DEVELOPMENT AND EVALUATION OF RSSI-BASED LOCALIZATION SCHEMES FOR WIRELESS SENSOR NETWORKS TO MITIGATE SHADOWING EFFECTS

by

Ndubueze O. Chuku

A dissertation submitted to the faculty of
The University of North Carolina at Charlotte
in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in
Electrical Engineering

Charlotte

2020

Approved by:

____
Dr. Asis Nasipuri

____
Dr. James Conrad

____
Dr. Robert Cox

____
Dr. Andrew Willis

____
Dr. Srinivas S. Pulugurtha

ABSTRACT

NDUBUEZE O CHUKU. Development and Evaluation of RSSI-Based Localization Schemes for Wireless Sensor Networks to Mitigate Shadowing Effects. (Under the direction of DR. ASIS NASIPURI)

Received radio frequency (RF) signal strength provides a cost-effective mechanism for distance estimation that is popularly used for range-based localization in wireless sensor networks (WSN). The typical method of determining sensor location using range-based localization methods is multilateration and this involves combining RSSI (received signal strength indicator) information from a number of beacons that is greater than the minimum number required for localization using accurate distance estimates. Multilateration using RSSI-based distance estimates are severely affected by shadowing and result in erroneous sensor location estimation. As a result of this, there was the need to come up with ways to overcome the effects of these shadowed measurements.

The objective of this research is to minimize the effects of shadowing in sensor location estimation. To address this problem, several methods were explored. First, a scheme that applies a spatial correlation mechanism to eliminate RSSI signals that are affected by obstructions (i.e. shadowed signals) is presented. It is shown that the scheme is effective in minimizing the adverse effects of shadowing on RSSI signals hence sensor node localization. Next, outlier detection schemes were explored as a method to minimize the effects of shadowing. The effectiveness of the correlation-based localization scheme and the outlier detection schemes are validated using simulations and experimental data and have been shown to improve on sensor localization compared to other popular localization schemes.

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my advisor Dr. Asis Nasipuri for his tremendous guidance during the course of my doctoral program. I will forever be grateful for the research aptitude skills that he impacted in me in his characteristic calm and patient demeanor.

I also give hearty thanks to my dissertation committee members: Dr. James Conrad, Dr. Robert Cox and Dr. Andrew Willis for their invaluable assistance, advice and encouragement as I navigated the course of my dissertation research.

My appreciation will not be complete without a special mention of my family for their love, patience, encouragement and support during these years without which this would not have been possible. I will like to thank my lovely children: Ndubueze A. Chuku, Jasmeen I. Chuku and Mikayelle O. Chuku. A big thank you to my late father, Dr. Eliezer C. O. Chuku, my mother, Mrs. Felicia Igbeaku Chuku, my siblings: Barr. Chijioke Chuku, Mrs. Ijeoma Mba and Dr. Nnenna Igwe.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ADO	Arrival and departure overlap
AOA	Angle of arrival
BN	Blind node
BS	Base station
CC	Closeness centrality
CR	Correlation region
CorrReg	Correlation region-based algorithm
EKF	Extended kalman filter
GPS	Global positioning system
IF	Information filter
kHz	Kilo Hertz
LOS	Line of sight
Multi	Multilateration
MMSE	Minimum mean square error
NLOS	Non line of sight
OD_CTRD	Outlier Detection_Centroid
OD_MSC	Outlier Detection_Mean Shift Cluster
PF	Particle Filter
RAM	Random Access Memory
RF	Radio frequency
RSSI	Received Signal Strength Indicator
ROSLAM	Range Only Simultaneous Localization and Mapping

SEIF	Sparse extended information filter
SLAM	Simultaneous Localization and Mapping
TOA	Time of arrival
TDOA	Time difference of arrival
UKF	Unscented kalman filter
UN	Unknown Node
WSN	Wireless Sensor Network

CHAPTER 1: INTRODUCTION

In that last couple of decades, substantial amount of research has been carried out towards enabling Wireless Sensor Networks (WSNs) to be a sustainable and inexpensive solution for distributed monitoring tasks. WSNs are networks of many small low-cost wireless sensors, which are electronic devices with embedded hardware for sensing physical quantities of their immediate environment and a radio for transferring data through multi-hop routing to a data-aggregating device called a sink. Sensor technologies integrated in wireless sensor nodes include: humidity, infrared light, temperature, acoustic, pressure, vibration, radar etc. WSNs apply ad-hoc networking principles that enable self-configuration and adaptations to changes such as addition and removal of wireless sensors. Typically, WSNs have a tree topology running from the data source (nodes that sense data) down to the data sink as shown in Fig. 1.1. WSNs are capable of applying distributed processing for applications monitoring employing multi-modal data got from onboard sensors to handle onboard processing. These applications include medical and health monitoring, environmental monitoring, logistics, agriculture, disaster relief, security etc. WSNs are also easily deployable and can be reprogrammed to adjust to changes in conditions. There are a wide range of WSN applications. However, in terms of communications requirements and network operations, they can be broadly classified into three broad groups [67]:

- Periodic measurements – all wireless sensor nodes deployed are set to periodically measure values and report such measurements to the sink.
- Tracking – sensor nodes measure the updated position of a mobile object or an event and report these updates to the sink. To achieve this the various sources have to cooperate prior to updates been transmitted to the sink.
- Event detection – sensors detect the occurrence of a specific event and report such detection to the sink. Individual sensors can detect an event independently (e.g. presence of a person in a room) or can work collaboratively with neighbor sensors to detect more complex events. Communication occur only when events are detected.

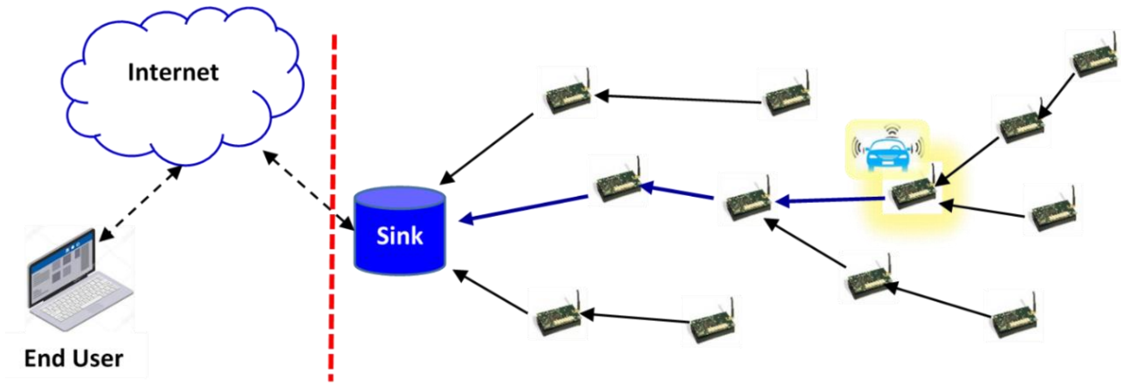


Figure 1.1: A typical Wireless Sensor Network

Due to the vast range of application types that can utilize WSNs, no single realization of a WSN has the ability to handle all the application types. However, some common characteristics and mechanisms of these systems exist. Achieving the said characteristics using new mechanisms poses a key challenge in the realization of WSNs. A few of the common characteristics of most WSN applications include: Quality of Service, Type of Service, Lifetime, Scalability, programmability, Maintainability etc. These characteristics can be realized using the following mechanisms: Multi-hop wireless communication, Energy-efficient operation, collaboration, Data centric, Locality etc.

The key problem affecting the development of long-term applications using WSNs lies in the fact that sensor nodes are resource constrained in terms of communication, energy resources and computational capabilities, and as such these constraints must be considered while designing tasks.

Effective utilization of WSN has generated significant research in a large number of different areas which includes media access control, routing, application-based communications, collaboration etc. However, in this research, our focus is on position estimation of wireless sensor nodes, popularly termed as localization, which is an important aspect of WSN that has also generated a lot of research.

Localization is a very important task required for most applications and also for helping networking protocols, such as routing, in WSNs. In many situations, it is very important for a wireless sensor node to know its position in the real-life environment. Let us consider a WSN deployed in an Intelligent Parking Lot to monitor free parking spots for cars. If a given sensor node raises an alert for a certain free parking space but that sensor node does not know its location within the Parking Lot, then the information/alert raised by the sensor node to alert drivers of a free parking spot is useless. This problem can be extrapolated to other WSN deployments and as such is a very important topic in WSN research. How can this sensor node position awareness be solved?

The importance of Localization in WSN cannot be over-emphasized. Measurement data used in WSNs applications require the knowledge of the particular location in an environment where such measurements were taken. However, despite the remarkable progress achieved in the advancement of geographical positioning systems (GPS), node localization needs specialized solutions for WSNs because advanced positioning systems, like the GPS, are impractical for WSNs because of cost as well as energy limitations of sensor nodes. Also, GPS needs satellite signals and these signals may be unavailable for sensor nodes deployed in areas that have some form of covering over the deployment region. For most WSN deployments, it is prohibitive to manually deploy and record the locations of every sensor node because of the large number of sensor nodes involved. Consequently, a mechanism is needed by which the locations of deployed sensor nodes can be determined automatically post deployment. Hence, a lot of ongoing research work is ongoing in the development of cost-effective localization methods for WSNs that are able to function under the resources-limitations of the nodes.

1.1. The Localization Problem in Wireless Sensor Networks

A wide range of localization problems in WSN have been addressed in literature. The requirements for localization schemes in WSNs mostly hinge on the specific properties of the applications and also on the hardware and network infrastructure limitations. A few of the most important properties of localization in WSN include [12]:

- Absolute vs. relative coordinates: GPS and other positioning systems determine absolute coordinates of objects on earth. Conversely, the requirement may simply be an arbitrary reference location at the application area, such as a conference room located inside an office building, and as such relative coordinates are used. With relative coordinates, sensor nodes may have coordinates that are accurate in relation to one another but are unrelated to the absolute coordinates in earth coordinates.

- Accuracy and precision of node localization: Localization accuracy may depend on several factors such as the mechanism or technology employed. Granularity of measurements of WSN localization methods depends on the application and may range from a few centimeters up to tens of meters or even greater. Precision on the other hand describes the consistency of the location estimates. An example use of precision is a system which estimates the location of a node with 15cm accuracy and 90% precision.

- Mobility vs. Immobility of sensor nodes: With mobile sensor nodes, the localization schemes encounters the problem of continuously taking measurements in other to track the location of the nodes. The rate at which the measurements are taken depends on the dynamics of the sensor nodes. Most WSNs applications require the sensor nodes to be static after they are deployed. However, there are also dynamic cases whereby sensors embedded on robots are used. Therefore it is a requirement that a localization scheme for WSNs is capable of tracking the movement of sensor nodes where applicable.

- Scale of measurements: The type of WSN application determines the scale of measurements used. A scale of measurement defines the farthest distance over which the localization scheme can cover. Different scales include: outdoor deployment such as a football field or an indoor deployment such as a school auditorium.

- Communication requirements: Current systems differ in their communications requirements for estimating positions of objects. Some devices, e.g. a GPS receiver, does not transmit signals but simply implements localization employing radio frequency signals obtained from a system of satellites. Conversely other systems such as cellular telephone localization methods depend on 2-way communication between the phone and the base stations. In WSNs, communication between sensor nodes and beacon nodes or other sensor nodes offers such vital benefits like accuracy/precision improvements as well as time synchronization. Nevertheless, a fundamental problem in WSNs is the reduction in the communication requirements of the sensor nodes to extend battery life. Moreover, this presents unique considerations in the design of localization schemes.

- Self-localization vs. remote localization: Some WSN localization schemes require the nodes to localize themselves using received beacons from location-aware nodes or devices. Conversely, some localization schemes may permit a remote device, for example a base station, to localize the sensors nodes using beacons received from the nodes. The main difference between self-localization and remote localization is where the computations occur and this may require the sensor nodes to communicate with the remote base station. Hence the disadvantages of communication cost and scalability issues experienced with remote localization schemes.

- Cost: It is imperative that wireless sensor nodes do not have additional hardware, for localization, that will introduce increased cost in size and capital.

- Form factor: A key requirement of wireless sensor nodes is that they must be small in size. The sensor node's size therefore is highly important in the determination of an effective mechanism that can be employed for localization in WSNs.

- Passive vs. active localization: In some localization systems, the UN may be required to actively participate in location estimations whereas in other schemes, no active participation is required of the UN. The role and level of participation of wireless sensor nodes in location estimation is dependent on the limitations caused by the nodes' hardware and cost constraints.

1.1.1. Challenges in Wireless Sensor Network localization

The main challenges prevalent in the design of a localization scheme for WSNs come from the necessity to deal with the small node size, low hardware complexity, cost of implementation of the sensor nodes, as well as their arbitrary deployment environments. The typical approach for estimating a sensor node's location is triangulation, which involves estimating distances or angles from several fixed reference points. We will briefly describe the challenges inherent in determining these estimates in sensor networks.

A. Ranging Issues:

Using RF signals to measure time of flight (TOF) is very challenging for applications in WSNs because RF signals travel at the speed of light and measuring these very short flight times especially in the sensor network field poses a technical problem as it would need very accurate clock synchronization between the sensor node and the transmitter. Therefore, researchers have explored TOF measurements using ultrasonic and acoustic signals that have considerably slower speeds than the radio frequency. Acoustic range-based measuring devices are commonly used because they are inexpensive and accurate when used indoors. They are capable of achieving accuracies within tens of centimeters. The relatively higher frequency ultrasonic signals (typically 24−40KHz) capable of achieving accuracies of less than 5 centimeters, have smaller range than the acoustic signals. Other challenges faced due to the use of acoustic or ultrasonic signals for ranging in WSNs include the following:

- Acoustic sources and detectors are larger in size than the RF sources because of their larger wavelength hence are a problem for small sensor nodes.
- Also due to their larger wavelength, they cannot travel through physical barriers.
- The effect of severe multipath effects which makes it difficult to design a reliable range estimation system for arbitrarily deployed sensor nodes in unfamiliar environments.

If the RF signal is received immediately, the corresponding delay of the considerably slower ultrasound signal provides the required range estimate. Using RSSI for range estimates needs the knowledge of the corresponding signal propagation model. However, even with extensive channel estimation and modeling, RSSI-based range estimates experiences inaccuracies caused by shadowing, scattering effects, refractions, and multipath reflections. Consequently, due to the simplicity of implementation, significant amount of research has been carried out on RSSI-based localization schemes.

A. Angle of Arrival (AOA) Estimation:

An antenna array can be employed in the estimation of the direction of arrival of a wireless signal because of its capability to achieve extremely small beam width. However, this will be extremely big for use in small form factor wireless sensor nodes. Therefore, for AOA estimations for application in wireless sensor networks, alternative techniques are required. One of such approaches used in the reduction of antenna arrays is the use of ultrasound signals [69] and this can be used to calculate angles from phase and time difference of arrivals on an ultrasound pulse on several detectors positioned in a defined pattern in a very small area typically a few centimeters apart and has the ability to obtain the angle of arrival with a very high accuracy.

A critical prerequisite for the realization of this mechanism is that a line of sight from the source must exit. Multipath and scattering are sources of problems in AOA estimation because they rely on phase differences.

B. Other Technical Challenges:

To reduce the errors experienced in distance or angle measurements, one may take several measurements at a sensor node or use measurements from neighboring nodes. This concept is termed collaboration. To achieve either of the above, optimization operations is required and these require extensive computations as well data transmission which is not ideal especially in resource constrained sensor nodes.

A method popularly used in localization, which does not require additional hardware for the wireless sensor nodes, is the use of distance estimates of such sensor nodes from beacons with known locations using the radio frequency (RF) received signal strength indicator (RSSI). However, because of the irregularities of the RF signal propagation, RSSI is not an accurate measure of distance. Thus a greater part of the research work on RSSI based localization methods have focused on creating effective methods for minimizing errors of such distance estimates. A typical method of achieving this is by multilateration, which involves combining RSSI measurements from a number of beacon nodes (BNs) more than the lowest number needed for localization when accurate distance estimates are used (in other words D+1 beacon nodes for a D-dimensional space). A major motivation of our research stems from the idea that multilateration using RSSI-based distance estimates is adversely affected by shadowing effects which results in some of the RSSI measurements being unusually more erroneous than others. Hence we propose an approach that utilizes a simple spatial correlation mechanism to choose a subset of a large number of beacon signals to perform multilateration. The thought behind this proposed approach is that for any sensor node in a typical WSN, some beacon signals will not be obstructed and therefore distances estimates from their RSSI values will have less errors than those from obstructed beacon signals. Consequently, multilateration using these RSSI measurements from the unobstructed beacon signals would yield greater accuracy of the location of the sensor node.

We show from simulations that using such a scheme improves the localization accuracy more than the approach that applies multilateration to all the beacon signals together. We also show performance evaluations attained from experimental testbeds showing the effectiveness of this approach.

1.1.2. Approaches to localization in Wireless Sensor Networks

There are some localization schemes, depending on the application, that require accurate XY-coordinates and on the other hand there are those schemes than simply require proximity information. Localization approaches for WSNs can be broadly classified into range-free and range-based approaches. Range-free localization schemes do not utilize range measurements in the estimation of the location of a sensor node. However, the approaches do not try to offer accurate location estimates. For example, some range-free localization schemes employ RF-based proximity approach (i.e. use communication ranges) to estimate the area in an environment where a sensor node is located.

(a) (b)

Figure 1.2: (a) Cross-section of the Paradise substation where the ParadiseNet was deployed. (b) a wireless sensor node for monitoring circuit-breakers. [68]

Range-based localization approaches use RSSI, time of arrival (TOA), time difference of arrival (TDOA) or angle-of-arrival (AOA) to measure the range or angle between a BN and a sensor node and then employ trilateration or triangulation to estimate the location of the sensor nodes. When the range or angle estimates are not highly erroneous, range-free localization approaches are effective. Range-based approaches are more accurate for localization but need additional hardware for AoA and TDoA: for example Acoustic Module for TDoA and Radio or array microphone for AoA.

1.2. System Model

The dissertation research was motivated by practical observations from the actual deployment of a WSN that was developed for monitoring the health of equipment at the TVA-operated power substation in Kentucky. The project, sponsored by EPRI and initiated in 2006, led to the deployment of a 122-node WSN called the ParadiseNet at the power substation and is shown in Fig. 1.2 [68]. The challenges experienced in manually recording and tracking the exact locations of the sensor nodes as well as the fact that several energy-conserving protocols and algorithms being developed require the sensor nodes to be location-aware necessitated the development of a self-localization scheme for this network. For such a deployment site, utilizing a mobile robotic beacon transmitter can be an effective approach in realizing sensor self-localization. Although this approach is technically practical, it has the challenging problem given that a location like this contains large

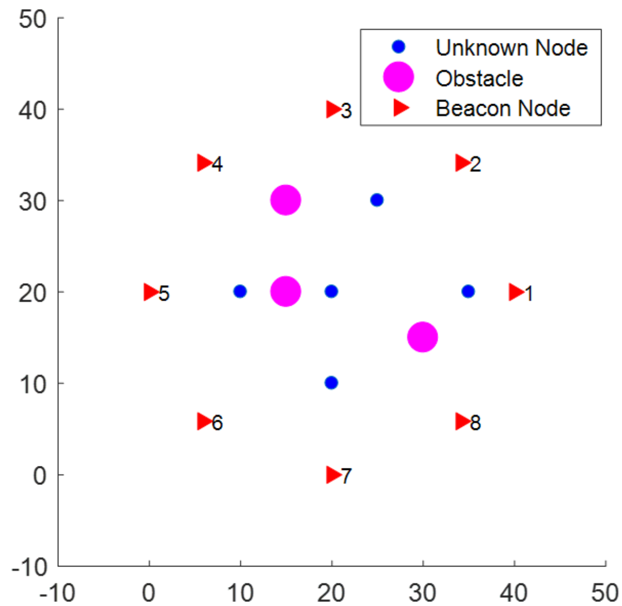


Figure 1.3: Wireless Sensor Network Layout showing beacon node positions, position of an unknown node and obstacles.

metallic objects like capacitor banks, transformers, circuit breakers, bus- bars etc. in the network area which causes large errors to the RSSI measurements due to shadowing. Therefore the use of a beacon mounted robotic platform in the localization system will yield better localization results if the sensor nodes have the ability to select beacon signals that are not severely affected by shadowing, and in essence deselecting those adversely affected by shadowing.

This type of deployments are common in real-life WSN applications. We hereby pose the localization problem thus. We assume that in a given environment, a set of wireless sensor nodes are randomly deployed. Also deployed are a set of "B" number of beacon nodes, with known locations, which will assist in the sensor nodes' localization process. The BNs broadcast RF signals that contain their location. Several statically place BNS can be used to achieve this. Another effective approach for implementing beacon generation is the use of a mobile robot fitted with a GPS and communication hardware to transmit its locations periodically as it moves through the network area. Sensor nodes on receiving the RF signals from the BN estimate their distances, from RSSI measurements, by employing a path loss model that is assumed to be calculated from offline channel measurements. A sensor node can estimate its location by employing beacon signals transmitted from at least three non-collinear locations and can apply multilateration to a larger number of received beacon signals for improving this location estimate. In this research, we have concentrated on using RSSI-based distance estimation for sensor localization although other methods exist as earlier described. The main reason, as we have stated earlier, is that distance estimation using RSSI dos not require the sensor node to have additional hardware and hence does not add to the cost of the sensor node. The problem thus, is to design an RSSI-based self-localization scheme for the sensor nodes such that they utilize a portion of the received beacon signals to achieve the least error of their location estimate from multilateration.

1.2.1. Assumed network layout

A typical network model used in our evaluation of the proposed localization approaches is shown in Fig. 1.3. A sensor network area of 40m x 40m is used. The red triangles depict beacon or anchor nodes, the blue circles depict the unknown nodes and the purple circles depict the obstacles that cause shadowing on the transmitted signals.

1.2.2. Wireless channel model

Here we describe the radio channel propagation model on which the localization scheme is based on. We assume that the deployment location include natural or man-made objects such as trees, pillars, and metallic objects that serve as obstacles in the path of beacon signals. To effectively model the radio channel propagation characteristics obtainable in these types of environments, we employed the lognormal shadowing model with added attenuation loss accounting for obstructions.

The lognormal shadowing model is typically used to represent terrestrial path loss along with a random (Gaussian) long-term fading component. According to this model, the power of a received beacon signal at a sensor node that is located at a certain distance from a beacon node can be calculated using the equation below:

$$P_r(d) = P_r(d_0) - 10n\log\left(\frac{d}{d_0}\right) + N[0,\sigma] \qquad (1.1)$$

where $P_r(d_0)$ represents the received power at a reference distance d_0, n represents the path loss exponent (usually between 2 and 4) and $N[0, \sigma]$ represents a zero-mean Gaussian random variable with σ standard deviation originating from the channel noise and shadowing effects.

The radio channel can be modeled offline employing a small portion of RSSI measurements to determine parameters like n and $P_r(d_0)$ and these parameters used as the input to any chosen localization scheme. Due to the spatio-temporal characteristics of the radio channel in the estimated model, these parameters are likely inaccurate representations of the true radio channel of the environment at any specific time, which introduce some errors in the RSSI and also in the distance estimation. Furthermore, in an environment containing obstacles, the received signal power is greatly perturbed because of obstacles located in the path between a beacon node and the unknown node. The unknown nodes can also dynamically acquire these channel parameters.

The channel propagation model is described below:

Received power from unobstructed beacon signals is calculated with the equation shown below:

$$P_r(d) = P_r(d_0) - 10n\log\left(\frac{d}{d_0}\right) + N\left[0,\ \sigma_f\right] \tag{1.2}$$

Received power from obstructed beacon signals is calculated with the equation shown below:

$$P_r(d) = P_r(d_0) - 10n\log\left(\frac{d}{d_0}\right) + N\left[0,\ \sigma_f\right] + S(\sigma_s)_{NLOS} \tag{1.3}$$

where σ_f and σ_s represent standard deviations of the fading component experienced in line of sight (LOS) and non-line of sight (NLOS) beacon signals respectively..

1.3. Effect of erroneous distance estimates in Wireless Sensor Network Localization

Distance estimates calculated from RSSI from beacon signals are used in calculating the location of an unknown node. As we stated earlier, the typical approach used in the estimation of the location of an UN is the multilateration. Multilateration applies a minimum-mean-square-error (MMSE) technique employing range estimates from a large number of BNs. This approach normally uses a number of BNs which exceeds the minimum number needed for node localization when accurate range estimates are used (i.e. D+1 beacons for D-Dimensional space).

1.3.1. Distance estimation from RSSI measurements

In WSN localization, distance and angle measurements are measured in physical media which introduces errors such as time-varying and static error, environment-dependent errors [34]. Time-varying errors (e.g., errors from additive noise and interferences) can be minimized by taking an average of several measurements taken over time. Environment-dependent errors, on the other hand, are the result of the real placements of objects (e.g., metallic objects, trees, and furniture) within a wireless sensor network environment. Since the environment, hence measurement errors are unpredictable, we have modelled them as random. However, in environments where objects are stationary and also most of the sensor nodes are stationary, environment-dependent errors as earlier described will be mostly constant over time. Distances used in estimating the location of an UN are calculated using the formula shown in equation 1.4.

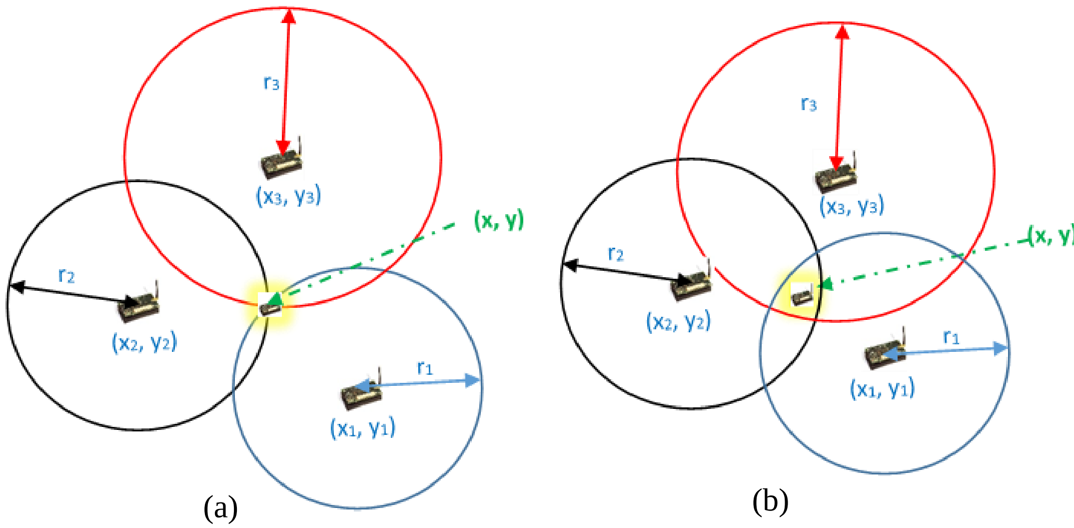
$$d_{Est} = 10^{(P_t(d_0) - P_r(d))/10n} \tag{1.4}$$

Where d_{Est} denotes the estimated distance between a BN and an UN and $P_t(d_0)$ denotes path loss at reference distance d_0.

1.3.2. Effect of error in localization

When all the distance estimates d_{Est}, used in a multilateration problem are accurate, i.e. $d_{Est} = d$, the result of the sensor location estimate is also accurate. The challenge for localization using distances is that the distance measurements, as we described in the previous sub-section, are not perfect but mere estimates of the true distance. This can be illustrated with the figures in Fig. 1.4. With perfect distance estimates, depicted in Fig. 1.4(a) all the circular rings formed by the RF propagation region, with centers at each of the beacon nodes, intersect at a point. Conversely, when at least one of the distance estimates is inaccurate, the intersection of the rings will not be at a single point but rather be either a region as is shown in Fig. 1.4(b).

$$d_{Est} = \begin{cases} d, & \text{for ideal situations} \\ d + \varepsilon, & \text{for real situations} \end{cases} \qquad (1.5)$$



(a) (b)

Figure 1.4: Sensor node localization using (a) error-free distance estimates and (b) erroneous distance estimates.

