Towards Secure Localization in Randomly Deployed Wireless Networks

Marcelo Costa †, Marko Beko † and Slavisa Tomic *
† Instituto Superior Técnico, Lisboa, Portugal
* COPELABS, Universidade Lusófona de Humanidades e Tecnologias, Lisboa, Portugal

***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 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.**

***Index Terms*—Secure Localization, Wireless Sensor Networks, Target Localization, Received Signal Strength, Spoofing Attacks.**

I. Introduction

Recently, wireless sensor networks (WSNs) have attracted much interest of the scientific community, partially due to their ability to work in harsh environments, ease and low costs of implementation [1–3], and wide variety of applications [4–9]. From the localization perspective, generally, WSNs are composed of two distinct types of nodes: 1) anchor nodes, whose locations are known and serve as reference points in the localization process and 2) target nodes, whose locations are unknown and one desires to determine them. In this work, a non-cooperative network, where target nodes are only allowed to communicate with anchor nodes, is considered.

In most applications, data acquired by sensors are only useful if they can be associated with the respective physical location. However, most of existing localization systems are designed under the consideration that there are no security threats in the environment where they are integrated [10–13]. Therefore, if these systems are exposed to spoofing attacks, especially in applications that highly depend on accurate localization, they can result in catastrophic outcomes (e.g., failure in a self-driving car collision system, change in drone trajectory, etc.). Mainly for this reason, localization systems should be developed for adversarial environments, where a malicious (or damaged) sensor present in the network can produce false distance measurements (spoof attacks) in order to manipulate the localization estimation.

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 infeasibility in some environments (e.g., indoor, urban areas, forests, etc.). In addition, from the security point of view, GPS is considered a civilian localization system (e.g., uses unencrypted signals); thus, is not very difficult to manipulate (spoof) it. This work makes use of terrestrial radio signals only (nevertheless, it can be used as a complement to the GPS system or even as its main algorithm) from already deployed technologies and proposes a novel voting-based scheme to achieve reliable (secure) localization in adverse settings [14].

This thesis is organized as follows. Section II introduces prior work in secure localization techniques for WSN. Section III formulates the problem of interest. Section IV provides a detailed description of the proposed solution. Section V evaluates the performance of the proposed solution in terms of computational complexity, detection and localization accuracy. Lastly, Section VI concludes the paper and summarizes the main findings of the work.

II. Related Work

Secure localization in WSNs has been addressed in the literature with different detection schemes to tackle this problem [15–19]. Nonetheless, both the achieved detection rates and localization accuracy can be further improved with the possibility to reduce the algorithm's computational complexity.

The work in [17] exploits the problem's geometry in order to obtain an initial location estimate based on WCM. Subsequently, it computes distance estimates from the initial location estimate to all anchors. The distance estimates are compared to a predefined threshold in order to distinguish and remove malicious from the location process. Lastly, the localization problem is transformed into a generalized trust region sub-problem (GTRS) and solved by a bisection method. Nevertheless, the authors in [17] considered only two-way TOA measurements, where attackers could only enlarge distance measurements.

The work in [18] is a generalization of [17], where the

authors considered a general range-based scenario and employed generalized likelihood ratio test (GLRT) for detection and law of cosines (LC) to convert the problem into a GTRS.

In [19], the authors presented two RSS-based solutions for target localization in the presence of malicious nodes, namely, secure weighted least squares (SWLS), and l1-norm-based solution (LN-1E). In the SWLS algorithm, the authors first compute an estimate of the noise power for each anchor. The malicious anchors are identified by comparing the estimated noise standard deviation with a specified (empirical) threshold. Lastly, WLS criterion is applied considering only the (supposedly) honest anchors. However, the threshold employed in the detection procedure proposed in [19] is based on the true knowledge of noise power, which compromises SWLS's , performance in practice, since perfect knowledge of the noise power is not available. In the LN-1E algorithm, the authors employ a 3-D plane fitting based on data points corresponding to the RSS measurements, where the non-malicious anchors are expected to lay further away from the plane. Afterwards, the authors use the K-means clustering algorithm to separate malicious anchors from genuine ones. Lastly, another plane fit is computed based on non-malicious anchor nodes and an estimate of the target location is obtained.

