Malicious and Malfunctioning Node Detection via Observed Physical Layer Data

THESIS

Tyler J. Hardy, 2LT, USAF

AFIT/GE/ENG/11-14

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

The views expressed in this paper are those of the author, and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the U.S. Government. This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States.

AFIT/GE/ENG/11-14

Malicious and Malfunctioning Node Detection via Observed Physical Layer Data

THESIS

Presented to the Faculty

Department of Electrical and Computer Engineering

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Master of Science in Electrical Engineering

Tyler J. Hardy, B.S.E.E.

2LT, USAF

March 2011

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

AFIT/GE/ENG/11-14

Malicious and Malfunctioning Node Detection via Observed Physical Layer Data

Tyler J. Hardy, B.S.E.E.
2LT, USAF

Approved:

/signed/	08 Mar 2011
Dr. Richard Martin, PhD (Chairman)	date
/signed/	08 Mar 2011
Maj. Ryan Thomas, PhD (Member)	date
/signed/	08 Mar 2011
Maj. Mark Silvius, PhD (Member)	date

AFIT/GE/ENG/11-14

Abstract

There are many mechanisms that can cause inadequate or unreliable information in sensor networks. A user of the network might be interested in detecting and classifying specific sensors nodes causing these problems. Several network layer based trust methods have been developed in previous research to assess these issues; in contrast this work develops a trust protocol based on observations of physical layer data collected by the sensors. Observations of physical layer data are used for decisions and calculations, and are based on just the measurements collected by the sensors. Although this information is packaged and distributed on the network layer, only the physical measurement is considered. This protocol is used to detect faulty nodes operating in the sensor network. The context of this research is Wireless Network Discovery (WND), which refers to modeling all layers of a non-cooperative wireless network. The focus in particular is the localization of transmitters, and detection of sensors affecting the localization. To accomplish this, a model for faulty sensors and two methods of detection are developed. Detection rates are analyzed with Receiver Operating Characteristic (ROC) curves, and the trade-off of detection versus localization error is discussed. Classification between faulty sensors is also considered to determine appropriate response to potential network attacks.

Acknowledgements

I would first like to thank my wife for all of her support throughout this whole process. I would not have made it without her. I also owe a lot of thanks to my advisor Dr. Martin. He presented challenging work, and gave me opportunities to do interesting research and present it. I would also like to thank my friends in the RF Signal Exploitation Lab for all their assistance. They assisted in both my education here at AFIT, and learning about being an officer in the Air Force. Finally, I would like to thank members of my committee for reviewing, editing, and challenging my work. Thank you everyone!

Tyler J. Hardy

Table of Contents

Page

List of Figures

List of Tables

List of Abbreviations

Malicious and Malfunctioning Node Detection via Observed Physical Layer Data

I. Introduction

This chapter will cover relevant background material to this research, including the development of localization via different methods and trust metrics in wireless networks. The motivation and research objectives for this work will also be discussed.

1.1 Background

With the abundance of wireless devices in use today, there is a need to identify essential pieces information about them. Examples of this type of information include:

- The frequency the device is communicating at
- Location of the device
- Communication patterns
- Type of information being shared
- Antenna patterns on the device
- Transmission power
- Environmental fading

This work in discovering information is called Wireless Network Discovery (WND) [1].

A primary user can use WND to discover information about two different types of devices. The first is a Non-cooperative Network (NN), a network or single device being operated by an outside actor. Information about the NN is not accessible to an outside network directly. The second case is a Cooperative Network (CN), under the

control of the primary user. Information about a NN may be shared or not. The CN might also not have the capabilities to share the information needed.

A device in the CN might not be complex enough to know its own location or other important statistics. This is often the case with large, cheap sensor networks. In this situation the CN can use known transmitter locations to locate its own sensors. An example of this might be a mobile cell phone user locating itself by using transmitted signals from cell towers.

This research focuses on one of the previously mentioned areas of interest, the location of the transmitter. WND, in particular the localization of a transmitter, can be completed using several different methods. These methods, including benefits and how they are completed, will be described in more detail in the literature review. In this case the localization is done by using the Received Signal Strength (RSS) measurements. Although it is useful information, the location of the transmitter is not the final goal of this research. Using the calculated location of the transmitter and the RSS values from the sensors in the CN, the objective is to determine if any of the nodes in the CN are not performing as expected. Differences between the expected RSS and the actual RSS could point to a variety of problems in the CN, including an attack on the network or poorly performing sensor nodes.

This type of WND may be used for many different reasons, but one particular area of interest is physical security. A user might want to deploy a sensor network along a perimeter of a secured area, and monitor Radio Frequency (RF) traffic from a transmitter. Trust in the sensor network measurements is important in maintaining a secure site. The user would like to accurately know the location as well as other information about the RF transmitter. Using the location already found, this research gives methods to determine trust and classify types of attacks on the sensor network.

These anomalies on the network are classified into two categories and have different implications. The first is when a sensor has been reprogrammed or attacked by someone with the intention of affecting the localization accuracy of the SN. This

type of node will be referred to as a *malicious nodes*. The sensors in the network can also have a variety of physical problems not caused by intentionally. This type of node is called *malfunctioning nodes*. The malfunctioning nodes can have low battery power, obscured communication paths, or any other environmental factors that are causing the sensor to not accurately report the RSS. The details will be described in the methodology section of this report.

1.2 Research Objectives

The main research objective is to determine trust in sensors in a network that is controlled by a user. This includes identifying faulty nodes and classifying the type of problem each is experiencing. This involves several different smaller objectives that must also be met to achieve the overall goal. One of these additional objectives is creating models that match both realistic situations and plausible scenarios where this system will be operated in. Models of the types of attacks the sensor network might suffer and the physical environment are important to correctly identifying malicious sensors. The trust being discussed here is the trust in the accuracy of the RSS measurements used to locate a transmitter of interest. Locating the transmitter is an important step in determining trust; therefore another research objective is to accurately locate a transmitter using plausible and applicable environmental and situational factors. The final objective is to develop an algorithm that takes the RSS measurements from the CN and uses them to determine trust. The algorithms will be applied and their ability to detect and classify will be compared. The final outcome is an algorithm that is able to be used in a sensor network deployed to monitor RF traffic, that is able to detect and classify anomalies in said network based on the physical layer data it is collecting.

1.3 Motivation

There are several motivating factors that provide the background for this work. One such factor is the increase in availability of wireless sensors. They can have uses

from monitoring industrial machines to observing weather and RF patterns [2]. With the growth in availability and use comes more interest in new applications for these sensors.

A second key motivation is the need for low complexity trust metrics. As it is described in the background, simple low power sensors are often better for this type of network. As a result of this it is beneficial to use simple algorithms and data collections to achieve as many objectives as possible.

The third motivation is a need to monitor or track RF activity near an area of interest. This could applied anywhere from a personal security system to military base level security. Using these three motivating factors the focus of the research was determined. A large majority of people carry some sort of RF transmitter on their person. By utilizing this knowledge, it is possible to locate or track people near an area of interest. Since these devices may not have similar communication protocols or willingly share information the RSS gives a way to locate a device for a large majority of potential transmitters. In using RSS to both locate and determine trust in the network, a more simple sensor can be used to accomplish both tasks.

1.4 Organization

In Chapter II, the literature review and key technical background research areas are discussed. This includes previous work in similar disciplines and how this research differs. In Chapter III, the derivations and simulation details are described. Chapter IV provides the results for simulations and analyzes how the proposed algorithms perform. Finally, Chapter V describes the future potential of this work and where this research stops and other research should begin.

II. Literature Review

This chapter will discuss related theory and experimental results to this research. The review will address several main areas that are covered in this work. The first of these areas is Wireless Network Discovery.

2.1 Wireless Network Discovery

As discussed in the introduction WND is one key area of related research. The objective of a sensor network, such as the type being studied here, is often to obtain information about the RF environment that it is operating in. One specific research area in this field is Cognitive Radio (CR). In the case of CR information discovered about the environment can help better utilize the RF spectrum [3]. Cognitive radio uses environmental monitoring along with cooperative communication to change and adapt the communication frequency, protocol, and other important signal characteristics, depending on the RF spectrum usage in the area. This is done through a combination of adaptive hardware and software. There are many examples of where cognitive radio is used in current technology [4].

Being able to adapt to a changing RF environment requires building a model of the environment to predict and adapt to. In [5] they develop two methods of radio spectrum information being shared in the network. The first is a Radio Environment Map (REM). REM is *a priori* information that is gathered and stored at one location and shared with other users in the network. The type of information included in REM could be locations, geographical information, service and networks, regulation and policy, activity profile of radio devices, and previous experience [6]. For this research building a REM type of database would not be likely because the transmitter is not part of the network, so little to no information is given freely. All information about the transmitter needs to be obtained through another sensor network. The second type of network map better suits this type of application. An Available Resource Map (ARM) is described by [4] as a map of the radio frequency spectrum obtained and updated by a sensor network. In previous work, [4], building an ARM has been done

with non-RSS based techniques. The idea of using physical layer data localization techniques has been proposed as an alternative to more traditional methods [7]. The advantage is that RSS based ARM building requires less sophisticated sensors in the network, reducing cost and power consumption. Next, a description of localization techniques and the building of a RF map with these methods will be described.

2.2 Localization

Technological advances have led to the use of Wireless Sensor Networks (WSN) in many different practical areas [8]. Localization in these WSN can come in many forms, from locating a cell phone using RSS for emergency purposes [9], to using an array of acoustic sensors to locate a vehicle [10].

Often it is useful to have a large group of sensors working together to accomplish tasks [2]. Due to the large number of sensors and the desire to be cost effective, low power and complexity sensors are used if possible. It is undesirable to need to change sensors repeatedly because of battery life. In [2] it is shown that in many cases the location of the device is essential in the functionality of the network. Conventional ways of determining locations of sensors such as Global Positioning System (GPS) require a more complex sensor and have higher power consumption. These concerns present a need for a method to determine the location of sensors without conventional methods. If the user is trying to locate or obtain information about a sensor that is not in the network, information will not be shared with the CN willingly, even if the transmitter has GPS available. Both situations provide motivation for research in the area of sensor localization.

The need for alternative localization methods has resulted in several different approaches being developed and researched. In [11] it is shown that there are four common measurements using communications signals to gain information about the sensors transmitting them. They are the RSS, Time of Arrival (TOA), Angle of Arrival (AOA), and Time Difference of Arrival (TDOA). The common theme of these methods is that they utilize the physical layer data being collected by the sensor.

Each method uses a different aspect of the signal, and has distinct advantages and disadvantages over the other methods. Each method will be discussed, including how they are able to locate a transmitter and benefits and disadvantages.

2.2.1 Time of Arrival and Time Difference of Arrival. TOA and TDOA are very similar location techniques. Both methods utilize the time it takes to receive a transmitted waveform and the known speed of propagation of the signal to determine the distance. These techniques also can work in two distinct cases. The first is where the location of multiple transmitters is known, but a single receiver is unknown. This is called *navigation*. In the other case multiple receivers with known locations are used to receive the signal from a single transmitter. This is called *source localization* [12]. The research is focused on the latter, source localization.

TOA and TDOA require a high level of synchronization on the network to achieve realistic error measurements [11]. TOA uses the time of flight from the known transmitter to the receiver. TDOA on the other hand shares the arrival time with another transmitter or receiver, depending on the type of location or navigation, and the distance is based on the difference between the two arrival times.

Figure 2.1 demonstrates the concept behind a system using TOA to determine a location.

There are several important considerations and assumptions when using TOA for location. The first is that the locations of the receiver or transmitter are known. When using a TOA system for locating a transmitter, the location of the receivers are assumed to be known, and when navigating using multiple transmitters their location is assumed known. TOA and TDOA systems also rely on synchronization and timing to be accurate [2]. The sensors also need to be able to determine characteristics in the waveforms so the time of arrival for the signal can be determined.

In general a sensor using this type of localization will need to be more complex. Timing and synchonization, along with the necessary hardware and software to compute when the same message arrives at two different sensors requires added

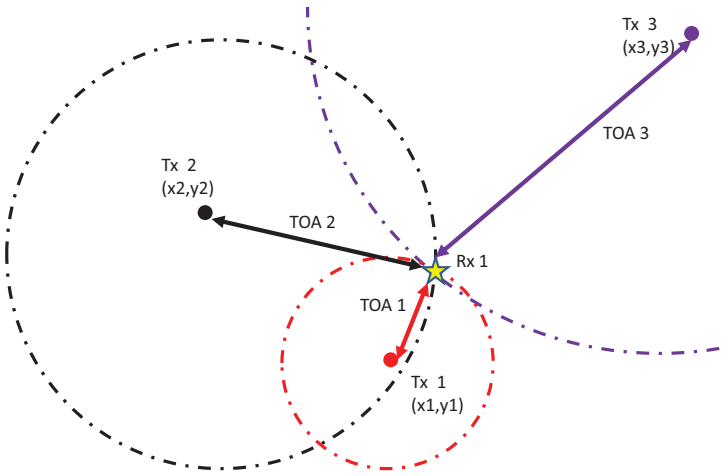

Figure 2.1: Concept drawing for a TOA system.

complexity and bandwidth. The benefit of this is that the TDOA systems are more accurate than other methods that will be discussed. It is less susceptable to noise in the atmosphere than RSS systems. The main source of error in this type of system is multi-path. In multi-path, the signal takes not only the direct path from transmitter to receiver, but also several other indirect paths off objects.

2.2.2 Angle of Arrival. Angle of Arrival is a method that uses an array of sensing devices to determine the direction of the arrival of the signal. This information is often used in addition to TDOA and RSS methods to provide an additional piece of information [2]. The array of sensors, acoustic or antennas, can collect two types of information to determine the angle of arrival. Using differences in signal arrival between sensors the phase delay information can be used to determine the angle of arrival [13]. The signal measurements can be averaged prior to computing the AOA, or AOA estimates can be made then averaged. In [14], the authors compare these two approaches.

The downfall of using AOA is similar to that of TDOA, noise and multi-path effects cause error in the estimation. Another key point is that AOA is able to

determine the direction, but not the distance to the transmitter of interest. Depending on the situation, this might not be sufficient information about the transmitter.

Figure 2.2 demonstrates how the AOA method works using a phase delay system. The four circles represent sensors that record the information in the waveform $f(t)$. Each sensor experiences a different phase delay. This comparison of phase delay allows the user to determine the direction of arrival for the waveform.

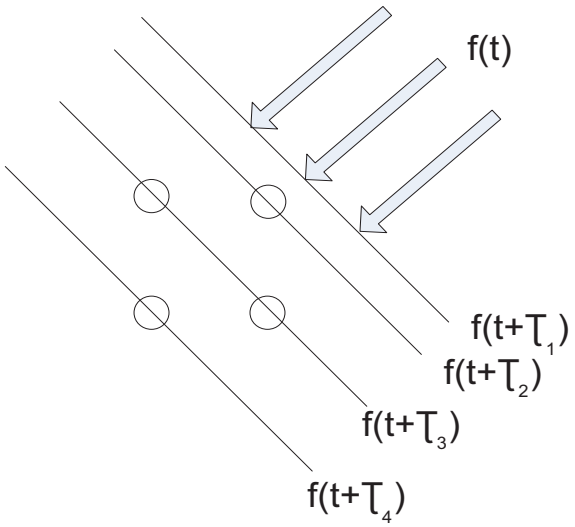

Figure 2.2: This figure demonstrates the concept behind AOA based techniques.

