TÉCNICO LISBOA

Towards Secure Localization in Randomly Deployed Wireless Networks

Marcelo Salgueiro Costa

Thesis to obtain the Master of Science Degree in

Electrical and Computer Engineering

Supervisor(s): Prof. Marko Beko
Prof. Slaviša Tomic

Examination Committee

Chairperson: Prof. João Luís Costa Campos Gonçalves Sobrinho
Supervisor: Prof. Marko Beko
Member of the Committee: Prof. Francisco Monteiro

November 2022

To my brother and father.

Declaration

I declare that this document is an original work of my own authorship and that it fulfills all the requirements of the Code of Conduct and Good Practices of the Universidade de Lisboa.

Acknowledgments

I would like to start by express my gratitude towards my supervisors, professor Marko Beko and professor Slaviša Tomic, for their support, patience, and guidance throughout my academic path. I will never be able to fully express how thankful I am.

This work would also not have been possible without the constant support and motivation from my parents and brother, who have always believed in me and pushed me forwards. I would never have gotten this far without you.

I would like to thank my closest friends: Bruno Esparteiro, Mário Ferreira, and Diogo Nicolau. For what looks like a lifetime of friendship, you have always been by my side through the highs and lows, for that, I am eternal grateful. Thank you!

I would also like to thank my cousins, Carlos Salgueiro, and Pedro Salgueiro. Thank you for always listen me whenever I needed, for your advice in my most difficult choices, and for keeping my sanity intact during the most difficult times.

Last but not least, I would like to thank my grandmother, Fernanda Salgueiro. All that is good in me started in you.

Resumo

Este trabalho aborda o problema de localização de um alvo numa rede de sensores wireless (RSW) aleatoriamente distribuídos na presença de nós maliciosos capazes de manipular medidas de distância (i.e., efetuar ataques de *spoofing*) e assim, dificultando a localização precisa. Este problema torna-se extremamente importante com a expansão de IoT e aplicações de smart cities, que podem vir a depender em processos de localização precisos, e a presença de um nó atacante na rede pode representar uma ameaça de segurança se não for considerado. Para além disso, maior parte dos sistemas de localização existentes foram desenvolvidos para configurações não adversas, em que nenhuma ameaça de segurança é contemplada, tornando estes sistemas vulneráveis a ataques de *spoofing*. Portanto, é proposto um esquema de votos baseado em *clustering* e *weighted central mass* (WCM) para resolver o problema de localização de uma forma segura e detetar atacantes na rede. A solução proposta prevê duas fases: 1) Escolher um cluster de pontos de interseção por atribuição de votos a fim de localizar o alvo, e 2) Analisar a estimativa de localização e aplicar métodos de estatística para a deteção de atacantes na rede. O método proposto é estudado em termos de precisão de localização, sucesso de deteção de atacantes, e complexidade computacional para diferentes configurações. O desempenho do novo algoritmo é estudado por meio de simulações computacionais, que corroboram a eficácia do esquema proposto em relação aos métodos do estado da arte.

Keywords: Localização Segura, Redes de Sensores sem Fios, Localização, Intensidade do Sinal Recebido, Ataques de Spoofing.

Abstract

This thesis addresses the problem of target localization in randomly-deployed wireless sensor networks (WSNs) in the presence of malicious nodes capable of manipulating distance measurements (i.e., performing spoofing attacks) and thus, hindering accurate localization. This problem becomes extremely important with the forthcoming expansion of IoT and smart cities applications, that might depend on accurate localization, and the presence of malicious attackers can represent serious security threats if not taken into consideration. In addition, most existing localization systems are intended for non-adversarial settings, in which no security threats were contemplated, making them highly vulnerable to spoofing attacks. Therefore, this work proposes a novel voting scheme based on clustering and weighted central mass (WCM) to securely solve the localization problem and detect attackers. The proposed solution has two main phases: 1) Choosing a cluster of suitable intersection points by assigning votes in order to localize the target, and 2) Attacker detection by exploiting the location estimate and fundamental statistics. The proposed method is studied in terms of localization accuracy, success in attacker detection, and computational complexity for different settings. Performance of the new algorithm is studied through computer simulations, which corroborate the effectiveness of the proposed scheme compared to state-of-the-art methods.

Keywords: Secure Localization, Wireless Sensor Networks, Target Localization, Received Signal Strength, Spoofing Attacks.

Contents

List of Tables

List of Figures

Acronyms

ADMM	Alternating direction method of multipliers
AOA	Angle of arrival
ARMMSE	Attack-resistant minimum mean squares estimation
CRLB	Cramér-Rao lower bound
DV-Hop	Distance Vector Hop
GPS	Global positioning system
GRLT	Generalized ratio likelihood test
GTRS	Generalized trust region sub-problem
IoT	Internet of things
LB	Lower bond
LRT	Likelihood ratio test
LMS	Least median of squares
LN-1	l1-norm solution
MAP	Maximum a posterior
MC	Monte Carlo
ML	Maximum likelihood
MLE	Maximum likelihood estimation
MMSE	Minimum mean squares estimator
MSE	Mean square error
NNSS	Nearest neighbor in signal space
PDF	Probability density function.
PLE	Path loss exponent

RF	Radio frequency
RMSE	Root mean square error
RSS	Received signal strength
RTT	Round trip time
SDP	Secure weighted least squares
SNR	Signal-to-noise-ratio
SOCP	Semidefinite programming
SWLS	Secure weighted least squares
TDoA	Time difference of arrival
TOF	Time of flight
TOA	Time of arrival
VS	Voting scheme
WCM	Weighted central of mass
WLS	Weighted least squares
WSN	Wireless sensor network

Nomenclature

Greek symbols

δ	Attack intensity power
γ	Path loss exponent

Roman symbols

$\hat{\mathbf{x}}$	Targets location estimate
$\mathbf{A}$	Matrices: upper bold letter
$\mathbf{a}$	Vectors: lower bold letter
$\mathbf{a_i}$	Location coordinate of the i-th anchor node
$\mathbf{x}$	Location coordinate of the target node
$\mathbf{x_{att}}$	Location the malicious nodes agree to perform the attack
$\mathbf{0}_{m\times n}$	$m \times n$ matrix of all zero entries
$\boldsymbol{Q}_i$	i-th column of matrix $\boldsymbol{Q}$
$\cup$	Union between sets
$\hat{d}_i$	Estimated euclidean distance between the i-th anchor node and the target node
$\|x\|$	Euclidean norm
$\mathbb{R}$	Set of real numbers
$\mathbb{R}^n$	Set of n-dimensional vectors
$\mathbb{R}^{m\times n}$	$m \times n$ real matrices
$\mathcal{H}$	Honest node set
$\mathcal{M}$	Malicious node set
$\mathcal{N}(\mu, \sigma^2)$	Real-valued Gaussian distribution with meah μ and variance σ^2
$\sim$	Distributed according to

$\subseteq$ Subset of

$C_{ij}^{(u)}, C_{ij}^{(l)}$ Upper and lower half spaces point clusters, respectively

d_0 Short reference distance

d_i Euclidean distance between the i-th anchor node and the target node

$f(\cdot)$ Probability density function

$H^{(u)}, H^{(l)}$ Set of points for the upper and lower half spaces, respectively

H_{ij} Hyperplane between two anchor nodes

K Number of RSS observations

N Number of anchor nodes

$n_{i,k}$ Measurement noise for the k-th RSS observation from the i-th anchor node

p_0 Anchor node predefined transmit power

P_D Probability of detection

P_{FA} Probability of false alarm

$p_{i,k}$ k-th RSS observation from the i-th anchor node

p_i Median value of K RSS observation from the i-th anchor node

$proj_H(\boldsymbol{p})$ Distance of point $\boldsymbol{p}$ to it's respective projection on the hyperplane H

v Vote

Subscripts

i, j, k, h Computational indexes

l Reference to lower half space

u Reference to upper half space

Superscripts

T Transpose

Chapter 1

Introduction

A wireless sensor network (WSN) is communication network composed of sensor nodes distributed over a specific area with the goal of retrieving information from the environment to execute a certain task, such as detection (forest fires) [1], monitoring (agricultural, healthcare) [2–6], exploration (deep water) [7], and many more. In many applications, the acquired data are only relevant if they can be associated with the physical position (accurate and reliable) of the sensor. Section 1.1 starts with the motivation and importance of secure localization in WSN's. Section 1.2 presents the state-of-the-art, where some methods related to secure localization are described. Section 1.3 and Section 1.4 describe, respectively, the objectives and organization of the dissertation.

1.1 Motivation

Wireless sensor network (WSN) refers to a wireless communication network composed of several sensors scattered over a certain environment. The interest in WSN has been increasing, partially due to their ability to work in harsh environments, huge application potential [1–7] (owing to the fast growth of IoT), autonomous work in terms of human interaction, and ease and low cost of implementation [8–11].

From the localization perspective, a WSN is composed of two types of sensors: 1) anchor nodes, whose locations are known a priori and are used as reference points in the localization process and 2) target nodes, whose locations are unknown and one desires to determine. The role of a sensor in an WSN is to acquire data from the environment in which it was deployed (e.g. temperature, sound, movement, etc.) for later use by other devices in order to execute a specific task. A possible application is fire detection in a forest. For this application, some sensors are placed at different locations in a forest to measure the temperature in their vicinity. When a sensor detects a rise in temperature it sends a warning together with its location. Intuitively, in most applications, the data acquired by a sensor are only useful if one is capable to associate the measured data to its physical location. This can be done in various ways [12–15], but most existing systems are designed under the consideration that there are no security threats. Therefore, as a consequence, if exposed to spoofing attacks or, simply, malfunctions, most existing localization systems could result in catastrophic outcomes (e.g.,

failure in a collision-prevention system of an autonomous car, change in drone trajectory, etc.). For this reason, emerges the necessity to develop a localization system considering an adversarial setting, where a malicious (or damaged) sensor is capable of disrupting the localization process by producing false distance measurements (spoof attacks) [16].

Besides the security threats, sensors have limited communication and data processing capabilities due to their battery driven characteristics (low power); hence, a localization algorithm has to be computational efficient in order to be used in practical in real-time problems. Furthermore, it is beneficial the use of terrestrial radio frequency (RF) technologies when determining the target's location, since these technologies are well studied and can be easily integrated in WSN's. Nowadays, localization information based on RF signals can generally be acquired through range-free or range-based observations. This work focuses on the latter type exclusively, since they offer, in most cases, higher location estimation accuracy. Therefore, the target location is determined by using range-based techniques and the known positions of reference points (anchor nodes) in the case where a portion of them is malicious/damaged.

Perhaps the easiest and most common way of localization is to equip sensors with global positioning system (GPS) receivers. However, this solution has several undesirable consequences, such as increased implementation costs and unfeasibility in some environments (e.g., indoor, urban areas, forests, etc.). From the security perspective, GPS is considered a civilian localization system (e.g., uses unencrypted signals); thus, it is fairly easy to manipulate (spoof) it [17]. Therefore, with the objective to preserve low implementation costs and network integrity, a small number of the sensors might be equipped with GPS receivers or placed manually (reference points), while the remaining sensors are dependent on a localization algorithm in use and the reference points to establish their locations.

To tackle the security threats, a novel voting-based scheme for non-cooperative networks (where target nodes are only allowed to communicate with anchor nodes) to achieve a reliable localization with minimum computational complexity possible is proposed/described/introduced. Moreover, even though the novel solution takes advantage of terrestrial radio signals only from already deployed technologies in this thesis, it could also be used as a complement (or even the main localization scheme) of GPS system..

It is also important to mention that, although the present work focuses on localization in WSNs, the proposed solution can be extended to cellular and local area networks, since a base station or an access point can be seen as an anchor node, while other devices (e.g., cell phones, laptops, etc.) can be considered as targets. Therefore, it is expected that the proposed algorithm will be applicable in emerging 5G applications where secure localization plays an important role, such as in autonomous vehicles, smart city applications, swarm of robots, smart agriculture, etc.

1.2 Related Work

Recently, secure localization in WSNs has been addressed in the literature with numerous detection schemes [18–25] to tackle security threads in an adversarial scenario for both uncoordinated (where malicious nodes perform spoof attack independently) and coordinated attacks (where malicious nodes

perform spoof attacks collaboratively). Nonetheless, some of them are designed for very specific use cases and under a set of (little realistic) assumptions, while for others there is still plenty of room to achieve better localization precision and attack detection rates, while maintaining (or even reducing) the algorithm's computational complexity.

In [18], the authors explore two secure localization methods for two classes of localization schemes: triangulation, and RF fingerprinting, namely, least median of squares (LMS) and nearest neighbor in signal space (NNSS). Instead of minimizing the sum of the residue squares, the LMS solution minimizes the median of the residue squares in order to reduce the effect of outliers (altered measurements from the malicious node). Afterwards, the set of anchors are divided in different subsets, and for each subset an estimator is obtained. The final target location is given by a subset with least median residue. In the case of radio frequency fingerprinting, the authors use a variant of the k-nearest neighbor algorithm and a median-based distance metric to obtain the position estimate of a target.

The work in [19] investigates two types of secure localization techniques for range-based systems, namely, attack-resistant minimum mean squares estimation (ARMMSE) and a voting scheme. In the ARMMSE solution, firstly, the authors estimate the target's location with a standard minimum mean squares estimator (MMSE) and then compare the position estimate with a known genuine location reference in order to form a consistent set of reference points and remove the most inconsistent ones. To check the consistency of location reference points, each sensor uses the mean square error (MSE) of the distance measurements in order to estimate its own location. Intuitively, the more inconsistent a set of location references is, the greater the corresponding MSE should be. Therefore, a threshold based on the measurement error model is used to determine if a set of location references is (or not) consistent. However, the threshold has to be predefined and stored in each sensor before network deployment by conducting field experiments. The voting-scheme divides the deployment area into a grid and each anchor assigns votes to the grid cells based on the distance measurements. The highest voting cells define the area where the target is located. Accuracy of such solution obviously depends on the grid resolution, and its computational complexity is directly proportional to the grid resolution.

The work in [20] presented an iterative secure localization algorithm for both coordinated and uncoordinated attacks based on gradient descent. Once the gradient descent converges, the residues corresponding to malicious anchors tend to be higher than those from genuine anchors. Thus, the terms with high residues can be excluded from the cost function. Afterwards, the algorithm uses the terms with low residues together with the gradient descent to estimate the target's location.

Two types of secure localization techniques for range-based systems were investigated in [21], where the location points are divided into normal and abnormal clusters. The authenticity of an anchor is assessed by a sequential probability ratio test based on the consistency characteristics of received signal strength (RSS) and time of arrival (TOA) distance measurements (see Chapter 2 for more detail on distance measurements). The clustering algorithm is self-adaptive, which avoids outliers to be categorized in normal clusters.

The authors in [22] studied the problem of target location and velocity estimation for secure localization in mobile WSNs. The target location and velocity is estimated by a maximum a posterior (MAP)

criterion using distance and Doppler measurements. The problem was then formulated using the MAP estimator solved iteratively through a variational message passing algorithm. The detection of a malicious node is achieved through an auxiliary indicator vector of the MAP estimator. Similar as for [19], there is a trade off between localization accuracy and computational burden, which depend on the number of particles employed.

The solution proposed in [23] introduces a geometrical approach and threshold-based keying to detect corrupted anchors in range-based systems. This solution begins with an initial estimate where all anchors are treated as genuine ones. The initial location estimate is determined via WCM of a set of points of interest and weights based on the distance measurements. Subsequently, it computes distance estimates from the initial estimate to all anchors. The distance estimates are compared to a threshold in order to detect (and therefore remove) malicious anchors from the localization process. The predefined threshold is computed for a desired value of the probability of false alarm. Lastly, the localization problem is transformed into a generalized trust region sub-problem (GTRS) and solved by a bisection method using only *genuine* anchors. It is worth mentioning that the authors only consider two-way TOA measurements, where the malicious anchors can only enlarge the distance measurements.

The work in [24] is a generalization of [23], where it is assumed the use of any range-based technique (e.g., received signal strength, time of arrival, etc.), which results in a more challenging setting since the attackers can either enlarge or reduce their measurements. The presented solution is similar to the one in [23] in the sense that the initial estimate is obtained by WCM of a set of points of interest (some of which are fabricated). However, the detection scheme is based on the generalization likelihood ratio test (GLRT) to detect attackers, which requires estimating the noise standard deviation. For the location estimation, the proposed solution takes advantage of the law of cosines to convert the problem into a GTRS and solve it by a bisection method.

In [25], the authors presented two RSS-based solutions for target localization in the presence of malicious nodes, namely secure weighted least squares (SWLS) for uncoordinated, and l1-norm-based solution (LN-1) for coordinated attacks. In SWLS, the authors first compute the noise standard deviation for each anchor node. The malicious anchors are identified (and later removed from the localization process) by comparing the noise standard deviation of an anchor with a specific (empirical) threshold. Lastly, the weighted least squares (WLS) criterion is applied considering only the (supposedly) genuine anchors. However, the threshold employed is not very realistic in the sense that it is based on the true knowledge of noise power. In practice, one cannot exactly determine this value, which might compromise the performance of the SWLS solution in practice. In the LN-1 solution, the authors convert the problem into a 3-D plane fitting problem, where the data points correspond to the RSS measurements. Ideally, the malicious anchors would display greater variance due to the attack component present in the distance measurement and the genuine anchors would fit a plane. For a coordinated attack setting two planes can be found, one for the genuine anchors, and other for the malicious ones. The plane fitting problem is then solved using alternating direction method of multipliers (ADMM).

1.3 Objectives and Deliverables

This thesis addresses the problem of range-based localization in WSNs in the presence of malicious anchor nodes, which are capable of performing spoof attacks (enlarge or reduce distance measurements). The proposed solution takes advantage of the problem geometry to compute a set of intersection points (points of interest) between all pairs of anchors. Afterwards, a voting scheme to assign beliefs (votes) to each intersection point is employed. The intuition behind the voting scheme is not to exclude completely a detected malicious node, since they can still provide valuable information for the localization process (for instance, when the attack intensity is low). Besides, mistakenly removing a genuine anchor can severely degrade the localization accuracy. This way, by employing a voting scheme, the detected malicious nodes could still be used in the localization process and have less negative impact than some noise-corrupted genuine ones. A set of the highest votes of the intersection points are then converted into probabilities, which are used as weights to estimate the target's location via WCM. All anchors are subjected to scrutiny in order to estimate potential attack intensities in each link according to the maximum likelihood (ML) criterion and the obtained location estimate. The final decision on the attacker detection is founded on confidence intervals, with a predefined confidence level. It is worth mentioning that, due to the applied geometric approach, the presented algorithm can be easily adapted to any range-based measurement.

Therefore, the main focus of this work is to go beyond the traditional localization systems and develop a secure localization method capable of reliably detecting malicious nodes (if any) and securely (accurately) localizing a target node. The main contributions of this thesis are twofold:

1. Design of a novel solution for target localization in randomly-deployed sensor network in the presence of (un)coordinated (see Figure 3.1) spoofing attacks in a general setting, based on a new voting scheme. The proposed localization estimator is a very simple single-iteration scheme that matches or even outperforms more complex state-of-the-art solutions.

2. Proposal of a novel attacker detection scheme that exploits the target's location estimate and is based on confidence intervals.

1.4 Thesis Outline

This thesis is organized as follows. In Chapter 2, a brief theory of localization methods and distance measurements is presented. Chapter 2 also describes the considered scenario and formulates the problem of interest. Chapter 3 describes the derivation of the proposed algorithm. This chapter is organized in three parts: 1) a preliminary part, where points of interest are determined, 2) a detailed description of the proposed localization estimator based on the voting scheme, and 3) the detection procedure to identify attackers. In Chapter 4, evaluation of the performance of the proposed solution in terms of computational complexity, detection, and localization accuracy is given. Lastly, Chapter 5 summarizes and discusses the main findings of this thesis and lists possible directions for future work.

Publications. The results of this thesis have been submitted for publishing in: M. Costa, S. Tomic, M. Beko, ”Voting Scheme to Strengthen Localization Security in Randomly-deployed Wireless Networks”, IEEE Internet of Things Journal.

Previous work. Previous related work have been published in: M. Costa, S. Tomic, M. Beko, ”An SOCP Estimator for Hybrid RSS and AOA Target Localization in Sensor Networks”, Sensors, vol. 21, no. 5, March, 2021.

Chapter 2

Background

In this section, some basic concepts about estimation, together with detection theory and localization techniques are provided in order to better understand the background of the proposed work. Section 2.1 is based on the book [26] and briefly describes the Cramér-Rao lower bound (CRLB), the least squares (LS) method and the maximum likelihood estimation (MLE). Section 2.2 describes some fundamental concepts important for understanding detection theory, mainly the Neyman-Pearson theorem, and is based on the book [27]. Lastly, Section 2.3 introduces some of the classical localization techniques used in practice and is based on [28].

2.1 Basic Concepts of Estimation Theory

2.1.1 Cramér-Rao Lower Bound

Intuitively, all the information useful to derive an estimator is compressed in the probability density function (PDF) of the observed data. Therefore, the estimator is evaluated in terms of the PDF. For instance, imagine two Gaussian PDFs of unknown variance σ_1^2 and σ_2^2 and some data x, such that $\sigma_1^2 < \sigma_2^2$. Since the variance of the first PDF is smaller, then one should be able to estimate more accurately some unknown parameter of x from it [26]. The CRLB places a lower bound for the variance of an unbiased estimator. Hence, the CRLB provides us a benchmark against which one can compare how good an (unbiased) estimator is. Although there are numerous variance bounds [26], the CRLB is the most popular and relatively ease to determine. If an estimator meets this bound, then that estimator (if it exists) is fully efficient.