The proposed solution takes advantage of the problem geometry to compute a set of intersection points between all pairs of anchors in order to obtain an estimate of the target's location through a voting-scheme and WCM, which is then exploited to detect attackers based on confidence intervals. A set of the highest votes of the points are then converted into probabilities, which are used as weights to estimate the target's location via WCM. A potential attack intensity is estimated for each node according to the maximum likelihood (ML) criterion. 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.

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

1) Design of a novel and single-iteration solution for target localization in randomly-deployed WSN in the presence of spoofing attacks 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.

III. 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. 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. Nevertheless, both types of spoofing attacks are described in the following two subsections.

A. Uncoordinated Attack

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

$$p_{i,k} = p_0 - 10\gamma \log_{10}\left(\frac{d_i}{d_0}\right) + \delta_i + n_{i,k} \ , \qquad (1)$$

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 $n_{i,k} \sim \mathcal{N}(0, \sigma^2_{i,k})$, 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 [19].

It is important to note 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 $\sigma^2_{i,k} = \sigma^2, i = 1, ..., N$ 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 (1) from the i-th anchor node (p_i) is used in the following derivations.

Figure 1(a) shows an uncoordinated attack, where three of the seven anchors are malicious and report, independently, false distance measurements, aggravating the localization process. From (1) (and by employing p_i) the conditional PDF of the RSS model for a single anchor is given by

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

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

From (2), the location of the target node can be estimated based on the ML criterion [20]. However, the ML estimator is non-convex and therefore difficult to tackle directly. In a later section, a voting scheme and a detection scheme are introduced to estimate the target location and detect a malicious node.

B. Coordinated Attack

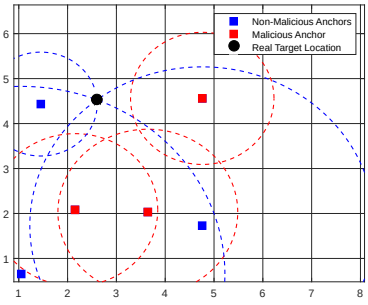
In coordinated attacks the malicious anchors communicate with each other to agree with a (false) location for the target. The idea is to make the network believe that the target is at a different location than it actually is. Similar to [19], the coordinated attack can be modeled as

$$p_{i,k} = \begin{cases} p_0 - 10\gamma \log_{10}\left(\dfrac{\|\boldsymbol{x} - \boldsymbol{a}_i\|}{d_0}\right) + n_{i,k} & \text{if } i \in \mathcal{H}, \\ p_0 - 10\gamma \log_{10}\left(\dfrac{\|\boldsymbol{x}_{att} - \boldsymbol{a}_i\|}{d_0}\right) + n_{i,k} & \text{if } i \in \mathcal{M}, \end{cases} \tag{3}$$

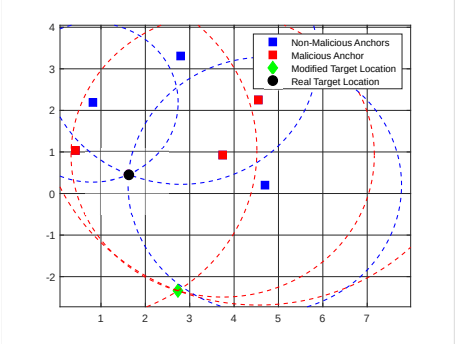
where $\boldsymbol{x}_{att}$ is the location the malicious nodes agree to perform the attack, $\mathcal{M}$ and $\mathcal{H}$ are, respectively, the set of malicious and honest nodes. The distance between the target's true location and the fabricated one is scaled by a factor of $\frac{\|\boldsymbol{x}_{att}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

Figure 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 ($\boldsymbol{x}$).

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.

Fig. 1: Types of spoofing attacks done by malicious anchors in a WSN for noise-free measurements.

IV. The Proposed Method

This section describes the derivation of the proposed algorithm for secure localization in randomly deployed WSNs. It is organized into three parts: 1) a preliminary part, where points of interest are determined, and two main parts in which 2) the proposed localization estimator is described in detail based on a voting scheme, and 3) the proposed detection procedure to identify attackers is introduced.

