Virus: Cooperative Localization for Indoor Positioning Systems

Carla Marreiros
Instituto Superior Técnico, Av. Rovisco Pais, 1049-001 Lisboa, Portugal
carla.marreiros@tecnico.ulisboa.pt

***Abstract*—The work done has as starting point an indoor localization system developed by Thales. It uses Open Beacon based sensors and relies on RSS measurements. A method is studied to obtain a target localization (mobile antenna) from RSS measurements between the latter and the anchors - standing sensors deployed in a strategic way and at a given space. The goal was not to attain the exact target coordinates, an Euclidean representation, but the one relative to the anchors: a visual representation of the target positions over time that shows the trajectory.**

Among the used methodologies, there are filters resembling the Kalman ones and others, MDS algorithm and Procrustes Analysis. Other processes were discarded as they proved to be unnecessary and/or unsuitable.

To obtain the needed data (RSS measurements) were performed many instances of the anchors in various different physical scenarios were tested. The selected setup to illustrate the chosen methodologies was one of the ones that produced the best results, showing the preferred features to this kind of operation.

***Keywords* - RSS; Power; Sensors; Cooperative; Localization; Algorithm;**

1. Introduction

An indoor localization system is a system that can determine the position of an object or person. These systems are required by Wireless Sensor Networks, (WSNs).

The most famous localization system in the world is GPS. The fact that it has a satisfactory precision in outdoor environments [1] and is in consumer-oriented technologies contributed to its large scale use. However, this system is not fit for certain environments, such as indoor and aquatic, producing erroneous data or not working at all. As it locates devices using the Time of Flight (TOF) of radio signals transmitted by satellites, in environments where the sky view is obstructed, it is impossible to obtain the location of the first ones. With the increasing complexity of the infrastructures, the need to produce a system capable of performing indoor localization have arisen.

1.1. Objectives

The main goal of this work is to develop a method to obtain the location of a mobile sensor - target - from the measurements of Received Signal Strength (RSS) between it and the anchors - sensors strategically implemented in space under study. The system must be as efficient as possible, using the same sensors provided by Thales [2]. Given the poor quality of the measurements, the realistic goal for this work was only to obtain a rough estimate of the trajectory of the target relative to a set of anchors, rather than to try to attain a high-quality Euclidean representation.

1.2. Wireless Sensor Network

A WSN is a network in which the sensors (tags) are spatially distributed and communicate with each other, if the network is cooperative, or only with a Central Computer (CC) in case the network is centralized (star topology). The CC is necessary to act as a bridge between the sensors and the user, but can also be used to make the necessary calculations (which depend on the chosen algorithms and equipment). The site where the network is implemented can have arbitrary dimensions: from a single room to an entire building. The object to be positioned is called a target, while the remaining are anchors , if their location is known *a priori*. All sensors perform one or more types of measurements. In the present work, each of the sensors of the network measures RSS , which is the value of the power with which the transmitted signal reaches the receiver, that is, each tag reports to the CC the received power value from each other tags.

Unlike networks with devices that support GPS, the sensors used in WSNs usually are inexpensive and easy to assemble, although the life of their batteries can be reduced, which it is undesirable. Another advantage of WSNs is mobility, which for indoor systems has a key role. This allows sensors to detect changing phenomena and it also allows the existence of dynamic targets - dynamic location .

2. State-of-the-Art on Localization Systems

2.1. Types of signals

In the recent years, the increase in interest in technologies that allow mobility for users has boosted the creation of devices equipped with wireless communication technologies. In this subsection, the main focus will be on Radio Frequency (RF) signals, since this type is used in this study.

2.1.1. Radio Frequency. Some of the most used RF technologies are briefly described below:

- **Radio Frequency Identification (RFID):** this technology is widely used for proximity detection systems [3], e.g., at the gates of the Lisbon metro. It consists of a tag, a reader and an antenna. A tag is embedded in some object, such as a card, that wants to be detected. The reader, which is no more than a transducer (emits energy to power the RFID tag), is able to read the information (ID) provided by the nearest tag.
- **Ultrawideband (UWB):** it is used for transmitting information over a broadband ($>$ 500 MHz) and takes advantage of a large portion of the available spectrum [4]. The UWB signals have a short duration, which makes it possible to filter the original signals from the reflected ones and, therefore, attain better reliability in the measurement of TOF and/or TOA, which results in better accuracy. Another appealing feature is that they can penetrate various obstacles, such as walls, although liquids and metals are sources of interference.
- **Wi-Fi:** the mid-range Wireless Local Access Networks (WLAN), operating in the ISM band (from 2.4 to 2.485 GHz), have served to support localization schemes, with the addition of a CC. With a range of up to 100 m, IEEE 802.11 is the standard dominant in these networks [5]. Any device with a Wi-Fi antenna can be a network node. This node communicates with one or more Access Points (APs), and vice versa, and records the RSS measurements sending them to the CC, which determines the its location.
- **Bluetooth:** the presence of Bluetooth technology is very common on mobile devices. For information transmission, its range is about 15 m with a maximum bit rate around 1 Mb/s. This technology is a lightweight and flexible standard that supports services other than localization systems. However, in itself it is not enough: due to its range, the system is not accurate. It is then necessary to carry out other measurements, such as RSS. With the creation of new applications for healthcare, security, among others, the Bluetooth Low Energy (BLE) [6], presents a considerable reduction in energy consumption, continuing to operate in the ISM band.

2.2. Metrics

To determine the location of all tags on the network, the localization systems need measurements that the latter take. These measurements are based on several types of metrics.