The typical approach used for such a problem is to employ more than D + 1 number of BNs and redundant distance estimates in a D-dimensional scenario. The objective here is to minimize the effect of the errors in the distance estimates. To solve this mathematically, we set up a set of overdetermined system of equations and seek a solution that minimizes the mean square error.

1.4. Research Outline

The primary objective of the dissertation research is to minimize the effects of obstructed beacon signals used in sensor node localization. In Fig. 1.5, we illustrate a typical WSN deployment scenario using the multilateration method for estimating the location of an unknown node. When there are no obstructions to the beacon signal(s), the distance estimates used in the multilateration will be less erroneous and hence produce sensor node location estimate close to the true location of the sensor node. In contrast,

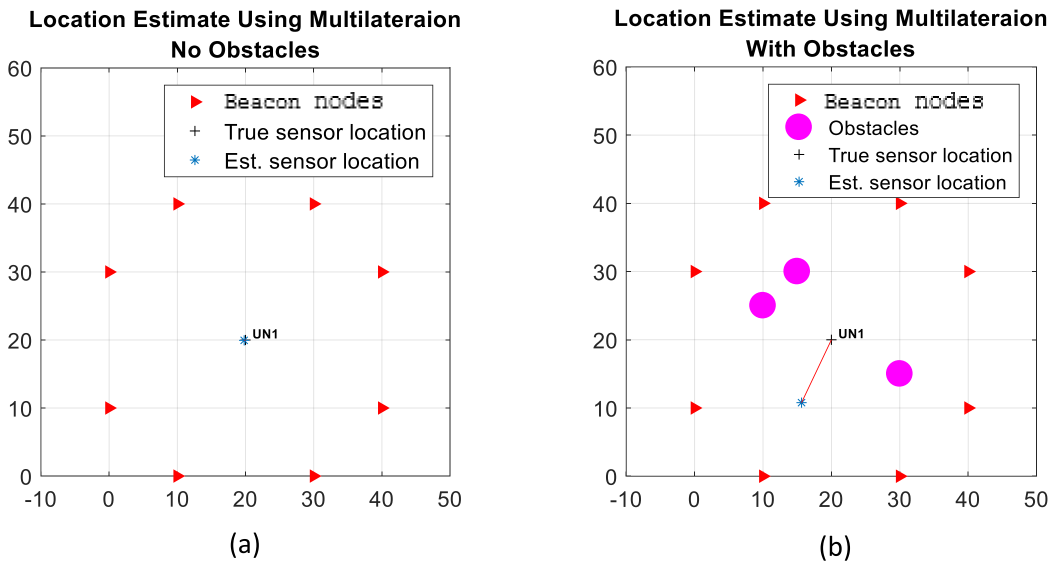


Figure 1.5: Node Location estimate in a wireless sensor network area using multilateration: a) No obstacles present in the area and b) Obstacles present in the area.

when there are obstructions in some or all the beacon signals received by the sensor node, then the distance estimates will be inaccurate thereby inducing significant error in the sensor node location estimate. This is due to the fact that in multilateration, all received beacon signals used in estimating the location of the unknown node are given equal weight whether those signals are obstructed or not. What this means is that beacon signals that are obstructed introduce errors in the node location estimate.

In this research, we have investigated three approaches that aim to solve the problem stated above. The three approaches are the Correlation region based Localization scheme, the Outlier Detection-Centroid and Outlier Detection-Mean Shift Clustering Localization schemes. The idea behind these approaches is that for any sensor node in a typical WSN deployment, some of the beacon signals will not be obstructed, and therefore distances estimated from their RSSI values will be less inaccurate than those from obstructed beacon signals. Therefore, multilateration with RSSI measurements from these unobstructed signals would yield better accuracy in estimating the location of the node. These design approaches are outlined below:

Correlation region based Localization Algorithm (corrReg): The corrReg employs spatial correlation to several independent trilaterations and combining them to deemphasize the impact of the obstructed beacon signals to the sensor node localization. This approach minimizes the contribution of those obstructed beacon signals towards the localization of a sensor node.

Outlier Detection - Centroid Algorithm (OD-CTRD): An N number of subsets of three BNs are employed to produce the same number of location estimates called candidate location estimates. These location estimates are produced using some beacon signals that are obstructed and other beacon signals that are unobstructed. We have assumed that candidate location estimates produced from distance estimates from unobstructed beacon signals will be less inaccurate and their corresponding locations will be spatially correlated and close to one another. However the location estimates produced from highly inaccurate beacon signals will be spatially uncorrelated and therefore be farther apart from each other and these we have called outliers.

The goal of this localization scheme is to utilize the above assumption to systematically combine the spatially correlated candidate location estimates and thus remove the outliers. Then the residual and less erroneous candidate location estimates are used to determine the final estimate of the UN's location.

Outlier Detection – Mean Shift Clustering (OD-MSC): The idea and assumption made in proposing this algorithm is the same for corrReg and OD-CTRD methods. The OD-MSC scheme is a cluster-based localization algorithm that employs a sliding-window that aims to locate dense regions of the candidate location estimates (data points). It uses a centroid-based mechanism to find the centers of every cluster and then updates candidates for centers to be the mean of the data points located in the sliding-window. The candidate windows undergo a post-processing filtering process to remove duplicates or near-duplicates, yielding the final set of centers and their corresponding clusters.

Fig. 1.6 shows a summary of the key design considerations we investigated and these will be outlined in depth in later sections.

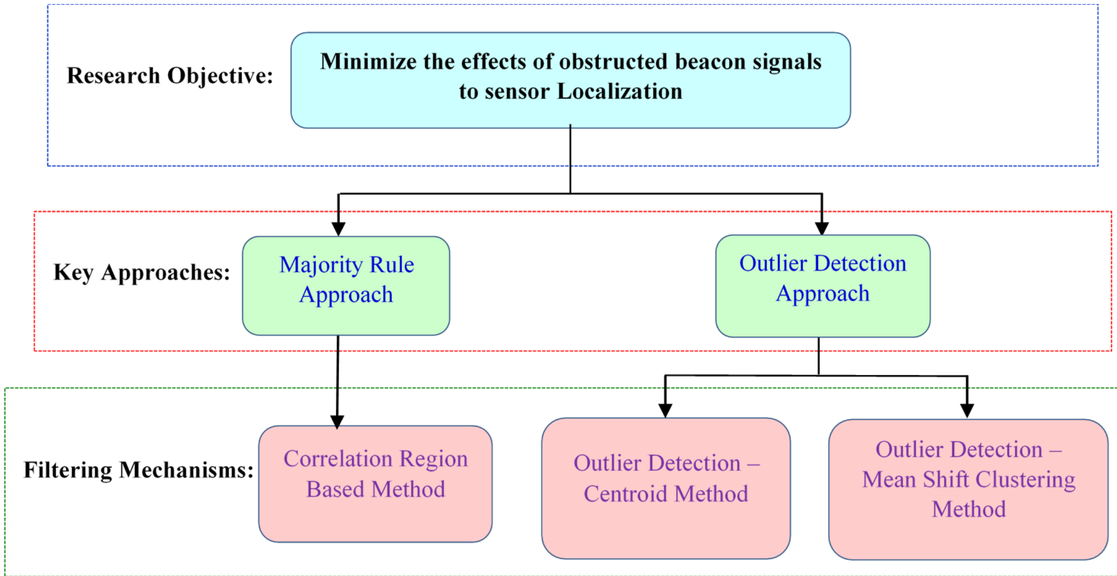


Figure 1.6: Dissertation Research Outline

1.5. Dissertation Organization

The rest of the dissertation is organized as follows. In Chapter 2, we discuss related work. Chapter 3 analyzes the Simultaneous Localization and Mapping (SLAM), with emphasis on Range-Only SLAM (ROSLAM) with performance results. SLAM employs a weighting mechanism that can reduce the effects of obstructed beacons and this weighting mechanism is similar in principle to the mechanism used in our proposed localization scheme. Chapter 4 describes our correlation region based localization scheme which employs a Majority Rule mechanism that reduces the effect of obstructed beacons. We will show performance results from simulations and discuss the computation issues that arise as a result. In Chapter 5, we present our Outlier detection schemes: OD-CTRD and OD-MSC with performance results. We show, in Chapter 6, performance comparison for all the localization schemes for simulations as well as experimental testbeds. We summarize our contributions in chapter 7 as well as lay out proposed recommendations for future work.

CHAPTER 2: RELATED WORK

We will present a brief literature review of related work. To streamline this review, we first present a classification in taxonomy of relevant localization schemes for WSNs. WSN Localization schemes can be broadly grouped into two classes: distributed and centralized algorithms. Fig. 2.1 shows the classification of localization schemes used in wireless sensor networks.

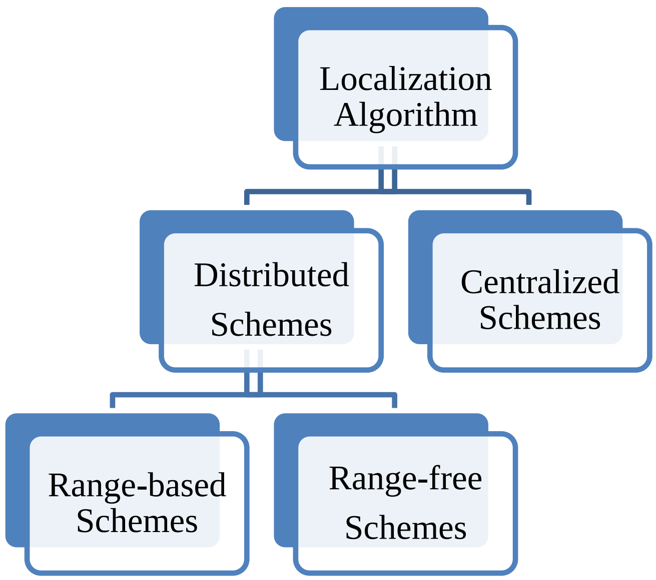


Figure 2.1: Classification of wireless sensor localization schemes.

2.1. Centralized localization schemes

In Centralized localization schemes, all computation is done at a central station thereby resolving the computational constraints of sensor nodes. However, in these schemes, there is a need for sensor nodes to communicate with the Base Station (BS) thereby consuming more energy than consumed in computation.

A few of the centralized localization schemes include MDS-MAP [27], RSSI-based centralized localization scheme [28] and stochastic optimization localization scheme [25, 26]. The authors in [27] introduced a 3-step centralized localization scheme which in the first step computes the shortest paths between all pairs of sensor nodes within the area of interest using such Shortest Path First (SPF) algorithms as the Dijkstra's algorithm or

Floyd-Warshall algorithm. Then the shortest path distances are then employed in the construction of the distance matrix. The second step involves the application of the MDS to the distance matrix and keeping the highest couple of eigenvalues and eigenvectors to create a 2-D or 3-D relative map which produces a location for every sensor node. The relative map is randomly flipped and rotated relative to the sensor nodes' actual positions. Finally the relative map is transformed to an absolute map which is based on the BNs' absolute positions in other to reduce the sum of squares of the deviation between the transformed positions in the MDS map and the true positions of the BNs. A benefit of the MDS approach is that BNs are not required in other to commence because it has the capability to build a relative map of the nodes with no BNs. Then with 3 or more BNs, the relative map is transformed into absolute coordinates. The MDS approach is effective in scenarios where there are low BNs ratios. A major drawback of the MDS-MAP is that the approach needs global information of the sensor network as well as the fact that computation is done centrally.

The authors in [29] proposed a Simulated Annealing approach to sensor nodes' localization in a centralized fashion. This is a 2-stage approach which minimizes the issue of flip ambiguity. First, simulated annealing is applied in other to find location estimates of the localizable sensor nodes utilizing distance limitations. Secondly, error caused by flip ambiguity is eliminated.

The RSSI-based centralized localization scheme [28] estimates of sensor nodes' locations through RF beacon attenuation in Electromagnetic (EM) waves. The three stages of the scheme comprises:

1. RF mapping of the network created by transmitting short packets at different power levels through the network and then the average RSSI value of the received packets is stored in memory tables.
2. Ranging model is created by recording the entire ordered pair between the two BNs which are then processed at the central device to correct the non-linearity and also for model calibration.
3. The optimization problem is then solved resulting in the sensor node's location using the Centralized localization model.

This scheme has the advantage of being both practical as well as self-organizing and can be used for any outdoor environment. Its drawback is its high level of power-consumption as it needs vast packet generation and must transmit significant amount of data to the central device.

2.2. Distributed localization scheme

In distributed localization schemes, computation is distributed among sensor nodes resulting in less energy being consumed as only inter node communication takes place unlike in the centralized schemes. We will further classify the various distributed localization schemes into range-based and range-free localization schemes.

2.2.1. Range-free Schemes

Range-free localization schemes [3, 5, 30, 70] do not employ range measurements in the estimation of location of a sensor node. These localization schemes are used in applications where there is no requirement to provide accurate estimates of a node's location. Some range-free schemes employ communication ranges to estimate the area in an environment that may hold a sensor node.

The authors in [30] employed a mechanism called the expected hop progress which involves a study of the hop progress from a WSN model, in other to determine the location estimate of a node in a network region. This scheme realizes better localization results and also has less communication overhead compared to other schemes like the RAW and the DV-Hop schemes.

The authors in [70] introduced an area-based range-free localization scheme known as Approximate Point in Triangle (APIT) which needs a heterogeneous network of sensor nodes where a subset of the devices, called anchors, are fitted with powerful transmitters and positioning information. In this approach, beacons from the anchors are used in isolating the network environment into triangular areas between beaconing nodes as shown in Fig. 2.2. A sensor node chooses three anchors from all the anchors it hears beacons from and tests whether or not it is located within the triangle formed by connecting to the three anchors. The algorithm repeats these tests using all combinations of three of the audible anchors. Hence the sensor node's presence within or outside the triangular regions created

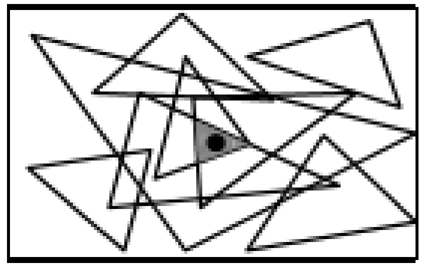


Figure 2.2 Overview of the Area-based APIT Algorithm [70]

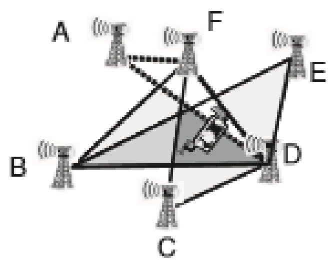


Figure 2.3 Position estimates using overlapping triangles [36]

from the different combinations of three anchors, estimates the region in which it can likely be in.

We will illustrate the idea using Fig. 2.3. Assuming the sensor node has learned that it is located within the triangles BDE, BDF and CDF, it can estimate that it is located within the dark shaded area in Fig. 2.3. To determine whether a sensor node is located within or outside a triangle formed by three anchors, a test is carried out which utilizes the intuition that when a node located within a triangle is moved in any direction, the node must come closer to at least one of the corners of the triangle more than it was prior to its movement. Conversely, if the sensor node is outside the triangle, there must exist at least a single direction for which the node's distance to all the corners of the triangle increases. Mobile sensor nodes within imaginary triangles is not practical in real-life situations. A practical way of achieving this is for the sensor node to determine from its neighboring nodes their distance to the same set of three anchors and compare the distances with the probing node's distance. Checking distances of all its neighbors, if there exists at least one corner where the neighbor is closer to the corner than the probing node, one assumes that the node is located within the triangle, otherwise it is assumed to be outside the triangle.

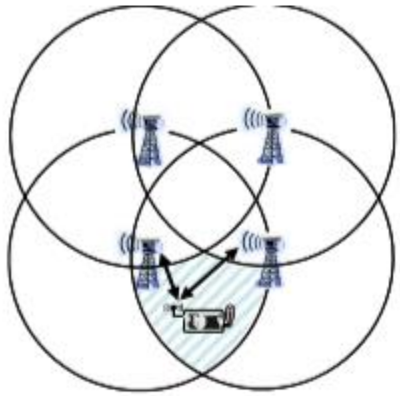


Figure 2.4 Positioning using connectivity information to multiple beacon nodes [5]

APIT has the advantage of being simple in its implementation but needs a high anchor-to-sensor node ratio and also needs longer range beacons to achieve more accurate position estimates. With low beacon density, APIT has shown not to achieve accurate results.

The authors in [5] present a scheme using the Overlapping connectivity approach, where they used only the availability of connectivity to a group of anchors or beacon nodes to estimate a sensor node’s location as shown in Fig. 2.4. They authors have made the assumption that beacon signals transmitted using a pre-determined radio transmit power can be received within a certain circular region of known radius. A sensor node on receiving beacons from anchors, determines that it is located at the center of the area where the circles around these anchors overlap. Assuming the sensor node is aware of all deployed anchors, the fact that the nodes do not receive some beacon signals may indicate that the sensor node is outside these circles and this further narrows the possible position of the node. The accuracy level of this approach is dependent on the number of anchors used. Higher percentage of anchors used in a network area results in a better location estimation.

2.2.2. Range-based Schemes

Range-based localization schemes [2, 32, 33, 34] use RSSI, time of arrival (TOA), angle-of-arrival (AOA), or time difference of arrival (TDOA) to estimate the range/distance or angle between a BN and a sensor node. Then they use trilateration or triangulation to estimate the sensor node’s position. When the distance or angle estimates are relatively accurate, these methods prove to be effective. Range-based techniques provide more accurate localization results but need additional hardware for AoA and TDoA.

Another class of beacon nodes popularly used in many sensor localization schemes is mobile beacons. A single mobile beacon node can be used instead of several static beacon nodes. The mobile node is used in obtaining the required distance or angle estimates for node position estimates. The authors in [6] introduced a distributed localization algorithm which employs a mobile BN where they have assumed that a sensor node is in an area termed the Arrival and Departure Overlap (ADO). The ADO is created by the intersection of the arrival constraint area and departure constraint area which is formed by beacon signals as the BN approaches and leaves the sensor nodes sensing range. The authors in [9] introduced the MRTP scheme which leverages the distance upper bound constraints to obtain accurate sensor location estimates in obstructed environments. Their approach is similar to that of the Centroid technique [10], however they apply the distance upper bound constraints in other to achieve additional further localization accuracy. In [2], the grid-based MLE method is used to perform localization. To perform error detection, the authors employed the MinMax scheme to overcome significant attenuation measurement errors caused by the obstructions and then use the compensated RSS measurements to further reduce the localization error.

Researchers have also studied other range-based localization schemes that combine range estimates from multiple beacons in estimating sensor nodes' locations. These approaches include the convex optimization [14], systems of complex equations [15], and the Kalman filters [16]. These schemes have the disadvantage of requiring substantial cost in computation and communications.

Probabilistic approaches [7, 8, 9, 10] have also been proposed. The authors in [7] presented a distributed scheme for outdoor deployments that recognizes distance measurement errors. They first collect and process measurement data and then create a map of the probability distribution of individual signal strength value versus the distance. At first every node in the network area is assumed to exist everywhere in the network region with equal probability and then repeatedly refines their location estimates by incorporating their current location estimates with calculated positive and negative constraints. This algorithm is novel as well as practical but has the drawback of requiring the tedious task of taking offline calibration of measured data and can also result in sensor nodes unable to self-localize. As far we know, there has not been an indoor implementation of this scheme. Authors in [9] introduced an RSSI-based object tracking algorithm used to assess the extent a sensing field is covered or observed. The authors made the assumption that the sensor node's location is known and also they computed probabilities that the network will detect a node's location. However, the authors have not stated whether their algorithm can be implemented on sensor nodes to self-localize. In [10], they authors presented a method for estimating a node's location using several sample power measurements from beacon nodes. They compute the expected value of the received power and using a steepest descent approach, they combine the computed expected value with the mean and standard deviation of the measurement samples.

2.2.2.1. Multilateration scheme

A very popular method of estimating sensor node location is the multilateration. Multilateration determines the estimated location of the node that minimizes the mean error leveraging range or angle measurements from all BNs or anchors. It is an effective method for reduction of errors in range-based localization methods. To further reduce the level of errors, the number of beacons used should be increased.

The mathematical basics for the multilateration problem is outlined below [36]. We assume a sensor network of three beacon nodes at known positions $(x_i, y_i), i = 1,.,3$, a sensor node at an unknown position (x_u, y_u) and precise distances $(r_i), i = 1,.,3$ from the sensor node to the beacon nodes. A depiction of such a scenario is shown in Fig. 1.4a.

The set of equations can be derived using Pythagoras theorem:

$$(x_i - x_u)^2 + (y_i - y_u)^2 = r_i^2 \text{ for } i = 1,.,3 \quad (2.1)$$

Setting equations 2.1 as a set of linear equations in x_u and y_u, and removing the quadratic terms x_u^2 and y_u^2 by subtracting equations 3 from equations 1 and 2, yields the following equations:

$$(x_1 - x_u)^2 - (x_3 - x_u)^2 + (y_1 - y_u)^2 - (y_3 - y_u)^2 = {r_1}^2 - {r_3}^2 \tag{2.2}$$

$$(x_2 - x_u)^2 - (x_3 - x_u)^2 + (y_2 - y_u)^2 - (y_3 - y_u)^2 = {r_2}^2 - {r_3}^2 \tag{2.3}$$

Then rearranging the terms yields the set of equations:

$$2(x_3 - x_1)x_u + 2(y_3 - y_1)y_u = ({r_1}^2 - {r_3}^2) - ({x_1}^2 - {x_3}^2) - ({y_1}^2 - {y_3}^2) \tag{2.4}$$

$$2(x_3 - x_2)x_u + 2(y_3 - y_2)y_u = ({r_2}^2 - {r_3}^2) - ({x_2}^2 - {x_3}^2) - ({y_2}^2 - {y_3}^2) \tag{2.5}$$

Rewriting the above set of equations as a linear matrix equation yields:

$$2\begin{bmatrix} x_3 - x_1 & y_3 - y_1 \\ x_3 - x_2 & y_3 - y_2 \end{bmatrix}\begin{bmatrix} x_u \\ y_u \end{bmatrix} = \begin{bmatrix} ({r_1}^2 - {r_3}^2) - ({x_1}^2 - {x_3}^2) - ({y_1}^2 - {y_3}^2) \\ ({r_2}^2 - {r_3}^2) - ({x_2}^2 - {x_3}^2) - ({y_2}^2 - {y_3}^2) \end{bmatrix} \tag{2.6}$$

The matrices on the left and right sides contains only known constants. Hence the unknown variables x_u and y_u can be calculated easily. Hence the accurate position of the sensor node can be computed assuming accurate distance estimates are known as has been described.

However getting accurate distance estimates without sophisticated distance measuring tools is not feasible in real life situations. So distance estimates $\tilde{r}$ with an unknown error ε are used for the multilateration as shown in Fig.1.4b. Solving equation 2.6 using $\tilde{r_i} = r_i + \varepsilon$ will typically not produce the accurate position (x_u, y_u) of the sensor node.

The popular approach to solving this is to use an overdetermined system of equations as shown in equation 2.7, which entails using greater than the required minimum number of BNs and distance estimates. In the above example scenario given, we will have to use more than 3 beacon nodes and distance estimates.

$$\underbrace{2\begin{bmatrix} x_n - x_1 & y_n - y_1 \\ \vdots & \vdots \\ x_n - x_{n-1} & y_n - y_{n-1} \end{bmatrix}}_{\text{A}}\underbrace{\begin{bmatrix} x_u \\ y_u \end{bmatrix}}_{\text{X}} = \underbrace{\begin{bmatrix} ({r_1}^2 - {r_n}^2) - ({x_1}^2 - {x_n}^2) - ({y_1}^2 - {y_n}^2) \\ \vdots \\ ({r_{n-1}}^2 - {r_n}^2) - ({x_{n-1}}^2 - {x_n}^2) - ({y_{n-1}}^2 - {y_n}^2) \end{bmatrix}}_{\text{b}} \tag{2.7}$$

To solve the overdetermined system of equations, a solution of (x_u, y_u) can be calculated that minimizes the mean square error (MSE), $\|\mathrm{AX} - \mathrm{b}\|_2$ where A, X and b represent the matrices as depicted in equation 2.7.

To minimize the mean square error, we utilize the square of the Euclidean norm above. For any given vector v, $\|\mathrm{v}\|_2^2 = \mathrm{v}^T\mathrm{v}$. This yields the equation below:

$$\|\mathrm{AX} - \mathrm{b}\|_2^2 = (\mathrm{AX} - \mathrm{b})^T(\mathrm{AX} - \mathrm{b}) = \mathrm{X}^T\mathrm{A}^T AX - 2\mathrm{X}^T\mathrm{A}^T b + \mathrm{b}^T\mathrm{b} \tag{2.8}$$

Minimizing the mean square error, i.e. getting the directive of the expression with respect to x and setting it equal to zero yields the normal equation below:

$$2\mathrm{A}^T AX - 2\mathrm{A}^T b = 0 \Leftrightarrow \mathrm{A}^T AX = \mathrm{A}^T b \tag{2.9}$$

Equation 2.9 has a unique solution if matrix A has full rank. However other methods, such as the Cholesky or QR factorization, can yield a solution to the equation.

To find a solution to the issue of computationally complex minimum mean square error (MMSE) especially when all the multiple range estimates may not be available at every sensor node, other approaches like iterative and collaborative multilateration [1] have been identified for sensor network.

- Iterative and collaborative multilateration

The typical multilateration approach estimates distances between unknown nodes (UNs) and BNs (or anchors) in order to apply multilateration on the BNs. As introduced in the previous subsection, there may arise situations where there are insufficient number of BNs for node location estimate. In such situations, normal or unknown nodes after being localized, can be used as “Pseudo beacon nodes” and act as BNs in the location estimates process. We have an example scenario in Fig. 2.5 [36-37] which shows 3 UN: A, B and C which do not know their positions. Node A can determine its position using 3 of the available BNs and after Node A localizes, then node B can uses the now localized node A and 2 other BNs to localize itself and then be available for node C to localize itself.

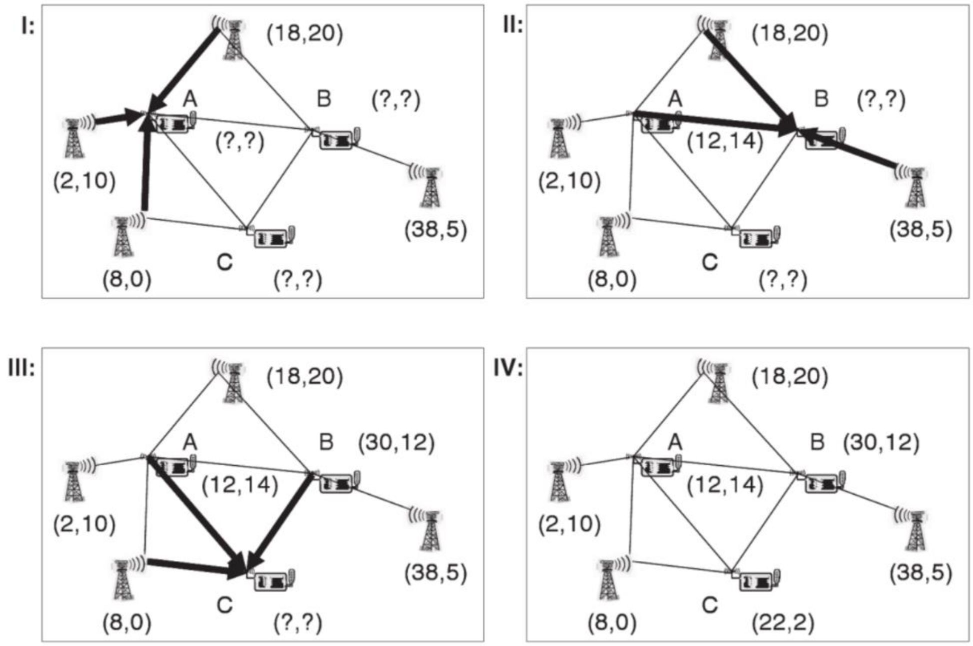


Figure 2.5 Iterative multilateration showing UNs A, B and C that can find their positions in multiple iterations. The thick arrows depict the beacon signals whose distance estimates are employed in a given iteration [36].

