

Article

Localization Algorithm for 3D Sensor Networks: A Recursive Data Fusion Approach

Rafaela Villalpando-Hernandez ^{1,*} , Cesar Vargas-Rosales ¹  and David Munoz-Rodriguez ²¹ School of Engineering and Sciences, Tecnológico de Monterrey, Monterrey C.P. 64849, Mexico; cvargas@tec.mx² Aerospace Development Center, Instituto Politécnico Nacional, Ciudad de México C.P. 06000, Mexico; dmunoz@itesm.mx

* Correspondence: rafaela.villalpando@tec.mx

Abstract: Location-based applications for security and assisted living, such as human location tracking, pet tracking and others, have increased considerably in the last few years, enabled by the fast growth of sensor networks. Sensor location information is essential for several network protocols and applications such as routing and energy harvesting, among others. Therefore, there is a need for developing new alternative localization algorithms suitable for rough, changing environments. In this paper, we formulate the Recursive Localization (RL) algorithm, based on the recursive coordinate data fusion using at least three anchor nodes (ANs), combined with a multiplane location estimation, suitable for 3D ad hoc environments. The novelty of the proposed algorithm is the recursive fusion technique to obtain a reliable location estimation of a node by combining noisy information from several nodes. The feasibility of the RL algorithm under several network environments was examined through analytic formulation and simulation processes. The proposed algorithm improved the location accuracy for all the scenarios analyzed. Comparing with other 3D range-based positioning algorithms, we observe that the proposed RL algorithm presents several advantages, such as a smaller number of required ANs and a better position accuracy for the worst cases analyzed. On the other hand, compared to other 3D range-free positioning algorithms, we can see an improvement by around 15.6% in terms of positioning accuracy.

Keywords: recursive localization; position information fusion; 3D sensor networks



Citation: Villalpando-Hernandez, R.; Vargas-Rosales, C.; Munoz-Rodriguez, D. Localization Algorithm for 3D Sensor Networks: A Recursive Data Fusion Approach. *Sensors* **2021**, *21*, 7626. <https://doi.org/10.3390/s21227626>

Academic Editors: Raquel Caballero-Aguila and Andrzej Stateczny

Received: 13 September 2021
Accepted: 6 November 2021
Published: 17 November 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

New paradigms such as the Internet of things, Industry 4.0, health monitoring, assisted living and physical and environmental conditions monitoring, among other applications, have been introduced due to the increased reliability, improved autonomy and low-cost trend of sensor networks which enable the market growth for both military and civil applications. Data fusion techniques can be applied to find solutions in several challenges that must be addressed when designing new applications or techniques for sensor networks. Connectivity and network topology in ad hoc and sensor networks change dynamically due to variations in the propagation conditions and mobility, [1]. At the same time, in nowadays network scenarios, position location awareness becomes more relevant in order to achieve network goals such as fast routing, efficient resource management, sensor energy harvesting and reliable deployment of location-aware services. Besides GPS and other GNSS systems meant to provide three-dimensional geolocation for outdoor applications, most existing positioning techniques deal with planar location environments that are not sufficiently suitable for many applications. Although GPS has been in place for many years for several uses, its availability and accuracy are insufficient for indoor applications and other compromised scenarios such as tunnels, mines and high-vegetation environments. Furthermore, in sensor networks, range-based location techniques might not be reliable due to range-noisy estimations obtained from complex physical environments. In real scenarios,

sensors might fail, burn power batteries or be under different sources of interference; so, it is required to design alternative estimation methods to address these random restrictions.

Therefore, data fusion-based techniques represent a crucial alternative for multiple network applications, such as localization. Note that, in sensor positioning, data fusion can be understood as the set of techniques or methods that process or combine data from several sensors in order to obtain a better location estimation of a node of interest [2,3]. Therefore, alternative 3D localization schemes are necessary for applications such as emergency call services in urban and metropolitan environments, museum guides, warehouse stock management, industrial plant monitoring and assisted living, among many others [4].

In the literature, we can find range-based and range-free sensor localization techniques, most of them designed for 2D scenarios. Basically, range-based localization techniques are focused on the improvement of the estimated distances (through TOA, DTOA and RSSI) between anchor nodes (ANs) and the node to locate or node of interest, [5]. Then, available ANs perform classical trilateration and the centroid of the intersected areas is selected as the final location estimation of the node of interest. Some of these proposals use Kalman filters and/or modifications of the Kalman filter in order to improve estimated distances [6–8] and thus the location estimation. Others have focused on the improvement of the centroid calculation of intersected areas, while some authors have proposed some variations such as the weighted centroid calculation [9].

One of the first range-based 3D localization techniques was proposed in [6], where they introduced a method based on two stages. During the first stage, nodes with known locations estimate distances based on the received signal strength indicator (RSSI) to a mobile location-assistant (LA) such as an aircraft, a balloon, a robot, a vehicle, etc. Then, each location-unaware sensor discovers its position by observing the moving, location-aware LA nodes. In the second stage, the location-unaware sensor nodes use an unscented Kalman filter for self-localization. This technique reports the requirement of a high number of known location anchor nodes in order to achieve moderate position errors. For instance, it is reported that, in low complexity scenarios with 200 sensor nodes deployed in a $1000 \times 1000 \times 100$ feet space, they require 5% of anchor nodes in order to achieve position errors of around 11 feet for flat terrains. As they evaluate more complex scenarios, the accuracy is degraded to positioning errors of about 100 feet. Although this method does not rely on network density and topology, its performance depends on the observation of mobile devices that might not be available in some hostile scenarios; additionally, its performance degrades as the terrain complexity increases.

Another popular range-based approach was introduced in [7] for sensor node localization in planar scenarios, using a multidimensional scaling (MDS) technique. An extrapolated concept for 3D scenarios is presented in [10], where the authors propose an MDS-MAP-based localization technique where they calculate the shortest distances between every pair of nodes, then they apply classical multidimensional scaling to the distance matrix. Finally, the relative map with relative location for each node is obtained. In order to transform the relative map into a global map, at least four anchor nodes are needed. The main limitation of this algorithm is a strong dependency on the input information for the distance matrix and the high dependency on the network connectivity. The reported performance of the proposed algorithm considers a scenario with 100 nodes in a $100r \times 100r \times 100r$ (r is the unit length distance) space, varying connectivity. The position error reported is of about $12\%R$ (R is the maximum transmission range) with high connectivity and a high number of anchors (10 anchors).

For comparison purposes, we can observe that the two algorithms just discussed in the previous paragraphs require at least 10 ANs to be implemented, while our proposed algorithm can be implemented using only three ANs. In terms of accuracy, the purpose of this paper is not to find a direct comparison between the worst analyzed cases. However, we achieve position errors of around $10\%r_{\max}$ ($r_{\max}$ is the maximum transmission range) using only three ANs, while, in [6], the authors achieve an accuracy of around $45\%r_{\max}$ and, in [7], the authors achieve $12\%r_{\max}$ for the worst analyzed cases.

In [11], the authors present a range-based (RSSI estimations) sequence localization algorithm where they construct a 3D Voronoi diagram, which is used to divide the operational scenario depending on known location nodes. Then, they formulate a rank sequence table of estimated location nodes (virtual beacons) based on the previous partition. RSSI estimations between nodes are used to correct the measured distances and to fix the location sequence of unknown nodes. The location error is normalized to the transmission range r_0 , achieving a location error of around $0.12r_0$ for eight ANs.

