Deep Learning to Robustify a Geometric Interpretation of Trilateration for 3D RSS-based Localization

Hoang. M. Le[1,2], Dirk Slock[1], Jean-Pierre Rossi[2]

[1]Communication Systems Departement, EURECOM - Sophia Antipolis, France
[2]Orange Labs - Sophia Antipolis, France
Email: leho@eurecom.fr, hoang.m.le.1994@gmail.com

***Abstract*—Received Signal Strength (RSS) is ubiquitous in wireless communications. Despite the low accuracy, it is still attractive because of the simplicity and the ready availability in nearly every wireless system without any additional hardware or software required. This paper develops a geometric interpretation of trilateration in RSS-based 3D localization, which is presented in a previous paper but in 2D scenarios. In addition, to correct the final estimates, an iterative Maximum Likelihood (ML) estimator for position estimation is presented. The Artificial Neural Networks (ANNs) are then applied in estimating the related parameters to robustify the performance of the algorithm. When compared to earlier methods, simulation results demonstrate a considerable boost in performance.**

***Index Terms* - trilateration, positioning, RSS, Received Signal Strength, 3D localization, Maximum Likelihood, ANNs.**

I. Introduction

Several main positioning techniques have been used in determining the position of a point: Time of Arrival (ToA), Time Difference of Arrival (TDoA), Received Signal Strength (RSS), and Direction of Arrival (DoA) (in some documents, it is also called Angle of Arrival - AOA) [1]. ToA-based [2-4] and TDoA-based [5-6] positioning require highly accurate clock synchronization among all BSs and mobile device. DoA-based systems do not require such a synchronization [7-10], but it is sensitive to Non Line of Sight (NLoS) and requires antenna arrays. In RSS-based technique, the distance between the mobile device and each base station is firstly estimated by the strength of power received. Afterward, an algorithm of trilateration is then implemented. This technique has its own drawbacks. Shadowing and NLoS circumstances make RSS systems extremely vulnerable, as signal power drops dramatically, resulting in huge estimation mistakes. Local changes in the average received power have a significant impact on these systems (small-scale fading). RSS-based positioning algorithms, on the other hand, continue to play a crucial role in practice. RSS-based systems require the least amount of hardware (only a power detector), which can be found in a variety of applications such as WiFi, Zigbee, and Bluetooth chipsets. There is no need for synchronization in these designs. Since they do not rely on timing information, they perform well in short-range circumstances and are more resistant to multipath. [11].

RSS-based positioning algorithm is a popular topic of many previous works. In the paper [12], we illustrate a geometric interpretation for RSS-based trilateration in 2D. Concerning 3D localization, paper [13] demonstrates their study, but in the scenario where all path loss exponents are already given and considered to be equal, and so are the standard deviations of RSS measurement. In [14-15], positioning algorithms with unknown path loss exponent are studied, but they still consider that standard deviations are known. Recently, deep learning with Artificial Neural Networks is a popular approach to robustify positioning accuracy. The authors in [16] proposes a method using the neural network with RSS samples to estimate directly the coordinates of the unknown point.

In this paper, we continue our work in [12] in 3D scenarios. A geometric interpretation of trilateration in 3D with a Weighted Least Square estimation is presented in section II. In section III, we illustrates RSS-based localization using the proposed 3D trilateration algorithm with estimates of mean and variance of square of the distances, and the an iterative Maximum Likelihood procedure is studied to optimize the positioning algorithm. Section IV presents an utilization of deep learning to enhance the estimations. Instead of directly localize the mobile device by ANNs, we use ANNs to estimate the standard deviation (σ_i) of the each RSS measurement as well as the path loss exponent (α_i) of each transmission link at first, and then estimate the position of mobile device by the α_i and σ_i obtained . In the last two sections, we illustrate the simulations and the related results to prove the superiority of our proposed algorithm.

Notation: $\mathbf{exp}(x)$ denotes an exponentiation with the base e and the power x; $\mathbf{diag}(a_1,\ a_2,\ \ldots,\ a_n)$ is the diagonal matrix whose diagonal elements are $a_1, a_2, \ldots, a_n$ resp.; **[a;b)** is the interval of real numbers from a to b which includes a but excludes b, Γ means **Gamma function**, $\mathbf{log}(\cdot)$ stands for decimal logarithm, $\mathbf{ln}(\cdot)$ stands for natural algorithm, $\mathbf{E}(\cdot)$ stands for mean value, $\mathbf{var}(\cdot)$ stands for variance.