The accuracy of this iterative process depends on the following:

- Accuracy of the range estimation – the more accurate the range measurements are, the better the position estimate of the UNs.
- The initial position estimate – if the position estimate of the first UN that becomes a pseudo BNs is erroneous, the subsequent UNs will be using a pseudo BN whose position is inaccurate and this will result in other UNs' positions being inaccurate and the inaccuracies are propagated down to the last UN localized.
- Average number of neighbors – this is very critical as if at any step in the iterative multilateration process, then some UNs may not have sufficient BNs and/or pseudo BNs to use for localization.
- Number of BNs – If there are insufficient BNs in the initial location estimate of the first UN, then there will not be pseudo BNs to be used for other localizations down the iterative multilateration process.

The problem of insufficient BNs for location estimation can be addressed using the process of collaborative multilateration to estimate some of the UNs that ordinarily could not have been localized. Basic guidelines of the collaborative multilateration process include: ensure high connectivity, use a minimum of 5% BNs and to place BNs toward the perimeter of a network [37].

Collaborative multilateration can be effective in assisting the iterative multilateration in areas of the network where the number of BNs is low and the atomic multilateration requirement is not met [37]. The advantage of the iterative- and collaborative-multilateration methods is that they can be employed in the localization of nodes in areas of the network area where there are insufficient BNs or nodes who can act as pseudo nodes. Their drawback is that of error propagation across the node localization process.

Drawbacks of multilateration

Multilateration method of sensor node localization has inherent disadvantages in that they utilize distance estimates from beacons received from all BNs, shadowed and unshadowed beacons in localizing nodes. This results in the sensor node location estimate often being severely inaccurate. The intuition will be that any scheme that identify and deselect those distance estimates calculated from shadowed beacon signals, then the resultant node location estimate will be significantly less inaccurate. Substantial amount of study in this area has been conducted by researchers in designing algorithms that will mitigate the effects of shadowed beacons [43, 44].

2.2.2.2. Majority Rule in Wireless Sensor Localization

To counteract the challenge experienced with using multilateration in sensor localization due to erroneous distance estimates caused by shadowed beacon signals, researchers have developed various techniques to deal this. One of such techniques is called the majority rule method.

The authors in [45] proposed a majority rule-based sensor fusion system with particle filter to filter and suppress noisy sensor measurements. In their scheme, the authors have made the assumptions that biased distributions, which are different from distributions from other low noise sensors, are detected and subsequently removed using a majority rule approach.

2.2.2.3. Outlier detection in Wireless Sensor Localization

The other approach that has attracted significant work in the wireless sensor localization research community is the detection, elimination and suppression of outlier sensor measurements. Outlier measurements can be considered to be those measurements that greatly differ from the typical pattern of sensed data.

Several methods have been developed by researchers to detect, remove or suppress outlier beacon signals caused by noisy measurements [45,46]. The authors in [45] employed the graph embeddability employing rigidity theory to detect and filter outlier range measurements. The authors made the assumptions that measurements unaffected by noise are accurate and this is not practical with RF RSSI measurements. They also did not state whether there is an outdoor implementation of their scheme. In [46], the authors leveraged the closeness of sensor nodes in a network, employing the term called Closeness Centrality (CC) to represent the significance of a node within the network compared to other nodes. They then used the CC to remove those nodes with noisy measurements from being used in the estimation of the UNs' locations. The drawbacks of the algorithm are that it needs training data and also there was no explanation by the authors whether or not the clusters formed using the CC reduces or removes the adverse effect of obstructed signals.

2.2.2.4. Simultaneous Localization and Mapping (SLAM)

A robot can be tasked to estimate its position in an environment using a map of landmarks and inbuilt sensors that takes measurements of these landmarks. This is termed Localization. Also, if given accurate positions of the robot within the environment, a map of the landmarks can be obtained. This is termed Mapping. Then localizing the robot positions and at the same time build a map of the landmarks is termed Simultaneous Localization and Mapping (SLAM). The SLAM algorithm is a very important localization method in Robotics as can be seen in the works shown in [17, 19, 20, 22, 23]. When the inbuilt robot sensor described above can take only range measurements, a special case of range-only SLAM emerges [18, 21, 47, 48]. Researchers have varying implementations of the SLAM using mechanisms like the Extended Kalman Filter (EKF), sparse graphs, Sparse extended information filter (SEIF) and particle filter (PF).

The EKF SLAM algorithm has a number of limitations namely its quadratic computational complexity as well as the linearization technique. With the GraphSLAM [50], it accumulates information and is basically an offline algorithm. To develop an online filter algorithm that incorporates the benefits of the information filters, the SEIF was introduced. The SEIF SLAM maintains an information view of all knowledge making it run online as well as be computationally more efficient than the EKF and GraphSLAM. In FastSLAM [42, 49], particle filter is used in estimating the robot path and the map features are estimated by EKFs but using separate low-dimensional EKF for each feature. The advantages of FastSLAM include its logarithmic time complexity in the number of features and data association can be made on a per-particle basis. Another very important advantage is that, unlike the other techniques that require approximations of the non-linear motion model using linear functions, particle filters can handle non-linear robot motion models.

Our correlation-based localization approach is different from the described schemes because it uses the idea that not all beacon signals are adversely affected by obstructions located in the network area. It improves on the linear least square method by removing the effects of the obstructed beacon signals. It attempts to combine the RSSI measurements that contribute to a high degree of agreement towards the final location estimate. It achieves this by utilizing spatial correlation of candidate sensor location estimates that are computed using different combinations of a subset of all beacon signals in a trilateration process. Furthermore, our approach does not depend on the deployment environment as the nodes can dynamically calculate path loss exponent of the sensing region using beacon signals received from BNs.

CHAPTER 3: PERFORMANCE OF SLAM IN THE PRESENCE OF SHADOWING

A very popular sensor localization approach that has a similar weighting mechanism (Kalman Gain) for node position estimation is the simultaneous localization and mapping, popularly called SLAM. The SLAM has seen a rapid progress over the past decade. The problem of learning maps is a fundamental problem in mobile robotics. SLAM involves building a map of an environment using a mobile robot or platform and at the same time, localize the mobile robot in the environment. SLAM is a Chicken-or-Egg problem: a map of the environment is required to localize the robot and also knowledge of the robot poses is required to build a map of the environment. Hence, SLAM is regarded as a hard problem in robotics. Figure 3.1 illustrates the essential SLAM problem.

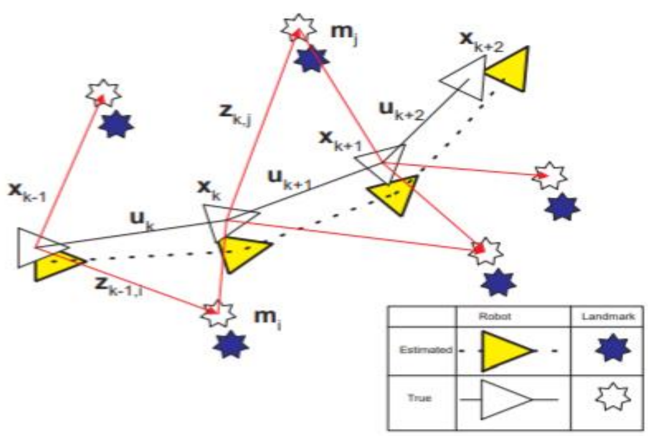


Figure 3.1: The true positions are unknown and are not measured directly but information from observations made between true robot and landmark (sensor node) positions are used to determine a simultaneous estimate of the robot and sensor node's positions. [17]

Before delving in depth of the SLAM problem, we will describe the following problems that often times are the objectives of SLAM algorithms, i.e. the problem that the SLAM algorithm wants to address:

- Localization – determine the position(s) of a robot/mobile node given a map.
- Mapping – determine the map (i.e. sensor nodes' locations) given the robot/mobile node's locations.
- SLAM – determine the map and at the same time the positions of the robot.

3.1. SLAM: Overview

With sensor localization, the positions or path of the mobile robot are determined using the known locations of the map/landmarks/sensor nodes. The converse is the case whereby the locations of sensor nodes within a sensor region are determined given that the positions or path of the robot within that sensor region are known.

Mapping is more relevant to the correlation-based localization scheme we have proposed. In SLAM using the EKF filter, a mechanism called the Kalman Gain is applied to reduce the effect of obstructed beacon signals by determining the contribution of such beacon signals to the state space estimation. This is similar, in principle, to our proposed localization scheme where the spatial-correlation mechanism is used to reduce the effect of obstructed beacon signals. The spatial-correlation applies a weighting function to give higher significance to those candidate estimates that are spatially correlated while giving less significance to those that are spatially uncorrelated. This is the motivation for our choosing to explore SLAM and compare it to our proposed scheme.

Let us take a scenario where a mobile robot with an onboard sensor is traversing an environment and taking measurements of some unknown landmarks or unknown nodes as shown in Fig. 3.1. At a time instance k, the following quantities are defined:

- $x_k -$ The state vector containing the robot's location and orientation.
- $u_k -$ The control vector.
- $m_i -$ A vector containing the location of the i^{th} landmark.
- z_{ik}: An observation of the i^{th} landmark measured from the robot at time instance k.

Representations of the observation model as well as the motion model are achieved by computing the prior and posterior distributions using probabilistic approaches. These algorithms are greatly affected by both measurement data and data association.

The two main forms of the SLAM problem in the probabilistic perspective are the full SLAM and the online SLAM. The full SLAM aims to estimate the entire path of the robot and also the map of the environment provided that all the control inputs and all the measurements are known. The full SLAM calculates the posterior over the entire robot path and the map instead of the current pose of the robot.

$$bel(x_{0:k}, m) = p(x_{0:k}, m \mid z_{0:k}, u_{0:k}) \tag{3.1}$$

The graphical representation of the full SLAM is shown in Figure 3.2.

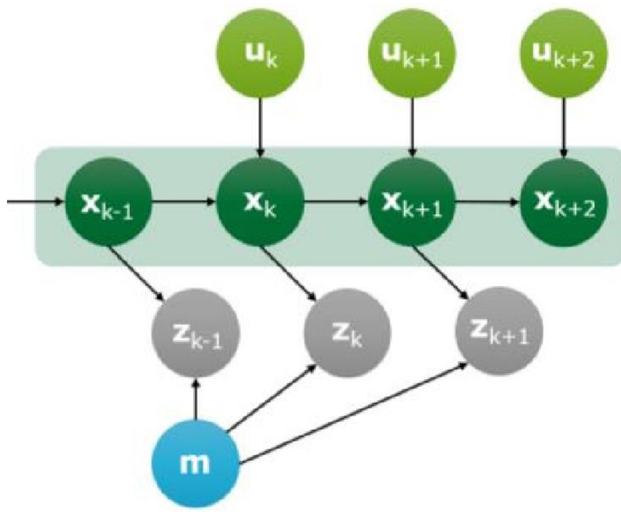


Figure 3.2: Graphical representation of the full SLAM problem [23].

Due to the growth in complexity as a result of the number of variables considered, the full SLAM problem can become very challenging to deal with in real time. To forestall the complexity growth problem of the full SLAM, the online SLAM is used. The online SLAM estimates the current position of the robot, given that the previous measurement information is known. Fig. 3.3 illustrates the graphical illustration of the online SLAM.

The Bayes' rule can be used to show the incremental nature of the problem as shown in below:

$$\begin{aligned} bel(x_k, m) &= p(x_k, m \mid z_{0:k}, u_{0:k}) \\ &\propto \; p(z_k \mid x_k, m) \int_{x_{k-1}} p(x_k \mid x_{k-1}, u_k)\, bel(x_{k-1}, m)\, dx_{k-1} \end{aligned} \tag{3.2}$$

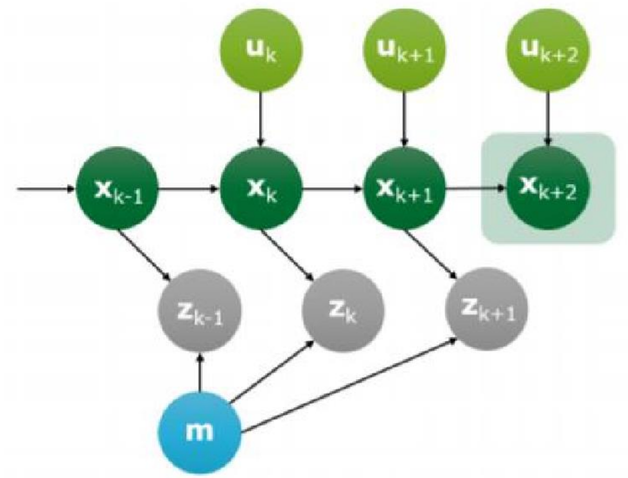


Figure 3.3: Graphical representation of the online SLAM problem [23]

There are two main categories of the estimation techniques used in SLAM. One is the filter-based method which we will discuss here. The other category, which we will not discuss in this dissertation, is the optimization-based method.

Filter-based SLAM

Filter-based SLAM methods are Bayesian filters and involve two-step iterative processes described below:

- Prediction phase: The prediction phase involves predicting the robot and map states. The prediction is made using the motion model as well as the control inputs u_k.
- Update/correction phase: In this phase, the current sensor observation z_k is compared to the map in other to correct the prediction made in the first phase. The model that relates the observation to the map is known as the observation model.

The prediction and update steps continue iteratively, incorporating measurement data as the robot moves along its path, to finally estimate the robot position as well as the map of the environment. A few of the filters used in filter-based SLAM are the Extended Kalman Filter (EKF), Unscented Kalman Filter (UKF) , Particle Filter (PF), Information Filter (IF) etc. The EKF, UKF and IF are examples of Gaussian Filters which are an essential family of recursive state estimators. On the other hand the PF belongs to the Non-Parametric family of filters.

We will discuss the EKF and PF variants in more detail. To describe the EKF, UKF and IF filters, we will introduce and describe the Kalman Filter (KF) which is the best studied technique for the implementation of Bayes filters. The KF is a recursive Bayes filter with prediction and correction steps but implements belief computation for non-discrete as well as non-hybrid state spaces. A mathematical representations of the 2 steps of the KF are shown in equations 3.3 and 3.4.

1. Prediction:

$$\overline{bel}(x_k) = \int p(x_k|u_k, x_{k-1}) bel(x_{k-1})\, dx_{k-1} \tag{3.3}$$

2. Measurement Update/Correction:

$$bel(x_k) = \eta\, p(z_k|x_k)\, \overline{bel}(x_{k-1}) \tag{3.4}$$

In the KF algorithm, the posterior must be Gaussian and the state transition probability must be a linear function in its argument but with added Gaussian noise. Table 3.1 shows a depiction of the KF algorithm.

Algorithm 1 shows the steps of the KF Algorithm:

Algorithm 1: $Kalman\_Filter(\mu_{k-1}, \Sigma_{k-1}, u_k, z_k)$:

1: KF_Prediction:
2: $\bar{\mu}_k = A\mu_{k-1} + Bu_k$
3: $\bar{\Sigma}_k = A_k \Sigma_{k-1} A_k^T + R_k$
4: KF_Measurement:
5: $S_k = \bar{\Sigma}_k C_k^T \,(C_k \bar{\Sigma}_k C_k^T + Q_k)^{-1}$
6: $\mu_k = \bar{\mu}_k + S_k\,(z_k - C_k \bar{\mu}_k)$
7: $\Sigma_k = (I - S_k C_k)\,\bar{\Sigma}_k$
8: $\boldsymbol{return}\ \mu_k, \Sigma_k$

To better understand the KF, we will define some common and vital parameters of the KF shown in the algorithm above:

- Kalman Gain – This parameter represented by the variable S_k in line 5 of the KF algorithm specifies the degree to which the measurement influences the estimation of the new state. The mean is manipulated, adjusting it in proportion to the Kalman Gain and the deviation between the predicted and actual measurements.
- Innovation – This is the deviation between the predicted measurement $C_k\bar{\mu}_k$ and the actual measurement taken z.

One of the critical and essential attributes that must exist for a KF to be complete is that observations are linear functions of the state and are linear functions of the previous state. However, in real life situations, state transitions and observations are mostly non-linear. As a result of this reality check and also the assumption of unimodal beliefs, the KF is really in applicable in most applications.

To forestall the linearity requirement of the KF, the EKF was introduced whereby the state transition probability as well as the measurement probabilities are ruled by nonlinear functions. The EKF approximates the basic belief representation of the KF and uses linearization to approximate the non-linear function.

We will now describe the EKF in more detail.

i. Extended Kalman Filter (EKF)

At its basic form KFs are developed to tackle linear systems and even though they have remarkable convergence properties, they are hardly used in SLAM because of that fact that real live processes are usually not linear in nature. On the other hand, the EKF is commonly used in non-linear filtering and so is used in SLAM. To overcome non-linear models, the EKF linearizes non-linear models. The linearization is done around the current estimate using a first order Taylor expansion. The EKF is effective provided the linearization is done around the true value of the state vector.

The various steps of the EKF SLAM are: state prediction, measurement prediction, measurement reading, data association and update.

Algorithm 2 shows the steps of the EKF Algorithm:

Algorithm 2: $Extended\_Kalman\_Filter(\mu_{k-1}, \Sigma_{k-1}, u_k, z_k)$:

1: EKF_Prediction:
2: $\bar{\mu}_k = g(u_k, \mu_{k-1})$
3: $\bar{\Sigma}_k = G_k \Sigma_{k-1} G_k^T + R_k$
4: EKF_Measurement:
5: $S_k = \bar{\Sigma}_k H_k^T (H_k \bar{\Sigma}_k H_k^T + Q_k)^{-1}$
6: $\mu_k = \bar{\mu}_k + S_k (z_k - h(\bar{\mu}_k))$
7: $\Sigma_k = (I - S_k H_k) \bar{\Sigma}_k$
8: $\boldsymbol{return}\ \mu_k, \Sigma_k$

ii. Particle Filter (PF)

The PF is a nonparametric and major approach in filtering SLAM algorithms. Particle filtering is an approach used for implementing recursive Bayesian filtering by Monte Carlo sampling. The major idea is to represent the posterior density at time k $p(x_k|z_{1:k})$ by a set of independent and identically distributed (i.i.d.) random particles $\left\{x_k^{(i)}\right\}$ according to the distribution. Every particle is paired with a weight $w_k^{(i)}$. Successive measurements and model-based predictions are then used to update the weights and particles.

Unlike for the EKF, the PF is very effective for non-Gaussian stochastic processes with non-linear dynamics and especially beneficial when the posterior $p(x_k|z_{1:k})$ does not have any parametric form or that the form is not known.

The PF algorithm involves the following steps:

- Filter initialization (The prior model)

  The filter is initialized on the reception of the first beacon signal from the robot. An initial distance and a corresponding variance on the distance are estimated from RSSI values. The prior considered is thus a uniform distribution on a spherical annulus, where the inner and outer radii are dependent on the estimated mean and variance.

- Prediction step

  The probability that at time step k, the unknown node is at position P_k given that it was at position P_{k-1} at the time step k-1. If static unknown nodes are employed, then the prediction step might be omitted (that is, with probability 1 each node is in the same position at time steps k and $k-1$). If that is the case, since there is a limited number of particles over the state space, a random move is introduced to the particles in order to search locally around the position of the previous time step.

- Update step

The functions $\mu(d)$ and $\sigma(d)$ are applied in the estimation process. Every time a new measurement is received, the weights of each particle is updated putting into consideration the likelihood of the received measurement.

- Resampling

  In particle filter localization, the weights of particles with high likelihood increases and those particles with low likelihood do not. Since the number of particles are limited, a resampling step is introduced in order to replicate particles with high weights and remove those with low weights. Doing so, particles with high likelihood (high weights) contribute towards the final estimation of the unknown node while ignoring particles with low likelihood (low weights).

  To deal with some of the obvious issues of the resampling step, two considerations are made: first, resample only when the effective number of particles N_{eff} goes below a threshold. The effective number is calculated using equation 3.5:

$$N_{eff} = \left[\sum_{i=1}^{L} \left(w_k{}^{(i)}\right)^2\right]^{-1} \tag{3.5}$$

  The second consideration is to use a low variance sampler which allows the spread of the particles over the areas of highest likelihood.

- Mean and standard deviation estimation

  The mean and standard deviation from the particle filter is computed as follows:

$$\mu_k = \sum_{i=1}^{L}\left[x_k^{(i)} w_k^{(i)}\right] \tag{3.6}$$

$$\sigma_k^2 = \sum_{i=1}^{L}(x_k^{(i)} - \mu_k)^2 w_k^{(i)} \tag{3.7}$$

The key advantage of the particle filter is that it can be used for multi-modal or nonparametric distributions. While the posterior distribution depends on the measurements during the transient state, the filter approximately converges to a Normal distribution at the position of the node. It can be considered that the filter converges when σk is below a certain threshold during a period of time.

There are several approaches to solving the SLAM problem:

- Range-bearing SLAM - Range and bearing measurements are used as parameters to localize an unknown node.
- Range-only SLAM – Range measurements only are used.
- Bearing-only SLAM - Bearing (or angle) measurements are used.

Since RSSI used in range estimation is the only measurement data the sensors we have investigated take, we will focus on the solution of the localization problem using range-only SLAM in this dissertation,

3.2. Range-Only SLAM (ROSLAM)

Range-Only SLAM, popularly known as ROSLAM, is an approach to the solution of the SLAM problem [18, 21, 41, 47, 48]. In scenarios whereby the embedded sensor on a robot can only detect range to landmarks, the special case of ROSLAM, whereby only range measurement is used as parameter in estimating the robot path and also localizing a landmark. This is unlike the more popular Range-bearing SLAM which requires not only range measurements but bearing measurements as well.

The ROSLAM is a very challenging SLAM problem and this is attributed to these characteristics: the presence of outliers due to the types of sensors used (typically radio pulses or sonar), and more notably the high ambiguity of the measurements [18]. Fig. 3.4 illustrates this issue. Here a robot determines the distance to the beacon shown from three separate positions moving along a straight path. For every position the beacon node takes measurement at, an annular-shaped region of the estimated position of the beacon exists given the measurement. The problems that arise with the sensors are:

i. The large area in the network environment where the BN could be located, given only a single measurement observation.
ii. Multiple measurements are required to fully initialize a BN's position in the state vector of the filter [66].
iii. The possibility of multiple locations that the beacon node could be as in Fig. 3.4.

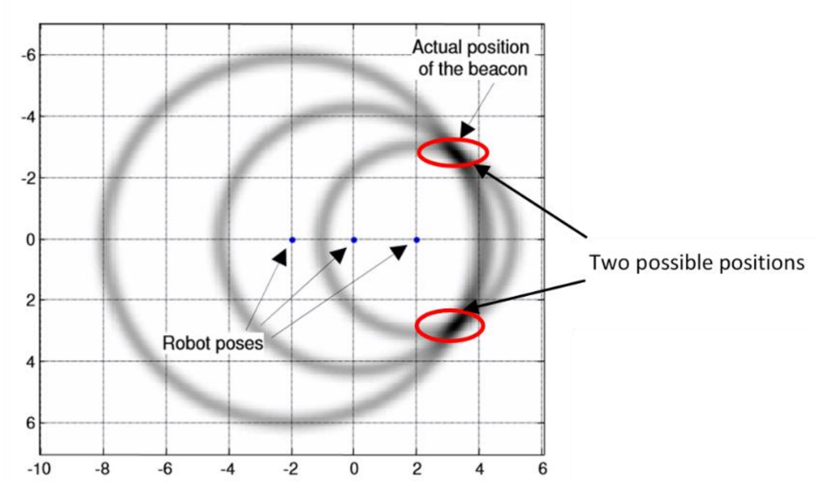


Figure 3.4: Range-Only SLAM: Map estimates may result in multi-modal densities [18].

A few of the advantages of the ROSLAM approach include:

- There is no data association problem, since the beacon nodes are capable of identifying themselves.
- Depending on the technologies, there is no need for line-of-sight (LOS) between beacon node and landmark.

To clearly describe the ROSLAM localization algorithm, we will discuss the EKF ROSLAM [66]. EKF-SLAM algorithm uses RF beacons with no prior knowledge of their position in the sensor network area. Here, the BNs provide only range information, hence estimating their positions is a challenging task.

In SLAM problems, one of the very crucial steps of the filter mechanism (EKF in this case) is to initialize the filter with an initial position of the landmark (unknown node in this case). With Range-bearing SLAM, it is easy to initialize an unknown node because the measurement function is bijective for range-bearing. Initialization can be achieved using equation (3.8).

$$\underbrace{\begin{pmatrix}\bar{\mu}_{j,x}\\ \bar{\mu}_{j,y}\end{pmatrix}}_{\text{Observed position of landmark j}} = \underbrace{\begin{pmatrix}\bar{\mu}_{k,x}\\ \bar{\mu}_{k,y}\end{pmatrix}}_{\text{Estimated position of the robot}} + \underbrace{\begin{pmatrix}r_k^i \cos(\ \phi_k^i + \bar{\mu}_{k,\theta})\\ r_k^i \sin(\ \phi_k^i + \bar{\mu}_{k,\theta})\end{pmatrix}}_{\text{Relative measurement}} \tag{3.8}$$

However, for ROSLAM, to be able to estimate the initial position of the UN, multiple range measurements have to be taken. For example, two range measurements from two robot's positions can be used to initialize two hypotheses for the BN's location, and are then entered into the filter as a Gaussian mixture. A detailed description of the ROSLAM implementation will be outlined in the next section.

3.3. ROSLAM Implementation

In this section, the ROSLAM implementation using simulations is outlined.

A. System Overview

In this ROSLAM implementation, we considered a mobile robot moving in a 40m x 40m outdoor environment. The robot is equipped with wheel encoders and gives travelled distance Δu (in meters) as well as change in orientation $\Delta\theta$ (in rad) between two time steps. However, the odometry measurement is perturbed with Gaussian noise.

The mobile robot travels in a circular region round the network region and only stops at certain locations to transmit beacon signals before continuing in its circular path. For the simulations, we considered different number of locations (6, 8, 10 and 12) from where the robot sent beacon signals. An unknown node is placed at an unknown location within the network area, $\mathbf{x} = (x, y)^T$.

i. State vector Initialization:

The state vector at time k where every beacons are initialized is of the form:

$$X_k = (x_k, y_k, x_k, \mathbf{x})^T \tag{3.9}$$

Where $(x_k, y_k, x_k)^T$ is the pose of the robot at time step, k.

After filter initialization, the next step is the prediction step and is described next.

ii. Prediction step:

Only the state variables of the robot are affected by the prediction. Equation 3.10 describes this step:

$$\begin{cases} x_{k+1} = \; x_k + \; \Delta u cos(\theta_k + \frac{\Delta\theta}{2}) \\ y_{k+1} = \; x_k + \; \Delta u sin(\theta_k + \frac{\Delta\theta}{2}) \\ \theta_{k+1} = \; \theta_k + \; \Delta\theta \end{cases} \tag{3.10}$$

The measurement noise matrix in an EKF ROSLAM

$$Q = \begin{pmatrix} K_{\Delta u}|\Delta u| & 0 \\ 0 & K_{\Delta\theta}|\Delta\theta| \end{pmatrix}$$

Since the specific locations where the robot sends beacons is known in our implementation, the noise matrix will an 2 x 2 all zeros matrix.

iii. Update step:

When the beacon's position converges to a unique solution, a Mahalanobis test is done before the update in order to drop those range measurements that are farther away from the prediction. The prediction is then computed. Also the covariance matrix for the Mahalanobis distance is calculated using the filter's innovation covariance matrix [66].

3.4. ROSLAM: Performance results in the presence of shadowing

We now present the performance results we achieved of simulating the ROSLAM algorithm in the assumed WSN environment that is characterized by unknown obstructions. We evaluate the performance results of the ROSLAM algorithm in compared with that of multilateration, in our assumed system model which is a WSN characterized by obstruction. When there is an obstruction blocking a BN's path to an UN, we add an additional attenuation is given in equation 1.3. We evaluated for varying degree of attenuation caused by obstructions as well as for varying number of obstructions.

We have also evaluated the ROSLAM using varying number of robot steps to show that with increased number of robot steps, the ROSLAM performance improves. By "number of robot steps", we refer to the number of unique positions in the network area where the robot transmits beacon signals. This is different from the number of beacon signals transmitted by a robot as the later may involve beacon signals transmitted by a robot from the same spot.