In range-free localization methods, the distance between ANs and the node of interest is based on the hop count criterion; therefore, the position estimation of the node does not depend on the accuracy of the estimated distances between anchor nodes and the node to locate. Most of the range-free localization methods base their approaches on the assumption that the Euclidean distance between nodes is proportional to the number of hops in the connecting route. Then, a heuristic method to estimate the coordinates of the node of interest is applied. In [12], the authors propose a range-free approach, where a position location (PL) method suitable for 3D environments is introduced. This method is based on the convenient deployment of access points (APs) or anchor nodes and on the discretization of the network space. They base distance estimation on the hop count criterion on a Manhattan scenario, [13]. A set of algebraic relations is defined for the node's coordinates based on the distances from the APs to the node of interest. In addition, a maximum a posteriori probability (MAP) criterion is used to improve the estimated distances and performance of the algebraic method. They deploy from 300 to 600 nodes in a $10l \times 10l \times 10l$ space (l = Manhattan step) and achieve localization errors of about $0.83r_{\max}$ (where $r_{\max}$ is the maximum transmission range), which can be considered a good result for the free range/multihop scenarios analyzed.

Comparing the proposed method against the last 3D positioning methods summarized, we achieve an improvement of around 15.6% for similar scenarios, using 4 ANs, 600 deployed nodes and hop count as distance estimation criterion.

In this paper, we propose a novel localization technique named Recursive Localization (RL) algorithm, suitable for 3D sensor node positioning. In the proposed RL algorithm, the initial node of interest's coordinates are estimated by classical trilateration techniques, after a reference plane is constructed based on the locations of only three available ANs. Then, the recently located node n_j becomes a new anchor node and one of the original ANs is now considered to be a node of interest to be located. Since the actual locations of the ANs are known, we obtain the difference between the AN's estimated location using n_j and the AN's true location by least squares. Finally, we use this difference (called recursive correcting vector, or **CV**) to recalculate the location of the node of interest n_j . The algorithm is explained in detail in the following sections. Note that, when more than three ANs are available, we can take advantage of having several reference planes to improve the first location estimation of n_j (this is the starting point of the RL). Note that observations are prone to impairments due to either the error in distance estimations, anchor node movements or the absence of direct connectivity from sensor nodes to ANs, rendering in ad hoc routes that may depart from a line of sight (LOS) trajectory. The performance of the proposed methodology is assessed for both LOS and multihop scenarios. Different scenarios are analyzed, where several network factors are modified, such as the number of ANs, their relative positions, the number of possible reference planes and the maximal transmission range, as well as the node densities for multi-hop scenarios. The performance in a noisy environment is considered. The proposed RL algorithm provides a significant improvement of the location accuracy for all the scenarios analyzed, without the need of additional information or processing for distance estimation improvement with a very low number of ANs.

In Table 1, we summarize the characteristics of the main positioning methods described above in order to clarify the differences between the state of the art and the proposed method. We see further that the proposed RL algorithm can be used under both the range-based or free-range-based approach.

Table 1. Summary of localization methods.

Localization Method	Requirements	Accuracy
Classical trilateration with Kalman filter [6] (range-based).	RSSI estimations. At least 5% of nodes must be ANs, starting with 200 nodes.	Achieves an estimation error of around 100 feet in $1000 \times 1000 \times 100$ feet-complex scenarios.
Multidimensional scaling (MDS)-based location techniques [7,10] (range-based).	Distance estimations, high node connectivity—10% of nodes must be ANs to achieve their best reported result.	Achieves an estimation error of around 0.12R in a $100r \times 100r \times 100r$ scenario with 10 anchor nodes.
Voronoi-based PL algorithm [11] (range-based).	RSSI estimations, high number of ANs.	Achieves an estimation error of around $0.12r_0$ for 8 ANs, for $r_0 = 8$ m
Heuristic localization method [12] (free-range-based).	Hop count distance estimation, specific deployment of ANs. Four ANs required.	Achieves an error estimation of around $0.83r_{\max}$ for a network with 600 nodes and 4 ANs.

In [14], the authors formulate a taxonomy for localization algorithms' classification. They also present a study of different main localization algorithms along with their improved versions. The challenges and research approaches of localization in sensor networks emerging applications are also discussed.

In Section 2, we introduce the scenario used for position location with the description of the Recursive Localization algorithm. Section 3 contains the simulation results showing the performance as several parameters are varied and, in Section 4, the conclusions are discussed.

2. Positioning System

In a basic network set up, we want to determine the location of a node in a defined 3D scenario. The node is considered to be able to establish direct or multihop routes to several ANs that provide global connectivity. The reachability of the node of interest to at least three anchor nodes, AN_1 , AN_2 and AN_3 , with corresponding known coordinates (x_k, y_k, z_k) , $k = 1, 2, 3$, is assumed. Note that the node to be located can be a radio source associated to a sensor with initial unknown location; this is a feasible scenario, as, in some cases, sensors may be deployed without precise location information. Anchor nodes are notified to perform distance estimations to the node of interest. This information is sent to the node so that it can estimate its own location. Any of three ANs (AN_k , AN_l , or AN_n) with original coordinates (x_k^o, y_k^o, z_k^o) , (x_l^o, y_l^o, z_l^o) and (x_n^o, y_n^o, z_n^o) , respectively, can be selected to define the original reference plane $P^{o_{kln}}$ so that coordinates (x_j^o, y_j^o, z_j^o) of the node to be located n_j can be expressed with respect to the reference plane. For instance, see the plane $P^{o_{123}}$ defined by AN_1 , AN_2 and AN_3 in Figure 1a forming a reference for the location of node n_j .