- **Angle of Arrival (AOA):** it is the angle of the signal intercepted by the receiver [7]. The direction of a signal can be calculated by exploring and detecting the phase difference between the antennas of the antenna array at the receiver. The angle is calculated by comparing the direction of the incoming signal with a reference orientation.
- **Time of Flight (TOF):** it's the time elapsed during the propagation of a signal since it was sent to the receiver. Its most desirable feature is that, if it were possible to measure with precision and synchronism, the distance calculation would be very easy: calculate the product between the measured TOF and the propagation speed. Due to synchronization limitations in consumer-oriented technologies, another metric has been created, the two-way TOF [8].
- **Received Signal Strength (RSS):** it is the power with which the signal emitted by a source reaches the receiver. It can be measured by any device that has its own antenna. Observations are used to determine the distance between the object to be located and a beacon, given the relationship between distance and power. However, RSS varies over time due to fading by multipath and other phenomena, especially in indoor scenarios. The greater the distance the greater the attenuation.

2.3. Multipath and Fading

Radio waves do not remain constant over time. They are influenced by the objects they encounter on their path and can suffer reflection which leads to the existence of several paths from an emitter to a receiver - multipath propagation [9].It results in attenuation or amplification of the original signal. The wave can also undergo diffraction, which results in splitting the signal into several, or scattering, when the it hits an uneven surface (fig.1). When hitting or passing

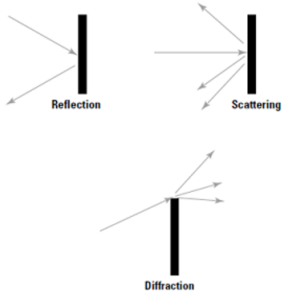


Figure 1. Reflection, Diffraction and scattering [10]

through objects, the waves incur in energy and amplitude

loss due to destructive interference, which is called fading. The physical properties of the medium also influence the signals; it's almost impossible to design an algorithm capable of predicting all the losses during propagation, starting with the great complexity of the input variables that would be necessary.

When the propagation medium is air and the emitter and receiver are in LoS (free space propagation), the Friis formula can be used to calculate the received power , in dB

$$P_{R_{[dB]}} = P_{E_{[dB]}} + G_{E_{[dB]}} + G_{R_{[dB]}} - L_{[dB]} - \gamma 10 log_{10}(\frac{\lambda}{4\pi d}) \quad (1)$$

in which P_R and P_E correspond to the power measured at the receiver and the output of the sender, respectively, G_R and G_E to the antennas gains, d the distance between them, λ the signal wavelength, L to losses and γ to the attenuation coefficient, Path Loss Exponent (PLE). In outdoor scenarios, the PLE is usually equal to 2, being higher in indoor scenarios (higher attenuation). The signal is being attenuated since it left the source and the Path Loss (PL) can be calculated from (2)

$$PL_{[dB]} = PL_{0_{[dB]}} + \gamma 10 log_{10}(\frac{d}{d_0}) \quad (2)$$

where d_0 is the reference distance for which the power is known. However, as already mentioned, the signal in the present work is strongly affected by multipath and fading, which means that it is not possible to know the correct value of PLE and the equations (1) and (2) cannot be used directly.

2.4. Algorithms

Detecting signals or obtaining certain metrics is not sufficient for estimating distances between nodes. Calculations are necessary and for that algorithms are used.

2.4.1. Range-based. These algorithms make use of the measurements acquired through the sensors. To perform the localization different techniques can be used, such as trilateration (or triangulation, when using AOA), accompanied by others that improve the accuracy of location [11]. Range-based algorithms doesn't require complex on-site settings, it consists in the tags deployment. This way the installation process is easier and quicker compared to Range-free.

- **Geometric - Trilateration.** Using this algorithm the measurements made, namely RSS, can be translated into distances using a propagation model. By estimating the distance from a node to at least 3 other nodes (not collinear and with known position) it is possible to find the location of the node in 2D: the intersection between the center circumferences in each beacon and radius equal to the respective estimated distance (fig.2). Due to the RSS variations, impossible to control, the measurements can contain a lot of noise. Increasing the number of beacons increases its accuracy, however, it also increases the area of uncertainty where the target may be. Considering N beacons used (N $\geq$ 3), a target A and the estimated distance d_N (from the target to beacon) using a propagation model, a system of N quadratic equations (3) is obtained,

$$\begin{cases} d_1^2 = (x - x_1)^2 + (y - y_1)^2 \\ d_2^2 = (x - x_2)^2 + (y - y_2)^2 \\ ... \\ d_N^2 = (x - x_N)^2 + (y - y_N)^2 \end{cases} \quad (3)$$

where x_N and y_N correspond to the coordinates of each beacon, $P_N(x_N, y_N)$. As the intersection of the N equations isn't a single point, it is then necessary to find the most likely position of the target, P_A, within that area. This is one of the reasons why geometric algorithms are often complemented with Optimization-based algorithms.

- **Optimization-Based - LLSQ** One of the algorithms proposed in [11] was the Linear Least Squares (LLSQ) which consists of linearizing a system and solving it in the sense of least squares, which is obtained by subtracting each of the N equations with the remaining N-1 . In R^2, the obtained linear system can be described by (4)

$$A(P_A) = b \quad (4)$$

where P_A is the targets unknown position

$$A = \begin{bmatrix} x_1 - x_N & y_1 - y_N \\ x_2 - x_N & y_2 - y_N \\ \cdots & \cdots \\ x_N - 1 - x_N & y_N - 1 - y_N \end{bmatrix} \quad (5)$$

$$b = \frac{1}{2} \begin{bmatrix} x_1^2 - x_N^2 + y_1^2 - y_N^2 + d_1^2 + d_N^2 \\ x_2^2 - x_N^2 + y_2^2 - y_N^2 + d_2^2 + d_N^2 \\ \cdots \\ x_{N-1}^2 - x_N^2 + y_{N-1}^2 - y_N^2 + d_{N-1}^2 + d_N^2 \end{bmatrix} \quad (6)$$

The solution found is expressed by 7

$$P_T = (A^T A)^{-1} A^T b \quad (7)$$

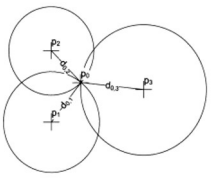


Figure 2. Triangulation with 3 anchors, located in p_1, p_2 and p_3, and a target located in p_0 [12]

2.4.2. Range-free. These solutions do not use measurements directly to infer distances. The measurements are compared with a set of reference values (taken *a priori*) and from there the location is found. For this reason, these algorithms may have to use complex processes, which can be expensive and time consuming.

