Approximate Dynamic Programming for Condition-Based Node Deployment in a Wireless Sensor Network

Nicholas T. Boardman Kelly M. Sullivan

Abstract

The flexibility of deployment strategies combined with the low cost of individual sensor nodes allow wireless sensor networks (WSNs) to be integrated into a variety of applications. Network operations degrade over time as sensors consume a finite power supply and begin to fail. In this work we address the selective maintenance of a WSN through a condition-based deployment policy (CBDP) in which sensors are deployed over a series of missions. The main contribution is a Markov decision process (MDP) model to maintain a reliable WSN with respect to region coverage. Due to the resulting high dimensional state and outcome space, we explore approximate dynamic programming (ADP) methodology in the search for high quality CBDPs. Our model is one of the first related to the selective maintenance of a WSN through the repeated deployment of new sensor nodes with a reliability objective, and one of the first ADP applications for the maintenance of a complex WSN. Additionally, our methodology incorporates a destruction spectrum reliability estimate which has received significant attention with respect to network reliability, but its value in a maintenance setting has not been widely explored. We conclude with a discussion on CBDPs in a range of test instances, and compare the performance to alternative deployment strategies.

Keywords — Approximate Dynamic Programming, Selective Maintenance, Network Reliability, Wireless Sensor Networks

---

Nicholas T. Boardman and Kelly M. Sullivan are with the Department of Industrial Engineering, University of Arkansas, 4207 Bell Engineering Center, Fayetteville, AR 72701. This material is based upon work supported by the National Science Foundation under Grant No. CMMI-1751191.
email: nboardma@uark.edu, ksulliv@uark.edu

Key Notation

$\mathcal{R}$	The set of all subregions.
$\mathcal{G}$	The Wireless Sensor Network.
α	The coverage requirement.
$C(\mathcal{G}(t))$	The coverage of the network at time $t \geq 0$.
M	The number of missions.
δ	The duration of a mission.
$N_{m,i,k}$	The number of functioning sensor nodes with age k in subregion i at the beginning of mission m.
$\bar{N}_m$	The total number of sensor nodes functioning in the network at the beginning of mission m.
B_m	The budget available for the remaining missions.
$\boldsymbol{S}_m$	The state variable, $\boldsymbol{S}_m = (\boldsymbol{N}_m, B_m)$.
$\mathcal{S}$	The set of all possible states.
x_m	The number of sensor nodes deployed at the beginning of mission m.
$\bar{x}_m$	The total number of sensors deployed in subregion i at the beginning of mission m.
$C_m(x_m)$	The cost of action x_m.
$R_m(\boldsymbol{S}_m, x_m)$	Network reliability given state $\boldsymbol{S}_m$ and action x_m.
$V_m(\boldsymbol{S}_m)$	Value function of state $\boldsymbol{S}_m$, defined as the maximum expected number of successful missions remaining among missions $m, m+1, \ldots, M-1$ if the network is in state $\boldsymbol{S}_m$ at the beginning of mission m.
$\boldsymbol{S}_m^x$	The post-decision state variable, the state variable immediately after a deployment action.
$V_m^x(\boldsymbol{S}_m^x)$	The value function of the post-decision state, defined as the maximum number of successful missions remaining among missions $m+1, m+2, \ldots, M-1$ given the post-decision state variable $\boldsymbol{S}_m^x$.
$s_{\alpha,i}^n$	The probability that the i^{th} sensor node failure results in coverage falling below α.
$b(x;n,p)$	The binomial pdf, $b(x;n,p) = \binom{n}{x} p^x (1-p)^{n-x}$.
$B(x;n,p)$	The binomial cdf, $B(x;n,p) = \sum_{i=0}^{x} \binom{n}{i} p^i (1-p)^{n-i}$.

1 Introduction

Through the cooperative effort of individual sensor nodes, a wireless sensor network (WSN) can be deployed to monitor and report data on an event of interest in a desired region. In environmental settings WSNs can be valuable to monitor a forest providing early detection of forest fires, or to monitor a coastline and warn about potential flooding [1]. WSNs have additionally been deployed to observe animals and their behavior in a natural habitat over a period of time with minimal disruption [2]. In commercial applications, WSNs can be utilized to track inventory or for temperature/climate control in buildings and warehouses [3]. Sensors have also been integrated into military and healthcare applications [4], illustrating the flexibility WSNs offer.

While a single sensor can monitor only a relatively small area, sensor nodes are able to communicate with each other to route information through the network. By sufficiently distributing nodes throughout a region of interest, the WSN is able to monitor a much larger region. The deployment of sensor nodes is typically categorized as either deterministic, where nodes are located at specified locations, or random [4]. Random deployment strategies can be attractive due to their ease of constructing a network, and is further supported by the low infrastructure (e.g., wires, cables) required required for operation [5]. Each individual sensor node contains the components necessary for sensing and sending/receiving data, as well as an individual battery that is drained over the course of network operation [6]. As an increasing number of sensor nodes fully consume their power supply and lose functionality, overall network coverage and connectivity begins to degrade. Sensor nodes may also fail for other reasons as well (e.g., malfunction, damage). While not the focus of this work, identifying faulty sensor nodes is an important problem as well to ensure data from the WSN is accurate [7]. In either case, node failures can have a significant impact on network capability and may have ripple effects in the network as the remaining sensor nodes are relied on more heavily, thereby increasing power consumption and the risk to other faults [8].

Methods to delay the impact of sensor node failures and extend network lifetime have received significant attention in the literature. A few areas include sleep/wake cycles [9–11] and power management techniques [12, 13]. Battery and/or sensor node replacement policies are examined in [14] and [15], but is not considered a viable strategy for a large network operating in an environment where it is not practical to access failed nodes individually [16]. In [17] and [18] WSN coverage and/or connectivity is restored by deploying a minimal number of relay nodes. A similar problem is addressed in [19] and [20], with an objective of providing a level of redundancy (i.e., k-connectivity objective) to ensure the next sensor node failure does not immediately require additional actions to restore the WSN.

The reliability of a WSN is an important metric as well as it can be used to justify the design, deployment, and operational policies for individual sensor nodes. While initial WSN reliability (i.e., for a WSN constructed at a single point in time) has been considered, research focusing on WSN node redeployment has diverged from research focusing on WSN reliability evaluation. Specifically, existing research related to WSN node deployment and redeployment typically considers a deterministic coverage, connectivity, or lifetime measure (e.g., time to first node failure) instead of an explicit measure of network reliability. An additional limitation of existing deployment models is that they are concerned with the deployment of sensors at a single point in time, and do not address the need to deploy sensors in multiple stages to maintain a WSN over a longer horizon.

In this work we consider the problem of selectively redeploying sensor nodes into a WSN over a series of maintenance actions subject to budget limitations. By redeploying sensor nodes, we aim to maximize a multiple-mission measure of a WSN's reliability of covering an area. The main contributions of this work are as follows:

1. We formulate the first Markov Decision Process (MDP) model to redeploy nodes into a WSN to maximize the reliability of region coverage over time. This model also contributes to the selective maintenance literature by addressing a large, complex network with hundreds of components that cannot be represented using traditional series, parallel, or combinations of simple subsystems.

2. We propose an Approximate Dynamic Programming (ADP) algorithm to solve the MDP approximately. Noting that the reward function of the MDP entails evaluating network reliability, we customize the ADP using a destruction spectrum approach for estimating network reliability in the presence of maintenance actions.

3. We demonstrate the model's value and the solution procedure's efficacy through numerical examples and comparison to simpler node deployment policies.

The remainder of this paper proceeds as follows. Section 2 summarizes the relevant literature in the areas of WSN reliability evaluation and selective maintenance modeling and characterizes this work's relationships to other closely related research. Section 3 formally states the problem and underlying assumptions, formulates an MDP model for the sensor node deployment problem, and prescribes an ADP approach to identify node deployment policies. Section 4 presents numerical results for a range of test instances and compares the ADP policies to alternative deployment strategies. Finally, Section 5 concludes the article, and provides directions for future research.

2 Literature Review

Failures in a WSN are frequently attributed to either link failures [21] or node failures [22–24]. The impact of component failures on reliability is reflected through WSN reliability measures that are commonly defined by traditional reliability definitions such as two-terminal, k-terminal, or all-terminal reliability, see [25–27], respectively. Recently, Xiang and Yang [23] introduced a generalized k-terminal measure to reflect the characteristic that a WSN can function as long as k arbitrary sensor nodes are connected to the sink node. In addition to battery depletion, WSN reliability in the presence of sensor node malfunctions or software errors has been addressed [28]. WSN reliability considering common cause failures is introduced in [29] where all nodes located in a certain region may be impacted. Performance based reliability measures, such as the amount of data that can delivered to a desired sink node, are also mentioned in [30]. Since WSNs are frequently deployed with a purpose of monitoring a desired region, WSN reliability definitions have also addressed reliability of area coverage [22, 28, 31].

For complex networks such as WSNs, network reliability evaluation problems are typically #P-Complete [32] and therefore pose a significant computational challenge. An exact approach to WSN reliability, such as a path-set approach in [28] or a derivation of the reliability polynomial as in [27] is limited to WSNs with only a few sensor nodes. Fault-tree analysis [33] and reliability block diagram [24] techniques have also been utilized, but are not practical for randomly deployed networks with complex sensor node communication paths. For large scale WSN's where exact methods become intractable, approximation methods such as a Monte Carlo simulation can be utilized [22, 29].

Literature addressing the reliability of a WSN has primarily focused on the evaluation of reliability. When addressing a WSN design problem a handful of works have considered an optimization objective involving reliability. In [27] one such problem is approached through the evaluation and reliability comparison for a small number of fixed network topologies. In [23] reliability is maximized by varying the transmission power based on the relationship between sensor node power consumption and sensor node lifetime. The extension of WSN reliability as an objective beyond initial network deployment, and in particular informing a maintenance policy to sustain operations for a large-scale WSN as new sensor nodes are deployed in the network has not been addressed.

Our problem of maintaining a WSN over an extended period of time subject to limitations on the available maintenance actions (e.g., a budget) relates closely to the selective maintenance problem. A mathematical formulation of the selective maintenance problem in a series-parallel system is discussed in [34], where models are presented that maximize system reliability subject to constraints on cost and maintenance time available, or minimize cost (time) subject to a constraint on the time (cost) and minimum system reliability requirement. In [35] the model is expanded to consider multiple maintenance actions (e.g., minimally repair failed components, replace failed components, replace functioning components), and model the lifetime of an individual component with a Weibull failure distribution. In both [34] and [35] the maintenance decision is based on maximizing or minimizing the objective for the next mission (i.e., until the next maintenance action). Since the system is likely maintained over a series of missions, a maintenance policy can be improved by considering the impact of a decision on future missions as well. This problem is first explored in [36] through an MDP model for a small series-parallel system, and later in [37] by applying ADP methodology to solve for a maintenance policy in a system comprised of a larger number of subsystems and components. MDP models are also presented for multi-state components for a K-out-of-N:G system in [38] and a moderately-sized series-parallel system in [39].

