Probabilistic Life Time Maximization of Sensor Networks

Hassan Jaleel, Amir Rahmani and Magnus Egerstedt

***Abstract*—Power aware sensor networks has been an active area of research in the last decade and a lot of solutions have been proposed for maximizing the lifetime of a sensor network. However, an underlying assumption in most of the existing work is that the performance of a sensor remains the same throughout its lifetime, which is not always true. In this paper, we identify the problem of the effects of power decay on the performance of an individual sensor and of the entire network. In particular, we examine networks with decaying footprints, akin to those of RF- or radar-based sensors and relate the performance of a sensor to its available power. Moreover, we propose probabilistic scheduling controllers that compensate for the effects of decrease in power while maintaining an adequate probability of event detection under two sensing models, namely, Boolean and non-Boolean. Finally we perform Monte Carlo simulations, which verify that our proposed controllers maintain the desired performance level throughout the lifetime of a network.**

I. Introduction

A sensor network consists of a large number of sensors that are typically low-cost, low-power devices with limited sensing, processing, and communication capabilities. Because of their low-power requirement, sensors are normally powered by batteries, which deplete with time. Unfortunately, the replacement of batteries in a large number of sensors is virtually impossible, particularly in an inaccessible or potentially hostile environment, so the lifetime of each sensor is limited ([1] and [5]). Therefore, a critical problem, which is a subject of active research in the wireless sensor networks community, is power conservation. One approach to conserving power is to turn sensors *off* when they are not needed. However, such an approach is risky because critical events can be missed and information lost while the sensor is *off*. Other approaches have been proposed, ([2], [3], [6], [7], [13], [15], and [17]), to name just a few.

Despite the numerous approaches to conserving power that have been proposed, one issue that still requires attention is the effect of power decay on both individual sensor performance and the entire network. Obviously, the extent of the impact depends on the type of devices used. For example, if a system consists of vision-based sensors, power levels may be related to the maximally available frame rate; for RF- or radar-based sensors, the footprint area may be reduced with a reduction in power; and in any system, latency issues may arise across the network. Thus, in this paper, we first explicitly couple the performance of a sensor network comprised of RF- or radar-based sensors with power consumption. Note that this choice of sensors is simply a starting point for more comprehensive future research. After establishing this relationship, we use it during the network design phase to ensure that the network maintains a desired level of performance through out its lifetime regardless of the adverse effects of power consumption.

The network that we examine in this paper consists of randomly-deployed sensors in a region of interest. Random deployment allows us to use well-established tools from probability theory and stochastic geometry for the design and the analysis of our system. From probability theory, we model random deployment as a stationary spatial Poisson point process, and from stochastic geometry, we model sensor coverage using the well-known germ-grain model. After we represent our system with these models, we design a controller that schedules the duty cycle of sensors so that we can ensure a constant probability of event detection throughout the lifetime of the network in the presence of power decay. Here, we denote the probability of event detection as our desired performance criterion. We start with common Boolean sensing scheme (i.e., an event is detected only when in the footprint of a sensor) and derive sensor scheduling control laws for persistent and non-persistent events. We also propose non-Boolean sensing model, which is more closely related to practical systems, and derive a similar scheduling scheme. The preliminary findings along these lines were presented in [10].

II. System Description

Consider a domain $\mathcal{D} \subset \mathbb{R}^2$ in which a large number of sensors are randomly deployed such that the location of each sensor is independent of the locations of all the other sensors. For example, such a scenario can arise when sensors are dropped from the air into a region of interest. From [16], we know that this sensor deployment can be modeled as a stationary spatial Poisson point process with constant intensity λ (the expected number of sensors in a unit area). Given a set in $\mathcal{D}$ with area A, the probability of having n sensors in this area is given by

$$P_n(A) = \frac{(\lambda A)^n e^{-\lambda A}}{n!}. \tag{1}$$

To analyze the random deployment of sensors, we assume the following about our system. We first assume that the total number of sensors in the region of interest goes to

Hassan Jaleel and Magnus Egerstedt are with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, Georgia; Emails: {*hjaleel3,magnus*}*@ece.gatech.edu*. Amir Rahmani is with the Department of Mechanical and Aerospace Engineering, University of Miami; Email: *a.rahmani@miami.edu*

