Energy Efficient Spectrum Aware Clustering for Cognitive Sensor Networks

Akrem Latiwesh

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 and Computer Engineering) at

Concordia University

Montreal, Canada

April 2016

CONCORDIA UNIVERSITY
SCHOOL OF GRADUATE STUDIES

This is to certify that the thesis prepared

By: Akrem Latiwesh

Entitled: "Energy Efficient Spectrum Aware Clustering for Cognitive Sensor Networks"

and submitted in partial fulfillment of the requirements for the degree of

Master of Applied Science

Complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

____ Chair
Dr. R. Raut

____ Examiner, External
Dr. A. Youssef (CIISE) To the Program

____ Examiner
Dr. A. Agarwal

____ Supervisor
Dr. Dongyu Qiu

Approved by: ____
Dr. W. E. Lynch, Chair
Department of Electrical and Computer Engineering

____________20_____ ________________________________
Dr. Amir Asif, Dean
Faculty of Engineering and Computer Science

ABSTRACT

Energy Efficient Spectrum Aware Clustering for Cognitive Sensor Networks

Akrem Latiwesh

Combining cognitive radio technology with wireless sensor networks has been introduced in the literature as a solution to the spectrum deficiency problem. Many clustering algorithms have been proposed for wireless sensor networks. However, most of them are not suitable for cognitive sensor networks as they operate on a fixed channel settings. In this work, we propose a low energy spectrum aware clustering algorithm, CogLEACH-C, based on CogLEACH algorithm in order to improve the performance in terms of system lifetime.To accomplish that, energy level of each sensor node should be taken into account during the selection process. CogLEACH-C uses not only the number of channels sensed idle by the node but also the node energy level in determining the probability for each node to be a cluster head. Hence, allows the base station to select K cluster heads that are in a better position for the nodes in the network.

Moreover, as the network consists of primary users (PU) and secondary users sharing the same channels with the priority for the primary users, secondary nodes that are within the coverage of primary users will sense a low number of idle channels depending on the primary user's activity. As a result, those nodes will experience difficulty to connect with a cluster head and that affects the performance of the network.In particular, when the probability of a channel to be idle is low and the majority of the channels are within the coverage of the PUs. We provide an analysis of the probability of connectivity of a sensor node in cognitive wireless sensor network when CogLeach-

C algorithm is used, and we show how that affects the performance of the network in terms of delay and network lifetime.

ACKNOWLEDGEMENTS

I would like to express my gratefulness to Dr. Dongyu Qiu for his continues guidance and help throughout this research. I would also like to thank my family for their support.

Akrem Latiwesh

Contents

List of Figures

List of Tables

List of Symbols

P_i	probability of the i^{th} node to be a cluster head
K	number of clusters
C_i	number of channels sensed idle by the i^{th} node
C_T	total number of channels sensed idle by all the nodes
q_i	probability of the i^{th} channel to transit from busy mode to idle mode.
p_i	probability of the i^{th} channel to transit from idle mode to busy mode.
p_f^i	stationary probability of the i^{th} Channel
m	Number of Channels
E_i	residual energy of the i^{th} node
E_T	total residual energy in the network
N	number of nodes in the network
M	width of the area covered by the network
d	width of one cell in the grid
P_c	connectivity probability of a node to the sink

X	random variable represents the number of common channels sensed idle by two nodes
A_{sec}	area covered by secondary user
A_{total}	area covered by the network
p_{pu}	probability that a primary user is in the coverage of a secondary node
p_{pui}	probability that there is at least one primary user in the coverage of a secondary node
r_{sec}	diameter of the area covered by a secondary node
N_{pu}	number of primary users in the network
p_{idle}	probability a channel is sensed idle
p_{f1}	probability a channel is sensed idle by a node that has primary users in its coverage.
p_{f2}	probability a channel is sensed idle by a node that does not have primary users in its coverage.
p_{cin}	probability that a node is connected with a cluster head that is located in the same cell.
k_{pu}	number of primary users in the coverage of a secondary node
x_a	x coordinate of point a
x_b	x coordinate of point b
y_a	y coordinate of point a
y_b	y coordinate of point b

$f_{x_a}(x_a)$	probability density function of x_a
$f_{x_b}(x_b)$	probability density function of x_b
$f_{y_a}(y_a)$	probability density function of y_a
$f_{y_b}(y_b)$	probability density function of y_b
$f_{x_a,x_b,y_a,y_b}(x_a,x_b,y_a,y_b)$	joint probability density function of x_a,x_b,y_a and y_b
D	distance between a node and a neighbour cluster head
p_{cn}	probability that a node is connected with a neighbour cluster head
p_{ch}	probability of successfully having a cluster head located in a d^2 area
c_n	number of neighbouring cells
p_{chn}	probability of successfully having a cluster head located in a $c_n * d^2$ area
n_{i+1}	number of messages in the departure of the $i+1$ message
n_i	number of messages in the departure of the i message
a_{i+1}	number of arrivals during the service time of message $i+1$
$I(i)$	an indicator of node connectivity to the sink.
λ	arrivals rate
μ	service rate
$p(z)$	probability generating function of number of messages in the system
$A(z)$	probability generating function of number of arrivals
$m(t)$	probability density function of service time

$D(s)$	Laplac transform of the probability density of message delay
ρ	system load
$d(t)$	probability density of message delay
$\overline{d}$	average message delay
n	number of cycles until the nodes dies
n_{idle}	number of idle periods
$\overline{BS}$	average time of busy periods
n_{busy}	number of busy periods
N_{life}	node lifetime
B_{life}	buttery lifetime

List of Abbreviations

ADC	Analog to Digital Converter
BS	Base Station
CH	Cluster Head
CogLEACH	Cognitive Low-Energy Adaptive Clustering Hierarchy
CR	Cognitive Radio
CRWSN	Cognitive Radio Wireless Sensor Networks
GPS	Global Positioning System
HEED	Hybrid, Energy-Efficient, Distributed
LEACH	Low-Energy Adaptive Clustering Hierarchy
PUs	Primary Users
TDMA	Time Division Multiple Access
WSNs	Wireless Sensor Networks

Chapter 1

Introduction

1.1 Introduction

Communication networks, wired or wireless, have become a vital part of our life, as they have been applied to many facilities such as, education, health care, banking etc. Frequency spectrum and energy are deficient and expensive resources in wireless networks. In particular, microsensor networks where battery size is small and energy level is limited, energy consumption becomes a crucial factor.

The use of static spectrum assignment is a technique that provides licensed users with the right of using the licensed spectrum, in away that it can not be shared with other users. Radio spectrum is a finite resource. Therefore, with the increased demand on wireless services and with the use of static spectrum assignment, spectrum scarcity has been a major issue. Cognitive radio has been introduced to overcome spectrum deficiency in WSNs . In cognitive radio, primary users are allowed to use the licensed band, while secondary users have the right to utilize the licensed channels opportunistically.

Clustering is a technique used to gather sensor nodes into clusters to improve the overall network scalability, network lifetime and energy consumption. Since WSNs should operate as long as possible, system lifetime is one of the important goals of implementing clustering algorithms in WSNs. Due to the difficulties in recharging or replacing nodes' batteries, it is essential that

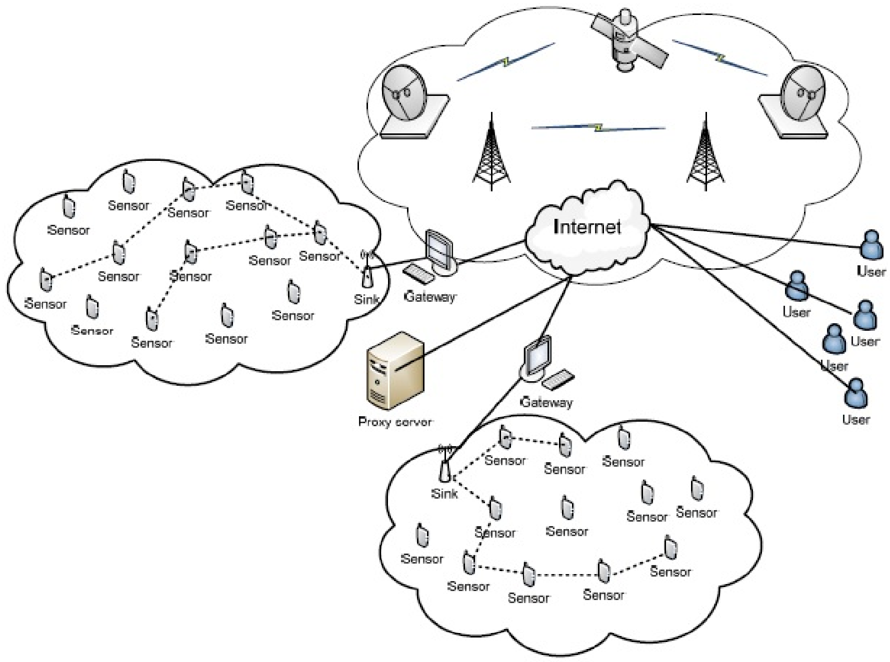

Figure 1.1: Traditional Wireless Sensor Networks

clustering algorithms be energy efficient as much as possible. Clustering for CRWSNs implies more constraints during the design of the algorithms, as dynamic spectrum access has to be addressed.

1.2 Traditional wireless sensor Networks - WSNs

Wireless sensor networks are constructed by groups of wireless sensor nodes. Those nodes, produce a bursty data whenever an event occurs. Therefor, they are described as an event-driven devices [2]. [1] describes a WSN as an ad hoc network that contains number of sensor nodes, which are distributed over the targeted area randomly or uniformly.

However, wireless sensor nodes are deployed over the given area with constrains on the distance separating them from each other. When a sensor node detects an event, it generates the corresponding data. Then, the sensor node sends the data to the base station (sink node), which responsible on delivering the collected data from the nodes to the user via a gateway. This kind of transmission usually occurs over the Internet or another channel as shown in figure. 1.1 [2].

WSNs operate in the unlicensed band, which is overcrowded, as it is used by many other technologies. The coexistence with many other applications operating on the same band can affect the performance of WSNs. In particular. when WSNs coexist with WiFi networks, which has high transmission power[2].

1.3 Cognitive Radio as a Solution

Adapting cognitive radio (CR) with WSNs has been proposed in the literature as a solution for spectrum scarcity issue [1-7].

CR can alter its operating parameters based on what it learns from the field where it is operating, which is a main advantage for this technology. Moreover, CR has the ability to determine unused spectrum and make it available for unlicensed users (secondary users) opportunistically, with keeping the priority of the licensed spectrum for the licensed users (primary users). On other words, the channels that are being used by primary users can not be utilized by secondary users until the primary users vacate them. Applying such a technique in WSNs can improve energy efficiency.

Cognitive radio enabled sensor nodes have additional capabilities compared to traditional wireless sensor nodes such as, detecting unutilized white space in the spectrum band, using the unutilized white space and sharing information with other sensors, and distributing spectrum usage opportunistically between sensor nodes.

1.4 Cognitive Radio Wireless Sensor Networks - CRWSNs

Several conceptual designs for CRWSNs have been introduced in literature [1,7]. A CR-WSN consists of a large number of wireless sensors, which are provided with cognitive radio capabilities and distributed over a given area. Cognitive radio sensors are self-organized and selfaware devices. However, those nodes still have the same energy limitation issue as in traditional sensor nodes.

In CRWSNs, a secondary node senses the spectrum looking for vacant channels and uses the best channel available for it. However, as the priority for using the licensed spectrum is for the primary users, secondary nodes have to vacate the channel they are using in case a primary user wants to use the channel.

1.4.1 Advantages of CRWSNs

Adapting CR with the traditional wireless sensors can bring some advantages, which discussed below [2-4,6]

1.4.1.1 Dynamic Spectrum Utilization:

WSNs operate in the unlicensed band, which is also called industrial, scientific and medical radio bands. In addition, WSNs use fixed channel settings over highly utilized bands, which lead to a decrease in the performance. To operate in a licensed band, the right of using that band has to be purchased from the country where the network is deployed, which comes with a really high cost. Enabling Cognitive radio in WSNs will give them the ability to share a licensed band with a licensed user without the need for spectrum usage purchase.