A. Determining Points of Interest

At the beginning, all anchor nodes are treated as honest. Hence, the set of malicious nodes can be initiated as $\mathcal{M} = \varnothing$, and the set of honest nodes as $\mathcal{H} = \{i : 1 \leq i \leq N\}$. Afterwards, one can construct circles, c_i, centered at the known locations of the anchor nodes and radii equivalent to the distance estimate, $\widehat{d}_i = d_0 10^{\frac{p_0 - p_i}{10\gamma}}$, obtained from (1), of the respective anchor node (we refer the reader to see Figure 2(a)). The intersection points between all pairs of circles are used as points of interest for the voting scheme. The intersection points between a pair of circles (given that they exist) can be calculated as follows

$$\boldsymbol{q}'_{ij} = \boldsymbol{q}_0 + \boldsymbol{t} \text{ and } \boldsymbol{q}''_{ij} = \boldsymbol{q}_0 - \boldsymbol{t}, \text{ for } i = 1, ..., N-1, \quad j = i+1, ..., N, \tag{4}$$

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 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 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

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

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

$$
\begin{aligned}
\boldsymbol{q}'_{ij} &= \frac{\boldsymbol{q}_{i,1} + \boldsymbol{q}_{j,1}}{2} \\
\boldsymbol{q}''_{ij} &= \frac{\boldsymbol{q}_{i,2} + \boldsymbol{q}_{j,2}}{2}
\end{aligned}
\tag{5}
$$

This idea has been implemented in [18], 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.

B. The Proposed Voting-based Scheme for Target Localization

The voting scheme is a process to cluster and assign votes to the intersection points 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 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 it is computed a hyperplane, $H_{ij} = \{g : \widehat{\boldsymbol{b}}^\intercal \boldsymbol{Q}_h = \widehat{\boldsymbol{b}}^\intercal \boldsymbol{s}_0\}$, 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. All things considered, the vote, for the h-th intersection point, given the hyperplane H_{ij}, is calculated as

$$
\begin{aligned}
v_h = v_h &+ w_h \frac{proj_{H_{ij}}(\boldsymbol{Q}_h)}{\sum_{h:h\in C_{ij}^{(u)}} proj_{H_{ij}}(\boldsymbol{Q}_h)} \\
&+ w_h \frac{proj_{H_{ij}}(\boldsymbol{Q}_h)}{\sum_{h:h\in C_{ij}^{(l)}} proj_{H_{ij}}(\boldsymbol{Q}_h)},\ h \in C_{ij}^{(u)} \cup C_{ij}^{(l)}
\end{aligned}
\tag{6}
$$

where

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

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

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, $\boldsymbol{v} = [v_h]^\intercal$, 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 (6).

