2014

Rss Based Localization In A Rayleigh Fading Environment

Rojina Adhikary
University of Mississippi

RSS BASED LOCALIZATION IN A RAYLEIGH FADING ENVIRONMENT

A Thesis
Presented for the
Master of Science Degree in Engineering
Science with Emphasis in
Telecommunications
The University of Mississippi

Rojina Adhikary

MAY, 2014

ABSTRACT

The objective of this thesis is to quantify the improvement that can be obtained in sensor agent localization accuracy as a function of the number of multipath components that can be resolved. We assume that a known number sensor agents are located at unknown coordinates within a rectangular grid having anchors at the corner locations, whose locations are known. Further, we assume fading is Rayleigh and that the propagation constant is constant but unknown. Also, we assume that modulation is spread spectrum and that either the sensors or agents are capable of resolving multipath components down to the chip level and are capable of measuring the received signal strength in each of the resolved multipath components. An error function is formulated based upon the square of the distances between the actual sensor locations and their model-predicted locations, which are functions of the received signal strength of the various multipath components and the propagation constant, and the optimal sensor location estimates and propagation constant are determined through a multistage process of formulating and minimizing error functions. The effectiveness of this approach is investigated via extensive simulations in which the Saleh-Valenzuela model is used to generate multipath components. The simulation results indicate that for a given fixed propagation constant, resolving multipath results in improved localization accuracy and that this improvement is a non decreasing function of the propagation constant. For a distance-squared propagation environment, the results indicate that resolving 6 multipath components improves localization accuracy by at least 20%, the improvement being with respect to the localization accuracy based on aggregate received signal strength.

ACKNOWLEDGEMENTS

First and foremost, I would like to express my sincere gratitude to my advisor, Dr. John N. Daigle, for the continuous support, guidance and encouragement during the course of this work. I am especially grateful for his insightful suggestions and inputs over numerous discussions, which helped me understand the research problems clearly and carrying out the further research with much enthusiasm. My heartfelt thank goes to him for the opportunity that he provided to me to start my masters here at OleMiss.

I would also like to thank Dr. Lei Cao and Dr. Ramanarayanan"Vish" Viswanathan for consenting to be a member on the committee for my master's defense.

Lastly, I would like to thank my family. Their love and continuous support have helped me come to this point of my life. I thank my husband and my best friend Amrit for all his support, advice and love.

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1

INTRODUCTION

This work seeks to quantify the improvement that can be obtained in localization of sensor nodes in a wireless sensor network as a function of number of resolvable multipath components. We assume that multipath components arrive in clusters and fading follows the Rayleigh model. We also assume that modulation technique used is direct sequence spread sprectrum, and that the sensors can resolve the multipath components down to the chip level and measure the received signal strength in each of the resolved multipath components. We propose a method for localization that includes resolving multipaths and reflecting the measurements to the minimum delay path. A cost function is formulated and minimized to obtain an estimate of the agent location and propagation constant of the environment.

The goal of localization is to determine the physical coordinates of a group of sensor nodes in a wireless sensor network. In any wireless sensor network, there are a number of anchor nodes, also known as *beacon* nodes, and multiple agent nodes. Anchor nodes are simply ordinary sensor nodes that know their global coordinates a priori. This knowledge could be hard coded, or acquired through some additional hardware such as a GPS receiver. Based on the location information of the anchor nodes, other ordinary sensor nodes attempt to localize themselves; such nodes are called *agent* nodes. As a specific example, suppose we have a wireless sensor network with anchor nodes and agent nodes, then each of the agent's objective is to determine its exact location in global coordinates by using the global position information and received signal strength from anchor nodes.

We assume that a known number of agent nodes are located at random two-dimensional coordinates within a rectangular grid and anchors at the corners, whose locations are known. We assume that the propagation constant is an unknown constant value. An error function is formulated based upon the square of the distances between the actual sensor locations and their model-predicted locations, which are functions of the received signal strength of the various multipath components and the propagation constant. The optimal sensor location estimates and propagation constant are determined through a multistage process of formulating and minimizing cost functions. The Saleh-Valenzuela channel model which is a statistical multipath model, is used to generate the multipath components and path gains of each of the components.

The localization accuracy of this approach is investigated via extensive simulations for the single agent localization and the multiple agent localization case. The simulation results indicate that for a given fixed propagation constant, resolving multipath results in improved localization accuracy. Improvement in localization accuracy was found to be a non decreasing function of number of multipaths that can be resolved and propagation constant of the environment. We found that improvement due resolving additional multipath components are marginal beyond 7 components.

1.1 Motivation and goals

Location-awareness is essential for many wireless network applications, such as the localization service in next generation cellular networks [1], search-and-rescue operations [2], [3], logistics [4], and blue force tracking in battlefields [5] and many more. Each node is equipped with a transmitter, receiver or a transceiver, and localization is accomplished using signals passed between agents and their neighboring anchors. Since resources such as available power, bandwidth, size of the nodes and the equipments installed in a given node in a network are limited, one may consider efficient methods of localizing agent nodes that does not require bulky and expensive equipments. There are numerous RSS based localization

techniques that have very large localization errors and similarly there are many localization techniques that localize agent nodes efficiently however have computational complexity, require heavy and expensive processors and other equipments in sensor node. We investigate a localization method that uses received signal strength information obtained from anchor nodes, a simple efficient numerical computation, and does not require additional equipments in sensors for processing.

The motivation of this thesis is to achieve higher localization accuracy for agent nodes in a wireless sensor network using the received signal strength information, and to explore the means by which they could be achieved in a simple decentralized fashion without the burden of heavy computations and complex centralized network schemes. We then evaluate the localization performance obtained from our methodology as a function of number of resolvable multipaths.

1.2 State of the art

Location estimation is an important task in wireless cellular and sensor networks [6]. Most location technologies are based on either time-of-arrival (TOA), time-difference-of-arrival (TDOA), angle-of-arrival (AOA), or received-signal-strength (RSS) measurements [6], [7], [8], [9]. The RSS approach exploits signal attenuation with distance to determine the source location. This technique has been employed in both cellular and sensor networks utilizing radio-frequency (RF) fading channel [10]. It is inexpensive compared with TOA/TDOA and AOA-based approaches has been identified as the prime candidate [9] for location estimation in wireless sensor networks.

In a wireless communication system the signals may travel through multiple paths between a transmitter and a receiver [11]. This effect is called a multipath propagation. Due to the multiple paths, the receiver of the signal will observe variations of amplitude, phase and angle of arrival of the transmitted signal. These variations originate the phenomenon referred as multipath fading. The variations are characterized by large-scale and small-scale

fading. The large-scale fading, refers to path loss caused by the effects of the signal traveling over large areas. It characterizes the losses due to considerably big physical objects in the signals path like hills or forests. On the other hand small-scale fading characterizes the effects of small changes in the separation between a transmitter and a receiver. These changes can be caused by mobility of the transmitter, receiver or the intermediate objects in the path of the signal. Small scale changes result in considerable variations of signal amplitude and phase. Small-scale fading is also known as Rayleigh fading since the fluctuation of the signal envelope is Rayleigh distributed when there is no predominant line of sight between the transmitter and receiver. When there is a predominant line of sight between the transmitter and receiver the fluctuations are statistically described by a Rician distribution [12].

Fundamental limits of localization accuracy of wideband wireless networks in harsh multipath environments that can be achieved with wideband localization is proposed in [13]. The localization accuracy is characterized in terms of a performance measure called the squared position error bound (SPEB), and the notion of equivalent Fisher information has been proposed to derive the SPEB in expressions. The methodology provides insights into the essence of the localization problem by unifying localization information from individual anchors and information from a priori knowledge of the agents position in a canonical form. Their analysis begins with the received waveforms themselves rather than utilizing the signal metrics extracted from these waveforms. The resulting SPEB serves as a fundamental limit of localization accuracy [13].

Maximum likelihood estimator for ranging measurements exploiting the received signal strength is proposed in [14]. The bias and uncertainties of the RSS based ranging procedure are analyzed, considering a path loss model of an indoor ultra-wideband (UWB) network under line of sight (LOS) conditions. The nonlinearity of the path loss model is first taken into account and the statistics of the observed RSS are approximated by a Taylor sequence of first order. The obtained metric describes the weighted least squares method, and the metrics of the estimator are analytically derived in closed form. The performance of the

derived estimator is investigated in Monte-Carlo simulations and compared with a simple least squares (LS) method and another method exploiting RSS fingerprints.

1.3 Outline of the thesis

The thesis is organized as follows. The next chapter is detailed study and analysis of the Saleh-Valenzuela Channel model. Chapter 3 contains details of the localization techniques that we have proposed. It presents detailed description of the approach and framework that we have developed for localization. The analysis and performance in terms of number of resolvable multipaths, propagation constant and the number of agent nodes, is demonstrated by computer simulations and numerical plots in chapter 4. Chapter 5 concludes the thesis.

Chapter 2

THE SALEH-VALENZUELA CHANNEL MODEL

This chapter presents an overview of the channel model [15] we are using in our thesis. Its detailed literature review and technical overview is presented, and concepts and terminology related with the work is explained in detail.

The Saleh-Valenzuela (SV) indoor channel model was proposed as the prototype for the IEEE 802.15.3a reference channel. This model is a statistical multipath model, whose formulation was based on the results of multipath delay spread and attenuation measurements within a office building and some other indoor environments, carried out by Adel A. M. Saleh and Reinaldo A. Valenzuela in 1987. This model is extendable (by adjusting its parameters) to represent channels within other buildings and outdoor environment.

The Saleh-Valenzuela channel model is a statistical multipath model whose basic assumption is that multipath components (MPCs) arrive in clusters, formed by the multiple reflections from the objects in the vicinity of receiver and transmitter. The channel is represented by multiple paths or rays that have positive gains $\{\beta_k\}$, propagation delays $\{\tau_k\}$, and associated phase shifts $\{\theta_k\}$, where k is the path index; in principle, k extends from 0 to ∞.

The complex, low-pass impulse response of the channel is given by

$$h(t) = \sum_{l=0}^{\infty} \sum_{k=0}^{\infty} \beta_{kl} \exp^{j\theta_{kl}} \delta(t - T_l - \tau_{kl}), \tag{2.1}$$

where l is the cluster index, T_l is the arrival time of the lth cluster, τ_{kl} is the arrival time of the kth ray measured from the beginning of the lth cluster, while β_{kl} and θ_{kl} are the gain and phase of the kth ray of the lth cluster. In principle, rays and clusters extend over an infinite time, as signified by the double infinite sum.

The rays have independent uniform phases, and independent Rayleigh amplitudes with variances that decay exponentially with cluster and ray delays [15]. The clusters, and the rays within the cluster, arrive according to Poisson processes with different rates and have inter arrival times that are exponentially distributed. The cluster arrival times T_l are modeled as a Poisson arrival process with fixed rate Λ and ray arrival time τ_{kl} are modeled as another Poisson process with fixed rate λ. T_l and τ_{kl} are described by the independent inter arrival exponential probability density functions

$$f_{\tilde{T}_l}\left(T_l \mid T_{l-1}\right) = \Lambda \exp\left[-\Lambda\left(T_l - T_{l-1}\right)\right], \quad l > 0, \tag{2.2}$$

$$f_{\tilde{\tau}_{kl}}\left(\tau_{kl} \mid \tau_{(k-1)l}\right) = \lambda \exp\left[-\lambda\left(\tau_{kl} - \tau_{(k-1)l}\right)\right], \quad k > 0, \tag{2.3}$$

where Λ and λ is the cluster and ray arrival rate, respectively and since each cluster contains many rays, $\lambda >> \Lambda$.

The $\{\theta_{kl}\}$ are statistically independent uniform random variables over $[0, 2\pi)$ and the corresponding path gains $\{\beta_{kl}\}$ are statistically independent positive random variables, whose mean square values $\overline{\beta_{kl}^2}$ are monotonically decreasing functions of $\{T_l\}$ and $\{\tau_{kl}\}$

$$\begin{aligned}\overline{\beta_{kl}^2} &= \overline{\beta^2\left(T_l, \tau_{kl}\right)} \\ &= \overline{\beta^2\left(0,0\right)} \exp^{-T_l/\Gamma} \exp -\tau_{kl}/\gamma,\end{aligned} \tag{2.4}$$

where $\overline{\beta^2\left(0,0\right)} = \overline{\beta_{00}^2}$ is the average power gain of the first ray of the first cluster, and Γ and γ are power-decay constants for the clusters and rays, respectively. This $\overline{\beta^2\left(0,0\right)}$ is directly

related to the average multipath power gain

$$\overline{G}\,(r) = G\,(1m)\; r^{-\alpha},$$

where $G\,(1m)$ can be approximated by

$$G\,(r) = \frac{G_T\, G_R\, \lambda_0}{4\,\pi\, r^2}, \tag{2.5}$$

with $r = 1\,m$, where G_T and G_R are the gains of the transmitting and receiving antennas, λ_0 is RF wavelength, and r is the antenna separation and α is the propagation constant which is different for different environments. Now,

$$\begin{aligned} \overline{G}\,(r) &= \lambda \int_0^\infty \overline{\beta^2\,(t)}\, dt \\ &\simeq \gamma\lambda\, \overline{\beta^2\,(0,0)} \sum_{l=0}^{L} \exp^{-T_l/\Gamma}. \end{aligned} \tag{2.6}$$

The summation term in (2.6) is usually dominated by the first term, ie. first cluster. By using (2.2), we can see that the average value of the summation is $1+\Gamma\,\Lambda$. The 1 corresponds to the first cluster and $\Gamma\,\Lambda$ to the subsequent clusters. It is found that $1 >> \Gamma\,\Lambda$, hence the $\Gamma\,\Lambda$ due to subsequent clusters can be safely neglected, and we get

$$\overline{\beta^2\,(0,0)} \approx (\gamma\lambda)^{-1}\; G\,(1m)\; r^{-\alpha}. \tag{2.7}$$

The power decays exponentially with cluster decay as well as excess delay within a cluster as in Figure 2.1.