Table 3.1: ROSLAM Simulation Parameters.

Simulation Parameters	
Network Region	40m x 40m
Number of Beacon node positions	10
Position of unknown node	{20,20}
Number of obstructed beacon nodes	1, 2 and 3
Line of sight (LOS) fading factor	1 dB
Additional attenuation due to obstructions	4 dB
Transmit Frequency	900 MHz
Path Loss Exponent (n)	2.3
Number of trials (or runs)	10

The simulation parameters are shown in Table 3.1 and we have simulated a 40m x 40m wireless sensor network area, with 1 mobile BN transmitting at 10 different positions, and one UN placed position {20,20}. We have assumed the network region contains one, two and three obstacles. These obstacles obstruct RF beacon signals and we have modeled these as describe in section 1.2.2.

i. Error performance of ROSLAM compared with Multilateration:

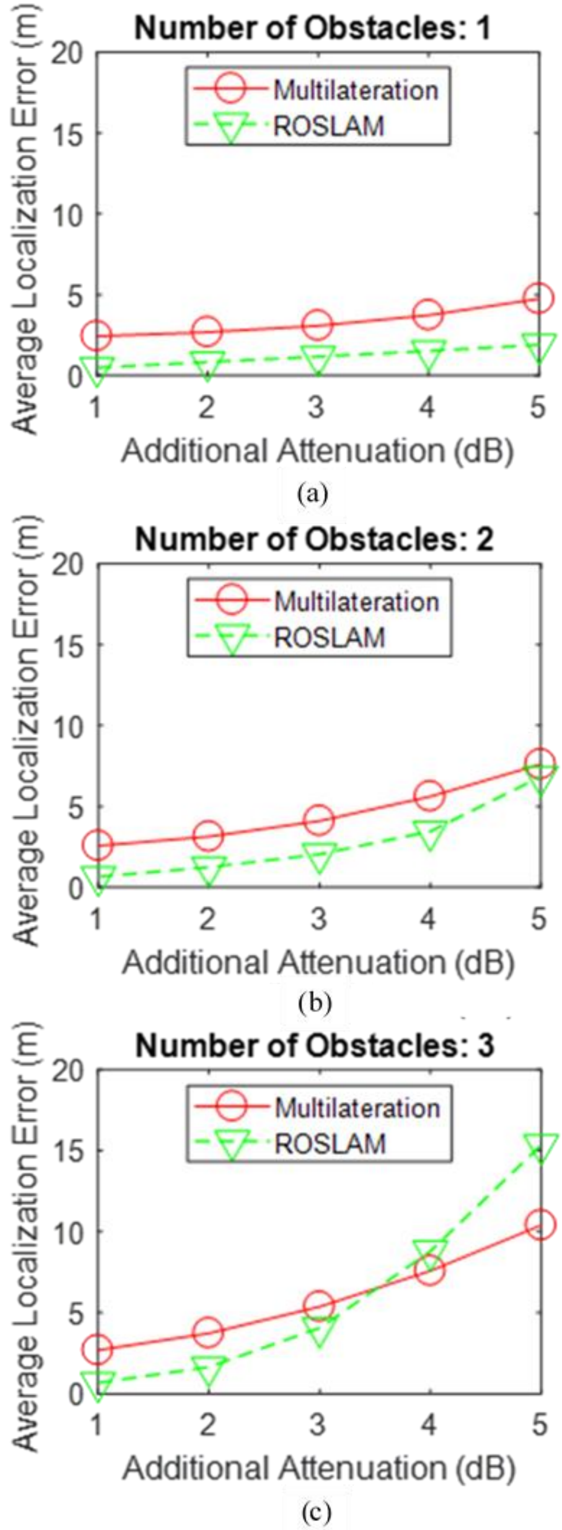


Figure 3.5: Localization accuracy comparison Multilateration versus ROSLAM using varying number of obstacles: 1, 2 and 3.

ii: Effect of number of robot steps on ROSLAM localization accuracy:

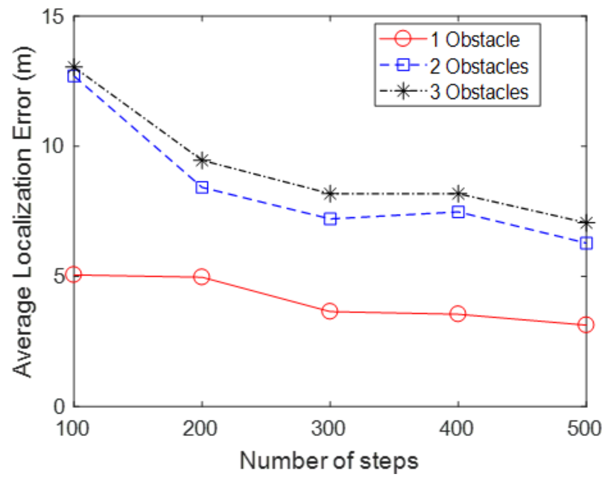


Figure 3.6: Shows the effect of the number of robot steps on ROSLAM localization accuracy

As we stated earlier, to evaluate the effect of the number of robot steps in ROSLAM, we simulated the robot steps as the number of unique positions from which the robot transmits beacon signals. The robot moves from location to location and at each location, it transmits beacon signals containing its location. So for 100 number of steps, the robot transmits beacons from 100 unique locations in the network area. This is same for 200 up to 500 number of steps.

The performance results in Fig. 3.5(a) show that ROSLAM outperforms the multilateration scheme for all values of additional attenuation factor evaluated for the 1-obstacle case. However, for the 2-obstacles and 3-obstacles cases, the multilateration method performs worse than the ROSLAM but for the case where there are obstructed beacon nodes and the added attenuation reaches 5dB as shown in Fig. 3.5(b)-(c). The cause of the dismal performance of the ROSLAM is that the small number of unique steps taken by the robot (BN) is not sufficient to effectively localize the unknown node. With increase in the number of unique steps taken by the robot (e.g. 200 and above), the localization accuracy of the ROSLAM will improve. Fig. 3.6 shows that the localization accuracy improves when the total number of unique steps taken by the robot grows from 100 to 500.

3.5. Performance Evaluation: Conclusion

A summary of our findings from simulations shows the following: ROSLAM scheme outperforms the multilateration method as the added attenuation fading factor increases. However, with a small number of unique positions where measurement data is transmitted and with a high number of obstacles, ROSLAM localization performance deteriorates.

However, increasing the number of unique robot steps improves the localization accuracy of the ROSLAM. The reason for the improvement in performance with increased number of BN/robot steps is that an increase in the number of steps introduces more unique measurement data to be used in refining the UN's location estimates until the system converges.

CHAPTER 4: SENSOR LOCALIZATION USING MAJORITY RULE

Our work is motivated by the knowledge that multilateration using RSSI-based distance estimates is extremely affected by shadowing effects that adversely affects some of the RSSI measurements more than others. This results in highly erroneous location estimates of an UN's position in a sensor region. The main problem of the use of multilateration in sensor localization is that multilateration uses beacon signals from all BNs, obstructed and unobstructed beacon signals alike. It will be very useful if there was a way to identify and discard those beacon signals that are shadowed and use only unshadowed beacon signals in sensor localization. However, it is overly challenging or even downright impractical to identify which received beacon signals by an UN are shadowed and those that are not. To address this issue, in this chapter we present an approach that uses a spatial correlation mechanism to select a fraction of a large number of received beacon signals to implement multilateration. The spatial correlation mechanism introduces the concept of Majority Rule in sensor node localization [6, 45, 46]. Majority rule (or Consensus) approach has engendered tremendous research interest as it has shown great promise in its effectiveness in sensor location estimation. This is especially the case in WSNs that experience shadowing due to obstacles present in the sensor region. This approach uses spatial correlation concept to remove those RSSI beacon signals that are adversely affected by obstructions.

The idea behind the Majority Rule approach is to use those beacon signals that contribute to a location estimate matching the majority of the beacons completely disregarding the beacons that tend to disagree with the majority. This is fundamentally different from the multilateration approach that considers all beacons as equally significant and generates a location estimate that is the minimum mean square error location calculated from all the beacon signals. It is beneficial to observe that the proposed approach is likely to work if two conditions are met. First, if the number of erroneous or NLOS beacons is not too large so that the majority is evident. Second, since all beacon signals have some error due to fading, a mechanism to determine the agreement with other beacons signals within a given margin of error exists.

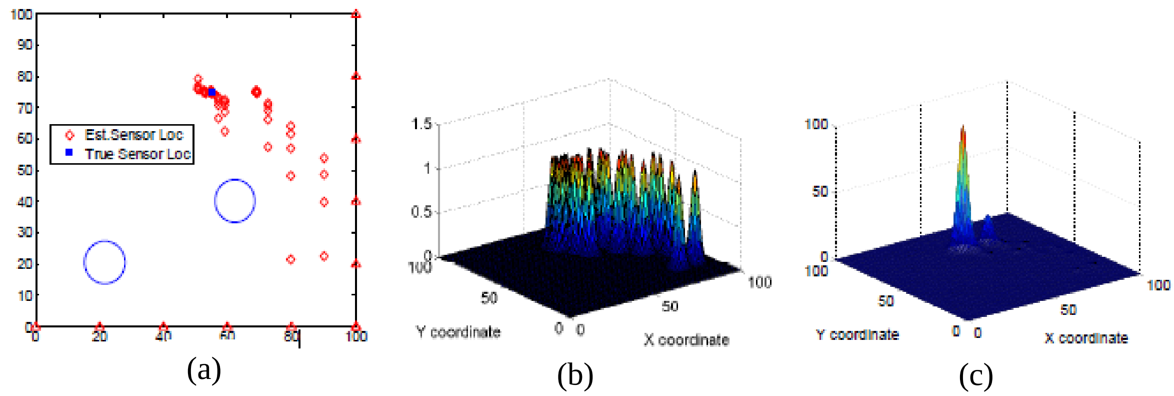


Figure 4.1: Depiction of our correlation based localization approach: (a) Network topology where blue circles represent obstructions, triangles represent beacon nodes' positions, and diamonds represent corresponding candidate location estimates from subsets of 3 beacons; (b) bivariate Gaussian pdfs with mean centered at the candidate location estimates; and (c) summation of the Gaussian pdfs, shows final estimate of node's location [6].

In this chapter, we will propose a spatial correlation mechanism that uses location estimates from small subsets of beacon signals and applies a majority rule on these location estimates.

4.1. Approach for imparting spatial correlation

As discussed in the earlier section, the reason for the introduction of the spatial correlation concept is to solve the problem of deselecting or removing those beacon signals that we do not want to have high significance in the solution of the UN localization problem.

We illustrate the spatial correlation concept using Fig. 4.1. We assume there are B number of BNs deployed in the sensor network area and these BNs will be used in localizing the unknown nodes (UNs). The BNs transmit RF beacon signals, and each of these beacon signals contain the ID as well as the location of the BN from which it was transmitted. The scheme takes subsets of M BNs ($M \ll B$) and implements multilateration to every subset to achieve as many as $\binom{B}{M}$ multilaterations for each UN. The scheme then applies a spatial correlation mechanism to choose the location where a majority of the multilateration results are in agreement. This phenomenon is termed Majority rule.

Fig. 4.1(a) illustrates an example of a network area that is 100m x100m square and is served by 11 BNs deployed in an L-shaped manner and 20m apart from each other. We consider 2 obstacles and 1 UN in locations as indicated. We have made the assumption that the RF signals transmitted by the BNs experience lognormal fading and the RF signals from BNs that are obstructed by obstacles experience shadowing effect in addition to fading. Assuming $M = 3$, there can be $\binom{11}{3} = 165$ distinct subsets of BNs, and the corresponding multilateration results, as obtained from simulations, are depicted by the red diamonds in Fig. 4.1 (a). Note that each of these multilaterations generate an estimated location of the UN that is expected to have an error that depends on the errors in the estimated distances from the BNs used in the corresponding subset. To illustrate these errors, we represent each location estimate by a bivariate Gaussian pdf of the estimated location centered at the corresponding location (Fig. 4.1(b)). The variance of the Gaussian pdfs will depend on the error in the distance estimates, i.e. a high fading rate will lead to a wider Gaussian pdf. It is expected that the location estimates that are obtained from unshadowed or LOS BN signals would be relatively consistent, i.e. located close to each other. Moreover, those location estimates from any subset that includes one or more shadowed BN signals will have much higher errors and will be widely dispersed on the 2-dimesional plane. This phenomenon is evident in the simulation results illustrated in Figures 4.1(a) and 4.1(b), where a number of location estimates are found to be close to the UN and several others, potentially from subsets of BN of which at least one BN is shadowed, are further away. Consequently, taking the summation of all the bivariate Gaussian pdfs result in superposition of the pdf that are close together (agree) while those pdfs that do not superimpose (disagree) can be ignored. This results in achieving spatial correlation of these multilaterations is obtained by a superimposition of a set of bivariate Gaussian distributions with centers located at the multilateration results (Fig. 4.1(b)). Summing these bivariate Gaussian distributions, shown in (Fig. 4.1(c)) and the final location estimate of the UN is determined by choosing the {X,Y} position of the peak of the resultant sum of pdfs.

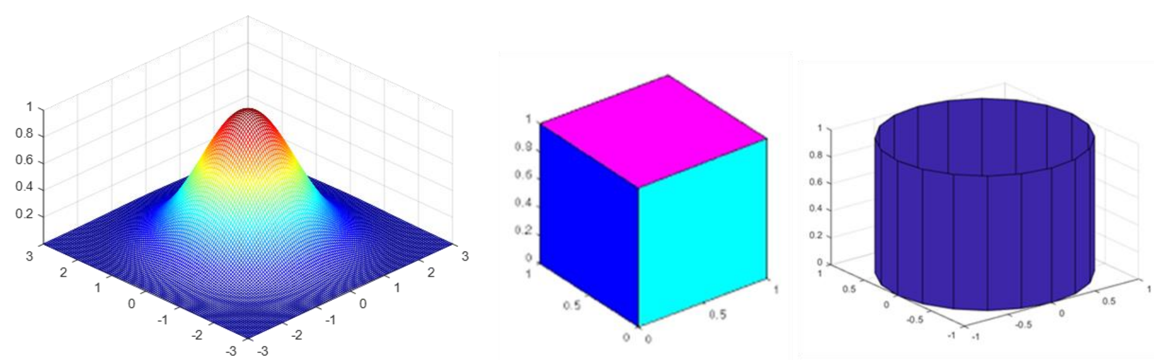
Design considerations of the spatial correlation mechanism include:

1. Shape of the superposition pdf

The Gaussian pdf as shown in Fig. 4.1 is one of the most effective pdfs that can be used because of their effectiveness in capturing Gaussian noise. However, computing and adding Gaussian pdfs is computationally complex to implement, especially on resource constrained sensor motes. To reduce such computation complexity, simpler shapes can be used like rectangular or cylindrical pdfs as depicted in Figs. 4.2(b) and (c) respectively.

2. Size of the footprint of the selected superposition pdf

   The size of the pdf superposed at any location estimate represents the error variation of the distance estimates used in calculating that location estimate. A very small CR size has adverse effect in the majority rule voting mechanism in that the individual pdfs will have minimal total overlap and as such may not effectively localize the UN while the converse is the case whereby a very large CR size may introduce erroneous location estimates which in turn will result in an erroneous final solution to the location of the UN. There has to be a tradeoff between reducing the number of erroneous location estimates in one hand and not having a reasonable total overlap of majority location estimates on the other.



(a) (b) (c)

Figure 4.2: Shows several pdfs considered in the implementation of the spatial correlation mechanism: (a) a bivariate Gaussian pdf, (b) rectangular pdf [63] and (c) cylindrical pdf [64].

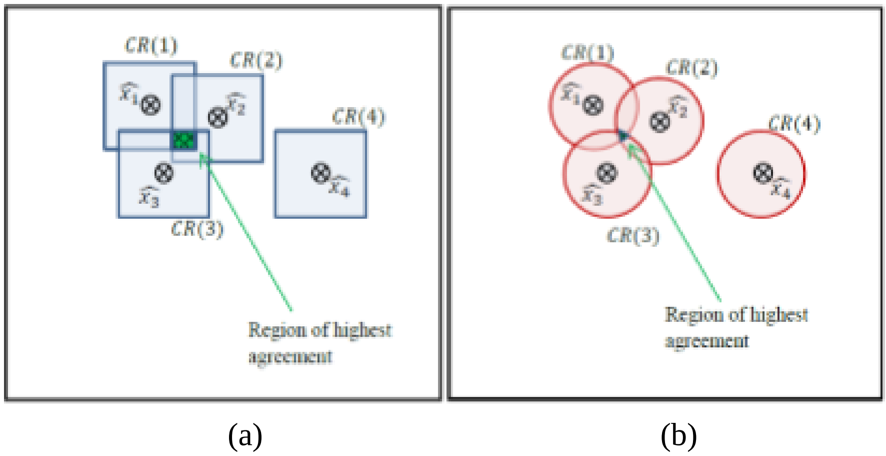


Figure 4.3: Depictions of simple approaches of realizing spatial correlation of several location estimates. In (a), correlation regions are depicted as squares, and in (b), correlation regions are depicted as circles. The region of highest overlap, depicted as the area shaded in green in both cases, is the solution [38].

To further illustrate our proposed correlation-based sensor localization scheme, we have used 2D depictions of the correlation regions as depicted in Fig. 4.3(a) and (b). We will us consider a sensor localization scenario, as described earlier in this section, comprising a static sensor node, known as an UN, using a set of B number of BNs: $\mathrm{BN}_1, \mathrm{BN}_2, \mathrm{BN}_3, \dots, \mathrm{BN}_\mathrm{B}$. These BNs are located at certain known reference points $(R_1, R_2, R_3, \dots, R_B)$, respectively in a wireless sensor region. The correlation-based localization approach takes random subsets of M beacons, performs multilateration to each of the subsets to achieve $\binom{B}{M}$ location estimates. Note that multilateration over a 2-dimensional plane requires ≥ 3 . Then employing the spatial correlation concept, the correlation-based localization scheme uses the majority rule approach to agree on a single location of majority agreement as the final estimation location of the UN. The idea behind this proposed approach is that for those subsets of beacon signals that include beacon signals severely affected by shadowing, their location estimates would be spatially uncorrelated. On the other hand, those subsets of beacon signals obtained from unobstructed beacon signals, would have a high degree of spatial correlation and will be reasonably close to the true position of the UN. Combining these spatially correlated location estimates as described above, will greatly reduce the effect of beacon signals shadowing, with their resultant errors in sensor location estimates.

Typical environments showing obstacles such as trees, metallic tanks, air-conditioning units etc. which severely affect beacon signals are shown in Fig. 4.4. To effectively model the radio channel propagation characteristics of these environments in simulations, we used the log normal shadowing model given in equation 1.1. As we pointed out earlier in the section, in our correlation-based localization approach, the radio channel parameters (n, σ) are determined offline with a small number of RSSI measurements. However, UNs can also determine these parameters dynamically and values used to perform localization in an online manner.

4.2 Proposed Correlation Based Localization Scheme

We will describe our proposed correlation-based localization scheme in three stages:

Stage 1: Data gathering

(a) (b)

Figure 4.4: Typical examples of an obstructed WSN area (a) Wooded environment, (b) Outdoor environment with metallic obstacles [6].

As the UN receives beacon signals from BNs, it forms *N* subsets of non-collinear BNs it receives signals from as calculated from the BNs' locations which are imbedded in the received beacon messages. The unknown node stores the distance estimates computed from all the beacons from their corresponding RSSI measurements using the channel model described in equations 1.2 and 1.3 of section 1.2.2. These stored distances will be used in the next stage of the scheme.

Stage 2: Multilaterations

The UN uses a linear least square approach to calculate a location estimate using multilateration from each of the N subset of M BNs (trilateration for $M = 3$) formed in stage 1. The multilateration results in a total of N location estimates. Each of the location estimates is obtained as $\hat{x} = (A^T A)^{-1} A^T \tilde{b}$, where

$$A = \begin{bmatrix} 2(x_1 - x_2) & 2(y_1 - y_2) \\ \vdots & \vdots \\ 2(x_1 - x_M) & 2(y_1 - y_M) \end{bmatrix} \tag{4.1}$$

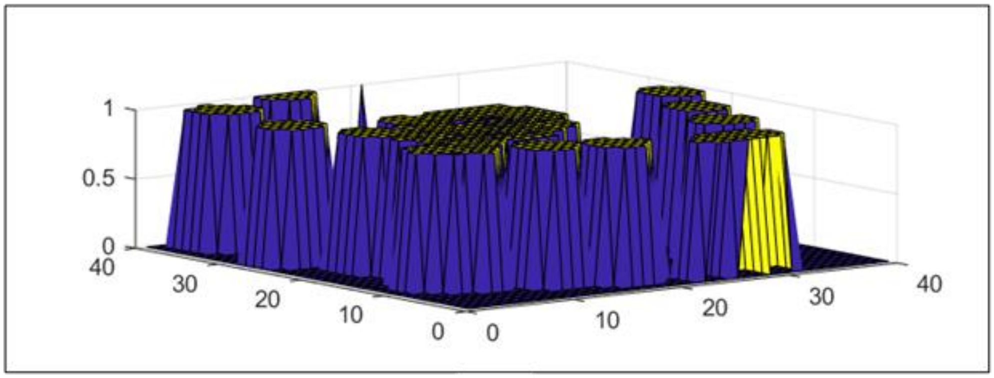
and $\tilde{b}$ is given by:

$$\begin{bmatrix} \tilde{b}_1 \\ \vdots \\ \tilde{b}_M \end{bmatrix} = \begin{bmatrix} \left(\tilde{d}_1{}^2 - \tilde{d}_i{}^2\right) - (x_1{}^2 - x_i{}^2) - (y_1{}^2 - y_i{}^2) \\ \vdots \\ \left(\tilde{d}_1{}^2 - \tilde{d}_M{}^2\right) - (x_1{}^2 - x_M{}^2) - (y_1{}^2 - y_M{}^2) \end{bmatrix} \tag{4.2}$$

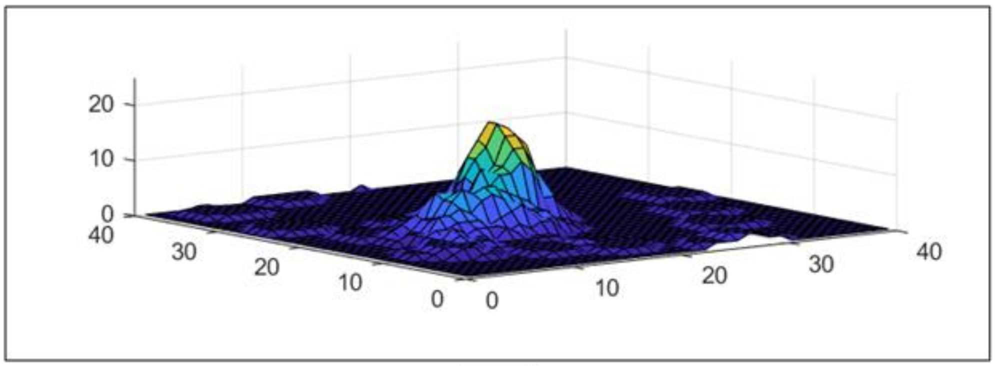
In equations 4.1 and 4.2, $\tilde{d}_i$ represents the distance estimate from the UN to BN_i that is considered in the BN subset, and (x_i, y_i) represents the location of BN_i .

Stage 3: Correlation

In the third and final stage of the scheme, the UN determines which of the N location estimates computed in stage 2 have the highest agreement using the spatial correlation mechanism termed majority rule. The location estimates will be describe by $\hat{x}_i \; \forall \; i \in (1, \ldots, N)$. To achieve spatial correlation, we consider correlation regions (CRs). A CR is the region around a location estimate whereby the center of the CR is centered at that location estimate. The size of a CR represents the margin of error of the individual location estimates. The larger the CR size, the larger the margin of error and vice versa. With the considered CRs with their centers located at each of the location estimates $\hat{x}_i$, we find the areas that have the highest overlap of CRs. We then consider this area of highest overlap as the solution to the localization problem.



(a)



(b)

Figure 4.5: Shows (a) a pdf constructed with center at each of the candidate location estimates and (b) sum of the pdfs from a: signifying consensus in the sensor localization.

To determine this location of highest agreement, we represent each individual position estimates $\hat{x}_i$ by a pdf of corresponding estimate centered at that position estimates as we have shown in Fig. 4.3. The dimension and shape of the pdf represent the error from each location estimate. Since the Gaussian pdf is complex in implementation on resource constrained sensor motes, we have used simpler shapes like squares or circles. We also shown in Fig. 4.3 the CR concept using circular and square CRs around four multilateration results achieved from four different subsets of BNs. The overlap of multiple such pdfs will clearly indicate the agreement of such position estimates. These pdfs are added and the peak point of the resultant sum we have assumed is the final location estimation of the UN. Fig. 4.5(a) shows the square pdf centered at of the individual position estimates and Fig 4.5(b) shows the sum of the individual square pdfs.

4.3. Numerical Analysis

To effectively describe and analyze the scenario, we first present a numerical analysis of the proposed scheme using a small number of BNs. We use the example network region broken up into L x L grid elements as shown in Fig. 4.6 [38]. We also consider a subset of BNs, $S_i = \{BN_x, BN_y, BN_z\}$ of BNs to be used in determining a candidate solution using the trilateration method. To obtain a solution to the trilateraion, we will denote the distance estimates from the UN located at h to the BN_s as d_s, where $S \in \{x, y, z\}$.

We will denote the trilateration operation as a mapping shown below:

$$T: \{ \, d_x, d_y, d_z \, \} \rightarrow \hat{x}_l \tag{4.3}$$

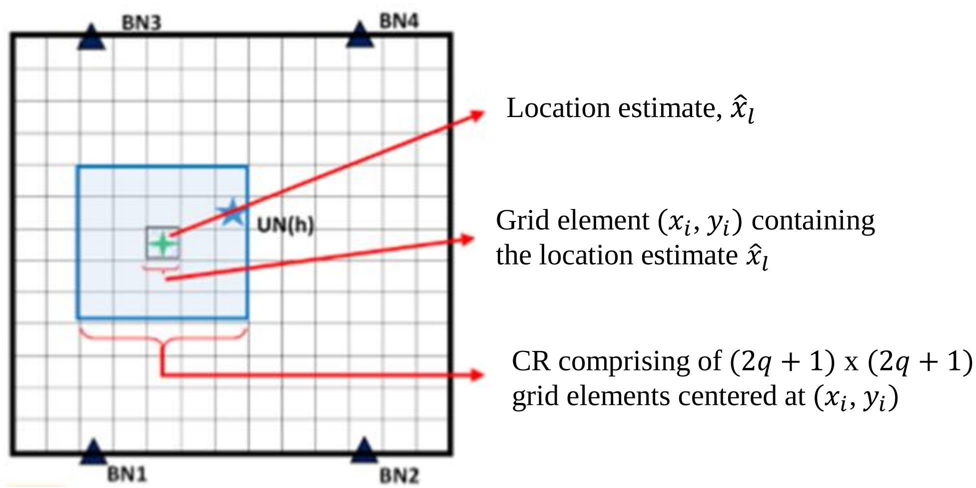


Figure 4.6: Illustration of Network area divided into grids, with four BNs and one UN. Location estimate $\{\boldsymbol{x_i}, \boldsymbol{y_i}\}$ is mapped to grid, {5, 7}. The CR shown here involves $(\boldsymbol{2q+1}) \times (\boldsymbol{2q+1})$ grid elements where $\boldsymbol{q}$ is the number of hop grid elements from the grid that contains the location estimate [38].

The mapping implies that each location estimate $\hat{x}_l$ is mapped to a grid element k located at position (x_i, y_i), which contains the candidate location estimate $\hat{x}_l$. We then introduce the corresponding correlation region, $CR(k)$ to the mapped position estimate by constructing a $(2q + 1) \times (2q + 1)$ region comprising of grid elements who are $2q$ hop neighbors of the grid element centered at position (x_i, y_i).