This work is partially supported by the Office of Naval Research through the MURI-HUNT.

infinity. This assumption is for the purpose of analysis only, for the results presented in this paper are still applicable to any practical system comprising a sufficiently large number of sensors with identical battery power and sensing capabilities when deployed. Secondly, we take only the sensing capability of the network into account, and we are not concerned with the communication among sensors. Finally, we assume that all sensors are RF or radar based. Therefore, the footprint of each sensor i is a closed ball, $\mathcal{B}_{r(t)}(x_i)$, of radius $r(t)$, centered at x_i, which is the position of the sensor. In addition to these assumptions, we define the footprint of a sensor as the region in which a sensor can detect any event and communicate with other sensors. The union of all these footprints, centered at the locations of the corresponding sensors, form the famous germ-grain model of stochastic geometry in which sensors are the germs and their footprints are the grains ([8] and [16]). In the rest of the paper, we will use the terms "grains" and "footprints" interchangeably.

To conserve power, we let the sensors be *on* with probability q. Each sensor can switch its state from *on* to *off* or vice versa only at discrete time instances $k\Delta t$ (or simply at instance k), where Δt is the sample time. The state of a sensor at instance k is maintained throughout interval $[k, k+1)$ of length Δt. A sensor can sense only when it is *on*, and for an event to be detected, it should be within the footprint of at least one *on* sensor. One obvious fact is that when a sensor is *on*, it consumes power, and its battery is depleted. Using the discrete time version of the battery dynamics in [14], we model the power of each sensor in the *on* state using the following dynamics

$$\eta(k+1) = \eta(k) - \Delta t \gamma \eta(k),$$

where γ is the decay constant and $\eta(k)$ represents the remaining battery power available for sensing at time instant k. We define a switching signal $\sigma(k)$ as

$$\sigma(k) = \begin{cases} 1 & \text{if a sensor is } \textit{on} \text{ at time instant } k \\ 0 & \text{if a sensor is } \textit{off} \text{ at time instant } k \end{cases}$$

Since a sensor is *on* with probability q, the expected value of $\sigma(k)$ is $E\{\sigma(k)\} = \hat{\sigma}(k) = q(k)$. We know that power is consumed only when a sensor is *on*, so we can modify the power model as

$$\eta(k+1) = \eta(k) - \Delta t \gamma \sigma(k) \eta(k). \tag{2}$$

Since $\hat{\sigma}(k) = q(k)$ and $\sigma(i)$ is independent of $\sigma(j)$ for all $i \neq j$, the expected power level of each sensor is

$$\hat{\eta}(k+1) = \left[\prod_{i=0}^{k} (1 - \Delta t \gamma q(i))\right] \eta(0). \tag{3}$$

Moreover, for all $t \in [k, k+1)$, we assume that $\eta(t) = \eta(k)$.[1]

[1]We should note that in this analysis, we are not considering the potential power consumption resulting from switching between the *on* and *off* states.

III. Probability of Event Detection

Consider a non-persistent event that takes place at some point $x_e \in \mathcal{D}$ at some arbitrary time $t \in [k, k+1)$. A non-persistent, which a sensor can detect only at the time of its occurrence, does not leave a mark in the environment. Hence, this event is detected if it is within the footprint of at least one sensor in the *on* state at time k. We first investigate the probability of event detection for a non-persistent event in a non-decaying sensor network (i.e., a network of sensors whose footprints and probabilities q do not change with time). From [9], we know that the probability of an event going undetected by a non-decaying sensor network deployed randomly with intensity λ is

$$P_u = e^{-\lambda A q}, \tag{4}$$

where A is the area of the sensor footprint with radius r and q is the probability of a sensor being *on*. The proof of Equation (4) is based on the observation that the probability of an event going undetected is equal to the sum of probabilities of the event being inside the range of $n \in [0, \infty]$ sensors, all of which are *off*.