Recent attention on the selective maintenance problem has focused on variations to a number of assumptions common in the previous works. In [40], the authors present a model addressing stochastic imperfect maintenance. In addition to a do-nothing and perfect maintenance action, the decision can be to perform imperfect maintenance but the exact outcome/improvement to the system is uncertain. In much of the selective maintenance literature, the time between maintenance actions is also assumed to be constant. The model in [41] introduces uncertainty in mission duration, resulting in an unknown time until the next maintenance action. Meanwhile, in [42] structural dependencies between components are introduced in which improving system performance might require maintenance to several components in a group instead of a single individual component.

Compared to the selective maintenance problems discussed in [34–37], WSNs typically lack the well defined structure of a series-parallel system which complicates the estimation of network reliability. A survey of selective maintenance problems is provided in [43], with mention to several works that address complex configurations. However the definition of a complex system in the selective maintenance literature typically refers to a bridge system, a K-out-of-N:G system, or a system that is comprised of multiple structures (e.g., series-parallel) [43]. In this work a complex network refers to a network that cannot be represented by a combination of series, parallel, or other well-known configurations.

In extending the selective maintenance problem to a WSN approximation methods such as a Monte Carlo simulation or an estimate for a reliability bound might be considered. However relying on such an approach that requires repeated implementation to optimize a policy is not computationally tractable.

We address the complexity present in a reliability objective by incorporating the destruction spectrum (D-spectrum) to estimate network reliability [44]. In the presence of independent and identically distributed (i.i.d.) sensor failures the D-spectrum is only a function of the network structure, and does not depend on the failure distribution of sensors in the network [45]. While it is possible to compute the D-spectrum of a network exactly it is more common to use an approximation method, particularly when applied to a large, complex system. A Monte Carlo estimation of the D-spectrum has been shown to be more efficient compared to a traditional Monte Carlo simulation that directly estimates network reliability [46]. The lower computational effort required in estimating the D-spectrum, algorithms of which are outlined in [47] and [48], becomes significant when reliability estimation is embedded in an optimization problem and may need to be repeated over a large number of replications. The D-spectrum has received significant attention in network reliability literature, but its application in a maintenance setting is still emerging. The D-spectrum is applied in [49] to develop a preventive maintenance policy for a network subject to external shocks causing node failures with equal probability. The D-spectrum is incorporated in an expected cost-estimate dependent upon either a preventive maintenance action if components are repaired prior to network failure, or emergency repair if the network has failed. The resulting policy determines the number of failed components before a preventive maintenance action is necessary to minimize the long-run cost.

Research that is most closely related to this paper and addresses elements from each of the previous topics is found in [47] and [50]. In [47] a time-based deployment policy (TBDP) for a WSN is explored where the network is restored to a fixed size at periodic time intervals. Sensor nodes are randomly deployed in the network, and the D-spectrum is incorporated to estimate both the cost and WSN reliability over a wide range of deployment policies. Closely related to a TBDP is one in which a fixed number of sensors are deployed in the network at constant time intervals. This now results in a varying network size, but [50] address how the D-spectrum remains valuable in estimating

WSN reliability. The myopic condition-based deployment policy in [50] deploys new sensors to maximize reliability for a single mission, without considering the impacts on future missions. To the best of our knowledge, [47, 50] are the only sensor node redeployment policies in the literature to optimize WSN reliability over time. In this work we discuss how the D-spectrum can be adapted into a model to estimate WSN reliability in the presence of a condition-based deployment policy in which the decision also includes the number of new sensor nodes to deploy in the network, and complications that arise concerning a dynamic network topology, dynamic network size, and a dynamic age composition of sensors.

Our research extends the prior work [47, 50] by formulating the node deployment problem as an MDP. Although MDPs have been applied to a variety of WSN problems [51], the work of [15] appears to be the only previous MDP model focusing on the problem of replacing failed nodes over time. We note that the MDP of [15] makes a significant assumption that all failed nodes equally affect WSN performance, thereby disregarding network topology. By comparison, our work specifically considers network topology within the MDP.

3 Problem Description and Model

In this section we discuss a condition-based node deployment MDP model in which a limited budget is available to deploy additional sensors in the network. The WSN, represented by $\mathcal{G}$, is comprised of a collection of sensor nodes and a sink node deployed throughout a region of interest. Sensor nodes in the network are responsible for communicating with neighboring nodes to route information through the network, with a desired destination at the sink node. In addition to a communication capability, sensor nodes are tasked with monitoring the surrounding area and desired target locations in the region. We assume a unit disk graph model in which a pair of sensor nodes can communicate directly if their distance from each other is no more than d_1. Similarly, we assume a functioning sensor node can monitor any target within a distance of d_2.

For a target to be covered in the network it must not only be within the monitoring radius of a functioning sensor; there must also be a communication path through a sequence of functioning sensor nodes (that can communicate directly) from the monitoring sensor back to the sink node. The ability of sensors to communicate with one another declines over time as a result of sensor node failures, which also impacts the collection of targets covered. The lifetime of an individual sensor node is modeled by a survival function $\bar{F}(t) = 1 - F(t)$, where $F(t)$ represents the cumulative distribution function (cdf) of sensor node lifetime and is assumed to be identically distributed for all sensors. At time $t \geq 0$, the WSN $\mathcal{G}$ is represented by $\mathcal{G}(t)$ and consists of sensors that remain functioning at time t. The proportion of targets covered, or WSN coverage, is denoted $C(\mathcal{G}(t))$ and informs the condition of the network.

Note that WSN coverage is dependent upon the number of targets within range of a functioning sensor node (influenced by d_2), and the ability of a sensor node within range of a target to route information to the sink node, communicating through multiple hops if necessary to route information over a longer distance (influenced by d_1). The survival function $\bar{F}(t)$ is defined for each individual

(a) (b) (c)

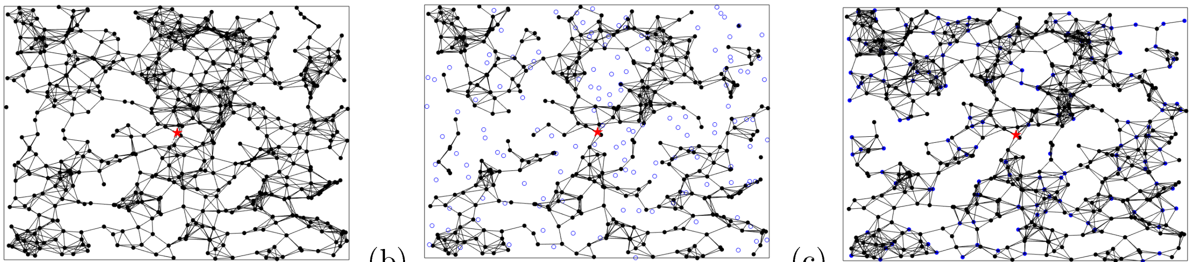

Figure 1: (a) Initial WSN with sink node ($\star$) and functioning sensor nodes ($\bullet$) ; (b) WSN with failed sensors ($\circ$) ; (c) WSN with newly deployed sensors ($\bullet$).

sensor node, and impacts $C(\mathcal{G}(t))$ as nodes fail over time and network communication/monitoring capabilities degrade. The timing and locations of newly deployed sensor nodes also impact $C(\mathcal{G}(t))$ as the new sensors may reestablish coverage over certain targets and/or restore connectivity with a group of sensor nodes that remain functioning from previous missions but were isolated from the sink node due to sensor node failures along a communication path. The redeployment decision must consider these factors in order to maximize WSN reliability at key points in time, where reliability is defined as $P[C(\mathcal{G}(t))] \geq \alpha$ for a given coverage requirement α.

An example of the WSN evolution over time is illustrated in Figure 1. In Figure 1(a) the WSN contains a large number of sensors and covers a significant portion of the region. Over time sensors fail and can dramatically impact network performance, as illustrated in Figure 1(b). To prevent a further drop in coverage and restore network capability, new sensors are deployed in the network, demonstrated in Figure 1(c). New sensors can be deployed in the network with an objective to improve the ability of sensors to communicate with one another, in addition to re-establishing coverage in portions of the network that were severely impacted by failures.

The desire of deploying new sensors in the WSN is to enable the region of interest to be monitored over a sequence of missions $\{0, 1, \ldots, M-1\}$. Each mission is of equal duration δ, and mission m corresponds to the duration of time between $m\delta$ and $(m+1)\delta$. Additionally, it is assumed that the initial network is provided (i.e., node positions at time $t = 0$ are known). The first redeployment action therefore corresponds to mission 1 at time $t = \delta$. At the beginning of mission 1, and each subsequent mission, the network is observed and a decision is then made on how many new sensors are deployed in the network. In our discussion throughout we adopt the convention that network observation and the deployment of any new sensors always occur at the beginning of a mission. Since the end of mission $m-1$ corresponds to the beginning of mission m, an equivalent statement is that the network is observed at the end of mission $m-1$, the deployment of new sensors occurs, and then mission m starts. For consistency purposes and ease of state variable and decision variable definitions introduced later, we always refer to both actions occurring at the beginning of a mission.

3.1 Deployment Actions and Template Structures

To avoid the computational effort in explicitly modeling the location of each newly deployed node, we instead allow the decision maker to specify a subregion into which each new node is deployed, and assume the new node is deployed randomly and uniformly within that subregion. Accordingly, the region of interest is partitioned into a number of subregions represented by the index set $\mathcal{R} = \{1, 2, \ldots, r\}$. As specified in Definition 1 and Assumption 1, we further restrict the redeployment decision by assuming new nodes are assigned to a given subregion based on the network size and the current number of sensor nodes in a subregion.

Definition 1. For a given network size, the *template structure* for a WSN specifies how many sensor nodes should be located in each subregion. That is, if there are n sensor nodes in the network, a *template structure*, $h^n = (h^n(1), h^n(2), \ldots, h^n(r))$, specifies the number of nodes, $h^n(i)$, located in each subregion $i \in \mathcal{R}$.

Definition 1 introduces the idea of a template structure, which might be informed by desired performance goals (e.g., a highly reliable network). For example, sensor nodes located near the sink node contribute significantly in routing information from nodes farther away that cannot directly communicate with the sink node. Therefore, it might be desirable to deploy sensor nodes in a higher density near the sink to provide redundant communication paths, and prevent a single node failure from disconnecting a large portion of the WSN. In addition to influencing the initial deployment of sensors, a template structure(s) can be informative in a WSN maintenance policy that deploys new sensor nodes by advising the subregion new sensor nodes should be deployed in.