For this numerical analysis, we will consider four BNs with subsets of M=3 BNs in other to keep the computations simple. Using four BNs and choosing subsets of three BNs from the four BNs considered will yield four different subsets. We then aim to find the probability distributions $P_i(k)$, where $i \in \{1,2,3,4\}$ and,

$$P_i(k) = Prob[solution\ from\ S_i\ includes\ element\ k] \qquad (4.4)$$

Since the environment we are analysis is best modeled using lognormal fading model, we represent the distance estimate d_s by a Gaussian random variable with pdf $f_s(d_s) \sim N\left[\bar{d}_s, \sigma\right]$, where $\bar{d}_s$ depicts the true value of d_s and σ depicts the standard deviation of the lognormal fading. Each joint probability $P_i(k)$ can then be expressed as shown below:

$$P_i(k) = \iiint_{d_x\, d_y\, d_z \in \chi} f_x(d_x) f_y\left(d_y\right) f_z(d_z)\, dd_x\, dd_y\, dd_z \qquad (4.5)$$

where χ depicts the 3-dimensional space of the values of d_x, d_y and d_z whose trilateration solution $\hat{x}_l$ maps to a grid element.

To determine the final solution, we considered any grid element whereby trilateration results from all the four subsets agree on that specific grid element. Our decision to determine the final solution this way is borne out of the need to keep computations at a minimum as against using the region of maximum overlap of the four CRs. Having laid out the model for this numerical analysis, we then consider the result that determines that the final solution is located at element k as:

$$P_L(k) = \prod_{i=1}^{4} P_i(k) \qquad (4.6)$$

We performed numerical calculations, using the example scenario described below, to evaluate the average localization error of our correlation-based scheme in comparison with that obtained from multilateration [38]. Table 4.1 shows the parameters of the network region considered:

Table 4.1: Numerical performance analysis parameters

Parameter description	Values
Network Region	50m x 50m
Number of beacon nodes (or Anchors)	4
Position of beacon nodes	{0,0}, {10,40}, {30,0}, {40,30}
Position of unknown node, h	{15,20}

We have considered $L = 50$ which means that each grid is considered to be $1m^2$ area. In other to compute equation 4.5 numerically, we discretized each of the distance estimates d_x into random variables with values $\{\tilde{d}_{x1}, \tilde{d}_{x2}, \dots, \tilde{d}_{xS}\}$ whose probabilities are derived from their Gaussian pdfs. Our numerical analysis is outlined in the flow chart in Fig. 4.7.

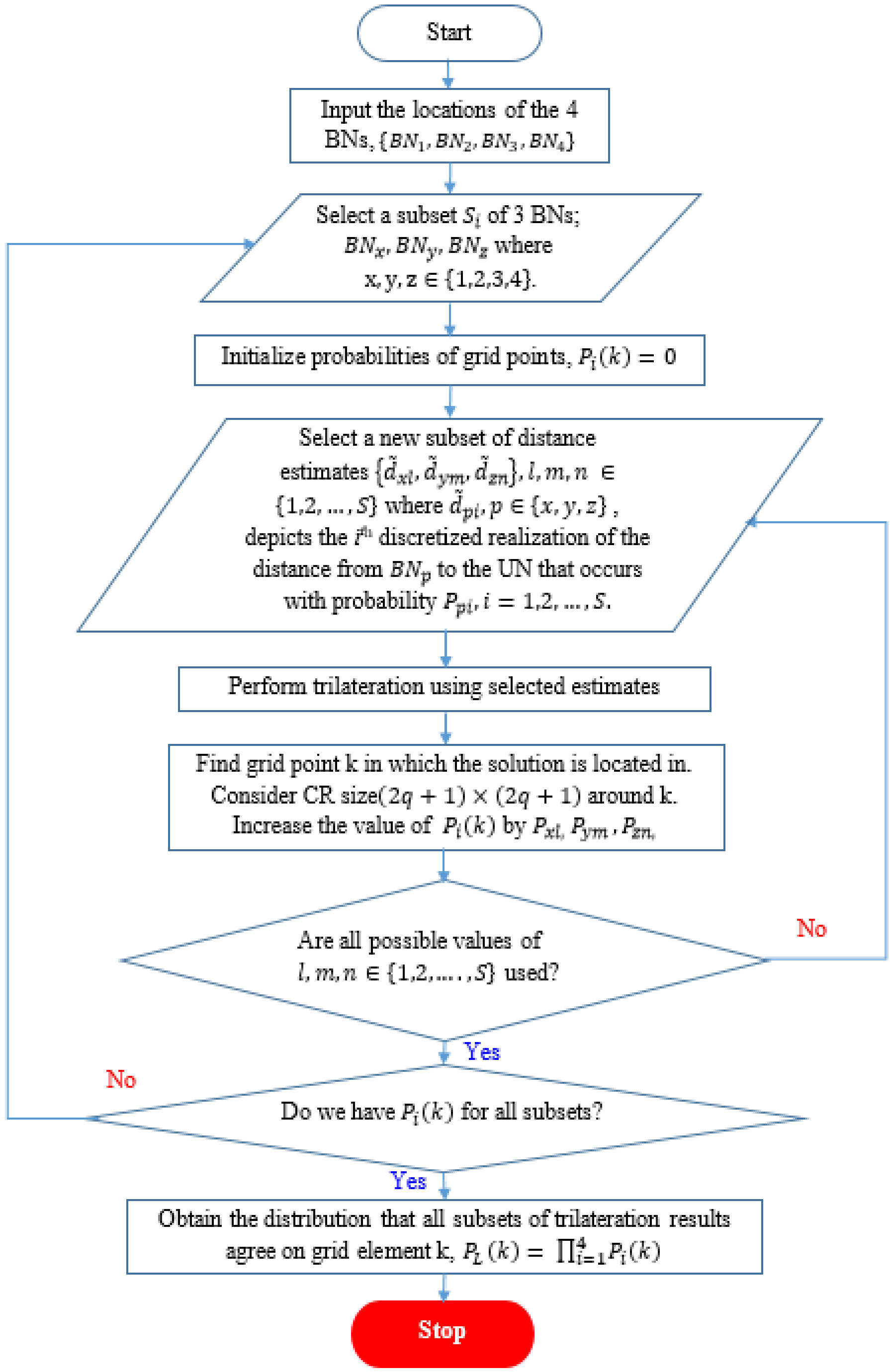


Figure 4.7: Flowchart of numerical analysis of the proposed scheme.

To calculate localization error for our proposed localization scheme, we simply use the Euclidean distance between the actual location of the UN and the grid element k. We show the results of the localization error distribution for varying values of the shadowing parameter in Fig. 4.8. The plots show that while the localization error increases as the values of σ increases, our localization scheme has much lower error on average compared to the multilateration. This is because while the multilateration uses distance estimates from all the BNs, our proposed localization scheme uses spatial correlation mechanism to deselect certain candidate location estimates.

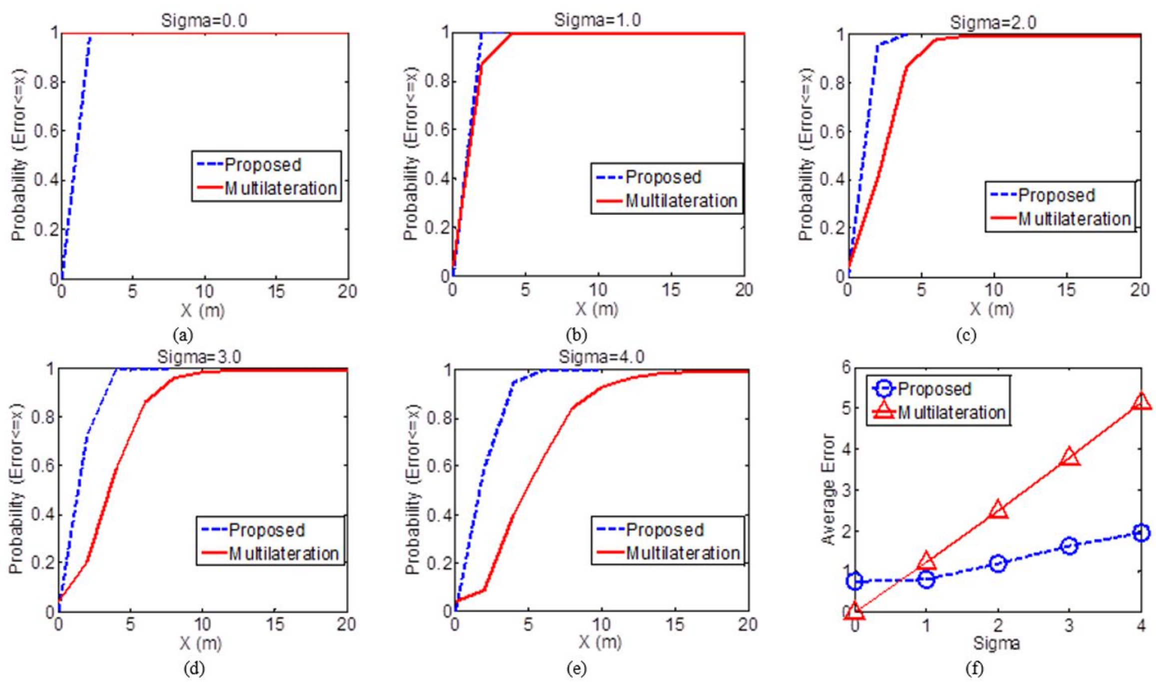


Figure 4.8 (a)-(e): Distribution of localization error for correlation-based localization and multilateration approaches using varying values of σ. (f) Shows average localization error for the two approaches with varying σ [6].

4.4. Simulation-based Performance Results

We also evaluated the performance of our correlation-based localization scheme in an environment containing varying number of obstacles. These obstacles obstruct RF beacon signals and we have modeled the channel characteristics of this environment as shown in section 1.2.2 in the following way:

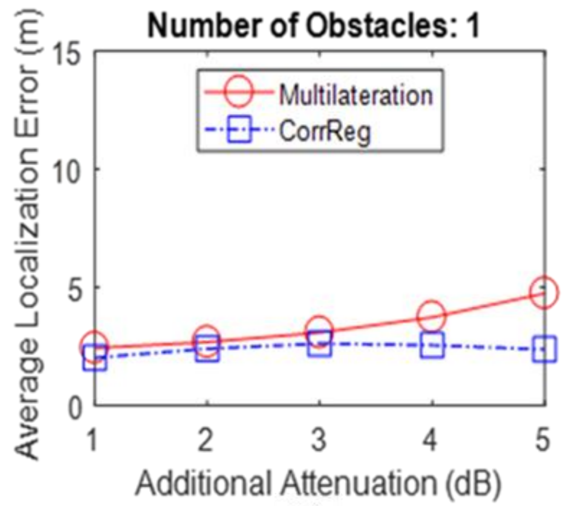
- For unobstructed beacons, the RSSI measurements is modeled as in equation 1.2.
- For the obstructed beacons, the RSSI measurements is modeled as in equation 1.3 whereby further attenuation is added as a result of the shadowing introduced by the obstacle(s).

Table 4.2: Correlation Based localization scheme: Simulation Parameters.

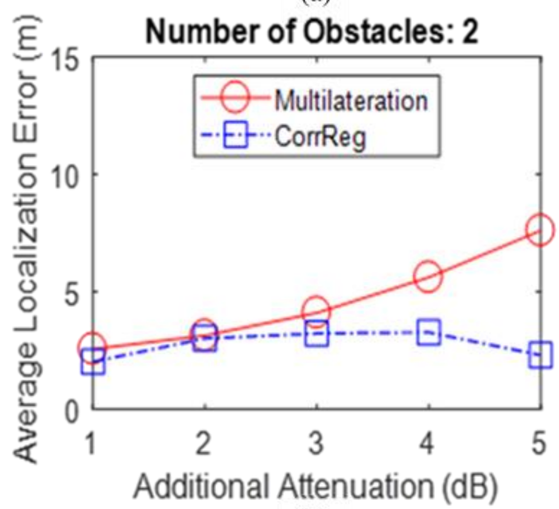
Parameter description	Values
Network Region	40m x 40m
Number of beacon nodes	10
Position of unknown node	{20,20}
Number of obstructed beacon nodes	1, 2 and 3
Line of sight (LOS) fading factor	1 dB
Additional attenuation due to obstructions	4 dB
Transmit Frequency	900 MHz
Transmit Power	0 dBm
Path Loss Exponent (n)	2.3

4.4.1. Error performance: proposed scheme versus multilateration

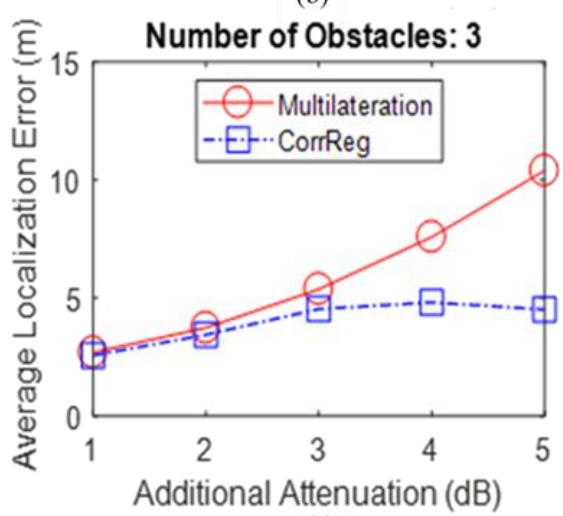
For the simulations, we used the network environment as depicted in Fig. 1.4 and modeled the channel using a log-normal shadowing model with a path loss exponent of 2.3 and the standard deviation (σ) of the additional attenuation due to shadowing is assumed to range from 1 to 5 dB. We used varying number of obstacles: 1, 2 and 3 in the simulations and the radio transmit power of the BNs are 0 dBm. A list of the simulation parameters used for our simulations is shown in Table 4.2. We have evaluated the schemes for scenarios where the network region contains 1 obstacle, 2 obstacles and 3 obstacles. We compared our scheme's performance with those of multilateration thereby showing the benefits of using our scheme instead of the typical multilateration method. Fig. 4.9(a)-(c) show the average localization error achieved using our proposed localization algorithm and that achieved from the multilateration algorithm which uses all the beacon signals together (linear least square estimate) for varying degree of noise. One can observe that our correlation-based localization algorithm minimizes localization errors especially when there are obstacles than in the multilateration method and this we have shown in separate line plots for scenario where there are one, two as well as three obstacles in the network area. This can be attributed to the fact that multilateration uses much more than three beacons signals, unlike in our correlation-based localization algorithm, to determine the UN's location.



(a)



(b)



(c)

Figure 4.9: Localization accuracy comparison: Multilateration vs Correlation based (corrReg) using added attenuation of 1-5dB, 10 beacon nodes and varying number of obstacles.

4.4.2. Effect of number of beacon nodes on performance results

We have shown the localization performance improved for the correlation-based localization scheme over that of the multilateration. In the performance results shown in Fig. 4.9, the evaluation was done using ten beacon nodes or anchors. Using lower number of BNs in the multilateration will result in less accurate localization result. The reverse will be the case when higher number of BNs are used. The reason behind this is that the multilateration employs the linear least square approach and using higher number of beacon signals improves the UN location estimate. This assertion we have shown to be true in Fig. 4.10 with performance plots using varying number of obstacles present in the sensor network region.

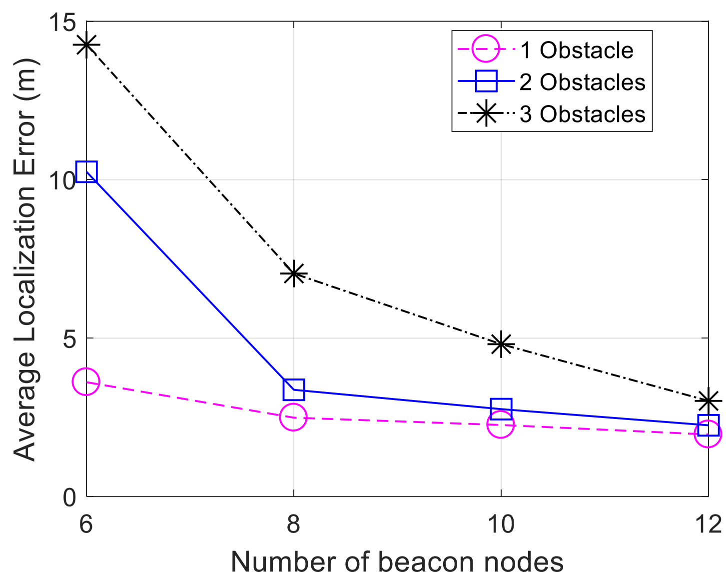


Figure 4.10: Effect of number of beacon nodes on Localization accuracy.

Improvement in localization accuracy as a result of increase in the number of BNs used comes with a cost in computation time. Fig. 4.11 shows how the number of BNs affects the algorithm's computation time. In evaluating computation time, we have used the execution time of the algorithms using an Intel(R) Core(TM) i5-7200U @ 2.70GHz laptop system with 8 GB of RAM, 64-bit Operating system, x64-based processor. We evaluated the computation time for the localization algorithm for three different number of obstacles: 1, 2 and 3 located in the sensor region. The channel propagation model is consistent with that used in the previous chapter and we used an additional attenuation (due to obstructions) of 5dB.

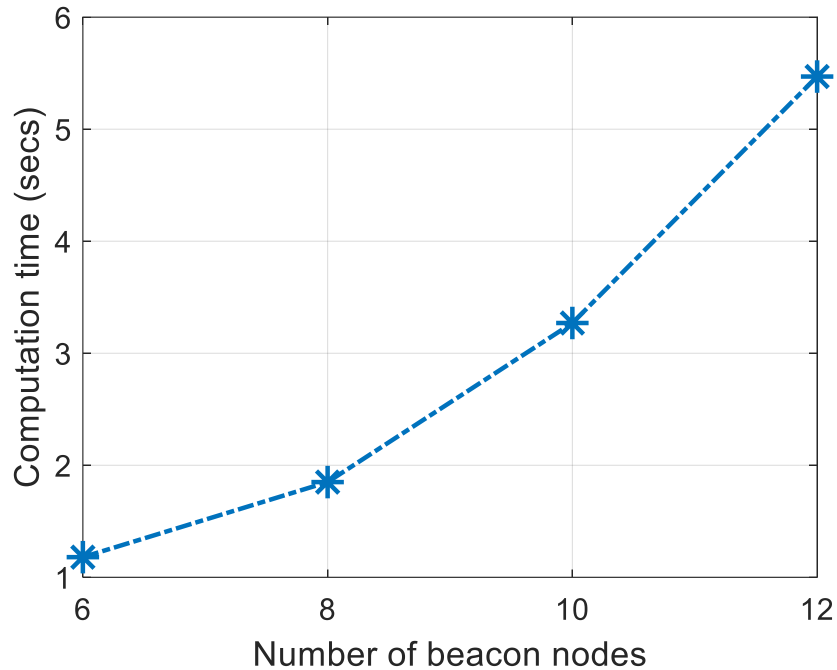
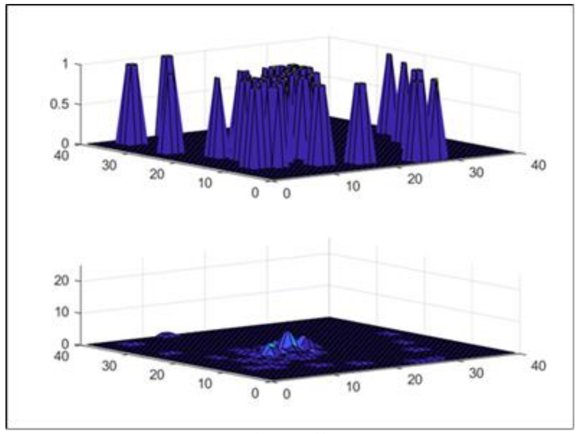


Figure 4.11: Computation time versus the number of beacon nodes used for localization.

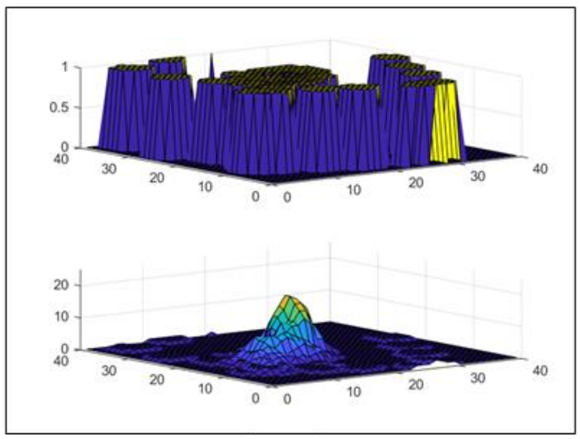
Fig. 4.11 shows computation time for our correlation-based localization algorithm and it shows that as the number of BNs or anchors used in the simulation increases, the computation time increases as well. A tradeoff will have to be made between creating an algorithm that produces reasonably accurate results on one hand and one that is computationally efficient on the other.

4.4.3. Effect of correlation size on localization scheme performance

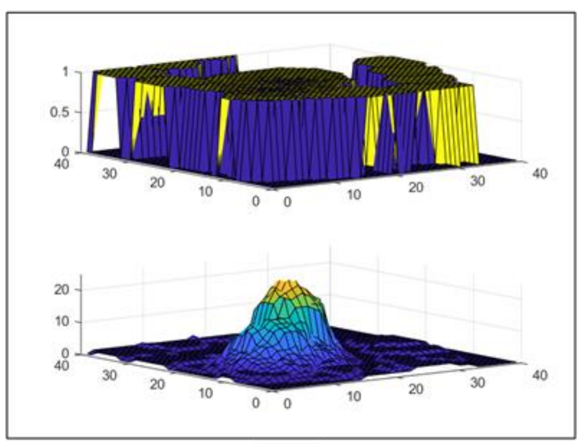
The main mechanism of our proposed localization scheme is the spatial correlation concept which uses the correlation region mechanism to facilitate consensus or majority rule in all the candidate location estimates. As discussed in the previous subsequent sections in this chapter, our scheme considered using various shapes of correlation regions, summing them and then taking the peak of the resultant sum as the location of the UN.



(a)



(b)



(c)

Figure 4.12: Figures above show how correlation region size (CR-size): (a) 1 x 1 (b) 3 x 3 and (c) 5 x 5, influences consensus in sensor localization

However, we have not introduced the effect of the size of these CRs can have on the following: the localization error as well as probability of localization success (the probability of localizing an UN). Before we present the effect of the CR size using simulations of an example network scenario, we will illustrate using Fig. 4.12 the need for carefully choosing the CR size used in the correlation region concept of our scheme.

Fig. 4.13 below shows the effect of the CR size to the performance evaluation criteria. As discussed earlier in this section, a very small CR size has adverse effect in the majority rule voting mechanism in that the individual pdfs as shown in Fig. 4.12(a) will have minimal total overlap and as such may not effectively localize the UN. On the other hand, a very large CR size, as depicted in Fig. 4.12(c), would increase the probability of overlapping of CRs, but may introduce erroneous candidate location estimates which in turn will result in an erroneous final solution to the location of the UN. Fig. 4.12(b) is a tradeoff between the two CR sizes discussed and may offer advantages from these two.

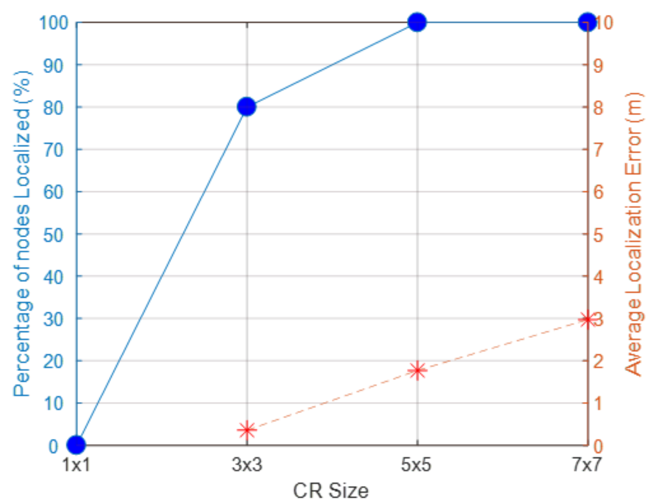


Figure. 4.13: Effect of correlation region size on Probability Of localization success and localization accuracy.

4.4.4. Computation considerations

Our proposed correlation-based localization scheme overcomes problems caused by obstructed signals in sensor location estimation by employing spatial correlation. As pointed out in our correlation-based localization scheme we considered that a bivariate Gaussian function is placed at every location estimate $\hat{x}_i$, which are then summed and the peak position of the sum of bivariate Gaussian functions is deemed to be the final location estimate of the UN.

However, a sum of bivariate Gaussian functions is computationally expensive. To lower this complexity, we considered simpler functions that can be easily computed especially in sensor nodes that are resource constrained. Another criteria we considered is the size of the number of BNs in each subset of selected BNs. In our scheme, we chose M = 3 for the following reasons:

i. To reduce the computational complexity of multilateration as this is critical especially for low-power sensor nodes with low processing capabilities.
ii. Using M = 3 provides a large number of possible location estimates and this aids in achieving a good CR function summation density.

4.4.5. Performance Evaluation: Conclusion

A summary of our findings from the simulations shows the following: our proposed localization scheme outperforms the multilateration method as the added attenuation fading factor increases. With little or no shadowing, the multilateration performs reasonably well but its performance dips as the attenuation increases. We have also shown that increasing the number of BNs (or robot positions) used in sensor localization helps in improving the localization accuracy of the proposed scheme. The reason for the improvement in performance with increased number of BNs is that an increase in the number of BNs introduces a large number of possible location estimates which in turn helps in achieving a good CR function summation density hence better location estimate(s).

Another crucial finding that the correlation-based localization scheme simulations have revealed is the effect of the CR size to the localization accuracy of the scheme. Very small

CR size on one hand and very large CR size on the other hand have adverse effects on the UN location estimation. With CR size of 1x1, the probability of plurality of candidate location estimates being agreement is very minimal and may result in the UN(s) not been localized as shown in Fig. 4.13. As shown, as the CR size is increased to 3x3, 5x5 and above, the probability of this agreement increases. However, the localization error also increases. Fig. 4.13 also shows that correlation size 3x3 localizes the UN 80% of the time (as against 100% for 5x5 and 7x7), however 3x3 has the least localization error. So in essence, as the probability of localization success increases, localization error tends to increase as well. A tradeoff has to be made between increasing the probability of localizing an UN on one hand and keeping the localization error relatively low on the other.

CHAPTER 5: SENSOR LOCALIZATION USING OUTLIER DETECTION

In chapter 4, we introduced a correlation-based localization scheme which applies spatial correlation to minimize the effect of shadowing in range based sensor node localization. In this chapter, we introduce a localization approach that applies clusterization of candidate location estimates to determine an agreement in the location estimation while removing the effects of shadowed beacon signals. In effect, the clusterization scheme helps in removing outliers from candidate location estimates: hence we term this approach as an outlier detection scheme. We consider two different clusterization algorithms for this approach and evaluate their performances.

5.1. Approach for Sensor Localization using Outlier Detection

There is a similarity between this set of schemes and our correlation-based localization discussed earlier in that they employ the spatial correlation concept on a large number of candidate location estimates. Some of these candidate location estimates are results of subsets of N number of BNs with one or more of the BNs' signals being shadowed and thereby introducing error to the distance estimates. Like our underlying assumption in our research has clearly stated, those candidate location estimates resulting from unshadowed beacon signals will exhibit spatial correlation and be close to each other while those estimates resulting from shadowed beacon signals will be spatially uncorrelated and hence not be close to each other. The later location estimates, i.e. resulting from shadowed range estimates, we have termed "Outliers" and the main objective of these localization schemes is to effectively filter out these outliers by employing a clustering mechanism to reduce the effect of the aforementioned outliers in the final estimation of the UN location estimate.

Clustering is a technique that identifies groups of data points that have well-defined similarities. It is heavily used in data science to gain valuable insights from large sample sets of data points, which is an important consideration for machine learning applications [51, 52, 53, 54, 55, 56].

Numerous clustering algorithms have been developed to employ the grouping/clustering technique to effectively classify groups of objects based on defined or learned criteria.