After examining a non-decaying network, we take into account a decaying network in which the power of the sensors is consumed when they are *on*, resulting in a decrease in the area of the sensor footprints, which is proportional to the decay in power [4]. In [12], it was shown that if the sensor range model is based on the RF-power-density function for an isotropic antenna, the sensor footprint is proportional to the available power of the sensor node, i.e.,

$$r^2(t) \propto \eta(t), \tag{5}$$

where $r(t)$ is the radius of the sensor footprint at time $t \in [k, k+1)$. Hence, the area of footprint of a sensor at time t is

$$A(t) = \pi r^2(t) = \alpha \eta(t), \tag{6}$$

where $\alpha = \zeta\pi$ is a constant with ζ being the constant of the proportionality in Equation (5). If we substitute the power, η, in Equation (6) with the expected power, $\hat{\eta}$, from Equation (3), we will get the expected footprint area of a sensor which is

$$\hat{A}(k) = c \left[\prod_{i=0}^{k-1} (1 - \Delta t \gamma q(i))\right], \tag{7}$$

where $c = \alpha \eta(0)$ is a constant.

Lemma 3.1: *The probability of an event being detected by a decaying sensor network is given by*

$$P_d(k) = 1 - e^{-\lambda \hat{A}(k) q(k)}, \tag{8}$$

where $\hat{A}(k)$ is the expected footprint area of all the sensors.

Proof: From Equation (4), we know that an event at $x_e \in \mathcal{D}$ is detected in a non-decaying sensor network if at least one sensor in the *on* state is present in $\mathcal{B}_r(x_e)$, where r is the radius of the sensor footprint. For a decaying network, this reasoning cannot be applied directly. Although all sensors are initially identical, we have no reason to believe that the battery power and the footprint areas are the same throughout the network at any time $k \neq 0$ because of sensor switching and power decay.

From stochastic geometry [8], we know that the probability of any given point $x \in \mathcal{D}$ not being covered by set $\bigcup_i B_{r_i}(x_i)$ in the germ-grain model is

$$P(x \text{ not covered}) = e^{-\lambda \hat{A}}, \tag{9}$$

where $B_{r_i}(x_i)$ is the grain corresponding to the germ x_i with area A_i and $\hat{A}$ is the expected area of grains $B_{r_i}(x_i)$ for all i. Our scenario slightly differs from that in [8], because in our system, a sensor is *on* only with probability $q(k)$ at any instance k. Therefore, even if $x \in B_{r_i}(x_i)$ for any arbitrary i, it may still not be covered since that sensor can be *off*. As a result, the probability of an event being undetected, P_u, can be obtained by updating Equation (4) as

$$P_u(k) = e^{-\lambda \hat{A}(k) q(k)}. \tag{10}$$

Note that, we are using the expected area of a grain, $\hat{A}(k)$, instead of $A(k)$. Finally, to conclude the proof, we substitute the value of $\hat{A}(k)$ in Equation (10) with Equation (7) and use relationship $P_d = 1 - P_u$. ■

Equation (8) confirms that, if the probability of sensors being *on*, q, is constant then the chance of an event being detected, P_d, clearly decreases with time.

IV. Duty Cycle Scheduling for Constant Event Detection Probability

A key requirement in many practical applications of sensor networks is to maintain a minimum satisfactory probability of event detection. In order to maintain the desired probability of event detection, we propose a controller that adjusts $q(k)$, the probability of a sensor being on at time k, which is the main goal of this paper. In other words, we wish to find $u(k) \in [0, 1]$ such that

$$q(k+1) = u(k), \tag{11}$$

yields a scheduling scheme for the duty cycle of sensors that maintains the probability of event detection for the overall network at a desired level.

Definition 4.1: *The desired network performance, P_{des}, is the minimum satisfactory probability of an event being detected.*

Consider a case in which a desired probability of event detection is the given performance parameter P_{des}. Hence, $\beta = 1 - P_{des}$ is the probability of an event going undetected. The duty cycle scheduling of sensors can help maintain such performance in the presence of decreasing sensor power.

Theorem 4.1: [Main Result] *A feedback scheduling controller of the form*

$$u(k) = \min\left\{1, \frac{1}{1 - \Delta t \gamma q(k)} q(k)\right\}, \tag{12}$$

will guarantee that the desired network performance is maintained for the lifetime of the network.