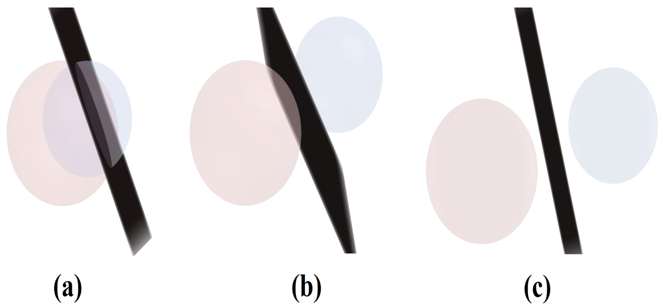

Figure 1: Radical plane of 2 non-concentric spheres when they intersect each other at a circle (a), only one point (b) or zero point (c).

II. Proposed geometric approach

A. Geometric interpretation

Trilateration is a mathematical method which determines the position of an unknown point based on the distances from itself to several base stations around. In 2D scenarios, each base station, with the corresponding distance from itself to the mobile device, forms a circle. In [12], it is discussed that two non-concentric circles have a **radical axis**, which is the locus of a point having equal power with regard to them. The estimated position of the mobile device is supposed to be the common point of these axes. To be continued in 3D scenarios, we consider the term **radical planes** of the two non-concentric spheres (Fig. 1), which is also the locus of a point having the same power with regard to the two spheres concerned. Localization in 3D requires at least 4 base stations; meanwhile determining a point demands at least 3 planes. Therefore, with N base stations, we take $N-1$ planes, where each of them is the radical plane of the first sphere and the i-th sphere ($i \geq 2$). The common point of these planes is considered to be the position of the mobile device. If such a point does not exist, the Weighted Least Square method will be applied to obtain the most appropriate position.

B. Matrix approach

The following equation demonstrates the sphere whose center is the i-th base station and radius is d_i, where d_i is the distance from the i-th base station to the mobile device:

$$(x-x_i)^2+(y-y_i)^2+(z-z_i)^2 = d_i^2 \tag{1}$$

In the equation above, (x,y,z) are the coordinates of the mobile device, (x_i,y_i,z_i) are the coordinates of the i-th base station, i is from 1 to N and N is the number of base stations ($N \geq 4$).

The equation of the radical plane of the spheres corresponding to the first and i-th base station ($i \geq 2$) is

$$\begin{aligned}&(x-x_i)^2+(y-y_i)^2+(z-z_i)^2\\ &-(x-x_1)^2-(y-y_1)^2-(z-z_1)^2 = d_i^2-d_1^2\end{aligned} \tag{2}$$

As a result:

$$\begin{aligned}&(-2x_i+2x_1)x+(-2y_i+2y_1)y+(-2z_i+2z_1)z\\ &= d_i^2-x_i^2-y_i^2-z_i^2-d_1^2+x_1^2+y_1^2+z_1^2\end{aligned} \tag{3}$$

In matrix formulation, we define

$$\boldsymbol{A} = \begin{bmatrix} -2x_2+2x_1 & -2y_2+2y_1 & -2z_2+2z_1 \\ -2x_3+2x_1 & -2y_3+2y_1 & -2z_3+2z_1 \\ \dots & \dots & \dots \\ -2x_N+2x_1 & -2y_N+2y_1 & -2z_N+2z_1 \end{bmatrix}; \ \boldsymbol{x} = \begin{bmatrix} x \\ y \\ z \end{bmatrix};$$

$$\hat{\boldsymbol{b}} = \begin{bmatrix} \widehat{d_2^2}-x_2^2-y_2^2-z_2^2-\widehat{d_1^2}+x_1^2+y_1^2+z_1^2 \\ \widehat{d_3^2}-x_3^2-y_3^2-z_3^2-\widehat{d_1^2}+x_1^2+y_1^2+z_1^2 \\ \dots \\ \widehat{d_N^2}-x_N^2-y_N^2-z_N^2-\widehat{d_1^2}+x_1^2+y_1^2+z_1^2 \end{bmatrix}$$

where $\widehat{d_i^2}$ is the estimate of d_i^2.