Assumption 1. When new sensor nodes are deployed in the WSN, the number of new nodes deployed in each subregion is determined by the template structure for the resulting network size. That is, suppose there are currently n_i sensor nodes located in each subregion $i \in \mathcal{R}$. Let $n' = \sum_{i\in\mathcal{R}} n_i$ and suppose we wish to deploy n'' new nodes. Let z_i denote the number of nodes deployed to subregion $i \in \mathcal{R}$. We choose $z_i,\ i \in \mathcal{R}$, to minimize $\max\{h^{n'+n''}(i) - n_i - z_i : i \in \mathcal{R}\}$. Note that the resulting number of nodes $n_i + z_i$ in each subregion $i \in \mathcal{R}$ will be equal to $h^{n'+n''}(i)$ unless $n_j > h^{n'+n''}(j)$ for some subregion $j \in \mathcal{R}$, in which case we ensure the resulting number of nodes in each subregion $i \in \mathcal{R}$ is not too much smaller than $h^{n'+n''}(i)$.

To illustrate Assumption 1, suppose there are $r = 3$ subregions with $(n_1, n_2, n_3) = (3, 4, 6)$ nodes currently in each subregion. Note that $n' = n_1+n_2+n_3 = 13$ and suppose we wish to locate $n'' = 7$ new nodes. Suppose the template structure for size $n = n' + n'' = 20$ is $(h^{20}(1), h^{20}(2), h^{20}(3)) = (6, 9, 5)$. Then the new nodes will be assigned to subregions according to either $(z_1, z_2, z_3) = (2, 5, 0)$ or $(3, 4, 0)$. Note that in this example it is not possible to deploy new nodes in a manner that achieves the template structure for a 20 node network. However, either of these actions ensures that the resulting number of nodes $n_i + z_i$ in each subregion $i \in \{1, 2, 3\}$ is no more than 1 less than $h^{20}(i)$.

Assumption 1 leverages Definition 1 by using the template structure to inform the subregion new sensor nodes should be deployed in. This assumption simplifies the decision problem in that the primary decision must now address how many sensor nodes to deploy, and the resulting network size and template structure will influence the subregion a new node is deployed in.

Collectively, Definition 1 and Assumption 1 support the idea there there is some insight beforehand into how the network should be designed, and the decision should reflect that new nodes are deployed to achieve something close to this design over time. Even when exact sensor node placement is not feasible introducing a number of subregions and defining a template structure allows new nodes to be deployed with varying density throughout the network, the advantages of which are explored in [52].

It is important to note that the deployment action (specifically the subregion new nodes are deployed in) is determined by the template structure, however selecting the "best" or "optimal" template structure is not a trivial task. In fact, one of the implications of Assumption 1 is that the decision avoids the complexity present in an optimal network design problem as well, allowing our model to place a larger focus on the impact of WSN maintenance, specifically the timing and the number of nodes to deploy.

We recognize that there are a large number of strategies in defining subregions (both the number and size) as well as approaches to inform the template structure. Given the complexity present by addressing network reliability and a series of actions to maintain a WSN over time, Definition 1 and Assumption 1 are present to avoid introducing additional levels of difficulty in the model and/or decision. However, they are designed to be flexible and allow the model to address relaxations of these assumptions in future work. In this manner, one of the benefits of introducing subregions is that the model is flexible and able to address the scenario in which sensors are randomly deployed over the entire region (i.e., $r = 1$), as well as scenarios in which a more controlled deployment (i.e., $r > 1$) is possible [52, 53]. The number of subregions can also be influenced by the application of the WSN, thus the model is not dependent on a specific number of subregions. The focus of this work is therefore not on the number of subregions or how to optimally partition the region, but rather allow the model to address these different scenarios.

3.2 MDP Formulation

When new sensors are deployed in the WSN, a fixed cost c_F is incurred if at least one sensor is deployed in addition to a variable cost c_V for each sensor deployed. The fixed-plus-variable cost model relates to the hardware plus non-hardware model discussed in [15], and is also used in a related work investigating time-based redeployment policies [47]. It is assumed that all sensors deployed in the network are homogeneous, in the sense that all sensor capabilities are identical and sensors follow an i.i.d. failure distribution, F.

Since new sensors are deployed in the network over a sequence of missions, the collection of sensors is heterogeneous in the sense that sensors have different ages, and therefore different residual life distributions. Let k be the age of a sensor in the network, where sensors are deployed with initial age $k = 0$. The age of a sensor therefore corresponds to how many missions the sensor has survived. Define $\mathcal{K} = \{0, 1, \ldots, K\}$ as the set of all possible ages, where K is some upper bound on the age of a sensor in the network.

The state space consists of two main components, the first of which is the observed distribution of sensors in the network and is defined as

$$\boldsymbol{N}_m = (N_{m,i,k})_{i \in \mathcal{R}, k \in \mathcal{K}} \equiv (N_{m,1,0}, N_{m,1,1}, \ldots, N_{m,1,K}, N_{m,2,0}, \ldots, N_{m,r,K}), \tag{1}$$

where $N_{m,i,k}$ denotes the number of functioning sensors with age $k \in \mathcal{K}$ in subregion $i \in \mathcal{R}$ (immediately *prior* to the deployment of any new sensors) at the beginning of mission m . The total number of functioning sensors in the network is denoted by $\bar{N}_m = \sum_{i \in \mathcal{R}} \sum_{k \in \mathcal{K}} N_{m,i,k}$. The second component of the state space is the budget available to deploy sensors during mission m (and all future missions), denoted B_m. Combining these two components, the state of the system at the beginning of mission m is defined by $\boldsymbol{S}_m = (\boldsymbol{N}_m, B_m) \in \mathcal{S}$, where $\mathcal{S}$ is the set of all possible states.

After observing the state of the network, a decision must be made on how many new sensors are deployed. Let x_m denote the number of sensors deployed at the beginning of mission m. We define $\bar{x}_{mi}$ as the number of sensors deployed in subregion $i \in \mathcal{R}$ at the beginning of mission m, and note that $\bar{x}_{mi}$ is determined based on Assumption 1. The resulting cost from implementing action x_m is denoted $C_m(x_m)$, where

$$C_m(x_m) = \begin{cases} c_F + c_V x_m, & \text{if } x_m > 0, \\ 0, & \text{othwerwise.} \end{cases} \tag{2}$$

The transition probability functions can now be used to characterize how the system evolves from one state to another. First, note that an individual sensor with age k survives the current mission with probability

$$p_k = \frac{\bar{F}((k+1)\delta)}{\bar{F}(k\delta)}. \tag{3}$$

Using the survival probability for an individual sensor, the transition probability for the number of sensors with age k in subregion i is determined by

$$\Pr(N_{m+1,i,k} | N_{m,i,k-1}, x_m) = \begin{cases} b(N_{m+1,i,1}; \bar{x}_{mi}, p_{k-1}), & \text{if } k = 1 \text{ and} \\ & 0 \leq N_{m+1,i,k} \leq \bar{x}_{mi}, \\ b(N_{m+1,i,k}; N_{m,i,k-1}, p_{k-1}), & \text{if } k > 1 \text{ and} \\ & 0 \leq N_{m+1,i,k} \leq N_{m,i,k-1}. \end{cases} \tag{4}$$

where $b(n; x, p)$ is the binomial probability of n successes in x trials with probability of success p. The overall transition probability given maintenance action x_m can now be determined by

$$\Pr(\boldsymbol{N}_{m+1} | \boldsymbol{N}_m, x_m) = \prod_{i \in \mathcal{R}} \prod_{k \in \mathcal{K}} \Pr(N_{m+1,i,k} | N_{m,i,k-1}, x_m). \tag{5}$$

The second component of the state variable is the budget, which transitions based on the corresponding cost of the action implemented,

$$B_{m+1} = B_m - C_m(x_m). \tag{6}$$

The state transition function is defined as $\boldsymbol{S}_{m+1} = S^M(\boldsymbol{S}_m, x_m, \boldsymbol{W}_{m+1})$, where $\boldsymbol{W}_{m+1}$ represents information on sensor failures that occur during mission m.

Given a starting budget B_0, the objective is to deploy sensors in the network to maximize the expected number of successful missions. For a given coverage requirement α, an individual mission is successful if WSN coverage over the duration of the mission remains above this requirement. Network reliability is also defined with respect to α, and is defined as the probability the coverage requirement is satisfied over the mission duration. From an observed network state $\boldsymbol{S}_m$ and implementing action x_m, the resulting network reliability is denoted $R_m(\boldsymbol{S}_m, x_m) = P[C(\mathcal{G}(m\delta + \delta)) \geq \alpha]$ where $m\delta + \delta$ refers to the period of time at the end of mission m *prior* to the deployment of new sensors at the beginning of the subsequent mission. Let $X_m^\pi(\boldsymbol{S}_m)$ be a policy that determines the sensor deployment action (when and how many sensor nodes are deployed) for each state $\boldsymbol{S}_m \in \mathcal{S}$. For a given number of missions M, the objective is

$$\max_{\pi \in \Pi} \mathbb{E}^\pi \Big\{ \sum_{m=0}^{M-1} R_m(\boldsymbol{S}_m, X_m^\pi(\boldsymbol{S}_m)) \Big\}. \tag{7}$$

Constraining a decision each mission is first the budget available, B_m, to deploy sensors in the network. Additionally, there may be some desired minimum reliability (i.e., probability of mission success), ϕ, that each mission must achieve. This constraint is intended to prevent the scenario where network reliability is completely sacrificed (i.e., unacceptably low reliability and almost certain network failure) one mission, while the reliability of a later mission is near one. Finally, there may exist an upper limit on the number of sensors allowed in the network, n^{max}, to prevent the region from becoming saturated with sensors at any given time. Overall the set of feasible actions, $\mathcal{X}_{\boldsymbol{S}_m}$, during mission m is therefore defined by

$$\mathcal{X}_{\boldsymbol{S}_m} = \Big\{ x_m : C_m(x_m) \leq B_m, R_m(\boldsymbol{S}_m, x_m) \geq \phi, \bar{N}_m + x_m \leq n^{max} \Big\}. \tag{8}$$

A complicating aspect in determining the set of feasible actions is the reliability requirement an action must satisfy. Because network reliability problems commonly fall in the #P-Compete class of problems determining the exact set of feasible actions as defined by (8) is not a trivial task. Section 3.3.1 addresses this difficulty by outlining an efficient method to estimate network reliability and instead apply the constraint to the estimated reliability of an action, $\hat{R}_m(\boldsymbol{S}_m, x_m)$. In doing so the set of feasible actions is now approximated as well, and it is possible our approximation includes actions that are not feasible to (8). That is, the estimated reliability of an action may satisfy the constraint and therefore appear in our approximated action set, but the true value might be below the requirement. However, this should only occur for a small number of actions, and the actions *are* feasible to the original problem with only a cost constraint.

