# Robust Deployment and Control of Sensors in Wireless Monitoring Networks

A Thesis

Submitted to the Faculty

of

Drexel University

by

David J. Dorsey

in partial fulfillment of the

requirements for the degree

of

Doctor of Philosophy

July 2014

This thesis is dedicated to my sun, moon, and stars:

☼, ☾ , ★

Renée, Parker, and June

## Acknowledgements

I consider myself to be extraordinarily fortunate for my experience in Drexel's ECE department and my time working in the Data Fusion Laboratory under my advisor, Dr. Moshe Kam. The people who inhabit the laboratories, offices, and classrooms of the Drexel's Bossone building are creative, interesting, smart, hard-working, and fun. It has been a pleasure to work and study here.

First I want to thank my thesis committee for their support and feedback. Their patient involvement in this process made this thesis possible.

I also want to acknowledge the teachers who have influenced and inspired me in the classroom. After attending the first lecture in Moshe's undergraduate course on systems and signals, I knew that I wanted to work in his laboratory. Dr. Hande Benson's courses on linear and nonlinear programming were immensely useful and her real experience in this area helped to make the difficult subject approachable. Dr. Steven Weber's excellent graduate courses in networking gave many of us our first real experiences with analytical treatments of network design and performance. Dr. Lazar Trachtenberg is a serious scholar and a thoughtful teacher who patiently guided us through the art of proving logical arguments.

I have also had the privilege of working on projects with many of the bright professors in the ECE department, specifically Dr. John Walsh, Dr. Kapil Dandekar, and Dr. Nagarajan Kandasamy.

The helpful and patient ECE administrators deserve special acknowledgement: Chad Morris, Kathy Bryant, Tanita Chapelle, Amy Ruymann, and Phyllis D. Watson

have been indispensable over the years.

My friends from the Data Fusion Laboratory and the Drexel Wireless Systems Laboratory have made my time at Drexel special. I want to thank them all for great discussions, both academic and personal. Many of the former members played a significant role in guiding me through graduate school, and many are among my best friends. These include Dr. Pramod Abichandani, Dr. Rich Primerano, Gabe Ford, Boris Shishkin, Dr. Gus Anderson, Dr. Leonardo Urbano, Ryan Measel, Dr. Brian Hipszer, Ray Canzaneze, Brad Boyle, Alex Fridman, Gaurav Naik, Chris Lester, Nishant Dhawan, Dr. John Kountouriotis, Jeff Wildman, Wade Kirkpatrick, Dr. Ram Aiyar, Dr. Mianyu ”Jeremy” Wang, Dr. Wendi Chen, Zayd Hammoudeh, Chris Gaughan, George Sworos, Donald Bucci, Sayandeep Acharya, Dr. Li Bai, Dr. Saturnino Garcia, Dr. Lit Hsin Loo, Dahn Nguyen, Kevin Wanuga, and my wife, Renée Dorsey.

Finally I want to give special thanks my mentor and advisor, Dr. Moshe Kam. His guidance and example are the ultimate legacy of the time I have spent here. Learning from Moshe’s ability to intuitively see the important themes of a problem amidst a sea of challenges and details has helped me immensely in my career. When presented with situations where co-workers and project teammates have legitimate and conflicting ideas of how to best solve a problem, I often think ”how would Moshe handle this?” and at times I have called on him for suggestions. Moshe has an enviable ability to hear arguments and ideas from all sides, integrate them into a final proposal that is more than the sum of the individual parts, and eloquently communicate his reasoning to others. Having seen so many members of Data Fusion Laboratory from the past and present mirror this ability in their own lives and work, we wonder if this isn’t the true meaning of our lab’s appellation.

## Table of Contents

## List of Tables

## List of Figures

## Abstract

Advances in Micro Electro-Mechanical Systems (MEMS) technology, including MEMS sensors, have allowed the deployment of small, inexpensive, energy-efficient sensors with wireless networking capabilities. The continuing development of these technologies has given rise to increased interest in the concept of wireless sensor networks (WSNs). A WSN is composed of a large number (hundreds, even thousands) of sensor nodes, each consisting of sensing, data processing, and communication components. The sensors are deployed onto a region of interest and form a network to directly sense and report on physical phenomena. The goal of a monitoring wireless sensor network is to gather sensor data from a specified region and relay this information to a designated base station (BSt).

In this study, we focus on deploying and replenishing wireless sensor nodes onto an area such that a given mission lifetime is met subject to constraints on cost, connectivity, and coverage of the area of interest. The major contributions of this work are (1) a technique for differential deployment (meaning that nodes are deployed with different densities depending on their distance from the base station); the resulting clustered architecture extends lifetime beyond network lifetime experienced with a uniform deployment and other existing differential techniques; (2) a characterization of the energy consumption in a clustered network and the energy remaining after network failure, this characterization includes the overhead costs associated with creating hierarchies and retrieving data from all sensors ; (3) a characterization of the effects and costs associated with hop counts in the network; (4) a strategy for replenishing nodes consisting of determining the optimal order size and the allocation over the deployment region. The impact of replenishment is also integrated into the network control model using intervention analysis. The result is a set of algorithms that provide differential deployment densities for nodes (clusterhead and non-clusterhead) that maximize network lifetime and minimize wasted energy. If a single deployment is not feasible, the optimal replenishment strategy that minimizes deployment costs and penalties is calculated.

## Notation and Abbreviations

| Symbol | Description |
|---|---|
| $\mathcal{A}$ | area of deployment, a subset of $\mathbb{R}^2$ |
| ADC | analog-to-digital converter |
| BECR | biased energy consumption rate |
| BSt | base station or sink node |
| CSMA | Carrier Sense Multiple Access |
| CDMA | Code Division Multiple Access |
| $C_j$ | number of cluster-heads in band $j$ |
| $c_s$ | cost of an individual sensor node |
| $C_t(y_t)$ | cost associated with ordering $y_t$ nodes in period $t$ |
| CH | cluster-head node |
| $d$ | cost associated with deploying too few nodes in a band |
| DSSS | Direct Sequence Spread Spectrum |
| $\mathbb{E}$ | expected value operator |
| $\mathcal{E}_0$ | initial energy of a sensor node |
| $E_r^j$ | energy consumed by all nodes in band $j$ in a single round |
| $E_{rcv}$ | energy consumed in receive mode |
| $E_{tx}$ | energy consumed in transmit mode |
| $E_w^j$ | wasted energy (residual energy in the network after network failure) in band $j$ |
| $F_{jt}$ | marginal cumulative distribution function (CDF) of $w_{jt}$ |
| $F_{jt+L}$ | leadtime CDF |
| $\bar{G}_t$ | CDF of $\bar{W}_t$ |
| GK | reference to Gupta and Kumar [2000] |
| $G(r)$ | geometric random graph with parameter $r$, the radius of connectivity |
| $G_t$ | CDF of $W_t$ |
| $G(V, E)$ | a graph consisting of a set of vertices ($V$) and edges ($E$) |
| $h$ | cost associated with deploying too many nodes in a band |
| $hc$ | hop count |
| $J$ | number of bands |
| $j$ | band index |
| $K$ | Fixed cost for deploying a batch of nodes |
| $\mathcal{L}$ | network lifetime |
| $L$ | leadtime |
| $\bar{M}_t$ | $= \sum_{j=1}^{J} \mu_{jt}$; mean of $\bar{W}_t$ |
| MCU | microcontroller unit |
| $N_i$ | number of nodes in band $i$ |
| NCH | non-cluster-head node |
| $\mathcal{P}(\lambda)$ | Poisson distribution with parameter $\lambda_j$ |

Continued...

| Symbol | Description |
|---|---|
| $P_c$ | continuous power consumption |
| $p_c$ | probability of connectivity |
| $p_j$ | probability of being a cluster-head in band $j$ |
| $PPP$ | Poisson point process |
| QoS | quality of service |
| $R$ | radius of the $\mathcal{A}$ |
| $r_s$ | sensing radius of a node |
| $r_{tx}$, $r_c$ | maximum range of communication |
| RX | radio receiver |
| SINR | Signal to Interference plus Noise |
| $S_j$ | number of non-cluster-heads in band $j$ |
| $\bar{S}_t^2$ | $= \sum_{s=t}^{t+L-1} S_t^2$; variance of $\bar{W}_t$ |
| $S_t^2$ | variance of $W_t$ |
| $t$ | time index |
| $T$ | terminal stage/ mission lifetime |
| TDMA | time division multiple access |
| TX | radio transmitter |
| $\bar{W}_t$ | failures in all bands during the interval $t \dots t+L$ |
| $w_{jt}$ | number of failures in band $j$ in period $t$ |
| $W_t$ | failures in all bands during period $t$ |
| WSN | wireless sensor network |
| $\mathbf{x} = (x_1, \dots, x_N)$ | sensor locations |
| $\tilde{X}_t^{\Delta}$ | number of nodes plus outstanding orders minus target levels over all bands |
| $\tilde{x}_{jt}$ | number of nodes minus target level in band $j$ |
| $\tilde{X}_t$ | number of nodes minus target levels over all bands |
| $x^t$ | $= (x_{jt})_{j=1}^{J}$ |
| $x_{jt}$ | Number of active nodes in band $j$ at the beginning of period $t$ |
| $\hat{y}^t$ | $= (y_s)_{s=t-L}^{t-1}$; vector of orders placed in the last $L-1$ periods (not yet allocated) |
| $y_t$ | number of nodes ordered at the beginning of period $t$ |
| $z_{jt}$ | number of nodes delivered to band $j$ in period $t$ (allocation) |
| $\alpha$ | path loss exponent |
| $\alpha_{\text{cov}}$ | multiplier to the minimum density for connectivity that will provide minimum coverage |
| $\lambda_j$ | parameter of a Poisson point process (node density) in band $j$ |
| $\lambda_{\text{con}}$ | value of $\lambda$ that provides minimum density for connectivity |
| $\hat{\mu}_{jt}$ | $= \sum_{s=t}^{t+L} \mu_{jt}$; expected number of lead-time failures in band $j$ from period $t$ |
| $\mu_{jt}$ | $= \mathbb{E}(w_{jt})$; expected number of failures in band $j$ during period $t$ |
| $\Phi(\cdot)$ | $\phi(\cdot)$; standard normal cdf and pdf |

Continued...

| Symbol | Description |
|---|---|
| $\rho_{jt}$ | target level for band $j$ at $t$ |
| $\hat{\sigma}_{jt}^2$ | $= \sum_{s=t}^{t+L} \sigma_{jt}^2$; variance of the leadtime distribution $F_{jt+L}$ |
| $\sigma_{jt}^2$ | $= Var(w_{jt})$; variance of the distribution $F_{jt}$ |
| $\sigma$ | duty cycle: number of sensor reports per period |

# 1. Wireless Sensor Networks (WSNs) and Deployment Strategies

## 1.1 WSNs and their Applications

A wireless sensor network (WSN) [Akyildiz and Vuran 2010, Pottie 1998b, Stankovic 2008, Swami et al. 2007, Zhao and Guibas 2004] consists of spatially distributed autonomous sensors that monitor physical or environmental processes such as temperature, sound, vibration, pressure, motion or pollutants. The sensor nodes are equipped with radio transceivers, a processing unit, power supply, and one or more sensors (see Figure 1.1). They are deployed onto an area of interest ($\mathcal{A}$) in order to sample the physical environment. After deployment, the nodes communicate with other nearby nodes (within their transmission radius) to form a network. The goal of this network is to relay data from the sensors to a central processing station, called a *sink node*, or *base station* (BSt).

Unlike conventional ad-hoc wireless communication networks, which are mainly used to exchange data between nodes, WSNs provide a direct interface to the physical world. Some other important differences between WSNs and other wireless ad-hoc networks are related to the severe limitations on energy capacity, computational power, transmission power, and memory in sensor nodes. In addition, the number of nodes that are deployed in WSNs is typically much higher, as is the density of deployment. Sensor nodes usually do not have unique addresses as do nodes in ad-hoc networks, which typically employ TCP/IP [Akyildiz et al. 2002].

Decisions about how many sensors should be deployed, where the sensors should be deployed, and how the network should be organized are crucial to designing a WSN that can meet mission objectives. These objectives include the quality of sensing coverage (does the WSN sample the area with sufficient resolution?), the quality of

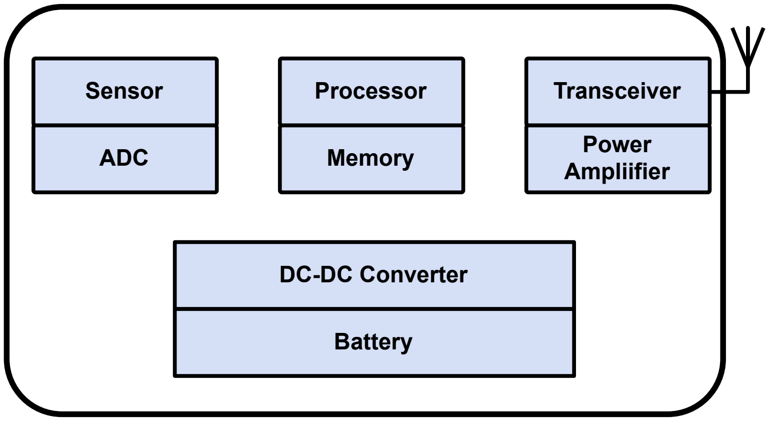


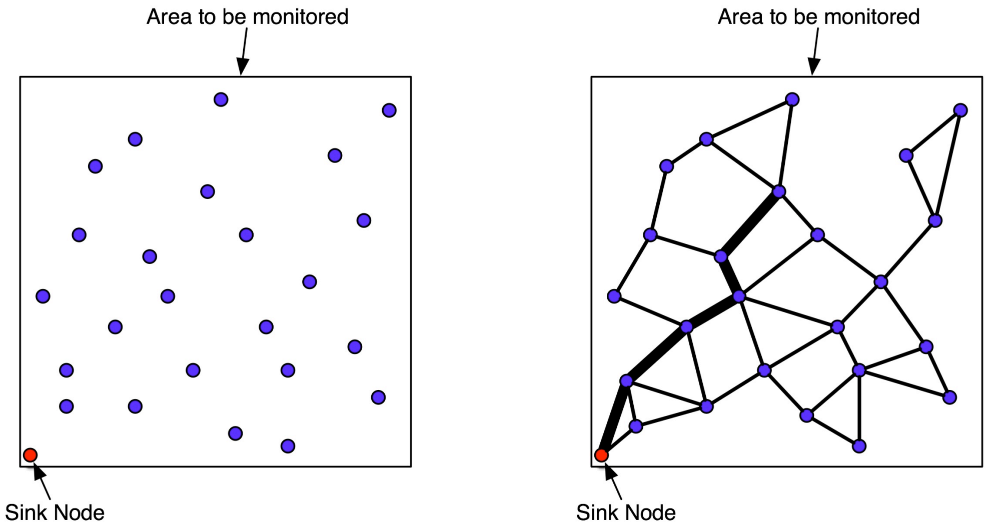
Figure 1.1: The components of a generic wireless sensor node.

estimation (how well does the WSN represent the physical phenomenon of interest over the area?), and the lifetime of the network (how long can the WSN provide a representation of the physical space without intervention?). The issue of sensing coverage is related to the density of sensors over the area, while the issue of estimation is related to both the density of the sensors and the capacity of the network. If the available bandwidth is insufficient, sensors may not be able to communicate their estimates of the area with sufficient fidelity. Since sensor nodes possess limited energy supplies, nodes will begin to fail due to energy depletion. The lifetime of the network is determined by the rate that nodes fail.

Unlike in peer-to-peer *ad hoc* networks, data gathered in a monitored area must often be delivered from many sources to a single destination (sink node). Nodes in WSNs must forward traffic for other nodes that are further away from the sink node. Consequently, nodes that are closest to the sink node typically expend their energy at a faster rate than more remote nodes. When enough nodes have failed, there will not be sufficient number of sensor nodes to provide sensing coverage and/or relay traffic (*i.e.*, the network will experience connectivity loss).

A key question is how to deploy a sensor network in order to meet lifetime, capacity, and performance requirements. This question is related to the scalability of WSNs. The scalability challenge is related to the traffic pattern exhibited by most WSNs (many-to-one communication). As the area of deployment becomes larger, and as more nodes are deployed to cover the area, the amount of traffic being relayed (and the energy expended) by nodes close to the sink node increases, while the available bandwidth decreases.

While traditional networks aim to provide high quality of service (QoS) levels, WSNs often aim to reduce energy consumption and prolong the lifetime of the network [Duarte-Melo and Liu 2006] while maintaining a minimum quality of monitoring (QoM). WSNs also have different constraints on deployment that depend on energy consumption and lifetime. Sensor nodes carry limited, generally irreplaceable, power sources that must power the node for months or years with no human intervention. For this reason, and because sensor nodes are prone to unpredictable failures, nodes should be 'over-deployed' beyond the nominal requirements of the application. Since sensor networks are to be deployed in large numbers and with high density, manual placement of nodes is often infeasible. Manual placement may also be impractical because the monitored area is dangerous (*e.g.*, chemical or biological agents may be present), or inaccessible (*e.g.*, environmental monitoring or scientific data collection takes place over inhospitable terrains). One approach to deployment is for nodes to be placed randomly over the monitored region through airdrop or from moving vehicles. The nodes then self-organize to form a network (Figure 1.2). The number of nodes to be deployed and the density of nodes in different regions must be chosen to meet requirements on connectivity, coverage, capacity, cost, and the lifetime of the network. We refer to the choice of node densities (and their organization) as the *deployment strategy*.



(a) Random Deployment

(b) Link Formation and Routing to Base Station

Figure 1.2: Sensor nodes are randomly distributed across an area to be monitored. After deployment, nodes communicate with their neighbors to form links. Sensed data are forwarded through one or more hops until they reach the base station.

### 1.1.1 Applications of WSNs

Applications of dense distributed sensor networks include monitoring climate (*e.g.*, Simic and Sastry [2003]), seismic activity (*e.g.*, Werner-Allen et al. [2005]), and acoustic signals. Some WSNs are used for medical and intelligence data gathering. Others for inventory monitoring. Examples of working systems include a habitat monitoring network [Osterweil et al. 2004] and a sensor network used to monitor volcanic activity [Werner-Allen et al. 2005]. WSNs are also a promising solution for military *command, control, computing, intelligence, surveillance, reconnaissance, and targeting* (C4ISRT) systems (*e.g.*, Akyildiz and Vuran [2010]). However, the vast potential of wireless sensor networks remains at present largely unrealized. The few sensor networks that have been deployed have consisted of few nodes (typically less than 30) that were placed manually inside the monitored region. Nevertheless, recent successes in the development of cheap, ultra-low power devices with processors, sensors and radios built into a single system-on-a-chip (SoC) provide hope that large scale realizations of WSNs consisting of tens or even hundreds of thousands of tiny nodes may soon be a reality. In the meantime, there remain many open problems and opportunities for improvements in WSN design and control. The research in this area has focused on a broad range of applications, including battlefield monitoring, environmental monitoring, and scientific data gathering [Simic and Sastry 2003]; inventory management [McKelvin et al. 2005], and several others.

## 1.2 WSN Deployment Optimization and Control

Developing deployment strategies that meet lifetime, capacity, and cost requirements is complicated by the unique traffic patterns for messages in WSNs. Unlike processing in peer-to-peer *ad hoc* networks, data gathered in a monitored area must often be delivered from many sources to a single destination (the base station, BSt). Because sensor nodes have limited communication range, these messages must be repeated over multiple hops until they reach the BSt (assuming that the monitored area is large compared to the communication range of a sensor node). Suppose the region of deployment, $\mathcal{A}$, is modeled as a disk of radius $R$ with the BSt located at the origin (Fig. 1.3(a)). Nodes are deployed uniformly on $\mathcal{A}$ to cover the region and maintain connectivity from all nodes to the BSt (Fig 1.3(b)). Now suppose that $\mathcal{A}$ is divided into annular bands of equal width 1.3(c), and define the *energy density* [Duarte-Melo et al. 2003] in each band as the amount of energy available in the band per unit area.

Figure 1.4 gives an illustration of the impact of the many-to-one communication over multiple hops on the energy consumption in different bands under these model assumptions [1]. The $x$-axis depicts time and the $y$-axis is the energy density (the

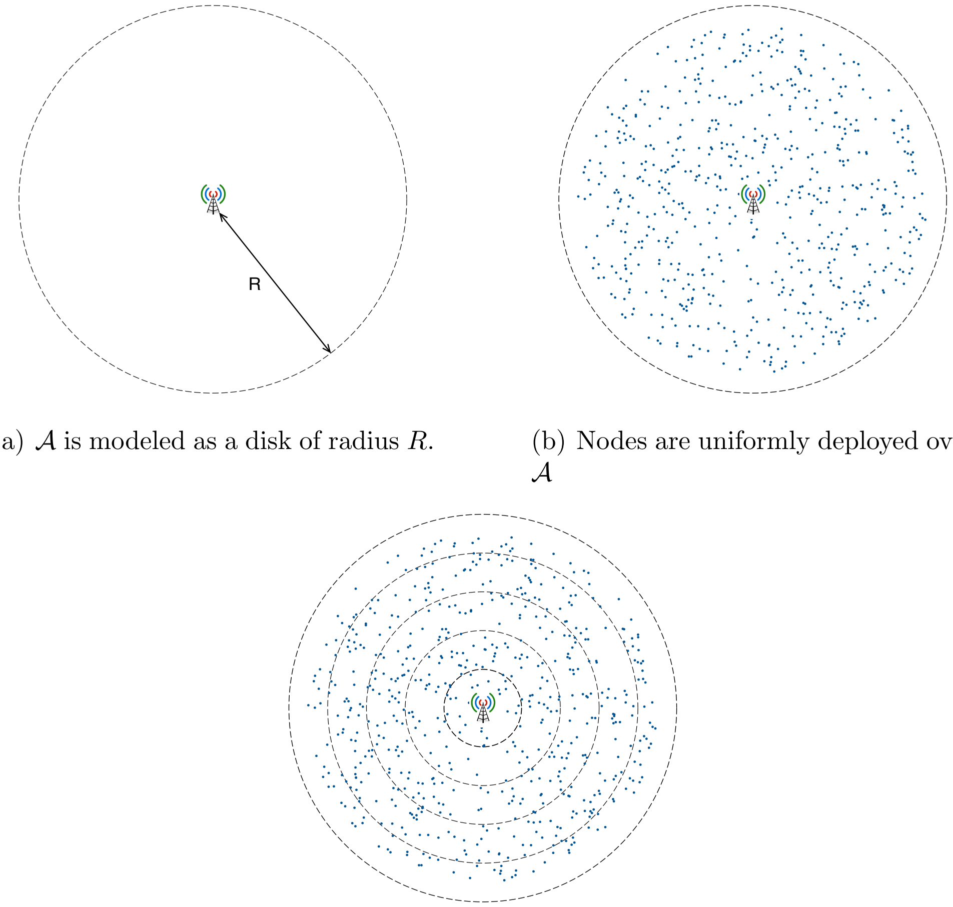
[1] Figure 1.4 is a simplified rendering drawn for illustration of the BECR effect. Figure 3.14 shows the same trend using data supplied from simulation.

amount of energy in the band per unit area). Each line in the figure represents the energy density for a single band over time in a uniformly deployed WSN. Initially, each band has equal energy density (because nodes are uniformly deployed). However, as time progresses, the nodes that are nearest the BSt will be burdened with a much greater amount of traffic, and will therefore “die” more quickly (this phenomenon is referred to in [Xu et al. 2005] as the *biased energy consumption rate*, BECR ). The network is no longer operable once the nodes closest to the BSt are unable to forward messages from the next band outward; the lifetime of the network will thus be quite often the lifetime of the first band of nodes near the BSt.

### Unbalanced energy consumption

The introduction of an organizational structure into the network helps to mitigate the unbalanced consumption of energy across the WSN by allowing nodes to share local measurements and reduce the number of messages that are forwarded through the network. Organization designs generally fall into one of two categories: *flat* or *hierarchical* [Sadler 2005]. Figure 1.5 shows a flat organization and a hierarchical organization. Figure 1.5(a) shows nodes that are all connected to their nearest neighbors. The bold line through the network depicts an example data path from a single sensor back to the BSt. Figure 1.5(b) represents the hierarchical organization. The nodes are divided into relay nodes (clusterheads) and sensor nodes. Instead of connecting to all of their neighbors, sensors connect to their nearest relay node in order to send data to the BSt. Thus, all data are gathered from sensors to by the assigned relay node, and the relay nodes form a separate network (shown as the“upper tier” in the figure) to forward sensor data to the BSt.

In a flat organization (Figure 1.5(a)), nodes are peers in the sense that all nodes act as both a relay and a sensing device at all times. Node management is performed



(a) $\mathcal{A}$ is modeled as a disk of radius $R$.

(b) Nodes are uniformly deployed over $\mathcal{A}$

(c) $\mathcal{A}$ is divided into annular bands of equal width.

Figure 1.3: Introduction of the model for $\mathcal{A}$ and the use of annular bands to compute energy density with respect to distance from the BSt.

through consensus protocols, where nodes cooperate to select a subset of nodes that will remain active while others “go to sleep” (*e.g.*, the Coverage-Centric Active Nodes Selection (CCANS) protocol, Zou and Chakrabarty [2005]). These protocols are designed to increase the likelihood that the active node subset is sufficient to meet connectivity and coverage requirements. In hierarchical (clustered) organizations, a subset of nodes (clusterheads) are designated to control the communications of the other nodes within this communication range (non-clusterheads). The clusterheads

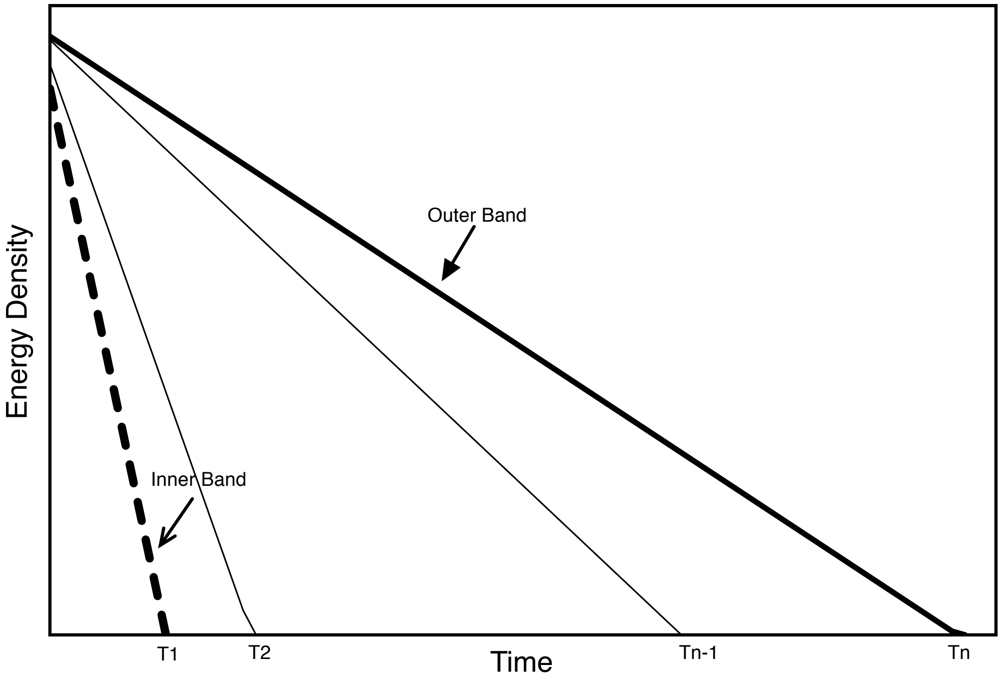


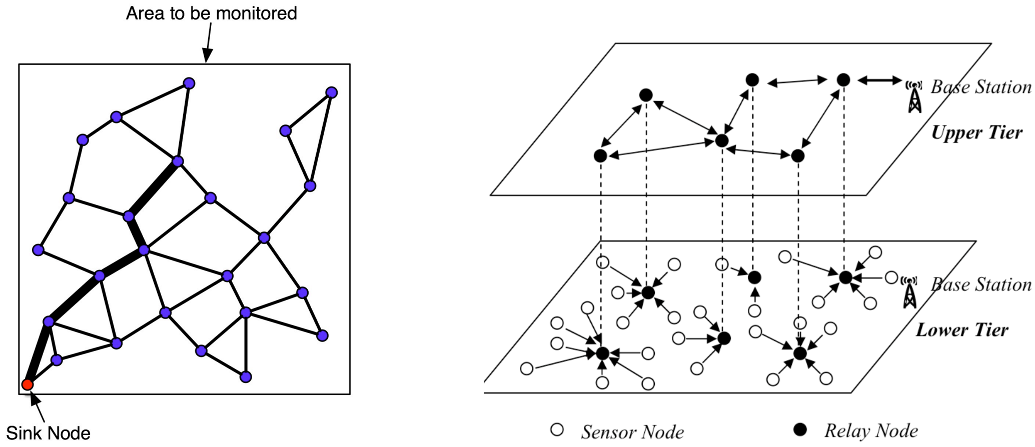
Figure 1.4: Energy is consumed at a greater rate when nodes are closer to the base station. The dashed line represents the energy density of the nodes nearest the base station; the bold solid line represents the energy density of the nodes in the outer band. In a uniform deployment, the lifetime of the network is the lifetime of the inner band.

are exclusively responsible for forwarding messages to the base station after collecting sensed data from the non-clusterheads [2]. The number and placement of clusterheads need to be sufficient to support a connected overlay network for delivering data to the base station. The advantage of clustered WSNs is that they provide a suitable structure for fusing spatially correlated data received from non-clusterheads and forwarding a compressed version of the data along the clusterhead overlay network.

However, this compression cannot be repeated on increasing hierarchical levels because the sensor data will de-correlate in clusters that cover large areas. Because

[2] The terms *clusterhead* and *relay* are often used interchangeably. However, a clusterhead is a relay that is also responsible for node management, while a relay is only responsible for forwarding messages from other nodes.

the clusterhead overlay is also performing many-to-one communication over multiple hops, the lifetime of the overlay is also the lifetime of the band that contains the clusterheads that are forwarding the most information. In Chapter 3, we show that when nodes are *uniformly* distributed over $\mathcal{A}$ and nodes alternate the responsibility of being a clusterhead (*e.g.*, Heinzelman et al. [2000]), the lifetime of the entire network is the lifetime of the nodes nearest the BSt.



(a) Depiction of a flat organization.

(b) Depiction of an hierarchical organization

Figure 1.5: Flat vs. hierarchical network organization.

### Balanced energy consumption is not possible

In the next two sections, we motivate the need for new approaches to WSN deployment that can maximize the lifetime of the network while reducing the residual energy remaining in the network at failure. We first emphasize that a deployment that would balance the energy consumption per unit area in all bands of the network simultaneously is not possible. We then select a recent proposed approach to suboptimal energy balancing in bands around the sink node, and show that the density requirements to implement the procedure are not practical, even for modest network sizes.

In [Wu et al. 2008a], the authors model $\mathcal{A}$ as a 2-dimensional disk with radius $R$ and a sink node at the center. $\mathcal{A}$ is divided into annular bands of width $r = 1$, which is the transmission radius of any node (see Figure 1.3(c)). The work was motivated by the problem of unbalanced energy consumption due to the effect of many-to-one multi-hop communication in WSNs discussed above. The goal was to minimize the imbalance in the total energy consumed by nodes across the annular bands. The authors proposed to deploy nodes in a non-uniform fashion over the network, providing more nodes for the bands near the sink node.

In a flat network without any data compression, when a node is relaying a packet, it has to expend two units of energy for every relayed packet, while to send its own data it expends one unit energy. Obviously, if a large fraction of a node's communication activities are dedicated to relaying, it will expend energy at a much higher rate than a node that has no relaying responsibilities. Letting $\mathcal{E}_o$ denote the initial energy stored in each node at deployment, $E_i$ denote the total energy consumed in band $i$ per round [3], and $N_i$ denote the number of nodes in band $i$, the network lifetime (assuming energy consumption across all bands is balanced) is given by

$$\frac{\mathcal{E}_o N_1}{E_1} = \frac{\mathcal{E}_o N_2}{E_2} = \ldots = \frac{\mathcal{E}_o N_{R-1}}{E_{R-1}} = \frac{\mathcal{E}_o N_R}{E_R}. \tag{1.1}$$

In words, the expected lifetime of a single band is the ratio of the total available energy in the band and the rate of energy consumption per round. If if this ratio is equal for all bands, then we say that the energy consumption is *balanced*, and the

[3] A round is defined as a single reporting cycle where all nodes have completed sending $K$ measurements to the BSt.

lifetime of the network (in rounds) is given by any of the ratios in Eq. 1.1.

It is easy to show that, since the outermost band does not forward any traffic, completely balanced energy depletion is impossible (this is counter to suggestions for fully balanced deployments, *e.g.*, Liu [2006]). This fact is expressed in Theorem 4.1 in [Wu et al. 2008a]; consequently, the authors define *sub-balanced energy depletion* as the balance attained when nodes in all bands *except the outermost band* exhaust their energy simultaneously.

In order to determine if it is possible to achieve sub-balanced energy depletion, the authors define the lifetime of a band as the ratio of the total energy in the band and the energy consumed in the band per unit time, and set each of these terms equal to one another (that is, Eq. 1.1 without the last term). This expression establishes a relationship between the lifetime of the network and the criteria for sub-balanced energy depletion. Starting with this expression, the authors prove that sub-balanced energy depletion is achievable only if the number of nodes grows in geometric progression with a common ratio $q > 1$ from the outer bands to the inner bands except the outermost one.

### Example of a network designed for sub-balanced energy depletion

[Wu et al. 2008a] provide an example of a network deployment where sub-balanced energy depletion is achieved ($q$-switch). However, their approach requires careful placement of nodes in the bands, and a special routing protocol. The approach involves deploying nodes from the outer band to the inner band in numbers that satisfy the constraint:

$$\frac{N_i}{N_{i+1}} = \begin{cases} q, & q > 1, 1 \leq i \leq R-2 \\ q-1, & i = R-1, \end{cases}$$

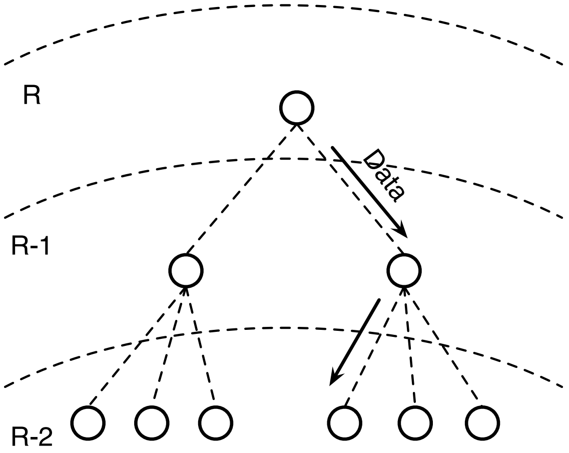


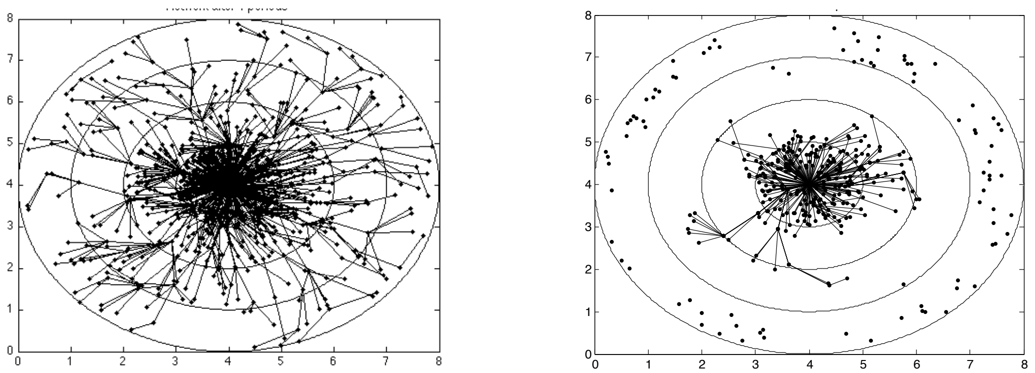
Figure 1.6: This figure shows the placement of nodes in a network that can achieve sub-balanced energy depletion. Each node from band 1 to band $R-2$ can communicate directly with $q$ different nodes in the adjacent band. This arrangement requires a disjoint set of $q$ nodes in the neighboring band for each node. This placement is used to support the $q$-switch routing protocol, which allows nodes to evenly distribute their messages to each of their $q$ downstream neighbors. Graphic is adapted from [Wu et al. 2008a].

The number of nodes deployed in the outermost band (*i.e.*, $i+1=R$) is determined by the coverage and connectivity constraints of the network. Then, these nodes are placed in such a way that each node in band $R$ can communicate directly with $(q-1)$ different nodes in band $R-1$, and nodes in band $i+1$ can communicate with $q$ different nodes in band $i$, for $i=1\ldots R-2$. In Figure 1.6 this placement is shown for $q=3$. The placement strategy supports the proposed routing protocol, $q$-switch routing, where nodes evenly distribute messages to be relayed to the sink node among their $q$ neighbors in the adjacent band.

### Simulation of the $q$-switch algorithm for a random deployment

The $q$-switch algorithm provides a method for deploying a WSN with sub-balanced energy consumption. However, the approach has a significant drawback for the large-scale WSNs considered here and in [Wu et al. 2008a]. $q$-switch requires specific placement of nodes, which would be impractical for a large network. To illustrate the density requirements for a *random deployment* over a modest area size, we can consider employing $q$-switch over an area of radius $R = 4$ with transmission range $r = 1$. In order to choose a density for the outermost band, we apply results from [Xue and Kumar 2004a], who proved that a network with $n$ randomly placed nodes is asymptotically connected with probability one as $n$ increases if each node is connected to more than $5.17 \ln(n)$ neighbors. They also show through simulations that connectivity can be achieved with high probability for only $1.5 \ln(n)$ neighbors. If we assume that the nodes are deployed according to a Poisson point process (PPP) (see Section 1.3.3 for an explanation of a PPP) with mean $\lambda$, the expected number of one-hop neighbors of a sensor with communication radius $r$ is $\lambda \pi r^2$. Since $\lambda$ is approximately the density of nodes in a fixed area, we can use the minimum criteria for connectivity to derive the minimum number of required nodes in band $R$ by solving $\frac{N_R \pi}{A_R} = 1.5 \ln(N_R)$ for $N_R$, where $A_R$ is the area of the outer band and $N_R$ is the number of nodes to deploy in the outer band. For a network with $R = 4$, the solution is $N_R = 120$. If we choose $q = 2$, the number of nodes for the $R - 1$ band is $N_R(q - 1) = 120$. The remaining values are $N_2 = 240$ and $N_1 = 480$. This deployment (using random placement) is shown in Figure 1.7(a).

Implementing the $q$-switch routing algorithm for a random deployed network is difficult since the nodes are not carefully placed within the bands. For example, it was not possible to find $N_2 = 240$ disjoint subsets of $q$ nodes from band 1 for the random deployment in this example. Even if it were possible, the complexity of the algorithm used to find such pairings would make the routing protocol impractical. In order to approximate the intent of the $q$-switch protocol, the simulation was designed so that nodes would select $q$ nodes from the adjacent band at random and uniformly distribute messages across this subset ($q$-switch requires a unique set of $q$ neighbors



(a) Initial deployment

(b) Network after 600 nodes have failed

Figure 1.7: Example simulation of the non-uniform random deployment strategy proposed in [Wu et al. 2008a].

for each node). After 98 iterations of the simulation, the number of node failures in each band was $(304, 160, 102, 32)$ for bands 1-4 respectively. Figure 1.7(b) shows the remaining nodes after 600 nodes have consumed all of their energy. There are no nodes remaining in the third band (bands are counted from 1 to $N$ from the BSt outward) and these failures prevent messages from being sent from bands 4 and 5 to the BSt. However, many nodes are still active in the first band. A key observation is that the distribution of residual energy among nodes that are still active in band 1; a majority of the nodes remaining in band 1 have nearly half of their energy reserves at network failure. From this simulation it appears that our random version of $q$-switch does not provide sub-balanced energy depletion. The reason for the imbalance in the energy consumption is that, as nodes exhausted their energy in outer bands, fewer messages were being sent through the inner bands.

