

# Robust and Probabilistic Power-aware Scheduling of Wireless Sensor Networks

Hassan Jaleel and Magnus Egerstedt

**Abstract**—In this paper, we investigate randomly deployed wireless sensor networks with decaying footprints, akin to those of RF- or radar-based sensors, and relate the performance of a sensor to its available power. More importantly, to compensate for the effects of decrease in available power, we propose probabilistic scheduling controllers that maintain a desired probability of event detection for any power-decay model with no a-priori knowledge of the model itself. In addition to maintaining desired event detection probability, the proposed scheme is robust to node failures and adaptively adjusts the control parameter in the case of the failure of nodes to maintain desired performance. Furthermore, the scheme is completely decentralized since it requires neither any global information nor any local communication for computing the control parameter at any time.

## I. INTRODUCTION

Wireless sensor networks are used in a wide range of real-life applications such as intruder detection, precision agriculture, smart buildings and monitoring hazardous and potentially hostile environments like detecting fires in a forest or oil spillage in the ocean [1]. One key aspect of such large-scale systems is that they are inherently power sensitive. Therefore, a critical problem, which is a subject of active research in the wireless sensor networks community, is power conservation [17]. In static networks, i.e., networks comprising of agents with no mobility, power management has always been a part of design methodology and redundancy in sensor deployment is utilized to maximize the lifetime of a network by developing intelligent switching schemes for turning the nodes on or off (see e.g., [2], [13], [5], [16], [19], [18], and [11]). Despite the numerous approaches to conserving power that have been proposed, there is an important issue that requires thorough investigation.

In [12] we identified the problem of effects of decrease in available power on network performance for the first time. To address this problem, we developed an explicit relationship between sensor performance and available power for sensors that use RF- or radar-based antennas for sensing and communication. Using this relationship, we further proposed power-aware controllers to compensate for the deteriorating performance of a sensor. However, the analysis and design in [12] was limited to a scenario in which the available power decreases according to an exponential decay law. In this paper we extend our previous work for a generalized system model and propose power-aware control

laws to maintain desired performance and minimize power consumption. However, we do not assume any particular power-decay model in the design and analysis of our proposed scheme. Moreover, our proposed scheme can handle increase in available power, which is important because of the two reasons. First, the power delivered by batteries is affected by atmospheric temperature implying that it may increase with the increase in temperature [22]. Secondly, with the advancements in energy harvesting technologies, it is becoming common that sensing devices are equipped with small devices like solar panels to harvest energy. In such scenarios, the available power may increase with time and if each node does not incorporate this increase in power, the system will consume more power than is required to maintain the desired performance. Our proposed scheme takes into account this increase in available power and adjusts the control parameter to reduce power consumption.

The decrease in available power with time is a critical problem, but it is a missing link in the existing literature on the wireless sensor networks. In almost all the existing literature on the design of wireless sensor networks, it is assumed that the performance of sensing nodes remain constant throughout their lifetime. However, in the summer of year 2002, a wireless sensor network comprising approximately 50 nodes was deployed on an uninhabited island for habitat monitoring. This was one of the first networks of this size that ran unattended for a period of four months over which 1.1 million readings were received from the network. Using this data, the designers analyzed the performance of the network to deepen their understanding of the practical issues in the network design and later published their findings in [15]. The crux of their analysis was that the performance of the network was far below the level it was designed for. One important observation that they made after analyzing the data was that the batteries of these nodes were unable to maintain constant terminal voltage<sup>1</sup>. In fact, the terminal voltage decreased continuously not just at the end but throughout the operational lifetime of a node. Since the power that is delivered to a device is directly related to its terminal voltage, this implies that for each node the available power kept on decreasing with time. This decrease in available power must have a negative impact on the performance of sensing devices, which violates the assumption of constant performance. Therefore, the basic objective of this work is to understand the relationship between system performance and available power and propose power-aware controllers to

Authors are with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332; Emails: {hjaleel,magnus}@ece.gatech.edu.

This work was sponsored in part by a grant from NSF.

<sup>1</sup>The nodes were powered by two standard AA batteries

maintain desired performance.

In addition to compensating the effects of power decay on sensor performance, the scheme we are proposing in this paper is robust to node failures. The scheduling controller adjusts the control parameter to minimize the effects of node failures and maintain desired performance for the maximum possible time. Moreover, the proposed scheme is capable of handling the situation in which additional nodes are deployed at some later time to improve network performance. In such a scenario, the load on the existing nodes should decrease. The proposed scheme is adaptive in a sense that it takes into account addition of nodes in the network and again adjusts the control parameter accordingly to minimize power consumption.