Different types of clustering methods include:

1. Partitioning clustering

   These are clustering techniques that divide a data set into k number of pre-defined groups. Various types of partitioning clustering methods exist, the most popular is the K-means clustering [51]. In K-means clustering algorithm, every cluster is represented by the center or mean of the cluster's data points. A drawback of the K-means clustering technique is that it is sensitive to outlier data points.

2. Hierarchical clustering

   In contrast to the Partitioning clustering method, the Hierarchical clustering does not require the number of clusters for the data sets to be defined apriori. The hierarchical clustering method results in a tree-based depiction of the objects and this is generally referred to as a dendrogram. There are two versions of the dendrograms namely proximity and threshold dendrograms. The proximity dendrogram retains information on the clusters they represent while the threshold dendrograms do not. A very popular hierarchical clustering algorithm is the Mean shift clustering algorithm [60].

   Mean shift clustering involves shifting a chosen kernel iteratively through the set of data points to regions of higher density until the system convergences. So, every shift of the Mean shift clustering kernel is defined by a Mean shift vector which always points in the direction of the maximum density.

3. Density-based clustering

   In this method of clustering, clusters are differentiated from other clusters by their different densities. A cluster with a dense group of data points may be surrounded by other groups of low density data points.

   A very important Density-based clustering algorithm is the Density-Based Spatial Clustering and Application with Noise (DBSCAN) [55]. DBSCAN needs only a single input parameter and even helps in obtaining an appropriate value for the parameter. It determines clusters with arbitrary shapes and is effective for various sizes of databases.

4. Model-based clustering

   This clustering method considers the source of the data set to be from a distribution that is a mixture of multiple clusters and tries to recover the distribution model from that data set. This method employs a concept whereby every data point is assigned a probability or maximum likelihood of being a member of each cluster. Important work done by early researchers of model-based clustering is shown in [52, 53]. The K-means clustering is also a special case of the model-based clustering with all the distributions assumed to be Gaussians of equal variance.

   The drawbacks of the model-based clustering include: false assumption made that the data sets originates from particular probability distributions, slow execution time especially in large data sets and the requirement to specify basic models for mixture.

5. Fuzzy clustering

   In the Fuzzy clustering technique, each data point belongs to more than one cluster. Every data point is assigned a probability of being an element in each of the clusters [57, 58, 59]. This is in contrast to the K-means or K-medoids clustering where every data point can belong to only a single cluster and this is referred to as hard or non-fuzzy clustering. Fuzzy clustering on the other hand is referred to as soft clustering. The extent to which a data point belongs to a cluster is assigned a numerical value ranging from 0 to 1. A very popular fuzzy clustering algorithm is the fuzzy c-means (FCM) algorithm where the centroid of a cluster is computed as the mean of all data points and weighted by the degree to which they belong to the cluster.

Our main criteria for choosing the clustering algorithms is not to classify data points but to identify a region in the network area where data points are dense. The Mean shift clustering as well as the Centroid clustering method meet this criteria.

5.2. Proposed Clustering Based Localization Schemes

We will now introduce our pair of outlier detection localization schemes and the steps by which they achieve the filtering of outliers in other to effectively estimate the location of an UN.

5.2.1. Centroid Method

The first of the two outlier detection localization schemes is the Outlier Detection-Centroid localization scheme herein called the OD_CTRD. We will describe this localization scheme using the example network scenario shown in Fig. 5.1. In this scenario, we have considered a 40m x 40m network area, eight BNs deployed in a circular area of 20m radius and an UN. The network region also contains some obstacles which obstruct beacon signals transmitted from some BNs to the UN.

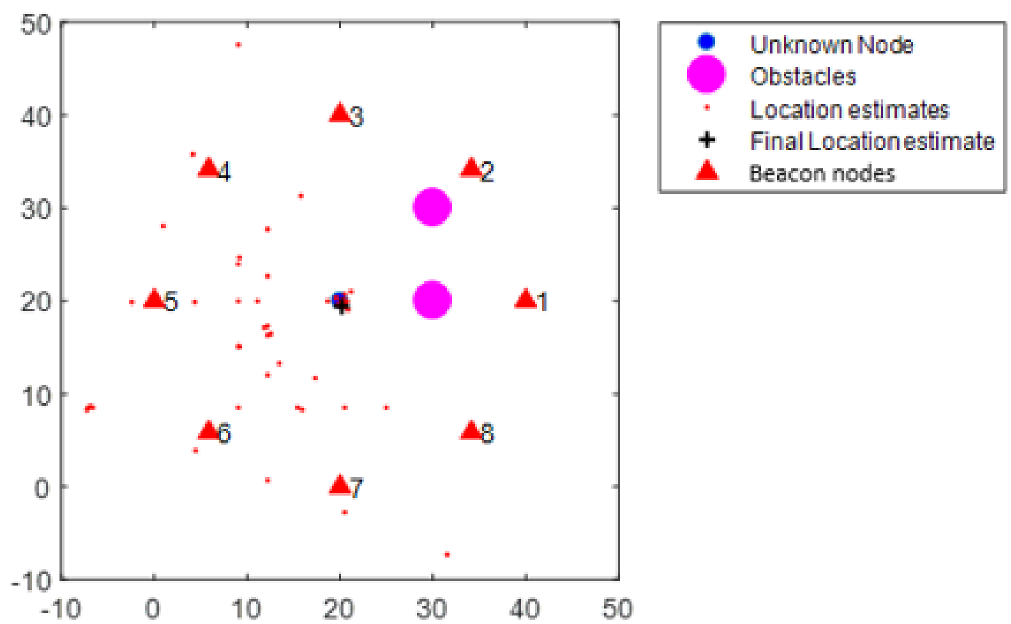


Figure 5.1: Illustration of the 40m x 40m Network area containing one UN, eight BNs (or anchors) and obstacles [39].

From Fig. 5.1, the red triangle represents static BNs or can be used to represent locations where a mobile robot sends beacon signals. The blue circle is the true location of the UN. The purple circle represents obstacles in the network region. The red dots represent candidate location estimates which are results of the individual multilaterations using subsets of BNs. And finally, the black plus sign represents the final estimated location of the UN.

Beacon nodes 1 and 2 are obstructed by the obstacles as shown in the Fig 5.1 and beacon signals sent from these BNs to the UN will be shadowed. Consequently, those multilateration results calculated using distance estimates from beacon signals from any one or both of these obstructed BNs will be erroneous and spatially uncorrelated and will be widely dispersed. Conversely, those multilaterations that do not include distance estimates from beacon signals from either of these two obstructed BNs will yield more accurate and spatially correlated location estimates. These spatially correlated location estimates, though small in number, will be close to each other. Our goal is to employ an effective clustering mechanism to focus on those areas of high density and discard areas of lower density. The objective here is to find the location of the UN in the network region and we will now outline the approach of the schemes. The UN on receiving beacon signals from the BNs, obtains the RSSI of the signals and based on these, estimates its distance from the BNs it received signals from using a channel propagation model. This localization approach takes subsets of N beacons selected B beacons each time. Then multilateration is applied to each of these subsets using their respective estimated distances resulting in a total of $\binom{N}{B}$ location estimates. These location estimates, referred to here as candidate location estimates, are obtained using some shadowed beacon signals and others from unshadowed beacon signals. The scheme then applies a clustering mechanism to efficiently choose the most probable location in the network region where a majority of the candidate location estimates have a consensus on or are in agreement with. We have made the assumption that the RSSI of the beacon signals received from all the BNs at the UN experience lognormal fading. We have used the same channel propagation model described in section 1.2.2 with equations 1.1 to 1.3 whereby beacon signals from unobstructed BNs follow equation 1.2 and beacon signals from obstructed BNs follow equation 1.3 where they experience further shadowing.

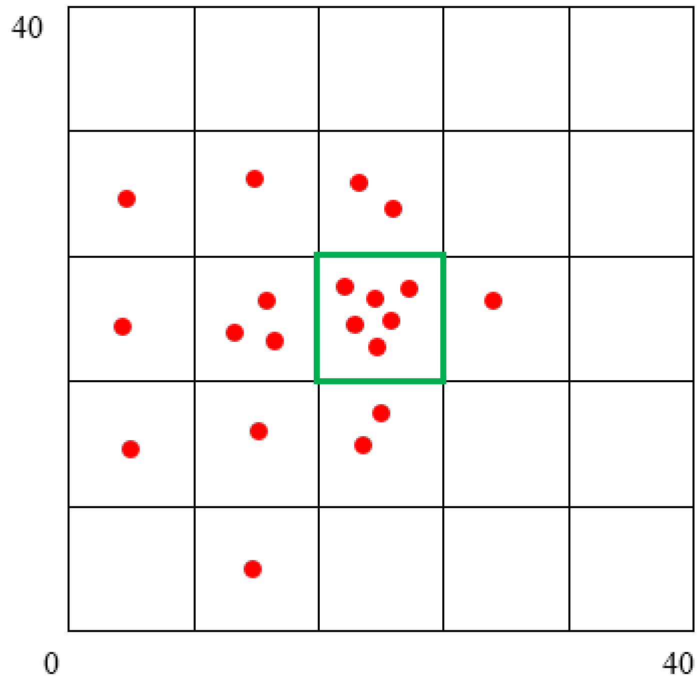


Figure 5.2: showing the OD_CTRD process. A 40 x 40 network region broken into 5 x 5 grids. Red dots represent the mapped trilateration results and green box shows the grid with the highest number of mapped location estimates. The centroid of the grid box is assumed to be the final location of the UN.

To properly illustrate this scheme, we have shown trilateration results from all subsets of B, (B=3) BNs at the sensor nodes are depicted by the red dots in Fig. 5.2. The candidate location estimates are fed into a clustering mechanism which utilizes the spatial correlation of the location estimates to deselect outlier location estimates. The output of the clustering algorithm is considered to be the final location estimate of the UN and is obtained by choosing the centroid of the grid containing the highest number of location estimates.

The network region shown in Fig. 5.2 is broken into a 5 x 5 grids. A grid element occupancy counter is initialized, i.e. every grid element initial occupancy count is set to zero. Then each of the candidate location estimates is mapped to the grid element where they fall into. We describe the mapping operation done here with the expression shown in equation 5.1.

$$T_i\colon \{\, d_1, d_2, d_3 \,\} \rightarrow \, < \widehat{x, y} >_j \begin{cases} i \in \left(1, \binom{N}{B}\right) \\ j \in (1, L^2) \end{cases} \tag{5.1}$$

This mapping means that every candidate location estimate resulting from the trilateration $T_i\colon \{ d_1, d_2, d_3 \}$ is mapped to a grid element located at position $< \widehat{x, y} >_j$.

As the candidate locations estimates are mapped, a count of the number of location estimates contained in each grid is kept. On mapping all the candidate location estimates to grids, the grid element that contains the maximum number of mapped location estimates is chosen as the grid containing the location of the UN. The clustering algorithm outputs the grid ID of this selected grid and the centroid of the selected grid is considered as the final location estimate of the UN. However there may arise special cases whereby more than one grid has the same number of maximum mapped location estimates. In such special cases, further processing is done to select one grid as the chosen grid. One way of choosing this grid is to check amongst the selected grid elements and determine the degree of spread of the constituent candidate location estimates in each of the selected grids and the grid with the tightest spread is the winner and becomes the chosen grid. These special cases will not be explored further in this dissertation. The flowchart shown in Fig. 5.3 outlines the steps of the localization scheme described. In the next subsection, we will describe the other outlier detection localization scheme called Outlier Detection_Mean Shift Clustering.

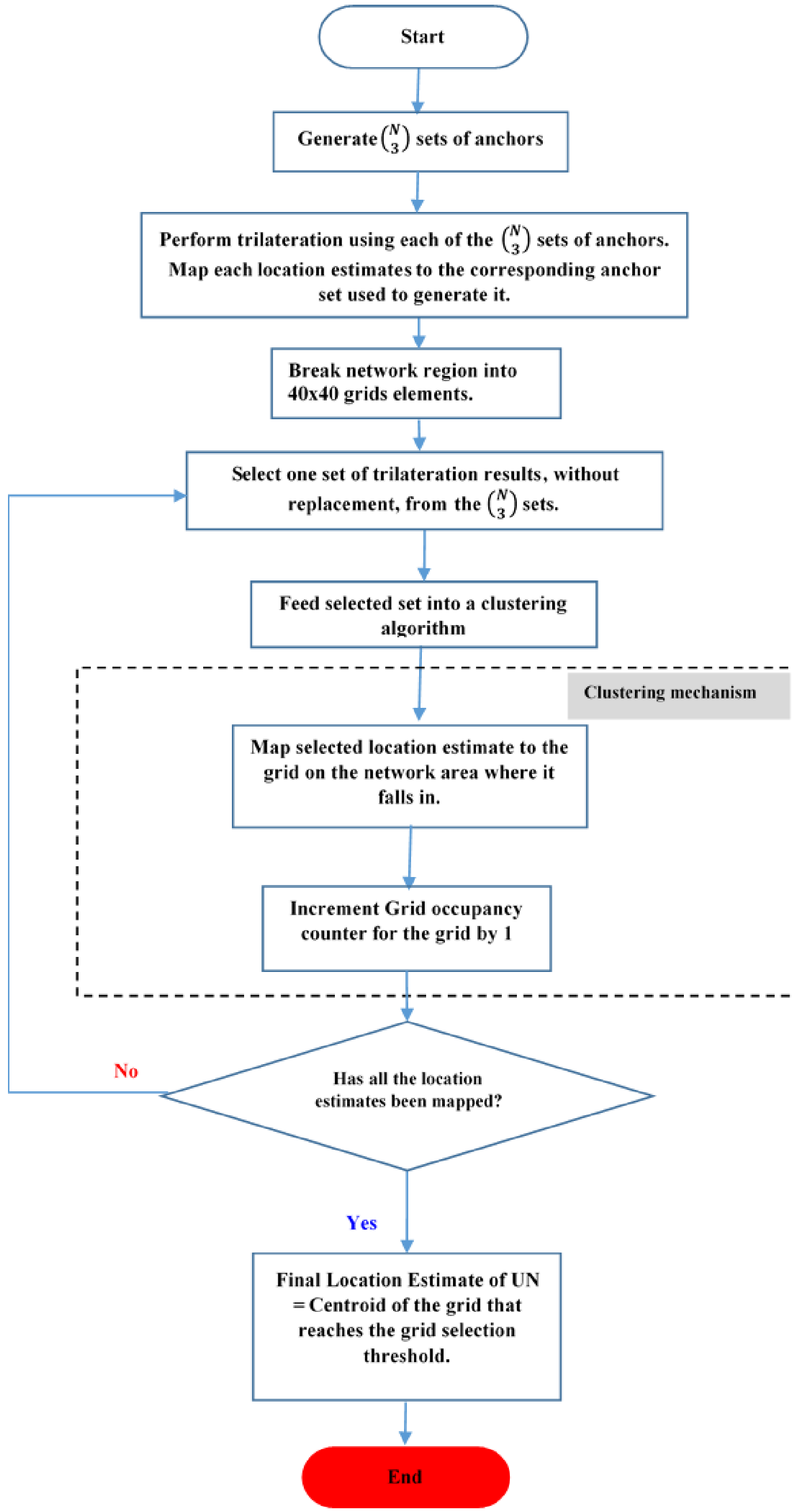


Figure 5.3: Flowchart of the OD_CTRD Outlier detection localization scheme

5.2.2. Mean Shift Clustering Method

The Outlier Detection_Mean Shift Clustering (OD_MSC) is the second outlier detection localization scheme we have proposed. The candidate location estimates generation phase of the scheme is similar to that of the OD_CTRD. The unique part of the OD_MSC localization scheme is that the candidate location estimates generated are fed into the mean-shift clustering mechanism.

Mean shift clustering is a non-parametric sliding-window-based algorithm that is used to find dense regions of data points and also used to locate the center points of each of these regions. The algorithm works by incrementally updating candidate data points for center points to be the mean data points within a chosen sliding-window. A post processing procedure is initiated in other to filter the candidate windows and this is done to eliminate duplicates or near duplicates and finally forming the final set of center points and group tuples. Mean shift clustering was proposed by Fukunaga, K. & Hostetler, L. in 1975 and later adapted by Cheng for use in image analysis. It is very popular in the computer vision research community and have been found to be beneficial in other disciplines as well.

Mean shift is built on the concept of the Kernel Density Estimation (KDE) which is a method used to estimate the probability density function of a given set of data. A kernel is placed on each data point in the data set and the individual kernels are summed up to generate a new probability density function which may vary depending on the kernel bandwidth used. The Mean shift clustering process does not require that the number of clusters is specified prior as this is determined by the algorithm and the data being processed. Two of the popular kernel functions used in Mean shift clustering are the Flat or uniform kernel and the Gaussian [61].

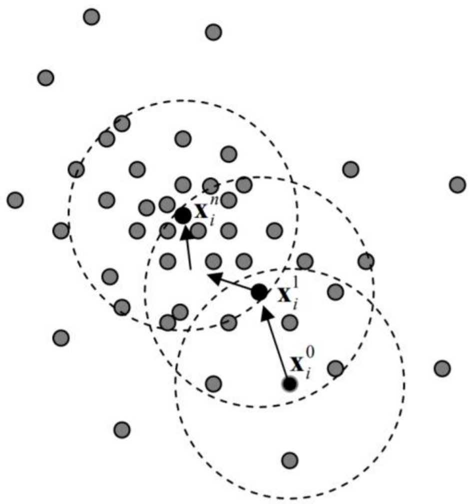


Figure 5.4: showing the Mean shift process [62].

Fig. 5.4 shows an illustration of the Mean shift process. The first step is to represent the data set as data points as shown in the figure. The algorithm process starts at any chosen data point X_i, then finds the stationary points of the density function. The superscripts shown in Fig. 5.4 represent the Mean shift process iteration, the shaded dots represent the input data points and the black dots represent the successive centers of the windows. The dotted circles represent the density estimation windows.

Advantages of the Mean shift clustering algorithm include:

- Requires only one parameter, window size or bandwidth
- It is robust to outliers
- It is independent of the type of model used
- Ability to find different number of modes

Disadvantages of the Mean shift clustering algorithm include:

- Result of the Mean shift cluster is dependent on the bandwidth and its selection is not trivial
- Can be computationally expensive, $O(n^2)$

- This can be overcome by employing certain special steps like lowering the number of data points search without adversely affecting the output of the scheme.

With the knowledge of the benefits of the Mean shift clustering and its success in the computer vision research community, we decided to introduce the concept to the sensor localization problem in other to filter out the outlier location estimates generated using erroneous distance estimates caused by shadowed beacon signals. Due to spatial correlation of candidate location estimates, the mean shift clustering mechanism systematically moves through these location estimates moving from regions of low density, i.e. low agreement by individual location estimates to regions of higher density, i.e. higher agreement.

To illustrate the OD_MSC localization scheme, we will outline the implementation steps using the flowchart in Fig. 5.5:

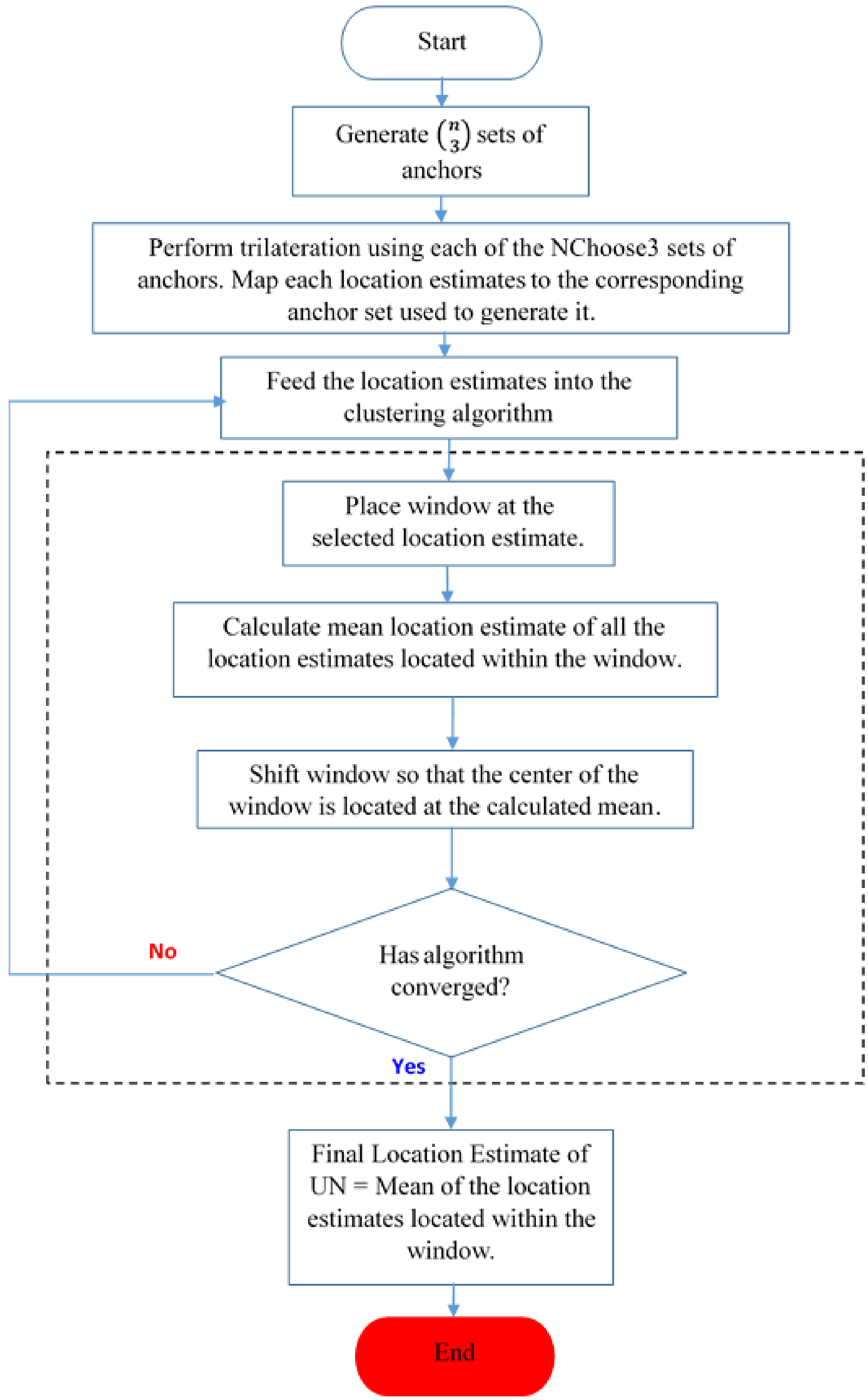


Figure 5.5: Flowchart of the OD_MSC Outlier detection localization scheme

5.3. Performance results

We present our simulation evaluation of the OD_CTRD and OD_MSC localization schemes. We considered the network area shown in Fig. 5.1. For the simulation, we have employed a channel propagation model as outlined in section 1.2.2 whereby a lognormal shadowing model comprising fading factor is used for LOS beacons signals and additional attenuation factor for obstructed or NLOS beacon signals. Just as the other localization schemes earlier discussed, we have evaluated this scheme for scenarios with 1 obstacle and multiple obstacles present in the network area.

Table 5.1 shows the network parameters we have used in simulating the outlier detection localization schemes. Obstacles in the network region is modeled in the following manner. For LOS beacon signals, we used lognormal fading factor parameter σ_f = 1dB and for NLOS beacon signals, we have introduced an added attenuation factor due to shadowing σ_s for which we have used varying values ranging from 1dB to 5dB. We evaluated the schemes using average localization error in meters and also evaluated the how the number of BNs affects the localization accuracy for the two outlier detection schemes in the 3-obstacle scenarios.

Table 5.1: Outlier detection schemes: Simulation Parameters

Parameter description	Values
Network Region	40m x 40m
Number of grids	40 x 40
Number of beacon nodes	8
Position of unknown node	{20,20}
Number of obstructed beacon nodes	1, 2 and 3
Line of sight (LOS) fading factor σ_f	1 dB
Additional attenuation due to obstructions	1dB - 5 dB
Transmit Frequency	900 MHz
Transmit Power	0 dBm
Path Loss Exponent (n)	2.3
Bandwidth (for OD_MSC)	5

5.3.1 Design considerations for the Outlier detection localization schemes

For the OD_CTRD method, the following design considerations were taken:

- Number of BNs in every BN subset – To have a reasonable number of location estimates to be used for the UN localization, we used
- Size of network grids

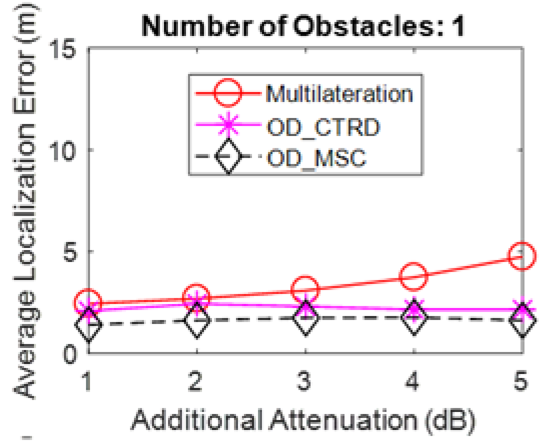
For the OD_MSC method, the following design considerations were taken:

- The window-size or bandwidth - This is the only parameter required for the OD_MSC clusterization mechanism and our consideration of the size of window we selected for our evaluation was through a trial run of several window sizes: 1, 3, 5, 7 and 9. Selection of a very small window-size results in increase in computation time as it will take the mechanism more time to loop through all location estimates (or data points) and using a very large window-size will result in degraded clustering.
  From the various window-sizes evaluated, window-size 5 gave the best result given the execution time.

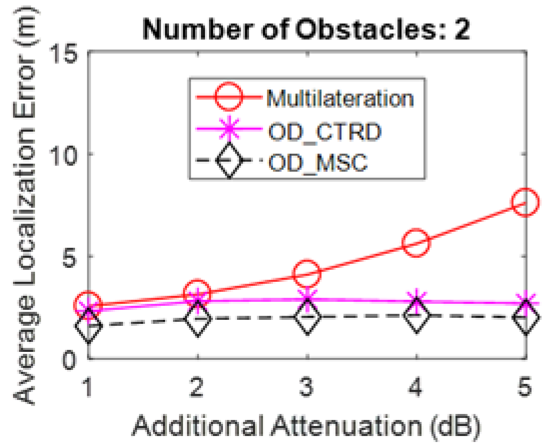
5.3.2. Localization Error performance compared with multilateration

Fig. 5.6 shows the performance of the OD_CTRD and the OD_MSC compared with multilateration with varying added attenuation. Fig 5.5(a) was simulated for a network region that has one obstacle within it while Fig 5.5(b) and Fig 5.5(c) are two and three obstacles respectively.

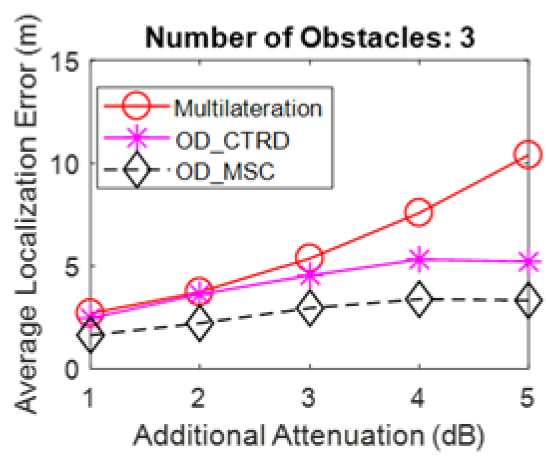
We can observe that for all 3 scenarios, the average localization for all schemes increase as the added attenuation increases. It is also evident from the plots that our pair of outlier detection schemes perform better than the multilateration scheme.



(a)



(b)



(c)

Figure 5.6: Localization accuracy comparison: Multilateration vs Outlier Detection Methods for 1, 2 or 3 obstacles.

