoS_{C_n})^{1/n})},$$

the optimal problem can be rewritten as follows:

Optimization Problem 2. *To find the maximum k value under the following conditions:*

1. $D \leq QoS_{DD}$,
2. $P_d \geq QoS_{DP}$,
3. $1 \leq k \leq \frac{r}{a(1 - (1 - QoS_{C_n})^{1/n})}$, and
4. $n = c$,

where QoS_{DD} , QoS_{DP} , and QoS_{C_n} are pre-defined QoS constraints, and c is a constant value.

Theorem 1. *The above optimal problem has an optimal solution, if*

$$QoS_{DD} < \frac{(Q - 1 + s)(Q^2 - 1 + s)}{2Q(Q + 1)} \left[1 - \left(1 - \frac{r}{a} \right)^c \right],$$

$$\frac{r}{a(1 - (1 - QoS_{C_n})^{1/c})} \geq 1, \quad 1 - \left(1 - \frac{r}{a} \right)^c \geq QoS_{DP} > 0,$$

and $1 > QoS_{C_n} > 0$, where c is a constant. In other words, The following set

$$S_a = \left\{ k \mid D \leq QoS_{DD} < \frac{(Q - 1 + s)(Q^2 - 1 + s)}{2Q(Q + 1)} \left[1 - \left(1 - \frac{r}{a} \right)^c \right], \right.$$

$$\left. P_d \geq 1 - \left(1 - \frac{r}{a} \right)^c \geq QoS_{DP} > 0, \right.$$

$$\left. 1 \leq k \leq \frac{r}{a(1 - (1 - QoS_{C_n})^{1/n})}, \quad 1 > QoS_{C_n} > 0, n = c \right\}$$

is nonempty, and is bounded.

Proof. Based on Lemma 10, D is an increasing function of k . Based on proof of Lemma 7, D tends to a function independent of k when k is large enough and

$$\lim_{k \rightarrow \infty} D = \frac{(Q-1+s)(Q^2-1+s)}{2Q(Q+1)} \left[1 - \left(1 - \frac{r}{a} \right)^n \right].$$

According to Lemma 10, D is an increasing function of k . The maximum possible value of D is $\lim_{k \rightarrow \infty} D$. Assume QoS_{DD} is valid, then $QoS_{DD} < \lim_{k \rightarrow \infty} D$. Therefore,

$$\left\{ k \mid D \leq QoS_{DD} < \frac{(Q-1+s)(Q^2-1+s)}{2Q(Q+1)} \left[1 - \left(1 - \frac{r}{a} \right)^n \right], \quad n = c \right\}$$

is nonempty, and is bounded.

Based on Lemma 4, P_d is a decreasing function of k and $\lim_{k \rightarrow \infty} P_d = 0$. Therefore, $\{k \mid P_d \geq 1 - (1 - \frac{r}{a})^c \geq QoS_{DP} > 0, n = c\} = \{1, 2, \dots, Y\}$ is bounded. It is not empty since $P_d|_{k=1} = 1 - (1 - \frac{r}{a})^c$ implies $Y \geq 1$.

$$\left\{ k \mid 1 \leq k \leq \frac{r}{a(1 - (1 - QoS_{C_n})^{1/n})}, \quad n = c \right\} \\ = \{1, 2, \dots, Z\}, \quad Z = \left\lfloor \frac{r}{a(1 - (1 - QoS_{C_n})^{1/n})} \right\rfloor$$

is nonempty, and is bounded. Therefore, the following set

$$S_a = \left\{ k \mid D \leq QoS_{DD} < \frac{(Q-1+s)(Q^2-1+s)}{2Q(Q+1)} \left[1 - \left(1 - \frac{r}{a} \right)^n \right], \quad n = c \right\} \\ \cap \left\{ k \mid P_d \geq 1 - \left(1 - \frac{r}{a} \right)^c \geq QoS_{DP} > 0, \quad n = c \right\} \\ \cap \left\{ k \mid 1 \leq k \leq \frac{r}{a(1 - (1 - QoS_{C_n})^{1/n})}, \quad n = c \right\} \\ 1 > QoS_{C_n} > 0, \quad n = c$$

is bounded. It is also not empty since $1 \in S_a$. Since values of k are positive integers and the set S_a is bounded so that the set S_a is closed too. \square

Since C_n , D , and P_d are monotonic functions of k as shown in [9], Lemma 4, and Lemma 10, respectively, k can be found by using a procedure similar to binary search. From Theorem 1, we know that

$$\left\{ k \mid 1 \leq k \leq \frac{r}{a(1 - (1 - QoS_{C_n})^{1/n})} \right\},$$

then the maximum number of steps to find the best k is

$$O\left(\log_2 \frac{r}{a(1 - (1 - QoS_{C_n})^{1/n})}\right),$$

if such k exists for the set of QoS constraints. The algorithm is shown in Algorithm 1. We refer the best k to as the optimal k , denote it as k_{opt} .

Algorithm 1. Optimal k searching algorithm

```

Input:  $k_{min}, k_{max}$ 
Result:  $k_{opt} > 0$  or 0. If 0,  $k_{opt}$  does not exist
while  $k_{max} - k_{min} \geq 1$  do
   $k_c = (k_{max} + k_{min})/2;$ 
  if IsTooSmall ( $k_c$ ) then
    if  $k_c == k_{max}$  then
       $k_c = 0;$  break;
    end
     $k_{min} = k_c$ 
  end
  else if IsTooGreat ( $k_c$ ) then
    if  $k_c == k_{min}$  then
       $k_c = 0;$  break;
    end
     $k_{max} = k_c;$ 
  end
   $k_{opt} = k_c;$ 
Input:  $k_c, QoS_{C_n}, QoS_{DP}$ , and  $QoS_{DD}$ 
Result: true if  $k_c$  is too small; false otherwise
Function IsTooSmall ( $k_c$ )
  calculate  $C_n(k_c) = C_n|_{k=k_c}, P_d(k_c) = P_d|_{k=k_c}$ , and
   $D(k_c) = D|_{k=k_c};$ 
  if  $C_n(k_c) \geq QoS_{C_n} \wedge P_d(k_c) \geq QoS_{P_d} \wedge$ 
   $D(k_c) \leq QoS_{DD}$  then
    return true;
  else
    return false;
  end
Input:  $k_c, QoS_{C_n}, QoS_{DP}$ , and  $QoS_{DD}$ 
Result: true if  $k_c$  is too great; false otherwise
Function IsTooGreat ( $k_c$ )
  calculate  $C_n(k_c) = C_n|_{k=k_c}, P_d(k_c) = P_d|_{k=k_c}$ , and
   $D(k_c) = D|_{k=k_c};$ 
  if  $C_n(k_c) \leq QoS_{C_n} \vee P_d(k_c) \leq QoS_{P_d} \vee$ 
   $D(k_c) \geq QoS_{DD}$  then
    return true;
  else
    return false;
  end

```

Another way of looking up the definition of network lifetime in (20) is that the network lifetime is defined as (20) together with the first three conditions defined in Optimization Problem 1. In other words, a network is defined alive if the first three conditions defined in Optimization Problem 1 can be satisfied, where QoS parameters are specified by users. Formally, it can be redefined as (21). It also can be defined by reducing one or two conditions in (21).

$$T_{Nife} = kT_{Sufe}, \quad \text{where} \\ \{D \leq QoS_{DD}\} \cap \{P_d \geq QoS_{DP}\} \cap \{C_n \geq QoS_{C_n}\}. \quad (21)$$

Fig. 10a shows the maximum k value versus QoS_{C_n} (i.e., QoS constraints of C_n) with fixed QoS constraints on P_d and D , where $n = 10,000$, $a = 10,000$, $r = 30$, $T = 1$, $L = 1$, $QoS_{DD} = 0.15$, and $QoS_{DP} = 0.6$. As illustrated in the figure, the maximum k value remains flat when QoS_{C_n} is small, but when QoS_{C_n} is large enough, it decreases sharply as QoS_{C_n} increases.