Proof: Using the result of Lemma 3.1, we have

$$\left[\prod_{i=0}^{k-1} (1 - \Delta t \gamma q(i))\right] q(k) = \frac{\ln(\frac{1}{\beta})}{\lambda c}. \tag{13}$$

Replacing the value of k in the above equation with 0, we can compute the initial value of q as

$$q(0) = \frac{\ln(\frac{1}{\beta})}{\lambda c}. \tag{14}$$

Rearranging the terms of Equation (13) results in a feedback controller for $q(k)$ as

$$q(k+1) = \frac{1}{1 - \Delta t \gamma q(k)} q(k). \tag{15}$$

Since the input of the controller is a probability, it cannot have a value greater than 1. Taking into account this fact, our proposed probabilistic scheduling controller takes on the form

$$u(k) = \min\left\{1, \frac{1}{1 - \Delta t \gamma q(k)} q(k)\right\}.$$

As long as $u(k)$ is less than 1, $q(k)$ evolves according to Equation (15) and the desired performance is maintained. However, the lifetime of the sensor network is completer when $u(k)$ reaches its maximum value, which is proved in the next Lemma. Now, solving resulting controlled dynamical Equation (15) with initial condition (14) produces an expression for $q(k)$ as

$$q(k) = \min\left\{1, \frac{-1}{\gamma k \Delta t + \frac{\lambda c}{\ln(\beta)}}\right\}. \tag{16}$$

■

Note that Equation (16) provides a scheduling strategy for the duty cycle of sensors, which guarantees a constant probability of event detection. However, this desired probability can be maintained for a limited time only.

Theorem 4.2: *The maximum achievable event detection probability in a sensor network with given spatial distribution intensity λ is $1 - e^{(-\lambda c)}$.*

Proof: Consider Equation (14), which yields the initial probability of a sensor being in the *on* state. This probability should always remain inside interval $[0, 1]$. Since $\beta \in [0, 1]$, it is guaranteed that

$$0 \leq q(0) = \frac{\ln(\frac{1}{\beta})}{\lambda c} \leq 1,$$

for all given β, λ, and c. To ensure that $q(0) \leq 1$, the condition

$$\beta \geq e^{-\lambda c},$$

must be satisfied. Hence, $P_{des} \leq 1 - e^{-\lambda c}$. ■

Definition 4.2: *The lifetime of the sensor network is the maximal time beyond which the desired network performance cannot be achieved.*

Characterizing the lifetime of sensor network is essential to its design.

Lemma 4.3: *The lifetime of the sensor network with desired event detection probability* P_{des} *is given by* $\frac{-1}{\gamma}\left(1+\frac{\lambda c}{\ln(1-P_{des})}\right)$.

Proof: At the end of the lifetime of a sensor network, all sensors should be *on*, i.e., $q(k_f) = 1$, where k_f denotes the final time instance. Suppose $q(k_f) \neq 1$, which suggests that one of the sensors can still be in the *off* state. This implies that, turning this sensor *on* will increase the detection probability by an amount equal to the probability of an event being in its footprint. This increase in the detection probability will in turn increase the lifetime of the sensor network, which results in a contradiction since we have already assumed that the lifetime of the network has ended. Therefore, we have proven that $q(k_f) = 1$ at the end of the lifetime of a network. Substituting $q(k_f)$ in Equation (16) by 1 yields $k_f\Delta t = \frac{-1}{\gamma}\left(1+\frac{\lambda c}{\ln(1-P_{des})}\right)$. ■

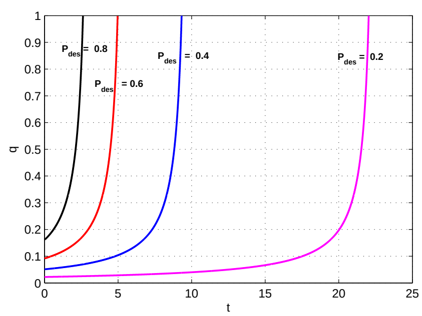

Fig. 1. The evolution of the probability of a sensor being *on* for given desired performance P_{des} where $\lambda = 10$, $c = 1$, and $r(0) = 2$. In each case, the lifetime of the network is achieved when $q = 1$.