## II. SYSTEM DESCRIPTION

Consider a domain  $\mathcal{D} \subset \mathbb{R}^2$  in which a large number of sensors are randomly deployed for monitoring purposes. In random deployment, sensors can either be dropped from a plane flying over the region of interest or any vehicle driving across it [1]. This deployment scheme is preferred for large networks because it allows to deploy a large number of nodes with minimum effort. However, random deployment does not provide any information about the particular locations of sensors which prohibits design and analysis of such networks. Thus, the first task is to model this sensor deployment in an efficient manner. For this task, we consult stochastic geometry, which is the study of random patterns in a plane or in any high dimensional space. The fundamental concept in stochastic geometry is that of a point process where

**Definition 2.1:** [9] *A point process  $\Phi$  on  $\mathbb{R}^2$  is a random sequence of points,*

$$\Phi = \{x_1, x_2, \dots\} \quad x_i \in \mathbb{R}^2,$$

*that is locally finite and simple.*

*Locally finite* means that any bounded subset of  $\mathbb{R}^2$  contains only a finite number of points of  $\Phi$  and *simple* means that two points cannot overlap, i.e.,  $x_i \neq x_j$  if  $i \neq j$ . Since the location of each sensor is a point in  $\mathbb{R}^2$  and the locations of all the sensors form a random sequence of points, the random deployment of sensors can be modeled as a spatial point process.

The next step is to select a particular point process. In random deployment, the location of each sensor is completely independent of the location of all the other sensors. If we combine this fact with an assumption that the number of deployed sensors goes to infinity in the limiting case such that  $N/\|\mathcal{D}\| \rightarrow \lambda$ , where  $\|\mathcal{D}\|$  is the area of domain  $\mathcal{D}$  and  $\lambda$  is the expected number of sensors per unit area, then the sensor deployment can be modeled as a Poisson point process. More formally,

**Definition 2.2:** [8] *A point process  $\Phi$  is a Poisson point process if it satisfies the following two conditions*

- 1)  $P(\bigcup_i (\phi(B_i) = n_i)) = \prod_i P(\phi(B_i) = n_i)$ , where  $B_1, B_2, \dots, B_n$  are disjoint subsets of  $\mathbb{R}^2$  and  $\phi(B_i)$

*is the number of points of  $\Phi$  in the set  $B_i$ .*

- 2) *For any set  $B \subset \mathbb{R}^2$ ,  $\phi(B)$  is Poisson distributed.*

Moreover, if  $\lambda$  is constant throughout the area, then we have a homogeneous Poisson process. Once it is established that the sensor deployment is modeled as a stationary Poisson point process with some intensity  $\lambda$ , then the number of sensors in any region of area  $A$  can be determined using Poisson distribution.

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

where  $P_n(A)$  is the probability of having  $n$  sensors in some area  $A$ .

Next we specify the model of the sensing agents we have considered in this paper. Each sensing agent located at  $x_i \in \mathbb{R}^2$

- is a static device that uses RF- or radar-based omnidirectional antenna for sensing and communication.
- is battery powered where the available power decreases with time.
- has a circular disk of radius  $\Delta$ , centered at  $x_i$ , as its sensing region called the footprint of a sensor.

We assume that all the deployed sensors are identical, i.e., they have same battery power and same sensing, processing and communication capabilities. To conserve power, sensors are switched between on and off states. To simplify the analysis, we assume that power is consumed only when a sensor is in the on state. However, the results derived in the later sections can easily be extended for a more generalized power model. Moreover, we assume that sensors switch between on and off states at discrete time instants  $k\Delta t$ , where  $\Delta t$  is the length of an interval in which a sensor maintains its state.<sup>2</sup> At each switching time, a sensor decides to be in the on state with probability  $q(k)$  and in the off state with probability  $1 - q(k)$ .

To design power-aware controllers, we first need to establish a relationship between power that is available to a sensor for transmission and the desired performance criterion. In this paper, we have selected event detection probability as our desired performance criterion because for a randomly deployed network in which sensors randomly switch between on and off states, event detection probability is a natural performance criterion. Let  $\eta(k)$  is the power that a sensor can transmit in order to monitor its footprint at time  $k$ . Then from RF antenna theory [4], we know that for an RF- or radar-based antenna, area of the footprint is directly related to transmitted power, i.e.,

$$A(k) = \alpha \eta(k), \quad (2)$$

where  $\alpha$  is proportionality constant. Equation (2) is an explicit relationship between the performance of an individual sensor and its available power. However, we want to extend this relationship to the performance of the entire network, i.e., event detection probability.