2.2.3 Received Signal Strength. This method of localization uses the power contained in the communication signal measured at the receiver. A sensor with the ability to collect the power in a signal can be utilized for this method. RSS localization relies on the fact that in free space signal power fades as the square of the distance, d^2. If the transmission power is known or estimated, the distance from the receiver can be estimated from the RSS. Using multiple receivers provides the ability to more accurately estimate the location of the receiver. This approach is used for this research. More details on how the location is determined from the RSS measurements are included in Chapter III.

The main advantage to using the signal power to determine location is the low complexity sensors. In RSS, the necessary data is more easily collected than other methods. As a result it can be used with a greater number of sensors for the same cost.

There are several downsides to using an RSS based localization technique. The first is that it is more susceptible to noise or interference affecting the estimation. Noise in the environment will directly impact what the measured RSS value is. However in TDOA, the important quantity, time, is indirectly affected by noise. Also, many effects on the signal power such as interference from objects are grouped together and modeled as random noise. Therefore the model for received power does not exactly model the propagation of the wave.

2.3 Received Signal Strength Research

There is a lot of research ongoing in the area of RSS and localization. The topics in this area can be broken down into three main focus areas: the RSS model, optimizations and algorithms, and applications and feasibility. In [15], a sequential Monte-Carlo simulation is used to model a mobile RSS sensing network, and performed favorably versus a Maximum Likelihood Estimation (MLE) method, with 40% improved localization error. In [16], a model is built to estimate the 2-D position with the presence of nuisance parameters such as a third dimension of position or number of sensors in the network. RSS localization is also used in a variety of applications from an alternative to more complex cell phone location algorithms [17], to using acoustic energy for multiple source vehicle location [10].

Optimization and more efficient algorithms to utilize RSS based localization is the major focus of research in this area. This has a large subset as well. In [18], for instance, an algorithm is developed to deal with physical layer data attacks on the SN. The authors discuss how reflective or absorbing materials can affect the RSS measurements. The proposed algorithm deals with any type of attack on physical layer data. The Signal Strength Difference (SSD) is used between two sensors with

known locations. The SSD method maintains accuracy while the previous methods were degraded by the attacks. In [19], localization was improved by utilizing outlier theory. To more accurately compare algorithms, research has been done to include the error in RSS measurement [20]. This helps to compare location algorithms at multiple distances because RSS error might have a different effect on each algorithm.

2.4 Detection and Estimation

Another key area of development relevant to this research is detection and estimation. The data is collected by all of the sensors in the network and reported back to a command center. This is where the decisions are made by detection and estimation theory. To determine the location from a collection of power measurements Maximum Likelihood Estimation (MLE) is used. MLE is asymptotically unbiased and efficient for large data sets [21]. Other methods have been used in estimating position from data [15, 18]. The derivations for how this estimator is used are included in Chapter III. The MLE of θ, a parameter of vector $\underline{X}$, is found by:

$$\hat{\theta}_{ML} = \arg\max_{\theta} \mathcal{L} \tag{2.1}$$

$$\mathcal{L} = \ln p(\underline{X}|\theta) \tag{2.2}$$

After using the MLE to find the location of the transmitter, the same data that was collected is also used to determine trust. To do this, hypothesis testing is applied. Hypothesis testing uses data collected to decide between two or more claims or assertions about population characteristics or parameters [22]. In this case the claim is that the node is trusted or not trusted.

2.5 Trust Metrics

In a WSN, the sensor nodes and data are vulnerable to attacks and sources of error. To help determine when attacks or errors happen a trust metric is calculated. Trust is found in different networking research areas. In [23], the authors describe trust as having two important categories; communication trust and data trust. The first of these is the trust in the sensor node that it is communicating as expected. Network attacks can cause sensor nodes to not report packets, misdirect packets, and a variety of other communication issues. These issues affect the performance, and cause the communication to not be as expected. The second area is data trust. The purpose of most WSNs is to monitor an area and gather data. The information collected can also be attacked. This could involve the hardware attacks, changing the environment, or causing the sensor to report false or misleading data points. The authors in [23] argue that a trust metric based on both categories gives a better determination of a sensor's overall trust level.

Trust in a WSN can be obtained by several methods. In [24] intruders in a network are detected by comparing a sensors decisions about neighboring sensor nodes. Other proposed methods in use include majority voting [25], and using witnesses to verify correct data transfer from a sensor network to the data center [26]. Another approach is to build a trust metric that suits the particular need of the network. In [27] the authors describe important aspects to the network functionality and then build a model that account for how each sensor in the network is performing in each area. This includes both communication and data metrics. In general these techniques build trust metrics based on many observations of occurrences in the network. This work on the other hand uses observations of data that is collected at the physical layer. The difference is that this method uses the collected data to make binary decisions on specific sensors. Other techniques may use trust for other purposes then a binary decision. For example if a sensor is always reporting high RSS values, the pattern may lead to the sensor being identified and the RSS values can be included but corrected.

2.5.1 Outlier Detection. An outlier in the context of a WSN is sensed data that differs from the normal pattern [28]. The authors also describe how outlier detection can be broken into three sources and categories:

- *Fault Detection*, where errors cause large deviations from expected results. This can be short or long term depending on the type of error. Errors also have a high probability of occurring.
- *Events*, where a physical phenomenon that will cause a sensor to be a long term outlier. The probability of occurrence for an event is low in general.
- *Malicious Attacks*, where an intruder will intentionally change the functionality of the sensor. A sensor subject to a malicious attack is diffuclt to classify as an outlier. The intruder will attempt to be hard to notice.

Outliers have been detected using statistics, nearest neighbor, cluster, classification, and spectral decomposition based approaches. Taking into account the low cost and low battery life of the WSN that this research is focusing on, the statistic based method of determining the outliers is used. This method is specifically tailored to this application, because the physical measurements are used to determine another quantity, the location of a transmitter. Although these techniques are similar to outlier detection, these metrics will not correlate directly with them.

2.5.2 Physical Layer Trust. The goal with physical layer trust is to determine the trust in a sensor node without using high level network communication protocol. The metric is based on the information gathered from the physical signal. In this case, the signal power. Many of the approaches described in the previous section require more information to determine trust. The advantage of the physical layer trust metric is that it utilizes data that is already collected for the localization, the main objective of the sensor network. Therefore there are no additional requirements on the sensors in the network. This type of information would lead to the data trust mentioned earlier. This research does not specifically utilize data trust. Decisions

are made over a set of observations, but no patterns of behavior are shared between sets of observations. To truly consider data trust, a metric would be placed on a sensor of how much trust it contains based on multiple sets of observations and the determination of if it is faulty or correctly functioning.

2.6 Contributions

This research utilizes many different disciplines to detect malicious nodes with only observations of physical layer data. These methods are combined and used to create an algorithm that is able to accomplish the research objectives. The RSS localization theory used to find the location of the transmitters of interest is used from [2,7], the model is applied to the specific potential scenarios that were developed in this research. Models of potential malfunction or malicious nodes were created. This is a main area of contribution of this research. These models describe how a potential attack on a network might appear in data. Detection and estimation theory such as MLE and hypothesis testing were used to develop an algorithm that applied to this specific model and Receiver Operating Characteristic (ROC) theory is used to analyze and compare results. This method of determining trust also differs from previous outlier detection techniques [19]. The authors use a window operation prior to their localization technique. The window gives a value of confidence in each data set. The filter consists of a Hampel Filter, to detect outliers in the data, and a Kernel Density Estimator. In this research two approaches are used based on distributions of sensor models and propagation models.

III. Methodology

This chapter will detail the methodology used to develop and test the algorithms used to detect and classify broken and malicious sensor nodes. A system model description is given to demonstrate the model used. Following the system model, a description of the development of algorithms is given.

3.1 System Overview

The system consists of three major components: the cooperative sensor network deployed by the user, the command center used for collecting and processing data, and the "hostile" transmitter that is being tracked.

- *Cooperative Neetwork*: The first component is the cooperative network. This network is a group of sensors that are able to measure the signal power contained in a signal they receive. For this research, the sensors have knowledge of the frequency that the transmitter is operating at. This could be possible by a few methods; a separate process to find any devices communicating in a predetermined area, WND, or prior knowledge of the device. The latter is the case here, and it can be reasonably assumed that someone wanting to track a transmitter of interest would know details about the device, e.g cell phone. The sensors in the cooperative network also have independent knowledge of their location and are stationary. The location could be from a GPS unit on each sensor, or the command center could have *a priori* knowledge of their location from the user deploying the sensors in the field.
- *Transmitter*: The second key aspect of the system is the transmitter. In reality, there are many unknown aspects to the device and channel that will affect how accurately the sensor network will be able to locate the transmitter. Some of the factors are the polarity of the antenna, the original transmission power, and various channel and environmental factors. In [7] techniques are shown to estimate these factors without prior knowledge of them. For this research, these factors are considered known either from estimation or prior knowledge.

- *Command Center*: The command center is the final part of the system. All data is reported back and collected here. The algorithms will be implemented and decisions will be made here. In simulation this is done with MATLAB, but in theory it would be done with a device that is able to communicate with the sensor network, and has enough processing power to complete the necessary calculations.

Each of these components are shown in Figure 3.1, a layout of how this system may be used.

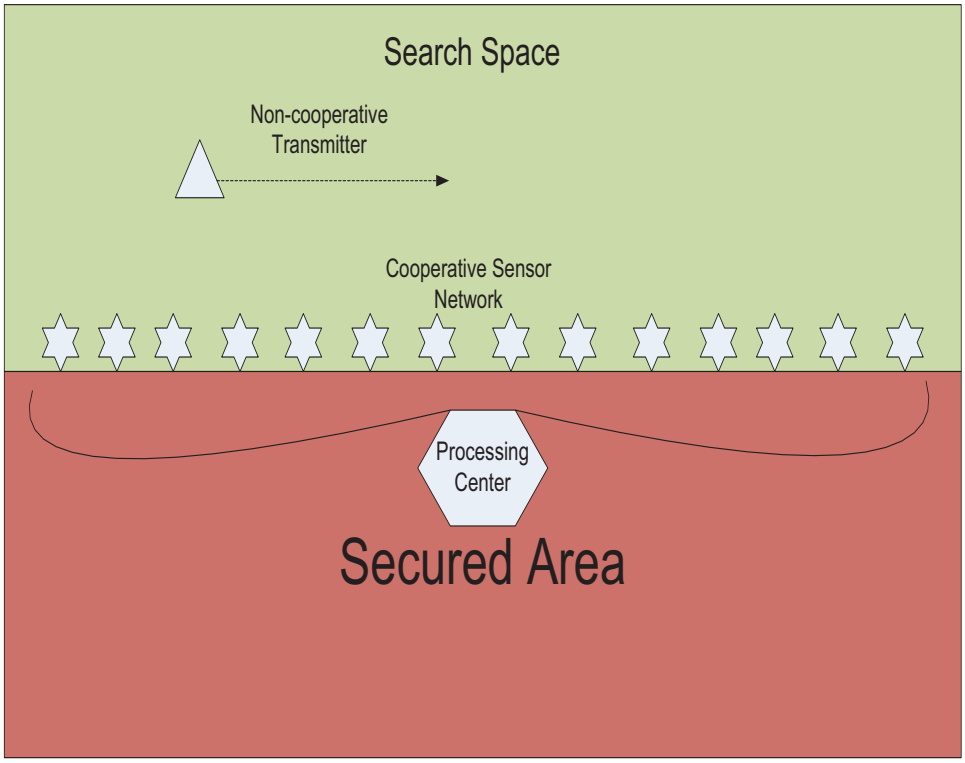

Figure 3.1: A system block diagram of how the sensor network, transmitter, processing center, and area of interest all interact.

The system shown in Figure 3.1 is implemented into MATLAB, and it can be seen in Figure 3.2. In this figure the sensors in the CN are distributed randomly in a predefined area. The faulty sensor is also noted with a large trianle. The plot shows the transmitter moving along a path for each time step T. Finally, the estimation

of the transmitter location is shown both with the faulty sensor included and the non-faulty sensor data.

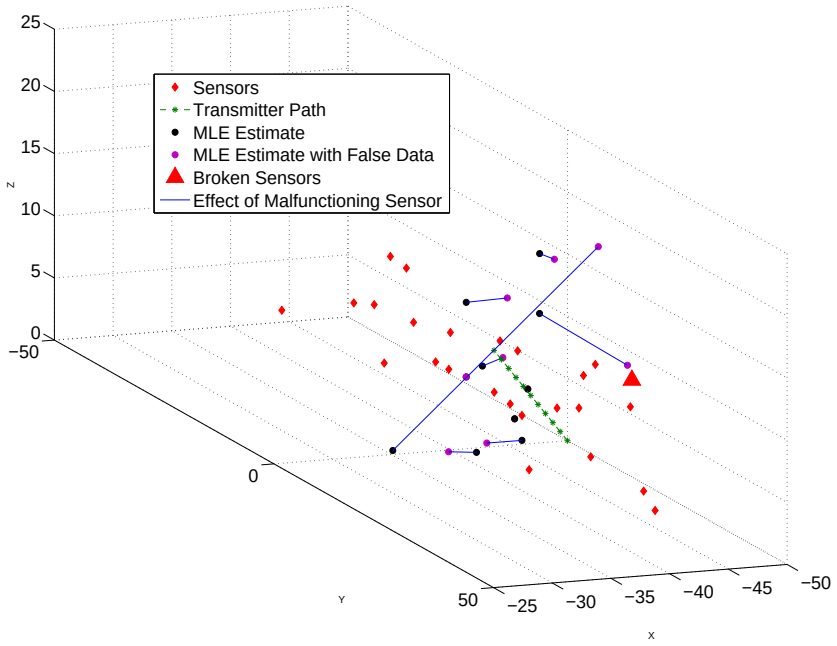

Figure 3.2: A simulation diagram example.

Figure 3.2 shows the effect faulty sensors can have on the localization ability of the network. The lines show the potential distance error introduced by just one fauly sensor in the network.

3.2 RSS Model

Table 3.1 is a collection of variables used in this research.