Figure (1) depicts how the duty cycle of sensors (the probability of sensors being *on*) needed to maintain a constant event detection probability, varies with time. For a constant event detection probability, P_{des}, the lifetime of the network is achieved when all sensors are turned *on*, as is shown in the proof of Lemma 4.3. Moreover, as P_{des} increases, the lifetime of the network decreases.

Corollary 4.4: *Given a desired lifetime of the sensor network,* t_f*, the maximum probability of event detection that can be maintained in time interval* $[0, k_f]$ *is* $P_d = 1 - e^{\frac{-\lambda c}{1+\gamma k_f \Delta t}}$, *where* $t_f \in [k_f, k_f + 1)$.

Proof: As mentioned in the proof of Lemma 4.3, at the end of the lifetime of a sensor network, all nodes are *on* to maintain the desired network performance, i.e., $q(k_f) = 1$. Therefore, we substitute $q(k_f)$ in Equation (16) with 1, which results in

$$\gamma k_f \Delta t + \frac{\lambda c}{\ln(\beta)} = -1.$$

Finally, we solve the above equation for β and replace it with $1 - P_d$, which conclude the proof. ■

V. Simulations

To confirm the validity of the proposed duty cycle scheduling strategy, we implement a Monte Carlo simulation of a sensor network that is deployed randomly. For this simulation, we consider a 10 by 10 unit rectangular area with $A_{total} = 100$. Sensors are deployed in this area according to a spatial stationary Poisson point process with constant intensity per unit area of $\lambda = 10$. This means that the expected number of sensors in the area of interest is $\lambda A_{total} = 1000$. The initial footprint of each sensor is set to be a closed ball of the unit radius centered at the position of the sensor. Events are generated randomly at each time instant throughout the area of interest. To increase the accuracy of the results, each value of P_d is averaged over 100 iterations of simulation.

In order to ensure that a decaying network maintains the desired performance throughout its lifetime, we need to vary q according to Equation (16) as is shown in Figure (1). The effects of varying q according to Equation (16) are depicted in Figure (2), which illustrates the simulation results for a decaying network with and without our proposed scheduling scheme. We set the desired network performance $P_{des} = 0.63$ (constant dashed line). First, we simulate the system without applying our proposed scheduling scheme and plot the probability of event detection (decaying dashed line). Then, we apply our proposed scheduling scheme and simulate the system again. The results for the later simulation (solid line) reveal that the probability of event detection is $P_d \approx 0.62$. The simulated event detection probability, P_d, is very close to the desired performance, P_{des}, which indicates the validity of our scheme. Moreover, the improvement in the performance measure because of our proposed scheme is obvious by comparing the plots for the two simulations (solid line vs decaying dashed line).

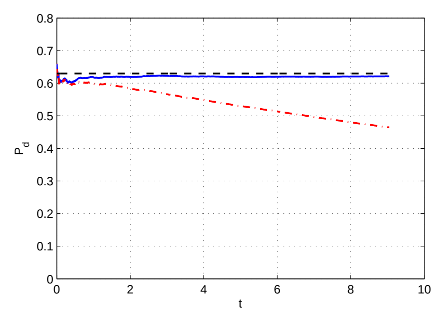

Fig. 2. Event detection probability P_d vs time t for decaying networks with given $P_{des} = 0.63$ (constant dashed line); with the scheduling scheme (solid line) and without the scheduling scheme (decaying dashed line). Here $\lambda = 10$, $A(0) = 1$, $\gamma = 1$, and $c = 1$.

VI. Detection Probability for Persistent Events

Until now, we have been analyzing a non-persistent event, which a sensor can detect only at the time of its occurrence. However, in practical systems, we regularly encounter events that persist for some time duration t_{ev}. It is intuitive that the probability of detection of a persistent event is more than a non-persistent event, and this probability must increase with the increase in t_{ev}. However, this relationship between the probability of event detection and t_{ev} cannot be linear because of the shrinking footprints. Therefore, in this section we find the exact event detection probability for persistent events.