<sup>2</sup>For brevity of notation, we will denote switching time by  $k$  instead of  $k\Delta t$  in the rest of this paper.

### III. POWER-AWARE SCHEDULING

Let  $x_e \in \mathcal{D}$  be the location of a non-persistent event and we want to find the probability of detection for this event.<sup>3</sup> To find event detection probability,  $P_d$ , we start with the probability of event not being detected,  $P_u$ . Event at  $x_e$  is not detected if either of the two conditions is satisfied.

- 1)  $x_e$  does not belong to the footprint of any sensor.
- 2) All the sensors such that  $x_e$  belong to their footprints are in the off state.

Mathematically,

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

In the above equation,  $n = 0$  corresponds to Condition 1 while the summation of terms having  $n \neq 0$  corresponds to Condition 2. This implies that event detection probability is

$$P_d = 1 - e^{-\lambda A q},$$

where  $A$  is the area of the footprint of each sensor. However, we are interested in the scenario when the available power is decaying with time. We model the power decay by a difference equation

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

where  $f : \mathbb{R}_+ \times \mathbb{N} \rightarrow \mathbb{R}_+$  is a function of  $\eta(k-1)$  and  $k$ . The negative sign in the above model represents power decay. We can replace the negative sign with a positive sign for scenarios in which the available power increases with time owing to energy harvesting and derive all the subsequent results in the similar manner. Now, we know that power is consumed only when a sensor is *on*, and a sensor is *on* with probability  $q(k)$ , we can modify the power model as

$$\eta(k) = \begin{cases} \eta(k-1) - \Delta t f_k & q(k-1) \\ \eta(k-1) & 1 - q(k-1) \end{cases}$$

Thus, the expected power level of each sensor is

$$\hat{\eta}(k) = \hat{\eta}(k-1) - q(k-1) \Delta t \hat{f}_k. \quad (3)$$

As the expected power level  $\hat{\eta}(k)$  is the same for all the sensors in the network, it is a global property that we will use to relate available power to global network performance. We have already established that the area of the footprint is directly related to available power, so this relationship can simply be extended to expected area vs expected power, i.e.,

$$\hat{A}(k) = \alpha \hat{\eta}(k) \quad (4)$$

Next we propose a scheduling scheme to maintain desired event detection probability. In order to decide whether to be in the on or off state at some time instant  $k$ , each sensor at location  $x_i$ , where  $i \in \{1, 2, \dots\}$ , performs the following steps:

<sup>3</sup>A non-persistent event is one that does not leave a mark in the environment and can only be detected at the time of its occurrence

#### Scheme 1: Sensor Scheduling Scheme

**Given:** Desired event detection probability ( $P_{des}$ )  
Sensor deployment intensity  $\lambda$

**At Step 0:**

- Initializes its probability of being on

$$q(0) = \frac{\ln\left(\frac{1}{1-P_{des}}\right)}{\lambda \alpha \eta(0)} \quad (5)$$

- Selects a number  $m_i$  such that  $m_i \sim \text{unif}[0, 1]$ .
- Turns on if  $m_i < q(0)$ .

**At Step k:**

- Selects a number  $m_i$  such that  $m_i \sim \text{unif}[0, 1]$ .  
**if**  $m_i < q_i(k)$ 
  - Decides to be in the on state in interval  $[k, k+1]$ .
  - At time  $k+1$ , computes

$$q_i(k+1) = \frac{\eta(0)}{\eta_i(k+1)} q(0). \quad (6)$$

**else**

- Decides to be in the off state in interval  $[k, k+1]$ .

$$q_i(k+1) = q_i(k)$$

**Theorem 3.1:** *If each sensor makes its switching decisions according to Scheme 1, then the network maintains  $P_{des}$  throughout its lifetime.*

*Proof:* Using Equation (4) and standard results from stochastic geometry [10], we can easily show that the event detection probability of a network in which available power decreases with time is given by

$$P_d = 1 - e^{-\lambda \hat{A}(k) \hat{q}(k)}, \quad (7)$$

where  $\hat{A}(k)$  is the expected area of the footprint at time  $k$  and  $\hat{q}(k)$  is the average of the on probabilities of all sensors, i.e.,

$$\hat{q}(k) = \frac{1}{N} \sum_{i=1}^N q_i(k), \quad (8)$$

where  $q_i(k)$  is the probability of being on of the  $i^{\text{th}}$  sensor. If the power-decay model is known as in [12], then  $q_i(k) = q(k)$  for all the sensors and as a result  $\hat{q}(k) = q(k)$ . However, in Scheme 1, each sensor is updating its control parameter  $q_i(k)$  based on its current power level  $\eta_i(k)$ . Moreover, the current power level of each sensor depends on  $q_i(k-1)$ . Therefore, we have two discrete time random variables  $\eta_i(k)$  and  $q_i(k)$  whose values are updated according to first-order difference equations and whose initial values are same for all the sensors, so their expected values will also be the same.