The value function, $V_m(\boldsymbol{S}_m)$, is defined as the maximum number of successful missions remaining among missions $m, m+1, \ldots, M-1$ if the system is in state $\boldsymbol{S}_m$ at the beginning of mission m. To determine an optimal policy to (7) we must find a solution to Bellman's equation,

$$V_m(\boldsymbol{S}_m) = \max_{x_m \in \mathcal{X}_{\boldsymbol{S}_m}} \Big\{ R_m(\boldsymbol{S}_m, x_m) + \mathbb{E}\big[V_{m+1}(\boldsymbol{S}_{m+1}) \big| \boldsymbol{S}_m, x_m\big] \Big\}. \tag{9}$$

3.3 ADP Formulation

The previous section provides an initial MDP model for the condition-based sensor deployment problem over a sequence of M missions. Common to many dynamic programming problems, this model suffers from the curses of dimensionality [54]. The large size of the state space can be illustrated by examining the distribution of sensor nodes in the network. For a network containing i sensor nodes, these nodes can be allocated to different subregions of the network $\binom{r+i-1}{i}$ different ways. Due to node failures and the deployment of new sensor nodes, the total number of sensor nodes in the network also varies between 0 and n^{max}. As a result, the size of the state space considering only the distribution of sensor nodes in the network is $\sum_{i=0}^{n^{max}} \binom{r+i-1}{i}$. Note that this does not include any information about the age composition of nodes, which further complicates the size of the state space. The remaining budget is also a factor, and can be bounded between 0 and B_0. Assuming integer values of c_F and c_V then the budget for mission m can also assume integer values between 0 and B_0, and the size of the state space can be bounded by $B_0 \sum_{i=0}^{n^{max}} \binom{r+i-1}{i}$ for a single mission. The large deployment action space and outcome space (i.e., observing sensor node failures) are additional components that limit exact algorithms to be applied for only small problem instances.

For large-scale WSNs of interest, ADP can be applied to the condition-based sensor deployment problem. First, the optimality equations can be reformulated around the post-decision state variable, $\boldsymbol{S}_m^x = (\boldsymbol{N}_m^x, B_m^x)$, which is the state at the beginning of mission m immediately after new sensor nodes have been deployed. In the post-decision state, the number of sensors functioning in each subregion immediately after new nodes have been deployed and the total number of sensor nodes in the network are represented by $\boldsymbol{N}_m^x$ and $\bar{N}_m^x$, respectively. Analogous to Equation 1, $\boldsymbol{N}_m^x$ is a vector with rK components such that $N_{m,i,k}^x$ refers to the post-decision number of functioning sensor nodes of age $k \in \mathcal{K}$ in subregion $i \in \mathcal{R}$. Collectively, the post-decision state variable $\boldsymbol{S}_m^x = (\boldsymbol{N}_m^x, B_m^x)$ is defined by (10) and (11) below.

$$N_{m,i,k}^x = \begin{cases} \bar{x}_{mi} & \text{if } k = 0 \\ N_{m,i,k} & \text{if } k \in \{1, 2, \ldots, K\} \end{cases} \tag{10}$$

$$B_m^x = B_m - C_m(x_m). \tag{11}$$

Similarly, B_m^x refers to the remaining budget after implementing the deployment action. Let $V_m^x(\boldsymbol{S}_m^x)$ denote the value of being in the post-decision state $\boldsymbol{S}_m^x$, and is defined as the maximum number of successful missions among missions $m+1, m+2, \ldots, M-1$ given the post-decision state variable $\boldsymbol{S}_m^x$. The relationship between V_m^x and V_m is given by

$$V_{m-1}^x(\boldsymbol{S}_{m-1}^x) = \mathbb{E}[V_m(\boldsymbol{S}_m)|\boldsymbol{S}_{m-1}^x], \tag{12}$$

where

$$V_m(\boldsymbol{S}_m) = \max_{x_m \in \mathcal{X}_{\boldsymbol{S}_m}} \Big\{ R_m(\boldsymbol{S}_m, x_m) + V_m^x(\boldsymbol{S}_m^x) \Big\}. \tag{13}$$

Substituting (13) into (12) we obtain the optimality equations around the post-decision state variable

$$V_{m-1}^x(\boldsymbol{S}_{m-1}^x) = \mathbb{E}\Big\{ \max_{x_m \in \mathcal{X}_{\boldsymbol{S}_m}} \big(R_m(\boldsymbol{S}_m, x_m) + V_m^x(\boldsymbol{S}_m^x) \big) | \boldsymbol{S}_{m-1}^x \Big\}. \tag{14}$$

One of the advantages of utilizing the post-decision state variable is the expectation is now outside of the maximization problem. The resulting maximization problem in (14) is less complicated than the original formulation in (9), but still requires an evaluation of network reliability. Due to Assumption 1 the network structure with respect to the post-decision state $\boldsymbol{S}_m^x$ will be similar to the template structure given by Definition 1. Based on this observation, the following section describes an approach for approximating $R_m(\boldsymbol{S}_m, x_m)$ based on estimating the D-spectrum of the template structure.

3.3.1 Destruction Spectrum Reliability Estimation

The D-spectrum is an approach to evaluating reliability, $R_m(\boldsymbol{S}_m, x_m)$, which appears both in Equation (8) and (14). The action of deploying new sensors in the network influences the network structure and the number of sensors functioning in each subregion of the network. From information available in the post-decision state variable we can apply the network D-spectrum to estimate reliability, but must first define a number of state aggregation functions. Let $\mathcal{S}^{(a)}$ be the state space at the ath level of aggregation, where the aggregation function A^a maps the original state space $\mathcal{S}$ to $\mathcal{S}^{(a)}$. Define A^1 as the function that aggregates over the age composition of sensors in a subregion, resulting in the number of sensors in each subregion, $N_m^{(1)} = (N_{m1}^{(1)}, N_{m2}^{(1)}, \ldots, N_{mr}^{(1)})$. The second aggregation function, A^2, aggregates over the subregions in the network, resulting in the number of sensors with a given age, $N_m^{(2)} = (N_{m0}^{(2)}, N_{m1}^{(2)}, \ldots, N_{mK}^{(2)})$.

Applying the first aggregation function to the post-decision state variable, we can determine the number of sensors functioning in each subregion which informs the current WSN structure. Alternatively, as a result of Assumption 1 this closely matches a predefined template structure. This is of significance because we can estimate the D-spectrum for each of our template structures, an in turn WSN reliability from the post-decision state. The D-spectrum estimate with respect to template structure h^n is denoted $\hat{s}_{\alpha,i}^{h^n}$, and is the probability the ith sensor failure results in network coverage falling below the requirement α in a network where there are $h^n(j)$ sensor nodes randomly and uniformly located in subregion $j \in \mathcal{R}$. A Monte Carlo simulation is implemented to estimate the D-spectrum, which further illustrates the value of Assumption 1. Instead of requiring a Monte Carlo simulation repeatedly throughout the MDP/ADP model, the D-spectrum is only required for the template structures which can be estimated prior to solving the model, avoiding the need to constantly estimate the D-spectrum within the model itself.

From the second aggregation function we can determine the probability of randomly selecting a sensor with age k in the network by

$$\tilde{\rho}_k = \frac{N_{mk}^{x,(2)}}{\bar{N}_m^x}, \ k \in \mathbb{Z}_{\geq 0}. \tag{15}$$

With (15), the residual life distribution for a sensor randomly selected in the network is now given by the cdf

$$\tilde{G}(t;\delta) = \sum_{k=0}^{\infty} \frac{F(k\delta+t)-F(k\delta)}{\bar{F}(k\delta)}\tilde{\rho}_k, \tag{16a}$$

$$= \sum_{k=0}^{\infty} \frac{F(k\delta+t)-F(k\delta)}{\bar{F}(k\delta)}\frac{N^{x,(2)}_{mk}}{\bar{N}^x_m}, \tag{16b}$$

and note that this follows a development similar to that in [47], based upon [55]. From the D-spectrum estimate and residual life distribution in (16b), network reliability over the next mission, given the observed state $\boldsymbol{S}_m$ and action x_m can be estimated by

$$\hat{R}_m(\boldsymbol{S}_m, x_m) = \sum_{i=1}^{\bar{N}^x_m} \hat{s}^{h^n}_{\alpha,i} B(i-1; \bar{N}^x_m, \tilde{G}(\delta;\delta)), \tag{17}$$

where $B(i-1; \bar{N}^x_m, \tilde{G}(\delta;\delta))$ is the cumulative binomial probability distribution of no more than $i-1$ successes in $\bar{N}^x_m$ trials with probability of success $\tilde{G}(\delta;\delta)$ [45].

One of the limitations of the proposed approach to reliability estimation is that it uses the stable residual life distribution derived in (16b), which relies on a probability distribution of sensor ages aggregated over the entire network. Since we observe information on the age distribution of sensors within a subregion, it is reasonable to question why this level of detail is not retained and incorporated in our estimation method. That is, the residual life distribution can be subregion dependent and more accurately reflect the state of the network. A disadvantage of this approach is it now requires an application of the multi-dimensional D-spectrum [56] which is more complicated to estimate. To alleviate introducing further complexity into the model, we leave an in-depth investigation for this consideration for future work.

3.3.2 Value Function Approximation

Due to the large state space, we approximate the value function through the use of the previously defined aggregation functions and lookup tables. This is based on the observation that the age composition of sensors in the network and the distribution of sensors contribute greatly to the size of the state space. The former is necessary to estimate the stable residual life distribution while the latter is necessary to estimate the destruction spectrum, both of which are required to estimate reliability of the current mission. It is reasonable to expect that while both of these components will impact future missions as well, the primary factor impacting future missions can be summarized by the number of nodes in the network. Therefore, we aggregate over the age composition and subregion distribution of sensors in approximating the value function. This is defined as the aggregation function $A^3 \equiv \sum_{i\in\mathcal{R}}\sum_{k\in\mathcal{K}} N_{m,i,k}$, which is equivalent to $\bar{N}_m$.

Additionally, the starting budget B_0 influences the size of the state space and impacts the ability to deploy new sensors in the network. Assuming the variable cost of deploying additional sensors is relatively small (particularly compared to the total cost), deploying one or two additional sensors has a minor impact on the budget remaining. It is also reasonable to assume that the impact of deploying one or two additional sensors has a minor increase to the overall value function, particularly when compared to the impact of deploying 15 to 20 additional sensors. As a result we can aggregate the budget into different intervals corresponding to a range of values that result in a similar state value. If the budget is aggregated into intervals of size d, there are now $\bar{B}_0 = \lceil \frac{B_0}{d} \rceil$ different budget states.