Another drawback of the $q$-switch approach is that the number of nodes required grows very quickly with the number of bands. Figure 1.8 shows a semi-log plot of the required number of nodes needed, for the first band only, to meet the requirements of the approach. Even for modest sized networks and the smallest value of $q$, the

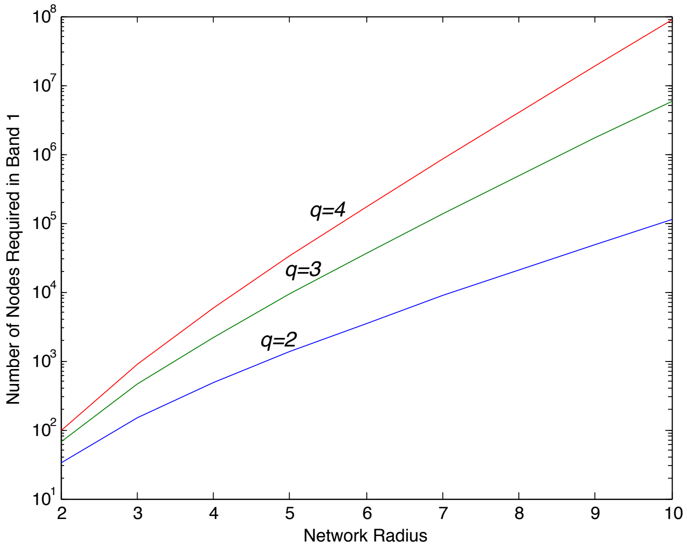


Figure 1.8: This figure shows a semi-log plot of the required number of nodes to be deployed in the first band only as the network radius increases from $2r$ to $10r$. The requirements are computed for common ratios $q = 2, 3, 4$.

number of nodes required is not practical. The large number of nodes required for balanced energy consumption emphasizes the need for techniques, such as clustering, that reduce the overall number of redundant messages in the network.

## 1.3 Fundamentals of Energy Constrained WSNs

The deployment strategy influences the limits of many properties of a WSN, such as energy, coverage, connectivity, capacity, cost, and lifetime. In this section, we will discuss each of these properties in an effort to derive a model for the constraints and objectives towards the deployment strategy proposed in Chapters 3 and 4, and the replenishment strategy discussed in Chapter 5. For more comprehensive analysis of these properties, we refer to the surveys in [Akyildiz et al. 2002, Ghosh and Das 2008, Sadler 2005].

### 1.3.1 Stages of a Wireless Sensor Network

The activity of a WSN can be divided into four primary stages: *deployment*, *clustering* and/or *route selection*, *data retrieval*, and *replenishment*. In this section we briefly describe each one of these stages in order to introduce some particular requirements, constraints, and associated challenges before discussing them in detail in later sections. We also highlight some of the interplay between the requirements for these stages.

#### Deployment

The deployment of a sensor network can be either random or deterministic; we focus exclusively on random deployments. Since the deployment random, control over where the sensors are placed in an area of interest is limited to a coarse resolution. It is for this reason that we focus on the density of sensors over large areas and not the exact number and configuration on the monitored regions, $\mathcal{A}$. As mentioned earlier, we assume that the nodes are deployed from air or from a moving vehicle. The density of sensors must be chosen to meet criteria for network lifetime, coverage of the monitored area, average connectivity of the network, and cost. The selection of sensor densities must also be made in conjunction with decisions about network hierarchy, Multiple Access Control (MAC) schemes, and routing. Deploying many nodes will require additional command overhead related to hierarchy, MAC, and routing to control congestion, while too few nodes will require more frequent replenishments in order to meet coverage and connectivity requirements.

### Command messaging (clustering and route selection)

Once nodes are deployed, they are required to self-organize to create a connected network. In a flat network (no hierarchical strategy), each sensor must find a path back to the sink node in order to deliver measurements. This task requires forming routes by passing control messages between local nodes to create a connected network. If a clustering strategy is used, then nodes must be provided an algorithm for selecting a clusterhead, forming clusters, and exchanging measurements between non-clusterheads and their associated clusterhead.

### Data retrieval

Once sensors have taken measurements, the data must be delivered back to the sink node. If a flat network is chosen, each node is responsible for delivering its readings to the sink node through the network. In order to avoid collisions and delays, a MAC scheme must be specified in addition to a routing algorithm. In a clustered network, separate MAC/routing schemes must be provided for intra-cluster communication (where non-clusterheads (NCHs) exchange data with the clusterheads (CH)) and inter-cluster communication (where CHs exchange messages in order to pass measurements to the sink node.

### Replenishment

The replenishment of a WSN involves introducing additional sensors to the monitored area in order to maintain connectivity and coverage until the end of the network's mission lifetime. The motivation for adding additional sensors when (or just before) they are needed is to reduce network cost and increase network lifetime. Simply increasing the number of initially deployed sensors while keeping all of them in use will eventually begin to decrease the network lifetime due to the additional energy consumed by control overhead messages and increased data retrieval messages. Network capacity will diminish due to increased contention to access the medium. Also, the cost of the deployment will increase due to the large number of sensors being deployed. Decisions about when, where, and how many additional sensor should be added to the network in order to reduce cost and extend lifetime are discussed in Chapter 5.

### 1.3.2 Energy consumption

Energy resources of a sensor node are consumed by four main components (Figure 1.9): the *processor*, *radio*, *sensing module*, and *timing element*. The subset of components that are active at any given time depends on the state of the node. These states include *transmit*, *receive*, *idle*, *sensing*, and *DSP active* (other states are possible, depending on the application). The energy consumed in a node will depend upon the proportion of time that nodes spend, and the amount of energy consumed, in each of these states. Some of these states will cost considerably more than others, so managing node state transitions is critical to reducing energy consumption and extending lifetime; this observation is especially true for the transceiver.

As an example of the power consumed in each state, the FireFly [Rowe et al. 2004] node designed at Carnegie Mellon University draws a total of 24.8mA of current (with a 3V supply) when the radio is in use and the CPU is active. The device is comprised of an Atmel ATmega1281 8-bit micro-controller with 8KB of RAM and 128KB of ROM and a Texas Instrument CC2420 IEEE 802.15.4 standard-compliant radio transceiver for communication. According to the data sheet [Chipcon 2004], the receive state requires 18.8mA, while the CPU requires 6mA. When both the radio and the CPU are idle, the node draws only 0.2$\mu$A.

The processor in a sensor node is primarily used for digital signal processing (DSP) and will therefore be active whenever the transceiver is active in order to provide support for (de)modulation and (de)coding and any other functions performed on the signal before the DAC in the transmit state and after the ADC in the receive state. Additionally, the processor will be active in the sensing state to perform DSP, compress data, and, occasionally, execute decision logic on the sensor input. The amount of power consumed by the processor for these tasks depends on the complexity of the functions, the frequency of the processor, and the amount of information that must be stored and retrieved to complete operations. The processor will also consume power in its idle state due to leakage current. The processor can be turned off when idle in order to reduce power consumption, though the benefit of turning the processor on and off diminishes when the unit is used often since there is a cost associated with powering up the device from the off state [Wang and Yang 2007].

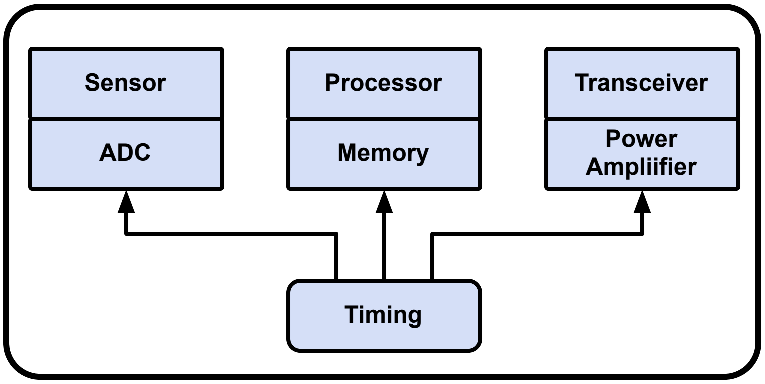


Figure 1.9: Energy is consumed by four main components in a typical sensor node: the sensing component, the timing component, the processor, and the radio transceiver.

According to data sheets for most standard WSN transceiver modules (*e.g.*, Chip-

Table 1.1: Power consumption in transmit mode for the Texas Instruments CC2420 SmartRF ®chip. The device consumes 18.8mA in receive mode.

| dBm level | current (mA) | power (mW) |
|---|---|---|
| -25 | 8.5 | 15.3 |
| -15 | 9.9 | 17.82 |
| -10 | 11 | 19.8 |
| -5 | 14 | 25.2 |
| 0 | 17.4 | 31.32 |

con [2004]), the power required to transmit a single bit is comparable to the power required to receive a single bit when the transmitter is being used at full power (typically 0dBm). Table 1.1 shows the power consumption in transmit mode for all 5 power levels provided on the Texas Instrument CC2420 chip with a regulated 1.8V supply. In receive mode, the module consumes 18.8mA (33.84mW).

In other cases, however, the transceiver may require as much as 2-3 times more power in the receive state than in the transmit state (*e.g.*, Shih et al. [2001]). This incremental requirement is due to additional functionality including carrier acquisition and synchronization and decoding. The complexity of these functions may be further increased when robust signaling, such as direct sequence spread spectrum, is employed. The major factor that determines the transmit energy consumption $E_{tx}$ is the path loss, which requires an adjustment in the transmit power level, while the power consumption in the receive mode $E_{rcv}$ is determined by the complexity of the modulation and coding scheme. In general, when considering the cost per bit, $E_{tx} \geq E_{rcv}$. However, the cost $E_{tx}$ is only borne during transmission, whereas $E_{rcv}$ is continually a factor whenever the node is 'listening.' Consequently, the fraction of time that the receiver is in the listening state is a major factor in the total energy cost at the transceiver. The percent of time a node is in an active state (*e.g.*, transmit or receive) is referred to as the *duty cycle*.

To approximate the energy consumption, we can use a first-order radio model including blocks for the transceiver and signal processing similar to [Heinzelman et al. 2000, Shih et al. 2001]. The transmit energy is described by

$$E_{tx} = e_{sp} + d^{\alpha} \times e_{amp}, \tag{1.2}$$

where $e_{sp}$ is the energy cost of the signal processing associated with generating the signal, $e_{amp}$ is the output energy determined at the power amplifier of the transmitter, and a simple geometric path loss model is assumed. The propagation loss is assumed [Sadler 2005] to be proportional to

$$\frac{1}{d^{\alpha}},\ 2 \leq \alpha \leq 4, \tag{1.3}$$

where $\alpha$ is the path loss exponent and $d$ is the distance (meters). At the receiver, let $E_{rcv}$ denote the receiver energy and let $E_{sp}$ = signal processing energy. The units are Joules/bit, and $E_{tx}$ and $E_{rcv}$ are then scaled by the packet length $M$ (bits). Finally, let $E_{idle} = \mu E_{rcv}$ be the energy consumed in the idle state, where $\mu \ll 1$.

Using this model, we can illustrate how the duty cycle affects the energy consumption. As an example, let $E_{rcv} = E_{tx} = 1$ and $\sigma =$ duty cycle. Figure 1.10 shows the radio energy consumption versus the duty cycle for values of $\mu$ =0.001, 0.01, and 0.1. The figure shows that for low duty cycles, the energy consumption is dominated by $E_{idle}$, while at high duty cycles, the energy consumption is dominated by the cost of receiving and transmitting. A high duty cycle can be a consequence of high message rates, long periods of time in the 'listening' state, or a combination of both. Therefore, in order to reduce the energy consumption with respect to the duty cycle, nodes should minimize the number of bits transmitted by compressing the data before forwarding to the base station and avoiding redundant messages. Also, nodes can reduce

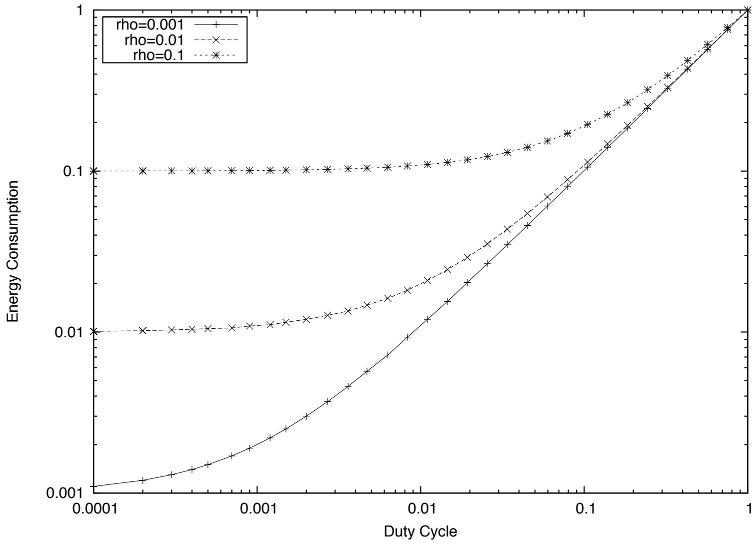


Figure 1.10: The duty cycle (percent of time nodes are active) determines the extent to which the energy in the node is predominantly consumed by the transceiver or the idle state components (*e.g.*, timing element).

the time spent in the 'listening' state through scheduling techniques and network organization such as clustering [Bandyopadhyay and Coyle 2003b]. If the message rates are relatively low, then nodes can be placed into an idle state until the next scheduled communication. This type of coordination requires nodes to be synchronized, and the precision of this synchronization depends on the accuracy of the timing element (as discussed in Section 1.3.10).

### 1.3.3 Connectivity

A WSN is connected when a path exists between all nodes. These paths need not be direct; it is usually sufficient that every node can reach the base station, which implies a path between all nodes. However, we note that interference, fading, dynamic clustering protocols, and random node failures cause transient topology changes, which may result in a short-term connectivity loss. For this reason, it is better to consider the average connectivity over time. Connectivity is a function of the node locations (density and coverage area), radio channels, transmission power, and traffic patterns [4]. In some applications, initial connectivity can be ensured by careful node placement along with channel measurement and power adjustment. However, in most situations, nodes will be deployed randomly on the monitored area, so a natural model for analyzing connectivity in WSNs is a random geometric graph [Penrose 1999].

A *graph* $G$ is a pair $(V, E)$ of vertices (nodes) $V$ and edges (links) $E \subseteq [V]^2$. A pair of vertices $v_i$ and $v_j \in V$ are directly connected if there is an edge $v_i v_j \in E$. A path is a subset of $G$ of the form $V = \{v_0, v_1, v_2, \ldots, v_k\}$ and $E = \{v_0v_1, v_1v_2, \ldots, v_{k-1}v_k\}$. If any two vertices in $G$ are linked by a path, then $G$ is connected. *Random graphs* are graphs whose properties, such as the number of vertices or edges, are determined randomly. In a *geometric random graph* , $G(r)$, a parameter $r$ is introduced and a set of vertices are distributed uniformly at random in a metric space, $\mathcal{R}$. Then, for any pair of vertices $v_i$ and $v_j \in V$, there is an edge $v_i v_j \in E$ if and only if the distance between $v_i$ and $v_j$ is less than $r$ (the communication range) in Euclidean space. The assumption that the vertices are distributed uniformly allows us to model the spatial distribution function (the probability that there are $n$ vertices within a unit space) using a homogeneous *spatial Poisson distribution* or *Poisson point process* (PPP) [Hall 1988, Kingman 1992, Stoyan et al. 1995]. Just as a stationary 1-D Poisson process has a constant rate $\lambda$ that determines the expected number of "events" or "arrivals" that occur per unit time, $\lambda$ is the expected number of vertices in a unit area on $\mathcal{R}$, or the density. Thus, if the density of the PPP is $\lambda$, the number of vertices located in a region of area $A$ is $\lambda A$ and the probability that there are $k$ nodes in this region is

[4] What we are calling "traffic patterns" is usually characterized by a traffic matrix that gives the volume of data between origins and destinations in a network.

distributed as

$$P(N(A) = k) = \frac{\mathrm{e}^{-\lambda \mathrm{A}} \left(\lambda \mathrm{A}\right)^{\mathrm{k}}}{k!}. \tag{1.4}$$

.

Now consider a randomly deployed sensor network with $n$ total nodes, where the nodes are placed under a homogeneous spatial Poisson distribution with parameter $\lambda$ into a 2-dimensional area of size $A$. Assume the geometric path loss model of Eq. 1.2. Furthermore, any two nodes within distance $r = r_c$ are able to form a link. Let $A_r = \lambda r^2$, which is the area covered by a transmission with radius $r_c$. Then, $N = \lambda A_r$ is the expected number of nodes within the transmission radius of the node (sometimes called the "average degree" of the nodes in the network). Given this model, we can consider the problem of finding a value of $\lambda$ that will ensure connectivity in $\mathcal{R}$. Extending the setup above with a slotted ALOHA model (a contention-based multiple access model where nodes access a shared channel by sending messages at the start of discrete time slots, [Abramson 1970]) and setting the value of $r_c$ equal for all nodes, [Kleinrock and Silvester 1978] performed average throughput analysis to show that $N = \lambda A_r \approx 6$ achieved the best trade-off between throughput and connectivity. This value of the optimal average degree came to be called the "magic number." However, as pointed by [Philips et al. 1989], as the value of $A$ increases under the Poisson model, there is a finite probability of a network partition using this (or any) static value of $N$. More recently, [Xue and Kumar 2004b] showed that the average degree of sensor nodes should scale with $\mathcal{O}(\log N)$, so that $P(\mathit{connected}) \rightarrow 1$ as $N \rightarrow \infty$.

In many studies that consider connectivity in geometric random graphs and sensor networks, (*e.g.*, Bettstetter [2004], Dousse et al. [2002], Gupta and Kumar [1998], Wan and Yi [2004], Xue and Kumar [2006]), the problem of ensuring connectivity in randomly deployed sensor networks is posed in the following way: given a density $\lambda$

and an area $A$, what value of $r$ will ensure connectivity? Other studies aim to find a density $\lambda$ for fixed $A$ and $r$ that will ensure connectivity. The results are equivalent for either form, and both suggest a trade-off between connectivity and capacity: increasing $r$ will increase the likelihood of connectivity, but larger transmission radii suggest more interference between nodes, impacting the network's throughput. On the other hand, increasing $\lambda$ for a fixed $A$ and $r$ will ensure connectivity while also creating more interference between nodes. [Xue and Kumar 2006] show that the critical transmission range for connectivity on a graph of unit area is given by

$$\pi r_c^2(n) = \frac{logn + c}{n}. \tag{1.5}$$

In an extensive study of properties of geometric random graphs, [Bettstetter 2004] shows that a random graph $G_n(r_c)$ with $n$ total nodes over area $A$ and a communication radius $r_c$ assigned to every node is connected with probability $p_c$ if:

$$r_c > \sqrt{\frac{A}{n\pi}(\ln n - \ln\ln\frac{1}{p_c})}. \tag{1.6}$$

This result takes into consideration the complicated boundary effects introduced by finite graphs that are ignored in a classical analysis of geometric random graphs, since the spatial Poisson distribution is assumed to be sampled from an infinite space. [Wan and Yi 2004] extend these results for the case of $K$-connectivity, where each node is to be connected to at least $K$ other nodes.

### 1.3.4 Effect of interference

A condition for successful transmission from node $i$ to node $j$ is that the *Signal to Interference plus Noise Ratio (SINR)* should be above a predetermined threshold $\beta$:

$$\frac{P_r(i,j)}{N_o + \gamma \sum_{k \neq i,j} P_r(k,j)} > \beta, \tag{1.7}$$

where $P_r(x,j)$ is the power of the signal received at $j$ from $x$, $N_o$ is the thermal background noise, and $\gamma$ is the orthogonality factor, which models the extent to which nodes in each others tx/rx range are coordinated. For example, if the nodes are all coordinated by a single clusterhead and perfectly adhere to a slotted schedule where no transmissions overlap, then this situation corresponds to $\gamma = 0$. In the case of $\gamma = 0$, the SINR is replaced with the SNR and the connectivity analysis is performed using a Boolean model (if nodes are in range, then are connected, otherwise they are not). When $\gamma > 0$, the connectivity is influenced by the summation in the denominator of 1.7. This model suggests that there is an upper limit on the node degree; the bound was shown to be $K \triangleq 1 + \frac{1}{\gamma\beta}$ in [Dousse et al. 2005]. In order to improve connectivity in this case, the value of $\gamma$ should be made small by coordinating transmissions so that they do not occur simultaneously (*e.g.*, using TDMA, CDMA, and CSMA schemes).

### 1.3.5 Capacity

The capacity of a network refers to the maximum theoretical rate that the network can deliver data from sources to destinations. The per-node capacity can be computed from the capacity, and gives the average maximum transmission rate for individual nodes. We may also speak about the *bandwidth capacity*, or available bandwidth in bit/s; it defines the maximum throughput across a communication channel. Therefore, the capacity refers specifically to maximum theoretical bounds on throughput for a channel, a network, or a single node. From the perspective of a single node, there are two common ways to measure data delivery. It may be measured by the number of bits per second per node that the network can deliver, referred to as the *per-node throughput.* Per node capacity may also be measured by the number of bits-meter per second per node, referred to as the *per-node transport capacity.* This metric was first defined by [Gupta and Kumar 2000] (GK) for *ad hoc* wireless networks. The transport capacity is the total transport length (in meters) of all of the bits in the network per unit time; this definition is different from the Shannon capacity since it involves physical distance. GK found that for $n$ nodes in a peer-to-peer *ad hoc* network, with a common shared channel of bandwidth $W$ Hz, the best total network transport capacity scales like $\mathcal{O}(W\sqrt{n})$[5] The transport capacity is expressed in terms of the per-node capacity by dividing by $n$, yielding $\mathcal{O}(W/\sqrt{n})$. The conclusion is that networks with a large number of nodes are probably not feasible, since as the number of nodes becomes large the available throughput to each node tends to zero.

The applicability of GK's work for characterizing the capacity of WSNs is somewhat limited because the results are based on the assumption that communications are made between randomly selected source and destination pairs transmitting uncorrelated information. In a study of the feasibility of large scale WSNs, [Servetto 2002] showed that although the per-node throughput of the network does tend to zero as the network size increases, so does the amount of information generated by each transmitter, due to correlation. In [Wang et al. 2005], the authors address the energy constraints and many-to-one communication in WSNs by analyzing capacity with respect to the cumulative amount of information that a relay can handle subject to lifetime and energy constraints. [Duarte-Melo and Liu 2003] and [Marco

[5] Order notation $\mathcal{O}(x)$ generally indicates that the largest term scales with $x$.

et al. 2003] extend the results from GK's work to account for many-to-one traffic as is encountered in WSN applications, comparing results for clustered and flat network organizations.

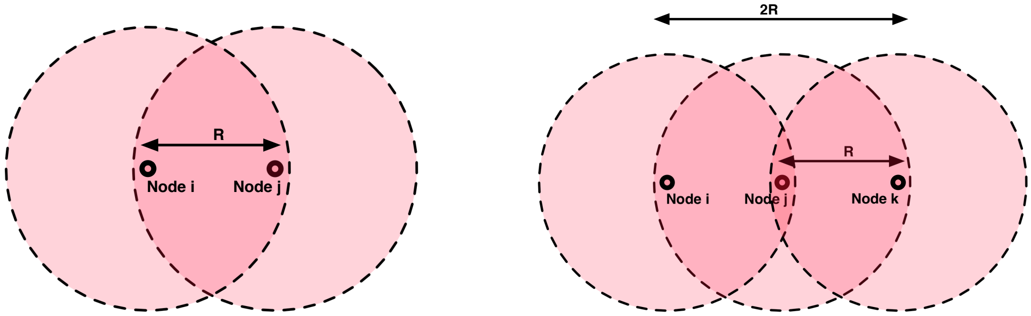
In a clustered network, one must make a distinction between the capacity of an individual cluster and the capacity of the clusterhead (CH) overlay network. The capacity of the CH overlay network will depend heavily on whether CHs compress the data received from the nodes in the cluster or act as simple relays of all packets sent from their non-clusterheads (NCHs). A common assumption is that the CH overlay network is on a separate channel from the channel that the clusters use for communication, so the two layers do not interfere.

The foundation for analyzing either inter- or intra- cluster capacity is the interference model. Let $s_i$ and $s_j$ be two nodes with distance $d_{i,j}$ between them. The transmission from $s_i$ to $s_j$ will be successful if

$$d_{i,j} \leq r_c \text{ and } d_{k,j} > r_c \tag{1.8}$$

for any node $s_k$ that is simultaneously transmitting. There are two ways that the nodes may interfere with each other. First, a node will obviously interfere with another node if it is within the communication radius $R$ (see Figure 1.11(a)). Second, a node may interfere with another node that is transmitting if it is within distance $2R$ because if node $s_i$ is within $2R$ of $s_k$ and the intended receiver $s_j$ is located within the overlapping area, the transmission will fail due to interference (Figure 1.11(b)). The interference model in Eq. 1.8 implies that no node can receive more than one transmission at a time and that no node can send and receive at the same time.

Now consider a single cluster operating on a separate channel with $S$ NCHs. For simplicity, we will assume that sources share the resource (time) by transmitting



(a) Nodes i and j are interfering with one another

(b) Node k is interfering with communication between nodes i and j

Figure 1.11: Two causes of node interference.

following a schedule consisting of time slots[6]. Then we have a sequence of $S$ time slots, one for each NCH to transmit to the CH. If $W$ is the capacity of the shared channel, the maximum throughput is achieved when the CH is busy 100% of the time, which implies a intra-cluster capacity of $W/S$. However, in most WSN applications nodes transmit a few small packets per unit time (usually on the order of hours). In addition, if the CH is energy constrained, then achieving maximum throughput (*i.e.*, being 100% busy) would be undesirable. The same result applies to CHs if every CH is directly connected to the base station, as is assumed in many studies (*e.g.*, Duarte-Melo and Liu [2003], Heinzelman et al. [2000]).

In the case of the inter-cluster capacity, when CHs are not directly connected to the base station, the capacity becomes more of an issue; capacity of $W/S$ is no longer achievable. As the network scales in size, the clusterheads must support more relay traffic from the outer regions of the network, in addition to collecting data from the sensors. The capacity of the overlay will be determined by whether or not CHs compress data before forwarding it; the average number of nodes in a cluster; the number of clusters; and the ratio of the deployment region's radius and the communication

[6] The same analysis could be used for different shared resources, such as frequency or codes.

radius. Although [Duarte-Melo and Liu 2003] do not consider multi-hop communication between CHs and the base station, we can apply their results for multi-hop traffic in a flat network, assuming that CHs compress the data from their NCHs and forward a single packet of constant size at each sampling of the region. The authors show that the capacity of the flat network with $n$ nodes, each with communication radius $r_c$ on an area of radius $R$, can achieve a maximum throughput of

$$C = \frac{R^2 W}{n\,(2\,R^2 - {r_c}^2)} \tag{1.9}$$

which is slightly worse than the results of GK. In this model, as the communication radius increases, the capacity improves, since nodes will send messages over fewer hops, thus reducing the number of transmissions. Also, larger transmission radius $r_c$ implies that more nodes are directly communicating with the base station. The reason why the results do not necessarily imply that large scale WSNs are infeasible is that, besides the argument made in [Servetto 2002] regarding correlated data, WSNs do not, in general, require high throughput in order to function.

If high throughput is required, then it has been shown by [Hu and Li 2004b] that the limited energy in the network would be the limiting factor on the network's capacity, not the interference. The authors compare *energy-constrained network capacity* and *interference-constrained network capacity.* Energy-constrained capacity is the maximum number of bits that can be injected into the network by each node without causing network failure as a result of energy depletion. The definition of energy-constrained capacity follows from the observation that the maximum amount of data that can be transmitted in any given time period are limited by the energy available during the same time period. This definition also holds for WSNs that use renewable energy sources, such as solar. [Hu and Li 2004b] show that for fixed densities the energy-constrained capacity scales much worse (with $n$) than interference-constrained capacity (assuming that the area of the region is scaling with $n$ to preserve the fixed density.) When the area is fixed and the density of nodes is increased, the authors show that the energy-constrained capacity is comparable to the interference constrained capacity.

Our consideration is the amount of time that nodes will spend waiting to access the channel after they make a measurement. We use this period of time to estimate how much energy is consumed by nodes in each sensing round as a function of the number of active nodes in a unit area.

### 1.3.6 Coverage

The concept of *sensor coverage* is central to the goal of WSN deployment. Since the sensors are being deployed for the purpose of measuring a physical space, it is natural to ask "how well will the sensors represent the desired features they are deployed to measure?" As pointed out in [Meguerdichian et al. 2001], coverage is a measure of the quality of service (QoS) of the sensing function and is subject to a range of interpretations across a variety of sensor types and applications. Nevertheless, a general definition of coverage in the literature is the guarantee that each location in the targeted physical space is within the *sensing range* of at least one sensor. Some of the work in coverage studies use the ratio of the covered area to the size of overall deployment region as a metric for the quality of coverage [Huang and Tseng 2005]. However, most recent literature has focused on the worst case coverage, or least exposure, which measures the probability that a target would travel across an area without being detected (*e.g.*, [Clouqueur et al. 2002]).

In developing a model for sensor coverage, most studies begin by using a stationary 2-D Poisson point process (PPP) to describe the locations of the sensors. Two widely adopted sensing models are the *Boolean sensing model* and the *general sensing model*

[Liu and Towsley 2004]. In the Boolean model (applied in, for example, Shakkottai et al. [2003]), each sensor can sense the environment within its sensing range, $r_s$. A location is said to "covered" by a sensor if it lies within the sensor's sensing area. The entire space is partitioned into two regions: the *covered region* (the region covered by at least one sensor) and the *vacant region.* An object is detected if it passes through the covered region. If two sensors are within $2r_s$ of each other, they are said to form a cluster. Since an object cannot traverse a cluster without being detected, the objective is to form enough clusters as to prevent a target from passing through the region undetected. This interpretation of coverage is problematic, however, because the definition of *sensing range* is not clear, since the quality of a sensor's measurement across a distance is related to the signal-to-noise ratio, not just the distance. Also, many sensors are *point* sensors (*e.g.*, chemical sensors); they do not detect objects at a distance but rather must come in physical contact with the object in order to sense it.

The general model is meant to incorporate the signal degradation of the sensor's sensing capability as the distance between the sensor and the target increases. For a sensor $s$, a *sensing signal* at a point $x$ on the region is given by [Liu and Towsley 2004]

$$S(s,x) = \begin{cases} \frac{\alpha}{d(s,x)^\beta} & \text{if } A \leq d(s,x) < B \\ 0 & \text{otherwise} \end{cases}, \tag{1.10}$$

where $d(s,x)$ is the distance between $s$ and $x$, $\alpha$ is the energy emitted by the events at $x$, and $A$ and $B$ define the range of a sensor's sensing capability. The sensing signal decays according to a power law with exponent $\beta$. The sensor field intensity $I_x$ at point $x$ is defined as the sum of the sensing signals of all sensors in the region, *i.e.*,

$$I_x = \sum_{i=1}^{\infty} S(s_i, x). \tag{1.11}$$

A point $x$ is deemed covered if $I_x$ is greater than or equal to some threshold. This definition of coverage fits naturally into a decision fusion framework, where sensors collaborate to determine whether or not an event occurred in a region of surveillance.

In this general sensing model, the sensing range is not an explicit boundary, but rather can be thought of as a threshold of the false alarm rate (probability that the sensors detect a target or event that is not present.) This definition of sensing range incorporates the statistical nature of WSNs with regard to the nature of the event being sensed and the capabilities of the sensor (a noise term could also be added to Eqns. 1.10 and 1.11). This definition is primarily useful for distributed detection with a network of sensors that take measurements from one of two hypotheses and compare a likelihood criterion to a threshold (*e.g.*, [Anandkumar et al. 2008], who seek to maximize the Neyman-Pearson detection error exponent subject to a constraint on average (per node) energy consumption.) But not all sensor networks are deployed to detect the presence or absence of an event. The goal of a data-gathering WSN or field-gathering WSN [Duarte-Melo and Liu 2006] is to periodically sense the environment and report an analog sensor value to the base station at regular intervals. This application is not compatible with a binary detection framework. As an example, consider a network deployed for the purposes of monitoring the air quality near a major metropolitan area (*e.g.*, Hamel et al. [2006]). Each sensor would report the density of particulate matter and the location of the sensor at predefined intervals. The density of deployment in this case would be the required resolution of spatial data required in order to apply distributed *parameter estimation* methods for localizing the diffusive source, determining its space-time concentration distribution, and predicting its cloud envelope evolution. Other examples of such *diffusive sources* are biological agents, toxic chemicals, explosives, hazardous materials, and temperature fields.

Although some research has been done recently on the problem of designing optimal estimators under bandwidth [Ribeiro and Giannakis 2006] and energy constraints [Li and Al Regib 2007] in wireless networks, and specifically for the case of monitoring diffusive sources [Zhao and Nehorai 2007], the subject of a minimum sensor density for these estimators, as far as we can tell, does not appear to have been covered. It appears that the notion of a sensing range is more meaningful in the context of finding a deployment to optimize *detection* of an event [Anandkumar et al. 2008, Rajagopalan et al. 2005], while for data gathering sensor networks, solving a parameter estimation problem, it is less useful.

A suitable criterion for coverage in monitoring networks is the *monitoring quality*. The monitoring quality of a sensor network is a measure of how well the sensor reports represent the underlying measured process in space and time. We can define these quantities in terms of the distortion between the temporal and spatial measurements and the real process by modeling the underlying reality as a spatio-temporal random process or random field in space and time. Temporal distortion arises due to a mismatch between the rate of changes in the random field and the sampling rate of the sensor network. If the sampling rate is less than the Nyquist frequency required to reconstruct the spatio-temporal random signal, the reconstruction will exhibit aliasing and other distortions.

Spatial distortion results from an undersampling of the area of interest, and this is the quantity of interest here. Thus we assume that the monitoring network is synchronized in the sense that all sensors take a measurement at about the same time, report to the base station, and "go back to sleep." We also assume that the rate of the sensing-reporting cycle satisfies the Nyquist rate and that the field does not change faster than the time it takes to collect samples from every sensor. The spatial distortion depends on the number and placement of actively participating sensors over the area. The average spatial distortion $D_{mse}$ over an area $A$ is defined as

$$D_{mse}(A) = \|A\|^{-1} \int_{x \in A} \mathbb{E}\Big[Z(x) - \hat{Z}(x)\Big]^2 dx, \tag{1.12}$$

where $Z$ and $\hat{Z}$ represent, respectively, the real and reconstructed values at points $x \in A$. The inner expectation is with respect to the spatial distribution of nodes in $A$.

Without making assumptions about prior knowledge (knowledge that could be used for an application specific data fusion technique), we could compute the value of $\hat{Z}$ by taking linear combinations of values reported by sensors in a neighborhood. This standard technique is called *kriging*[Williams 1998].

We do not consider specific underlying processes or monitoring quality requirements in this study, although a specific expression for $D_{mse}$ can be added as a constraint to the lifetime optimization problem. Instead we will assume, as is done in much of the literature, that the minimum density for coverage is linearly proportional to the minimum density for connectivity.

### 1.3.7 Lifetime

The lifetime of a WSN is the time span from the initial deployment to the instant that the network is considered non-functional. When a network is considered non-functional is, however, application specific. Various definitions of network lifetime are possible, such as time to first node failure (*e.g.*, Wang et al. [2006]), or time to appearance of the first network partition (*i.e.*, connectivity failure). Lifetime analysis is difficult since the network lifetime depends on many factors including network architecture and protocols, data collection initiation, lifetime definition, channel characteristics, and energy consumption model. The use of energy-aware routing protocols and efficient MAC protocols can increase the lifetime by reducing the amount of energy consumed by redundant messages, maximizing the amount of time nodes spend in the idle state, and reducing packet collisions that result in costly retransmissions. Data collection initiation refers to the manner in which nodes are triggered to sense and communicate. In event-based networks, nodes will communicate sensor data if a threshold on the sensor is exceeded, for example if a motion sensor detects an intruder. In time-triggered networks, the class to which data gathering networks belong, nodes sense and report according to a schedule. In some clustered networks, the schedule is coordinated by the clusterheads; the trigger may also be maintained by a timer on each node, assuming nodes are synchronized. The lifetime definition is determined by the nature of the application (critical or non-critical) and the number of redundant nodes that are deployed. Finally, an analysis of lifetime will depend on the models used for energy consumption in the nodes and the assumptions about the channel characteristics. For example, if the energy consumed by a component in the idle state is high relative to the total costs of communication, the energy model given in Eq. 1.2 will not provide an accurate lifetime result.

Upper bounds on lifetime have been derived for various WSNs. [Bhardwaj and Chandrakasan 2001] and [Hu and Li 2004b] derive upper bounds on network lifetime in flat architectures based on the assumption that all data are relayed to the base station via an optimal number of hops. Many studies use network flow techniques to bound the lifetime of flat networks with known topology (*e.g.*, Bhardwaj and Chandrakasan [2002], Duarte-Melo et al. [2003], Giridhar and Kumar [2005]). Several efforts have derived lifetime upper bounds in flat organizations However, studies on the lifetime analysis of of hierarchical organizations are relatively scarce. In one study by [Duarte-Melo and Liu 2002], the authors bound the lifetime of the WSN by optimally allocating energy to sensors in a clustered network.

[Chen and Zhao 2005; 2007] study the average lifetime of WSNs in a general setting; they do not specify a network architecture, the data collection initiation, or the channel and the energy consumption model. The energy consumed in the network is divided into two types: continuous energy consumption (energy consumed by the clock, current leakage, analog sensors, *etc.*), and reporting energy (sensing and communication, not including any sensors that are on continuously). Chen and Zhao provide a theorem, derived from the strong law of large numbers, stating that in a WSN with total non-rechargeable initial energy $\mathcal{E}_{total}$, the average network lifetime $\mathbb{E}[\mathcal{L}]$, measured as the average amount of time until the network dies, is given by

$$\mathbb{E}[\mathcal{L}] = \frac{\mathcal{E}_{total} - \mathbb{E}[E_w]}{P_c + \sigma \mathbb{E}[E_r]}, \tag{1.13}$$

where $P_c$ is the constant continuous power consumption over the whole network, $\mathbb{E}[E_w]$ is the expected wasted energy (*i.e.*, the total unused energy in the network when it dies), $\sigma$ is the average sensor reporting rate defined as the number of data collections per unit time, and $\mathbb{E}[E_r]$ is the expected reporting energy consumed by all sensors in a randomly chosen data collection. Not surprisingly, Eq. 1.13 implies that reducing the expected reporting energy and the expected wasted energy leads to prolonged network lifetime. This theorem assumes that lifetime is defined as the time span until any sensor in the network dies (the first death) or no sensor has enough energy for transmission during a data collection (the first failure in data collection). A more general framework for analyzing lifetimes for situations where lifetime is defined with respect to a percent of nodes failing (called $\alpha$-lifetime) can be found in [Zhang and Hou 2005a], where a loose upper bound on the maximum $\alpha$-lifetime is given.

The lifetime of a sensor network will also depend on other factors beyond (but not independent of) the deployment strategy, hardware, routing, network architecture, *etc.* For example, the desired quality of the output of the network (*e.g.*, how well it estimates a source) will affect the number of nodes required to sense the phenomenon as well as the rate required to convey the minimum amount of information to the base station. In these cases, rate-distortion based information theoretic arguments would provide a fundamental characterization of the quality-rate tradeoff [Pottie 1998a]. These additional cases can also be added into the general framework of the lifetime theorem above.

### 1.3.8 MAC protocols

Due to the unique operating environment that WSNs occupy, a large number of research studies have been published on the topic of media access control (MAC) protocols for sensor networks. As we have emphasized in the previous sections, the application often plays a significant role in the design of particular metrics and protocols; this is certainly the case with controlling multiple access communication in possibly dense (but likely low-bandwidth) applications related to WSNs. The specifics of the available hardware such as radios and clocks (discussed in the next section) all have different capabilities, costs, and energy consumption that affect the applicability of a particular MAC protocol.

WSN MAC protocols can be classified into two general classes: scheduled protocols and random (or unscheduled) protocols. The most common scheduled method employs time division multiple access (TDMA), where a single sensor uses a particular time slot. The most popular unscheduled protocols use channel sense multiple access (CSMA), where a sensor measures the assigned frequency to see if it is busy before transmitting. There are many protocols that specify some hybrid of these two schemes. A good survey of the most popular MAC protocols was published by [Kredo and Mohapatra 2007].