Given  $P_{des}$ , the desired probability of event detection that we want to maintain, then

$$1 - P_{des} = e^{-\lambda \hat{A}(k) \hat{q}_{des}(k)}.$$

After performing simple algebraic manipulations, we get

$$\hat{q}_{des}(k) = \frac{\ln\left(\frac{1}{1-P_{des}}\right)}{\lambda \hat{A}(k)} \quad (9)$$

Thus, to maintain  $P_{des}$ , the average of the on probabilities of all the sensors must be equal to  $\hat{q}_{des}(k)$  for all the time.

Simply replacing  $k$  with 0 in Equation (9) gives us the desired initial condition  $q(0)$  as in Equation (5). Moreover, by considering the fact that  $\hat{A}(k) = \alpha\hat{\eta}(k)$ , and by multiplying and dividing Equation (9) with  $\eta(0)$  yields

$$\hat{q}_{des}(k) = \frac{\eta(0)}{\hat{\eta}(k)}q(0). \quad (10)$$

In Equation (6),  $\eta(0)$  and  $q(0)$  are deterministic and same for all the sensors, so the only random variable is  $\eta_i(k)$  whose expected value is  $\hat{\eta}(k)$ . This implies that the expected value of  $q_i(k)$  in Equation (6) is

$$\hat{q}_i(k) = \frac{\eta(0)}{\hat{\eta}(k)}q(0),$$

and this expected probability is same for all the sensors in the network. Therefore, using the strong law of large numbers

$$\hat{q}(k) = \frac{1}{N} \sum_{i=1}^N q_i(k) \rightarrow \hat{q}_i(k) \quad \text{as } N \rightarrow \infty$$

We have already shown that  $\hat{q}_i(k) = \hat{q}_{des}(k)$  for all the time, which concludes the proof. ■

Equation (6) is a power-aware controller that maintains desired event detection probability throughout the lifetime of the network. It is important to note that this controller is independent of the power-decay law, since the only quantity a sensor needs to compute is its available power at time instant  $k$ , which can be computed easily. Therefore, this controller can be implemented even on the sensing devices that have low computational capabilities.

#### A. Simulation

To verify the performance of Scheme 1, we performed Monte Carlo simulations in Matlab. We considered a rectangular domain of dimensions  $[30 \times 30]$  in which sensors were deployed randomly according to Poisson point process of intensity  $\lambda = 10$ . Each sensor had an initial footprint area  $A(0) = 1$  and initial power level  $\eta(0) = 1$ . To compare the performance of this scheme with our previous work in [12], we considered an exponential power decay law given by,

$$\eta(k) = \eta(k-1) - \gamma\Delta t\eta(k-1), \quad (11)$$

where  $\gamma$  is decay constant and for this simulation  $\gamma = 1$ . We want to point out that in [12], each sensor had a complete knowledge of its power-decay law and it used this information to update its control parameter. However, in this scheme, no sensor has any information about the decay law, which makes this work more attractive for practical implementation. The event detection probability of the simulated system under the proposed scheme is demonstrated in Figure (1). The dotted line corresponds to the desired detection probability  $P_{des} = 0.67$ . The figure shows that after the initial settling phase, the system maintains desired event detection probability throughout the lifetime of the network. In the sensor network literature, network lifetime has been

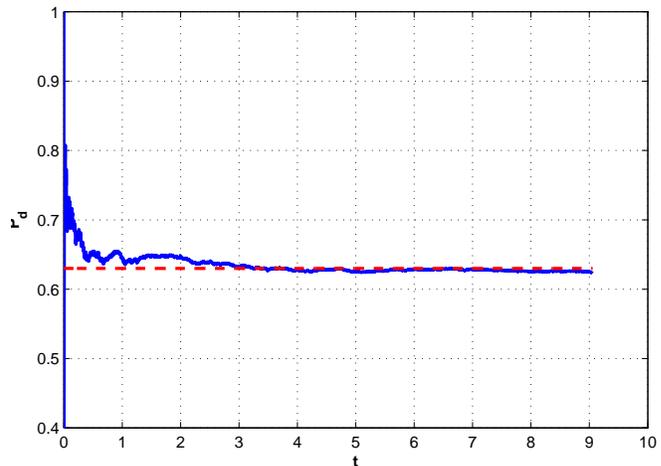


Fig. 1. Event detection probability  $P_d$  vs time  $t$  for a sensor networks with Scheme 1 (solid line) and given  $P_{des} = 0.63$  (constant dashed line). Here  $\lambda = 10$ ,  $A(0) = 1$ , and  $\gamma = 1$ .