The approximate value function for a given post-decision state $\boldsymbol{S}_m^x$ is denoted $\bar{V}_m(\boldsymbol{S}_m^x)$, and with an aggregated state space size of approximately $\bar{B}_0 \times n^{max}$ is significantly smaller than the original state space. We recognize that there are alternative methods to approximate the value function (e.g., using basis functions with a regression model, nonparametric models, etc.), and a look-up table significantly simplifies this step. For the results in Section 4 we will demonstrate that the resulting CBDP performs favorable in comparison to existing node deployment policies [47, 50].

3.3.3 Determining An Optimal Action

The primary question that remains is addressing how the maximization problem in (14) is solved for the optimal value and corresponding action. From the observed state $\boldsymbol{S}_m$, we can first determine an upper bound on the number of sensors that can be deployed by $\tilde{n} = n^{max} - \bar{N}_m$ (assuming the budget does not limit us first). This results in a range of $(0, \tilde{n})$ to search for the the number of new sensor nodes to deploy. Since the D-spectrum is independent of the failure distribution of sensors in the network, the reliability for each post-decision state evaluated in this search can be quickly estimated without re-evaluating the D-spectrum. The only step that is required is to update the residual life distribution (16b), after which reliability can be estimated by applying (17). While estimating the D-spectrum is more efficient than a traditional Monte Carlo simulation to estimate reliability, repeatedly estimating the D-spectrum for different network structures becomes computationally burdensome. With Assumption 1, we can estimate the D-spectrum for template structures over a range of network sizes (e.g., for a network with 300 to n^{max} sensors) once at the very beginning of the problem and store the estimates for use later in the ADP model.

As a further enhancement, we revisit the discussion from Section 3.3.2 in which we noted that a single additional sensor has a minor impact on network reliability and the future number of successful missions. Based on this observation, we can search the range $(0, \tilde{n})$ in intervals of d nodes, i.e., resulting in approximately $\tilde{n}/d$ reliability evaluations instead of the $\tilde{n}$ evaluations that would be required to do a complete search of $(0, \tilde{n})$.

3.3.4 Initializing the Value Function

A more simplistic policy considers the impact of deploying sensors on only the upcoming mission. This is a version of a myopic policy, and can be informative in our ADP formulation as well. Since a myopic policy is interested in reliability of a single mission, the policy will always deploy sensors until a constraint limits the action. That is, the myopic policy will never skip a deployment opportunity, and deploy sensors every mission until a constraint is reached (e.g., budget no longer available or maximum network size reached). When considering a myopic policy it is therefore more appropriate to consider, or allocate, a small budget to each mission to ensure there is a budget available to missions near the end of the planning horizon as well. A myopic CBDP is explored in [50], and is of value to our ADP model in two ways. First, as discussed in [50], a myopic CBDP results in a relatively consistent network size. Applying Assumption 1 when a fixed number of sensor nodes are deployed across all missions results in a consistent template structure over time. Through a comparison with an ADP policy we can now highlight the significance of allowing greater control on the number of sensor nodes deployed (and therefore budget allocated to) each mission. Second, the resulting reliability estimate of a myopic CBDP can be of value in the ADP formulation to initialize the value function. In the ADP problem, if there is a budget B_m remaining then one option is to evenly allocate this budget to the remaining $M-1-m$ missions. This essentially corresponds to a myopic policy with each mission receiving $\frac{B_m}{M-1-m}$ of the budget. The reliability of the myopic policy can then be used to estimate the number of successful missions in the remaining $M-1-m$ missions and initialize the value function.

3.4 Approximate Value Iteration Algorithm

Algorithm 1 outlines an approximate value iteration (AVI) algorithm utilizing a value function approximation based on a lookup table representation on the aggregated state space, adapted from [54]. The AVI algorithm updates our value function approximation over a sequence of iterations $y = 1, 2, \ldots, Y$, which in turn updates the CBDP. $\boldsymbol{S}_m^y$ represents the observed state at the beginning of mission m in iteration y, and $\boldsymbol{S}_m^{x,y}$ represents the post-decision state variable given action x_m. $\bar{V}_{m-1}^y(\boldsymbol{S}_{m-1}^{x,y})$ represents the value function approximation for the post-decision state variable $\boldsymbol{S}_{m-1}^{x,y}$ during iteration y, and is updated based on the step size parameter η_y. While Algorithm 1 outlines a relatively standard AVI algorithm, we hope to show that the resulting CBDP are a significant improvement over both a myopic condition-based deployment policy and a time-based deployment policy. As this is also one of the first ADP applications for the maintenance of a complex WSN with respect to a reliability evaluation, the performance of the AVI algorithm can identify components of the model to focus more on in future work.

4 Numerical Example

In this section we illustrate the performance of the ADP formulation and provide results for a number of test instances. The lifetime of each sensor node is distributed according to a Weibull distribution, which is also selected to model failures in [57] and [27], with a shape parameter $\beta = 1.5$ and a scale parameter $\lambda = 10$. Sensor capabilities are defined by on a common communication radius $d_1 = 0.075$ and a monitoring radius of $d_2 = 0.075$. Values for the sensor node capabilities are selected to provide a notional instance with reasonable parameter values. An increasing failure rate (IFR) distribution (i.e., $\beta > 1$) is selected to reflect that the expected remaining life of a sensor node should decrease as the node consumes limited available energy. In practice, the scale parameter λ would depend upon the hardware, application, and environment; however, since an equivalent problem results upon scaling λ and δ by a constant factor the results are easily generalizable to other values of λ. Values for the communication and monitoring radius are selected to provide a

Algorithm 1 AVI for Finite Horizon Problem Using the Post-Decision State

1: **function** AVI

2: Initialization: approximation of the value function $\bar{V}_m^0(\boldsymbol{S}_m^x)$ for all post-decision states, and an initial state $\boldsymbol{S}_0^{x,1}$. Set $y = 1$.

3: For $m = 0, 1, 2, \ldots, M-1$,

4: Determine $\hat{v}_m^y$ by

$$\hat{v}_m^y = \max_{x_m \in \mathcal{X}_{\boldsymbol{S}_m}} \left(R_m(\boldsymbol{S}_m^y, x_m) + \bar{V}_m^{y-1}(\boldsymbol{S}_m^{x,y}) \right)$$

and let x_m^y be the optimal action.

5: Update $\bar{V}_{m-1}^{y-1}$ using

$$\bar{V}_{m-1}^y(\boldsymbol{S}_{m-1}^{x,y}) = (1 - \eta_{y-1})\bar{V}_{m-1}^{y-1}(\boldsymbol{S}_{m-1}^{x,y}) + \eta_{y-1}\hat{v}_m^y.$$

6: Sample $\boldsymbol{W}_{m+1}^y$ and compute the next state $\boldsymbol{S}_{m+1}^y = S^M(\boldsymbol{S}_m^y, x_m^y, \boldsymbol{W}_{m+1}^y)$.

7: Increment y. If $y \leq Y$ go to step 3.

8: Return the value functions $(\bar{V}_m^n)_{m=0}^{M-1}$.

9: **end function**

balance between the capability of an individual node, and the number of sensor nodes required for overall network function.

The cost of deploying sensors in the network is determined by the variable cost $c_V = 1$, with a fixed cost $c_F = 100$ incurred each time one or more sensors are deployed. Fixed and variable costs are selected to balance in the ratio of the fixed cost of accessing the network, which may be large when the network is in a hostile environment, and the individual cost of a single sensor node. The region of interest is a $[0, 1] \times [0, 1]$ unit square that is partitioned into $r = 16$ equal sized subregions of size 0.25×0.25. This partitioning of subregions is selected to provide the model flexibility in focusing the deployment of new nodes either toward the middle of the region or toward the boundaries as needed. Additionally, 441 targets are uniformly spaced as a 21×21 grid representing target locations where the WSN must provide coverage. The number of targets and their distribution is selected to ensure coverage throughout the entire region is sufficiently captured.

In the results that follow the number of sensor nodes in a given subregion of the WSN, which defines the template structure, is based on a subregion weight and is inversely proportional to (1) the distance from the center of the subregion to the sink node, and (2) the probability that all sensor nodes in a subregion are connected (see [58] for details). Defining template structures in this manner is influenced by two factors. First, if the number of sensor nodes in each subregion is approximately equal, then it is desirable to deploy a new node near the sink and provide a level of redundancy in communication paths with the sink node. Second, if a subregion is farther away from the sink node but has a very small number of functioning sensor nodes then they are likely disconnected from one another and/or cover a small fraction of the subregion. Therefore we also desire to deploy a number of new sensor nodes in this subregion as well, and constructing template structures in this manner is designed to balance these two competing objectives.

Defining template structures in this manner also accounts for the overall size of the WSN. Smaller sized networks require a more uniform distribution of sensor nodes to balance coverage in exterior regions and sensor nodes located near the sink to support network connectivity. Meanwhile, once a sufficient number of sensor nodes are deployed in the exterior regions a larger sized network will focus more on the subregion surrounding the sink node in an attempt to increase the redundancy in communication paths.

The step size influences the rate at which the value function approximation is updated and the convergence of the AVI algorithm. Since the value functions are initialized with a myopic CBM policy, the initial step size for updating the value function approximation is $\eta_0 = 0.7$, and the step size is updated according to

$$\eta_y = \eta_0 \frac{a}{a + y - 1}, \tag{18}$$

with $a = 20$. This step size rule allows the rate at which η drops to zero to be influenced by the parameter a, with larger values slowing the rate at which η decreases.

For the test instances, the inspection interval δ varies among $\{2, 3, 4\}$, and the number of missions is selected so that the total time horizon ($M * \delta$) is approximately the same. The coverage requirement is set at $\alpha = 0.8$, meaning if the WSN covers less than 80% of target locations the network is in a 'failed' state. The maximum network size is also fixed at $n^{max} = 950$ sensors for every test instance, with an initial number of $\bar{N}_0 = 650$ sensors deployed in the region. Parameter values for each test instance, to include the starting budget, B_0, and reliability requirement, ϕ, are provided in Table 1. To force exploration in the decision space, each mission there is a 5% chance a random non-optimal deployment action is implemented.