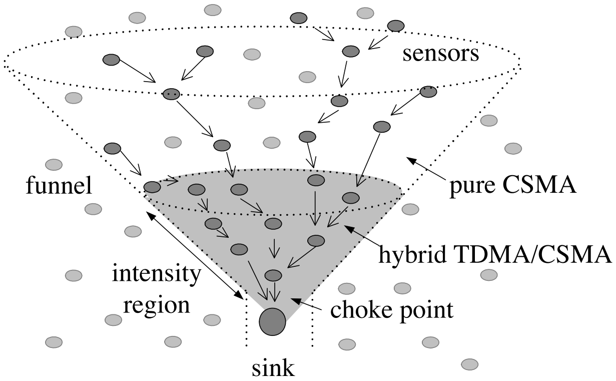


Figure 1.12: Illustration of the funneling-MAC protocol [Ahn et al. 2006]

### Funnel effect on the MAC

One of the most important issues that arises in the selection and design of a MAC protocol is the many-to-one data flow that challenges most WSN applications. Where the choice of MAC protocols is concerned, this unique traffic flow results in very different traffic characteristics over different spatial areas. In much of the literature (*e.g.*, Wang and Liu [2011]), the critical challenge stems from the fact that the closer a sensor node is to the base station, the more packets it needs to relay. The effect is sometimes referred to as the *funneling effect*. The funneling effect means that the region close to the base station is heavily burdened and will experience significant collisions if the MAC layer uses a CSMA-based protocol, but the regions that are farthest from the base station will not have nearly as much traffic, so the use of CSMA is more practical there. The use of TDMA, however, requires additional overhead and organization in order to assign nodes to time slots (and the time slots may need to be synchronized often). As an example of a proposed solution to the funneling effect, Figure 1.12 illustrates the funneling-MAC protocol [Ahn et al. 2006], where nodes closer to the sink node communicate using a hybrid of TDMA and CSMA, while nodes that are farther away from the sink use pure CSMA. The funneling effect also has an impact on the choice of clustered vs. flat network hierarchies, as we discuss in Section 3.6.

### 1.3.9 Routing

When categorizing routing protocols for WSNs, the literature tends to consider network hierarchical schemes as well as route selection techniques as part of the routing protocol. To an extent, the hierarchical arrangement is also prominent in the WSN MAC literature. Intuitively, the organization of the network plays a role in both resource sharing and in routing. This cross-layer emphasis is more pronounced in sensor networks because the data are spatially related; local measurements have more correlation, so there is more opportunity for increased efficiency through compression. The common categories for routing protocols are *flat*, *hierarchical*, *geometric* or *location-based*, and *data-centric*. Geographic routing requires that each node can determine its own location and that the source is aware of the location of the destination. In data-centric routing, the sink sends queries to certain regions and waits for data from the sensors located in the selected regions. Since data are being requested through queries, attribute-based queries are necessary to specify the properties of data. Directed Diffusion [Heinzelman et al. 1999] is a popular example of a data-centric protocol. Hierarchical routing schemes involve the use of clustering (low-energy adaptive clustering hierarchy (LEACH) [Heinzelman et al. 2000] is probably the most well-known of this category). In flat protocols, each node is employed in the same way to cooperate with neighbors and exchange data to the sink node.

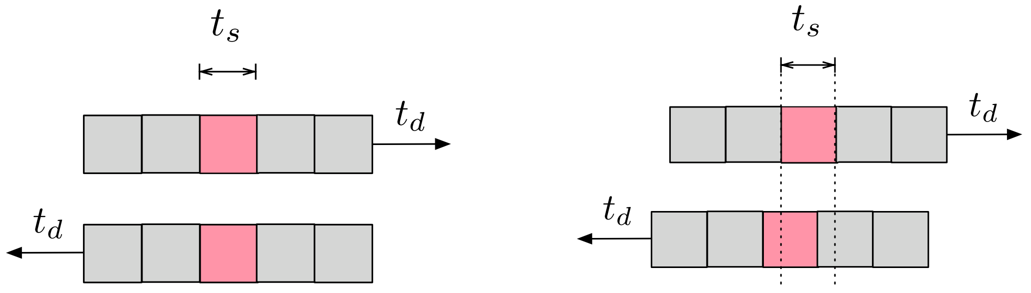
### 1.3.10 Synchronization and timing hardware

A final design consideration that we will discuss is the choice of the timing component [Barooah and Swami 2008]. A trade-off exists between the cost of the timing device, the amount of time nodes spend in the active state awaiting messages, and the energy consumed by nodes transmitting synchronization messages. To maintain synchrony across the network, each node may use its own clock, and then rely on communications between nodes to account for the clock drift between nodes. If the node clocks are very accurate, then synchronization messages will be infrequent. However, accurate clocks are much more expensive and require much more energy than cheaper, less accurate clocks (for a survey of oscillator types, their cost, power consumption, and accuracy, see [Sadler 2005] and [Schmid et al. 2009]).

The clock signal is a periodic signal with some nominal frequency $f_0$. The clock signal increments a hardware counter every $1/f_0$ seconds. Every clock signal will deviate from its intended nominal frequency for various reasons (*e.g.*, changes in pressure or temperature). This deviation is termed frequency error, denoted as $f_e(t) = f_0 - f(t)$, where $f(t)$ is the frequency of the clock signal at time $t$. This error, called the *frequency drift*, is commonly expressed in parts per million (ppm).

In the example given in Section 1.3.2, the exceptionally low current draw in the idle state for the Firefly is attributable to the use of two separate clocks connected to the processor (a third clock is provided for the radio); one clock for active states that operates at 8Mhz and a low-power clock that operates at 33Khz during idle states. The 33Khz clock is driven by a low-power crystal oscillator with a frequency tolerance of ±20 ppm, which implies an accuracy of $20 \times 10^{-6}$ seconds after one second.

Let $t_d$ be the clock drift of a single node; then the worst case drift between two nodes (drifting in opposite directions) is $2t_d$. Now suppose that nodes are scheduled to communicate in non-overlapping time slots of width $t_s$; nodes are also scheduled to receive messages (enter the listening state) from other nodes in these scheduled time slots. If we assume that, in order for two nodes to communicate their time slots must overlap by at least 90%, then the communications will begin to fail after the clocks



(a) 100% overlapping time slots.

(b) The relative times for nodes A and B has drifted far apart so that less than 90% of the time slots are overlapping

Figure 1.13: The top bar depicts the perceived time from the perspective of a node A. The highlighted block is the reserved time slot for communicating with node B (bottom bar).

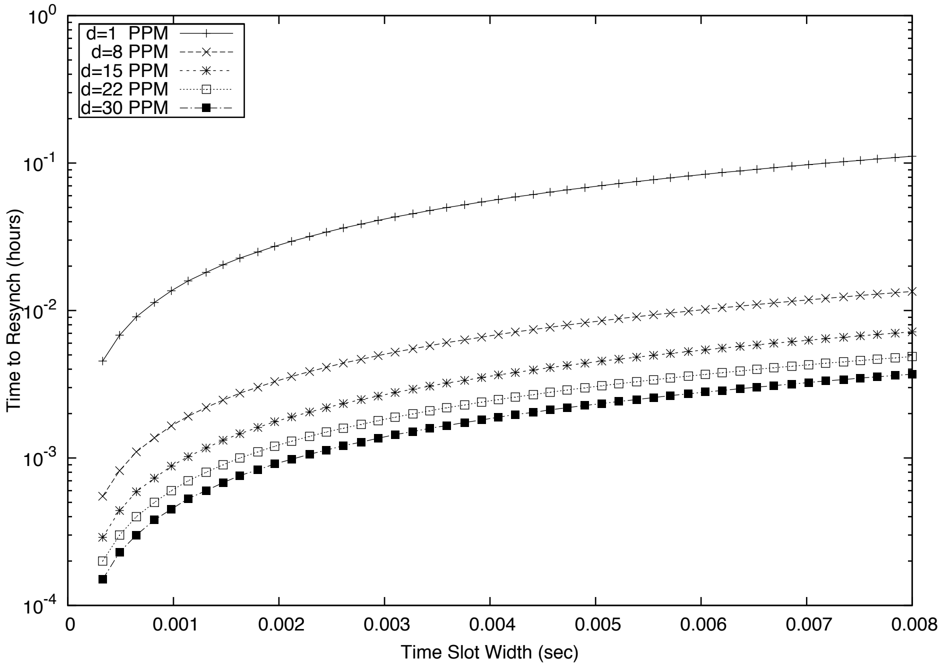


Figure 1.14: Relationship between the required frequency of resynchronization messages and the width of communication time slots for five different clock accuracies.

each drift by $t_d = \frac{1}{2}(0.1)t_s$ (see Fig. 1.13). If $d$ is the accuracy in ppm of the oscillators on the nodes, the period of time until the clocks drift this far apart is $t_f = \frac{t_d 10^6}{d}$. Every $t_f$ seconds the nodes will have to send ‘resynch’ messages to recalibrate their clocks. In the case of an oscillator with 20 ppm accuracy, these resynchronization messages would have to be sent 6 times per minute. Figure 1.14 shows the number of times per hour that nodes will have to resynchronize with respect to the size of the time slots, for several oscillator accuracies.

## 1.4 Main Contributions and Outline of the Remainder of this Thesis

The previous sections introduced the key elements of a WSN design. In the remaining chapters of this thesis we will introduce techniques for planning a deployment and replenishment strategy for a large-scale monitoring WSNs, developed from models built with many of these design elements. In the next chapter, we introduce related work in WSN deployment strategies. The differences between our proposed approach and the related literature are discussed and summarized in Tables 2.1 and 2.2.

In Chapter 3, we develop a model for the message traffic and energy consumption in WSNs (Section 3.2), using random point processes to model the expected number of clusterheads and the expected sizes of clusters. This model is used to formulate an optimization problem whose objective is to maximize the lifetime of the network while minimizing the wasted energy remaining in the network upon network failure (Section 3.3). The solution to this optimization problem provides an approach for deploying WSNs to extend lifetime. Observations from calculations and simulations support the claim that controlling both the clusterhead densities and the total node densities with respect to the distance from the base station results in longer lifetimes over strategies that either deploy nodes uniformly and/or only control the clusterhead density. This work was first presented in [Dorsey and Kam 2009]. To our knowledge, this was the first paper to study random differential deployments of sensor nodes where both the node density and the clusterhead densities are decision variables in the optimization.

In Chapter 4, we revisit the deployment issue to discuss the impact of some of the simplifying assumptions that we (and most other studies) have included in the network model. All of the studies discussed in the related works section assume an *ideal MAC*. This assumption excludes the impact of nodes waiting to access the medium or possible collisions between nodes. Additionally, in all of the studies we have surveyed that use concentric bands to model distances from the base station, there is an assumption that a message traverses exactly one hop to next band toward the base station. Chapter 4 studies the impact of the hop-count distributions as messages are forwarded from sensors to the base station, the variance in the size of clusters, and the impact of intra-cluster contention on the energy consumption. We show that the hop-count distribution and the energy consumed by nodes as they wait to access the medium do have an impact on the expected lifetime of the network.

In Chapter 5, we consider the problem of extending the lifetime of an initial deployment by added additional nodes in batches, subject to lead times for delivery. The problem is cast as a dynamic programming problem, which is then modified to reduce the dimensionality, resulting in an approximate dynamic programming problem consisting of two parts: the size of the optimal batch and the optimal allocation of nodes across the region. The problem formulation originates from the *inventory control* literature in Operations Research. We adapt this formulation for the problem of replenishing WSNs by adding a node failure forecast that considers the impact of deploying new nodes onto the network, and a myopic allocation strategy suited to node failure patterns. Results from simulation are provided. This work was initially presented in [Dorsey and Kam 2010], and it is the only study, to our knowledge, that deals with node replenishment to extend the lifetime of large WSNs.

Chapter 6 concludes the thesis with a discussion of the results and their implications.

# 2. Survey of WSN Deployment Strategies

## 2.1 Overview of WSN Deployment Strategies

We present a review of recent strategies developed for WSN deployment. We focus on the deployment of stationary sensors, and so we will not discuss recent work on mobile sensors (*e.g.*, Wang et al. [2008]) or deployments during which nodes are adjusted after the initial placement. For a comprehensive survey on the state of research on optimized node placement in WSNs, see [Younis and Akkaya 2008],[Younis et al. 2006] and the references therein.

In most of the literature, static deployment strategies typically employ one of

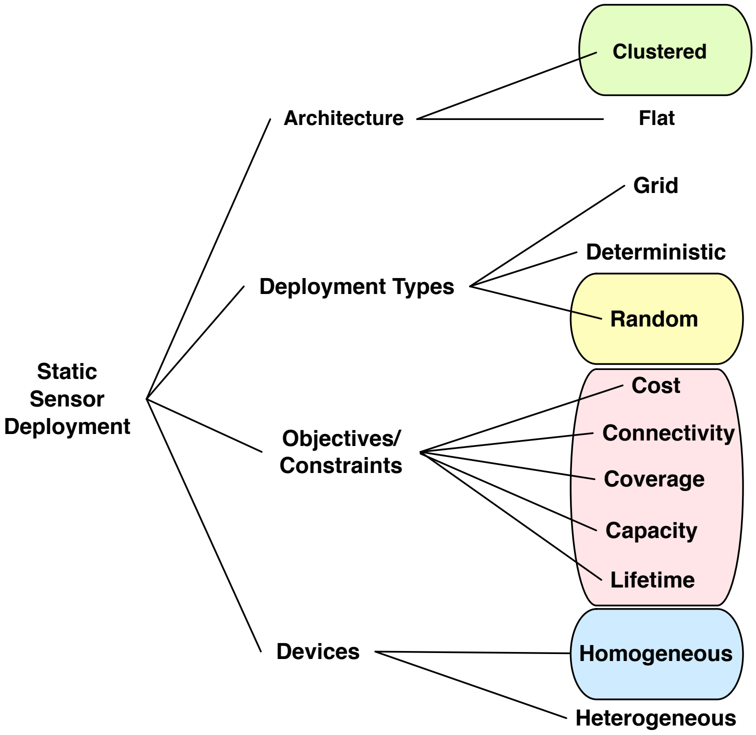


Figure 2.1: A taxonomy of WSN deployment strategies. The highlighted boxes indicate the architecture, deployment type, the various constraints and objectives, and the device type used in the proposed approach. We also study the flat architecture for comparison.

three possible approaches: **deterministic deployment**, **grid-based deployment**, and **random deployment**. In a deterministic deployment, each node placed exactly at an arbitrary location in the sensing field. In a random deployment, nodes are placed on the field according to a random spacial distribution. In this thesis, we focus on the set of deployment types, objectives, and constraints highlighted in the WSN deployment taxonomy in Figure 2.1. We consider clustered, or hierarchical, architectures (although the flat architecture is analyzed as a special case where all nodes are clusterheads), and discuss only random deployments. Devices are assumed to be homogenous (*i.e.*, all sensor nodes have the same communication capabilities). Objectives and constraints on the network include cost, connectivity, capacity, and the lifetime of the sensor network.

## 2.2 Related Work

One of the first studies to analyze the problem of random device deployment in a large-scale WSN was [Xu et al. 2005]. In their study, Xu *et al.* propose three random deployment strategies for relay nodes in a heterogeneous WSN: *connectivity-oriented*, *lifetime-oriented*, and a *hybrid* strategy. In each of the strategies, the sensor nodes are initially distributed randomly according to a uniform distribution with a density that will ensure the desired coverage. A distribution of relay nodes is then provided to meet the primary objective. The simplest of the three strategies is the connectivity-oriented strategy, since it involves deploying both sensor nodes and relays according to a uniform distribution, providing maximal connectivity everywhere on the monitored area, $\mathcal{A}$. The study introduces the *biased energy consumption rate* (BECR) phenomenon in WSNs, where relay nodes that are closer to the base station will consume energy more quickly than relays that are farther away. This phenomenon is addressed by a lifetime-oriented strategy, which computes an optimal weighted random deployment designed to provide relay node densities that are proportional to the expected energy consumption at locations on $\mathcal{A}$.

[Mhatre et al. 2005] derive a random deployment of heterogeneous nodes that minimizes cost while meeting lifetime, connectivity, and coverage constraints. However, they consider a scenario where a satellite or aircraft periodically passes over the field of deployment and gathers data from the relay nodes. Under these circumstances, there is no convergence of data into the base station because the relays are communicating via a single hop to a mobile base station, and therefore the problem of biased energy consumption is not considered. There are several other proposed deployment techniques that assume that relay nodes or clusterheads can communicate with the base station in a single hop, including the analysis of the LEACH protocol by [Heinzelman et al. 2002].

Other related works assume that the nodes will be deployed in stages. [Sun and Shayman 2007] propose initially deploying nodes uniformly over the region to provide coverage, and then deploying nodes with more energy and greater transmission range in order to provide a connected network. [Wang et al. 2007] propose a three-stage deployment where nodes are uniformly deployed first. Then relays (with more energy and greater range) are deployed to provide a minimum set covering by "putting sensors to sleep" that provide redundant coverage. Finally, additional relays are added to ensure connectivity. Many of the papers assume that relays and sensors are equipped with different hardware. Generally, as specified in commercial wireless sensor network standards such as Zigbee [Alliance 2005] , these authors assume that the relays are either tethered to an external power supply or have a much longer battery life than that of a sensor node, and that the transmission range is much larger than the sensor nodes.

Table 2.1: Summary of objectives, decision variables and constraints used in selected related literature.

| **Author** | **Objective** | **Decision Variable(s)** | **Constraint(s)** |
|---|---|---|---|
| Bandyopadhyay and Coyle [2003a] | Min total energy consumption | CH density and max number of hops between NCHs and CHs | Connectivity |
| Mhatre et al. [2005] | Min cost | Node densities and initial energy | Lifetime, connectivity, and coverage |
| Sun and Shayman [2007] | Max lifetime | Ratio of relays to SNs | Connectivity and energy |
| Wang et al. [2007] | Min cost | Ratio of relays to SNs | Lifetime and connectivity |
| Sheldon et al. [2005] | equal CH energy consumption | Density of CHs | None specified |
| Liu et al. [2006] | Max lifetime | Node densities with respect to distance from BSt | Coverage and connectivity |
| Wang et al. [2006] | Max lifetime | Density of nodes in bands | Number of available nodes |
| Liu [2006] | Maximize percent coverage | Density of nodes in bands | Traffic loads among different bands must be equal; number of nodes available |
| Iranli et al. [2005] | Max lifetime | Number and locations of CHs | Number of available nodes |
| Wu et al. [2008b] | Min residual energy/ max lifetime | Density of nodes in bands | Minimum data delivery ratio |
| Dorsey and Kam [2009] | Min residual energy/ max lifetime | Node and CH densities with respect to distance from BSt | Coverage and connectivity |

In most of the proposed strategies, the deployment is derived from the solution to a constrained optimization problem. The objectives of the optimization problems range from minimizing total cost of deploying the network (including the cost of sensors and the number of deliveries), to minimizing the probability of false alarm in detecting an intruder, depending on the application and assumptions about the hardware. Constraints are typically related to connectivity, coverage of the region, energy, cost, and network lifetime. The definitions of these constraints and also vary.

Table 2.1 lists the objectives, constraints, and decision variables for some of the related literature, and Table 2.2 contains information about definitions, models, and key assumptions.

Table 2.2: Summary of device assumptions, deployment types, and lifetime definitions for various related work.

| **Author** | **Device Types** | **Stages** | **Network Model** | **Organization** | **Lifetime** |
|---|---|---|---|---|---|
| Bandyopadhyay and Coyle [2003a] | Relays and sensors are equal | Single | Disk with BSt at center | Clustered | N/A |
| Mhatre et al. [2005] | Relays have more energy and greater range | Single | Disk with BSt at center | Clustered | First loss of coverage/ connectivity |
| Sun and Shayman [2007] | Relays and sensors are equal | Two | Disk with BSt at center, divided into bands | Flat | First loss of coverage |
| Wang et al. [2007] | Relays have more energy and greater range | Three | Square field with BSt at the middle of one side | Flat | N/A |
| Sheldon et al. [2005] | Relays have more energy and greater range | Single | Disk with BSt at center, divided into bands | Clustered; CHs do not sense | N/A |
| Liu et al. [2006] | Homogeneous nodes | Single | Disk with BSt at center | Flat | First loss of coverage |
| Wang et al. [2006] | Homogeneous nodes | Single | Disk with BSt at center, divided into bands | Flat | First node fails |
| Liu [2006] | Homogeneous nodes | Single | Disk with BSt at center, divided into bands | Clustered | Coverage < 70% |
| Iranli et al. [2005] | Relays have more energy and greater range | Single | Square field with BSt at the center | Clustered | First node fails |

Continued. . .

Table 2.2: Summary of device assumptions, deployment types, and lifetime definitions for various related work.

| **Author** | **Device Types** | **Stages** | **Network Model** | **Organization** | **Lifetime** |
|---|---|---|---|---|---|
| Wu et al. [2008b] | Homogeneous nodes | Single | Disk with BSt at center, divided into bands | Flat | First node fails |
| Dorsey and Kam [2009] | Homogeneous nodes | Single | Disk with BSt at center, divided into bands | Clustered | Reports received at BSt $< 70\%$ |

[Wang et al. 2006] provide a differential node density method to deploy sensors in order to increase the lifetime of the network. They define the lifetime of the network as the cumulative active time of the network until the first sensor is out of power. The field of deployment is modeled as a 2-D disk with a radius $R$, divided into $n$ bands of width $\frac{r_c}{2}$ each, based on the distance from the base station. Figure 2.2 shows the model used in their work. They divide the area into $n$ 'levels' from the sink to the outside (the outer level has radius $R_n = R$). The boundaries of level $i$ are the circles of radius $R_i$ and $R_{i-1}$. The network architecture is flat. [Wang et al. 2006] do not consider the case where nodes form clusters and aggregate data before forwarding them to the base station. Their study also does not consider requirements for connectivity or coverage. Assuming that the amount of data originating from a band is proportional to the area of the band, they propose to increase the node density near the base station in order to reduce the traffic load that each node in the neighborhood of the base station will have to bear. Their simulations show that the lifetime of this deployment is greater than that of a uniform deployment.

Using a similar approach, [Liu 2006] divides $\mathcal{A}$ into bands of equal width and considers homogeneous nodes. The objective is to maximize the fraction of $\mathcal{A}$ that is covered by at least one sensor under the constraint that the average rate of messages being relayed (average load) through nodes should be equal in all bands. The constraint is used to ensure load balancing so that, on average, nodes will become inoperable at the same time. After initial deployment, the nodes communicate with one another to form clusters using the approach proposed by [Bandyopadhyay and Coyle 2003b]. Once the clusters are formed, a routing tree is created among the clusterheads to forward packets toward the base station. The authors assume that when a clusterhead is depleted of energy, the base station will send a network reorganization request to all of the nodes in order to reconstruct the gradient. The reconstruction algorithm requires that the nodes be aware of their own location and the location of their neighbors through the use of a GPS device.

[Wu et al. 2008b] show that when $\mathcal{A}$ is modeled as a circle with concentric bands surrounding the base station, equal energy dissipation is not possible for all bands because the outer-most band is not forwarding any traffic. They suggest that the number of nodes deployed in each band should increase geometrically from the band nearest the base station to the outer band in order to minimize the residual energy left in the network once connectivity is lost due to node failures. The deployment

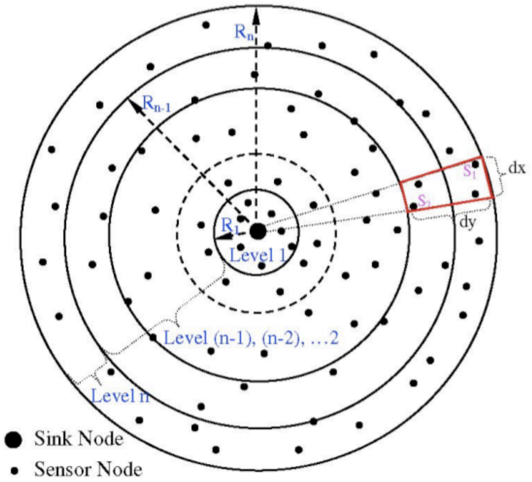


Figure 2.2: Model of $\mathcal{A}$ divided into concentric bands of equal width (from Wang et al. [2006])

strategy is proposed in conjunction with a routing protocol that serves to balance traffic among the nodes. However, this protocol requires significant control over the placement of the nodes in order to be effective.

In order to balance energy consumption in a clustered network, both intra-cluster and inter-cluster communications should be considered. One approach, proposed in [Shu et al. 2005] and also in [Li et al. 2005], is to assign larger cluster sizes to CHs that are further from the BSt and have fewer packets to relay (see Figure 2.3). However, this approach is constrained by the maximum power transmission level on the nodes. In [Gun et al. 2007], the authors propose a deployment strategy where nodes are deployed with differential densities from the BSt. In order to balance the load between nodes toward the BSt and nodes toward the outer region, they deploy clusterheads with variable battery capacities. This approach is limited by the cost of providing different battery hardware for different nodes and the availability of batteries with their calculated energy capacities.

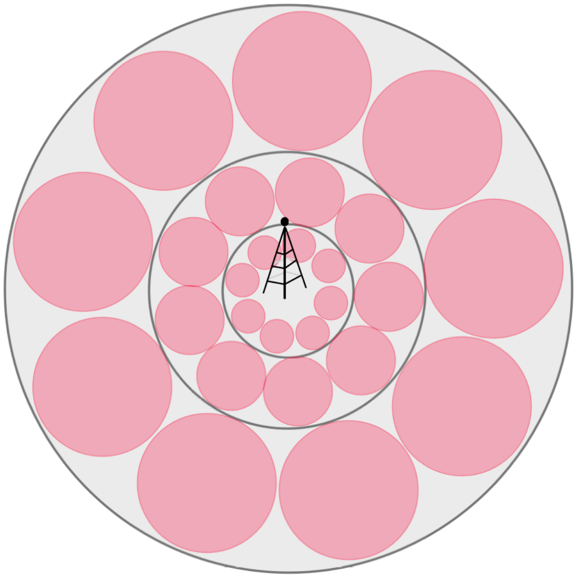


Figure 2.3: Larger clusters are assigned to bands that are farther from the base station in order to balance clusterhead energy consumption.

The WSN deployment approach proposed in the next chapter extends our paper, [Dorsey and Kam 2009], where we employ probabilistic CH selection while varying the density of nodes and probability of being a CH over discrete distances from the BSt. The objective is to maximize the network lifetime (defined as the time until a percentage of the sensor reports do not reach the BSt), while minimizing the energy remaining in the network at network failure. We show that by optimizing the expected lifetime over node densities and CH densities, the lifetime is extended over other approaches discussed in the literature.

# 3. Non-Uniform Deployment in Clustered WSNs – Part I

## 3.1 Introduction

We present and extend our previous work on a strategy for deploying a large-scale clustered wireless sensor network with random (or coarse-grain controlled) placement [Dorsey and Kam 2009]. The nodes organize clusters using a distributed clustering algorithm; clusterhead (CH) selection is performed in the manner originally proposed by [Heinzelman et al. 2000]. The strategy includes differential densities for both sensor nodes and clusterheads with respect to the distance from the base station in order to maximize lifetime.

In Section 3.2, we present the network model and assumptions, and derive expressions for energy consumption. An expression is derived to approximate the lifetime of a differentially deployed random network using the density of clusterheads and non-clusterheads (NCHs) as variables. In section 3.3 we show how this lifetime expression provides an objective function to be maximized, subject to constraints on coverage, connectivity, capacity, and cost. The numerical results in section 3.4 and the simulation results in 3.5 show that a differential node deployment with a uniform CH density increases the lifetime of the network over a Uniform deployment. Moreover, the addition of a suitable differential CH density further increases lifetime over the differential node deployment with uniform CH density. In the following sections, the three variations (deployment with uniform node density and CH density, a differential node deployment with a uniform CH density, and a differential deployment with a differential CH density) will be referred to as the *Uniform*, *Static p*, and *Dynamic* deployments, respectively.

## 3.2 Network Model and Problem Formulation

This section describes the basic components of the model used to formulate an optimization problem for WSN design. We make several simplifying assumptions here for the sake of tractability. They include, (1) the model used to characterize spatially varying node densities (annular bands of uniformly distributed nodes, each with different parameters); (2) an assumption about the number of hops a message will travel until it reaches the base station; and (3) an assumption of an ideal channel access method. The last two simplifications will be removed in Chapter 4, where we compute the expected number of hops each message will travel in order to reach the next band and describe a channel access model that accounts for contention.

Models for the distribution of nodes over the deployment area, the network organization, and the energy consumption for individual nodes are combined in the next subsections under the simplifying assumptions. These models are used to derive expressions for the average energy consumption in each band and the energy remaining in each band at network failure. The energy consumption and residual energy expressions are used as objective functions in the optimization problem outlined in section 3.3.

### 3.2.1 Area of deployment ($\mathcal{A}$) and distribution of nodes

The monitored area is modeled as a disk of radius R that is divided into $J$ annular bands of equal width (see Figure 3.1). The width of each band is equal to the communication radius, $r_{tx}$. The base station is located at the center of the disk. We assume that the nodes are to be distributed on the region according to a set of homogeneous spatial Poisson processes (see Sec. 1.3.6), one process for each band, with intensity $\lambda_j$ for the $j^{th}$ band.

The number of nodes in band $j$ is a Poisson random variable, $N_j \sim \mathcal{P}(\lambda_j)^1$. The area of band $j$ is $A_j$ and the expected number of nodes in band $j$ is $\mathbb{E}[N_j] = \lambda_j A_j$, where $A_j = \pi\left(r_j^2 - r_{j-1}^2\right)$. The widths of the bands are chosen to be equal to the communication radius of each node, $r_{tx}$, so that the number of bands, $J$ is $\lceil \frac{R}{r_{tx}} \rceil^2$.

### Justification of the use of annular bands

The choice of modeling the region as a set of annular bands is meant for ease of analysis, but it requires some justification. First, we assume that the spatial node density is symmetric about the sink node. Consider that, as the width of the bands approaches zero, the set of spatial Poisson processes become a single spatial Poisson process with an intensity $\lambda(r)$, a continuous function of the distance $r$ from the base station. We assume that $\lambda(r)$ is a smooth and continuous function of $r$. Then, by the Mean Value Theorem, there exists a radius $r_j \in [(j-1)r_{tx}, jr_{tx}]$ such that the number of nodes in band $j$ is given by

$$N_j = 2\pi\lambda(r_j)\int_{(j-1)r_{tx}}^{jr_{tx}} r dr = \pi r_{tx}^2 \lambda_j (2j-1), \tag{3.1}$$

where $\lambda_j = \lambda(r_j)$. So the annular bands of uniform spatial Poisson processes are an approximation of a true underlying process with a continuous parameter that is assumed to be a smooth function of the distance from the center of the disk. This assumption appears to be supported by the optimal densities that are computed in this chapter, which also appear to be discrete samples of a smooth function (*e.g.*, Figure 3.5).

[1] The expression $X \sim \mathcal{P}(\lambda)$ means "X is (asymptotically) distributed as $\mathcal{P}(\lambda)$", where $\mathcal{P}(\lambda)$ denotes a Poisson distribution with parameter $\lambda$

[2] $\lceil x \rceil$ is the smallest integer not less than $x$.

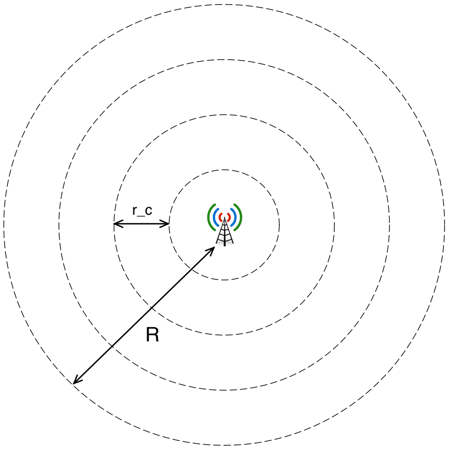


Figure 3.1: The area to be monitored is separated into annular bands of width $r_{tx}$, the communication radius. The base station is located at the origin of the circle.

### 3.2.2 Network organization

The network lifetime is measured in terms of *rounds*, which are of arbitrary length. During each round, nodes initiate CH selection once and then sample the environment and report their data to the base station $K$ times.

According to the LEACH protocol [Heinzelman et al. 2000], an individual node becomes a CH in a given round with probability $p$. The decision to become a CH is made at each node by choosing a random number between 0 and 1 at preset time intervals and comparing this number to the threshold $T(n)$, which is a function of $n$, the sensor number. This threshold is

$$T(n) = \begin{cases} \frac{p}{1-p\left(r mod \frac{1}{p}\right)} & \text{if } n \in G \\ 0 & \text{otherwise,} \end{cases} \tag{3.2}$$

where $r$ is the current round and G is the set of nodes that have not been CHs in the last $\lceil \frac{1}{p} \rceil$ rounds. If a node becomes a CH, it broadcasts this decision, and the neighboring nodes that have not become CHs will align to the CH whose advertisement has the highest received signal strength (since we use the radio model in Eq. 1.2, the highest signal strength in free space will correspond to the CH with the shortest Euclidean distance to the node).

### 3.2.3 Expected number of clusterheads and non-clusterheads in each band

The network model assumes that all nodes are distributed on the region according to a Poisson point process(PPP) $N_j \sim \mathcal{P}(\lambda)$. Since CHs are selected randomly in each round (*see* sction 3.2.2), we can think of CH selection as a series of independent Bernoulli trials $\{\xi_n\}$ with mean $p$ over the points of $N_j$. That is, each of the $N_j$ points is retained with probability $p$ by a process $X_1 = \sum_{n=1}^{N} \xi_n$, and $X_1 \sim \mathcal{P}(p\lambda)$. Thus, CH selection constitutes a *thinning* of $N_j$. Now the clustered network can be thought of as a random geometric graph $G_n(r_{tx})$ with vertices distributed according to two different types of PPPs; a process with intensity $\lambda_{j,1} = \lambda_j p_j$ for CHs and a process of intensity $\lambda_{j,0} = (1 - p_j)\lambda_j$ for NCHs. The expected number of nodes in band $j$ is $\lambda_j A_j$, so the expected number of CHs, $\mathbb{E}[C_j]$, in band $j$ is $\lambda_j A_j p_j$. The expectations are conditioned on the event $\{N_j = n\}$.

Each NCH joins the nearest CH to form a Voronoi tessellation [Preparata and Shamos 1985], dividing the graph into cells (see Figure 3.2) bounded by lines that are equidistant from two points (CHs). Each Voronoi cell corresponds to a Poisson process point with intensity $p\lambda$, called the *nucleus*. Using the results from [Zuyev 1996] and [Bandyopadhyay and Coyle 2003b], if $N_v$ is a random variable denoting the number of Poisson process points with intensity $(1 - p)\lambda$ in each Voronoi cell, then the expected value of $N_v$ is

$$\mathbb{E}[N_v] = \frac{(1-p)\lambda}{p\lambda} = \frac{1-p}{p}. \tag{3.3}$$

Thus, the number of NCHs associated with a CH in band $j$ is a random variable $S_j$,

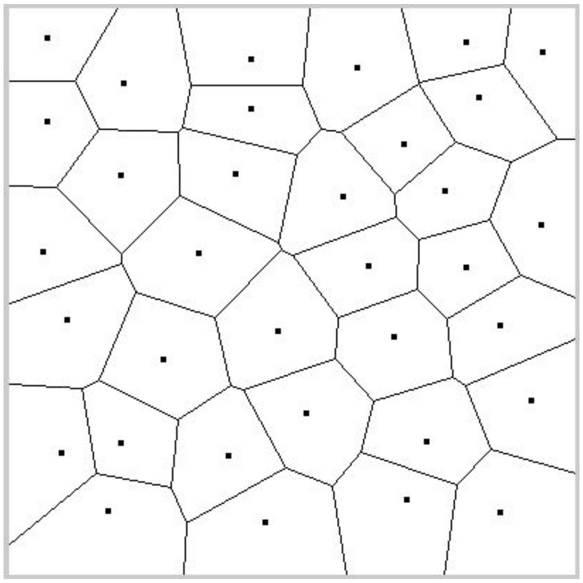


Figure 3.2: An example of a Voronoi Diagram; each line is equidistant from two points, where the points represent CHs.

and the expected number of NCHs in each cluster is

$$\mathbb{E}[S_j] = \frac{1-p_j}{p_j} = \frac{\lambda_{j,0}}{\lambda_{j,1}}. \tag{3.4}$$

The following are some additional key assumptions in the development of the model.

1. An ideal MAC layer is assumed; nodes are scheduled to transmit and receive according to a sequence that will prevent packet collisions and retransmissions.
2. A message transmitted from a CH in band $j$ to be delivered to the base station will be forwarded by one node in each of the bands $j-1$, $j-2$,... 1 in order to reach the base station. The distance between a node and the base station is equivalent to $\lceil \frac{distance}{r_{tx}} \rceil$ hops.
3. The nodes all have the same capabilities (homogeneous) and are not able to adjust their transmit power level.

4. A shortest-path routing protocol is assumed for the CH overlay network. When a CH node communicates data to the base station, only CH nodes on the path to the base-station forward the message.

5. Each sensor node is assumed to be deployed with the same initial energy, $\mathcal{E}_0$.

6. No data compression occurs between CHs; all CH messages are forwarded to the base station.

Assumptions 1 and 2 will be removed in Chapter 4.

### 3.2.4 Energy consumption

**Energy consumed during reporting**

We can now consider the energy consumed in each band of the network during cluster formation, communication between CHs and NCHs, and communication between CHs. We will also assume that the nodes in different bands can use different values of $p$ ($p_j$ for the $j$th band). Then, at the start of each round, the nodes each select a random variable and compare it to the threshold, which is a function of $p_j$. The expected amount of energy used in band $j$ during cluster formation is the energy consumed as each of the $C_j$ CHs broadcast their status and each of the $S_j$ nodes surrounding each CH responds to the broadcast. Each CH receives $S_j$ responses and transmits once, and each NCH transmits once, so the energy consumed by nodes in band $j$ in a single round is $C_jS_jE_{rx} + C_jE_{tx} + C_jS_jE_{tx} =$

$$\mathrm{C}_j \left( E_{tx} + \mathrm{S}_j E_{rcv} + \mathrm{S}_j E_{tx} \right). \tag{3.5}$$

Next, each of the NCHs sends a message to their CH after sampling the environment. The energy consumed is from the NCHs transmitting once each ($C_jS_jE_{tx}$)

and the CHs receiving these transmissions ($C_jS_jE_{rx}$). This communication occurs $K$ times per round, so the energy consumed in the $jth$ band is

$$K\mathrm{C}_j\mathrm{S}_j\left(E_{rcv}+\ E_{tx}\ \right). \tag{3.6}$$

After each CH aggregates the data from the surrounding nodes, a message is forwarded to a CH in the next band forward to deliver to the base station node. The number of messages received by CHs in the $jth$ band is the sum of messages from the $j-1$ band, $j-2$ band,...,$J$ band. The CHs then forward to the next band all of the messages received from the previous band in addition to the messages from their own aggregation. The energy consumed in the $jth$ band is therefore

$$KE_{rcv}\sum_{k=j+1}^{J}\mathrm{C}_k\ +KE_{tx}\sum_{k=j}^{J}\mathrm{C}_k\ . \tag{3.7}$$

Summing all of the above energies and substituting $\mathbb{E}[C_j]=\lambda_jA_jp_j$ for $C_j$ and $\mathbb{E}[S_j]=\frac{(1-p_j)}{p_j}$ for $S_j$ gives an expression for the expected total energy consumed in each round for the $jth$ band as a function of $p_j$ and $\lambda_j$, namely:

$$\begin{aligned}\mathbb{E}[E_r^j]\ &=\ \lambda_jA_jp_j\left(E_{tx}+\frac{(1-p_j)}{p_j}E_{rcv}+\frac{(1-p_j)}{p_j}E_{tx}\right)\\ &\quad+\ K\lambda_jA_jp_j\frac{(1-p_j)}{p_j}\left(E_{rcv}+\ E_{tx}\ \right)\ +\\ &\quad KE_{rcv}\sum_{k=j+1}^{J}\lambda_kA_kp_k\ +KE_{tx}\sum_{k=j}^{J}\lambda_kA_kp_k\ .\end{aligned} \tag{3.8}$$

This expression can be simplified to give the expected energy cost for communication per round in each band,

$$\mathbb{E}[E_r^j] = (K+1)\, E_{tx}\lambda_j p_j A_j + (K+1)\, E\lambda_j A_j\,(1-p_j) + KE \sum_{k=j+1}^{J} \lambda_k A_k p_k, \qquad (3.9)$$

where $E = E_{tx} + E_{rx}$.