CRLB-Unknown Scalar Parameter

Let us consider a single sample

$$x = A + n, \tag{2.1}$$

where A is the unknown parameter to be estimated, n is a random variable with normal distribution, $n \sim \mathcal{N}(0, \sigma^2)$. The conditional PDF is expressed as follows

$$p(x|A) = \frac{1}{\sqrt{2\pi\delta^2}} \exp - \frac{(x-A)^2}{2\delta^2}. \tag{2.2}$$

The CRLB theorem states that under independent and identically distributed random variables, the regularity condition $\mathbb{E}\left[\frac{\partial \log(p(x|A))}{\partial A}\right] = 0$ is verified, the lower bound of any unbiased estimator is given by

$$Var(\widehat{A}) \geq \frac{1}{I(A)}. \tag{2.3}$$

The function $I(A)$ is known as the Fisher information and it works as measure of how much information a random variable carries about the unknown parameter A. The Fisher information is defined as

$$I(A) = -\mathbb{E}\left[\frac{\partial^2 \ln(p(x|A))}{\partial A^2}\right]. \tag{2.4}$$

Replacing (2.4) in (2.3) gives the final formulation of the CRLB

$$\begin{aligned} Var(\widehat{A}) &\geq \frac{1}{-\mathbb{E}\left[\frac{\partial^2 \ln(p(x|A))}{\partial A^2}\right]}, \\ Var(\widehat{A}) &\geq \sigma^2. \end{aligned} \tag{2.5}$$

Furthermore, an unbiased estimator attains the CRLB if and only if

$$\frac{\partial \ln(p(x|A))}{\partial A} = I(A)(\widehat{A} - A). \tag{2.6}$$

2.1.2 Least Squares

The LS method attempts to approximate the solution of a system by minimizing the residual error between the observed data and a transformed data according to a model function. In the LS method the goal is to minimize the squared difference between the input data and the observed data at a system output. In signal processing sense, a transmitted signal, $y(n)$, goes through a noisy channel and the receiver observes a set of noisy measurements, $x(n)$. Therefore, the goal is to determine an estimate to the signal $y(n)$ by minimizing the residual error between the observed data and a model function. For better understanding, let us consider a set of independent measured data, $\boldsymbol{x} = [x_1, x_2, ..., x_N]$, and dependent data to be estimated, $\boldsymbol{y} = [y_1, y_2, ..., y_N]$.

Linear Model

If the model is linear, the LS problem can be formulated as

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{\theta} \tag{2.7}$$

where

$$\boldsymbol{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix}, \boldsymbol{H} = \begin{bmatrix} h_1(x_1) & \dots & h_p(x_1) \\ \vdots & \ddots & \vdots \\ h_1(x_N) & \dots & h_p(x_N) \end{bmatrix}, \boldsymbol{\theta} = \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_N \end{bmatrix}, \tag{2.8}$$

where each entry of matrix $\boldsymbol{H}$ is composed of a linear function applied to the observation x, and $\boldsymbol{\theta}$ is the vector formulation of the model parameters. The cost function can be written as

$$J(\boldsymbol{\theta}) = \|\boldsymbol{y} - \boldsymbol{H\theta}\|^2 = (\boldsymbol{y} - \boldsymbol{H\theta})^T(\boldsymbol{y} - \boldsymbol{H\theta}) = \boldsymbol{y}^T\boldsymbol{y} - 2\boldsymbol{y}^T\boldsymbol{H\theta} + (\boldsymbol{H\theta})^T\boldsymbol{H\theta}. \tag{2.9}$$

Following the same reasoning as before, the gradient of the cost function is given as

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = -2\boldsymbol{y}^T\boldsymbol{H} + 2\boldsymbol{H}^T\boldsymbol{H\theta}. \tag{2.10}$$

Lastly, as it was done previously, the parameters estimate are obtained by setting the gradient of the cost function to zero. Therefore, the estimate is given by

$$\widehat{\boldsymbol{\theta}} = (\boldsymbol{H^T H})^{-1}\boldsymbol{H}^T\boldsymbol{y} \tag{2.11}$$

The invertibility of $\boldsymbol{H^T H}$ is guaranteed since $\boldsymbol{H}$ is a full rank matrix [29]. This assumption is perfectly valid since the measured data are linear independent.

Quadratic Model

Figure 2.1 shows a set of noisy measurements, with residual error e_i, that defines a convex function. Therefore, the goal is to fit y_i by a quadratic function (model function).

$$\widehat{y}_i = \beta_1 x_i^2 + \beta_2 x_i + \beta_3, \tag{2.12}$$

where $\widehat{y}_i$ is the estimated value of y_i and the model parameters are given by $\boldsymbol{\theta} = [\beta_1, \beta_2, \beta_3]$

The cost function is as follows,

$$J(\boldsymbol{\theta}) = \sum_{i=1}^{N}(y_i - \widehat{y}_i)^2 = \sum_{i=1}^{N}(y_i - \beta_1 x_i^2 - \beta_2 x_i - \beta_3)^2 \tag{2.13}$$

To find the best model, the LS error criterion can be minimized by taking the gradient with respect to $\boldsymbol{\theta}$ and equal it to zero. Therefore, a necessary condition to find the minimum is

$$\nabla_{\theta} J(\boldsymbol{\theta}) = \begin{bmatrix} \frac{\partial J(\boldsymbol{\theta})}{\partial \beta_1} \\ \frac{\partial J(\boldsymbol{\theta})}{\partial \beta_2} \\ \frac{\partial J(\boldsymbol{\theta})}{\partial \beta_3} \end{bmatrix} = 0 \tag{2.14}$$

Calculating the partial derivatives and reorganizing the variables into a matrix form, the problem can be expressed as

$$\boldsymbol{H\theta} = \boldsymbol{R}, \tag{2.15}$$

where

$$\boldsymbol{H} = \begin{bmatrix} \sum_{i=1}^{N} x_i^2 x_i^2 & \sum_{i=1}^{N} x_i x_i^2 & \sum_{i=1}^{N} x_i^2 \\ \sum_{i=1}^{N} x_i^2 x_i & \sum_{i=1}^{N} x_i x_i & \sum_{i=1}^{N} x_i \\ \sum_{i=1}^{N} x_i^2 & \sum_{i=1}^{N} x_i & 1 \end{bmatrix}, \boldsymbol{\theta} = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \end{bmatrix}, \boldsymbol{R} = \begin{bmatrix} \sum_{i=1}^{N} y_i x_i^2 \\ \sum_{i=1}^{N} y_i x_i \\ \sum_{i=1}^{N} y_i \end{bmatrix}, \tag{2.16}$$

Therefore, the parameter estimate $\widehat{\boldsymbol{\theta}}$ can be obtained using (2.15) as

$$\widehat{\boldsymbol{\theta}} = \boldsymbol{H}^{-1}\boldsymbol{R}, \tag{2.17}$$

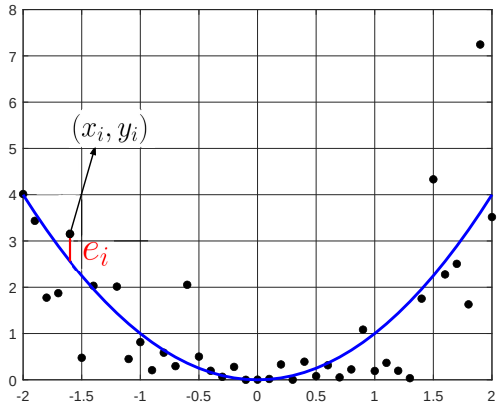


Figure 2.1: Illustration of least squared method for a quadratic model function.

2.1.3 Maximum Likelihood Estimation

The MLE is a method that determines values for the parameters of a given model. It is probably the most popular approach due to ease of implementation for complicated estimation problems, achieving the CRLB, and asymptotically efficient (when presented with enough data).

For better understanding this definition let's consider a realization $\boldsymbol{x} = [x_1, x_2, ..., x_N]$ of a sequence of variables with normal distribution such that $\boldsymbol{x} \sim \mathcal{N}(\mu, \sigma^2)$, the PDF is given by

$$f(\boldsymbol{x}) = \frac{1}{\sigma\sqrt{2\pi}} \exp - \frac{(\boldsymbol{x} - \mu)^2}{2\sigma^2}, \tag{2.18}$$

which is called the likelihood function.

The goal of the MLE is to determine μ and σ so that the Gaussian curve produced using the estimated parameters match the data. However, in order to use MLE there are two important assumptions: 1) the data must be independently distributed, and 2) the data must be identically distributed.

We can start by maximizing the probability of observed data given the parameter $\boldsymbol{\theta} = [\mu, \sigma]$. The goal is to maximize the probability density of the observed data, as a function of its parameters. From

probability theory it is known that the joint probability of independent events can be expressed as follows:

$$\begin{aligned} f(x_1, x_2, ..., x_N|\boldsymbol{\theta}) &= f(x_1|\boldsymbol{\theta}) \cdot f(x_2|\boldsymbol{\theta}) \cdot ... \cdot f(x_N|\boldsymbol{\theta}) \\ &= \prod_{i=1}^{N} f(x_i|\boldsymbol{\theta}) \end{aligned} \tag{2.19}$$

The goal is to estimate the parameter $\hat{\boldsymbol{\theta}}$ by maximizing (2.19). Therefore, the optimization problem can be expressed as

$$\widehat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} \prod_{i=1}^{N} f(x_i|\boldsymbol{\theta}). \tag{2.20}$$

In order to find the maximum value, (2.20) can be equated to the scenario when the derivative with respect to $\boldsymbol{\theta}$ equals zero. To facilitate, the derivative term is changed using the monotonic increasing function natural logarithm (log-likelihood function).

$$\sum_{i=1}^{N} \frac{\partial}{\partial\theta} \ln[f(x_i|\theta)] = \frac{\partial}{\partial\theta} \ln\left[\frac{1}{\sigma\sqrt{2\pi}} \exp - \frac{\sum_{i=1}^{N}(x_i-\mu)^2}{2\sigma^2}\right] = 0 \tag{2.21}$$

Using logarithmic properties, equation (2.21) can be simplified as follows

$$-\frac{\partial}{\partial\theta}\left[\sum_{i=1}^{N}\frac{(x_i-\mu)^2}{2\sigma^2}\right] = 0 \Leftrightarrow \begin{cases} -\frac{\partial}{\partial\mu}\left[\sum_{i=1}^{N}\frac{(x_i-\mu)^2}{2\sigma^2}\right] = 0 \\ -\frac{\partial}{\partial\sigma}\left[\sum_{i=1}^{N}\frac{(x_i-\mu)^2}{2\sigma^2}\right] = 0 \end{cases} \tag{2.22}$$

The final solution of the MLE is given by solving these equations in respect to the respective parameters.

$$\begin{aligned} \hat{\mu} &= \frac{1}{N}\sum_{i=1}^{N} x_i \\ \hat{\sigma} &= \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(x_i-\mu)^2} \end{aligned} \tag{2.23}$$

Note that in (A.6) the term $\frac{1}{N}$ was replaced by $\frac{1}{(N-1)}$. This is because, typically, if we calculate the standard deviation of each sample using $\frac{1}{N}$, and then average all the supposed estimates of σ, we find out that their averages is less than the actual value of σ. Dividing by $(N-1)$ is a way to compensate and get a more precise estimator.

To summarize, the MLE method is a way to find an unbiased estimator for the parameters of an assumed PDF, given enough observed data. This is achieved by maximizing the likelihood function (or PDF) so that the parameters estimated describe the observed data. If the likelihood function is differentiable, the maxima can be found using the derivative and set it to zero. In order to simplify the problem, we can take the natural logarithm of the likelihood function, since it is a monotonic function. Therefore, the estimator is given by solving the derivatives of the log-likelihood function with respect to the respective parameters and set it to zero.

2.2 Basic Concepts on Detection Theory

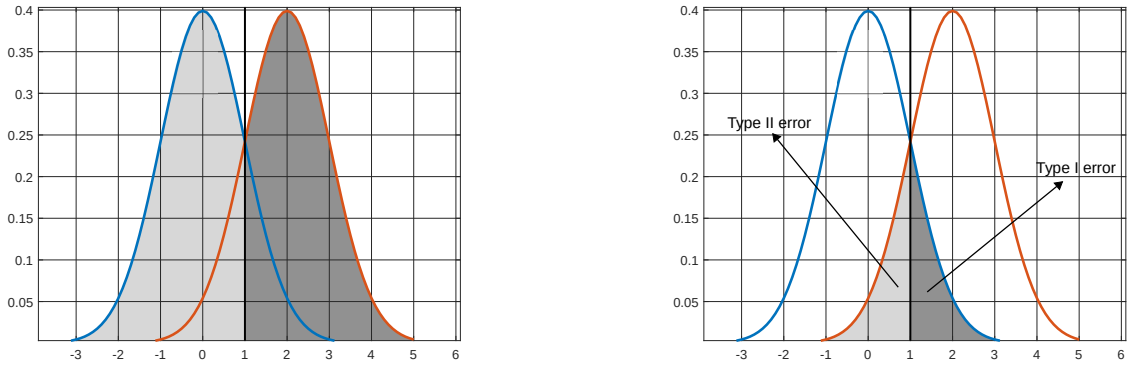
This section presents some basic concepts on decision theory, such as hypothesis test, type of errors, probability of false alarm and probability of detection for a single value observation. Lastly, the Neyman-Pearson theorem is addressed, where it is defined the likelihood ration used to decide between hypothesis for a sequence of observations.

2.2.1 Neyman-Pearson Theorem

For the sake simplicity lets start by the binary hypothesis testing of a single value,x. Lets assume two Gaussian distributions with different mean and variance 1, $\mathcal{N}(0,1)$ and $\mathcal{N}(1,1)$. The goal is to decide which distribution better describes x. Therefore, our hypotheses are defined as:

- $\mathcal{H}_0 : \mu = 0$;
- $\mathcal{H}_1 : \mu = 1$;

Intuitively, the first approach that comes to mind is to threshold the PDFs at the PDFs intersection points. This way, if x is below the threshold it is more likely to belong to the PDF defined by $\mathcal{H}_0$ and vice versa (see Figure 2.2(a)).



(a) (b)

Figure 2.2: Binary hypothesis testing.

On the other hand, Figure 2.2(b) defines two types of errors.

- Type I error: decide $\mathcal{H}_1$ while $\mathcal{H}_0$ is the correct hypothesis with error probability $p(\mathcal{H}_1|\mathcal{H}_0)$;
- Type II error: decide $\mathcal{H}_0$ while $\mathcal{H}_1$ is the correct hypothesis with error probability $p(\mathcal{H}_0|\mathcal{H}_1)$;

Interestingly, these errors are defined by a threshold in the sense that if the threshold is shifted to the right side the type I error decreases but the type II error increases. Therefore, it is not possible to reduce both errors simultaneously. A typical approach to design optimal detector is to fix one of the probability errors and minimize the other, for example, set the probability of type I error to $p(\mathcal{H}_1|\mathcal{H}_0) = \zeta$. This way, $p(\mathcal{H}_1|\mathcal{H}_0)$ is referred to as probability of false alarm, P_{FA}, and $p(\mathcal{H}_1|\mathcal{H}_1)$ is referred to as probability of

detection P_D. The probability of false alarm and the probability of detection are defined in terms of the Q function as

$$\begin{aligned} P_{FA} &= Prob\{x > \epsilon|\mathcal{H}_0\} = \int_{\epsilon}^{\infty} \frac{1}{\sqrt{2\pi}} \exp^{-0.5x^2}\, dx = Q(\epsilon), \\ P_D &= Prob\{x > \epsilon|\mathcal{H}_1\} = \int_{\epsilon}^{\infty} \frac{1}{\sqrt{2\pi}} \exp^{-0.5(x-1)^2}\, dx = Q(\epsilon - 1), \end{aligned} \tag{2.24}$$

where ϵ is the threshold.

For better understanding lets see an example, imagine we want a probability of false alarm of 10^{-5}. Then

$$\begin{aligned} &P_{FA} = Q(\epsilon) \Leftrightarrow \epsilon \approx 4, \\ &\text{and} \\ &P_D = Q(4-1) = 1.3 \times 10^{-3}. \end{aligned} \tag{2.25}$$

The generalization to a sequence of observed data, $\boldsymbol{x} = \{x_1, x_2, \ldots, x_N\}$, is fairly similar. The goal is for the detector to map each data value to a decision. This way, one can define a set of values that would result in a $\mathcal{H}_1$ decision, $R_1 = \{\forall x_i : \text{results in } \mathcal{H}_1 \text{ decision}\}$, and another set of values that would result in a $\mathcal{H}_0$, $R_0 = \{\forall x_i : \text{results in } \mathcal{H}_0 \text{ decision}\}$, decision. Therefore, the probability of false alarm and the probability of detection are defined as

$$\begin{aligned} P_{FA} &= \int_{R_1} p(\boldsymbol{x}|\mathcal{H}_0)\, d\boldsymbol{x}, \\ P_D &= \int_{R_1} p(\boldsymbol{x}|\mathcal{H}_1)\, d\boldsymbol{x}. \end{aligned} \tag{2.26}$$

The Neyman-Pearson theorem gives a solution to R_1 that maximizes P_D for a given $P_{FA} = \epsilon$.

$$L(\boldsymbol{x}) = \frac{p(\boldsymbol{x}|\mathcal{H}_1)}{p(\boldsymbol{x}|\mathcal{H}_0)} > \epsilon, \tag{2.27}$$

where $L(\boldsymbol{x})$ is known as the likelihood ratio and (2.27) is referred as likelihood ratio test (LRT). The threshold ϵ is obtained, once more, by fixing the probability P_{FA}. Therefore, if $L(\boldsymbol{x}) > \epsilon$ decide $\mathcal{H}_1$; otherwise decide $\mathcal{H}_0$.

2.3 Localization Measurements, Models and Classical Methods

Distance measurements can be obtained through *geometric-based*, *range-based*, or *range-free* [30] methods. The range-based observations use additional devices, such as antennas, timers, intensity signal receivers, to compute a measurement of distance/position. These methods use properties of a received radio signal, such as time of arrival (TOA) [31, 32], time difference of arrival (TDoA) [33],

round trip time (RTT) [34], time of flight (TOF) [35], angle of arrival (AoA) [36, 37], received signal strength (RSS) [38, 39], etc. Besides the traditional range-based methods (where one property of the received radio signal is exploited), there are also hybrid techniques [8, 13, 15, 40–42], where more than one of the signal characteristics are acquired to obtain more information in order to achieve a better location estimate. In contrast, the range-free methods [43–45] do not require additional hardware, instead the prior knowledge of the network properties (e.g., hops between transmitter and receptor, proximity between nodes, etc.) are used to compute the target's location. Lastly, the geometric-based methods, as the name indicates, are based on the geometry of the problem. Normally, these methods explore the scenario geometric properties to estimate the target's location.

This thesis focuses on the integration of range-based methods since, generally, they achieve higher localization accuracy. Which type of measurement to use depends on the implementation, for example, a network that uses TOA measurements requires clock synchronization between all sensors in the network [28], increasing the network complexity, while AoA only requires the sensors to be equipped with an array of antennas [28]. The proposed method, as it will be shown later, is developed under the assumption that RSS measurements are employed, but, due to its geometry-based nature, it can easily be generalized to any range-based setting. The reason for the particular interest in RSS measurements is that it is a popular method in the literature since RSS is available in practically all devices, and does not require additional hardware besides a radio frequency emitter and receiver.

2.3.1 Range-free Methods/Localization

Range-free localization requires no distance or angle measurements among nodes to estimate a target location. Instead, this type of localization techniques takes advantage of the networks properties. This subsection briefly describes some of the most popular range-free methods nowadays.

RF fingerprinting:

Perhaps the most famous range-free method is RF fingerprinting. RF fingerprinting is a method to identify wireless devices using the received RF signal properties (e.g., different transmitted power for each sensor), which are difficult to imitate [46–48]. During the process of localization using RF fingerprints, one can distinguish between two phases: 1) a data collection phase, to store all the fingerprints from the devices present in communication network, and 2) a localization phase, where a received radio signal is compared with all the fingerprints stored to decide the best fit location to estimate the target location. However, fingerprinting relies on having collected a sufficient amount of data, which can be time consuming and prone to error, especially in cases where the environment conditions can change relatively rapidly.

Distance Vector Hop (DV-Hop): DV-Hop is a localization technique based on the number of hops a device is from another one [49–51]. In DV-Hop the anchor nodes broadcasts their location and the initialized hop counter $(x_i, y_i, 0)$. When a device receives a message of this type, it stores the information and broadcast the same message plus an increment on the hop counter. This cycle is repeated until all devices have stored the number of hops from the anchor node. Afterwards, the number of hops is

replaced by a parameter called hop size, which gives a physical perspective of distance between hops. Lastly, the target position is estimated via trilateration (see Figure 2.5(b)) using the hop size and the number of hops. Unfortunately, DV-Hop uses the average distance, which can result in large estimation errors.

2.3.2 Range-based Methods/Localization

Range-based localization methods are widely used due to high accuracy and applicability to most RF technologies. This subsection describes some of the currently most used range-based methods.

Time of Arrival (TOA)

There are two types of TOA systems. The first one assume that the anchors have fully synchronized clocks. Basically, a signal is sent from the target to an anchor with the departure time. At the anchor side, the departure time is compared with the time of arrival and a estimate of distance can be done based on the time of flight. It is important to notice that this type of system requires clock synchronization and, as it was already mentioned, increases the complexity of the network. The second type of TOF system is the two-way TOA and does not require full synchronization across the devices in the network. In this system a device sends a message and waits for a reply. This way the time of flight is obtained locally. However, this implies reduced vacancy of the channel (since the communication is in two directions). Given the Figure 2.3, the time for a RF signal to travel from a target to an anchor and come back can be modeled as

$$t = \frac{d}{v_p} + \frac{T}{2} \tag{2.28}$$

where T is processing time at the anchor side, and t is the total time of this process. The propagation time, t_p, is equal to the distance between transmitter and receiver sensors, d, divided by the propagation velocity v_p.