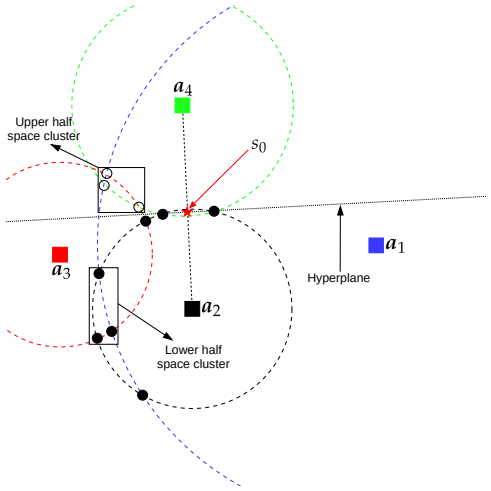


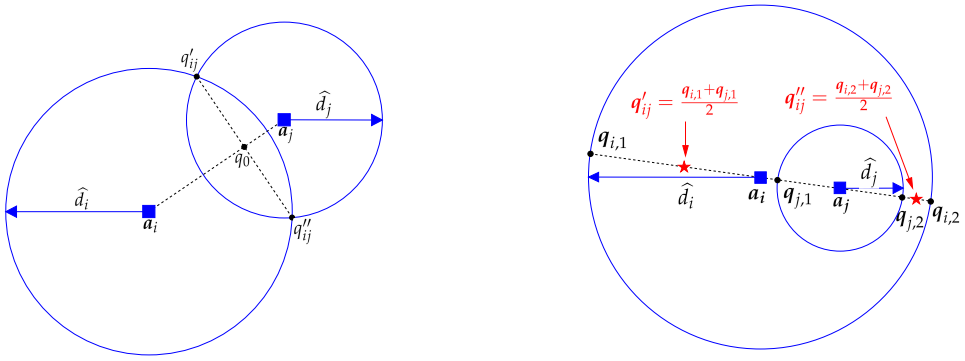
Fig. 3: Illustration of the voting process between two anchor nodes when $N = 4$.

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

$$
\widehat{\boldsymbol{x}} = \sum_{h=1}^{N-1} \tilde{w}_h \boldsymbol{Q}_h \tag{7}
$$

with

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



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

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

Fig. 2: Illustration of the possible scenarios in finding circles intersections.

C. Attack Detection

Considering (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 (7)) based on the ML principle, as well as the noise standard deviation and the expected (honest) RSS value for the estimated target location.

$$\widehat{\delta}_i = \frac{\sum_{k=1}^{K} p_0 - p_{i,k} - 10\gamma \log_{10} \|\widehat{\boldsymbol{x}} - \boldsymbol{a_i}\|}{K}, \tag{8}$$

$$\widehat{\sigma} = \frac{1}{N} \sum_{i=1}^{N} \sqrt{\frac{1}{(K-1)} \sum_{k=1}^{K} \left(\alpha_i - \widehat{\delta}_i\right)^2},$$

$$\widehat{p}_i = p_0 - 10\gamma \log_{10} \|\widehat{\boldsymbol{x}} - \boldsymbol{a_i}\|. \tag{9}$$

where $\alpha_i = p_{i,k} - p_0 + 10\gamma \log_{10} \|\widehat{\boldsymbol{x}} - \boldsymbol{a_i}\|$.

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 (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 $\mathcal{M} = \{i : \widehat{p}_i + \widehat{\sigma} < p_i < \widehat{p}_i - \widehat{\sigma}\}$.

It is important to note that, in contrast to [17–19], 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.

V. 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.

A. 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 I 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 [18] and the LN-1 algorithm proposed in [19] 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 [18] requires two estimation steps to reach its final solution. Moreover, it is noteworthy that all approaches in [18] and [19] 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). 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 [18] requires two iterations to obtain the final estimation, whereas the proposed solution solves the localization problem in a single iteration.

B. Localization and Attacker Detection Analysis

This section presents an assessment in localization accuracy and success of attacker detection for the considered methods in Table I 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 $N_D = 800$

Algorithm	Complexity	Average Running Time (s)	Coordinated Attacks	Uncoordinated Attacks
VS in Chapter IV	$\mathcal{O}(N)$	0.0139	✓	✓
SWLS in [19]	$\mathcal{O}(N)$	0.0015	✗	✓
LN-1 in [19]	$\mathcal{O}(B_{ADMM}N)$	0.0063	✓	✓
GTRS in [18]	$\mathcal{O}(N \times B_{max})$	0.0254	✓	✓

TABLE I: Worst-case computational complexity and average running time of the considered methods.

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.

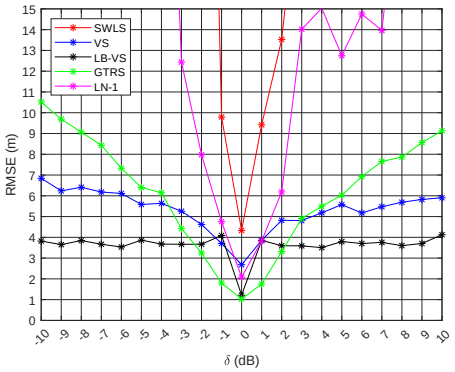
The RSS measurements were obtained through (1), where the transmit power at the target node is set to $p_0 = -10$ dBm, the PLE is fixed at $\gamma = 3$, and $K = 10$. The simulations for uncoordinated attacks considered single and two malicious nodes and the simulations for coordinated attacks considered 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.