### Estimation of residual energy within bands

Next, we consider the balance of energy usage *within a band* and the expected amount of residual (wasted) energy $\mathbb{E}[E_w^j]$ left over after a portion of the nodes in the band have failed. We show that, given a deployment of nodes employing a balanced CH selection in each round, the nodes that remain alive in a single band after the first node(s) fail (due to energy loss) will have insufficient energy remaining to perform the duty of a CH.

That nodes are being designated as CH in a balanced manner (via the LEACH protocol) suggests that, at a given time (before any node failures) and in a specific band, the number of times that any node has held the role of CH should be nearly the same. If there are $C_j$ CHs at any given time among $N_j$ nodes in band $j$, then all nodes should be a CH approximately once every $N_j/C_j = 1/p_j$ rounds. Therefore, if a particular node has been a CH $n$ times in round $t$, we will assume that the other nodes in the band will have been a CH $n$ or $n \pm 1$ times. This is a strong assumption; there will likely be more variance in the distribution of CH responsibilities. The result that follows should be considered a lower bound on the wasted energy in each band.

First we consider the energy consumed by a particular node in band $j$ that has been a CH $n$ times in $t$ rounds. If we denote the energy consumed in a round by a NCH as $E_{NCH}$ and the energy consumed by a CH in a round as $E_{CH}$, then the total energy consumed by a node after $t$ rounds in band $j$ is

$$\begin{aligned} x_j(t,n) &= (t-n)E_{NCH} + nE_{CH} \qquad (3.10)\\ &= \left(t - \frac{C_j}{N_j}t\right)E_{NCH} + \left(\frac{C_j}{N_j}t\right)E_{CH}. \end{aligned}$$

Then, this node will fail in round $t_f \in \mathbb{Z}$. The value of $t_f$ is computed by solving $\mathcal{E}_0 - x_j(t,n) = 0$ for $t$ to get

$$t_f = \left\lceil \frac{tC_j\left(E_{NCH} - E_{CH}\right) + N_j\mathcal{E}_0}{N_jE_{NCH}} \right\rceil. \qquad (3.11)$$

All other nodes in band $j$ that have also been designated as CH $n = \lceil\frac{C_j}{N_j}\rceil t$ times before (or in round $t = t_f$) are also failing. The question now is: how much energy do the remaining nodes have after $t_f$? The nodes that remain after this round have either been CH exactly $n$ times and will soon fail or they have been CH $n-1$ times (or less). The amount of energy remaining in surviving nodes that have been a CH $n-1$ times in $t_f$ rounds is

$$\begin{aligned} \mathbb{E}[E_w^j] &= \mathcal{E}_0 - x_j(t_f, n-1) \qquad (3.12)\\ &= \mathcal{E}_0 - \left(t_f - \frac{C_j}{N_j}t_f\right)E_{NCH} - \left(\frac{C_j}{N_j}t_f\right)E_{CH} \qquad (3.13)\\ &= E_{CH} - E_{NCH}. \qquad (3.14) \end{aligned}$$

Expression 3.14 is obtained by substituting the expression in 3.11 into 3.13 for $t_f$ and simplifying. The residual energy is insufficient for the CH's responsibility in a single round (because $E_{CH} - E_{NCH} < E_{CH}$). This result suggests the existence of a critical phase transition in the network, where after the first nodes have failed, the remaining nodes will either soon fail or will not be able to perform their duties. The length of this phase (and the number of nodes to fail in each period after the phase transition) will be proportional to $1/p_j$. The expression for $\mathbb{E}[E_w^j]$ is only valid for the first band that fails (at $t_f$). The remaining bands will not yet have failed, and will have more residual energy left; the objective is to minimize this unusable energy (Section 1.3.7, Equation 1.13 states the relationship between residual energy and network lifetime). This observation means that we need an expression for the energy in the nodes in remaining bands at time $t_f$, the time that the first band has failed.

The reason that all bands do not fail at the same time is due to the biased energy consumption rate effect (BECR, Section 1.2). Some bands must relay more data from outer bands than others, resulting in different values of $E_{CH}$ for each band (the value of $E_{NCH}$ is the same for all bands). We will now use the notation $E_{CH}^j$ to denote the energy consumed by a node as a CH in band $j$ during a single round. The expected number of CHs in band $j+1$ that relay their traffic to a particular CH in band $j$ is $\mathbb{E}[C_{j+1}]/\mathbb{E}[C_j]$. Then, the expected value of $E_{CH}^j$ is

$$\mathbb{E}[E_{CH}^j] = K \sum_{k=j}^{J-1} \frac{\mathbb{E}[C_{k+1}]}{\mathbb{E}[C_k]} E + K\mathbb{E}[S_k]E_{rx} + KE_{tx}, \tag{3.15}$$

and the expected residual energy per node in band $j$ at $t_f$ is[3]

$$\begin{aligned}
\mathbb{E}[E_w^j] &= \lambda_j A_j \left(\mathcal{E}_0 - x_j(t_f, n-1)\right) \qquad (3.16)\\
&= \lambda_j A_j \left(\mathcal{E}_0 - \left(t_f - \frac{\mathbb{E}[C_j]}{\mathbb{E}[N_j]} t_f\right) E_{NCH} - \left(\frac{\mathbb{E}[C_j]}{\mathbb{E}[N_j]} t_f\right) \mathbb{E}[E_{CH}^i]\right)\\
&= \lambda_j A_j \left(\mathcal{E}_0 - t_f\left(1-p_j\right) KE_{tx} - p_j t_f \mathbb{E}[E_{CH}^i]\right).
\end{aligned}$$

### 3.2.5 Expected lifetime

Each node starts with the same battery level, $\mathcal{E}_0$, and the expected number of nodes in band $j$ is $\lambda_j A_j$. Therefore, the number of rounds until a band is expected to

[3] The notation $t_f$ in this expression refers to the time (rounds) that the first band failed, not the mathematical expression in 3.11.

run out of energy (the expected lifetime) is $\mathcal{L}_j$:

$$\mathbb{E}[\mathcal{L}_j] \;=\; \frac{\lambda_j A_j \mathcal{E}_0 - \mathbb{E}[E_w^j]}{K\mathbb{E}[E_r^j]}. \tag{3.17}$$

The lifetime of the network is the lifetime of the first band to fail, thus:

$$\mathbb{E}[\mathcal{L}_{net}] = \min_j \mathcal{L}_j. \tag{3.18}$$

Our problem is to find a deployment strategy that will maximize $\mathbb{E}[\mathcal{L}_{net}]$ subject to constraints on the node densities regarding the total number of available nodes, the connectivity of the network, and the coverage of the sensors over $\mathcal{A}$.

## 3.3 WSN Lifetime Optimization

### 3.3.1 Computation of optimal CH probability, p and node density, $\boldsymbol{\lambda}$

We denote the CH probability in band $j$ as $p_j$ and the density of nodes in band $j$ as $\lambda_j$. We use the boldface $\mathbf{p}$ and $\boldsymbol{\lambda}$ to denote the vectors containing $J$ values for CH probabilities and node densities, respectively. We seek values of $p_j$ and $\lambda_j$ for $j = 1 \ldots J$ that will maximize the lifetime of the entire network. This is equivalent to maximizing the minimum lifetime for all bands. Adding nodes to the area closest to the base station may increase the expected lifetime beyond the lifetime of the second band, so the lifetime of the second band will be the lifetime of the network. However, adding nodes to the second band will increase the amount of traffic to the innermost band, decreasing its expected lifetime. Optimal values of $\mathbf{p}$ and $\boldsymbol{\lambda}$ will need to balance the energy consumption over all bands simultaneously.

### 3.3.2 Constraints

The lifetime of the network is maximized subject to constraints for connectivity between CHs and coverage of the monitored area. In our calculation, we estimate the minimum number of nodes required to provide adequate coverage in band $j$, and the number of CHs that provide a connected CH subnetwork. These are expressed as functions of the transmit radius $r_{tx}$ of the nodes and the area of band $j$, $A_j$. Since all nodes in all bands have a fixed communication radius and area, we can solve Equation 1.6 (which is Equation 28 in Bettstetter [2004]) for $n$ to obtain a minimum CH density for all bands to provide connectivity $\lambda_{\text{con}}$. The minimum number of CHs in each band that guarantee connectivity is:

$$C_j^{conn} = \frac{-A_j \cdot W\left(-\frac{\pi r_{tx}^2 \ln\left({p_c}^{-1}\right)}{A_j}\right)}{\pi r_{tx}^2}, \tag{3.19}$$

and therefore the minimum density is given by

$$\lambda_{\text{con}} = -\frac{1}{\pi r_{tx}^2} W\left(-\frac{\pi r_{tx}^2 \ln\left({p_c}^{-1}\right)}{A_j}\right). \tag{3.20}$$

The function $W$ in Eqns. 3.19 and 3.20 is the *LambertW* function[4] [Corless et al. 1996], which satisfies $W(x)e^{W(x)} = x$. The domain of the LambertW function is the interval $\left[-\frac{1}{\mathrm{e}}, 0\right]$. If $x$ is a real number, two real values for $W(x)$ are possible for $-\frac{1}{\mathrm{e}} \leq x \leq 0$, the *principal branch* $W_0(x)$ and a *non-principal branch* $W_{-1}(x)$. This non-principal branch ($B = -1$) gives the desired real result for $C^{conn}$ (the plot of the LambertW function is shown in Figure 3.3; the desired branch is shown in red). As a reminder, the value $p_c$ is the desired probability of connectedness. We can show that the argument of the *LambertW* function will always be within the required interval

[4] A MATLAB function that uses Halley's method to compute the LambertW is described in http://blogs.mathworks.com/cleve/2013/09/02/the-lambert-w-function/ (accessed 9/20/2013)

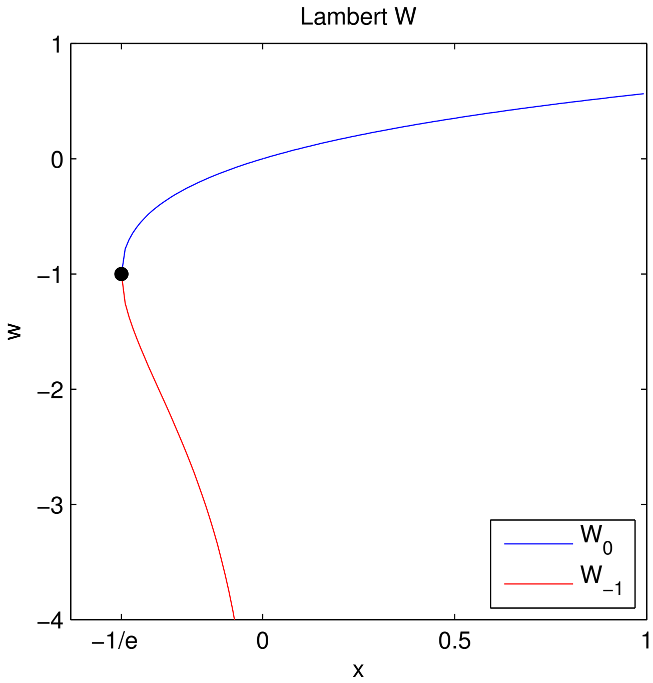


Figure 3.3: The LambertW function. The non-principal branch is shown in red.

to compute the constraint if the parameter $p_c \geq 0.7$. We want to show that

$$-\frac{1}{\mathrm{e}} \leq -\frac{\pi r_{tx}^2 \ln\left(\frac{1}{p_c}\right)}{A_j} \leq 0,\ \forall i \in 1, 2, \ldots J. \tag{3.21}$$

First, we show that the argument is within the interval for $j = 1$. Since the number of bands is given by $J = \lceil R/r_{tx} \rceil$, each annulus has width equal to $r_{tx}$ (assuming that the radius of the deployment area is a multiple of $r_{tx}$). Therefore, the first band has area $A_1 = \pi r_{tx}^2$ and

$$-\frac{\pi r_{tx}^2 \ln\left(\frac{1}{p_c}\right)}{A_1} = -\ln\left(\frac{1}{p_c}\right). \tag{3.22}$$

Solving $-\ln(1/p_c) = -1/\mathrm{e}$ for $p_c$ gives the minimum connectivity probability that will yield valid results, $p_c^{min} = \mathrm{e}^{-\frac{1}{\mathrm{e}}} \approx 0.69$. The upper bound of $p_c$ is 1. In order to show that the *LambertW* argument is in the function's domain for $j = 2, 3, \dots J$, note that, for the $j^{th}$ band,

$$-\frac{\pi r_{tx}^2 \ln\left(\frac{1}{p_c}\right)}{A_j} = -\frac{\pi r_{tx}^2 \ln\left(\frac{1}{p_c}\right)}{\pi\left(\left(jr_{tx}\right)^2 - \left(\left(j-1\right)r_{tx}\right)^2\right)}. \tag{3.23}$$

Now remove $\ln(1/p_c)$ from the right side of 3.23 and note that the sequence defined as

$$S_j^J \triangleq -\frac{\pi r_{tx}^2}{\pi\left(\left(jr_{tx}\right)^2 - \left(\left(j-1\right)r_{tx}\right)^2\right)} \tag{3.24}$$

is greater than $-1$ for all values of $j \in [1, J]$. Expressions 3.23 and 3.24 show that the argument of the LambertW function is is never less than $-1/\mathrm{e}$ when $p_c \geq 0.7$. Also, since

$$S_j^J = \left\{-1, -\frac{1}{3}, -\frac{1}{5}, -\frac{1}{7}, \dots\right\} \to 0$$

as $J \to \infty$, the argument of the *LambertW* function in 3.19 and 3.20 is never greater than 0.

We also set a coverage constraint in terms of the connectivity constraint $\lambda_{\text{con}}$. As noted in Sec. 1.3.6, derivations for coverage bounds are usually arrived at through the same arguments as for connectivity in random graphs. Also, several authors have provided results that show a linear relationship between connectivity and coverage (*e.g.*, Zhang and Hou [2005b]). Thus, the constraint for coverage is simply $\lambda_j \geq \alpha_{\text{cov}}\lambda_{\text{con}}$, where $\alpha_{\text{cov}} > 0$ is a parameter that is determined by the application. Finally, we assume that there is a fixed number of nodes to be deployed (inventory).

### 3.3.3 Optimization problem

The optimization problem is to maximize the minimum band lifetime over the CH probabilities $\mathbf{p}$ and the node densities $\boldsymbol{\lambda}$, subject to the inventory, coverage and connectivity constraints:

$$\max_{p,\lambda} \mathcal{L}_{net} \tag{3.25}$$

*subject to*

$$\begin{aligned} \lambda_j p_j &\geq \lambda_{\text{con}} && (3.26) \\ \lambda_j &\geq \alpha_{\text{cov}}\lambda_{\text{con}} && (3.27) \\ \sum_{j=1}^{J} N_j &\leq \text{inventory} && (3.28) \\ 0 \geq p_j &\leq 1 && (3.29) \\ \lambda_j &\geq 0 \end{aligned}$$

The constraints are set to ensure that the following are satisfied:

**1)** The initial density in each band is greater than the minimum density required for coverage (3.26) and connectivity (3.27).

**2)** The total number of sensors deployed is not greater than the number of sensors available (3.28).

### Problem transformation

The objective function (3.25) is a maximization over the band with the minimum lifetime. In order to make the problem more tractable and to retain a value of the network lifetime during optimization, we introduce an additional *scalar* variable, $z$ and then reformulate the problem with new objectives and an additional constraint:

$$\max_{p,\lambda} z \tag{3.30}$$

*subject to*

$$
\begin{aligned}
\mathcal{L}_j - z &\geq 0 && (3.31)\\
\lambda_j p_j &\geq \lambda_{\text{con}} && (3.32)\\
\lambda_j &\geq \alpha_{\text{cov}} \lambda_{\text{con}} \\
\sum_{j=1}^{J} N_j &\leq \text{inventory} \\
0 \geq p_j &\leq 1 \\
\lambda_j &\geq 0
\end{aligned}
$$

This formulation seeks to maximize $z$ subject to $\mathcal{L}_j - z \geq 0$, or $\mathcal{L}_j \geq z$ for all bands $j \ldots J$. However, since $z$ is a scalar it does not provide an active bound for all bands, but only the band with the minimum lifetime. Thus, this formulation is equivalent to the max-min optimization implicit in 3.25.

### 3.4 Numerical Results

The objective function and constraints above were written in *A Modeling Language for Mathematical Programming* (AMPL) [Fourer et al. 1989] and calculated using a numerical solver (Sparse Nonlinear OPTimizer, SNOPT [Gill et al. 2002]) for three variations on the variables $p$ and $\lambda$. In the first variation, we assumed that $\lambda$ and $p$ were scalars; the lifetime was maximized over a single value of $p$ and a single value of $\lambda$. This formulation is equivalent to assuming that the network is to be deployed with uniform density and a single parameter $p$ determines the CH density over the network. This is the case of distributed clustering algorithms that use a random selection to determine CHs. In the second case, we assumed that the density of nodes could vary over the various regions on the network ($\boldsymbol{\lambda}$ is a vector), but $p$ was still a global scalar parameter; this is similar to the approach described in [Wang et al. 2006] and [Liu 2006]. Finally, we compute optimal vectors $\boldsymbol{\lambda}$ and $\mathbf{p}$, to demonstrate the case where the density in each region and the probability of being a CH may vary between regions. As a reminder, the three cases (deployment strategies) are referred to as the *Uniform*, *Static p*, and *Dynamic* deployments, respectively.

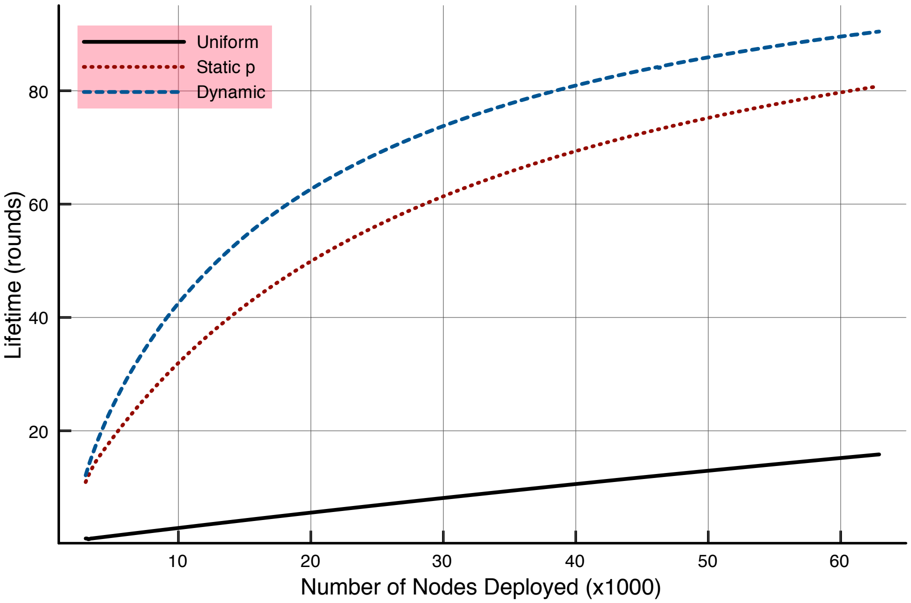


Figure 3.4: Expected lifetimes for the three deployment variations.

In all of the numerical calculations in this section, the radius of the region of deployment is assumed to be $R = 100$ arbitrary units. The radius of communication for all nodes is $r_{tx} = 6.6$ and $J = \lceil R/r_t x \rceil = 15$. The parameter $\alpha_{\text{cov}}$ for the coverage constraint is set to 1.2. Figure 3.4 shows the expected network lifetimes for the three variations as the number of nodes initially deployed grows. Both Static $p$ and the Dynamic deployments perform significantly better than the Uniform deployment. Also, the expected lifetime for the Dynamic deployment is greater than or equal to

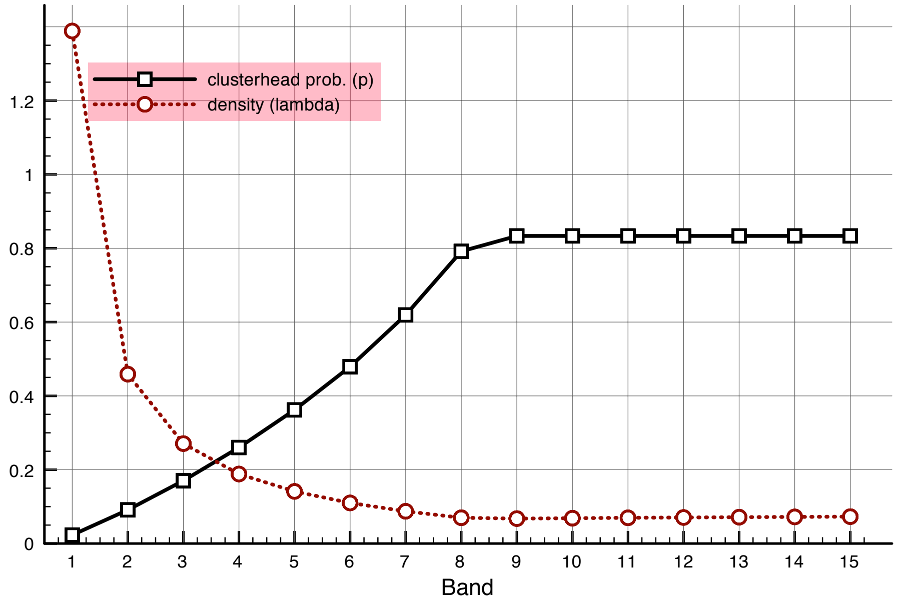


Figure 3.5: Optimal values of $\lambda$ and $p$ for 5000 nodes deployed over a region with radius $R = 100$ and communication range $r_{tx} = 100/15 \approx 6.6$.

the Static $p$ deployment.

The computed values of $\boldsymbol{\lambda}, \mathbf{p} \in \mathbb{R}^J$ for a Dynamic deployment of a 5000-node network with $R = 100$ and $r_{tx} = R/15$ are shown in Figure 3.5. The strategy is to deploy nodes very densely near the base station, but to keep the number of CHs to a minimum in order to meet the connectivity constraint. In the outer bands, the objective is to deploy just enough nodes to meet connectivity and coverage constraints. In the specific example of Figure 3.5, the bands beyond band 8 are deployed with the minimum density of nodes and CHs to satisfy the constraints

$$\begin{aligned} \lambda_j p_j &\geq \lambda_{\text{con}} \\ \lambda_j &\geq \alpha_{\text{cov}} \lambda_{\text{con}}. \end{aligned}$$

The coverage parameter is $\alpha_{\text{cov}} = 1.2$, so 80% of the nodes are meeting the minimum

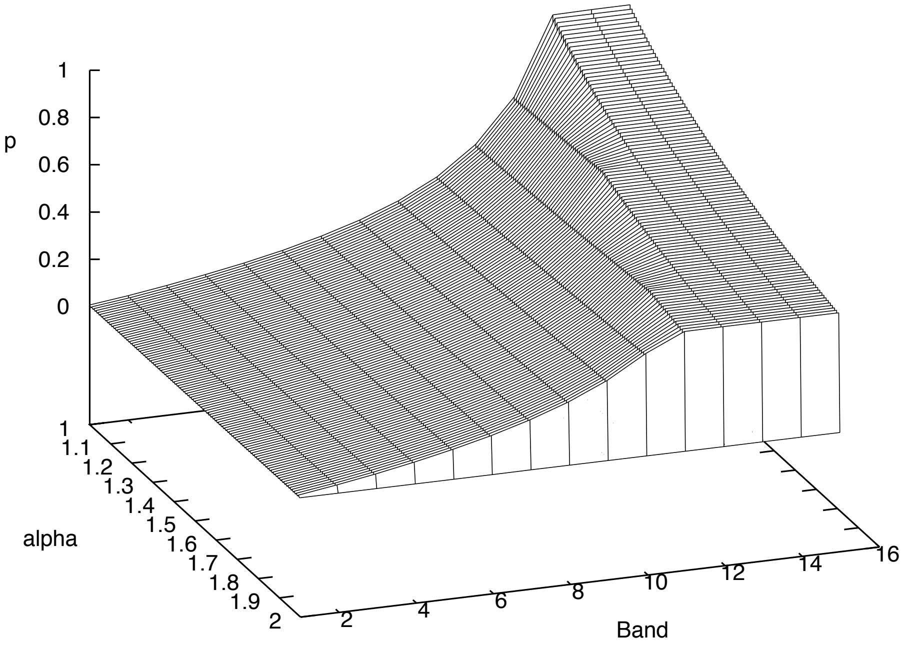
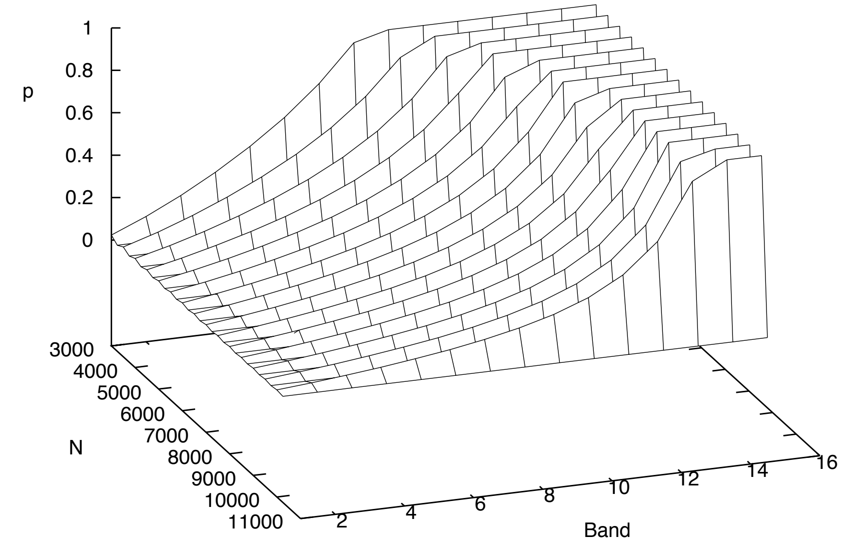


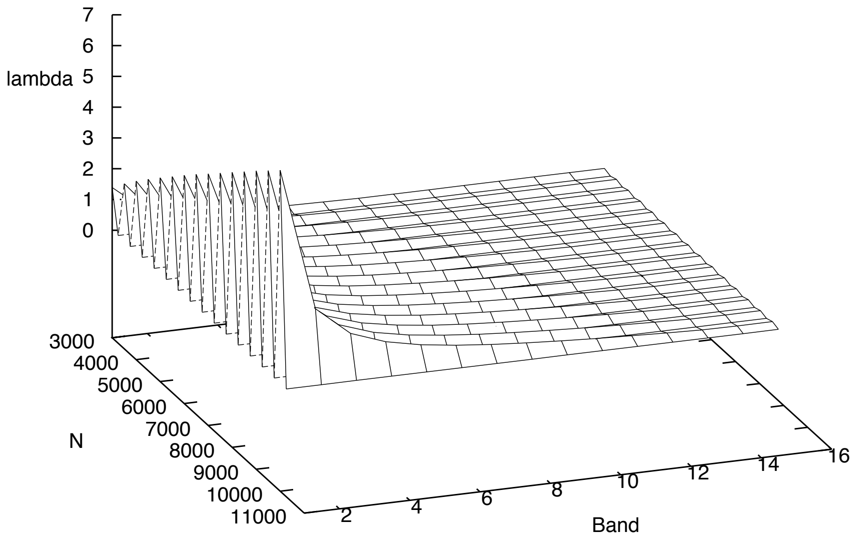
Figure 3.6: As the coverage constraint (parameterized by $\alpha_{\text{cov}}$) increases, the percent of nodes that are designated to be CHs in the outer bands decreases.

CH requirement while the remaining 20% provide enough nodes for coverage (we assume that CHs can sense the environment as well). If we increase the density of nodes required for coverage by increasing $\alpha_{\text{cov}}$, the value for $p$ in the outer bands will decrease (see Figure 3.6).

As the total number of nodes increases, the Dynamic deployment strategy will increase the density of nodes near the base station until the expected lifetimes of these bands are close to those of the middle bands. Then additional nodes will be added to the middle bands to maintain sub-balanced energy consumption. As more nodes are added to the outer bands, the probability of being a CH is decreased (the CH probability is always kept as close as possible to the value required by the connectivity constraint). This process is documented in Figure 3.7; the $x$ and $y$ axes are the band number and the number of nodes available for deployment. The $z$-axis



(a) *Values of $p$ across bands as the number of deployed nodes $N$ increases.* As more nodes are added to the outer bands, the probability of being a CH is decreased.



(b) *Values of $\lambda$ across bands as the number of deployed nodes $N$ increases.* The density of nodes near the base station increases until the expected lifetimes of these bands are close to those of the middle bands. Then additional nodes are added to the middle bands to maintain sub-balanced energy consumption.

Figure 3.7: Samples of optimal values of $\mathbf{p}$ and $\boldsymbol{\lambda}$ as the number of deployed nodes increases.

in Figure 3.7(a) shows the value of $p$ in each band as the total number of nodes grows and Figure 3.7(b) shows the selected densities ($\lambda_j$) for all bands $j = 1 \ldots 15$.

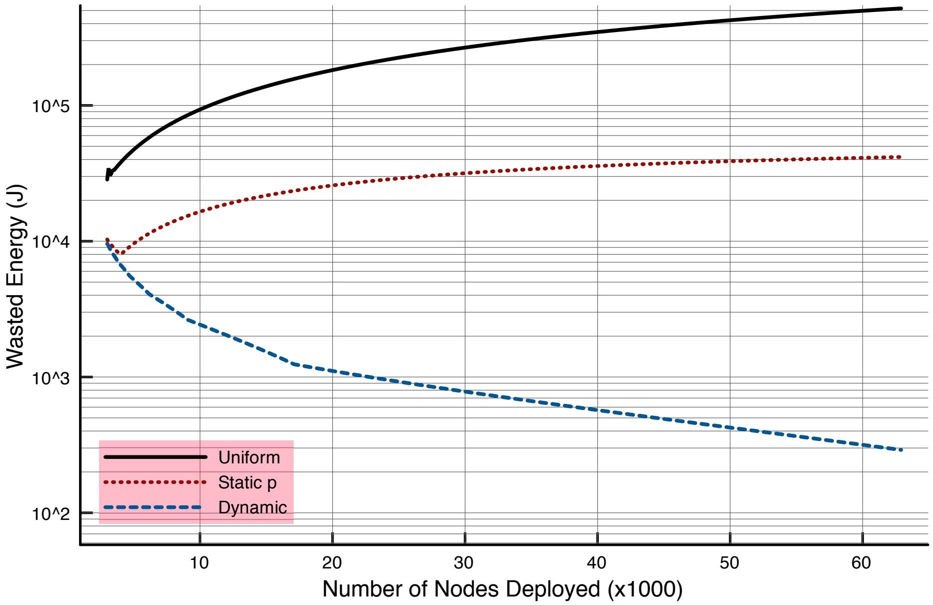


Figure 3.8: In the Dynamic $p$ deployment, the amount of wasted energy decreases as more nodes are added to the network.

Although the Dynamic deployment has an expected lifetime that is always greater than that of the Static $p$ deployment, the lifetimes appear to scale similarly with both methods. However, the Dynamic deployment appears to have an advantage over the Static $p$ deployment with respect to the expected residual energy in the network. Figure 3.8 shows the energy remaining in the network after the first band fails. The amount of wasted energy is plotted logarithmically in order to show the general trend as the initial number of nodes to be deployed grows. The figure shows that the amount of wasted energy in the network for both the Static $p$ and Uniform deployments *grows* as the number of nodes increases, while for the Dynamic deployment, the residual energy *tends to zero for an increasing number of nodes*. The plot suggests that the Dynamic deployment is able to balance more effectively the amount of energy per unit area consumed, or *energy density* for each band. Thus, the expected lifetimes for all bands will tend to converge as the number of nodes $N$ increases. The difference in the ability to balance the energy consumption between Static $p$ and Dynamic deployments is evident from Figures 3.9 and 3.10. In Fig 3.9, the expected lifetimes for each of the outer bands continue to grow as $N$ increases (we will revisit this result in the next chapter), indicating that more energy will be wasted when the first band (and thus the network) fails. In the Dynamic deployment, the expected lifetimes for bands far from the base station remain constant until they converge onto the maximum network lifetime.

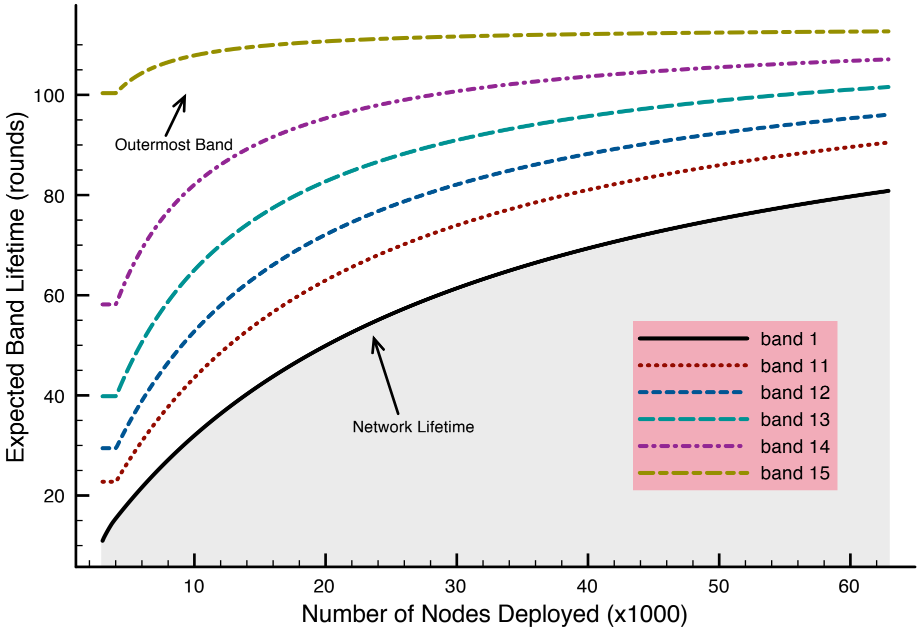


Figure 3.9: The expected lifetime for each band in the Static $p$ deployment (band 1 includes the nodes nearest the base station).

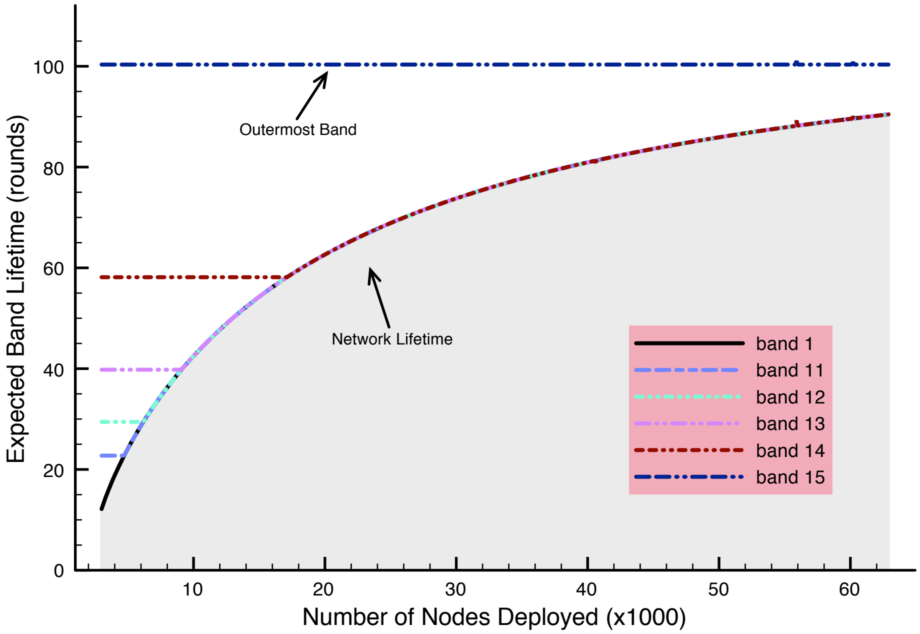


Figure 3.10: The expected lifetime for each band in the Dynamic $p$ deployment (band 1 includes the nodes nearest the base station).

## 3.5 WSN Lifetime Simulation With Optimal Deployment

The optimization problem in Section 3.3.3 was solved for increasing numbers of deployed nodes and the results were used as parameters in a simulation. In the simulated network, the lifetime of the WSN was defined as the first iteration when less than 70% of messages reached the base station. The parameters of the simulation were the same as used in the optimization problem (see Table 3.1). The simulations were performed using *MATLAB* for each of the three variations over initial node inventories of 1500 - 2500 and 6 bands; each data point represents an average of the results from 10 different random deployments. For each experiment, nodes were deployed according to PPPs with intensities obtained by solving the optimization problem; an example simulation setup is shown in Figure 3.11. At the start of each period, each node decides to become a CH according to the LEACH protocol [Heinzelman

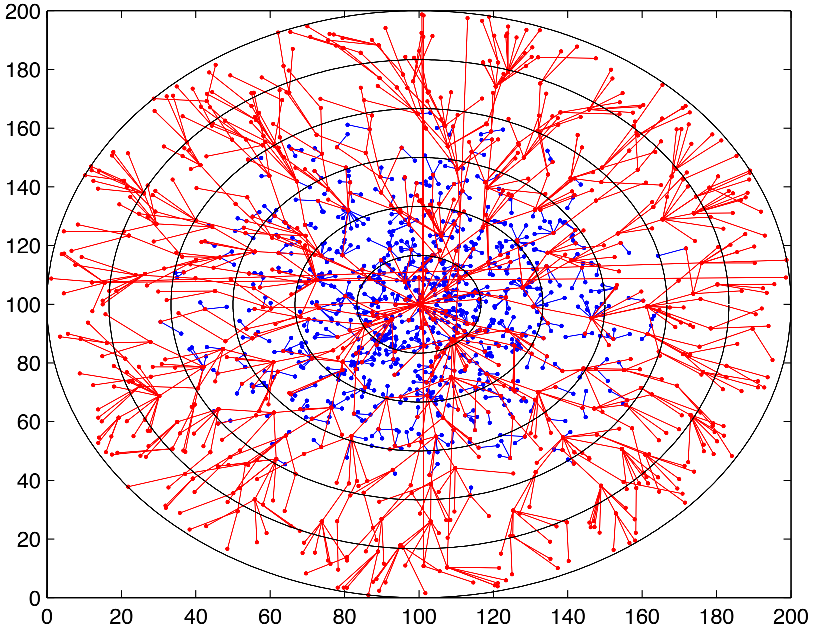


Figure 3.11: *Example simulation topology.* The lines indicate potential paths, the red dots are NCHs, and the blue dots are CHs. This is an example of a Dynamic deployment where the inner bands are using a clustered organization while the outer bands are using a flat organization.

et al. 2000] , announces its status, and links with nearby nodes. The CHs also form links with other CHs in range. If a CH does not have any neighboring CHs, it will force one of its neighbors to perform to become a CH. Then, a Bellman-Ford shortest path algorithm [Bellman 1956] is applied to the adjacency matrix containing distance metrics so that each CH can find paths from itself to the base station (we assume shortest path routing). Once the paths are formed, each node sends 10 messages to its CH and each CH forwards 10 messages along a path to the BSt. Energy levels are tracked for each node, and if a node's energy falls below a minimum energy level, it is removed from the network and the adjacency matrix is updated. This process continues as long as more than 30% of CH messages are received at the base station (BSt).

Table 3.1: Simulation parameters

| Parameter | Value |
|---|---|
| Initial energy for each node | 10 J |
| Transmit cost ($E_{tx}$) | 180 nJ/bit |
| Receive cost ($E_{rx}$) | 200 nJ/bit |
| Length of each packet | 400 bits |
| Radius of $\mathcal{A}$ | 60m |
| Communication radius ($r_{tx}$) | 10m |
| Width of each band | 10m |

### 3.5.1 Simulations

The simulated lifetimes for each of the variations are shown in Figure 3.12(a); numerical calculations of the expected lifetimes for the same network setup are shown in Figure 3.12(b). The simulated network lifetimes are close to the expected lifetimes (comparing Figures 3.12(a) and 3.12(b)). In addition, the relative lifetimes for the different variations are very close to those from the expected lifetimes computed in Section 3.4. Variations in the expected lifetimes for individual tests were due to two factors: (1) the paths taken from CHs to the base station were not always direct, and (2) the number of NCH nodes associated with a CH varied from the average $((1-p)/p)$. These deviations are due our use of expected values for point processes that are derived from asymptotic results. Nevertheless, the variations appear to have a negligible effect with respect to expected lifetimes in simulation.