1.4.1.2 Opportunistic Usage of Available Channels

When traditional sensor nodes detect an event, they generate a traffic that is bursty in nature. Then, they try to secure access to the same channel to send the generated data to the sink node. As a result, the chance of having high number of collisions will increase as all the nodes are contending to use the same channel, which affects the overall communication reliability and increases packet losses. CR can decrease the number of collisions as sensor nodes will have the ability to operate on multiple channels.

1.4.1.3 Decreasing Energy Consumption

So much energy is wasted in WSNs as a result of high number of collisions and retransmissions. That waste of energy can be alleviated by adding CR capabilities to sensors nodes, as it enables sensor nodes to adjust their parameters to variable channel conditions.

1.4.1.4 Operating Independently of Spectrum Usage Rules

Different countries use different regulations for spectrum utilization. Therefore, it is difficult to deploy a WSN that has it's operating frequency already set in different countries. CR can resolve that issue as it is able to operate in any country, and that is due to it's ability of changing frequency band.

1.4.1.5 Coexistence of Multiple WSNs

Wireless sensor networks that overlap spatially can coexist efficiently by enabling CR in their sensor nodes. That is due to secondary users ability of sharing the use with primary users of the licensed band.Hence, interference can be alleviated.

1.4.2 Architecture of CR Wireless Sensor

Conceptual designs of cognitive wireless sensor node were introduced in [1,2 and 4], Figure 1.2 [2]. Some of the main components are discussed below.

1. **Sensing Unit**

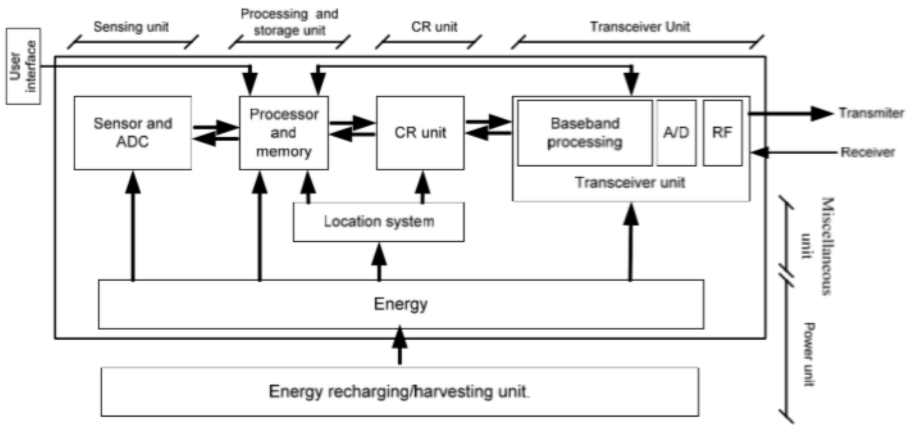

Figure 1.2: Conceptual Design of Cognitive Wireless Sensor Node

This unit is responsible on sensing the spectrum and it contains an ADC, which is responsible for converting the signals that result from the sensing process from analog to the digital.

2. **Cognitive Radio Unit:**

   Cognitive radio unit is responsible on adjusting operation parameters dynamically. In addition, it is responsible on managing spectrum sharing between the nodes and finding the best vacant channel for the node. Moreover, Cognitive radio unit makes secondary users vacate the channel in the case of primary user arrival.

3. **RF Unit:**

   This unit contains a transceiver, which is responsible on receiving and sending analog signals.

4. **Energy Unit:**

   This unit represents the battery, which supplies all the components of the sensor node with power.

Each sensor node has a processing unit that is responsible on manipulating the signals received from the sensing unit and from the cognitive radio unit. Charging unit, miscellaneous unit and location unit are optional and they depend on the application. Although, many designs have been introduced to cover the basic characteristics of CR, it is really sophisticated to design the hardware of CRWSNs. However, designing the hardware of CRWSNs depends on different principles, such as security, sophistication, amenability [2].

1.4.3 Cognitive Radio Wireless Sensor Network Topologies

Cognitive radio wireless sensor networks are application specific. As a result, different kinds of topologies have been propose based on the requirements of the application.

1.4.3.1 Clustered Topology:

a clustered topology is suitable for effective management of the dynamic spectrum and the energy in cognitive radio wireless sensor networks. In this kind of topology, it is essential to use a specific channel allocated only for the purpose of exchanging management and control messages. Those messages could carry information about spectrum sensing, spectrum usage, and primary users arrival. It is difficult sometimes to find such a channel with specific applications. Nevertheless, the use of space correlation can increase the potential of dedicating a channel only for control and management purposes.

In clustered topology, wireless sensor nodes alternate to be cluster heads in each round. the selection process can be done individually in away that each node decides its rule by itself or it can be centralized. Nodes that are selected to be cluster heads in a certain round are responsible for collecting the data from all the ordinary nodes in the same cluster and sending the collected data to the sink node (the base station-BS). However, other duties can be assigned for cluster heads like spectrum bargaining. Figure 1.3 shows a cluster based CRWSN [4].

1.4.3.2 Hierarchical Heterogeneous Topology:

A cognitive radio wireless sensor network with this kind of topology contains regular cognitive radio sensors and high powered cognitive radio sensors. Those high powered nodes are

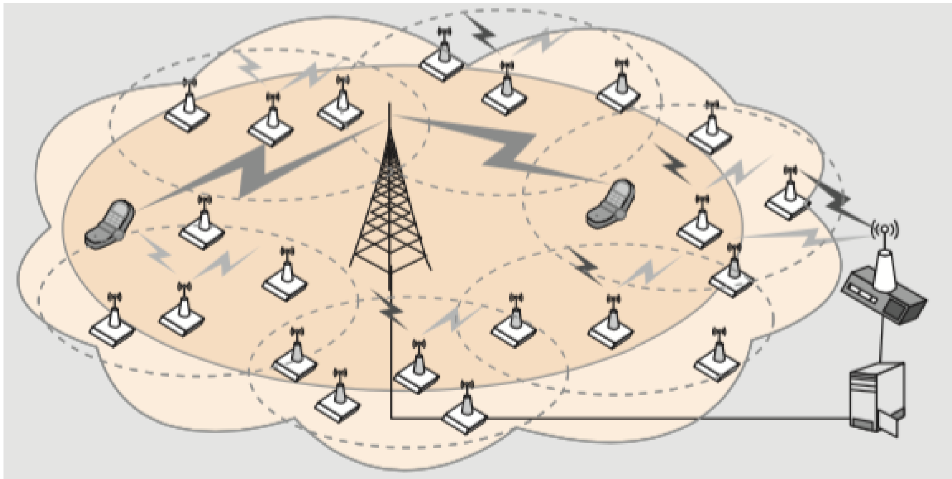

Figure 1.3: Traditional Wireless Sensor Networks

responsible on sending the data to the sink node. Using such a topology implies new challenges due to the existence of the high powered nodes. The challenges are represented in the difficulty of deploying the high powered sensors and their efficient use of the spectrum.

1.4.3.3 Ad Hoc Topology:

This type of topology does not need infrastructure. Nodes perform spectrum sensing separately or cooperatively. Moreover, nodes use multihop routing techniques to deliver the data to the BS.

1.4.4 Challenges of CRWSNs:

CRWSNs have new features that differ them from traditional WSNs. Hence, new challenges rise by adapting cognitive radio with WSNs and we discuss some of them below [1,2].

1. **Clustering:**

   many algorithms have been proposed for clustering in WSNs. However, none of those algorithms is suitable for CRWSNs as they do not address the dynamic spectrum access. As a result, new clustering algorithms need to be designed for CRWSNs.

2. **Network Deployment:**

   The existence of primary users, which have the priority of using the spectrum, made it indispensable to develop effective deployment techniques for secondary users, which lead to prolonging the lifetime of the network and enhancing communication reliability.

3. **Node Operation in Dynamic Spectrum Environment:**

   Each node in the network has to perform several tasks, such as channel detection, spectrum hand-off, channel allocation and spectrum sharing. Those tasks can be accomplished in an individual fashion or cooperative fashion depending on what is the best for using the limited resources of the network.

4. **Quality of Service:**

   Besides the factors used in WSNs to determine the quality of service, CRWSNs has an additional factor, which is protecting primary users' right of use of the spectrum. Any violation of that right affects the performance of the network, and that creates more challenge.

1.4.5 Clustering in CRWSNs

Clustering is a technique used to gather sensor nodes into clusters to improve the overall network scalability, network lifetime and energy consumption. Since WSNs should operate as long as possible, system lifetime is one of the most important goals of implementing clustering algorithms in WSNs. Due to the difficulties in recharging or replacing nodes' batteries, it is essential that clustering algorithms be energy efficient as much as possible. Clustering for CRWSNs implies more constraints during the design of the algorithms, as dynamic spectrum access has to be addressed.

Several factors should be taken into account during the design process of clustering algorithms. clustering algorithms must be able to gather sensor nodes in the best possible formation to reduce energy consumption needed to transmit the data, and the number of control messages exchanged to create those clusters must be as low as possible. Moreover, clustering algorithms should support variable application requirements and they should support secure transmission of data.

1.4.5.1 Clustering Factors:

Below, we explain some of the important parameters used in the design process of clustering algorithms.

- **Number of Clusters:** The number of formed clusters can be random or predetermined. This number has an impact on data routing and also on energy consumption. A very high or a very low number of clusters can lead to an increase in energy consumption. Therefore, some solutions have been proposed in the literature to find the optimal number of cluster heads that gives the best performance[2,11].
- **Communication Inside the Cluster:** The communication between the nodes and the cluster head inside a cluster can be done in single hop or multihop. Selecting the appropriate way depends on the range of sensor nodes and also the number of sensors available in the network.
- **Type of Nodes:** Usually most of nodes has the same characteristics and rules. However, in some topologies that can change. For example, in the heterogeneous typology, some of the nodes have higher power than the others and they have additional rules.
- **Cluster Heads Selection:** The selection of cluster heads can vary from one algorithm to another. Those cluster heads can be predetermined in case the network has some nodes with higher energy. Moreover, cluster heads can be selected based on a certain probability calculated for each node or in a completely random fashion.
- **Algorithm Complexity:** The time needed to implement the algorithm and forming the best clusters is one of the main parameters in the designing process of clustering algorithms.
- **Different Levels of Clusters:** Clusters can be formed in different level. For example, if two levels of clusters are formed. nodes send the data to the cluster heads in the first level. Then,

those cluster heads, which are considered regular nodes in the second level, send the data to the cluster heads in the second level, which will send the aggregated data to the sink node.

1.5 Literature Review

Next, we provide a brief discussion of the past work in clustering algorithms and lifetime analysis for sensor networks.

CogLEACH [1] is a spectrum aware algorithm and an extension of Low Energy Adaptive Clustering Hierarchy (LEACH) protocol [10]. It uses the number of channels sensed idle by the node as the main coefficient instead of the number of rounds used in LEACH. It uses Eq.(1) to determine the probability of each node to become cluster head.

$$P_i = min(K\frac{C_i}{C_T}, 1) \tag{1.1}$$

where :

$$C_T = \sum_{i=1}^{N} C_i,$$

K is the number of cluster heads in the network in each round. N is the number of nodes in the network. C_i represents the number of channels sensed idle by node i.

As a result, nodes that sense a higher number of idle channels have an increased probability to become cluster head. Using this procedure in selecting cluster heads improves spectrum utilization. However, it does not take energy level of the nodes into account. In other words, nodes that have a high number of idle channels and low residual energy can be cluster heads, which leads to a faster death of the node.

LEACH-C [2] is a centralized clustering algorithm for wireless microsensor networks that uses the residual energy of the node as a weight of finding the probability of each node to be a cluster head. This algorithm can not be used in Cognitive sensor networks as it was designed for fixed channel settings.

In [6], an event driven clustering algorithm is introduced. Clusters are formed when there is an event. As well, nodes are considered eligible for clustering depending on their position to the event and the BS. Cluster heads are selected based on their degree, number of channels and distance to the BS. This algorithm needs intensive message exchange and it is not practical in very active environments.