Obviously, we have the approximation $\boldsymbol{Ax} \approx \hat{\boldsymbol{b}}$.

A Weighted Least Square estimation [17] is then applied. The estimate of $\boldsymbol{x}$ is compute by:

$$\hat{\boldsymbol{x}} = \left(\boldsymbol{A}^T\boldsymbol{W}^{-1}\boldsymbol{A}\right)^{-1}\boldsymbol{A}^T\boldsymbol{W}^{-1}\hat{\boldsymbol{b}} \tag{4}$$

where $\boldsymbol{W}$ is the covariance matrix of $\hat{\boldsymbol{b}}$:

$$\boldsymbol{W} = \mathrm{var}(d_1^2)\bullet\mathbf{1}\bullet\mathbf{1}^T + \mathrm{diag}\left(\mathrm{var}(d_2^2),\mathrm{var}(d_3^2),\dots,\mathrm{var}(d_N^2)\right) \tag{5}$$

$\mathrm{var}(d_i^2)$ is the variance of d_i^2.
$\mathbf{1} = [1\ \ 1\ \ \dots\ \ 1]^T$ is the all-one vector.

III. RSS-based localization

To estimate the coordinates of mobile device in (4), estimations of d_i^2 and $\mathrm{var}(d_i^2)$ are required. This section analyzes these estimations in RSS-based model and proposes an iterative Maximum Likelihood procedure to optimize the position obtained.

A. RSS model

The average power received via a wireless link is referred to as RSS. The disturbance in RSS caused by shadowing is log-normally distributed, according to field testing. As a result, the log-normal path loss model can be written as follows:

$$P_i = P_0 + 10\alpha_i \log d_0 - 10\alpha_i \log d_i + n_i \tag{6}$$

where P_0 is the power received at a reference point at the distance d_0; d_i is the true distance from the i-th base station to the mobile station; α_i is the path loss exponent of the corresponding transmission link, n_i is the log normal fading. The Path Loss Model (PLM) assumes that this fading is in Gaussian distribution with zero-mean and variance of σ_i^2.

The RSS measurement is simplified to

$$\gamma_{\mathrm{RSS},i} \triangleq P_i - P_0 - 10\alpha \log d_0 \tag{7}$$

As a result, from (6), we have:

$$\gamma_{\mathrm{RSS},i} = -10\alpha_i \log d_i + n_i \tag{8}$$

B. RSS-based Weighted Least Squares estimation

In the Appendices A and B, we prove the following equations to estimate d_i^2 and $\mathrm{var}(d_i^2)$:

$$\widehat{d_i^2} = \exp\left(\frac{-(\ln 10)\gamma_{\mathrm{RSS},i}}{5\alpha_i} - \frac{(\ln 10)^2\sigma_i^2}{50\alpha_i^2}\right) \tag{9}$$

$$\begin{aligned}&\mathrm{var}(d_i^2)\\ &= \exp\left(\frac{-(\ln 10)\gamma_{\mathrm{RSS},i}}{2.5\alpha_i} - \frac{(\ln 10)^2\sigma_i^2}{12.5\alpha_i^2}\right)\left[\exp\left(\frac{(\ln 10)^2}{25\alpha_i^2}\sigma_i^2\right) - 1\right]\end{aligned} \tag{10}$$

The position of the mobile device is then computed by Weighted Least Squares estimation, shown in equation (4).

C. Iterative Maximum Likelihood Procedure

To optimize $\hat{\boldsymbol{x}}$ obtained in (4), an iterative Maximum Likelihood estimator is applied.