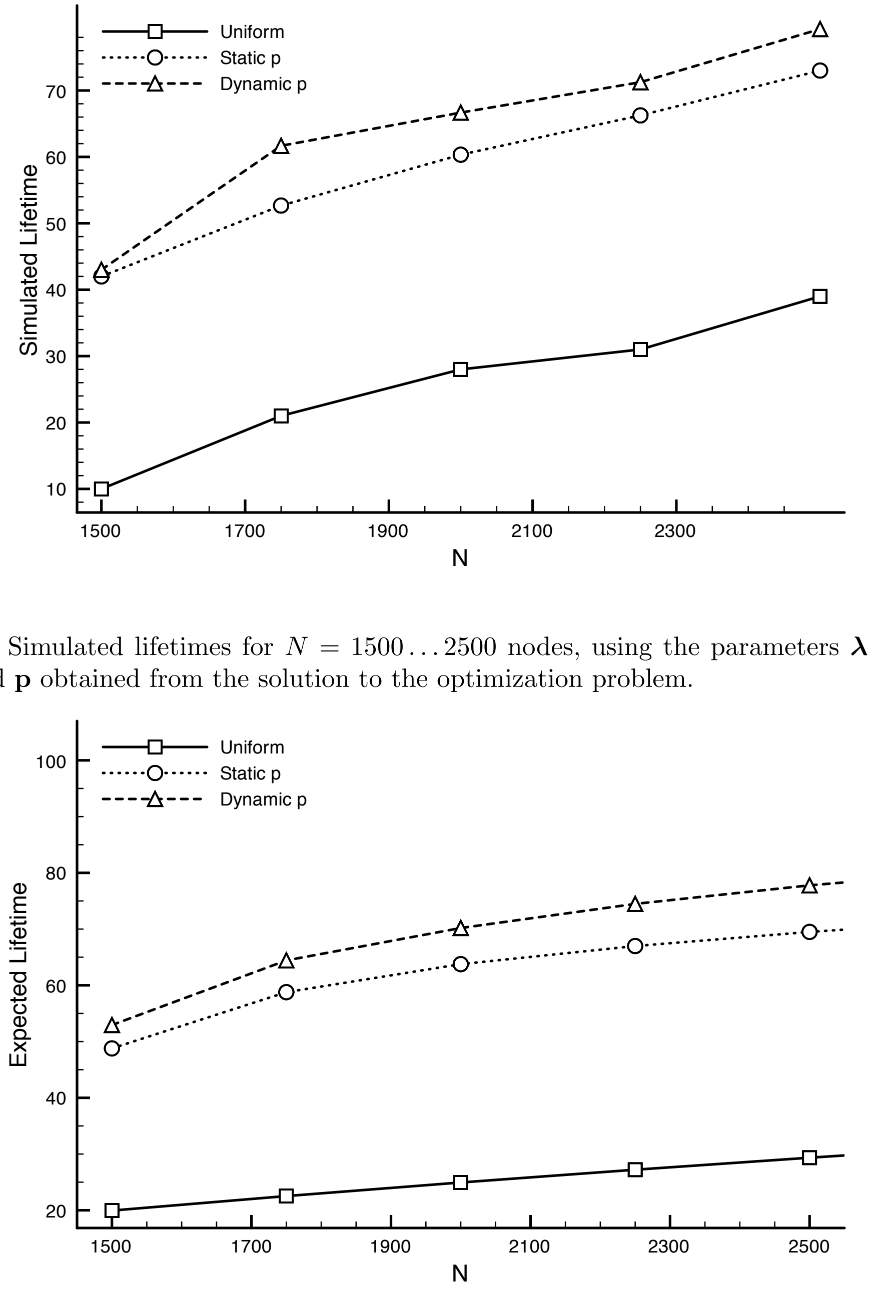
In Section 1.1 we described the effect of biased energy consumption in many-to-one wireless sensor networks over multiple hops. Figure 1.4 depicted the nature of energy consumption (in terms of energy density) closer to the BSt and showed that we could expect large amounts of wasted energy after the first band is depleted of energy and is no longer able to forward messages. Figure 3.14 shows that the simulation values for the energy densities in a Uniform deployment exhibit the same behavior (the bold line indicated the energy density of the innermost band). The energy density for the first band begins much higher in the Dynamic deployment than the rest of the bands (Figure 3.13). Therefore, although the energy density is this band is depleting at a faster rate than any other band, it has sufficient reserves to remain active until other bands deplete their energy.

## 3.6 Observations on the Deployment Strategies

The optimization problem and the model for energy consumption for the Dynamic deployment allowed nodes in different bands to have different values of $p$ and $\lambda$. Currently, clustering algorithms program a value of $p$ for all nodes that is determined before deployment. This value of $p$ is set to ensure that there are enough CHs to provide a connected network and to minimize the amount of energy consumed in the process of monitoring the area and delivering messages to the base station. We studied a strategy where each node is programmed with a look-up table of $p$ values that correspond to distance from the base station. By adding an additional degree of freedom in the network design, we are able to extend the lifetime of the network and reduce the wasted energy in the network at failure. The advantage to this approach in planning and deploying large WSNs is that, with no additional cost to a traditional clustering protocol, and no additional hardware, our proposed strategy will increase the lifetime of a WSN deployment over a Uniform deployment.

One of the few studies analyzing the advantages of using a flat vs. hierarchical network organization for many-to-one WSNs was written by [Duarte-Melo and Liu 2003]. It provides evidence that small networks would benefit from the use of clusters. For larger networks, they argue that it is better to use a flat network. The reasoning for this observation is that in large networks, many nodes need to be deployed in order to cover the area in the outer regions, so organizing clusters may be a waste of energy since the size of the clusters will be small. However, the flat network will result in poor capacity since all of the nodes near the BSt will be competing for the shared channel to transmit their data (as discussed in Sec. 1.3.5). There seems to exist a trade-off between capacity and energy consumption when choosing between a flat network or a hierarchical network. Viewing the Dynamic deployment strategy computed in Sec. 3.4 with this trade-off in mind, however, we see that the Dynamic deployment provides a strategy that balances this trade-off by providing a *hybrid* approach (an example of a hybrid network obtained from Dynamic deployment was shown in Figure 3.11).

If the objective for a large network is to cover the region while minimizing communications (energy) and maximizing capacity (by minimizing messages to the base station), then an appropriate strategy would be to use a hierarchical organization near the base station and to gradually reduce the number of clusters moving toward the outer bands. This is the approach that the Dynamic deployment is providing. In Figure 3.5, the value of $p$ for the outer bands is near 80%. That is, only 20% of the nodes in the outer bands are NCHs. However, if most CHs do not have NCHs associated with them, then the outer bands are essentially flat networks with few NCHs to help meet the coverage requirements. If the coverage parameter $\alpha_{\text{cov}}$ is equal to 1, meaning that the coverage and connectivity constraints are the same, all nodes in the outer bands are CHs. Figure 3.5 also shows that the value of $p$ close to the base station is small (approx. 2%), which means that very few nodes will be competing to send data to the base station. This value of $p$ for the innermost band is close to the optimal value for CH probability computed in [Heinzelman et al. 2002] for a Uniform deployment.



(a) Simulated lifetimes for $N = 1500 \ldots 2500$ nodes, using the parameters $\boldsymbol{\lambda}$ and $\mathbf{p}$ obtained from the solution to the optimization problem.

(b) Expected lifetimes for $N = 1500 \ldots 2500$ nodes given by the solution to the optimization problem.

Figure 3.12: Comparison of the simulated lifetimes and the expected lifetimes for $N = 1500 \ldots 2500$ nodes. The three lines represent the lifetimes for the Uniform, Static $p$, and Dynamic deployments.

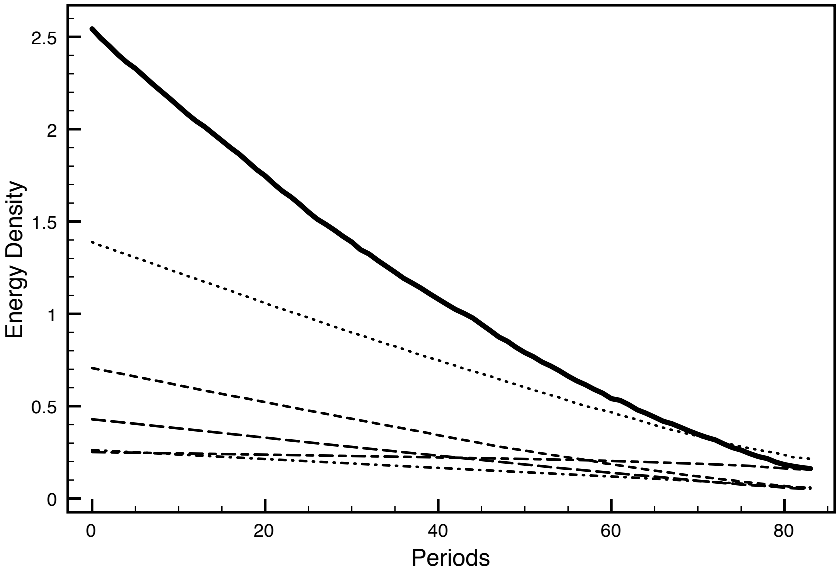


Figure 3.13: Simulated energy densities for the Dynamic deployment; the bold line is the inner band.

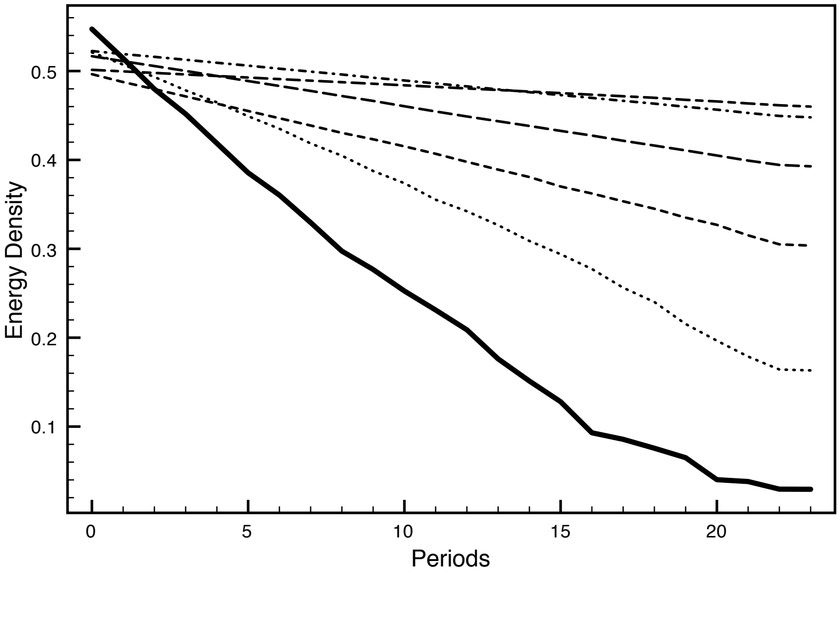


Figure 3.14: Simulated energy densities for each band in a Uniform deployment; the bold line is the inner band.

# 4. Non-Uniform Deployment in Clustered WSNs – Part II

## 4.1 Introduction

In the previous chapter, we described the network model and constraints for an optimal deployment of sensor nodes for a monitoring network. We derived network constraints, and expressions for energy consumption and residual energy using some simplifying assumptions. We compared the optimal solutions for three deployment strategies: Uniform deployment, Static $p$ deployment, and Dynamic deployment. We computed expected lifetimes using the the corresponding optimization problems and showed that the best performance, in terms of expected lifetime and residual energy at the time of network failure, was achieved under the Dynamic deployment strategy. Simulations verified the optimization results.

In this chapter, we remove some of the simplifying assumptions, reformulate the optimization problem for the more realistic model, update some the numerical results, and discuss the consequences. Expressions for the cost of intra-cluster communication are extended to include the costs associated with setting up clusters and the energy consumed in each cluster on exchanging data with the clusterhead (CH). These extensions involve specifying a MAC scheme for intra-cluster communication and re-writing the energy equations to include the effects of cluster size. In the previous chapter, we relied on the expected number of non-clusterheads (NCHs) associated with each CH to compute the expected cost of intra-cluster communication. In this chapter we include the variance of the cluster sizes in order to account for the overhead introduced by large clusters. We also remove the assumption that each CH is communicating with a CH in the neighboring band each time data are forwarded toward the sink node. Thus, the inter-cluster communication overhead will include the costs of additional hops that may be required to cross a band. We analyze the effect of these additional hops, and redesign the deployment strategies.

## 4.2 Network Model

The network model and notation remains the same as the previous chapter (see section 3.2). The probability that a node will be a CH in band $j$ is $p_j$, and $\lambda_j$ is the density of nodes in band $j$. The monitored area is a disk of radius $R$ that is divided into $J$ annular bands of width $r_{tx}$ each, with the base station located at the center of the disk. Nodes are assumed to be distributed in the region according to a set of homogeneous spatial Poisson processes with intensity $\lambda_j$ for the $j$th band. The area of band $j$ is denoted $A_j$.

The number of nodes in band $j$ is a Poisson random variable, $N_j \sim \mathcal{P}(\lambda_j)$. The area of band $j$ is $A_j$ and the expected number of nodes in band $j$ is $\mathbb{E}[N_j] = \lambda_j A_j$, where $A_j = \pi\left(r_j^2 - r_{j-1}^2\right)$. The widths of the bands are chosen to be equal to the communication radius of each node, $r_{tx}$, so that the number of bands, $J$ is $\lceil \frac{R}{r_{tx}} \rceil$

The next two sections derive expressions for the total energy consumed in each round of sensing and reporting in the $jth$ band as a function of $p_j$ and $\lambda_j$. We consider the energy consumed in each band of the network during

A1 **Cluster formation**;

A2 **Intra-cluster communication**: communication between CHs and NCHs;

A3 **Inter-cluster communication**: communication between CHs.

## 4.3 Cluster Formation and Intra-Cluster Energy Consumption

In this section, we derive expressions for A1 and A2. We describe the MAC protocol for the intra-cluster communication and derive a new expression for energy consumption that includes the effects of command messaging, idle periods, and exchanging data between the NCHs and the CH.

### 4.3.1 Media access control (MAC)

The MAC scheme assumed for intra-cluster communication is a hybrid of carrier sense multiple access (CSMA) and time division multiple access (TDMA). The combination of TDMA and CSMA has been shown to reduce energy consumption in WSNs (*e.g.*, the Z-MAC protocol, [Rhee et al. 2008]). CSMA is used between the NCHs and CHs after the initial cluster set-up in order to synchronize to the CH clock and to receive a time slot and frequency channel for sending messages. Once the NCHs have obtained time slots and are synchronized to the CH, the NCHs can avoid contestation within the cluster using the contention-free TDMA MAC. At the end of a round, new CHs are selected and the procedure is repeated.

In the energy consumption derivations for the intra-cluster communication, we distinguish between the energy consumed while transmitting and receiving data and the energy consumed by control messages. We also introduce additional notation for energy spent while the radio is in an idle mode (the radio is sensing the channel but is not receiving nor transmitting data), awaiting access. Let $E_{tx}$ and $E_{rx}$ denote the energy expended to transmit and receive a single data packet, respectively. Let $E_{txc}$ and $E_{rxc}$ represent the energy expended in transmitting and receiving control messages. Finally, let $E_l$ denote the energy expended while the node is in idle for a single time slot.

### 4.3.2 Cluster formation (set-up period)

The intra-cluster MAC protocol consists of three periods: the *set-up period*, the *contention period*, and the *steady-state period*, shown in Figure 4.1. During the set-up

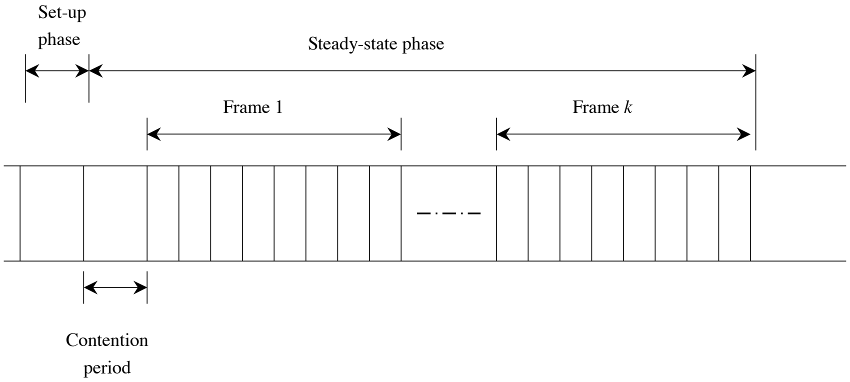


Figure 4.1: Illustration of the separation of the set-up period, the contention period, and the steady-state TDMA period assumed in the intra-cluster MAC scheme.

period, each node decides whether it would become a CH. This determination is based on the node's energy level and the number of times it has served as CH. Elected CHs broadcast an advertisement message to all other nodes, announcing their becoming new CHs. Next, each NCH node joins the cluster whose announcement has the highest received signal strength. The expected amount of energy used in band $j$ during cluster formation is the energy consumed as each of the $C_j$ CHs broadcast their announcement and each of the $S_j$ nodes surrounding each CH responds to the advertisement. Each CH receives $S_j$ responses and transmits once and each NCH transmits once, so the energy consumed is $C_jS_jE_{rxc} + C_jE_{txc} + C_jS_jE_{txc} =$

$$\mathrm{C}_j\left(\mathrm{E}_{txc} + \mathrm{S}_j\mathrm{E}_{rxc} + \mathrm{S}_j\mathrm{E}_{txc}\right). \tag{4.1}$$

Once the clusters are declared, the system enters into the *steady-state* period.

### 4.3.3 Steady-state period and the Contention Period

The steady-state period begins with the initial contention period, where NCHs communicate with the CH using nonpersistent CSMA to obtain a time slot (the time slots are assigned to the NCHs by the CH during the contention period). The choice of nonpersistent CSMA is motivated by the fact that, although it may incur high delays, it is very efficient [Bruno et al. 2002].

The steady-state period is divided into a *contention period* and *frames*. The duration of each frame is fixed. During the contention period, all nodes keep their radios on. The CH builds a TDMA schedule and broadcasts it to all nodes within the cluster. There is one data slot allocated to each node in each frame. Each source node turns its radio on and sends its data to the CH over its allocated slot-time. It keeps its radio off at all other times. With the basic TDMA scheme, a node always turns its radio on during its assigned time slot whether it has data to transmit or not. If the node has no data to send, the node operates in idle mode, which is an energy-consuming state. When a frame ends, the next frame begins and the procedure is repeated. The CH collects the data from all the source nodes and forwards the aggregated data to the base station. After a predefined time, the system begins the next round and the process is repeated.

### 4.3.4 Energy consumption During the Contention Period

During the contention period, the communication between the CH and all other nodes is accomplished by using non-persistent CSMA. Suppose $\eta$ is the throughput of non-persistent CSMA when there are $S_j$ attempts to access the channel per packet time[1]. The energy consumption by a single node during the contention period is

$$\frac{1}{\eta}E_{txc} + \frac{S_j - 1}{\eta}E_l + E_{rxc}. \tag{4.2}$$

[1] $\eta$ is the maximum throughput for the nonpersistent CSMA used in the contention period of the TDMA scheduling. We shall use the capacity derived by [Kleinrock and Tobagi 1975], $\eta = 0.815$.

The CH node receives control packets and dissipates energy in the amount

$$S_j E_{rxc} + E_{txc}. \tag{4.3}$$

Therefore the average total energy consumed due to contention at each cluster is

$$\begin{aligned} & \left[\frac{1}{\eta} E_{txc} + \frac{S_j - 1}{\eta} E_l + E_{rxc}\right] + S_j E_{rxc} + E_{txc} \\ = \quad & S_j \frac{1}{\eta} E_{txc} + E_{txc} + \frac{S_j (S_j - 1)}{\eta} E_l + 2 S_j E_{rxc}. \end{aligned} \tag{4.4}$$

### 4.3.5 Total energy consumption for intra-cluster set-up and communication

A round consists of $K$ sessions or frames. Energy consumed exchanging one frame from each NCH in all clusters in a band is $C_j S_j \left(E_{rx} + E_{tx}\right)$, and this occurs $K$ times per round, so the energy consumed in a band due to frame exchanges in all clusters is:

$$K C_j S_j \left(E_{rx} + E_{tx}\right). \tag{4.5}$$

The energy consumed due to initial broadcast for cluster group decisions is

$$C_j \left(E_{txc} + S_j E_{rxc} + S_j E_{txc}\right) \tag{4.6}$$

The energy consumed per band due to the contention period preceding delivery of TDMA slots is

$$C_j \left(S_j \left(\frac{E_{txc}}{\eta} + \frac{\left(S_j - 1\right) E_l}{\eta} + E_{rxc}\right) + S_j E_{rxc} + E_{txc}\right) \tag{4.7}$$

Summing all of these expressions, the energy consumed in each band due to intra-cluster communication is

$$C_j\left(E_{txc}+S_jE_{rxc}+S_jE_{txc}\right)+C_j\left(S_j\left(\frac{E_{txc}}{\eta}+\frac{\left(S_j-1\right)E_l}{\eta}+E_{rxc}\right)+S_jE_{rxc}+E_{txc}\right)$$
$$+KC_jS_j\left(E_{rx}+E_{tx}\right). \tag{4.8}$$

After substitutions and collecting terms according to the different types of energy consumed (transmit and receive control, idle, transmit and receive data), we can write the energy consumed during cluster set-up and organization, and after $K$ rounds of data exchange in a single cluster as

$$\begin{aligned} E_{\mathrm{intra}} = &-\frac{\lambda_jA_j\left(-p_j-{p_j}^2\eta+{p_j}^2-p_j\eta\right)E_{txc}}{p_j\eta} \\ &-\frac{\lambda_jA_j\left(-Kp_j\eta+K{p_j}^2\eta\right)E_{tx}}{p_j\eta} \\ &-\frac{\lambda_jA_j\left(-3\,p_j\eta+3\,{p_j}^2\eta\right)E_{rxc}}{p_j\eta} \\ &-\frac{\lambda_jA_j\left(-Kp_j\eta+K{p_j}^2\eta\right)E_{rx}}{p_j\eta} \\ &-\frac{\lambda_jA_j\left(-E_l+3\,E_lp_j-2\,E_l{p_j}^2\right)}{p_j\eta}. \end{aligned} \tag{4.9}$$

## 4.4 Routing protocol

We consider a basic greedy forwarding algorithm, where every node attempts to forward a packet to a node that is both within its transmission range and closer to the sink than itself (closer in the sense that the Euclidean distance between the sink and this target node is smaller). Greedy forwarding tries to bring the message closer to the destination in each step using only local information. Thus, each node forwards the message to the neighbor that is most suitable from a local point of view. The most suitable neighbor can be the one who minimizes the distance to the destination in each step. Since all CHs are attempting to forward data to the same destination, these routes are formed by creating a routing tree rooted a the sink node.

## 4.5 Sources of Uncertainty in the Energy Consumption Model

In Chapter 3, we stated the assumption that each CH communicates directly with a CH in the next band. We also based the energy consumption model for intra-cluster communications on the expected number of of NCHs associated with each CH using our derivation from homogeneous PPPs. In reality, the number of associated NCHs will have significant variance due to the fact that our network is not truly a homogeneous PPP, but a collection of annular bands with different intensities. Also, the number of hops that will be required for a message to reach the sink from band $j$ will not, in general, be exactly $j$. In this section, we examine the distributions of these factors and attempt to introduce them into the optimization problem in order to provide more a more realistic problem statement.

### 4.5.1 Cluster size

The expected value for the number of NCHs in a cluster in band $j$ is $\frac{1-p_j}{p_j}$. The distribution of cluster sizes is Poisson, so the variance is equal to the mean. In simulations, the variance of $\frac{1-p_j}{p_j}$ appears to be a valid approximation for the different cluster sizes. Figure 4.2 shows a typical deployment over 6 bands. The dark lines indicate the value $\frac{1-p_j}{p_j}$ for the band and the red points indicate the actual number of nodes in a cluster with respect to distance from the sink node.

### 4.5.2 Hop-count statistics

A significant source of uncertainty in the energy consumption model is the hop-count distribution. If we assume that each CH in a band communicates with exactly

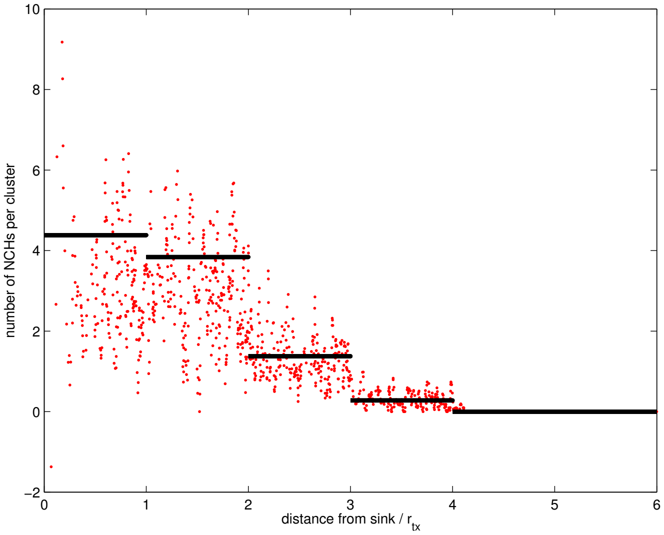


Figure 4.2: The number of NCHs associated with a cluster are shown by the red points. The dark lines indicate the expected value for cluster sizes in the band. The variances of the points appear to be approximately in line with the expected value.

one CH in the next band toward the sink, then the mean energy consumed by nodes in a particular band can be computed by summing the number of CH in each of the bands that are farther from the sink node. However, without enforcing such communications between bands, we cannot ensure that this count would provide a good approximation of the energy consumed in an area. As an illustration, Figure 4.3 shows the distributions for the number of hops to reach the sink node with respect to the distance from the sink node for the Dynamic deployment strategy. For example, the figure shows that, for a 6-band network, the maximum number of hops is as high as 8, and the distribution of hop counts is not the same for each band. This suggests that some bands will be consuming more energy than others while relaying messages across the band.

In order to see how the CH density in two adjacent bands can affect the hop count distribution, we perform a simple experiment. We simulate a two-band network and vary the density of CHs over the second band while holding the first band's density

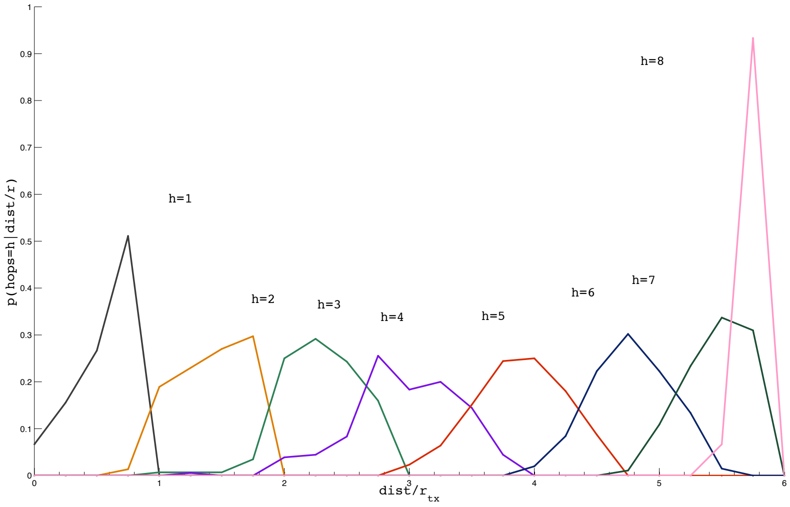
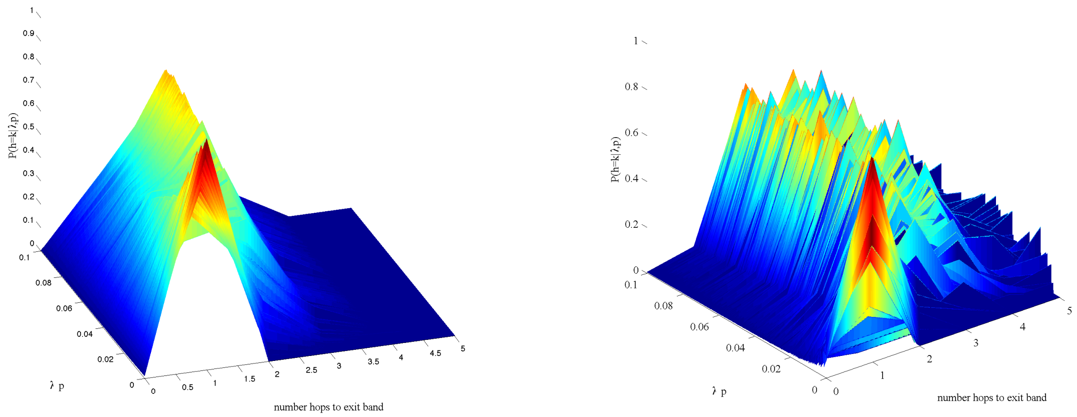


Figure 4.3: Simulation results illustrate the difficulties of the assumption that a message originating from a node in band $k$ travel over $k$ hops to reach the destination. The $x$-axis denotes the distance from the sink node divided by the annular band width. The $y$ axis shows the probability of a specific hop count to the sink, given that the message originates at a particular distance from the sink node. Eight (8) conditional distributions are shown (one for each hop count).

constant. Figure 4.4(a) shows the distribution of the number of hops required to reach a node in the next band when the CH density is fixed in the first band to be 0.1. The y-axis represents the density of CHs in the second band, varying from 0.01 to 0.1. Here we notice that, for CH densities that are lower (but high enough to provide connectivity), the distribution of hop counts peaks at a single hop, but quickly moves to a peak at 2 hops as the density increases. Figure 4.4(b) shows the same simulation, except the density of CHs in the first band is set to 0.01. Here we see that, as the density in the second band increases, the distribution of the number of hops required to exit the band spreads to include peaks at 2 and 3 hops.

In order to make an approximation of the energy consumed due to forwarding messages back to the sink, we require some statistics about the hop-count distribution on the WSN that we can include in the optimization problem. Specifically, we require a statistic related to the probability $P(hc = k|j)$, the probability that a CH in band $j$ is $hc$ hops from the nearest CH in band $j+1$, for all $j \in 1, \ldots, J-1$. The general problem of deriving hop-count distributions, even for completely homogeneous Poisson point process models, is difficult, due to the involved spatial dependence problem (see, for example, Rahmatollahi and Abreu [2012] and Zhang et al. [2012]). The spatial dependence problem arises because the event that a randomly chosen destination node is a $k$th hop node from a randomly chosen source node is not independent of the event that another randomly chosen node is a $i$th hop node for $1 < i < k$.

Denote by $A(x, R)$ the intersectional area of two disks, each with radius $R$, with a distance $x$ between them. This is called the *circle-circle intersection* (see Figure 4.5). The area of the intersection for equal sized circles is



(a) The fixed density of CHs in band 1 is 0.1

(b) Using a lower CH density in the first band (0.01)

Figure 4.4: Illustration of the affect of CH density on the hop count distribution. Distributions for hop counts for varying $p$ and $\lambda$ in band 2 and a fixed density for band 1. The $x$ axis indicates the number of hops required to reach the next band. The $y$ axis shows the density of CHs in the second band. The $z$ axis shows the values of the conditional distribution $P(hc = k|\lambda_2 p_2)$. The left plot shows the hop count distribution for a dense deployment; the right plot is a sparse (but connected) deployment.

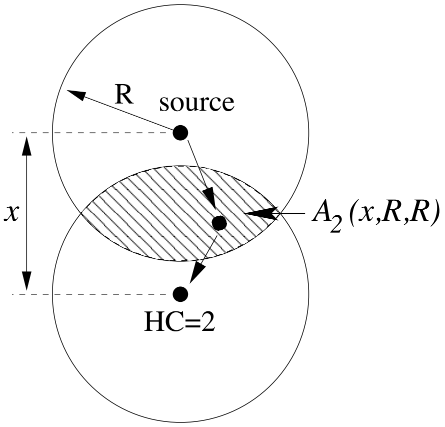


Figure 4.5: Illustration of the intersection of two disks separated by a distance $x$, used to compute the 2-hop distribution.

$$2\,R^2 \arccos\left(1/2\,\frac{x}{R}\right) - 1/2\,\sqrt{(4\,R^2 - x^2)\,x^2}. \tag{4.10}$$

If $x < R$, then $P(hc = 1|x) = 1$. When $R < x \leq 2R$, the probability that the hop count is 2 is given by $P(hc = 2|x) = 1 - e^{-\lambda A(x,R)}(\lambda A(x,R))^2/2!$. This expression gives the probability that the intersection (area shaded in Figure 4.6) contains at least one node. However, for hop counts greater than 2, the computation becomes very complicated, and closed-form solutions are not available for hop counts above 3 without making additional independence assumptions.

If we are using a greedy forwarding routing algorithm, then the problem of estimating the hop-count distribution simplifies somewhat. The simplification is realized since we are only interested in the number of hops required to forward a message from band $j$ to band $j-1$. As the bands are defined to have width equal to the transmission radius, and as the CH densities are selected to ensure that there is a neighbor within the band, we are only interested in the probability of a CH in one band having a reachable destination in the next band toward the sink node. In order to obtain the probability of there being a next hop neighbor in the next band, we

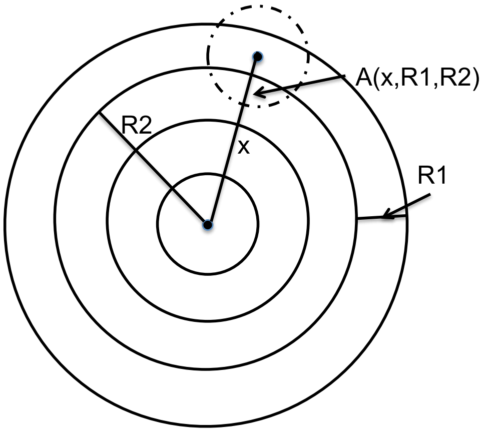


Figure 4.6: The dashed circle is the transmission radius of a node in the outer band. The area of intersection with the next band toward the sink node is computed using Eq. 4.11.

first need the area of intersection between the circle around a random node and the next band. We can compute the area of intersection using the circle-circle intersection formula for circles of different radii. Letting $r_1 = r_{tx}$, $r_2$ be the radius of the outer edge of the $j-1$ band, and $x$ the distance between a node in band $j$ and the sink node (see Figure 4.6), the area of intersection is given by

$$\begin{aligned} A_{cc}(x, r_1, r_2) = {r_1}^2 \arccos\left(1/2\,\frac{x^2+{r_1}^2-{r_2}^2}{x r_1}\right) + {r_2}^2 \arccos\left(1/2\,\frac{x^2+{r_2}^2-{r_1}^2}{x r_2}\right) \\ -1/2\sqrt{\left((r_1+r_2)^2-x^2\right)\left(x^2-(r_1-r_2)^2\right)}\,. \end{aligned} \tag{4.11}$$

We let $x_j = x - (j-1)r_{tx} = x - r_2$; this is the distance between a node in band $j$ and the outer edge of band $j-1$. Then denote the intersection of the transmit region of a node in band $j$ and band $j-1$ as $A_{ccj}(x_j, r_1, r_2)$. Then, the probability that the number of hops for a node in band $j$, a distance $x_j$ from the outer edge of band $j-1$ will be greater than one is

$$P(hc > 1|x_j) = \left( \frac{A_{j-1} - A_{ccj}(x_j, r_1, r_2)}{A_{j-1}} \right)^{C_{j-1}}. \tag{4.12}$$

We denote the area of an annular band inside of band $j$ whose inner edge is a distance $x_j$ from the outer edge of band $j$ and has width $\mathrm{d}x_j$ as $A_{\mathrm{d}x_j}$. Then,

$$P(x_j|j) = 1 - \left( \frac{A_j - A_{\mathrm{d}x_j}}{A_j} \right). \tag{4.13}$$

The probability of a node in band $j$ having a hop count greater than one to reach $j-1$ is

$$P(hc > 1|j) = \int_{x_j=0}^{x_j=r_{tx}} P(hc > 1|x_j)P(x_j|j)\mathrm{d}x_j. \tag{4.14}$$

By discretizing the width of the small annular bands and replacing the integral 4.14 with a summation, we can estimate the number of nodes in band $j$ that will require more than a single hop to reach the next band as $\lambda_j p_j A_j P(hc > 1|j) = C_j P(hc > 1|j)$. This expression is included in the energy cost for the inter-cluster communication in the optimization problem. As an example, Figure 4.7 shows the edges of the CH network in red for a sample deployment. The values of $\lambda$ and $p$ are $\lambda = [0.417, 0.198, 0.104, 0.059, 0.029, 0.029]$ and $p = [0.110, 0.122, 0.249, 0.463, 0.967, 1.0]$. The expected number of nodes that will require more than a single hop to exit the band is computed to be [0, 11.4, 25.5, 33.4, 41.2, 49.6]

We performed the same experiment that produced Figures 4.4(a) and 4.4(b), and used the discretized expression for the total probability in Eq. 4.14. We computed the expected number of nodes that have a hop count greater than one in band 2, given the density of CHs in band 1. The results are shown in Figure 4.8. Figure 4.8(a) shows the prediction when the density of CHs is fixed to 0.1 in band 1, and Figure 4.8(b) shows the prediction for the sparser density of 0.01 in band 1. The black lines are comprised of point estimates for each CH density value in band 2. From these

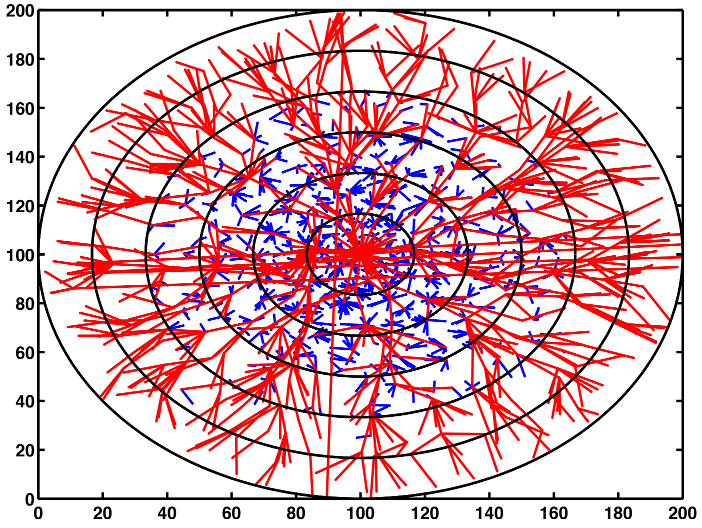
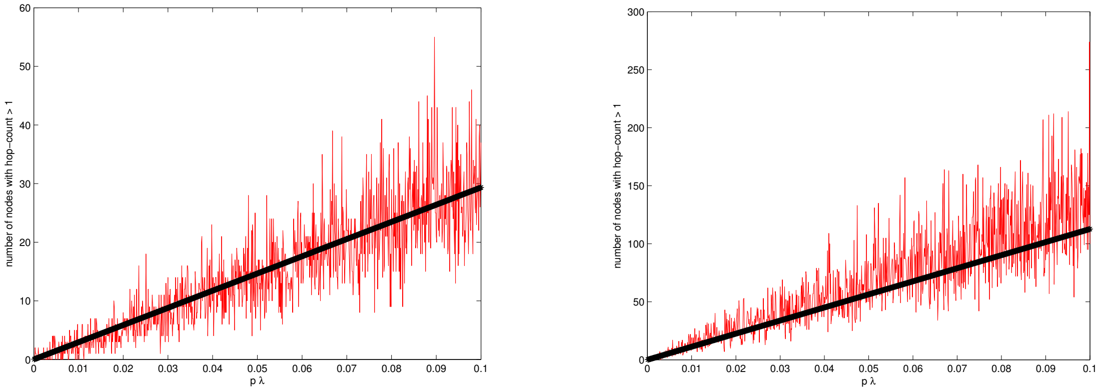


Figure 4.7: Example deployment with $\lambda$ =[0.417 , 0.198, 0.104, 0.059, 0.029, 0.029] and $p = [0.110, 0.122, 0.249, 0.463, 0.967, 1.0]$.

figures we see that although there is significant variance in the number of nodes with additional hops requirements to leave the band, our prediction follows the trend well.



