Lifetime Analysis for Wireless Sensor Networks

Helen Legakis

A Thesis

in

the Department

of

Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of Master of Applied Science (Electrical & Computer Engineering) at

Concordia University

Montreal, Quebec, Canada

January 2008

Canada

ABSTRACT

Lifetime Analysis for Wireless Sensor Networks

Helen Legakis

The communications industry anticipates that Wireless Sensor Networks (WSNs) are the emerging technology to greatly affect society. A WSN is composed of numerous sensor nodes which have the ability to sense, compute and communicate in order to gather information about their surroundings. The efficiency of a given WSN is determined by its lifetime. Network lifetime is the duration of time for which it can maintain sensing coverage and network connectivity, which respectively involves the ability to detect an event in a region and to report the sensed data to its destination. In much related work, the network is considered unserviceable the moment when the entire area cannot be fully covered or when the network is not completely connected. However, in many application scenarios, as long as the percentage of disconnected sensor nodes and uncovered areas are above a threshold value, the utility of the network will not be harmed. This latter view will be applied in this thesis. We first derive the probability distribution of the lifetime of a single sensor node by modeling a sensor node as an $M/M/1$ queue which alternates between idle and busy periods. Then, the network connectivity probability is determined by discovering the percentage of sensor nodes that can communicate with the destination given that the monitored region is partially covered. The sensor nodes are randomly deployed in a grid-based network according to the Poisson distribution. Given the average of the number of sensor nodes in a cell, the

connectivity probability of two adjacent cells is determined. From this result, we can then derive the probability that a sensor node can communicate with a sink. Finally, the results found for the probability distribution of the lifetime of a single sensor node and the network connectivity probability are integrated to determine the network lifetime.

ACKNOWLEDGEMENTS

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

Symbol	Definition
c_i	duration of the ith cycle
i_i	duration of the ith idle period
b_i	duration of the ith busy period
NL	lifetime of a sensor node
BL	lifetime of a sensor node's battery
IT_n	total duration of idle periods for n cycles
BT_n	total duration of busy periods for n cycles
CT_n	total duration of n cycles
λ	the arrival rate
μ	the service rate
ρ	the load
$I(s)$	Laplace transform of an idle period
$IT_n(s)$	Laplace transform of the sum of n idle periods
$f_{IT_n}(t)$	probability density function of the sum of n idle periods
$\overline{IT_n}$	average duration of the sum of n idle periods
$B(s)$	Laplace transform of a busy period
$M(s)$	Laplace transform of the service time
$BT_n(s)$	Laplace transform of the sum of n busy periods
$f_{BT_n}(t)$	probability density function of the sum of n busy periods

I_n	modified Bessel function of the first kind of order n
$\overline{BT_n}$	average duration of the sum of n busy periods
$F_{NL}(t)$	probability distribution of the lifetime of a sensor node
N	number of completed cycles when the battery dies
$F_{NL}(t\,\|\,n)$	conditional cumulative distribution function of NL given $N(t) = n$ cycles
$p_N(n)$	probability mass function of the number of completed cycles when the battery dies
$f_{NL}(t\,\|\,n)$	conditional pdf of the lifetime of a sensor node given n cycles have been completed
$\gamma(g,x)$	lower incomplete gamma function
$F_{BT_n}(BL)$	probability of the event $\{\,BT_n \le BL\,\}$
$F_N(x)$	cumulative distribution function of N
$u(x)$	unit step function
$\overline{N}$	average number of cycles completed during the lifetime of a sensor node
$\overline{NL}$	average lifetime of a sensor node
λ_A	sensor nodes per unit area
d	side length of a cell
s	cell area
ξ	average number of sensor nodes in a cell
f_a	area coverage
P_{N_c}	distribution of the number of sensor nodes within a cell area

$N_c(z)$	PGF for the distribution of the number of sensor nodes within a cell area
p_1	probability that a randomly chosen pair of sensor nodes, located in adjacent cells, are within range of one another
p_2	probability that a randomly chosen sensor node in one cell is connected to at least one sensor node in an adjacent cell
p	probability that two adjacent cells are connected
L	distance between two sensor nodes
n_r	number of sensor nodes in the right cell
$N_r(z)$	PGF of n_r
n_ℓ	number of sensor nodes sensor nodes in the left cell that has connectivity with at least one of the sensor nodes in the right cell
$N_\ell(z)$	PGF of n_ℓ
q	number of cells in a block
$Q(z) =$	PGF of q
$\bar{q}$	average length of a block in number of cells
k_n	number of cells in segment n
i_n	number of tagged blocks in segment n
ℓ_n	number of additional cells that segment n has compared to segment $n-1$ on the left side
r_n	number of additional cells that segment n has compared to segment $n-1$ on the right side
$L_n(z)$	PGF of ℓ_n
$R_n(z)$	PGF of r_n

$K_n(z)$	PGF of k_n
$\overline{k_1}$	average length of the first segment
d_n	number of gates between adjacent cells in segment n within the borders of segment n-1
f_n	number of gates from among d_n which are closed
$I_n(z)$	PGF of the distribution of i_n
$\overline{i_n}$	average number of tagged blocks in segment n
x_n	number of primary blocks in segment n
s_n	number of secondary blocks in segment n
$\theta(\ell, r)$	number of ways that an integer ℓ may be partitioned into r fragments
$\varphi(m)$	number of ways that α primary blocks may be partitioned into m fragments given β secondary blocks
ϕ	total number of ways that α primary and β secondary blocks may be partitioned to
R_m	probability that α primary blocks may be partitioned into m fragments given β secondary blocks
$\overline{m_n}$	average number of primary fragments in segment n
$\overline{g_n}$	mean number of blocks in a primary fragment in segment n
$\overline{h_n}$	average number of blocks in a frame in row n
$\overline{x_n}$	mean number of primary blocks in segment n
γ	Probability that a tagged blocked is a primary block
P_r	probability that a sensor node located within the first r segments will have connectivity with the sink

LIST OF ABBREVIATIONS

Abbreviation	Definition
WSN	Wireless Sensor Network
PDA	Personal Digital Assistant
QoS	Quality of Service
MAC	Medium Access Control
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
TDMA	Time-Division Multiple Access
iid	independent, identically distributed
LT	Laplace transform

CHAPTER 1:

INTRODUCTION

1.1 Introduction

In recent decades, wireless communications technology has grown very rapidly and it has greatly affected modern life. Wireless networks are typically based on an infrastructure, where base stations are used to connect wireless entities to a wired backbone network and relay data between them. Although infrastructure-based wireless networks are the most commonly deployed, problems arise in some scenarios. In disaster areas, infrastructure may be unavailable. In other cases, sites may be too inaccessible or very large where it may be excessively expensive or difficult to set up an infrastructure. In military operations, there may be no time to setup an infrastructure [1]. These situations can be treated by creating a wireless network where no infrastructure is needed. One of the most interesting infrastructureless wireless networks to have emerged to date is the wireless sensor networks (WSNs).

A WSN is composed of numerous independent sensor nodes, which are usually battery operated, that organize themselves into a network without being directed or controlled by an external entity. Sensor nodes are devices equipped with sensors which check, track and observe the state of the physical environment under observation, e.g. temperature, pollution, pressure, etc., throughout a region [2]. Although WSNs were

1

initially considered for military uses, it is anticipated that they will have many civilian applications. WSNs may provide many benefits over traditional communication networks in many areas including health, industry, environment, home, and commerce [1, 3, 4].

For a standard application, the sensor nodes are deployed in a specific area where we would like to obtain information. The deployment of sensor nodes can either be regular or random [1]. Regular deployment is when the positioning of the sensor nodes is planned (e.g. fixing sensors in machinery for surveillance and preventive maintenance). Random deployment is when a large number of sensor nodes are scattered in an area without having a specific pattern (e.g. dropping sensor nodes from an aircraft over a wildfire).

The sensor nodes in a WSN can play the role of a source and/or relay. The source is the sensor node which has detected an event or sensed data. This information is then reported back to the appropriate sink, as shown in Figure 1.1 [4].The sink is an entity found outside the network, like a gateway or Personal Digital Assistant (PDA), which communicates with a task manager via Internet or satellite. The sensed data from the source is usually relayed through multiple sensor nodes, called relays, to the sink. Relays receive and transmit data but not simultaneously.
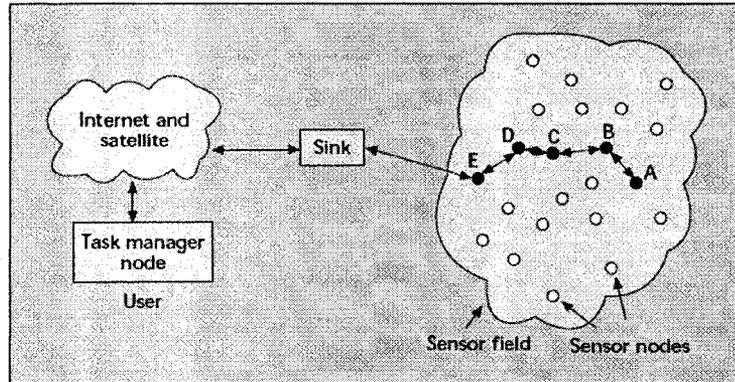
Figure 1.1: Sensor Nodes Scattered in a Sensor Field

The sources and the sink(s) can interact in different ways. The source can report to the sink(s) whenever it detects a particular event. The source can be assigned to periodically report its sensed data. When something anomalous occurs, the observer can poll certain sensors nodes to obtain more information about an area. The interaction between the sources and sink(s) can also be a combination of the interaction patterns mentioned above.

1.2 Main Characteristics of WSN

Although many diverse applications of WSN are under consideration, they all pose similar challenges on WSN, discussed below [1, 4-7].

- Quality of Service (QoS)

Unlike traditional QoS measures, where delay and jitter are main concerns, reliability is the most important QoS measure in WSN. A network is considered reliable if an event can be detected, the sensed data can be delivered to the appropriate sink, and the information received by the sink is accurate. Therefore, reliability can be categorized into three parts: detection reliability, information accuracy, and reliable data transfer.

3

Detection reliability and reliable data transfer respectively depend on the coverage and connectivity, which will be discussed in section 1.5.

- Fault Tolerance

A WSN is expected to tolerate and recover from failures. Faults occur when the battery has used up all its energy, the node gets damaged, or the communication between a pair of nodes gets interrupted for a long period of time. By deploying more nodes than necessary, the network will be robust against malfunctions.

- Lifetime

The efficiency of a given wireless sensor network is determined by its lifetime. The desirable goal is to have the whole network operating properly and performing its tasks for as long as possible. Inasmuch as the individual nodes have a limited power supply, energy needs to be preserved. This target can be attained if energy-efficient operations are performed. When more energy is consumed, the lifetime of a network shortens.

- Scalability

In many applications, the number of nodes used in the network is expected to be large. Hence, the network should have the ability to handle and support such a great number.

- Wide Range of Densities

Density is the number of nodes per unit area. The density value depends on the application. If an application requires the network to be densely inhabited, a higher value

will be assigned and vice versa. In addition, the density will change over time and space on account of node failure. These fluctuations should not affect the network.

- Programmability

When a new task is assigned, the sensor nodes in the network should be able to reprogram themselves so as to sense and monitor this event.

- Maintainability

A WSN has to adjust to changes and must sustain itself. The network should monitor its condition and adapt appropriately, if possible, when the energy is running low to maximize its lifetime.

In order for the above requirements to be met, a WSN needs the following mechanisms [1, 4-7]:

- Small-Scale Sensor Nodes

Recent advances have enabled the miniaturization of hardware components. Smaller features consume less power and are cost effective. Hence, small-scale sensor nodes facilitate in preserving energy and making a large network affordable.

- Multihop Wireless Communication

Communication between two nodes relies on the transmission range. Direct communication between entities that are far apart is possible by increasing the transmission range. However, this is inefficient since it requires more transmission power and the bandwidth is limited. By using multiple nodes along the path to the destination, the total required power is reduced as shorter transmission ranges are necessary.

- Energy-Efficient Operation

    By performing energy-efficient operations for communication, computation and sensing, the lifetime of the individual nodes and, more importantly, the network will last longer. Methods on how to consume less energy will be explained later.

- Auto-Configuration

    A WSN has to be able to organize itself given that no base stations are available to manually configure the network. The network should be aware of its surroundings and adapt to changes. Not only does the network need to tolerate node failures, it should support the integration of new nodes as well. Additionally, an entity is expected to be capable of locating itself in the network with the help of its surrounding nodes.

- Collaboration and in-network processing

    Sometimes a single node cannot provide enough information for an application (e.g. average temperature of an area). In this case, the combination of data of various nodes is required so as to sense an event. In addition, the processing of the information should be done in the network rather than at the edge of the network. In-network processing is energy-efficient since it reduces the amount of traffic in the network. The collaborating nodes join their data instead of having them all send their data individually to the sink.

- Data centric

    Conventionally in communication networks, each node contains a unique identifier in order to route packets between two distinct entities. This type of networking, known as address-centric, may be inefficient for WSNs. WSNs may contain a large

number of sensors. Therefore, embedding a unique ID to all the nodes would result in significant overhead and wastage of energy. Furthermore, the WSN applications focus on detecting and gathering information from the network and not which node provided it. A data-centric protocol would be better suited for WSNs. Also, collaboration and in-network processing is easier to implement with data-centric networking.

1.3 Architecture of a Sensor Node

Sensor nodes have the ability to sense, compute and communicate with the support of five main components, as shown in Figure 1.2 [1]: controller, communication device, sensor(s)/actuator(s), memory and power supply.
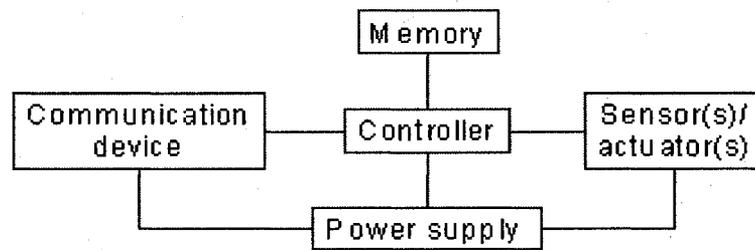
Figure 1.2: Components of a Sensor Node

The controller is the main part of the sensor node. The controller executes programs, processes data received by the sensor(s) or by other sensor nodes, determines when and where to transmit data, and manages the sensor node's other components. The communication device is responsible for sending and receiving information over a wireless medium. A receiver converts a radio wave into a sequence of bits and the reverse is carried out by a transmitter. Sensor nodes use a transceiver which joins the roles of a

transmitter and receiver into one unit. The sensors and actuators interact with the physical environment. Sensors monitor and measure the parameters of its surroundings, whereas actuators control an object (e.g. motor, light bulb) according to these physical parameters. Memory is used to store data and programs. Finally, the power supply is the component that provides the sensor node with energy for it to operate. In WSNs, the energy is usually supplied by batteries. Batteries can either be nonrechargeable, known as primary batteries, or rechargeable, referred to as secondary batteries. The secondary batteries are recharged by harnessing energy from the environment, like light, temperature gradients, vibrations, pressure vibration, and air/liquid flow.

Since energy is a limited source and recharging the battery is not always possible, the energy consumption of a sensor node is required to be highly restrained and managed. The components that mainly consume energy are the controller, the communication device, and to some level the memory and the sensors [1]. When sensor nodes have nothing to do, ideally it should be turned off in order to conserve the battery and wake up again once it is needed. As a sensor node cannot be turned off completely, it should operate in a state with reduced energy consumption, called the sleep state. This state can be employed by all the components of a sensor node and can save significant amount of energy. Therefore, whenever a sensor node is idle, it should go to sleep in order to prolong its lifetime.

Furthermore, studies have shown that the energy cost for communication is substantially higher than computation (data processing); the rule of thumb is that ten

times the power is required to transmit or receive a bit as is required to process a bit of information [1]. Hence, if achievable, a sensor node should process data locally instead of communicating it. This conclusion has inspired a number of key techniques in WSN, such as in-network processing and data aggregation mentioned earlier on, which help in reducing the overall power consumption.

1.4 WSN Protocol Architecture

The WSN protocol architecture consists of a physical, data link, network, transport, and application layer which are based on the OSI layering model. Since energy is a scarce resource in WSNs, energy-efficient operations should be incorporated into each of these layers. In order for the sensor nodes to perform correctly and efficiently, power and task management planes must be used by WSNs [4]. The power management plane controls how a sensor node makes use of its power. As explained earlier, the sensor node should operate in different states to reduce the power consumption. Hence, this plane is responsible for putting sensor nodes to sleep. The task management plane schedules the sensing tasks and synchronizes sensor nodes in a region. The protocols employed by WSNs must address these planes, which can lower the overall power consumption. Therefore, new protocols, in particular MAC and routing protocols, need to be developed for WSNs as traditional protocols are not cautious with energy.

1.4.1 MAC Protocols

The communication medium is shared by numerous entities. Medium Access Control (MAC) protocols are needed in order to determine when an entity can access the

medium. Traditional MAC protocols have been implemented to deal with delay, throughput, and fairness. However, these criteria are not the main concern for WSNs. WSNs need to perform energy-efficient operations to function as long as possible. Therefore, new MAC protocols need to be developed focusing on energy conservation instead of the conventional performance requirements.

Energy should be consumed only when needed. The following circumstances should be avoided since they futilely use up energy [1]:

- Collisions: When two packets collide because of near simultaneous transmission, energy is wasted at the source and destination and more energy needs to be consumed to retransmit the packet.

- Overhearing: In wireless communications, when a station transmits a packet, all the stations in its transmission range that have their receivers turned on hear the packet. If the packet is not intended for a station, the station discards the packet and wastes energy overhearing it.

- Idle listening: Idle listening is when a station is waiting to receive a packet but none of the stations in its surroundings have anything to send. Idle listening pointlessly consumes energy since the station is running at full operation for nothing to occur.

- Protocol overhead: Protocol overhead is extra data added to the body of the message which helps in transmitting packets between entities by describing the

message's content, for whom the message is intended to, and other protocol needs. Since the protocol overhead consumes extra energy, its size should be as small as possible.

MAC protocols for WSNs are expected to deal with at least one of these problems. If more of these energy problems are tackled, further energy will be conserved in the network.