3.2.1 RF Signal Propagation. The derivation begins with the model for signal power. In free space, a RF signal will decay with respect to the distance squared. In previous research [7] the model for received power in dB can be shown to be:

$$m_s(d_s) = P_0 - \eta 10 \log_{10}\left(\frac{d_s}{d_o}\right) \tag{3.1}$$

Table 3.1: Table of variables used in this research

Variable	Definition	Dimensionality	Units
S	Number of sensors	Scalar	Unitless
T	# of Observations	Scalar	Unitless
K	# of Independant Trials	Scalar	Unitless
V_0	Deterministic RSS reported from faulty sensor	Scalar	dBm
θ	Location of transmitter	1×3	Unitless
$\hat{\theta}$	Estimated location of transmitter	1×3	Unitless
$\hat{\theta}_{ML}$	ML estimate of location of transmitter	1×3	Unitless
ϕ_s	Location of s^{th} sensor	1×3	Unitless
p_0	Probability of reporting true RSS measurement	Scalar	Unitless
P_s	Power received at s^{th} sensor including AWGN	Scalar	dBm
P_0	Logarithmic transmitted power	Scalar	dBm
d_0	Reference distance	Scalar	meters
d_s	Euclidean distance between emitter and s^{th} sensor	Scalar	meters
$m(\phi, \theta)$	Received power at s^{th} sensor without noise present	Scalar	dBm
$\underline{m}$	Received power vector of all sensors without noise present	$1 \times$ S	dBm
$\underline{P}$	Received power vector of all sensors with AWGN	$1 \times$ S	dBm
$\underline{n}$	Additive White Gaussian Noise	$1 \times$ S	dBm
Γ_0	Power transmitted	Scalar	mW
I_S, I_T	Identity matrix	S $\times$ S, T $\times$ T	Unitless

where η is the path loss exponent and P_0 is the reference received power at a known reference distance, d_0. The effect of noise in this log-normal model is assumed to be Gaussian with a standard deviation of σ_e. Noise in this model is error due to log-normal fading, not interference. The distance between two objects in this method is the three dimensional distance between the transmitting sensor and the receiving sensor, $d_s = ||\phi_s - \theta||$, where $[x_0, y_0, z_0] = \theta$ is the location of the transmitter and $[x_s, y_s, z_s] = \phi_s$ is the location of the s^{th} sensor. After this point the received power will be described by the locations of the sensors and transmitters, $m(\phi_s, \widehat{\theta})$ if the transmitter is estimated or $m(\phi_s, \theta)$ for derivations where the transmitter location is known. Using this model, the $S \times 1$ received power vector, $\underline{P}$, has a distribution of

$$\underline{P} = \underline{m}(\underline{\phi}, \theta) + \underline{n} \tag{3.2}$$

$$\underline{n} \sim \mathcal{N}(\underline{0}, \sigma_e^2 I) \tag{3.3}$$

Where I is an $S \times S$ identity matrix. For this research S is defined as the number of sensors in the cooperative network, K is the number of independent trials, and T is the number of observations per trial. At each new observation the transmitter is moved one meter along a specified path.

3.2.2 Data Creation and Variation. To obtain data to test the theories proposed, the RSS model and pseudo random number generation are used. Creating synthetic RSS values in simulations requires access to some truth values that will not be known to the user. The location of the transmitter is not known, but used in the data creation and it gives the simulated RSS values. Then $m_s(d_s)$ from Equation (3.1) is calculated, the ideal RSS value. The value is then corrupted with the modeled noise, n.

$$P_i = m_s(d_s) + n_{sim} \tag{3.4}$$

where n_{sim} is AWGN generated by MATLAB. The noise is zero mean with unit variance, but is multiplied by the simulated variance, σ_e. Data is created for each independent trial of the simulation. To vary the simulations, the variance can be changed. Or as will be described later in the sensor node behavior section 3.2.5, the simulated RSS can be changed to have the malicious or malfunctioning node behavior.

3.2.3 Transmitter Localization. Using this distribution, and knowledge of the system model described above, a MLE can be made on the transmitter location θ. First a Probability Density Function (PDF) is built across all of the sensor observations

$$p(\underline{P}|\theta) = \prod_{s=1}^{S} \frac{1}{\sqrt{2\pi}\sigma_e} e^{-\frac{1}{2\sigma_e^2}(P_s - m(\phi_s,\theta))} \tag{3.5}$$

To find the MLE the log likelihood function, $\mathcal{L}$, needs to be maximized. $\mathcal{L}$ is found by taking the logarithm of the joint distribution and finding the argument θ that maximizes it [21].

$$\mathcal{L} = \ln\left[p(\underline{P}|\theta)\right] = \ln\left[\prod_{s=1}^{S} \frac{1}{\sqrt{2\pi}\sigma_e} e^{-\frac{1}{2\sigma_e^2}(P_s - m(\phi_s,\theta))^2}\right] \tag{3.6}$$

$$\mathcal{L} = \psi - \frac{1}{2\sigma_e^2} \sum_{s=1}^{S} (P_s - m(\phi_s,\theta))^2 \tag{3.7}$$

$$\widehat{\theta}_{ML} = \arg\max_{\theta}[\mathcal{L}] \tag{3.8}$$

where ψ is a scalar, and does not depend on θ. To find the values of θ that maximize $\mathcal{L}$, the gradient with respect to θ is found and set equal to $\underline{0}$. This is analogous to finding a maximum by taking the derivative and setting it equal to 0.

$$\nabla_\theta \mathcal{L} = \underline{0} = \nabla_\theta \left(\psi - \frac{S}{2\sigma_e^2} ||\underline{P} - \underline{m}(\underline{\phi},\theta)||^2\right) \tag{3.9}$$

Since ψ does not depend on θ, it will drop out of the gradient. Solving (3.9) for θ gives the MLE of the position of the transmitter, $\widehat{\theta}_{ML}$. This is difficult to solve analytically. To find the MLE, a numerical approach is used. Combining and reducing (3.8) and (3.9) the MLE can be found by

$$\widehat{\theta}_{ML} = \arg\min_{\theta_g} ||\underline{P} - \underline{m}(\underline{\phi}, \theta_g)||^2 \tag{3.10}$$

Where θ_g is a possible transmitter location that will be searched. From here, the MLE can be found numerically. This is done by using a search grid and calculating the cost for each θ_g, and finding the coordinates of the minimum value.

3.2.4 Location Estimation in Simulation. As mentioned above, solving analytically for the location is difficult. In MATLAB simulation, determining the MLE can be done more simply. First, a search space is defined. In this research an area of 100 meters in both x and y are used as well as 50 meters of elevation. The transmitter can be located anywhere in the search grid. To find the location, each index of the matrix in MATLAB is simulated as 1 meter. At each index the simulated RSS, P_i, is used to calculate cost $\mathbf{C}$:

$$\mathbf{C} = ||\underline{P}_i - \underline{m}(\underline{\phi}, \theta_g)||^2 \tag{3.11}$$

After the matrix is populated, the minimum value is found. Then the three coordinates that correspond to the minimum value of $\mathbf{C}$ give the values that minimizes (3.10). The three coordinates when put into a vector give $\widehat{\theta}_{ML}$.

3.2.5 Sensor Node Behavior. An objective of this research is to detect and classify two types of non-functioning sensors in a SN, malfunctioning and malicious based on the data trust. When referring to a sensor that is not working correctly, either malfunctioning or malicious, that sensor will be called *faulty*.

A faulty sensor in this model is a sensor that is not reporting the correct RSS between itself and the transmitter. The first example of this is a malfunctioning sensor. A malfunctioning sensor is a sensor that is not performing correctly, but it does not have the intent to disrupt the function of the SN localization. Examples of this type of classification might be physical damage, low battery power, and abnormal interference. The goal with this type of sensor is to detect and classify it so it can either be replaced, or the reported values can be ignored or adjusted. If the RSS is consistent with the true value plus an offset, this can be estimated or corrected in processing until the sensor can be fixed. For the s^{th} sensor in the network the RSS can be shown to be

$$P_s = V_0 + n \tag{3.12}$$

where V_0 is false RSS value. This values is considered constant throughout a simulation. V_0 represents a signal stregth that is independent of the distance between the transmitter and sensor.

The second type of non-functioning sensor being considered is a malicious node. This type of node is trying to intentionally disrupt the localization of the transmitter. A sensor with this intent could report values that greatly affect the location estimation, but this would be easy to detect. The sensor might try to report false values a specified percentage of the time to prevent detection. A malicious user trying to disrupt the localization would need to weigh the benefits of localization error with the cost of detection. A malicious sensor will report a RSS value of the attackers choosing. This value V_0 is then corrupted with artificial noise, $\widehat{n}$, and reported as the measured RSS value to the processing center. The artificial noise is chosen to have the same distribution as the noise in the environment, (3.3). This noise is to prevent the user from being able to easily detect the difference between malicious and true values. If the same value is reported for multiple time observations, it would be very easy to detect.

Table 3.2: Table of faulty sensor characteristics

Characteristic	**Malfunctioning**	**Malicious**
% of Sensors	X	X
V_0	X	X
p_0		X

$$P_s = V_0 + \widehat{n} \tag{3.13}$$

The model is developed to consider both of these cases. The probability p_0 is defined as the probability of a sensor to report the true RSS between itself and the transmitter. A correctly working sensor would utilize (3.13) with $p_0 = 1$. If there is a malicious sensor in the network, it might use a p_0 value somewhere between 0-1 depending on the amount of localization error needed versus the cost of detection, (3.12). A malicious sensor with p_0 of 0 can be considered a malfunctioning sensor. It always reports the false RSS value, V_o. Combining (3.12) and (3.13) with p_0 as described gives:

$$P[P_s = m(\phi_s, \theta) + n] = p_0 \tag{3.14}$$

$$P[P_s = V_0 + \widehat{n}] = 1 - p_0 \tag{3.15}$$

Once each sensor is determined malicious or broken, it will remain in that state for each time step t. For each independent trial K, different sensors are randomly selected.

3.2.6 Simulating Sensor Behavior. To simulate the effect these types of sensor will have on the performance of the localization, both needed to be simulated. There are several key characteristics of the sensors. These parameters and which sensors they can be used on are shown in Table 3.2.

In both situations the percentage of fault out of the total sensors is needed. To decide which sensors are faulty, a random number is selected for each faulty sensor. These numbers each represent a vector index of a sensor in the network. Once the sensors are selected to be faulty, the next step is determining whether they are malicious or malfunctioning. This is determined prior to the simulation. In Equation (3.4), it is shown how data is generated for a non-faulty sensor. For a faulty sensors, the simulated data is overwritten with the new values.

Both the original synthetic RSS data and the replaced false RSS data with faulty nodes are kept. They are utilized to determine errors in localization for comparison in techniques.

3.2.7 Power Map. After estimating the location of the transmitter, the RF propagation model is used to map the RSS values throughout the search space. The location of the sensor nodes are known, so the power map gives an expected RSS value for each sensor location. The expected RSS, when compared to the actual RSS, provides one method of determining trust in each sensor node. Figure 3.3 demonstrates how the power from the transmitter can be modeled.

The transmitter is modeled with an omni-directional antenna, as seen in the figure. The signal is propagating in all directions equally.

3.3 Algorithm Development

After all the data is collected and the transmitter location has been estimated, the data is used again to determine trust in the network. Two separate methods of determining trust from the data will be derived and tested. The first method compares errors between the expected and actual RSS, and the second method uses a GLRT to test each sensor against a threshold.

3.3.1 Generalized Likelihood Ratio Test. In this method, a Likelihood Ratio Test (LRT) is used. A LRT compares two hypotheses made about a set of data.

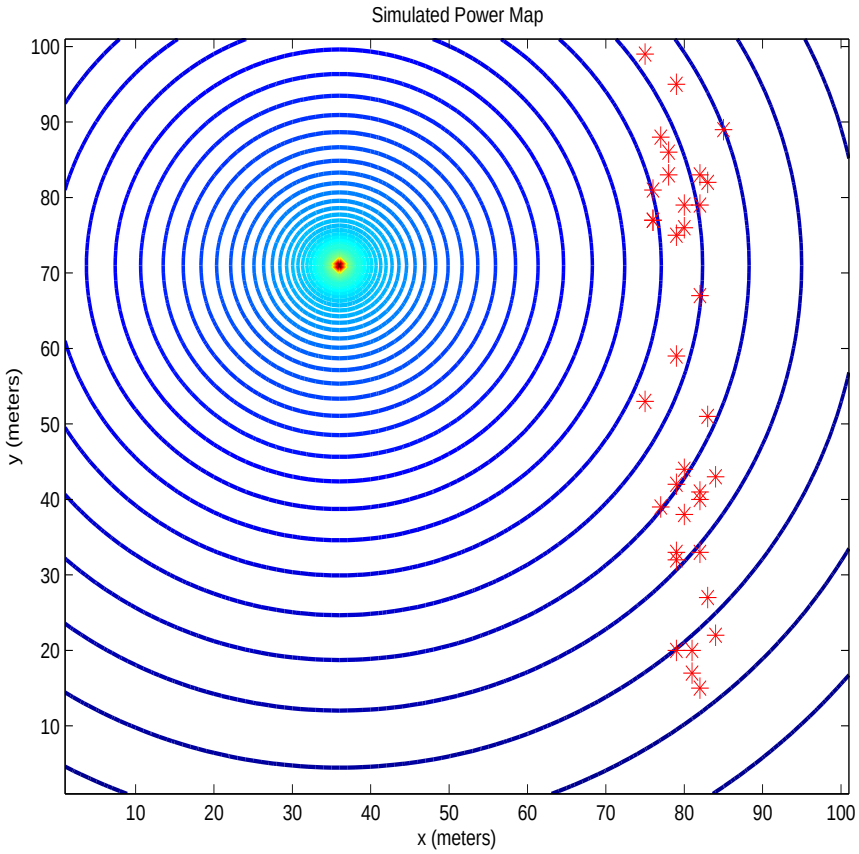

Figure 3.3: A simulated power map that demonstrates power contour levels in the search space.

When one of the parameters is estimated, the LRT becomes a Generalized Likelihood Ratio Test (GLRT) [29]. In this case, the estimate of the transmitter location, $\widehat{\theta}_{ML}$, is not the hypothesis being tested, but is still unknown. It needs to be estimated and included in the LRT.

In this model there are two possible hypotheses for the RSS of a specific sensor, s.

$$H_0: \quad P_{s,t} = m_{s,t}(\phi_s, \theta) + n_{s,t}, \quad t = 1, 2, ..., T \tag{3.16}$$

$$H_1: \quad P_{s,t} = V_0 + n_{s,t}, \qquad t = 1, 2, ..., T \tag{3.17}$$

where $n_{s,t}$ is Additive White Gaussian Noise (AWGN) for both cases. In the case of H_1, the noise is generated by the malicious sensor to simulate the noise in the true RSS measurements. In both cases the noise is considered independent.

$$n_{s,t} \sim \mathcal{N}(0, \sigma_e^2) \tag{3.18}$$

In the H_0 hypothesis the sensor is assumed to be working correctly. In the H_1 hypothesis the sensor is assumed to be faulty. Using these two hypotheses to create distributions yields the following:

$$P_{s,t}|H_0 \sim \mathcal{N}(m_{s,t}(\underline{\phi}, \underline{\theta}), \sigma_e^2) \tag{3.19}$$

$$P_{s,t}|H_1 \sim \mathcal{N}(V, \sigma_e^2) \tag{3.20}$$

As stated earlier one method to differentiate between the two hypotheses is a LRT. In this case the LRT compares the two hypotheses to a threshold γ:

$$\frac{p(\underline{P}_s|H_1)}{p(\underline{P}_s|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} \gamma \tag{3.21}$$

This is the case for each sensor s. The likelihood ratio can be compared to a threshold γ to decide between the two hypotheses.

$$\alpha_s(\underline{P}_s) = \prod_{t=1}^{T} \frac{p(P_{s,t}|H_1)}{p(P_{s,t}|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} \gamma \tag{3.22}$$

Where S represents the number of sensors in the cooperative network and T is how many observations are taken from each sensor. These observations can be made on a moving transmitter, as is the case in this simulation, or a stationary transmitter. Using the distributions (3.19) and (3.20) and the LRT (3.22), the following can be shown:

$$\alpha_s(\underline{P}_s) = \ln\left[\frac{\prod_{t=1}^{T}\frac{1}{\sqrt{2\pi\sigma^2}}e^{\frac{-(P_{s,t}-V_0)^2}{2\sigma_e^2}}}{\prod_{t=1}^{T}\frac{1}{\sqrt{2\pi\sigma_e^2}}e^{\frac{-(P_{s,t}-m_{s,t}(\phi_s,\theta))^2}{2\sigma^2}}}\right] \tag{3.23}$$

$$\alpha_s(\underline{P}_s) = \sum_{t=1}^{T}\left(\frac{-(P_{s,t}-V_0)^2}{2\sigma_e^2} - \frac{-(P_{s,t}-m_{s,t}(\phi_s,\theta)^2}{2\sigma_e^2}\right) \tag{3.24}$$

$$\alpha_s(\underline{P}_s) = \sum_{t=1}^{T}\left(-2P_{s,t}(m_{s,t}(\phi_s,\theta)-V_0) + m_{s,t}(\phi_s,\theta)^2 - V_0^2\right) \tag{3.25}$$

In 3.25, $\alpha_s(\underline{P}_s)$ can be considered optimal if there is only one faulty sensor in the network, the transmitter locations are known to the user, and the value of V_0 is known. In reality there are usually more than one faulty sensor, and the parameters are estimated.