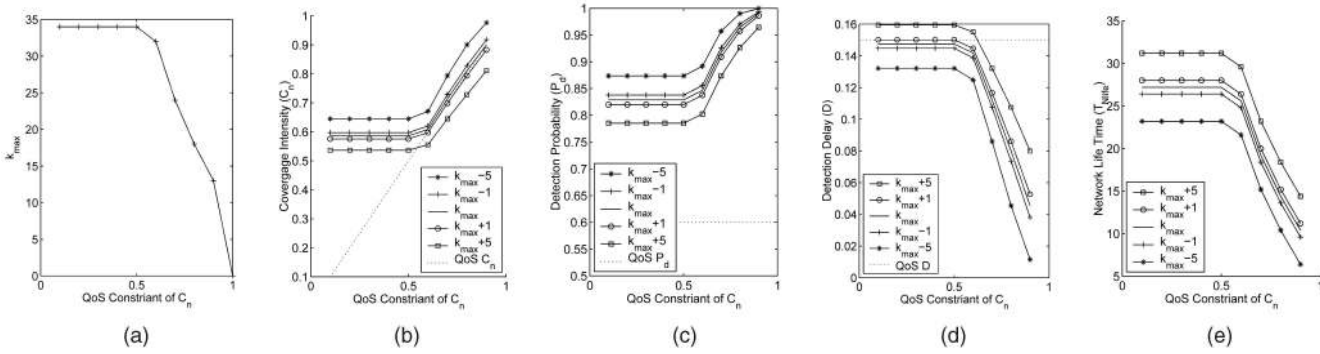


Fig. 10. Comparisons for QoS_{C_n} . (a) Maximum k versus QoS_{C_n} , (b) C_n versus QoS_{C_n} , (c) P_d versus QoS_{C_n} , (d) D versus QoS_{C_n} , and (e) lifetime versus QoS_{C_n} .

Figs. 10b, 10c, 10d, and 10e compare C_n , D , P_d , and T_{Nlife} with the maximum k values obtained from Fig. 10a with those not at the maximum k values under the same parameters as Fig. 10a. Although Fig. 10c shows that all five cases have higher P_d than the required $QoS_{DD}(=0.6)$, Fig. 10b shows that when QoS_{C_n} is large, the cases of $k_{max} + 1$ and $k_{max} + 5$ have smaller C_n than the required QoS_{C_n} , and Fig. 10d shows that when QoS_{C_n} is small, the cases of $k_{max} + 1$ and $k_{max} + 5$ have larger D than the required $QoS_{DD}(=0.15)$. In other words, the cases of $k_{max} + 1$ and $k_{max} + 5$ do not satisfy all QoS requirements. Furthermore, Fig. 10e shows that the cases of $k_{max} - 1$ and $k_{max} - 5$ have smaller T_{Nlife} than the case of k_{max} . In other words, the optimal one is the best among the five cases.

Fig. 11a shows the maximum k value versus QoS_{DD} (i.e., QoS constraints of D) with fixed QoS constraints of network coverage intensity and detection delay, where $n = 10,000$, $a = 10,000$, $r = 30$, $T = 1$, $L = 1$, $QoS_{DD} = 0.15$, and $QoS_{C_n} = 0.6$. As illustrated in the figure, the maximum k value remains flat when QoS_{DD} is small, but when QoS_{DD} is large enough, it decreases sharply as QoS_{DD} increases.

Figs. 11b, 11c, 11d, and 11e compare C_n , D , P_d , and T_{Nlife} with the maximum k values obtained from Fig. 11a with those not at the maximum k values under the same parameters as Fig. 11a. Fig. 11b shows that when QoS_{DD} is small, the cases of $k_{max} + 1$ and $k_{max} + 5$ have smaller C_n than the required $QoS_{C_n}(=0.6)$. Fig. 11c shows that when QoS_{DD} is large, the cases of $k_{max} + 1$ and $k_{max} + 5$ have smaller P_d than the required QoS_{DD} , and Fig. 11d shows that when QoS_{DD} is small, the cases of $k_{max} + 1$ and $k_{max} + 5$ have larger D than the required $QoS_{DD}(=0.15)$. In other

words, the cases of $k_{max} + 1$ and $k_{max} + 5$ do not satisfy all QoS requirements. Furthermore, Fig. 11e shows that the cases of $k_{max} - 1$ and $k_{max} - 5$ have smaller T_{Nlife} than the case of k_{max} . In other words, the optimal one is the best among five cases.

Fig. 12a shows the maximum k value versus QoS_{DD} (i.e., QoS constraints of D) with fixed QoS constraints on C_n and D , where $n = 10,000$, $a = 10,000$, $r = 30$, $T = 1$, $L = 1$, $QoS_{C_n} = 0.6$, and $QoS_{DD} = 0.6$. As illustrated in the figure, the maximum k increases when QoS_{DD} is small, and it remains flat when QoS_{DD} is large.

Figs. 12b, 12c, 12d, and 12e compares C_n , D , P_d , and T_{Nlife} with the maximum k values obtained from Fig. 12a with those not at the maximum k values under the same parameters as Fig. 12a. Although Fig. 12c shows that all five cases have higher detection probabilities than the required $QoS_{DD}(=0.6)$, Fig. 12b shows that when QoS_{DD} is large, the cases of $k_{max} + 1$ and $k_{max} + 5$ have smaller C_n than the required $QoS_{C_n}(=0.6)$, and Fig. 12d shows that when QoS_{DD} is small, the cases of $k_{max} + 1$ and $k_{max} + 5$ have larger D than the required $QoS_{DD}(=0.15)$. In other words, the cases of $k_{max} + 1$ and $k_{max} + 5$ do not satisfy all QoS requirements. Furthermore, Fig. 12e shows that the cases of $k_{max} - 1$ and $k_{max} - 5$ have smaller T_{Nlife} than the case of k_{max} . In other words, the optimal one is the best among the five cases.