MAC protocols for WSNs can be categorized as either contention-based or schedule-based protocols [1,4,8,9]. Contention-based protocols are based on a Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) scheme. A sensor node listens to the channel before attempting to transmit its packet. A station should suspend its transmission if the shared medium is sensed to be busy, seeing that another transmission is in progress, to circumvent collisions. If the channel is sensed idle, before a station transmits its packet, a signal is broadcasted to its neighbours notifying them that a data transfer will begin in order for them to suppress their transmission to avoid collisions. Although collision avoidance mechanisms are employed, they do not totally prevent collisions. Additionally, for contention-based protocols, idle listening can be tackled if a periodic wakeup scheme is added. The periodic wakeup scheme alternates between a sleep and listen period, as shown in Figure 1.3 [1]. During the wakeup period, the sensor node is asleep and wakes up periodically, during its listen period, to receive data. If no data is received, the node goes back to sleep. The sender needs to learn about the recipient's listen periods in order to transmit at the appropriate time.

Listen period

|← Wakeup period →| →| |←

Sleep period →
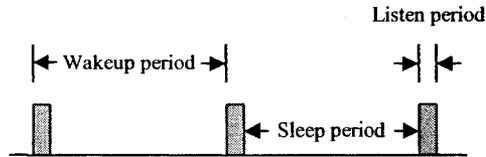
Figure 1.3: Periodic Wakeup Scheme

Unlike contention-based protocols where the sensor nodes contend each other to seize the channel, in schedule-based protocols coordination among the nodes is needed in order to determine which node can access the medium. These type of protocols are based on a Time-Division Multiple Access (TDMA) scheme. Time is divided into frames and each node is assigned to a specific slot in the frame. During a time slot, only the allotted sensor node is allowed to communicate, either transmit or receive data, whereas the other sensor nodes should be put to sleep. If a node does not use its slot, it remains idle. A schedule is required to store all the information indicating when the entities can access the medium and for how long. For entities to wake up right at the beginning of their allocated slot, time synchronization is necessary. The design of this protocol implicitly handles collisions, idle listening, and overhearing without the aid of additional mechanisms.

Even though schedule-based protocols solve collisions, idle listening, and overhearing, creating and maintaining the schedule and the use of time synchronization entail signalling traffic and memory. In addition, they are not as flexible and adaptable to changes in the network as contention-based protocols. Both schemes have their advantages and disadvantages making it hard to evaluate which design outperforms the

other; consequently, a standard MAC protocol cannot be elected since it highly depends on the application.

1.4.2 Routing Protocols

Routing protocols are responsible for selecting which sensor node to forward an incoming packet to and forming a route from the source to the sink. Based on the network structure, routing protocols can be classified into the following groups [9]:

- Flat-based routing: In flat-based routing protocols, all the nodes in the network are equivalent and perform the same tasks. Moreover, the nodes cooperate with each other in order to find routes.

- Hierarchical-based routing: For hierarchical-based routing, the nodes in the network are organized into clusters. Unlike flat-based routing, the nodes in hierarchical-based routing take part in different responsibilities: cluster head and member node of a cluster.

Unlike traditional routing protocols, where routes are usually computed based on the smallest hop count, the routing protocols for WSNs should determine the paths based on a metric that optimizes network lifetime. For example, if routes that consume less energy to transmit data packets or that have a higher available battery capacity are chosen, the lifetime of a network can be lengthened [4,9].

1.5 Coverage, Connectivity, and Lifetime

A WSN is considered reliable and effective depending on its ability to detect an event and deliver it to the sink. Coverage and connectivity are performance metrics that define the efficaciousness of sensing and communication, respectively. An area is totally covered if every point in that area is within the sensing range[1] of an active sensor node. Therefore, a network that provides complete coverage can detect all events. An area is considered connected if all the active sensor nodes in that area can communicate with each other; in other words, there exists at least one path between any pair of active nodes [10]. Hence, if an event is sensed in a fully connected network, this data will be guaranteed delivery to its destination. Thus, coverage and connectivity are key QoS measures.

Applications that use a higher degree of sensing coverage, meaning a region is monitored by more than one sensor node, make the network more accurate and fault tolerant. If a sensor node can report to the destination via various paths, the network becomes more robust and higher throughput is achieved. Safety-critical applications, such as surveillance, security and targeting, require a high degree of sensing coverage and network connectivity [11]. These applications do not abide even a small loss of coverage or connectivity. Ideally, a WSN should be accurate, fault tolerant, and robust; however, these features require more sensor nodes to remain active, which involves more energy

---

[1] The sensing range of a sensor node is the area where its sensor can detect an event; whereas the carrier sensing range of a sensor node is the area where the neighboring sensor nodes must suppress their transmission to avoid collision when the channel is busy.

14

consumption by the overall network resulting in shortening the lifetime of a WSN. If a network can effectively work without the need of a high degree of sensing coverage and network connectivity, the network lifetime can be considerably extended at a small cost of the accuracy, fault tolerance and robustness.

Although most studies related to sensing coverage have concentrated on the completely covered network problem, in some cases full coverage is unattainable or redundant. For example, a WSN for weather forecasting requires at least 90% of the monitored region to be covered in order for the network to be acceptable [12]. Moreover, the deployment of sensor nodes has an effect on the sensing coverage. For regular deployment, the sensor nodes can be placed in a fashion to ensure complete coverage. On the other hand, for random deployment, coverage of an area can never be guaranteed. Even with a high sensor node density, there is a chance that a small portion of a region may not be covered. In addition, as the WSN ages, sensor nodes die, because of their depletion of energy, and therefore the density of sensor nodes will decrease. Fewer sensor nodes in a region will eventually find it harder to monitor the whole area which will result in sections of the region not being monitored. Thus, depending on the application, partially covering a monitored region with a given coverage threshold is sufficient and more realistic than full coverage [12]. Furthermore, fewer sensor nodes are needed for partial coverage than for full coverage. Hence, with partial coverage, energy is conserved throughout the network, since more sensor nodes can be put to sleep, resulting in lengthening the lifespan of the network.

The lifetime of a network is the period of time during which it functions until it becomes unserviceable [13]. In much work, the network is considered unserviceable the moment that the whole area cannot be fully covered or when the network is not completely connected. Depending on the application, such as environmental monitoring, full sensing coverage and network connectivity is not required for the network to operate fruitfully. As long as the percentage of disconnected sensor nodes and uncovered areas are above a threshold value, the practicality of the network will not be harmed.

1.6 Related Literature

Next, we briefly discuss past work related to lifetime, coverage and connectivity of WSNs.

In [10], n sensor nodes are placed in a unit square grid-based network, which is a network divided into mini squares as shown in Figure 1.4.
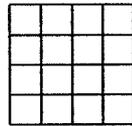
Figure 1.4: Square Grid-Based Network

The authors found that the necessary and sufficient condition for attaining both full area coverage and connectivity is

$$p(n)r^2(n) \approx \frac{\ln(n)}{n}$$

where $p(n)$ is the probability that a sensor node is active and $r(n)$ is the transmission radius.

In [14], the connected *k*-coverage problem was presented. The authors find a set of sensor nodes needed to cover every point in the monitored area by at least *k* sensor nodes and that are connected. In [11], by assuming that the transmission range of each sensor node is at least twice its sensing range, the authors show that *k*-coverage implies *k*-connectivity. Subsequently, an algorithm which provides different degrees of coverage and connectivity is introduced.

In [12], the partial coverage problem for a given coverage threshold is analyzed. The authors propose an algorithm that finds a subset of sensor nodes that will partially cover the monitored region with a given coverage guarantee and that will be connected. The performance of the algorithm is then evaluated through experimental simulations.

In [15], the lower bound of connectivity probability is examined for sensor networks. The monitored area is separated into a grid of blocks, illustrated in Figure 1.4, where *n* sensor nodes are uniformly distributed in a block. The connectivity probability is first found for each small block. Then, the small blocks are combined into a larger block until the entire network comes together. The connectivity probability of the whole sensor networks is found gradually and is based on the connectivity probability of the small blocks.

In [16], the area coverage (f_a) of an arbitrary circular area with radius *r*, equal to the sensing range of the sensor nodes, is found. This scenario is shown in Figure 1.5 [1].
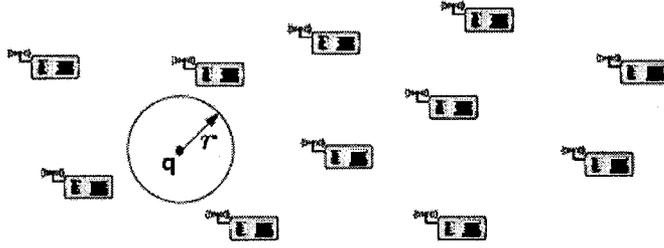
Figure 1.5: Determining Area Coverage

The authors have assumed that the deployment of sensor nodes in the network is according to the Poisson distribution with parameter λ. The probability that at least one sensor node is found in the area circling point q is given by

$$f_a = 1 - e^{-\lambda \pi r^2} \tag{1.1}$$

This probability also gives the percentage of the area coverage given that sensor density in the area is λ. Hence, from this equation, we can also find the density of sensor nodes required in an area to achieve a specific area coverage:

$$\lambda = -\frac{\ln(1 - f_a)}{\pi r^2}$$

Reference [17] reveals that a network with n sensor nodes randomly placed in the area is asymptotically disconnected with probability one as n increases if each node is connected to less than 0.074 $\ln(n)$ neighbours. On the other hand, the network is asymptotically connected with probability approaching one as n increases if each node is connected to more than 5.1774 $\ln(n)$ neighbours. In [18], the authors ameliorated the bound by proving that a network is asymptotically connected if each node has 2.718 $\ln(n)$ neighbours.

1.7 Research Objectives and Contributions of the Thesis

The objective of this thesis is to study the network lifetime of a WSN by evaluating the connectivity of the network as a function of time. Network lifetime highly depends on the sensing coverage and network connectivity. The modeling in this thesis inclusively handles coverage. In related work, a network is considered as connected when all the active sensor nodes in the network can communicate with each other. However, if two nodes are not connected but this path is not involved in relaying the sensed data to the sink, the sink will still receive this information, resulting in the network being serviceable. In this thesis, the definition of full network connectivity is changed to when all the active sensor nodes can communicate with the sink. Moreover, in contrast to the related literature, this analysis does not require the number of sensor nodes in the network to approach infinity in order to guarantee network connectivity.

The analysis is appropriate for a non-safety-critical area monitoring application where a large number of sensor nodes are randomly deployed over a region for some event to be monitored, e.g. environmental monitoring and weather forecasting. Unlike [11] and [14], this application does not demand a high degree of coverage and connectivity. Additionally, full sensing coverage and network connectivity is not needed for this application to function effectively as long as it remains above a given value, as presented in [12]. Therefore, the network lifetime, in this case, is considered as the time it takes before a percentage of sensor nodes that cannot communicate with the sink reaches a threshold value. The following explains the computation of the network lifetime.

First, the probability distribution of the lifetime of a single sensor node is derived by modeling a sensor node as an *M/M/*1 queue. In order to save energy, the sensor nodes are turned off when they have nothing to do and are reawakened when they are needed for communication (transmit or receive data). Hence, a contention-based protocol with a periodic wakeup scheme may be used for the MAC protocol. The transceiver of a sensor node alternates between idle and busy periods. A sensor node enters the busy period when it receives either measured data by its sensor or relayed data by its neighbouring sensor nodes in the network. During the idle periods, the sensor nodes are put to sleep, since they do not perform any tasks, and we assume that no energy is consumed.

Next, the connectivity probability between the sensor nodes in the network and the sink is determined. The sensor nodes are deployed over a large rectangular area. We assume that the deployment of the sensor nodes in the monitored area is according to the Poisson distribution. The monitored area is divided into a grid of virtual blocks, as described in [10] and [15], where the mini squares, called cells, each have a side equal to the transmission range of a sensor node. As a result, if a cell has connectivity to the sink, then all the sensor nodes residing in that cell will have connectivity to the sink. Thus, the problem under consideration reduces to the derivation of the connectivity probability of a cell to the sink. The sensor nodes in the network are equivalent and perform the same tasks, as in a flat-based routing protocol. We assume that data will be relayed from top to bottom, from right to left, and from left to right. The data will not be allowed to flow from bottom towards top, because it will be in the reverse direction of the sink. Given an average number of sensor nodes in a cell, the connectivity probability of two adjacent

cells is determined. From this result, we then discover if a sensor node can communicate with the sink to determine the network connectivity probability.

Finally, the results found for the probability distribution of the lifetime of a single sensor node and the network connectivity probability are integrated to determine the network lifetime.

1.8 Organization of Thesis

The thesis is organized as follows:

Chapter 1 *Introduction* presents the fundamental idea of wireless sensor networks, the reason of the study and the organization of the thesis.

Chapter 2 *Lifetime of a Sensor Node* explains the derivation of the probability distribution of the lifetime of a single sensor node. The results for different scenarios are shown at the end of the chapter.

Chapter 3 *Derivation of the Connectivity Probability for Two Adjacent Cells* obtains the connectivity probability of two adjacent cells for a grid-based network.

Chapter 4 *Network Connectivity* determines the percentage of sensor nodes in the network that can communicate with the sink. The numerical and simulation results are shown at the end of the chapter as a function of adjacent cell connectivity probability.

Chapter 5 *Network Lifetime* discovers the network connectivity of a WSN over time by integrating the results obtained in the previous chapters.

Chapter 6 *Conclusion* describes the main results obtained in this thesis.

CHAPTER 2:

LIFETIME OF A SENSOR NODE

2.1 Introduction

The objective of this chapter is to derive the probability distribution of the lifetime of a sensor node. The results should be useful in the design of WSNs. Since many sensor nodes in WSNs are battery-operated, the lifetime of a sensor node is the time between its activation and the expiration of its battery. When the battery dies, the sensor node dies. Energy-efficient operations must be performed in order to maximize the lifetime of the network. One approach to save energy is to turn the sensor nodes off when they have nothing to do and wake them up when they are needed for communication (transmitting or receiving data) since most energy is consumed by these activities.

2.2 Past Work on Sensor Node Lifetime

Next, we briefly discuss past work that has considered modeling of sensor nodes. The lifetime of a sensor node had also been studied in [19], which assumed that the arrival of messages to a sensor node is according to a Poisson process but messages have fixed lengths. The paper, without a proof, states that the probability density function of the lifetime of a sensor node follows an Erlang distribution. The degree and parameter of the Erlang distribution depend on the message arrival rate, message length, and the

amount of energy needed to transmit a message. In this thesis, we show that a sensor node lifetime has a more complicated distribution than an Erlang distribution.

A sensor node has been modeled as an $M/M/1$ queue in [20], which also assumed that a node has two states, active and sleep states. The node alternates between the active and sleep states and the residence time in each visit to a state is an independent exponentially distributed random variable with different parameters. The main difficulty of the model is that a node goes to sleep independent of its existing traffic backlog. The paper did not attempt to derive lifetime distribution of a sensor node, since the objective of the paper has been to study the network throughput and its energy consumption.

In [21], a sensor node has been modeled using a discrete time Markov chain. As in [20], a sensor node alternates between active and sleep states and the residence time during a visit to each state is assumed to be a geometrically distributed independent random variable. The model allows the beginning of a node's sleep time to be deferred until its queue becomes empty. However, during this deferral, the arrival of new messages to the node is not allowed. Then, the paper determines the probability transition matrix which is used to determine message throughput and mean message length of a node. The paper then studies network capacity and energy consumption assuming node independence except for loading.

In both [20] and [21], the amount of time that a node remains in the active state is independent of the state of the system, i.e. queue content. This will be a significant

disadvantage in those applications that there may not be any activity in the network for long periods of time and followed by a sudden burst of activity. During that burst of activity, it will be a far better strategy for a node not to sleep as long as its queue is nonempty, in order not to block the routing of the data to the sink. Thus, from this point of view, the model in this thesis is closer to the real world scenarios.

2.3 Structure of the Lifetime of a Sensor Node

The transceiver of a sensor node is divided into cycles, which are composed of an idle and a busy period, as shown in Figure 2.1
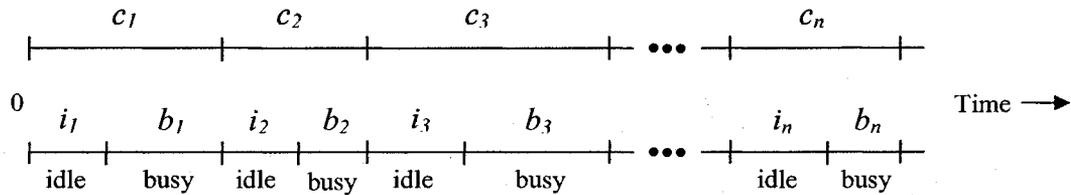
Figure 2.1: n Cycles of a Sensor Node

where c_i is the duration of the ith cycle, i_i is the ith idle period (the idle period belonging to the ith cycle), and b_i is the ith busy period (the busy period belonging to the ith cycle).

The transceiver of a sensor node is idle when it does not perform any tasks. Therefore, during the idle periods, the sensor nodes are put to sleep and we assume that no energy is consumed. On the other hand, the transceiver of a sensor node is busy when it needs to transmit or receive data. A sensor node enters the busy period when it receives either measured data by its sensor or relayed data by its neighboring sensor nodes in the network. It then goes back to sleep when it has transmitted all its acquired messages;

24

hence, energy is consumed during the busy periods. The transition between the idle and busy states requires extra energy. However, we assume that this energy consumption is negligible. As a result, the battery dies when the sum of all the busy periods up to a certain time t is greater than the battery life. This scenario is displayed in Figure 2.2.
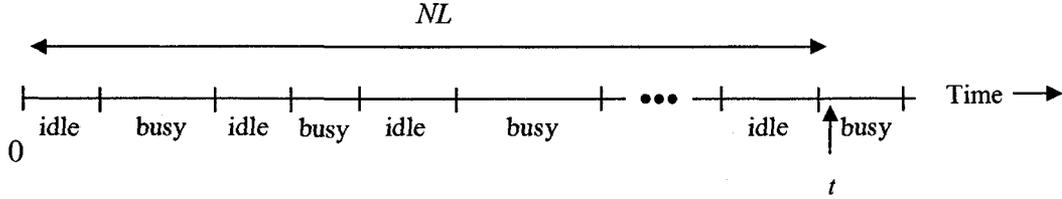
Figure 2.2: Lifetime of a Sensor Node

where NL is the lifetime of a sensor node. At time $t = NL$, the sensor node ceases to function because the end of the battery's lifetime (expressed as BL) is reached. Given that the sensor node alternates between idle and busy periods, the lifetime of a sensor node will be larger than the lifetime of its battery.