(a) The fixed density of CHs in band 1 is 0.1

(b) The fixed density of CHs in band 1 is 0.01

Figure 4.8: Predictions for the number of nodes with hop counts greater than one compared to actual number for varying $p$ and $\lambda$ in band 2 and a fixed density for band 1. The black lines are the model predictions and the red lines are the actual values from simulation.

## 4.6 Energy Consumption for Inter-cluster Communication

In this section, we state the expression for A3 of section 4.2, the energy consumed while performing inter-cluster communication. The expression for the energy consumed while relaying messages forward to the sink node through the CH overlay (Eq. 3.7) is

$$K\mathrm{E}_{rcv}\sum_{k=j+1}^{J}\mathrm{C}_k + K\mathrm{E}_{tx}\sum_{k=j}^{J}\mathrm{C}_k\ .$$

This expression counts one message for each of the expected number of CHs in each band as they are received and forwarded along toward the sink. We now include the term $C_jP(hc > 1|\lambda_j, p_j)$ to account for the expected amount of energy due to the density of CHs in adjacent bands. Then, the updated expression for the inter-cluster communication costs is

$$E_{\text{inter}} = KE_{rx}\sum_{k=j+1}^{J}C_k(1 + P(hc > 1|k) + KE_{tx}\sum_{k=j}^{J}C_k(1 + P(hc > 1|k). \qquad (4.15)$$

## 4.7 Maximum Lifetime Optimization Problem with Updated Energy Model

The formulation of the optimization problem is similar to the problem stated in Section 3.3.3, with the following additions. First, the energy consumption formula used to evaluate the expected lifetime $\mathbb{E}[\mathcal{L}_j]$ now include the energy consumed during intra-cluster communication, $E_{\text{intra}}$ (Eq. 4.9). These additions account for the amount of energy consumed in a cluster with respect to the size of the cluster. Control messages, idle time spent waiting to exchange data with the CH messages, and the amount of data shared with the CH all contribute to the intra-cluster energy expenditures. This addition to the energy model is important because it factors how the energy consumed due to congestion in large clusters will affect the expected network lifetime. The expected lifetime formulation in the previous chapter did not account for this effect. For the same reason, we also include the variance of the cluster sizes, instead of just the expected number of NCHs in a cluster, when computing the intra-cluster energy consumption. Also, the expected lifetime formulation now includes the expected amount of energy that will be consumed due to the number of hops required to relay messages to the next band (Eq. 4.15). The expected lifetime formulation in the previous chapter assumed that a source node in band $j$ could reach a destination node in band $j-1$ in a single hop. This addition is important in the expected lifetime formula because it accounts for the effect that choices of $\lambda$ and $p$ in adjacent bands will have an effect on the expected number of hops a message will require to exit a band. The computation of the expected remaining energy, $\mathbb{E}[E_w^j]$ is also updated with the new energy consumption terms. The constraints on the problem remain exactly the same as before, and the objective is still to find the values of $p$ and $\lambda$ that maximize the minimum expected lifetime over all bands:

$$\max_{p,\lambda} \ z \tag{4.16}$$

*subject to*

$$\begin{aligned} \mathcal{L}_j - z &\geq 0 \\ \lambda_j p_j &\geq \lambda_{\text{con}} \\ \lambda_j &\geq \alpha_{\text{con}} \lambda_{\text{con}} \\ 0 \geq p_j &\leq 1 \\ \lambda_j &\geq 0. \end{aligned}$$

### 4.7.1 Examples of expected lifetimes with the updated model

Again, we compare the three deployment strategies (Uniform, Static $p$, and Dynamic). However, now we remove the node inventory constraint from the problem (we set a an arbitrarily large total inventory) and solve the lifetime objective with a trade-off on the cost (total number of nodes). That is, we let a parameter $w \in [0, 1]$ vary while solving an objective $wz - (1 - w)\sum_J N_j$. The variable $z$ is maximum expected lifetime, and $N_j$ is the number of nodes deployed in band $j$. The results are plotted in Figure 4.9. As expected, it shows that the Static $p$ and the Dynamic deployments perform much better in terms of lifetime for a given cost than the Uniform deployment. While the Dynamic deployment always dominates the Static $p$ deployment for a given cost/lifetime solution, the margin is smaller than the results in the last chapter (see Figure 3.5) where we did not account for intra-cluster contention and the effects of the hop-count distributions. Also note that when the energy consumption due to congestion in the intra-cluster communication was considered in the expected lifetime calculation, the Uniform deployment could no longer increase lifetime beyond a certain point by adding more nodes.

With a better understanding of the way that the density of CHs in a given band affects the hop-counts in the network, the diminishing margins make sense. Increasing the value of $p$ for a band will generally increase the number of nodes that will require additional hops to reach the next band. Figure 4.8 shows that this increase is approximately linear in the density of the CHs, and the linear proportionality constant is determined by the number of available CHs in the next band. In the simulation of the last chapter, the Dynamic deployment did consistently better than the Static $p$ simulations (Figure 3.12(a)), and the numerical solutions for the optimization problem showed better expected performance for the Dynamic deployment. One advantage of the Dynamic deployment (as computed in the optimization problem, Eq. 3.30)

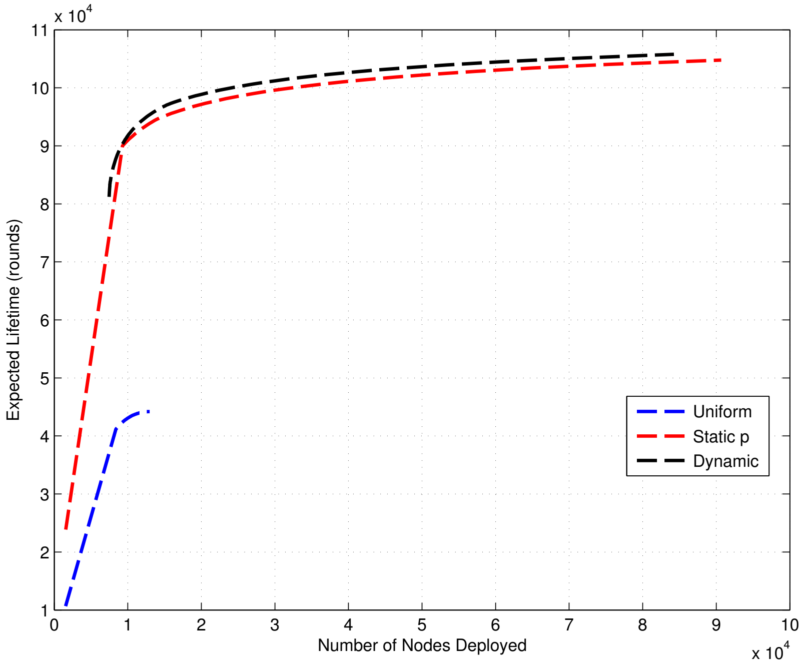


Figure 4.9: The numerical solution of the expected lifetime with respect to the number of nodes deployed for the the three compared approaches.

was its ability to minimize the wasted energy; the numerical solutions in Figure 3.8 showed that as the number of nodes deployed on the network increased, the amount of residual (wasted) energy would continue to decrease.

In Chapter 3, we saw numerical results suggesting that if one could vary both the CH density and the node density, this capability would reduce the amount of residual energy remaining after network failure. However, Figure 4.10 shows that both the Static $p$ and the Dynamic deployment strategies can find a solution that decreases the residual energy as the number of nodes increases. These observations suggest that when we include the energy consumed due to congestion and additional hops to reach the BSt in the lifetime function, the advantages of the Dynamic deployment over the Static $p$ deployment are reduced.

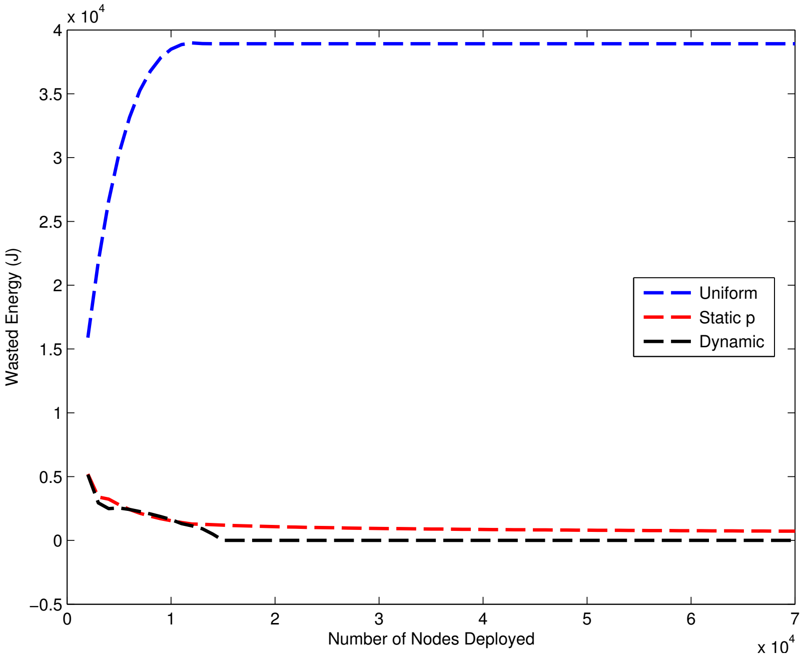


Figure 4.10: Comparison of the residual energy of the optimal deployments for each approach as the number of nodes increases.

# 5. Sensor Node Replenishment

## 5.1 Introduction

So far we have considered the problem of extending the lifetime of the WSN for a given inventory of sensor nodes through strategic placement over the field of deployment. In this chapter, we discuss a replenishment strategy for further extending the lifetime of a network to meet a mission requirement by deploying batches of additional nodes.

A natural question might be: "why not just deploy more nodes at the initial installation of the network?" There are two reasons why a replenishment strategy would be required instead of a larger initial deployment. First, as the calculations and simulations in Sections 3.4, 3.5, and 4.7.1 suggest, the marginal lifetime increase provided by deploying more nodes begins to diminish as the size of the sensor network grows. Consequently, the benefit-to-cost ratio is decreasing as the initial number of sensors deployed (denoted $N_0$) grows. Therefore, the number of initial nodes required to meet a long mission requirement will become prohibitively expensive. Second, as discussed in Section 1.3.5, the quality of the network will also deteriorate with increasing $N_0$ due to capacity limits on cluster-heads and at the base station (Hu and Li [2004a] argue that the *capacity constrained* lifetime of a WSN will decrease in the order of $1/\sqrt{N_0}$). That is, the number of nodes required to meet the lifetime requirement may result in a poor quality network.

A replenishment strategy adds new nodes to the network at subsequent stages in order to meet mission requirements (connectivity, coverage, lifetime, *etc.*). The number of nodes added in any period should be sufficient to meet the expected number of nodes that fail between replenishments. If too few nodes are added, there may be a loss of coverage or connectivity, or an increase in the average energy consumed in the network due to excessively long paths from a clusterhead to the base station. Adding too many nodes in a replenishment may result in higher costs than necessary to meet the mission requirements, while affecting network capacity and increasing the network energy consumption. The objective of the replenishment strategy is to meet mission requirements while minimizing the total cost of the mission. The total costs associated with a replenishment strategy are (1) penalties associated with having too many or too few active nodes, (2) the actual cost of the sensor nodes, and (3) the fixed cost of deploying them on the the monitored region.

#### Relation to Sleep Scheduling Approaches

Another class of solutions for sensor replenishment that has been discussed in the literature is to overdeploy nodes in the monitored area and then put a subset of them to sleep. Then, as the network begins to fail, the sleeping nodes are 'awakened' to replace the failing nodes. With suitable modifications to a few cost parameters, these *sleep scheduling* replenishment schemes can be treated as special cases within the general replenishment controller proposed in this chapter. However, there may be logistical reasons not to use a sleep scheduling strategy even though the effect is the same as replenishment. For example, prolonged exposure to harsh terrain and weather would make node failures more likely.

### 5.1.1 Replenishment Control Overview

To date, there has been little discussion of the specific problem of WSN replenishment in the literature. However, there exists a large body of closely-related work within the operations research, management science and decision science communities devoted to the control of inventories in supply chains (called inventory management, or inventory control, *e.g.*, Axsater [2006]). Related work in the systems and control engineering disciplines can be found in the optimal control literature (*e.g.*, Bertsekas [1995]). In general, inventory control problem are studied using mathematical models from optimal control, dynamic programming, and network optimization.

A block diagram describing the replenishment problem is illustrated in Figure 5.1; each step is numbered. The blocks *failure forecast* and *controller* are assumed to reside at the base station (BSt). (1) At the beginning of time period, $t$, there are $x_j$ nodes in band $j$. (2) During this time period $w_j$ nodes fail in band $j$. (3) The *failure forecast* block uses the number of failures in each band during this time period to compute the expected number of nodes to fail in each band over a selected planning horizon. These estimates are provided to the *controller*. (4) The replenishment controller uses the estimates to compute the number of nodes that should be ordered ($y$), and an *allocation* ($\mathbf{z}$) for these nodes across all $J$ bands, $L$ periods into the future. (5) If an order was placed at period $t - L$, the batch is received at time $t$. (6) When the order is delivered, the nodes are allocated to bands according to ($\mathbf{z}$). During period $t$, the number of nodes in each band is affected by node failures ($\mathbf{w}$); these failures are detected by the controller at the beginning of the next period.

The WSN deployment model introduced in Fig. 5.1 bears close resemblance to a classical problem in inventory control called the multi-period multi-location inventory and supply problem [Krishnan and Rao 1965]. In this problem, an inventory system consists of a central depot which supplies $J$ retailers where random demands for a single commodity must be filled. The inventories are reviewed and decisions are made periodically. The decisions to be made (centrally, at the depot) are 1. the amount of stock to order from the supplier to be delivered to the depot, and 2. the fraction of the order each retailer will receive.

The simplest form of the multi-period multi-location inventory and supply problem

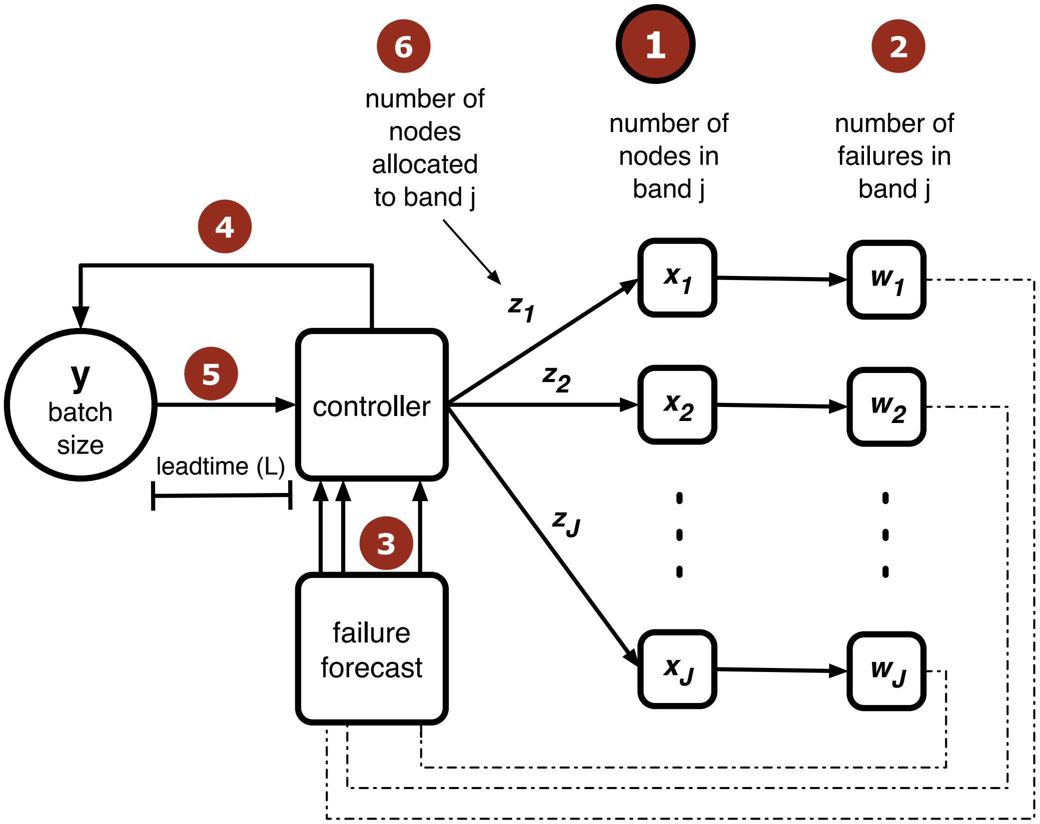

6
number of nodes allocated to band j
1
number of nodes in band j
2
number of failures in band j
4
y
batch size
5
controller
leadtime (L)
3
failure forecast
z1
z2
zJ
x1
x2
xJ
w1
w2
wJ

Figure 5.1: Overview of the replenishment control problem: The number of nodes in a band $j$ at any time is given by $x_j$. As nodes fail in each band according to the random processes $(w_j(t))$, the controller receives information about the number of failures in each band; a forecast of future failures for each band is used to choose a batch size $y_t$ of nodes to order and the placement over all bands that will minimize expected future costs. When the nodes arrive after a lead-time $(L)$, they are delivered to each of the $J$ bands according to the minimum cost allocation.

can be solved using dynamic programming, yielding an optimal policy consisting of a simple rule. This rule dictates that if the inventory at any of the retailers falls below a level $s$, then one should order enough supply to fill each of their inventories back up to $s$. Two basic assumptions are included in this simple formulation. First, the demand at each retailer is assumed to be stationary and normally distributed about a known mean with a constant, known variance. Additionally, no fixed cost is incurred for orders from the supplier. The only costs are a penalty for stockouts at a retailer, a holding cost for storing items, and the unit cost per item purchased. This pricing scheme means that the depot can place frequent orders for small batches in order to meet demand at the retailers without penalty.

However, even with these simplifying assumptions, the task of computing an optimal policy using dynamic programming can be formidable if the number of retailers is large. For $J$ retailers and a lead-time of $L$, the problem will have a $J + L$-dimensional state space, and the number of states in a dynamic programming problem grows exponentially in the number of dimensions. This is known as the "*curse of dimensionality*" in dynamic programming. For any but the smallest $J, L$, and terminal time, $T$, computing the exact solution will be impractical.

The version of this problem that we are considering will be more complicated. First, we assume that there is a nonzero fixed cost for deploying a batch of sensors (of any size). This value could include the cost of ordering nodes, outfitting nodes with sensors, and the cost of transporting these nodes to the field either by aircraft or vehicle. Second, we suppose that there is a delay between the placement of an order and the time it can be delivered to the field (lead-time). Finally, we know (from our previous analysis of the failure patterns of nodes) that the failure model will not be stationary. Therefore, in addition to the computational constraints provided by the dimensionality of the problem, our problem requires a failure model that can estimate with precision the number (and location) of failures beyond a short horizon. Since a dynamic program solves the multi-stage problem by starting at the terminal time ($T$) and computing the optimal decision path backwards in time, a full characterization of the failure process is required.

### 5.1.2 Summary of proposed replenishment controller

This chapter describes and extends work that was first presented in [Dorsey and Kam 2010], which. In the next section we will provide expressions for the costs associated with ordering new nodes, deploying nodes, and penalties for deploying too few or too many nodes to bands in the network. This effort will result in a full dynamic programming setup of the problem (Section 5.2.3). In the subsequent sections, we will introduce a tractable approximation of the problem through the use of a *myopic allocation policy* (Section 5.3.1), where an order received from $L$ periods ago is divided up among the bands in order to minimize the expected costs in the current period, ignoring costs in subsequent periods. Then, using results from [Zipkin 1982], an approximation of the myopic allocation problem (Section 5.3.2) in terms of *aggregate state variables* (introduced in Section 5.2.3) is described. This approximation is integrated back into the full dynamic program, to yield a new problem in terms of aggregate state variables (not in terms of the distribution of nodes over all bands). Finally, we describe a transformation of the cost function that allows us to shift future costs (incurred $L$ periods into the future) to the present. The final result is a single-dimensional dynamic program.

The myopic allocation policy presents a trade-off between optimality and computational complexity. If failures are observed as stationary and uncorrelated, then the cost of the myopic aggregation policy is a lower bound on the true cost of the optimal solution. The extent to which failures are observed as stationary will depend on the fluctuation in the distribution of the residuals from the failure forecasting model described in section 5.4. In the case of non-stationary, correlated failures, a recent paper [Truong 2012] shows that the myopic policy is the tightest known approximation bound for this problem, and that the expected cost is *at most* twice the expected holding cost plus the expected shortage cost of the optimal policy.

In Section 5.4, the Holt-Winters forecast model [Holt 1957] is introduced to estimate the number of node failures over the next lead-time period. We present simulation data that suggest that the residuals of the forecast model are Gaussian, but the parameters of this model are non-stationary. The costs of the myopic allocation policy and the full dynamic programming approach are compared in Section 5.8 for a three-band deployment. The simulations show that, for this scenario, the approximation results in total costs (shortage and surplus penalties, and fixed delivery costs) that are approximately 20% higher than those of the full dynamic programming controller. However, the full DP approach takes nearly 700 times as long to compute a solution as the approximate dynamic programming controller.

## 5.2 Description of WSN Replenishment Problem

### 5.2.1 Notation and definitions

The notation for the replenishment problem will follow the convention that variables in boldface are vectors. The 'hat' symbol above a variable indicates that it is a aggregation over time or a vector of values over time. Otherwise, vectors always have $J$ elements, one for each band. We identify some variables related to the state of the system, including the number of active nodes in a band at the beginning of a period, batches of nodes ordered for deployment, and the allocation of the order over bands. These are listed in the box below (**State variables**). Using this notation, we can describe the *state* of the system at the beginning of period $t$ as $(\mathbf{x}_t, \hat{\mathbf{y}}_t)$. Here $\mathbf{x}_t$ contains the number of nodes in each band at time $t$ and $\hat{\mathbf{y}}_t$ is the vector of all orders that have been placed before time $t$ that have not yet been delivered.

**State variables**

$L$ Lead time

$t$ Period index

$T$ Mission lifetime requirement

$j$ Band index ($j = 1 \dots J$)

$x_{jt}$ Number of active nodes in band $j$ at the beginning of period $t$

$\mathbf{x}_t$ $= (x_{jt})_{j=1}^{J}$

$y_t$ Number of nodes ordered at the beginning of period $t$

$\hat{\mathbf{y}}_t$ $= (y_s)_{s=t-L}^{t-1}$ Vector of orders placed in the last $L - 1$ periods that have not yet been allocated

$z_{jt}$ Number of nodes delivered to band $j$ in period $t$ (allocation)

$\mathbf{z}_t$ $= (z_{jt})_{j=1}^{J}$

Next we introduce some variables related to the failure model, and the cost variables.

**Notation related to failures**

$w_{jt}$ Number of failures in band $j$ in period $t$

$F_{jt}$ Marginal cumulative distribution function (CDF) of $w_{jt}$

$\mu_{jt} = \mathbb{E}(w_{jt})$ Expected number of failures in band $j$ during period $t$

$\sigma_{jt}^2 = Var(w_{jt})$ Variance of the distribution $F_{jt}$

$\hat{\mu}_{jt} = \sum_{s=t}^{t+L} \mu_{jt}$ Expected number of lead time failures in band $j$ starting in period $t$

$\hat{\sigma}_{jt}^2 = \sum_{s=t}^{t+L} \sigma_{jt}^2$ Variance of the lead time distribution $F_{jt+L}$

Also, let $\Phi$ and $\phi$ denote the standard normal cdf and pdf, respectively.

**Cost variables**

$C_t(y_t)$ Cost associated with ordering $y_t$ nodes in period $t$

$K$ Fixed cost for deploying a batch of nodes

$c_s$ Cost of an individual sensor node

$h$ Cost associated with deploying too many nodes in a band

$d$ Cost associated with deploying too few nodes in a band

The costs $h$ and $d$ are incurred in each period whenever the number of nodes in a band is above or below a specified target level. Let $\rho_{jt}$ denote the target for band $j$

in period $t$. Then, the cost associated with the level in band $j$ at the end of period $t$ is $d(x_{jt} - \rho_{jt})^- + h(x_{jt} - \rho_{jt})^+$, where the $+$ and $-$ superscripts denote functions that return the absolute value of the argument if it is positive or negative, respectively, and zero otherwise.

Since we are interested in the deviation of $x_{jt}$ from the target for our cost calculations, we will introduce the variables

$$\tilde{x}_{jt} = x_{jt} - \rho_{jt},$$

and

$$\tilde{\mathbf{x}}_t = (\tilde{x}_{jt})_{j=1}^{J}.$$

### 5.2.2 Costs

The costs associated with deploying nodes ($K$), and the actual price of the nodes ($c_s$) for a batch size $y_t$ is given by

$$C_t(y_t) = \mathbf{1}(y_t > 0)K + c_s y_t,$$

where

$$\mathbf{1}(y_t > 0) = \begin{cases} 1, & \text{if } y_t > 0, \\ 0, & \text{otherwise.} \end{cases}$$

If no nodes were ordered for period $t$, then the deployment cost $(y_t > 0)K$ and sensor cost $(c_s y_t)$ are zero. If $y_t > 0$, then the fixed cost $K$ is incurred independent of how many nodes are ordered. When $K$ is large, the controller will avoid ordering small batches and placing orders too often. If $K = 0$, the controller will replenish nodes in small batches to replace ones that failed in the previous time period.

The single period expected costs related to node levels (*penalty costs*) across all bands in period $t$ are

$$Q_t(\mathbf{z}_t, \tilde{\mathbf{x}}_t) = \sum_{j=1}^{J} q_{jt}\left(z_{jt}, \tilde{x}_{jt}\right),$$

where

$$q_{jt}\left(z_{jt}, \tilde{x}_{jt}\right) = h\mathbb{E}\left[\tilde{x}_{jt} + z_{jt} - w_{jt}\right]^{+} + d\mathbb{E}\left[\tilde{x}_{jt} + z_{jt} - w_{jt}\right]^{-}.$$

The formula $q_{jt}$ is used to compute the expected penalty for having more nodes or less nodes than the target level $\rho_{jt}$ in band $j$ at time $t$. Since $\tilde{x}_{jt}$ is the current deviation from the target, this expression for a single period is ideally zero (the target deviation and the allocation in band $j$ at time $t$ offset the node failures). The expectation in taken with respect to $w_{jt}$, the number of failures in band $j$ at time $t$.

We can rewrite this expression in terms of the marginal cumulative distribution function (CDF) of $w_{jt}$ by integrating $F_{jt}$ up to the sum of the target deviation and the allocation:

$$q_{jt}\left(z_{jt}, \tilde{x}_{jt}\right) = d\left(\mu_{jt} - \left(\tilde{x}_{jt} + z_{jt}\right)\right) + (d+h)\int_{-\infty}^{\tilde{x}_{jt}+z_{jt}} F_{jt}(u)du \tag{5.1}$$

This formulation will help to derive the myopic allocation policy and its approximation.

### 5.2.3 System dynamics and recursive equations

The initial state is given by the node level deviations from target and the outstanding node orders to be delivered in time period $t$, $(\tilde{\mathbf{x}}_t, \hat{\mathbf{y}}_t)$. During period $t$, and order may be placed to be delivered at $t+L$, and there are $\mathbf{w}_t \geq 0$ node failures. If an order is received in the current time period, it will be allocated to the monitored area according to $\mathbf{z}_t$. Then the state of the system in the next period will be $(\tilde{\mathbf{x}}_{t+1}, \hat{\mathbf{y}}_{t+1})$:

$$\tilde{\mathbf{x}}_{t+1} = \tilde{\mathbf{x}}_t + \mathbf{z}_t - \mathbf{w}_t,$$

$$\hat{\mathbf{y}}_{t+1} = (y_{t-L+1}, \ldots y_{t-1}, y_t).$$

We can now state the recursive Bellman equation whose solution provides the optimal policy for the problem. Let $f_t(\tilde{\mathbf{x}}_t, \hat{\mathbf{y}}_t)$ denote the minimum total expected costs in periods $t$ through $T$, given that the system starts in the initial state $(\tilde{\mathbf{x}}_t, \hat{\mathbf{y}}_t)$. Then, the Bellman equation computes the minimum cost from the current time $t$ to the final time period $T$ over the decision variables $y_t$ and $\mathbf{z}_t$, the number of nodes ordered and their allocations to bands in each time period. The Bellman equation is written as

$$\begin{aligned} f_{T+1} &= 0 \\ f_t(\tilde{\mathbf{x}}_t, \hat{\mathbf{y}}_t) &= \min_{y_t, \mathbf{z}_t} \left[ C_t(y_t) + Q_t(\mathbf{z}_t, \tilde{\mathbf{x}}_t) + \mathbb{E}\left[ f_{t+1}\left( (\tilde{\mathbf{x}}_t + \mathbf{z}_t - \mathbf{w}_t), \hat{\mathbf{y}}_t \right) \right] \right]. \end{aligned} \tag{5.2}$$

The function $f_t(\tilde{\mathbf{x}}_t, \hat{\mathbf{y}}_t)$ consists of the fixed costs $C_t(y_t)$ that include the sensor node costs and the fixed cost for delivery $(K)$ and the single period node level penalties $Q_t(\mathbf{z}_t, \tilde{\mathbf{x}}_t)$. The last term contains recursion that sums the fixed costs and node level penalties for the expected node failures in the future. The domain of $y_t$ and $\mathbf{z}_t$ are constrained by

$$\begin{aligned} \sum_{j=1}^{J} z_{jt} &= y_{t-L} \\ y_t &\geq 0 \\ \mathbf{z}_t &\geq 0. \end{aligned} \tag{5.3}$$

This formulation of the dynamic equation suffers from high dimensionality because it has to find order sizes $y_t$ and allocations $z_{jt}$ for all bands and for all time periods.

Moreover, the order sizes and allocations are not independent, since the possible allocations of nodes over bands depends on the number of nodes that are to be ordered (this is the first constraint in Eq. 5.3). Also, a failure probability distribution for all $t = 0 \ldots T$ and all $j$ is assumed. In order to make the problem easier to solve, we will amend the problem so that we may consider separately the problem of choosing the number of nodes to order, $y_t$, and the number of nodes to allocate to each band, $\mathbf{z}_t$. This effort will involve the introduction of a few aggregate variables, where the aggregation takes place over the $J$ bands and $L$ periods.

### Aggregate variables

We first define two aggregate failure random variables,

$$W_t = \sum_{j=1}^{J} w_{jt}$$

and

$$\bar{W}_t = \sum_{s=t}^{t+L-1} W_s.$$

$W_t$ is the sum of all node failures in time period $t$ over all $J$ bands. $\bar{W}_t$ is the sum of node failures in all bands and over the time period $t \ldots t + L$. Thus, $\bar{W}_t$ is the total number of failures over the next lead-time. $W_t$ and $\bar{W}_t$ are assumed to be normal random variables with cdfs $G_t$ and $\bar{G}_t$, and characterized by means

$$M_t = \sum_{j=1}^{J} \mu_{jt}, \text{ and } \bar{M}_t = \sum_{s=t}^{t+L-1} M_t$$

and variances

$$S_t^2, \text{ and } \bar{S}_t^2 = \sum_{s=t}^{t+L-1} S_t^2.$$

The assumption of normality is justified by the observed distribution of residuals of the failure model to be shown in Section 5.4. Finally, we define variables that store the aggregate number of active nodes over $J$ bands and $L$ periods, relative to the target levels. Let

$$\tilde{X}_t = \sum_{j=1}^{J} \tilde{x}_{jt} - \sum_{j=1}^{J} \rho_{jt},$$

$$\tilde{X}_t^{\Delta} = \tilde{X}_t + \sum_{s=t-L}^{t-1} y_s.$$

$\tilde{X}_t$ is the sum of the node level target deviations over all bands, and $\tilde{X}_t^{\Delta}$ is the sum of $\tilde{X}_t$ and all orders that have been placed but have not yet been delivered.

## 5.3 Approximation of the Dynamic Program

In this section, we introduce some results from [Federgruen and Zipkin 1984] and [Zipkin 1982] that allow separation of the ordering problem from the allocation problem and rewrite the dynamic programming problem (Eq. 5.2) as a one-dimensional dynamic program. First we define the myopic allocation problem.

### 5.3.1 Myopic allocation problem

A myopic allocation problem involves dividing a batch of nodes ordered $L$ periods ago among the bands in the network in order to minimize the expected costs in the current period, ignoring costs in subsequent periods. For any period $t \leq T$, the myopic allocation problem is given by

$$
\begin{aligned}
R_t(\tilde{\mathbf{x}}_t, y_{t-L}) = \min_{\mathbf{z}_t} Q_t(\mathbf{z}_t, \tilde{\mathbf{x}}_t) \\
\text{subject to: } \mathbf{z}_t \geq 0, \\
\sum_{j=1}^{J} z_{jt} = y_{t-L}.
\end{aligned}
\tag{5.4}
$$

$R_t$ is the minimum value of the node level penalty costs, $Q_t$, defined in Section 5.2.2, over all possible allocations of the order placed in period $t-L$, and the vector $\mathbf{z}_t$ that minimizes $Q_t$ is the optimal allocation.

### 5.3.2 Approximation of the myopic allocation problem

In [Zipkin 1982], the authors describe a method for approximating the minimal cost of an allocation problem, by a simple, closed-form aggregate cost function. This technique allows us to state $R_t$ as a function of the scalars $\tilde{X}_t$ and $y_{t-L}$ instead of the $J$-vector $\tilde{\mathbf{x}}_t$. Note, however, that although we are approximating the minimum *value* of the expected single-period penalty cost function $Q_t$, the actual myopic allocation is still given by Eq. 5.4 (we still need to choose a vector $\mathbf{z}_t$ such that $Q_t$ is equal to the approximation and meets the constraints). We begin by restating $Q_t$ from Section 5.2.2:

$$
Q_t(\mathbf{z}_t, \tilde{\mathbf{x}}_t) = \sum_{j=1}^{J} q_{jt}\left(z_{jt}, \tilde{x}_{jt}\right),
$$

where

$$
q_{jt}\left(z_{jt}, \tilde{x}_{jt}\right) = d\left(\mu_{jt} - \left(\tilde{x}_{jt} + z_{jt}\right)\right) + (d+h)\int_{-\infty}^{\tilde{x}_{jt}+z_{jt}} F_{jt}(u)du.
$$

The variables $\mu_{jt}$ and $\tilde{x}_{jt}$ are substituted with the aggregate variables $M_t$ and $\tilde{X}_t$, and the allocations $\mathbf{z}_t$ are replaced with the number of nodes to arrive in time $t$, $y_{t-L}$. Also, $F_{jt}(u)$ is replaced with the standard normal cdf $\Phi\left(\frac{u-\mu_{jt}}{\sigma_{jt}}\right)$. The approximate single-period penalty cost function $Q_t$ is written as

$$Q_t = \sum_{j=1}^{J} q_{jt} = d\left(M_t - \left(\tilde{X}_t + y_{t-L}\right)\right) + (d+h)\sum_{j=1}^{J}\int_{-\infty}^{\tilde{x}_{jt}+z_{jt}} \Phi\left(\frac{u-\mu_{jt}}{\sigma_{jt}}\right) du. \quad (5.5)$$

Since the minimization in (5.4) is computed over $\mathbf{z}_t$ (not the order size, $y$), and $d$ and $h$ are both positive, the myopic allocation problem reduces to

$$\begin{aligned} R_t(\tilde{\mathbf{x}}_t, y_{t-L}) = \min_{\mathbf{z}_t} & \left[\sum_{j=1}^{J}\int_{-\infty}^{\tilde{x}_{jt}+z_{jt}} \Phi\left(\frac{u-\mu_{jt}}{\sigma_{jt}}\right) du\right] \quad (5.6) \\ & \text{subject to: } \mathbf{z}_t \geq 0, \\ & \sum_{j=1}^{J} z_{jt} = y_{t-L}. \end{aligned}$$

Denote the approximation of the minimum value of the myopic allocation, $R_t$, as $\hat{R}_t$. Zipkin derives an expression for $\hat{R}_t$ in two steps. In the first step, the non-negativity constraint, $\mathbf{z}_t \geq 0$, in (5.4) and (5.6) is relaxed. Under this relaxation, all bands collapse into a single aggregate band. The allowance of negative values for some entries in $\mathbf{z}_t$ implies, in effect, that some nodes may be "taken" from one band and moved to another. With this relaxation, it is possible to separate the decision of how many nodes to order and the decision about where to allocate them once the order arrives.

This relaxation of the problem is justified by what is termed the "allocation assumption" in inventory control [Eppen and Schrage 1981]. This assumption states that, when a order arrives, we can make an allocation such that the probability of falling below the target levels in each band is the same in the next period. This assumption is contingent on the batch size being large enough to accommodate this allocation; if the batch size is sufficiently large, then the relaxation of the non-negativity constraint will not affect the solution to the minimal cost problem (5.4).

In the second step, the optimality conditions for the remaining problem (after relaxation of the constraints) are manipulated (see Zipkin [1982]) to yield (using $\Phi_{jt}^{-1}(\frac{u-\mu_{jt}}{\sigma_{jt}}) = \mu_{jt} + \sigma_{jt}\Phi^{-1}(u)$):

$$\sum_{j=1}^{J} \left( \mu_{jt} + \sigma_{jt}\Phi^{-1}\left(\frac{d+\xi}{d+h}\right)\right) = \tilde{X}_t + y_{t-L},$$

which is equivalent to

$$M_t + S_t^2\Phi^{-1}\left(\frac{d+\xi}{d+h}\right) = \tilde{X}_t + y_{t-L}, \tag{5.7}$$

where $\xi$ is the Lagrange multiplier. Since the Lagrange multiplier is equal to the derivative of the minimum value of the relaxed problem, *i.e.* $\xi = \partial R_t / \partial y_{t-L}$, we can obtain the approximation $\hat{R}_t$ by solving (5.7) for $\xi$,

$$\xi = -d + (d+h)\Phi\left(\frac{\tilde{X}_t + y_{t-L} - M_t}{S_t}\right)$$

and integrating over $y_{t-L}$ to get

$$\begin{aligned} & \hat{R}_t\left(\tilde{X}_t, y_{t-L}\right) && (5.8) \\ \equiv\ & d\left(M_t - \tilde{X}_t\right) - dy_{t_L} + (d+h)\int_{-\infty}^{\tilde{X}_t + y_{t-L}} \Phi\left(\frac{U - M_t}{S_t}\right) dU \\ =\ & d\left(M_t - \left(\tilde{X}_t + y_{t-L}\right)\right) + (d+h)\int_{-\infty}^{\tilde{X}_t + y_{t-L}} \Phi\left(\frac{U - M_t}{S_t}\right) dU, \end{aligned}$$

where the constant term of the integration is $d\left(M_t - \tilde{X}_t\right)$.