- **Fingerprinting:** requires two phases: offline and online. In the first one the space is mapped: samples of RSS are collected and associated with coordinates, forming a 'radio map' of the entire space. During the online phase, the system compares the measured observations with the ones on the radio map to determine where the target is located. The cost function c minimization can be solved using probabilistic methods, Support Vector Machines (SVM), among others. It is assumed that the RSS samples associated with the same site do not change over time, which constitutes a great challenge since it is known that the signals are affected by diffraction, reflection, scattering, etc.

3. Multidimensional Scaling

Multidimensional Scaling (MDS) emerged in the Psychology community, and is still used today by many professionals. It was created to be a model that would visually explain how people form opinions [13] as well as relationships between events. The objects of study are represented in the plan as points. In fig.14b it appears that the 'Murder' is closer to the 'Assault' than 'Robbery', which means the correlation between 'Murder' and 'Assault' is higher than the one between 'Murder' and 'Robbery'. The input of the MDS algorithm to produce the previous plot, proximities or dissimilarities, were the correlations. The designation of dissimilarity/proximity between two points comes from the fact that the greater the value of this input variable the smaller is the distance between the points.

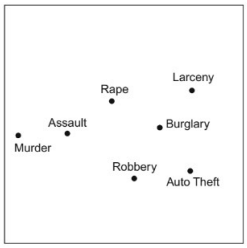


Figure 3. Proximities plot.

3.1. Non Metric MDS

Non Metric MDS is the most well-known MDS model, given its flexible character. It admits as proximities, p_{ij}, any type of metric or value derived from it, as long as it reflects a relationship between the data of the sample taken. The only condition imposed on this metric is that it must be ordinal, the distances between the points of the configuration X obtained by MDS, d_{ij}, will be ordered according to p_{ij} ((8) and (9)). Although d_{ij} distances are calculated as Euclidean distances, there is always a strong chance that they cannot be interpreted as actual distances but only as proximity between points. In the present work the proximities p_{ij} are the symmetric of the RSS values taken between the pair ij.

$$f : p_{ij} \rightarrow d_{ij}(X) \tag{8}$$

$$f : p_{ij} < p_{kl} \rightarrow d_{ij}(X) \leq d_{kl}(X) \tag{9}$$

To reach from the proximities p_{ij}, matrix P, or dissimilarities δ_{ij}, matrix Δ, up to distances d_{ij}, one has to solve a sequence of operations. It starts by centering the square of the Δ matrix ($n \times n$ matrix where n is the number of points), which prevents the configuration from "wandering" through the plane, obtaining B_{Δ}, by the equation (10). In this, J is obtained from (11).

$$B_{\Delta} = -\frac{1}{2} J \Delta^2 J \tag{10}$$

$$J = I - n^{-1} \mathbb{1}\mathbb{1}' \tag{11}$$

$$B_{\Delta} = Q \Lambda Q' \tag{12}$$

$$X = Q_{+} \Lambda_{+}^{\frac{1}{2}} \tag{13}$$

That said, X ($n \times m$ matrix being m the dimension of the plan) is obtained through (13), where Q_{+} corresponds to the first m columns from Q and $\Lambda_{+}^{\frac{1}{2}}$ to the first positive m eigenvalues of B_{Δ} eigen-decomposition (12). If the Δ matrix were filled with Euclidean distances, all eigenvalues would be positive or negative but very close to zero. The main advantage of using the decomposition (12) is that the configuration X is the one that minimizes the difference between Δ and the matrix containing the distances between the points in X.

3.2. Classic MDS vs Iterative MDS

The classic form of solving an MDS algorithm assumes that the surroundings are Euclidean distances and boils down to solving the algorithm. Ideally, the proximities p_{ij} and the distances between the points of the obtained configuration $d(X)_{ij}$, disparities, would be the same. However, when computing the difference between them, what is left is a matrix with values other than zero ($P - D$), except the diagonal ($d_{ii} = 0$). Given what p_{ij} represents in this model, the larger the module of the values of $P - D$, the further the configuration obtained from reality will be.

Using a MDS algorithm on an iterative way, it starts from a starting configuration (random or not) and focuses on minimizing a cost function that leads to the minimization of $P-D$. The points obtained are moved each iteration, very slowly, until a convergence value of the chosen cost function is obtained.