Under the assumption that no energy is consumed during the idle periods, the sensor node will die during a busy period. For example, n cycles are completed before the sensor node dies during the $(n+1)^{st}$ busy period. We assume that a sensor node will not start a busy period if the battery is low. Thus, the sensor node will stop operating at the end of the nth cycle. Put differently, NL will be equal to the duration of all the n cycles combined. Let IT_n and BT_n denote the total duration of idle and busy periods for n cycles respectively. Then,

$$IT_n = \sum_{i=1}^{n} i_i \quad , \quad BT_n = \sum_{i=1}^{n} b_i \tag{2.1}$$

Defining CT_n as,

$$CT_n = BT_n + IT_n \qquad (2.2)$$

From the explanation given above,

$$NL = CT_n \qquad (2.3)$$

Let us now examine the queueing system of a sensor node. Assume that when a sensor node's queue is empty, a received message will go into service instantly. In other words, the message will experience no waiting time when a sensor node is idle. A sensor node becomes idle once it has serviced all the messages waiting in its queue. The queueing system is illustrated in Figure 2.3, where M_i represents the ith message arrival.
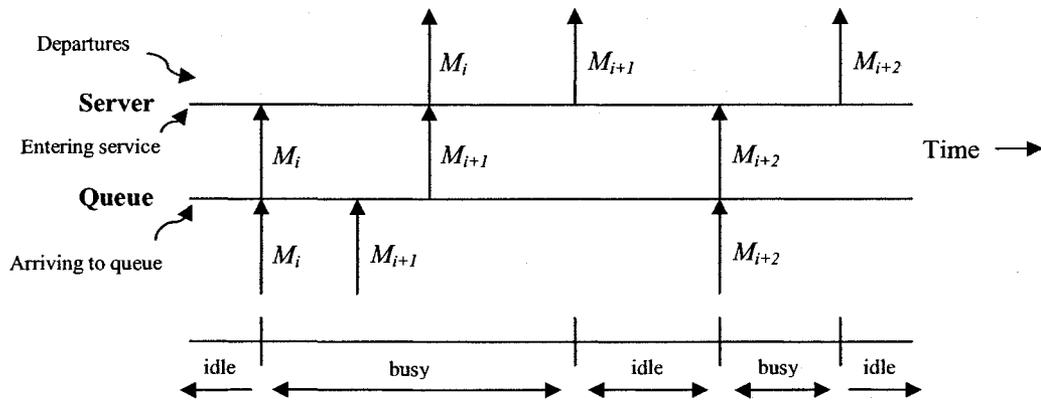
Figure 2.3: Queuing System of a Sensor Node

The arrow arriving at the queue line indicates that a message has entered the system and has been queued into a buffer with infinite capacity. The queued messages are served by a single server according to a first-come-first-served order of service. When the server is available, a message from the queue enters service, as shown by the arrow departing from the queue line and approaching the service line. Once the message has been served, it exits the system, as depicted by the arrow leaving from the service line.

The idle period is represented as the interval between the departure of a message which resulted in the server becoming idle and the arrival of a new message. Referring to Figure 2.3, the idle period is the interval between the departure of M_{i+1} and the arrival of M_{i+2}, where M_{i+2} is the newly incoming message.

The busy period begins when a message arrives to the sensor node while it is idle (no messages are waiting in the queue). During the service time of the message that initiated the busy period, if no other messages arrive to the sensor node, the busy period will be equivalent to the service time of this message. This is shown in Figure 2.3 where M_{i+2} arrives to an idle sensor node and no messages arrive during its service time.

On the other hand, if at least one message arrives to the sensor node during this service time, the busy period will be the sum of the service time of the message that initiated the busy period and of all the messages that arrive until the system is empty. The busy period terminates when the last message waiting in the queue departs from the system and no new messages arrive. This scenario is illustrated in Figure 2.3. The arrival of message M_i initiates the busy period. During the service time of M_i, message M_{i+1} arrives to the sensor node and waits in the queue until M_i is serviced. Once M_i departs the system, M_{i+1} will enter service. Since no messages enter the system during M_{i+1}'s service time, the busy period will terminate once M_{i+1} leaves the system. In this case, the busy period will be the sum of M_i's and M_{i+1}'s service times.

2.4 Modeling of a Sensor Node as an *M/M/*1 Queue

We will assume that the arrival of messages to a sensor node is according to a Poisson process and transmission time of a message is exponentially distributed. Thus, a sensor node will be modeled as an *M/M/*1 queue. A network of *M/M/*1 queues can be modeled as a Jackson Network [22, 23] in which the arrivals at any sensor node whether locally generated (sensed data) or from another sensor node (relay data) are Poisson. Therefore, the inter-arrival and service times of a sensor node are exponentially distributed, each with a different mean value, with λ as the arrival rate, μ as the service rate, and $\rho = \lambda / \mu$ as the load. Consequently, the idle and busy periods of a sensor node will correspond to the idle and busy periods for an *M/M/*1 queue. The idle period durations (i_i) and the busy period durations (b_i) are respectively independent, identically distributed (iid) random variables, each with a different distribution [23].

The idle period in an *M/M/*1 queue has the same distribution as the inter-arrival time exponential with mean $1/\lambda$. Therefore, the Laplace transform (LT) of an idle period is the Laplace transform of the pdf of the inter-arrival time,

$$I(s) = \frac{\lambda}{s+\lambda} \tag{2.4}$$

Next, for subsequent use we will also give LT of IT_n defined in (2.1). The LT of the sum of n idle periods is given by,

$$IT_n(s) = [I(s)]^n$$

Thus from Eq. (2.4) we obtain immediately that,

$$IT_n(s) = \left[\frac{\lambda}{s + \lambda} \right]^n \tag{2.5}$$

$IT_n(s)$ corresponds to the Laplace transform of an nth degree Erlang distribution. The corresponding probability density function of this distribution is

$$f_{IT_n}(t) = \frac{\lambda e^{-\lambda t}(\lambda t)^{n-1}}{(n-1)!}; \quad t \geq 0, n \geq 1 \tag{2.6}$$

The average duration of the sum of n idle periods is given by,

$$\overline{IT_n} = \frac{n}{\lambda} \tag{2.7}$$

The Laplace transform for the busy period (denoted as $B(s)$) for an $M/G/1$ queue can be expressed as [24]

$$B(s) = M(s + \lambda - \lambda B(s)) \tag{2.8}$$

where $M(s)$ is the Laplace transform of the service time, which is exponentially distributed. We then have

$$M(s) = \frac{\mu}{s + \mu} \tag{2.9}$$

With the help of (2.9), $B(s)$ is obtained by solving equation (2.8) as

$$B(s) = \frac{\mu}{s + \lambda - \lambda B(s) + \mu}$$

$$\lambda [B(s)]^2 - [\mu + \lambda + s]B(s) + \mu = 0$$

$$B(s) = \frac{\mu + \lambda + s - [(\mu + \lambda + s)^2 - 4\mu\lambda]^{\frac{1}{2}}}{2\lambda}; \quad |B(s)| \leq 1, \mathrm{Re}(s) \geq 0 \tag{2.10}$$

We note that the LT for the busy period can be solved for any service time distribution. However, it is difficult to find a closed form equation when the message transmission

time is not exponentially distributed; as an alternative, moments can be used to approximate this expression in future studies.

Next, for subsequent use we will also give LT of BT_n defined in (2.1). Since the busy periods (b_i) are iid random variables, the Laplace transform of the sum of n busy periods is given by,

$$BT_n(s) = [B(s)]^n$$

Thus from Eq. (2.10) we obtain immediately that

$$BT_n(s) = \left[\frac{\mu + \lambda + s - [(\mu + \lambda + s)^2 - 4\mu\lambda]^{1/2}}{2\lambda} \right]^n \qquad (2.11)$$

The probability density function of the sum of n busy periods, $f_{BT_n}(t)$, can be obtained by the inversion of the above equation. The inversion has been carried out with the use of the following transform pair [25]

$$\left[s - (s^2 - a^2)^{1/2} \right]^n \quad \Leftrightarrow \quad \frac{n a^n I_n(at)}{t}; \qquad Re(n) > 0$$

where

$$a = 2(\mu\lambda)^{1/2} \quad , \qquad I_n(at) = \sum_{m=0}^{\infty} \frac{\left(at/2 \right)^{n+2m}}{(n+m)!\,m!}$$

In the above, I_n is the modified Bessel function of the first kind of order n.

The density function is given by,

$$f_{BT_n}(t) = \frac{e^{-(\mu+\lambda)t}}{t\rho^{n/2}} n I_n(at); \qquad t > 0, n \geq 1 \qquad (2.12)$$

30

The average duration of the sum of n busy periods is given by,

$$\overline{BT_n} = \frac{n}{\mu - \lambda} \tag{2.13}$$

2.5 Derivation of the Lifetime Distribution of a Sensor Node

Next, we will determine the probability distribution of the lifetime of a sensor node defined as,

$$F_{NL}(t) = P[NL \le t] = 1 - P[NL > t]$$

We calculate the probability distribution of the lifetime of a sensor node by conditioning on N, which is defined as the number of completed cycles when the battery dies. The cumulative distribution function of the lifetime of a sensor node depends on N and may then be written as

$$F_{NL}(t) = P[NL \le t] = \sum_{n=1}^{\infty} P[NL \le t | N = n] \, P[N = n] \tag{2.14}$$

The subsequent subsections will comprise the following: first, given that n cycles have been completed throughout a sensor node's lifetime, the conditional cumulative distribution function of the lifetime of a sensor node ($P[NL \le t | N = n]$) is determined; second, the probability mass function of the number of completed cycles when the battery dies ($P[N = n]$) is obtained; third, the cumulative distribution function of NL ($F_{NL}(t)$) by unconditioning with respect to N is found; finally, for purposes of comparison, the average lifetime of a sensor node is determined using a purely average value approach.

31

2.5.1 Conditional CDF of NL Given N

Given that n cycles have been completed during the lifetime of a sensor node, $N = n$, then, from equations (2.2) and (2.3),

$$NL = BT_n + IT_n \qquad (2.15)$$

Since we have assumed that the battery dies exactly at the end of the nth cycle and energy is used up only during the busy periods, the total duration of the busy periods for n cycles, BT_n, will be equivalent to the battery's lifetime (BL), where BL is a constant of a known value. Then Eq. (2.15) becomes

$$NL = BL + IT_n \qquad (2.16)$$

The probability density function of the linear function given in Eq. (2.16) is determined as [26],

$$f_{NL}(t \mid n) = f_{IT_n}(t - BL)$$

Thus from Eq. (2.6), the conditional pdf of NL given $N = n$ is

$$f_{NL}(t \mid n) = \frac{\lambda e^{-\lambda(t - BL)}[\lambda(t - BL)]^{n-1}}{(n-1)!}; \qquad t \geq BL, n \geq 1 \qquad (2.17)$$

Next, the conditional cumulative distribution function of NL given $N = n$ is found by integrating the conditional pdf given in Eq. (2.17) as follows:

$$F_{NL}(t \mid n) = P[NL \leq t \mid N = n] = \int_{BL}^{t} f_{NL}(y \mid n) \, dy$$

$$= \frac{\lambda}{(n-1)!} \int_{BL}^{t} e^{-\lambda(y - BL)}[\lambda(y - BL)]^{n-1} dy$$

Following some substitutions, the above integral can be represented as a lower incomplete gamma function:

$$\gamma(g,x) = \int_0^x e^{-t} t^{g-1} dt$$

Therefore, the conditional CDF of NL given $N = n$ may be written as

$$F_{NL}(t \mid n) = \frac{1}{(n-1)!} \gamma(n, \lambda(t - BL)); \qquad t \geq BL, n \geq 1 \tag{2.18}$$

2.5.2 The Probability Mass Function of N

The probability mass function of n cycles being completed when the battery dies may be determined as follows [27]:

$$P[N = n] = P[N \geq n] - P[N \geq n+1] \tag{2.19}$$

If more than n cycles have been completed during a sensor node's lifetime, then the sum of n busy periods does not exceed BL since the battery does not die at the end of the nth cycle:

$$P[N \geq n] = P[BT_n \leq BL]$$

Therefore, the probability mass function of n cycles being completed when the battery dies can be written as

$$P[N = n] = P[BT_n \leq BL] - P[BT_{n+1} \leq BL] \tag{2.20}$$

The probability of the event $\{ BT_n \leq BL \}$ is found by integrating the pdf of BT_n (found in Eq. (2.12)) as follows

$$F_{BT_n}(BL) = P[BT_n \le BL] = \int_0^{BL} f_{BT_n}(t)\, dt \qquad (2.21)$$

This integral is difficult to evaluate in closed form. Therefore the alternative approach of numerical integration has been used in evaluation of $F_{BT_n}(BL)$.

Now Eq. (2.20) can be employed by using the results obtained by Eq. (2.21) in order to find the probability mass function of N,

$$p_N(n) = P[N = n] = F_{BT_n}(BL) - F_{BT_{n+1}}(BL) \qquad (2.22)$$

Finally, the cumulative distribution function of N is given by,

$$F_N(x) = \sum_{n=1}^{\infty} p_N(n)\, u(x - n) \qquad (2.23)$$

where $p_N(n)$ is the probability mass function of N given in Eq. (2.22) and $u(x)$ is the unit step function. We note that the average number of cycles completed during the lifetime of a sensor node, $\overline{N}$, may be computed as

$$\overline{N} = \sum_{n=1}^{\infty} n p_N(n) \qquad (2.24)$$

2.5.3 Unconditional CDF of Lifetime of a Sensor Node

Substitution of (2.18) and (2.22) into (2.14) gives the unconditional CDF of lifetime of a sensor node,

$$F_{NL}(t) = \sum_{n=1}^{\infty} \frac{1}{(n-1)!} \gamma\big(n, \lambda(t - BL)\big) \big[F_{BT_n}(BL) - F_{BT_{n+1}}(BL)\big], \qquad t \ge BL \qquad (2.25)$$

The above CDF is the main result of this chapter. It is a complicated expression involving modified Bessel function of first order, but it can be evaluated numerically.

2.5.4 Average Lifetime of a Sensor Node

In this subsection, we determine a closed form expression for the average lifetime of a sensor node. First, we determine the average number of cycles in the lifetime of a sensor node. We note that the number of cycles completed during the lifetime of a sensor node forms a stopping time for the sum of the busy periods in battery's lifetime. From Wald's identity, [27], the average number of cycles in the lifetime is given by,

$$BL = \frac{\overline{N}}{\mu - \lambda}$$

$$\overline{N} = BL(\mu - \lambda) \tag{2.26}$$

The above gives an alternative closed form expression to (2.24) for the average number of cycles completed during the lifetime of a sensor node.

From Eq. (2.7) and (2.13), the average lifetime of a sensor node is given by

$$\overline{NL} = \overline{BT}_n + \overline{IT}_n = \frac{\overline{N}}{\mu - \lambda} + \frac{\overline{N}}{\lambda} = \frac{\overline{N}\mu}{(\mu - \lambda)\lambda}$$

Substituting from (2.26) in the above,

$$\overline{NL} = \frac{\mu BL}{\lambda} \tag{2.27}$$

As it may be seen, the average lifetime of a sensor node depends on the battery life, message arrival and service rates.

2.6 Numerical Results

In this section, we present some numerical results regarding the analysis developed in this chapter. We plot various figures that show the probability distribution of the lifetime number of cycles (2.23) and the distribution of lifetime (2.25) for different parameter values. We also give some validity checks. As explained before the sensor node lifetime depends on battery life, message generation and service rates. We present results for three values of battery life, $BL =$72, 96, 120hours, which are the typical battery life values with the present day technology [28]. We set the value of message service rate to $\mu =$150messages/hour.

The evaluation of (2.23) and (2.25) involves an infinite summation that depends on the number of cycles completed during the lifetime of a sensor node. In the calculation of these distributions, the summation needs to be truncated and therefore the accuracy of this truncation is a concern. We have determined the validity of this approach by calculating the average number of the cycles in the lifetime of a sensor node from (2.24) and (2.26). The latter has a closed form expression while the former depends on the same infinite summation that the calculations in (2.23) and (2.25) depend. Table 2.1 presents this average for different values of battery life and message arrival rates. It may be seen that the difference between the results of two calculation methods is almost always less than one percent. This gives us the confidence that our numerical results are accurate.