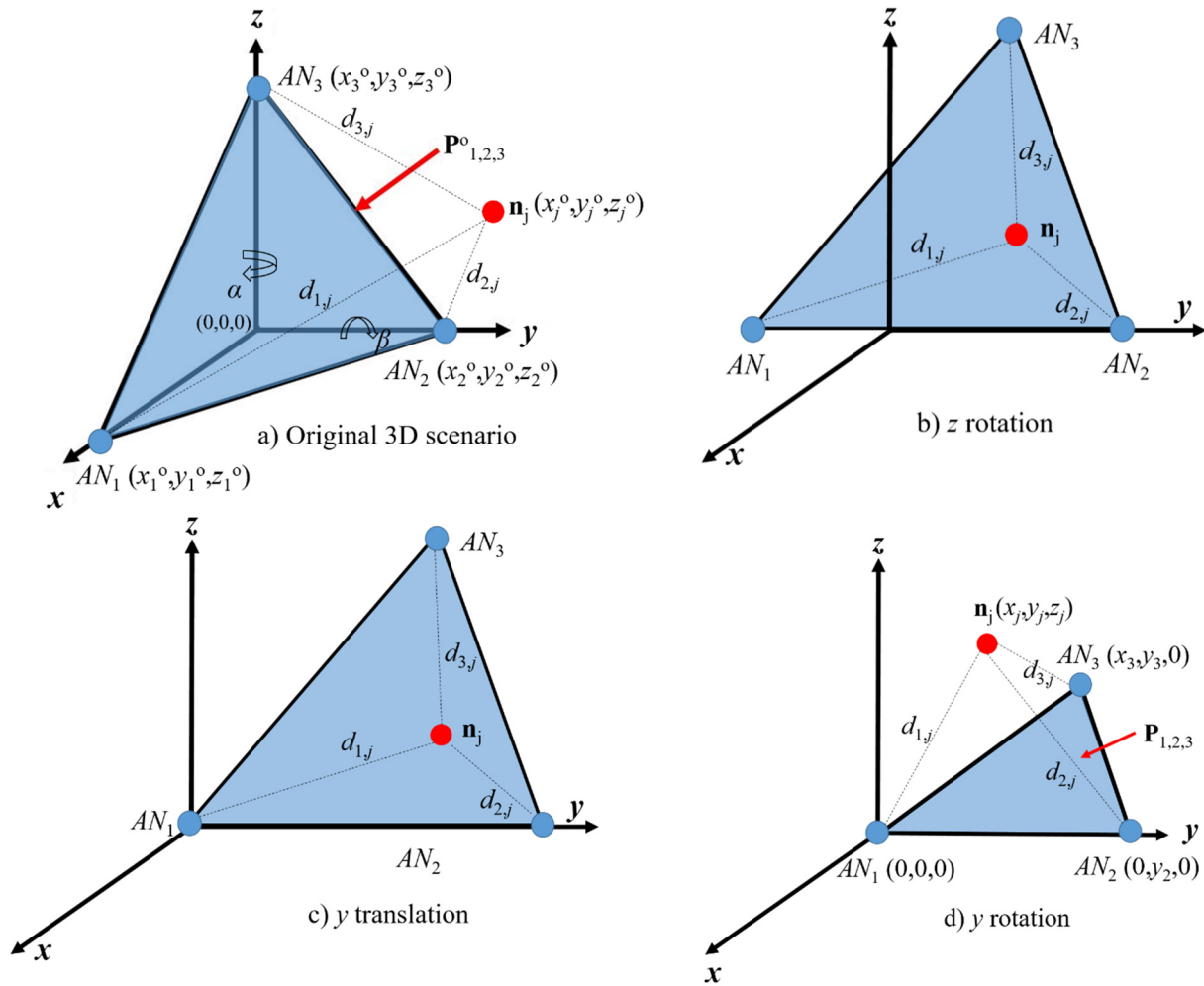


Figure 1. (a) Original 3D network scenario with three ANs. (b) Rotated reference plane with respect to the z -axis by angle α . (c) Translated reference plane. (d) Rotated reference plane with respect to the y -axis by angle β .

As illustrated in Figure 1, the location estimation of the node of interest, calculated by anchor nodes that define the reference plane (Figure 1a), is relative to the rotated/translated anchor's coordinates (Figure 1d). Note that, after rotation/translation, the coordinates of node AN_1 are $(0, 0, 0)$; therefore, in order to obtain the scenario illustrated in Figure 1b,c, a rotation by angles α and β is performed in the z - and y -axis, respectively. Therefore, the coordinates (x_j, y_j, z_j) of node n_j with respect to plane P_{123} can be obtained from the following distance relationships:

$$d_{1,j}^2 = x_j^2 + y_j^2 + z_j^2, \quad (1a)$$

$$d_{2,j}^2 = (x_2 - x_j)^2 + y_j^2 + z_j^2, \quad (1b)$$

$$d_{3,j}^2 = (x_3 - x_j)^2 + (y_3 - y_j)^2 + z_j^2, \quad (1c)$$

where we denote the true separation distance between node n_j and AN_k as $d_{k,j}$ and this is unaltered after plane rotation/translation. Thus, after the algebraic handling and in a noise free scenario, the coordinates (x_j, y_j, z_j) can be expressed as

$$x_j = \frac{d_{1,j}^2 - d_{2,j}^2 + x_2^2}{2x_2}, \quad (2a)$$

$$y_j = \frac{d_{1,j}^2 - d_{3,j}^2 + x_3^2 + y_3^2 - 2x_3x_j}{2y_3}, \quad (2b)$$

$$z_j = \sqrt{d_{1,j}^2 - (x_j)^2 - (y_j)^2}. \quad (2c)$$

For the scenario illustrated in Figure 1, the coordinates (x_j, y_j, z_j) can be translated back to the original coordinate system (x_j^0, y_j^0, z_j^0) , so that node n_j is meant (after the rotation in y - and z -axis) to be located at

$$\begin{aligned} x_j^0 &= (x_j \cos \beta + z_j \sin \beta) \cos \alpha - y_j \sin \alpha, \\ x_j^0 &= (x_j \cos \beta + z_j \sin \beta) \cos \alpha - y_j \sin \alpha, \\ z_j^0 &= -x_j \sin \beta + z_j \cos \beta, \end{aligned} \quad (3)$$

where α and β are the rotation angles illustrated in Figure 1a. This process can be conducted for any other combination of ANs. Note that, in real environments, physical as well as propagation impairments render in noisy distance estimations from ANs to the node of interest. These impairments are quantified by using an additive noise model for estimation distances. Thus, let us define the estimated distance between AN_k and the node of interest n_j as $d'_{k,j} = d_{k,j} + E_{k,j}$, where we assume that $E_{k,j}$ is exponentially distributed with parameter λ ($E_{k,j} \sim \exp(\lambda)$), that is proportional to the true distance $d_{k,j}$ from the AN_k to node n_j . With these distances estimated, the coordinates of the node of interest are not those obtained in a noise-free scenario and they are noisy coordinates (x'_j, y'_j, z'_j) obtained by using the same equation set (2), but with distances given by $d'_{k,j}$, $k = 1, 2, 3$. In the following section, we introduce the RL algorithm, where the coordinates of the node of interest are recursively recalculated and where we express (x'^n, y'^n, z'^n) as the n -th calculation of the node of interest's location coordinates. For the RL algorithm, the coordinates (x'_j, y'_j, z'_j) obtained using Equation set (2) with $d'_{k,j}$ are defined as (x'^n, y'^n, z'^n) for $n = 1$.

2.1. Recursive Localization (RL) Data Fusion Algorithm

As noted before, the estimated distances $d'_{k,j}$ may be corrupted, resulting in impaired coordinate calculations. In this section, we explain the data fusion-based Recursive Localization (RL) algorithm. This algorithm is based on the ANs' known locations and on the first location estimation (x'^1, y'^1, z'^1) of the node of interest n_j , as follows. Let us consider a scenario with three available ANs, AN_1 , AN_2 and AN_3 , where we can estimate the location of a node of interest n_j , (x'^n, y'^n, z'^n) , relative to plane P_{klm} (e.g., P_{123}), based on a set of estimated distances $d'_{k,j}$ (e.g., $k = 1, 2, 3$) as those in Equation set (2). Note that the super index of the coordinates (x'^n, y'^n, z'^n) starts with $n = 2$ for the first time that the recursive algorithm recalculates the location of n_j and it continues for values of $n > 2$ for the subsequent calculations. Once the coordinates (x'^n, y'^n, z'^n) are calculated, node n_j becomes an anchor node AN_j and three new different planes of reference, P_{12j} , P_{13j} and P_{23j} , can be generated by rotation, as shown in Figure 2. Note that, in each of these new planes, one of the original anchor nodes is disregarded, i.e., AN_3 is not part of plane P_{12j} ; we call this the extra anchor node. Then, the RL data fusion algorithm is as follows:

1. The extra anchor node AN_k (e.g., AN_3 for reference plane P_{12j} , see Figure 2a) performs self-localization (i.e., calculates the coordinates of AN_k^n , namely, x'_k^n, y'_k^n, z'_k^n) based on the two noise-free distances with the other two original ANs (e.g., d_{13} and d_{23} , if AN_3 is self-locating) and one noisy distance (e.g., $d'_{3,j}$, if AN_3 is self-locating) with the estimated location of the new anchor node n_j . For instance, if AN_3 is self-locating, $x_3'^n, y_3'^n$ and $z_3'^n$ are calculated based on the distances $d_{1,3}$, $d_{2,3}$ and $d'_{3,j}$ and on the previous found coordinates $(x'^{n-1}, y'^{n-1}, z'^{n-1})$, using Equation set (2). Let us recall that $n - 1 = 1$ for the first recursion.