B1 Uncoordinated Attacks - Single Malicious Node Scenario

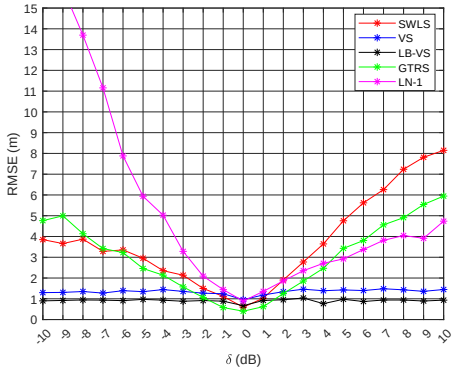
Figure 4 shows the RMSE (m) versus the attack intensity δ (dB) for different number of anchor nodes and $\sigma = 1$ dB. For the purpose of better understanding the performance of the proposed method, the lower bound, "LB-VS", on the performance of the proposed algorithm where the ideal detection is 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. 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. 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.

Figure 5 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, 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.

Figure 6 shows the probability of (successful) detection,

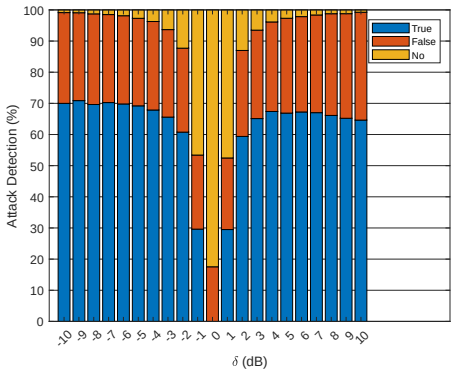


(a) $N = 4$

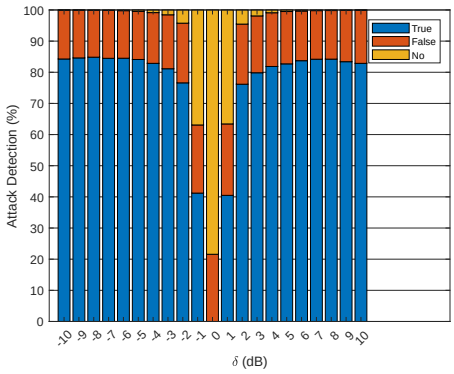


(b) $N = 6$

Fig. 4: 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.

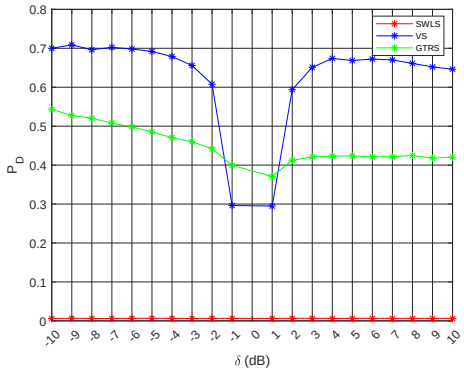


(a) $N = 4$

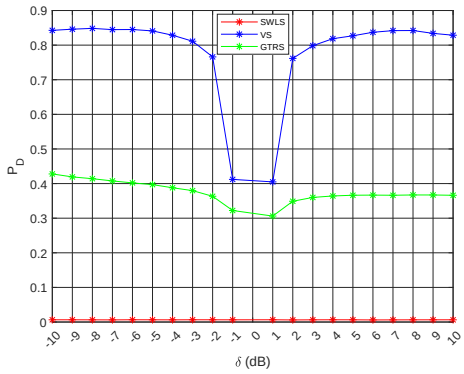


(b) $N = 6$

Fig. 5: 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.



(a) $N = 4$



(b) $N = 6$

Fig. 6: 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.