Table 2.1: Verification of the Average Number of Cycles Completed During a Sensor Node's Lifetime

BL	*λ*	*Eq.(2.26)*	*Eq. (2.23)*	*% Diff*
72	15	9720	9783.4	0.65
72	75	5400	5434.2	0.63
72	135	1080	1091.7	1.08
96	135	1440	1452	0.83
120	135	1800	1812.1	0.67

In Fig. 2.4, we present CDF of the number of cycles completed during the lifetime of a sensor node from (2.23) with battery life as a parameter for a constant value of message arrival rate λ = 135 messages/hour. As may be seen, initially all the curves have zero value, then, they approach one in a steep manner. Fig. 2.5 presents the same CDF with message arrival rate as a parameter for a battery lifetime of 72 hours. The comments made about the previous figure will also apply here.
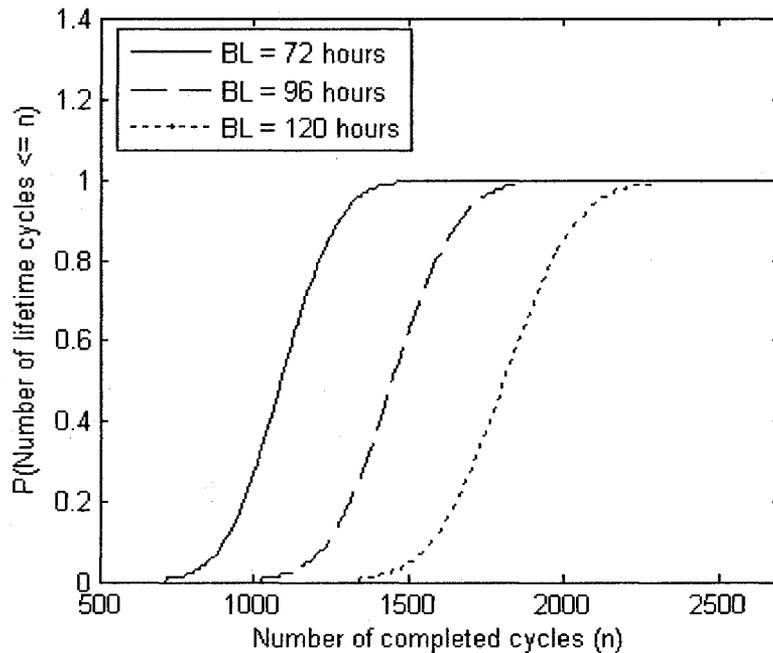
Figure 2.4: Cumulative Distribution Function of the Number of Cycles in the Lifetime for Different Values of *BL* given that *λ*=135 messages/hour
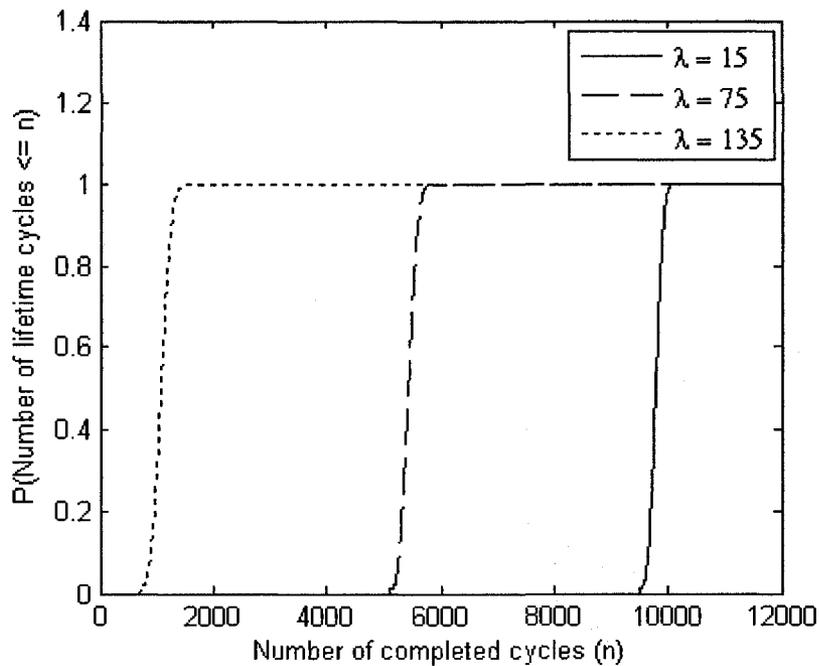
Figure 2.5: Cumulative Distribution Function of the Number of Cycles in the Lifetime for Different Values of λ given that BL=72 hours

In Fig. 2.6, we present the cumulative distribution of a sensor lifetime with battery lifetime as a parameter for a constant value of message arrival rate λ = 135 messages/hour. We note that this corresponds to a high traffic load and node utilization will be ρ = 0.9. This means that the node will be almost always busy, thus the battery will be drained continuously. It may be seen that initially the curves have zero values and very sharply they approach to one. As expected, the sharp rise occurs almost at the battery life. This observation gives further confidence that our results are correct.
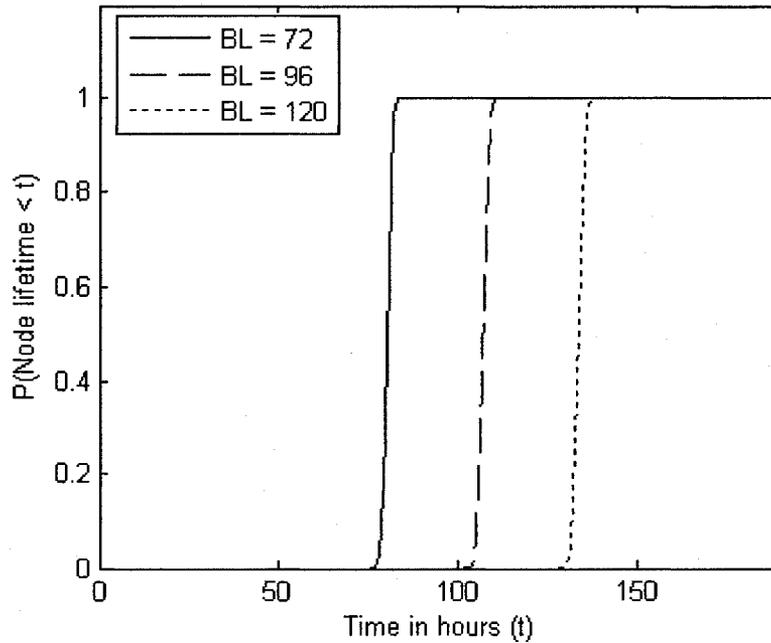
Figure 2.6: Cumulative Distribution Function of the Lifetime of a Sensor Node for Different Values of *BL* given that λ=135 messages/hour

Fig. 2.7 also shows the CDF of a sensor node's lifetime with message arrival rate as a parameter $\lambda = 135, 75, 15$ for a constant battery life of $BL = 72$ hours. As before, initially, the curves have a zero value and then sharply they approach to one. It may be seen that as the message arrival rate drops down, the sensor node life time increases. For $\lambda = 15$, the node will be under a light traffic load with server utilization $\rho = 0.1$. For this system, the probability that a sensor's lifetime will be less than 700hours is zero, which is a tenfold increase in sensor's lifetime compared to battery life. In WSNs that experience long periods of inactivity and then a sudden burst of activity, the node utilization will be low. Therefore, the model with the assumed traffic loading may characterize a sensor node in such a network.
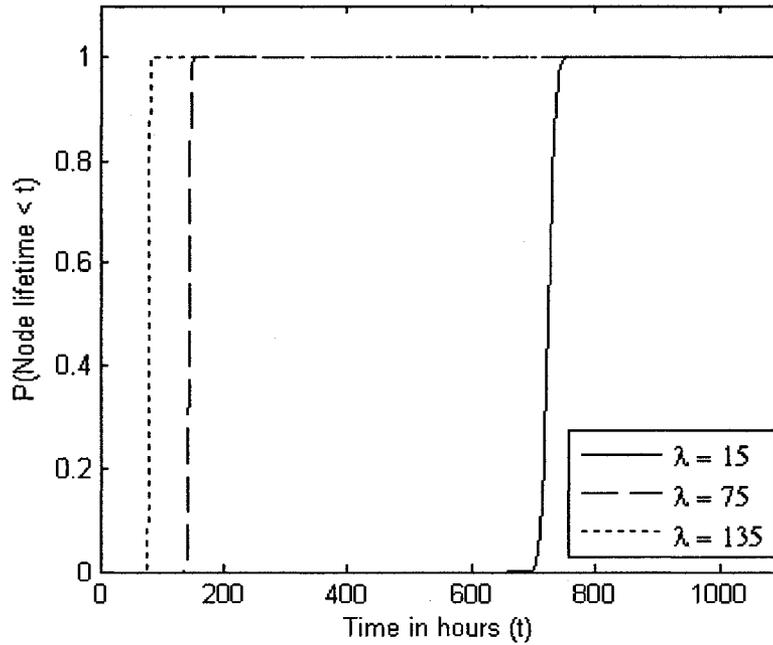
Figure 2.7: Cumulative Distribution Function of the Lifetime of a Sensor Node for Different Values of λ given that BL=72 hours

Finally, in Tables 2.2 and 2.3, we present the 90, 95 and 99th percentiles of the distribution of sensor node lifetime for battery life and message arrival rate as a parameter respectively.

Table 2.2: Percentiles for the CDF of NL with different BL given that λ=135 and μ=150

Percentile	BL=72	BL=96	BL=120
90%	81.62	108.49	135.31
95%	81.93	108.9	135.83
99%	82.8	109.83	136.82

Table 2.3: Percentiles for the CDF of NL with different λ given that BL=72 and μ=150

Percentile	λ=15	λ=75	λ=135
90%	736.41	146.97	81.62
95%	739.89	147.76	81.93
99%	746.37	148.99	82.8

According to the results obtained above, we can conclude that when we increase the battery life of a sensor node, increase the service rate or decrease the arrival rate, the lifetime of a sensor node is expected to increase. By equipping a sensor node with a battery that supplies more energy, the sensor node can last longer. By augmenting the arrival rate, a sensor node will be receiving more messages resulting in performing more tasks and sleeping less. Hence, the sensor node will be consuming more energy and the sensor node will die faster. If the service rate is high, the sooner a sensor node will transmit its messages and go back to sleep. Thus, low service rates consume more energy since the sensor node is operating for a longer time.

CHAPTER 3:

DERIVATION OF THE CONNECTIVITY PROBABILITY FOR TWO ADJACENT CELLS

3.1 Introduction

In this chapter, the sensor nodes will be assumed to be deployed uniformly over a large square area. Deployment area will be logically divided into mini squares to be referred to as *cells*. The objective of this chapter is to derive the communications connectivity probability of two adjacent cells. This result will be then used, in Chapter 4, to determine the connectivity probability between a sensor node and the sink for the entire network.

3.2 Sensor Node Population in a Cell

In this section, we will consider the distribution of the number of sensor nodes in a cell. The sensor nodes are randomly deployed over a square area according to a Poisson process with density λ_A sensor nodes per unit area. The deployment area is then divided into cells where each has a side of d units, as illustrated in Figure 3.1.
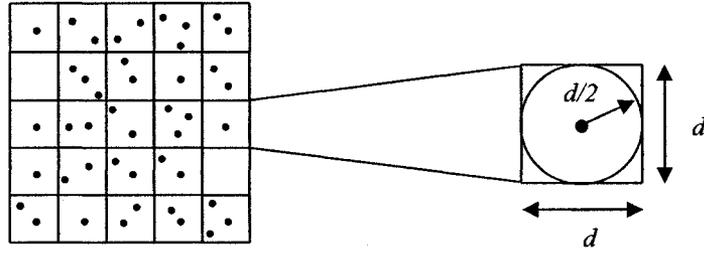
Figure 3.1: Deployment Area Divided into Cells and Representation of a Single Cell

Let us define s as the cell area and ξ as the average number of sensor nodes in a cell,

$$s = d^2 \quad , \quad \xi = \lambda_A s$$

By neglecting the border effects, we assume that each cell is covered by a circular area with radius $d/2$, as shown in Figure 3.1. In other words, a sensor node's sensing range is set to $d/2$. With the help of equation (1.1), we can discover the area coverage, f_a, of a cell given a specific mean number of sensor nodes in a cell as follows,

$$f_a = 1 - e^{-\xi \pi / 4} \tag{3.1}$$

f_a determines the probability that an event will be detected in a cell, which in fact also provides the area coverage of the network since an average of ξ sensor nodes reside in all the cells. The area coverage for different cell densities is in Table 3.1. We will prove that this model inclusively handles coverage in Chapter 5.

Table 3.1: Area Coverage for Different Cell Densities

ξ	2	3	4	5
f_a	0.79212	0.90522	0.956786	0.980297

The distribution of the number of sensor nodes within a cell area is given by the Poisson distribution

$$P_{N_c} = \Pr[k \text{ nodes in a cell}] = \frac{e^{-\xi}(\xi)^k}{k!} \qquad (3.2)$$

The probability generating function (PGF) of the Poisson distribution is given by

$$N_c(z) = \sum_{k=0}^{\infty} z^k P_{N_c} = \sum_{k=0}^{\infty} z^k \frac{e^{-\xi}(\xi)^k}{k!}$$

$$N_c(z) = e^{-\xi(1-z)} \qquad (3.3)$$

It is a property of the Poisson process that, given k sensor nodes in a cell, the nodes will be independently, uniformly distributed throughout a cell. This property will be used extensively in the rest of the chapter.

3.3 Derivation of the Cell Connectivity Probability

In this section, we will determine the connectivity probability of two adjacent cells. In the following, it will be assumed that the side of each cell (d) will be equal to a sensor node's transmission range. As a result, all the sensor nodes within the same cell will be connected. If two cells contain sensor nodes within transmission range of each other, then, these two cells will have connectivity. If two cells are connected, messages can be transmitted from one cell to the other in both directions.

In the subsequent subsections, the following will be determined: first, the probability that a randomly chosen pair of sensor nodes, located in adjacent cells, are within range of one another (p_1); second, the probability that a sensor node in a cell has connectivity with at least one sensor node located in the adjacent cell (p_2); finally, the

44

probability that at least one of the sensor nodes in a cell has connectivity with at least one of sensor nodes in the adjacent cell (p).

3.3.1 Determining p_1

First, we consider two neighboring cells with one sensor node lying in each as shown in Figure 3.2.
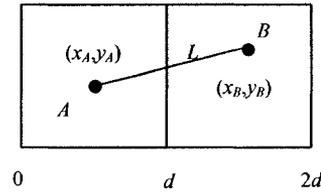
Figure 3.2: Two Adjacent Cells Containing One Node

The coordinates of the sensor nodes satisfy the following constraints,

$$0 \le x_A, y_A, y_B \le d \quad , \quad d \le x_B \le 2d$$

The locations of sensor nodes A and B are distributed uniformly in the left and right cell respectively. The probability density functions of these random variables are given by:

$$f_{X_A}(x_A) = f_{X_B}(x_B) = f_{Y_A}(y_A) = f_{Y_B}(y_B) = \begin{cases} 1/d & 0 \le x_A, y_A, y_B \le d \quad , \quad d \le x_B \le 2d \\ 0 & elsewhere \end{cases} \tag{3.4}$$

Two sensor nodes will be able to communicate directly if $L \le d$ where L is the distance between them and d is the maximum allowed distance for direct communication (the transmission range). The distance between the two sensor nodes is given by

$$L = \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2} \tag{3.5}$$

Let us define p_1 as the probability that the distance between a randomly chosen pair of sensor nodes is less than or equal to the transmission range:

$$p_1 = \Pr[L \le d]$$

If y_A and y_B can be any value between $[0, d]$, the following constraints must be applied in order to satisfy $L \le d$. From Eq. (3.5),

$$L^2 \le d^2$$

$$(x_B - x_A)^2 + (y_B - y_A)^2 \le d^2$$

$$(x_B - x_A)^2 \le d^2 - (y_B - y_A)^2$$

$$-\sqrt{d^2 - (y_B - y_A)^2} \le x_B - x_A \le \sqrt{d^2 - (y_B - y_A)^2}$$

$$x_A - \sqrt{d^2 - (y_B - y_A)^2} \le x_B \le x_A + \sqrt{d^2 - (y_B - y_A)^2}$$

The lower limit $x_A - \sqrt{d^2 - (y_B - y_A)^2}$ is always less than or equal to d. Hence,

$$d \le x_B \le x_A + \sqrt{d^2 - (y_B - y_A)^2} \tag{3.6}$$

since $x_B \ge d$.

The upper limit of x_B in Eq. (3.6) is always less than or equal to $2d$ but may also be equal to a number less than d. We need to find the limits of x_A where the upper limit of x_B is greater or equal to d.

$$x_A + \sqrt{d^2 - (y_B - y_A)^2} \ge d$$
$$x_A \ge d - \sqrt{d^2 - (y_B - y_A)^2}$$

Therefore, x_A ranges between

$$d - \sqrt{d^2 - (y_B - y_A)^2} \le x_A \le d \tag{3.7}$$

With the limits given in equations (3.5) and (3.6), we can now determine p_1:

$$p_1 = \int_0^d \int_0^d \int_{d-\sqrt{d^2-(y_B-y_A)^2}}^d \int_d^{x_A+\sqrt{d^2-(y_B-y_A)^2}} f_{X_A,X_B,Y_A,Y_B}(x_A,x_B,y_A,y_B)\,dx_B dx_A dy_B dy_A \qquad (3.8)$$

where $f_{X_A,X_B,Y_A,Y_B}(x_A,x_B,y_A,y_B)$ is the joint pdf of X_A, X_B, Y_A, and Y_B. Since these random variables are independent of each other, their joint pdf becomes

$$f_{X_A,X_B,Y_A,Y_B}(x_A,x_B,y_A,y_B) = f_{X_A}(x_A)f_{X_B}(x_B)f_{Y_A}(y_A)f_{Y_B}(y_B) = \frac{1}{d^4}$$

Then, p_1 in equation (3.8) is given by

$$p_1 = \frac{1}{d^4}\int_0^d \int_0^d \int_{d-\sqrt{d^2-(y_B-y_A)^2}}^d \int_d^{x_A+\sqrt{d^2-(y_B-y_A)^2}} dx_B dx_A dy_B dy_A$$

$$p_1 = \frac{1}{d^4}\int_0^d \int_0^d \int_{d-\sqrt{d^2-(y_B-y_A)^2}}^d \left[x_A + \sqrt{d^2-(y_B-y_A)^2} - d\right]dx_A dy_B dy_A$$

$$p_1 = \frac{1}{d^4}\int_0^d \int_0^d \left[\frac{d^2}{2} - \frac{y_B^2}{2} + y_A y_B - \frac{y_A^2}{2}\right]dy_B dy_A$$

$$p_1 = \frac{1}{d^4}\int_0^d \left[\frac{d^3}{3} + \frac{y_A d^2}{2} - \frac{y_A^2 d}{2}\right]dy_A$$

$$p_1 = \frac{1}{d^4}\left[\frac{d^4}{3} + \frac{d^4}{4} - \frac{d^4}{6}\right] = 5/12 \qquad (3.9)$$

3.3.2 Determining p_2

Now, let us consider two adjacent cells with one sensor node lying in a cell and k sensor nodes in the other as shown in Figure 3.3.
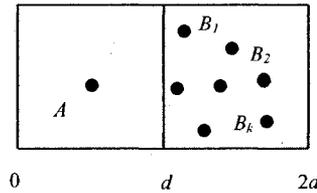
Figure 3.3: Two Adjacent Cells Containing One and k Nodes

47

Sensor nodes A and B_1, B_2, ..., B_k are distributed uniformly in the left and right cell respectively. We would like to determine p_2, which is the probability that a randomly chosen sensor node in one cell is connected to at least one sensor node in an adjacent cell. In this case, the neighboring cells are connected. Since the locations of sensor nodes are independent of each other, the connectivity between a pair of sensor nodes is then independent of any other pair in the network.

As explained earlier, the distribution of the number of sensor nodes within the right cell is given by the Poisson distribution. The probability generating function (PGF) of the number of nodes in the right cell, $N_c(z)$, is given in Eq. (3.3).