In Fig. 13, all three QoS parameters are fixed, whereas in Figs. 10a, 11a, and 12a, only two QoS parameters are fixed. Fig. 13 shows QoS versus k , where $n = 10,000$, $a = 10,000$, $r = 30$, $L = 1$, and $T = 1$. The optimal k is 32 when QoS requirements are $QoS_{DD} = 0.6$, $QoS_{C_n} = 0.6$, and $QoS_{DD} = 0.15$. Fig. 13a shows that if k is larger than 32, C_n cannot be satisfied, i.e., being smaller than 0.6. Fig. 13b shows that if k is

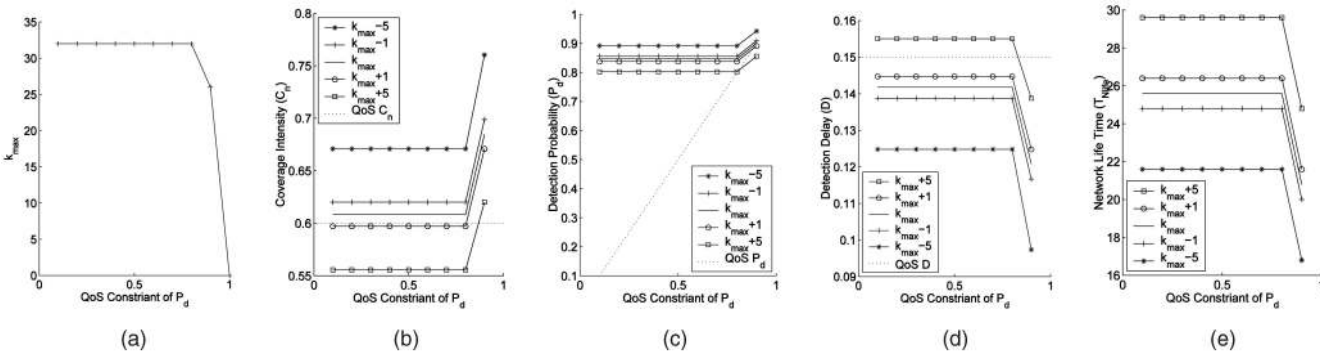


Fig. 11. Comparisons for QoS_{P_d} . (a) Maximum k versus QoS_{P_d} , (b) C_n versus QoS_{P_d} , (c) P_d versus QoS_{P_d} , (d) D versus QoS_{P_d} , and (e) lifetime versus QoS_{P_d} .

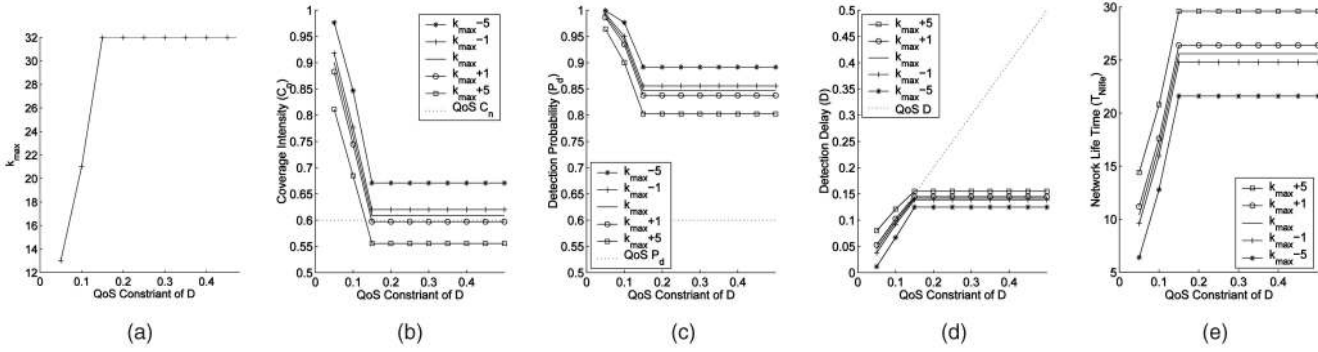


Fig. 12. Optimal performance over QoS_{DD} . (a) Maximum k versus QoS_{DD} , (b) C_n versus QoS_{DD} , (c) P_d versus QoS_{DD} , (d) D versus QoS_{DD} , and (e) lifetime versus QoS_{DD} .

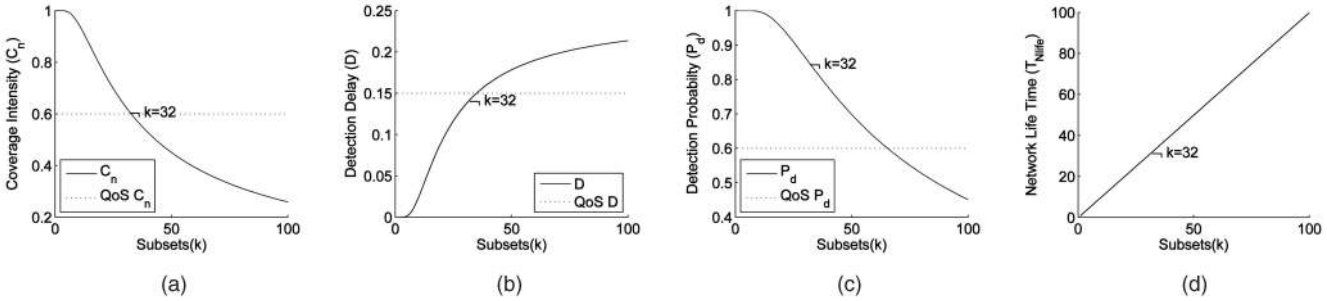


Fig. 13. Optimality. (a) C_n versus k , (b) D versus k , (c) P_d versus k , and (d) lifetime versus k .

larger than 34 (> 32), D cannot be satisfied, i.e., being larger than 0.15. Fig. 13c shows that if k is larger than 65 (> 32), P_d cannot be satisfied, i.e., being smaller than 0.6. In other words, all integers in $[1, 32]$ satisfy QoS requirement for C_n via Fig. 13a; all integers in $[1, 34]$ satisfy QoS requirement for D via Fig. 13b; all integers in $[1, 65]$ satisfy QoS requirement for P_d via Fig. 13c; and $[1, 32] \cap [1, 34] \cap [1, 65] = [1, 32]$ is the set satisfying all three QoS requirements. Clearly $k = 32$ is the maximum k value among all integers in $[1, 32]$, and is the optimal solution. Although Fig. 13d shows that a larger k has a larger T_{Nlife} , for integers larger than 32, at least one QoS constraint cannot be guaranteed.

8 EFFECTS OF SIMULATION DURATION

It is important to determine when we should halt a simulation and calculate the defined performance metrics. The simulation duration (or length) is the number of intrusion detection rounds. We vary the simulation duration in Fig. 14a. For each simulation, we run the simulation for 20 times. Fig. 14a shows that the standard deviation and the coverage intensity obtained from 20 simulations at different simulation duration, where $k = 4$, $a = 10,000$, and $r = 30$. When the simulation duration is small (e.g., 10^1), the standard deviation is large as shown in the figure. When the simulation duration is large enough (e.g., 10^6), regardless of n , C_n from simulation is almost identical to the one from analytical model as shown in Fig. 14a and the standard deviation of the simulation result is so small that it cannot be shown in the figure. Throughout the paper, our simulation duration is chosen no less than 10^6 and we do not plot the standard deviation in our figures.

Fig. 14b compares C_n obtained from both simulations and analytical model with different n and r . In this figure, $k = 4$ and $a = 10,000$. Both simulation and analytical results

match well even regardless of n and r . The figure also shows that as r increases, C_n increases.