Both HEED protocol [8] and LEACH-B protocol [9] are designed for fixed channel settings and were proposed as enhancements for LEACH protocol in terms of prolonging the lifetime of the network. HEED selects cluster heads depending on energy level and node proximity to its neighbours. LEACH-B uses a two-stage selection method to elect cluster heads. Firstly, LEACH-B chooses cluster heads as LEACH protocol does. Secondly, a second selection is performed depending on the residual energy of the selected nodes.

In [15], an analysis for network connectivity probability was done. The network was divided into blocks. Then, The probability of connectivity for a single block was determined. The probability of connectivity for the entire network then was found gradually by gathering some blocks creating a bigger block to which the probability of connectivity was obtained. A geometric probabilistic analysis was used to obtain the results.

Two models were developed in [16], one for sensor nodes, which is a queueing model, and another for the entire network, which is a performance model. The aim of the analysis is to shed a light on the trade-offs of adapting a random active-sleep model and network performance. This work showed that the use of sleep-active model has a major effect of energy consumption and packet delay in the network.

In [17], a grid approach was used to determine the probability of connectivity of wireless sensor networks using the node density in a block. It assumes that nodes deliver their data to the sink node by passing the data from one node to another until it reaches the sink. To achieve the goal of this work, The probability of a node to be connected to the sink was obtained by determining the probability that the node is located in a path to the base station. Moreover, the probability of two adjacent nodes was introduced.

1.6 Research Objectives

The objective of this work is to enhance CogLEACH algorithm, which is a clustering algorithm designed for CRWSNs based on the well-known LEACH algorithm, in a way that prolongs the lifetime of the network. First, we try to improve cluster heads selection process . In CogLEACH, each node decides its rule in a certain round based on a probability calculated by the node itself. That probability is compared to a certain threshold, and if it is higher than the threshold then the node is a cluster head. The main factor in calculating that probability is the number of channels sensed idle by the node, which means nodes with high number of idle channels can be cluster heads even if they have a low level of residual energy. To improve that, we included the average residual energy in the calculation of each node's probability to be a cluster head or not.However, Nodes can not determine the average residual energy in the network in each round. As a result, the algorithm has to operate in a centralized fashion.

Efficient energy consumption is a main key in prolonging the lifetime of the network. To take advantage of the centralized functionality of the algorithm, we made the base station does some of the functions that were allocated to the sensor nodes. In each round nodes send their information to the base station, which determines the probability of being cluster head in each round. In addition, to reduce the communication needed between the cluster heads and the other nodes to form the clusters, we allow the base station to form the clusters on behalf of the nodes using K-Neighbours algorithm. Moreover, it has been introduced in the literature that using high number of clusters can lead to an increase in energy consumption, which means a faster death to the network. Therefore, we give the right to the base station to select K cluster heads from those who had a probability higher than the threshold, in case they were more than K nodes.

In CRWSNs, secondary sensor nodes exist in the same area with primary users, which have the priority to use the licensed band. If a channel is being used by a primary user, a secondary user can not use that channel. Moreover, if a secondary user is using a certain channel, it has to leave that channel in case a primary user arrived to the channel. That can leave secondary users unconnected for certain rounds even if they have data to send to the base station, which can affect the performance of the network. In particular, if there is a high number of primary users and they have high probability to be active. Therefore, we develop an analysis for CRWSN when CogLEACH-C is used for clustering. We provide an analysis for the connectivity probability of a node in the three modes proposed in CogLEACH algorithm, which are arbitrary model, spectrum similarity model and spatial-spectrum similarity model. To do so, we first divide the network into blocks were nodes are spread uniformly over the covered area. Then, we find the probability of a node to be connected with a cluster head that exists in the same block, which is the probability of having at least one intersection between the two nodes in terms of the channels sensed idle by both of them. After that, we find the probability of a node to be connected with a neighbouring cluster head, which is the probability that there is an intersection between the two nodes and the cluster head is withing the coverage of the node. Finally, we find the probability of a node to be connected with the sink, which is the probability of a node to be connected with a cluster head in the same block or a neighbouring cluster head.

Next, the probability of connectivity for the node is used to show its impact on network performance. To do so, we analyze a sensor node as M/M/1 queue and we determine the probability density function of message delay in the system. We compare the results we obtained from the analysis with the results that we obtained from the simulation. Then, we determine the average lifetime of a sensor node.

1.7 Thesis Structure

The thesis is arranged as following:

Chapter 1 *Introduction* illustrates the principles of CRWSNs, the objectives of the thesis and the organization of the thesis.

Chapter 2 *CogLEACH-C* introduces CogLEACH-C algorithm and how it prolongs the lifetime of the network compared to CogLEACH. Results are shown at the end of the chapter.

Chapter 3 *Node Connectivity Analysis* presents an analysis of node node connectivity when three models are used, arbitrary model, spectrum similarity model, and spatial-spectrum similarity model.

Chapter 4 *The Impact of Node Connectivity on Network Performance* shows the affect of node connectivity on massage delay in the system and average lifetime of the node.

Chapter 4 *Conclusion* explains the results introduced in the thesis.

Chapter 2

CogLEACH-C Algorithm

2.1 Introduction

A sensor node runs on a battery which is a limited energy resource. Once the battery is expired, normally, it can not be recharged, which means the death of the sensor node. Therefor, the lifetime of the sensor node is considered a major design factor in cognitive wireless sensor networks. In order to prolong the lifetime of the network, energy has to be consumed wisely.

In probabilistic clustering algorithms, a node's rule during a cycle is decided by the node itself or by the sink (centralized algorithm) depending on a probability determined by certain factors. Different algorithms uses different factors to compute that probability. For example, LEACH algorithm uses the number of rounds, while CogLEACH algorithm uses the number of channels sensed idle by each node. In both of those algorithms, the residual energy of a sensor node is not taking into account in determining the probability of a node to be a CH, which means that a node with a low residual energy level can be selected as a CH and that leads to a quick death for the node.

The aim of this chapter is to introduce CogLEACH-C algorithm, which is a centralized probabilistic algorithm uses the remaining energy of a sensor node as a second factor along with the number of channels sensed idle by the node to determine it is rule in a cycle.

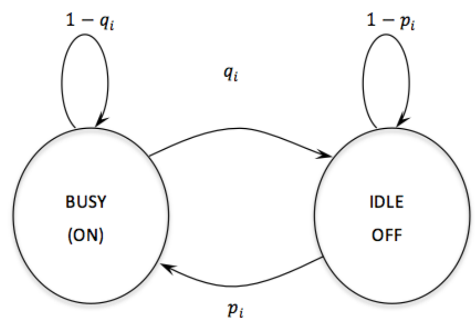

Figure 2.1: Semi-Markov Process for Primary Users

2.2 CogLEACH-C

In CogLEACH-C, nodes do not make the decision to be cluster heads or not, the BS is responsible for making the decisions on behalf of the nodes. This is based on a probability computed for each node using the node's vacant channels and the node's residual energy level. CogLEACH-C has to be centralized algorithm to be able to compute the average residual energy in the network. For a better energy consumption CogLEACH-C gives the right to the BS not only to determine a node's rule in a cycle, but also to form the clusters based on the node's idle channels and it is distance to the CH.

2.3 Primary Users Model

We used the same semi-Markov process to model the behaviour of the primary users PUs on a channel as in [10], in a way that a PU is either in an ON state or OFF state (as shown in Fig 1). The stationary probability of a channel i to be idle is given by:

$$p_f^i = \frac{q_i}{p_i+q_i} \tag{2.1}$$

where p_i and q_i are parameters representing the time intervals when a PU alternates between ON and OFF states respectively.

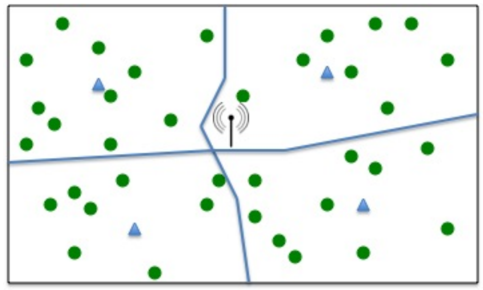

Figure 2.2: Network in Spatial and Spectrum Similarity Model

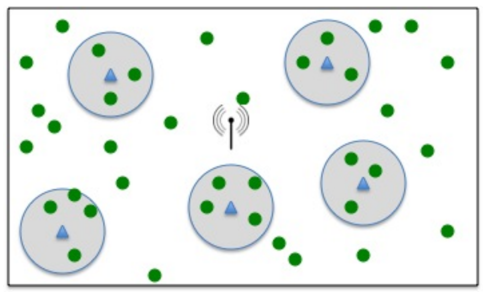

Figure 2.3: Network in the Spectrum Similarity Model and the Arbitrary Model.

2.4 Network Model

The network has *N* nodes which act like secondary users and *P* nodes which act like primary users (nodes that appear as triangles in Figure (2.2) and Figure (2.3) distributed randomly over a square area. The frequency band is assigned first to the primary users and then shared with the secondary users. To avoid interference between the secondary users and the primary users, the same interference protection range used in [10] is used here. As a result, none of the secondary users located within the coverage range of the primary users can sense a channel idle, if that channel is used by the PU.

Moreover, we assume that the BS is located in the middle of the network. For fair comparison between CogLEACH and CogLEACH-C, we use the same models used in [10]. Nevertheless, the algorithms are different. We started by the arbitrary model where even CogLEACH has to be centralized.

2.4.1 Arbitrary Model

In this scenario some of the nodes are within the coverage of the PUs. As well, channels have different p_f in a PU system and all the PUs have different p_f (p_f of channel1 in PU1 is different to the p_f of channel1 in PU2). As channels in a PU system have different values of p_f, it is not possible for the nodes to determine the average of channels sensed idle in the network. As a result, all the nodes send their vacant channels, their residual energy and their locations to the base station, which determines the probability for each node to be cluster head.

In CogLEACH-C a node can be a cluster head based on a certain probability $P_i(t)$ determined in each round. If use the number of channels sensed idle by the node to obtain $P_i(t)$, then $P_i(t)$ will be as following:

$$P_i(t) = min(\frac{C_i}{C_T}, 1) \tag{2.2}$$

While if we use the residual energy to obtain $P_i(t)$, then $P_i(t)$ will be as following:

$$P_i(t) = min(\frac{E_i}{E_T}, 1) \tag{2.3}$$

$$\text{where: } C_T = \sum_{i=1}^{N} C_i$$

$$E_T = \sum_{i=1}^{N} E_i$$

In both cases mentioned above, the summation of $P_i(t)$ for the N nodes is equal to 1.

$$\sum_{i=1}^{N} P_i(t) = 1$$

by multiplying (4) and (5) we get:

$$P_i(t) = min(\frac{C_i * E_i}{C_T * E_T}, 1) \tag{2.4}$$

$$\text{where: } \sum_{i=1}^{N} P_i(t) = \frac{1}{N}$$

Our objective is to make $E[\text{CHs}] = k$

$$E[\text{CHs}] = \sum_{i=1}^{N} P_i(t)$$

So, we multiply Eq (2.4) by $N * K$ to make the summation of $P_i(t)$ for the N nodes equal to K, where K represents the number of cluster heads in the network in each round. Eq (2.4) becomes:

$$P_i(t) = min(N * K * \frac{C_i * E_i}{C_T * E_T}, 1) \tag{2.5}$$

E_i is the energy level of node i. While C_i represents the number of channels sensed idle by node i.

2.4.2 Spatial and Spectrum Similarity Model:

All the nodes are within the coverage of the PUs as shown in Figure (2.2). As well, all the PUs have the same p_f for idle channels. All the nodes send their energy level, their location and their number of channels sensed idle to the BS. then, the BS determines the probability rule of the each node in the current round using Eq.(2.5).