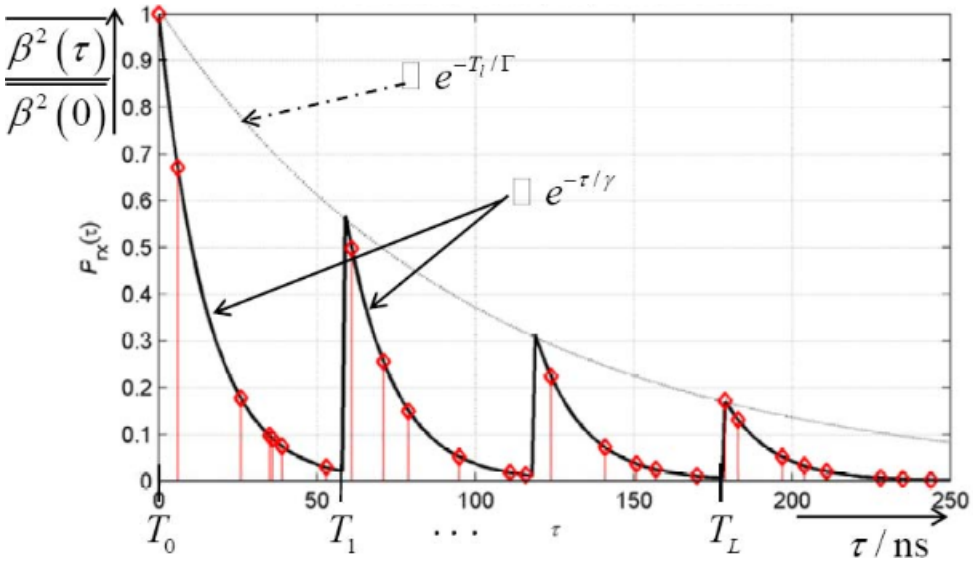

Figure 2.1. A schematic representation of the model: exponentially decaying ray and cluster average powers.

2.1 Model parameters and physical interpretations

In this section, a brief introduction of the parameters of the channel model and their physical meaning is presented. The first arriving cluster of rays is formed by the transmitted wave following a more-or-less direct path to the receiver. Such a path, which is not usually a straight line, goes through a few, but not too many reflecting surfaces. Subsequent clusters result from reflections from the building superstructures, large walls, trees, cars, etc. The first cluster is always present, however subsequent clusters depends on the type environment. The number of clusters is a function both of the measurement bandwidth and of the considered environments. The number of clusters in indoor environment varies from 1 to 4 for 2.4 GHz bandwidth around a 5 GHz carrier in indoor environments and an average of 5 clusters in office environments, however up to 14 clusters are found in outdoor environments [16].

SV model assumes that the ray arrival rate for all the clusters are same, that is, all the clusters have same λ. However, some UWB measurements indicate that the arrival rate is larger for later clusters [16]. By resolving the individual rays in about 200 power profile measurements, the $1/\lambda$ is estimated to be in the range 5-10 ns. The range unceratinity came

from the fact that the ray-resolving algorithm coupled with the measurement sensitivity was unable to detect many weak rays, those falling near strong rays.

The expected value of the ray power as a function of time, measured from the arrival point of the first ray of the first cluster, is given by

$$\overline{\beta^2(t)} = \overline{\beta^2(0,0)} \sum_{l=0}^{L} \exp^{-T_l/\Gamma} \exp^{-(t-T_l)/\gamma} \mathrm{u}(t-t_l), \tag{2.8}$$

where $\mathrm{u}(t)$ is the unit step function.

The power level of each successive ray reaching the receiver is found to be proportional to the time delay of that ray, which results in exponential power decay characteristic as obtained in SV channel model.

The expected value of the path power gain was expressed as a function of the associated cluster and ray delays T_l and τ_{kl}. A new assumption is then made, which is reasonably supported by the observations, that the probability distribution of the normalized power gain is independent of the associated delays, or for that matter, of the location. The cumulative distribution of normalized power gain $\beta_{kl}^2/\overline{\beta_{kl}^2}$, is unity-mean exponential cumulative distribution, which results in an exponential probability density function,

$$f_{\tilde{\ell}}(\ell;\sigma) = \frac{1}{\sigma} \exp(-\ell/\sigma), \tag{2.9}$$

for the path power gain, where the $\tilde{\ell} = \tilde{z}^2 = \tilde{\beta}_{kl}^2$ and $\mathrm{E}\left[\tilde{\ell}\right] = \mathrm{E}\left[\tilde{z}^2\right] = \mathrm{E}\left[\tilde{\beta}_{kl}^2\right] = \sigma$. Equivalently, the Rayleigh distribution is

$$f_{\tilde{z}}(z;\sigma) = \frac{2z}{\sigma} \exp\left(-z^2/\sigma\right), \tag{2.10}$$

for the magnitude of the path voltage gain.

Chapter 3

LOCALIZATION METHODOLOGY

In this chapter, we discuss our localization method in the context of sensor network. The chapter begins by stating the problem we are solving and introducing the system model we are using. System parameters and assumptions are presented and then our localization approach is described in detail. Later the formulation we have used for localization is analyzed in terms of power received by receiver which is apart from a radiating transmitter antenna and the objective functions are analytically derived. An algorithm for computing received signal power and a second for formulating the cost function is presented. The chapter ends with a brief discussion of the Python programming language which we are using for the simulation and *scipy.optimize.fmin_powell* used in optimization of cost functions.

The Problem addressed here is as follows: To localize sensor nodes in a Rayleigh fading environment and to quantify the improvement that can be obtained in agent localization accuracy as a function of the number of resolvable multipath components.

Simulation is the method used to quantify the improvements that can be obtained when we resolve miultipath. Setting up a simulation requires modeling the system under study, capturing the essential entities and their interactions, and then writing code to make the model executable. A simulator provides, first, a framework with which the system can be thoroughly studied in its current state and second, a framework that enables the user to modify any parameters of his choice that might play any part in the efficiency of the scheduler as well as controlling the way that those parameters are being used.

We set up a network model with a known number of agents located at unknown coordinates within a rectangular grid of length X and width Y having anchors at the corners,

whose locations are known. We assume that the transmitter signals reaches the receiver through multiple paths. The Figure 3.1 shows a randomly generated sensor network with $X = 200$ and $Y = 100$. Our target is to determine the exact location of the agent node using the only received signal strength information. The actual coordinates of the agent node is (x, y), which we want to localize.

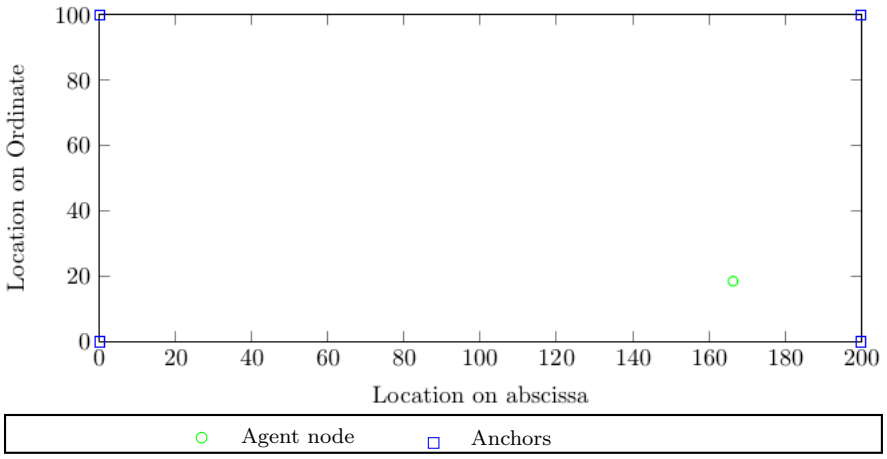

Figure 3.1. An example network with one agent node placed randomly in a rectangular area with a anchor at each corner.

Anchors transmit beacons which is in the form of radio waves and the agent nodes receive the transmitted signal. We assume that the propagation environment is same for all the waves from all the anchors and hence the propagation coefficient is α for all of the rays.

In mathematics, optimization is the process of optimizing an objective function with respect to some variables in the presence of constraints on those variables. The objective function is the function that is to be minimized. A loss function or cost function is a function that maps an event or values of one or more variables onto a real number intuitively representing some *cost* associated with the event. An optimization problem seeks to minimize a loss function.

3.1 System parameters and assumptions

We begin with a randomly located agent node within a rectangular 2-dimensional space with four corner anchors and a propagation constant for a given environment. We assume that there is exactly one propagation constant for a given environment and its range of possible values are $[2, 4]$. The value of α depends on specific propagation environment. In free space, $\alpha = 2$, and when obstructions are present, α will have a larger value [12].

Environment	**Path Loss Exponent,** α
Free space	2
Urban area cellular radio	2.7 to 3.5
Shadowed urban cellular radio	3 to 5
Obstructed in building	4 to 6
Obstructed in factories	2 to 3

Since we are transmitting the signal, we know the transmitted signal power, P_T. All the agents are equipped with transceivers and hence can measure the received signal power, P_R. It is well known that the transmitted signals travel through multiple paths. In a real scenario, at receiver, we cannot determine the actual number of multipaths and hence cannot determine whether the received signal is from LOS path or NLOS path, but we can measure the arrival power. Since no two environments are exactly the same, we consider a range of possible values from 2 to 4.

Once we measure the path power gain of each image coming through multipaths at the receiver antenna, we obtain error functions. Using those error functions, we develop cost function and minimize it to localize the agent.

3.2 Location based on received signal power

The path voltage gain β_{kl} are statistically independent positive random variables whose mean square values $\overline{\beta_{kl}}$ are monotonically decreasing functions of $\{T_l\}$ and $\{\tau_{kl}\}$. The mean

square values of the gain are given by (2.4) as

$$\begin{aligned}\overline{\beta_{kl}^2} &= \overline{\beta^2\left(T_l, \tau_{kl}\right)} \\ &= \overline{\beta^2\left(0,0\right)} \exp^{-T_l/\Gamma} \exp^{-\tau_{kl}/\gamma}, \end{aligned} \tag{3.1}$$

where $\overline{\beta^2\left(0,0\right)}$ is the average power gain of the first ray of the first cluster, and Γ and γ are power-decay constants for the clusters and rays, respectively. The power decays exponentially with cluster delay as well as excess delay within a cluster.

We first get the path power gain for each arriving multipaths from an anchor node to the agent node. We know the arrival time of all the ray images at the receiver. We calculate the total elapsed time from the first image of the first cluster to the very last image of last arriving cluster. Next, we divide the total time of ray arrival into different time bins of width 20 ns (1 chip time at 50 Mc/s) and check the bins that have at least one image and ignore the bins that have none. Now, in each bin, we sum the path power gains of all the images, and average the path power gain, and fix its arrival time as exactly mid of each bin width. Thus we get a single average path power gain for each bins. Thus from the (2.4),

$$\overline{\beta_{kl}^2} = \overline{\beta^2\left(0,0\right)} \exp^{-T_l/\Gamma} \exp^{-\tau_{kl}/\gamma}, \tag{3.2}$$

and

$$\overline{\beta^2\left(0,0\right)} = \overline{\beta_{kl}^2} \exp^{T_l/\Gamma} \exp^{\tau_{kl}/\gamma}. \tag{3.3}$$

We denote average path power gain for bin b by $\overline{\beta_b^2}$, the start time of bth bin by T_b, the mid point of bin b by τ_b and the average power gain of the first ray of the first cluster due to bth bin by $\overline{\beta_b^2\left(0,0\right)}$. Since τ_b is always measured from the begining of bin b, $\tau_b = 10$ ns. Thus,

$$\overline{\beta_b^2\left(0,0\right)} = \overline{\beta_b^2} \exp^{T_b/\Gamma} \exp^{\tau_b/\gamma}, \quad \text{with} \quad b = \left\{1, 2, 3, ..., N\right\}, \tag{3.4}$$

where N is the total number of bins. This equation (3.4) maps each $\overline{\beta_b^2}$ to $\overline{\beta_b^2(0,0)}$. Thus we attain N different samples of $\overline{\beta_b^2(0,0)}$. Now we average over all N values of $\overline{\beta_b^2(0,0)}$ and obtain a value $\overline{\beta_{b_{\text{avg}}}^2(0,0)}$ which has contribution from all the received ray images in each bins;

$$\overline{\beta_{b_{\text{avg}}}^2(0,0)} = \frac{1}{N}\sum_{b=1}^{N}\overline{\beta_b^2(0,0)}. \tag{3.5}$$

We have four anchors (A, B, C, D), located at (0,0), (0,X), (X,Y) and (0,Y), respectively and an agent P at (x, y)

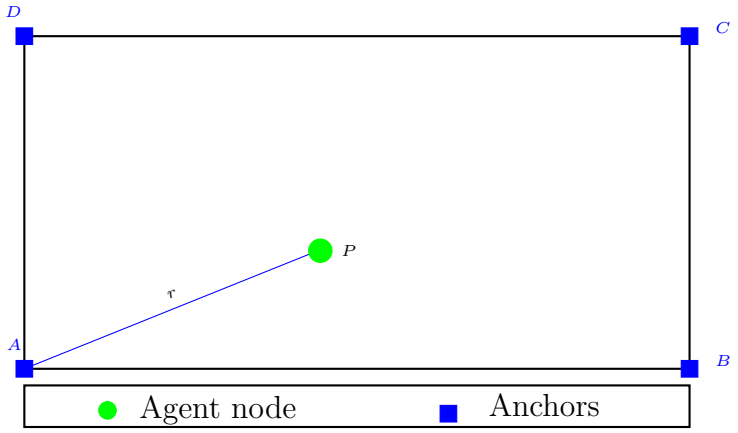

Figure 3.2. An example network with one agent node placed randomly in a rectangular area with a anchor at each corner. r is the distance between anchor A and agent P.

Using (2.7), we can write

$$\overline{\beta^2(0,0)} \approx (\gamma\lambda)^{-1}\, G(1m)\, r^{-\alpha}, \tag{3.6}$$

$$r^{-\alpha} = \overline{\beta^2(0,0)}\,(\gamma\lambda)\, G(1m)^{-1},$$

$$r = \left[\overline{\beta^2(0,0)}\,(\gamma\lambda)\, G(1m)^{-1}\right]^{-\frac{1}{\alpha}}. \tag{3.7}$$

$(\gamma\lambda)\, G(1m)^{-1}$ is a constant for a given scenario, and we can write $(\gamma\lambda)\, G(1m)^{-1} = K$. Hence

$$r = \left[K\,\overline{\beta^2(0,0)}\right]^{-\frac{1}{\alpha}}. \tag{3.8}$$

This is the required equation that we will use to calculate errors in distance measurement obtained from the channel model. Using $r = \sqrt{x^2 + y^2}$ in (3.8), we have

$$\sqrt{x^2 + y^2} = \left[K\,\overline{\beta^2\,(0,0)}\right]^{-\frac{1}{\alpha}}, \tag{3.9}$$

and

$$\epsilon\,(x, y, \alpha) = \sqrt{x^2 + y^2} - \left[K\,\overline{\beta^2\,(0,0)}\right]^{-\frac{1}{\alpha}}. \tag{3.10}$$

Corresponding to each anchor node, we have a different value of $\overline{\beta^2_{b_{\text{avg}}}\,(0,0)}$ since

$$\overline{\beta^2\,(0,0)} = \overline{\beta^2_{b_{\text{avg}}}\,(0,0)}\,.$$

Now the error function (3.10) becomes

$$\epsilon\,(x, y, \alpha) = \sqrt{x^2 + y^2} - \left[K\,\overline{\beta^2_{b_{\text{avg}}}\,(0,0)}\right]^{-\frac{1}{\alpha}}. \tag{3.11}$$

3.2.1 Cost function and minimization

Now using the above derived error function, (3.58), we generate four equations corresponding to the four anchors for our cost function.