9 CONCLUSION

In this paper, we evaluate several issues for a randomized scheduling algorithm in sensor networks through both analysis and simulation. We study network coverage intensity, asymptotic coverage intensity, detection probability, and detection delay. We analyze the problem of maximizing network lifetime under QoS constraints such as the bounded detection delay, detection probability, and coverage intensity. We study properties and asymptotic properties, disclose that the optimal solution exists, and present the conditions of the existence of the optimal solutions. Our results can provide people with better understanding of the network design and parameter selection. This work also lays a foundation for our future work on sensor network scheduling algorithms. Evidently, we have extended partially this work and investigated the properties of randomized scheduling algorithms in sensor networks

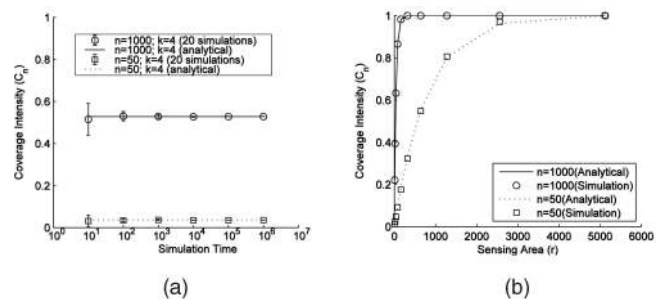


Fig. 14. Effects of simulation duration and sensing area. (a) Effect of simulation duration on simulation results and (b) C_n versus r .

where sensor nodes are deployed either on two-dimensional plane or in three-dimensional space and intrusion objects occupy either areas in a two-dimensional plane or volumes in three-dimensional space, respectively [23].

APPENDIX A

Proofs of Lemmas 1-5 are omitted due to limited space.

PROOFS OF LEMMAS 7-10

Proof of Lemma 7. We first prove the followings: 1) D tends to a function independent of k when k is large enough; 2) we have

$$\lim_{k \rightarrow \infty} D = \frac{\frac{L}{T} \left(\left[\frac{L}{T} \right]^2 - \left[\frac{L}{T} \right] + \frac{L}{T} \right)}{2 \left[\frac{L}{T} \right] \left(\left[\frac{L}{T} \right] + 1 \right)} \left[1 - \left(1 - \frac{r}{a} \right)^n \right].$$

1) We proceed to show that M_j tends to a function independent of k when k is large enough, from which we conclude the same property for D .

When k is large enough, we have in particular $\phi_1 = \lceil \frac{L}{T} \rceil = Q$ and $\phi_2 = \lceil \frac{L}{T} \rceil + 1 = Q + 1$. Then, we have

$$M_j = \frac{(1-s) \sum_{i=2}^Q \left[\binom{k-i+1}{k}^j - \binom{k-i}{k}^j \right] \left(i - 1 - \frac{1-s}{2} \right)}{\sum_{i=1}^Q \left[\binom{k-i+1}{k}^j - \binom{k-i}{k}^j \right]} + \frac{s \sum_{i=2}^{Q+1} \left[\binom{k-i+1}{k}^j - \binom{k-i}{k}^j \right] \left(i - \frac{3}{2} - \frac{1-s}{2} \right)}{\sum_{i=1}^{Q+1} \left[\binom{k-i+1}{k}^j - \binom{k-i}{k}^j \right]}.$$

We note that for each fixed j , all $\binom{k-i+1}{k}^j - \binom{k-i}{k}^j$, where $1 \leq i \leq Q + 1$, are asymptotically equal when k is large enough. To see this, apply the Mean Value Theorem to the function $f(x) = x^j$ in the interval $[\frac{k-i}{k}, \frac{k-i+1}{k}]$. Since $\frac{df(x)}{dx} = jx^{j-1}$, we then obtain

$$\begin{aligned} \left(\frac{k-i+1}{k} \right)^j - \left(\frac{k-i}{k} \right)^j &= \left(\frac{k-i+1}{k} - \frac{k-i}{k} \right) j x_i^{j-1} \\ &= \frac{j}{k} x_i^{j-1} \end{aligned}$$

for some $x_i \in [\frac{k-i}{k}, \frac{k-i+1}{k}]$. All such x_i where $1 \leq i \leq Q + 1$ will tend to 1 as k tends to infinity, since we have $\lim_{k \rightarrow \infty} \frac{k-i}{k} = 1 = \lim_{k \rightarrow \infty} \frac{k-i+1}{k}$. Hence, we know that $\binom{k-i+1}{k}^j - \binom{k-i}{k}^j$ will all tend to $\frac{j}{k}$. From this we obtain, as $k \rightarrow \infty$,

$$\begin{aligned} \frac{\left[\binom{k-i+1}{k}^j - \binom{k-i}{k}^j \right]}{\sum_{i=1}^Q \left[\binom{k-i+1}{k}^j - \binom{k-i}{k}^j \right]} &\approx \frac{1}{Q}, \\ \frac{\left[\binom{k-i+1}{k}^j - \binom{k-i}{k}^j \right]}{\sum_{i=1}^{Q+1} \left[\binom{k-i+1}{k}^j - \binom{k-i}{k}^j \right]} &\approx \frac{1}{Q+1}, \end{aligned}$$

and hence

$$M_j \sim \frac{1-s}{Q} \sum_{i=2}^Q \left(i - 1 - \frac{(1-s)}{2} \right) + \frac{s}{Q+1} \sum_{i=2}^{Q+1} \left(i - \frac{3}{2} - \frac{1-s}{2} \right),$$

which is independent of k .

(2) By the above proof, we have,

$$\begin{aligned} M &\triangleq \lim_{k \rightarrow \infty} M_j = \frac{1-s}{Q} \sum_{i=2}^Q \left[i - 1 - \frac{1-s}{2} \right] \\ &\quad - \frac{s}{Q+1} \sum_{i=2}^{Q+1} \left[i - \frac{3}{2} - \frac{1-s}{2} \right] \\ &= \frac{(Q-1+s)(Q^2-1+s)}{2Q(Q+1)}. \end{aligned}$$

Hence,

$$\begin{aligned} \lim_{k \rightarrow \infty} D &= \lim_{k \rightarrow \infty} \sum_{j=1}^n M_j \binom{n}{j} \left(\frac{r}{a} \right)^j \left(1 - \frac{r}{a} \right)^{n-j} \\ &= M \sum_{j=1}^n \binom{n}{j} \left(\frac{r}{a} \right)^j \left(1 - \frac{r}{a} \right)^{n-j} \\ &= M \sum_{j=0}^n \binom{n}{j} \left(\frac{r}{a} \right)^j \left(1 - \frac{r}{a} \right)^{n-j} - M \binom{n}{0} \left(1 - \frac{r}{a} \right)^n \\ &= M \left[\left(\frac{r}{a} + 1 - \frac{r}{a} \right)^n - \left(1 - \frac{r}{a} \right)^n \right] \\ &= \frac{\frac{L}{T} \left(\left[\frac{L}{T} \right]^2 - \left[\frac{L}{T} \right] + \frac{L}{T} \right)}{2 \left[\frac{L}{T} \right] \left(\left[\frac{L}{T} \right] + 1 \right)} \left[1 - \left(1 - \frac{r}{a} \right)^n \right]. \end{aligned}$$