Referring to Figure 3.3, let n_r denote the number of sensor nodes in the right cell that sensor node A has connectivity with among k sensor nodes. Defining u_i to be a Bernoulli variable as follows,

$$u_i = \begin{cases} 1, & \textit{if two nodes are connected} \\ 0, & \textit{no connectivity} \end{cases}$$

$$U_i(z) = p_1 z + 1 - p_1$$

Then, n_r is given by the following random summation:

$$n_r = \sum_{i=0}^{k} u_i$$

Since we have a random sum of independent random variables, the probability generating function can be shown to be

$$N_r(z) = N_c(z)\big|_{z=U_i(z)}$$

48

We note that $N_r(0)$ gives the probability that none of the sensor nodes in the right cell have connectivity with sensor node A. Hence, the adjacent cells in this case are connected with probability

$$p_2 = 1 - N_r(0)$$

$$p_2 = 1 - e^{-\xi(1-p_1 z - 1 + p_1)}\bigg|_{z=0} = 1 - e^{-\xi p_1} = 1 - e^{-\lambda_A d^2 p_1} \qquad (3.10)$$

3.3.3 Determining p

Finally, let us consider two adjacent cells with k and j sensor nodes lying in each as shown in Figure 3.4.
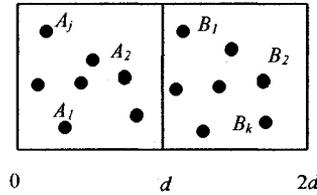
Figure 3.4: Two Adjacent Cells Containing j and k Nodes

Sensor nodes A_1, A_2, ..., A_j and B_1, B_2, ..., B_k are distributed uniformly in the left and right cells respectively. We would like to determine p, which is the probability that two adjacent cells will have connectivity. If the distance of at least one pair of sensor nodes in different cells is less than the transmission range, then, the neighboring cells will be connected.

As before, the distribution of the number of sensor nodes within a cell area is given by the Poisson distribution. The PGF for the distribution of k sensor nodes lying is a cell is given in Eq. (3.3).

49

Referring to Fig 3.4, let n_ℓ denote the number of sensor nodes among j sensor nodes in the left cell that has connectivity with at least one of the sensor nodes in the right cell. Defining v_i to be a Bernoulli variable as follows,

$$v_i = \begin{cases} 1, & \textit{if a node in the left cell has connectivity with at least one node in the right cell} \\ 0, & \textit{no connectivity} \end{cases}$$

$$V_i(z) = p_2 z + 1 - p_2$$

Then, n_ℓ is given by the following random summation:

$$n_\ell = \sum_{i=0}^{j} v_i$$

We note that the random variables v_i will not be independent of each other because it depends on the connectivity of each sensor node in the left cell to the same sensor nodes in the right cell. However, in the following, it will be assumed that these random variables are independent of each other. Since the number of sensor nodes in the left cell also has the Poisson distribution,

$$N_\ell(z) = N_c(z)\big|_{z=V_i(z)}$$

The accuracy of the independence assumption will be determined through simulation. We note that $N_\ell(0)$ gives the probability that none of the sensor nodes in the left cell have connectivity with any of the sensor nodes in the right cell. Hence, the adjacent cells are connected with probability

$$p = 1 - N_\ell(0)$$

$$p = 1 - e^{-\xi(1 - p_2 z - 1 + p_2)}\big|_{z=0} = 1 - e^{-\xi p_2} = 1 - e^{-\lambda_A d^2 p_2} \tag{3.11}$$

50

3.4 Numerical and Simulation Results

In this section, the numerical and Monte Carlo simulation results for the connectivity probability for two adjacent cells are compared for different values of ξ. In Table 3.2, we present the numerical and simulation cell connectivity probabilities, p, as a function of sensor node density. Also shown in the table is p_2. As may be seen, p_2 provides a lower bound for cell connectivity probability p except for low density nodes. These results are plotted in Figure 3.5.

Table 3.2: Comparison of Numerical and Simulation Cell Connectivity Probabilities as a Function of Cell Node Density

ξ	1	2	3	4	5	6	7	8	9	10
p Sim	0.228	0.555	0.748	0.897	0.958	0.975	0.992	0.992	0.999	0.999
p Num	0.2888	0.6772	0.8824	0.961	0.9874	0.9959	0.9987	0.9996	0.9998	0.9999
p_2 Num	0.3408	0.5654	0.7135	0.8111	0.8755	0.9179	0.9459	0.9643	0.9765	0.9845
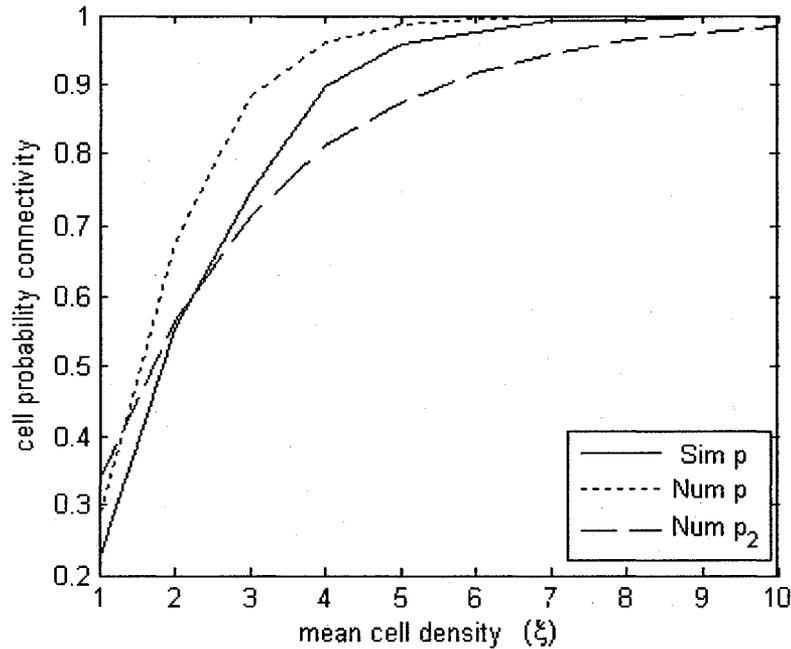
Figure 3.5: Simulation and Lower and Upper Bound of Numerical Results

Since we have assumed that the random variables v_i are independent of each other, the simulation and numerical results differ when the average number of sensor nodes in a cell is small. In Figure 3.5, the numerical results found for p_2 and p were plotted in order to give a lower and upper bound respectively to the results obtained by simulation. We observe that for approximately $\xi < 2$, the calculated values of p_2 and p give useful upper bounds. It is not likely that such low node density would be of interest. For approximately $\xi > 2$, the calculated values of p_2 and p give lower and upper bounds, respectively. Therefore, the calculated results will provide useful tools during the design process of WSN.

CHAPTER 4:

NETWORK CONNECTIVITY PROBABILITY

4.1 Introduction

In this chapter, we will determine the probability that a sensor node has connectivity to the sink. As in chapter 3, the sensor nodes will be assumed to be deployed uniformly over a very large rectangular area. Deployment area will be logically divided into mini squares to be referred to as cells. The side of each cell will be equal to the transmission range of a sensor node. As a result, if a cell has connectivity to the sink, then all the sensor nodes residing in that cell will also have connectivity to the sink. Thus, we have reduced the problem under consideration to the derivation of the connectivity probability of a cell to the sink.

4.2 Block Structure of the Cells

It will be assumed that the sensor nodes will be deployed to the entire upper Cartesian plane. Figure 4.1 shows the logical division of the deployment area into the cells, where a cell is represented by a square with a letter in it. The sink will be located at the origin, the mid-point of the bottom of the deployment area. We assume that the flow of data will be from top to bottom, from right to left, and from left to right. The data will

not be allowed to flow from bottom towards top, because it will be in the reverse direction of the sink.

e	e	e	f	f	f	e	e	f	f	f	e	e	e	f	f	f	f	e	e	e	row 3
c	c	c	d	d	c	c	c	c	d	d	d	c	c	c	d	d	d	d	c	c	row 2
a	a	a	a	b	b	b	b	a	a	a	a	a	b	b	b	b	a	a	a	a	row 1

... -10 -9 -8 -7 -6 -5 -4 -3 -2 -1 0 1 2 3 4 5 6 7 8 9 10 ...
Sink
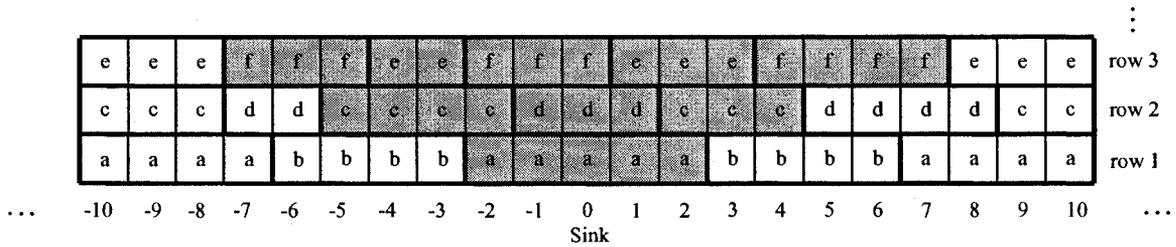
Figure 4.1: Division of the Deployment Area into Cells

We note that each cell will have four neighbors, above, below, right and left, except for the cells in the first row which will have only three neighbors. It will be assumed that the connectivity of a cell to each of its neighbors will be independent of the other neighbors. Two neighboring cells will have connectivity with probability p, discussed in Chapter 3. The connectivity of a cell to its neighbors above, below, right and left will be referred to as higher, lower, right and left connectivity respectively. Next, we define a *block* as a group of consecutive cells in a row such that adjacent cells have connectivity with each other. In each block, the traffic may flow across the cell boundaries to the left or right. The block is an important entity which plays a large role in the sequel. In Figure 4.1, each group of consecutive cells in a row that contain the same letter forms a block. In other words, the adjacent cells in a block will be connected to each other with probability p and a new block will begin with probability $1-p$, as shown in Figure 4.2.

block   block   block

Neighboring cells
are connected
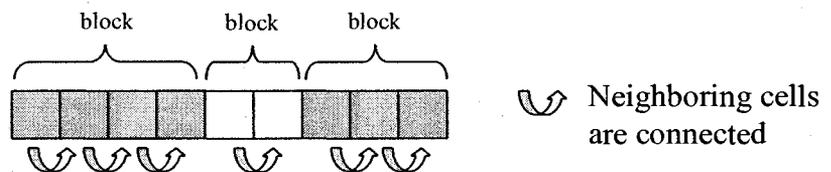
Figure 4.2: Formation of Blocks

54

The probability that a block contains k cells is given by the geometric distribution [23]:

$$\Pr(q = k) = \Pr(a\ block\ contains\ k\ cells)$$

$$\Pr(q = k) = (1 - p)p^{k-1} \quad , \quad k = 1,2,3,\ldots \tag{4.1}$$

The PGF of the geometric distribution is expressed as

$$Q(z) = E[z^q] = \frac{(1-p)z}{1-pz} \tag{4.2}$$

From above the average length of a block in number of cells is given by,

$$\bar{q} = \left. \frac{dQ(z)}{dz} \right|_{z=1} = \frac{1}{(1-p)} \tag{4.3}$$

The traffic in a block will be trapped if none of the cells of that block has lower connectivity.

4.3 Distribution of the Number of Tagged Cells in a Row

The blocks, in any row, that have the potential to be connected with the sink will be called *tagged blocks* and the corresponding cells of that block *tagged cells*. If at least one cell of a block is a neighbor to a tagged cell in the row immediately below, then that block becomes a tagged block and all its cells tagged cells. Row one always has a single tagged block since the cell at the origin is considered to be the sink. The tagged cells in a row will be collectively referred to as a *segment* which will be identified by its row number. All the cells in row n which interface the cells in segment n-1 will be tagged cells. However, segment n may have additional cells than segment n-1 because the border blocks of segment n may extend beyond the boundaries of segment n-1. Clearly, each segment will consist of contiguous cells. In Figure 4.1, the shaded blocks are tagged

blocks; they have the potential of communicating with the sink. Thus the tagged blocks in segment n are those blocks that have complete or partial borders with the tagged blocks in segment n-1. Clearly, it is only the tagged cells that may have connectivity to the sink. The tagged cells form a wedge on the plane. The following analysis excludes the non-tagged cells since they can not have connectivity to the sink. As a result of this exclusion, the monitored area becomes smaller resulting in it looking like an upside down pyramid with a sink located at the bottom, as illustrated in Figure 4.1. However, by adding three more sides, it approximately gives a rectangular area with the sink in the middle, as shown in Figure 4.3. From symmetry, connectivity in one of the wedges indicates connectivity in the entire square.
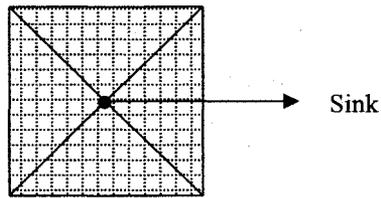
Figure 4.3: Combining Wedges in a Square Area

Next, we will determine the average number of cells in a segment. Let us define

k_n = number of cells in segment n

i_n = number of tagged blocks in segment n

ℓ_n = number of additional cells that segment n has compared to segment n-1 on the left side

r_n = number of additional cells that segment n has compared to segment n-1 on the right side

Figure 4.4 illustrates these symbols.
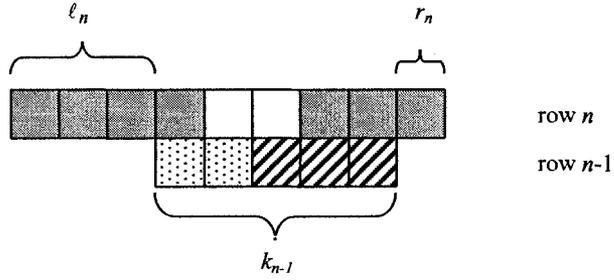
Figure 4.4: Example of Two Consecutive Segments

In Figure 4.4, three blocks in segment n (i_n=3) are bordering the tagged blocks in row n-1 (i_{n-1}=2). Row n and n-1 contain k_{n-1}=5 and k_n=9 tagged cells respectively. Compared to the tagged block in segment n-1, segment n comprises three additional tagged cells to the left (ℓ_n) and an additional tagged cell to the right (r_n).

As pointed above, the border blocks of segment n may extend beyond the boundaries of segment n-1. The probability that segment n has j additional cells compared to segment n-1 on the left and right side is given by the geometric distribution:

$$\Pr(\ell_n = j) = \Pr(r_n = j) = (1-p)p^j \quad , \quad j = 0,1,2,\dots \tag{4.4}$$

The PGFs of ℓ_n and r_n are found below:

$$L_n(z) = R_n(z) = E[z^{\ell_n}] = E[z^{r_n}] = \frac{1-p}{1-pz} \tag{4.5}$$

Since $L_n(z)$ and $R_n(z)$ do not depend on the row number n, the subscript n may be dropped as shown below,

$$L(z) = R(z) = \frac{1-p}{1-pz} \tag{4.6}$$

Then, the number of cells in segment n is given by,

57

$$k_n = k_{n-1} + \ell_n + r_n \qquad\qquad (4.7)$$

Since we are assuming infinite number of cells in each row, then, the random variables k_{n-1}, ℓ_n and r_n are independent random variables. Therefore, the PGF of k_n is expressed as

$$K_n(z) = K_{n-1}(z)L(z)R(z)$$

The above recursion results in the following explicit formula for $K_n(z)$,

$$K_n(z) = K_1(z)[L(z)R(z)]^{n-1} \qquad\qquad (4.8)$$

where $K_1(z)$ is the PGF of number of cells in the first segment. Since $L(z) = R(z)$, $K_n(z)$ can be expressed as

$$K_n(z) = K_1(z)[L(z)]^{2(n-1)} \qquad\qquad (4.9)$$

Next we will determine $K_1(z)$. From the renewal theory [27], the tagged block in row one will be "special" and it will have length bias since longer blocks are more likely to include the sink. The probability that this block contains j cells is given by

$$\Pr(k_1 = j) = \frac{j\,\Pr(q = j)}{\overline{q}} \qquad\qquad (4.10)$$

where $\Pr(q = j)$ is expressed in (4.1). The PGF of k_1 is found as

$$K_1(z) = \sum_{j=1}^{\infty}\Pr(k_1 = j)z^j = \frac{1}{\overline{q}}\sum_{j=1}^{\infty} j\,\Pr(q = j)z^j \qquad\qquad (4.11)$$

With the use of the following z-Transform property [29]

$$\sum_{n=0}^{\infty} nf_n z^n = z\frac{d}{dz}F(z) \qquad , \qquad \text{where } F(z) = \sum_{n=0}^{\infty} f_n z^n$$

$K_1(z)$ is obtained as

$$K_1(z) = \frac{1}{q} z \frac{d}{dz} Q(z)$$

where $Q(z)$ is given by 4.2. Substituting from (4.2) into the above, we find

$$K_1(z) = \frac{1}{q} z \frac{d}{dz} \left[\frac{(1-p)z}{1-pz} \right]$$

$$K_1(z) = \frac{1}{q} z \left[\frac{1-p}{(1-pz)^2} \right]$$

With the substitution of (4.3), $K_1(z)$ is expressed as

$$K_1(z) = z \left[\frac{1-p}{1-pz} \right]^2 \tag{4.12}$$

Finally, substitution of (4.6) and (4.12) into (4.8) gives the PGF of the number of cells in segment n,

$$K_n(z) = z \left[\frac{1-p}{1-pz} \right]^2 \left[\frac{1-p}{1-pz} \right]^{2(n-1)}$$

$$K_n(z) = z \left[\frac{1-p}{1-pz} \right]^{2n} \tag{4.13}$$

From (4.12), the average length of the first segment is given by,

$$\overline{k_1} = \frac{dK_1(z)}{dz} \bigg|_{z=1} = \frac{1+p}{1-p} \tag{4.14}$$

Table 4.1 gives the comparison of the numerical and simulation results for the average length of the first segment. The numerical results we obtained through (4.14). As we see, the correspondence between numerical and simulation results is excellent and proves that the tagged block in row one is in fact "special." Hence, the hypothesis that the first segment will have length bias is correct.