2.4.3 Spectrum Similarity Model:

Some of the nodes are within the coverage of the PUs as shown in Figure 2.3. As well, channels have the same p_f in a PU system and different PUs have different values of p_f (p_f of channel1 in PU1 is different to the p_f of channel1 in PU2). Since, CogLEACH-C is a centralized algorithm, all nodes send their residual energy, their location and their vacant channels to the BS, which computes the probability of each node to be a cluster head using Eq.(2.5).

2.5 Base Station Cluster Head Selection

All the nodes start by sensing the spectrum looking for idle channels. Then, each node sends a message to the base station over a common control channel containing its ID, its vacant channels,its residual energy level and its position in the network, which can be done using a GPS or other wireless positioning devices. Once the base station receives all the information from all the nodes, it calculates the total residual energy in the network and the total number of channels sensed idle in the network. Next, the BS determines the probability for each node to become a cluster head or not using Eq.(2.5).

In [11], it is shown that the optimal number of clusters in the network in terms of energy dissipation is $K = 5$. Therefor, as the number of clusters increases, the total energy dissipated in the network per each round increases. CogLEACH algorithm does not guarantee that the number of clusters in each round is K, and it was noticed that the number of clusters exceeds K in many rounds, which means more dissipation of network energy. As a result, in case the number of nodes elected to be cluster heads in a round is above K, CogLEACH-C allows the BS to use K-nearest neighbours algorithm to choose the closest K nodes among those elected nodes to be the cluster heads of the current round. These K cluster heads are selected depending on their position for the nodes in the network to achieve lower energy consumption.

K-nearest neighbours is a learning algorithm that is used to classify data based on certain measures. It uses different distance measures, such as the Euclidean and Mahalanobis distance. It this work we use k-nearest neighbours algorithm to choose k cluster heads using the Euclidean distance as a measure for classification.

After choosing the K cluster heads, the BS broadcasts clustering information to the nodes and when a node receives those information, it validates its rule in the current round. if the node is a cluster head it waits for joining messages from the normal nodes. Then, it creates TDMA frame. While if the node is a normal node it sends join message to the cluster head with the minimum communication cost. These control messages exchanged between the BS and the nodes are short and they are sent over a common communication channel.

In addition, each round a node sends two control messages. A message to the BS, which contains its ID, its number of channels sensed idle and its residual energy. Then, it sends another message to the cluster head that was assigned for it by the BS to join the cluster.

2.6 Experiment and Results

2.6.1 Experiment Setup:

As it is not easy to model the network analytically, we used Matlab to simulate and evaluate CogLEACH-C and CogLEACH algorithms in terms of network lifetime. We made CogLEACH algorithm operates in a centralized fashion in all of the three models for a better comparison between the two algorithms. In our simulation, we have a network that consists of 100 nodes (secondary users) and 10 nodes (primary users) spread randomly over an area of $100 * 100m^2$. As well, the network consists of 5 channels, interference protection region with radius of $20m$, packet size of 4000 bits, K number of clusters $K = 10$, node initial energy of $2J$. The BS is located in the middle of the area. We used the same energy model proposed in [11] to implement this experiment.

2.6.2 Experiment Results:

Network Lifetime: All the nodes have data to send in every round. correspondingly, the residual energy level is low for most of the nodes after first node death, and the rate of dead nodes increases dramatically. Therefore, we consider system lifetime as the time elapses till the death of the first node. It can be noticed in Figure (2.4), Figure (2.5), and Figure (2.6) that system lifetime prolongs more when CogLEACH-C is used at different settings of p_f, and that is due to cluster heads selection mechanism used by CogLEACH-C, which depends on the nodes' energy level and their number of channels sensed idle. Thus, nodes with higher residual energy have better chance

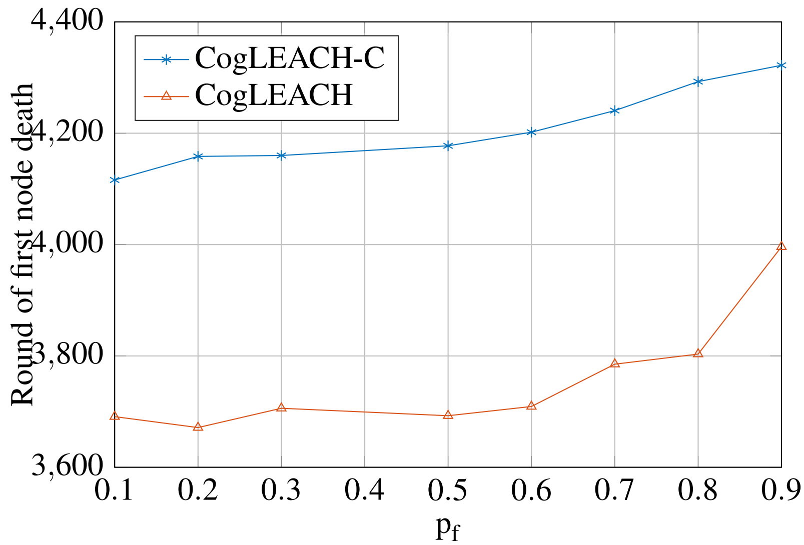

Figure 2.4: Round of First Node Death at Different p_f Settings for the Network for the spectrum similarity model

to be cluster heads. As well, the BS in CogLEACH-C ensures that the number of clusters in each round does not exceed K clusters, which provides an improvement in energy dissipation.

In spatial and spectrum similarity model, all the secondary nodes are within the coverage of the PUs. As a result, when p_f has a low value (lower than 0.3 as shown in figure (2.6)) the number of channels sensed idle by the nodes, those were elected to be cluster heads in a given round, is small compared to the number of channels used in the system. Subsequently, it becomes difficult for secondary users to join a cluster, which means that they do not consume energy, as they do not send any data in the current round. In contrast to the other two models, It can be seen that the life time of the network increases dramatically as the value of p_f decreases. We explain that in the next chapter when we do an analysis for the node's probability of connectivity

Data Received at the BS: When the number of nodes elected to be cluster heads in a given round is higher than the optimal value of K, the BS is allowed to choose the closest K nodes. Correspondingly, the amount of energy dissipated in the network decreases, which prolongs the lifetime of the network and allows more data to be sent to the BS (as shown in Figure (2.7), Figure (2.8) and Figure (2.9)). As can be seen, the best results were obtained when the arbitrary model and the spectrum similarity model were used. In this part of the simulation, p_f was set to 0.3.

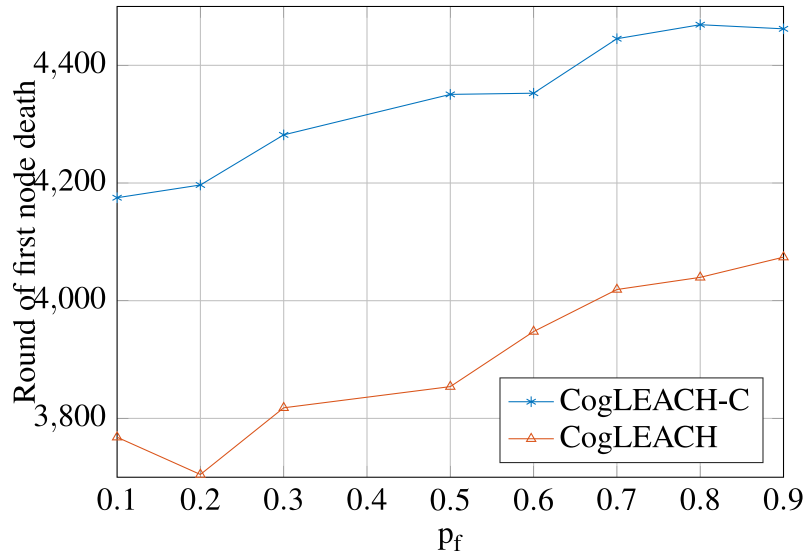

Figure 2.5: Round of First Node Death at Different p_f Settings for the Network for the arbitrary model

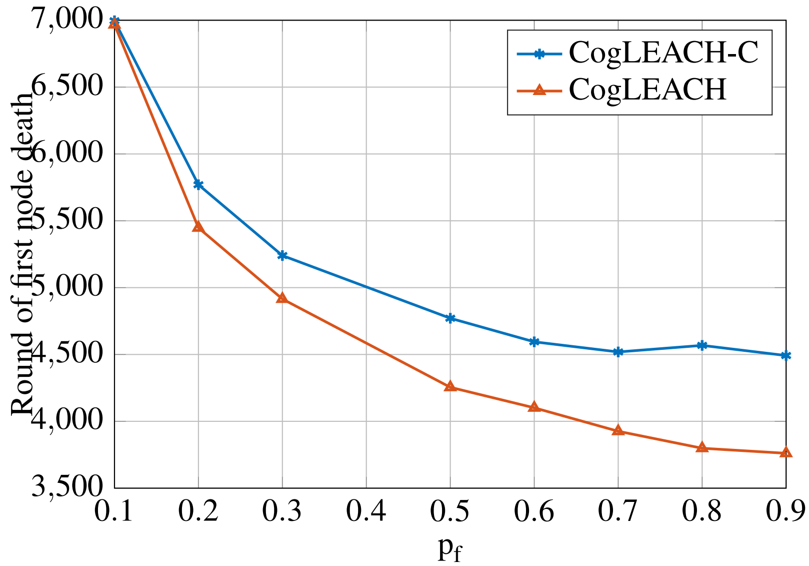

Figure 2.6: Round of First Node Death at Different p_f Settings for the Network for the Spatial and Spectrum similarity model

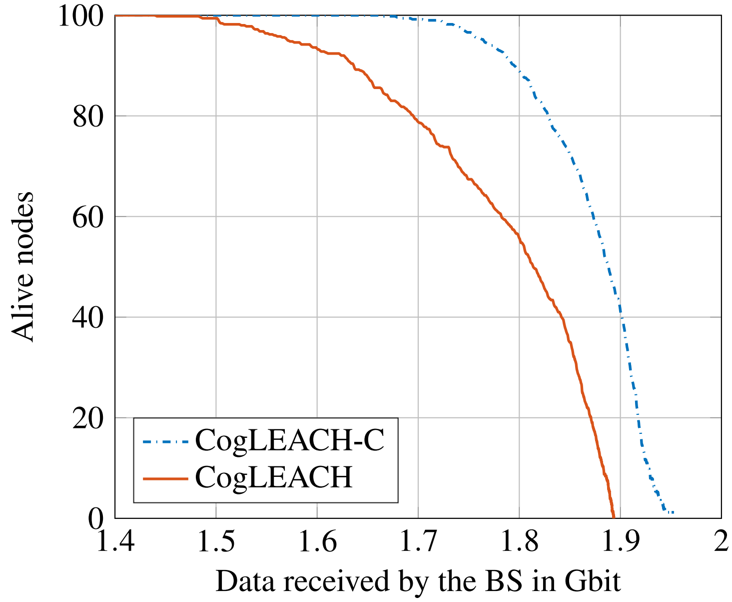

Figure 2.7: Number of Alive Nodes vs the Volume of Data Received by the BS for the Arbitrary Model

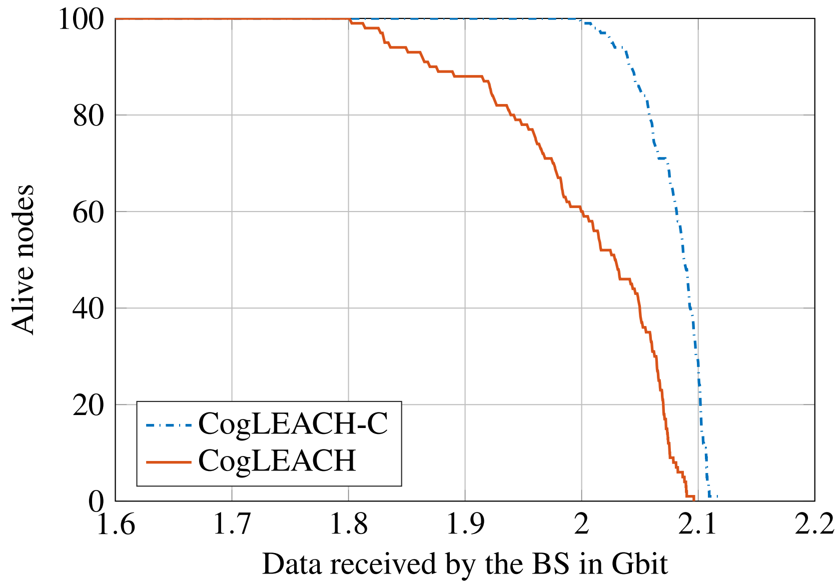

Figure 2.8: Number of Alive Nodes vs the Volume of Data Received by the BS for the Spatial and Spectrum Similarity Model