There are two sources of of error in TOA systems: 1) multipath and 2) additive noise. The TOA estimate is given by the peak in the cross-correlation between the received and the known transmitted signal. Therefore the problem of measurement noise is well studied [52] and does not represent great adversity.

Setting aside the hardware and software time delays, the main source of error in TOA systems is due to multipath channels, where there are early and late components of the transmitted signal, with different arriving times, that interfere with the transmitted signal and decrease the signal-to-noise-ratio (SNR). Adding to this problem, the strongest signal arriving at the receiver can be any multipath component, and not necessarily the direct-path one. Intuitively, the time measured is given by the first arriving peak of the cross-correlation that reach a predefined threshold, which, obviously, give rise to time measurement errors. Therefore, it is defined the early-arriving multipath problem, where the multipath components arrive sooner than the direct signal and meet the cross correlation threshold and therefore obscure the peak from the transmitted signal. The power of the direct-path component increases when the distance

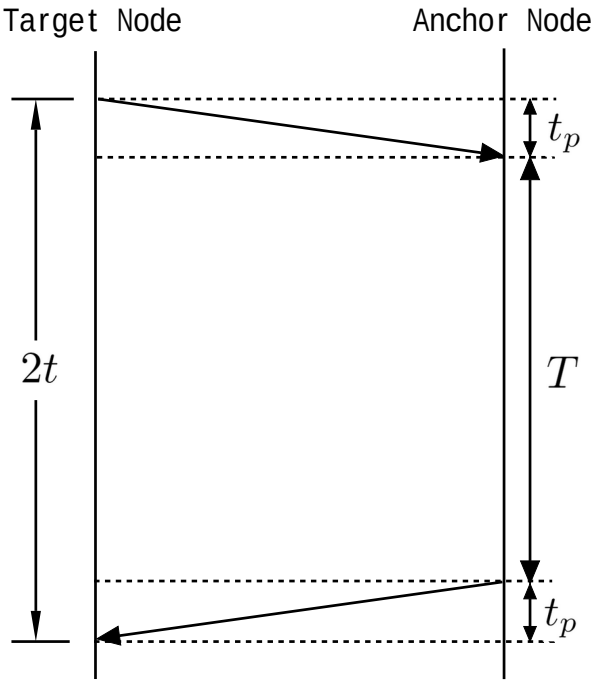


Figure 2.3: Illustration of TOF measurements in the presence of noise.

between sensors decrease [28]. Therefore, the early-arriving problem is diminished for dense sensor networks. Besides the early-arriving multipath problem there is the attenuation of the transmitted signal compared to the late-arriving multipath components. However, even though late-arriving components generate greater errors, the early-arriving problem is more difficult to tackle. In order to distinguish between the early multipath components and the transmitted signal it is essential to obtain a narrow auto-correlation peak. However, the most popular approaches, such as ultra-wideband, require wide signal bandwidths which increase the device costs, energy costs and demand higher processing speed.

TOA Model

Measured time delay between a target sensor, $\boldsymbol{x}$, and the i-th sensor can be modelled as Gaussian distribution [28], $t \sim \mathcal{N}\left(\frac{d_i}{v_p} + \mu, \sigma^2\right)$, where μ and σ are, respectively, the mean and variance of the time delay error. Several experiments have been conducted using wideband direct-sequence spread-spectrum and showed $\mu = 10.9$ ns and $\sigma = 6.1$ ns [53], and for ultra-wideband systems, where it was shown $\mu = 0.3$ ns and $\sigma = 1.9$ ns [54].

Received Signal Strength (RSS)

Perhaps the most popular range-based method, the RSS measurements can be easily implemented without great costs. The RSS is defined as the power received by a sensor. In WSNs, a sensor communicates with its neighbors via RF signals, hence, the distance between a target and an anchor (reference point with known position) in a WSN can be characterized by the received power of the RF signal at the target side. Figure 2.4(a) illustrates how the distance measurements ($\widehat{d_1}, \widehat{d_2}, \widehat{d_3}$) are obtained in a WSN composed of three anchors (a_1, a_2, a_3). The presence of noise/interference prevents precise measurements, hence the red area defines a set of possible positions. It is worth mentioning that the distances

$\widehat{d}_i$ are observations and not the true distance to the target.

In real-world applications, the RSS measurement depends on the environment characteristics, which can result in *shadowing* (e.g., loss in signal power due to obstructions in the path) and *multipath fading* (e.g., due to reflections and refractions, multiple versions of the original signal reach the receiver via many paths with changed phase and power). In multipath fading, the overall received signal at the receiver is a sum of all the signals received. The received signals will add or subtract depending on the relative phase. This type of fading can be fought using spread-spectrum methods (i.e., frequency hopping, direct sequence, pseudo noise, or linear frequency modulation).

RSS Model

Considering now the attenuation of the signal due to obstructions in the propagation path (shadowing), this effect can me modeled as

$$P_R = P_T - 10\gamma \log_{10}\left(\frac{d}{d_0}\right), \tag{2.29}$$

where P_R and P_T are the received and transmitted power, respectively, and d is the distance between the emitter and the receiver for a distance reference d_0. One can notice that the received power decays proportionally to γ, where γ is the path loss exponent, which characterizes the losses along the way (power decays with $d^{-\gamma}$) and, typically, ranges between 2 and 6 (see Table 2.1).

MLE distance estimate from RSS measurements.

Considering the received power (dBm), P_i, at a target sensor, $\boldsymbol{x}$, and transmitted by the i-th sensor, $\boldsymbol{a}_i$ is distributed as a Gaussian PDF with mean $P_T - 10\gamma \log_{10} \frac{d_i}{d_0}$ and the received power standard deviation in dB, σ^2, which is constant with the distance. Therefore, the conditional PDF is given as

$$f(P_i|\boldsymbol{x}) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{\left(P_i - P_T + 10\gamma \log_{10}\left(\frac{d_i}{d_0}\right)\right)^2}{2\sigma^2}\right\}, \tag{2.30}$$

The distance between sensors can be estimated using the maximum likelihood estimate. The log-likelihood of p_i is

$$\log\left[f(P_i|\boldsymbol{x})\right] = -0.5\log\left(2\pi\sigma^2\right) - \frac{\left(P_i - P_T + 10\gamma \log_{10}\left(\frac{d_i}{d_0}\right)\right)}{2\sigma^2}, \tag{2.31}$$

The distance estimator, $\widehat{d}_i^{MLE}$ is given by the derivative of the log-likelihood expression with respect to the distance set to zero. Therefore, the estimator that maximizes the log-likelihood is

$$\widehat{d}_i^{MLE} = d_0 10^{\frac{P_T - P_i}{10\gamma}}, \tag{2.32}$$

and

$$\mathbb{E}[\widehat{d}_i^{MLE}] = \exp\left\{\frac{10\gamma}{2\sigma \log 10}\right\} d_i, \tag{2.33}$$

which makes the estimator biased with a multiplicative factor. Since the received power variance is constant with distance, the bias factor is also constant with distance. For example, for a true distance

Environment	Path loss exponent
Free space	2
Urban area	2.7 to 3.5
In building with line-of-sight	1.6 to 1.8
Obsctucted in building	4 to 6

Table 2.1: Measured γ for different environments [55]

of 10m between sensors and bias factor of 1.2, the measured distance would be 12m, an error of 2m. Naturally, the error increases with the distance between sensors.

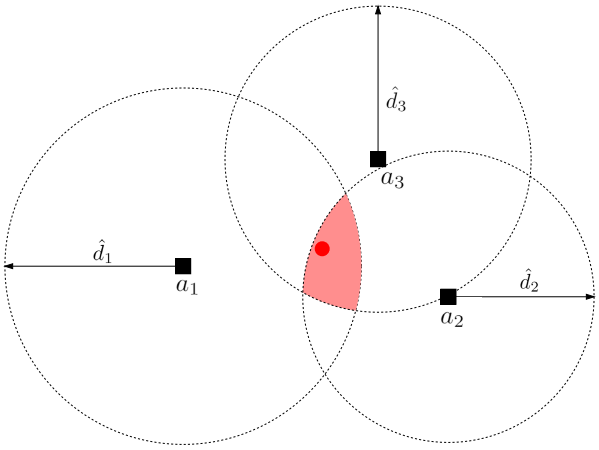
´

Angle of Arrival (AOA)

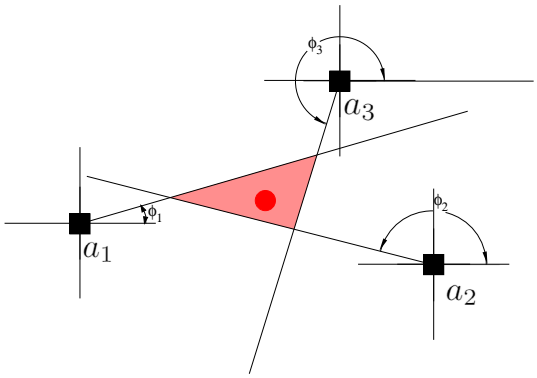
The AOA measurement is based on the direction of the received signal. In contrast to other localization measurements, AOA requires the sensors to be equipped with an antenna array to capture the angle of the received signal. The AOA estimate is obtained from the differences in time of arrival for a received signal at each of the antennas in the array. However, the accuracy of AOA is greater in comparison to RSS [56] (due to the usage of antenna arrays and signal processing). Figure 2.4(b) illustrates how AOA can be used to find a target position in a WSN where, once again, the red area represents a set of possible positions. The major sources of errors are the same as discussed for the TOA, multipath and measured noise.

Hybrid Methods

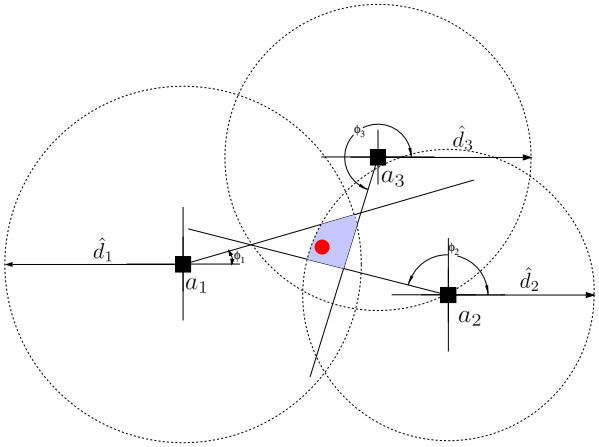
Recently, hybrid systems that introduce two range-based measurements have gained some interest [8, 13, 40]. These hybrid systems take advantage of combined measurements to reduce the area of possible solutions introduced by the noise (and possible attackers). Nevertheless, it seems that range and bearing measurements are good complements. Figure 2.4 illustrates a scenario where distance (2.4(a)) and angular (2.4(b)) measurements are degraded by the presence of noise. The distance measurements $(\widehat{d_1}, \widehat{d_2}, ..., \widehat{d_N})$ are used to obtain a circumference which describes the possible positions for a respective anchor node. However, as it was seen previously, due to noise, instead of a single point, we have an area of possible solutions. Similarly, in the case of angular information, the observed angle, ϕ_i, is used to define a line in the direction of the received signal, in other words, the target's direction. The area of possible solutions is given by the intersection points of, at least, three lines. Figure 2.4(c), shows a decrease in the area of possible solutions due to the integration of distance and angular information. To conclude, the use of multiple measurement techniques can improve the target localization accuracy. However, this improvement usually comes at the cost of increased complexity.



(a) RSS measurements



(b) AOA measurements



(c) RSS/AOA hybrid measurements

Figure 2.4: 2-D Illustration of the impact of noise in range-based, bearing-based, and combined range-bearing-based measurements.

2.3.3 Classical Localization Methods

Geometric-based localization techniques are better known for their simplicity. In this subsection it is intended to give a geometric interpretation of the most known geometric-based techniques.

- *Trilateration*:

  Trilateration is a technique that takes advantage of distance/range measurements and the known positions of the anchors to describe a circle centered at the respective anchor position and radius equal to the distance measurement obtained through the RSS measurement. The position of the target is given by the intersection of the circles. Intuitively, for a 2D scenario, it is required to have at least three anchors to obtain a single point for the target's position. This technique has only one (undemanding) restriction, the anchors cannot be collinear.

- *Multilateration*:

  Multilateration is given as the difference between a reference measurement and the remaining ones, where the reference measurement is subtracted to the other measurements [57]. The problem formulation is similar to the trilateration. However, it offers improvements in accuracy.

  The problem of localization using multilateration can be formulated as a LS problem. Let $\widehat{d}_1, \widehat{d}_2, ..., \widehat{d}_n$ be the distance estimations (RSS) between anchors, $\boldsymbol{a_1}, \boldsymbol{a_2}, ..., \boldsymbol{a_n}$, and a target, $\boldsymbol{x}$ in a 2D setting.

$$\begin{cases} ||\boldsymbol{x}-\boldsymbol{a_1}||_2^2 = \widehat{d}_1^2 \\ ||\boldsymbol{x}-\boldsymbol{a_2}||_2^2 = \widehat{d}_2^2 \\ \vdots \\ ||\boldsymbol{x}-\boldsymbol{a_n}||_2^2 = \widehat{d}_n^2 \end{cases} = \begin{cases} ||\boldsymbol{x}||_2^2 - 2\boldsymbol{a_1}^T\boldsymbol{x} + ||\boldsymbol{a_1}||_2^2 = \widehat{d}_1^2 \\ ||\boldsymbol{x}||_2^2 - 2\boldsymbol{a_2}^T\boldsymbol{x} + ||\boldsymbol{a_2}||_2^2 = \widehat{d}_2^2 \\ \vdots \\ ||\boldsymbol{x}||_2^2 - 2\boldsymbol{a_n}^T\boldsymbol{x} + ||\boldsymbol{a_n}||_2^2 = \widehat{d}_n^2 \end{cases} = \begin{cases} 2(\boldsymbol{a_2}-\boldsymbol{a_1})^T\boldsymbol{x} = \widehat{d}_1^2 - \widehat{d}_2^2 + ||\boldsymbol{a_2}||_2^2 - ||\boldsymbol{a_1}||_2^2 \\ 2(\boldsymbol{a_3}-\boldsymbol{a_1})^T\boldsymbol{x} = \widehat{d}_1^2 - \widehat{d}_3^2 + ||\boldsymbol{a_3}||_2^2 - ||\boldsymbol{a_1}||_2^2 \\ \vdots \\ 2(\boldsymbol{a_n}-\boldsymbol{a_1})^T\boldsymbol{x} = \widehat{d}_1^2 - \widehat{d}_n^2 + ||\boldsymbol{a_n}||_2^2 - ||\boldsymbol{a_1}||_2^2 \end{cases} \tag{2.34}$$

  Therefore, the localization problem can be approximated to a LS problem as

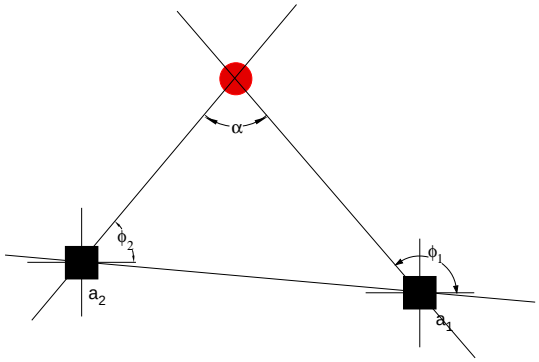
$$\min_x ||\boldsymbol{A}\boldsymbol{x}-\boldsymbol{b}||_2^2$$
$$A = \begin{bmatrix} 2(\boldsymbol{a_2}-\boldsymbol{a_1})^T \\ 2(\boldsymbol{a_3}-\boldsymbol{a_1})^T \\ \vdots \\ 2(\boldsymbol{a_n}-\boldsymbol{a_1})^T \end{bmatrix}, \; b = \begin{bmatrix} d_1^2 - d_2^2 + ||\boldsymbol{a_2}||_2^2 - ||\boldsymbol{a_1}||_2^2 \\ d_1^2 - d_3^2 + ||\boldsymbol{a_3}||_2^2 - ||\boldsymbol{a_1}||_2^2 \\ \vdots \\ d_1^2 - d_n^2 + ||\boldsymbol{a_n}||_2^2 - ||\boldsymbol{a_1}||_2^2. \end{bmatrix} \tag{2.35}$$

  This way, we can have a closed form solution.

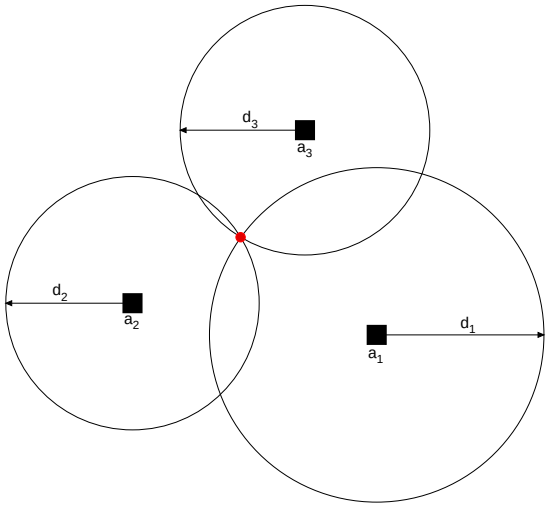
$$\widehat{\boldsymbol{x}} = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T\boldsymbol{b} \tag{2.36}$$

- *Triangulation*:

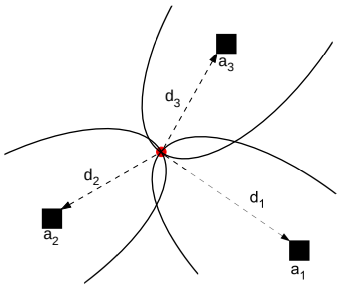
Triangulation is a geometric approach that uses the angles between pairs of anchors and the target (see Figure 2.5(a)) instead of distance. The angle information can be computed through trigonometric laws of cosines and sines. In Figure 2.5(a), the target position, x, can be obtained using the law of cosines [58]. Figure 2.5 illustrates the range-based techniques described.



(a) Triangulation



(b) Trilateration



(c) Multilateration

Figure 2.5: Geometric illustration of classical/traditional localization techniques in a WSN.

However, in the presence of noise (see Figures 2.4(a) and 2.4(b)), the solution that these techniques offer are not feasible. The lines instead of intersecting in a single point, hence giving a single solution, we have an area which describes a set of possible solutions. The larger the area formed by the noisy intersections, the larger the set of possible solutions becomes and, consequently, more difficult the estimation task becomes. After presenting this adversity, it is clear that it is very difficult to obtain the exact target location.

Chapter 3

The Proposed Method

This Chapter is organized as follows. Section 3.1 formulates the problem of interest. Section 3.2 and 3.3 provide a detailed description of the proposed solution namely, how points of interest are obtained and the voting-scheme. Lastly, Section 3.4 describes the detection scheme used.

3.1 Problem Formulation

Let us consider a 2-dimensional (generalization to 3-dimensions is straightforward), non-cooperative and static WSN, where a single target node, whose true location is unknown and denoted by $\boldsymbol{x}$, is located at a time by the help of a set of anchor nodes whose true locations are known and denoted by $\boldsymbol{a}_i, i = 1, ..., N$. It is assumed that some of the anchors are malicious and try to disrupt the location process by manipulating their distance measurements (spoofing attacks). The target node receives radio signals (from which it measures the RSS values) from the anchors[1]. There are two types of spoofing attacks that can be performed to disrupt the localization process: coordinated and uncoordinated. The algorithm presented in this thesis is suitable for both types of attacks. Since in practice, it is highly unlike that one knows under which type of attacks (if any) the network is a priori, here, no assumptions regarding the type of attacks will be made. Nevertheless, both types of spoofing attacks are described in the following two subsections.

1. *Uncoordinated Attack*

In an uncoordinated attack, the malicious anchors perform attacks independently. In this setting, the genuine anchors have a predefined transmitted power, while the malicious anchors change the transmitted power arbitrarily without notifying the network about this change. The k-th RSS measurement sample ($k = 1, ..., K$) between the target node and the i-th anchor node can be modeled as

$$p_{i,k} = p_0 - 10\gamma \log_{10}\left(\frac{d_i}{d_0}\right) + \delta_i + n_{i,k} \ , \tag{3.1}$$

[1]Note that this work opted to consider RSS-based localization due its *global* availability in almost all devices and also for convenience of comparison with existing methods (namely with [25]). Nevertheless, its generalization to any range-based setting is straightforward, since it only requires distance information as it will be seen in Chapter 3.

p_0 is the RSS at a short reference distance d_0, for simplicity referred to as the transmitted power, $d_i = \|\boldsymbol{x} - \boldsymbol{a}_i\|$ is the true distance for the i-th link, γ is the path loss exponent (PLE) which represents the decay of signal strength with distance, $n_{i,k}$ is the noise term modeled as a zero-mean Gaussian random variable with variance $\sigma_{i,k}^2$, i.e., $n_{i,k} \sim \mathcal{N}(0, \sigma_{i,k}^2)$, and $\delta_i \in \mathbb{R}$ represents the intensity of the spoofing attack of the i-th anchor node, where $\delta_i = 0$ refers to an honest node and $\delta_i \neq 0$ to a malicious one [25].

Lastly, it is important to note that, in contrast to [23], where the malicious anchor nodes could only enlarge their distance measurements, in this work it is assumed that the attackers can either reduce or enlarge distance measurements. For simplicity, it is assumed that the measurements variance is equal for every link and sample, i.e., for $\sigma_{i,k}^2 = \sigma^2, i = 1, ..., N$ and $k = 1, ..., K$. Moreover, to easier combat outliers and also for the sake of notation simplicity and without loss of generality, the median of all K RSS measurements in (3.1) from the i-th anchor node (p_i) is used in the following derivations.