5.3.3. Effect of number of beacon nodes on localization accuracy

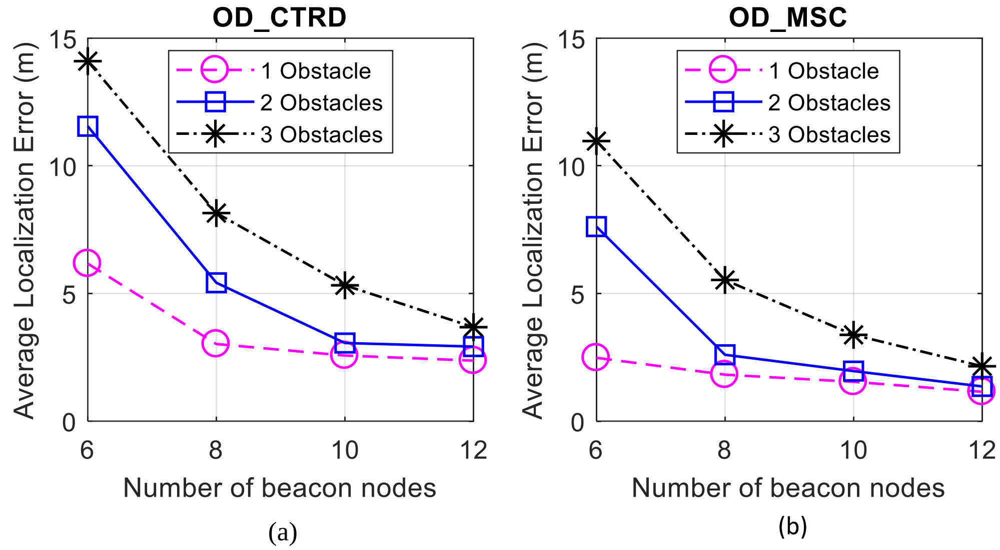


Figure 5.7: Effect of the number of beacon nodes on Localization accuracy comparison.

In this subsection, we show in Fig. 5.7 how the number of BNs used affects the effectiveness of the two outlier localization schemes: OD_CTRD (shown in Fig. 5.7a) and the OD_MSC (shown in Fig. 5.7b). We evaluated the outlier localization schemes using 6, 8, 10 and 12 BNs. The plots show that as the number of BNs used in estimating the location of the UN increases, the localization error of the schemes decrease. This is true for both OD_CTRD and OD_MSC. The reason behind this is because with increase in the number of unique BNs or beacon positions, hence increased beacon signals, the number of candidate location estimates used in estimating the location of the UN increases, thereby improving the localization efficiency of the scheme. However, the improvement in localization accuracy comes with a cost in computation time. Fig. 5.8 shows how the number of BNs affects the localization algorithm's computation time. We evaluated the computation time for OD_CTRD and OD_MSC algorithms. To evaluate computation time, we have used the execution time of the algorithms on an Intel(R) Core(TM) i5-7200U @ 2.70GHz laptop system with 8 GB of RAM, 64-bit Operating system, x64-based processor. The channel propagation model is consistent with that used in the previous chapters and we used an additional attenuation (due to obstructions) of 5dB.

Fig. 5.8(a) shows computation time for the OD_CTRD algorithm and one can observe that with increase in the number of BNs or anchors used in the simulation, the computation time increases as well. Same characteristic is exhibited for OD_MSC shown in Fig. 5.8(b). Also the two computation time plots show that OD_MSC is computationally more efficient than the OD_CTRD. We will reiterate that a tradeoff will have to be made between building an algorithm that produces reasonably accurate results on one hand and one that is computationally efficient.

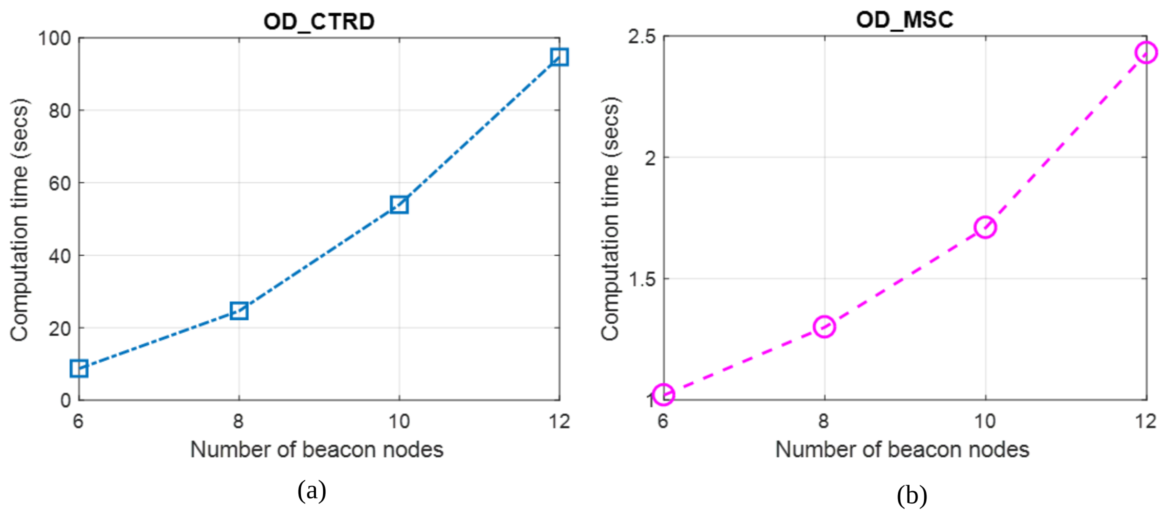


Figure 5.8: Computation Time versus number of beacon nodes used for, (a) OD_CTRD and (b) OD_MSC.

5.4. Chapter Conclusion

With the success of the mean shift clustering algorithm in the computer vision research community, we introduced that mechanism to our spatial correlation based sensor node localization scheme, an algorithm we termed Outlier detection_Mean Shift Clustering (OD_MSC) and the result has been shown to be very promising. The second outlier detection scheme we introduced is the Outlier detection_Centroid method (OD_CTRD) which has also produced comparable results to the OD_MSC.

A summary of our findings from the simulations shows the following: the pair of outlier detection localization schemes: OD_CTRD and OD_MSC outperform the multilateration method as the added attenuation fading factor increases. We have also shown that increasing the number of BNs (or robot positions) used in sensor localization improves the localization performance of the proposed scheme. The reason for the improvement in performance of the localization schemes with increased number of BNs is that an increase in the number of BNs introduces a large number of possible location estimates to be used in the location estimation process which in turn helps in localizing the UN.

In this and the previous chapters, we have explored and presented theoretical analysis and evaluated various localization schemes. In chapter 6, we will discuss and evaluate all the localization schemes we have discussed so far using experimental data from real life environments.

CHAPTER 6: PERFORMANCE COMPARISON

In this chapter, we present a performance comparison of all the various localization schemes we evaluated, from simulations as well as from experiments. The localization schemes we have evaluated are as follows: Multilateration, Range-Only SLAM (ROSLAM), Correlation-based (CorrReg), Outlier detection-Centroid (OD_CTRD) and Outlier detection-Mean Shift Clustering (OD_MSC) methods.

6.1. Simulation Performance Evaluation

For simulations, we implemented the localization schemes using MATLAB R2017a. We have used a common system model to evaluate all the five localization schemes. We have presented these simulation parameters in tables shown in earlier chapters. However, a summary of the parameters is shown below:

- We implemented the schemes assuming a 40m x 40m network area
- 10 beacon nodes (for all 4 localization schemes expect ROSLAM which uses one mobile BN transmitting at 10 different positions)
- UN at position {20,20}
- Varying number of obstacles (1, 2 and 3 obstacles) in the network area
- LOS fading factor of 1dB
- Additional attenuation (due to obstructions) of 4 dB
- Transmit frequency of 900MHz
- Radio transmit Power of 0dBm
- Path loss exponent of 2.3
- Results are averaged over a total of 10 independent trials.

The network environment is depicted in Fig. 1.4 and the channel was modelled using a log-normal shadowing model as described in section 1.2.2.

6.1.1. Simulation Results

We described the system model used for simulations and will now present the results achieved with the aim of comparing the effectiveness of the five localization schemes discussed using our assumed model where a number of obstacles are present in the network area.

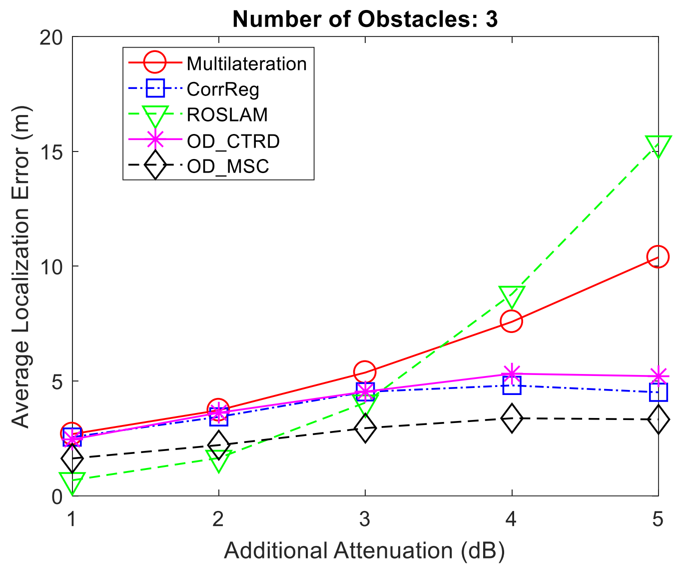


Figure 6.1: Localization accuracy comparison using varying added attenuation of 1dB to 5dB

The plot in Fig. 6.1 shows a comparison plot of all the five localization schemes evaluated using simulations. As shown, the corrReg, OD_CTRD and OD_MSC methods perform better than the multilateration and ROSLAM when added attenuation factor goes from 3dB and above. The reason that corrReg, OD_CTRD and OD_MSC perform significantly better, as we have observed, is that these three schemes unlike the multilateration and ROSLAM, employ systematic mechanisms to deselect erroneous distance estimates for localization.

Another key reason why the ROSLAM did not perform well in these simulations, as we have pointed out earlier in chapter 3, is because the number of unique locations where the mobile BN transmits beacons is not sufficient to effectively localize the UN. We have shown in chapter 3 that increasing the number of these unique robot or mobile BN steps improves ROSLAM localization performance, albeit the increase in computational time introduced.

6.2 Experimental Performance Evaluation

Next, we present experimental evaluations of the proposed correlation based and outlier based localization schemes using off-the-shelf wireless sensor nodes. We consider several different deployment environments, all of which are characterized with various shadowing obstructions. We obtain measurement data from each of these environments and apply them to the five localization schemes implemented in MATLAB. We will cover the experimental system model which includes the description of the locations where we collected the experimental data, the nature of the sensor nodes deployment, the number and types of sensor nodes used as well as the objective(s) of the experiments we have conducted.

6.2.1 Experimental system model

Prior to discussing the evaluation of the localization schemes, we will describe the locations where we collected measurement data. We collected measurement data to be used in evaluating our localization scheme as well as the other four localization schemes in three different physical locations: an open outdoor courtyard, an indoor location (LAB), and a wooded location, all of which are located on the campus of the University of North Carolina at Charlotte (UNCC). We also obtained measurement data from a fourth location at an off campus wooded area in Simpsonville South Carolina. For each of the testbeds evaluated, we took a significant number of RSSI measurements which we used to calculate the path loss exponent for the environment. Also we used Micaz motes [65] for each of the experimental testbeds and estimated locations were calculated offline on a laptop from data transmitted to a base station connected to the laptop. We will describe in detail, the type of motes used in a subsequent sub-section and Fig. 6.2 shows a Micaz mote.

Figure 6.2: Shows a Micaz Mote with radio platform MPR2400 based on the Atmel ATmega128L [65].

The environmental locations where we collected measurement data are outlined below:

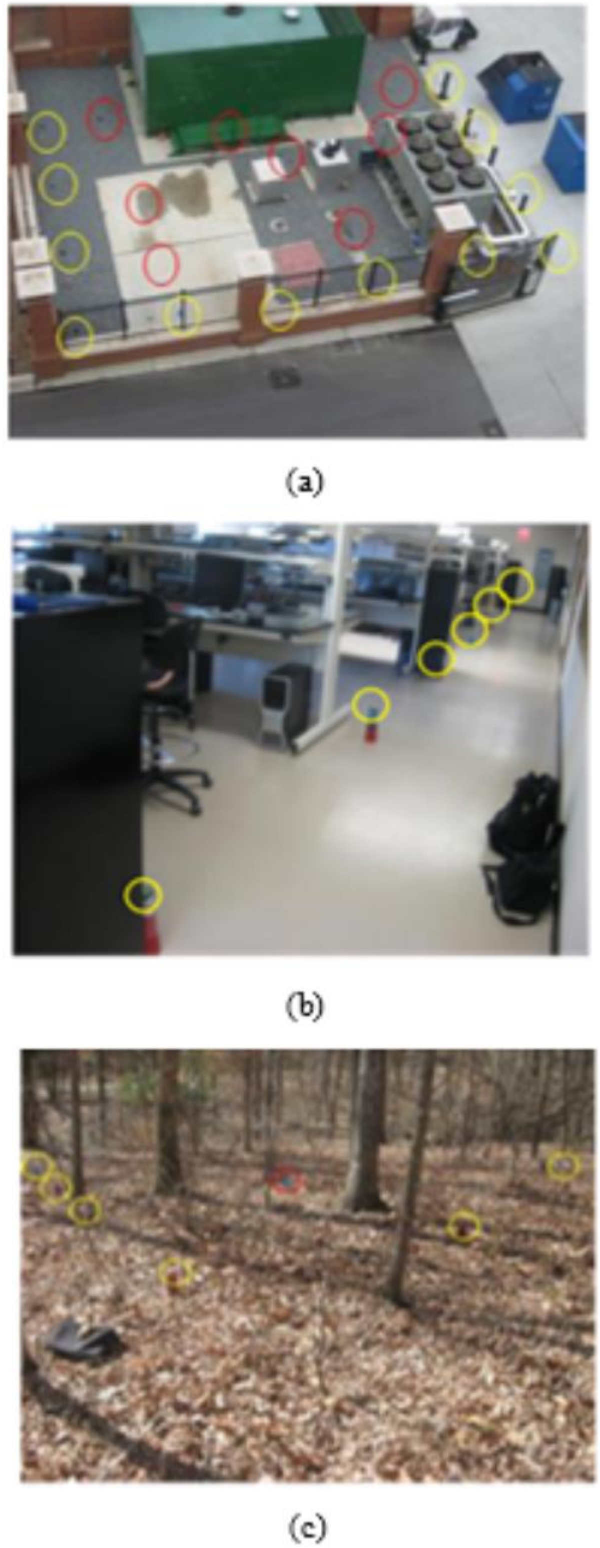


Figure 6.3: (a) Outdoor WSN localization test bed (b) Indoor location and (c) Wooded location. [6]

Outdoor environment, UNC Charlotte

We took measurement readings in an open air location behind the EPIC building at UNC Charlotte that contains metallic structures such as metallic tanks, air handling units, poles etc. as shown in Fig. 6.3(a). These obstacles obstruct RF beacon signals transmitted from BNs. We deployed twelve BNs (circled with yellow rings) and eight UNs (circled with red rings) in a $25m \times 25m$ region. The path loss exponent obtained using the RSSI from received beacon signals was determined to be 3.15 in this location. We also show in Fig. 6.8(a) the localization error results obtained using the correlation-based scheme and results obtained using multilateration method. The plot shows that the correlation-based scheme performs better than the multilateration method in an obstructed environment.

Indoor environment, UNC Charlotte

We also took measurement data in an indoor testbed located in the Embedded Systems laboratory in the Electrical and Computer Engineering department of University of North Carolina at Charlotte. The laboratory contained metallic cubicles, cabinets, robotic devices, computer systems, chairs as well as tables. These obstacles obstruct beacon signals transmitted from BNs to the UNs. We deployed twelve BNs and six UNs in a $25m \times 5m$ area within the laboratory. We also calculated, using measurement data taken from this environment, a Path loss exponent which came out to be 2.34. As expected, the Path loss exponent here was very low and is because of the effects of reflections on beacon signals due to the presence of cubicles, metallic file cabinets, chairs and other metallic objects in the laboratory. Fig. 6.8(b) shows the localization error results obtained using our correlation-based scheme as well as that from the multilateration approach. This plot also shows that the correlation-based scheme performs better than the multilateration method.

Wooded environment, UNC Charlotte

We set up the third experimental testbed at a wooded area located behind the EPIC building at the University of North Carolina at Charlotte. Fig. 6.3(c) shows the network area which contained trees, shrubs and has an undulating landscape. The trees and shrubs obstruct beacon signals transmitted from BNs to the UNs. We deployed twelve BNs and four UNs in a $25m \times 25m$ area. Using measurement data taken from this wooded area, we calculated a Path loss exponent of 3.23. Fig. 6.8(c) shows localization error results obtained using correlation-based scheme and results obtained using the multilateration method. From the plot, we can show that the correlation-based scheme outperforms the multilateration method.

Wooded environment, Simpsonville, SC

We also set up a fourth experimental testbed also in a wooded area but at an off-campus location in Simpsonville, South Carolina. This sensor network area, shown in Fig. 6.3(a) and (c), contains trees and shrubs which obstruct beacon signals from BNs to the UNs. Here, we deployed five UNs and one BN and the BN was moved around to eight or more different positions around the $20 \times 20\ m^2$ area. From the RSSI measurements we took in this location, we calculated the path loss exponent of 2.3. We evaluated our correlation-based localization scheme as well as the multilateration, ROSLAM, OD_CTRD and OD_MSC methods. In Fig. 6.9 we show the localization error results obtained using the correlation-based scheme and localization error results obtained using the other four localization schemes. From the plot, it also shows that the correlation-based scheme outperforms the other localization schemes.

6.2.2. System description

A description of the wireless sensor node system we have used in the experimental testbed is illustrated subsequent sub-sections.

6.2.2.1. Hardware description

In Table 6.1, we describe the hardware components we have used for the experiment. The components include 5 unknown nodes (UN), a beacon node (BN) which also serves as a base station (BS), a MIB510 interface board and finally a laptop computer.

Table 6.1: Hardware descriptions

Hardware	Description
Unknown node	Micaz Motes using IEEE 802.15.4 compliant RF radio platform MPR2400 based on the Atmel ATmega128L, that operate on 2.4 GHz ISM band
Beacon node / Base Station	Micaz Motes using IEEE 802.15.4 compliant RF radio platform MPR2400 based on the Atmel ATmega128L, that operate on 2.4 GHz ISM band
MIB510	MIB510 Mote Interface Board
Laptop computer	Dell XPS L502X with Intel® Core(TM) i7-2670QM CPU @ 2.20 GHz, 8 GB RAM.

6.2.2.2. System Components

A depiction of the experimental testbed system components is shown in Fig. 6.4 while Table 6.2 lists the hardware and their respective functions.

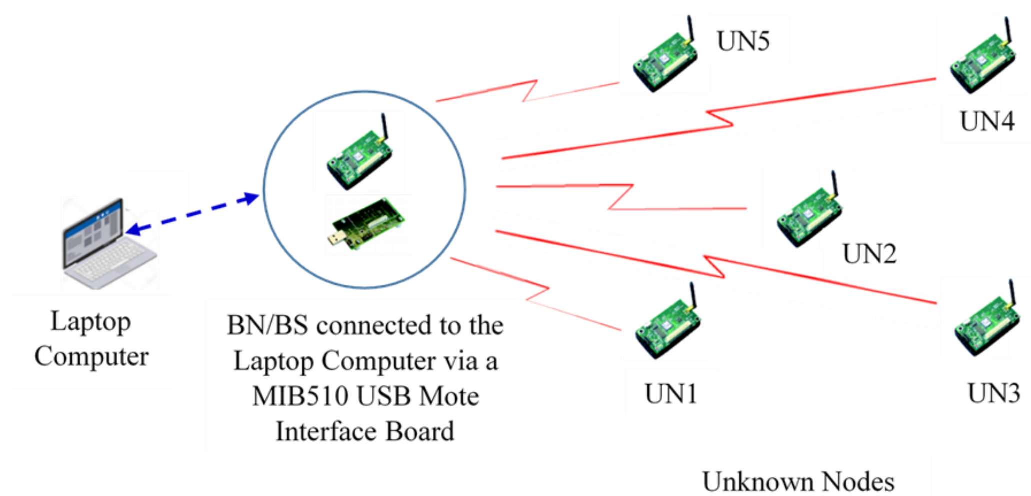


Figure 6.4: System schematic showing the individual components of the experimental testbed: Laptop, BN/BS, UN1 to UN5 and the MIB510 interface board that connects the BS to the laptop for data aggregation.

Table 6.2: System components and Functions.

Hardware	Function
Unknown node	Micaz Motes programmed to transmit beacon signals to the mobile BN.
Beacon node	Micaz Motes programmed to receive beacon signals from UNs at different positions in the network area.
Base Station	Same as the BN and connected to a Dell laptop using the MIB510 interface board.
Laptop computer	BN/BS connects to the Dell XPS L502X laptop through a MIB510 interface board and aggregates RSSI measurements received by the BN and feeds the measurement data to the localization algorithms.

Figure 6.5: Wooded location in Simpsonville, SC. (a) Cross section of the wooded area showing trees and shrubs, (c) deployment area showing the stands on which the sensor motes: BNs and UNs are mounted on, (b) & (d) shows UN1 and UN3 mounted on top of stands to elevate the height of the deployed sensor nodes.

Figure 6.6: Shows the MIB510 Mote Interface Board used in our experiments and a base station connected to it.

Beacon nodes (BN) and sensor motes that are not aware of their locations, i.e. unknown nodes (UNs) were all implemented using Micaz Motes manufactured by Memsic. These low-cost wireless sensors are equipped with IEEE 802.15.4 compliant RF radio platform MPR2400 based on the Atmel ATmega128L, which operate on 2.4 GHz ISM band and RF power set at 0dBm. The UNs were configured to transmit RF signals at intervals of 10 seconds and the BN configured to receive and return RSSI value of the received signals. For aggregating of the sensor data onto a laptop, we used, as shown in Fig. 6.6, a Micaz Mote as base station connected to a Dell laptop using the MIB510 which provides a serial/USB interface for programming and data communications.

We mounted the BN and UNs on top of wooden stands as shown in Fig. 6.5 (a) – (d) and secured with Velcro to avoid fall and damage to the motes. The Omnidirectional dipole antennas of all the sensor motes: BN and UNs alike, were placed in a vertical alignment to ensure consistent polarization of the antennas hence proper receipt of the signals by the UNs. Each of the UNs as we have stated earlier were configured to transmit beacon signals at intervals of 10 seconds. A significant amount of measurement data was collected and processed offline to first calculate the path loss exponent of the location and more importantly to estimate the location of the five UNs.

6.2.3. Experimental Results

Having described the experimental testbed, we will discuss how the measurement data used in evaluation the localization schemes were collected. Then we will show using Fig. 6.9 how the various localization schemes discussed in the course of this dissertation stack up to each other.

6.2.3.1. Measurements and data collection

For the experimental testbeds shown in Fig. 6.3(a) – (c), we statically placed the BNs and UNs as specified in section 6.2. In Fig. 6.8(a) – (c), we show results of these testbeds evaluated for our proposed correlation-based localization scheme and the multilateration scheme. The plots shown here indicates that our correlation-based scheme performed better than the multilateration scheme for all UNs nodes and at all 3 testbeds evaluated. The reason behind the good performance of the correlation-based scheme in obstructed or shadowed environments is that whereas multilateration uses distance estimates from all BNs, whether they are obstructed or not, thereby introduces errors which adversely affects the UNs location estimate, our correlation-based methodically deselects those distance estimates calculated from obstructed BNs, thereby reducing the effects of obstructions in the network region.

However, for the last off campus wooded area testbed, we used a different deployment approach. Here, we used a single BN and moved this BN to several different positions around the perimeter of a 20m x 20m wooded environment where the BN transmits beacon signals. Our main goal is to emulate the movement of a mobile robot around a network region. Fig. 6.7 illustrates a layout of the deployment area showing true locations of the deployed UNs, BN positions and trees that obstruct and hence attenuate beacon signals. The positions around the network region where the BN transmitted beacon signals are:

- BN position 1: {5,0}
- BN position 2: {0,15}
- BN position 3: {20,5}
- BN position 4: {20,15}
- BN position 5: {15,20}
- BN position 6: {5,20}
- BN position 7: {0,15}
- BN position 8: {0,5}.

For the UNs, we statically placed them at the following positions:

- UN position 1: {5,5}
- UN position 2: {10,10}
- UN position 3: {15,5}
- UN position 4: {15,15}
- UN position 5: {5,15}

Fig. 6.5 (a) and (c) show a cross section of the deployment of the sensor motes in the sensor area. Each of the five UNs had an ID tag ranging from 1 to 5 and these IDs matched the sensor node configured on the UN. Similarly the BNs were also labelled with ID tags, ranging from 1 to 8 corresponding to the ID configured on the nodes. The BN is a Micaz mote acting as a sink and connected to the laptop through a MIB510 interface board. The measurement data collection process starts with the BN at position 1 at {5,0} where it sends packets at intervals of 10 seconds. Each of the UNs may or may not receive the transmitted packets depending on the distance between the UN and the BN position as well as the presence of obstacles or lack thereof in the Line of sight between the UN and the BN. The process is repeated at BN positions 2 through 8.

The data collected is aggregated and processed offline on the Dell laptop. Table 6.3 shows average RSSI values received from the UNs at the mobile BN as it is moved from positions 1 through 8 in the network region. The path loss exponent of the testbed environment is calculated. In this case, the path loss exponent calculated was 2.3. The RSSI data collected is used to calculate the distance estimates of each of the UNs to each of the BN positions. The resulting data and path loss exponent are fed into each of the localization schemes for the final location estimate of the five UNs. Then for each localization schemes, we calculated the localization error e_i using the Euclidean distance between the UNs' estimated location estimates $\widehat{xy}_i$ and their true location xy_i as shown in equation 6.1.

$$e_i = \sqrt{(\hat{x}_i - x_i)^2 + (\hat{y}_i - y_i)^2} \tag{6.1}$$

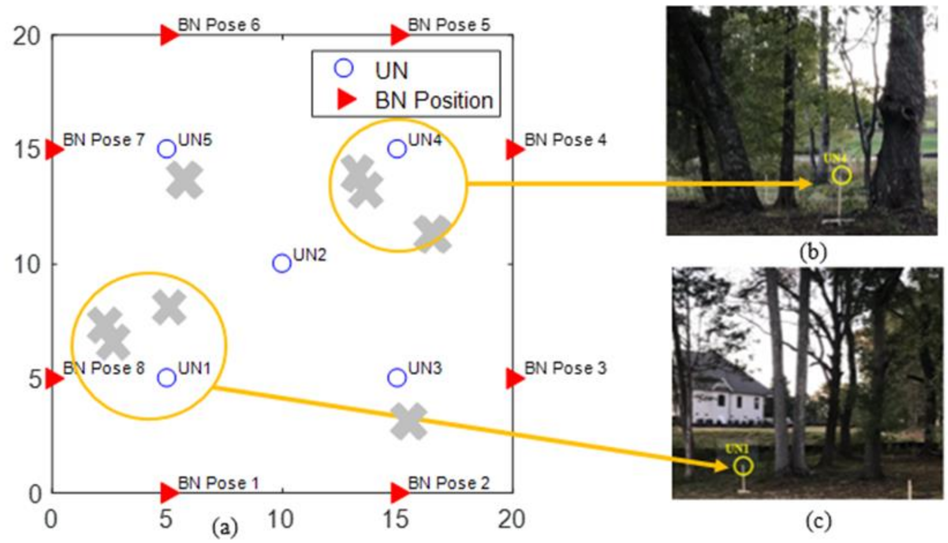


Figure 6.7: (a) Experimental testbed layout showing BNs as red triangles, UNs as blue circles and gray crosses as obstacles (trees), (b) and (c) show UN1 and UN4 obstructed by cluster of trees respectively.

Table 6.3: Average RSSI values received by mobile beacon node from the unknown nodes.