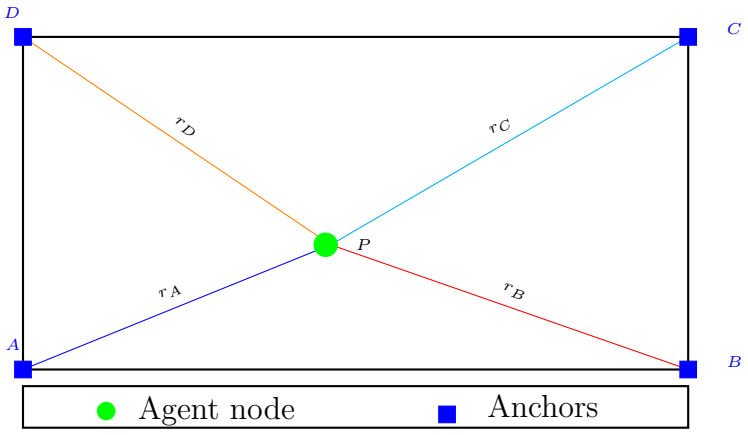

Figure 3.3. An example network with one agent node placed randomly in a rectangular area with a anchor at each corner, showing distance between agent and anchors.

Since $\overline{\beta^2_{b_{\text{avg}}}(0,0)}$ is different at the agent node for each anchor, we now add indexing to differentiate for each anchor, and we can write

$$K_j = K\,\overline{\beta^2_{bj_{\text{avg}}}(0,0)} \quad \text{for } j \in \{A,B,C,D\}\,. \tag{3.12}$$

The error function corresponding to jth anchor node is

$$\epsilon_j(x,y,\alpha) = r_j - K_j^{-\frac{1}{\alpha}} \quad \text{for j} \in \{\text{A, B, C, D}\}, \tag{3.13}$$

which results in

$$\epsilon_A(x,y,\alpha) = \sqrt{x^2+y^2} - K_A^{-\frac{1}{\alpha}}, \tag{3.14}$$

$$\epsilon_B(x,y,\alpha) = \sqrt{(X-x)^2+y^2} - K_B^{-\frac{1}{\alpha}}, \tag{3.15}$$

$$\epsilon_C(x,y,\alpha) = \sqrt{(X-x)^2+(Y-y)^2} - K_C^{-\frac{1}{\alpha}}, \tag{3.16}$$

and

$$\epsilon_D(x,y,\alpha) = \sqrt{x^2+(Y-y)^2} - K_D^{-\frac{1}{\alpha}}. \tag{3.17}$$

Now we can write our cost function $f(x,y,\alpha)$ as

$$f(x,y,\alpha) = \sum_{j\in\{A,B,C,D\}} \epsilon_j^2(x,y,\alpha)\,. \tag{3.18}$$

Once we have our cost function, we minimize the function and find the $\hat{x}$, $\hat{y}$, and $\hat{\alpha}$ that minimizes the cost function.

$$(\hat{x},\hat{y},\hat{\alpha}) = \text{argmin}\left\{f(x,y,\alpha)\right\} = \text{argmin}\left\{\sum_{j\in\{A,B,C,D\}} \epsilon_j^2(x,y,\alpha)\right\} \tag{3.19}$$

Thus obtained $\hat{x}$, $\hat{y}$ and $\hat{\alpha}$ is the estimate of agent's location x, y and propagation constant

α in our system.

In a real scenario, we have multiple number of agents and anchors as shown in Figure 3.4. The same method and procedure is extended to localize the multiple agents simultaneously.

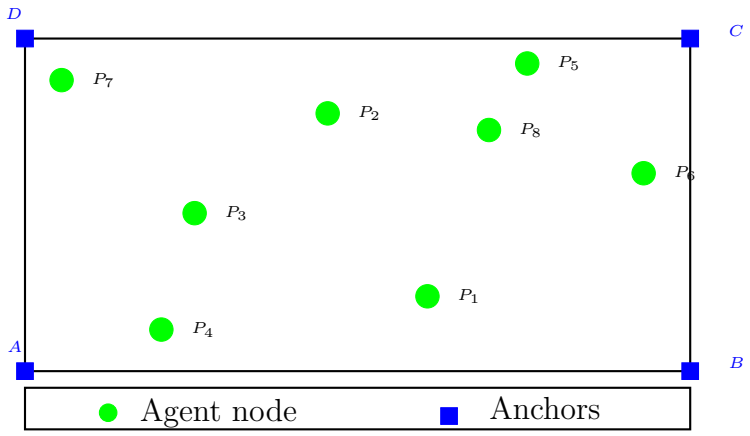

Figure 3.4. An example network with multiple agent nodes placed randomly in a rectangular area with a anchor at each corner.

For the given scenario with n_a agent nodes, error functions corresponding to the ith agent as in (3.14), (3.15), (3.16), and (3.17) can be written as

$$\begin{aligned}
\epsilon_{A_i}(x_i, y_i, \alpha) &= \sqrt{x_i^2 + y_i^2} - K_{A_i}^{-\frac{1}{\alpha}}, \\
\epsilon_{B_i}(x_i, y_i, \alpha) &= \sqrt{(X - x_i)^2 + y_i^2} - K_{B_i}^{-\frac{1}{\alpha}}, \\
\epsilon_{C_i}(x_i, y_i, \alpha) &= \sqrt{(X - x_i)^2 + (Y - y_i)^2} - K_{C_i}^{-\frac{1}{\alpha}},
\end{aligned}$$

and

$$\epsilon_{D_i}(x_i, y_i, \alpha) = \sqrt{x_i^2 + (Y - y_i)^2} - K_{D_i}^{-\frac{1}{\alpha}} \quad \text{for } P_i(x_i, y_i). \tag{3.20}$$

The cost function thus becomes

$$\begin{aligned}
f(x_1, y_1, \cdots, x_{n_a}, y_{n_a}, \alpha) &= \epsilon_{A_1}^2(x_1, y_1, \alpha) + \epsilon_{B_1}^2(x_1, y_1, \alpha) + \cdots + \epsilon_{C_{n_a}}^2(x_{n_a}, y_{n_a}, \alpha) + \epsilon_{D_{n_a}}^2(x_{n_a}, y_{n_a}, \alpha), \\
&= \sum_{j \in \{A,B,C,D\}} \sum_{i=1}^{n_a} \epsilon_{j_i}^2(x_i, y_i, \alpha).
\end{aligned} \tag{3.21}$$

Now minimizing the cost function will give us an estimate of all the agent locations and the propagation constant of the environment.

3.3 Analysis of the cost function

The choice of routines for minimizing the cost function depends upon the characteristics of the cost function itself. Since our cost function is nonlinear, it is of interest to know whether or not the cost function is convex. In this section, we check the cost function for convexity and find regions where the function is convex. Here we derive conditions of convexity for one error function only, and similar conditions can be derived for all other error functions as well.

If $f(x)$ has a second derivative in $[a, b]$, then a necessary and sufficient condition for the function to be convex on that interval is that $f''(x) \geq 0$ for all x in $[a, b]$. For an agent at (x, y), the error functions are given in (3.14), (3.15), (3.16) and (3.17). Those error functions are a function of x, y and α. In order to get a feel for the difficulty of minimizing the error function, we do a preliminary analysis of the error function to determine the region over which it is convex within the rectangular grid in which the sensors are placed.

3.3.1 Convexity with respect to x

In this section, we do a brief investigation of the characteristics of the error function at specific points and over a few regions with respect to x. We begin with the equation (3.14)

$$\begin{aligned} \epsilon_A(x, y, \alpha) &= \sqrt{x^2 + y^2} - K_A^{-\frac{1}{\alpha}}, \\ \epsilon_A^2(x, y, \alpha) &= \left[\sqrt{x^2 + y^2} - K_A^{-\frac{1}{\alpha}}\right]^2, \end{aligned} \tag{3.22}$$

and find its partial second derivative with respect to x.

$$\frac{\partial}{\partial x}\epsilon_A^2(x,y,\alpha) = \frac{\partial}{\partial x}\left[\sqrt{x^2+y^2} - K_A^{-\frac{1}{\alpha}}\right]^2$$
$$\frac{\partial}{\partial x}\epsilon_A^2(x,y,\alpha) = 2\left(\sqrt{x^2+y^2} - K_A^{-\frac{1}{\alpha}}\right)\frac{1}{2}\frac{2x}{\sqrt{x^2+y^2}}$$
$$\frac{\partial}{\partial x}\epsilon_A^2(x,y,\alpha) = 2\left(x - K_A^{-\frac{1}{\alpha}}\frac{x}{\sqrt{x^2+y^2}}\right)$$

Now using the first derivative, we find its second derivative

$$\frac{\partial^2}{\partial x^2}\epsilon_A^2(x,y,\alpha) = 2\left[1 - \left(K_A^{-\frac{1}{\alpha}}\frac{1}{\sqrt{x^2+y^2}} - x\frac{x}{(x^2+y^2)^{\frac{3}{2}}}K_A^{-\frac{1}{\alpha}}\right)\right],$$
$$\frac{\partial^2}{\partial x^2}\epsilon_A^2(x,y,\alpha) = 2 - 2\left(K_A^{-\frac{1}{\alpha}}\frac{1}{\sqrt{x^2+y^2}} - K_A^{-\frac{1}{\alpha}}\frac{x^2}{(x^2+y^2)^{\frac{3}{2}}}\right). \tag{3.23}$$

For $\frac{\partial^2}{\partial x^2}\epsilon_A^2(x,y,\alpha) \geq 0$,

$$\left(K_A^{-\frac{1}{\alpha}}\frac{1}{\sqrt{x^2+y^2}} - K_A^{-\frac{1}{\alpha}}\frac{x^2}{(x^2+y^2)^{\frac{3}{2}}}\right) \leq 1,$$
$$\frac{K_A^{-\frac{1}{\alpha}}}{\sqrt{x^2+y^2}}\left(1 - \frac{x^2}{x^2+y^2}\right) \leq 1,$$
$$K_A^{-\frac{1}{\alpha}}\frac{y^2}{(x^2+y^2)^{\frac{3}{2}}} \leq 1. \tag{3.24}$$

This is the required condition under which the error function (3.14) is convex with respect to x. Following similar steps for y as we did for x, we get a condition under which the error function is convex with y

$$K_A^{-\frac{1}{\alpha}}\frac{x^2}{(x^2+y^2)^{\frac{3}{2}}} \leq 1. \tag{3.25}$$

We next derive the regions where the condition (3.24) holds in our network layout. First, we need to know a range of K_A that we encountered in the simulation work. By evaluating the values of K_A encountered in the simulation for different agent locations and for different

values of α, following is the observed range of K_A,

$$10^{-4} \leq K_A \leq 9.9 \times 10^{-2}, \quad \text{for}\, \alpha = 2 \tag{3.26}$$

$$10^{-6} \leq K_A \leq 9.9 \times 10^{-4}, \quad \text{for}\, \alpha = 3 \tag{3.27}$$

$$10^{-8} \leq K_A \leq 9.9 \times 10^{-4}, \quad \text{for}\, \alpha = 4 \tag{3.28}$$

The range of K_A is different for different value of α used in the simulation. Using the K_A, now we derive the regions where the error function (3.14) is convex with respect to x. We first consider the point $(x, y) = (0, 0)$. On taking limit of in (3.24) as x and y go to zero, the result is $0/0$. Two applications of L'Hôpital's rule shows that the function is not convex at $x = y = 0$. Now we consider a region $x > 0$ and $y = 0$. On substituting the y in (3.24), it reduces to

$$K_A^{-\frac{1}{\alpha}} \frac{0}{x^3} \leq 1,$$

$$0 \leq 1,$$

and the function is convex for all $x > 0$ and $y = 0$. Next, we consider the region where $x = 0$ and $y > 0$. Putting those values in (3.24), reduces it to

$$K_A^{-\frac{1}{\alpha}} \frac{y^2}{y^3} \leq 1 \quad \Rightarrow \quad y \geq K_A^{-\frac{1}{\alpha}}. \tag{3.29}$$

Now using (3.26), (3.27) and (3.28), we find

$$3.7182 \leq K_A^{-\frac{1}{\alpha}} \leq 100, \quad \text{for}\, \alpha = 2,$$

$$10.03355 \leq K_A^{-\frac{1}{\alpha}} \leq 100, \quad \text{for}\, \alpha = 3,$$

and

$$5.6375 \leq K_A^{-\frac{1}{\alpha}} \leq 100, \quad \text{for}\, \alpha = 4.$$

Along the $x = 0$ axis, for $\alpha = 2$, if $K_A^{-\frac{1}{\alpha}} = 3.7182$, then the error function is convex in the region $y > 3.7182$. But if the $K_A^{-\frac{1}{\alpha}} = 100$, the error function is not convex for the given network layout. Along the $x = 0$ axis, depending upon the K_A and α used, the error function is convex in certain region or may not be convex at all. For all $y = x$ with $x > 0$ and $y > 0$, the (3.24) reduces to

$$K_A^{-\frac{1}{\alpha}} \frac{y^2}{(y^2 + y^2)^{\frac{3}{2}}} \leq 1 \quad \Rightarrow \quad y \geq \frac{K_A^{-\frac{1}{\alpha}}}{2\sqrt{2}}. \tag{3.30}$$

Now using (3.26), (3.27), and (3.28), we find

$$1.3145 \leq \frac{K_A^{-\frac{1}{\alpha}}}{2\sqrt{2}} \leq 35.355, \quad \text{for}\, \alpha = 2,$$

$$3.547 \leq \frac{K_A^{-\frac{1}{\alpha}}}{2\sqrt{2}} \leq 35.355, \quad \text{for}\, \alpha = 3,$$

and

$$1.993 \leq \frac{K_A^{-\frac{1}{\alpha}}}{2\sqrt{2}} \leq 35.355, \quad \text{for}\, \alpha = 4.$$

The error function is convex for all $x = y > 35.355$ for all α. For $\alpha = 2$, $\alpha = 3$, and $\alpha = 4$, depending on the value of K_A, the error function may be convex for other regions smaller than $x = y > 1.3145$, $x = y > 3.547$, and $x = y > 1.993$, respectively.