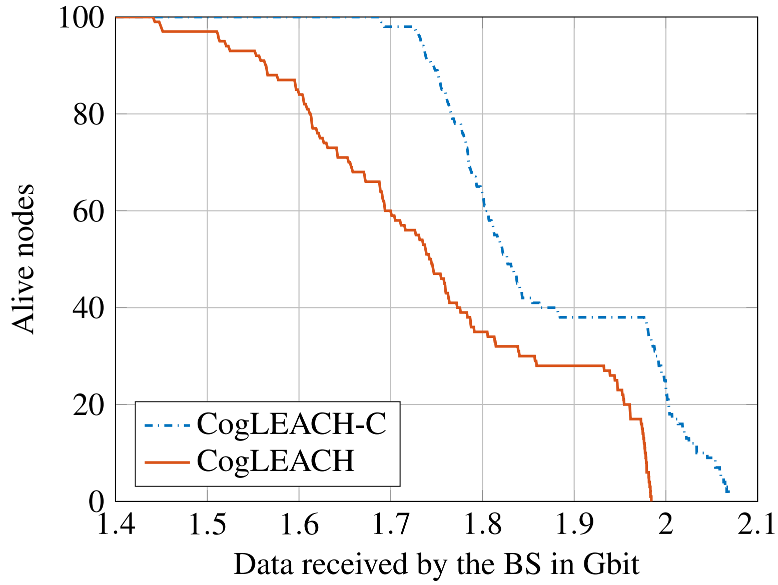

Figure 2.9: Number of Alive Nodes vs the Volume of Data Received by the BS for the Spectrum Similarity Model

Chapter 3

Node Connectivity Analysis

3.1 Introduction

In this section, we determine the probability of connectivity for a sensor node to the sink. To achieve that, we obtain the probability that a node is able to connect with a CH , assuming that CHs can always connect with the BS. We perform the analysis with the three models mentioned before in chapter 1. The result of this analysis will be used the next chapter to determine the impact of connectivity probability on the performance of the network.

To do so, we initially divide the area covered by the network into a grid of squares as shown in the figure below, where all nodes are spread over the area uniformly.

We assume that nodes are always ready to transmit unless they are in sleeping mode, which only occurs when they are not able to connect with a cluster head (CH). Node connectivity depends on its connectivity with one of the CHs, which depends on the channels sensed idle by the two nodes. If there is an intersection between them and the CH is within the coverage of the node then the node can connect to the CH. Otherwise, the node attempts to connect with a neighbor CH.

Network connectivity is the ratio between the average number of connected nodes and the number of the nodes in the network. The number of connected nodes in the network has binomial distribution with N and p_c, where N is the number of nodes in the network and connectivity probability. As a result, we can say that by finding the probability of connectivity for a node, we

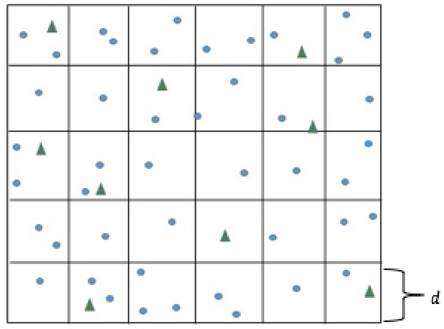

Figure 3.1: A Circle Represents a Secondary Node and a Triangular Represents a Primary Node in the Grid Network

determine the probability of connectivity of the network.

To determine the connectivity probability of a sensor node, we assume that the coverage area of a node is equal to a circle with diameter $\sqrt{2}d$, where d is the width of a cell in the grid. By doing that, we ensure that a CH in the same square with the node is within the coverage of that node.

3.2 Connectivity Probability between a Node and a Cluster Head within the Same Square

In the use that the node and the cluster head are in the same square, it is certain that they are within each others coverage. Therefore, the connectivity probability here depends on the channels sensed idle by the two nodes. if there is an intersection then the node can connect with the CH, Otherwise it will have to connect with a neighboring CH.

We define a random variable X, which is the number of intersections between the two nodes

$$P_r[\text{ there is at least one intersection}] = P[X \geq 1] = 1 - P[X = 0] \tag{3.1}$$

We defined three models for CogLEACH-C algorithm and each one of them has different assumptions. Hence, we will determine the probability of connectivity of a node in the network in the three cases we have.

3.2.1 Spectrum Similarity Model:

We assume that we have m channels in our network. In this model some of the nodes are within the coverage of primary users (PUs) and some of them are outside of it. In addition, nodes that are within the coverage of the PUs sense a channel idle with probability p_{f1}, while the nodes that are out of the coverage of the PUs sense a channel idle with probability p_{f2}, which is always equal to 1. Moreover, p_{f1} is the same for all PU systems.

Equation (3.1) shows that to find the probability of having at least one intersection between the two nodes, we need to find the probability that there is no intersection between the two nodes. That means, there is no common channel sensed idle by the two nodes. Therefore, we first determine the probability that a channel is sensed idle by a node as shown below.

$$\begin{aligned} P_r[\text{ a channel is sensed idle by a node}] = P[(\text{channel idle})|(\text{node is in PU coverage})] \\ *P[\text{node is in PU coverage}] \\ +P[(\text{channel idle})|(\text{node is not in PU coverage})] \\ *P[\text{node is not in PU coverage}] \end{aligned} \tag{3.2}$$

Instead of finding the probability that a node is within the coverage of the PUs, we will determine the probability that a PU is within the coverage of the node. As the primary users are spread over the area covered by the network uniformly, the probability that a PU exists within the coverage of a secondary node can be found by dividing the area covered by the secondary user A_{sec} on the total area covered by the network A_{total}.

$$P[\text{a PU is in the coverage of the node}] = p_{pu} = \frac{A_{sec}}{A_{total}} = \frac{\Pi r_{sec}}{M^2}$$

$$p_{pu} = \frac{\Pi(\sqrt{2}d)^2}{M^2} = \frac{2\Pi d^2}{M^2}$$

We have N_{pu} PU nodes in the network so there could be more than one PU within the coverage of the node. As result, using binomial distribution we find the probability that there is at least one PU within the coverage of the node. In decentralized uncoordinated sensing techniques, sensors do not cooperate with each other in sensing the spectrum, which means that the sensing is performed independently. If a node detects primary user arrival it vacates the channel without acknowledging the other nodes [23]. In this analysis we assume that sensors detect the channels independently.

$$P[\text{there is at least one PU in the coverage of the node}] = p_{pui} = 1 - P[\text{there is no PUs in}]$$

$$= 1 - (1 - p_{pu})^{N_{pu}} = 1 - (1 - \frac{2\Pi d^2}{M^2})^{N_{pu}} \tag{3.3}$$

So Eq(3.3) becomes as following:

$$P_r[\text{a channel is sensed idle by a node}] = p_{idle} = p_{f1} * p_{pui} + p_{f2} * (1 - p_{pui})$$

$$= p_{f1} * p_{pui} + (1 - p_{pui})$$

$$= 1 - p_{pui} * (1 - p_{f1}) \tag{3.4}$$

As sensing the channel by a node is independent from the sensing process of other nodes, the probability of a channel to be sensed idle by the two nodes is:

$$P[\text{a channel is sensed idle by the two nodes}] = p_{idle} * p_{idle} = {p_{idle}}^2$$

p_f	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
p_{cin}	0.86	0.91	0.94	0.97	0.98	0.99	1	1	1

Table 3.1: Node Connectivity Probability to Cluster Head in the Same Cell in Spectrum Similarity Model

So, the probability of having no common channels sensed idle by the node and the CH is as the following:

$$P[X=0] = (1 - {p_{idle1}}^2) * (1 - {p_{idle2}}^2) ... (1 - {p_{idlem}}^2)$$

P_{idle1} is the probability that channel 1 is idle and P_{idlem} is the probability that channel m is idle. As P_{idle} is the same for all channels then $P[X=0]$ becomes:

$$P[X=0] = (1 - {p_{idle}}^2)^m$$

By substituting $p[X=0]$ in (3.1) we get:

$$P_{cin} = P[X \geq 1] = 1 - (1 - {p_{idle}}^2)^m \tag{3.5}$$

Table (3.1) shows the connectivity probability P_{cin} at different values of p_f when the spectrum similarity model is used with m = 5 channels and $N_{PU} = 10$.

3.2.2 Spatial and Spectrum Similarity Model:

In this model, all the nodes are within the coverage of the PUs and all the PUs have the same p_f.

$$P_{idle} = p_f$$

$$P_{cin} = P[X \geq 1] = 1 - (1 - {p_{idle}}^2)^m$$

p_f	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
p_{cin}	0.049	0.185	0.376	0.582	0.763	0.893	0.965	0.99	1

Table 3.2: Node Connectivity Probability to Cluster Head in the Same Cell in Spatial and Spectrum Similarity Model

Table (3.2) shows the connectivity probability at different values of p_f when the spatial and spectrum similarity model is used with $m = 5$ channels.

By comparing the two tables we see that there is a slight difference between the values of p_{cin} at different values of p_f when spectrum similarity model is used, while there is a gradual increase in the value of p_{cin} as the value of p_f increases when spatial and spectrum similarity model is used. That explains why the lifetime of the network, when that model is used, is really long when p_f has low values compared to the lifetime when p_f has high values Fig (2.6). When p_f is low, the connectivity probability is low, which means that nodes have difficulty to connect with the CH. As a result, they stay in the sleeping mode waiting for the next cycle and that is why it appears in the figure that they longer lifetime at low values of p_f.

3.2.3 Arbitrary Model:

In this model, some of the nodes are within the coverage of the PUs. As well, different PU systems have different values for p_f. In this case, channels have different values of the probability to be idle p_{idle}, as there could be more than one PU in the coverage of the node sensing the spectrum. Therefore, a channel can be sensed idle, if there is no PUs in the coverage of the node or it is not utilized by any PU system within the coverage of the node.

$$p_{idle} = p_{f2} * (1 - p_{pui}) + p_{pui} \prod_{j=1}^{k_{pu}} p_{fj}$$

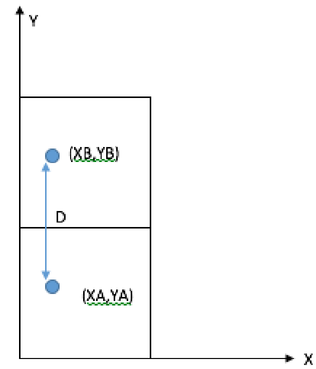

Figure 3.2: Distance between Two Neighbours

Where k_{pu} is the number of PUs in the coverage of the node, and $p_{f2} = 1$

$$p_{idle} = (1 - p_{pui}) + p_{pui} \prod_{j=1}^{k_{pu}} p_{fj} \tag{3.6}$$

Eq. (3.5) becomes as the following:

$$P[X = 0] = \prod_{j=1}^{m} (1 - p_{idle_j}{}^2)$$

3.3 Connectivity Probability between a Node and a Neighbour Cluster Head:

As we can see from the Fig (3.2), in order for the node and the cluster head to connect with each other, the distance between them must be less than or equal to $\sqrt{2}d$. Moreover, there must be at least one intersection between them in the channel sensed idle by the two of them.