Let t_{ev} be the total time for which an event persists and P_{dp} be its detection probability. If $t_{ev} < \Delta t$ and $t_{ev} \subset [k, k+1)$, then $P_{dp} = P_d(k)$. The two probabilities are equal because Δt is the sample time for which a sensor remains *on* or *off*. Now, consider a case in which an event persists for two time slots. Then,

$$\begin{aligned} P_{dp} &= 1 - P \text{ (event is undetected in both slots k and k+1)}, \\ &= 1 - P_u(k) P_u(k+1|k), \end{aligned}$$

where

$$P_u(k) = \sum_{n_1=0}^{\infty} (1 - q(k))^{n_1} \frac{(\lambda \hat{A}(k))^{n_1} e^{-\lambda \hat{A}(k)}}{n_1!}.$$

In the above equation, the number of sensors that can detect the event during interval $[k, k+1)$ is n_1, which ranges from 0 to ∞ and cannot increase over the subsequent time intervals because the footprints of all sensors decrease with time. Therefore, the number of sensors that can detect the event during interval $[k+1, k+2)$ is at maximum n_1. As a result,

$$P_u(k+1|k) = \sum_{n_2=0}^{n_1} (1 - q(k+1))^{n_2} \frac{(\lambda \hat{A}(k+1))^{n_2} e^{-\lambda \hat{A}(k+1)}}{n_2!}.$$

Now, we can generalize the above equation for j time slots

$$P_{dp} = 1 - \prod_{i=0}^{j-1} P_u(k+i|k+(i-1)).$$

Thus,

$$P_{dp} = 1 - \prod_{i=0}^{j-1} \left[\sum_{n_i=0}^{n_{i-1}} (1 - q(k+i))^{n_i} P_{n_i}(\hat{A}(k+i)) \right], \quad (17)$$

where from Equation (1), $P_{n_i}(\hat{A}(k+i))$ is the probability of having n_i sensors in $\hat{A}(k+i)$.
For $i = 0$, the number of sensors in the sensing region ranges from 0 to ∞.

VII. Event Detection Probability for a Non-Boolean Sensing Model

All the results in the previous sections were derived for Boolean sensing model, i.e., an event is either detected with probability one, if it is within the footprint of a sensor that is *on*, or is undetected. In fact, if r is the radius of the footprint of a sensor, then an event is detected if it occurs at a distance of $r - \epsilon$ from a sensor, and it is undetected if it occurs at a distance of $r + \epsilon$ (for any arbitrary $0 < \epsilon < r$) from a sensor. Therefore, the Boolean sensing model is a relatively simple model, which may not be applicable to all the physical systems. In this section we will analyze a more realistic sensing scheme in which the probability of event detection is a function of the distance from the sensor location.

Let l be the distance of an event from a sensor. Then, the probability of event detection increases as l decreases and vice versa, which means that the probability of an event being detected is a direct function of l, i.e., $P_d(k) \propto \alpha(l, k)$. Here, $\alpha(l, k)$ relates the probability of event detection to the distance of an event from a sensor, l, and the expected power level of a sensor at instance k, $\hat{\eta}(k)$. This relationship depends on type of the sensing devices being used and can be described in various forms. In this paper we define $\alpha(l, k)$ as

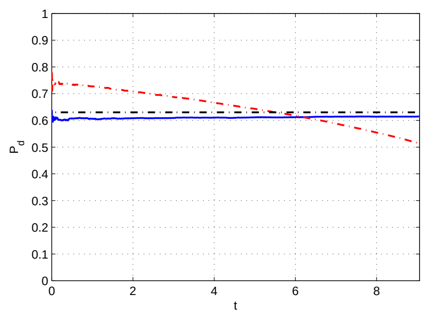

Fig. 3. Comparision of Boolean sensing and non-Boolean sensing under the scheduling scheme (15). Here $s = 2$ for non-Boolean sensing. $P_{des} = 0.63$ (constant dashed line); Boolean sensing (solid line) and non-Boolean sensing (decaying dashed line). Here $\lambda = 10$, $A(0) = 1$ and $\gamma = 1$.

$$\alpha(l, k) = e^{-\frac{s}{\hat{\eta}(k)} l}. \quad (18)$$

According to this model, the probability of event detection decrease exponentially as result of increase in l or decrease in $\hat{\eta}$. The third parameter in the model is s, which is a constant defining the rate of decay of the event detection probability. Even though, the model defined in Equation (18) is just one choice, but it relates the parameters of interest in a logical manner and is close to the behaviour of many sensors of interest.