Table 4.1: Numerical and Simulation Results for the Average Length of the First Segment as a Function of Cell Connectivity Probability, p

p	NUM	SIM
0.1	19	19.0602
0.3	5.666667	5.6718
0.5	3	2.9912
0.7	1.857143	1.8525
0.9	1.222222	1.2144

4.4 Distribution of the Number of Tagged Blocks in a Row

Next, we will derive the distribution of the number of tagged blocks in a segment. As shown in Figure 4.5 the boundary of adjacent cells is like a gate, if the two cells have connectivity then the gate is opened otherwise it is closed. All the gates in a block are open and on the other hand the closed gates form the borders between the blocks in a row.
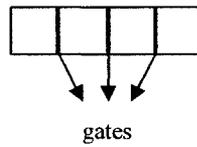
gates

Figure 4.5: Gates of a Block

Next let us consider the cells in segment n that directly interface the cells in segment n-1. The number of these cells will be given by k_{n-1}. Clearly, the number of blocks in segment n will be determined only by these k_{n-1} cells. The extra cells in segment n compared to segment n-1 do not create new blocks in segment n. Let us define,

d_n = number of gates between adjacent cells in segment n within the borders of segment n-1

f_n = number of gates from among d_n which are closed.

Figure 4.6 illustrates these symbols.

$f_n = 2$

row n

row n-1

$d_n = 4$
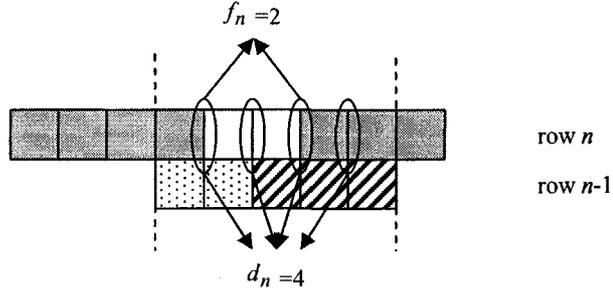
Figure 4.6: Example of how to Calculate the Number of Blocks in Segment n

Then, we have,

$$d_n = k_{n-1} - 1 \qquad (4.15)$$

$$i_n = f_n + 1 \qquad (4.16)$$

Since each gate will be open independent of the other gates with probability p and closed with probability 1-p, the conditional distribution of the number of closed gates is given by the binomial distribution,

$$\Pr(f_n = i \mid k_{n-1} = k) = \binom{k-1}{i}(1-p)^i p^{k-1-i}, \quad i = 0,1,2,\ldots,k-1 \qquad (4.17)$$

The PGF of the binomial distribution may be expressed as

$$F_n(z \mid k_{n-1} = k) = \sum_{i=0}^{k-1} z^i \Pr(f_n = i \mid k_{n-1} = k)$$

$$F_n(z \mid k_{n-1} = k) = [p + (1-p)z]^{k-1} \qquad (4.18)$$

From (4.16), the conditional PGF of the distribution of i_n is given by,

$$I_n(z \mid k_{n-1} = k) = zF_n(z \mid k_{n-1} = k)$$

Substituting (4.18) into the above equation gives the following conditional PGF

$$I_n(z \mid k_{n-1} = k) = z(p + (1-p)z)^{k-1} \qquad (4.19)$$

Finally, the PGF of the distribution of i_n is found by unconditioning (4.19) with respect to the probability distribution of the number of cells in segment n-1,

$$I_n(z) = \sum_{k=1}^{\infty} I_n(z \mid k_{n-1} = k) \Pr(k_{n-1} = k)$$

$$I_n(z) = \sum_{k=1}^{\infty} z(p + (1-p)z)^{k-1} \Pr(k_{n-1} = k)$$

$$I_n(z) = \frac{z}{p + (1-p)z} \sum_{k=1}^{\infty} (p + (1-p)z)^k \Pr(k_{n-1} = k)$$

$$I_n(z) = \frac{z}{p + (1-p)z} \left[\sum_{k=1}^{\infty} z^k \Pr(k_{n-1} = k) \bigg|_{z = p + (1-p)z} \right]$$

$$I_n(z) = \frac{z}{p + (1-p)z} \left[K_{n-1}(z) \big|_{z = p + (1-p)z} \right]$$

Substituting (4.13) into the above equation,

$$I_n(z) = \frac{z}{p + (1-p)z} (p + (1-p)z) \left[\frac{1-p}{1 - p(p + (1-p)z)} \right]^{2(n-1)}$$

$$I_n(z) = z \left[\frac{1-p}{(1-p)(1+p-pz)} \right]^{2(n-1)}$$

$$I_n(z) = z \left[\frac{1}{(1+p-pz)} \right]^{2(n-1)} \tag{4.20}$$

The above gives the PGF of the distribution of the number of tagged blocks in segment n and it is the main result of this section. The average number of tagged blocks in segment n is given by,

$$E[i_n] = \frac{dI_n(z)}{dz} \bigg|_{z=1}$$

$$E[i_n] = \left[[(-pz + p + 1)]^{-2(n-1)} + z(2(n-1)p)[(-pz + p + 1)]^{-2(n-1)-1} \right]_{z=1}$$

$$E[i_n] = 1 + 2(n-1)p \tag{4.21}$$

The above may be written as,

$$E[i_n] = 1 + 2(n - 2 + 1)p$$

$$E[i_n] = 1 + 2(n - 2)p + 2p$$

62

which may be written as,

$$E[i_n] = E[i_{n-1}] + 2p \qquad (4.22)$$

The above equation expresses average number of tagged blocks recursively. As may be seen, the average number of tagged blocks in segment n is higher by $2p$ than in segment n-1, which makes sense since the probability of one or more cells in the right or left hand block is p. As a result of the special length of the border blocks, the average number of tagged blocks in each row increases by the constant amount $2p$.

Figure 4.7 gives the comparison of the numerical and simulation results for the average number of tagged blocks in each row. As we see, the correspondence between numerical and simulation results is excellent.
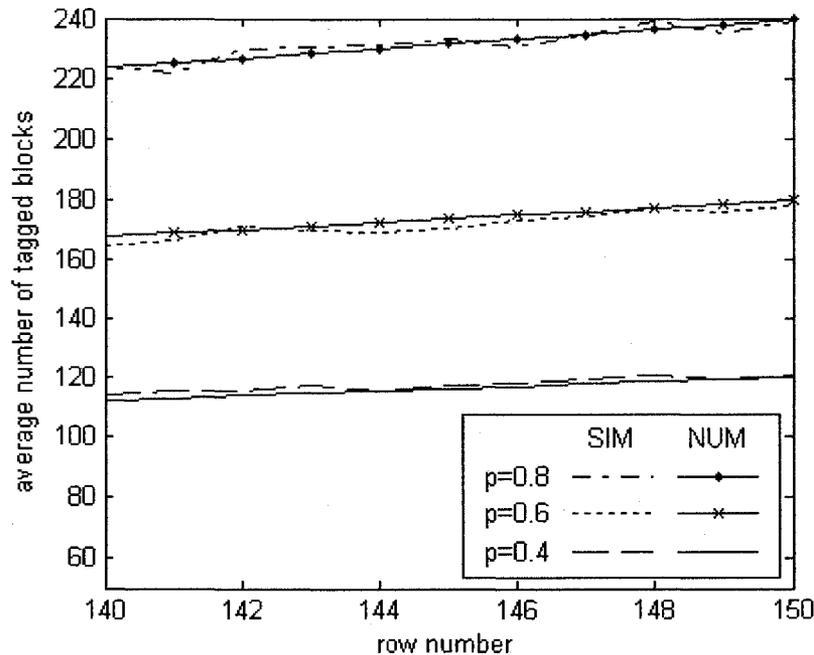
Figure 4.7: Numerical and Simulation Results for the Average Number of Tagged Blocks as a Function of Row Number for Different Cell Connectivity Probability, p

4.5 Length Distribution of the Tagged Blocks

Next, we will determine the distribution of the number of cells in a tagged block. From the previous section, the number of closed gates interfacing the cells in segment n-1 determines the number of tagged blocks in segment n. Since each gate performs an independent Bernoulli trial, then, the length of the blocks will have geometric distribution given by (4.1). As pointed out earlier on, the border blocks of segment n may extend beyond the boundaries of segment n-1 (see Figure 4.4). The distribution of this extension length is given by the geometric distribution given in (4.4). The extension introduces length bias of the renewal processes to the border blocks. As a result, the length distribution of the border blocks is given by (4.10). Figure 4.8 shows the make up of these "special" blocks.
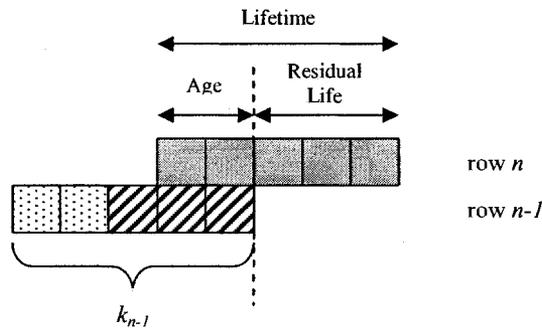
Figure 4.8: Length of Border Blocks According to the Renewal Theory

The simulation results have confirmed that there are two types of tagged blocks: *border* and *non-border* blocks. Figures 4.9 a, b present the mean number of cells as a function of cell connectivity probability, p, in regular and special blocks from (4.3) and (4.14) respectively. In each figure we also plot the corresponding simulation results. As may be seen, the numerical and simulation results have excellent agreement.
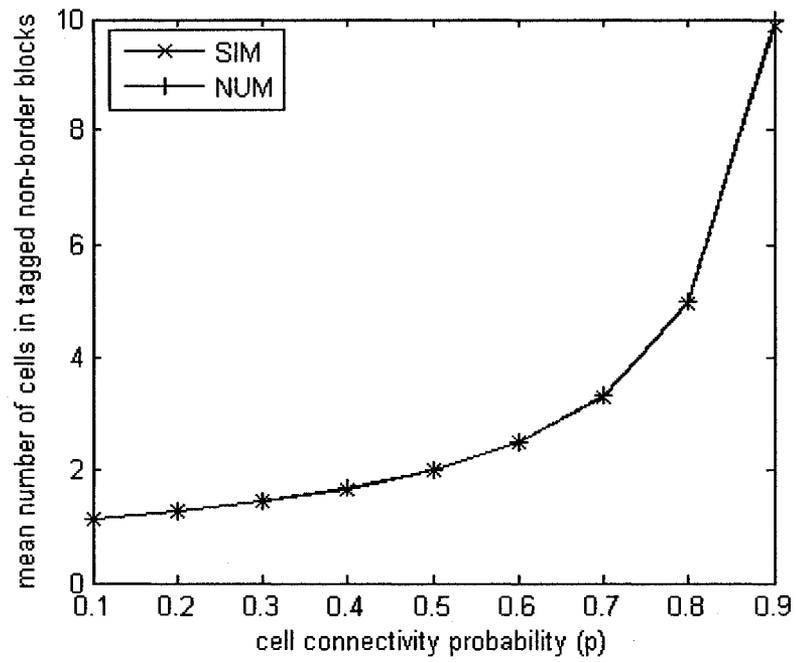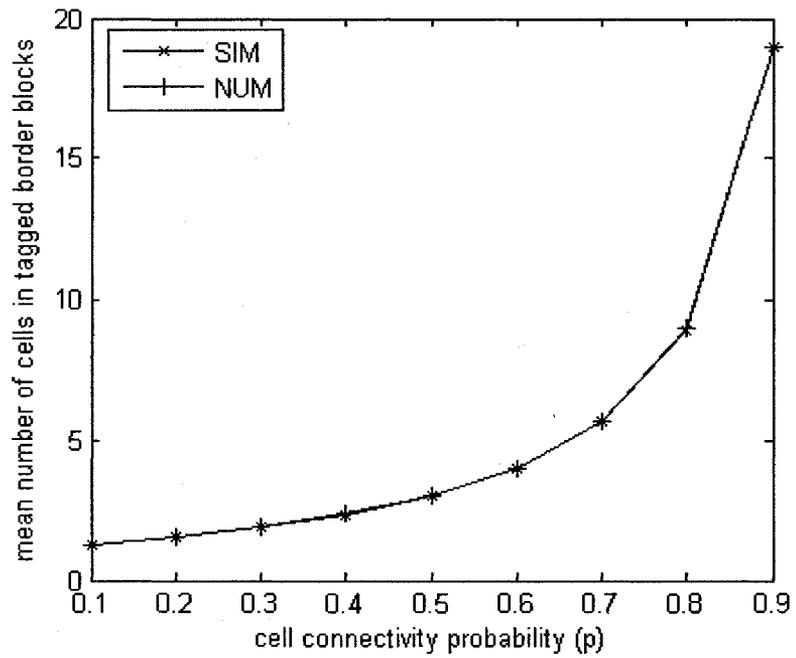
Figure 4.9: Numerical and Simulation Results for the Mean Number of Cells as a Function of Cell Connectivity Probability, p, in Regular (a) and Special Blocks (b)

4.6 Classification of the Tagged Blocks

Next, we will classify the tagged blocks in a segment into two groups as *primary* and *secondary* blocks. Those tagged blocks that have connectivity to the sink will be referred to as primary blocks and the remainder will be referred to as secondary blocks. A tagged block will be identified as a primary block, if and only if, it has connectivity to one or more primary blocks in the row below itself. If at least a single cell of a tagged block has connectivity to a primary block in the lower row, then this tagged block becomes a primary block. In contrast, none of the cells in secondary block connect to a cell in a primary block in the row immediately below. The single block in row one will always be a primary block. If there are no primary blocks in a row, there will not be any primary blocks in higher rows. The cells of primary (secondary) blocks will be referred to as primary (secondary) cells. Consecutive primary cells form paths that allow the flow of traffic towards the sink. There is at least a single path connecting each primary cell to the sink. These paths may merge or split like the tributaries of rivers (see Figure 4.10)

Secondary fragment
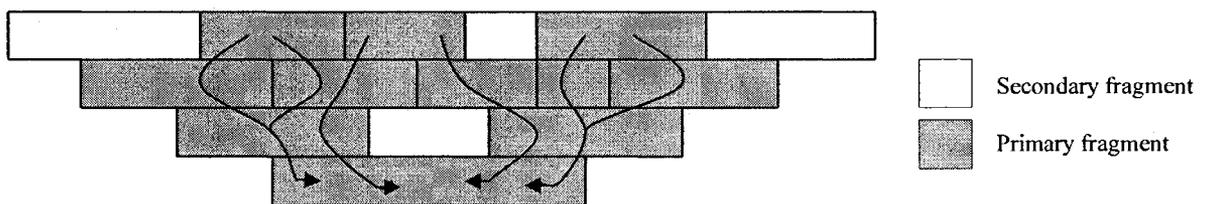
Primary fragment

Figure 4.10: Merging and Splitting Paths Connecting a Primary Fragment to the Sink

Let us define,

x_n = number of primary blocks in segment n.

s_n = number of secondary blocks in segment n.

We note that,

$$i_n = x_n + s_n \qquad (4.23)$$

The primary and secondary blocks will interleave with each other. Let us define a primary (secondary) *fragment* as a number of primary (secondary) blocks which are contiguous. The primary and secondary fragments will alternate with each other. For example, let $x_n=3$ and $i_n=5$, then, the primary blocks may form one to three fragments, as illustrated in Figure 4.11.
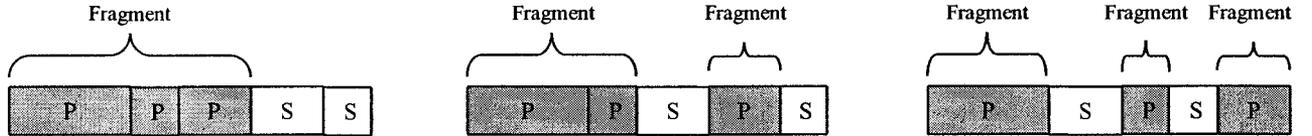
Figure 4.11: Possible Fragments for Three Primary Blocks and Five Tagged Blocks

Next, we will determine average number of primary fragments in segment n which will be used in the following section. Let us assume a constant number of primary and secondary blocks in segment n and then determine the number of ways that these primary and secondary blocks may be partitioned. Let

$\theta(\alpha, m)$ = number of ways that α primary blocks will form m fragments.

$\theta(\beta, f)$ = number of ways that β secondary blocks will form f fragments.

$\theta(\alpha, m)$ and $\theta(\beta, f)$ are found by the following well known recursive relation [30]:

$$\theta(\ell, r) = \theta(\ell - 1, r - 1) + \theta(\ell - r, r) \qquad (4.24)$$

$$\text{where } \theta(\ell, r) = \begin{cases} 0 & for \quad \ell < r \\ 0 & for \quad either \quad \ell < 1 \quad or \quad r < 1 \\ 1 & for \quad \ell = r \\ 1 & for \quad r = 1 \end{cases}$$

$\theta(\ell, r)$ gives the number of ways that an integer ℓ may be partitioned into r fragments where $1 \le r \le \ell$. In our problem, the number of primary and secondary fragments is dependent since they alternate with each other. Clearly, there are three possibilities,

- The segment begins and ends with a primary fragment, then, $f = m - 1$ where f and m are, respectively, the number of secondary and primary fragments.

- The segment begins with a primary fragment and ends with a secondary fragment or the converse of this, then, $f = m$.

- The segment begins and ends with a secondary fragment, then, $f = m + 1$.

Let us define,

$\varphi(m)$ = number of ways that α primary blocks may be partitioned into m fragments given β secondary blocks.

ϕ = total number of ways that α primary and β secondary blocks may be partitioned to.

Then,

$$\varphi\,(m) = \theta(\alpha, m) * \sum_{\substack{f = m-1 \\ \forall f \le \beta}}^{m+1} \theta(\beta, f), \quad \text{for } m > 1 \qquad (4.25)$$

$$\text{where } \varphi\,(1) = \begin{cases} 1, & \text{if } \beta = 0 \\ 0, & \text{otherwise} \end{cases}$$

$$\phi = \sum_{m=1}^{\alpha} \varphi\,(m) \qquad (4.26)$$

Next, we will assume that all the partitions are equally likely to occur which will be justified later on through simulations. Let us define R_m as the probability that α primary blocks may be partitioned into m fragments given β secondary blocks, then,

68

$$R_m = \frac{\varphi(m)}{\phi} \qquad (4.27)$$

Finally, the average number of primary fragments is given by

$$\overline{m} = \sum_{j=1}^{\alpha} j * R_j \qquad (4.28)$$

Next, we assume that the above result also holds when the numbers of primary and secondary blocks in a segment are not constant but chosen to be equal to the average number of primary and secondary blocks in segment n, $\alpha = \overline{x}_n$, $\beta = \overline{s}_n$. Letting $\overline{m}_n$ denote the corresponding average number of primary fragments in segment n, then,

$$\overline{m}_n = \sum_{j=1}^{\overline{x}_n} j * R_j \qquad (4.29)$$

The above assumption will be justified by the simulation results to be presented later on.