As we see from Fig (3.2) each node is represented by x and y coordinates, which are random variable. So, in order to find the probability that the distance between the CH and the node is less than or equal to $\sqrt{2}d$, we first find the probability density function of the four variables x_a, x_b, y_a

and y_b. Nodes are uniformly distributed over each square, which has a width of d, the probability density function of the coordinates is shown bellow.

$$f_{x_a}(x_a) = f_{x_b}(x_b) = f_{y_a}(y_a) = f_{y_b}(y_b) = \begin{cases} \frac{1}{d-0} = \frac{1}{d} & \text{for} \quad 0 \leq x_a, x_b, y_a \leq d, d \leq y_b \leq 2d \\ 0 & \text{for} \quad Otherwise \end{cases} \tag{3.7}$$

Since x_a,x_b,y_a and y_b are independent random variables, the joint probability density function is as shown below:

$$f_{x_a,x_b,y_a,y_b}(x_a, x_b, y_a, y_b) = \frac{1}{d^4}$$

we need the distance between the two nodes D be equal or less than $\sqrt{2}d$ to ensure that the CH is within the coverage of the node.

$$D \leq \sqrt{2}d$$

$$D = \sqrt{(x_b - x_a)^2 + (y_b - y_a)^2}$$

$$(x_b - x_a)^2 + (y_b - y_a)^2 \leq 2d^2$$

$$(y_b - y_a)^2 \leq 2d^2 - (x_b - x_a)^2$$

$$-\sqrt{2d^2 - (x_b - x_a)^2} \leq (y_b - y_a) \leq \sqrt{2d^2 - (x_b - x_a)^2}$$

However, y_b is always bigger than y_a. Hence, the negative side can be ignored.

$$(y_b - y_a) \leq \sqrt{2d^2 - (x_b - x_a)^2}$$

The high limit of y_b is alwayes higher than or equal to d. We use that to find the limits of y_a, which is always lower than or equal to d.

$$y_a - \sqrt{2d^2 - (x_b - x_a)^2} \leq y_b \leq y_a + \sqrt{2d^2 - (x_b - x_a)^2}$$

y_b is always higher than or equal to d. So, the limits of y_b become:

$$d \leq y_b \leq y_a + \sqrt{2d^2 - (x_b - x_a)^2} \tag{3.8}$$
$$y_a + \sqrt{2d^2 - (x_b - x_a)^2} \geq d$$

$$d \geq y_a \geq d - \sqrt{2d^2 - (x_b - x_a)^2} \tag{3.9}$$

We use Eq.(3.7),Eq.(3.8) and Eq.(3.9) to find $p[D \leq \sqrt{2}d]$

$$p[D \leq \sqrt{2}d] = \int_0^d \int_0^d \int_{d-\sqrt{2d^2-(x_b-x_a)^2}}^d \int_d^{y_a+\sqrt{2d^2-(x_b-x_a)^2}} \frac{1}{d^4} dy_b dy_a dx_b dx_a$$
$$= \frac{1}{d^4} \int_0^d \int_0^d \int_d^{d-\sqrt{2d^2-(x_b-x_a)^2}} (y_a + \sqrt{2d^2 - (x_b - x_a)^2} - d) dy_a dx_b dx_a$$
$$= \frac{1}{d^4} \int_0^d \int_0^d (\frac{y_a^2}{2} + y_a \sqrt{2d^2 - (x_b - x_a)^2} - dy_a) \mid_{d-\sqrt{2d^2-(x_b-x_a)^2}}^d dx_b dx_a$$
$$= \frac{1}{d^4} \int_0^d \int_0^d (\frac{d^2}{2} + d\sqrt{2d^2 - (x_b - x_a)^2} - d^2)(\frac{d^2 - 2d\sqrt{2d^2 - (x_b - x_a)^2} + 2d^2 - (x_b - x_a)^2}{2}$$
$$+d\sqrt{2d^2 - (x_b - x_a)^2} - (2d^2 - (x_b - x_a)^2)$$
$$-d^2 + d\sqrt{2d^2 - (x_b - x_a)^2}) dx_b dx_a$$
$$= \frac{1}{d^4} \int_0^d \int_0^d d^2 - \frac{x_b^2}{2} + x_b x_a - \frac{x_a^2}{2} dx_b dx_a$$
$$= \frac{1}{d^4} \int_0^d d^2 x_b - \frac{x_b^3}{6} + \frac{x_b^2 x_a}{2} - \frac{x_b x_a^2}{2} \mid_0^d dx_a$$
$$= \frac{1}{d^4} \int_0^d d^3 - \frac{d^3}{6} + \frac{d^2 x_a}{2} - \frac{d x_a^2}{2} dx_a$$
$$= \frac{1}{d^4} (d^3 x_a - \frac{d^3 x_a}{6} + \frac{d^2 x_a^2}{2} - \frac{d x_a^3}{2}) \mid_0^d$$
$$p[D \leq \sqrt{2}d] = \frac{11}{12}$$

For a node to be able to connect with a neighbouring cluster head, the cluster head has to be within the coverage of the node and there has to be intersection between the two nodes in the channels sensed idle by both of them. As, the two events are independent from each other, we can find the probability of connectivity by multiplying the probabilities of the two events.

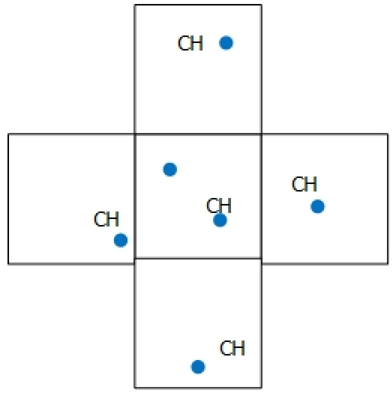

Figure 3.3

$$P_r[\text{a node is connected with a neighbor CH}] = p_{cn} = p[D \leq \sqrt{2}d] * p[x \geq 1] = \frac{11}{12} p_{cin}$$

$$p_{cn} = \frac{11}{12}(1 - (1 - p_{idle}^2)^m) \tag{3.10}$$

3.4 Connectivity Probability of a Node when There Is a CH in the Same Cell and Neighbour CH:

When CogLEACH-C is used, the BS assures that the number of cluster heads in each round does not exceed K cluster heads. In order for a node to be connected to the BS, it has to secure a connection with a CH in the same cell or a neighboring cell. In every round we have K CHs in the network so to find the probability that we have one of those clusters in the same cell of the node, we use binomial distribution.

Nodes are spread uniformly over the area. Therefore, the selected CHs follow a uniform distribution when it comes to their location in the network. Hence, the probability of successfully having a CH located in a d^2 area, which is the area of one cell, is $p_{ch} = \frac{d^2}{M^2}$. Moreover, the probability of successfully having a CH located in a $c_n * d^2$ area, where c_n is the number of neighbouring

p_f	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
p_c Spectrum	0.86	0.91	0.94	0.97	0.98	0.99	1	1	1
p_cSpatial and Spectrum	0.049	0.185	0.376	0.582	0.763	0.893	0.965	0.99	1

Table 3.3: Probability of Connectivity at different values of p_f

cells, is $p_{chn} = \frac{c_n * d^2}{M^2}$

$$P[\text{J CHs in the targeted area}] = \binom{K}{j} p_{ch}^{l} (1 - p_{ch})^{K-j} \tag{3.11}$$

As mentioned above, in order for a node to connect with the sink, it has to be connected with a CH in the same cell or a neighbouring CH. Using (3.5), (3.7) and (3.8) we can find the probability of connectivity of a node as following:

$$\begin{aligned} P[\text{a node is connected to the sink}] = p_c = p[&\text{a node is connected with CH in cell}|\text{there is a CH in cell}] \\ &*p[\text{there is a CH in cell}] \\ &+p[\text{a node is connected with a neighbor CH}|\text{there is at least one neighbor CH}] \\ &*p[\text{there is at least one neighbor CH}] \end{aligned}$$

$$\begin{aligned} p_c &= p_{cin} * \binom{K}{1} p_{ch}(1 - p_{ch})^{K-1} + p_{cn}(1 - \binom{K}{0} p_{chn}(1 - p_{chn})^K) \\ &= p_{cin} * K * \frac{d^2}{M^2}(1 - \frac{d^2}{M^2})^{K-1} + p_{cn}(1 - (\frac{C_n * d^2}{M^2})^K) \\ &= p_{cin}(K * \frac{d^2}{M^2}(1 - \frac{d^2}{M^2})^{K-1} + p[D \leq \sqrt{2}d](1 - (\frac{C_n * d^2}{M^2})^K)) \end{aligned} \tag{3.12}$$

Where c_n is the number of neighboring cells, which depends on the position of the node on the grid.

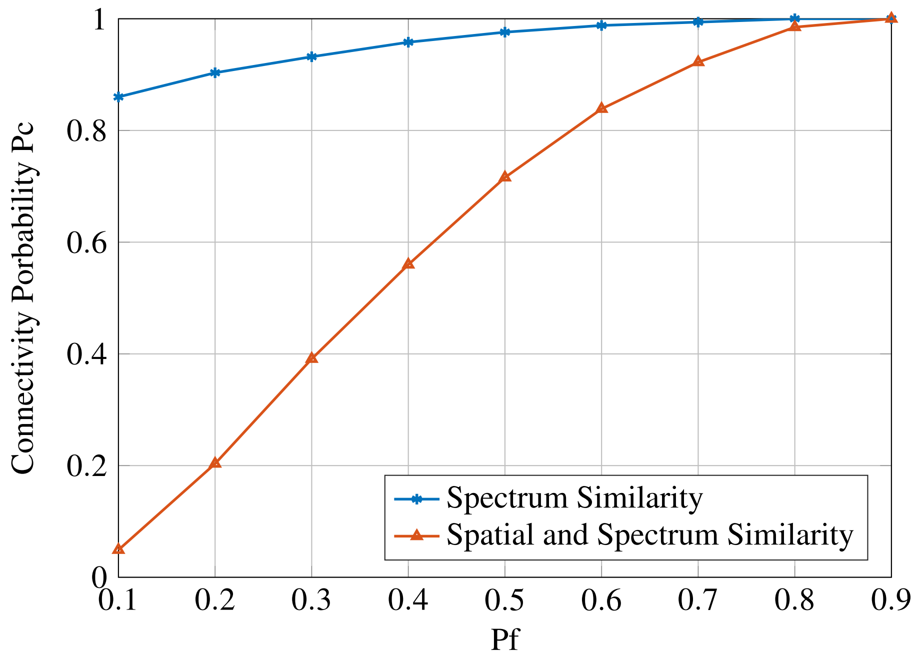

Figure 3.4: Connectivity Probability for Spatial and Spectrum Similarity Model and Spectrum Similarity Model

Table shows the probability of connectivity p_c for a node when the spectrum similarity and the spatial and spectrum similarity models are used. The following values were used: $M = 100m$,$K = 10$,$d = 10m$,c_n=5. Here we consider cells to be neighbors when they share the same border. So, the maximum number of neighbors is 4.

Chapter 4

The Impact of Node Connectivity on Network Performance

4.1 Introduction

In this chapter, we model a sensor node as M/G/1 queue to find the probability density function of message delay in the system when CogLEACH-C is used as the clustering algorithm. We first model the node as M/G/1 queue then we use exponential distribution for service time to model the node as M/M/1 queue. We assume that the node has Poisson arrivals. Moreover, the Node is always capable of providing service unless it is not able to connect with a cluster head. As a result, it changes to sleeping mode and it goes back to the busy mode when it has connectivity to the sink with probability p_c.

4.2 Modeling Sensor Node as M/G/1 Queue

In order to find the probability density function of message delay, we first need to find the distribution of number of messages in the system.

4.2.1 The distribution of Number of Messages in The System

We determine embedded Markov chain for M/G/1 queue. To do so, We define an indicator $I(i)$, which is equal to 1 if there is a connection and 0 if there is no connection.

$$n_{i+1} = n_i - I(i) * U(n_i) + a_{i+1}$$

Where a_{i+1} is the number of arrivals during the service time of message $i+1$. n_i is the status of the queue when message i departs the system . $U(n_i)$ has a value of 0, if the queue is empty after message i departs or a value of 1, if it is not empty.