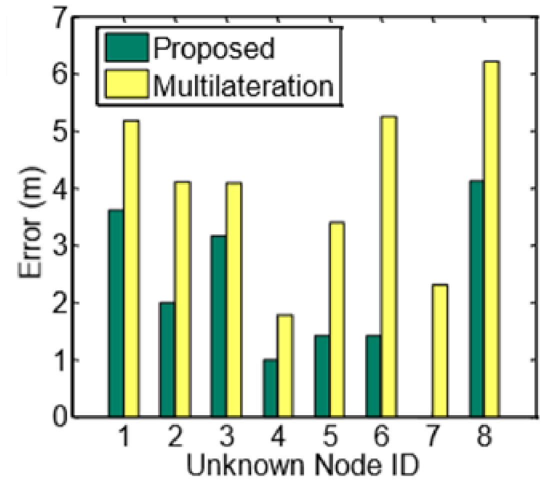
	Average RSSI values (dBm)							
	BN Pose 1	**BN Pose 2**	**BN Pose 3**	**BN Pose 4**	**BN Pose 5**	**BN Pose 6**	**BN Pose 7**	**BN Pose 8**
UN1	-73.23	-84.43	-76.40	-89.07	-87.90	-93.13	-81.30	-63.00
UN2	-76.70	-84.43	-70.93	-84.27	-76.07	-74.20	-73.20	-77.63
UN3	-73.87	-70.10	-66.83	-74.23	-88.63	-88.07	-78.87	-78.10
UN4	-91.47	-76.67	-89.83	-67.53	-68.83	-72.57	-73.80	-90.37
UN5	-75.67	-80.07	-70.30	-82.63	-74.13	-66.00	-59.77	-70.07

Table 6.4: Error in estimated distances from the unknown nodes to the beacon node.

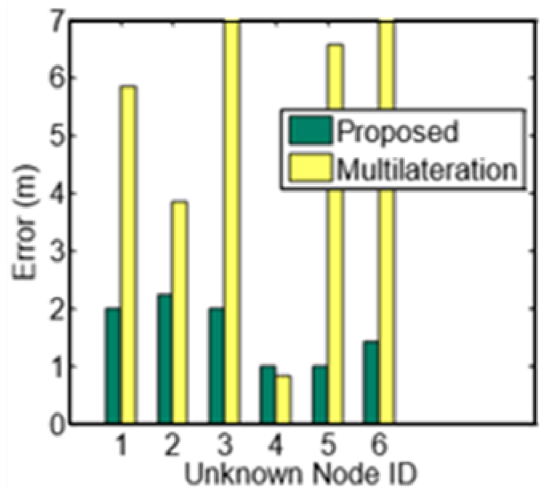
	Distance Estimate Errors (m)							
	BN Pose 1	**BN Pose 2**	**BN Pose 3**	**BN Pose 4**	**BN Pose 5**	**BN Pose 6**	**BN Pose 7**	**BN Pose 8**
UN1	0.98	7.27	6.77	11.39	2.21	29.24	2.29	2.86
UN2	2.70	7.27	6.43	6.97	3.22	4.59	5.22	1.87
UN3	4.80	13.66	1.86	4.57	13.14	8.57	7.48	5.24
UN4	19.41	6.55	20.57	1.63	1.16	5.58	8.67	15.49
UN5	7.35	6.90	13.57	0.39	4.63	2.11	3.45	6.83

6.2.3.2. Experimental result plots

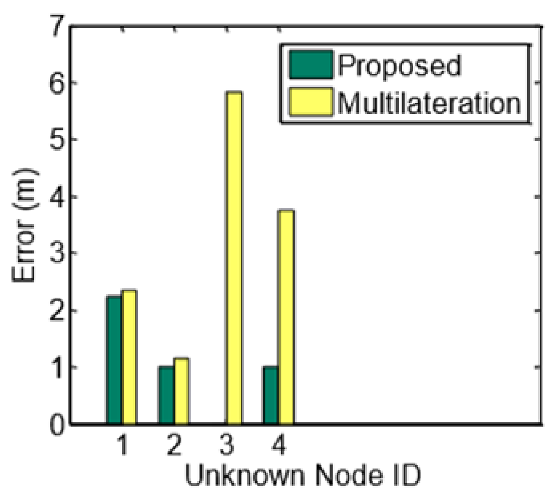
Fig.6.6 shows results of the evaluation of our correlation-based localization and the multilateration schemes. In Fig. 6.9 we show a comparison plot of all the five localization schemes explored.



(a)



(b)



(c)

Figure 6.8: Performance evaluation of Multilateration and correlation-based localization schemes for environments shown in Figure 6.3 (a) –(c) respectively using experimental data.

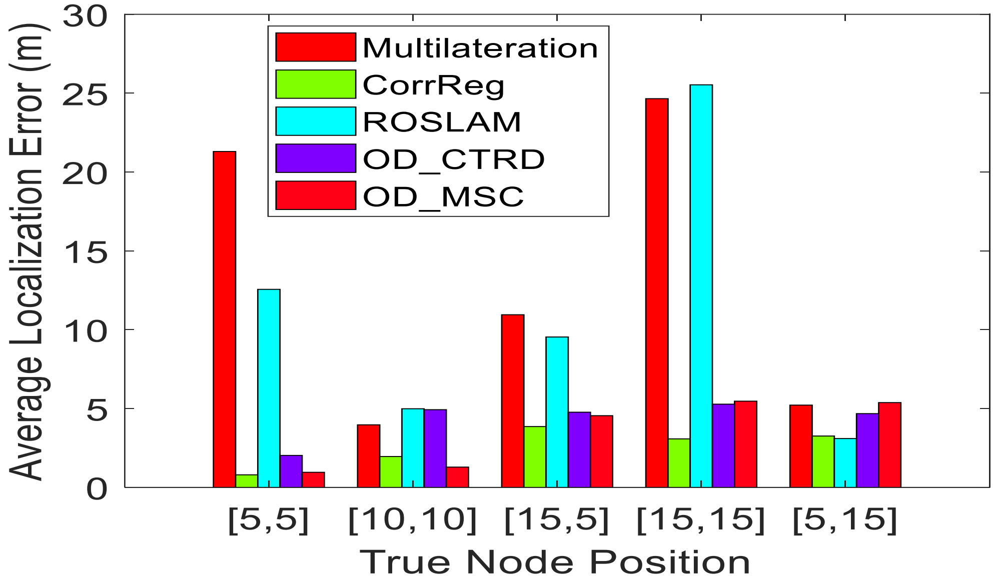


Figure 6.9: Performance evaluation of all five localization schemes localizing five unknown nodes using experimental data.

Results of the accuracy of the location estimates of the five UNs using the five evaluated localization schemes is compared as shown in Fig. 6.9. From the plots, one can observe that the correlation-based localization scheme performs better than the other four evaluated schemes. Ranking the localization schemes according to their total performance for all five UNs localized shows that the correlation-based scheme came out 1st, followed by the OD_MSC and then OD_CTRD, ROSLAM and Multilateration in that order.

The plots show a very high localization error for UN1 (~22m) and UN4 (~25m) for the multilateration. Table 6.4 reveals very high errors in the estimated distances between the following UNs and the BN positions:

- UN1 to BN position 4 (11.39m) and UN1 to BN position 6 (29.24m).
- UN3 to BN position 2 (13.66m) and UN1 to BN position 5 (13.14m).
- UN4 to BN position 1 (19.41m), UN4 to BN position 3 (20.57m) and UN4 to BN position 8 (15.49m).
- UN5 to BN position 3 (13.57m).

Multilateration performed poorly especially for UNs 1 and 4 because they were based on multiple reference distance estimates that had very high errors. The high errors are due to added attenuation to the received beacon signals caused by the clusters of trees around UN1 and UN4 as shown in Fig. 6.7(b) and (c) respectively. The result for UN3 had comparatively lower error since two of the reference distance estimates had high errors but not as high as those for UN1 and UN4. The reference distance error for UN5 was lower, resulting in a smaller error in multilateration. This clearly depicts the problems with multilateration utilizing distance estimates from all the received beacon signals, shadowed and non-shadowed alike, which was the motivation behind our research.

The same can be said for ROSLAM that came in second from the bottom even though it uses the Kalman Gain to weight received measurement data and hence determine whether or not to give the measured data more influence in the location estimation process. Another reason ROSLAM did not perform as well can be attributed to the fact that there was not a significant number of unique locations from which the BN received signals. The top three localization schemes are the ones that employ spatial correlation. As was seen in the simulations shown in previous chapters, the experimental evaluation also proves that the schemes that employ spatial correlation mechanism are effective in deselecting those erroneous location estimates due to shadowing, hence produce better final location estimate for the UNs.

6.3. Performance Evaluation: Conclusion

In analyzing the performance plots from simulations, we have been able to show the effectiveness of the correlation-based localization and also the outlier detection schemes in reducing the effects of shadowing caused by obstacles in wireless sensor networks. We have shown this by comparing the simulations results of all five localization schemes.

Also a summary of our findings from the experimental testbeds shows that the correlation-based localization scheme outperforms all the other evaluated localization schemes, namely: the Multilateration, ROSLAM, OD_CTRD and OD_MSC.

With the evaluation of all five localization schemes, we have shown that the correlation-based localization schemes is very effective in estimating the location of an UN even in an obstructed environment.

We have also shown that generally the three localization schemes that employ spatial correlation performed better than the two that do not. Our findings showed the effectiveness of the spatial correlation mechanism in deselecting shadowed location estimates or outliers and producing more accurate final location estimate of an unknown node.

CHAPTER 7: CONCLUSIONS AND FUTURE WORK

This research addressed practical and effective solutions for improving the localization accuracy of wireless sensor nodes using distance estimates from RSSI that are impacted by obstructions. RSSI is a no-cost method for obtaining distance estimates from a set of reference points or beacons, which is the primary reason behind its popularity for localization in WSNs. The typical challenge for using RSSI is reducing the effect of errors in distance estimates that are unavoidable in terrestrial radio propagation channels characterized by long-term fading. A significant amount of prior research has been reported to find effective solutions for tackling this problem. For instance, the conventional approach of multilateration successfully reduces reasonably small errors in distance estimates by using redundancy and minimizing the mean square error, the performance of which improves with increasing redundancy. However, localization error using these methods can be unacceptably high when there are physical obstructions in the WSN deployment area that cause an inordinate amount of errors in the corresponding distance estimates from RSSI. The corresponding outliers of distance estimates do more harm than good even when redundancy is available. The reason for this adverse effect can be attributed to the fact that multilateration uses distance estimates calculated from all beacon signals: obstructed and unobstructed, in the sensor localization process. This is the primary focus of this research.

Intuition will give thought that if we develop a scheme which can identify and deselect those erroneous distance estimates caused by shadowing, the sensor location estimate will be more accurate. We proposed two approaches aimed at minimizing the effects of obstructed beacon signals to sensor node location estimation in WSN. Both approaches exploit the fact that in a typical WSN, not all beacon signal sources (beacon/anchor nodes/robots) are obstructed. The unobstructed beacon signals essentially confirm the true location estimate within an expected margin of error that is determined by the long-term fading characteristics. However, the unobstructed beacon signals, although not identifiable, essentially have no correspondence to the location estimate agreed upon by the unobstructed beacons. We present approaches to systematically deselect the beacons that disagree with a subset of beacons that have a confirmed opinion.

In our first proposed approach, instead of applying multilateration using beacon signals from all beacon nodes, we apply multilateration using beacon signals from multiple subsets of the beacon nodes. Results from these several multilaterations are combined in such a way that those position estimates that agree, within a given threshold, add up while those that do not agree are outliers and can be ignored. We developed mathematical models to analyze the benefit of our proposed approach. Performance evaluations obtained from theoretical formulation, simulations and experiments demonstrate the benefits of our correlation-based localization scheme over the multilateration method.

Our second approach was based on outlier detection as applied in the field of data science. Specifically, we propose two outlier detection localization schemes named OD_CTRD and OD_MSC. We explored these two outlier detection schemes because they employ clustering mechanisms that aims to influence which location estimates that participate in the final estimation of the sensor node. Again performance evaluations we obtained from simulations and experiments demonstrate the benefits of both OD_CTRD and OD_MSC over the multilateration method.

We also presented and evaluated ROSLAM using EKF, especially in the area of mapping. Our reason for exploring ROSLAM with EKF is because EKF uses a weighting mechanism called Kalman Gain in influencing how measured data influences the sensor location estimate. This Kalman gain function is similar to our correlation-region based weighting mechanism which we used to influence which location estimates are used in the final sensor location estimate. Performance evaluations obtained from simulations and experiments show that using a beacon signals from a small number mobile BN or robot positions does not perform as well as when you use beacon signals from a large number of robot positions. Our results demonstrate the benefits of ROSLAM over the multilateration method.

In summary, we presented an extensive set of approaches for localization in WSNs for effectively dealing with obstructions of beacons signals. This includes two proposed modifications to multilaterations and a popular approach using EKF. We present extensive performance evaluations of all five schemes discussed using simulations as well as experimental testbeds. Performance was evaluated from the perspective of localization errors as well as computational complexity/cost (computation time) of implementation. Our results show that the correlation-based localization scheme (corrReg) performs better than all the other four schemes, with the OD_MSC and OD_CTRD coming second and third respectively. The three proposed localization schemes performed better than the multilateration approach but with an increase in computation cost: corrReg and OD_MSC schemes having marginal increase in computation time.

We will outline below: the main goal of this research, how we have met the goal and the contributions of our work to the WSN localization research community.

Minimize the effects of shadowing caused by obstructions to beacon signals to sensor localization:

Although several approaches exist, our proposed approaches are important due to the following:

- We presented effective and inexpensive range-based localization approaches to minimize the effects of shadowing caused obstacles in a sensor network environment on sensor node localization.
- Unlike most of the state of the art range-based localization approaches, our proposed schemes do not require extra (or specialized) hardware that raises the cost and/or size of the sensor nodes.

Following the conclusions we have drawn from the work we have presented, further research in this area has been identified. Further work to be done include:

- Further exploration of effective method(s) to reduce computational complexity of the OD_CTRD localization method.

- Explore using Particle Filter (PF) and the correlation region concept in localization sensor nodes in obstructed WSNs. Develop mathematical model to utilize correlation regions in calculating weights of particles and finally simulate the correlation region-based PF Localization Algorithms where the weights of the particles are calculated using correlation regions.

REFERENCES

[1] Savvides, C.-C. Han, and M. B. Srivastava, “Dynamic fine-grained localization in ad-hoc networks of sensors,” in MOBICOM, 001, pp.166–179.

[2] Y.-B. Kong, Y.-G. Kwon, and G.-T. Park, “Practical robust localization over obstructed interferences in wireless sensor networks,” in ICCE, 009.

[3] B. Xiao, H. Chen, and S. Zhou, “Distributed localization using a moving beacon in wireless sensor networks,” IEEE Trans. Parallel Distrib. Syst., vol. 19, no. 5, pp. 587–600, 2008.

[4] C. Wang, L. Xiao, and R. Jin, “Sensor localization in an obstructed environment,” in International Conference on Distributed Computing in Sensor Systems, 2005.

[5] N. Bulusu, J. Heidemann, and D. Estrin, “GPS-less low cost outdoor localization for very small devices,” IEEE Personal Communications Magazine, 2000.

[6] N.Chuku, A.Pal and A.Nasipuri, “An RSSI Based Localization Scheme for Wireless Sensor Networks to Mitigate Shadowing Effects” 013 IEEE SouthEastCon paper presentation.

[7] R. Peng and M. Sichitiu,”Probabilistic localization for outdoor wireless sensor networks", ACM SIGMOBILE Mobile Comput. Commun. Rev., vol. 11, no. 1, pp.53 -64 2007.

[8] V. Ramadurai and M. L. Sichitiu, “Localization in wireless sensor networks: A probabilistic approach”, In Int’t Conf. on Wireless Networks (ICWN), pages 275–281, June 2003.

[9] S.-P. Kuo, Y.-C. Tseng, F.-J. Wu, and C.-Y.Lin.,”A probabilistic signal-strength-based evaluation methodology for sensor network deployment”, In Proc. Int. Conference on Advanced Information Networking and Applications (AINA), pages 319–324, 2005.

[10] C. Papamanthou, F.P. Preparata and R. Tamassia,” Efficient Localization for Wireless Sensor Networks Using Power Measurements Sampling”, Int. J. of Ad Hoc and Ubiquitous Computing.

[11] B. Xiao, H. Chen, and S. Zhou, “Distributed localization using a moving beacon in wireless sensor networks,” IEEE Trans. Parallel Distrib. Syst., vol. 19, no. 5, pp. 587–600, 2008.

[12] A. Nasipuri, “Location Discovery in Sensor Networks”, in Algorithms and Protocols for Wireless Ad Hoc and Sensor Networks, edited by Azzedine Boukerche, Wiley & Sons, ISBN 978-0-471-79813-2, pp. 341 - 364, 2009.

[13] Bachrach J., Taylor C., “Localization in Sensor Networks”, In I. Stojmenović (Ed.), Handbook of sensor networks: algorithms and architectures, John Wiley & Sons, Inc. 277-310, 2005.

[14] B. Dil, S. Dulman, and P. Havinga, “Range-based localization in mobile sensor networks”, in Proc. European Wireless Sensor Networks (EWSN), pages 164–179, 2006.

[15] D. Niculescu and B. R. Badrinath, “Ad hoc positioning system (APS) using AOA”, in Proc. IEEE Conference on Computer Communications (INFOCOM), 2003.

[16] A. Gasparri, S. Panzieri, F. Pascucci, and G. Ulivi, “An Interlaced Extended Kalman Filter for sensor networks localization. Int. J. Sen. Netw. 5, 3 (June 2009), 164-172.

[17] H. Durrant-Whyte and T. Bailey, “Simultaneous localization and mapping (SLAM): Part I, the essential algorithms,” IEEE Robotics and Automation Magazine, vol. 13, no. 2, pp. 99–110, Jzn. 2006.

[18] Jose-Luis Blanco, Juan-Antonio Fernández-Madrigal, Javier González, “Efficient Probabilistic Range-Only SLAM.” IEEE/RSJ IROS 2008.

[19] S. Thrun, W. Burgard and D. Fox, “Probabilistic Robotics”, 3rd ed.; The MIT Press: Cambridge, UK, 2005.

[20] Cyrill Stachniss, “Photogrammetry II – 13 – EKF SLAM (2015/2016)”, YouTube videos.

[21] J. Djugash, S. Singh, G. Kantor, and W. Zhang, “Range-only SLAM for robots operating cooperatively with sensor networks”, in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2006, pp. 2078–2084.

[22] J. Aulinas, Y. Petillot, J. Salvi, and X. Llado, “The SLAM Problem: A ´ Survey”, In Proceedings of the International Conference of the Catalan Association for Artificial Intelligence, pages 363–371. IOS Press, 2008.

[23] G. Bresson, Z. Alsayed, L. Yu, and S. Glaser, “Simultaneous localization and mapping: A survey of current trends in autonomous driving,” IEEE Transactions on Intelligent Vehicles, vol. 2, no. 3, pp. 194–220, September 2017.

[24] F. Caballero, L. Merino, I. Maza, and A. Ollero, “A particle filtering method for wireless sensor network localization with an aerial robot beacon”, In Proc. of IEEE ICRA, pages 596–601, 2008.

[25] A. A. Kannan, G. Mao and B. Vucetic, “Simulated annealing based localization in wireless sensor network”, In Proceedings of the IEEE Conference on Local Computer Networks 30th Anniversary (LCN’05), Sydney, Australia, 17 November 2005.

[26] King-Yip Cheng, King-Shan Lui and Vincent Tam, "Localization in Sensor Networks with Limited Number of Anchors and Clustered Placement", in Proceedings of Wireless Communications and Networking Conference, 2007 (IEEE WCNC 2007), March 2007, pp. 4425 – 4429. http://dx.doi.org/10.1109/WCNC.2007.806

[27] Y. Shang, W. Ruml, Y. Zhang, and M. Fromherz, "Localization from mere connectivity", In Proceedings of ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc'03), June 2003, Annapolis, Maryland, USA, pp.

[28] Cesare Alippi, Giovanni Vanini, "A RSSI-based and calibrated centralized localization technique for Wireless Sensor Networks", in Proceedings of Fourth IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOMW'06), Pisa, Italy, March 2006, pp. 301-305.

[29] Anushiya A Kannan, Guoqiang Mao and Branka Vucetic, "Simulated Annealing based Wireless Sensor Network Localization", Journal of Computers, Vol. 1, No. 2, pp 15-22, May 2006.

[30] Y. Wang, X. Wang, D. Wang, and D. P. Agrawal, "Range-free localization using expected hop progress in wireless sensor networks," IEEE Trans. Parallel Distrib. Syst., vol. 20, no. 10, pp. 1540–1552, 2009.

[31] N. Patwari, A. O. H. III, M. Perkins, N. S. Correal, and R. J. O'Dea, "Relative location estimation in wireless sensor networks," IEEE Transactions on Signal Processing, vol. 51, no. 8, pp. 2137–2148, 2003.

[32] C.-C. Pu and W.-Y. Chung, "An integrated approach for position estimation using rssi in wireless sensor network," Journal of Ubiquitous Convergence Technology, 2008.

[33] A. Awad, T. Frunzke, and F. Dressler, "Adaptive distance estimation and localization in wsn using rssi measures," in DSD, 2007, pp. 471–478.

[34] M.L. Ruz, J. Garrido, J. Jiménez, R. Virrankoski, and F. Vázquez, "Simulation Tool for the Analysis of Cooperative Localization Algorithms for Wireless Sensor Networks", Sensors (Basel, Switzerland), 2019.

[35] A. Nasipuri and K. Li, "A directionality based location discovery scheme for wireless sensor networks," in WSNA, 2002, pp. 105–111.

[36] H. Karl, A. Willig, "Protocols and Architectures for Wireless Sensor Networks", first ed., Localization and Positioning, Wiley, 2005. June 24 (Chapter 9).

[37] C. Savarese, J. Rabay, and K.langendoen, "Robust Positioning Algorithms for Distributed Ad-Hoc Wireless Sensor Networks" In Proceedings of the Annual USENIX Technical Conference, Monterey, CA, 2002.

[38] N. Chuku and A. Nasipuri. 2014. "Performance evaluation of an RSSI based localization scheme for wireless sensor networks to mitigate shadowing effects". In IEEE Wireless Communications and Networking Conference (WCNC'14).

[39] N. Chuku and A. Nasipuri. 2019. "Wireless Sensor Localization Using Outlier Detection". In 2019 IEEE 16th International Conference on Smart Cities: Improving Quality of Life Using ICT & IoT and AI (HONET-ICT)

[40] N, J. Ash, S. Kyperountas, R. M. Moses, A. O. Hero III, and N. S. Correal, "Locating the nodes: Cooperative localization in wireless sensor networks," IEEE Signal Processing Mag., vol. 22, no. 4, pp. 54–69, July 2005.

[41] J. Blanco, J. Gonzalez, and J. Fern´andez-Madrigal, "A Pure Probabilistic Approach to Range-Only SLAM," in Proceedings of the IEEE International Conference on Robotics and Automation, 2008.

[42] A. Doucet, N. de Freitas, K. Murphy, and S. Russell, "RaoBlackwellised particle filtering for dynamic Bayesian networks," in Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence, 2000, pp. 176–183.

[43] K. Fukunaga, and L. Hostetler, "The estimation of the gradient of a density function, with applications in pattern recognition". IEEE Transactions on Information Theory, 21(1), 32–40, 1975.

[44] N. Ohashi, Y. Funabora, S. Doki and K. Doki," Majority Rule Sensor Fusion System with Particle Filter for Robust Robot Localization" Proceedings of the 2018 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM), Auckland, New Zealand, July 9-12, 2018.

[45] L. Jian, Z. Yang and Y. Liu," Beyond Triangle Inequality: Sifting Noisy and Outlier Distance Measurements for Localization" in Proc. IEEE INFOCOM, 2010, pp. 1–9.

[46] T. Ahmad, X. Jun Li and B. C. Seet, ''3D Localization Using Social Network Analysis for Wireless Sensor Networks,'' In Proceedings of the 2018 IEEE 3rd International Conference on Communication and Information Systems (ICCIS), Singapore, 28-30 December 2018; pp. 88-92.

[47] E. Menegatti, A. Zanella, S. Zilli, F. Zorzi, and E. Pagello, "Range-only SLAM with a mobile robot and a Wireless Sensor Networks," in Proc. IEEE Int. Conf. Robot. Autom., 2009, pp. 8–14.

[48] A. Torres-González, J. R. Martinezdedios, and A. Ollero, "Range-only SLAM for robot-sensor network cooperation," Auton. Robots, vol. 42, no. 3, pp. 649–663, 2018.

[49] G.Grisetti, G.D.Tipaldi, C.Stachniss, W.Burgard, and D.Nardi, "Fast and accurate SLAM with rao-blackwellized particle filters". Journal on Robotics and Autonomous Systems (RAS), 55(1):30–38, 2007.

[50] J. Huang, D. Millman, M. Quigley, D. Stavens, S. Thrun, and A. Aggarwal, “Efficient, generalized indoor wifi graphslam”. In IEEE International Conference on Robotics and Automation, ICRA 2011, Shanghai, China, 9-13 May 2011, pages 1038–1043. IEEE, 2011.

[51] J. MacQueen,, “Some Methods for Classification and Analysis of Multivariate Observations,” Proc. of rhe Fifh Berkeley Symposium on Math., Srar. And Prob., Vol. 1, pp. 281-296, 1967.

[52] A.W.F. Edwards, and L. L. CavalliSf orza, “A method for cluster analysis.” Biometrics 3, 362-75, 1965.

[53] J. H. WOLFE, “Pattern Clustering by Multivariate Mixture Analysis”, Multivariate Behavioral Research, 5, 329–350, 1970.

[54] G. Gan, C. Ma, and J. Wu, 2007, “Data Clustering: Theory, Algorithms, and Applications (ASA-SIAM Series on Statistics and Applied Probability). SIAM”.

[55] M. Ester, H. Kriegel, J. Sander, and X. Xu, “A density-based algorithm for discovering clusters in large spatial databases with noise,” in Proc. 2nd Int. Conf. Knowledge Discovery and Data Mining (KDD’96), 1996, pp. 226–231.

[56] G. Carlsson and F. M´emoli, “Characterization, stability and convergence of hierarchical clustering methods.”, Journal of Machine Learning Research 11, 1425–1470 (2010)

[57] R. Davé and R. Krishnapuram, “Robust clustering methods: A unified view,” IEEE Trans. Fuzzy Syst., vol. 5, no. 2, pp. 270–293, May 1997.

[58] A. Geva, “Hierarchical unsupervised fuzzy clustering,” IEEE Trans. Fuzzy Syst., vol. 7, no. 6, pp. 723–733, Dec. 1999.

[59] R. Hammah and J. Curran, “Validity measures for the fuzzy cluster analysis of orientations,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 22, no. 12, pp. 1467–1472, Dec. 2000.

[60] A. Miguel Carreira-Perpinan, “A Review of Mean-shift Algorithms for Clustering”, Handbook of Cluster Analysis, CRC, 2015.

[61] D. Comaniciu and P. Meer, "Mean shift: a robust approach toward feature space analysis," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 5, pp. 603-619, May 2002.

[62] K. G. Derpanis. Mean shift clustering. Tutorial, 2005.

[63] http://www.thphys.nuim.ie/CompPhysics/matlab/help/techdoc/umg/chpatch4.html

[64] https://www.mathworks.com/help/matlab/ref/cylinder.html.

[65] http://www.memsic.com/userfiles/files/Datasheets/WSN/micaz_datasheet-t.pdf

[66] L. Génevé, O. Kermorgant, E. Laroche, "A composite beacon initialization for EKF range-only SLAM", in: IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2015.

[67] H. Karl, A. Willig, "Protocols and Architectures for Wireless Sensor Networks", first ed., Introduction, PatwariWiley, 2005. June 24 (Chapter 1).

[68] A. Nasipuri, R. Cox, J. Conrad, L. V. der Zel, B. Rodriguez, and R. McKosky, "Design considerations for a large-scale wireless sensor network for substation monitoring," in LCN, 2010, pp. 866–873

[69] N. Priyantha, A. Miu, H. Balakrishnan & S. Teller, "The Cricket Compass for Context-Aware Mobile Applications", Proceedings of the Annual International Conference on Mobile Computing and Networking, MOBICOM 2001. 10.1145/381677.381679.

[70] T. He, C. Huang, B. Blum, J. Stankovic, and T. Abdelzaher, "Range-free localization schemes in large scale sensor networks", In Proceedings of the Ninth Annual International Conference on Mobile Computing and Networking (MobiCom'03), September 2003, San Diego, CA, USA.