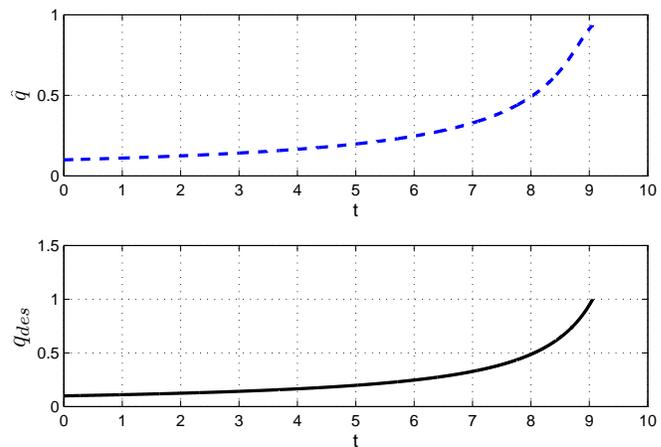


Fig. 2. Evolution of average probability of a sensor being *on* for a given desired performance  $P_{des}$  where  $\lambda = 10$ ,  $A(0) = 1$ , and  $\gamma = 1$ . In each case, lifetime of the network is achieved when  $\hat{q}_{des} = 1$ .

defined in a variety of ways [6]. However, in this work, we define network lifetime as

**Definition 3.1:** *The lifetime of the sensor network is the maximal time beyond which the desired network performance cannot be achieved.*

In [12], we presented an exact expression of network lifetime for exponential decay model and using that expression, the lifetime of the simulated system is  $T_f = 9.0578$ . Figure (2) shows the evolution of the average probability of sensors being on. In the top figure,

$$\hat{q}(k) = \frac{1}{N} \sum_{i=1}^N q_i(k),$$

where  $N$  is the total number of sensors. In the bottom figure  $\hat{q}_{des}(k)$ , which is the desired probability of being on given

by Equation (10), and the two probabilities are almost the same for all the time as required by Theorem 3.1 to hold.

#### IV. ROBUST POWER-AWARE SCHEDULING SCHEME

##### A. Decentralized Scheduling Scheme

In Section III, we proposed a probabilistic power-aware scheduling scheme that can maintain desired event detection probability throughout the lifetime of a network. One important aspect of this scheme was that each sensor was making its switching decision completely independent of all the other sensors. However, there was one global parameter that was used by each sensor in its decision making and that parameter was  $\lambda$ , i.e., expected number of sensors per unit area. In this section, we will update Scheme 1 to remove the dependence on  $\lambda$ . As a result, the updated scheme will be decentralized in true sense as neither global knowledge nor local communication will be used to make switching decisions.

**Scheme 2: Updated Sensor Scheduling Scheme**  
**Given:** Desired event detection probability ( $P_{des}$ )  
**At Step 0:**

- Senses its footprint  $\mathcal{B}_{\Delta(0)}(x_i)$  having area  $A_i(0)$  and computes  $\lambda_i$ .

$$\lambda_i = \frac{\text{Number of sensors in } \mathcal{B}_{\Delta(0)}(x_i)}{A(0)}.$$

- Initializes its probability of being on

$$q_i(0) = \frac{\ln\left(\frac{1}{1-P_{des}}\right)}{\lambda_i \alpha \eta(0)} \quad (12)$$

- Selects a number  $m_i$  such that  $m_i \sim \text{unif}[0, 1]$ .
- Turns on if  $m_i < q_i(0)$ .

**At Step k:**

- Follow Scheme 1

In the above scheme,  $\lambda_i$  is the estimate of  $\lambda$  generated by sensor  $i$ . Since the expected number of sensors in unit area is still  $\lambda$ ,

$$\frac{1}{N} \sum_{i=1}^N \lambda_i = \lambda.$$

Each sensor uses its estimate of  $\lambda$  to compute  $q_i(0)$  according to Equation 12, which is a convex function of  $\lambda_i$ . From Jensen's inequality, we know that for a convex function, say  $f$ ,

$$f(E(x)) \leq E(f(x)).$$

This implies that

$$q(0) \leq \frac{1}{N} \sum_{i=1}^N q_i(0), \quad (13)$$

where  $q(0)$  is given by Equation (5) and  $q_i(0)$  is given by Equation (12). The effect of this inequality is obvious from Figure (3), which demonstrates event detection probability of the same system as simulated in Section III-A but using Scheme 2. Since  $E(q_i(0)) \geq q(0)$ , event detection probability is higher than desired value (dotted line) in the beginning, which seems good. However, the disadvantage of high

initial probability is that the sensors consume more power than desired and consequently the detection probability falls below the desired level before the lifetime is over. This is the price we have to pay for making the system completely decentralized.