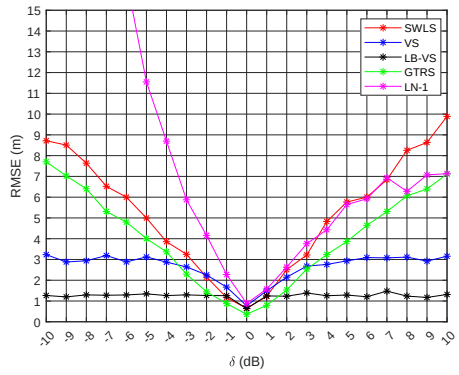
P_D, against δ (dB) for different choices of N when $\sigma = 1$ dB. The results obtained by using the proposed scheme show superior detection rates over the considered solutions in all scenarios considered with the exception of $N = 4$, where GTRS outperforms the proposed solution for small magnitudes of attack intensity. Nevertheless, as stated earlier, this behavior could even be desirable for small magnitudes of the attack. The results obtained for the SWLS solution are a consequence of the threshold tuning.

B2 Uncoordinated Attacks - Two Malicious Node Scenario

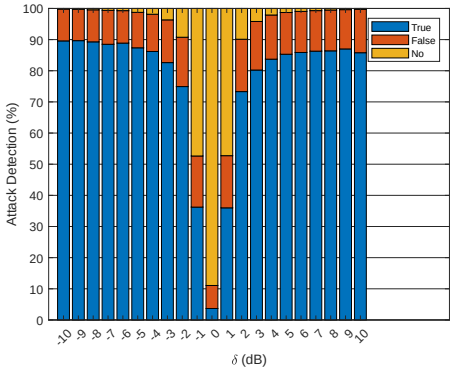
Figure 7(a) shows the RMSE (m) versus the attacker intensity δ (dB) for $\sigma = 1$ dB and $N = 6$. As expected, the RMSE values are slightly worse when the number of malicious nodes increases. The results obtained between the considered solutions are similar to the case where only one malicious node 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 7(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 5. 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. Figure 7(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.

B3 Coordinated Attacks - Two Malicious Node Scenario

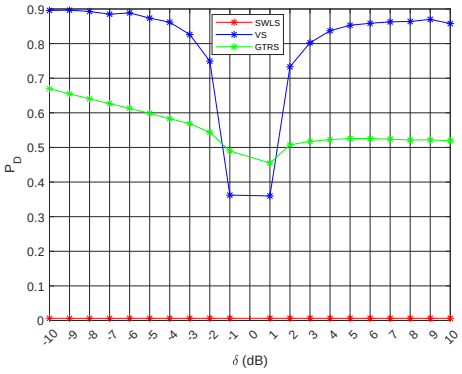
In the coordinated attack scenario the attack position, i.e., the position agreed by the malicious nodes, is obtained by choosing a random point in a circle centered at the true target and predefined radii, i.e., $\|\boldsymbol{x} - \boldsymbol{x}_{att}\|$. The simulations for coordinated attacks consider two malicious nodes. The simulation parameters are the same as in the uncoordinated



(a) RMSE versus attack intensity δ (dB)



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

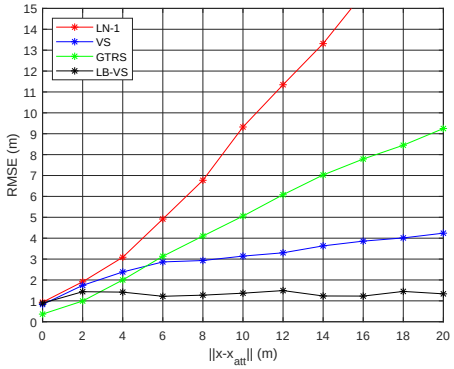


(c) P_D versus attack intensity δ (dB)

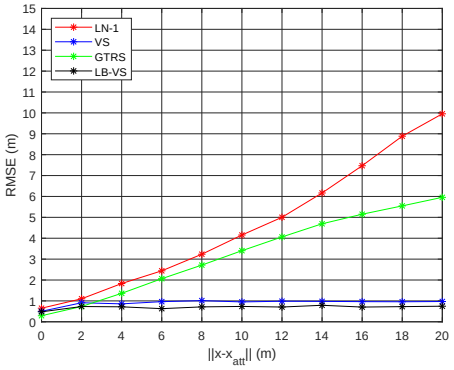
Fig. 7: Proposed solution performance for $N = 6$, $\gamma = 3$, $B = 25$ m, and $\sigma = 1$ dB, with two malicious nodes.