In vector form, we denote

$$\boldsymbol{\gamma}_{\mathrm{RSS}} = \begin{bmatrix}\gamma_{\mathrm{RSS},1} & \gamma_{\mathrm{RSS},2} & \dots \gamma_{\mathrm{RSS},N}\end{bmatrix}^T \tag{11}$$

$$\boldsymbol{f}(\boldsymbol{x}) = \begin{bmatrix} -10\alpha_1 \log\sqrt{(x-x_1)^2+(y-y_1)^2+(z-z_1)^2} \\ -10\alpha_2 \log\sqrt{(x-x_2)^2+(y-y_2)^2+(z-z_2)^2} \\ \dots \\ -10\alpha_N \log\sqrt{(x-x_N)^2+(y-y_N)^2+(z-z_N)^2} \end{bmatrix} \tag{12}$$

where $\boldsymbol{x} = \begin{bmatrix}x & y & z\end{bmatrix}^T$.

The vector $\boldsymbol{\gamma}_{\mathrm{RSS}}$ is Gaussian distributed with mean vector of $\boldsymbol{f}$ and covariance matrix $\boldsymbol{C}_{\mathrm{RSS}}$, we have the probability density function (pdf) [19]:

$$\mathrm{p}(\boldsymbol{\gamma}_{\mathrm{RSS}}|\boldsymbol{x}) = \frac{(2\pi)^{-\frac{N}{2}}}{|\boldsymbol{C}_{\mathrm{RSS}}|^{\frac{1}{2}}}\exp\left(\frac{-1}{2}(\boldsymbol{\gamma}_{\mathrm{RSS}}-\boldsymbol{f})^T\boldsymbol{C}_{\mathrm{RSS}}^{-1}(\boldsymbol{\gamma}_{\mathrm{RSS}}-\boldsymbol{f})\right) \tag{13}$$

where

$$\boldsymbol{C}_{\mathrm{RSS}} = \mathrm{diag}(\sigma_1^2, \sigma_2^2, \dots, \sigma_N^2) \tag{14}$$

Maximizing the pdf in (13) is equivalent to finding

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}}(\boldsymbol{\gamma}_{\mathrm{RSS}} - \boldsymbol{f}(\boldsymbol{x}))^T\boldsymbol{C}_{\mathrm{RSS}}^{-1}(\boldsymbol{\gamma}_{\mathrm{RSS}} - \boldsymbol{f}(\boldsymbol{x})) \tag{15}$$

which we shall perform alternatingly. We consider Gauss-Newton procedure [20] for $\hat{\boldsymbol{x}}$. At iteration $(u+1)$:

$$\hat{\boldsymbol{x}}^{(u+1)} = \hat{\boldsymbol{x}}^{(u)} + (\boldsymbol{G}_{\mathrm{RSS}}^T\boldsymbol{C}_{\mathrm{RSS}}^{-1}\boldsymbol{G}_{\mathrm{RSS}})^{-1}\boldsymbol{G}_{\mathrm{RSS}}^T\boldsymbol{C}_{\mathrm{RSS}}^{-1}\left(\boldsymbol{\gamma}_{\mathrm{RSS}} - \boldsymbol{f}(\hat{\boldsymbol{x}}^{(u)}\right) \tag{16}$$

where G_{RSS} is the Jacobian matrix of $\boldsymbol{f}(\boldsymbol{x})$

$$\boldsymbol{G}_{\mathrm{RSS}} = \boldsymbol{G}(\boldsymbol{x}^{(u)}),\ \boldsymbol{G}(\boldsymbol{x}) = \frac{\partial \boldsymbol{f}(\boldsymbol{x})}{\partial \boldsymbol{x}^T}. \tag{17}$$

A procedure is expected to terminate when $\|\boldsymbol{x}^{(u+1)} - \boldsymbol{x}^{(u)}\|_2 < \varepsilon_{\mathrm{RSS}}$, for the stopping criterion $\varepsilon_{\mathrm{RSS}}$ sufficiently small. Then, the final position of the procedure is considered to be the coordinates of the mobile device in the xyz space.

However, iterative procedures do not always converge. In [21], we show that there are **three** possible outcomes for an iterative procedure: **Convergence, Divergence** and **Oscillation**. We take the final position of a converging procedure as the estimated position for the mobile device. As for a diverging procedure or an oscillating procedure, the initial position is selected.

In a nutshell, the Algorithm 1 is proposed for the Gauss-Newton iterative procedure of Maximum Likelihood estimator.