The most popular cost function is Stress. This can be obtained by the expression (14), where w_{ij} are weights corresponding to each dissimilarity δ_{ij}. It is a residual sum of squares, positive and the lower its value the closer to reality is the configuration obtained (or not, if this value is a local minimum). Stress can serve as 'fit index', it aims

to assess whether the configuration approximates reality or not, given the dissimilarities provided.

$$\sigma_r(X) = \sum_{i<j} w_{ij}(\delta_{ij} - d_{ij}(X))^2 \tag{14}$$

3.3. SMACoF

This algorithm [14] consists of minimizing a complex function, in this case Stress, through iterative majorization. The main idea is to take a function that is (literally) complicated ($f(x)$) and replace it with a simpler one ($g(x, z)$) and greater or equal ($f(x) \leq g(x, z)$) regardless of the number of variables, such that z has a fixed value. These algorithms are less likely that the solution found has been "stuck" in a local minimum than others. SMACoF algorithm has as input

Algorithm .1: SMACoF

Data: X_0, Δ
Result: Points configuration, X
k=0;
ϵ=1e-4;
compute $d_{ij}(X_0)$;
apply PAVA in $d_{ij}(X_0)$: $\hat{d}_{ij}$;
standardize $\hat{d}_{ij}$;
compute Stress: $\sigma_r^{(0)} = \sigma_r(\hat{d}, X^{(0)})$;
$\sigma_r^{(-1)} = \sigma_r^{(0)}$;
while *k=0 ou ($\sigma^{(k-1)} - \sigma^{(k)} > \epsilon$ e k maximum of iterations)* **do**
  compute Guttman transformed: $X^k = n^{-1}B(Z)Z$;
  compute $d(X^{(k)})_{ij}$;
  apply PAVA in $d_{ij}(X)$: $\hat{d}_{ij}$;
  $\delta_{ij} = \hat{d}_{ij}$;
  standardize $\hat{d}_{ij}$;
  compute Stress: $\sigma_r(\hat{d}, X^{(k)})$;
  $Z = X^{(k)}$;
end

an initial configuration, X_0, which may have been generated through the classic, random or other MDS provided by the user and also the dissimilarities δ_{ij}, Δ, which led to X_0. The distances $d_{ij}(X_0)$ are the Euclidean distances between the points of the latter. Another way is to denote the distances $d_{ij}(X_0)$ as a transformation of δ_{ij}, as in (15). Thus, $d_{ij}(X_0)$ is expected to have "weak monotonicity" with respect to δ_{ij}, that the distances $d_{ij}(X_0)$ are monotonically non-decreasing in relation to δ_{ij}, (16).

$$d_{ij}(X_0) = f(\delta_{ij}) \tag{15}$$

$$if \quad p_{ij} < p_{kl} \quad then \quad d_{ij}(X) < d_{kl}(X) \tag{16}$$

In order to guarantee (16) and later to obtain the disparities $\hat{d}_{ij}$, a Pool-Adjacent-Violators Algorithm (PAVA), is used and it is also with this algorithm that the distances are "untied"; sometimes it can happen that there are two equal dissimilarities ($\delta_{ij} = \delta_{kl}$) and these can translate into equal disparities or not ("primary approach") which gives more freedom to the data. When SMACoF ends, it is assumed (but not guaranteed) that the obtained point configuration corresponds to the global minimum of the Stress function.

3.4. MDSCALE - MATLAB

In this work the function 'mdscale' provided by MATLAB was also used. In this function, an algorithm is implemented, which uses MDS. The big difference between this and SMACoF is in the process of updating the point configuration. That is, to minimize the Stress, MDSCALE resorts to the "Polak-Riviere" line search method [15].

4. Procrustes Analysis

The goal of Procrustes Analysis is to fit a configuration of points Y in a configuration X, that is, the points of Y occupy the same positions as those of X. This can be done through rigid transformations (rotation, reflection, translation and expansion) through minimization of the sum squared error (SSE), $L(T) = ||X - YT||$ being T some linear transformation. After some simplification one can reaches

$$L(T) = c - 2trX'YT \tag{17}$$

Algebraically it can be translated to the following steps:

- Computation of matrix C

$$C = X'JY \tag{18}$$

- Singular value decomposition of C

$$C = P\phi Q' \tag{19}$$

- Computation of optimal rotation matrix , T

$$T = QP' \tag{20}$$

- If $det\ T < 0$ it is a sign that is necessary to perform a reflection. In order to do that it equals matrix Q to its symmetric,$Q = -Q$, and recomputes T;
- Computation of the scaling factor s

$$s = \frac{trX'JYT}{trY'JY} \tag{21}$$

- Computation of the optimum translation vector t

$$t = n^{-1}(X - sYT)'\mathbb{1} \tag{22}$$

- Compute the configuration Z

$$Z = sYT + t \tag{23}$$

The matrix Z is the best possible approximation of Y points to X points.

5. Thales Experimentation

The experience developed by Thales for the "Viral" exhibition at Pavilhão do Conhecimento Ciência Viva in 2015/16 aimed to recreate a propagation of a virus in the community. It followed the scheme in Fig.4. The marker tags behave in this work as anchors, except for a single one that acts as a target (there were many targets in the Thales setup, but in the present study there is only one). The sink corresponds to the base station for the tags (is the bridge between them and the CC). As demonstrated in [16], the

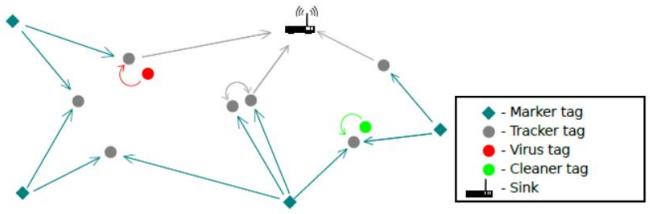


Figure 4. Packets exchange between tags and base station [16].

results using Thales' approach are too far from optimal. In the following text a different approach will be described, using the same hardware that Thales and [16] did.

6. Experimental Work

6.1. Setups

Here, the deployment of a specific configuration of anchors in a specific area is denoted as a "setup". It covers the implantation of the sensors and the communication between them and with the CC, through an Open Beacon base station, to acquire the RSS measurements.

6.2. Collaborative Scenario

In all setups all tags communicated with each other, with a transmission power of 4 dBm.