In order to quantify the performance loss, we ran 100 iterations of the Monte Carlo simulations under the same setup as described in Section III-A. As long as the detection probability remains greater than or equal to  $P_{des}$  our performance criterion is met. However, when it falls below  $P_{des}$ , we have performance loss, which is shown in Figure (IV-A). For each iteration of the simulation, Figure (4(a)) shows the percentage average performance loss while Figure (4(b)) shows the percentage maximum performance loss and the solid line in both the figures represent average of the 100 values. These figures show that the maximum performance loss is on average 11% where as the average performance loss is 4.5 % on average.

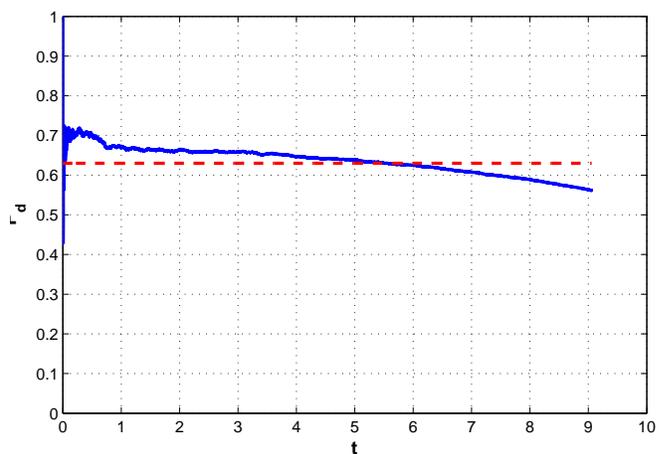


Fig. 3. Event detection probability  $P_d$  vs time  $t$  for a sensor networks (solid line) with given  $P_{des} = 0.63$  (constant dashed line) under Scheme 2. Here  $\lambda = 10$ ,  $A(0) = 1$ , and  $\gamma = 1$ .

##### B. Robust and Decentralized Scheduling Scheme

Until now, we have assumed that all the sensors in a network remain operational throughout the lifetime of the network. However, we are dealing with networks comprising of extremely cheap and low quality devices that are dropped in the region of interest randomly. Therefore, it is possible that some of the sensors stop working unexpectedly before their operational lifetime is over. This will obviously have a negative impact on the performance of the network since we have not incorporated failures of the devices in the design of our scheduling scheme. There is also a possibility that at some time, we want to deploy more sensors to improve the performance of the network and reduce the workload of existing ones. Again, this is a scenario that our proposed scheme cannot handle. Therefore, we update Scheme 3 to detect the failure of existing devices or addition of new devices and adjust the control parameter in a manner so that the desired performance

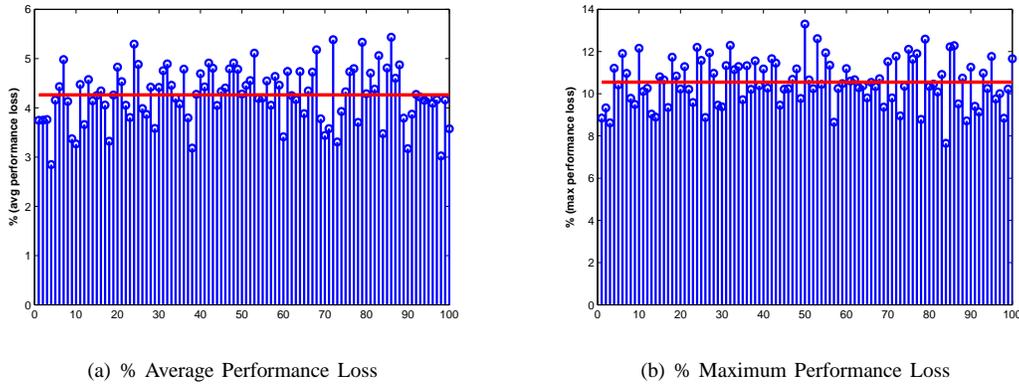


Fig. 4.

criterion is maintained while reducing power consumption.

### Scheme 3: Robust Sensor Scheduling Scheme

**Given:** Desired event detection probability ( $P_{des}$ )

**At Step 0:**

- Follow Scheme 2.