Figure 3.1(a) shows an uncoordinated attack, where three of the seven anchors are malicious and report false distance measurements, aggravating the localization process. It is important to note that, indeed, the malicious nodes act in an independent manner by changing the predefined value of transmitted power. From (3.1) (and by employing p_i) the conditional PDF of the RSS model for a single anchor is given by

$$f(p_i|\boldsymbol{x}) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{ -\frac{\left(p_i - p_0 + 10\gamma \log_{10}\left(\frac{d_i}{d_0}\right) - \delta_i\right)^2}{2\sigma^2} \right\}, \tag{3.2}$$

where $\exp\{\bullet\}$ denotes the exponential function.

From (3.2), the location of the target node can be estimated based on the ML criterion [59] as

$$\widehat{\boldsymbol{x}} = \arg\min_{x} \sum_{i=1}^{N} \left(p_i - p_0 + 10\gamma \log_{10}\left(\frac{\|\boldsymbol{x} - \boldsymbol{a}_i\|}{d_0}\right) - \delta_i \right)^2 \tag{3.3}$$

The first step is to verify if the problem is convex or non-convex in order to choose an appropriate strategy to solve it. In this case, the estimator is non-convex due to the norm term in (3.3). In conclusion, the ML estimator in (3.3), even though asymptotically efficient, is non-convex and therefore difficult to tackle directly. Resorting to grid search algorithms could be exhaustive and iterative approaches, such as gradient descent, could produce poor results due to the non-convexity of the problem. Another way to tackle this problem could be through (tight) approximation of (3.3) by another estimator (e.g., second order cone programming (SOCP), semidefinite programming (SDP), etc.). However, this alternative would increase the computational complexity of such a solution. In a later section, a geometric approach based on a voting scheme to estimate the target node location is introduced instead. Afterwards, the location estimate is exploited in order to detect the attackers in the network.

2. *Coordinated Attack*

In coordinated attacks the malicious anchors communicate with each other to agree with a (false) location for the target. The idea is to make the network believe that target is at a different location

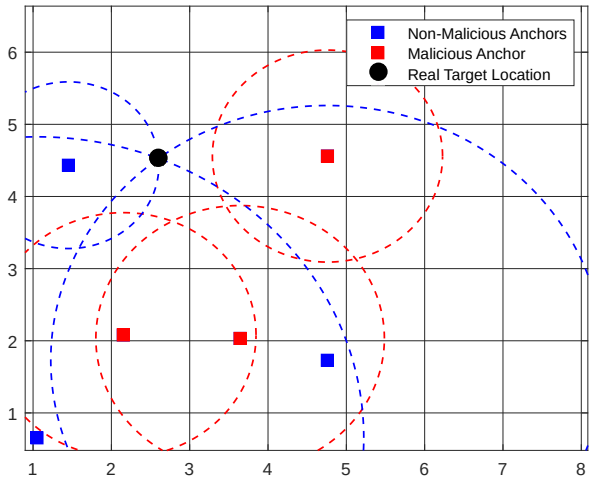
(ideally for attackers, *far* away from the real one) than it actually is. Similar to [25], the coordinated attack can be modeled as

$$p_{i,k} = \begin{cases} p_0 - 10\gamma \log_{10}\left(\dfrac{\|\boldsymbol{x} - \boldsymbol{a}_i\|}{d_0}\right) + n_{i,k} & \text{if the } i\text{-th anchor is honest,} \\ p_0 - 10\gamma \log_{10}\left(\dfrac{\|\boldsymbol{x}_{att} - \boldsymbol{a}_i\|}{d_0}\right) + n_{i,k} & \text{if the } i\text{-th anchor is malicious,} \end{cases} \tag{3.4}$$

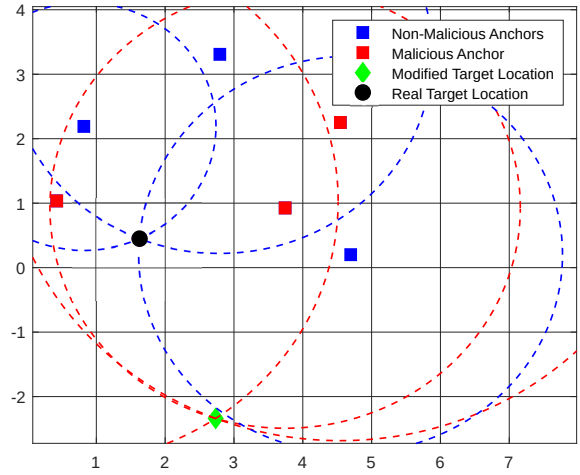
where $\boldsymbol{x}_{att}$ is the location the malicious nodes agree to perform the attack. The distance between the target's true location and the fabricated one is scaled by a factor of $\frac{\|\boldsymbol{x}_{att} - \boldsymbol{a}_i\|}{\|\boldsymbol{x} - \boldsymbol{a}_i\|}$.

Figure 3.1(b) shows an example of a coordinated attack, where three malicious anchor nodes (represented by a red square) attempt to make the network think that the position of the target is the one represented by a green diamond ($\boldsymbol{x}_{att}$) instead of the real target position represented by a black circle ($\boldsymbol{x}$). Moreover, $\boldsymbol{x}_{att}$ is given by the intersection of the red dashed circles, which represent the erroneous distance measurements, whereas the blue dashed circle represent the distance measurements between honest anchors and the real target, $\boldsymbol{x}$.

Similarly to the uncoordinated attack scenario, the ML estimator obtained for the coordinated attack scenario is non-convex and therefore cannot be tackled directly.



(a) Uncoordinated attack.



(b) Coordinated attack.

Figure 3.1: Types of spoofing attacks done by malicious anchors in a WSN for noise-free measurements.

The remainder of this chapter describes the derivation of the proposed algorithm to solve the problem of secure localization in WSNs. Therefore, it is organized in three sections, where

1. a preliminary part where points of interest are determined;
2. description of the proposed localization estimator based on a voting scheme;
3. proposed detection procedure to identify attackers.

3.2 Determining Points of Interest

At the beginning, all anchor nodes are treated as honest. Hence, we can define two sets of nodes, the set of malicious nodes, $\mathcal{M} = \varnothing$, and the set of honest nodes, $\mathcal{H} = \{i : 1 \leq i \leq N\}$. Afterwards, one can construct circles, c_i, centered at each anchor node with known locations and radii equivalent

to the distance estimate, $\widehat{d}_i = d_0 10^{\frac{p_0 - p_i}{10\gamma}}$, obtained from (3.1) as the median of K measurements for a specific anchor node (we refer the reader to see Figure 3.2(a)). The intersection points between all pairs of circles are known as points of interest and are used for the voting scheme. The intersection points between a pair of circles (given that they exist) can be calculated as follows

$$\boldsymbol{q}'_{ij} = \boldsymbol{q}_0 + \boldsymbol{t} \text{ and } \boldsymbol{q}''_{ij} = \boldsymbol{q}_0 - \boldsymbol{t}, \text{ for } i = 1, ..., N-1,\ j = i+1, ..., N, \tag{3.5}$$

where

$$\boldsymbol{q}_0 = (\boldsymbol{a_j} - \boldsymbol{a_i}) \frac{\widehat{d_i}^{\,2} - \widehat{d_j}^{\,2}}{2\|\boldsymbol{a_j} - \boldsymbol{a_i}\|^2} + \frac{\boldsymbol{a_i} + \boldsymbol{a_j}}{2},$$

and

$$\boldsymbol{t} = \frac{\sqrt{u}}{2\|\boldsymbol{a_j} - \boldsymbol{a_i}\|^2} \boldsymbol{T}(\boldsymbol{a_j} - \boldsymbol{a_i}) \ , \boldsymbol{T} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix},$$

with

$$u = \left((\widehat{d}_i + \widehat{d}_j)^2 - \|\boldsymbol{a_j} - \boldsymbol{a_i}\|^2\right) \left(\|\boldsymbol{a_j} - \boldsymbol{a_i}\|^2 - (\widehat{d}_j - \widehat{d}_i)^2\right),$$

according to Figure 3.2. However, due to the presence of noise and possibly malicious nodes, a pair of circles might not intersect (we refer the reader to see Figure 3.2(b)). In this case, we define the tuple set of anchor nodes without intersection as $\mathcal{C} = \{(i,j) : c_i \cap c_j = \varnothing\}$, where the notation $c_i \cap c_j = \varnothing$ is used to denote that the circles corresponding to the i-th and j-th anchor nodes do not intersect. In that case, one can draw a line that passes through the respective pair of anchor nodes, and compute the intersection points between the circles and the drawn line as follows

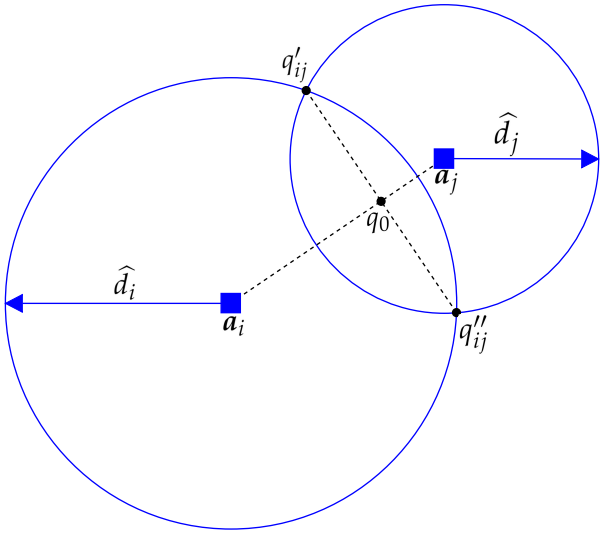
$$\begin{aligned}
\boldsymbol{q}_{i,1} &= \boldsymbol{s}_0 + \left[(\boldsymbol{a}_i - \boldsymbol{s}_0)^\intercal \widehat{\boldsymbol{b}} + \sqrt{[(\boldsymbol{a}_i - \boldsymbol{s}_0)^\intercal \widehat{\boldsymbol{b}}]^2 - (\boldsymbol{s}_0^\intercal \boldsymbol{s}_0 + \boldsymbol{a}_i^\intercal \boldsymbol{a}_i - d_i - 2\boldsymbol{s}_0^\intercal \boldsymbol{a}_i)}\right], \\
\boldsymbol{q}_{i,2} &= \boldsymbol{s}_0 + \left[(\boldsymbol{a}_i - \boldsymbol{s}_0)^\intercal \widehat{\boldsymbol{b}} - \sqrt{[(\boldsymbol{a}_i - \boldsymbol{s}_0)^\intercal \widehat{\boldsymbol{b}}]^2 - (\boldsymbol{s}_0^\intercal \boldsymbol{s}_0 + \boldsymbol{a}_i^\intercal \boldsymbol{a}_i - d_i - 2\boldsymbol{s}_0^\intercal \boldsymbol{a}_i)}\right], \\
\boldsymbol{q}_{j,1} &= \boldsymbol{s}_0 + \left[(\boldsymbol{a}_j - \boldsymbol{s}_0)^\intercal \widehat{\boldsymbol{b}} + \sqrt{[(\boldsymbol{a}_j - \boldsymbol{s}_0)^\intercal \widehat{\boldsymbol{b}}]^2 - (\boldsymbol{s}_0^\intercal \boldsymbol{s}_0 + \boldsymbol{a}_j^\intercal \boldsymbol{a}_j - d_j - 2\boldsymbol{s}_0^\intercal \boldsymbol{a}_j)}\right], \\
\boldsymbol{q}_{j,2} &= \boldsymbol{s}_0 + \left[(\boldsymbol{a}_j - \boldsymbol{s}_0)^\intercal \widehat{\boldsymbol{b}} - \sqrt{[(\boldsymbol{a}_j - \boldsymbol{s}_0)^\intercal \widehat{\boldsymbol{b}}]^2 - (\boldsymbol{s}_0^\intercal \boldsymbol{s}_0 + \boldsymbol{a}_j^\intercal \boldsymbol{a}_j - d_j - 2\boldsymbol{s}_0^\intercal \boldsymbol{a}_j)}\right],
\end{aligned}$$

where $\boldsymbol{s}_0 = \frac{\boldsymbol{a}_i + \boldsymbol{a}_j}{2}$ is the position vector of the line, and $\widehat{\boldsymbol{b}} = \frac{(\boldsymbol{a}_i - \boldsymbol{a}_j)}{\|\boldsymbol{a}_i - \boldsymbol{a}_j\|}$ is the unit vector that describes the line's direction. Afterwards, the intersection points to be used in the voting-scheme, $\boldsymbol{q}'_{ij}$, $\boldsymbol{q}''_{ij}$, are obtained as

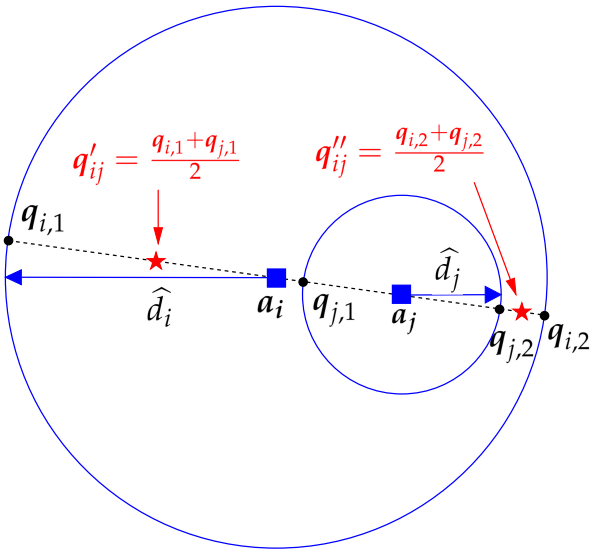
$$\begin{aligned}
\boldsymbol{q}'_{ij} &= \frac{\boldsymbol{q}_{i,1} + \boldsymbol{q}_{j,1}}{2} \\
\boldsymbol{q}''_{ij} &= \frac{\boldsymbol{q}_{i,2} + \boldsymbol{q}_{j,2}}{2}
\end{aligned} \tag{3.6}$$

This idea has been implemented in [24], and the reasoning behind it is that when a pair of anchor

nodes are attack-free, the intersection points would lay in the vicinity of the line, making these forged intersection points a reasonable approximation of the real ones.



(a) True intersections between a pair of anchor nodes.



(b) Forged intersections between a pair of anchor nodes.

Figure 3.2: Illustration of the possible scenarios in finding circles intersections.

3.3 The Proposed Voting-based Scheme for Target Localization

The voting scheme is a process to cluster and assign votes to some objects of interest based on some criterion (for instance, their physical proximity). The main idea is to assign a value (vote) to each intersection point in order to find the most trustworthy set of points. For the sake of simplicity, let us define the matrix $\boldsymbol{Q} = [\boldsymbol{q}'_{ij}\boldsymbol{q}''_{ij}] \in \mathbb{R}^{2\times\binom{N}{2}}$ which contains the intersection points obtained for all combinations of anchor nodes, and the vector $\boldsymbol{Q}_g \in \mathbb{R}^2$ as the g-th column of $\boldsymbol{Q}$. This process iterates all pairs of anchor nodes, and for each pair a hyperplane, $H_{ij} = \{g : \widehat{\boldsymbol{b}}^\mathsf{T}\boldsymbol{Q}_g = \widehat{\boldsymbol{b}}^\mathsf{T}\boldsymbol{s}_0\}$ is computed, which divides the problem space into two half spaces. The intersection points are assigned to the upper half space,

$H_{ij}^{(u)} = \{g : \widehat{\boldsymbol{b}}^{\intercal}\boldsymbol{Q}_g > \widehat{\boldsymbol{b}}^{\intercal}\boldsymbol{s}_0\}$, or to the lower half space, $H_{ij}^{(l)} = \{g : \widehat{\boldsymbol{b}}^{\intercal}\boldsymbol{Q}_g < \widehat{\boldsymbol{b}}^{\intercal}\boldsymbol{s}_0\}$, according to their physical location with respect to the hyperplane. Next, we build clusters composed of $N-1$ elements that are physically the closest to each others in each half space ($C_{ij}^{(u)} \subseteq H_{ij}^{(u)}$ as the upper cluster and $C_{ij}^{(l)} \subseteq H_{ij}^{(l)}$ as the lower cluster). Lastly, votes, v_h, are assigned to the points that belong to a cluster (if these exist), based on their distance to the hyperplane, see Figure 3.3. All things considered, the vote, for the h-th intersection point, given the hyperplane H_{ij}, is calculated as

$$v_h = v_h + w_h \frac{proj_{H_{ij}}(\boldsymbol{Q}_h)}{\sum_{h:h\in C_{ij}^{(u)}} proj_{H_{ij}}(\boldsymbol{Q}_h)} + w_h \frac{proj_{H_{ij}}(\boldsymbol{Q}_h)}{\sum_{h:h\in C_{ij}^{(l)}} proj_{H_{ij}}(\boldsymbol{Q}_h)},\ h \in C_{ij}^{(u)} \cup C_{ij}^{(l)} \tag{3.7}$$

where

$$w_h = \begin{cases} \frac{\widehat{d}_j}{\widehat{d}_i+\widehat{d}_j}, & \text{if } h \in H_{ij}^{(u)} \\ \frac{\widehat{d}_i}{\widehat{d}_i+\widehat{d}_j}, & \text{if } h \in H_{ij}^{(l)} \end{cases},$$

$$proj_{H_{ij}}(\boldsymbol{Q}_h) = \|\boldsymbol{Q}_h - (\boldsymbol{e}\boldsymbol{e}^{\intercal}\boldsymbol{Q}_{\boldsymbol{h}} + (\mathbb{I}_2 - \boldsymbol{e}\boldsymbol{e}^{\intercal})\,\boldsymbol{s}_0)\,\|$$

with

$$\boldsymbol{e} = \boldsymbol{T}\widehat{\boldsymbol{b}},$$

and $\mathbb{I}_2$ is the identity matrix of order two, $proj_{H_{ij}}(\boldsymbol{Q}_h)$ denotes the distance of an intersection point, $\boldsymbol{Q}_h$, to it's respective projection on the hyperplane, and w_h is a weight based on the distances $\widehat{d}_i$ and $\widehat{d}_j$ which takes into account the anchor node and the point $\boldsymbol{Q}_h$. It is important to refer that the vector of votes, $\boldsymbol{v} = [\boldsymbol{v_h}]^{\intercal}$, is initialized as a zero column vector $\left(\boldsymbol{v} = \boldsymbol{0}_{1\times\binom{N}{2}}\right)$ and each entry is incremented according to (3.7).

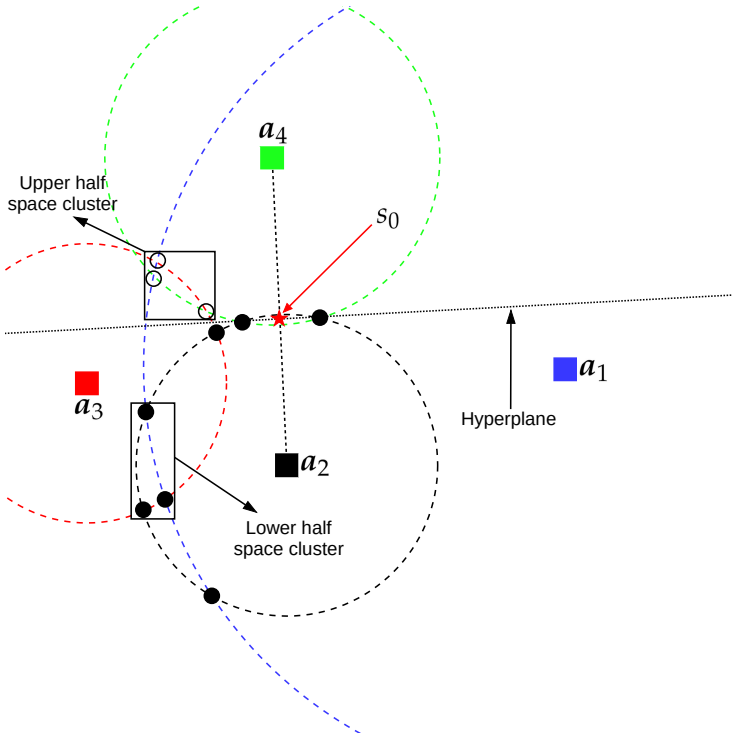


Figure 3.3: Illustration of the voting process between two anchor nodes when $N = 4$.