- The location error is defined as the Euclidean distance between the true position of $AN_k(x_k, y_k, z_k)$ and the current n -th estimate of $AN'_k(x'_k{}^n, y'_k{}^n, z'_k{}^n)$, i.e.,

$$e_k{}^n = \sqrt{(x_k - x'_k{}^n)^2 + (y_k - y'_k{}^n)^2 + (z_k - z'_k{}^n)^2}.$$

- Then, we formulate the recursive correcting vector $\mathbf{CV}_k$, which is used to re-calculate the node of interest coordinates, based on the difference between the estimated and the true position of AN_k . This is calculated as $\mathbf{CV}_k{}^n = [cx_k{}^n = |x'_k{}^n - x_k|, cy_k{}^n = |y'_k{}^n - y_k|, cz_k{}^n = |z'_k{}^n - z_k|]$ (e.g., $\mathbf{CV}_3{}^n = [cx_3{}^n = |x'_3{}^n - x_3|, cy_3{}^n = |y'_3{}^n - y_3|, cz_3{}^n = |z'_3{}^n - z_3|]$).
- Then, to find a local minimum of n_j coordinates, we calculate a new estimate of the coordinates $(x'_j{}^n, y'_j{}^n, z'_j{}^n)$ of node n_j by applying $\mathbf{CV}_k$ as follows:

$$\begin{aligned} x'_j{}^n &= x'_j{}^{n-1} - cx_k{}^n, \\ y'_j{}^n &= y'_j{}^{n-1} - cy_k{}^n, \\ z'_j{}^n &= z'_j{}^{n-1} - cz_k{}^n, \end{aligned} \quad (4)$$

where n is the number of the current coordinate calculation in the RL algorithm.

- Distances involved in equation set (2) are used again to re-calculate the location $(x'_k{}^n, y'_k{}^n, z'_k{}^n)$ of anchor $AN'_k{}^n$; this process (steps 1–4) is repeated until a local minimum is found, that is, when the current error calculation $e_k{}^n$ is larger than the previous error $e_k{}^{n-1}$. If this criterion is not met, the recursive algorithm continues until $|e_k{}^n - e_k{}^{n-1}| < \xi$, or when an unfeasible location is found, in order to proceed with the next step. Note that we are assuming that the node of interest is located at the first octant, as illustrated in Figure 1d.
- Finally, the coordinates $(x'_j{}^n, y'_j{}^n, z'_j{}^n)$ calculated in the last iteration are considered as the final location estimation of n_j .

Please note that the convergence condition of the RL algorithm is defined in step 5. Simulations over different ξ in the range from $0.10e_k{}^{n-1}$ to $0.30e_k{}^{n-1}$ were performed and results showed a similar behavior.

In addition, note that the RL algorithm does not rely on the knowledge of the true position of n_j . Let us recall that the local minimum as well as the threshold error ξ (step 5) are formulated based on successive calculations of the estimated locations of AN_k . Note that the correcting factor is used to correct the position estimation of n_j (step 4) and this new position is used to repeat steps 1–4 until the local minimum is found or the threshold error is reached (step 5).

In order to provide a better understanding of the complete localization process, we provide the following points of the execution of the algorithm:

- Calculate the initial location estimation $(x'_j{}^1, y'_j{}^1, z'_j{}^1)$ of n_j using Equation set (2).
- Then, node n_j now becomes an anchor node.
- Next, obtain the reference planes $\mathbf{P}_{12j}$, $\mathbf{P}_{13j}$ and $\mathbf{P}_{23j}$ using the new anchor node n_j .
- Then, follow steps 1–6 of the RL algorithm.

This algorithm can be implemented using a single extra anchor node AN or using the three ANs as extra anchor nodes and averaging the final n_j location, which we call mean RL. We anticipate that using all the ANs involved might improve the n_j location estimation.

Let us recall that the reference planes illustrated in Figure 2 are obtained by rotation and translation of the original plane by applying Equation set (3). Finally, when the location estimation of anchor $AN'_k{}^n(x'_k{}^n, y'_k{}^n, z'_k{}^n)$ is obtained, it can be transformed by translation/rotation to the original reference plane in order to calculate the corresponding recursive CV.

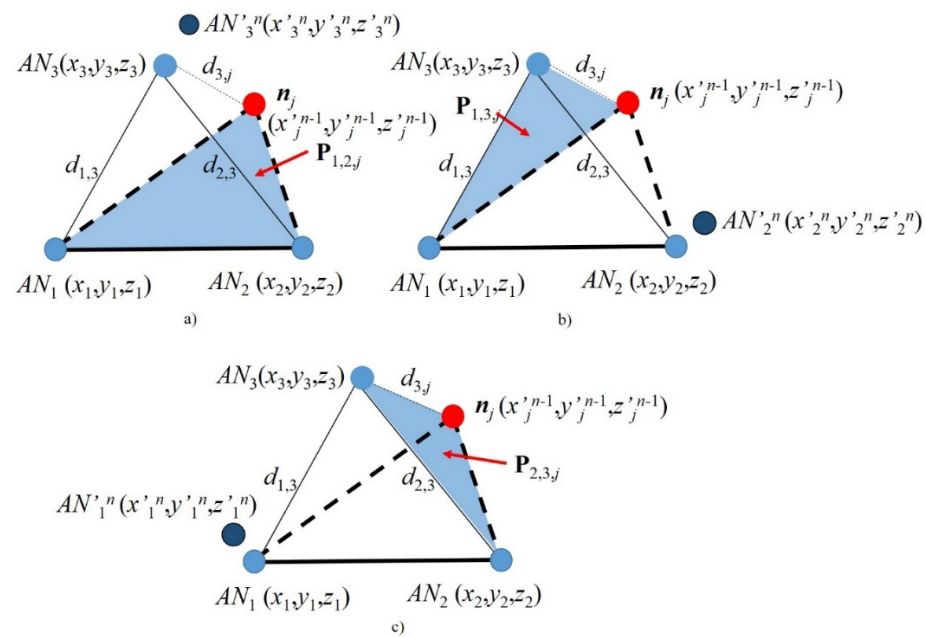


Figure 2. Recursive localization illustration. (a) AN_3 self-localizing, (b) AN_2 self-localizing and (c) AN_1 self-localizing.