Next, we consider the points where $y = n\,x$ and $x > 0$, $y > 0$. We first evaluate the points where $y = 2\,x$,

$$K_A^{-\frac{1}{\alpha}} \frac{(2x)^2}{\left(x^2 + (2x)^2\right)^{\frac{3}{2}}} \leq 1 \quad \Rightarrow \quad y \geq K_A^{-\frac{1}{\alpha}} \frac{8}{5\sqrt{5}}. \tag{3.31}$$

Now using (3.26) , (3.27), and (3.28), we find

$$2.66 \leq K_A^{-\frac{1}{\alpha}} \frac{8}{5\sqrt{5}} \leq 71.55, \quad \text{for } \alpha = 2,$$

$$7.1794 \leq K_A^{-\frac{1}{\alpha}} \frac{8}{5\sqrt{5}} \leq 71.55, \quad \text{for } \alpha = 3,$$

and

$$4.0338 \leq K_A^{-\frac{1}{\alpha}} \frac{8}{5\sqrt{5}} \leq 71.55, \quad \text{for } \alpha = 4.$$

The error function is convex for all $y = 2x > 71.55$ for all α. For $\alpha = 2$, $\alpha = 3$, and $\alpha = 4$, depending on the value of K_A, the error function may be convex for other regions smaller than $x = y > 2.66$, $y = 2x > 7.1794$, and $y = 2x > 4.0338$, respectively. Using similar evaluation for $y = 3\,x$, we find

$$y \geq K_A^{-\frac{1}{\alpha}} \frac{27}{10\sqrt{10}}. \tag{3.32}$$

Now using (3.26), (3.27), and (3.28), we find

$$3.17465 \leq K_A^{-\frac{1}{\alpha}} \frac{27}{10\sqrt{10}} \leq 85.38, \quad \text{for } \alpha = 2,$$

$$8.566 \leq K_A^{-\frac{1}{\alpha}} \frac{27}{10\sqrt{10}} \leq 85.38, \quad \text{for } \alpha = 3,$$

and

$$4.81338 \leq K_A^{-\frac{1}{\alpha}} \frac{27}{10\sqrt{10}} \leq 85.38, \quad \text{for } \alpha = 4.$$

The error function is convex for all $y = 3x > 85.385$ for all α. For $\alpha = 2$, $\alpha = 3$, and $\alpha = 4$, depending on the value of K_A, the error function may be convex for other regions smaller than $y = 3x > 3.17465$, $y = 3x > 8.566$, and $y = 3x > 4.81338$, respectively. For all $y = n\,x$,

the necessary condition for (3.24) to be convex with respect to x is

$$y \geq K_A^{-\frac{1}{\alpha}} \frac{n^3}{(n^2+1)^{\frac{3}{2}}}. \tag{3.33}$$

Using (3.26), (3.27), and (3.28), the range corresponding to different values of α is

$$3.7192\frac{n^3}{(n^2+1)^{\frac{3}{2}}} \leq K_A^{-\frac{1}{\alpha}} \frac{n^3}{(n^2+1)^{\frac{3}{2}}} \leq 100\frac{n^3}{(n^2+1)^{\frac{3}{2}}}, \quad \text{for } \alpha = 2,$$

$$10.03355\frac{n^3}{(n^2+1)^{\frac{3}{2}}} \leq K_A^{-\frac{1}{\alpha}} \frac{n^3}{(n^2+1)^{\frac{3}{2}}} \leq 100\frac{n^3}{(n^2+1)^{\frac{3}{2}}}, \quad \text{for } \alpha = 3,$$

and

$$5.6375\frac{n^3}{(n^2+1)^{\frac{3}{2}}} \leq K_A^{-\frac{1}{\alpha}} \frac{n^3}{(n^2+1)^{\frac{3}{2}}} \leq 100\frac{n^3}{(n^2+1)^{\frac{3}{2}}}, \quad \text{for } \alpha = 4.$$

The error function is convex for all $y = nx > \left(100\frac{n^3}{(n^2+1)^{\frac{3}{2}}}\right)$ for all α. For $\alpha = 2$, $\alpha = 3$, and $\alpha = 4$, depending on the value of K_A, the error function may be convex for other regions smaller than $y = nx > \left(3.7192\frac{n^3}{(n^2+1)^{\frac{3}{2}}}\right)$, $y = nx > \left(10.03355\frac{n^3}{(n^2+1)^{\frac{3}{2}}}\right)$, and $y = nx > \left(5.6375\frac{n^3}{(n^2+1)^{\frac{3}{2}}}\right)$, respectively.

For all $x = n\,y$ and $x > 0$, $y > 0$, we proceed the same way we did for $y = n\,x$, and the required condition for (3.24) to be convex with respect to x at $x = n\,y$ is

$$y \geq K_A^{-\frac{1}{\alpha}} \frac{1}{(n^2+1)^{\frac{3}{2}}}. \tag{3.34}$$

Using (3.26) , (3.27) and (3.28), the range corresponding to different values of α is

$$3.7192\frac{1}{(n^2+1)^{\frac{3}{2}}} \leq K_A^{-\frac{1}{\alpha}} \frac{1}{(n^2+1)^{\frac{3}{2}}} \leq 100\frac{1}{(n^2+1)^{\frac{3}{2}}}, \quad \text{for } \alpha = 2,$$

$$10.03355\frac{1}{(n^2+1)^{\frac{3}{2}}} \leq K_A^{-\frac{1}{\alpha}} \frac{1}{(n^2+1)^{\frac{3}{2}}} \leq 100\frac{1}{(n^2+1)^{\frac{3}{2}}}, \quad \text{for } \alpha = 3,$$

and

$$5.6375\frac{1}{(n^2+1)^{\frac{3}{2}}} \leq K_A^{-\frac{1}{\alpha}}\frac{1}{(n^2+1)^{\frac{3}{2}}} \leq 100\frac{1}{(n^2+1)^{\frac{3}{2}}}, \quad \text{for}\, \alpha = 4.$$

The error function is convex for all $x = n\,y > \left(100\frac{1}{(n^2+1)^{\frac{3}{2}}}\right)$ for all α. For $\alpha = 2$, $\alpha = 3$, and $\alpha = 4$, depending on the value of K_A, the error function may be convex for other regions smaller than $x = n\,y > \left(3.7192\frac{1}{(n^2+1)^{\frac{3}{2}}}\right)$, $x = n\,y > \left(10.03355\frac{1}{(n^2+1)^{\frac{3}{2}}}\right)$, and $x = n\,y > \left(5.6375\frac{1}{(n^2+1)^{\frac{3}{2}}}\right)$, respectively.

Since the x and y are interchangeable in (3.24) and (3.25), we can obtain similar regions where the error function (3.14) is convex with respect to y, as it is with respect to x. Thus, we conclude that the error function is convex with respect to x and y in some regions and it is non-convex in some other regions, which depends on the value of α used. The region where the error function is convex with respect to x and y varies greatly with the values of α.

3.3.2 Evaluation with respect to α

In this section, we do a brief investigation of the characteristics of the error function with respect to α.

$$\frac{\partial}{\partial\alpha}\epsilon_A^2\,(x, y, \alpha) = \frac{\partial}{\partial\alpha}\left[\sqrt{x^2+y^2} - K_A^{-\frac{1}{\alpha}}\right]^2 \tag{3.35}$$

The K_A in the equation is always a positive quantity, smaller than 1. Here we define

$$K_A = \exp^{-\lambda_A} \quad \Rightarrow \quad \lambda_A = -\ln\left(K_A\right). \tag{3.36}$$

Substituting K_A in equation (3.35) gives

$$\begin{aligned}\frac{\partial}{\partial\alpha}\epsilon_A^2(x,y,\alpha) &= \frac{\partial}{\partial\alpha}\left[\sqrt{x^2+y^2}-\exp^{\frac{\lambda_A}{\alpha}}\right]^2 \\ &= 2\left(\sqrt{x^2+y^2}-\exp^{\frac{\lambda_A}{\alpha}}\right)\frac{\lambda_A}{\alpha^2}\exp^{\frac{\lambda_A}{\alpha}},\end{aligned}$$

and

$$\begin{aligned}\frac{\partial^2}{\partial\alpha^2}\epsilon_A^2(x,y,\alpha) &= \frac{\partial}{\partial\alpha}\left[2\left(\sqrt{x^2+y^2}-\exp^{\frac{\lambda_A}{\alpha}}\right)\frac{\lambda_A}{\alpha^2}\exp^{\frac{\lambda_A}{\alpha}}\right] \\ &= 2\sqrt{x^2+y^2}\,\lambda_A\frac{\partial}{\partial\alpha}\left(\frac{\exp^{\frac{\lambda_A}{\alpha}}}{\alpha^2}\right)-2\lambda_A\frac{\partial}{\partial\alpha}\left(\frac{\exp^{\frac{2\lambda_A}{\alpha}}}{\alpha^2}\right) \\ &= 2\sqrt{x^2+y^2}\,\lambda_A\left(\frac{-2}{\alpha^3}\exp^{\frac{\lambda_A}{\alpha}}-\frac{\lambda_A}{\alpha^4}\exp^{\frac{\lambda_A}{\alpha}}\right)-2\lambda_A\left(\frac{-2}{\alpha^3}\exp^{\frac{2\lambda_A}{\alpha}}-\frac{2\lambda_A}{\alpha^4}\exp^{\frac{2\lambda_A}{\alpha}}\right) \\ &= -4\sqrt{x^2+y^2}\,\frac{\lambda_A}{\alpha^3}\exp^{\frac{\lambda_A}{\alpha}}-2\sqrt{x^2+y^2}\,\frac{\lambda_A^2}{\alpha^4}\exp^{\frac{\lambda_A}{\alpha}}+\frac{4\lambda_A}{\alpha^3}\exp^{\frac{\lambda_A}{\alpha}}+\frac{4\lambda_A^2}{\alpha^4}\exp^{\frac{2\lambda_A}{\alpha}} \\ &= -2\frac{\lambda_A}{\alpha^3}\exp^{\frac{\lambda_A}{\alpha}}\left(2\sqrt{x^2+y^2}+\sqrt{x^2+y^2}\,\frac{\lambda_A}{\alpha}-2\exp^{\frac{\lambda_A}{\alpha}}-2\frac{\lambda_A}{\alpha}\exp^{\frac{\lambda_A}{\alpha}}\right).\end{aligned} \tag{3.37}$$

For $\frac{\partial^2}{\partial\alpha^2}\epsilon_A^2(x,y,\alpha)\geq 0$,

$$\begin{aligned}-2\frac{\lambda_A}{\alpha^3}\exp^{\frac{\lambda_A}{\alpha}}\left(2\sqrt{x^2+y^2}+\sqrt{x^2+y^2}\,\frac{\lambda_A}{\alpha}-2\exp^{\frac{\lambda_A}{\alpha}}-2\frac{\lambda_A}{\alpha}\exp^{\frac{\lambda_A}{\alpha}}\right)&\geq 0, \\ \lambda_A\left(2\sqrt{x^2+y^2}+\sqrt{x^2+y^2}\,\frac{\lambda_A}{\alpha}-2\exp^{\frac{\lambda_A}{\alpha}}-2\frac{\lambda_A}{\alpha}\exp^{\frac{\lambda_A}{\alpha}}\right)&\leq 0.\end{aligned} \tag{3.38}$$

In terms of K_A as specified in (3.36), we find

$$\begin{aligned}-\ln[K_A]\left(2\sqrt{x^2+y^2}+\sqrt{x^2+y^2}\,\frac{(-)\ln[K_A]}{\alpha}-2\exp^{\frac{(-)\ln[K_A]}{\alpha}}-2\frac{(-)\ln[K_A]}{\alpha}\exp^{\frac{(-)\ln[K_A]}{\alpha}}\right)&\leq 0, \\ \ln[K_A]\left(2\sqrt{x^2+y^2}-\sqrt{x^2+y^2}\,\frac{\ln[K_A]}{\alpha}-2\exp^{\frac{-\ln[K_A]}{\alpha}}+2\frac{\ln[K_A]}{\alpha}\exp^{\frac{-\ln[K_A]}{\alpha}}\right)&\geq 0.\end{aligned}$$

Here the K_A is a positive quantity and $K_A < 1$. This makes $\ln[K_A]$ always a negative quantity. Hence

$$\left(2\sqrt{x^2+y^2} - \sqrt{x^2+y^2}\,\frac{\ln[K_A]}{\alpha} - 2\exp^{\frac{-\ln[K_A]}{\alpha}} + 2\frac{\ln[K_A]}{\alpha}\exp^{\frac{-\ln[K_A]}{\alpha}}\right) \leq 0. \qquad (3.39)$$

The (3.39) can be evaluated in similar manner, as we did in previous subsection and different regions where (3.14) is convex with respect to α can be obatined. As we saw in above section, the error function (3.14) is convex with respect to x and y in some regions and it is non-convex in some other regions, which varies greatly with the values of α. We conclude that the error function is convex with respect to α in certain interesting regions, as it is with respect to x and y .