With the above proof, since

$$\lim_{n \rightarrow \infty} \left(1 - \left(1 - \frac{r}{a} \right)^n \right) = 1,$$

we have

$$\begin{aligned} \lim_{n=km, k \rightarrow \infty} D &= \frac{(Q-1+s)(Q^2-1+s)}{2Q(Q+1)} \lim_{n \rightarrow \infty} \left(1 - \left(1 - \frac{r}{a} \right)^n \right) \\ &= \frac{(Q-1+s)(Q^2-1+s)}{2Q(Q+1)} \\ &= \frac{\frac{L}{T} \left(\left[\frac{L}{T} \right]^2 - \left[\frac{L}{T} \right] + \frac{L}{T} \right)}{2 \left[\frac{L}{T} \right] \left(\left[\frac{L}{T} \right] + 1 \right)}. \end{aligned}$$

Proof of Lemma 8. Recall (18) and (19). Since for any h ,

$$\begin{aligned} B_j(h, m) &\triangleq \sum_{i=h}^m \left[\binom{k-i+1}{k}^j - \binom{k-i}{k}^j \right] \\ &= \left(\frac{k-h+1}{k} \right)^j - \left(\frac{k-m}{k} \right)^j, \end{aligned}$$

we have

$$\begin{aligned} B_j(1, m) &= 1 - \left(\frac{k-m}{k} \right)^j, \\ \frac{1}{B_j(1, m)} &= \frac{1}{1 - \left(\frac{k-m}{k} \right)^j} = \sum_{i=0}^{\infty} \left(\frac{k-m}{k} \right)^{ij}, \\ C_j(m) &\triangleq \sum_{i=2}^m i \left[\binom{k-i+1}{k}^j - \binom{k-i}{k}^j \right] \\ &= \left[\left(\frac{k-1}{k} \right)^j - \left(\frac{k-m}{k} \right)^j \right] \\ &\quad + \sum_{i=1}^{m-1} \left[\left(\frac{k-i}{k} \right)^j - \left(\frac{k-m}{k} \right)^j \right], \end{aligned}$$

and

$$\begin{aligned}
M_j &= \frac{(1-s)[C_j(\phi_1) + B_j(2, \phi_1)(-1 - \frac{1}{2}(1-s))]}{B_j(1, \phi_1)} \\
&\quad + \frac{s[C_j(\phi_2) + B_j(2, \phi_2)(-\frac{3}{2} - \frac{1}{2}(1-s))]}{B_j(1, \phi_2)} \\
&= (1-s) \left\{ \begin{array}{l} \left[\left(\frac{k-1}{k} \right)^j - \left(\frac{k-\phi_1}{k} \right)^j \right] \\ + \sum_{i=1}^{\phi_1-1} \left[\left(\frac{k-i}{k} \right)^j - \left(\frac{k-\phi_1}{k} \right)^j \right] \\ + \left[\left(\frac{k-1}{k} \right)^j - \left(\frac{k-\phi_1}{k} \right)^j \right] \\ (-1 - \frac{1}{2}(1-s)) \end{array} \right\} \sum_{i=0}^{\infty} \left(\frac{k-\phi_1}{k} \right)^{ij} \\
&\quad + s \left\{ \begin{array}{l} \left[\left(\frac{k-1}{k} \right)^j - \left(\frac{k-\phi_2}{k} \right)^j \right] \\ + \sum_{i=1}^{\phi_2-1} \left[\left(\frac{k-i}{k} \right)^j - \left(\frac{k-\phi_2}{k} \right)^j \right] \\ + \left[\left(\frac{k-1}{k} \right)^j - \left(\frac{k-\phi_2}{k} \right)^j \right] \\ (-\frac{3}{2} - \frac{1}{2}(1-s)) \end{array} \right\} \sum_{i=0}^{\infty} \left(\frac{k-\phi_2}{k} \right)^{ij} \\
&= \sum_{i=0}^{\infty} \left\{ (1-s) \left(\frac{k-\phi_1}{k} \right)^{ij} V_a + s \left(\frac{k-\phi_2}{k} \right)^{ij} V_b \right\},
\end{aligned}$$

where

$$\begin{aligned}
V_a &= \left\{ \begin{array}{l} \left[\sum_{h=2}^{\phi_1-1} \left[\left(\frac{k-h}{k} \right)^j - \left(\frac{k-\phi_1}{k} \right)^j \right] \right] \\ + \left[\left(\frac{k-1}{k} \right)^j - \left(\frac{k-\phi_1}{k} \right)^j \right] \\ (2 - 1 - \frac{1}{2}(1-s)) \end{array} \right\}, \\
V_b &= \left\{ \begin{array}{l} \left[\sum_{h=2}^{\phi_2-1} \left[\left(\frac{k-h}{k} \right)^j - \left(\frac{k-\phi_2}{k} \right)^j \right] \right] \\ + \left[\left(\frac{k-1}{k} \right)^j - \left(\frac{k-\phi_2}{k} \right)^j \right] \\ (2 - \frac{3}{2} - \frac{1}{2}(1-s)) \end{array} \right\}.
\end{aligned}$$

So, M_j is the linear sum of the terms $\left[\left(\frac{k-h}{k} \right)^j - \left(\frac{k-m}{k} \right)^j \right] \left(\frac{k-m}{k} \right)^{ij}$, $0 \leq h \leq m-1$, $m \in \{\phi_1, \phi_2\}$, $i \geq 0$, with non-negative coefficients. Therefore,

$$\begin{aligned}
D &= \sum_{j=1}^n M_j \binom{n}{j} \left(\frac{r}{a} \right)^j \left(1 - \frac{r}{a} \right)^{n-j} \\
&= \sum_{j=1}^n \left[\sum_{i=0}^{\infty} U(\dots) \right] \binom{n}{j} \left(\frac{r}{a} \right)^j \left(1 - \frac{r}{a} \right)^{n-j} \\
&= \sum_{i=0}^{\infty} \left[\sum_{j=1}^n U(\dots) \binom{n}{j} \left(\frac{r}{a} \right)^j \left(1 - \frac{r}{a} \right)^{n-j} \right]
\end{aligned}$$

is an infinite sum with non-negative coefficients. Thus, it is sufficient to prove that each term in D ,

$$\begin{aligned}
&\sum_{j=1}^n \left[\left(\frac{k-h}{k} \right)^j - \left(\frac{k-m}{k} \right)^j \right] \left(\frac{k-m}{k} \right)^{ij} \\
&\binom{n}{j} \left(\frac{r}{a} \right)^j \left(1 - \frac{r}{a} \right)^{n-j}
\end{aligned}$$

is a decreasing function. Now we have

$$\begin{aligned}
&\sum_{j=1}^n \left[\left(\frac{k-h}{k} \right)^j - \left(\frac{k-m}{k} \right)^j \right] \left(\frac{k-m}{k} \right)^{ij} \\
&\binom{n}{j} \left(\frac{r}{a} \right)^j \left(1 - \frac{r}{a} \right)^{n-j} \\
&= \sum_{j=0}^n \left[\left(\frac{k-h}{k} \right)^j - \left(\frac{k-m}{k} \right)^j \right] \\
&\left(\frac{k-m}{k} \right)^{ij} \binom{n}{j} \left(\frac{r}{a} \right)^j \left(1 - \frac{r}{a} \right)^{n-j} \\
&= \left(\frac{k-h}{k} \left(\frac{k-m}{k} \right)^i \frac{r}{a} + 1 - \frac{r}{a} \right)^n \\
&\quad - \left(\frac{k-m}{k} \left(\frac{k-m}{k} \right)^i \frac{r}{a} + 1 - \frac{r}{a} \right)^n \\
&= \left(1 - \frac{r}{a} + \frac{r}{a} \left(1 - \frac{m}{k} \right)^i \left(1 - \frac{h}{k} \right) \right)^n \\
&\quad - \left(1 - \frac{r}{a} + \frac{r}{a} \left(1 - \frac{m}{k} \right)^i \left(1 - \frac{m}{k} \right) \right)^n.
\end{aligned}$$