6.3. Chosen Methodologies

This subsection illustrates the deployment made in ISR-Taguspark, Fig.5. Here, all the anchors were placed at the same height with respect to the floor and the only time they might not be in LoS would be when the target interposes between them. In this room there is a lot of glass and some polished and unpolished metal structures, as well as the existence of a water tank.

The coordinates occupied by the anchors can be seen in Fig.6, considering a virtual space of 5.4 m x 5.45 m around the anchor configuration.

The target described a square trajectory around the anchors for about 30 seconds.

Figure 5. Anchors configuration - DSOR.

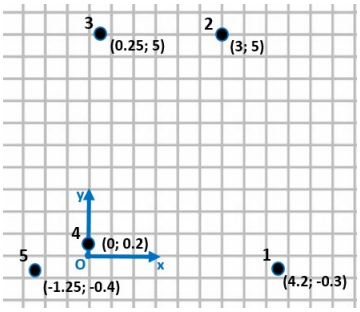


Figure 6. Anchors coordinates, with origin in tag 4.

6.3.1. Power vs. Distance. Considering the equation (1), it is easily inferred that if both variables are on the logarithmic scale, their relationship is linear and decreasing

$$P_{ij} = a - \gamma d_{ij} \tag{24}$$

where P_{ij} is the RSS value between tags i and j and d_{ij} the distance between them.

However, the plot in Fig.7 shows that condition (24) is far from being verified in practice, and it is almost impossible to discern any analytical relationship between both variables.

6.3.2. Cleaning, Completion and Signaling. One of the objectives of this study is to use the MDS algorithm as a mean to improve the robustness of relative positioning using spatial cues (RSS values) that are only weakly correlated with physical distances. It was decided from the beginning that its entry, the Δ dissimilarity matrix, would be measurements of RSS, power values. To calculate the position of the target every second, it is necessary to have a matching value of RSS and only one between each pair of tags. For that, Algorithm .2 is proposed

- **Cleaning and Completion:** in the cleaning phase it was ensured that there was only one measurement per instant t (second). In case there is more than one, the value of $P_{ij}(t)$ to be saved was the average between them. On the other hand, sometimes there

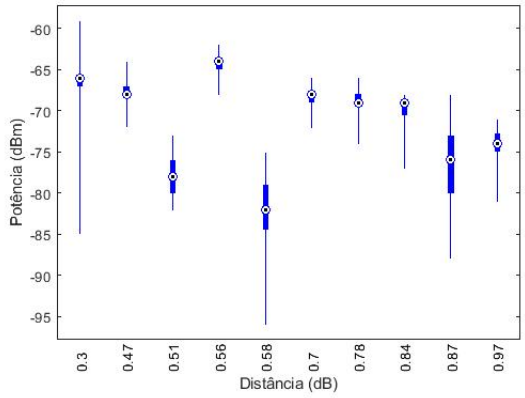


Figure 7. Power variation for each distance (between anchors) - setup "j01".

was no recorded power value; then, in the Completion phase that value was filled. It was assumed that the RSS measurements would have a monotonic behaviour so, for a given $P_{ij}(t \neq 0) = 0$ if the value other than zero immediately before was less than the non-zero value immediately afterwards then, in this time, the powers would increase (increasing monotony) and vice-versa.

- **Signaling:** the target was carried by a person. It was considered that, on average, the normal walking human speed is about 1 m/s. In the case of RSS vectors referring to 'anchor-target' pairs there were sometimes large variations (greater than 10 dB) from instant to instant. Taking into account the real conditions of the experiment an attempt was made to somehow limit the variations. A variable $delta$ was defined, which served as "goal": a measurement of RSS could only be $delta$ units greater or less than the previous one: $P_{ij}(t) \in [P_{ij}(t-1) - delta; P_{ij}(t+1) + delta]$. If $P_{ij}(t)$ was not "within the goal", it would take the average between its current value and $P_{ij}(t-1)$. The $delta$ value considered to be optimal was 2.5 dB.

As can be seen in Fig. 8, the graphs in b) show less abrupt variations, are smoother than those in a), maintaining a shape that resembles their original one.

6.3.3. Kalman Filter. There was no intention to eliminate variations in P_{ij} vectors corresponding to 'anchor-target' pairs, but only to mitigate their fluctuations so that they were not too abrupt and "unreal". On the other hand, in 'anchor-anchor' pairs the variations are inconvenient. Since the latter are standing still, RSS measurements would be expected to remain constant. However, this is not the case, as these are contaminated with noise, sources of constructive or destructive interference, and other unquantifiable losses. It is necessary to infer power values of anchor pairs closer to the real ones and more reliable than those obtained by Algorithm .2. One of the most frequently used filters

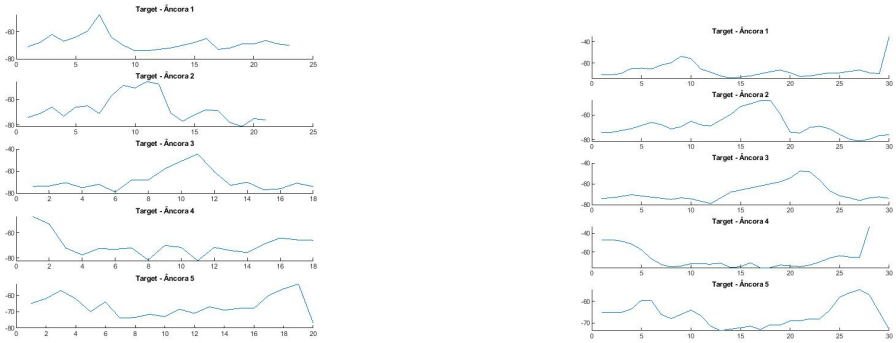
Algorithm .2: Cleaning, Completion and Signaling

```
Data: P_ij, time, delta
Result: P_ij
// Cleaning
for each instant t do
    if there is more than one P_ij(t) then
        p = average(P_ij(t));
    end
end
// Completion
if P_ij(1) = 0 then
    equal all P_ij(t) equals to zero and previous to
     the 1st P_ij(t) different from zero,to the 1st
     P_ij(t) different from zero;
end
c=0;
for each instant t do
    if P_ij(t) = 0 then
        if P_ij(t-1) = 0 then
            c = c+1;
        end
        else
            c=2;
        end
    end
    if P_ij(t) ≠ 0 e c ≠ 0 then
        dif = (P_ij(t) - P_ij(t-c))/3; while c do
            P_ij(t-c-1) = P_ij(t-c-2)+ dif
            c = c-1;
        end
        c=0;
    end
end
// Signaling
for for each instant t do
    if P_ij(t) - P_ij(t-1) > delta then
        P_ij(t) =average(P_ij(t), P_ij(t-1));
    end
end
```