Algorithm 1: Proposed Maximum Likelihood estimtor with Gauss-Newton procedure

```
Given path loss exponents α_i and standard deviations
  σ_i. Take the measured Received Signal Strength P_i.
Compute γ_RSS,i by (8), d_i^2 (hat) by (9), var(d_i^2) by (10).
Assign u = 1 and ε_RSS sufficiently small.
Compute the estimation x̂ by (4) as the first estimated
  coordinates of the mobile device.
repeat
    Compute the following estimated coordinates
      x̂^(u+1) of the mobile device by (16).
    u = u + 1;
until ||x^(u+1) − x^(u)||_2 < ε_RSS or u > 1000 or
  X^(u+1) = ±∞;
if u > 1000 or x̂^(u+1) = ±∞ then
    x̂^(1) is the estimated position of the mobile device;
else
    x̂^(u) is the estimated position of the mobile device;
```

D. Cramer Round Lower Bound (CRLB)

To evaluate the quality of the algorithm based on DDoA, we compare to the Cramer-Rao Lower Bound (CRLB), via the Fisher Information Matrix (FIM):

$$\boldsymbol{I}(\boldsymbol{x}) = \boldsymbol{G}_{\mathrm{RSS}}^T(\boldsymbol{x})\boldsymbol{C}_{\mathrm{RSS}}^{-1}\boldsymbol{G}_{\mathrm{RSS}}(\boldsymbol{x}). \tag{18}$$

The CRLB is the trace of the inverse of FIM:

$$\mathrm{CRLB} = tr(\mathrm{I}^{-1}) \tag{19}$$

IV. An application of Deep Learning in estimating standard deviations of RSS measurements and path loss exponents of environments

A. Formulation of Artificial Neural Network

Artificial neural systems are computing frameworks ambiguously motivated by the neural systems of animal brains. ANNs are distributed parallel systems made up of neurons, which are interconnected processing units that perform mathematical calculations. An ANN contain an input layer with single or multiple inputs, an output layer with singe or multiple outputs, and the hidden layers, each containing the nodes (Fig. 2).

Each connection can transmit a signal to other nodes. The nodes and connections ordinarily have a weight that adjusts as learning proceeds. After traversing the layers numerous times, signals move from the first layer (the input layer) to the last layer (the output layer). Neural networks are trained by working with instances that have known inputs and outputs,

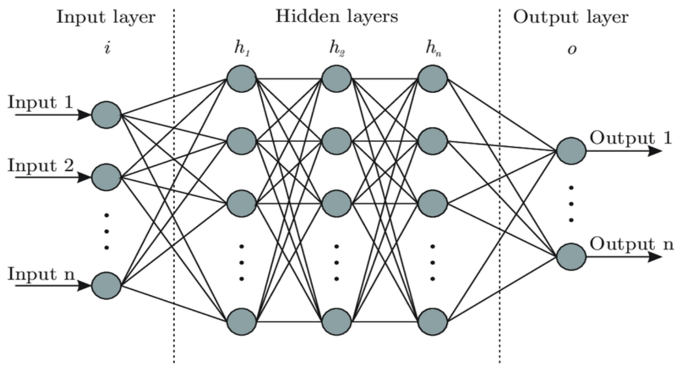

Figure 2: Artificial Neural Network

generating probability-weighted affiliations between them that are saved in the net's structure. The steps to train ANNs are presented in [22].

B. Estimation of path loss exponents and standard deviations

The proposed method is a form of supervised learning which maps the inputs to the outputs. Learning is accomplished using Levenberg-Marquardt algorithm [23].

To improve the performance of RSS-based localization, estimations of path loss exponents and standard deviations using ANNs are implemented. This task has two following phases:

- **Offline phase (training phase)**. 500 reference points are selected. From each reference point (known coordinates), the signals are transmitted from the mobile device to the network of base stations. The experiments are conducted in M = 100 times for each point. Let $P_i^{(m)}$ be the measured RSS at the i-th base station in the m-th experiment. The estimated RSS $\hat{P}_i$ is the mean of the M measured value:

$$\hat{P}_i = \frac{1}{M}\sum_{m=1}^{M} P_i^{(m)} \tag{20}$$

As a results, the estimated path loss exponent of the corresponding transmission link can be expressed by

$$\hat{\alpha}_i = \frac{\hat{P}_i - P_0}{10(\log d_0 - \log d_i)} \tag{21}$$

As the coordinates of the reference point are known, the distance d_i are already defined.
The unbiased estimate of the standard deviation $\hat{\sigma}_i$ of each RSS measurement is estimated by the following equation

$$\hat{\sigma}_i = \frac{1}{c_M}\sqrt{\frac{1}{M-1}\sum_{m=1}^{M}\left(P_i^{(m)} - \hat{P}_i\right)^2} \tag{22}$$

where c_M is a correction factor in terms of **Gamma function** [26].