3.3.3 Summary of convexity findings

In this subsection, we are listing all the necessary conditions for the error function (3.14)-(3.17) to be convex with respect to x, y, and α, respectively. The the observed range of value of K_A, K_B, K_C, and K_D encountered in the simulation for different agent locations and for different values of α is same. Thus we can write

$$10^{-4} \leq K_j \leq 9.9 \times 10^{-2}, \quad \text{for}\, \alpha = 2,$$

$$10^{-6} \leq K_j \leq 9.9 \times 10^{-4}, \quad \text{for}\, \alpha = 3,$$

and

$$10^{-8} \leq K_j \leq 9.9 \times 10^{-4}, \quad \text{for}\, \alpha = 4,$$

where $j \in \{A, B, C, D\}$. The range of K_A is different for different value of α used in the simulation. Using the K_A, now we derive the regions where the error function (3.14) is convex with respect to x. The required condition for equation (3.14) to be convex with

respect to x, y, and α are

$$K_A^{-\frac{1}{\alpha}} \frac{y^2}{(x^2+y^2)^{\frac{3}{2}}} \leq 1, \tag{3.40}$$

$$K_A^{-\frac{1}{\alpha}} \frac{x^2}{(x^2+y^2)^{\frac{3}{2}}} \leq 1, \tag{3.41}$$

and

$$\left(2\sqrt{x^2+y^2} - \sqrt{x^2+y^2}\,\frac{\ln[K_A]}{\alpha} - 2\exp^{\frac{-\ln[K_A]}{\alpha}} + 2\frac{\ln[K_A]}{\alpha}\exp^{\frac{-\ln[K_A]}{\alpha}}\right) \leq 0. \tag{3.42}$$

Following similar steps, required conditions for equation (3.15) to be convex with x, y, and α, respectively are

$$K_B^{-\frac{1}{\alpha}} \frac{y^2}{\left((X-x)^2+y^2\right)^{\frac{3}{2}}} \leq 1, \tag{3.43}$$

$$K_B^{-\frac{1}{\alpha}} \frac{(X-x)^2}{\left((X-x)^2+y^2\right)^{\frac{3}{2}}} \leq 1, \tag{3.44}$$

and

$$\left(2\sqrt{(X-x)^2+y^2} - \sqrt{(X-x)^2+y^2}\,\frac{\ln[K_B]}{\alpha} - 2\exp^{\frac{-\ln[K_B]}{\alpha}} + 2\frac{\ln[K_B]}{\alpha}\exp^{\frac{-\ln[K_B]}{\alpha}}\right) \leq 0. \tag{3.45}$$

Required condition for equation (3.16) to be convex with x, y, and α, respectively are

$$K_C^{-\frac{1}{\alpha}} \frac{(Y-y)^2}{\left((X-x)^2+(Y-y)^2\right)^{\frac{3}{2}}} \leq 1, \tag{3.46}$$

$$K_C^{-\frac{1}{\alpha}} \frac{(X-x)^2}{\left((X-x)^2+(Y-y)^2\right)^{\frac{3}{2}}} \leq 1, \tag{3.47}$$

and

$$\left(2\sqrt{(X-x)^2+(Y-y)^2}-\sqrt{(X-x)^2+(Y-y)^2}\,\frac{\ln[K_C]}{\alpha}-2\exp^{\frac{-\ln[K_C]}{\alpha}}+2\frac{\ln[K_C]}{\alpha}\exp^{\frac{-\ln[K_C]}{\alpha}}\right)\leq 0. \tag{3.48}$$

Required condition for equation (3.17) to be convex with respect to x, y, and α, respectively are

$$K_D^{-\frac{1}{\alpha}}\frac{(Y-y)^2}{\left(x^2+(Y-y)^2\right)^{\frac{3}{2}}}\leq 1, \tag{3.49}$$

$$K_D^{-\frac{1}{\alpha}}\frac{x^2}{\left(x^2+(Y-y)^2\right)^{\frac{3}{2}}}\leq 1, \tag{3.50}$$

and

$$\left(2\sqrt{x^2+(Y-y)^2}-\sqrt{x^2+(Y-y)^2}\,\frac{\ln[K_D]}{\alpha}-2\exp^{\frac{-\ln[K_D]}{\alpha}}+2\frac{\ln[K_D]}{\alpha}\exp^{\frac{-\ln[K_D]}{\alpha}}\right)\leq 0. \tag{3.51}$$

3.3.4 Convexity of the overall cost function

We know that the sum of any two or more convex functions is a convex function. If we are able to show that the error functions are convex with respect to x, y, and α, the cost function will also be convex with x, y, and α. We derived the condition of convexity for (3.14) only and obtained the regions where it is convex with respect to x. Similar approach can be used for (3.15), (3.16) and (3.17), and all the necessary conditions and regions where they are convex can be obtained. Given that our analysis shows that the error function is not convex over the entire region of interest, a systematic way of verifying the correctness of the numerical results may be needed. It is possible that a theoretical method of establishing the corectness of the numerical results can be developed, but developing such a method is beyond the scope of this thesis. Instead, we rely on numerical investigations,

and defer development of a formal method of proving optimality to future work. The means by which we checked optimality are mentioned in Section 3.6, of this thesis.

3.4 Channel parameterization and simulation procedure

In this section, an algorithm to simulate the channel between a transmitter and receiver at a distance r, with channel model parameters Λ, λ, Γ and γ is presented. In what follows, n_c is the number of arriving clusters and n_{rc} is the number of rays in cth cluster with $l \in \{0, 1, 2, \cdots, n_c - 1\}$ and $k \in \{0, 1, 2, \cdots, n_{rc} - 1\}$. $n_A = 4$ is the number of anchor nodes, and n_a is the number of agent nodes in the network. j is the anchor index and i is the agent index, where $j \in \{A, B, C, D\}$ and $i \in \{1, 2, \cdots, n_a\}$. r_{ij} is distance between ith agent node and jth anchor. This is followed by our algorithm to localize agent nodes for a single agent case.

Algorithm 1: Algorithm for computing received signal power

Choose Λ, λ, Γ and γ
for all n_A **do**
  Set n_c and n_{rc}
  for all n_a **do**
    Compute r_{ij}
  end for
  for all n_c **do**
    Generate T_1 , $T_2, \cdots, T_{n_c-1}$ using (2.2) with $T_0 = 0$
    for all n_{rc} **do**
      Generate τ_{kl} using (2.3) with $\tau_{0l} = 0$
      if $k = 0$ and $l = 0$ **then**
        Compute $\overline{\beta^2(0,0)}$ using (3.6)
      end if
      Generate β_{kl} using (2.10), with $\overline{\beta_{kl}^2}$ given by (2.4).
    end for
  end for
end for

Algorithm 2: Algorithm to formulate the cost function

Generate network with n_A and n_a .
Choose Λ, λ, Γ and γ .
for all n_A **do**
  Set n_c and n_{rc}.
  for all n_a **do**
    Compute r_{ij}.
  end for
  for all n_c **do**
    Generate $T_1, T_2, \cdots, T_{n_c-1}$ using (2.2) with $T_0 = 0$.
    for all n_{rc} **do**
      Generate $\{\tau_{kl}\}$ using (2.3) with $\tau_{0l} = 0$.
      if $k = 0$ and $l = 0$ **then**
        Compute $\overline{\beta^2(0,0)}$ using (3.6).
      end if
      Generate β_{kl} using (2.9), with $\overline{\beta_{kl}^2}$ given by (2.4).
    end for
  end for
  Compute $\tau_d = \tau_{(n_{rc}-1)(n_c-1)} - \tau_{00}$.
  Divide the τ_d into bins of width 20 ns. This gives N non-empty bins.
  for all N **do**
    Find average path power gain $\overline{\beta_b^2}$ in each bins.
    Position the $\overline{\beta_b^2}$ at mid of the corresponding bin.
    Reflect the $\overline{\beta_b^2}$ to $\overline{\beta_b^2(0,0)}$ using (3.4).
  end for
  Obtain $\overline{\beta_{b_{\text{avg}}}^2(0,0)}$ of the N obtained $\overline{\beta_b^2(0,0)}$.
  Compute P_R from the $\overline{\beta_b^2(0,0)}$.
  Create error function using the P_R and distance formula.
end for
Create cost function from error functions.
Minimize the cost function and obtain $\hat{x}$, $\hat{y}$ and $\hat{\alpha}$.

3.5 A second look at mapping multipath resolution

The transmitted signal power($\mathrm{P_T}$) is

$$\mathrm{P_T} = \lim_{\mathrm{T}\to\infty} \frac{1}{\mathrm{T}} \int_0^{\mathrm{T}} s^2(t)\, dt, \tag{3.52}$$

where $s(t)$ is the transmitted signal. Power received by a receiving antenna which is separated from a radiating transmitter antenna by a distance d, is given as

$$P_R = \frac{P_T\, G_T\, G_R\, \lambda^2}{(4\pi)^2\, d^\alpha\, L}, \tag{3.53}$$

where P_T is the transmitted power, P_R is the received power, G_T is the transmitter antenna gain, G_R is the receiver antenna gain, λ is wavelength in meters, α is the propagation coefficient of the environment and d is transmitter-receiver separation in meters.

$$P_R = \frac{P_T\, K}{d^\alpha}, \qquad \text{(deterministic quantity)} \tag{3.54}$$

where $k = \frac{G_T\, G_R\, \lambda^2}{(4\pi)^2\, L}$.

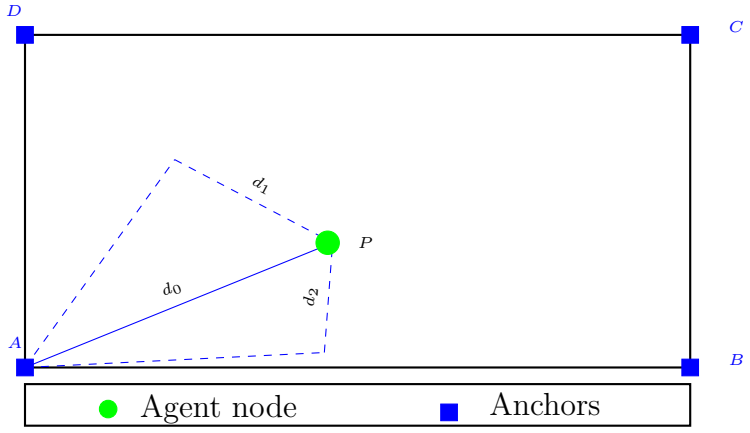

Figure 3.5. An example network with one agent node placed randomly in a rectangular area with a anchor at each corner, showing multipaths between the agent and anchor A.

In reference to the above figure, we can write

$$P_{R_0} = \frac{P_T\, K}{d_0^{\alpha}}, \quad P_{R_1} = \frac{P_T\, K}{d_1^{\alpha}}, \quad \text{and} \quad P_{R_2} = \frac{P_T\, K}{d_2^{\alpha}}.$$

Here d_0 is the shortest, ie. LOS path, and P_{R_0} is the received power through the LOS path, d_1 is the first reflected path (NLOS) and P_{R_1} is the received power corresponding to the first NLOS path and d_2 is the second reflected path (NLOS) between the anchor node A and the agent node P, and P_{R_2} is the received power corresponding to the second reflected path.

Now, we express the P_{R_0} in terms of P_{R_1},

$$\begin{aligned} P_{R_0} &= \frac{P_T\, K}{d_0^{\alpha}} \\ &= \frac{P_T\, K}{d_1^{\alpha}} \left(\frac{d_1}{d_0}\right)^{\alpha} \\ &= \frac{P_T\, K}{d_1^{\alpha}} \left(1 + \frac{\Delta d_1}{d_0}\right)^{\alpha} \\ P_{R_0} &\approx P_{R_1}\, \exp^{\frac{\Delta d_1}{d_0}\,\alpha}. \end{aligned} \tag{3.55}$$

Similarly,

$$P_{R_0} \approx P_{R_2}\, \exp^{\frac{\Delta d_2}{d_0}\,\alpha}.$$

Hence we can write

$$P_{R_0} \approx P_{R_i}\, \exp^{\frac{\Delta d_i}{d_0}\,\alpha},$$

where i is the multipath index.

Next we consider Rayleigh fading,

$$\tilde{P}_{R_i} = \overline{P}_{R_i}\, \tilde{x}_i,$$

where $\tilde{x}_i \sim \mathbb{E}\,[1]$, and $\overline{P}_{R_i}$ is the real average power received over a path of length d_i. At

distance d_0, this would be larger. We have samples of P_{R_0} given by

$$P_{R_i} = \overline{P}_{R_0} \exp^{-\frac{\Delta d_i}{d_0}\alpha} \tilde{x}_i,$$

so that

$$P_{R_i} \exp^{\frac{\Delta d_i}{d_0}\alpha} = \overline{P}_{R_0} \tilde{x}_i.$$

Let n be total number of multipath components from anchor A to the agent node. Summing over all the multipaths, we find

$$\frac{1}{n}\sum_{i=1}^{n} \tilde{P}_{R_i} \exp^{\frac{\Delta d_i}{d_0}\alpha} = \frac{1}{n}\overline{P}_{R_0} \sum_{i=1}^{n} x_i,$$

so that

$$\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n} \tilde{P}_{R_i} \exp^{\frac{\Delta d_i}{d_0}\alpha}\right] = \mathbb{E}\left[\frac{1}{n}\overline{P}_{R_0} \sum_{i=1}^{n} x_i\right] = \overline{P}_{R_0}. \tag{3.56}$$

Because $\mathbb{E}[x_i] = 1$ and since

$$P_{R_0} = \frac{P_T K}{d_0^{\alpha}},$$

we have

$$d_0 = \left(\frac{P_T K}{P_{R_0}}\right)^{\frac{1}{\alpha}} = \left(\frac{P_T K}{\frac{1}{n}\sum_{i=1}^{n} \tilde{P}_{R_i} \exp^{\frac{\Delta d_i}{d_0}\alpha}}\right)^{\frac{1}{\alpha}}. \tag{3.57}$$

We know $d_0 = \sqrt{x^2 + y^2}$. Hence

$$\sqrt{x^2 + y^2} = \left(\frac{P_T K}{\frac{1}{n}\sum_{i=1}^{n} \tilde{P}_{R_i} \exp^{\frac{\Delta d_i}{d_0}\alpha}}\right)^{\frac{1}{\alpha}},$$

and

$$\epsilon\left(x,y,\alpha\right)=\sqrt{x^2+y^2}-\left(\frac{P_T\,K}{\frac{1}{n}\sum_{i=1}^{n}\tilde{P}_{R_i}\,\exp^{\frac{\Delta d_i}{\sqrt{x^2+y^2}}\alpha}}\right)^{\frac{1}{\alpha}}, \tag{3.58}$$

where $\tilde{P}_{R_i}$ is the measured power of each ray images that we get from SV channel model. Thus we could define an error function at each corner in this way and find the best x, y, α from a sum of square of errors. $\epsilon\left(x,y,\alpha\right)$ is the error that comes up due to the difference in actual distance and the approximate distance obtained from the channel model.

3.5.1 Cost function

In this section, we formulate the cost function by using the error functions.

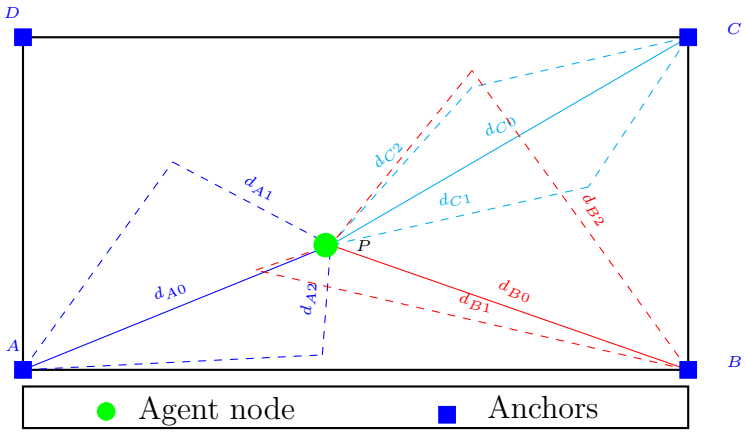

Figure 3.6. An example network with one agent node placed randomly in a rectangular area with a anchor at each corner, showing multipaths between the agent and anchor A, B and C.

Error corresponding to anchor A

$$\epsilon_A\left(x,y,\alpha\right)=\sqrt{x^2+y^2}-\left(\frac{P_T\,K}{\frac{1}{n_A}\sum_{i=1}^{n_A}\tilde{P}_{R_{Ai}}\,\exp^{\frac{\Delta d_{Ai}}{\sqrt{x^2+y^2}}\alpha}}\right)^{\frac{1}{\alpha}}, \tag{3.59}$$

where n_A is the number of multipath images from anchor A, $\tilde{P}_{R_{Ai}}$ is the received power of the ith image at the agent node due to anchor A and d_{Ai} is the path length of the ith image.