In the previous equations, the received power without noise, $m_{s,t}(\phi_s,\theta)$, depends on which sensor, s, and which observation, t, are being tested. For each sensor, a three dimensional location, ϕ_s, is needed and for each observation, the estimate of the transmitter location, $\widehat{\theta}_{ML}$. Combining (3.21) and (3.25) yields a final GLRT to use to compare various system components and conditions. Also, the factor of $\frac{1}{\sqrt{2\pi\sigma^2}}$ is divided and included in the γ term, because it does not change with s or t. A GLRT is used in this case because the offset value, V_o is estimated and then used in the LRT.

$$\alpha_s(\underline{P}_s) \underset{H_0}{\overset{H_1}{\gtrless}} \gamma \tag{3.26}$$

The offset value V is estimated with MLE. The distribution of a faulty sensor with false RSS value V_0 is shown in 3.13. Building a joint distribution of the s^{th} sensor over T observations can determine the MLE for that sensor.

$$p(\underline{P}_s|V_0) = \prod_{t=1}^{T}\frac{1}{\sqrt{2\pi}\sigma_e}e^{\frac{-1}{2\sigma_e^2}(P_{s,t}-V_0)^2} \tag{3.27}$$

Taking the log likelihood function and setting the derivative to zero gives:

$$\mathcal{L} = \ln\left[\prod_{t=1}^{T} \frac{1}{\sqrt{2\pi}\sigma_e} e^{\frac{-1}{2\sigma_e^2}(P_{s,t}-V_0)^2}\right] \tag{3.28}$$

$$\mathcal{L} = 0 = \frac{d}{dV_0}\left[\psi - \frac{1}{2\sigma_e^2}\sum_{t=1}^{T}(P_{s,t}-V_0)^2\right] \tag{3.29}$$

where ψ does not depend on V_0. Solving for V_0 gives the MLE.

$$0 = \frac{d}{dV_0}\left[\sum_{t=1}^{T}(P_{s,t}^2 - 2P_{s,t}V_0 + V_0^2)\right] \tag{3.30}$$

$$\widehat{V_0} = \frac{1}{T}\sum_{t=1}^{T} P_{s,t} \tag{3.31}$$

where the $\widehat{V_0}$ does not depend on knowing the parameters of the distribution, particularly σ_e.

The threshold test in (3.26) is used to compare to a sliding threshold γ, to determine the Receiver Operating Characteristic (ROC) for this method.

3.3.2 Mean Squared Error. Another method used to determine the ability to identify and classify working and non-working sensors is the Mean Squared Error (MSE). In this method the RSS is compared to the expected value of the RSS, given the coordinates of the sensors and the MLE of the transmitter. In this case there are also two hypotheses that are used

$$H_0 : e_s = \underbrace{(m(\phi_s,\theta)+n)}_{P_{s,t}} - m(\phi_s,\widehat{\theta}_{ML}) \tag{3.32}$$

$$H_1 : e_s = (V_0+n) - m(\phi_s,\widehat{\theta}_{ML}) \tag{3.33}$$

The distributions here depend on a couple of things that may be changed by the attacker. The value of $\widehat{\theta}_{ML}$ is used to determine the expected RSS. If the attacker controls enough of the sensors, this estimate becomes so poor the correctly working sensors appear to have high error measurements. Similarly, the value of V_0 is assumed known, and error in the knowledge or estimation of this value determines how accurate this method is. Although the model for errors is defined, this model does not take them into account. Instead, the MSE is used. The logic is that the error between faulty sensors actual and expected RSS is higher than the error for a non-faulty sensor. The second algorithm is shown below:

$$\beta_s(P_{s,t}) = \frac{1}{T}\sum_{t=1}^{T} {e_{s,t}}^2 \underset{H_0}{\overset{H_1}{\gtrless}} \gamma \tag{3.34}$$

where $e_{s,t}$ is the error between the actual RSS value, $P_{s,t}$, and the expected value based on the power map, $m(\phi_s, \widehat{\theta}_{ML})$ as shown in equation (3.32)

This MSE for each sensor is also compared to a threshold γ to create a ROC curve. It is important to note that this method does not account for the different distribution for possible errors in (3.32) and (3.33). This method determines the error regardless of distribution. As stated earlier, this method is not an optimal detector; it assumes several key factors that may hamper the detection ability. This will be seen in the results section as the performance degrades under specific circumstances.

In simulation, a matrix $\mathbf{e}$ is created using each sensor s and observation t. The MSE is taken across observations to give an error measurement for each sensor.

3.4 Identification, Classification, and Correction

Once the two algorithms are derived, they needed to be tested against metrics to determine their accuracy in detection and other important factors. The first of these is the probability of detection, P_d, and the probability of false alarm, P_f. The vectors of test statistics for the two methods, α_s and β_s are compared to a threshold γ. This threshold helps determine the ROC. In simulation the threshold slides between

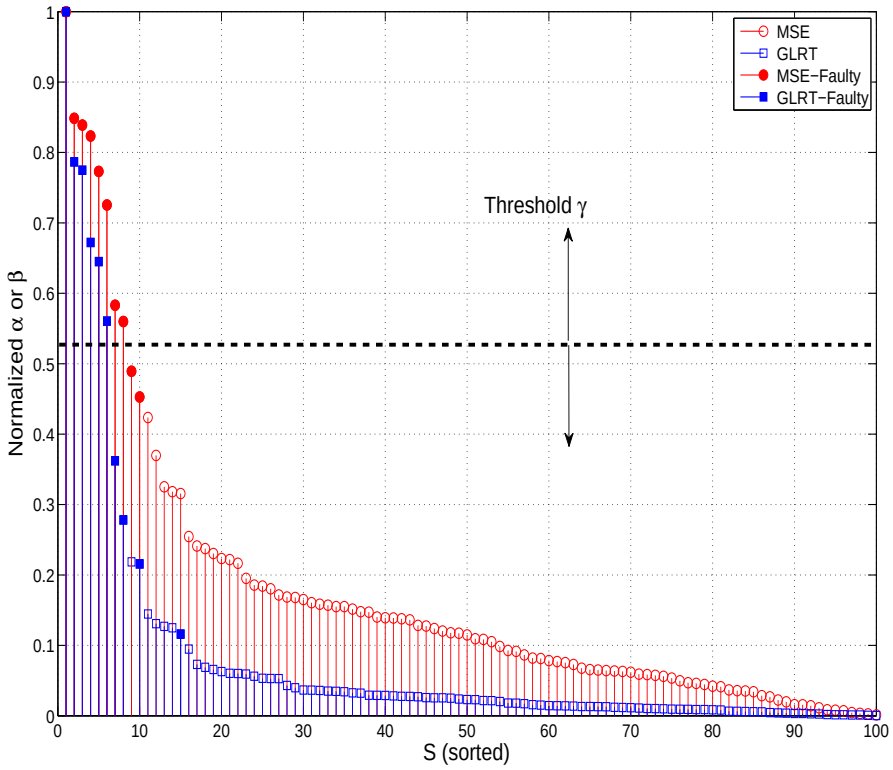

Figure 3.4: Example threshold test showing normalized α and β versus sorted sensor index.

$\max(\alpha_s)$ and $\min(\alpha_s)$ in equal steps. Figure 3.4 demonstrates an example threshold test with both normalized algorithms and the sensor index.

As Figure 3.4 shows, there are sensors that are below or above the threshold that are incorrectly categorized. As the threshold starts at the maximum value of the test statistic, all of the values are less than the threshold. As a result, they are all categorized as not broken. At this point the P_d is zero because there are no sensors classified as faulty, but there are also no false alarms, P_f. When the threshold is at the minimum value, all sensors are classified as faulty so $P_d = 1$, but the false alarm rate is also 1. The ROC values in between these points help classify how well the algorithm works.

3.4.1 Sensor Classification. As described in section 3.2.5, there are two different types of faulty sensor nodes. They do not have the same implications on the network. The first type, malfunctioning nodes, will affect the localization but have no intention of doing so. The second type, malicious nodes, have the intent of disrupting the functionality of the network. They may or may not be more difficult to detect depending on what the objective of the attack is. There are two possible methods of classifying a sensor into these types of problems.

The first method of classification is observing the variation in the RSS values reported by the sensor. After a sensor has been deemed a faulty node, the standard deviation gives insight on the type of faulty node. As previously stated, a malicious sensor is modeled to report the true RSS value with probability p_0 which can vary between 0 and 1. The attacker might want to be more or less likely to be detected, or may want more or less localization error. Since the sensor is reporting from two sets, the false value and true value, there is a deviation between the two. Taking advantage of this the user of the network can determine the standard deviation of the RSS values reported by each sensor.

$$\hat{\sigma}_s = \sqrt{\frac{1}{T-1}\sum_{t=1}^{T}(P_{s,t} - \bar{P}_s)} \tag{3.35}$$

where $\bar{P}_s$ is the sample mean of the RSS values for the s^{th} sensor.

Figure 3.5 shows both types of faulty sensors. To show differences in the two sensor types a histogram is built. The standard deviation, $\hat{\sigma}_s$ is taken across all observations of each sensor $s \in \{1 : S\}$. Each $\hat{\sigma}_s$ is then placed in the correct bin. In (a), the sensors are malicious with $p_0 = 0.50$, meaning they report the correct measured RSS 50% of the time. In (b), the sensors are malfunctioning, and always report the incorrect value V_0. A correctly functioning sensor would be the same as (b). A malfunctioning sensor has the same $\hat{\sigma}_s$, but may have a few sensors with a different mean RSS, V_0.

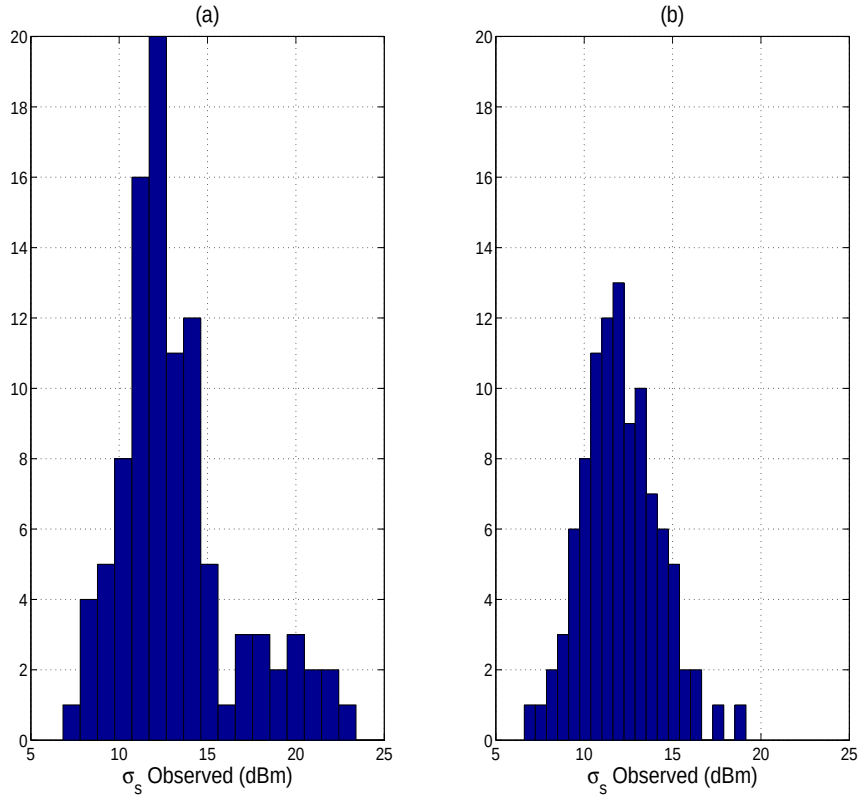

Figure 3.5: Comparison of standard deviation between a malicious sensor (a), and a malfunctioning sensor (b).

The histogram of the observed standard deviation, $\hat{\sigma}_s$, shows a clear distinction between the two types of faulty sensors. In the first case (a) the distribution of the measurements shows two distinct lobes. This is because the malicious sensors are reporting between two values giving two possible sets of standard deviations. In (b) the standard deviations are distributed as more Gaussian. When classifying sensors, only the nodes detected as faulty will be considered in determining the average standard deviation.

The second method is estimating the offset V_0. As shown in Equation (3.31), the offset can be estimated from the RSS values. In the case of a malfunctioning node, estimating the offset is helpful in determining how much the sensor is affecting the localization. From here, the user can determine the course of action; whether it

is removing the sensor from the localization, physically replacing or fixing the sensor, or accounting for the known offset. Combining these two techniques gives a way of determining the probable cause of the faulty sensor and information about the problem sensor.

3.5 Simulation Flow

Figure 3.7 is a block diagram of the simulation flow. With each new simulation a variety of parameters need to be determined. The first of these is whether the faulty sensor is malicious or malfunctioning. After this has been determined, the number of faulty sensors is chosen. This is done as a percentage of the total number of sensors S.

When S is chosen, the locations of the sensors are determined. For simulations in this research, the sensors are distributed randomly in a specified space. The search space is 10m in x and 100m in y. The sensors can also be in a 5m vertical space. The space replicates sensors being randomly placed along a fence or building. After the sensor locations are known, the time steps begin. At each time step the transmitter is moved a specified amount. With each new transmitter location, new RSS data is created, and MLE of location is computed. After all time steps, T, the data is used to calculate the algorithm values α_S and β_S. These values are used to complete a threshold test to determine P_f and P_d. After one complete time step, the process begins again. After all independent trials, P_f and P_d are averaged to get a final ROC curve for that simulation. Figure 3.6 is a list of the process described here, showing the process of a simulation. Figure 3.7 is a block diagram showing important simulation blocks.

1. Choose Parameters
2. K Independent trials
   (a) Choose sensor locations
   (b) Choose which sensors are faulty
   (c) T Time steps
      i. Generate RSS data
      ii. Calculate location - true and faulty sensor data
      iii. Move transmitter
   (d) Create power map and find expected RSS
   (e) Find α_S and β_s
   (f) Complete threshold test
3. Average P_f and P_d over K trials

Figure 3.6: Simulation flow with steps shown.

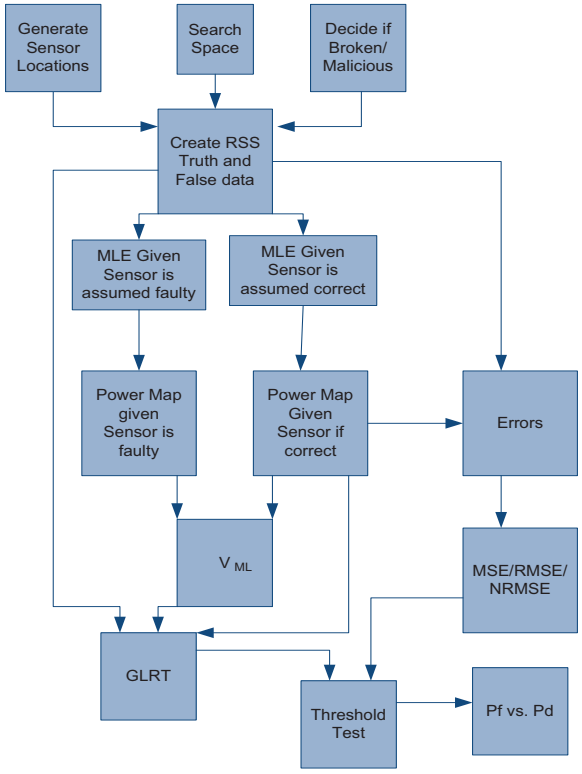

Figure 3.7: Flow diagram of how the simulation is run.