An estimate of the target's location can be obtained by re-ordering the vote vector in a descending fashion, $\tilde{\boldsymbol{v}} = [\tilde{v}_h]$, such that $\tilde{v}_1 \geq \tilde{v}_2 \geq \ldots \geq \tilde{v}_{\binom{N}{2}}$. The first $N-1$ votes correspond to the most trustworthy points; hence, the estimate is obtained by applying the WCM principle for the $N-1$ (normalized) vote values as

$$\widehat{\boldsymbol{x}} = \sum_{h=1}^{N-1} \tilde{w}_h \boldsymbol{Q}_h \tag{3.8}$$

with

$$\tilde{w}_h = \frac{\tilde{v}_h}{\sum_{h=1}^{N-1} \tilde{v}_h}$$

3.4 Attack Detection

Considering (3.2), in the case of malicious anchor, it is intuitive that the malicious attack would shift the probability distribution according to the attack intensity. This shift of the distribution could cause that the mean of the random variable falls outside of a certain confidence interval. In other words, one could take advantage of it to detect attackers. However, since we do not know the attack intensity, we can estimate it by exploiting the estimated target location (obtained from (3.7)) based on the ML principle, as well as the noise standard deviation and the expected (honest) RSS value for the estimated target location (refer Appendix A for the derivation).

$$\widehat{\delta}_i = \frac{\sum_{k=1}^{K} p_0 - p_{i,k} - 10\gamma \log_{10} \|\widehat{\boldsymbol{x}} - \boldsymbol{a_i}\|}{K}, \tag{3.9}$$

$$\widehat{\sigma} = \frac{1}{N} \sum_{i=1}^{N} \sqrt{\frac{1}{(K-1)} \sum_{k=1}^{K} \left(p_{i,k} - p_0 + 10\gamma \log_{10} \|\widehat{\boldsymbol{x}} - \boldsymbol{a_i}\| - \widehat{\delta}_i \right)^2},$$

$$\widehat{p}_i = p_0 - 10\gamma \log_{10} \|\widehat{\boldsymbol{x}} - \boldsymbol{a_i}\|. \tag{3.10}$$

From statistics, for a normal distribution, we know that 68% of the data fall within one standard deviation of the mean. Therefore, if the measured RSS from (3.1) lays outside of the confidence interval $[\widehat{p}_i - \widehat{\sigma}, \widehat{p}_i + \widehat{\sigma}]$, the respective anchor node is categorized as malicious; hence, the set of malicious nodes becomes $\mathcal{M} = \{i : \widehat{p}_i + \widehat{\sigma} < p_i < \widehat{p}_i - \widehat{\sigma}\}$. Figure 3.4 illustrates how the detection scheme works. The blue curve represents the distribution of an honest anchor, while the orange curve represents the distribution of a malicious one, shifted by the attack intensity, δ. The gray area represents the trust region defined by the threshold at $-\delta$ and δ, hence, any measurement that falls inside the threshold is deemed an honest measurement, while a measurement that falls outside the threshold is categorized as a measurement from a malicious anchor.

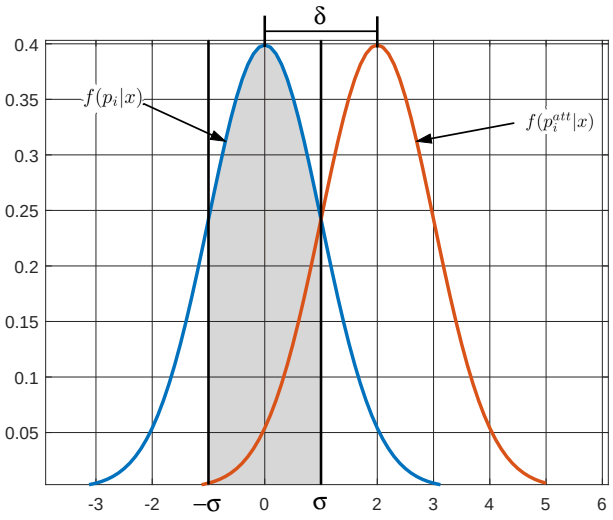


Figure 3.4: Illustration of the detection scheme employed.

It is important to note that, in contrast to [23–25], the malicious node is possibly exploited in the localization process. This can be advantageous when the attack intensity is not high compared to noise power.

Finally, to conclude this chapter, a pseudo-code for the proposed method is presented in Algorithm 1.

Algorithm 1 Pseudo Code for the Proposed VS algorithm

Inputs:
$\boldsymbol{a}_i, p_{i,k}, i = 1, ..., N, k = 1, ..., K$
Initialize:
$\mathcal{H} = \{i : 1 \leq i \leq N\}, \mathcal{M} = \varnothing$
// Calculate of forge intersection points
$\boldsymbol{q}_{ij} \leftarrow$ (3.5) or (3.6)
$\boldsymbol{q}_{ij} \leftarrow$ (3.5) or (3.6)
//Voting-scheme
for i := 1 to N-1 **do**
  for j := i+1 to N **do**
    // Compute hyperplane
    $H_{ij} = \{g : \widehat{\boldsymbol{b}}^{\mathsf{T}} \boldsymbol{Q}_h = \widehat{\boldsymbol{b}}^{\mathsf{T}} \boldsymbol{s}_0\}$
    // Classify intersection points by halfspace
    $H_{ij}^{(u)} = \{g : \widehat{\boldsymbol{b}}^{\mathsf{T}} \boldsymbol{Q}_g > \widehat{\boldsymbol{b}}^{\mathsf{T}} \boldsymbol{s}_0\}$
    $H_{ij}^{(l)} = \{g : \widehat{\boldsymbol{b}}^{\mathsf{T}} \boldsymbol{Q}_g < \widehat{\boldsymbol{b}}^{\mathsf{T}} \boldsymbol{s}_0\}$
    // Calculate votes for each point
    $v_h \leftarrow$ (3.7)
  end for
end for
// Estimate target's location estimate
$\widehat{\boldsymbol{x}} \leftarrow$ (3.8)
for i := i to N **do**
  if $\widehat{p}_i + \widehat{\sigma} < p_i < \widehat{p}_i - \widehat{\sigma}$ **then**
    $\mathcal{M} \leftarrow \mathcal{M} \cup \{i\}$
    $\mathcal{H} \leftarrow \mathcal{H} \setminus \{i\}$
  end if
end for

Chapter 4

Numerical Results

This chapter presents a series of numerical results in order to assess the performance of the proposed solution. It presents analysis based on computational complexity, localization accuracy and success in detecting malicious attackers. Thus, it is organized correspondingly.

4.1 Complexity Analysis

The complexity analysis is highly relevant for the applicability of the algorithm, especially in real-time scenarios. Given that B_{max} is the maximum number of iterations for the GTRS algorithm and B_{ADMM} is the number of iteration on average for the ADMM algorithm to converge, Table 4.1 summarizes the worst-case computational complexity together with the average running time of the considered methods. The latter evaluation was performed with 5000 Monte Carlo (MC) simulations on a machine with the following characteristics: CPU: Intel(R) Core(TM) i5-8250U CPU @ 1.60 GHz, RAM: 8 GB, OS: Windows 10, running MATLAB R2021a. From the table, one can note that the considered algorithms have linear complexity in N. However, the solution proposed in [24] and the LN-1 algorithm proposed in [25] have slightly higher computational burden, since they are executed in an iterative fashion. It is worth mentioning, that the proposed solution has a single position estimate, while the solution in [24] requires two estimation steps to reach its final solution. Moreover, it is noteworthy that all approaches in [24] and [25] require matrix operations (such as inversion and transpose), which have certain computational costs associated with them (e.g., $\mathcal{O}(m^3)$ is the cost of matrix inversion, where m represents the size of the matrix), unlike the proposed solution which does not require such matrix operations. However, although the operations in VS are computationally the least demanding, the proposed algorithm requires repeated actions (for each pair of anchors), which results in somewhat higher average running time than SWLS and LN-1. Moreover, the approach in [24] requires two iterations to obtain the final estimation, whereas the proposed solution solves the localization problem in a single iteration. Nevertheless, the table shows that all four algorithms are extremely fast, which is very important for real-time applications.

Algorithm	Complexity	Average Running Time (s)	Coordinated Attacks	Uncoordinated Attacks
VS in Chapter 3	$\mathcal{O}(N)$	0.0139	✓	✓
SWLS in [25]	$\mathcal{O}(N)$	0.0015	✗	✓
LN-1 in [25]	$\mathcal{O}(B_{ADMM}N)$	0.0063	✓	✓
GTRS in [24]	$\mathcal{O}(N \times B_{max})$	0.0254	✓	✓

Table 4.1: Worst-case computational complexity and average running time of the considered methods.

4.2 Localization and Attacker Detection Analysis

This section presents an assessment in localization accuracy and success of attacker detection for the considered methods in Table 4.1 through MC simulations. The simulations disclose the results for N randomly deployed anchor nodes and one target, within a two-dimensional area of 25×25 m^2 for scenarios with a single and multiple malicious nodes. Furthermore, all anchor nodes were randomly placed $N_D = 800$ times and, for each positioning settings, each anchor node (or each combination of anchor nodes for the scenario of multiple attacks) was considered malicious $N_A = 50$ times.

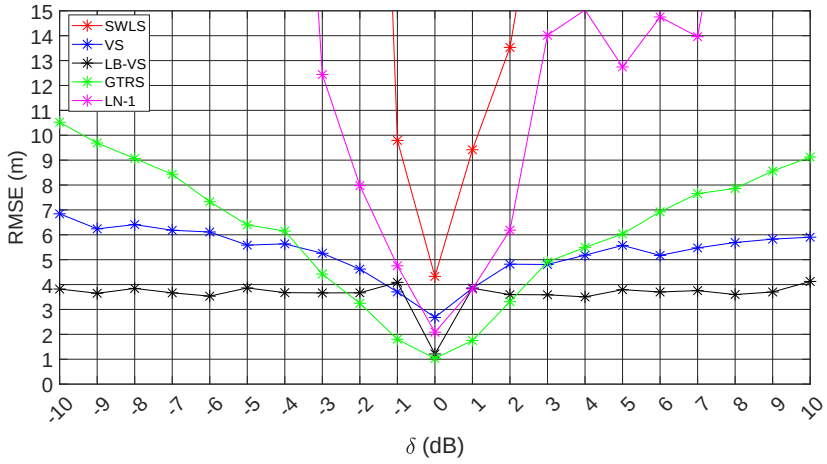
The RSS measurements were obtained through (3.1), where the transmit power at the target node is set to $p_0 = -10$ dBm, the PLE is fixed at $\gamma = 3$, and $K = 10$. The simulations for uncoordinated attacks considered single and two malicious nodes and the simulations for coordinated attacks considered two malicious nodes. The main metric used to assess the localization performance is the root mean squared error (RMSE), RMSE $= \sqrt{\sum_{i=1}^{Mc} \frac{\|\boldsymbol{x}_i - \widehat{\boldsymbol{x}}_i\|^2}{MC}}$ (m), where x_i and $\widehat{x}_i$ are, respectively, the true target location and the estimated target location in the i-th MC run, with $MC = N_D \times N_A \times N$ (single malicious node) or $MC = N_D \times N_A \times \binom{N}{2}$ (two malicious nodes). It is worth reminding the reader that SWLS requires tuning the detection threshold by studying an empirical parameter, ζ. Based on our experience, the best results for SWLS were obtained for $\zeta = 1.6$, and all results presented for SLWS in the remainder of this section are obtained by setting ζ to this value. The LN-1 algorithm does not detect attackers, hence it is only used as comparison for localization accuracy.

4.2.1 Uncoordinated Attacks

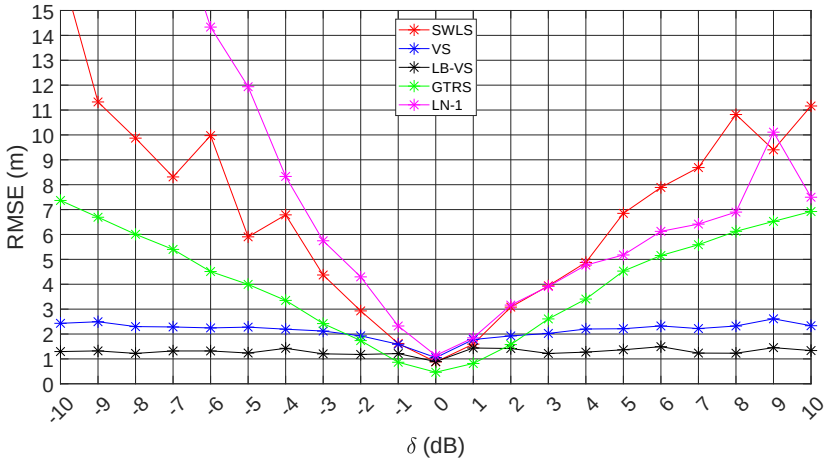
Single Malicious Node Scenario

Figure 4.1 shows the RMSE (m) versus the attack intensity δ (dB) for different number of anchor nodes and $\sigma = 1$ dB. For the purpose of better understanding the performance of the proposed method, the lower bound, "LB-VS", on the performance of the proposed algorithm where the ideal detection is given to the method is also shown. As expected, the figure exhibits that as N increases, the RMSE decreases. It can be seen that, for large values of $|\delta|$, the proposed method outperforms the considered algorithms and, for small values, the GTRS solution presents better results. However, for $\delta \in [-3, 3]$ dB,

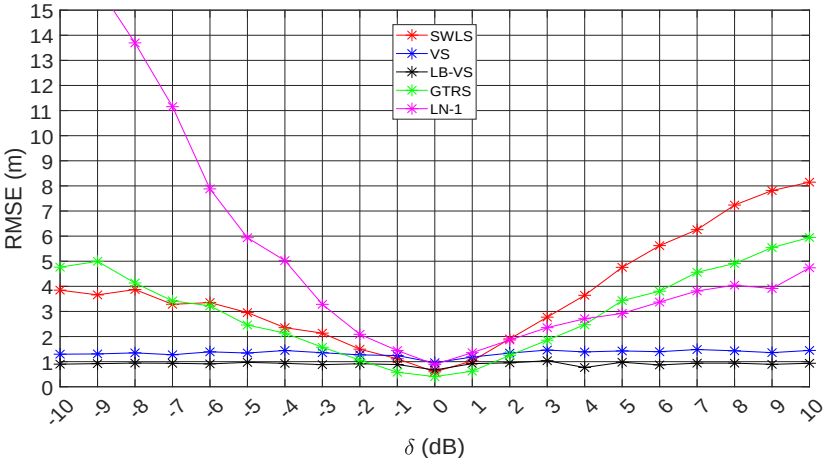
when $N = 4$ and $\sigma = 1$ dB, the maximum difference in RMSE between the VS and the GTRS is about 1.7 m in favor of GTRS, while for higher magnitudes of the attack, the maximum difference can reach 4.3 m in favor of the proposed method. Naturally, these differences become smaller when N increases, since all methods improve their performance. From the results, it is obvious that SWLS suffers significant performance loss for low N and only surpasses the proposed solution when $N = 6$ and $\delta \in [-1, 1]$ dB, where the difference is practically insignificant. The LN-1 solution similar performance to SWLS with slightly better results for $N = 4$, similar performance for $N = 5$ and $\delta > 0$ dB and shows significant performance loss for $\delta < 0$ dB, regardless of the number of anchors. In conclusion, the LN-1 algorithm does not surpass the proposed solution in any manner, and still presents great accuracy loss when $\delta < 0$ dB.



(a) $N = 4$



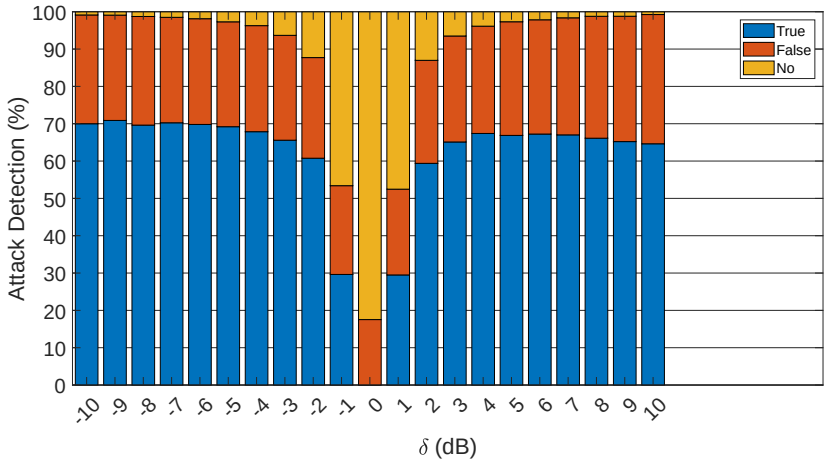
(b) $N = 5$



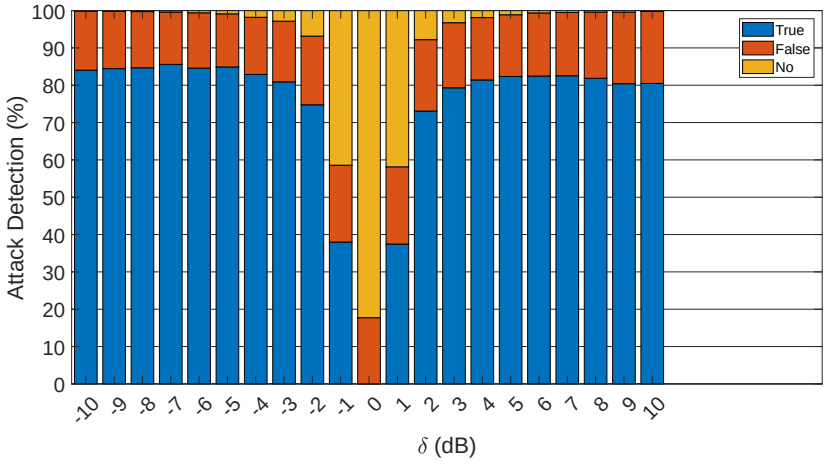
(c) $N = 6$

Figure 4.1: RMSE versus attack intensity δ (dB) for different number of anchor nodes, N, when $\gamma = 3$, $B = 25$ m, $\sigma = 1$ dB, with a single malicious node.

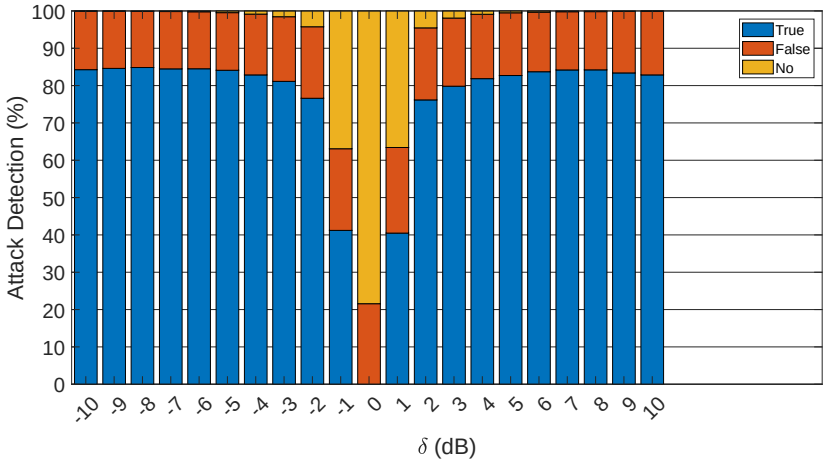
Figure 4.2 illustrates the rate of success, failure, and no detection of the proposed estimator in the previously mentioned setting. The figure presents large detection rates for large values of $|\delta|$. This is intuitive, since when $|\delta|$ is small, the presence of an attacker is concealed within the measurement noise, which makes it more difficult to detect it. On the other hand, when $|\delta|$ is large it becomes more difficult for the attacker to hide its presence. The figure also shows high rates of no detection for small values of $|\delta|$, which is mostly due to the noise measurements surpassing the attack intensity. However, considering Figure 4.1, this phenomenon does not pose a major threat to localization performance and the attack intensity can be treated as a slightly increased measurement noise. It is worth mentioning that, for the case where $\delta = 0$ dB (and $|\delta| \approx 0$), the desirable outcome is a high rate of no detection, since there are no malicious nodes in the network (or their attack intensity is low) and, ideally, all anchors would be used in the localization process. Finally, one can notice that detection performance betters with the increase of N, as anticipated, since the rate of honest/malicious nodes grows in this case.



(a) $N = 4$



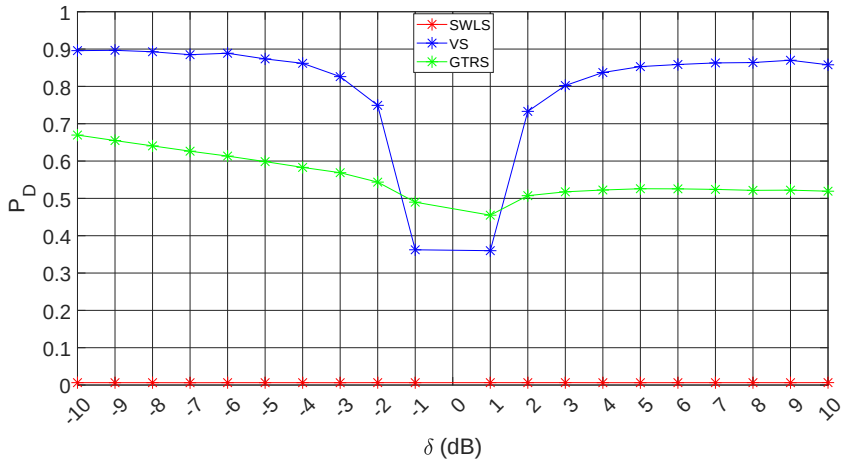
(b) $N = 5$



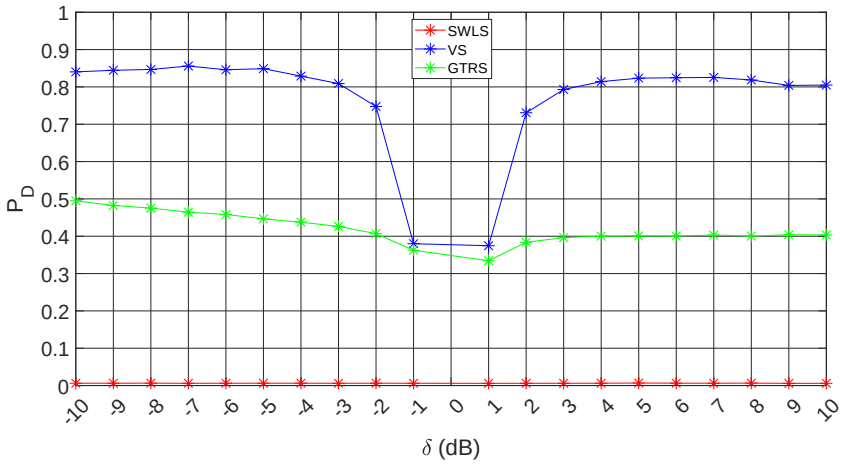
(c) $N = 6$

Figure 4.2: Attack detection (%) versus attack intensity δ (dB) for different number of anchor nodes, N, when $\gamma = 3$, $B = 25$ m, and $\sigma = 1$ dB, with a single malicious node.