### 5.3.3 Integration of the approximate myopic allocation into the dynamic program

The Bellman equation for the original dynamic programming formulation of the replenishment problem stipulated a minimization of the total expected costs over the number of nodes to order $y$ and where the nodes should be deployed $\mathbf{z}$. Eq. 5.2 is now restated for reference:

$$f_t(\tilde{\mathbf{x}}_t, \hat{\mathbf{y}}_t) = \min_{y_t, \mathbf{z}_t} \left[C_t(y_t) + Q_t(\mathbf{z}_t, \tilde{\mathbf{x}}_t)) + \mathbb{E}\left[f_{t+1}\left((\tilde{\mathbf{x}}_t + \mathbf{z}_t - \mathbf{w}_t), \hat{\mathbf{y}}_t\right)\right]\right].$$

Using the aggregate variables introduced in Section 5.2.3, an approximation to $f_t$ is

$$\hat{f}_t\left(\tilde{X}_t, \hat{\mathbf{y}}_t\right) = \min_{y_t, \mathbf{z}_t} \left\{C_t(y_t) + Q_t(\mathbf{z}_t, \tilde{\mathbf{x}}_t) + \mathbb{E}\left[\hat{f}_{t+1}\left((\tilde{X}_t + y_{t-L} - W_t), \hat{\mathbf{y}}_t\right)\right]\right\}.$$

In the recursion term $\hat{f}_{t+1}$, the allocation $\mathbf{z}_t$ was replaced with $y_{t-L}$, and the node level deviation vector, $\tilde{\mathbf{x}}_t$, was replaced with the sum level deviation over all bands, $\tilde{X}_t$. Thus, the single period node level penalty $Q_t(\mathbf{z}_t, \tilde{\mathbf{x}}_t)$ is independent of both the fixed costs $C_t(y_t)$ and the recursion $\hat{f}_{t+1}$. Therefore, we can rewrite $\hat{f}_t$ as two independent minimizations,

$$\begin{aligned}\hat{f}_t\left(\tilde{X}_t, \hat{\mathbf{y}}_t\right) &= \\ &\min_{y_t} \left\{C_t(y_t) + \mathbb{E}\left[\hat{f}_{t+1}\left((\tilde{X}_t + y_{t-L} - W_t), \hat{\mathbf{y}}_t\right)\right]\right\} \\ &+ \min_{\mathbf{z}_t} \left\{Q_t(\mathbf{z}_t, \tilde{\mathbf{x}}_t)\right\},\end{aligned}$$

where the second minimization is the myopic allocation. Therefore, the minimizations over $y_t$ and $z_t$ are separate. This suggests that, up to the approximation $\hat{f}_{t+1}$, the myopic allocation is optimal in period $t$. At this point, the dimensionality of the minimization over $y_t$ is $L+1$, instead of $L+J$, and the allocation minimization has dimension $J$. Since the cost minimizations for orders and their allocations can be separated, the problem can now be reduced to a problem in terms of $\tilde{X}$ and $\mathbf{y}_t$ by replacing $\min_{\mathbf{z}_t}\{Q_t(\mathbf{z}_t, \tilde{\mathbf{x}}_t)\}$ with the value of the approximate myopic allocation $\hat{R}_t$ (Eq. 5.8). This yields a problem with $L+1$ dimensions:

$$
\begin{aligned}
\hat{f}_t\left(\tilde{X}_t, \hat{\mathbf{y}}_t\right) \quad = & \\
& \hat{R}_t\left(\tilde{X}_t, y_{t-L}\right) \\
& + \min_{y_t}\left\{C_t(y_t) + \mathbb{E}\left[\hat{f}_{t+1}\left((\tilde{X}_t + y_{t-L} - W_t), \hat{\mathbf{y}}_t\right)\right]\right\}.
\end{aligned}
$$

### 5.3.4 Cost shifting

The final step in reducing the dynamic program that solves the replenishment problem is to shift the way that the costs are counted. Note that, once an order is placed, there is nothing that can be done at time $t$ to affect the situation in any band until period $t+L$. Therefore, we do not change the nature of the problem by counting in period $t$ the *expected* penalty costs previously incurred in period $t+L$.

Viewed from period $t$, the number of nodes over the entire network in period $t+L$ will be

$$
\begin{aligned}
\tilde{X}_{t+L} &= \tilde{X}_t + \sum_{s=t-L}^{t-1} y_s - \sum_{s=t}^{t+L-1} W_s \qquad (5.9) \\
&= \tilde{X}_t^{\Delta} - \bar{W}_t,
\end{aligned}
$$

Where $\tilde{X}_t^{\Delta}$ is the sum of $\tilde{X}_t$ and all orders that have been placed but have not yet been delivered, and $\bar{W}_t$ is the sum of all expected node failures overall bands and through time periods $t \ldots t+L$. The expected penalty cost in period $t+L$ is therefore

$$\begin{aligned} & P_t\left(\tilde{X}_t^{\Delta}, y_t\right) && (5.10) \\ & = \mathbb{E}_{\bar{W}_t}\left[\hat{R}_{t+L}\left(\tilde{X}_t^{\Delta} - \bar{W}_t, y_t\right)\right] \\ & = \int \left\{ d\left(M_{t+L} - \left(\tilde{X}_t^{\Delta} - \bar{W}_t + y_t\right)\right)\right. \\ & + \left.(d+h)\int_{-\infty}^{\tilde{X}_t^{\Delta} - \bar{W}_t + y_t} \bar{G}_{t+L}(U)dU \right\} d\bar{G}_t\left(\bar{W}_t\right), \end{aligned}$$

where $\bar{G}_t$ is the aggregate failure distribution (summed over $J$ and $L$) with mean and variance defined in Section 5.2.3. By reversing the order of integration

$$P_t\left(\tilde{X}_t^{\Delta}, y_t\right) = d\left(\left(M_{t+L} + \bar{M}_t\right) - \left(\tilde{X}_t^{\Delta} + y_t\right)\right) + (d+h)\int_{-\infty}^{\tilde{X}_t^{\Delta} + y_t} H(U)dU, \quad (5.11)$$

where $H_t$ is the convolution of the two normal distributions, $\bar{G}_t$ and $\Phi\left(\frac{\tilde{X}_{t+L} + y_t - M_{t+L}}{S_{t+L}}\right)$.

At this point, we can define $g_t\left(\tilde{X}_t^{\Delta}\right)$ as the minimum total expected ordering and penalty costs incurred in periods $t$ through $T$, given the network state is $\tilde{X}_t^{\Delta}$ in period $t$. Starting with the terminal costs, $g_{T-L+1} = 0$, and for $t \geq T-L$,

$$g_t\left(\tilde{X}_t^{\Delta}\right) = \min_{y_t \geq 0}\left\{C_t(y_t) + P_t\left(\tilde{X}_t^{\Delta}, y_t\right) + \mathbb{E}\left[g_{t+1}\left(\tilde{X}^{\Delta} + y_t - W_t\right)\right]\right\}. \quad (5.12)$$

This is a single-dimension dynamic program with no lead time, failures $W_t$, and convex penalty costs $P_t$. The form of this dynamic program (with fixed plus linear ordering costs) is known to result in a solution of type $(s, S)$ [Clark and Scarf 1960]. That is, if the number of nodes in the network is below $s$, then we order enough nodes to increase $\tilde{X}_t$ up to $S$.

In order to treat the non-stationarity of the lead-time failure rate, we require a forecasting method that will adapt to trends in the failure distribution parameters. So far we have explicitly stated that the lead-time failure distribution is a normal distribution $\bar{G}_t$ with time-varying means and variances $\bar{M}_t$ and $\bar{S}_t^2$. In the next section we will describe the lead-time failure model and justify the use of the normal distribution to characterize lead-time failures.

### 5.4 Failure Process Model

The future values of the failure process are unknown and therefore can be treated as a random variable. Since replenishment opportunities are constrained by the fixed lead-time, $L$, our primary focus is the expected number of node failures in all bands over a finite planning horizon, $T_h$ that is greater than or equal to the lead-time. These *lead-time failures* (LTF) are the aggregate of unknown future values $W_t$ defined as $\bar{W}_t = \sum_{t=1}^{t+T_h-1} W_t$.

We use a trend-corrected exponential smoothing method first introduced by [Holt 1957] in order to estimate the LTF distribution from the time series of observed failures. Essentially this model is a decomposition of the series of lead-time failures into three components: (1) the current base level of failures; (2) the rate of increase or decrease of the base level; and (3) an error term.

Subplots 2-4 in Figure 5.2 shows this decomposition for a simulation running for 85 periods until network failure (with no replenishment). Subplot 1 (top) shows the number of failures in the entire network for each period. Subplot 2 shows the base level parameter of the process. It appears that the parameters of the failure distribution are non-stationary as the base level is increasing with time. Although this figure suggests an increasing failure rate, the actual mean failure rate shows flat, decreasing, and increasing trends as we replenish the network with new nodes. Subplot 3 shows the rate of change in the base level, and the bottom subplot shows the residual errors between the model and the data. The model is updated after each observation. The general smoothing formula is

$$\begin{aligned} W_t &= l_{t-1} + b_{t-1} + e_t \\ l_t &= l_{t-1} + b_{t-1} + \alpha e_t \\ b_t &= b_{t-1} + \alpha\beta e_t \end{aligned} \tag{5.13}$$

where $l_t$ is the base level term, $b_t$ is the rate of change in the base level (trend), and $e_t$ is the random error component. The constants $\alpha$ and $\beta$ are called the smoothing parameters.

After each observation of the failures, the error $e_t$ is computed as $W_t - \hat{W}_t$, where $\hat{W}_t$ is the predicted number of failures in period $t$ and $W_t$ is the actual number of failures. The unknown parameters $\alpha$ and $\beta$ are determined by minimizing the squared prediction error. The model is fully specified by the distribution of the error term $e_t$. We assume that the errors are independent and identically distributed, following a normal distribution with mean 0 and unknown variance $\sigma^2$. Figure 5.3 shows the histogram of the residual error term and a Q-Q plot comparing the data to a normal distribution. The data set was tested for normality using a Shapiro-Wilk test [Shapiro and Wilk 1964], resulting in a $p$-value of 0.98.

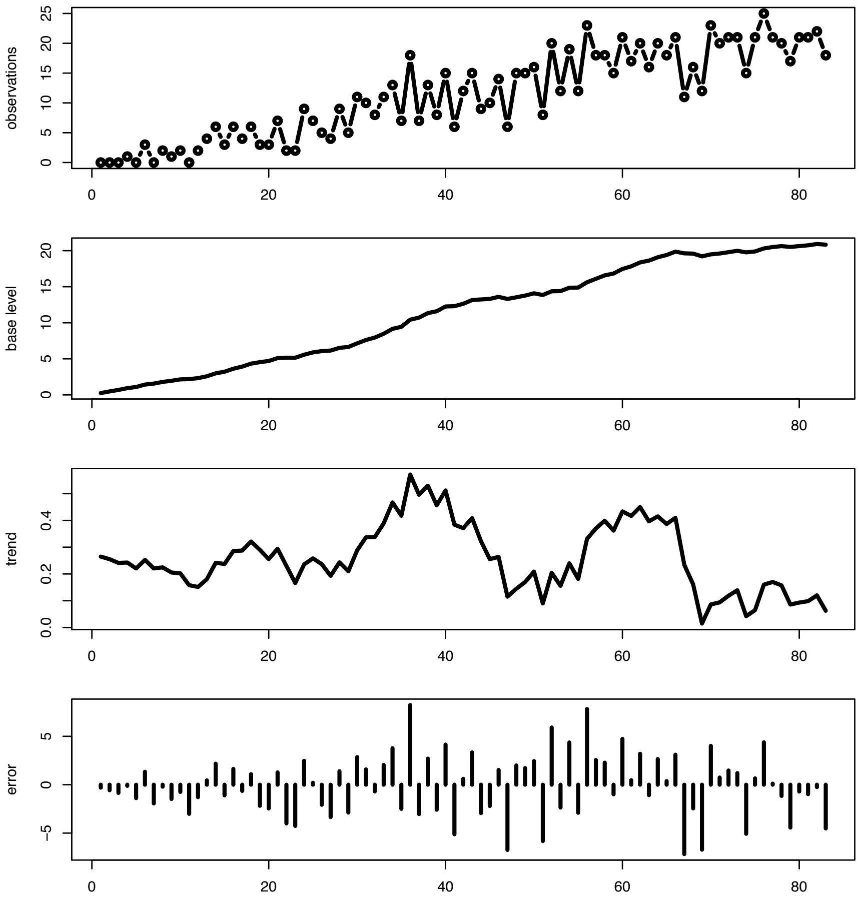


Figure 5.2: Subplot (1) shows the total failures for a single simulation run (to network failure). Subplots 2-4 show the decomposition of the failure process into (2) base level, (3) trend, and (4) error.

The calculations for forecasting the sensor lead-time failures are

$$\begin{aligned}
\hat{W}_t &= \hat{l}_{t-1} + \hat{b}_{t-1} \\
\hat{l}_t &= \hat{l}_{t-1} + \hat{b}_{t-1} + \alpha\left(W_t - \hat{W}_t\right) \\
\hat{b}_t &= \hat{b}_{t-1} + \alpha\beta\left(W_t - \hat{W}_t\right).
\end{aligned} \tag{5.14}$$

In order to obtain an estimation of the failures for period $t \ldots t+k$ at the end of period $t$, we compute $\hat{l}_t + k\hat{b}_t$. The mean and variance of the lead-time failures are

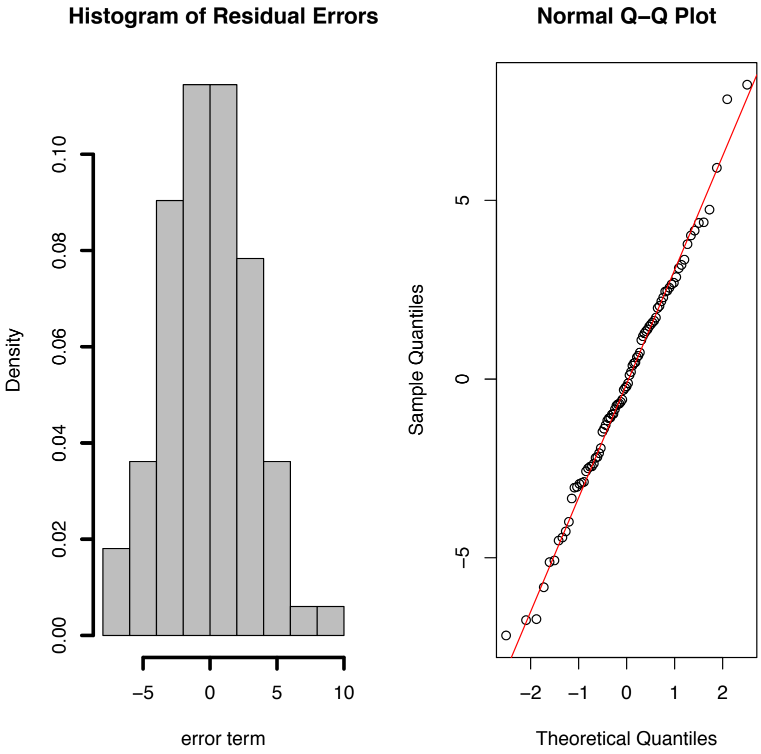


Figure 5.3: Testing normality of error data.

computed as:

$$\bar{M_t} = l_t + T_h b_t$$

and

$$\bar{S}_t^2 = S_t^2 \left[ 1 + \sum_{k=1}^{T_h - 1} (\alpha + k\beta) \right].$$

The variance depends only on the smoothing parameters and the length of the planning horizon, not on the observed failure rates. In fact, the smoothing parameters $\alpha$ and $\beta$ are determined using a least-squares minimization of the forecast error over the planning horizon from simulation data. Different values of $T_h$ will yield different parameter values. The result of the trend-corrected smoothing and a 20 period forecast of the failure rate are shown in Figure 5.4 for the initial test data. The colored bands around the forecast in the bottom subplot are the 0.95 confidence intervals of the forecast.

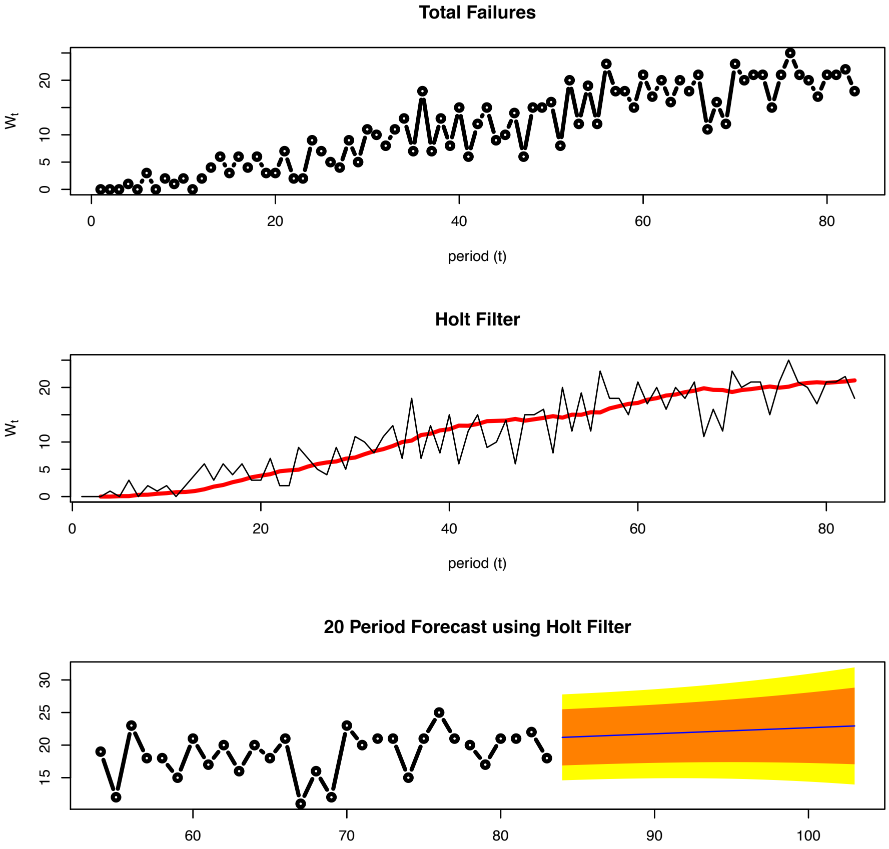


Figure 5.4: Holt smoothing technique and forecasting for a planning horizon of 20 periods.

## 5.5 Allocation Strategy

Once a batch order is received at the controller, the total batch size is to be divided among the individual bands according to an allocation strategy. Many heuristics are described in the literature for allocating stock to multiple retailers. Most of these studies use the fraction of customer demand at each retailer to distribute orders. Since we are observing failures in each band while waiting for an order to arrive, it makes sense to distribute the new batch among the bands according to which bands are experiencing the most failures. However, in our case, we have overdeployed nodes at the bands near the base station at the initial deployment, and it is neither necessary nor desirable to maintain these high densities as the network approaches the mission lifetime. Therefore, we introduced a moving target level for each band $\rho_{jt}$ in Sec. 5.2.1; these target levels decrease linearly from the initial deployment levels in each band down to the critical levels required for connectivity and coverage over the length of the mission time. The intent is to have the minimum number of nodes remaining on the field as the mission ends.

We are considering two variables in the allocation decision. The first is the fraction of failures that have occurred in a band $j$ over the last $L$ periods; this fraction is denoted $v_{1,j}$. The second is the fraction of deviation from the moving target level $\rho_{jt}$ for band $j$, denoted $v_{2,j} = (\rho_{jt} - x_{jt}) / \sum_{j=1}^{J} (\rho_{jt} - x_{jt})$. The allocation is then given as a linear combination of $v_{1,j}$ and $v_{2,j}$:

$$z_j = y_{t-L} \left[ a v_{1,j} + (1-a) v_{2,j} \right], \tag{5.15}$$

where $a$ is a constant expressing the weight given to the two variables.

An updated block diagram (revised from Figure 5.1) of the approximate replenishment controller is shown in Figure 5.5. The approximate DP controller and the myopic allocation blocks are now separate. The input to the failure forecast is an aggregate of all of the node failures in all bands at the end of the last period. The failure forecast provides the approximate DP controller with an estimate of the mean and variance of the expected node failures over the next planning horizon. The ap-

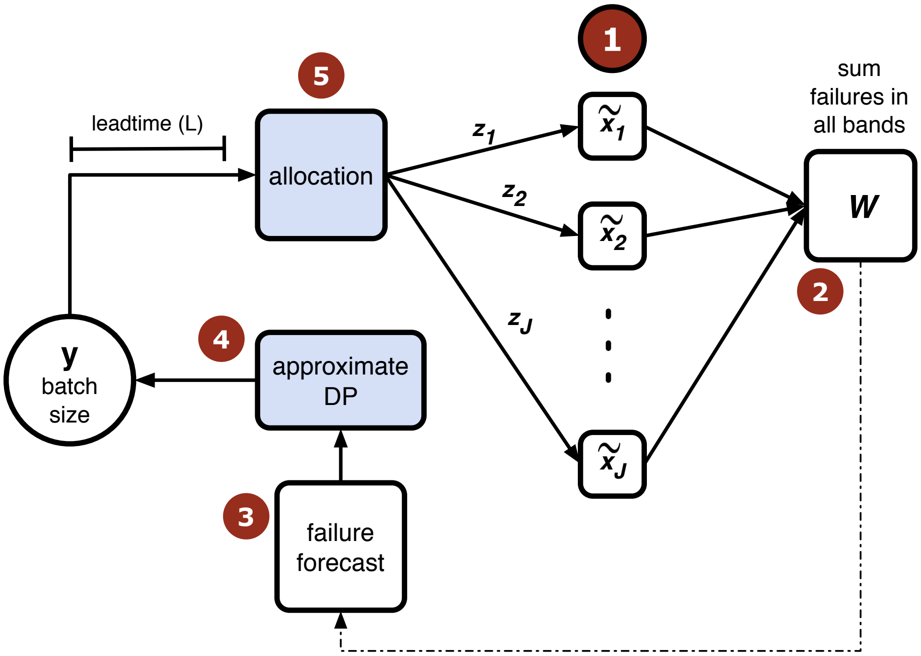


Figure 5.5: *Overview of the approximate replenishment controller.* (1) The deviation of the node level from the target level in a band $j$ at any time is given by $\tilde{x}_j$. (2) Node failures in all bands are provided to the approximate controller as the aggregate variable $W$. (3) The failure forecast method uses this information to compute an estimate of the mean and variance of the aggregate failures in all bands over the next planning horizon. (4) The forecast of future failures and the aggregate of node level deviations ($\tilde{X}$) is used to compute the batch size $y_t$ to order. (5) When the order arrives after the $L$ periods, a myopic allocation policy is used to decide how to divide the order among the $J$ bands.

proximate DP controller computes the using the output from the failure forecast, and the sum of node level deviations over all bands and all orders that have been placed but not delivered. When computing the optimal order size, the approximate DP controller uses an approximation of the minimum *cost* of a myopic allocation, $\hat{R}_t$, instead of computing the allocation $\mathbf{z}$ and the order size $y$ that will minimize expected future costs. The allocation is computed when the nodes arrive, using a myopic allocation strategy.

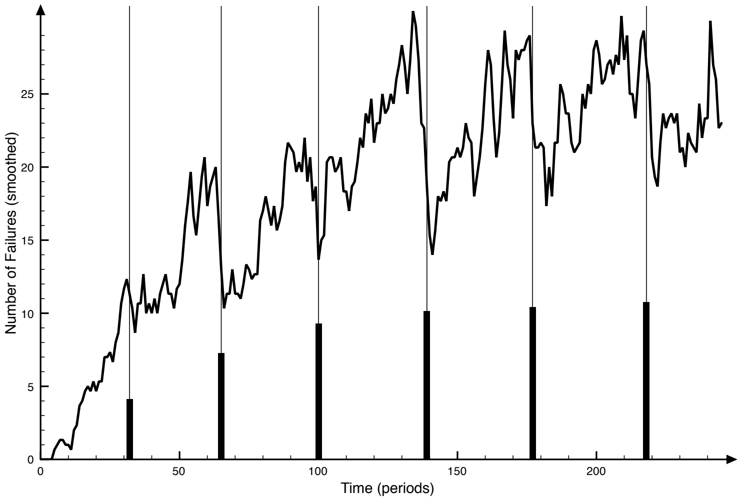


Figure 5.6: Plot of the total failures in each period during a typical simulation run. The vertical bars represent replenishments.

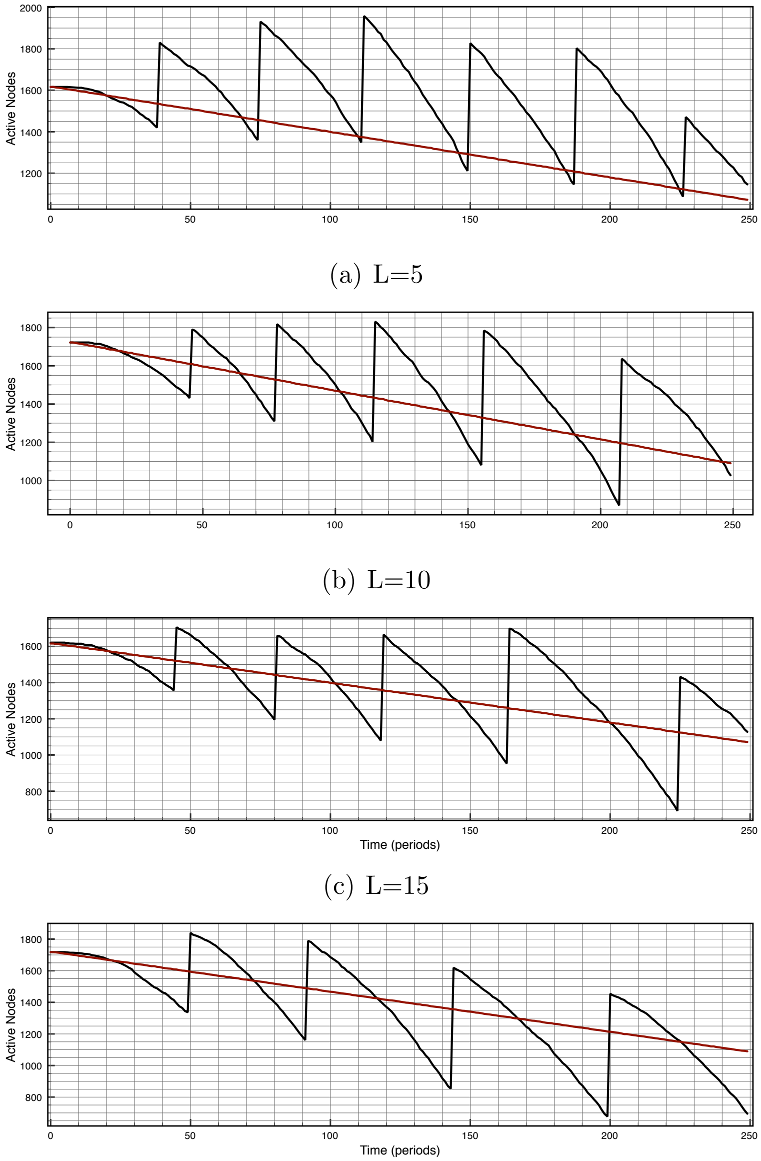
### 5.5.1 Effect of replenishments on failure rates

Each time a replenishment is delivered to the network area, nodes that have limited energy resources due to assuming the role of clusterhead will be relieved of this role as new nodes that have not yet been assigned a clusterhead role will be more likely to do so (this is a function of the clusterhead selection protocol). Therefore, each replenishment has an effect on the failure process. Without accounting for this effect, the LTF predictions near the time of replenishment will be subject to error, and the failure process model parameters will become unstable. Figure 5.6 shows the effect of replenishments on the failure process. The figure shows the aggregate number of failures in each period (summed over all bands). Beneath the graph of the failure process are bars that represent the times a batch of nodes was delivered. The height of the bars indicate the number of nodes delivered (divided by 100). The failure trend appears to remain relatively stable, while the base level abruptly drops by some amount right after replenishment. In order to account for the effect of the replenishments, we introduce an additional term to the base level update equation, $-\omega y_0 u[t - t_0]$ where $y_0$ is the number of nodes scheduled to be delivered at time $t_0$, $u$ is the unit step function, and $\omega$ is a scaling factor whose value is obtained through offline simulation. The additional term has the effect of reducing the expected number of failures beyond the time of replenishment. Once an order has been placed, the controller factors the reduction in the failure rate in its forecast of the lead-time failures, and the result is that the controller places larger orders less often, avoiding the large fixed cost for deployment.

## 5.6 Simulations

To test the replenishment strategy, we implemented the replenishment controller in the MATLAB simulation from Chapter 3. The values for the smoothing parameters were computed using a separate series of simulation runs where we allowed the network to run to failure. We also fixed the value of $a$, the allocation parameter from Sec. 5.5 to a constant value of 0.25. The simulation topology was similar to Figure 3.11. We used 6 bands and an initial deployment of 1700 nodes. The replenisher computed forecasts of failures for 20 periods into the future at each stage, hence the dynamic program was solved for a 20-period planning horizon. In order to see the effects of varying lead-times, we ran the simulation for lead-time values of 5, 10, 15, and 20. In each simulation, the mission lifetime was set to 250, more than 3 times the maximum lifetime of the initial deployment. We set the penalty costs to $d = 3$ and $h = 1.5$, and the fixed cost $K$ was set to 500.

Figures 5.7(a)-5.7(c) show the total number of nodes active in the network with respect to time for 5, 10, 15 and 20 period lead-times and a 20-period planning horizon. The straight lines represent the sum of the target levels for all bands over time, $\sum_{j=1}^{6} \rho_{jt}$.



(a) L=5

(b) L=10

(c) L=15

(d) L=20

Figure 5.7: Simulation results: node level of the entire network over time, with replenishments. The black oscillating curves are the total number of active nodes in the network. The straight red line is the sum of the target levels over all bands.

Figure 5.8 shows the number of active nodes in each band scaled by the area of each band (the node densities) over time. The allocation levels given to each band are evident from the “jumps” in the density. Note that at the end of the mission lifetime, the bands densities are near equal (*i.e.*, the network ends in a uniform deployment).

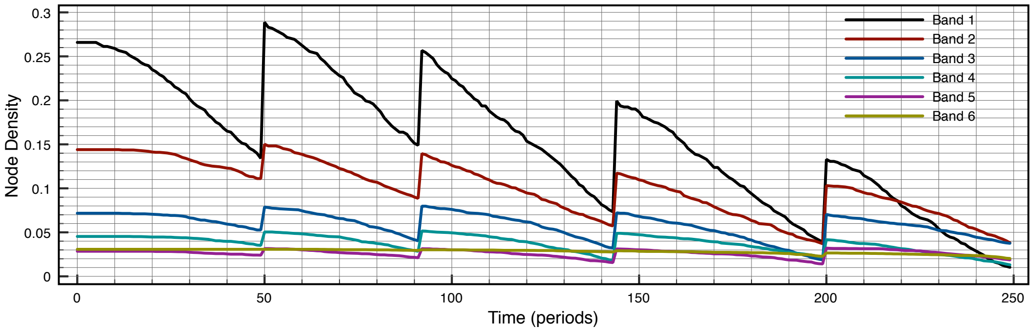


Figure 5.8: Node densities in each band over time with allocations.

## 5.7 Discussion of Replenishment Simulations

In this chapter, we have extended the deployment strategy from Chapter 4, where we have added a replenishment controller in order to extend the lifetime of an initial deployment to meet a mission lifetime. The use of a replenishment strategy is justified by the fact that, as we add additional nodes to the initial deployment, the increase in the lifetime of the network begins to flatten out. Moreover, the quality of the network, in terms of capacity and interference, begins to decrease. We have shown that the problem of replenishing networks with differential node densities is similar to the problem of multi-location inventory control. We showed that the formulation of this type of problem requires a dynamic program with a large state space, and that the non-stationarity of the failure rates further increases the complexity of the problem. Using a technique introduced by Zipkin, we reduced the problem size to a single dimension. The technique requires us to apply the *allocation assumption*, which holds that, given an order arriving from $L$ periods in the past, the order size will be sufficient to replenish each location in such a way that the probability that any of the bands will fail over the next lead-time is roughly equal. This assumption holds for our problem the initial deployment and the target levels for replenishment are obtained from the deployment strategy from chapter 4. The initial deployment strategy specifies a differential deployment that aims to balance the residual energy in each band so that a minimum amount of energy is wasted at the end of the network.

Another reason that the allocation assumption holds in our case is the way that nodes are allocated to bands in the myopic allocation strategy. The myopic allocation considers both the deviation at the current time period from band target levels and the fraction of failures over the past lag time in each band.

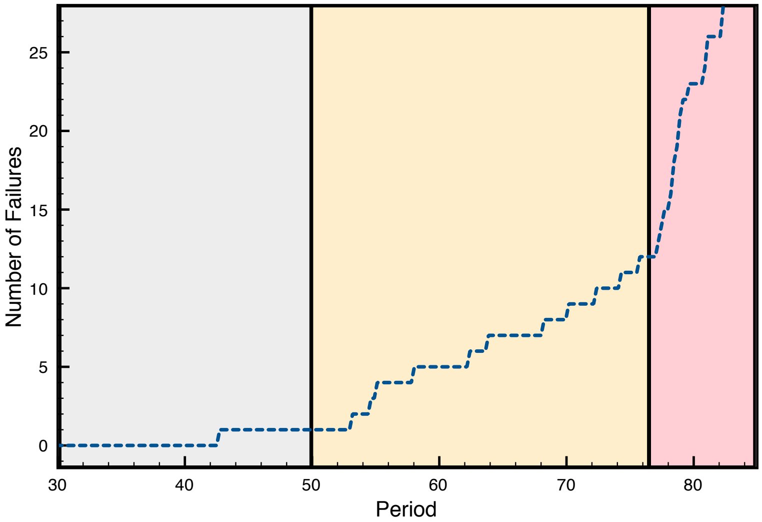


Figure 5.9: An example of phase transitions in the node failure rate for a single band.

Figure 5.9 shows the cumulative failures of a single band over time. The graph exhibits what appears to be three phases of failure (each phase is boxed in the figure): (1) an initial phase, where no nodes are failing, (2) a “linear” phase, where nodes begin failing at an approximately constant rate, and (3) a critical phase, where nodes are failing with an exponentially increasing rate. The interpretation of the transition between the linear and critical phases is that, at the beginning of the critical phase, the number of nodes in the band is near the minimum levels set for connectivity guarantees; thus, messages sent from a CH must traverse many more hops to exit the band, and so more energy is being consumed in the band. If bands were allowed to enter this critical phase, the exponential smoothing method used to forecast failures would not be able to track the failure trend accurately, insufficient nodes would be ordered to balance the bands, and the network would fail prematurely. By considering the fraction of failures as well as the target level deviations, we are reacting to both the long-run failures and the recent changes in the failure trends. The result is that the myopic allocation helps to prevent bands from exiting the linear phase.

We tested the replenisher for various lead-time values. The effect of a longer lead-time appears in Figure 5.7(d) for $L$=20. In this case the lead-time is equal to the planning horizon. We can see from the figure that, in this case, the replenisher consistently under-ordered, resulting in node levels well below the targets. A trend between the lead-time and the amounts that node levels fall below the targets becomes clear as we look at Figures 5.7(a)-5.7(d), which show the number of active nodes over time for $L = 5$, 10, 15, and 20. Since a penalty is incurred for node levels below the target in each period, the effects of the lead-time can be easily seen by comparing the total cost (penalties and actual costs) of a replenishment versus the lead-time as shown in Figure 5.10.

In Section 5.3 we derived a cost function $g_t$, which represents the minimum total expected ordering and penalty costs incurred in periods $t$ through $T$. In our implementation, $g_t$ is computed in each period over the planning horizon, since the non-stationarity of the failure process precludes us from planning out until the end

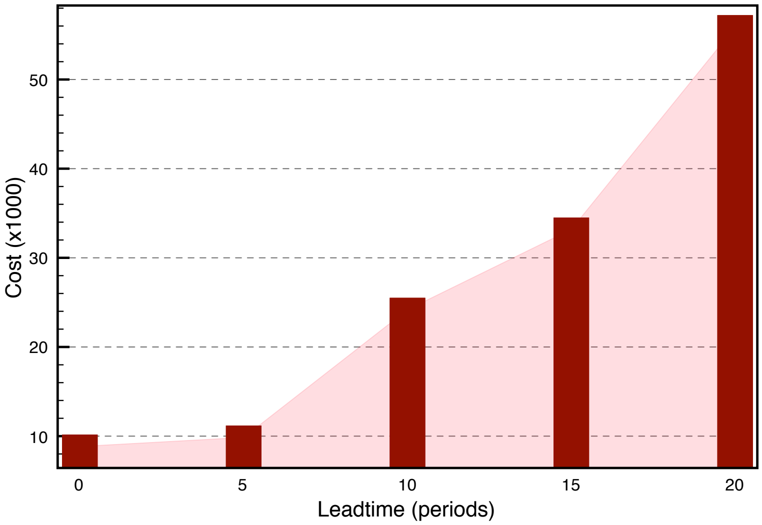


Figure 5.10: The effect of the lead-time on the total cost of the mission.

of the mission lifetime. The value of $y_t$ (the order) that minimizes $g_t$ is the optimal batch size for period $t$. Figure 5.11 shows two instances of $g_t$ from our simulations. The dashed line represents the case where the optimal batch size is zero, thus no order is placed. The solid line represents a case where the minimum value for $g_t$ corresponds to $y_t = 477$. The jump in the cost functions at $y_t = 0$ corresponds to the fixed cost for ordering and deploying the nodes. If the fixed cost is increased, the time that an order is placed will be delayed, and the size of the order will increase. The values of the penalty costs $d$ and $h$ will also affect the frequency of orders and the size of the batches. Therefore, given a fixed lead-time, it may be possible to simulate a network prior to deployment in order to find penalty values that will result in better performance.

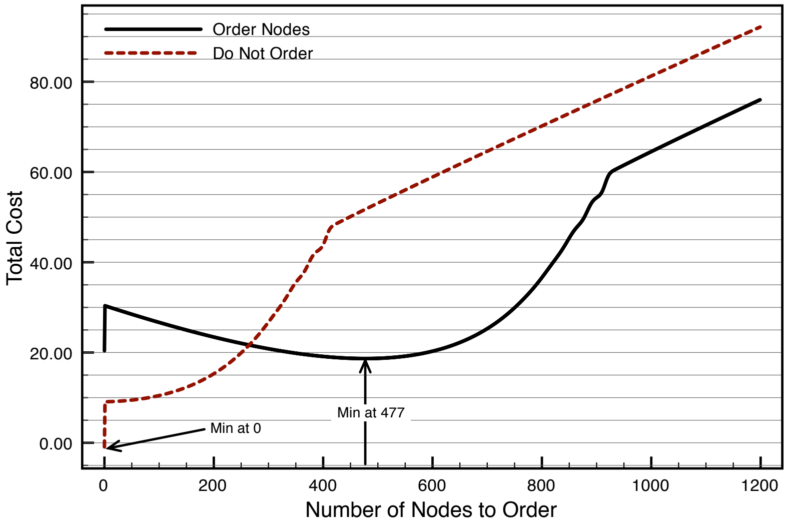


Figure 5.11: Characteristic shape of the cost function, $g_t$. The minimum of the cost function specifies the number of nodes to order. The dashed line shows a curve where the optimal decision is not to order any nodes. The solid line shows a minimum at 477, so the decision is to order nodes.

## 5.8 Comparison with the Full Multi-band DP Controller

In this section we compare the total costs, including the fixed cost for deployment, the penalties for surpluses and shortages of nodes, and the number of nodes delivered, for a three-band deployment using both the multi-band dynamic programming controller from Eqs. 5.2 and 5.3, and the proposed approximation given in Eq. 5.12. Recall that the motivation for introducing the relaxed version of the full dynamic programming problem was due to the potentially large dimensionality of the state space. The dimensionality of the full DP problem grows exponentially with the mission time, the number of bands, and the lead-time. The approximate controller reduces the problem to a single-dimension dynamic program with zero lead-time by separating the computation of the optimal number of nodes to order over the next lead-time and the allocation of these nodes among the bands once the order arrives.

This relaxation provides means that significantly fewer points need to be computed to obtain the optimal solution (in the sense that the number of nodes to order and the bands over which to allocate them minimizes the cost function in Eq. 5.12).

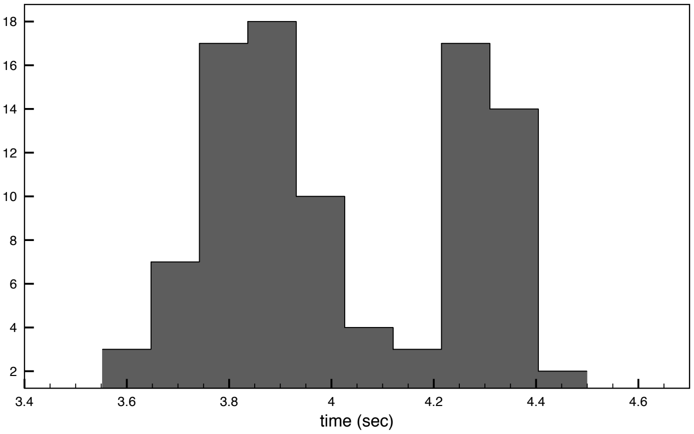
As an illustration, consider the number of feasible points that would have to be computed for the optimal allocation of *a given number of nodes* across 7 bands. The number of ways to allocate $Y$ nodes among $J$ bands is equivalent to the number of ways to write $Y$ as a sum of $J$ positive integers. This is known in combinatorics as the *Bars and Stars* problem, where one wants to find all the possible ways to partition $Y$ stars with $J$ bars. For example, one possible partition of 7 stars with 2 bars is $2 + 2 + 3 = 7$ or $\star\star \mid \star\star \mid \star\star\star$. The total number of possible ways to partition $Y$ stars with $J$ bars (or allocate $Y$ nodes to $J$ bands) is given by $\binom{Y-1}{J-1}$, so the number of points to compute grows very rapidly. For 7 bands and an order of 300 nodes, there are nearly 944 billion possible allocations. The approximate DP avoids this calculation when ordering nodes by approximating the *minimum cost* of of the allocation using only the number of nodes ordered and the sum of all node level deviations (Section 5.3.2). When a batch arrives the approximate DP controller allocates them according to the most recent information about node level deviations and the fraction of failures across bands. In the full DP approach, we would calculate the cost function for each of these possible allocations for every possible order size over $L$ time steps. By comparison, the relaxation of the full DP problem allows us to compute the optimal order size with respect to the aggregate lead-time failures as if there were only a single band, and then compute the best allocation once the nodes arrive.