IV. Results and Analysis

This chapter presents the results from the simulations described previously. The results will be analyzed and discussed in reference to the objectives of the research.

4.1 Experimental Parameters

To keep track of the parameters being tested, and values of the parameters, Table 4.1 was developed. Several important factors need to be tested to show that the algorithm can work in a sensor network. The simulations used to gain insight on the system are shown. Each parameter value is given, and in some cases the simulation is run to compare several different parameter values. The figure that the simulations correspond with are also given.

Table 4.1: Table of experimental parameters

Test #	S	# Malic	# Malf	p_0	V_0	σ_e	Figure
1	50,100, 200	0	20	0	-25	12	4.1
2(a)	100	0	20	0	-25	6	4.2(a)
2(b)	100	20	0	0.75	-25	6	4.2(b)
3	100	0	25	0	-40	2,6,12,25	4.3
4	100	0	20	0	-40,-30,-20, -10,0,10	4,8,12	4.4
5	100	20	0	0,0.25,0.50, 0.75,1.0	-25	12	4.5
6(a)	100	5,25,50, 75,95	0	0	-25	12	4.7(a)
6(b)	100	0	5,25,50, 75,95	0.25	-25	12	4.7(b)
7	100	0	20	0	-40,-30,-20, -10,0,10	12	4.9
7	100	20	0	0.50	-40,-30,-20, -10,0,10	12	4.9
8	100	20	0	0.25,0.50, 0.75	-40,-30,-20, -10,0,10	12	4.10

Where:

- S: Number of sensors in SN
- # Malic: Number of malicious sensors simulated
- # Malf: Number of malfunctioning sensors simulated
- p_0: Probability of reporting true RSS measurements in malicious sensors
- V_0: Offset RSS value reported in malfunctioning and malicious sensors (dBm)
- σ_e: Simulated noise standard deviation (dB)

4.2 Algorithm Detection Capabilities

This section will cover how well each algorithm, the GLRT and MSE algorithms are able to detect faulty sensors based on three important factors. The first factor is the size of the sensor network being used. The second factor is the RSS value offset reported by the faulty sensor, and the final factor is the noise variance in the environment. The algorithms will also be compared against each other to determine if one has a distinct advantage in detection over the other.

There are some aspects of the simulation that are necessary for simulation, but aren't considered when evaluating the performance. The first of these variables is η, the path loss exponent. This can range from 2 in free space, to 5 in dense urban environments [7]. The second variable is the transmitted power, P_0. This is the original transmitted power from the transmitter. This was not tested because the model did not include the accuracy of the RSS sensors. Therefore, all values of RSS can be accurately measured by the system, and the original transmission power does not affect the detection capabilities. One distinction is that the RSS value offset V_0 can be varied to be in or out of the expected RSS value based on P_0.

The first important consideration is how the algorithms are able to detect a faulty sensor in different sized sensor networks. In Figure 4.1, sensor networks of different sizes are used, with 20% malfunctioning sensors and $V_0 = -25$ dBm.

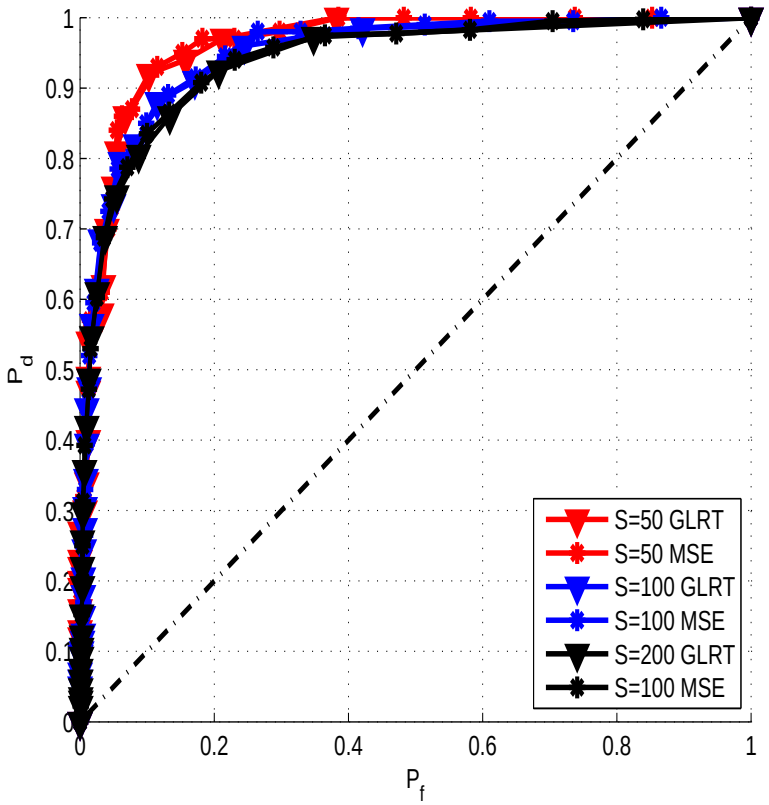

Figure 4.1: ROC for varying sizes of sensor networks(S). In simulation 20% of sensors are malfunctioning with an offset of $V_0 = -25$ dBm.

The ROC curves for the three sizes of sensor networks are similar. The GLRT and MSE algorithms both perform similarly in the sizes of sensor networks simulated. The difference is ROC changes for different sized networks. As the number of sensors increases the performance of both algorithms decreases. The highest ROC curve is for 50 sensors, while the ROC curve for 200 sensors is the lowest.

The offset value, V_0 might also have an impact on which algorithm is more able to detect faulty sensors. To determine this, a simulation is run with several potential offsets for malicious and malfunctioning sensors. Figure 4.2 shows two separate cases. In (a) the offset value V_0 is set to three different values, and 20% of sensors are simulated to be malfunctioning. In (b) 20% of sensors are simulated to be malicious with $p_0 = 0.75$.

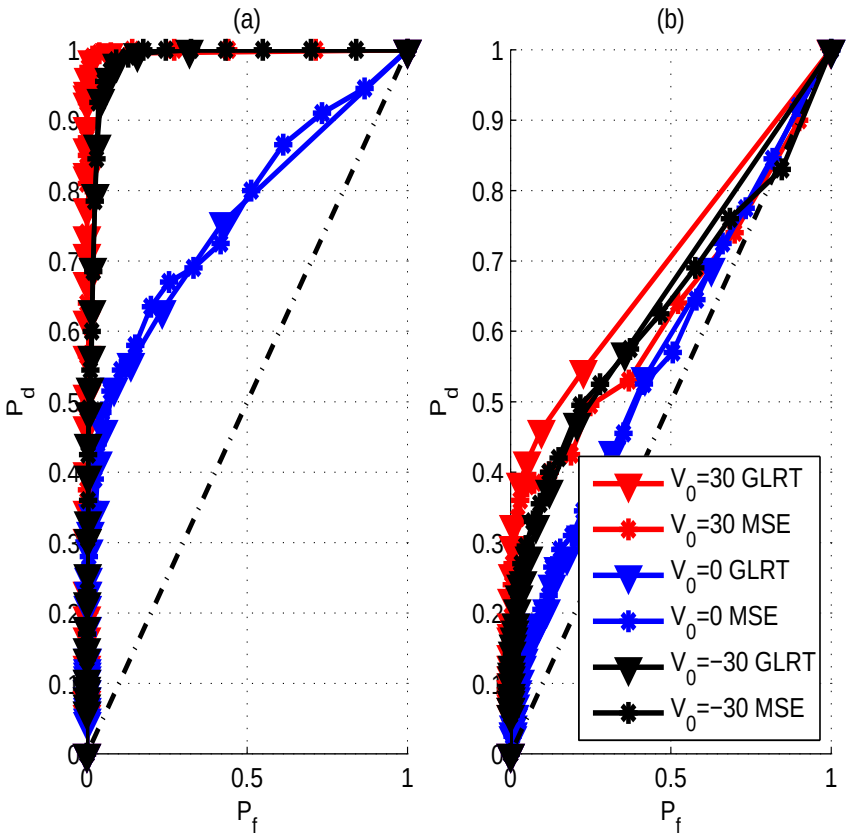

Figure 4.2: ROC for varying V_0. In (a) the sensors are simulated to be 20% malfunctioning, and in (b) 20% of sensors are simulated to be malicious with $p_0 = 0.75$. In both cases $S = 100$ sensors are used.

Looking at the ROC curves in Figure 4.2 (a), both algorithms have similar performance for all values of V_0. The difference in detection ability is with the value of V_0 the sensor is using. The transmitter has an initial transmitted power of 20 dbm. When the signal reaches the sensors typical RSS values are between -5 and -20 dBm. When the sensor reports that a received value is outside that range (30 and -30 dBm), the algorithms are more likely to be able to detect these large difference in expected and actual RSS. This is reflected by how both ROC curves for these values perform much better than the $V_0 = 0$ case. In (b) the sensors are simulated as malicious, and report the actual measured RSS value 75% of the time. As a result the V_0 value does not have as much of an impact because it is used only one quarter of the time. All the ROC curves in (b) are lower than (a) for the same V_0 values. There is also less

of a difference in detection rates between the three values of V_0. The GLRT performs 0.05% better on average for $V_0 = 30$.

The standard deviation of the noise also affects the ROC for both types of sensors. Changing the simulated standard deviation σ_e between several values allows analysis of algorithm comparison and performance. In Figure 4.3 σ_e is varied between 2 and 25 dB. A sensor network with $S = 100$ nodes is used, with an offset of $V_0 = -40$ dBm. The faulty sensor nodes are simulated to be malfunctioning with 25% of total being malfunctioning.

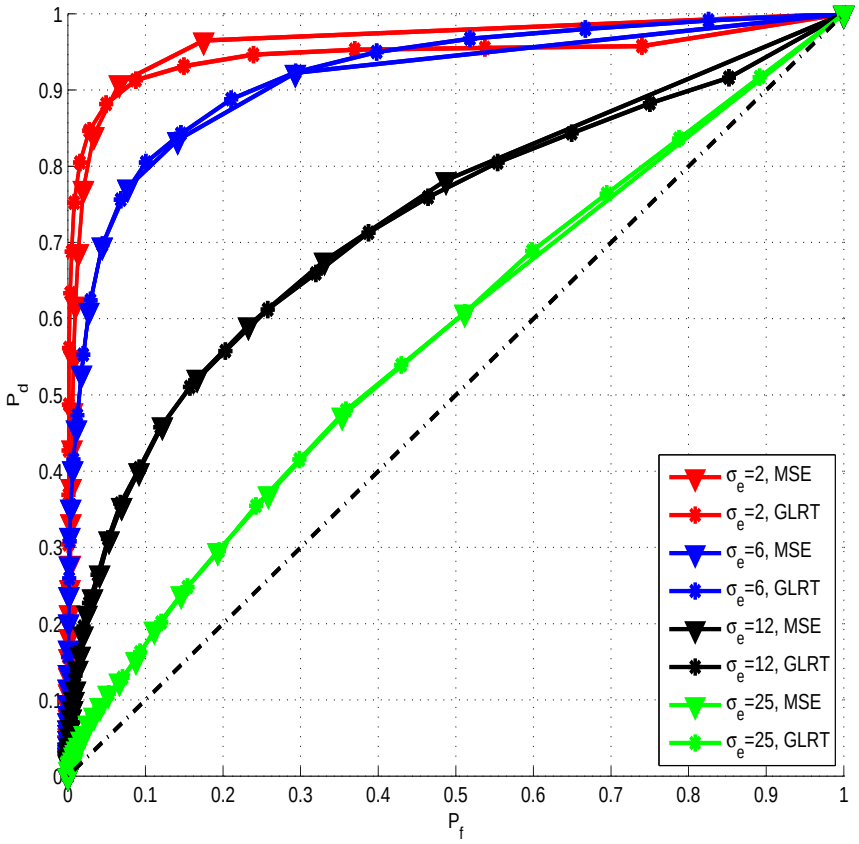

Figure 4.3: ROC for varying σ_e values. 25% of $S = 100$ sensors are simulated to be malfunctioning, with an offset $V_0 = -40$ dBm.

As Figure 4.3 shows, the value of σ_e affects the detection ability of both algorithms. The GLRT and the MSE perform similarly for all σ_e values. As in most cases simulated, the algorithms perform very similarly. The advantages besides detection

rate of each algorithm will be discussed later. The lower the noise variance, the better both methods perform. As the noise is increases to 25 dB, the detection rate becomes close to a guess, with the detection rate and false alarm rate being equal. Noise variance rates of 4 -12 dB can be found in many potential environments [7]. In that range, with $P_f = 0.20$ the probability of detection can range from 0.55 to 0.95. Depending on the environment there can be up to 40% difference in ability to detect a faulty sensor. This demonstrates the susceptibility of RSS measurements to noise.

Figure 4.4 shows the detection rate for several different values of σ_e and V_0. This demonstrates the detection ability for several different possible configurations. The detection rate shown is for a false alarm rate of $P_f = 0.15$.

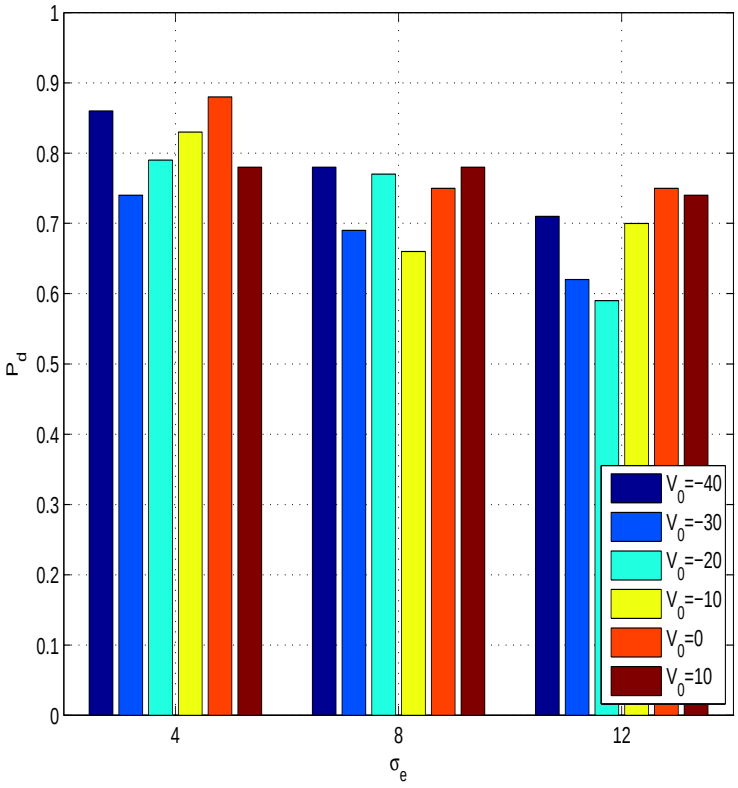

Figure 4.4: Plot of detection rate for various values of V_0 and σ_e. The GLRT algorithm is used, along with 20% malfunctioning sensors out of a total of $S = 100$. The detection rate P_d, is found with $P_f = 0.15$ in all cases.