for predicting and approximating values to reality is the Kalman Filter [17]. The filter used was a very simplified version of Kalman's.The filter was divided into two phases: prediction and updating. It also assumes a x_k 'state' and a z_k 'observation', where k indicates the instant.

The state x_k is described by $x_k = ax_{k-1} + bu_{k-1} + w_{k-1}$ with a being the transition state model applied to x_{k-1} (relates the current and the previous states), b_k the variable that relates the control input to the state x_{k-1} applied to the latter u_{k-1} and w_{k-1} is the process noise that is assumed to have normal distribution and covariance q (mean equals to zero). At each instant of time k a observation z_k is obtained: $z_k = hx_k + v_k$ where h related the real state x_k with the observation z_k and v_k is the observation noise, which is also normal distributed its covariance is r.

- **Prediction phase:** being $\hat{\Sigma}_{apriori}$ the covariance of



(a) (b)

Figure 8. Plots P(d) before (a) and after (b) Algorithm .2 was applied - horizontal axis: time(s); vertical axis: power(dBm).

the error

$$\hat{x}_{apriori} = a\hat{x}_{k-1} + bu_{k-1} \tag{25}$$

$$\hat{\Sigma}_{apriori} = a\hat{\Sigma}_{k-1} + q \tag{26}$$

- **Update phase:** the optimal Kalman gain K_k is estimated (27) as well as the state (28) and $\hat{\Sigma}_k$ (29).

$$K_k = \hat{\Sigma}_{apriori}h'(h\hat{\Sigma}_{apriori}h' + r)^{-1} \tag{27}$$

$$\hat{x} = \hat{x}_{apriori} + K_k(z_k - h\hat{x}_{apriori}) \tag{28}$$

$$\hat{\Sigma}_k = (1 - K_k h)\hat{\Sigma}_{apriori} \tag{29}$$

Considering each pair of anchors as a system and that it remains immutable, it is assumed that $a = 1$ and that the only item exchanged between nodes is the RSS measurement, with no control entry and $u = 0$. The variance of the process $q = 10^{-5}$ and r was taken as the variance of each power vector between anchors. For initial values of state and observation, $\hat{x}_0$ and $\hat{\Sigma}_0$, the average of the obtained measurements was taken and 1, respectively. Given that, the prediction phase simplified to (30) and the update to (31).

$$\begin{cases} \hat{x}_{apriori} = \hat{x}_{k-1} \\ \hat{\Sigma}_{apriori} = \hat{\Sigma}_{k-1} + q \end{cases} \tag{30}$$

$$\begin{cases} k = \frac{\hat{\Sigma}_{apriori}}{\hat{\Sigma}_{apriori}+r} \\ \hat{x}_k = \hat{x}_{apriori} + k(z_{k-1} - \hat{x}_{apriori}) \\ \hat{\Sigma}_k = (1-k)\hat{\Sigma}_{apriori} \end{cases} \tag{31}$$

The application of the filter proved to be useful. Since in this assembly there are 10 'anchor-anchor' pairs it would be time consuming to present all comparisons, but as a proof in Figure 9 it is possible to compare the RSS values measured between the anchors 1 and 3 before applying the filter (a) and after (b). Although there is no horizontal line, the variation that is observed is very small ($< 1dB$) so the values can be considered constant, approaching what would be ideal but still preserving a certain experimental component.

6.3.4. MDS Input. After the RSS values (related to the pairs of sensors) are prepared, the created algorithm using MDS begins. It receives as input a matrix of dissimilarities Δ and each entry δ_{ij} corresponds to the symmetric RSS value P_{ij}, that is, $\Delta = -P_{ij}$.

6.3.5. MDSCALE VS SMACoF. To obtain at each instant of time a configuration of points X from which it is possible to extract the coordinates of the target, to visualize its trajectory, the two algorithms SMACoF and MDSCALE were compared; both had the same Δ dissimilarity matrix as input.