Next, we design a scheduling scheme for non-Boolean sensing. One possible solution is to use controller (12), which was designed for Boolean sensing. This approach can be verified easily by including non-Boolean sensing scheme in Monte Carlo simulation of Section V. The results are presented in Figure (3), which clearly proves that our previous controller does not maintain constant event detection probability under non-Boolean sensing. Thus, using the similar concept as in [18], we propose a new scheme for maintaining desired performance under non-Boolean sensing:

Theorem 7.1: *For a non-Boolean sensing model with $\alpha(l, k)$ as defined in Equation (18), the probability of event detection for a non-persistent event is*

$$P_d(k) = 1 - e^{-\lambda \hat{s} \left[\prod_{i=0}^{k-1} (1 - \Delta t \gamma q(i))\right]^2 q(k)} \quad (19)$$

where $\hat{s} = \frac{2\pi \eta(0)^2}{s^2}$.

The proof is similar to that of Theorem 4.1 and can be found in [11].

By following the same technique as used for Boolean sensing in Section IV, we design a controller that maintains the desired network performance, P_{des}, for non-Boolean sensing. From Equation (19) we know that

$$\left[\prod_{i=0}^{k-1}(1-\Delta t\gamma q(i))\right]^2 q(k)=\frac{ln(\frac{1}{\beta})}{\lambda\hat{s}}. \quad (20)$$

From the above expression, we can find the initial value of q as

$$q(0)=\frac{ln(\frac{1}{\beta})}{\lambda\hat{s}}.$$

Rearranging the terms in Equation (20) yields the dynamics of $q(k)$ as

$$q(k+1)=\min\left\{1,\frac{1}{(1-\Delta t\gamma q(k))^2}q(k)\right\}. \quad (21)$$

Hence, we have a non-linear control law for duty cycle scheduling that maintains the desired performance measure throughout the lifetime. To verify the validity of our proposed scheme, we run a series of Monte Carlo simulations. The set up used for simulations is the same as was described in Section V. Figure (4) depicts the results for non-Boolean sensing. The proposed scheme maintains the desired performance throughout the lifetime of the network. The lifetime of the network in this case depends also on the parameter s and can be found numerically from Equation (21). As s increases, lifetime decreases and vice versa. Here we set $s=2$.

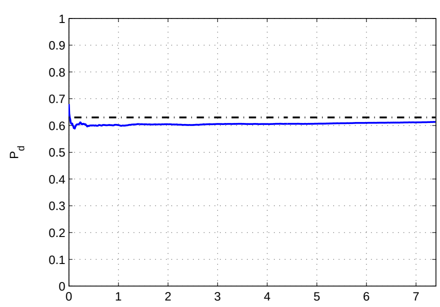

Fig. 4. Event detection probability P_d (solid line) vs time t for decaying networks and non-Boolean sensing model with given $P_{des}=0.63$ (constant dashed line). The values of the parameters are $s=2$, $\gamma=1$, $\eta(0)=1$.

VIII. Conclusions

In this paper, we presented a scheduling scheme for the duty cycle of dynamic sensor networks comprised of RF- or radar-based sensors, whose footprints shrink with the decrease in available power. In particular, we examined networks in which sensors were deployed randomly according to a stationary spatial Poisson point process, which simplified the computation of the probability of event detection and allowed us to use this probability as our performance criterion. To establish the relationship between the desired performance criterion and the lifetime of a network, we performed a theoretical analysis in which we investigated both persistent and non-persistent events and proposed scheduling schemes for both the scenarios that maximized the lifetime of a network. Moreover, we examined two sensing models, namely, Boolean and non-Boolean, in order to incorporate varied physical systems in our system model. The results generated through the theoretical analysis were validated by the Monte Carlo simulations of proposed controllers, which proved that our proposed schemes maintained the desired performance through out the lifetime of the network.