The first component of information shown is how the detection rate decreases as the standard deviation of the noise increases. The detection rate also has a loose correlation with the false RSS measurement reported, V_0. The values that are farther outside of the expected range have higher detection rates than those in the expected range. In simulations using $P_0 = 20$dBm, and the plausable distances from the transmitter to receivers in the sensor network, RSS measurements range from -5 to -20 dBm.

4.3 Detection of Malicious Sensors

As described in Chapter III, a malicious sensor can have three adjustable properties. The first property is p_0, how often the sensor correctly reports the true measured RSS. The attack on the network would have to weigh the trade-off of this parameter for two reasons: the probability that it will be detected, and the ability to affect localization error by the network. Table 4.2 demonstrates this concept. Simulating 25% malicious sensors in the network, the localization error D_e was found for varying probabilities of p_0.

Table 4.2: Error (D_e) for varying probability of reporting correct value (p_0)

Malicious Probability (p_0)	0.0	0.25	0.50	0.75	1.0
D_e (meters)	20.8	17.0	10.9	5.8	5.4

$$D_e = \frac{1}{T}\sum_{t=1}^{T} ||\widehat{\theta}_t - \theta_t|| \tag{4.1}$$

where D_e is computed for each independent trial K, and averaged. $[x_{0,t}, y_{0,t}, z_{0,t}] = \theta_t$ is the truth location at observation t, and $[x_{s,t}, y_{s,t}, z_{s,t}] = \widehat{\theta}_t$ is the MLE. This metric, a Mean Absolute Error (MAE), uses truth data (the location of the transmitter) which in practice the processing center will not have. This value is a good metric in determining the ability to detect the sensor node verses the error in localization it causes for specific conditions. D_e does not account for any detection and removal

of any sensors in the network. This metric measures strictly the error caused by the faulty sensor.

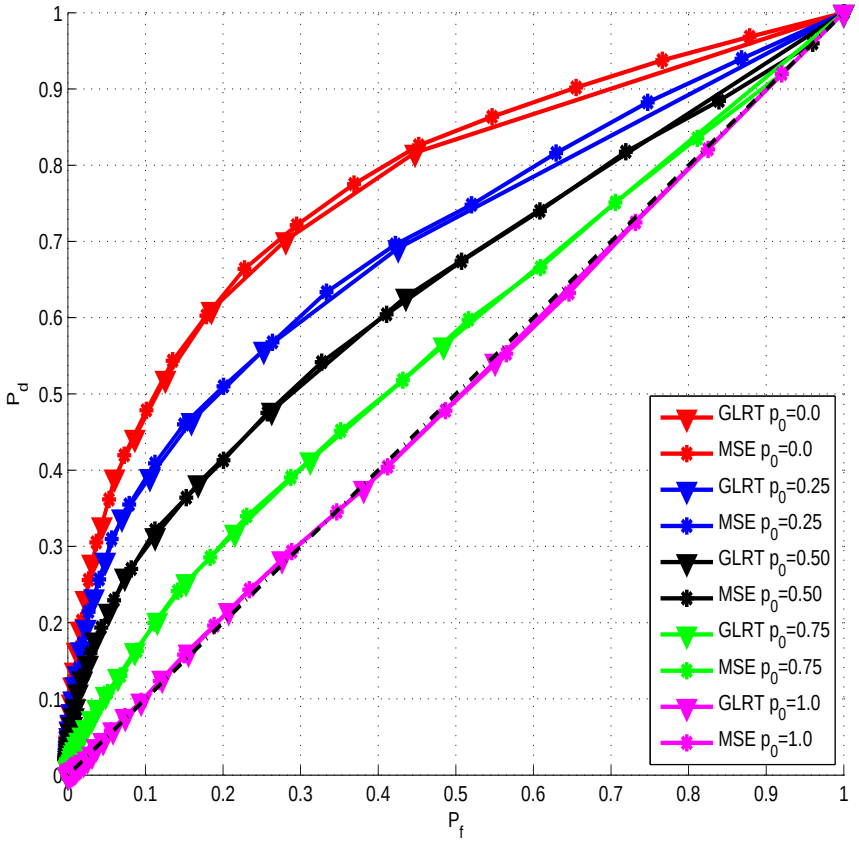

Figure 4.5: ROC for varying values of p_0. The simulation uses $S = 100$ sensors 20% of which are malicious and an offset of $V_0 = -25$ dBm.

Figure 4.5 is a ROC plot of varying p_0 values for malicious sensors in a network. The first detail of note is that as p_0 is increased, the probability of detection for the same probability of false alarm drops. As the attacker reports the true RSS value more, the ability to detect the sensor decreases. In Table 4.3 this is shown. If the user of the network wants a Constant False Alarm Rate (CFAR) of $P_f = 0.20$, then the probability of detection can be compared for each p_0. Values of detection for a given CFAR, $P_d|P_f$ are also shown in the table. A CFAR would be used when false positive detections have a significant impact. For example, if the cost to go out and replace a sensor that is falsely classified as a malicious sensor is high, the user of the network might want a lower CFAR to decrease cost. As a result of this, the probability of

detection goes down, so it is more likely that there will be localization error on the network.

Table 4.3: Error (D_e) and metrics for varying probability of reporting correct value (p_0)

Malicious Probability (p_0)	0.0	0.25	0.50	0.75	1.0
D_e (meters)	20.8	17.0	10.9	5.8	5.4
$P_d\|_{P_f=0.2}$	0.62	0.51	0.41	0.30	0.20
$P_d\|_{P_f=0.5}$	0.85	0.73	0.63	0.57	0.48
$G_{P_f=0.2}$ (meters)	8.02	8.43	6.51	4.12	4.36
$G_{P_f=0.5}$ (meters)	3.52	4.94	4.34	2.77	3.04

To quantify the relationship between localization error and probability of detection a new metric is used. The expected localization error in meters for a given false alarm rate, $G(P_f)$, is this metric.

$$G(P_f) = D_e(1 - P_d|_{P_f}) + D_e|_m(P_d|_{P_f}) \tag{4.2}$$

where $P_d|_{P_f}$ is the detection rate at the specified false alarm rate, P_f, and D_e is the localization error at the specified p_0 value. $D_e|_m$ is the localization error when the sensors have $p_0 = 1.0$, meaning 100% of the sensors are malicious.

$G(P_f)$ gives a number for each p_0 value that combines the error and detection probability. In Table 4.3, $G(P_f)$ is shown for two particular CFAR. For each P_f value chosen, the probability of detection, P_d, that corresponds is found. The lower the CFAR required by the user of the network, the higher $G(P_f)$ is. This means that the attacker is able to more successfully attack the network without being detected. When P_f is increased allowing for more false alarms, the attack will be less likely to have the same effect on the network for the same P_d. Figure 4.6 shows a plot of $G(P_f)$ for different values of P_f.

Figure 4.6 shows that in general, $G(P_f)$ decreases as p_0 increases, or as the true value is reported more. As p_0 approaches 1, $G(P_f)$ begins to increase again. At $p_0 = 1.0$, the true RSS value is always reported, therefore the detection becomes a

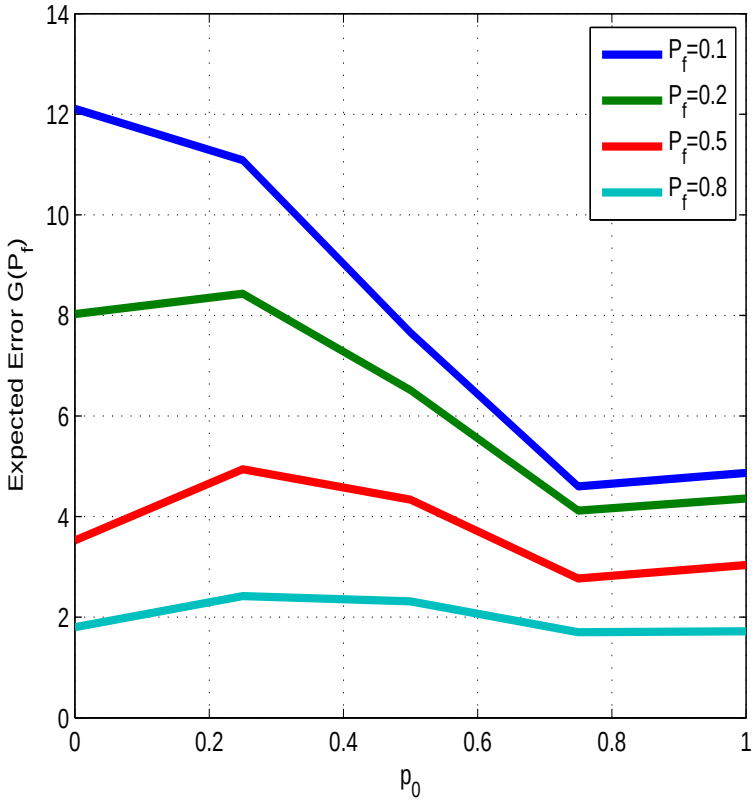

Figure 4.6: Plot of $G(P_f)$ for varying p_0. Plots are shown for several values of P_f. Data is simulated with $S = 100$ sensors 20% of which are malicious and $V_0 = -25$ dBm.

guess. At this point D_e still exists, but is not affected by the malicious sensors. When $p_0 = 1.0$ there is still error in the localization due to noise and multi-path. This metric $G(P_f)$ does not have much meaning because the attacker no longer has any control of the errors in the localization. The attacker would want to determine the best trade-off between detection capability and localization error, and this plot gives a metric to determine that.

The metric $G(P_f)$ does not alter characteristics as a function of P_f. All four plots have a similar shape and give approximately equal decrease with p_0. This is important because the choice of a threshold for the test does not affect how $G(P_f)$ more in some cases. The values of $G(P_f)$ are different for each P_f, but decrease at a

constant rate. The user of the network should make a choice for the threshold based on costs associated with false alarms.

4.4 Number of Faulty Sensors

The number of faulty sensors in the network also has an effect on the detection ability of the network. The control center does not have knowledge of how many sensors in the network are faulty. The algorithms described thus far are based on the fact that a small percentage of the sensors are faulty.

The GLRT is built to optimally detect a single faulty sensor in the network. As more are added, the algorithm is no longer ideal. Similarly, the MSE algorithm is based on creating a power map. This power map uses all sensors to locate the transmitter, and then recreate what the RF power is like in the environment. Any outliers from this power map are considered faulty. If however, a larger percentage of the sensors are faulty, the correctly functioning sensors will appear to the decision center as the outlier. This will allow the malicious user to completely control the localization ability of the network.

This type of attack can cause huge problems with the sensor network. To demonstrate how the percentage of faulty sensors in the network can affect the efficiency of the algorithms, a simulation is done to determine the detection capabilities. In this simulation, two cases were considered. In the first case, the network has varied numbers of malfunctioning sensors (a). The second case is simulated to have varying numbers of malicious sensors (b). In (b), the probability to report actual RSS measurements from the malicious sensors is $p_0 = 0.25$. In both cases values of $\sigma_e = 12$ dB and $V_0 = -25$ dBm are used.

Figure 4.7 shows ROC curves for both types of faulty nodes. In (a), the malfunctioning case, there are five simulations run. Each simulation runs with a different percentage of the total sensors considered malfunctioning. In the 5% case, the ROC is the highest, because the small number of total sensors allows for the majority of

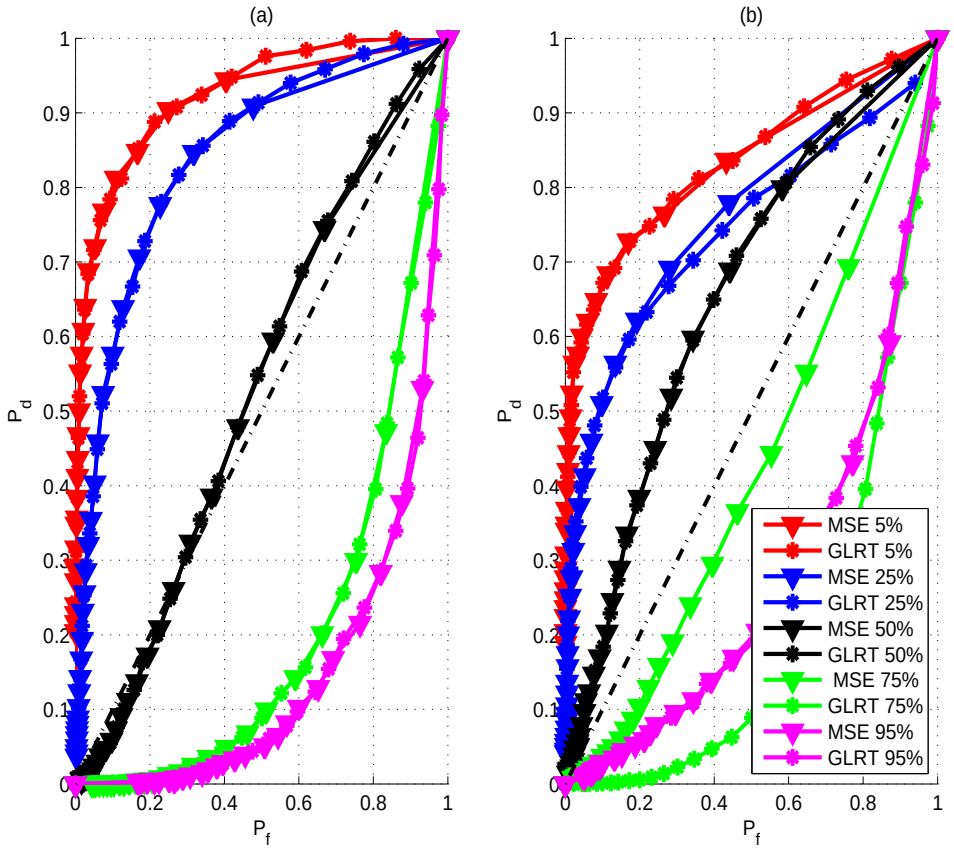

Figure 4.7: ROC plot for varying number of faulty sensors. In (a) the sensors are malfunctioning, and in (b) the sensors are malicious with $p_0 = 0.25$. Both are simulated using $S = 100$ sensors with $\sigma_e = 12$ dB and $V_0 = -25$ dBm.

other sensors to detect it. At 25% malfunctioning sensors the algorithms are still able to detect the faulty sensors adequately. The detection rate at $P_f = 0.20$ is approximately 10% lower. When the malfunctioning sensors are half of the total the detection becomes much more difficult. It is essentially a guess which sensors are the correctly functioning ones, because the detection rate and false alarm rate are equal. When more than half of the sensor network is malfunctioning, it presents a major challenge to the localization. Now the malfunctioning sensors are the majority, and cause the correctly working sensors to appear out of place. In the MSE algorithm, the power map is now largely based on false readings, and the correct RSS values appear to be the outliers. As a result, the detection now performs less effectively than a guess.

In (b) the same five simulations are run, but with the sensors modeled as malicious. The first observation to note is that the detection rate for the same false alarm rate is lower for malicious nodes, approximately 15% at $P_f = 0.20$. This is true when the malicious nodes are less than half of the total sensor network. At 50% malicious nodes, the algorithms are also more able to detect these nodes when compared to malfunctioning nodes. While malfunctioning nodes have equal P_f and P_d at 50%, malicious are able to obtain $P_d = 0.60$ at $P_f = 0.40$ with the same percentage of faulty nodes.