2.2. Reference Plane Redundancy

Note that, if more ANs are available, then we can obtain alternative planes that can be used for the location estimation of n_j and this may improve the first n_j location estimation (x_j^1, y_j^1, z_j^1) used in the first step of the RL algorithm. The number of different planes that we can select to be used in the algorithm is given by the number of combinations ${}_NC_3$, where N is the number of available ANs. Thus, for $N = 4$, four different sets of three ANs are obtained; therefore, four different calculations of coordinates (x_j, y_j, z_j) are possible (see Figure 3). In a noise-free environment, these four sets of equations lead to the same location estimation result. However, in practice, the estimated distances $d'_{k,j}$ are obtained from field strength or delay measurements and are impaired by different propagation phenomena or routing strategies. All these impairments render in different coordinate calculations that can be averaged to smooth out the difference. In [15], the authors proposed a PL method for a different 2D scenario based on a set of equations similar to those in (1); however, they did not take advantage of anchor point redundancy as we do in this paper. They related the equations in (1) by subtracting them and arranging terms in a matrix form, solving for a 2D scenario.

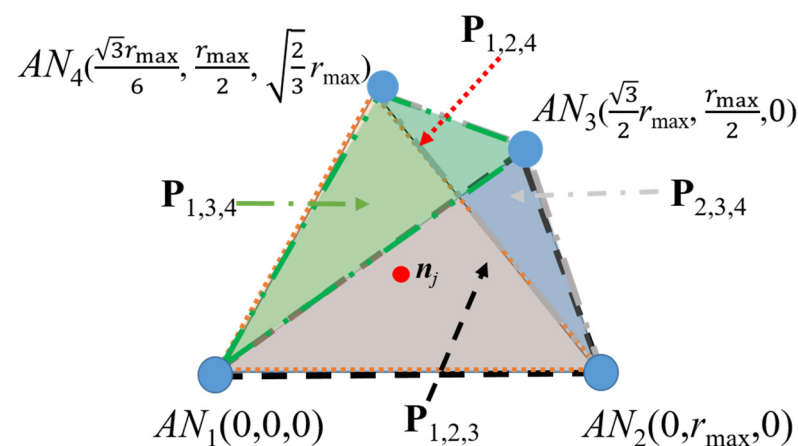


Figure 3. Different reference planes for several position location estimations of n_j in a four-AN scenario.

3. Simulations and Results

Note that, for scenarios as those illustrated in Figure 1, the ANs' locations can be arbitrarily set. In order to assess the performance of the proposed 3D PL technique and without loss of generality, we define the working operational space where three ANs are located at the corners of an equilateral triangle $AN_1(0,0,0)$, $AN_2(x_2 = r_{\max}, 0, 0)$ and $AN_3(x_3 = r_{\max}/2, 0)$, where $r_{\max}$ is the maximal transmission range. For cases when four ANs are available, let us define a scenario such as that illustrated in Figure 3, where the ANs' locations are placed at the corners of a regular tetrahedron. Therefore, for simulation purposes and without loss of generality, the ANs' locations are set as $AN_1(0,0,0)$, $AN_2(0, r_{\max}, 0)$, $AN_3(\sqrt{3}r_{\max}/2, r_{\max}/2, 0)$ and $AN_4(\sqrt{3}r_{\max}/6, r_{\max}/2, \sqrt{2}r_{\max}/3)$.

Let us recall that we expect to have a distance estimation noise derived from signal propagation impairments. In the following, we present the results of the mean squared Euclidean distance error ε between the estimated and true locations, normalized to $r_{\max}$, i.e., $\varepsilon/r_{\max}$. The normalization allows us to observe the error as a proportion of the reachability radius and to see if, for a larger value of $r_{\max}$, this proportion does not change. This normalized location error is shown for an additive observation error $E_{k,j}$ of the true separation between AN_k and the node to be located, n_j . Let us recall that $E_{k,j}$ is exponentially distributed with parameter λ and proportional to the true distance $d_{k,j}$ from AN_k to node n_j , $E_{k,j} \sim \exp(\lambda) * d_{k,j}$. This is the observed distance $d'_{k,j}$ from AN_k to n_j and can be expressed as $d'_{k,j} = d_{k,j} + E_{k,j}$. The simulations were conducted for a random location of node n_j and the maximum coverage radius was initially set to $r_{\max} = 10$ m.

For the results shown in Figure 4, a line of sight (LOS) from the ANs to node n_j is considered, to observe how having noisy distance estimations may affect the performance of the PL 3D technique. In this figure, the results of the normalized Euclidean error $\varepsilon/r_{\max}$ are presented as a function of the parameter λ of the random variable—let us recall that $E_{k,j} \sim \exp(\lambda) * d_{k,j}$. Please note that both the mean and the variance of the exponential random variable of the error are related to λ (i.e., $E[E_{k,j}] = 1/\lambda$, $\text{Var}[E_{k,j}] = 1/\lambda^2$). Please note that, as λ increases, so does the error dispersion. In the context of distance estimation, this means that we evaluate distance errors with high variability. We select λ from 0 to 0.1, to provide a high error dispersion. This simulation presents the results when using a single plane (three ANs available) and when single and mean RL are introduced on the node's coordinates calculation. As noted in Figure 4, the RL algorithm achieves significant improvements, of about 30% and 40%, when using one AN and three ANs, respectively.

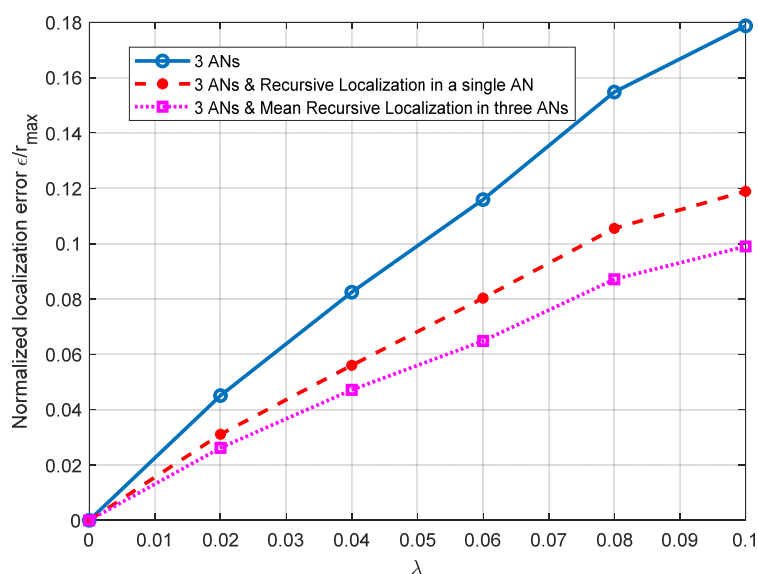


Figure 4. Normalized mean squared error $\varepsilon/r_{\max}$ vs. exponentially distributed with parameter λ , considering three ANs, single and mean RL for three ANs and $r_{\max} = 10$ m.

In Figure 5, we present a histogram of the percentage of improvement obtained with the RL algorithm over the location estimation using a single plane for $\lambda = 0.1$ and for 1000 analyzed cases. The improvement percentage ρ is defined as $\rho = \frac{\varepsilon - \varepsilon_{RL}}{\varepsilon} * 100\%$, where ε is the Euclidean distance error using a single plane and ε_{RL} is the Euclidean distance error using the RL algorithm, between the estimated and the true node location. In Figure 5, we observe that the RL algorithm provides a significant improvement in all the analyzed cases, because all the values of ρ in Figure 5 are non-negative. The histograms show that the probability density function changes as more ANs are considered on the RL algorithm, concentrating improved values of ρ around the mean (see Figure 5b). Note that the improvement percentage ρ pdf for the mean RL (three involved ANs) is different from that of the ρ pdf using one AN. For the mean RL, ρ seems to have a normal pdf with a mean of around $\rho = 50\%$ and approximately 94.7% of the analyzed cases presented an improvement from $\rho = 20\%$ to $\rho = 80\%$. When we take the average (mean RL) of the three ANs' correction vectors, it is possible that the correction vector associated with a single AN is compensated by the other two estimations, providing a better behavior and avoiding any possible bias.