- **Effect of Procrustes Analysis:** it was found that this analysis positively contributed to the improvement of the results. To simplify, in order to support the latter sentence, only the anchors were taken into account. As seen in Figs.10 and 11, the Procrustes Analysis was beneficial to both algorithms usages.
- **Impact of Target Addition:** as it has already been verified, the coordinates of the anchors change over time. It would be important that with the addition of the 'target-anchor' dissimilarities the anchors (approximately) maintain their configuration. Only then can it be considered minimally reliable to equate visualizing the trajectory of a target. In Figs. 12 and 13 the shown points have already undergone Procrustes analysis. After observing these, it was considered that, even though they are not the same, they still seem to be concentrated in 5 different zones. In fact, there seems to be a greater demarcation of those areas, which may indicate that a greater number of nodes tags improves the quality of the solution.
- **Procrustes Analysis - Target:** having proved to be a useful tool for this work, the intention was to extend Procrustes Analysis to the coordinates of the target as well. This constitutes a problem since this analysis presupposes a real configuration for which another will be adapted and in the case of the target its position is not known *a priori* (contrary to the anchors). The solution found was, at each instant,

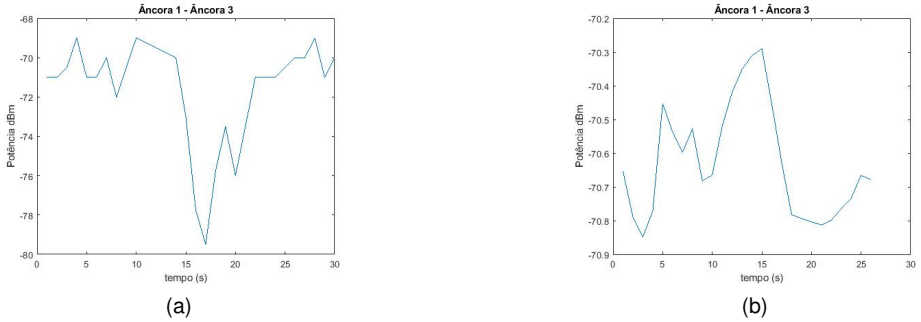


Figure 9. RSS values between anchors 1 and 3 before (a) and after (b) the approximated Kalman Filter.

to compute the transformation (23) that fitted the anchors configuration obtained by MDS, and apply it to the target. The graphics that were obtained are shown in Fig.14.

- **Distances Filtering:** the filtering applied to the RSS values was intended not to be too restrictive, since it is complicated to define what is right or wrong when it comes to the RSS metric. And so it is evident that errors will continue to exist. Once again, a light approach was chosen to perform a filtration of the distances.
  At this point, the target coordinates, at each instant, are already computed.
  The filter used is almost another type of filtering, but now applied to distances. The walking speed of the human being, of approximately 1 m/s, was also considered. This means the distance between a point and its consecutive one should not be greater than 1 m. In Algorithm .3 it is illustrated that, if the previous condition does not happen, a weighted average is made, in which the last estimated point $\hat{X}_{target(t-1)}$ has a weight four times greater than the point being analyzed $X_{target(t)}$. It was assumed that the $X_{target}(1)$ (first point in time) was always correct, which may not be true. The trajectories obtained can be seen in Fig15. Although not perfectly, the trajectory of a) (MDSCALE) traces a square-like shape: it starts with something that appears to be u-turns, but then it goes to something that resembles the real path. The trajectory in b) (SMACoF) fell slightly behind, although the sense of direction feels right.
  The Thesis provides experimental results for deployments in other areas of IST-Taguspark.

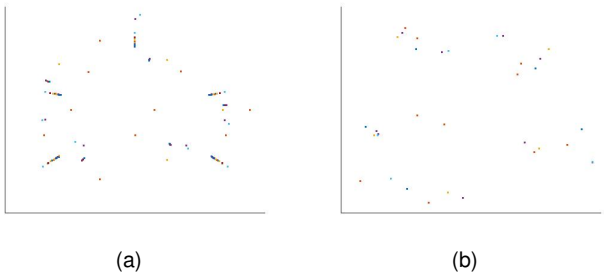


Figure 10. Anchors' coordinates: a) before and b) after Procrustes Analysis - MDSCALE.

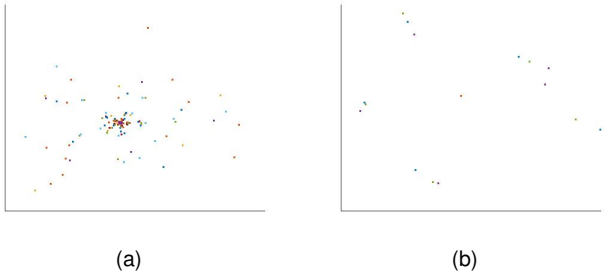
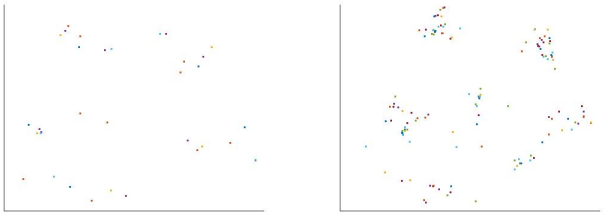


Figure 11. Anchors' coordinates: a) before and b) after Procrustes Analysis - SMACoF.

7. Conclusion

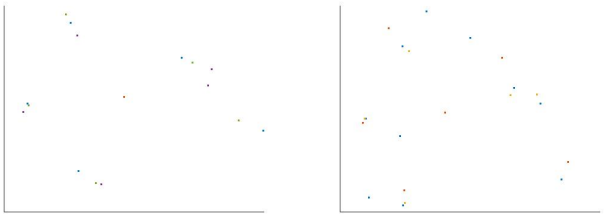
Numerical values of the stress cost function played only a minor role when evaluating the results. This happened because, although stress is an indicator of "goodness-of-fit" it is not an absolute indicator. As a matter of fact, in the literature it is pointed out that: a low stress (< 1 %, does not necessarily indicate a good MDS solution [18]. It is always necessary to take into account the data being studied, because if it is known it contains many errors, as in this work, it's not a good idea to be guided exclusively by a measure that depends on them. Based on this, it was considered that the best weapon for evaluating the solution would be the human eye, since the actual assembly is known.

Better performance was observed using the MDSCALE algorithm. The trajectory obtained is quite similar to the real one, considering the enormous amount of noise contained in the RSS measurements. With respect to SMACoF, when it did not converge to a local minimum, the corresponding coordinates were discarded. This occured several times which resulted in few target coordinates and a worse result. It is



(a) (b)

Figure 12. Anchors' coordinates: a) before and b) after target addition - MDSCALE.