Table 1 also provides performance results of Algorithm 1 with $Y = 300$ replications, where column 5 (labeled V_0) reports the expected number of successful missions from the resulting ADP policy. The final column in the table, labeled Monte Carlo Policy Evaluation (MC-PE), reports the average number of successful missions observed when the optimal ADP policy is evaluated through a Monte Carlo Simulation, assisting a later discussion on a comparison of the expected vs observed policy performance. Starting with $\delta = 4$ and the largest budget $B_0 = 8700$, the WSN is not overly strained and a sufficient number of new sensors can be deployed when needed to maintain the WSN at a high level. The budget is also large enough that enforcing a minimum reliability requirement on every mission has little impact on the performance of the optimal deployment policy. The next pair of test instance reduces the budget by $1,100$ which corresponds to a smaller number of sensors that can be deployed, and a larger emphasis on deploying sensors effectively to avoid the fixed cost consuming a large portion of the budget. While the budget is more constraining in this instance, the expected number of successful missions of 23.66 (23.99 without a reliability requirement) is still relatively high. The following pair of test instances result in a similar decline in WSN performance, particularly when a reliability requirement is present. Compared to the previous group of test

Table 1: Test Instances and Policy Performance

δ	M	B_0	ϕ	V_0	MC-PE
4	25	8700	0	24.97	24.95
4	25	8700	0.95	24.97	24.97
4	25	7600	0	23.99	23.85
4	25	7600	0.89	23.66	23.66
4	25	7400	0	23.13	22.69
4	25	7400	0.79	22.97	22.65
3	33	8050	0	31.89	31.71
3	33	8050	0.85	31.88	31.69
3	33	7650	0	29.45	28.14
3	33	7650	0.65	26.27	27.42
2	50	8700	0	49.95	49.89
2	50	8700	0.95	49.96	49.94
2	50	7600	0	48.54	47.54
2	50	7600	0.89	48.05	46.73
2	50	7400	0	47.19	45.55
2	50	7400	0.79	46.33	44.89

instances the budget has decreased slightly to $7,400$, while the decline in the expected number of successful missions is comparable to lowering budget from $8,700$ to $7,600$. This pair of test instances also help illustrate the value in providing a minimum reliability requirement for each mission. When no requirement is imposed and there is no penalty for WSN failure then network reliability for a given mission can be sacrificed to avoid the fixed cost. This allows a larger number of sensors to be deployed over the remaining missions. When the reliability requirement is set to $\phi = 0.79$ this ensures that the probability a single mission is successful is still relatively high and also has little impact on the expected number of successful mission over the planning horizon.

In the next grouping of test instances the inspection interval is lowered to $\delta = 3$, and for the total time horizon to remain approximately the same the planning horizon for the number of missions is increased to 33. The noticeable result from this grouping is again observed in the smallest budget instance with a reliability requirement in place. With a budget of $7,650$ and a minimum reliability requirement of $\phi = 0.65$, the expected number of successful missions is significantly smaller compared to the case when no requirement is in place. This is again a result of not penalizing WSN failure, and by sacrificing performance to avoid incurring the fixed cost the budget for the remaining missions is large enough to maintain a highly reliable network.

The last grouping of test instances contain the shortest inspection interval with $\delta = 2$ and the largest number of missions with 50, influencing the policy in a number of areas. With a smaller inspection interval the network is observed more frequently, and there is an opportunity to observe a network state that might fail during the next mission that would not be observed under a larger inspection interval. In this scenario, new sensors can be deployed to avoid the potential network failure, and the overall number of successful missions should increase. Alternatively, with a shorter time between inspections it might be more advantageous to avoid deploying sensors in the network if the reliability of the upcoming mission is already at a sufficient level. While this does not improve reliability for the next mission, the fixed cost is avoided and allows a larger number of sensors to be deployed in the network over the remaining missions. For the largest starting budget of $8,700$ the ADP policy again results in an expected number of successful missions that is near the total number. Even though the smaller inspection interval results in more frequent network observation and more flexibility in when sensors are deployed, the decline in the expected number of successful missions as the starting budget decreases remains noticeable.

4.1 Monte Carlo Policy Performance

The optimal CBDP identified by the ADP algorithm is also implemented in a Monte Carlo simulation to observe the average number of successful missions the policy achieves, and is reported in the "MC-PE" (Monte Carlo Policy Evaluation) column of Table 1. These results help demonstrate the performance of the deployment policy in a simulated setting obtain results close to the predicted values. In several of the test instances with a larger inspection interval the performance of the ADP policy matches the expected number of successful missions. The largest difference between the expected and observed number of successful missions occurs for the smallest budget and smallest inspection interval test instance. In this test instance, the observed number of successful missions is slightly smaller than the expected number. Observing the largest deviation in this test instance is somewhat expected since this corresponds to a more difficult scenario. A smaller δ results in more missions, which implies a larger number of decisions are made. This instance is also more resource constrained since it has the smallest budget. While the observed performance of the ADP policy does begin to deviate as the test instances become more difficult, the overall observed number of successful missions remains relatively high.

The observed MC-PE also provides a more appropriate comparison on the results for an inspection interval of $\delta = 4$ with an inspection interval of $\delta = 2$. For each test instance, the resulting ADP policy with an inspection interval of $\delta = 4$ is also a feasible policy for the corresponding $\delta = 2$ test instance. As a result, the observed number of successful missions in an optimal ADP policy for the $\delta = 2$ instance should be at least twice that of the corresponding $\delta = 4$ test instance. However in a majority of the test instances the observed number of successful missions for the $\delta = 2$ ADP policy is approximately double that of the corresponding $\delta = 4$ ADP policy, and is lower than expected in the $B_0 = 7400, \phi = 0.89$ test instance. This again highlights the difficulty of the test instance and the impact of reducing the time between network observations. When the network is inspected more frequently a larger number of deployment decisions must be made regarding when and how many sensors are deployed. The comparison in the observed performance of the ADP policy for different inspection intervals further demonstrates the complexity of a policy related to the repeated deployment of sensors in a WSN, and suggest there is an opportunity for future work to focus on improving a policy when the planning horizon increases.

4.2 ADP Comparison to Myopic Policy

In addition to initializing the value function, the myopic deployment policy provides a good comparison to demonstrate the improvement of the ADP policy. For this purpose, the myopic CBDP [50] is also implemented in a Monte Carlo simulation with a budget of B_0/M available to deploy sensors per mission. To the best of our knowledge this is only prior work that focuses on WSN reliability with region-based node redeployment over time. The observed number of successful missions for the myopic policy is provided in Table 2, along with the previous ADP results.

Table 2: Observed ADP and Myopic Policy Comparison

δ	M	B_0	ϕ	ADP MC-PE	Myopic Policy
4	25	8700	0.95	24.97	23.96
4	25	7600	0.89	23.66	21.44
4	25	7400	0.79	22.65	19.34
3	33	8050	0.85	31.69	28.31
3	33	7650	0.65	27.42	18.98

In each of the test instances the ADP policy results in a larger number of successful missions, and is more noticeable with a smaller budget. This result is somewhat expected since the ADP policy is allowed to deploy a variable number of sensors and reallocate the budget as necessary, saving when able and deploying a larger number of sensors when needed. However the magnitude of this improvement is quite significant particularly when the budget is more constraining, clearly seen in the instance with $\delta = 3$ and a budget of $B_0 = 7650$. With the small budget available in this instance the myopic policy performs quite poorly and only 19 of the 33 missions are successful, compared to the ADP policy which is able to achieve over 27 successful missions. A similar outcome is observed with an inspection interval of $\delta = 4$, in which the ADP policy again performs noticeably better than the myopic policy in each instance. The significant improvement of the ADP policy over a myopic policy illustrates the value of a deployment policy that considers the impact on network performance over a planning horizon, compared to traditional policies that focus on an immediate effect.

4.3 ADP Policy Investigation

We are also interested in investigating the impact any test instance parameters have on the resulting ADP Policy. One observation is that the optimal policy is more likely to skip a deployment opportunity (i.e., deploy zero sensors at the start of a mission) as the starting budget B_0 and/or the inspection interval δ decrease. For a large starting budget, it may be possible to incur the fixed cost every mission and still deploy a sufficient number of sensors to maintain a highly reliable network. As the budget decreases, the fixed cost of deploying sensors every mission consumes a larger proportion of the overall budget which results in fewer sensors deployed each mission. Therefore, it becomes more desirable to skip a maintenance opportunity when allowed to avoid the fixed cost, providing a larger budget over the remaining missions and increasing the proportion of the budget consumed by the variable cost, which equates to a new sensor in the network. Similarly, as the inspection interval decreases the amount of time the network must function until the next deployment window is also smaller. Compared to a larger inspection interval, it is likely that fewer sensors will fail in a shorter time interval and the network will more often be observed in a state providing the opportunity to skip sensor deployment while ensuring the upcoming mission remains successful with high probability.

The average percent of the budget consumed by the variable cost in each policy is reported in Table 3. For each test instance the column labeled "No Reliability Requirement" implies $\phi = 0$, while the column "With Reliability Requirement" refers to the non-zero reliability requirement for the corresponding test instance defined in Table 1. When $\delta = 4$, the significant drop in the starting budget between the first and second test instance impacts both the total number of missions in which sensors are deployed and the number of sensors deployed. However given the longer time between inspection intervals it is more difficult to skip a deployment opportunity and maintain a highly reliable network, which is observed by the decrease from 71.26% to 68.42% (71.41% to 67.25% with a reliability requirement) of the the overall budget dedicated to variable cost. Meanwhile, the budget allocation appears to be impacted less for the smaller inspection intervals. For example, when $\delta = 3$ the overall proportion of the budget consumed by the variable cost is approximately the same when the starting budget decreases from $8,050$ down to $7,650$. Additionally, for the inspection

Table 3: Percent of Budget Dedicated to Variable Cost

δ	M	B_0	No Reliability Requirement	With Reliability Requirement
4	25	8700	71.26%	71.41%
4	25	7600	68.42%	67.25%
4	25	7400	68.52%	67.38%
3	33	8050	61.00%	61.02%
3	33	7650	62.51%	61.24%
2	50	8700	70.75%	70.61%
2	50	7600	69.53%	69.71%
2	50	7400	69.94%	70.21%

interval $\delta = 2$ the decrease in the percent of budget allocated to the variable cost is not as significant compared to the larger interval of $\delta = 4$. This result is somewhat expected since the network does not have to operate as long until the next deployment decision, and there is more flexibility for the ADP policy to control when sensors are deployed in the network providing a better balance between the fixed and variable cost.

The discussion at the end of Section 4.1 also highlighted the difficulty encountered in the $\delta = 2, B_0 = 7400, \phi = 0.79$ test instance. Compared to the corresponding test instance with $\delta = 4$, a larger proportion of the overall budget is allocated to the variable cost under the smaller inspection interval of $\delta = 2$. This suggests that, as expected, the ADP policy in the $\delta = 2$ instance is skipping a deployment opportunity more often, but based on the observed policy performance compared to the $\delta = 4$ policy is struggling to do so in the most effective manner. This suggests that the ADP policy can potentially be improved by focusing more on the timing of when a deployment opportunity is skipped.

It is also interesting to note that for the smaller starting budgets and $\delta = 3$ or $\delta = 4$, the variable cost consumes a larger proportion of the budget when there is no reliability requirement present. The reason for this is that the ADP policy is actually more likely to skip a deployment opportunity when there is no minimum reliability to maintain. With no penalty for network coverage falling below the requirement and no minimum reliability the network must maintain the ADP policy is freely able to sacrifice network performance. By avoiding the deployment costs for the current mission, there is a larger budget for the remaining missions which likely contributes to an increase in the number of sensors deployed. When there is a minimum reliability requirement the policy must be more strategic in when a deployment opportunity is skipped to ensure reliability of every mission is sufficiently high. As a result, the opportunity to skip a deployment window likely arises by deploying a larger number of sensors at the beginning of a previous mission, and/or a favorable network observation in which only a small number of sensors failed during the prior mission. Compared to an instance with no reliability requirement, where an increase in the overall number of successful missions can be achieved by low network performance over one or more missions.