In the case of (a), the malfunctioning nodes in both algorithms perform equally on all percentages of faulty sensors. When the faulty sensors are modeled as malicious, there is some separation between the two methods. For 75% malicious sensors, the GLRT performs significantly better than the MSE. At $P_f = 0.20$ the GLRT has 25% better detection than the MSE. As the malicious percentage approaches 100%, the algorithms again have the same performance.

Similar to the probability of reporting a true value (p_0), developing a metric to understand the relationship between errors in the localization and the total percentage of faulty sensors can provide insight into the system. Table 4.4 gives the localization error for the simulations in 4.7. The G_{P_f} metric is used to obtain a number to relate the error and detection rate.

Table 4.4: Error (D_e) and metrics for varying percentages of faulty sensors

Percent Faulty (%)	5	25	50	75	95
D_e Malicious (meters)	7.7	20.6	38.5	71.4	63.0
D_e Broken (meters)	5.2	29.8	58.2	105.4	121.7
$G(P_f = 0.20)$ (meters)	2.19	8.27	23.50	64.27	59.85
$G(P_f = 0.60)$ (meters)	1.18	3.88	7.87	37.29	49.81

Table 4.4 shows that as the percentage of faulty sensors in a network increases, the localization error increases as well. This is to be expected, because if a larger number of sensors aren't performing and reporting the actual RSS, the localization

will be affected. G_{P_f} shows that for the same false alarm rate, the more effective the error is, taking into account the probability of detection.

Figure 4.8 is a plot of the $G(P_f)$ metric for varying values of P_f. This plot shows how effective changing the number of faulty sensors can be in compromising the localization functionality of the network.

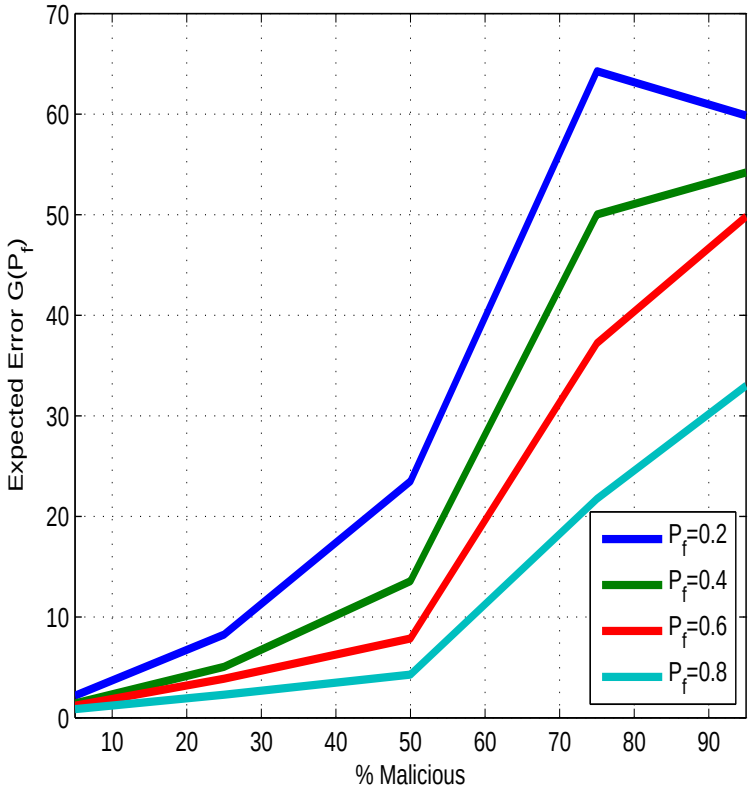

Figure 4.8: Plots of the $G(P_f)$ metric for varying percentages of malicious sensors with $p_0 = 0.25$. This data was simulated with $S = 100$ sensors with $\sigma_e = 12$ dB and $V_0 = -25dBm$.

Figure 4.8 shows that having knowledge of the possible attack can help to improve the detection of malicious nodes in the network. If the user has a low CFAR due to the high operation cost of false alarms, the attack can be more effective with the same percentage of malicious nodes. For low percentages of malicious nodes, all four CFAR are similar in the effectiveness metric $G(P_f)$. As the malicious percentage increases, the lower CFAR requirements have a much larger $G(P_f)$ value. For exam-

ple, if the user specifies a CFAR of $P_f = 0.20$ and the attacker is able to maliciously attack 60% of the sensors in the network, $G(P_f)$ is 276% more effective than if the user specified a CFAR of $P_f = 0.40$. On the other hand $P_f = 0.40$ is only 144% more effective than a $P_f = 0.60$.

4.5 Classification

The analysis thus far has been involved with how well each method is able to handle different conditions. This section covers the next step after the sensors have already been determined. The algorithms will mark each sensor as either faulty or working correctly. This may not be sufficient information to the user of the network.

As previously discussed, the two different sensor models influence how the problem may be dealt with. If a sensor is malicious, it is intentionally misleading the localization. This could compromise the operation of the network and allow someone with a RF transmitter to get inside the gate without knowledge. On the other hand, a malfunctioning sensor node affects the localization without the intent of doing so.

After the RSS data has been collected and the algorithms have made decisions about all of the sensors in the network, the sensors are classified. The first possibility considered is whether they are malicious or not. As stated in the sensor model, a malicious sensor reports between a false value V_0 and the actual RSS value with probability p_0. The more frequently the attacker reports a false value, the more they are able to affect the localization.

Figure 4.9 shows a plot of the measured standard deviation versus the two types of faulty sensors. This data was generated with $\sigma_e = 12$. The GLRT algorithm is used to detect faulty sensor nodes, and the sensors are noted. The average observed standard deviation is then plotted for each case. The two cases are malfunction and malicious. In each simulation multiple RSS value offsets, V_0 are used. The dotted line represents the expected observed standard deviation of 12. The average observed noise standard deviations, $\widehat{\sigma}_e$ can be found by:

$$\widehat{\sigma}_e = \sqrt{\frac{1}{T-1}\sum_{t=1}^{T}(P_{s,t}^{\mathrm{F}} - \bar{P}_s^{\mathrm{F}})} \tag{4.3}$$

where $P_{s,t}^{\mathrm{F}}$ is the t^{th} observation of RSS by the s^{th} sensor detected as faulty, and $\bar{P}_s^{\mathrm{F}}$ is the sample mean for s^{th} sensor detected as faulty.

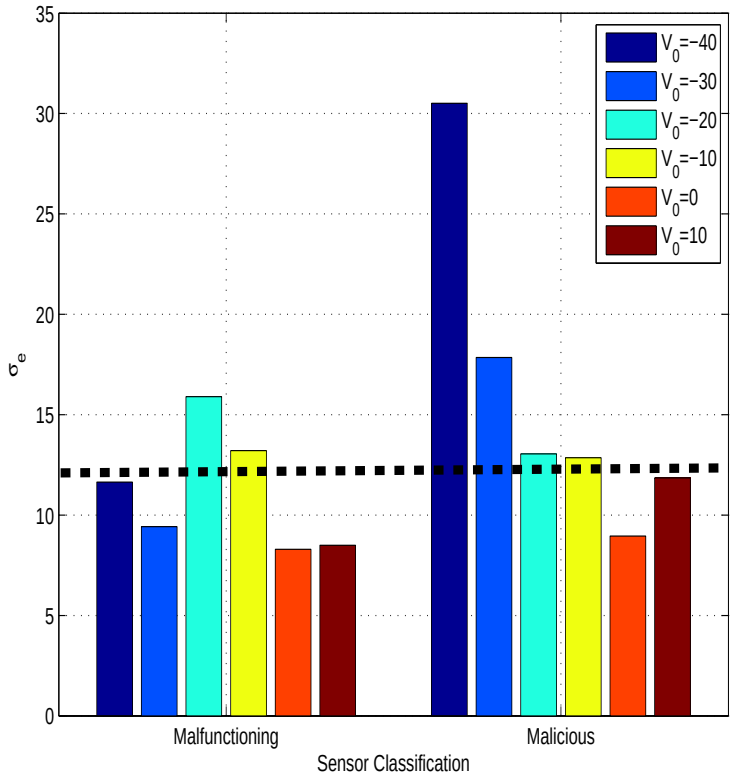

Figure 4.9: Plot of the classification for malicious and malfunctioning sensors. Using $P_f = 0.15$ the average noise variance, σ_e of sensors classified as faulty are shown. The classification is tested for several different offset values V_0. Values of $p_0 = 0.50$ and $\sigma_e = 12$ were used, and 20% of sensors are faulty. A line is used to demonstrate the simulated noise variance.

The first observation is that the standard deviation is lower on average in the malfunctioning sensor simulation. This is because in this case the sensors, if the sensors are malfunctioning, they are consistently reporting the same value. In the malicious sensors, the standard deviation is higher because the faulty sensors now

report two different numbers, which may be far apart in value. The line that depicts simulated σ_e is the "cutoff" for what might be considered malicious sensors. If the observed standard deviation is much higher than the cutoff, the user can conclude that the faulty sensors are malicious. If the values are close to expected, the sensors can be classified as malfunctioning.

The value of V_0 is an important factor in the ability to classify sensors for two reasons. As shown previously, V_0 can affect the ability of the GLRT algorithm to detect faulty sensors. If correctly working sensors are included in the average standard deviation, the values will likely be lowered, causing sensors to seem more like malfunctioning ones. The second consideration is that the larger separation between V_0 and the expected RSS values, the larger the standard deviation will be. At $V_0 = -40$, there are very few RSS measurements close to this value. However, with a $V_0 = -20$, the difference between reporting true information and false information is small.

Figure 4.10 is a plot of observed noise standard deviation versus the probability to report true values for a malicious sensor. For this data all faulty sensors are considered malicious. The goal is to determine the ability to classify for various values of V_0 and p_0.

Once again $P_f = 0.15$ is used with the GLRT algorithm to detect the faulty nodes, in this case malicious. The average standard deviation of these detected nodes is shown. In the first case $p_0 = 0.25$ is used, and the standard deviation is calculated for five V_0 values. For V_0 outside the expected range, the σ_e observed are higher than the values in the expected range. This is the case for all three p_0 simulations.

As expected, the standard deviation is highest when $p_0 = 0.50$. This is because the RSS measurement reported is changed with the highest probability. For lower or higher p_0, one of the true or false RSS measurements is being reported with a higher probability. The closer to p_0 the attacker is using, the higher the probability of correctly classifying the sensor as malicious.

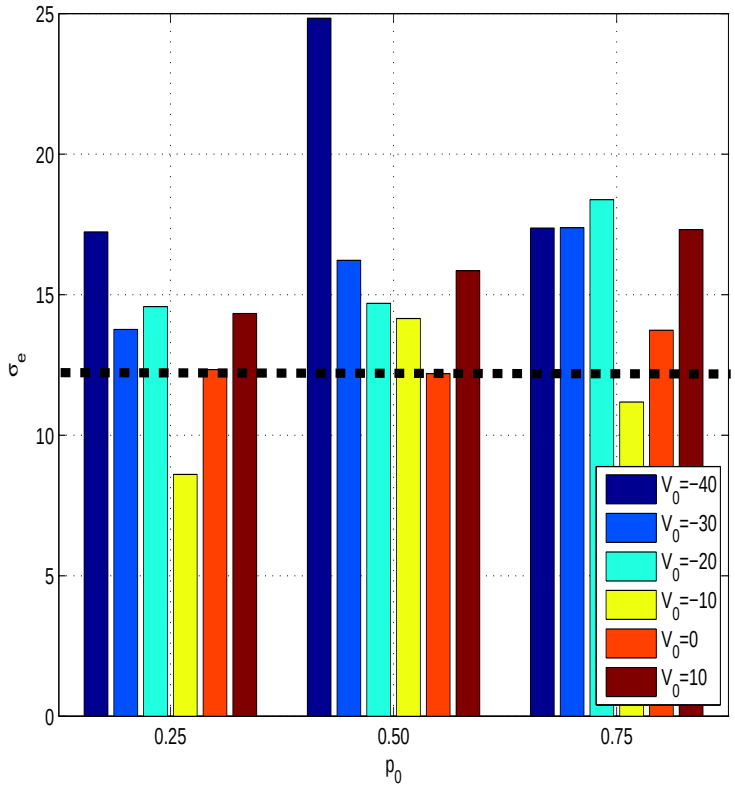

Figure 4.10: Plot of malicious sensors average noise variance. The value is shown for three cases of p_0, and varying levels of RSS offset V_0. A simulated noise variance of $\sigma_e = 12$ is used, and 20% of sensors are faulty. The line shows the simulated noise value.

4.6 Conclusions

This section summarizes the results and provides an overview of how they meet the objectives of this research. The first key component to consider is the feasibility of this concept as a whole. It is necessary to determine if it is a practical thing to use a system like this to accomplish faulty node detection. The algorithms developed are the other key objective of the research. The two algorithms will be compared in their performance in the simulated conditions.

4.6.1 Malicious Node Detection Feasibility. The system model and algorithms described here are able to detect malicious node behavior in a network just via

physical layer data alone. One of the objectives of this research is to determine the feasibility of this system to determine trust in sensors, and detect malicious attacks on them.

The first criteria is that the detection utilizes only observations of physical layer data. This means that the information used to make decisions is the RSS measurements collected from the sensors on the node. This data is packaged and sent back to the processing center, so it involves some layer of network interaction. The only information considered are the physical measurements taken by the sensors. This allows for low complexity and low power sensors to be used to assess the performance and functionality of this system, without considering network layer occurrences. These could include missing packets, misrouting information, and other techniques.

The functioning of this system depends on sevearl key assumptions. These will be addressed to show that the assumptions are practical.

- *Transmitter power known*: The original purpose of the network is to locate a transmitter of interest. It is safe to assume the user would know what the device being operated is. For example, the object might be a cell phone with a known average transmission power.
- *Omni-directional antenna*: Many of the devices that might be tracked have small or omni-directional antennas. At the distances used in this research, it is assumed that they all could be accurately modeled as omni-directional.
- *Log-normal fading model*: This model has been developed and is used for many different environments and purposes. It has also been tested in physical systems.
- *Sensor models*: The sensor models developed detail two potential ways that sensors might not be correctly working. There may be other methods of attacking a sensor that affect the ability to communicate, but this research was limited to attacks on the physical data collection and reporting.

- *Environmental factors*: The environmental factors including the path loss exponent and the noise variance used were taken from typical measurements of these values.
- *RSS measurement errors*: The error in RSS measurement will change the overall detection rate of the system, but not the comparisons between the two algorithms. These errors would be a next step to include in future research.

Under these assumptions, the system is able to detect both malicious and malfunctioning sensors. This research began without a directive of achieving a specific detection rate for a certain false alarm rate. As a result, the feasibility is not able to be concretely determined from a set of data. The feasibility of the system depends instead on a large variety of details involved with the system.

The first area tested is the detection ability in varying sized sensor networks. The sizes tested ranged from 50-200 sensors. There are slight variations in the detection ability of the network, around 5%, but all of the sizes were able to detect more than 90% of the faulty sensor nodes with $P_f = 0.20$. The number of the sensors has a greater effect on the localization accuracy. As a result, an important consideration for number of sensors is not the detection ability, but the accuracy that is needed for detection.