4.7 Derivation of the Average Number of Blocks Connected to the Sink

In this section, we will derive a recursive relation for the average number of primary blocks in a segment. First, let us consider the structure of a primary fragment. A primary fragment has the same structure as a segment, it consists of contiguous number of blocks and the lengths of border blocks in number of cells should be larger than that of non-border blocks. However, in practice, the border blocks may not always have the "special" length. Each primary fragment will interface a number of blocks in the higher row which will be called a *frame*. The border blocks of a frame will have the length bias of renewal processes. The blocks of each frame may generate zero, one, two or higher number of primary fragments in the row above. When the border block of a frame

happens to be also border block of a primary fragment, then, that border block will have a special length otherwise it will have a regular length (see Figure 4.12). Thus the special border blocks of a primary fragment have the special, length dependent, distribution given by (4.10), while all the other blocks have the geometric distribution given by (4.1).
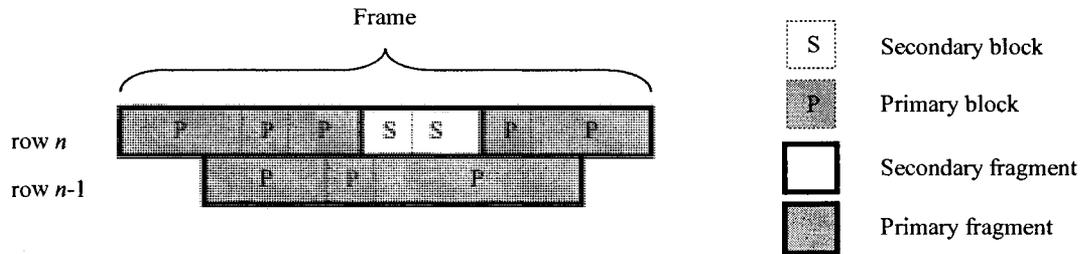
Figure 4.12: Example of a Frame Interfacing a Primary Fragment

Figures 4.13 a, b present the mean number of cells as a function of cell connectivity probability, p, in primary non-border and border blocks of a frame respectively. In each figure we also plot the corresponding numerical results given by (4.3) and (4.14) respectively.
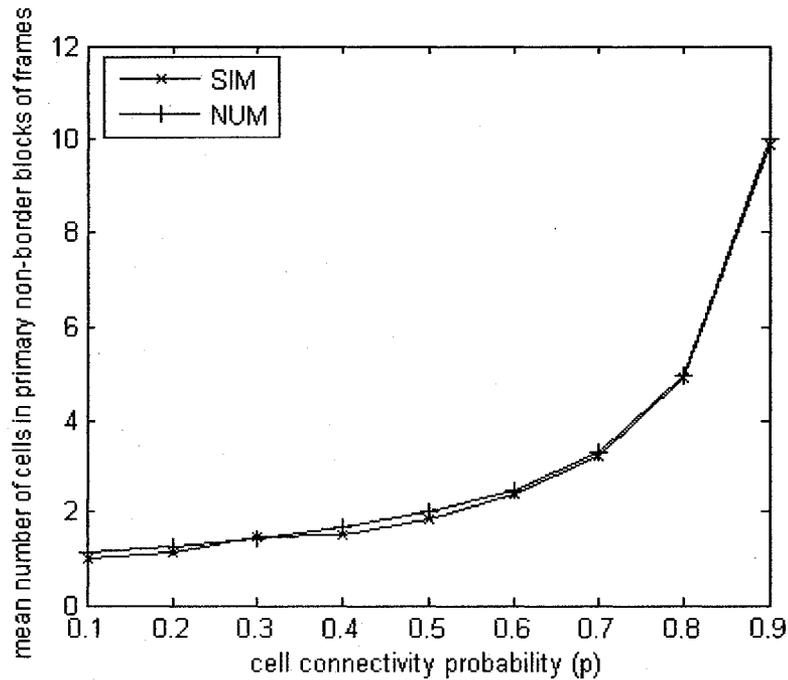
a)
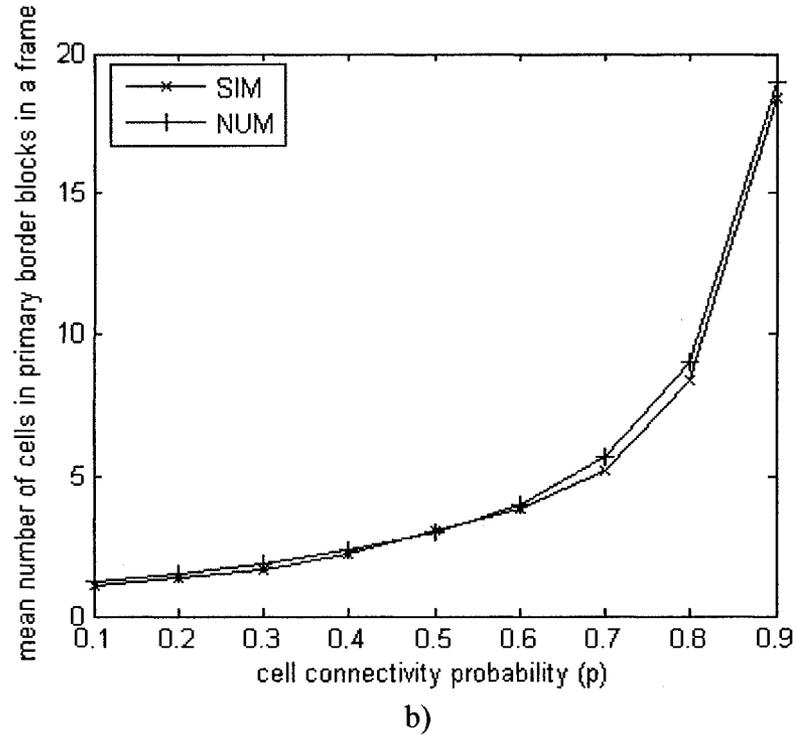
Figure 4.13: Numerical and Simulation Results for the Mean Number of Cells as a Function of Cell Connectivity Probability, p, in Primary Non-Border (a) and Border Blocks (b) of a Frame

Figure 4.14 shows the simulation results for the average size of a border block of a primary fragment. Also shown are the numerical results for the average of the geometric distribution, $1/(1-p)$, average of the geometric distribution with length bias, $(1+p)/(1-p)$, and the arithmetic mean of these two averages. It may be seen that the simulation results are best tracked by the arithmetic mean of the two averages. This corresponds to the case that one of the border blocks of a primary fragment is normal and the other is special confirming our expectation discussed above.
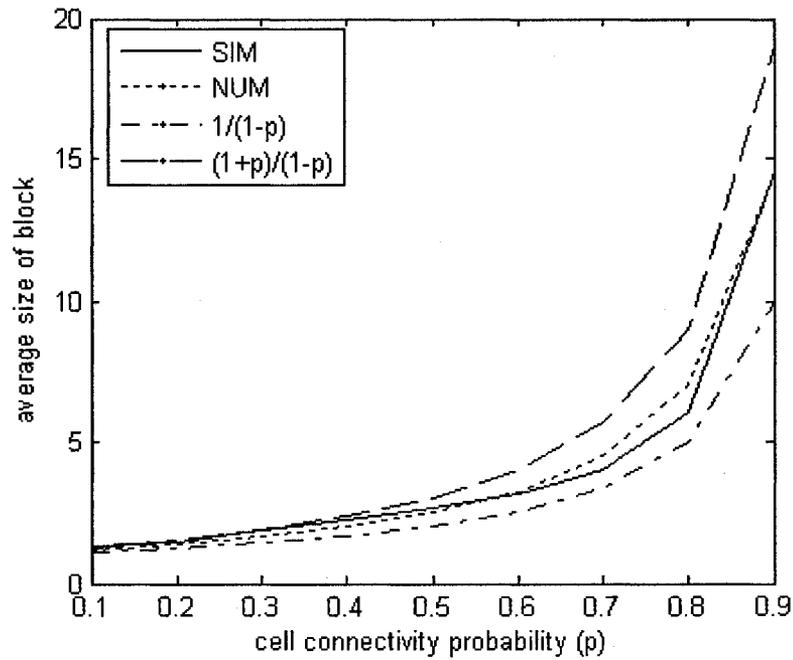
Figure 4.14: Average Size of Border Blocks of the Primary Fragments as a Function of Cell Connectivity Probability, p

Let us define,

$\overline{g}_n$ = mean number of blocks in a primary fragment in segment n.

$\overline{h}_n$ = average number of blocks in a frame in row n.

$\overline{x}_n$ = mean number of primary blocks in segment n.

γ = Pr (a tagged block is a primary block)

We would expect that the following relation holds,

$$\overline{x}_n = \overline{g}_n \overline{m}_n \qquad (4.30)$$

The average number of blocks in primary fragments and frames in consecutive rows will have the same relationship as the segments in consecutive rows. However, since on

average only a single border block of a primary fragment will have a special length, then, (4.22) is modified as follows,

$$\bar{h}_{n+1} = \bar{g}_n + p \qquad (4.31)$$

Let us multiply both sides of the above equation by the number of primary fragments in row n,

$$\bar{h}_{n+1}\bar{m}_n = \bar{g}_n\bar{m}_n + \bar{m}_n p \qquad (4.32)$$

Substituting (4.30) on the right hand side of the above equation,

$$\bar{h}_{n+1}\bar{m}_n = \bar{x}_n + \bar{m}_n p \qquad (4.33)$$

Next, we multiply both sides of the above equation with probability that a tagged block will be a primary block,

$$\bar{h}_{n+1}\bar{m}_n\gamma = (\bar{x}_n + \bar{m}_n p)\gamma \qquad (4.34)$$

We note that the left side of the above equation corresponds to the average number of the primary blocks in segment $n+1$,

$$\bar{x}_{n+1} = (\bar{x}_n + \bar{m}_n p)\gamma \qquad (4.35)$$

Next, we determine the probability that a tagged block is a primary block. According to [27], the portion of the border block partially interfacing the primary fragment in the row below (the length of the age given in Figure 4.8) is the same size as the non-border blocks. Since we are not concerned with the extension length given in (4.4), the average length of the blocks we are dealing with are all normal. Hence,

$$\gamma = \sum_{k=1}^{\infty} \Pr\left(\begin{array}{l} \textit{at least one cell from the} \\ \textit{block has lower connectivity} \end{array} \middle| \textit{a tagged block contains k cells} \right) \Pr(q = k)$$

73

Substituting for the probability distribution of block length from (4.1),

$$\gamma = \sum_{k=1}^{\infty} [1 - (1-p)^k](1-p)p^{k-1}$$

$$\gamma = 1 - \sum_{k=1}^{\infty} (1-p)^k (1-p)p^{k-1}$$

$$\gamma = 1 - (1-p)^2 \sum_{k=1}^{\infty} [(1-p)p]^{k-1}$$

$$\gamma = 1 - \frac{(1-p)^2}{1-(1-p)p} = \frac{p}{1-(1-p)p} \qquad (4.36)$$

The above result enables the calculation of the average number of primary blocks in a segment n recursively with $\overline{m}_n$ given by (4.29).

4.8 Derivation of the Sensor Node Connectivity Probability to the Sink

Finally, we are ready to determine the probability that a sensor node has connectivity to the sink. First, we determine the connectivity probability of a cell to the sink in the first r rows. Let us define,

P_r = Prob(a cell from the first r segments is connected to the sink)

$$P_r = \frac{mean\ number\ of\ primary\ blocks\ in\ the\ first\ r\ segments}{mean\ number\ of\ tagged\ blocks\ in\ the\ first\ r\ segments}$$

$$P_r = \frac{\sum_{n=1}^{r} \overline{x}_n}{\sum_{n=1}^{r} \overline{i}_n} \qquad (4.37)$$

where $\overline{x}_n$ and $\overline{i}_n$ are given in equations (4.35) and (4.22) respectively. Since the nodes are uniformly distributed in deployment area, P_r also gives probability that a sensor node

located within the first r segments will have connectivity with the sink. P_r will be referred to as either sensor or network connectivity probability. We note that sensor connectivity probability is not a constant but a function of the distance of a node to the sink. Another, may be more useful interpretation of P_r, is that it corresponds to the fraction of the nodes that have connectivity to the sink.

Figure 4.15 illustrates the numerical and simulation results of the network connectivity probability (P_r) as a function of the adjacent cell connectivity probability, p, for a network that contains 100 rows. As may be seen, the network connectivity probability is very low for $p<0.5$, it increases linearly in the range $0.5 <p<0.8$ reaching to the value of $P_r = 0.9$. The linear increase continues beyond that but more slowly towards one. The numerical and simulation results show very good agreement which validates the approximations and assumptions made in the analysis.

Figure 4.16 presents the fraction of the nodes that have connectivity to the sink as a function of the row number with adjacent cell connectivity, p, as a parameter. As may be seen for low values of p, the connectivity probability drops to zero quickly with increasing row number. For moderate values of p, the connectivity probability initially drops fast and then at a much slower rate with increasing row number. Finally, for high values of p, following an initial dip, the connectivity probability increases with increasing row number. This is the reason why at high values of p, the percentage of nodes that have connectivity to the sink approaches to one.
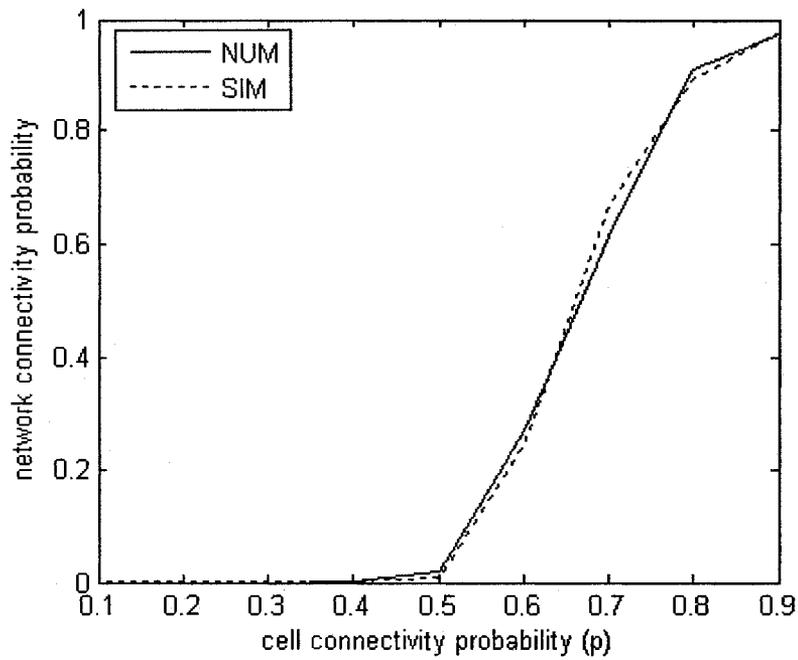
Figure 4.15: Numerical and Simulation Results of the Network Connectivity Probability (P_r) as a Function of Adjacent Cell Connectivity Probability, p, for a Network that Contains 100 Rows
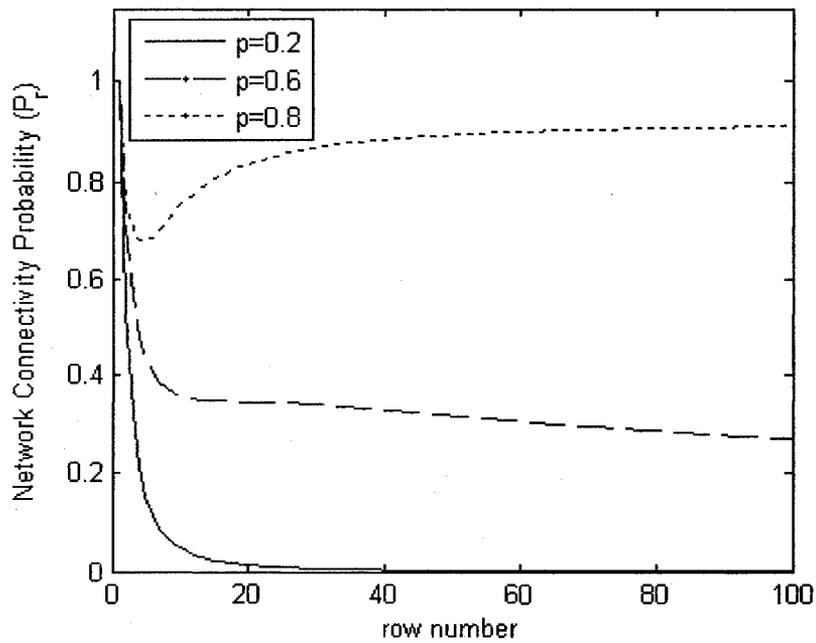
Figure 4.16: Network Connectivity Probability (P_r) as a Function of Row Number

76

CHAPTER 5:

NETWORK LIFETIME

5.1 Introduction

In this chapter, by integrating the results obtained from the previous chapters, we study network connectivity as a function of time. We determine the amount of time that the percentage of the nodes having connectivity is above a threshold value. Before calculating the network lifetime of a WSN, we determine the average number of sensor nodes per cell that are alive over time. In addition, we show that our design inclusively handles coverage.