**At Step k:**

**if**  $\text{mod}(l, k) = 0$

- Senses its footprint  $\mathcal{B}_{\Delta(k)}(x_i)$  having area  $A_i(k)$  and computes  $\lambda_i$ .

$$\lambda_i = \frac{\text{Number of sensors in } \mathcal{B}_{\Delta(k)}(x_i)}{A_i(k)}.$$

- Updates its probability of being on

$$q_i(k) = \frac{\ln\left(\frac{1}{1-P_{des}}\right)}{\lambda_i \alpha \eta_i(k)} \quad (14)$$

- Set  $q_i(0) = q_i(k)$  and  $\eta_i(0) = \eta_i(k)$ .
- Selects a number  $m_i$  such that  $m_i \sim \text{unif}[0, 1]$ .
- Turns on if  $m_i < q_i(k)$ .

**else**

- Selects a number  $m_i$  such that  $m_i \sim \text{unif}[0, 1]$ .  
**if**  $m_i < q_i(k)$ 
  - Decides to be in the on state in interval  $[k\Delta t, (k+1)\Delta t]$ .
  - Computes

$$q_i(k+1) = \frac{\eta_i(0)}{\eta_i(k+1)} q_i(0). \quad (15)$$

**else**

- Decides to be in the off state in interval  $[k\Delta t, (k+1)\Delta t]$ .

$$q_i(k+1) = q_i(k)$$

In this scheme, each sensor initializes its estimate of deployment intensity,  $\lambda_i$ , as in Scheme 2 and based on this estimate computes its initial probability of being on, i.e.,  $q_i(0)$ . Then, at each step it follows the same control law to update its control parameter,  $q_i(k)$  as in Equation (6). However, after every  $l$  steps, each sensor updates its estimate  $\lambda_i$  and uses this updated estimate to compute its probability of being on. This parameter  $l$  can either be given a priori based on the quality of devices or can be selected randomly by each device itself. By updating  $\lambda_i$ , each sensor takes into account the failure or addition of nodes in its sensing region and uses this information to adjust its probability of being on. In the case

of node failures,  $\lambda_i$  will decrease which will force the sensor to increase  $q_i(k)$  to maintain desired detection probability. Similarly, in the case of the addition of new nodes,  $\lambda_i$  will increase which will allow the sensor to decrease  $q_i(k)$  and reduce energy consumption while maintaining desired detection probability. Thus, Scheme 3 is robust to node failures and tries to maintain desired detection probability as long as a minimum number of sensors is available.

The performance of a network simulated under the same settings as in Section III-A, except  $\gamma = 0.4$ , is shown in Figure (5). In this simulation, sensors keep failing with probability  $P_f = 0.05$  after random intervals. The detection probability for this network is shown in Figure (5) as decaying solid line and it is apparent that the network cannot maintain  $P_{des}$ . By applying Scheme 3, the performance of the network clearly improves as shown in the dotted decaying plot. However, the network is still unable to maintain  $P_{des}$  throughout its lifetime because of two reasons. Firstly, as discussed for Scheme 2, when sensors estimate  $\lambda$ , there is always going to be a performance loss. Secondly, in this case, sensors are failing constantly and after some time it becomes simply impossible for the network to maintain  $P_{des}$  because of insufficient sensor deployment.

## V. CONCLUSIONS

In this paper we considered a randomly deployed wireless sensor network consisting of Rf- or radar-based sensors and analyzed the effect of decrease in available power on the performance of individual sensors, and of the entire network. Then, we proposed a probabilistic power-aware scheduling controller to compensate for the effects of power decay and maintain desired performance throughout the lifetime of the network. The proposed controller is independent of the power-decay model and completely decentralized since it does not require any global information or communication. Moreover, we presented a robust version of the scheduling scheme that can detect node failures and adjust the control parameter to compensate for the lost nodes while reducing energy consumption.

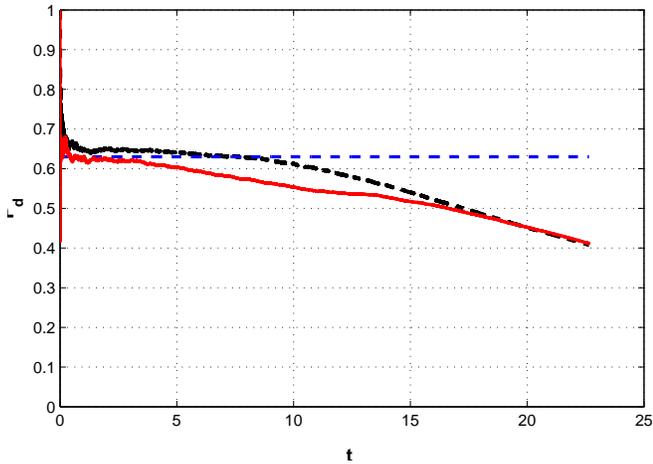


Fig. 5. Event detection probability  $P_d$  vs time  $t$  for a sensor network with failing nodes and no compensation (solid decaying line), sensor network with failing nodes and Scheme 3 (solid decaying line) with given  $P_{des} = 0.63$  (constant dashed line). Here  $\lambda = 10$ ,  $A(0) = 1$ ,  $\alpha = 1$ , and  $\gamma = 0.4$ .