Similarly,

$$\epsilon_B(x,y,\alpha) = \sqrt{(X-x)^2+y^2} - \left(\frac{P_T K}{\frac{1}{n_B}\sum_{i=1}^{n_B} \tilde{P}_{R_{Bi}} \exp^{\frac{\Delta d_{Bi}}{\sqrt{(X-x)^2+y^2}}\alpha}} \right)^{\frac{1}{\alpha}}, \tag{3.60}$$

where n_B is the number of multipath images from anchor B, $\tilde{P}_{R_{Bi}}$ is the received power of the ith image at the agent node due to anchor B and d_{Bi} is the path length of the ith image. Also,

$$\epsilon_C(x,y,\alpha) = \sqrt{(X-x)^2+(Y-y)^2} - \left(\frac{P_T K}{\frac{1}{n_C}\sum_{i=1}^{n_C} \tilde{P}_{R_{Ci}} \exp^{\frac{\Delta d_{Ci}}{\sqrt{(X-x)^2+(Y-y)^2}}\alpha}} \right)^{\frac{1}{\alpha}}, \tag{3.61}$$

where n_C is the number of multipath images from anchor C, $\tilde{P}_{R_{Ci}}$ is the received power of the ith image at the agent node due to anchor C and d_{Ci} is the path length of the ith image. Also,

$$\epsilon_D(x,y,\alpha) = \sqrt{x^2+(Y-y)^2} - \left(\frac{P_T K}{\frac{1}{n_D}\sum_{i=1}^{n_D} \tilde{P}_{R_{Di}} \exp^{\frac{\Delta d_{Di}}{\sqrt{x^2+(Y-y)^2}}\alpha}} \right)^{\frac{1}{\alpha}}, \tag{3.62}$$

where n_D is the number of multipath images from anchor D, $\tilde{P}_{R_{Di}}$ is the received power of the ith image at the agent node due to anchor D and d_{Di} is the path length of the ith image. Now we can write our cost function $f(x,y,\alpha)$ as

$$f(x,y,\alpha) = \epsilon_A^2(x,y,\alpha) + \epsilon_B^2(x,y,\alpha) + \epsilon_C^2(x,y,\alpha) + \epsilon_D^2(x,y,\alpha),$$

and

$$(\hat{x},\hat{y},\hat{\alpha}) = \operatorname{argmin}\{f(x,y,\alpha)\} = \operatorname{argmin}\left\{ \sum_j \epsilon_j^2(x,y,\alpha) \right\}, \tag{3.63}$$

where $j \in \{A, B, C, D\}$. Once we have our cost function, we minimize the obtained cost function and get the required $\hat{x}$, $\hat{y}$, and $\hat{\alpha}$ value that minimizes the cost function. Thus obtained $\hat{x}$, $\hat{y}$, and $\hat{\alpha}$ is the estimate of the agent location and propagation constant, respectively.

3.6 Optimization method and programming language

We are using Python for our simulation purpose. Python is an interpreted, object-oriented, high-level programming language. Python supports modules and packages, which encourages program modularity and code reuse. The fundamental package for scientific computing in Python is Numerical Python (NumPy), and Scientific Python (SciPy) is also a package which provides a lot of scientific routines that runs on top of NumPy. Since our earlier examination of the cost function shows that it is not convex, it is necessary to choose an optimization routine that does not depend on convexity. We chose a sub-package of Scipy, *optimize* for our optimization purposes and within optimize routines that do not depend upon the cost function being convex.

We chose *scipy.optimize.fmin_powell* for minimization of cost functions. This method minimizes a function of N variables using modified Powell's method. Powell's conjugate direction method, commonly known as Powell's method, is an efficient method for finding the minimum of a function of several variables without calculating derivatives [17]. This method is used for unconstrained minimization. To get more confidence that our cost functions were minimized properly, we thoroughly compared the results obtained from Powell's method with the results of two other methods; namely *scipy.optimize.fmin* and *scipy.optimize.fmin_l_bfgs_b*, separately.

The scipy.optimize.fmin uses a Nelder-Mead simplex algorithm to find the minimum of function of one or more variables. This algorithm has a long history of successful use in applications. But it is slower than an algorithm that uses first or second derivative information. In practice it can have poor performance in high-dimensional problems and is

not robust to minimizing complicated functions. Additionally, there currently is no complete theory describing when the algorithm will successfully converge to the minimum, or how fast it will if it does [17]. This method is also used for unconstrained minimization.

The scipy.optimize.fmin_l_bfgs_b minimizes a function using the L-BFGS-B algorithm. The L-BFGS-B algorithm extends L-BFGS (Limited-memory Broyden-Fletcher-Goldfarb-Shanno) to handle simple bound constraints on variables; that is, constraints of the form $l_i \leq z_i \leq u_i$ where l_i and u_i are per-variable constant lower and upper bounds, respectively (for each z_i, either or both bounds may be omitted). The method works by identifying fixed and free variables at every step (using a simple gradient method), and then using the L-BFGS method on the free variables only to get higher accuracy, and then repeating the process [17].

We chose Powell's method over the other two methods, as convergence time taken by the Powell's method was less, and this method converged even when the scipy.optimize.fmin failed to converge. Powell's method and L-BFGS-B include gradient-based search methods, both the methods converge quickly, but the minimum values obtained using Powell's method are better than the ones from L-BFGS-B method [18].

We also verified those routines by doing numerical checks over the entire grid. We first divided the network grid into subgrids of dimension 1×1. This gave us $201 \times 101 = 20301$ coordinate pair over the entire grid. We next evaluated the cost function throughout the entire 20301 points, and recorded the minimum value of cost function and corresponding coordinate pair that was giving the minimum. Thus obtained coordinate pair were equal to the results obtained by minimizing the same cost function using the Python minimizing routines.

Chapter 4

NUMERICAL RESULTS

In the previous chapter, our localization strategy was developed, the cost functions were analytically derived and were evaluated to see if they are convex. The parameters and settings used to carry the simulations, the channel model, and our localization technique was presented. In this chapter, the procedure and approach used in the simulations are described. The simulation results and related discussions conclude the chapter.

We investigated performance of our method for two cases. In first case, we evaluate the technique with one agent placed at a random position. For this, we investigated a large sample of received signals at the same location and saw how well that worked. The variables are the propagation constant, the location, and the number of multipaths resolved. We then plotted localization error as a function of number of multipath resolutions as a function of some anchor nodes.

In the other case, we have multiple randomly generated user agents and we resolve all of their locations using a single measurement set. This time the parameters are the propagation constant, number of user agents, and number of multipaths resolved. We obtained a sample by fixing the propagation constant, the number of user agents, and the number of mutipath resolutions, but placing the user agents at random locations. In real world cases, we cannot measure time any better than the resolution of the chips, and we have modeled the images as individual multipath components instead of clusters.

4.1 Single agent case

We begin with a randomly generated agent node, four anchor nodes and a propagation constant (α) for the given environment. Setting the number of resolvable multipaths, for the agent location, we run the simulation 100 times and measure 100 different samples of recieved signals and in turn, 100 different sets of $\hat{x}_{\text{raw}}$, $\hat{y}_{\text{raw}}$ and $\hat{\alpha}_{\text{raw}}$ that minimizes the cost functions. We term those result sets as 'raw data'. In those 100 raw data sets, some of the locations obtained are very close to the actual agent location but some of them are distinctly different from the original agent location. However, the obtained minimized alpha values $\hat{\alpha}_{\text{raw}}$ are closer to each other and usually fall in a range of values lower than the original α, ie. $\hat{\alpha}_{\text{raw}} < \alpha$.

We now use an iterative approach to improve the raw data sets and obtain more accurate revised data sets. Since finding the exact agent location is our primary objective, we use an arbitrary method of averaging all the values of $\hat{\alpha}_{\text{raw}}$ and set the value of obtained average as $\hat{\alpha}$.

Replacing the original α by $\hat{\alpha}$ and re-minimizing all the 100 cost functions to obtain an estimate of agent location only, we get a new revised data set of 100 values of $\hat{x}$ and $\hat{y}$. The new sets of $\hat{x}$, $\hat{y}$ and $\hat{\alpha}$ are termed as 'revised data' and those data sets are, in general, closer to the original agent location than raw data sets.

Now we check accuracy of our localization procedure by investigating the position error (e_p) for agent locations with number of resolvable multipaths, the agent location, and propagation constant as parameters.

$$e_p = \sqrt{(x - \hat{x})^2 + (y - \hat{y})^2}, \tag{4.1}$$

which is expressed in meters. By running our simulation for different agent locations and for different values of α, we have found that if we are able to choose $\hat{\alpha}$ accurately by some other methods, the localization error is almost zero.

4.1.1 Position error as a function of number of resolvable multipaths

By running our simulation 100 times for an agent location corresponding to each number of resolvable multipaths from 1 to 7, with fixed propagation constant ($\alpha = 2$), we obtained the minimum position error ($e_{p_{\min}}$), the maximum position error ($e_{p_{\max}}$) and the average position error ($\mathbb{E}[e_p]$). The number of clusters varies from 1 to 4 for a 2.4 GHz bandwidth in indoor environments and an average of 5 clusters in office environments, however up to 14 clusters are found in outdoor environments [16]. We set the maximum number of resolvable multipaths to 7 and the minimum number of resolvable multipaths to 1. We have plotted the errors obtained from raw data and from revised data as a function of number of resolvable multipaths for six different agent locations. The agent locations that we have used are (P_1, P_2, P_3, P_4, P_5, P_6), at $(42.472, 23.907)$, $(11.529, 76.381)$, $(32.151, 96.089)$, $(177.340, 46.573)$, $(185.115, 43.491)$ and $(167.731, 94.984)$, respectively.

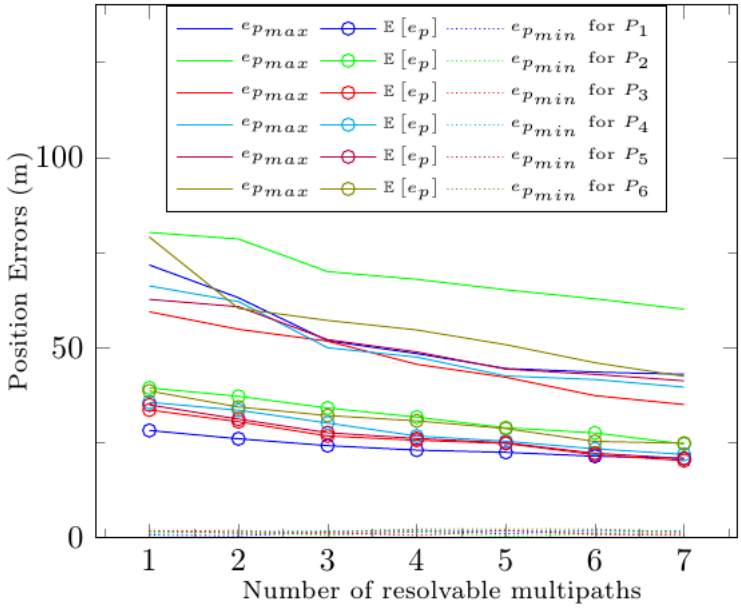

Figure 4.1. Observed minimum, maximum, and average position errors of raw data for a system with propagation constant 2 as a function of resolvable multipaths for six arbitrarily chosen locations in a 200×100 meter grid.

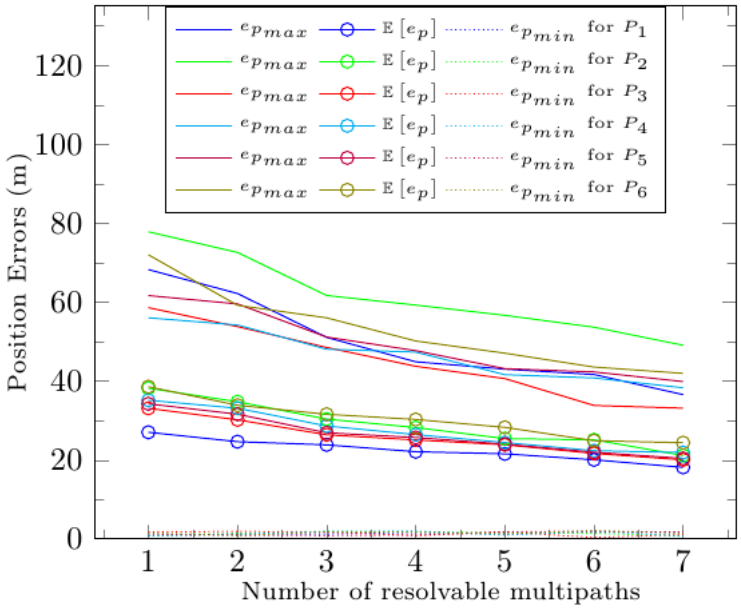

Figure 4.2. Observed minimum, maximum, and average position errors of revised data for a system with propagation constant 2, as a function of resolvable multipaths for six arbitrarily chosen locations in a 200×100 meter grid.

From the Figure 4.1 and Figure 4.2, we can clearly see that as the number of resolvable multipaths increases, the position error gradually decreases. The larger the number of resolvable multipaths, the higher is the localization accuracy. We can also see that localization errors in the revised data are smaller then the errors for the raw data. The revision of propagation constant has definitely contributed to better localization, however the improvement is very small, on the order of a few meters. However, if we are able to develop an efficient method to choose the exact value of $\hat{\alpha}$ from $\hat{\alpha}_{\text{raw}}$, we would see great improvement in localization accuracy with revised data sets.

We next want to examine how the values of $\hat{\alpha}_{\text{raw}}$ are distributed, and we have plotted the cumulative distribution $\hat{\alpha}_{\text{raw}}$ as a function of number of resolvable multipaths. From Figure 4.3, we can see that for every agent location, for the same propagation constant and for the same number of resolvable multipaths, $\hat{\alpha}_{\text{raw}}$ values seem to be distributed over a narrow range. For the number of resolvable multipaths equals 1, we can see distribution of $\hat{\alpha}_{\text{raw}}$ in one narrow range and for the number of resolvable multipaths equals 4, there is a

different narrow range of distribution of $\hat{\alpha}_{\text{raw}}$, irrespective of the agent location.