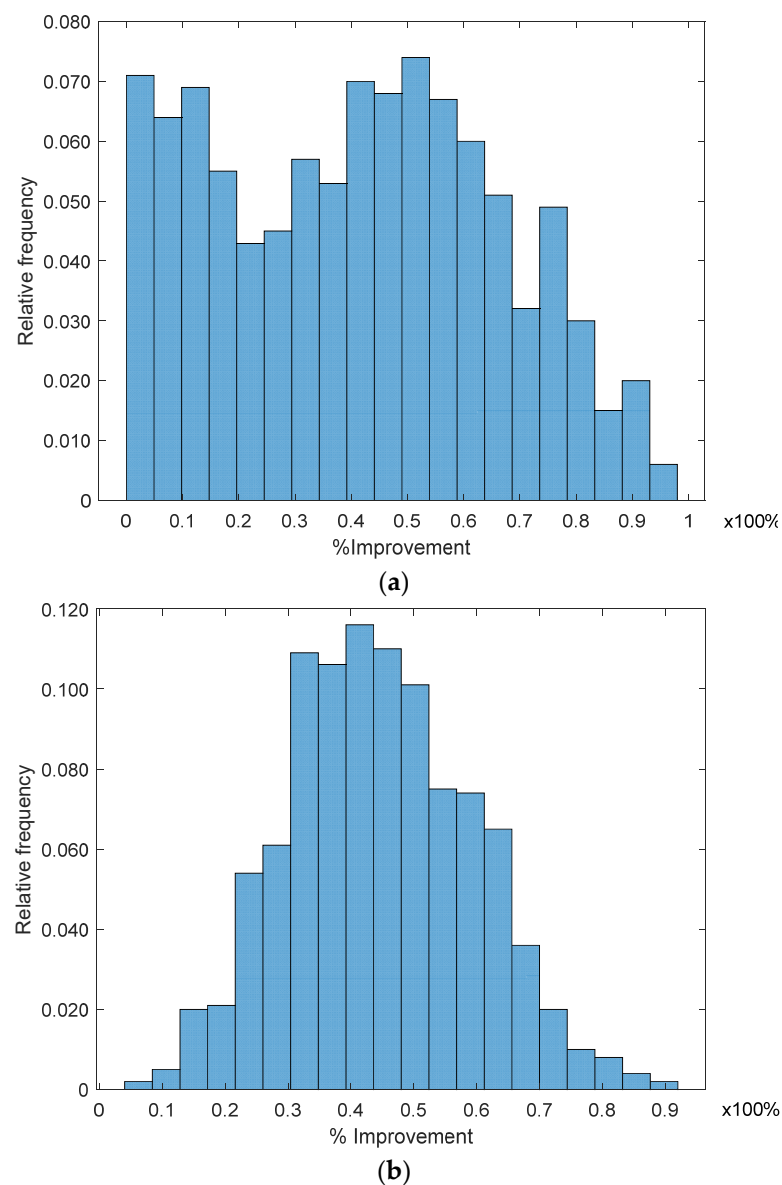


Figure 5. Improvement histogram achieved using RL: (a) single AN and (b) mean 3 ANs.

In Figure 6, additional planes are considered (four ANs) for (x'_j, y'_j, z'_j) calculation and it can be observed that, by using additional planes, we achieve better results than those shown in Figure 4, where only a single plane (three ANs) is used. Note that we can perform recursive localization in a single AN or in the three ANs on the working plane and take the average of the three estimations (mean RL). This can be performed on a single plane or on several planes.

Moreover, it can be seen how the use of the RL algorithm in a single plane provides better results than using multiple reference planes alone. As it can be observed in Figure 4, the proposed RL method provides significant improvements in both scenarios. These improvements can also be increased when the mean RL is used in the four ANs involved—an improvement of about 26% over the results obtained using the RL algorithm on a single plane. As in the scenario analyzed in Figure 4, in this analysis, we can observe how the RL algorithm achieves significant improvements (single and mean) for all the cases analyzed σ).

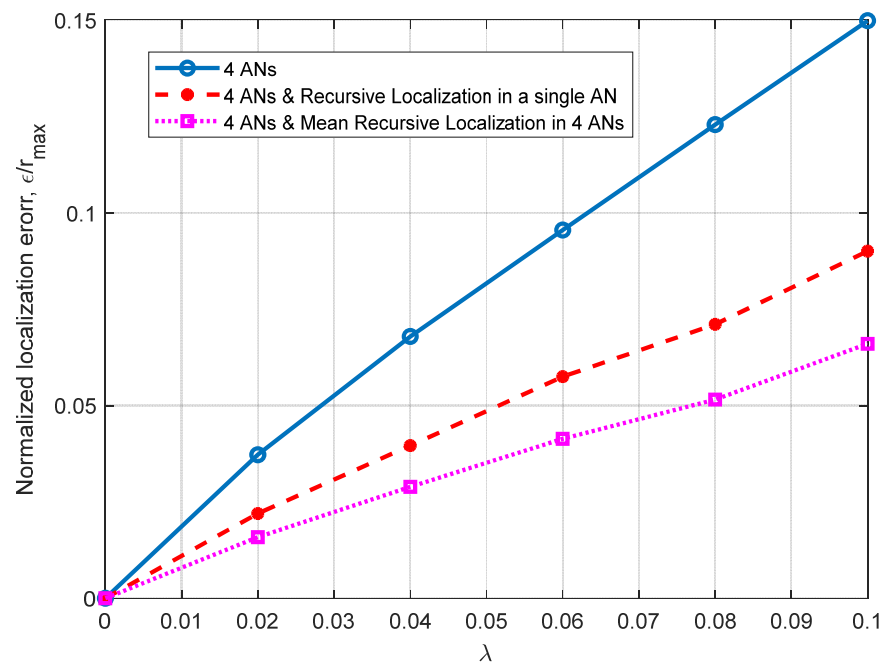


Figure 6. Normalized mean squared error $\epsilon/r_{\max}$ vs. exponentially distributed distance error with parameter λ , considering four ANs, single and mean recursive localization and $r_{\max} = 10$ m.

In Figure 7, we present the histogram of the improvement percentage ρ , obtained using the RL algorithm in the four ANs involved and averaging recursive vectors in order to obtain the final location estimation for n_j . In this figure, we can observe that, as more ANs are considered in the RL algorithm, the pdf of the improvement ρ is more concentrated around the mean. For this evaluation, ρ seems to have a normal pdf with a mean of around $\rho = 50\%$ and approximately 99% of the analyzed cases presented an improvement from $\rho = 20\%$ to $\rho = 80\%$.

In Figure 8, we analyze the RL performance based on the improvement percentage ρ using the mean RL in three ANs, for the scenario illustrated in Figure 1b, where three ANs are located at the corners of an equilateral triangle, $AN_1(0,0,0)$, $AN_2(x_2 = r_{\max}, 0, 0)$ and $AN_3(x_3 = r_{\max}/2, r_{\max}/2, 0)$, and $r_{\max}$ is increased gradually. In this figure, we can observe a very desirable behavior, as the RL performance is not degraded as the operational space is increased.