Taking derivative on the term, and noticing that

$$\begin{aligned}
1 - \frac{r}{a} + \frac{r}{a} \left(1 - \frac{m}{k} \right)^i \left(1 - \frac{m}{k} \right) &< 1 - \frac{r}{a} + \frac{r}{a} \left(1 - \frac{m}{k} \right)^i \\
\left(1 - \frac{h}{k} \right) &< 1, \quad 1 \leq h < m,
\end{aligned}$$

we have

$$\begin{aligned}
&\frac{d}{dn} \left[\left(1 - \frac{r}{a} + \frac{r}{a} \left(1 - \frac{m}{k} \right)^i \left(1 - \frac{h}{k} \right) \right)^n \right] \\
&\quad - \left(1 - \frac{r}{a} + \frac{r}{a} \left(1 - \frac{m}{k} \right)^i \left(1 - \frac{m}{k} \right) \right)^n \\
&= \left(1 - \frac{r}{a} + \frac{r}{a} \left(1 - \frac{m}{k} \right)^i \left(1 - \frac{h}{k} \right) \right)^n \\
&\quad \log \left(1 - \frac{r}{a} + \frac{r}{a} \left(1 - \frac{m}{k} \right)^i \left(1 - \frac{h}{k} \right) \right) \\
&\quad - \left(1 - \frac{r}{a} + \frac{r}{a} \left(1 - \frac{m}{k} \right)^i \left(1 - \frac{m}{k} \right) \right)^n \\
&\quad \log \left(1 - \frac{r}{a} + \frac{r}{a} \left(1 - \frac{m}{k} \right)^i \left(1 - \frac{m}{k} \right) \right) \\
&\rightarrow \left(1 - \frac{r}{a} + \frac{r}{a} \left(1 - \frac{m}{k} \right)^i \left(1 - \frac{h}{k} \right) \right)^n \\
&\quad \log \left(1 - \frac{r}{a} + \frac{r}{a} \left(1 - \frac{m}{k} \right)^i \left(1 - \frac{h}{k} \right) \right) \\
&\leq \left(1 - \frac{r}{a} + \frac{r}{a} \left(1 - \frac{1}{k} \right) \right)^n \log \left(1 - \frac{r}{a} + \frac{r}{a} \left(1 - \frac{1}{k} \right) \right) \\
&= \left(1 - \frac{1}{k} \frac{r}{a} \right)^n \log \left(1 - \frac{1}{k} \frac{r}{a} \right) < 0,
\end{aligned}$$

as $n \rightarrow \infty$. By the last inequality above, we can choose a positive integer N such that when $n > N$, every term in D starts to be decreasing. Therefore, D itself is a decreasing function of n when n is large enough.

Proof of Lemma 9. By the formula

$$\begin{aligned}
x^j - y^j &= (x-y)(x^{j-1}y^0 + x^{j-2}y^1 \\
&\quad + \dots + x^1y^{j-2} + x^0y^{j-1}),
\end{aligned}$$

we have

$$\begin{aligned} \left(\frac{k-i+1}{k}\right)^j - \left(\frac{k-i}{k}\right)^j &= \left(\frac{k-i+1}{k} - \frac{k-i}{k}\right) \\ &\sum_{u=0}^{j-1} \left(\frac{k-i+1}{k}\right)^{(j-1)-u} \left(\frac{k-i}{k}\right)^u \\ &= \frac{1}{k} \sum_{u=0}^{j-1} \left(\frac{k-i+1}{k}\right)^{j-u-1} \left(\frac{k-i}{k}\right)^u, \end{aligned}$$

and hence obtain

$$\begin{aligned} M_j &= (1-s) \frac{\sum_{i=2}^{\phi_1} \left[\left(\frac{k-i+1}{k}\right)^j - \left(\frac{k-i}{k}\right)^j \right] \left(i-1 - \frac{(1-s)}{2}\right)}{\sum_{i=1}^{\phi_1} \left[\left(\frac{k-i+1}{k}\right)^j - \left(\frac{k-i}{k}\right)^j \right]} \\ &+ s \frac{\sum_{i=2}^{\phi_2} \left[\left(\frac{k-i+1}{k}\right)^j - \left(\frac{k-i}{k}\right)^j \right] \left(i - \frac{3}{2} - \frac{(1-s)}{2}\right)}{\sum_{i=1}^{\phi_2} \left[\left(\frac{k-i+1}{k}\right)^j - \left(\frac{k-i}{k}\right)^j \right]} \\ &= (1-s) \frac{\sum_{i=2}^{\phi_1} \sum_{u=0}^{j-1} \left(\frac{k-i+1}{k}\right)^{j-u-1} \left(\frac{k-i}{k}\right)^u \left(i-1 - \frac{(1-s)}{2}\right)}{\sum_{i=1}^{\phi_1} \sum_{u=0}^{j-1} \left(\frac{k-i+1}{k}\right)^{j-u-1} \left(\frac{k-i}{k}\right)^u} \\ &+ s \frac{\sum_{i=2}^{\phi_2} \sum_{u=0}^{j-1} \left(\frac{k-i+1}{k}\right)^{j-u-1} \left(\frac{k-i}{k}\right)^u \left(i - \frac{3}{2} - \frac{(1-s)}{2}\right)}{\sum_{i=1}^{\phi_2} \sum_{u=0}^{j-1} \left(\frac{k-i+1}{k}\right)^{j-u-1} \left(\frac{k-i}{k}\right)^u} \\ &\leq (1-s) \frac{\sum_{i=2}^{\phi_1} j \left(\frac{k-i+1}{k}\right)^{j-1} \left(i-1 - \frac{(1-s)}{2}\right)}{\sum_{i=1}^{\phi_1} \sum_{u=0}^{j-1} \left(\frac{k-i+1}{k}\right)^{j-u-1} \left(\frac{k-i}{k}\right)^u} \\ &+ s \frac{\sum_{i=2}^{\phi_2} j \left(\frac{k-i+1}{k}\right)^{j-1} \left(i - \frac{3}{2} - \frac{(1-s)}{2}\right)}{\sum_{i=1}^{\phi_2} \sum_{u=0}^{j-1} \left(\frac{k-i+1}{k}\right)^{j-u-1} \left(\frac{k-i}{k}\right)^u}. \end{aligned}$$

Since

$$\begin{aligned} \sum_{i=1}^m \sum_{u=0}^{j-1} \left(\frac{k-i+1}{k}\right)^{j-u-1} \left(\frac{k-i}{k}\right)^u \\ \geq \left(\frac{k-i+1}{k}\right)^{j-u-1} \left(\frac{k-i}{k}\right)^u \Big|_{i=1, u=0} = 1, \end{aligned}$$

we have, by simply dropping the denominators,

$$\begin{aligned} M_j &\leq (1-s) \sum_{i=2}^{\phi_1} j \left(\frac{k-i+1}{k}\right)^{j-1} \left(i-1 - \frac{(1-s)}{2}\right) \\ &+ s \sum_{i=2}^{\phi_2} j \left(\frac{k-i+1}{k}\right)^{j-1} \left(i - \frac{3}{2} - \frac{(1-s)}{2}\right) \triangleq M'_j. \end{aligned}$$