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 I lists the methods considered for this setting.

Figure 8 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

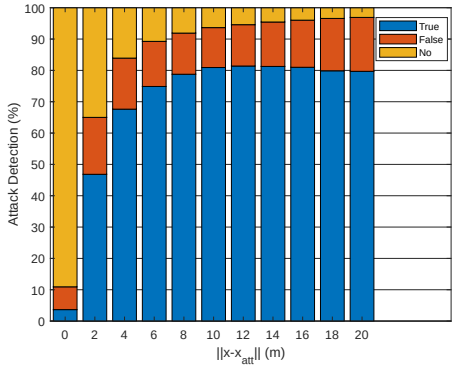


(b) $N = 8$ and $\sigma = 1$ dB

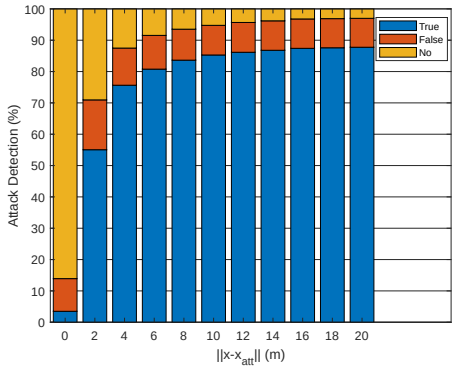
Fig. 8: RMSE versus attack distance $\|\boldsymbol{x} - \boldsymbol{x}_{att}\|$ (m) for different number of anchor nodes, N, when $\gamma = 3$, $B = 25$ m, $\sigma = 1$ dB, with two coordinated malicious node.

Figure 9 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 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 5, the results show slightly worse detection performance, which makes sense since cooperated attacks

are more complex than the uncoordinated ones. However, considering Figure 8, 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.



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

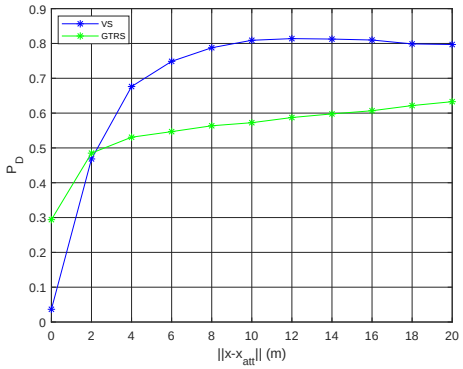


(b) $N = 8$ and $\sigma = 1$ dB

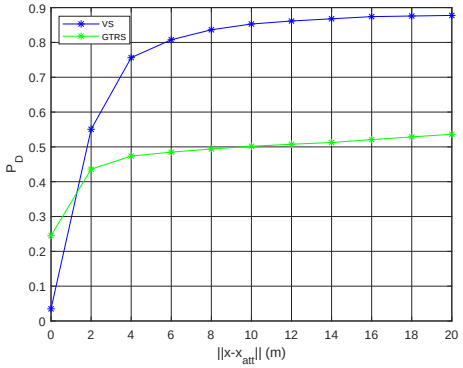
Fig. 9: Attack detection (%) versus attack distance $\|\boldsymbol{x} - \boldsymbol{x}_{att}\|$ (m) for different number of anchor nodes, N, when $\gamma = 3$, $B = 25$ m, and $\sigma = 1$ dB, with two coordinated malicious node.

Figure 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.

Figure 11(a) shows the RMSE (m) versus the distance between the target location($\boldsymbol{x}$) and the attack location ($\boldsymbol{x}_{att}$) when $N = 8$ and $\sigma = 2$ dB for two coordinated malicious nodes. 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 that the proposed solution shows far greater advantage for high values of attack distance, while the GTRS solution presents slightly better performance results for small values of attack distance and the LN-1



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



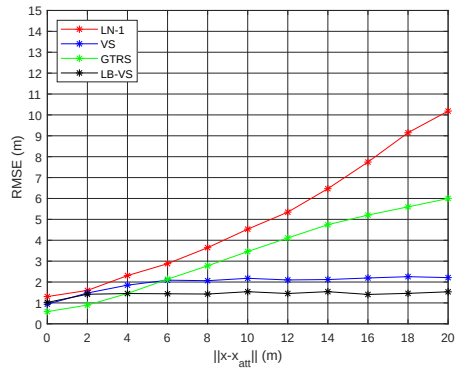
(b) $N = 8$ and $\sigma = 1$ dB

Fig. 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.