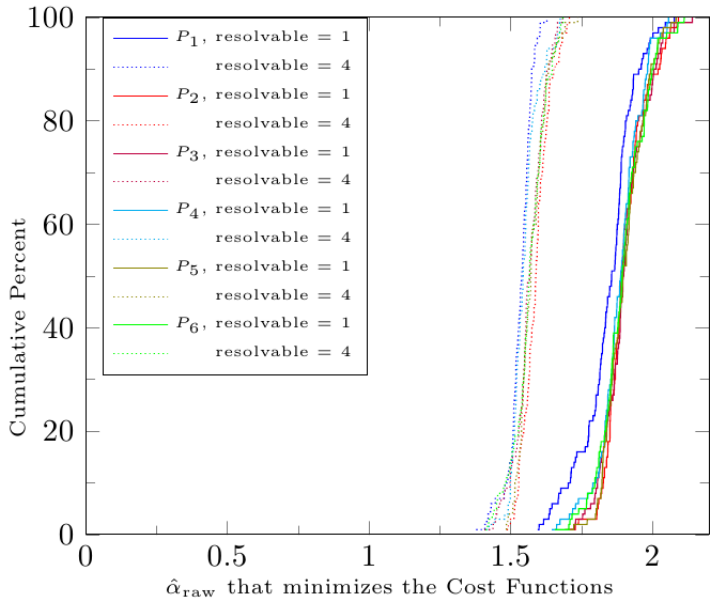

Figure 4.3. Observed cumulative distribution of $\hat{\alpha}_{\text{raw}}$ for a system with propagation constant 2, resolvable multipaths as 1 and 4 for six arbitrarily chosen locations in a 200×100 meter grid.

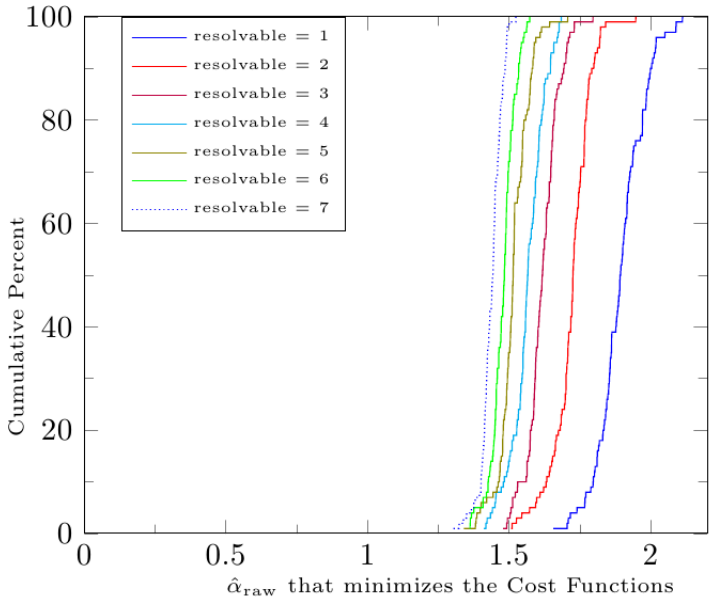

Figure 4.4. Observed cumulative distribution of $\hat{\alpha}_{\text{raw}}$ for a system with propagation constant 2, as a function of resolvable multipaths for an arbitrarily chosen location P_1 in a 200×100 meter grid.

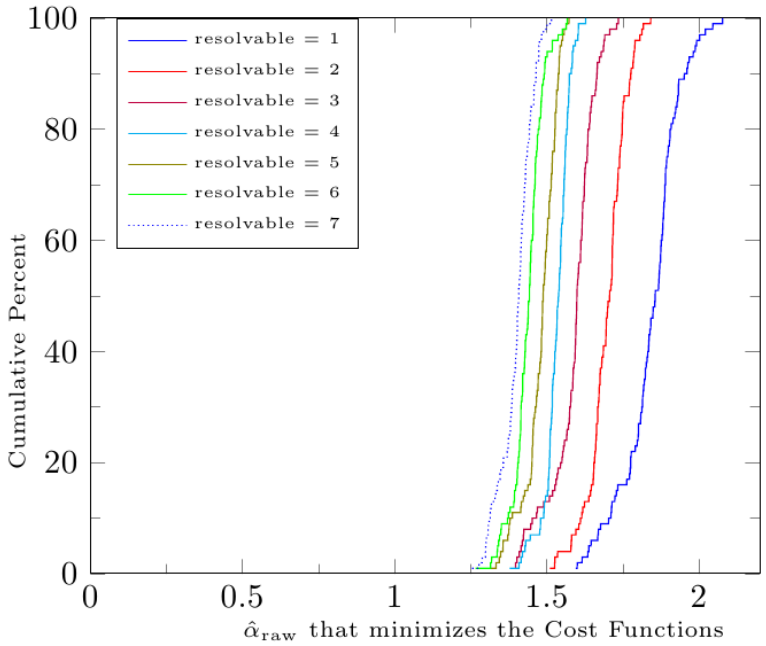

Figure 4.5. Observed cumulative distribution of $\hat{\alpha}_{\text{raw}}$ for a system with propagation constant 2, as a function of resolvable multipaths for an arbitrarily chosen location P_6 in a 200×100 meter grid.

Figure 4.4 and Figure 4.5 show the cumulative distribution of $\hat{\alpha}_{\text{raw}}$ obtained by varying the number of resolvable multipaths from 1 to 7 for two different agent locations. The values of $\hat{\alpha}_{\text{raw}}$ obtained are gradually decreasing with the increase in number of resolvable multipaths.

We also studied the nature of localization errors obtained as a function of propagation constant of the environment with the number of resolvable multipaths fixed. We plotted the localization errors as a function of α of the environment for two different systems with an arbitrarily generated agent locations. For an agent location, with number of resolvable multipaths as 6 and running our simulation for 100 times as before, we have obtained minimum, average and maximum position errors of revised data as a function of α.

From Figure 4.6 and Figure 4.7, we can see that position error gradually decreases as the value of propagation constant increases from 2 to 4. The larger the propagation constant of the environment, the smaller is the localization errors obtained.

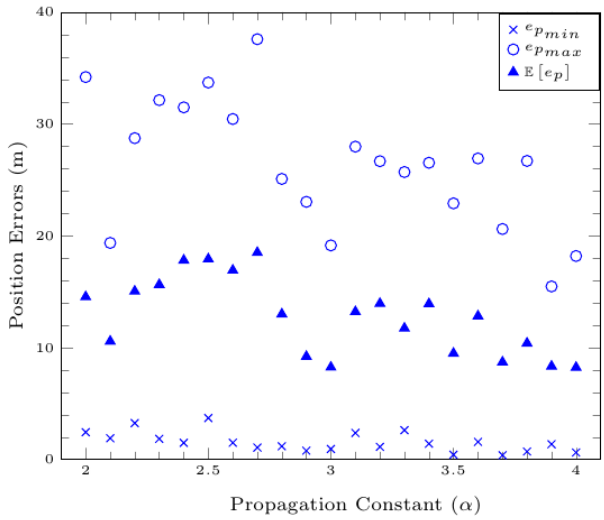

Figure 4.6. Observed minimum, maximum, and average position errors for a system having 6 resolvable multipaths as a function of propagation constant for an arbitrarily chosen agent P_7, located at $(90.768418, 27.294762)$ in a 200×100 meter grid.

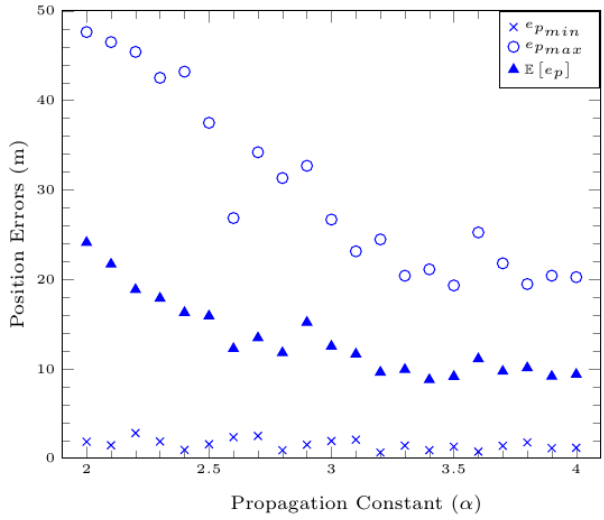

Figure 4.7. Observed minimum, maximum, and average position errors for a system having 6 resolvable multipaths as a function of propagation constant for an arbitrarily chosen agent P_8, located at $(112.54376, 97.430651)$ in a 200×100 meter grid.

4.2 Multiple agents case

We generate a network layout with multiple agents, create a cost function using measured signal information of all the agents and minimize it to obtain location of all the agents and propagation constant. The number of resolvable multipaths is a parameter. We take a single measurement corresponding to each resolvable multipath for each agent in the system.

Since we are localizing n_a arbitrarily chosen agent nodes in a system, agents located at a larger distance have larger position errors and the agents at smaller distance have smaller localization errors, it seems that plotting the actual e_p would not be sufficient. Instead of plotting location errors of each nodes, we calculate a root mean square error (e_{rms}) for each system using location errors corresponding to each agent, and plot the root mean square error of different system as a function of number of resolvable multipaths. e_{p_i} is the location error corresponding to ith agent in a network, where $i \in \{1, 2, \cdots, n_a\}$. Then the e_{rms} of the system (network) is

$$e_{\text{rms}} = \sqrt{\frac{1}{n_a} \sum_{i=1}^{n_a} e_{p_i}^2} \tag{4.2}$$

We generated 10 different networks each with 10 arbitrarily generated agent nodes, computed the e_{rms} of each system using (4.2) and plotted them as a function of number of resolvable multipaths. As we can see in Figure 4.8, for all the systems, e_{rms} decreases with the increase in number of resolvable multipaths.

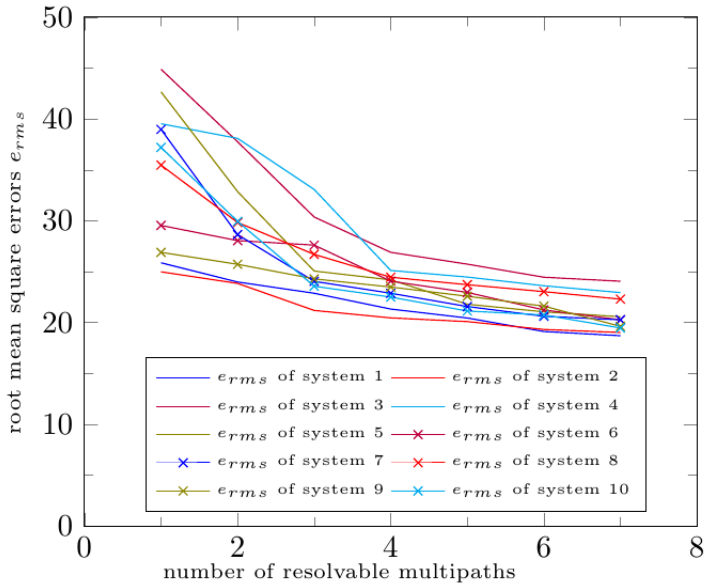

Figure 4.8. Observed e_{rms} for 10 different systems having propagation constant as 2, as a function of resolvable multipaths, for 10 arbitrarily chosen agent locations in a 200×100 meter grid, all agents localized at once.

For each system, we localized each agent node in the network separately using individual

cost functions as done in the single agent case, and compared the localization accuracy measured in terms of e_{rms} of a system obtained in group localization with e_{rms} obtained for the system when localizing each agents individually. The observed e_{rms} for individual localizations are shown in Figure 4.9.

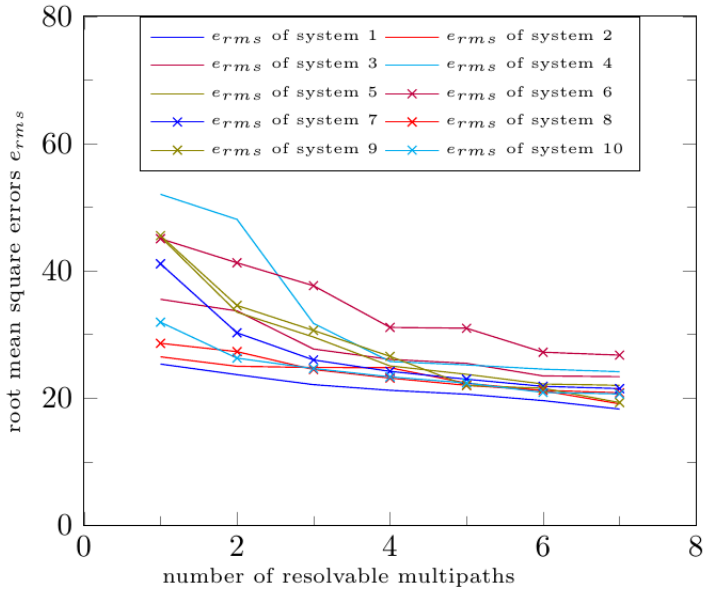

Figure 4.9. Observed e_{rms} for 10 different systems having propagation constant as 2, as a function of resolvable multipaths, for 10 arbitrarily chosen agent locations in a 200×100 meter grid, all agents localized individually.

Performance of group localization and individual localization in terms of e_{rms} of each system is found to be similar. For some networks, group localization yields smaller root mean square errors and for other networks individual agent localization performs better. Based on this, we can say that performance of our localization technique is similar for single as well as multiple agent localization.

We worked with 100 different systems, 50 of those with 10 arbitrarily chosen agent locations and another 50 with 20 arbitrarily chosen agent locations. We then computed root mean square errors (e_{rms}) of each system by varying the number of resolvable multipaths from 1 to 7 for each of the system. From the computed e_{rms}, we obtained maximum ($e_{\text{rms}_{\text{max}}}$), minimum ($e_{\text{rms}_{\text{min}}}$), and average ($\mathbb{E}\left[e_{\text{rms}}\right]$) root mean square errors throughout all the systems

and plotted them as a function of number of resolvable multipaths in Figure 4.10. As we can see, the $e_{\text{rms}_{\text{max}}}$, $\mathbb{E}[e_{\text{rms}}]$ and $e_{\text{rms}_{\text{min}}}$ gradually decreases with increase in the number of resolvable multipaths.