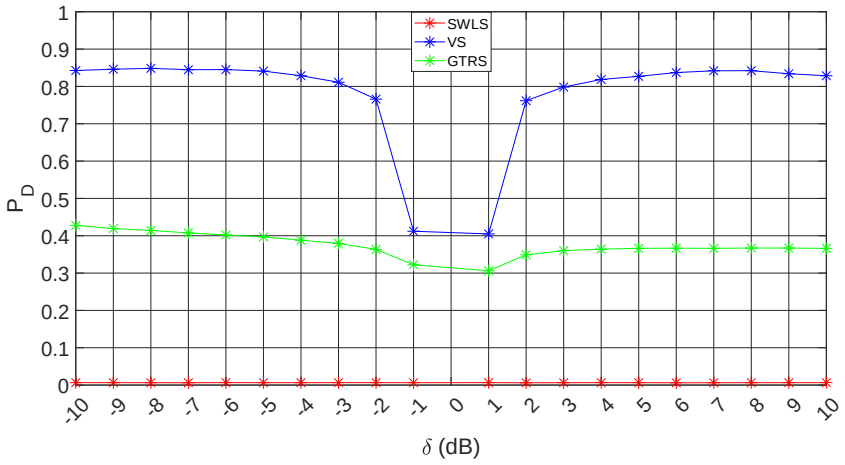
Figure 4.3 shows the probability of (successful) detection, P_D, against δ (dB) for different choices of N when $\sigma = 1$ dB. The results obtained by using the proposed scheme show superior detection rates over the considered solutions in all scenarios considered with the exception of $N = 4$, where GTRS outperforms the proposed solution for small magnitudes of attack intensity. Nevertheless, as stated earlier, this behavior could even be desirable for small magnitudes of the attack. The results obtained also show practically no detection for the SWLS solution, which is a direct consequence of the threshold tuning (otherwise, SWLS does not work in the considered settings due to low N).



(a) $N = 4$



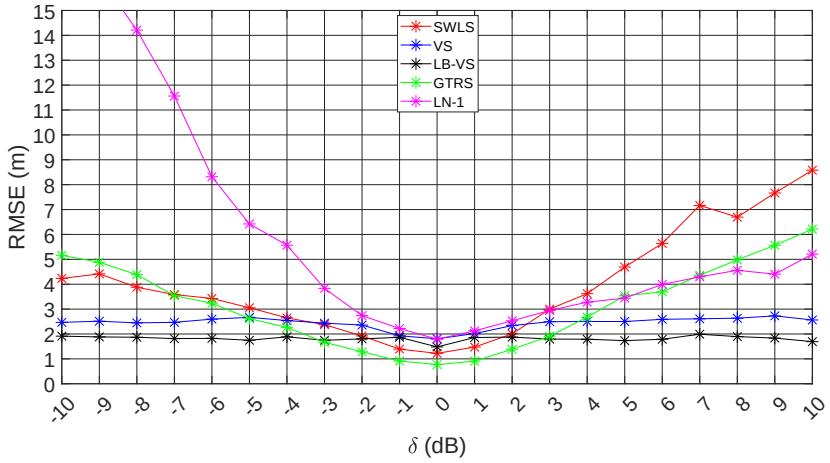
(b) $N = 5$



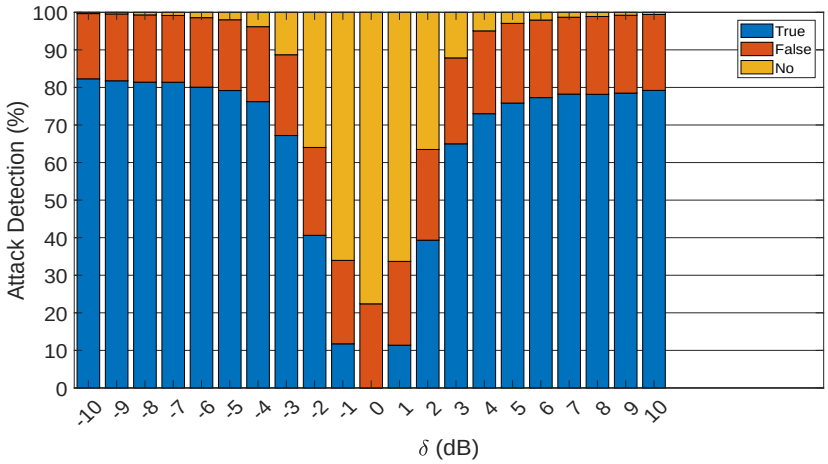
(c) $N = 6$

Figure 4.3: Probability of detection, P_D, versus attack intensity δ (dB) for different number of anchor nodes, N, when $\gamma = 3$, $B = 25$ m, and $\sigma = 1$ dB, with a single malicious node.

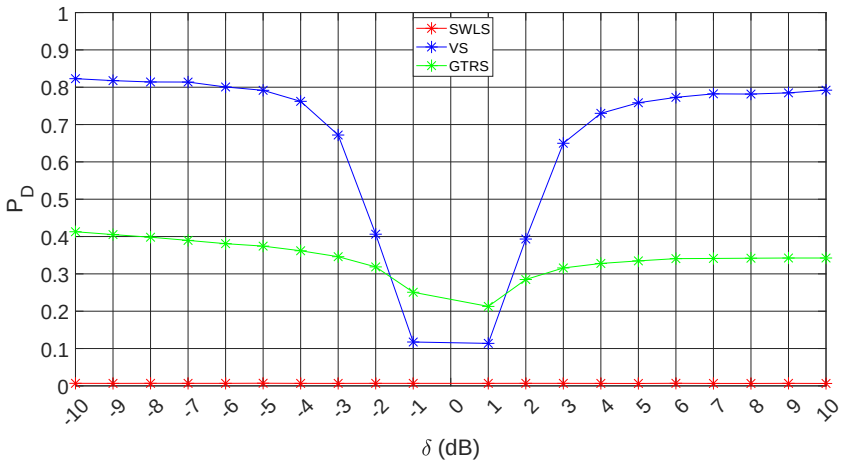
Figure 4.4(a) shows the RMSE (m) versus the attack intensity δ (dB) for $\sigma = 2$ (dB) and $N = 6$. Compared with Figure 4.1, the figure exhibits a deterioration in localization performance since the measured noise is increased. This phenomenon is intuitive due to a larger measurement noise which gives more chances to the attacker to conceal its presence. As we can see, the proposed solution exhibits slightly worse performance than the existing solution for small magnitudes of attack intensity, and outperforms the considered methods for large magnitudes. The figure also shows closer performance of the proposed solution in comparison with "LB-VS" (< 1 m of difference). It is concluded the same as in Figure 4.1, where, for low $|\delta|$, the difference in RMSE between the proposed method and the existing ones is about 1 m in favor of the existing solutions, where for larger magnitudes this difference can reach values of about 3.5 m in favor of the proposed solution. Figure 4.4(b) illustrates the rate of success, failure, and no detection for $\sigma = 2$ (dB), when $N = 6$. This figure shows that the proposed solution achieves at least 80% of detection success for high $|\delta|$. The figure also exhibits high no detection rates for low magnitudes of attack intensity, which, as previously mentioned, can be acceptable since the attack influence can be seen as noise in the measurements. Figure 4.4(c) shows the probability of successful detection, P_D, against δ (dB) for $\sigma = 2$ (dB) and $N = 6$. The result obtained by the proposed scheme show superior detection rates over the considered solutions with the exception of $\delta \in [-1, 1]$ dB, where GTRS outperforms the proposed solution. However, the cost of detection, for the GTRS solution, in the case of large magnitudes of attack intensity is far greater than the cost for the proposed solution in the case of small attack magnitudes.



(a) RMSE versus attack intensity δ (dB)



(b) Attack detection (%) versus attack intensity δ (dB)

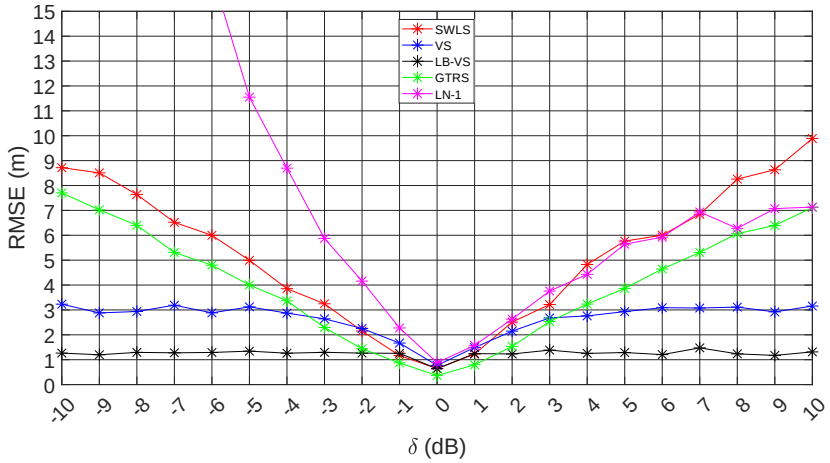


(c) Probability of detection, P_D versus attack intensity δ (dB)

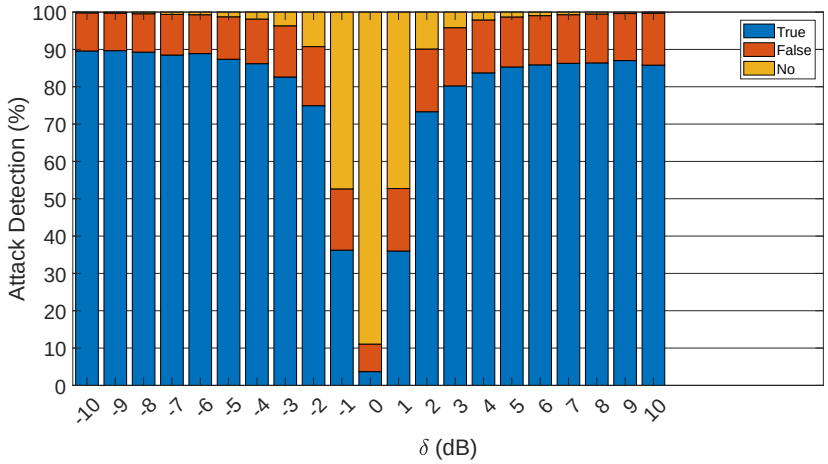
Figure 4.4: Proposed solution performance for $N = 6$, $\gamma = 3$, $B = 25$ m, and $\sigma = 2$ dB, with a single malicious node.

Multiple Malicous Nodes Scenario

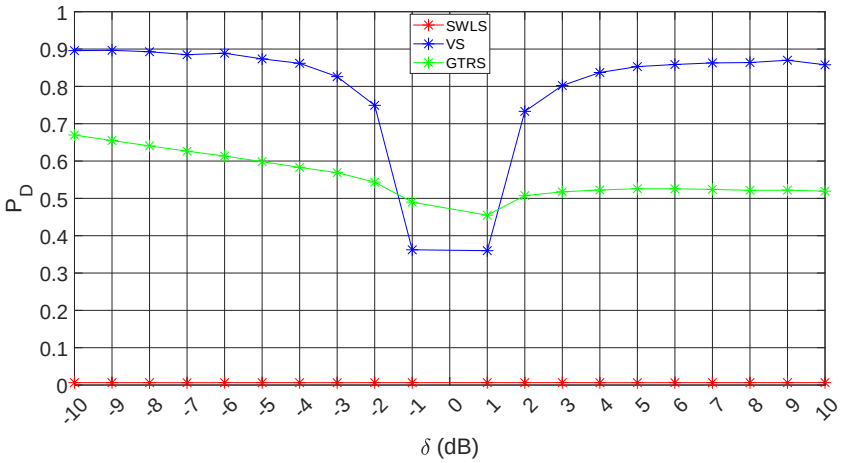
Figure 4.5(a) shows the RMSE (m) versus the attacker intensity δ (dB) for $\sigma = 1$ dB and $N = 6$. As expected, the RMSE values are slightly worse when the number of malicious nodes increases. The results obtained between the considered solutions are similar to the case where only one malicious node was admitted (i.e., the proposed solution outperforms the existing ones for larger magnitudes of attack intensity and underperforms for low magnitudes). However, it is important to notice that the difference in RMSE between the proposed solution and GTRS increases (about 75%) from one malicious node (the max difference is about 2.55 m) to two malicious nodes (the difference goes up to 4.47 m). Figure 4.5(b) illustrates the rate of success and failure of the proposed solution in the considered setting against δ (dB). The figure shows that with the increase of malicious nodes the proposed solution increases the rate of success, achieving at least 90% of detection success for high magnitudes of attack intensity, and decreases the rate of false detection for any value of δ. One can also notice an increase in rate of success in comparison to Figure 4.2. This is due to the fact that the algorithm might detect there is more than one malicious node, when in reality there is only one; hence, the number of false detection is higher for a single malicious node scenarios. Figure 4.5(c) presents P_D comparison of the existing methods in the considered setting. From this figure we can see that the probability of detection increases, significantly (around 30%) for GTRS. However, the proposed solution still displays a better performance over the magnitudes of attack intensity (except for $|\delta| = 1$ (dB)). Compared with the setting with a single attacker, one can notice that, even though P_D is somewhat higher in this case, every miss has larger cost in terms of RMSE in this scenario. Hence, although P_D grows in this setting (at the cost of reduced false detection), the localization performance does not improve in this case.



(a) RMSE versus attack intensity δ (dB)



(b) Attack detection (%) versus attack intensity δ (dB)



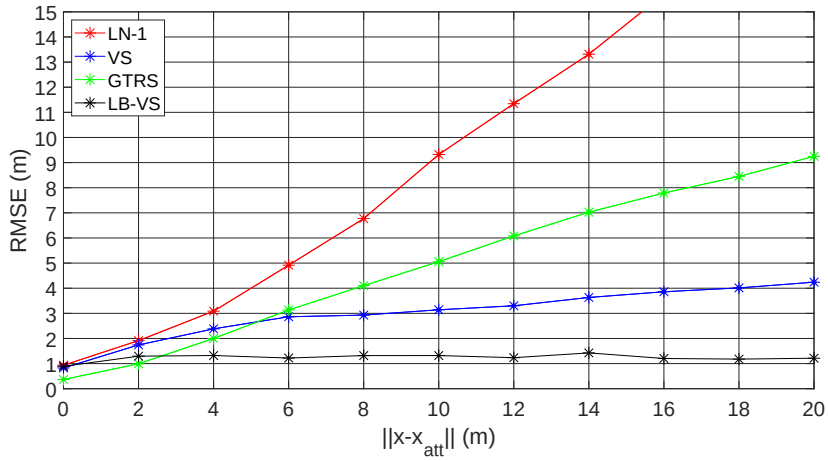
(c) Probability of detection, P_D versus attack intensity δ (dB)

Figure 4.5: Proposed solution performance for $N = 6$, $\gamma = 3$, $B = 25$ m, and $\sigma = 1$ dB, with two malicious nodes.

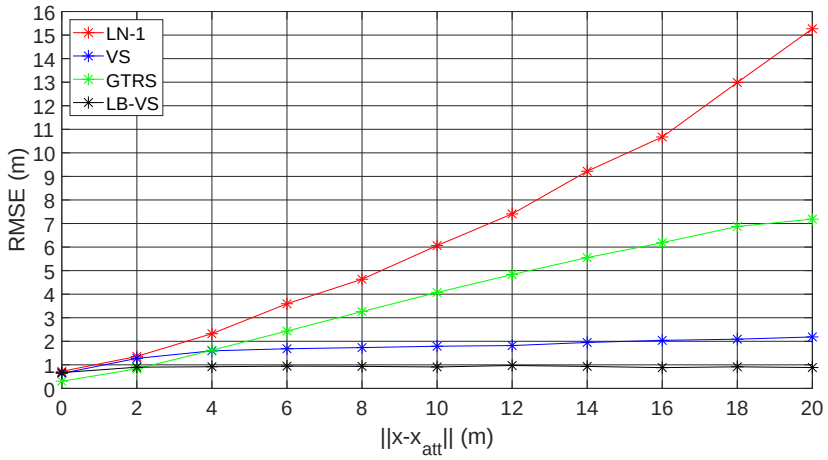
4.2.2 Coordinated Attacks

In the coordinated attack scenario the attack position, i.e., the position agreed by the malicious nodes, is obtained by choosing a random point in a circle centered at the true target and predefined radii, i.e., $\|\boldsymbol{x} - \boldsymbol{x}_{att}\|$. The simulations for coordinated attacks consider two malicious nodes. The simulation parameters are the same as in the uncoordinated scenario, i.e., the main metric to assess accuracy remains the RMSE and the number of MC simulations used is $MC = N_D \times N_A \times \binom{N}{2}$, where $N_D = 800$ and $N_A = 50$. Table 4.1 lists the methods considered for this setting.

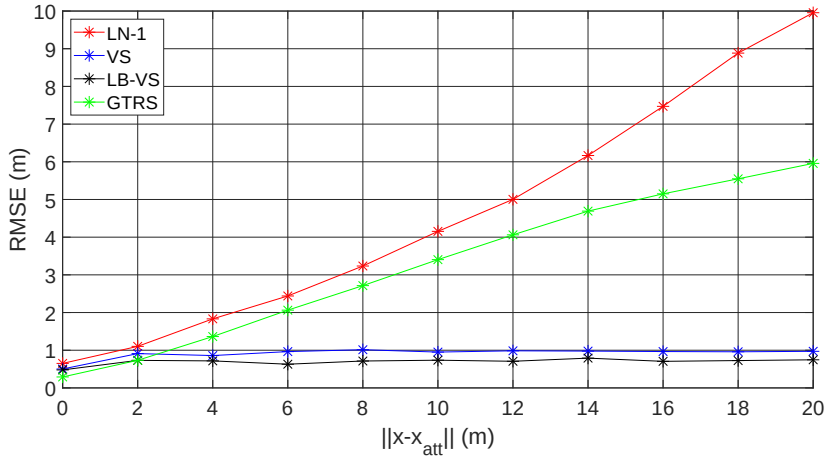
Figure 4.6 shows the RMSE (m) versus the distance between the target location ($\boldsymbol{x}$) and the attack location ($\boldsymbol{x}_{att}$) for different number of anchor nodes and $\sigma = 1$ dB. As expected, the figure exhibits that as N increases, the RMSE decreases. Intuitively, when the distance of attack, $\|\boldsymbol{x} - \boldsymbol{x}_{att}\|$, the RMSE also increases. It can be seen that the proposed method outperforms the considered algorithms except for small values of $\|\boldsymbol{x} - \boldsymbol{x}_{att}\|$, where the GTRS solution presents slightly better results. However, the advantage of the proposed solution for $\|\boldsymbol{x} - \boldsymbol{x}_{att}\| > 4$ m is far more significant than the one presented by the GTRS for small values of distance of attack. From the results, it is obvious that LN-1 suffers significant performance loss for high attack distance and only shows competitive results for small values of $\|\boldsymbol{x} - \boldsymbol{x}_{att}\|$. It is also worth to mention that the RMSE value of the proposed solution shows a slow increase and, in the case of $N = 8$, the error seems to converge to 1 m, while the considered solutions show a steeper increase in RMSE. Lastly, for $N = 8$ the VS solution shows closer performance in comparison with "LB-SV", with a difference of about 0.5 m.



(a) $N = 6$ and $\sigma = 1$ dB



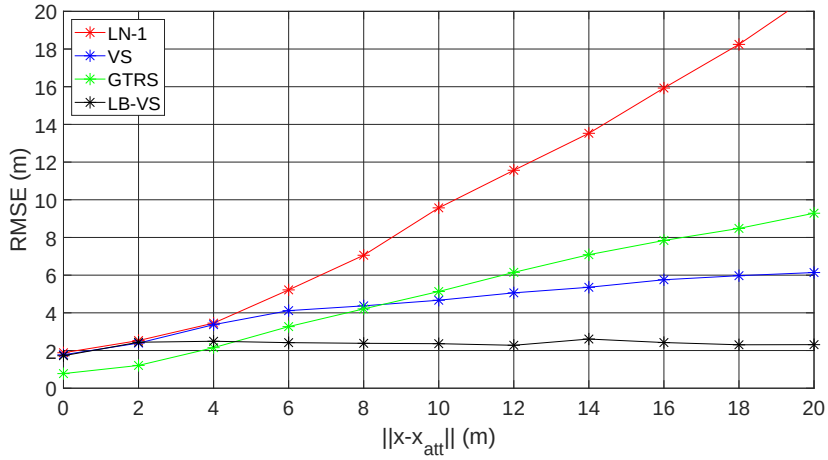
(b) $N = 7$ and $\sigma = 1$ dB



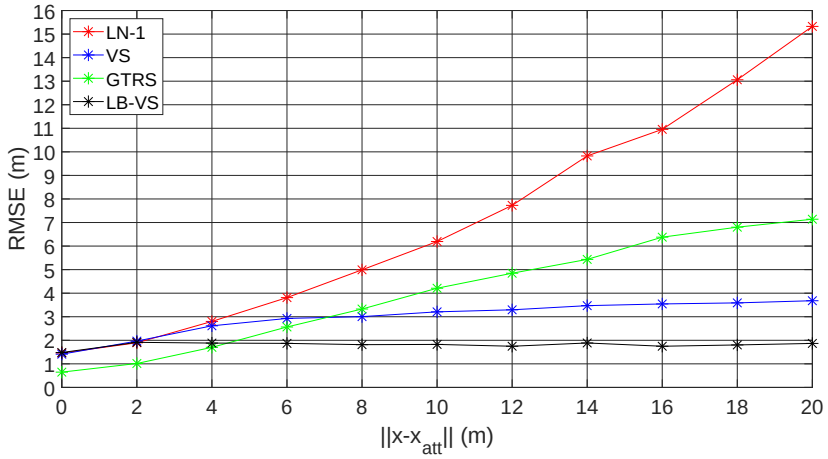
(c) $N = 8$ and $\sigma = 1$ dB

Figure 4.6: RMSE versus attack distance $\|\boldsymbol{x} - \boldsymbol{x}_{att}\|$ (m) for different number of anchor nodes, N, when $\gamma = 3$, $B = 25$ m, $\sigma = 1$ dB, with two coordinated malicious node.