$$c_M = \sqrt{\frac{2}{M-1}}\frac{\Gamma\left(\frac{M}{2}\right)}{\Gamma\left(\frac{M-1}{2}\right)} \tag{23}$$

Two ANNs are trained. As for the first ANN, the inputs are N estimated RSS measured received at N base stations, and the outputs are N path loss exponents. Similarly, the second ANN has the same N inputs but its outputs are N corresponding standard deviations. Accordingly, the size of training data set for each ANN is the number of reference points (500).

- **Online phase (standard deviation estimating phase)** When the mobile device at unknown position transmits signals, the N base stations receive them and compute the estimated RSS. By the ANNs trained, each standard deviation for each corresponding RSS measurement and each path loss exponent for each transmission link are then estimated.

V. Simulation Results

A. Simulation Setup

To compare the quality among of algorithms and CRLB, we use Root Mean Square Position Error (RMSE) which is defined by

$$\text{RMSE} = \sqrt{\text{E}(\|\hat{\boldsymbol{x}} - \bar{\boldsymbol{x}}\|^2)} \tag{24}$$

where $\bar{\boldsymbol{x}}$ is the true position of the mobile device and $\hat{\boldsymbol{x}}$ is its estimate. RMSE averaging is over 1000 mobile positions picked randomly in a cuboid of 1000m x 1000m x 50m (Fig. 3). 500 fixed reference points inside the red cuboid are chosen to take measurements. At the height of 25m, 8 base stations form a circle of radius 200m. Other 8 base stations form a simimar circle at the height of 150m. In MATLAB, we generate the RSS values with the fixed path loss exponent $\alpha = 2$ and the standard deviation (σ) varies from 0.5 to 3 dBm. The dataset for training each ANN (containing 3 hidden layers and 20 nodes in each layer) is obtained by transmitting signals from the 500 fixed reference points to the 16 base stations. Stopping criterion for the iterative ML estimator is ε_{RSS} = 0.01. Other parameters: d_0 = 1m and $P_0 = -45$ dBm.

B. Results

Instead of comparing the MSEs to the CRLB, we compare their square roots: The Root Mean Square Error (RMSE) = $\sqrt{\text{MSE}}$ and square root of CRLB ($\sqrt{\text{CRLB}}$).

Fig. 4 compares the RMSEs of the 4 algorithms:

(a) Position estimation by the iterative ML procedure when path loss exponent (α) and standard deviation of RSS measurement (σ) are already given, assuming that all the RSS measurements have the same standard deviation and all the transmission links have the same path loss exponents [13].
(b) Position estimation by the ML iterative procedure when α and σ are estimated by taking mean values of all the estimated $\hat{\alpha}_i$ and $\hat{\sigma}_i$, respectively, of the dataset, also assuming that all the RSS measurements have the same standard deviation and all the transmission links have the same path loss exponents.
(c) Using an ANN to directly determine the position of the mobile device [16].

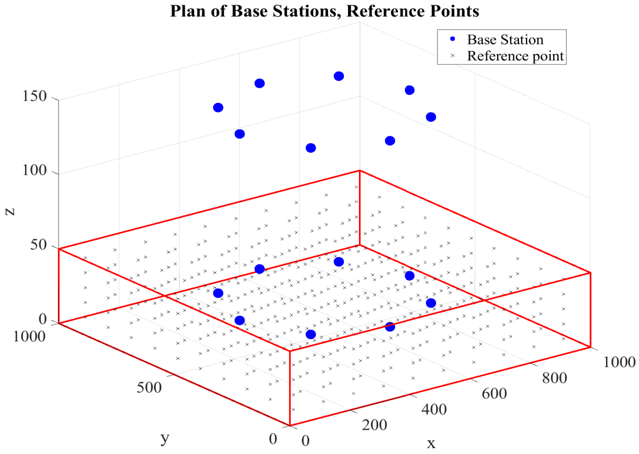

Figure 3: Map of base stations' network, reference points and the cuboid where the mobile device's position is randomly picked up inside