The system is also viable in varying levels of the RSS measurement offset, V_0. This implies that the system could detect varying levels of error due to battery life in a malfunctioning sensor, or false RSS values in malicious sensors. In the malfunctioning nodes, the detection workes extremely well in conditions where the offset is significantly outside the expected RSS range. When V_0 is inside the expected range, the detection capabilities decrease 30% at $P_f = 0.25$, but is still within an acceptable range. In malicious nodes, the detection rates are also acceptable within the expected range, but are lower than the malfunctioning sensor case. The difference here depends on the p_0 used. In the simulation, comparing malicious and malfunctioning in

V_0, $p_0 = 0.75$ is used. Values closer to one make the malicious sensor more difficult to detect.

In varying levels of noise, the physical layer data is also a feasible method of detection. In the realistic range of noise simulated, 4-12 dB, the detection rates for a $P_f = 0.20$ were between 0.55-0.90. The final factor that was tested is the probability of malicious sensors to report the correctly measured RSS value.

Table 4.5 outlines the ranges of values where the system will achieve viable detection rates, accomplishing the goal of the research.

Table 4.5: Comparison of the feasibility based on detection rates for simulations. x refers to an area where the algorithm has an advantage.

System Parameter	Feasible Value
p_0 (%)	0 - 50
V_0 (dBm)	Any
σ_e (dB)	4 - 12
S	50 - 200

4.6.2 Algorithm Comparison. A key objective is to develop and evaluate algorithms that are able to detect malicious sensor nodes. Table 4.6 gives an overall qualitative comparison of the two techniques in a variety of categories.

Table 4.6: Qualitative comparison of the two developed algorithms, GLRT and MSE.

System Characteristic	GLRT	MSE
Sensor Network Size	x	x
Algorithm Complexity		x
Varying % Faulty	x	
Noise Variance	x	x
V_0 Estimate	x	
Malicious Attacks	x	
Power Map Creation		x
Knowledge of Distribution		x
Improve with Knowledge	x	

Both algorithms have a few benefits that might be useful depending on the situation. The GLRT and MSE both perform equally well in all sensor size networks

and noise standard deviations that are simulated. If the user of the network is looking for extra information about the situation, there are benefits to each. As part of the process of detection, the MSE builds a power map of the area of interest. If the user is interested in knowing that information as well, the MSE might be a good option. Similarly if the value of the RSS measurement offset V_0 is of interest, the GLRT method estimates the value. This can be used to help classify the sensors severity of damage.

The GLRT method performs better in one area. For large percentages of the network under attack, the GLRT has better detection ability. It has equal P_f and P_d rates, but is an improvement over the MSE.

The MSE has the benefit of reduced complexity. The algorithm does not depend on a hypothesis specific to the sensor model. If the malicious attack on the network has a different model, the MSE will not need to change. On the other hand, the GLRT is built on the specific sensor model and hypothesis.

The GLRT algorithm has the ability to improve with more knowledge about the system. While the MSE can only improve with knowledge of the environmental factors such as σ_e, η, and P_0, the GLRT can improve with knowledge of these factors as well as V_0, p_0, and the number of faulty sensors. This means that if the sensor is more complex and other factors can be estimated the accuracy of the GLRT would improve.

Overall, both methods are feasible to use in a sensor network. As described in this research, they are able to detect reasonable amounts of faulty sensors in a RF localization network. This is done using only physical layer data that is already being collected for the purpose of the network. No extra data needs to be collected, and no extra hardware is needed. The GLRT algorithm is more complex and tailored to specific models and conditions, while the MSE performs similarly in most simulations, but can be limited in the detection ability due to the lack of usage of important information.

V. Recommendations and Future Work

This chapter will discuss the contributions of this research, and future research possibilities in this area of study. Current research discussed in Chapter II, along with work the research outlined in Chapter III and IV will be the basis for future research.

5.1 Summary

This research began with the goal of determining trust in a CN by using only physical layer data. The framework consists of a sensor network in place to localize a transmitter of interest. The background information for this thesis is from several different disciplines. These included localization techniques and applications, RSS based models and applications, network trust, and detection and estimation.

The work begins with finding and utilizing existing RSS based localization techniques and implementing and expanding them in simulation. Once the RSS localization is able to be simulated, the next task is to describe potential physical layer data attacks on the network. These attacks are grouped into two categories and models are developed for both. The next step is to use detection theory to find methods of determining which sensors in a network are faulty. With this research the GLRT and MSE algorithms are developed.

After the development of the algorithms, they are tested in a variety of potential situations. The goal is to determine the feasibility of the system as a whole and the algorithms comparatively. The results showthat both algorithms have benefits, and that this type of physical layer data is able to be used in the system described.

The work in this thesis was presented at the 2010 Asilomar Conference on Signals, Systems, and Computers [30].

5.2 Future Work

There are areas within this subject matter that justify future research. Each topic will be listed along with what future work might be interesting within that

area. These areas will be discussed in context of this research as well. The first area of potential future work is hardware implementation.

5.2.1 Hardware. The next logical step would be to implement these algorithms and derivations on physical data, rather than simulated data. Sentilla motes have been used to collect the necessary RSS measurements, and report them back to a command center. This is done by using one of the motes as a transmitter and the other motes as the receivers in the sensor network. The hardware implementation can be done in multiple steps. The first is replacing the synthetic RSS data with actual RSS data collected by the Sentilla motes. The rest of the processing can be done exactly the same as the simulation is done now. The faulty sensors will still be chosen in simulation. This would allow the researcher to determine the ability to use these specific sensors for the localization.

The next step would be to model how the hardware fails. In this research, two cases are considered. The malfunctioning and malicious models can be tested in hardware. The important questions for malfunctioning sensors would be:

- What is the model of battery life effect?
- What is the impact of interference RSS measurements?
- What level of physical damage causes bad RSS measurements?
- Does RSS measurement fail before the communication ability of sensor?

Questions that impact malicious sensors include:

- What ability does the sensor have to report different values?
- How much can a sensor affect localization in hardware?
- Can a sensor be reprogrammed without notice?
- Are there other models that affect localization more?

These questions would help to improve the models developed in this research.

5.2.2 Malicious Attack Modeling. Although malicious sensors are an important part of this research, more attention could be focused developing a model that would has the greatest effect on localization error. To better understand what types of attacks might occur on the network a researcher with the mindset and knowledge of attacking a network would be useful. Once these types of attacks are developed, the detection and classification of sensors can be done on these new malicious node models.

5.2.3 Parameter Estimation. Some of the quantities that are assumed known in this research are able to be estimated with the data available. In previous research [7], quantities such as antenna patterns, environmental fading, multiple transmitters, and transmission power have been developed. Including these estimations along with the ability to locate a transmitter and determine trust in the network would provide a well rounded and trustworthy RF map of an environment.

5.2.4 Localization Error Reduction and Faulty Node Handling. Another potential area of research in the future is using the detection of faulty sensors to improve the localization error. This might include various ways of disregarding, or correcting for false data. The metric would be the ability to improve the localization given the same environmental and network attack parameters.

Overall, this research demonstrates the feasibility of this type of system. The implementation in hardware would provide a final proof of concept, and other research could be included to give a broader scope.

Bibliography

1. R. Matin and R. Thomas, "Trust and Skepticism in Wireless Network Discovery," tech. rep., Air Force Institute of Technology, 2010.

2. N. Patwari, J. Ash, S. Kyperountas, A. H. III, R. Moses, and N. Correal, "Locating the Nodes," *IEEE Signal Processing Magazine*, vol. 22, pp. 54–69, July 2005.

3. V. Marojevic, J. Salazar, X. Revs, and A. Gelonch, "On Integrating Radio, Computing, and Application Resource Management in Cognitive Radio Systems," in *IEEE International Conference on Wireless and Mobile Computing, Networking and Communications*, 2007.

4. W. Krenik and A. Batra, "Cognitive Radio Techniques for Wide Area Networks," in *Proceedings of the Design Automation Conference*, pp. 409 – 412, Jun. 2005.

5. Y. Zhao, B. Le, and J. Reed, *Cognitive Radio Technology*, ch. Network Support: The Radio Environment Map, pp. 325–363. Academic Press, 2nd ed., Apr. 2008.

6. Y. Zhao, L. Morales, J. Gaeddert, K. Bae, J. Um, and J. Reed, "Applying Radio Environment Maps to Cognitive Wireless Regional Area Networks," in *IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks*, 2007.

7. R. Martin and R. Thomas, "Algorithms and Bounds for Estimating Location, Directionality, and Environmental Parameters of Primary Spectrum Users," *IEEE Transactions on Wireless Communications*, vol. 8, pp. 5692–5701, 2009.

8. D. Estrin, D. Culler, K. Pister, and G. Sukhatme, "Connecting the Physical World with Pervasive Networks," *IEEE Pervasive Computing*, vol. 1, pp. 59 – 69, Jan. 2002.

9. M. Ibrahim and M. Youssef, "CellSense: A Probabilistic RSSI-Based GSM Positioning System," in *IEEE Global Telecommunications Conference*, pp. 1 –5, Dec. 2010.

10. X. Sheng and Y. Hu, "Maximum Likelihood Multiple-Source Localization Using Acoustic Energy Measurements with Wireless Sensor Networks," *IEEE Transactions Signal Processing*, vol. 53, pp. 44–53, Jan. 2005.

11. X. Wei, L. Wang, and J. Wan, "A New Localization Technique Based on Network TDOA Information," in *International Conference on ITS Telecommunications Proceedings*, 2006.

12. C. Rondeau, "Navigation with Limited Prior Information Using Time Difference of Arrival Measurements from Signals of Oppurtunity," Master's thesis, Air Force Institute of Technology, Dec. 2010.

13. Y. Chan, B. Lee, R. Inkol, and Q. Yuan, "Direction Finding With a Four-Element Adcock-Butler Matrix Antenna Array," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 37, pp. 1155 –1162, Oct. 2001.

14. S. Wang, R. Inkol, S. Rajan, and F. Patenaude, "Comparison of two Angle of Arrival Averaging Strategies," in *Canadian Conference on Electrical and Computer Engineering*, pp. 1105–1110, May 2009.

15. W. Wang and Q. Zhu, "RSS-Based Monte Carlo Localisation for Mobile Sensor Networks," *IET Communications*, vol. 2, pp. 673–681, May 2008.

16. R. Malaney, "Nuisance Parameters and Location Accuracy in Log-Normal Fading Models," *IEEE Transactions On Wireless Communications*, vol. 6, pp. 937–947, 2007.

17. A. Weiss, "On the Accuracy of a Cellular Location System Based on RSS Measurements," *IEEE Transactions on Vehicular Technology*, vol. 6, pp. 1508–1518, 2003.

18. X. Li, "Designing Localization Algorithms Robust to Signal Strength Attacks," in *IEEE Communications Society Conference on Sensor Mesh and Ad Hoc Communications and Networks (SECON)*, 2010.

19. Y. Chen, W. Sun, and J. Juang, "Outlier Detection Technique for RSS-Based Localization Problems in Wireless Sensor Networks," in *SICE Annual Conference*, 2010.

20. J. Dulmage, R. Cioffi, M. Fitz, and D. Cabric, "Characterization of Distance Error with Received Signal Strength Ranging," in *Wireless Communications and Networking Conference*, 2010.

21. S. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Prentice Hall Signal Processing Series, 2009.

22. J. Devore and N. Farnum, *Applied Statistics for Engineers and Scientists*. Brooks/Cole, 2005.

23. M. Momani, S. Challa, and R. Alhmouz, "BNWSN: Bayesian Network Trust Model for Wireless Sensor Networks," in *Mosharaka International Conference on Communications, Computers and Applications, 2008*, pp. 110 –115, 2008.

24. T. Chen and V. Venkataramanan, "Dempster-Shafer Theory for Intrusion Detection in Ad Hoc Networks," *IEEE Internet Computing*, vol. 9, no. 6, pp. 35 – 41, 2005.

25. Y. Zhang and W. Lee, "Intrusion Detection in Wireless Ah-Hoc Networks," in *Proc 6th Ann ACM Int'l Conf. Mobile Computing and Networking (Mobicom)*, pp. 275–283, Aug. 2000.

26. W. Du, J. Deng, Y. Han, and P. Varshney, "A Witness-Based Approach for Data Fusion Assurance in Wireless Sensor Networks," in *IEEE Global Telecommunications Conference*, vol. 3, pp. 1435 – 1439 vol.3, 2003.

27. R. Khanna, L. Huaping, and C. Hsiao-Hwa, "Reduced Complexity Intrusion Detection in Sensor Networks Using Genetic Algorithm," in *IEEE International Conference on Communications*, pp. 1 –5, 2009.

28. Z. Yang, N. Meratnia, and P. Havinga, "Outlier Detection Techniques for Wireless Sensor Networks: A Survey," *IEEE Communications Surveys Tutorials*, vol. 12, no. 2, pp. 159 –170, 2010.

29. H. L. V. Trees, *Detection, Estimation, and Modulation Theory*. John Wiley and Sons, Inc., 1968.

30. T. Hardy, R. Martin, and R. Thomas, "Malicious Node Detection via Physical Layer Data," in *Asilomar Conference on Signals, Systems, and Computers*, Nov. 2010.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704–0188

The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704–0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202–4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. **PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.**

1. REPORT DATE *(DD–MM–YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From — To)*
24–03–2011	Master's Thesis	June 2009-March 2011

4. TITLE AND SUBTITLE

Malicious and Malfunctioning Node Detection via Observed Physical Layer Data

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Hardy, Tyler J., 2LT, USAF

5d. PROJECT NUMBER

None

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way
WPAFB OH 45433-7765 DSN: 785-3636

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT/GE/ENG/11-14

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Air Force Research Laboratory
Attn: AFRL/RYRE (Dr. Vasu Chakravarthy)
2241 Avionics Circle, Bldg 620
WPAFB OH 45433-7734
(937)255-5579
Vasu.Chakravarthy@wpafb.af.mil

10. SPONSOR/MONITOR'S ACRONYM(S)

AFRL/RYRE

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED;
THIS MATERIAL IS DECLARED A WORK OF THE U.S. GOVERNMENT AND
IS NOT SUBJECT TO COPYRIGHT PROTECTION IN THE UNITED STATES

13. SUPPLEMENTARY NOTES

14. ABSTRACT

There are many mechanisms that can cause inadequate or unreliable information in sensor networks. A user of the network might be interested in detecting and classifying specific sensors nodes causing these problems. Several network layer based trust methods have been developed in previous research to assess these issues; in contrast this work develops a trust protocol based on observations of physical layer data collected by the sensors. Observations of physical layer data are used for decisions and calculations, and are based on just the measurements collected by the sensors. Although this information is packaged and distributed on the network layer, only the physical measurement is considered. This protocol is used to detect faulty nodes operating in the sensor network. The context of this research is Wireless Network Discovery (WND), which refers to modeling all layers of a non-cooperative wireless network. The focus in particular is the localization of transmitters, and detection of sensors affecting the localization. To accomplish this, a model for faulty sensors and two methods of detection are developed. Detection rates are analyzed with Receiver Operating Characteristic (ROC) curves, and the trade-off of detection versus localization error is discussed. Classification between faulty sensors is also considered to determine appropriate response to potential network attacks.

15. SUBJECT TERMS

Detection and Estimation, Wireless Sensor Networks, Network Security, Received Signal Strength

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Dr. Richard K. Martin (ENG)
U	U	U	UU	74	19b. TELEPHONE NUMBER *(include area code)* (937)255-3636x4625; email:richard.martin@afit.edu