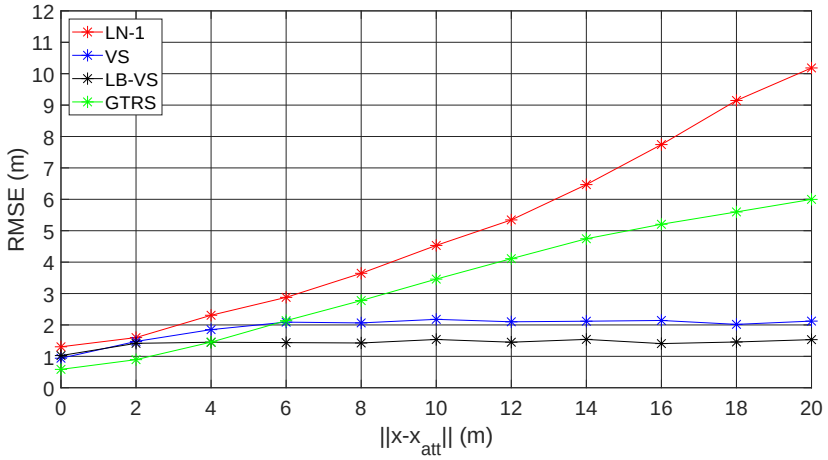
Figure 4.7 shows the RMSE (m) versus the distance between the target location($\boldsymbol{x}$) and the attack location ($\boldsymbol{x}_{att}$) for different number of anchor nodes and $\sigma = 2$ dB. As expected, in comparison with Figure 4.6, this figure exhibits a deterioration in localization performance since the measured noise is increased. It can be seen that the proposed method outperforms the considered algorithms except for $\|\boldsymbol{x} - \boldsymbol{x}_{att}\| < 6$ m, where the GTRS solution, once more, presents slightly better results. It is concluded the same as for Figure 4.6: the proposed solution shows far greater advantage for high values of attack distance in comparison with the considered methods, while the GTRS solution presents slightly better performance results for small values of attack distance and the LN-1 solution has severe performance loss for high values of attack distance.



(a) $N = 6$ and $\sigma = 2$ dB



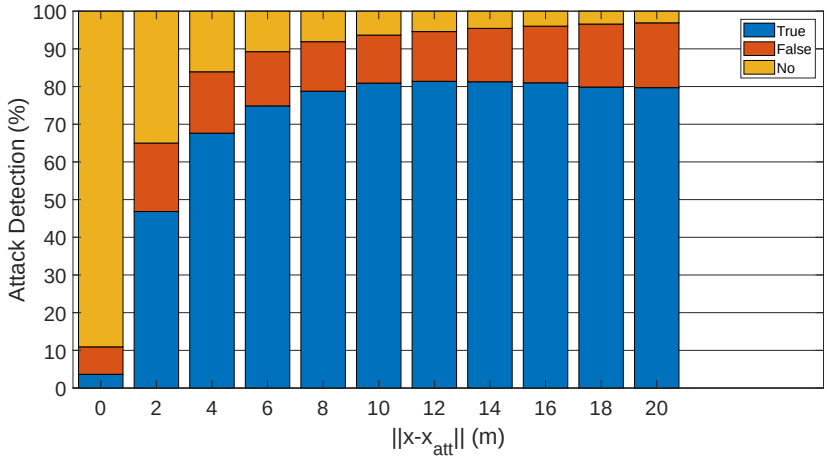
(b) $N = 7$ and $\sigma = 2$ dB



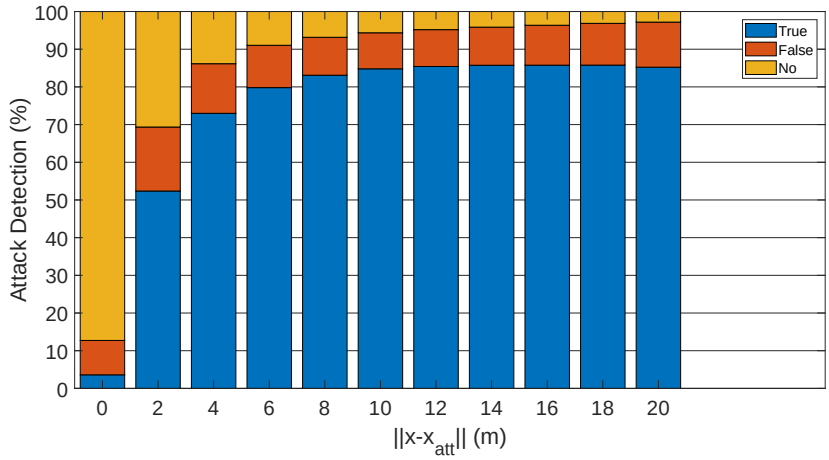
(c) $N = 8$ and $\sigma = 2$ dB

Figure 4.7: RMSE versus attack distance $\|\boldsymbol{x} - \boldsymbol{x}_{att}\|$ (m) for different number of anchor nodes, N, when $\gamma = 3$, $B = 25$ m, $\sigma = 2$ dB, with two coordinated malicious node.

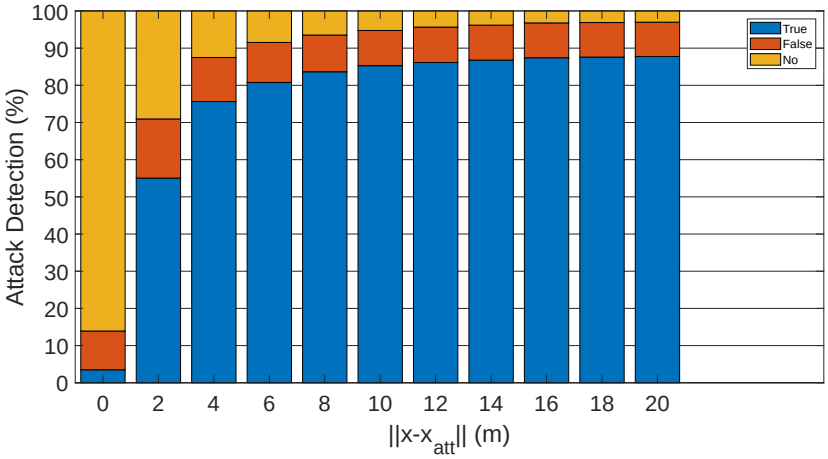
Figure 4.8 illustrates the rate of success, failure, and no detection of the proposed estimator for different number of anchor nodes and $\sigma = 1$ dB for two coordinated malicious nodes. The results show large detection rates (approximately 80% percent when $N = 6$ and 88% when $N = 8$) for large values of $\|\boldsymbol{x} - \boldsymbol{x}_{att}\|$. This is intuitive, since when $\|\boldsymbol{x} - \boldsymbol{x}_{att}\|$ is small, it is difficult to separate attack from measured noise, which makes it more difficult to detect it. In the same way, when $\|\boldsymbol{x} - \boldsymbol{x}_{att}\|$ is large, it becomes more difficult for the attacker to hide its presence. It is important to remember that LN-1 does not detect attackers, therefore cannot be used as comparison in detection assessments. In comparison with Figure 4.2, the results show slightly worse detection performance, which makes sense since cooperated attacks are more complex than the uncoordinated ones. However, considering Figure 4.6, this phenomenon does not pose a major threat to the localization performance of the proposed algorithm. It is worth mentioning that, for the case where $\|\boldsymbol{x} - \boldsymbol{x}_{att}\| = 0$ m the desirable outcome is a high rate of no detection, since there are no malicious nodes in the network and, ideally, all anchors would be used in the localization process. Finally, one can notice that detection performance betters with the increase of N, as anticipated, since the rate of honest/malicious nodes grows in this case.



(a) $N = 6$ and $\sigma = 1$ dB



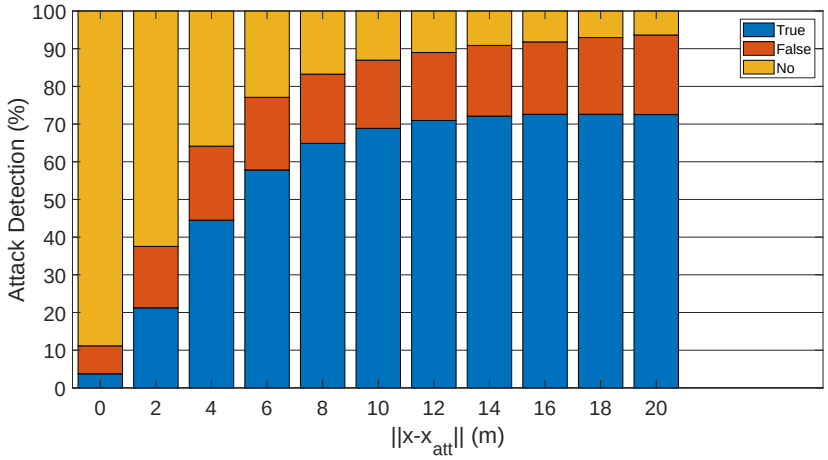
(b) $N = 7$ and $\sigma = 1$ dB



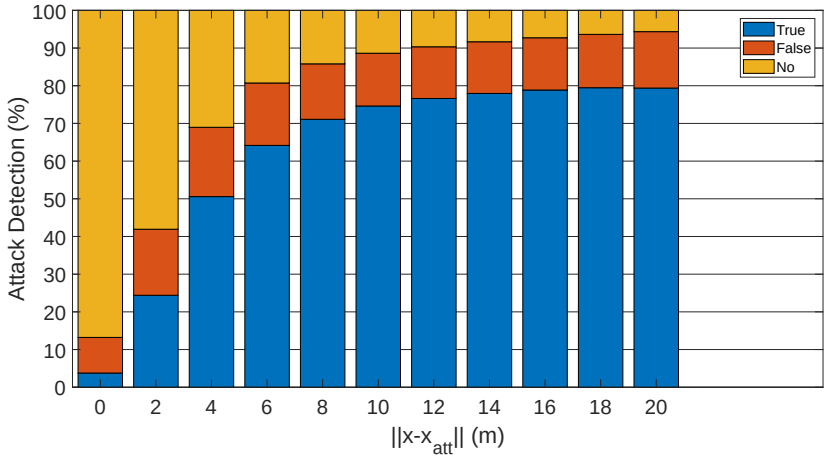
(c) $N = 8$ and $\sigma = 1$ dB

Figure 4.8: Attack detection (%) versus attack distance $\|\boldsymbol{x} - \boldsymbol{x}_{att}\|$ (m) for different number of anchor nodes, N, when $\gamma = 3$, $B = 25$ m, and $\sigma = 1$ dB, with two coordinated malicious node.

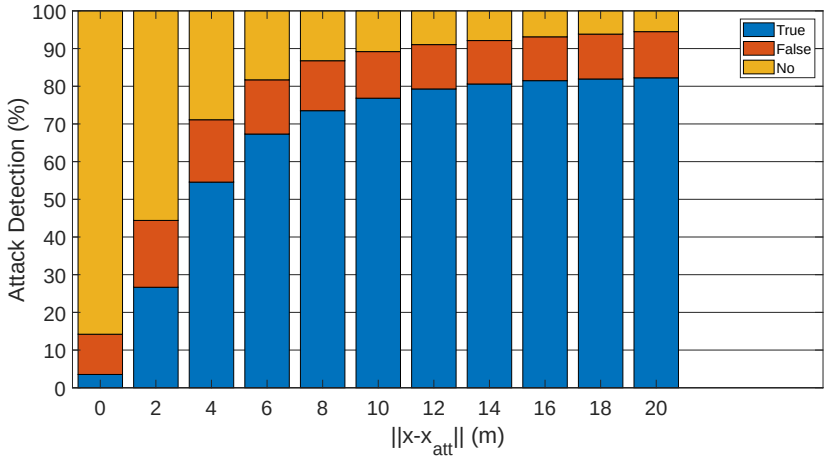
Figure 4.9 illustrates the rate of success, failure, and no detection of the proposed estimator for different number of anchor nodes and $\sigma = 2$ dB for two coordinated malicious nodes. The results show a degradation of detection rate, which is in line with our intuition since the attackers have a greater margin to hide their presence. In the case of $N = 6$, the proposed solution achieves a detection rate of 70% for high attack distances, when $N = 7$ the detection rate is about 78%, and when $N = 8$ the detection rate is about 83%. The results from this figure show a degradation of detection rate in comparison with 4.8. However, once more, considering the localization accuracy results presented in Figure 4.7, this phenomenon does not pose a major threat to the localization performance of the proposed solution.



(a) $N = 6$ and $\sigma = 2$ dB



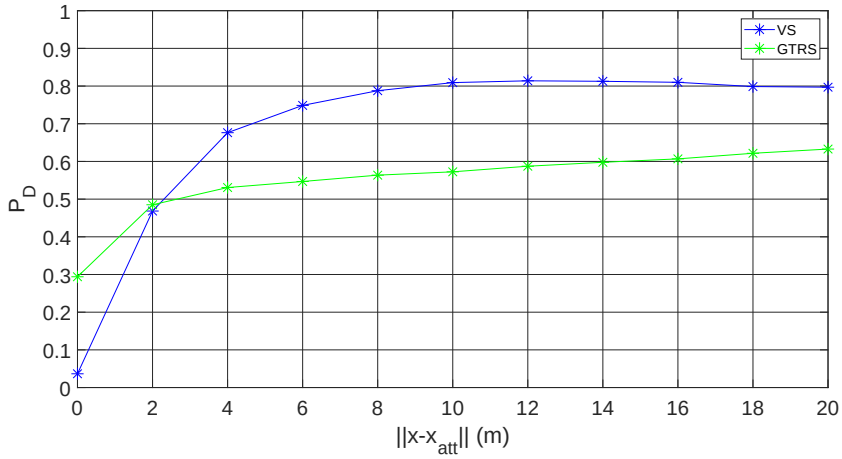
(b) $N = 7$ and $\sigma = 2$ dB



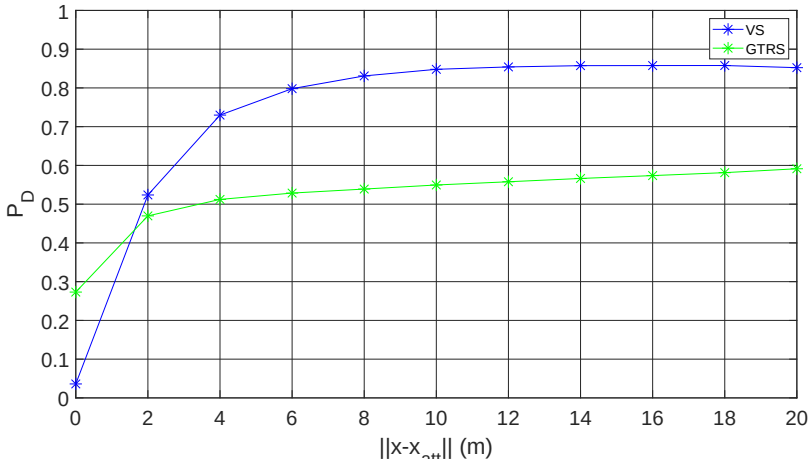
(c) $N = 8$ and $\sigma = 2$ dB

Figure 4.9: Attack detection (%) versus attack distance $\|\boldsymbol{x} - \boldsymbol{x}_{att}\|$ (m) for different number of anchor nodes, N, when $\gamma = 3$, $B = 25$ m, and $\sigma = 2$ dB, with two coordinated malicious node.

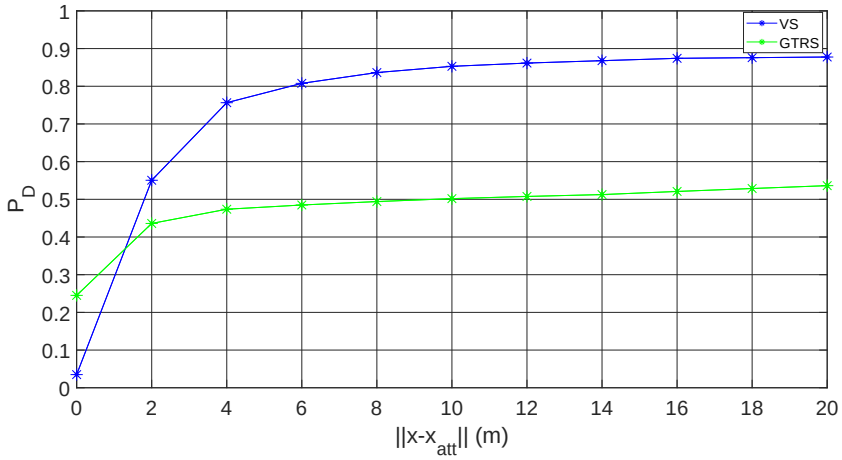
Figure 4.10 shows the probability of (successful) detection, P_D, against $\|\boldsymbol{x} - \boldsymbol{x}_{att}\|$ (m) for different choices of N when $\sigma = 1$ dB. The results obtained by using the proposed scheme show superior detection rates over the considered solution in all scenarios considered with the exception of very small $\|\boldsymbol{x} - \boldsymbol{x}_{att}\|$, where for GTRS outperforms the proposed solution.



(a) $N = 6$ and $\sigma = 1$ dB



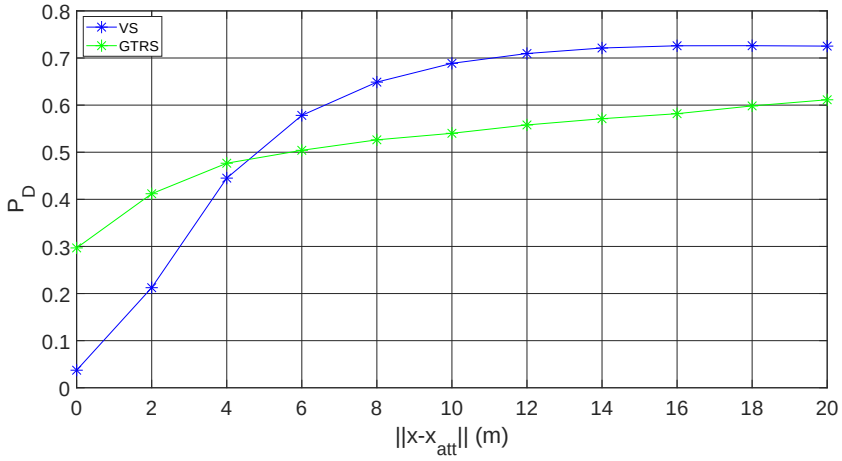
(b) $N = 7$ and $\sigma = 1$ dB



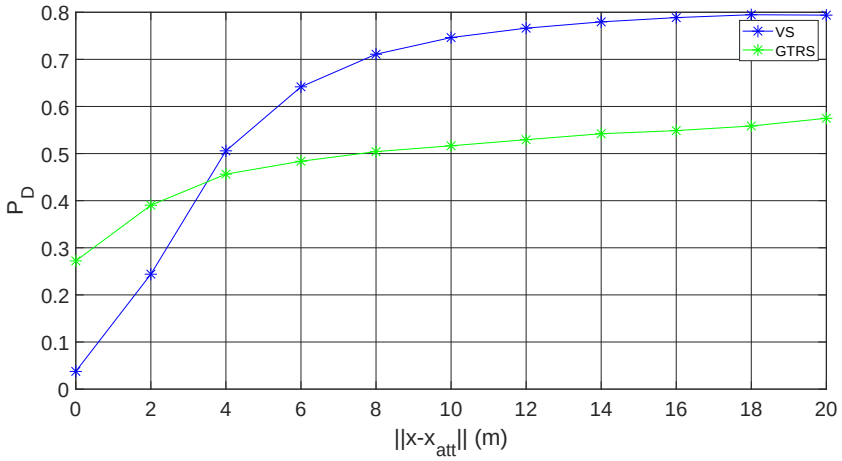
(c) $N = 8$ and $\sigma = 1$ dB

Figure 4.10: Probability of detection, P_D, versus attack intensity δ (dB) for different number of anchor nodes, N, when $\gamma = 3$, $B = 25$ m, and $\sigma = 1$ dB, with a two coordinated malicious nodes malicious node.

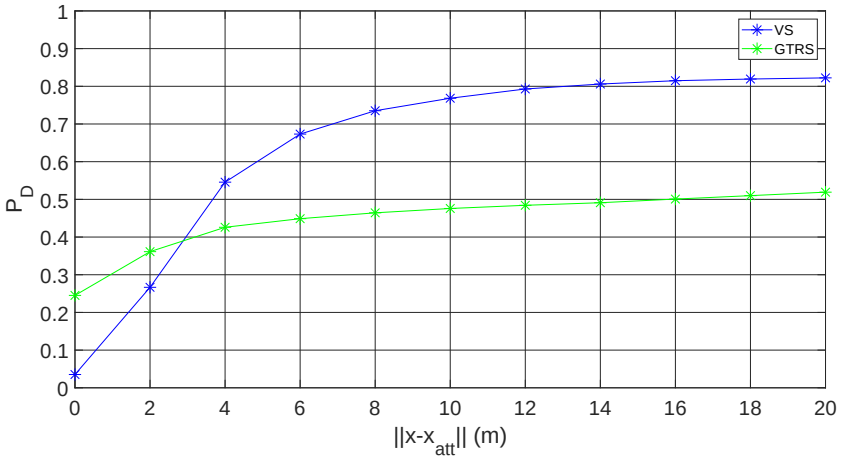
Figure 4.11 shows the probability of (successful) detection, P_D, against $\|\boldsymbol{x} - \boldsymbol{x}_{att}\|$ (m) for different choices of N when $\sigma = 2$ dB. This figure shows slightly worse results than the previous one, which is expected since the noise power has increased. Once more, the proposed method outperforms the considered one with exception for $\|\boldsymbol{x} - \boldsymbol{x}_{att}\| < 4$ m when $N = 6$, $\|\boldsymbol{x} - \boldsymbol{x}_{att}\| < 3.8$ m when $N = 7$, and $\|\boldsymbol{x} - \boldsymbol{x}_{att}\| < 3$ m when $N = 8$. Nevertheless, one should bear in mind that GTRS requires two iterations to reach its final solution, while the proposed scheme requires only a single iteration. In conclusion, the proposed solution outperforms the considered method for high values of attack distance in most settings.