References

[1] I. F. Akyildiz, E. Cayirci, W. Su and Y. Sankarasubramaniam, “A survey on sensor networks,” *IEEE Communications Magazine*, vol. 40, issue 8, August 2002, pp.102-114.

[2] A. Alfieri, A. Bianco, P. Brandimarte and C.F. Chiasserini, “Maximizing system lifetime in wireless sensor networks,” in *European Journal of Operational Research*, vol.181, issue 1, August 2007, pp. 390 402.

[3] C. G. Cassandras and X. Ning. “Optimal Dynamic Sleep Time Control in Wireless Sensor Networks,” in 47^{th} *IEEE Conference on Decision and Control*, Dec. 2008, pp. 2332 - 2337.

[4] K. Chang. *RF and Microwave Wireless Systems*, John Wiley and Sons, 2000.

[5] C. Y. Chong and S. P. Kumar, “Sensor Networks: Evolution, Opportunities and Challenges,” in *Proceedings of the IEEE* , vol 91, issue 8, August 2003, pp. 1247 - 1256.

[6] R. L. Cruz and M. Sarkar, “An Adaptive Sleep Algorithm for Efficient Power Management in WLANs,” in *Vehicular Technology Conference*, 2005, vol. 3, pp. 2101 - 2104.

[7] F. Fekri and R. Subramanian, “Sleep Scheduling and Lifetime Maximization in Sensor Networks: Fundamental Limits and Optimal Solutions”, in *Proc. Information Processing in Sensor Networks (IPSN)*, 2006, pp. 218 - 225.

[8] P. Hall, *Introduction to the Theory of Coverage Processes*, John Wiley and Sons, 1988.

[9] C. Hsin and M. Liu, “Network coverage using low duty-cycled sensors: Random and coordinated sleep algorithm,” in *Proceedings of the Third International Symposium On Information Processing In Sensor Networks*, April 2004, pp. 433 - 442.

[10] H. Jaleel, A. Rahmani and M. Egerstedt, “Duty Cycle Scheduling in Dynamic Sensor Networks for Controlling Event Detection Probabilities,” in *American Control Conference*, San Francisco, July 2011.

[11] H. Jaleel, A. Rahmani and M. Egerstedt, “Probabilistic Life Time Maximization in Dynamic Sensor Networks,” in Technical Report, 2011, Georgia Tech, *http://users.ece.gatech.edu/∼magnus/Papers/SensorNWTechRep.pdf*.

[12] P. Martin, R. Galvan-Guerra, and M. Egerstedt. “Power-Aware Sensor Coverage: An Optimal Control Approach,” in 19^{th} *International Symposium on Mathematical Theory of Networks and Systems MTNS*, Budapest, Hungary, July 2010.

[13] M. Potkonjak and S. Slijepcevic, “Power Efficient Organization of Wireless Sensor Networks,” in *IEEE International Conference on Communications*, vol. 2, June 2001, pp. 472 - 476.

[14] Z. Shi, W. Wolf, and F. Zhang. “A Hybrid Scheduling Model of Battery-Powered Cyber-Physical Systems,” accepted by *IEEE Transactions on Industrial Informatics*, 2009.

[15] D. Shuman and M. Liu. “Optimal Sleep Scheduling for a Wireless Sensor Network Node,” in *Proc. of 40th Asilomar Conference on Signals, Systems and Computers(ACSSC)*, Nov. 2006, pp. 1337 - 1341.

[16] D. Stoyan, W. S. Kendall and J. Mecke, *Stochastic Geometry and its Applications*, John Wiley and Sons, 1987.

[17] F. Ye, G. Zhong, J. Cheng, S. Lu, and L. Zhang. “Peas: A robust energy conserving protocol for long-lived sensor networks,” in *IEEE International Conference on Network Protocols (ICNP)*, 2002, pp. 28 - 37.

[18] Y. R. Tsai. “Sensing Coverage for Randomly Distributed Wireless Sensor Networks in Shadowed Environments,” in *IEEE Transactions on Vehicular Technology*, Jan. 2008, Vol. 57, No. 1, pp. 556 - 564.