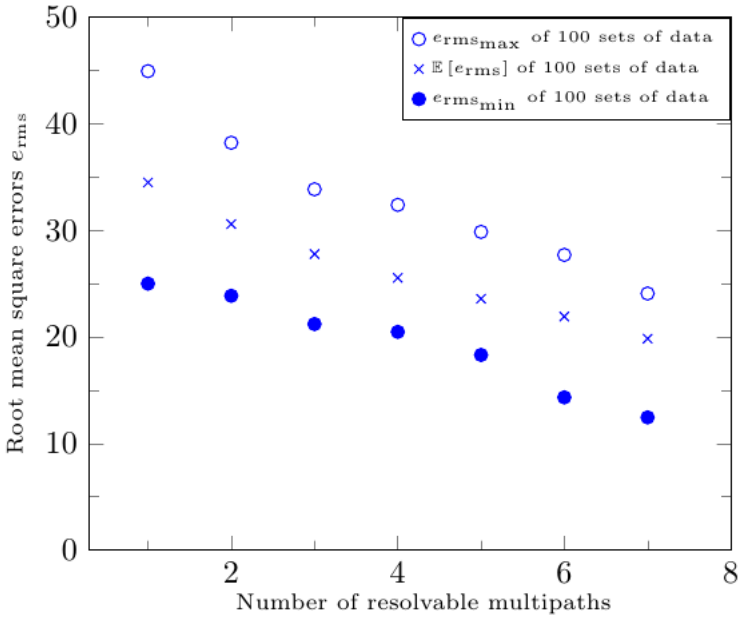

Figure 4.10. Observed $e_{\text{rms}_{\text{max}}}$, $\mathbb{E}[e_{\text{rms}}]$ and $e_{\text{rms}_{\text{min}}}$ from 100 different systems with propagation constant 2, as a function of resolvable multipaths for 10 and 20 arbitrarily chosen location in a 200×100 meter grid.

We also plotted the observed cumulative distribution of e_{rms} of same 100 networks as a function of number of resolvable multipaths in Figure 4.11. We can see that the larger the number of resolvable multipaths, the smaller is the root mean square errors for all the systems.

From all the plots, we see that the increase in the number of resolvable multipaths improves the localization accuracy in our systems, which was expected.

In Figure 4.12, we have plotted cumulative distribution of $\hat{\alpha}$ obtained as a function of number of resolvable multipaths for those 100 systems with group localization. Similar to the distribution of $\hat{\alpha}_{\text{raw}}$ observed in single agent localization case, the observed $\hat{\alpha}$ are distributed in a small range of values and with the increase in number of resolvable multipaths, the $\hat{\alpha}$ is decreasing.

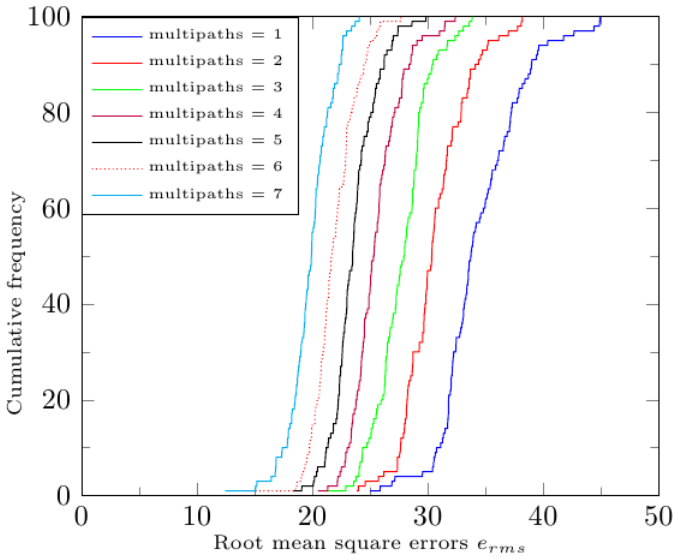

Figure 4.11. Observed cumulative distribution of e_{rms} of 100 different systems with propagation constant 2, as a function of resolvable multipaths for 10 and 20 arbitrarily chosen location in a 200×100 meter grid.

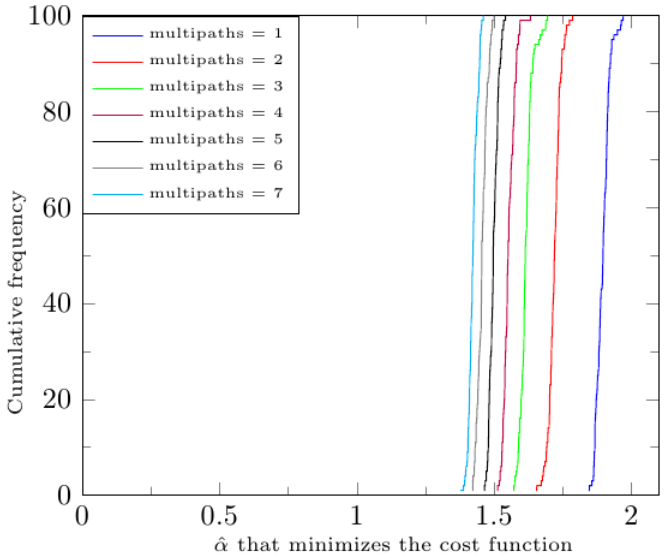

Figure 4.12. Observed cumulative distribution of $\hat{\alpha}$, that minimizes the cost functions of 100 different systems with propagation constant 2, as a function of resolvable multipaths for 10 and 20 arbitrarily chosen location in a 200×100 meter grid.

We also studied the nature of individual location errors corresponding to each agents observed with those 100 systems. We studied to see if there is any distribution pattern for the location errors of agents when plotted as a function of actual distance from one of the

anchor A. The localization errors of 50 systems with 10 agent nodes are plotted in a scatter plot and with 20 agent nodes are plotted in another scatter plot.

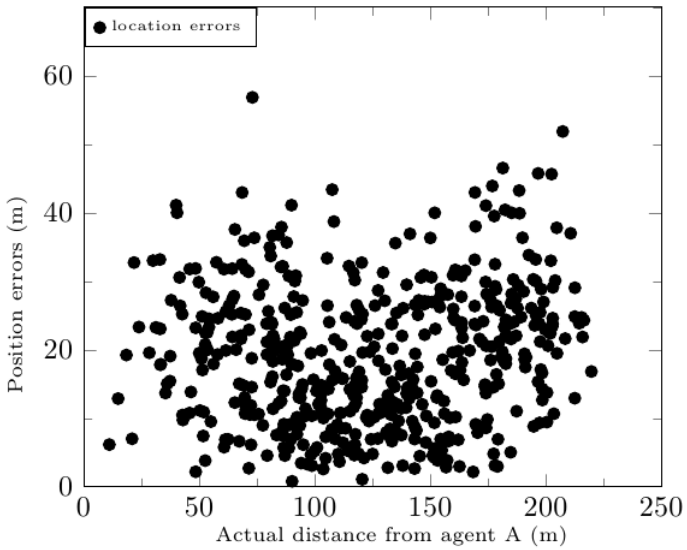

Figure 4.13. Observed scatter plot of localization errors of all the agents of 50 different systems with propagation constant 2, resolvable multipaths 6, for 10 arbitrarily chosen location in a 200×100 meter grid.

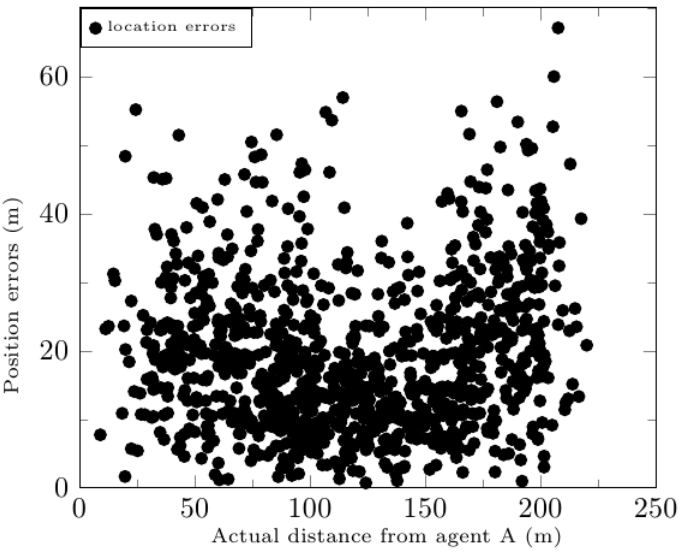

Figure 4.14. Observed scatter plot of localization errors of all the agents of 50 different systems with propagation constant 2, resolvable multipaths 6, for 20 arbitrarily chosen location in a 200×100 meter grid.

Figure 4.13 belongs to the 50 systems that have 10 agents and Figure 4.14 belongs to

rest of the systems each containing 20 arbitrarily chosen agents. From the plots, we could not see any specific distribution pattern of localization errors as a function of distance from anchor A. The errors are distributed equally at all the distances from the anchor. However, we observed that maximum value of location errors observed in systems that have larger number of agents is larger than the systems that have smaller number of agents when all of the agents are localized at once. This is to be expected since the error sizes are random and an increase in the number of samples naturally leads to selection of more outliers. Similarly, the maximum value of location error obtained with all the systems is about 60 m, and maximum number of agents have location error of about 30 m or below 30 m.

Chapter 5

CONCLUSIONS AND FUTURE WORK

A received signal strength based localization method was formulated for multipath induced fading in wireless channel. An analysis to determine conditions under which the error function is complex was presented in Chapter 3. We were able to establish that the error function is not convex over the entire region of interest. Therefore, we minimized the error function using multiple techniques and also conducted a number of experiments to verify that the minimization procedures are delivering the minimum of the error function. In future work, we intend to examine this issue in greater detail and develop a verification approach through which the error function is minimized with certainty. One possibility is to take advantage of the callback feature of some of the minimization routines to manage minimization. Optimal sensor location estimates and propagation constant were determined through a multistage process of formulating and minimizing error functions. Python was used for simulation and *scipy.optimize.fmin_ powell* for minimization of cost functions.

Investigation of the performance of our localization technique was done for a single agent case and multiple agent case. Localization errors obtained in both the cases were presented as a function of number of multipath components that can be resolved, and propagation constant of the environment. The simulation results indicate that the larger the number of resolvable multipaths for a given fixed propagation constant, the better is the localization accuracy achieved. Similarly, for a fixed number of resolvable multipaths, the localization accuracy is a increasing function of propagation constant. As the number of resolvable multipaths increases beyond 4, the improvement in localization accuracy of this approach

is reasonably low, which signifies that later arriving clusters do not contribute much in localization information.

Our localization technique is capable of giving reasonably accurate localization results in some cases and large errors in other. In worst case, we are getting position error of about 60 m, and an average position error is about 30 m and less. This error decreases as the number of resolvable multipaths increases. The localization error ranges from a few cm to tens of meters. Although we have not verified it through rigorous tests, performance of our localization method seems to be poor in the regions where cost function is non-convex. Because the range of localization error is so large, it would be good to develop a technique to quantify the quality of localization for each specific node. For example, in a real sensor network, the number of resolvable multipath would be different for each sensor. Therefore the localization error quality will vary. In future work, this issue should be analyzed to determine how to take advantage of this fact.

In the analysis, we use a simple statistical multipath channel model; an interesting extension would be to study the system under different channel models including shadow fading for the wireless channel. Exact conditions and regions where the cost function are convex as a function of x, y and α could be studied to see if the minimization can take advantage of convexity over regions. Furthermore, an extension could be carried out for 3-dimensional localization. It could be interesting to use different cluster delay rate and multipath delay rate for individual anchors instead of using same rates for all the anchors. It could also be extended to implement cooperative localization between agent nodes, but this will be left for future work.

BIBLIOGRAPHY

[1] A.H. Sayed, A. Tarighat, and N. Khajehnouri. Network-based wireless location: Challenges faced in developing techniques for accurate wireless location information. *Signal Processing Magazine, IEEE*, 22(4):24–40, July 2005.

[2] K. Pahlavan, Xinrong Li, and J.-P. Makela. Indoor geolocation science and technology. *Communications Magazine, IEEE*, 40(2):112–118, February 2002.

[3] J.J. Caffery and G.L. Stuber. Overview of radiolocation in cdma cellular systems. *Communications Magazine, IEEE*, 36(4):38–45, April 1998.

[4] M. Gavish and E. Fogel. Effect of bias on bearing-only target location. *Aerospace and Electronic Systems, IEEE Transactions on*, 26(1):22–26, January 1990.

[5] Chee-Yee Chong and S.P. Kumar. Sensor networks: evolution, opportunities, and challenges. *Proceedings of the IEEE*, 91(8):1247–1256, August 2003.

[6] Ali H Sayed and Nabil R Yousef. Wireless location. *Encyclopedia of Telecommunications*, April 2003.

[7] A.J. Weiss. On the accuracy of a cellular location system based on rss measurements. *Vehicular Technology, IEEE Transactions on*, 52(6):1508–1518, November 2003.

[8] Feng Zhao and Leonidas Guibas. *Wireless Sensor Networks: An Information Processing Approach*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2004.

[9] C. Savarese, J.M. Rabaey, and J. Beutel. Location in distributed ad-hoc wireless sensor networks. In *Acoustics, Speech, and Signal Processing, 2001. Proceedings. (ICASSP '01). 2001 IEEE International Conference on*, volume 4, pages 2037–2040 vol.4, May 2001.

[10] P. Bergamo and G. Mazzini. Localization in sensor networks with fading and mobility. In *Personal, Indoor and Mobile Radio Communications, 2002. The 13th IEEE International Symposium on*, volume 2, pages 750–754 vol.2, September 2002.

[11] B. Sklar. Rayleigh fading channels in mobile digital communication systems .i. characterization. *Communications Magazine, IEEE*, 35(7):90–100, July 1997.

[12] T.S. Rappaport. *Wireless Communications Principles and Practice*. Prentice Hall, 2008.

[13] Y. Shen and M.Z. Win. Fundamental limits of wideband localizationpart i: A general framework. *Information Theory, IEEE Transactions on*, 56(10):4956–4980, October 2010.

[14] A.E. Waadt, C. Kocks, S. Wang, G.H. Bruck, and P. Jung. Maximum likelihood localization estimation based on received signal strength. In *Applied Sciences in Biomedical and Communication Technologies (ISABEL), 2010 3rd International Symposium on*, pages 1–5, November 2010.

[15] R. A. Valenzuela A. M. Saleh. A stastical model for indoor multipath propagation. *IEEE Journal on Selected areas in communication*, 5, February, 1987.

[16] A.F. Molisch. Ultrawideband propagation channels-theory, measurement, and modeling. *Vehicular Technology, IEEE Transactions on*, 54(5):1528–1545, September 2005.

[17] Scipy reference guide; http://docs.scipy.org/doc/scipy/reference/.

[18] Craig Aaron Reimann. *Development of a General Methodology for Optimizing Acoustic Treatment using an Equivalent Source Method*. PhD thesis, North Carolina State University, 2007.

VITA

Rojina Adhikary received her Bachelor of Engineering degree in Electronics and Communication Engineering from Pulchowk Campus, Institute of Engineering, Tribhuvan University, Nepal in 2010. In January 2011, she joined the University of Mississippi for a M.S. degree with emphasis in Telecommunications. Since then, she has been a research assistant in the Center for Wireless Communications and she is completing her masters degree in the Department of Electrical Engineering at the university. Her research interest include localization of sensor nodes, ultrawideband signals, sensor networks, and optimization of localization networks. She is involved in wireless sensor localization and optimization techniques for wireless localization networks.