There will certainly be a loss of optimality when using the approximate DP formulation; the goal of this section is to provide a cost comparison of the two approaches for a single deployment scenario. The increased costs incurred by the approximation are a result of a loss of information to the controller when it views all lead-time failures as an aggregate statistic. Specifically, there are occasions where node levels in one band are below the target while levels were above target in another band. These deviation values from the target levels are received by the controller as aggregates where the negative and positive levels have cancelled each other out, indicating that no bands are below target level. The effect is that the controller may delay an order, or place too small an order.

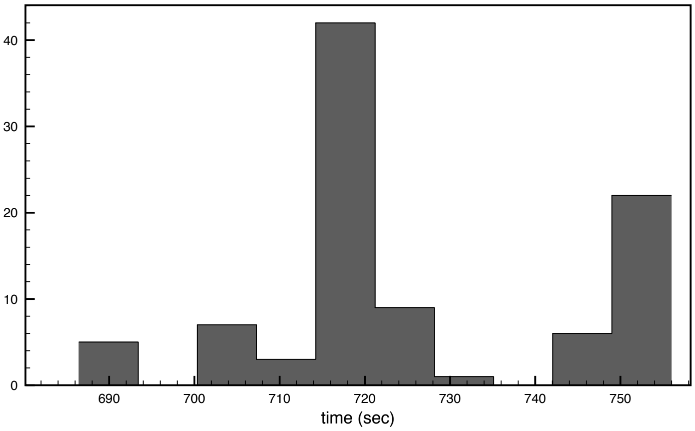
### 5.8.1 Simulation setup and cost comparison

To make the simulation times feasible, we chose a smaller scenario with three bands and an initial deployment of about 250 nodes to compare the costs between the full DP approach and the approximate controller. The mission lifetime was set to 200 rounds. The two controllers each begin with the same initial deployment computed using the Dynamic deployment strategy from Chapter 4. They both apply the same lead-time failure forecasting method from Section 5.4. The full DP approach receives a forecast of the distribution of lead-time failures for each band and each time step over the planning horizon. The approximate controller uses an aggregate distribution of the sum of lead-time failures over all bands and for the entire planning horizon. Both controllers can order up to 70 nodes in a batch. The fixed cost $K$ was set to 300, and the shortage ($d$) and surplus ($h$) costs were set to 0.9 and 0.1, respectively. The cost of a sensor ($c_s$) was set to 1.

We ran 80 simulations for the comparisons. Each controller was evaluated 10 times for each of the four lead-times, L = 5, 10, 15, and 20. The distribution of the average times to compute a solution for a single round for the approximate controller is shown in Figure 5.12(a). The approximate controller took about 4 seconds to compute a single order and allocation solution, including the time required to estimate the lead-



(a) Histogram of the average times to compute the optimal order size and allocation for approximate DP controller with myopic allocation.

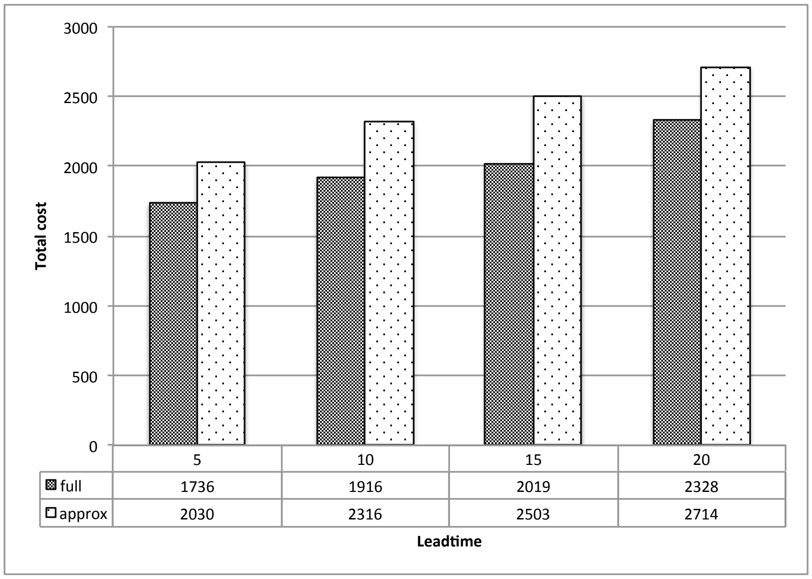


(b) Histogram of the average times to compute the optimal order size and allocation for full DP controller.

Figure 5.12: Comparison of the average times to compute the minimum cost order and allocation for the approximate DP and the full DP controllers. Each data point is the average for one simulation. The full DP controller typically requires more than 700 seconds to compute the solution for a single round in a three-band problem. The approximate DP controller requires an average of 4 seconds per round.

time failure distribution. The average time distribution for the full DP controller is presented in Figure 5.12(b). The full DP controller required over 700 seconds per round to evaluate the cost function over the 3 bands and a 20 period planning horizon;

each full DP simulation took approximately 40 hours to complete.



| | 5 | 10 | 15 | 20 |
|---|---|---|---|---|
| ▨ full | 1736 | 1916 | 2019 | 2328 |
| □ approx | 2030 | 2316 | 2503 | 2714 |

Figure 5.13: Comparison of the total costs (ordering costs and node level penalties) for the full DP controller and the approximate controller over lead-times of 5, 10, 15, and 20. The full DP approach is shown in dark columns.

The costs incurred by the replenishment controller include the ordering costs and penalties related to node levels in each round. The ordering cost as a function of the order size is given by

$$C_t(y_t) = (y_t > 0)K + c_s y_t.$$

The penalties across all bands in round $t$ are

$$Q_t(\mathbf{x}_t) = \sum_{j=1}^{J} h\left(x_{jt} - \rho_{jt}\right)^+ + d\left(x_{jt} - \rho_{jt}\right)^- .$$

The total cost is the sum of all of the ordering costs and penalty costs over the 200 round simulation time. Both controller types are evaluated using the same metric.

Figure 5.13 shows the average total costs over 10 simulations for each lead-time. The dark columns indicate the average cost of the full DP controller and the light columns are the costs for the approximate controller. In these simulations, the approximate controller is between 17% and 24% more costly than the full DP controller. Both controllers show an increasing total cost with respect to lead-time. It is not clear from these results whether the costs increase at the same rate as the lead-time increases.

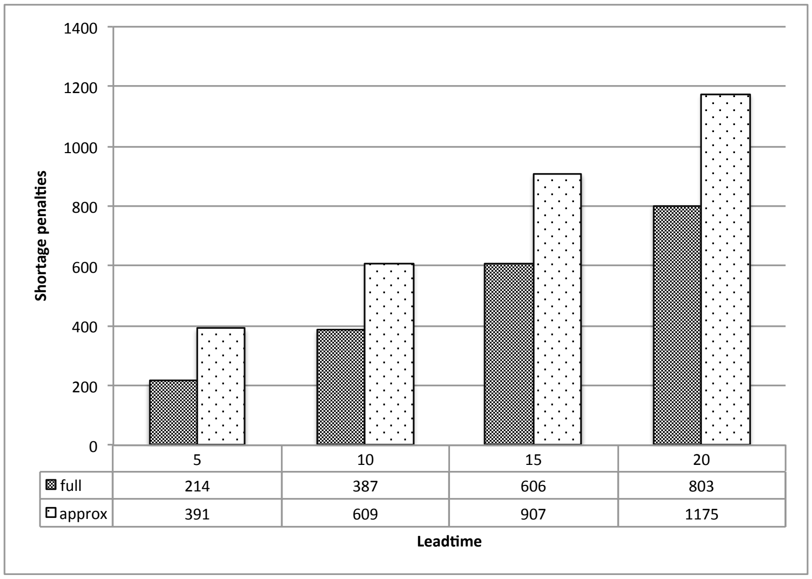


Figure 5.14: Comparison of the shortage penalties for the full DP controller and the approximate controller over lead-times of 5, 10, 15, and 20. The full DP approach is shown in dark columns.

Figure 5.14 shows the penalties incurred by both controllers for having node levels below the target level. The approximate controller shortage penalties are consistently higher than for the full DP controller. The rate of increase over the lead-times also appears higher for the approximate controller. Figure 5.15 shows the average shortage and surplus penalties as a percent of the total cost of the mission. Here we can see that the shortage penalties for the approximate controller are always a higher percentage of the total mission cost than for the full DP controller. However, the shortage costs are consistently a lower percentage of the total costs for the approximate controller than the full DP controller. These observations suggest that the approximate controller is either under-ordering, not ordering early enough, or not ordering as often as the full controller does. However, both controllers show the same trend as the lead-time increases. At shorter lead-times, both controllers pay a larger percentage of penalties for being above target thresholds. Then, as lead-times (and total costs) increase, both controllers pay a larger percentage of the total cost to shortages.

## 5.9 Cost Comparison for the Full DP and the Approximation

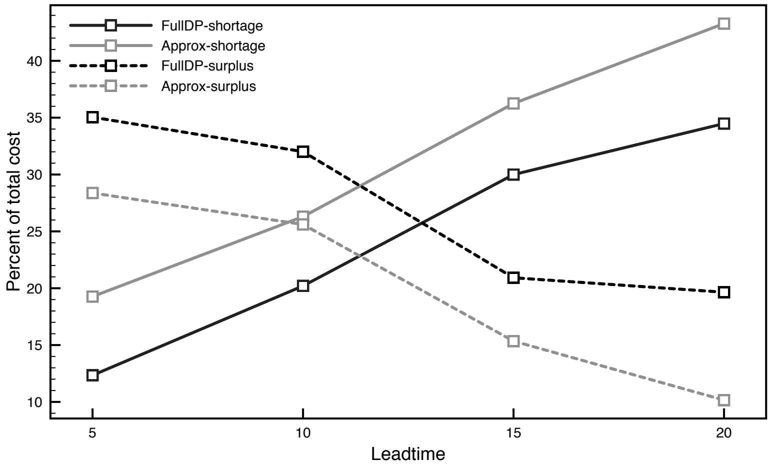
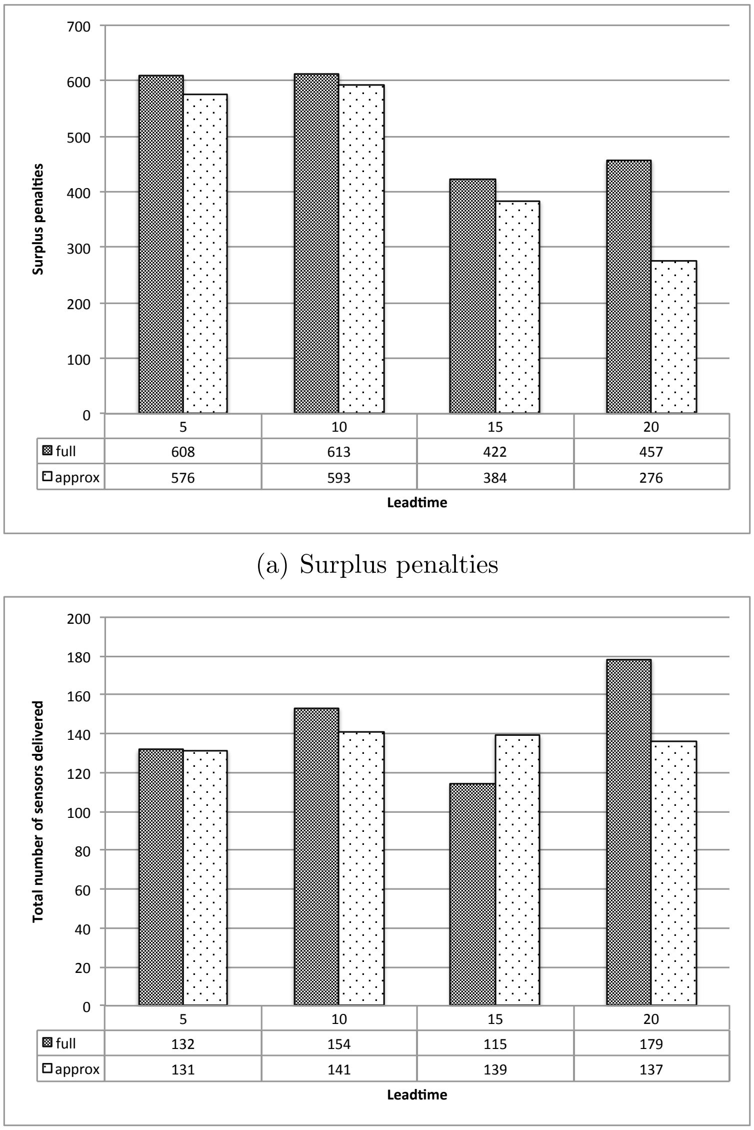


Figure 5.15: Comparison of the percentage of the total costs from shortages and surpluses for the full DP controller and the approximate controller over lead-times of 5, 10, 15, and 20.

We conclude that the difference in costs between the approximate controller and the full DP controller are primarily due to shortage costs. Figures 5.16(a) and 5.16(a) show the average surplus penalties and the average number of nodes deployed for both controllers. The average surplus penalties for both controllers is nearly the same with the exception of the 20 round lead-time (where the DP controller has higher surplus penalties). The number of sensors added to the network over the mission lifetime is also very similar. In the 15-round lead-time, the average number of sensors deployed is higher for the approximate controller than the DP controller, even though the average shortage penalties are lower for the full DP controller. It appears that the full DP controller is reducing costs over the approximation by ordering nodes sooner than the approximate controller.



(a) Surplus penalties

(b) Number of nodes added to the network

Figure 5.16: Comparison of surplus penalties and the number of nodes added to the network over the mission lifetime for the full DP controller and the approximate controller over leadtimes of 5, 10, 15, and 20. The full DP approach is shown in dark columns.

# 6. Conclusion

Over the past 15 years, the concept of Wireless Sensor Networks (WSNs) has invited a large amount of research. This research addressed a diverse set of topics related to wireless communications, large-scale modeling and simulation, distributed algorithm design, and power-aware versions of routing protocols, MAC protocol designs, time-synchronization, and signaling. Much of the appeal of WSN research derives from the elimination of some traditional constraints that are present in network research. Unlike typical wireless data networks, WSNs are not subject to the requirements of application and transport layer standardization such as TCP/IP. WSNs also introduce an interesting new set of issues pertaining to sensor coverage of a monitored area, event detection probability, the amount of energy consumed in an idle state, and the number of sensor nodes required to monitor an area. The relatively short life-span of a (presumably inexpensive) sensor node also leads to new and interesting definitions of *network state*. In wireless communication networks, the network state is typically given by metrics related to throughput, for example, queue length and channel state. In a WSN, the network state is often given in terms of the active set of nodes, or the set of awake nodes to control network coverage, connectivity, and lifetime. Thus, the questions one must ask when designing a controller for a sensor network are often about how many nodes to deploy, and where they should be deployed. Other related challenges are how these nodes should organize to form a network, and how the data should be routed to the base station. Another unique feature of sensor networks is the inherent many-to-one traffic pattern. Whether the WSN is event-driven, periodic, or user-driven (a network user requests information from the WSN), the data must be gathered from disparate physical locations and delivered to a single terminal node.

In this study, we have addressed problems associated with deploying a monitoring

WSN over a large area. These problems, each of which are related, include maximizing the active lifetime of the network (given a specified number of nodes to deploy), reducing the amount of wasted energy (the amount of energy remaining in the network upon failure), addressing the biased energy consumption rate (BECR) phenomenon inherent in WSNs, and ensuring connectivity and coverage as the network experiences node failures. We also address the problem of replenishing nodes after the initial deployment in cases where a single deployment is not feasible for longer lifetime requirements.

Some studies have addressed these issues by designing network architectures and algorithms whose focus is reducing the number of messages that are passed through the network. These include the efficient clustering algorithms developed in [Bandyopadhyay and Coyle 2003b] and [Heinzelman et al. 2000]. Other authors addressed the problem of unbalanced energy consumption and network lifetime through the addition of relay nodes with more powerful transmitters that can reach the base station in a single (or a few) hops. More recent work, emphasizing the role that BECR plays in determining the lifetime of the network, has focused on the use of a differential deployment in order to balance energy consumption in the network and extend lifetime. These include strategies for adding relay nodes in a clustered architecture [Xu et al. 2005], deploying sensor nodes in a flat architecture [Wang et al. 2006], and deploying sensor nodes that may be either clusterheads or non-clusterheads Liu [2006]. In our survey of the literature, we have not found any work that addresses the problem of replenishing WSNs.

We have focused on networks deployed with homogenous node types; all nodes act as either a clusterhead node or a non-clusterhead node. We further considered clusterhead selection similar to the LEACH protocol, where nodes act as a clusterhead with a specified probability. We have derived an expression for the expected lifetime of a network as a function of the specified probabilities of being a clusterhead and the density of nodes in annular regions surrounding the base station. We also derived an expression for the wasted energy remaining in the network at the time of network failure. These two expressions were employed as objective functions in an optimization problem (maximiziation of network lifetime) with decision variables $\mathbf{p} = \{p_j\}_{j=1}^{J}$, the vector whose elements are the specified clusterhead probabilities for each annular region, and $\boldsymbol{\lambda} = \{\lambda_j\}_{j=1}^{J}$, the node density in each region. This optimization problem is computed with respect to constraints on capacity, coverage, and connectivity, which we have derived using results from the literature. The result is, for given constraints on coverage, capacity, the number of available nodes for deployment, and the size of the region of deployment, the optimization program will provide an optimal node densities, $\boldsymbol{\lambda}$ and clusterhead probabilities $\mathbf{p}$ that maximizes the lifetime of the initial deployment. We have shown that this solution provides an initial deployment that will extend the lifetime of the network (by an average of 20%) beyond an optimal differential deployment where the clusterhead density is the same in all regions, as well as a uniform deployment (with an average percent increase of 200%). Furthermore, our approach minimizes the wasted energy remaining in the network. Thus, our approach to computing the optimal deployment for a given number of nodes improves existing techniques with no additional cost and little added complexity (all that is required is to pre-program nodes with the optimal values of $\mathbf{p}$, or provide a means for nodes to estimate their distance from the base station and select the appropriate value from a table).

Another benefit of our approach is that it generalizes work that has been done on computing optimal clusterhead densities in a clustered architecture. Usually, the decision of whether or not to use a clustered architecture is first made, then the optimal clusterhead densities are computed. The approach described in this study may be used to determine whether a clustered architecture is optimal, and will provide the appropriate densities. Furthermore, our approach provides a deployment strategy that combines both flat and hierarchical architectures; the output of the optimization problem specifies a *hybrid deployment* when appropriate.

Finally, we have provided a method for determining the order size and allocation of nodes over all regions given an initial deployment and a mission lifetime. We have shown that the problem of determining the orders and allocations of sensor nodes to replenish the network is analogous to a *single warehouse multi-location inventory replenishment problem* extensively studied in the Operations Research literature. Our results show that, for moderate lead-times, the *allocation assumption* is valid for this problem, and the full dynamic program required for solving the problem can be approximated by a single-dimensional problem without lead-times. We show a relationship between the lead-time and the cost of the network deployment. Our approach provides a minimum cost replenishment strategy that can easily be computed in a small fraction of the time that a dynamic programming approach requires. The solution minimizes the cost of keeping the network active until the end of the mission, subject to performance constraints (coverage and capacity).

# Index

# Bibliography

N. Abramson. The aloha system: Another alternative for computer communications. In *Proceedings of the November 17-19, 1970, Fall Joint Computer Conference*, pages 281–285. ACM, 1970.

G.-S. Ahn, S. G. Hong, E. Miluzzo, A. T. Campbell, and F. Cuomo. Funneling-mac: a localized, sink-oriented mac for boosting fidelity in sensor networks. In *Proceedings of the 4th international conference on Embedded networked sensor systems*, pages 293–306. ACM, 2006.

I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *Computer Networks*, 38(4):393–422, Jan 2002.

I. F. Akyildiz and M. C. Vuran. *Wireless sensor networks*, volume 4. John Wiley & Sons, 2010.

Z. Alliance. Zigbee Specification Document 053474r06. Technical report, 2005.

A. Anandkumar, L. Tong, and A. Swami. Optimal node density for detection in energy-constrained random networks. *IEEE Transactions on Signal Processing*, 56(10, Part 2):5232 – 5245, Oct 2008.

S. Axsater. *Inventory control*. Springer Verlag, 2006.

S. Bandyopadhyay and E. Coyle. An energy efficient hierarchical clustering algorithm for wireless sensor networks. *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies. IEEE*, 3:1713–1723 vol.3, 2003a.

S. Bandyopadhyay and E. Coyle. An energy efficient hierarchical clustering algorithm for wireless sensor networks. *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies. IEEE*, 3:1713–1723 vol.3, 2003b.

P. Barooah and A. Swami. Recursive time-synchronization in sensor networks. *Military Communications Conference, 2008. MILCOM 2008. IEEE*, pages 1 – 7, Oct 2008.

R. Bellman. On a routing problem. Technical report, DTIC Document, 1956.

D. Bertsekas. *Dynamic programming and optimal control*. Athena Scientific Belmont, MA, 1995.

C. Bettstetter. On the connectivity of ad hoc networks. *The Computer Journal*, 47 (4):432–447, 2004.

M. Bhardwaj and A. Chandrakasan. Upper bounds on the lifetime of wireless sensor networks. *Proc. of IEEE International Conference on Communications (ICC)*, 1, 2001.

M. Bhardwaj and A. Chandrakasan. Bounding the lifetime of sensor networks via optimal role assignments. In *IEEE INFOCOM*, volume 3, pages 1587–1596, 2002.

R. Bruno, M. Conti, and E. Gregori. Optimization of efficiency and energy consumption in p-persistent csma-based wireless lans. *IEEE Transactions on Mobile Computing*, 1(1):10–31, 2002.

Y. Chen and Q. Zhao. On the lifetime of wireless sensor networks. *Communications Letters*, Jan 2005.

Y. Chen and Q. Zhao. Law of Sensor Network Lifetime and Its Applications. *Wireless Sensor Networks: Signal Processing and Communications Perspectives*, page 93, 2007.

Chipcon. CC2420 2.4 GHz IEEE 802.15. 4/ZigBee-ready RF Transceiver. Technical report, 2004.

A. Clark and H. Scarf. Optimal policies for a multi-echelon inventory problem. *Management science*, 6(4):475–490, 1960.

T. Clouqueur, V. Phipatanasuphorn, P. Ramanathan, and K. Saluja. Sensor deployment strategy for target detection. *WSNA '02: Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, Sep 2002.

R. Corless, G. Gonnet, D. Hare, D. Jeffrey, and D. Knuth. On the LambertW function. *Advances in Computational mathematics*, 5(1):329–359, 1996.

D. Dorsey and M. Kam. Non-uniform deployment of nodes in clustered wireless sensor networks. *Information Sciences and Systems, 2009. CISS'09.*, Mar 2009.

D. Dorsey and M. Kam. Optimal deployment and replenishment of monitoring wireless sensor networks. In *Military Communications Conference, 2010 - MILCOM 2010*, pages 136–141, 2010.

O. Dousse, P. Thiran, and M. Hasler. Connectivity in ad-hoc and hybrid networks. In *INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 2, pages 1079–1088. IEEE, 2002.

O. Dousse, F. Baccelli, and P. Thiran. Impact of interferences on connectivity in ad hoc networks. *Networking, IEEE/ACM Transactions on*, 13(2):425–436, 2005.

E. Duarte-Melo and M. Liu. Analysis of energy consumption and lifetime of heterogeneous wireless sensor networks. *Global Telecommunications Conference, 2002. GLOBECOM '02. IEEE*, 1:21 – 25 vol.1, Oct 2002.

E. Duarte-Melo and M. Liu. Data-gathering wireless sensor networks: organization and capacity. *Computer Networks*, 43(4):519–537, 2003.

E. Duarte-Melo and M. Liu. Field gathering wireless sensor networks. *Mobile*, Jan 2006.

E. Duarte-Melo, M. Liu, and A. Misra. Lifetime bounds, optimal node distributions and flow patterns for wireless sensor networks. Technical report, 2003.

G. Eppen and L. Schrage. Centralized ordering policies in a multi-warehouse system with lead times and random demand. *Multi-level production/inventory control systems: Theory and practice*, 16:51–67, 1981.

A. Federgruen and P. Zipkin. Approximations of dynamic, multilocation production and inventory problems. *Management Science*, 30(1):69–84, 1984.

R. Fourer, D. Gay, and B. Kernighan. *AMPL: A mathematical programming language.* AT&T Bell Laboratories, 1989.

A. Ghosh and S. Das. Coverage and connectivity issues in wireless sensor networks: A survey. *Pervasive and Mobile Computing*, Jan 2008.

P. E. Gill, W. Murray, and M. A. Saunders. Snopt: an sqp algorithm for large-scale constrained optimization. *SIAM J. Optim.*, 12(4):979–1006 (electronic), 2002.

A. Giridhar and P. Kumar. Maximizing the functional lifetime of sensor networks. In *Proceedings of the 4th international symposium on Information processing in sensor networks.* IEEE Press Piscataway, NJ, USA, 2005.

M. Gun, R. Kosar, and C. Ersoy. Lifetime optimization using variable battery capacities and nonuniform density deployment in wireless sensor networks. *22nd International International Symposium on Computer and Information Sciences, ISCIS.*, pages 1 – 6, Oct 2007.

P. Gupta and P. Kumar. Critical power for asymptotic connectivity. In *Proceedings of the 37th IEEE Conference on Decision and Control, 1998.*, volume 1, 1998.

P. Gupta and P. Kumar. The capacity of wireless networks. *IEEE Transactions on Information Theory*, 46(2):388 – 404, Mar 2000.

P. Hall. *Introduction to the theory of coverage processes.* John Wiley & Sons Inc, 1988.

D. Hamel, M. Chwastek, B. Farouk, M. Kam, and K. Dandekar. A computational fluid dynamics approach for optimization of a sensor network. In *2006 IEEE International Workshop on Measurement Systems for Homeland Security, Contraband Detection and Personal Safety. Alexandria, VA. USA: IEEE*, pages 38–42, 2006.

W. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. *System Sciences, 2000. Proceedings of the 33rd Annual Hawaii International Conference on*, page 10 pp. vol.2, Jan 2000.

W. Heinzelman, A. Chandrakasan, and H. Balakrishnan. An application-specific protocol architecture for wireless microsensor networks. *IEEE Transactions on Wireless Communications*, 1(4):660 – 670, Oct 2002.

W. R. Heinzelman, J. Kulik, and H. Balakrishnan. Adaptive protocols for information dissemination in wireless sensor networks. In *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, pages 174–185. ACM, 1999.

C. Holt. Forecasting trends and seasonals by exponentially weighted moving averages. *ONR Memorandum*, 52, 1957.

Z. Hu and B. Li. Fundamental performance limits of wireless sensor networks. *Ad Hoc and Sensor Networks*, Jan 2004a.

Z. Hu and B. Li. On the fundamental capacity and lifetime limits of energy-constrained wireless sensor networks. In *The 10th IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS 2004)*, pages 2–9, 2004b.

C. Huang and Y. Tseng. The coverage problem in a wireless sensor network. *Mobile Networks and Applications*, 10(4):519–528, 2005.

A. Iranli, M. Maleki, and M. Pedram. Energy efficient strategies for deployment of a two-level wireless sensor network. *Low Power Electronics and Design, 2005. ISLPED '05. Proceedings of the 2005 International Symposium on*, pages 233–238, 2005.

J. F. C. Kingman. *Poisson processes*, volume 3. Oxford university press, 1992.

L. Kleinrock and J. Silvester. Optimum transmission radii for packet radio networks or why six is a magic number. In *Proceedings of the IEEE National Telecommunications Conference*, volume 4, pages 1–4, 1978.

L. Kleinrock and F. Tobagi. Packet Switching in Radio Channels: Part I–Carrier Sense Multiple-Access Modes and Their Throughput-Delay Characteristics. *Communications, IEEE Transactions on*, 23(12):1400–1416, 1975.

K. Kredo and P. Mohapatra. Medium access control in wireless sensor networks. *Computer Networks*, 51(4):961–994, 2007.

K. Krishnan and V. Rao. Inventory control in N warehouses. *Journal of Industrial Engineering*, 16(3):212–215, 1965.

C. Li, M. Ye, G. Chen, and J. Wu. An energy-efficient unequal clustering mechanism for wireless sensor networks. In *IEEE International Conference on Mobile Adhoc and Sensor Systems Conference, 2005*, page 8, 2005.

J. Li and G. Al Regib. Energy-constrained distributed estimation in wireless sensor networks. In *Proc. of Military Communications Conference (MILCOM 2007)*, pages 29–31, 2007.

B. Liu and D. Towsley. A study of the coverage of large-scale sensor networks. *Mobile Ad-hoc and Sensor Systems*, Jan 2004.

S.-C. Liu. A lifetime-extending deployment strategy for multi-hop wireless sensor networks. *Proceedings of the 4th Annual Communication Networks and Services Research Conference, CNSR 2006.*, page 8, Apr 2006.

Y. Liu, H. Ngan, and L. M. Ni. Power-aware node deployment in wireless sensor networks. *Sensor Networks, Ubiquitous, and Trustworthy Computing, 2006. IEEE International Conference on*, 1:128– 135, 2006.

D. Marco, E. Duarte-Melo, M. Liu, and D. Neuhoff. On the many-to-one transport capacity of a dense wireless sensor network and the compressibility of its data. *Lecture Notes In Computer Science*, Jan 2003.

M. McKelvin, M. Williams, and N. Berry. Integrated radio frequency identification and wireless sensor network architecture for automated inventory management and tracking applications. *Proceedings of the 2005 conference on Diversity in computing*, Jan 2005.

S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M. Srivastava. Coverage problems in wireless ad hoc sensor networks. In *IEEE INFOCOM*, volume 3, pages 1380–1387, 2001.

V. Mhatre, C. Rosenberg, D. Kofman, R. Mazumdar, and N. Shroff. A minimum cost heterogeneous sensor network with a lifetime constraint. *IEEE Transactions on Mobile Computing*, 4(1):4 – 15, Jan 2005. doi: 10.1109/TMC.2005.2(410)4.

E. Osterweil, J. Polastre, M. Hamilton, A. Mainwaring, and D. Estrin. Habitat monitoring with sensor networks. *Communications of the ACM*, Jan 2004.

M. Penrose. On K-connectivity for a geometric random graph. *Random Structures and Algorithms*, 15(2), 1999.

T. Philips, S. Panwar, A. Tantawi, I. Center, and Y. Heights. Connectivity properties of a packet radio network model. *IEEE Transactions on Information Theory*, 35 (5):1044–1047, 1989.

G. Pottie. Hierarchical information processing in distributed sensor networks. In *1998 IEEE International Symposium on Information Theory, 1998. Proceedings*, 1998a.

G. J. Pottie. Wireless sensor networks. In *Information Theory Workshop, 1998*, pages 139–140. IEEE, 1998b.

F. Preparata and M. Shamos. *Computational geometry: an introduction.* Springer, 1985.

G. Rahmatollahi and G. Abreu. Closed-Form Hop-Count Distributions in Random Networks with Arbitrary Routing. *IEEE Transactions on Communications*, 60 (2):429–444, 2012.

R. Rajagopalan, P. Varshney, C. Mohan, and K. Mehrotra. Sensor placement for energy efficient target detection in wireless sensor networks: A multi-objective optimization approach. In *Proc. of Annual Conf. on Information Sciences and Systems*, 2005.

I. Rhee, A. Warrier, M. Aia, J. Min, and M. L. Sichitiu. Z-mac: a hybrid mac for wireless sensor networks. *IEEE/ACM Transactions on Networking (TON)*, 16 (3):511–524, 2008.

A. Ribeiro and G. Giannakis. Bandwidth-constrained distributed estimation for wireless sensor networks-Part I: Gaussian case. *IEEE Transactions on Signal Processing*, 54(3):1131, 2006.

A. Rowe, R. Mangharam, and R. Rajkumar. FireFly: A Time Synchronized Real-Time Sensor Networking Platform. *Wireless Ad Hoc Networking: Personal-Area, Local-Area, and the Sensory-Area Networks*, 2004.

B. Sadler. Fundamentals of energy-constrained sensor network systems. *IEEE Aerospace and Electronic Systems Magazine*, 20(8):17–35, 2005.

T. Schmid, R. Shea, Z. Charbiwala, J. Friedman, M. Srivastava, and Y. Cho. On the interaction of clocks, power, and synchronization in embedded sensor nodes. *submitted to ACM Transactions on Sensor Networks (TOSN)*, March 2009.

S. Servetto. On the feasibility of large scale wireless sensor networks. In *Proceedings of the Annual Allerton Conference on Communication Control and Computing*, volume 40, pages 127–136, 2002.

S. Shakkottai, R. Srikant, and N. Shroff. Unreliable sensor grids: coverage, connectivity and diameter. *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies. IEEE*, 2:1073 – 1083 vol.2, Jan 2003.

S. S. Shapiro and M. B. Wilk. *An analysis of variance test for normality(complete samples).* PhD thesis, Doctoral dissertation, Rutgers, The State University., 1964.

M. Sheldon, D. Chen, M. Nixon, and A. Mok. A practical approach to deploy large scale wireless sensor networks. *Mobile Adhoc and Sensor Systems Conference, 2005. IEEE International Conference on*, pages 8 pp. EP –, 2005.

E. Shih, S. Cho, N. Ickes, R. Min, A. Sinha, A. Wang, and A. Chandrakasan. Physical layer driven protocol and algorithm design for energy-efficient wireless sensor networks. *Proceedings of the 7th annual international conference on Mobile computing and networking*, pages 272–287, 2001.

T. Shu, M. Krunz, and S. Vrudhula. Power balanced coverage-time optimization for clustered wireless sensor networks. *MobiHoc '05: Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*, May 2005.

S. Simic and S. Sastry. Distributed environmental monitoring using random sensor networks. *Lecture Notes In Computer Science*, pages 582–592, 2003.

J. A. Stankovic. Wireless sensor networks. *IEEE Computer*, 41(10):92–95, 2008.

D. Stoyan, W. S. Kendall, J. Mecke, and L. Ruschendorf. *Stochastic geometry and its applications*, volume 2. Wiley Chichester, 1995.

F. Sun and M. Shayman. Prolonging network lifetime via partially controlled node deployment and adaptive data propagation in wsn. *Information Sciences and Systems, 2007. CISS '07. 41st Annual Conference on*, pages 226–231, 2007.

A. Swami, Q. Zhao, Y.-W. Hong, and L. Tong. *Wireless Sensor Networks: Signal Processing and Communications.* John Wiley & Sons, 2007.

V.-A. Truong. Approximation Algorithm for the Stochastic Multi-period Inventory Problem via a Look-Ahead Optimization Approach. *submitted to Mathematics of Operations Research*, 2012.

P. Wan and C. Yi. Asymptotic critical transmission radius and critical neighbor number for k-connectivity in wireless ad hoc networks. *Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing*, pages 1–8, 2004.

D. Wang, Y. Cheng, Y. Wang, and D. P. Agrawal. Lifetime enhancement of wireless sensor networks by differentiable node density deployment. *Mobile Adhoc and Sensor Systems (MASS), 2006 IEEE International Conference on*, pages 546–549, 2006.

F. Wang and J. Liu. Networked Wireless Sensor Data Collection: Issues, Challenges, and Approaches. *Communications Surveys & Tutorials, IEEE*, 13(4):673–687, 2011.

Q. Wang and W. Yang. Energy consumption model for power management in wireless sensor networks. In *Fourth Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, pages 142–151, 2007.

Q. Wang, K. Xu, G. Takahara, and H. Hassanein. Locally optimal relay node placement in heterogeneous wireless sensor networks. *Global Telecommunications Conference, 2005. GLOBECOM '05. IEEE*, 6:5 pp. – 3553, Nov 2005.

Q. Wang, K. Xu, G. Takahara, and H. Hassanein. Device placement for heterogeneous wireless sensor networks: Minimum cost with lifetime constraints. *IEEE Transactions on Wireless Communications*, 6(7):2444 – 2453, Jul 2007.

Y. Wang, C. Hu, and Y. Tseng. Efficient placement and dispatch of sensors in a wireless sensor network. *IEEE Transactions on Mobile Computing*, Jan 2008.

G. Werner-Allen, J. Johnson, M. Ruiz, J. Lees, and M. Welsh. Monitoring volcanic eruptions with a wireless sensor network. *Wireless Sensor Networks, 2005. Proceeedings of the Second European Workshop on*, pages 108 – 120, Jan 2005. doi: 10.1109/EWSN.2005.1462003.

C. K. Williams. Prediction with gaussian processes: From linear regression to linear prediction and beyond. In *Learning in graphical models*, pages 599–621. Springer, 1998.

X. Wu, G. Chen, and S. Das. Avoiding energy holes in wireless sensor networks with nonuniform node distribution. *IEEE Transactions on Parallel and Distributed Systems*, 19(5):710 – 720, May 2008a.

X. Wu, G. Chen, and S. K. Das. Avoiding energy holes in wireless sensor networks with nonuniform node distribution. *IEEE Transactions on Parallel and Distributed Systems*, 19(5):710–720, 2008b.

K. Xu, H. Hassanein, and G. Takahara. Relay node deployment strategies in heterogeneous wireless sensor networks: multiple-hop communication case. *2005 Second Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks, IEEE SECON*, pages 575 – 585, Jan 2005.

F. Xue and P. Kumar. The number of neighbors needed for connectivity of wireless networks. *Wireless Networks*, 10(2):169–181, 2004a.

F. Xue and P. Kumar. The number of neighbors needed for connectivity of wireless networks. *Wireless Networks*, 10(2):169–181, 2004b.

F. Xue and P. Kumar. On the $\theta$-coverage and connectivity of large random networks. *IEEE/ACM Transactions on Networking (TON)*, 14:2289–2299, 2006.

M. Younis and K. Akkaya. Strategies and techniques for node placement in wireless sensor networks: A survey. *Ad Hoc Networks*, Jan 2008.

O. Younis, M. Krunz, and S. Ramasubramanian. Node clustering in wireless sensor networks: recent developments and deployment challenges. *Network, IEEE*, 20 (3):20–25, 2006.

H. Zhang and J. Hou. On the upper bound of $\alpha$-lifetime for large sensor networks. *ACM Transactions on Sensor Networks (TOSN)*, 1(2):272–300, 2005a.

H. Zhang and J. C. Hou. Maintaining sensing coverage and connectivity in large sensor networks. *Ad Hoc & Sensor Wireless Networks*, 1(1-2):89–124, 2005b.

Z. Zhang, G. Mao, and B. D. O. Anderson. On the Hop Count Statistics in Wireless Multihop Networks Subject to Fading. *IEEE Transactions on Parallel and Distributed Systems*, 23(7):1275–1287, 2012.

F. Zhao and L. J. Guibas. *Wireless sensor networks: an information processing approach.* Morgan Kaufmann, 2004.

T. Zhao and A. Nehorai. Distributed sequential bayesian estimation of a diffusive source in wireless sensor networks. *Signal Processing, IEEE Transactions on*, 55 (4):1511–1524, April 2007. ISSN 1053-587X. doi: 10.1109/TSP.2006.889975.

P. Zipkin. Exact and approximate cost functions for product aggregates. *Management Science*, 28(9):1002–1012, 1982.

Y. Zou and K. Chakrabarty. A distributed coverage-and connectivity-centric technique for selecting active nodes in wireless sensor networks. *IEEE Transactions on Computers*, 54(8):978–991, 2005.

S. Zuyev. On a voronoi aggregative process related to a bivariate poisson process. *Advances in Applied Probability*, pages 965–981, Dec 1996.