In order to prove that $\lim_{n \rightarrow \infty} D = 0$, we need to show that for any given $\varepsilon > 0$, there exists an integer $N > 0$ such that $D(n) < \varepsilon$ whenever $n > N$. To this end, we first note that, since $\lim_{j \rightarrow \infty} j \left(\frac{k-i+1}{k}\right)^{j-1} \rightarrow 0$ when $i > 1$, there exists an integer $N_0 > 0$ such that when $4j > N_0$, we have M'_j , hence M_j , is small enough, i.e., $M_j < \frac{\varepsilon}{2}$. On the other hand, since

$$\binom{n}{j} = \frac{n(n-1)\dots(n-j+1)}{1 \cdot 2 \cdot \dots \cdot j} \leq n^j,$$

and $\lim_{n \rightarrow \infty} n^j b^n = 0$ whenever $0 < b < 1$, we know that

$$\lim_{n \rightarrow \infty} \binom{n}{j} \left(1 - \frac{r}{a}\right)^{n-j} = 0$$

for each j such that $1 \leq j \leq N_0$, and, at the same time, M'_j and hence M_j are all bounded for j such that $1 \leq j \leq N_0$. Therefore, there exists an integer $N > N_0$ so that when $n > N$, we have

$$\binom{n}{j} \left(1 - \frac{r}{a}\right)^{n-j} < \frac{\varepsilon}{2N_0 \max\{M_1, \dots, M_{N_0}\}}.$$

We then have

$$\begin{aligned} D &= \sum_{j=1}^n M_j \binom{n}{j} \left(\frac{r}{a}\right)^j \left(1 - \frac{r}{a}\right)^{n-j} \\ &= \sum_{j=1}^{N_0} M_j \binom{n}{j} \left(\frac{r}{a}\right)^j \left(1 - \frac{r}{a}\right)^{n-j} \\ &+ \sum_{j=N_0+1}^n M_j \binom{n}{j} \left(\frac{r}{a}\right)^j \left(1 - \frac{r}{a}\right)^{n-j} \\ &\leq \max\{M_1, \dots, M_{N_0}\} \sum_{j=1}^{N_0} \binom{n}{j} \left(1 - \frac{r}{a}\right)^{n-j} \left(\frac{r}{a}\right)^j \\ &+ \frac{\varepsilon}{2} \sum_{j=N_0+1}^n \binom{n}{j} \left(\frac{r}{a}\right)^j \left(1 - \frac{r}{a}\right)^{n-j} \\ &< \max\{M_1, \dots, M_{N_0}\} \sum_{j=1}^{N_0} \frac{\varepsilon}{2N_0 \max\{M_1, \dots, M_{N_0}\}} \frac{1}{\left(\frac{r}{a}\right)^j} \\ &+ \frac{\varepsilon}{2} \sum_{j=0}^n \binom{n}{j} \left(\frac{r}{a}\right)^j \left(1 - \frac{r}{a}\right)^{n-j} \\ &= \frac{\varepsilon}{2N_0} \sum_{j=1}^{N_0} \left(\frac{r}{a}\right)^j + \frac{\varepsilon}{2} < \frac{\varepsilon}{2N_0} N_0 + \frac{\varepsilon}{2} = \varepsilon. \end{aligned}$$

This implies that $\lim_{n \rightarrow \infty} D = 0$.

Proof of Lemma 10. From the proof of Lemma 8, we know each M_j and hence D are linear sums of the terms

$$\left[\left(\frac{k-h}{k}\right)^j - \left(\frac{k-m}{k}\right)^j \right] / \left[1 - \left(\frac{k-m}{k}\right)^j \right]$$

where $0 \leq h \leq m-1$, $m \in \{\phi_1, \phi_2\}$, $i \geq 0$, with non-negative coefficients which are all independent of k . So, we only need to prove that each such a term is an increasing function of k .

Since

$$\frac{\left[\left(\frac{k-h}{k}\right)^j - \left(\frac{k-m}{k}\right)^j \right]}{\left[1 - \left(\frac{k-m}{k}\right)^j \right]} = \frac{(k-h)^j - (k-m)^j}{k^j - (k-m)^j},$$

we only need to prove that the function

$$f(x) = \frac{[(x-h)^j - (x-m)^j]}{[x^j - (x-m)^j]}$$

is an increasing function when $x \geq m$. To see this, we calculate its derivative as follows:

$$\begin{aligned}
f'(x) &= \frac{\begin{Bmatrix} [(x-h)^j - (x-m)^j]'[x^j - (x-m)^j] \\ -[(x-h)^j - (x-m)^j][x^j - (x-m)^j]' \end{Bmatrix}}{[x^j - (x-m)^j]^2} \\
&= \frac{\begin{Bmatrix} [j(x-h)^{j-1} - j(x-m)^{j-1}][x^j - (x-m)^j] \\ -[(x-h)^j - (x-m)^j][jx^{j-1} - j(x-m)^{j-1}] \end{Bmatrix}}{[x^j - (x-m)^j]^2} \\
&= \frac{\begin{Bmatrix} j(x-h)^{j-1}x^{j-1}h + j(x-h)^{j-1}(x-m)^{j-1}(m-h) \\ -j(x-m)^{j-1}x^{j-1}m \end{Bmatrix}}{[x^j - (x-m)^j]^2} \\
&= \frac{\begin{Bmatrix} j(x-h)^{j-1}x^{j-1}[x - (x-h)] \\ + j(x-h)^{j-1}(x-m)^{j-1}[(x-h) - (x-m)] \\ - j(x-m)^{j-1}x^{j-1}[x - (x-m)] \end{Bmatrix}}{[x^j - (x-m)^j]^2} \\
&= \frac{\begin{Bmatrix} jhm(x-h)^{j-1}(x-m)^{j-2} \\ \sum_{l=0}^{j-2} ((x-m)^{-l} - (x-h)^{-l})x^l \end{Bmatrix}}{[x^j - (x-m)^j]^2}.
\end{aligned}$$

Since $h < m$, we have $x - h > x - m > 0$ when $x > m$. Hence, $(x-m)^{-l} > (x-h)^{-l}$ when $l > 0$. It follows that $f'(x) > 0$ when $x \geq m$ and $j > 1$. Hence, $f(x)$ is an increasing function when $x \geq m$. This finishes the proof of Lemma 10.

ACKNOWLEDGMENTS

This work was supported in part by the US National Science Foundation (NSF) under grants CNS-0716211, CNS-0737325, and CCF-0829827.