(a) (b)

Figure 13. Anchors' coordinates: a) before and b) after target addition - SMACoF.

not clear that if it had not happened, it would have produced good results.

8. Future Work

The most obvious issue: is to test the experimental part with other equipment (sensors), more powerful and/or one the supports wi-fi, for instance. The measurements could be better or not.

There are numerous methodologies available which can be adapted and applied to the problem under study. It will be possible, with the same equipment, to test new filters, invented or those already available in the literature, which can allow the RSS values to be approximated to what would be the correct ones. This, in addition to adding more sensors to the setups should make it possible to single-out solutions that best adapt to reality.

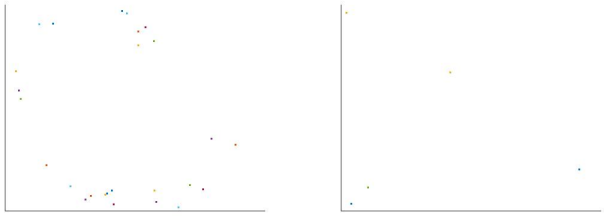
Finally, the most important aspect will be to compute the targets path (location) in real time, using adequate methodologies.

References

[1] Misra, Pratap and Burke, Brian P and Pratt, Michael M, *GPS performance in navigation*, Proceedings of the IEEE, 1999

[2] *Thales*, https://www.thalesgroup.com/en, 2020

[3] Cruz, Roselia Dela and Pedrasa, Jhoanna Rhodette, *RSSI-based Localization in an Emulated Active RFID System*, 2019 IEEE 10th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), 2019



(a) (b)

Figure 14. Target positions through a)MDSCALE and b)SMACoF.

Algorithm .3: Filter used on coordinates

Data: $X_{target}(t)$, time, velocity
Result: $\hat{X}_{target}(t)$
$\hat{X}_{target}(t)$ = vector 1 x length(time);
$\hat{X}_{target}(1) = X_{target}(1)$; **for** *each instant* t **do**
  if —$\hat{X}_{target}(t-1)$ - $X_{target}(t)$— ¿ *velocity* **then**
    $\hat{X}_{target}(t) = \frac{(4*\hat{X}_{target}(t-1)+X_{target}(t))}{5}$;
  end
  else
    $\hat{X}_{target}(t) = X_{target}(t)$;
  end
end

[4] A. F. Molisch, *Ultrawideband propagation channels-theory, measurement, and modeling*, IEEE transactions on vehicular technology, 2005

[5] Crow, Brian P and Widjaja, Indra and Kim, Jeong Geun and Sakai, Prescott T, *IEEE 802.11 wireless local area networks*, IEEE Communications magazine, 1997

[6] Maratea, Antonio and Salvi, Giuseppe and Gaglione, Salvatore, *Bagging to Improve the Calibration of RSSI Signals in Bluetooth Low Energy (BLE) Indoor Distance Estimation*, 2019 15th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS), 2019

[7] Niculescu, Dragos and Nath, Badri, *Ad hoc positioning system (APS) using AOA*, IEEE INFOCOM 2003. Twenty-second Annual Joint Conference of the IEEE Computer and Communications Societies, 2003

[8] Pettinato, Paolo and Wirström, Niklas and Eriksson, Joakim and Voigt, Thiemo, *Multi-channel two-way time of flight sensor network ranging*, European Conference on Wireless Sensor Networks, 2012

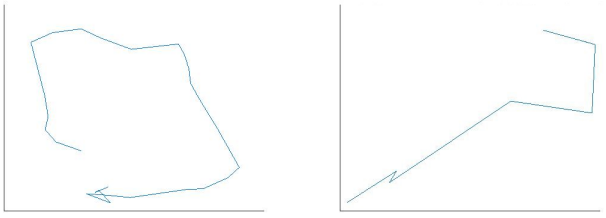
[9] Mahender, Kommabatla and Kumar, Tipparti Anil and Ramesh, KS, *Analysis of multipath channel fading techniques in wireless communication systems*, AIP Conference Proceedings, 2018

[10] A. Malik, *RTLS for Dummies*, John Wiley & Sons, 2009

[11] Will, Heiko and Hillebrandt, Thomas and Kyas, Marcel, *The FU Berlin parallel lateration-algorithm simulation and visualization engine*, 2012 9th Workshop on Positioning, Navigation and Communication, 2012

[12] Hoene, Christian and Willmann, Jorg, *Four-way TOA and software-based trilateration of IEEE 802.11 devices*, 2008 IEEE 19th International Symposium on Personal, Indoor and Mobile Radio Communications, 2008

[13] Borg, Ingwer and Groenen, Patrick JF and Mair, Patrick, *Applied multidimensional scaling*, Springer Science & Business Media, 2012



(a) (b)

Figure 15. Target trajectories - a)MDSCALE and b)SMACoF.

[14] Borg, Ingwer and Groenen, Patrick JF, *Modern multidimensional scaling: Theory and applications*, Springer Science & Business Media, 2005

[15] Aarti Singh, *Conjugate Gradient Descent*, Carnegie Mellon, School of Computer Science, 2020

[16] Rato, V., MSc Thesis *RSS-based Indoor Localization using a Network of Open-Source Wireless Tags*, Instituto Superior Técnico, 2018

[17] Welch, Greg and Bishop, Gary and others, *An introduction to the Kalman filter*, 1995

[18] Wagenaar, WA and Padmos, Peter, *Quantitative interpretation of stress in Kruskal's multidimensional scaling technique*, British Journal of Mathematical and Statistical Psychology, 1971