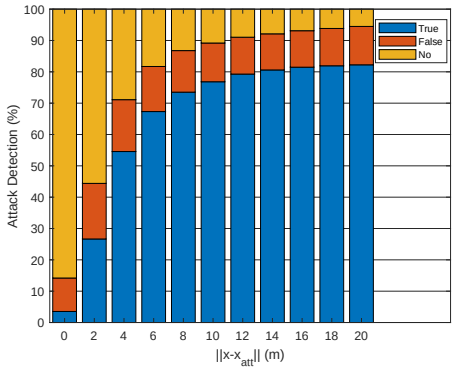
solution has severe performance loss for high values of attack distance.

Figure 11(b) illustrates the rate of success, failure, and no detection for the previously mentioned scenario. The proposed solution achieves a detection rate of 83% for high attack distances. In comparison with 9, as expected, the results show a degradation of detection rate. However, once more, considering the localization accuracy results presented in Figure 11(a), this phenomenon does not pose a major threat to the localization performance of the proposed solution.

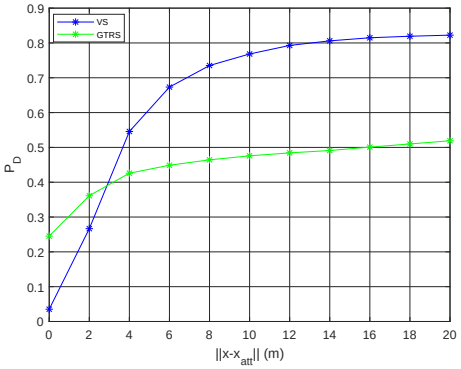
Figure 11(c) shows the probability of (successful) detection, P_D, against $\|\boldsymbol{x} - \boldsymbol{x}_{att}\|$ (m) for the previously mentioned scenario. This figure shows slightly worse results than Figure 10 due to the increased noise power. Once more, the proposed method outperforms the considered one with exception for $\|\boldsymbol{x} - \boldsymbol{x}_{att}\| < 3$.



(a) RMSE versus attack distance $\|\boldsymbol{x} - \boldsymbol{x}_a tt\|$ (m)



(b) Attack detection (%) versus attack distance $\|\boldsymbol{x} - \boldsymbol{x}_a tt\|$ (m)



(c) P_D versus attack distance $\|\boldsymbol{x} - \boldsymbol{x}_a tt\|$ (m)

Fig. 11: Proposed solution performance for $N = 8$, $\gamma = 3$, $B = 25$ m, and $\sigma = 2$ dB, with a two coordinated malicious node.

VI. Conclusions

In this thesis, it is presented a geometric approach for secure localization in randomly-deployed WSNs for non-coordinated spoofing attacks. The proposed solution requires a single iteration to solve the localization problem through a voting scheme. The location estimate is obtained via WCM using the 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. At last, the proposed algorithm has low and linear complexity in the number of anchor nodes. There are numerous possibilities for future work, some of them being: 1) UAV navigation and collision avoidance system using a similar detection scheme with multiple thresholds to detect obstacles and a voting scheme across an environment grid to assign votes to locations where it is more probable to have an obstacle; 2) secure tracking of multiple target with sensor cooperation; 3) conducting practical experiments in different setting (e.g., outdoor, indoor, etc.) to further validate this work.

References

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

[2] 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.

[3] 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.

[4] 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.

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

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

[7] 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.

[8] 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.

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

[10] 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.

[11] 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.

[12] 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.

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

[14] 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.

[15] 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.

[16] 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.

[17] M. Beko and S. Tomic. Toward secure localization in randomly deployed wireless networks. *IEEE Internet of Things Journal*, 8(24): 17436–17448, 2021.

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

[19] 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.

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