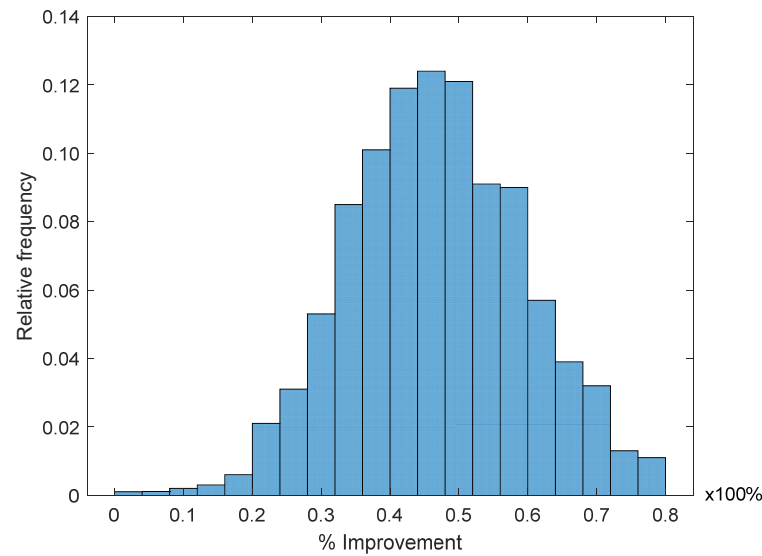


Figure 7. Improvement histogram for recursive localization using mean 4 ANs.

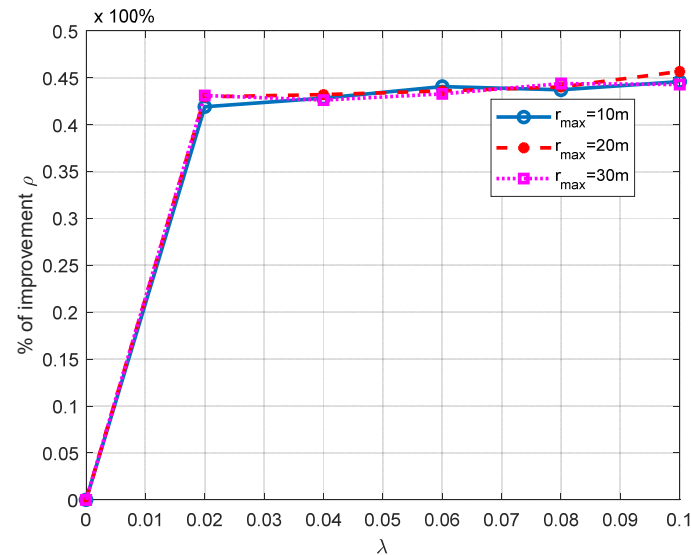


Figure 8. Performance improvement analysis of the RL algorithm for different $r_{\max}$.

Now, we evaluate the improvement ρ given by the RL algorithm when three ANs are available and for different locations of AN_2 and AN_3 . Note that, by changing the position of these ANs defining the reference plane as that in Figure 9, we obtain all the possible shapes for the triangle on the reference plane. The evaluation network scenario is generalized by modifying the coordinates of AN_2 and AN_3 , as illustrated in Figure 9.

The deployment of AN_2 is modified by adding to x_2 and y_2 random variables $\pm\zeta_{2x}$ and $\pm\zeta_{2y}$ respectively, while z_2 is kept without alteration. The same modification is conducted for the coordinates of AN_3 . Let us recall that plane normalization by translation/rotation was carried out as explained before; therefore, we can consider that all the ANs are in $z = 0$. So, let us define the distances ζ_{ix} and ζ_{iy} ($i = 1, 2, 3$) as two uniformly distributed variables, defined in the ranges $[0 \ \zeta_x]$ and $[0 \ \zeta_y]$, respectively. (See axis- x in Figure 10.) This is performed in order to generalize the results given by the RL algorithm. In Figure 10, we can observe that, for increments below the 30% of $r_{\max}$ ($\zeta_x = 1$, $\zeta_y = 1$), the accuracy improvement ρ is not severely degraded, as ρ only decreases by approximately 11%.

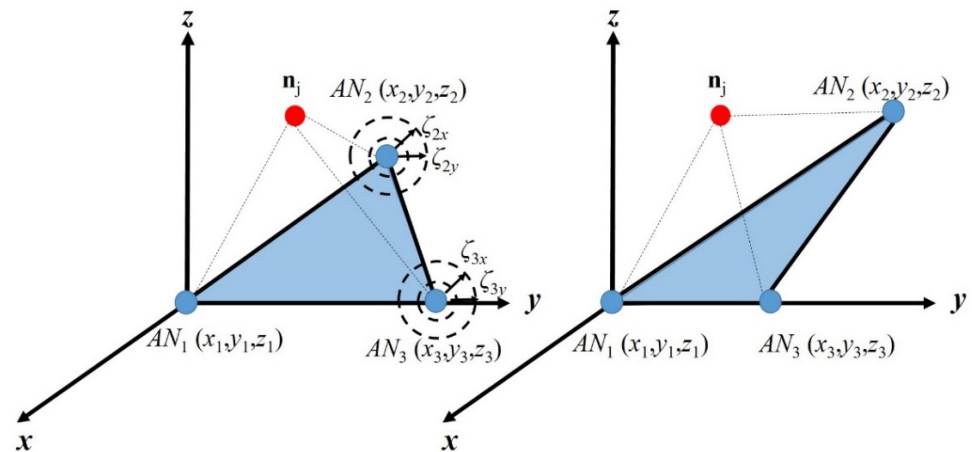


Figure 9. Generalization of the network scenario for evaluation.

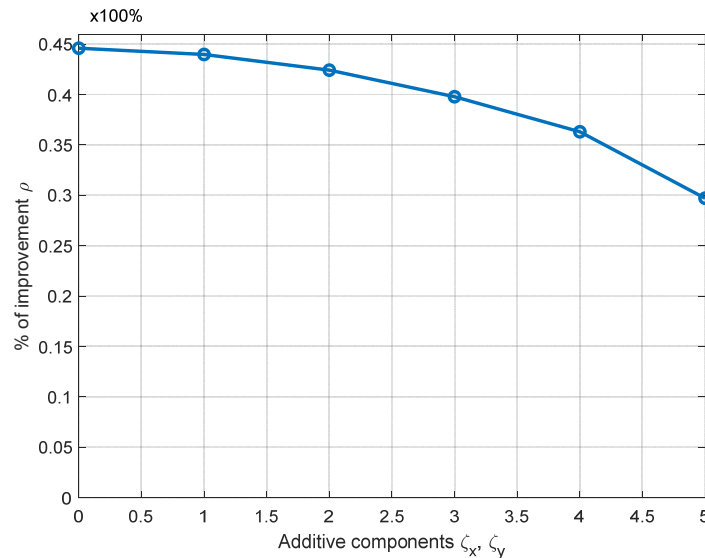


Figure 10. Improvement (ρ) evaluation of the RL algorithm for different locations of AN_2 and AN_3 .