5.2 Cell Density

In this section, we will determine the average number of sensor nodes per cell that are alive over time, $\xi(t)$. This value depends on the initial sensor node cell density and the probability that a sensor node is alive at a certain time t. The sensor nodes are randomly deployed over an area according to a Poisson process with an average of $\xi(0)$ sensor nodes per cell. Since the sensor nodes are dying randomly and independently, the population size of the live sensor nodes continues to follow a Poisson distribution. Hence, the mean cell density can be determined as,

$$\xi(t) = \xi(0) * P[NL > t] \qquad (5.1)$$

Using eq. (2.25), $P[NL > t]$ can be calculated as follows,

$$P[NL > t] = 1 - F_{NL}(t)$$

Figure 5.1 illustrates the average cell density as a function of time for a network with $\lambda = 15$ messages/hour, $\mu = 150$ messages/hour, and $BL=72$ hours where the initial cell density is 5, 10, 15 and 20 sensor nodes per cell. It may be seen that average cell density remains flat and then sharply drops down.
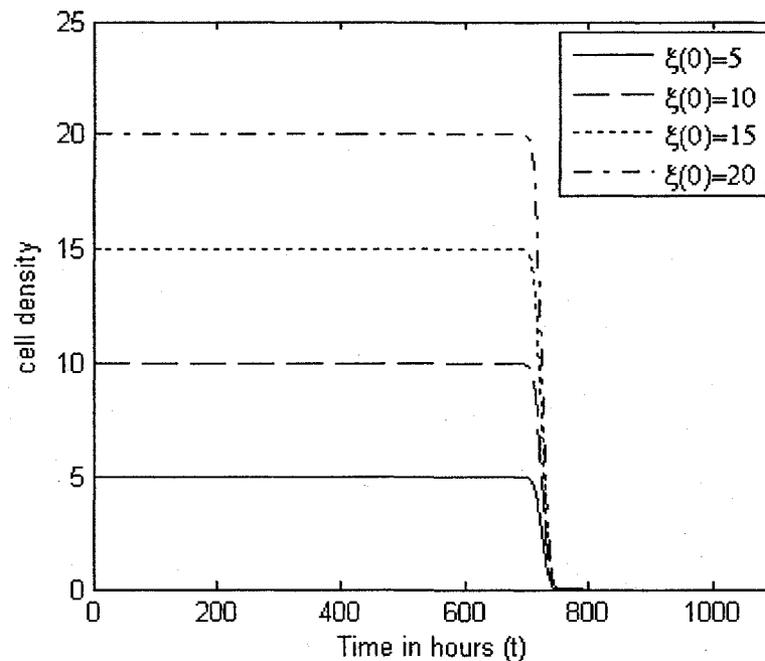
Figure 5.1: Average Cell Density as a Function of Time for $\xi(0) = 5, 10, 15,$ and 20

Table 5.1 illustrates the time it takes for the cell density of the network, with parameter values $\lambda = 15$ messages/hour, $\mu = 150$ messages/hour, and $BL=72$ hours, to drop to 3, 4, 5 sensor nodes per cell given that its initial cell density is 10, 15, and 20 sensor nodes per cell.

Table 5.1: Time it Takes a Network, with λ=15, μ=150, and *BL*=72, for $\xi(t)$ to reach 3, 4, and 5 Sensor Nodes Given Different $\xi(0)$ Values

$\xi(0)$	$\xi(t)=3$	$\xi(t)=4$	$\xi(t)=5$
10	729.0864	726.4931	723.8302
15	732.1055	730.0394	726.957
20	734.0234	732.1897	730.5274

As expected, a network with higher initial cell density will last longer. In addition, the network will last longer if it is serviceable at lower cell densities.

5.3 Network Lifetime

We define the network lifetime as the time that the percentage of sensor nodes can communicate with the sink is above a threshold value. In the following, we assume that all the nodes experience the same traffic load. Therefore, at a certain time t, we first calculate the average number of sensor nodes alive in a cell by using equation (5.1). Then, by using this result and the help of equations (3.9) and (3.10), we determine the lower and upper bounds of the adjacent cell connectivity probability. Then, using these bounds, we evaluate the corresponding bounds of the network connectivity probability, by using equation (4.37). Figure 5.2 illustrates the lower and upper bounds of the network lifetime for a network with λ=15 messages/hour, μ=150 messages/hour, and *BL*=72 hours where the initial cell density is 5 sensor nodes per cell.
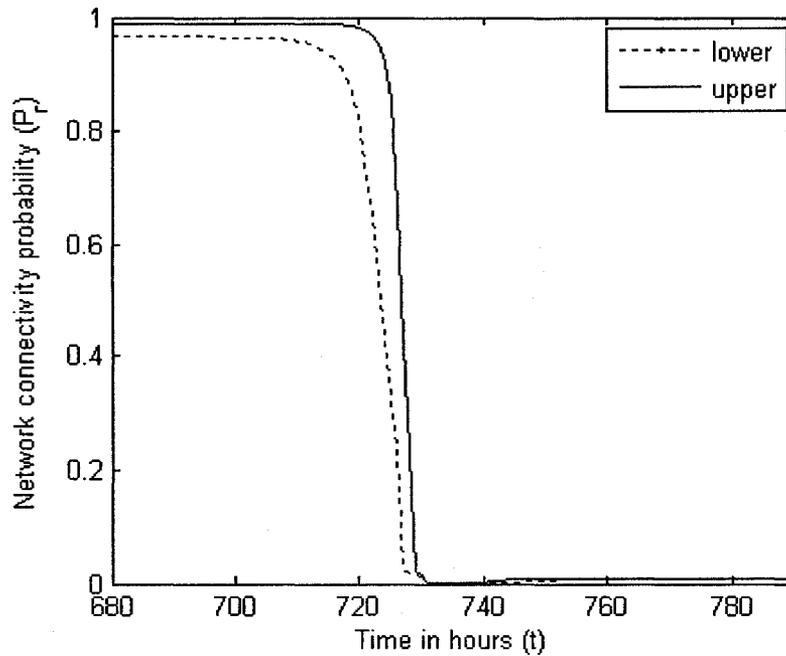
Figure 5.2: Lower and Upper Bounds of the Network Lifetime for a Network with λ =15, μ =150, BL=72, and $\xi(0)$ =5

Table 5.2 gives the lower and upper bounds of the adjacent cell connectivity

probability, p, and network connectivity probability, P_r, for different cell densities for a

network with λ =15 messages/hour, μ =150 messages/hour, and BL=72 hours. The area

coverage is also presented.

Table 5.2: Area Coverage, Bounds of Adjacent Cell Connectivity Probability, p, and Bounds of Network Connectivity Probability, P_r, as a Function of Cell Density for a Network with λ =15, μ =150, and BL=72

Cell density	Lower bound of p	Upper bound of p	Lower bound of P_r	Upper bound of P_r	Area Coverage f_a
2	0.565402	0.677226	0.1876	0.5197	0.79212
3	0.713495	0.882402	0.6715	0.9689	0.90522
4	0.811124	0.961012	0.923	0.9866	0.956786
5	0.875486	0.987442	0.9665	0.9889	0.980297

We have mentioned in chapter 3 that sensor node density less than two would not be of interest. When the cell density is two, the network connectivity ranges between 18.76 and 51.97% and the area coverage is about 80%. Even though the area coverage is mediocre, the network connectivity is too low. In this case, the collected data will probably not be delivered to the sink. For data transfer to be reliable, each cell must contain at least three sensor nodes. Therefore, the results found in this chapter will have area coverage of at least 90%.

We can determine the network lifetime for a specific connectivity threshold. For example, if the network is functional when at least 90% of the sensor nodes are connected, the network lifetime is determined by calculating when the lower network connectivity probability bound drops to 90%.

Figure 5.3 illustrates the lower bound of the network lifetime for a network with $\lambda = 15$ messages/hour, $\mu = 150$ messages/hour, and $BL=72$ hours where the initial cell density is 5,10,15, and 20 sensor nodes per cell.
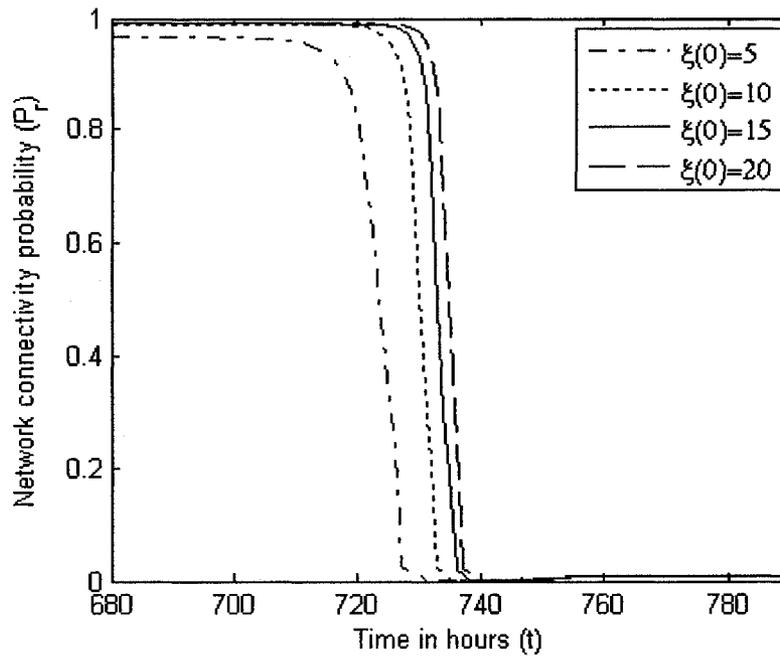
Figure 5.3: Lower Bound of the Network Lifetime, where λ=15, μ=150, **and** BL=72, **for Different** $\xi(0)$ **Values**

Table 5.3 presents the time it takes for the plots in Figure 5.4 for the network to have at least 70, 80 and 90% connectivity. As we can see, when the cell density is higher, the network lasts longer. In addition, if achievable, the lifetime of a WSN can be lengthened by selecting a lower threshold network connectivity.

Table 5.3: Amount of Time that the Percentage of Nodes Have Connectivity is Above a Threshold Value, where λ=15, μ=150, **and** BL=72, **Given Different** $\xi(0)$

$\xi(0)$	90%	80%	70%
5	717.7455	720.2911	721.49
10	727.1427	728.316	729.0144
15	730.4687	731.4457	732.0616
20	732.4965	733.3718	733.9206

CHAPTER 6:

CONCLUSION

6.1 Conclusion

In this thesis, we analyzed the lifespan of a WSN for a non-safety-critical monitoring area application. As time goes on, sensor nodes will die due to battery depletion. We have addressed the problem of determining the point at which the sensor node density falls below a critical density such that coverage and connectivity are lost.

First, the probability distribution of the lifetime of a single sensor node was derived. The sensor node was modeled as an $M/M/1$ queue alternating between idle and busy periods to represent the sleep and active modes, respectively. During the idle period, we assumed that no energy was consumed. Hence, the battery was only drained by the busy periods. The more time a sensor node spends in the busy period, the faster the battery will be depleted. The duration of the busy period depends on the arrival and service rate of a sensor node. Therefore, the lifetime of a sensor node is influenced by the following parameters: battery lifetime, arrival rate, and service rate. The probability distribution of the lifetime of a single sensor node was presented for different parameter values. We realized that the lifetime of a sensor node can be lengthened by increasing its battery's lifetime, increasing the service rate, or decreasing the arrival rate.

Next, the network connectivity probability was determined; in other words, the percentage of sensor nodes that can communicate with the sink was found. We considered a logical grid-network with a large number of sensor nodes randomly deployed over the area, where an average number of sensor nodes were uniformly placed in each cell. Since the transmission range of the sensor nodes was set to the side length of the cell, all the nodes residing in a cell were able to communicate with each other. As a result, if a cell had connectivity to the sink, then all the sensor nodes in that cell had connectivity to the sink. Thus, the problem under consideration reduced to the derivation of the connectivity probability of a cell to the sink. Given an average number of sensor nodes in a cell, the connectivity probability of two adjacent cells was first determined. Obviously, as the density of sensor nodes in the cells increases, the probability that two adjacent cells are connected approaches to one. Based on the connectivity probability of two adjacent cells, we then derived the ability of a cell to communicate with the sink in order to determine the network connectivity probability. Logically, as the connectivity probability of two adjacent cells approaches one, the network connectivity probability approaches one as well and vice versa. The network connectivity probability was examined for different values of the connectivity probability of two adjacent cells.

Finally, the results found for the probability distribution of the lifetime of a single sensor node and the network connectivity probability were integrated to determine the network lifetime. Initially, the number of sensor nodes that reside in a cell have a specified average value. As the WSN ages, sensor nodes start to die, because of battery depletion, causing the number of sensor nodes in the cells to decrease. We derived the

number of sensor nodes that will reside in a cell over time for a given battery lifetime, message arrival rate, and message service rate. From this result, we then determined the connectivity probability of two adjacent cells and the network connectivity probability over time. Lastly, the network lifetime was discovered by calculating the amount of time it takes for the network connectivity probability to reach a specific threshold value. Evidently, choosing parameters to lengthen the lifetime of a single sensor node will also prolong the lifetime of a network. Furthermore, by deploying a higher number of sensor nodes will increase the average number of sensor nodes in a cell which subsequently will increase the connectivity probability of two adjacent cells, network connectivity probability, and network lifetime. Moreover, the value for the threshold connectivity probability affects the network lifetime. A network lasts longer for smaller threshold values. Hence, if a WSN can operate effectively, a lower threshold value should be elected.

This thesis shows that several parameters have an impact on the network lifetime. The results presented in this work can be served as a guideline for WSN designers to optimally choose parameters in order to extend the network lifetime.

REFERENCES

[1]   H. Karl and A. Willig, *Protocols and Architectures for Wireless Sensor Networks*, John Wiley & sons, 2005.

[2]   http://www.informatik.uni-mannheim.de/~haensel/sn_book/chapter1.pdf

[3]   S. Hadim and N. Mohamed, "Middleware: Middleware Challenges and Approaches for Wireless Sensor Networks", *IEEE Distributed Systems Online, Vol. 7, No. 3, March 2006*

[4]   F.Akyildiz et al., "A Survey on Sensor Networks", *IEEE Communications Magazine, Vol. 40, No. 8, August 2002, pp. 102-114*

[5]   F. Zhao and L. Guibas, *Wireless Sensor Networks: an Information Processing Approach*, Morgan Kaufmann, 2004

[6]   E. Callaway, *Wireless Sensor Networks: Architectures and Protocols*, Auerbach Publications, 2004

[7]   B. Nirupama and J. Sanjay, *Wireless Sensor Networks*, Artech House, 2005

[8]   C.S. Raghavendra, K. M. Sivalingam, T. Znati, *Wireless sensor networks*, Springer, 2004

[9]   C. De Morais Cordeiro, *Ad Hoc & Sensor Networks: Theory and Applications*, World Scientific Publishing Co., 2006

[10]   S . Shakkottai, R. Srikant, N. Shroff, "Unreliable Sensor Grids: Coverage, Connectivity and Diameter", *IEEE INFOCOM 2003, Vol. 2, 30 March-3 April 2003 , pp. 1073-1083*

[11]   X. Wang, G. Xing, Y. Zhang, C. Lu, R. Pless, and C. Gill, "Integrated Coverage and Connectivity Configuration in Wireless Sensor Networks." *ACM Sensys '03, Nov.2003, pp. 28-39*

[12]   Y. Liu, W. Liang, "Approximate Coverage in Wireless Sensor Networks", *Proceedings of the IEEE Conference on Local Computer Networks 30th Anniversary (LCN'05), 15-17 Nov. 2005, pp. 68- 75*

[13]   D. Blough and P. Santi, "Investigating Upper Bounds on Network Lifetime Extension for Cell-Based Energy Conservation Techniques in Stationary Ad Hoc Networks", *Proceedings of the 8th Annual International Conference on Mobile Computing and Networking, 2002, pp. 183 - 192*

[14]   Z.Zhou, S. Das, and H. Gupta, "Connected *K*-Coverage Problem in Sensor Networks", *Proc. Of Intl. Conf. on Computer Communications and Networks (ICCCN), 11-13 Oct. 2004, pp. 373 – 378*

[15]   W. Jia, J. Wang, "Analysis of Connectivity for Sensor Networks Using Geometrical Probability", *IEE Proc.-Commun., Vol. 153, No. 2, April 2006, pp. 305 - 312*

[16]   B. Liu, D. Towsley, "A Study of the Coverage of Large-scale Sensor Networks", *2004 IEEE International Conference on Mobile Ad-Hoc and Sensor Systems, 25-27 Oct. 2004, pp. 475 - 483*

[17]   F. Xue, P.R. Kumar, "The Number of Neighbors Needed for Connectivity of Wireless Networks", *Wireless Network, Vol. 10, No. 2, March 2004, pp. 169-181*

[18]   P. Wan and C.W. Yi, "Asymptotic Critical Transmission Radius and Critical Neighbour for k-Connectivity in Wireless Ad Hoc Networks," *Proc. Fifth ACM Int. Symp. Mobile Ad-Hoc Networking and Computing (MobiHoc '04), May 24-26 2004, pp. 1-8*

[19]   J. Zhu and S. Papavassiliou, "On the Energy-efficient Organization and the Lifetime of Multi-hop Sensor Networks", *IEEE Communications Letters, Vol. 7, No. 11, November 2003, pp. 537 - 539*

[20]   J. Liu and T. T. Lee, "A Framework for Performance Modeling of Wireless Sensor Networks", *2005 IEEE International Conference on Communications, Vol. 2, 16-20 May 2005, pp. 1075 - 1081*

[21]   C. Chiasserini and M. Garetto, "An Analytical Model for Wireless Sensor Networks with Sleeping Nodes", *IEEE Transactions on Mobile Computing, Vol. 5, no. 12, December 2006, pp. 1706-1718*

[22]   L. Kleinrock, *Communication Nets: Stochastic Message Flow and Delay*, McGraw-Hill New York 1964, out of print; reprinted Dover, New York 1972.

[23] J.F Hayes and T.V.J. Ganesh Babu, *Modeling and Analysis of Telecommunication Networks*, Wiley 2004

[24] L. Kleinrock, *Queueing Systems, Volume I: Theory,* Wiley Interscience, New York, 1975, pp.206-216.

[25] F. Oberhettinger, L. Badii, *Tables of Laplace Transforms*, Springer-Verlag, Berlin, 1973, p.242.

[26] Alberto Leon-Garcia, *Probability and Random Processes for Electrical Engineering,* Addison-Wesley, Reading, MA, 2nd Edition, 1994, pp.121-122.

[27] Randolph Nelson, *Probability, Stochastic Processes, and Queuing Theory: The Mathematics of Computer Performance Modeling*, Springer-Verlag, New York, 1995

[28] S. Kumar, A. Arora and T. H. Lai, "On the Lifetime Analysis of Always-on Wireless Sensor Network Applications", *2005. IEEE International Conference on Mobile Adhoc and Sensor Systems Conference, 7-10 Nov. 2005*

[29] L. Kleinrock, *Queueing Systems, Volume II: Computer Applications*, John Wiley & sons

[30] N.M. Chase, "Global Structure of Integer Partition Sequences", Ph. D., Massachusetts College of Pharmacy and Health Sciences, 2003