Then, we find the probability generation function of the number of customers in the system as shown below:

$$E[z^{n_{i+1}}] = E[z^{n_i - I(i)*U(n_i)+a_{i+1}}]$$

We consider the two cases when the node is connected to the BS with probability $1-p_c$, and when it is not connected with probability p_c. Hence we get:

$$E[z^{n_{i+1}}] = (1-p_c)E[z^{n_i+a_{i+1}}] + p_c E[z^{n_i - I(i)*U(n_i)+a_{i+1}}]$$

$$\lim_{i\to\infty} a_{i+1} = a$$

$$E[z^{n_{i+1}}] = [(1-p)E[z^{n_i}] + pE[z^{n_i-U(n_i)}]]E[z^a] \tag{4.1}$$

We first determine $E[z^{n_i-U(n_i)}]$:

$$E[z^{n_i-U(n_i)}] = \sum_{k=0}^{\infty} z^{k-U(k)} P_r[n_i = k] = p(0) + z^{-1}(p(z) - p(0)) \tag{4.2}$$

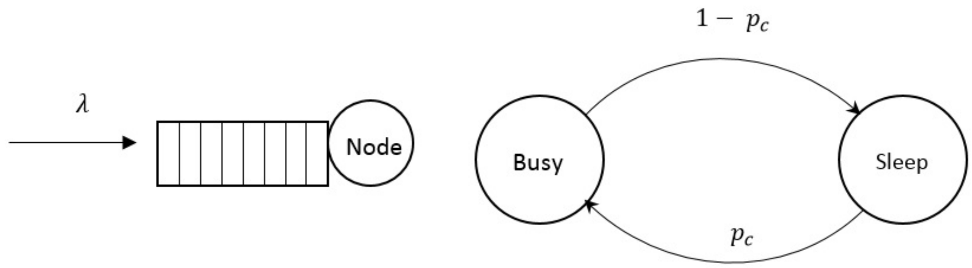

Figure 4.1: Sensor Node as M/M/1 System

By substituting from Eq.(4.2) in Eq.(4.1) we get:

$$p_{i+1}(z) = [(1-p_c)p_i(z) + p_c(p(0) + z^{-1}(p(z) - p(0)))] * E[z^a]$$

$$\lim_{i\to\infty} p_{i+1}(z) = \lim_{i\to\infty} p_i(z) = p(z)$$

$$p(z) = [(1-p_c)p(z) + p_c(p(0) + z^{-1}(p(z) - p(0)))] * E[z^a]$$

$$p(z) = \frac{p_c p(0)(z-1)A(z)}{z - zA(z) + zp_cA(z) - p_cA(z)} \tag{4.3}$$

4.2.2 Probability Density Function of Message Delay

In this section, we determine the distribution of message delay for M/G/1 queue, which will be used in the next section to find the probability density function of message delay for the node as M/M/1 queue. the Probability generating function of message arrivals can be represented as foloowing:

$$A(z) = \int_0^\infty e^{-\lambda t(1-z)} * m(t)dt = M(\lambda(1-z)) \tag{4.4}$$

$$z = 1 - \frac{s}{\lambda}$$

We are assuming that messages are being first come first served in the system. As, the number of messages in the system is equal to number of messages arrived while the departing customer

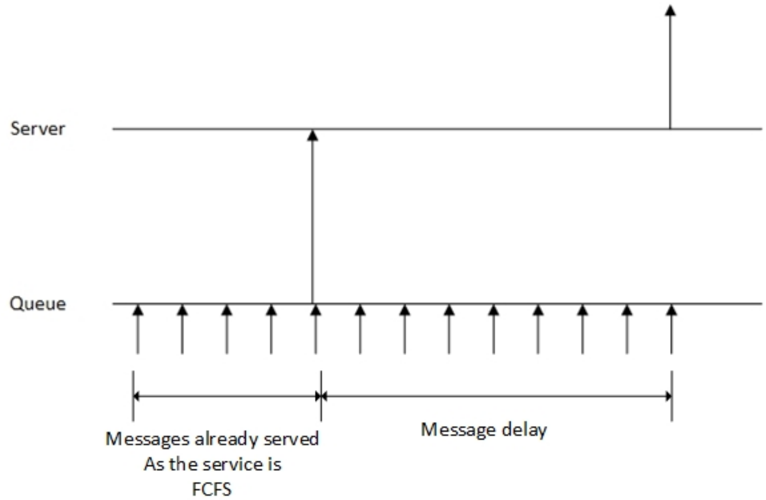

Figure 4.2: Queueing System

was served as shown in Fig (4.2), the probability generating function of number of customers in the system is equal to the delay under the condition showed below:

$$p(z) = D(s)\,\big|_{s=\lambda(1-z)} \tag{4.5}$$

Using Eq.(4.3),Eq.(4.4) and Eq.(4.5) we get Laplac transform of the probability density of message delay:

$$D(s) = \frac{p_c p(0)(z-1)M(s)}{z - zM(s) + zp_c M(s) - p_c M(s)} \tag{4.6}$$

4.3 Probability Density Function of Message Delay for M/M/1 Queue

In order to find the probability density function of message delay for node modeled as m/m/1 queue, we assume that service time for messages has exponential distribution. $p(0)$ and $M(s)$ are as the following:

$$p(0) = 1 - \rho, \rho = \frac{\lambda}{p_c \mu}, M(s) = \frac{\mu}{s + \mu}$$

By substituting $p(0) and M(s)$ in Eq.(4.6) we get:

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

Using Laplace inverse transform we can obtain the distribution of message delay in the system $d(t)$.

$$d(t) = (p_c \mu - \lambda) e^{(p_c \mu - \lambda) t} \tag{4.7}$$

Eq.(4.7) shows that message delay in the system has an exponential distribution that is affected by the probability of connectivity of the node. A lower probability of connectivity for the node leads to a higher delay for messages to be served. the average delay for a message in the queue is as shown below:

$$\overline{d} = \frac{1}{p_c \mu - \lambda} \tag{4.8}$$

4.3.1 Numerical Results

Table (4.1) illustrates the average delay per message when $\mu = 180, \lambda = 90$ for Spectrum similarity model and $\mu = 180, \lambda = 5$for spatial and spectrum similarity model.It can be noticed from the table that the average delay per packet decreases as the connectivity probability increases.

$\overline{d}$	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
$\overline{d}$ Spectrum	0.015	0.014	0.013	0.012	0.012	0.011	0.011	0.011	0.011
$\overline{d}$ Spatial and Spectrum	0.26	0.016	0.01	0.008	0.007	0.006	0.006	0.005	0.005

Table 4.1: Average Message Delay at Different Vlues of p_f

4.3.2 Simulation Results

In this section, we will simulate one of the nodes in our network as if it has a Poisson arrivals in order to compare the the numerical results we obtained for message delay in the system with the results we get from the simulation. In addition, in this scenario, we create a network one time when we start the simulation to ensure that the node has the same position in the network as we change the probability of a channel to be idle (p_f). Moreover, We use spectrum similarity for the comparison to show the affect of probability of connectivity on the network, which would be the same using the different models, but with different values.

Figure (4.3) shows the numerical results and the simulation results for message delay in the system. It can be seen that as the the probability of a channel to be idle increases, average message delay decreases correspondingly as a result of a higher probability of connectivity for the node.

4.4 Average Lifetime of a Sensor Node

Each node in the network alternates between idle and busy periods during each cycle with probabilities $1 - p_c$ and p_c respectively. A new cycle starts at the beginning of a busy period. The total lifetime of a sensor node is the sum of the average time of busy periods and idle periods. Idle periods have the same distribution of the arrival time, which has exponential distribution with rate

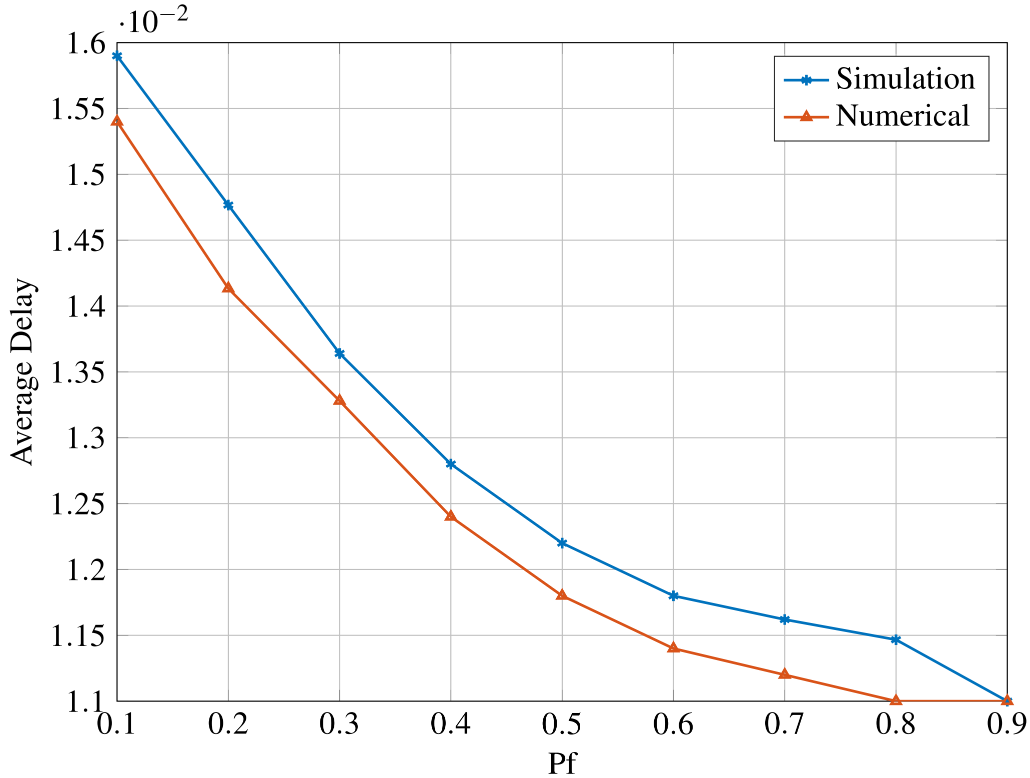

Figure 4.3: Average Message Delay as a Function of Probability of Connectivity

λ.

The average time of one idle period is $\frac{1}{\lambda}$ and idle periods are independent and identically distributed. Therefore, if we need n cycles until the node dies, we can say that the average time of n idle periods is as following:

$$n_{idle} = \frac{n}{\lambda} \tag{4.9}$$

The ratio between the average time of busy periods and the average time of one cycles is equal to the time when the node is busy providing service ρ.

$$\frac{\overline{BS}}{\overline{BS} + \frac{1}{\lambda}} = \rho = \frac{\lambda}{p_c \mu}$$

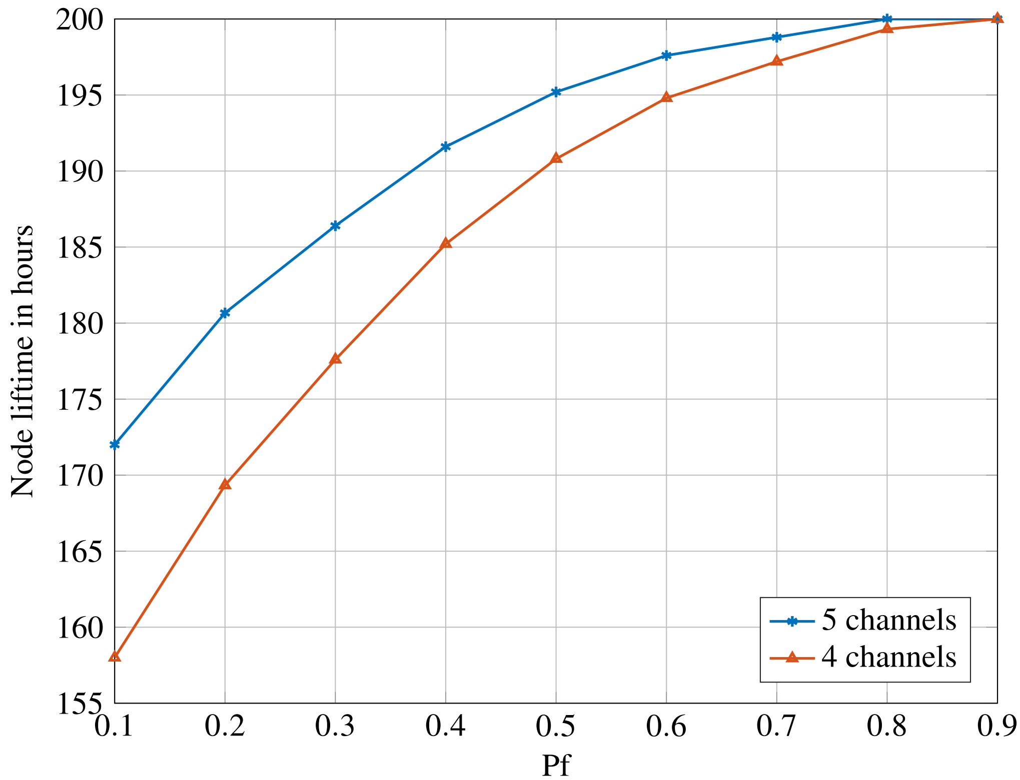

Figure 4.4: Node Lifetime as a Function of Probability of Connectivity

$$\overline{BS} = \frac{1}{p_c\mu - \lambda} \tag{4.10}$$

The average numbers of cycles until the nodes dies is n. So, the average time of the busy periods during n cycles n_busy is:

$$n_{busy} = \frac{n}{p_c\mu - \lambda} \tag{4.11}$$

We consider that the node consumes energy only during busy periods, which means that we consider the energy consumed during transmissions only and we neglect the energy consumed during processing and sensing. As a result, the average time of the busy periods is equal to battery life $B_{life} = n_{busy}$.We can determine the average life of a sensor node as shown below:

Node lifetime = N_{life}= average time of busy periods + average number of idle periods

From (4.9) and (4.11) we get:

$$N_{life} = \frac{n}{p_c\mu - \lambda} + \frac{n}{\lambda}$$

By using $n = B_{life} * (p_c\mu - \lambda)$ from Eq (4.11), we get:

$$N_{life} = \frac{p_c\mu B_{life}}{\lambda} \tag{4.12}$$

Figure (4.4) shows the lifetime of a sensor node at different values of the number of channels available in the network when $\mu = 180, \lambda = 90$ and $B_{life} = 100$ hours. It can be seen that a lower number of channels at a low connectivity probability leads to a shorter life for the sensor node. As, we assume that a node consumes energy only during the busy periods of the cycles, t does not matter how long the idle period of a cycle is. Hence, We can compare the numerical results we obtained in this section to the simulation results we showed in chapter 2 for the lifetime of the network (Figure (2.4), Figure (2.5) and Figure (2.6)), even that we assumed a node has a Poison message arrival in this section while we assumed in chapter 2 that nodes always have data to transmit. Both of the results we obtained, numerical and experimental, show that the lifetime of a sensor node increases as the the probability of connectivity for the nodes increases.

Chapter 5

Conclusion

In This thesis, we presented CogLEACH-C algorithm. A centralized clustering algorithm for CRWSNs, which is based on CogLEACH algorithm. We showed how CogLEACH-C lengthens the lifetime of the network compared to CogLEACH algorithm. In addition, We provided an analysis of the connectivity probability for sensor nodes in CRWSNs when CogLEACH-C is used for clustering.Moreover, we showed how that probability affects network performance in terms of message delay and the average lifetime of a sensor node.

Network lifetime is a key factor in CRWSNs and it depends mainly on the consumption of energy in the network. Therefor, we derived an a new equation that depends not only on the number of channels sensed idle by a sensor node, but also on the average residual energy in the network to determine the probability for each node to be a cluster head or not. By doing that, we succeeded in providing a better selection process for cluster heads, as nodes with higher residual energy had a higher chance to be cluster heads. Then, we worked on adding more features that can make energy consumption in the network more effective. CogLEACH-C is a centralized algorithm, and we used that as an advantage to give the BS the right to limit the number of nodes taking the rule of being cluster heads in each round. The BS ensures that the number of cluster heads in each round does not exceed the optimal number K, which is presented in literature as the best number of clusters for an effective energy consumption in CRWSNs. Moreover, we allowed the BS to form the clusters using K-Neighbour algorithm and inform each node with its rule during each round.By doing so, we minimized the number of messages needed to be exchanged between nodes and cluster heads for cluster formation. Sensor nodes have to send their information in the beginning of each round in a short message, which is sent over a common communication channel.

Then, We provided a comparison between CogLEACH-C and CogLEACH in terms of the number of nodes that stay alive as the network ages and the amount of data received by the sink node using three different models. Arbitrary model where different PU systems had different values of p_f and some of the secondary nodes were within the coverage of PU systems. Spatial and spectrum similarly model where all PU systems had the same p_f and all the secondary nodes were within the coverage of the PUs. Finally, spectrum similarity model where all PU systems had the same p_f and some of the secondary nodes were within the coverage of PU systems. We assumed that nodes always have data to transmit whenever they are connected. We showed that the number of nodes remained alive as the time elapsed was higher when CogLEACH-C was used than when CogLEACH was used in the three different models. As a result, the number of packets were received by the sink node was greater when CogLEACH-C was used.

Next, We introduced an analysis of cognitive radio sensor node connectivity probability. In CRWSNs, secondary nodes share the licensed band with the primary users, which gives CRWSNs the advantage of being able to operate everywhere. With that being said, secondary users have to respect the right of usage of the licensed band by primary users. As a result, secondary users can not use a channel that is used by a primary user, and in the case where a secondary user exists within the coverage of a very active primary user, it can be hard for the secondary user to connect with the sink node, which affects the overall performance of the network. To derive the probability of connectivity, We divided the area covered by the network into blocks, and we assumed that nodes were spread uniformly over the area. Moreover, we assumed that nodes coverage is equal to a circle that has a diameter of $\sqrt{2}d$, where d is the block width, to ensure that a node is always within the coverage of a cluster head in the same block. Then, we derived the connectivity probability of a node with a cluster head in the same block, which is equal to the potability that there is at least one intersection in the channels sensed idle by the two nodes. After, We determined the probability of connectivity of a node with a cluster head in a neighbouring block, which depends on the former probability and on the probability that the two nodes are within each other's coverage. At the end, we provided an analysis of the probability of connectivity of a node with the sink node.The analysis considered the three models mentioned earlier. Through this part, we showed how the probability of connectivity of a node increases as the probability of a channel to be idle increases.

Finally, We used the probability that was obtained in the previous part to show its impact on network performance in terms of message delay and average lifetime. To do so, we analyzed the sensor node as an M/M/1 queue and we found the probability density function of message delay. We assumed that message arrival has a Poisson distribution and nodes alternate between active and sleep mode depending on their connectivity with the sink.On other words, if the node is not connected to the sink it enters the sleeping mode. Then, we determined the average lifetime of a sensor node. We showed that message delay greatens as the probability of connectivity decreases and the average lifetime rises as the probability of connectivity rises.

Bibliography

[1] Abolarinwa J.A, SalawuN, Achonu A. "Cognitive Radio-based Wireless Sensor Networks As Next Generation Sensor Network: Concept, Problems and Prospects", Journal of Emerging Trends in Computing and Information Sciences, Vol. 4, No. 8 August 2013

[2] Gyanendra Prasad Joshi, Seung Yeob Nam, Sung Won Kim "Cognitive Radio Wireless Sensor Networks: Applications, Challenges and Research Trends", Sensors, 13, 11196-11228, 2013.

[3] Amir Sepasi Zahmati, Sattar Hussain, Xavier Fernando, Ali Grami "Cognitive Wireless Sensor Networks: Emerging Topics and Recent Challenges", TIC-STH, 593-596, 2009.

[4] Ozgur B. Akan, Osman B. Karli, Ozgur Ergul "Cognitive Radio Sensor Networks", IEEE Network, 34-40, 2009.

[5] Gayathri Vijay, Elyes Ben Ali Bdira, Mohamed Ibnkahla "Cognition in Wireless Sensor Networks: A Perspective", IEEE sensors journal, Vol 11, NO 3, 582-592, 2011

[6] Gayathri Vijay, Elyes Bdira, Mohamed Ibnkahla "COGNITIVE APPROACHES IN WIRELESS SENSOR NETWORKS: A SURVEY", 25th Biennial Symposium on Communications, 177-180, 2010

[7] Kok-Lim Alvin Yau, Peter Komisarczuk and Paul D. Teal "Cognitive Radio-based Wireless Sensor Networks: Conceptual Design and Open Issues", The 2nd IEEE Workshop on Wireless and Internet Services, 955-962, October 2009.

[8] Yan Zhang, Laurence T. Yang, Jiming Chen, "RFID and Sensor Networks: Architectures, Protocols, Security, and Integrations", 2009 by, CRC Press.

[9] Prakashgoud Patil, Umakant Kulkarni, N. H. Ayachit, "Some Issues in Clustering Algorithms for Wireless Sensor Networks", IJCA Special Issue on 2nd National Conference- Computing, Communication and Sensor Network CCSN, pp. 18-23, 2011.

[10] Rashad M. Eletreby, Hany M. Elsayed and Mohamed M. Khairy, "CogLEACH: A Spectrum Aware Clustering Protocol for Cognitive Radio Sensor Networks", International Conference on Cognitive Radio Oriented Wireless Networks, vol 9, 2014.

[11] Wendi B. Heinzelman, Anantha P. Chandrakasan, Hari Balakrishnan, "An Application-Specific Protocol Architecture for Wireless Microsensor Networks", IEEE Transaction on Wireless Communication, Vol 1, No. 4, pp. 660-670, October 2002.

[12] Mustafa Ozger, Ozgur B. Akan, "Event-driven Spectrum-Aware Clustering in Cognitive Radio Sensor Networks", Proceedings IEEE INFOCOM, pp. 1483-1493, 2013

[13] Ossama Younis, Sonia Fahmy, "HEED: A Hybrid, Energy-Efficient Distributed Clustering Approach for Ad Hoc Sensor Networks", IEEE transaction on mobile computing, VOL. 3, NO. 4, OCTOBER-DECEMBER 2004.

[14] Mu Tong, Minghao Tang, "LEACH-B:An Improved LEACH Protocol for Wireless Sensor Network", Wireless Communications Networking and Mobile Computing,pp 1-4, 2010

[15] W. Jia and J. Wang "Analysis of connectivity for sensor networks using geometrical probability ", IEE Proc.-Commun, Vol.153, No. 2,305-312, April 2006.

[16] Jianming Liu*, Xiaohong Jiang, Susumu Horiguchi,Tony, Tong Lee "Analysis of Random Sleep Scheme for Wireless Sensor Networks" Inderscience Enterprises Ltd.

[17] H. Legakis, M. Mehmet-Ali, J.F. Hayes "Lifetime Analysis for Wireless Sensor Networks", IEEE GLOBECOM, 2008.

[18] T. M. COVER, MEMBER, IEEE, AND P. E. HART, "Nearest Neighbor Pattern Classification", IEEE transaction on information theory, VOL. IT-IS, NO. 1, JANUARY 1967.

[19] Wendi Rabiner Heinzelman, Anantha Chandrakasan, Hari Balakrishnan,"Energy-Efficient Communication Protocol forWireless Microsensor Networks", Proceedings of the 33rd Hawaii International Conference on System Sciences, 2000.

[20] Moshe T. Masonta, Ntsibane Ntlatlapa, M. Mzyece1,"Energy and Spectrum Efficiency in Rural Areas based on Cognitive Radio Technology", Southern Africa Telecommunication Networks and Applications Conference SATNAC, 2009.

[21] J.F. Hayes and T. Ganesh Babu "Modeling and Analysis of Telecommunications Networks"

[22] Leonard Kleinrock "QUEUEING SYSTEMS", Vol J.

[23] Mansi Subheda, Gajanan Birajdar,"Spectrum Sensing Techniques in Cognitive Radio Networks: A Survey", International Journal of Next-Generation Networks (IJNGN),Vol.3, No.2, June 2011.

[24] Akrem Latiwesh, Dongyu Qiu "Energy Efficient Spectrum Aware Clustering for Cognitive Sensor Networks", 10th EAI International Conference on Communications and Networking, August 2015