## REFERENCES

- [1] I. F. Akyildiz, E. Cayirci, W. Su and Y. Sankarasubramaniam, "A survey on sensor networks," *IEEE Commun. Mag.*, vol. 40, issue 8, Aug. 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, Aug. 2007, pp. 390 - 402.
- [3] C. Y. Chong and S. P. Kumar, "Sensor networks: Evolution, opportunities, and challenges," in *Proc. of the IEEE*, vol. 91, issue 8, Aug. 2003, pp. 1247 - 1256.
- [4] K. Chang, *RF and Microwave Wireless Systems*, John Wiley and Sons, 2000.
- [5] R. Subramanian and F. Fekri, "Sleep scheduling and lifetime maximization in sensor networks: Fundamental limits and optimal solutions," in *Proc. IPSN*, ACM Press, 2006, pp. 218 - 225.
- [6] I. Dietrich and F. Dressler, "On the lifetime of wireless sensor networks," *ACM Transactions on Sensor Networks*, Feb. 2009, vol. 5, issue 1, pp. 1 - 39.
- [7] D. Tian and N. D. Georganas, "A coverage-preserving node scheduling scheme for large wireless sensor networks," in *Proc. of the 1st ACM Workshop on Wireless Sensor Networks and Applications.*, 2002, pp. 32 - 41.
- [8] P. Hall, *Introduction to the Theory of Coverage Processes*, New York, Wiley, 1988.
- [9] A. Karr, *Point processes and their statistical inference*, 2nd Ed. Marcel Dekker Inc., 1991.
- [10] D. Stoyan, W. S. Kendall and J. Mecke, *Stochastic Geometry and its Applications*, Wiley (Chichester W. Sussex and New York), 1987.
- [11] C. Hsin and M. Liu, "Network coverage using low duty-cycled sensors: random & coordinated sleep algorithms," in *Proc. IPSN*, ACM Press, Apr. 2004, pp. 433 - 442.
- [12] H. Jaleel, A. Rahmani, and M. Egerstedt, "Probabilistic Lifetime Maximization of Sensor Networks," in *IEEE Tran. on Automatic Control*, Issue 99, 2012.
- [13] S. Slijepcevic and M. Potkonjak, "Power efficient organization of wireless sensor networks," in *Proc. IEEE ICC*, vol. 2, June 2001, pp. 472 - 476.
- [14] 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," in *IEEE Trans. Parallel and Distributed Systems*, vol. 18, no. 3, pp. 334-350, Mar. 2007
- [15] R. Szwedczyk, J. Polastre, A. Mainwaring, and D. Culler. "Lessons from a sensor network expedition," in *First European Workshop on Wireless Sensor Networks*, Jan. 2004.
- [16] F. Ye, G. Zhong, J. Cheng, S. Lu, and L. Zhang. "PEAS: A robust energy conserving protocol for long-lived sensor networks," in *Proc. ICDCS*, 2003, pp. 28 - 37.
- [17] V. Rodoplu and T. Meng, "Minimum energy mobile wireless networks," in *IEEE Journal on Selected Areas in Communications*, vol. 1, no. 8, pp. 1333 - 1344, 1999.
- [18] Q. Cao, T. Abdelzaher, T. He, and J. Stankovic, "Towards optimal sleep scheduling in sensor networks for rare event detection," in *Proc. IPSN*, Apr. 2005.
- [19] C. Gui and P. Mohapatra, "Power conservation and quality of surveillance in target tracking sensor networks," in *Proc. of ACM Mobicom*, 2004.
- [20] Y. Xu, J. Heidemann, and D. Estrin, "Geography-informed energy conservation for ad hoc routing," in *Proc. of ACM MobiCom*, Rome, Italy, July 2001.
- [21] Q. Dong, "Maximizing system lifetime in wireless sensor networks," in *Proc. IPSN*, 2005.
- [22] R. Rao, S. Vrudhula, and D. N. Rakhmatov, "Battery modeling for energy-aware system design," in *Computer*, vol. 36, no. 12, pp. 7787, Dec. 2003