For multihop scenarios, a direct link from the ANs to the node to locate may not be available. In Figure 11, we analyze the performance of the localization technique using three ANs as in Figure 1, over multihop scenarios in a defined cube with side $l = r_{\max}$. The shortest route from AN_i to a node n_j , is referred to as R_{ij} and is obtained by Dijkstra's algorithm, considering a homogenous reachability radius $r_{\max} = 3$ m for all nodes in the network. For this analysis, the distance d_{ij} from AN_i to node n_j , is approximated by the hop count in the route linking the ANs to n_j . In Figure 11, we analyze the RL algorithm performance over a multihop route scenario for a different number of nodes η in the operational space. In this analysis, we can see that, as the number of nodes in the network grows, the distance error given by the single plane (three ANs) equations using the mean RL in three ANs tends to decrease. This can be explained by the fact that, when the number of nodes increases, the network connectivity improves and better routes may be obtained. Let us recall that, in this evaluation, distance estimation is based on the hop count method. Note that the improvement achieved by the RL algorithm is always above approximately 30%.

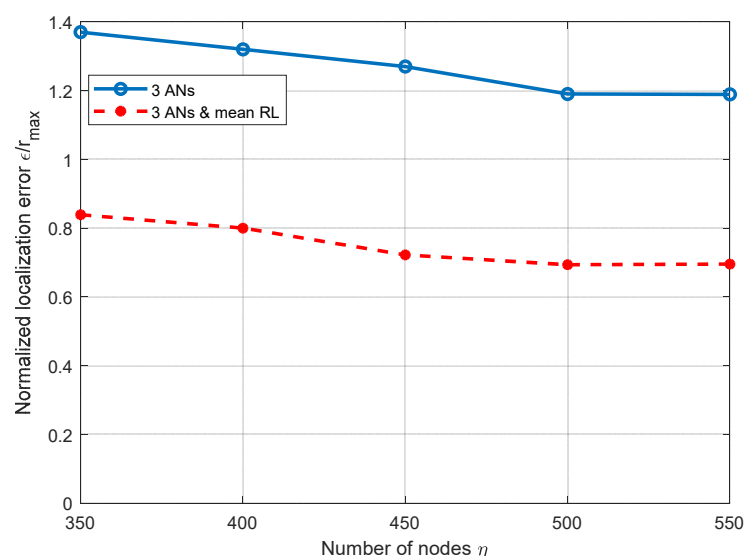


Figure 11. Normalized mean squared error $\epsilon/r_{\max}$ vs. number of nodes η for multihop scenarios, $r_{\max} = 3$ m.

4. Conclusions

A recursive data fusion-based localization method suitable for 3D sensor networks is here proposed and its performance assessed both in line-of-sight scenarios (single hop) and in the presence of propagation obstructions (multihop environments) with three and four ANs. The 3D PL method is based on the space formulated by arbitrarily located ANs near the network operational scenario. The RL algorithm was formulated and it was observed that it achieves significant improvements under different scenarios. It was shown through simulations that different locations of ANs do not impair the performance of the RL algorithm. For multihop environments, it was also observed that, as the number of nodes increases, the performance of the PL 3D technique improves, as the estimated distances are improved by means of better (straight) routes.

Author Contributions: Conceptualization, R.V.-H.; formal analysis, R.V.-H.; methodology, R.V.-H., C.V.-R. and D.M.-R.; project administration, C.V.-R.; resources, C.V.-R.; supervision, C.V.-R. and D.M.-R.; validation, R.V.-H. and D.M.-R.; visualization, R.V.-H.; writing—original draft, R.V.-H.; writing—review and editing, C.V.-R. and D.M.-R. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the SEP-CONACyT Research Project 256237.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors want to thank the support of SEP-CONACyT, the School of Engineering and Sciences and the Telecommunications and Networks Focus Group at Tecnológico de Monterrey.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Pahlavan, K.; Krishnamurthy, P. *Principles of Wireless Networks: A Unified Approach*, 1st ed.; Prentice Hall of India PTR: New Delhi, India, 2002.
2. Kansal, A.; Zhao, F. Data Fusion in Sensor Networks. In *Encyclopedia of Database Systems*; Springer: New York, NY, USA, 2018.
3. Abu-Mahfouz, A.M.; Gerhard, H. Localised Information Fusion Techniques for Location Discovery in Wireless Sensor Networks. *Int. J. Sens. Netw.* **2018**, *26*, 12–25. [[CrossRef](#)]

4. Jin, M.H.; Wu, H.K.; Hormg, J.T. An Intelligent Handoff Scheme Based On Location Prediction Technologies. In Proceedings of the 14th International Conference DEXA, Prague, Czech Republic, 1–5 September 2002; pp. 551–557.
5. Maung, N.; Kawai, M. Experimental evaluations of RSS threshold-based optimized DV-HOP localization for wireless adhoc networks. *Electron. Lett.* **2014**, *50*, 1246–1248. [[CrossRef](#)]
6. Zhou, X.; Zhang, L.; Cheng, Q. Landscape-3D a robust localization scheme for sensor networks over complex 3D terrains. In Proceedings of the 31st Annual IEEE Conference on Local Computer Networks (LCN'06), Tampa, FL, USA, 14–16 November 2006; pp. 239–246.
7. Jiand, X.; Zha, H. Sensor Positioning in Wireless Ad-hoc Sensor Networks Using Multidimensional Scaling. In Proceedings of the 23rd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'04), Hong Kong, China, 7–11 March 2004; pp. 2652–2661.
8. Lee, S.; Cho, B.; Koo, B.; Ryu, S.; Choi, J.; Kim, S. Kalman Filter Based Indoor Position Tracking with Self-Calibration for RSS Variation Mitigation. *Int. J. Distrib. Sens. Netw.* **2015**, *11*, 674635. [[CrossRef](#)]
9. Zhao, T.; Li, X.S.; Zhang, L. Weighted centroid localization algorithm based on node cooperation in coal mine underground. *Ind. Mine Autom.* **2018**, *44*, 32–38.
10. Stojkoska, B.R. Nodes Localization in 3D Wireless Sensor Networks Based on Multidimensional Scaling Algorithm. In Proceedings of the 10th Conference for Informatics and Information Technology (CIIT'13), Bitola, Macedonia, 18–21 April 2013; pp. 301–306.
11. Yang, X.; Yan, F.; Liu, J. 3D Localization Algorithm Based on Voronoi Diagram and Rank Sequence in Wireless Sensor Network. *Sci. Program. J.* **2017**, *2017*, 4769710. [[CrossRef](#)]
12. Villalpando-Hernandez, R.; Munoz-Rodriguez, D.; Vargas-Rosales, C.; Rizo-Dominguez, L. 3-D Position Location in Ad Hoc Networks: A Manhattanized Space. *IEEE Commun. Lett.* **2017**, *21*, 124–127. [[CrossRef](#)]
13. Gudmundsson, J.; Levcopoulos, C.; Narasimhan, G. Approximation Minimum Manhattan Networks. *Nord. J. Comput.* **2001**, *8*, 219–232.
14. Saad, E.; Elhoseini, M.; Haikal, A.Y. Recent achievements in sensor localization algorithms. *Alex. Eng. J.* **2018**, *57*, 4219–4228. [[CrossRef](#)]
15. Sayed, A.H.; Tarighat, A.; Khajehnouri, N. Network-based wireless location: Challenges faced in developing techniques for accurate wireless location information. *IEEE Signal Process. Mag.* **2005**, *22*, 24–40. [[CrossRef](#)]