4.4 Single Region Comparison

Finally, we explore the influence specifying the subregion a sensor is deployed in has on the overall number of successful missions. A simpler strategy to implement might involve randomly deploying a sensor over the entire region of interest, and is one of the more common assumptions when deploying a WSN [52,59]. The previous model formulation can easily address a single region by setting $r = 1$. It is interesting to note that since we previously defined a network structure by assigning weights to every subregion which determined how new sensors were deployed, a decision in the multiple subregion model is no more complex than the single subregion case. The only difference is that now sensors are randomly deployed over the entire region, whereas we previously used a rule-set to determine how sensors were allocated to each subregion.

Table 4 contains the expected number of successful missions from the optimal ADP policy when sensors are randomly deployed over the entire region. The final two columns of Table 4, under the 'Subregion' label, contain the results from the corresponding test instance with multiple subregions originally reported in Table 1. As expected, removing the ability to specify the subregion a sensor is deployed in lowers the expected number of successful missions compared to the original performance with multiple subregions. Even if the state variable definition remains the same (i.e., we are still able to observe the number and ages of sensors in various subregions in the network), there is now no guarantee that deploying new sensors based on observing a small number of sensors in one or more subregions at the beginning of a mission will improve the performance in the degraded areas of the WSN.

Table 4: Single Region Policy Performance

				Single Region		Subregion	
δ	M	B_0	ϕ	V_0	MC-PE	V_0	MC-PE
4	25	8700	0.95	24.91	24.89	24.97	24.97
4	25	7600	0.89	22.59	22.40	23.66	23.66
4	25	7400	0.79	20.79	21.12	22.97	22.65
3	33	8050	0.85	30.55	30.52	31.88	31.69
3	33	7650	0.65	24.53	25.35	26.27	27.42
2	50	8700	0.95	49.88	49.84	49.96	49.94
2	50	7600	0.89	45.73	44.03	48.05	46.73
2	50	7400	0.79	42.67	43.39	46.33	44.89

The decrease in expected number of successful missions resulting from randomly deploying sensors over the entire region compared to a smaller defined subregion is more noticeable for the smaller starting budgets. This can partially be attributed to the impact influencing network topology has on the probability of mission success in a smaller sized network compared to the impact in a larger network. In terms of the budget available, a decrease to the budget results in a decrease in the total number of sensors that are deployed over the planning horizon, and as a result the overall size of the WSN is generally smaller as well. For smaller sized networks it is less likely that randomly deploying sensors over the entire region of interest will result in sensors sufficiently distributed throughout the region for coverage purposes, and within the communication radius of nearby sensors necessary to route information to the sink node. While randomly deploying a sensor within a smaller subregion does not entirely remove this problem, it does provide the ability to avoid the situation in which one portion of the WSN is overly dense with sensor nodes whereas another portion of the network is uncovered and individual sensors are isolated. Therefore, there is a larger benefit (e.g., improvement in probability of mission success) in a smaller network when the subregion a sensor is deployed in can be specified compared to the benefit present in a larger sized network. This is observed several of the test instances, for example with $\delta = 4$ and $B_0 = 7400$ where the single region ADP policy achieves an expected number of successful missions of 20.79, while the previous results with 16 subregions achieve an optimal ADP policy with an expected 22.97 successful missions. Additionally, even if there is only a minor improvement for a single mission the cumulative impact over the entire planning horizon can be more substantial.

Exploring the performance in a single region model helps further illustrate the significance of the ADP policy and considering the impact of an action on future missions as well. Notice that the observed performance of the single region ADP policy, reported in the 'MC-PE' column of Table 4, is still able to outperform the myopic condition-based policy. This highlights the advantage of deciding if and how many sensors are deployed each mission, allowing an appropriate allocation of the budget to each mission as necessary. Even if new sensors are randomly deployed over the entire region of interest, rather than more controlled through a subregion deployment policy, the decision on when and how many sensors are deployed has a significant impact on WSN performance over an extended period of time.

A single region scenario also enables a more straightforward comparison with the TBDPs considered in [47], where sensors are deployed in order to restore the network to a fixed network size at periodic time intervals. Instead of a direct comparison with a TBDP, we can first note that there exists a close relationship between a TBDP and a corresponding myopic CBDP. In [47] an expression for the cost rate of an associated TBDP is derived based on the expected number of sensors that fail during a mission. The expected number of failed sensors informs the average cost of deploying sensors to reach a fixed network size, which can now be treated as a fixed budget available in a myopic CBDP. A TBDP differ from the myopic CBDPs in Section 4.2 in that sensors are randomly deployed over the entire region rather than a specified subregion. Since the myopic CBDP provides more control over how sensors are deployed, the performance of a myopic CBDP is at least as good as the related TBDP. With this similarity, and the previous discussion on the improvement of a single region ADP policy over a myopic CBDP, the ADP policy also improves upon a simpler time-based policy.

5 Conclusion

The coverage and communication capability of a WSN is made possible through the cooperative effort of a large number of sensor nodes. The flexibility with which WSNs can be established, randomly deploying sensors over a target region when exact placement is not feasible, enables their incorporation into a wide range of applications. It is important to consider not only the initial capability provided by a WSN, but performance over a period of time and the impact of eventual sensor failures. As the number of failed sensors increases the decline in network capability becomes more significant and appropriate actions must be taken to restore WSN coverage and communication abilities. A large focus on research related to this problem has been on deploying a small number of new sensor nodes in the network at a single point in time. The selective maintenance problem for a WSN over a prolonged period time in which sensors are repeatedly deployed in the network has received less attention.

In this work we have contributed an MDP model for the condition-based sensor deployment problem in which new sensors are deployed in the network over an extended period of time. While MDP models have been applied to a wide range of WSN related problems, our model is one of the few addressing maintenance through the repeated deployment of new sensor nodes, and one of the first ADP applications for the maintenance of a complex WSN. Whereas previous sensor deployment models have primarily been interested in extending a network lifetime metric, our work also addresses the complexity encountered by incorporating a reliability objective. A few of the difficulties that must be addressed in this problem include a variation in the age composition of sensors as well as a dynamic network topology as sensors fail and new sensors are deployed in the network. Our methodology has addressed both of these issues by the incorporation of the network D-spectrum. The D-spectrum has been widely research in network reliability problems, but only a handful of works discuss the D-spectrum in a maintenance optimization model as well [47, 49, 50]. Finally, we discussed an ADP solution approach using a value function approximations to determine optimal CBDPs, and presented results on a range of test instances.

The model also provides several directions for future work, focusing both on the modeling assumptions and ADP methodology discussed in Section 3. The reliability of a WSN is currently defined based on a given coverage requirement. The objective is to maximize reliability, but there is otherwise no detriment to not satisfying the coverage requirement over a mission. One possibility is to include a penalty based on the probability of network failure, which could also reflect need for immediate maintenance to provide a functioning WSN at all times. With respect to sensor failures, the model classifies sensors into an operating or failed state. Similar to the development of selective maintenance models for series-parallel systems, future work might allow multiple sensor states in which a sensor is partially degraded but still able to contribute towards WSN functions.

The current model also assumes the WSN is observed every δ time units and does not explicitly incorporate any cost associated with observation. A more complex decision might include whether the WSN is inspected/observed or not, where there is a cost associated with observing the network. Similarly network observation may be imperfect or there might be a time delay between our observation and deployment action. These directions begin to incorporate uncertainty in the true state of the network at the time sensors are deployed and might be better modeled as a partially observable MDP.

Our value function approximation was based on a combination of aggregation functions and lookup tables. Future work might consider the use of several basis functions and building a parametric model to approximate the value function. In this approach the previously defined aggregation functions may still be of use, but exploration is needed to define additional basis functions and an appropriate model representation (e.g., linear, nonlinear, etc.). A parametric model approximation of the value function is also of interest because it may provide additional opportunities to solve the optimality equation each stage, allowing the optimal action to determined more efficiently. Another direction for future work is to implement alternative solution methodologies, such as a Deep Q-Learning algorithm, to help address the large state and action space, which would help provide another point of comparison along with the myopic and time-based deployment policies.

Disclaimer

The views expressed in this article are those of the authors and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the U.S. Government.

References

[1] I. Dietrich and F. Dressler, "On the lifetime of wireless sensor networks," *ACM Transactions on Sensor Networks*, vol. 5, no. 1, pp. 5:1–5:39, Feb. 2009.

[2] T. Arampatzis, J. Lygeros, and S. Manesis, "A survey of applications of wireless sensors and wireless sensor networks," in *Proceedings of the 2005 IEEE International Symposium on, Mediterrean Conference on Control and Automation Intelligent Control, 2005.* IEEE, 2005, pp. 719–724.

[3] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no. 4, pp. 393 – 422, 2002.

[4] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer Networks*, vol. 52, no. 12, pp. 2292–2330, 2008.

[5] A. Tiwari, P. Ballal, and F. L. Lewis, "Energy-efficient wireless sensor network design and implementation for condition-based maintenance," *ACM Transactions on Sensor Networks (TOSN)*, vol. 3, no. 1, pp. 1–es, 2007.

[6] M. T. Thai, F. Wang, D. H. Du, and X. Jia, "Coverage problems in wireless sensor networks: designs and analysis," *International Journal of Sensor Networks*, vol. 3, no. 3, p. 191, 2008.

[7] U. Saeed, S. U. Jan, Y.-D. Lee, and I. Koo, "Fault diagnosis based on extremely randomized trees in wireless sensor networks," *Reliability Engineering & System Safety*, vol. 205, p. 107284, 2021.

[8] X. Fu and Y. Yang, "Modeling and analysis of cascading node-link failures in multi-sink wireless sensor networks," *Reliability Engineering & System Safety*, vol. 197, p. 106815, 2020.

[9] H. Zhang and J. C. Hou, "Maintaining sensing coverage and connectivity in large sensor networks," *Ad Hoc & Sensor Wireless Networks*, vol. 1, pp. 89–124, 2005.

[10] X. Wang, G. Xing, Y. Zhang, C. Lu, R. Pless, and C. Gill, "Integrated coverage and connectivity configuration in wireless sensor networks," in *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems*. New York, NY: ACM, 2003, pp. 28–39.

[11] L. Frye, L. Cheng, S. Du, and M. W. Bigrigg, "Topology maintenance of wireless sensor networks in node failure-prone environments," in *2006 IEEE International Conference on Networking, Sensing and Control*. IEEE, 2006, pp. 886–891.