REFERENCES

- [1] Z. Abrams, A. Goel, and S. Plotkin, "Set k-Cover Algorithms for Energy Efficient Monitoring in Wireless Sensor Networks," *Proc. Int'l Symp. Information Processing in Sensor Networks (IPSN '04)*, 2004.
- [2] C. Hsin and M. Liu, "Network Coverage Using Low Duty-Cycled Sensors: Random & Coordinated Sleep Algorithm," *Proc. Int'l Symp. Information Processing in Sensor Networks (IPSN '04)*, 2004.
- [3] S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M. Srivastava, "Coverage Problems in Wireless Ad-Hoc Sensor Networks," *Proc. IEEE INFOCOM*, 2001.
- [4] D. Tian and D. Georganas, "A Coverage-Preserving Node Scheduling Scheme for Large Wireless Sensor Networks," *Proc. Int'l Workshop Wireless Sensor Networks and Applications (WSNA '02)*, 2002.
- [5] K. Wu, Y. Gao, F. Li, and Y. Xiao, "Lightweight Deployment-Aware Scheduling for Wireless Sensor Networks," *ACM/Springer Mobile Networks and Applications*, special issue on Energy Constraints and Lifetime Performance in Wireless Sensor Networks, vol. 10, no. 6, pp. 837-852, Dec. 2005.
- [6] T. Yan, T. He, and J. Stankovic, "Differentiated Surveillance for Sensor Networks," *Proc. ACM Int'l Conf. Embedded Networked Sensor Systems (SenSys '03)*, 2003.
- [7] F. Ye, G. Zhong, J. Cheng, S. Lu, and L. Zhang, "Peas: A Robust Energy Conserving Protocol for Long-Lived Sensor Networks," *Proc. IEEE Int'l Conf. Network Protocols (ICNP '02)*, 2002.
- [8] L. Wang and Y. Xiao, "A Survey of Energy-Efficient Scheduling Mechanisms in Sensor Networks," *ACM/Springer Mobile Networks and Applications*, vol. 11, no. 5, pp. 723-740, Oct. 2006.
- [9] C. Liu, K. Wu, Y. Xiao, and B. Sun, "Random Coverage with Guaranteed Connectivity: Joint Scheduling for Wireless Sensor Networks," *IEEE Trans. Parallel and Distributed Systems*, vol. 17, no. 6, pp. 562-575, June 2006.

- [10] Y. Xiao, H. Chen, K. Wu, C. Liu, and B. Sun, "Maximizing Network Life Time under QoS constraints in Wireless Sensor Networks," *Proc. GLOBECOM*, 2006.
- [11] A. Cerpa and D. Estrin, "Ascent: Adaptive Self-Configuring Sensor Networks Topologies," *Proc. IEEE INFOCOM*, 2002.
- [12] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris, "Span: An Energy-Efficient Coordination Algorithm for Topology Maintenance in Ad Hoc Wireless Networks," *Proc. Mobicom*, 2001.
- [13] E. Elson and K. Romer, "Wireless Sensor Networks: A New Regime for Time Synchronization," *Proc. First Workshop Hot Topics in Networks*, Oct. 2002.
- [14] P. Godfrey and D. Ratajczak, "Robust Topology Management in Wireless Ad Hoc Networks," *Proc. Int'l Symp. Information Processing in Sensor Networks (IPSN '04)*, 2004.
- [15] H. Gupta, S. Das, and Q. Gu, "Connected Sensor Cover: Self-Organization of Sensor Networks for Efficient Query Execution," *Proc. MobiHoc*, 2003.
- [16] S. Ren, Q. Li, H. Wang, X. Chen, and X. Zhang, "Design and Analysis of Sensing Scheduling Algorithms under Partial Coverage for Object Detection in Sensor Networks," *IEEE Trans. Parallel and Distributed Systems*, vol. 18, no. 3, pp. 334-350, Mar. 2007.
- [17] C. Schurgers, V. Tsiatsis, S. Ganeriwal, and M. Srivastava, "Topology Management for Sensor Networks: Exploiting Latency and Density," *Proc. MobiHoc*, 2002.
- [18] S. Shakkottai, R. Srikant, and N. Shroff, "Unreliable Sensor Grids: Coverage, Connectivity and Diameter," *Proc. IEEE INFOCOM*, 2003.
- [19] S. Slijepcevic and M. Potkonjak, "Power Efficient Organization of Wireless Sensor Networks," *Proc. IEEE Int'l Conf. Comm. (ICC '01)*, 2001.
- [20] S. Tilak, N. Abu-Ghazaleh, and H.W., "Infrastructure Tradeoffs for Sensor Networks," *Proc. Int'l Workshop Wireless Sensor Networks and Applications (WSNA '02)*, 2002.
- [21] X. Wang, G. Xing, Y. Zhang, C. Lu, R. Pless, and C. Gill, "Integrated Coverage and Connectivity Configuration in Wireless Sensor Networks," *Proc. ACM Int'l Conf. Embedded Networked Sensor Systems (SenSys '03)*, 2003.
- [22] H. Zhang and J. Hou, "Maintaining Coverage and Connectivity in Large Sensor Networks," *Proc. Int'l Workshop Theoretical and Algorithmic Aspects of Sensor and Ad-Hoc Networks (WTASA '04)*, 2004.
- [23] Y. Xiao, Y. Zhang, M. Peng, H. Chen, X. Du, B. Sun, and K. Wu, "Two and Three Dimensional Intrusion Object Detection under Randomized Scheduling Algorithms in Sensor Networks," *Computer Networks*, vol. 53, no. 14, pp. 2458-2475, Sept. 2009.



member of the IEEE.



Yang Xiao (SM'04) is currently with the Department of Computer Science at the University of Alabama. He serves as an associate editor for several journals, e.g., *IEEE Transactions on Vehicular Technology*. His research areas are security, telemedicine, robot, sensor networks, and wireless networks. He has published more than 300 papers in major journals, refereed conference proceedings, and book chapters related to these research areas. He is a senior

Hui Chen (M'06) studied geophysics and computer science, and worked in industry. He is currently with the Department of Mathematics and Computer Science, Virginia State University. He primarily works in the area of computer networking. He served as journal guest editors and various IEEE conference program committees, and publishes frequently. He is a member of the IEEE.



Kui Wu received the PhD degree in computing science from the University of Alberta, Canada, in 2002, and joined the Department of Computer Science, University of Victoria, Canada, in 2002, where he is currently an associate professor. His research interests include performance analysis and protocol design of computer networks, wireless sensor networks, and network security. He is a senior member of the IEEE.



Xinyu Sun received the BSc degree in 1989 from Jilin University (Changchun, China), the MSc degree in 1995, and the PhD degree in 2004 from Temple University, majoring in mathematics. He spent three years in Texas A&M University as a visiting assistant professor from 2004 to 2007. He is a visiting professor in the Department of Mathematics at Tulane University.



Bo Sun is an assistant professor with the Department of Computer Science, Lamar University, Beaumont, Texas. His research interests include security issues of wireless networks and other communications systems. His research has been supported by the National Science Foundation and the 2006 Texas Advanced Research Program. He is a member of the IEEE.



Chong Liu received the PhD degree in the Computer Science Department from the University of Victoria, Canada, in 2006. He then joined Research In Motion Limited. His research interests include energy efficient node clustering, scheduling and data retrieval in sensor networks, corporation data, and application integration to mobile handheld. He is a student member of the IEEE.



Ying Zhang received the BSc degree in 1989 from Jilin University (Changchun, China), the MSc degree in 2001, and the PhD degree in 2005, from the National University of Singapore, majoring in mathematics. His research interest lies in low dimensional topology and geometry. He is currently a professor of mathematics in Soochow University (Suzhou, China).

▷ For more information on this or any other computing topic, please visit our Digital Library at www.computer.org/publications/dlib.