(a) $N = 6$ and $\sigma = 2$ dB



(b) $N = 7$ and $\sigma = 2$ dB



(c) $N = 8$ and $\sigma = 2$ dB

Figure 4.11: Probability of detection, P_D, versus attack intensity δ (dB) for different number of anchor nodes, N, when $\gamma = 3$, $B = 25$ m, and $\sigma = 2$ dB, with a single malicious node.

Chapter 5

Conclusions and Future Work

5.1 Conclusion

In this thesis a novel geometric approach for secure localization in randomly-deployed WSNs for both non-coordinated and coordinated spoofing attacks was presented. The proposed solution requires a single iteration to solve the localization problem and takes into consideration all nodes in the network in the beginning, including the malicious nodes, and uses a voting scheme to filter out/diminish the influence of potentially corrupted ones. The location estimate is obtained via WCM using the (normalized) votes as weights and the detection of attackers is based on confidence intervals calculated by taking advantage of the location estimate. The proposed solution was evaluated in terms of localization accuracy and probability of successful detection through MC simulations, where its efficiency outperformed the considered methods in most of the admitted scenarios, maintaining close localization performance to its lower bound. At last, the proposed algorithm has low and linear complexity in the number of anchor nodes.

5.2 Future Work

There are numerous possibilities for future work. The most interesting, and currently under research, is UAV navigation and collision avoidance system with some prior knowledge of the environment configuration. In a scenario where anchors do not have line of sight, the transmitted signal suffers some degree of degradation depending on the number of obstacles obstructing the path, which can be modeled as an attack. The idea is for a UAV to start the navigation with an assessment of the environment, where the detection scheme can be used to detect how many obstacles are in the way based on how much degradation the transmitted signal suffered. Compared with the proposed work, the detection scheme would have multiple thresholds to compute the number of obstacles and, obviously, in contrast with the work presented, the signal degradation would only decrease the transmitted power. Afterwards, a voting-scheme could be used to assign votes across an environment grid which are more probable to have an obstacle, i.e., the higher the vote the more probable a given position is to have an obstacle.

This idea would be a supplementary system for UAV navigation.

Another interesting project would be secure tracking of multiple target with sensor cooperation. This thesis is limited to single targets localization. However, its generalization it to a more complex settings would be interesting. Adding to multiple target tracking, the generalization to non-static WSNs seems interesting and might hold great practical interest. For example, the previously mentioned UAV navigation and collision avoidance system could support cooperation between multiple UAVs to obtain a better performance. Such setting seems challenging, since there would many more variables to take into consideration, especially computational complexity which is of high importance for real-time decisions.

Lastly, a more of personal interest would be a validation of the proposed solution by conducting practical experiments. This thesis evaluates the proposed solution through computational simulations and therefore the results are very limited. Practical experiment in different setting (e.g., outdoor, indoor, etc.) would be of great value to validate this work.

Bibliography

[1] C. Qiang. A forest early fire detection algorithm based on wireless sensor networks. *Sensors & Transducers*, 166(3):73, 2014.

[2] A. Khan, S. Aziz, M. Bashir, and M. U. Khan. Iot and wireless sensor network based autonomous farming robot. In *2020 International Conference on Emerging Trends in Smart Technologies (ICETST)*, pages 1–5, 2020. doi: 10.1109/ICETST49965.2020.9080736.

[3] M. Ilyas. Wireless sensor networks for smart healthcare. In *2018 1st International Conference on Computer Applications and Information Security (ICCAIS)*, pages 1–5, 2018. doi: 10.1109/CAIS.2018.8442038.

[4] K. Derr and M. Manic. Wireless sensor networks—node localization for various industry problems. *IEEE Transactions on Industrial Informatics*, 11(3):752–762, 2015. doi: 10.1109/TII.2015.2396007.

[5] A. Khan, S. Aziz, M. Bashir, and M. U. Khan. Iot and wireless sensor network based autonomous farming robot. In *2020 International Conference on Emerging Trends in Smart Technologies (ICETST)*, pages 1–5, 2020. doi: 10.1109/ICETST49965.2020.9080736.

[6] A. Oigbochie, E. Odigie, and B. Adejumo. Importance of drones in healthcare delivery amid a pandemic: Current and generation next application. *Open Journal of Medical Research (ISSN: 2734-2093)*, 2(1):01–13, 2021.

[7] L. Ghelardoni, A. Ghio, and D. Anguita. Smart underwater wireless sensor networks. In *2012 IEEE 27th Convention of Electrical and Electronics Engineers in Israel*, pages 1–5, 2012. doi: 10.1109/EEEI.2012.6376941.

[8] S. Tomic, M. Beko, R. Dinis, and L. Bernardo. On target localization using combined rss and aoa measurements. *Sensors*, 18(4), 2018.

[9] V. Kumar, G. Sakya, and C. Shankar. Wsn and iot based smart city model using the mqtt protocol. *Journal of Discrete Mathematical Sciences and Cryptography*, 22(8):1423–1434, 2019.

[10] A. Khalifeh, K. A. Darabkh, A. M. Khasawneh, I. Alqaisieh, M. Salameh, A. AlAbdala, S. Alrubaye, A. Alassaf, S. Al-HajAli, R. Al-Wardat, N. Bartolini, G. Bongiovannim, and K. Rajendiran. Wireless sensor networks for smart cities: Network design, implementation and performance evaluation. *Electronics*, 10(2), 2021.

[11] J. P. Matos-Carvalho, R. Santos, S. Tomic, and M. Beko. Gtrs-based algorithm for uav navigation in indoor environments employing range measurements and odometry. *IEEE Access*, 9:89120–89132, 2021. doi: 10.1109/ACCESS.2021.3089900.

[12] S. Tomic and M. Beko. A geometric approach for distributed multi-hop target localization in cooperative networks. *IEEE Transactions on Vehicular Technology*, 69(1):914–919, 2020. doi: 10.1109/TVT.2019.2952715.

[13] A. Coluccia and A. Fascista. On the hybrid toa/rss range estimation in wireless sensor networks. *IEEE Transactions on Wireless Communications*, 17:361–371, 2018.

[14] M. R. Gholami, S. Gezici, and E. G. Ström. Tw-toa based positioning in the presence of clock imperfections. *Digital Signal Processing*, 59:19–30, 2016.

[15] S. Tomic, M. Beko, L. M. Camarinha-Matos, and L. B. Oliveira. Distributed localization with complemented rss and aoa measurements: Theory and methods, 2020.

[16] Y. Zou:2016, J. Zhu, X. Wang, and L. Hanzo. A survey on wireless security: Technical challenges, recent advances, and future trends. *Proceedings of the IEEE*, 104(9):1727–1765, 2016. doi: 10.1109/JPROC.2016.2558521.

[17] K. C. Zeng, Y. Shu, S. Liu, Y. Dou, and Y. Yang. A practical gps location spoofing attack in road navigation scenario. In *Proceedings of the 18th International Workshop on Mobile Computing Systems and Applications*, HotMobile '17, page 85–90, 2017.

[18] Z. Li, W. Trappe, Y. Zhang, and B. Nath. Robust statistical methods for securing wireless localization in sensor networks. In *IPSN 2005. Fourth International Symposium on Information Processing in Sensor Networks, 2005.*, pages 91–98, 2005. doi: 10.1109/IPSN.2005.1440903.

[19] D. Liu, P. Ning, A. Liu, C. Wang, and W. Du. Attack-resistant location estimation in wireless sensor networks. *Electrical Engineering and Computer Science*, 11, 11 2007.

[20] R. Garg, A. L. Varna, and M. Wu. An efficient gradient descent approach to secure localization in resource constrained wireless sensor networks. *IEEE Transactions on Information Forensics and Security*, 7(2):717–730, 2012.

[21] X. Liu, S. Su, F. Han, Y. Liu, and Z. Pan. A range-based secure localization algorithm for wireless sensor networks. *IEEE Sensors Journal*, 19(2):785–796, 2019. doi: 10.1109/JSEN.2018.2877306.

[22] Y. Li, S. Ma, G. Yang, and K.-K. Wong. Secure localization and velocity estimation in mobile iot networks with malicious attacks. *IEEE Internet of Things Journal*, 8(8):6878–6892, 2021. doi: 10.1109/JIOT.2020.3036849.

[23] M. Beko and S. Tomic. Towards secure localization in randomly deployed wireless networks. *IEEE Internet of Things Journal*, 8(24):17436–17448, 2021. doi: 10.1109/JIOT.2021.3078216.

[24] S. Tomic and M. Beko. Detecting distance-spoofing attacks in arbitrarily-deployed wireless networks. *IEEE Transactions on Vehicular Technology*, 71(4):4383–4395, 2022. doi: 10.1109/TVT.2022.3148199.

[25] B. Mukhopadhyay, S. Srirangarajan, and S. Kar. Rss-based localization in the presence of malicious nodes in sensor networks. *IEEE Transactions on Instrumentation and Measurement*, 70:1–16, 2021. doi: 10.1109/TIM.2021.3104385.

[26] S. M. Kay. *Fundamentals of statistical signal processing: estimation theory*, volume 1. Prentice-Hall, Inc., 1993.

[27] S. M. Kay. *Fundamentals of statistical signal processing: detection theory*, volume 2. Prentice-Hall, Inc., 1993.

[28] N. Patwari. *Location estimation in sensor networks*. University of Michigan, 2005.

[29] J. C. A. Barata and M. S. Hussein. The moore–penrose pseudoinverse: A tutorial review of the theory. *Brazilian Journal of Physics*, 42(1-2):146–165, dec 2011. doi: 10.1007/s13538-011-0052-z. URL `https://doi.org/10.1007%2Fs13538-011-0052-z`.

[30] S. Messous, H. Liouane, O. Cheikhrouhou, and H. Hamam. Improved recursive dv-hop localization algorithm with rssi measurement for wireless sensor networks. *Sensors*, 21(12), 2021. doi: 10.3390/s21124152.

[31] Q.-l. Du, X. Sun, R. Ding, Z.-h. Qian, and S.-x. Wang. Toa-based location estimation accuracy for 3d wireless sensor networks. In *2009 5th International Conference on Wireless Communications, Networking and Mobile Computing*, pages 1–4, 2009. doi: 10.1109/WICOM.2009.5302827.

[32] J. He, Y. Geng, and K. Pahlavan. Toward accurate human tracking: Modeling time-of-arrival for wireless wearable sensors in multipath environment. *IEEE Sensors Journal*, 14(11):3996–4006, 2014. doi: 10.1109/JSEN.2014.2356857.

[33] H. Xiong, Z. Chen, B. Yang, and R. Ni. Tdoa localization algorithm with compensation of clock offset for wireless sensor networks. *China Communications*, 12(10):193–201, 2015. doi: 10.1109/CC.2015.7315070.

[34] S. K. Meghani and M. Asif. Localization of wsn node based on rtt toa using ultra wide band amp; 802.15.4a channel. In *Proceedings of the 11th IEEE International Conference on Networking, Sensing and Control*, pages 380–385, 2014. doi: 10.1109/ICNSC.2014.6819656.

[35] L. Yu, C. Hou, X. Li, and K. Yuan. An indoor positioning system based on tof for wireless sensor networks. In *IET International Conference on Information Science and Control Engineering 2012 (ICISCE 2012)*, pages 1–6, 2012. doi: 10.1049/cp.2012.2351.

[36] P. Biswas, H. Aghajan, and Y. Ye. Semidefinite programming algorithms for sensor network localization using angle information. In *Conference Record of the Thirty-Ninth Asilomar Conference*

onSignals, Systems and Computers, 2005., pages 220–224, 2005. doi: 10.1109/ACSSC.2005.1599736.

[37] A. Fascista, G. Ciccarese, A. Coluccia, and G. Ricci. Angle of arrival-based cooperative positioning for smart vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 19(9):2880–2892, 2018. doi: 10.1109/TITS.2017.2769488.

[38] S. Tomic, M. Beko, R. Dinis, and M. Raspopovic. Distributed rss-based localization in wireless sensor networks using convex relaxation. In *2014 International Conference on Computing, Networking and Communications (ICNC)*, pages 853–857, 2014. doi: 10.1109/ICCNC.2014.6785449.

[39] A. Coluccia and F. Ricciato. Rss-based localization via bayesian ranging and iterative least squares positioning. *IEEE Communications Letters*, 18(5):873–876, 2014. doi: 10.1109/LCOMM.2014.040214.132781.

[40] M. S. Costa, S. Tomic, and M. Beko. An socp estimator for hybrid rss and aoa target localization in sensor networks. *Sensors*, 21(5), 2021. doi: 10.3390/s21051731.

[41] S. Tomic, M. Marikj, M. Beko, R. Dinis, and N. Órfão. Hybrid rss-aoa technique for 3-d node localization in wireless sensor networks. In *2015 International Wireless Communications and Mobile Computing Conference (IWCMC)*, pages 1277–1282, 2015. doi: 10.1109/IWCMC.2015.7289266.

[42] S. Tomic, M. Beko, and R. Dinis. 3-d target localization in wireless sensor networks using rss and aoa measurements. *IEEE Transactions on Vehicular Technology*, 66(4):3197–3210, 2017. doi: 10.1109/TVT.2016.2589923.

[43] N. Bulusu, J. Heidemann, and D. Estrin. Gps-less low-cost outdoor localization for very small devices. *IEEE Personal Communications*, 7(5):28–34, 2000. doi: 10.1109/98.878533.

[44] L. Nian-qiang and L. Ping. A range-free localization scheme in wireless sensor networks. In *2008 IEEE International Symposium on Knowledge Acquisition and Modeling Workshop*, pages 525–528, 2008. doi: 10.1109/KAMW.2008.4810540.

[45] L. Doherty, K. pister, and L. El Ghaoui. Convex position estimation in wireless sensor networks. In *Proceedings IEEE INFOCOM 2001. Conference on Computer Communications. Twentieth Annual Joint Conference of the IEEE Computer and Communications Society (Cat. No.01CH37213)*, volume 3, pages 1655–1663 vol.3, 2001. doi: 10.1109/INFCOM.2001.916662.

[46] S. U. Rehman, K. W. Sowerby, S. Alam, and I. Ardekani. Radio frequency fingerprinting and its challenges. In *2014 IEEE Conference on Communications and Network Security*, pages 496–497. IEEE, 2014.

[47] D. A. Knox and T. Kunz. Secure authentication in wireless sensor networks using rf fingerprints. In *2008 IEEE/IFIP International Conference on Embedded and Ubiquitous Computing*, volume 1, pages 230–237, 2008. doi: 10.1109/EUC.2008.114.

[48] D. A. Knox and T. Kunz. Practical rf fingerprints for wireless sensor network authentication. In *2012 8th International Wireless Communications and Mobile Computing Conference (IWCMC)*, pages 531–536, 2012. doi: 10.1109/IWCMC.2012.6314260.

[49] G. Farjamnia, Y. Gasimov, C. Kazimov, and M. hashemi. A survey of dv-hop localization methods in wireless sensor networks. *Journal of Communication Engineering*, 9(2):359–398, 2020. doi: 10.22070/jce.2021.14405.1186.

[50] S. Kumar and D. Lobiyal. An advanced dv-hop localization algorithm for wireless sensor networks. *Wireless personal communications*, 71(2):1365–1385, 2013.

[51] Y. Hu and X. Li. An improvement of dv-hop localization algorithm for wireless sensor networks. *Telecommunication Systems*, 53(1):13–18, 2013.

[52] C. Knapp and G. Carter. The generalized correlation method for estimation of time delay. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 24(4):320–327, 1976. doi: 10.1109/TASSP.1976.1162830.

[53] N. Patwari, A. Hero, M. Perkins, N. Correal, and R. O'Dea. Relative location estimation in wireless sensor networks. *IEEE Transactions on Signal Processing*, 51(8):2137–2148, 2003. doi: 10.1109/TSP.2003.814469.

[54] N. Correal, S. Kyperountas, Q. Shi, and M. Welborn. An uwb relative location system. pages 394 – 397, 12 2003. ISBN 0-7803-8187-4. doi: 10.1109/UWBST.2003.1267871.

[55] J. Miranda, R. Abrishambaf, T. Gomes, P. Gonçalves, J. Cabral, A. Tavares, and J. Monteiro. Path loss exponent analysis in wireless sensor networks: Experimental evaluation. In *2013 11th IEEE International Conference on Industrial Informatics (INDIN)*, pages 54–58, 2013. doi: 10.1109/INDIN.2013.6622857.

[56] E. Niewiadomska-Szynkiewicz. Localization in wireless sensor networks: Classification and evaluation of techniques. *International Journal of Applied Mathematics and Computer Science*, 22: 281–297, 06 2012. doi: 10.2478/v10006-012-0021-x.

[57] A. Naguib. Multilateration localization for wireless sensor networks. *Indian Journal of Science and Technology*, 13:1213–1223, 03 2020. doi: 10.17485/ijst/2020/v13i10/150005.

[58] F. Mekelleche and H. Hafid. Classification and comparison of range-based localization techniques in wireless sensor networks. *Journal of Communications*, 12:221–227, 04 2017. doi: 10.12720/jcm.12.4.221-227.

[59] S. M. Kay. *Fundamentals of statistical signal processing: estimation theory*. Prentice-Hall, Inc., 1993.

Appendix A

Maximum Likelihood Estimators

From (3.1), the joint probability of the K RSS measurements from the i-th anchor to the target initial estimate is given as

$$f(p_{i,k};\ \sigma_i,\ \delta_i) = \prod_{k=1}^{K} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{ -\frac{\left(p_{i,k} - p_0 + 10\gamma \log_{10}\left(\frac{\widehat{d}_i}{d_0}\right) - \delta_i\right)^2}{2\sigma^2} \right\}\ , \tag{A.1}$$

where δ and σ are the attack intensity and noise standard deviation, respectively, to be estimated, and $\widehat{d}_i$ is the distance between the i-th anchor and the target initial location estimate. The log-likelihood function is obtained by applying the natural logarithm function to A.1.

$$\ln(f(p_{i,k};\ \sigma_i,\ \delta_i)) = -\frac{K}{2}\ln(2\pi\sigma^2) - \sum_{k=1}^{K} \frac{\left(p_{i,k} - p_0 + 10\gamma \log_{10}\left(\frac{\widehat{d}_i}{d_0}\right) - \delta_i\right)^2}{2\sigma^2}. \tag{A.2}$$

A.1 Attack Intensity MLE Derivation

The estimator for the attack intensity is given by

$$\widehat{\delta}_i = \arg\max_{\delta_i} \ln[f(p_{i,k};\ \sigma_i,\ \delta_i)]. \tag{A.3}$$

Taking the derivative of the log-likelihood function (A.2) in respect to δ_i produces

$$\frac{\partial}{\partial\delta_i} \ln[(f(p_{i,k};\ \sigma_i,\ \delta_i)] = -\sum_{k=1}^{K} p_0 - p_{i,k} - 10\gamma \log_{10}\left(\frac{\widehat{d}_i}{d_0}\right) + \delta_i\ ,$$

which being set to zero yields the MLE

$$\widehat{\delta}_i = \frac{\sum_{k=1}^{K} p_0 - p_{i,k} - 10\gamma \log_{10} \|\widehat{\boldsymbol{x}} - \boldsymbol{a_i}\|}{K}. \tag{A.4}$$

A.2 Noise Standard Deviation MLE Derivation

The estimator for the noise standard deviation for the i-th anchor is given by

$$\widehat{\sigma}_i = \arg\max_{\sigma} \ln[f(p_{i,k};\ \sigma_i,\ \delta_i)]. \tag{A.5}$$

Note that $\widehat{\delta}_i$ refers to the attack intensity estimation (A.4). Taking the derivative of the log-likelihood function (A.2) in respect to σ_i produces

$$\frac{\partial}{\partial \sigma_i} \ln[(f(p_{i,k};\ \sigma_i)] = -\frac{K}{\sigma_i} + \frac{1}{\sigma_i^3} \sum_{k=1}^{K} \left(p_{i,k} - p_0 + 10\gamma \log_{10}\left(\frac{\widehat{d}_i}{d_0}\right) - \widehat{\delta}_i \right)^2 ,$$

which being set to zero yields the MLE

$$\widehat{\sigma}_i = \sqrt{\frac{1}{(K-1)} \sum_{k=1}^{K} \left(p_{i,k} - p_0 + 10\gamma \log_{10}\left(\frac{\widehat{d}_i}{d_0}\right) - \widehat{\delta}_i \right)^2} . \tag{A.6}$$

The estimate of noise standard deviation is given by the average of the values $\widehat{\sigma}_i$.

$$\widehat{\sigma} = \frac{1}{N} \sum_{i=1}^{N} \sqrt{\frac{1}{(K-1)} \sum_{k=1}^{K} \left(p_{i,k} - p_0 + 10\gamma \log_{10}\left(\frac{\widehat{d}_i}{d_0}\right) - \widehat{\delta}_i \right)^2} .$$