[12] N. Li and J. C. Hou, "FLSS: a fault-tolerant topology control algorithm for wireless networks," in *Proceedings of the 10th Annual International Conference on Mobile Computing and Networking*. ACM, 2004, pp. 275–286.

[13] A. Sen, B. H. Shen, L. Zhou, and B. Hao, "Fault-tolerance in sensor networks: A new evaluation metric," in *INFOCOM 2006: 25th IEEE International Conference on Computer Communications*, 2006, p. 4146923.

[14] S. Parikh, V. M. Vokkarane, L. Xing, and D. Kasilingam, "Node-replacement policies to maintain threshold-coverage in wireless sensor networks," in *2007 16th International Conference on Computer Communications and Networks*. IEEE, 2007, pp. 760–765.

[15] S. Misra, S. R. Mohan, and R. Choudhuri, "A probabilistic approach to minimize the conjunctive costs of node replacement and performance loss in the management of wireless sensor networks," *IEEE Transactions on Network and Service Management*, vol. 7, no. 2, pp. 107–117, 2010.

[16] A. Jain and B. Reddy, "Node centrality in wireless sensor networks: Importance, applications and advances," in *2013 3rd IEEE International Advance Computing Conference*. IEEE, 2013, pp. 127–131.

[17] X. Cheng, D.-Z. Du, L. Wang, and B. Xu, "Relay sensor placement in wireless sensor networks," *Wireless Networks*, vol. 14, no. 3, pp. 347–355, 2008.

[18] E. L. Lloyd and G. Xue, "Relay node placement in wireless sensor networks," *IEEE Transactions on Computers*, vol. 56, no. 1, pp. 134–138, 2006.

[19] J. L. Bredin, E. D. Demaine, M. Hajiaghayi, and D. Rus, "Deploying sensor networks with guaranteed capacity and fault tolerance," in *Proceedings of the 6th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, 2005, pp. 309–319.

[20] H. M. Almasaeid and A. E. Kamal, "On the minimum k-connectivity repair in wireless sensor networks," in *2009 IEEE International Conference on Communications*. IEEE, 2009, pp. 1–5.

[21] G. Egeland and P. E. Engelstad, "The availability and reliability of wireless multi-hop networks with stochastic link failures," *IEEE Journal on Selected Areas in Communications*, vol. 27, no. 7, pp. 1132–1146, 2009.

[22] S. Chakraborty, N. K. Goyal, S. Mahapatra, and S. Soh, "A Monte-Carlo Markov chain approach for coverage-area reliability of mobile wireless sensor networks with multistate nodes," *Reliability Engineering & System Safety*, vol. 193, p. 106662, 2020.

[23] S. Xiang and J. Yang, "K-terminal reliability of ad hoc networks considering the impacts of node failures and interference," *IEEE Transactions on Reliability*, vol. 69, no. 2, pp. 725–739, 2019.

[24] S. Distefano, "Evaluating reliability of WSN with sleep/wake-up interfering nodes," *International Journal of Systems Science*, vol. 44, no. 10, pp. 1793–1806, 2013.

[25] H. M. AboElFotoh and C. J. Colbourn, "Computing 2-terminal reliability for radio-broadcast networks," *IEEE Transactions on Reliability*, vol. 38, no. 5, pp. 538–555, 1989.

[26] K. W. Pullen, "A random network model of message transmission," *Networks*, vol. 16, no. 4, pp. 397–409, 1986.

[27] J.-H. Park, "Time-dependent reliability of wireless networks with dependent failures," *Reliability Engineering & System Safety*, vol. 165, pp. 47–61, 2017.

[28] D. Deif and Y. Gadallah, "A comprehensive wireless sensor network reliability metric for critical Internet of Things applications," *EURASIP Journal on Wireless Communications and Networking*, vol. 2017, no. 1, p. 145, 2017.

[29] C. Chowdhury, N. Aslam, G. Ahmed, S. Chattapadhyay, S. Neogy, and L. Zhang, "Novel algorithms for reliability evaluation of remotely deployed wireless sensor networks," *Wireless Personal Communications*, vol. 98, no. 1, pp. 1331–1360, 2018.

[30] H. M. AboElFotoh, E. S. ElMallah, and H. S. Hassanein, "On the reliability of wireless sensor networks," in *2006 IEEE International Conference on Communications*, vol. 8. IEEE, 2006, pp. 3455–3460.

[31] Q. Liu, "Coverage reliability evaluation of wireless sensor network considering common cause failures based on d–s evidence theory," *IEEE Transactions on Reliability*, vol. 70, no. 1, pp. 331–345, 2020.

[32] J. S. Provan and M. O. Ball, "The complexity of counting cuts and of computing the probability that a graph is connected," *SIAM Journal on Computing*, vol. 12, no. 4, pp. 777–788, 11 1983.

[33] I. Silva, L. A. Guedes, P. Portugal, and F. Vasques, “Reliability and availability evaluation of wireless sensor networks for industrial applications,” *Sensors*, vol. 12, no. 1, pp. 806–838, 2012.

[34] C. R. Cassady, E. A. Pohl, and W. P. Murdock, “Selective maintenance modeling for industrial systems,” *Journal of Quality in Maintenance Engineering*, vol. 7, no. 2, pp. 104–117, 2001.

[35] C. R. Cassady, W. P. Murdock Jr, and E. A. Pohl, “Selective maintenance for support equipment involving multiple maintenance actions,” *European Journal of Operational Research*, vol. 129, no. 2, pp. 252–258, 2001.

[36] L. M. Maillart, C. R. Cassady, C. Rainwater, and K. Schneider, “Selective maintenance decision-making over extended planning horizons,” *IEEE Transactions on Reliability*, vol. 58, no. 3, pp. 462–469, 2009.

[37] K. Ahadi and K. M. Sullivan, “Approximate dynamic programming for selective maintenance in series–parallel systems,” *IEEE Transactions on Reliability*, vol. 69, no. 3, pp. 1147–1164, 2019.

[38] J. Xu, Z. Liang, Y.-F. Li, and K. Wang, “Generalized condition-based maintenance optimization for multi-component systems considering stochastic dependency and imperfect maintenance,” *Reliability Engineering & System Safety*, vol. 211, p. 107592, 2021.

[39] Y. Zhou, T. R. Lin, Y. Sun, and L. Ma, “Maintenance optimisation of a parallel-series system with stochastic and economic dependence under limited maintenance capacity,” *Reliability Engineering & System Safety*, vol. 155, pp. 137–146, 2016.

[40] A. F. Shahraki, O. P. Yadav, and C. Vogiatzis, “Selective maintenance optimization for multi-state systems considering stochastically dependent components and stochastic imperfect maintenance actions,” *Reliability Engineering & System Safety*, vol. 196, p. 106738, 2020.

[41] T. Jiang and Y. Liu, “Selective maintenance strategy for systems executing multiple consecutive missions with uncertainty,” *Reliability Engineering & System Safety*, vol. 193, p. 106632, 2020.

[42] C. D. Dao and M. J. Zuo, “Selective maintenance of multi-state systems with structural dependence,” *Reliability Engineering & System Safety*, vol. 159, pp. 184–195, 2017.

[43] W. Cao, X. Jia, Q. Hu, J. Zhao, and Y. Wu, “A literature review on selective maintenance for multi-unit systems,” *Quality and Reliability Engineering International*, vol. 34, no. 5, pp. 824–845, 2018.

[44] F. J. Samaniego, “On closure of the IFR class under formation of coherent systems,” *IEEE Transactions on Reliability*, vol. R-34, no. 1, pp. 69–72, April 1985.

[45] J. Navarro, F. J. Samaniego, N. Balakrishnan, and D. Bhattacharya, “On the application and extension of system signatures in engineering reliability,” *Naval Research Logistics*, vol. 55, no. 4, pp. 313–327, 2008.

[46] Y. Shpungin, "Networks with unreliable nodes and edges: Monte Carlo lifetime estimation," *Applied Mathematics and Computer Science*, vol. 27, pp. 168–173, 2007.

[47] N. T. Boardman and K. M. Sullivan, "Time-based node deployment policies for reliable wireless sensor networks," *IEEE Transactions on Reliability*, pp. 1–14, 2021.

[48] I. B. Gertsbakh and Y. Shpungin, *Models of Network Reliability: Analysis, Combinatorics, and Monte Carlo.* Boca Raton, FL: CRC press, 2016.

[49] M. Finkelstein and I. Gertsbakh, "Time-free preventive maintenance of systems with structures described by signatures," *Applied Stochastic Models in Business and Industry*, vol. 31, no. 6, pp. 836–845, 2015.

[50] N. T. Boardman and K. M. Sullivan, "Condition-based node deployment policies for reliable wireless sensor networks," *Proceedings of the 2021 Institute of Industrial and Systems Engineers (IISE) Annual Conference*, pp. 193–198, 2021.

[51] M. A. Alsheikh, D. T. Hoang, D. Niyato, H.-P. Tan, and S. Lin, "Markov decision processes with applications in wireless sensor networks: A survey," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 3, pp. 1239–1267, 2015.

[52] M. R. Senouci, A. Mellouk, and A. Aissani, "Random deployment of wireless sensor networks: a survey and approach," *International Journal of Ad Hoc and Ubiquitous Computing*, vol. 15, no. 1-3, pp. 133–146, 2014.

[53] M. Younis and K. Akkaya, "Strategies and techniques for node placement in wireless sensor networks: A survey," *Ad Hoc Networks*, vol. 6, no. 4, pp. 621–655, 2008.

[54] W. B. Powell, *Approximate Dynamic Programming: Solving the curses of dimensionality.* John Wiley & Sons, 2007, vol. 703.

[55] M. Finkelstein and J. Vaupel, "On random age and remaining lifetime for populations of items," *Applied Stochastic Models in Business and Industry*, vol. 31, no. 5, pp. 681–689, 2015.

[56] J. Navarro, F. J. Samaniego, and N. Balakrishnan, "Signature-based representations for the reliability of systems with heterogeneous components," *Journal of Applied Probability*, vol. 48, no. 3, pp. 856–867, 2011.

[57] F. Bistouni and M. Jahanshahi, "Evaluating failure rate of fault-tolerant multistage interconnection networks using Weibull life distribution," *Reliability Engineering & System Safety*, vol. 144, pp. 128–146, 2015.

[58] C. Bettstetter, "On the minimum node degree and connectivity of a wireless multihop network," in *Proceedings of the 3rd ACM International Symposium on Mobile Ad Hoc Networking & Computing*, 2002, pp. 80–91.

[59] M. Ishizuka and M. Aida, "Performance study of node placement in sensor networks," in *24th International Conference on Distributed Computing Systems Workshops, 2004. Proceedings.* IEEE, 2004, pp. 598–603.