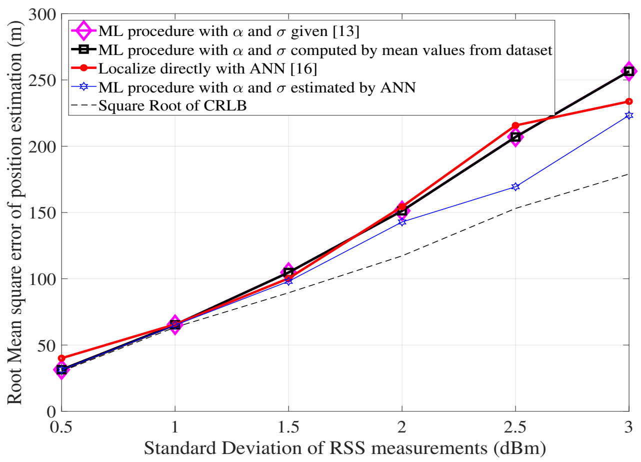

Figure 4: RMSE comparison among the algorithms

(d) Using ANNs to estimate each path loss exponent and each standard deviation, then using the proposed iterative ML procedure to estimate position of the mobile device (our proposed algorithm).

The $\sqrt{\text{CRLB}}$ is also added to validate the performance of our proposed algorithm.

From the results shown in Fig. 4, it is obvious that our proposed algorithm (d) **always** gives the best performance, with the lowest RMSE. This RMSE is still larger than the $\sqrt{\text{CRLB}}$, which proves that the estimation is unbiased.

Evidently, with the help of ANNs in estimating standard deviations and path loss exponents, our RSS-based algorithm is considerably improved with lower RMSE, specially when the log normal fadings are large, but still keeps the estimation unbiased.

VI. Conclusions

This paper develops the geometric interpretation of a trilateration algorithm in 3D, and propose an iterative Maximum Likelihood procedure to enhance the accuracy of the positioning. The prior estimations of standard deviations and path loss exponents by ANNs provide more exact results for localization than the ordinary estimation.

All of the conclusions offered, however, are based on simulations. It would be interesting to study how the proposed positioning algorithm performed on measurements.

Acknowledgment

EURECOM's research is partially supported by its industrial members: ORANGE, BMW, Symantec, SAP, Monaco Telecom, iABG. The Research of EURECOM and ORANGE was also supported in part by the French FUI project GEOLOC.

References

[1] J. Pearl-Rosado, R. Raulefs, J. A. Lopez-Salcedo, and R. Seco-Granados, "Survey of cellular mobile radio localization methods: From 1G to 5G", *IEEE Commun. Surveys Tuts.*, vol. 20, pp. 1124-1148, 2nd Quart. 2018.

[2] T. K. Le, N. Ono, *Closed-form and near closed-form solutions for TOA-based joint source and sensor localization*, IEEE Trans. Signal Process., vol. 64, no. 18, pp. 4751-4766, Sep. 2016.

[3] A. Moragrega, P. Closas, C. Ibars, *Supermodular game for power control in TOA-based positioning*, IEEE Trans. Signal Process., vol. 61, no. 12, pp. 3246-3259, Jun. 2013.

[4] N. H. Nguyen, K. Doanay, *Optimal geometry analysis for multistatic TOA localization*, IEEE Trans. Signal Process., vol. 64, no. 16, pp. 4180-4193, Aug. 2016.

[5] T. K. Le, N. Ono, *Closed-form and near closed-form solutions for TDOA-based joint source and sensor localization*, IEEE Trans. Signal Process., vol. 65, no. 5, pp. 1207-1221, Mar. 2017.

[6] L. Wang, T. K. Hon, J. D. Reiss, A. Cavallaro, *Self-localization of ad-hoc arrays using time difference of arrivals*, IEEE Trans. Signal Process., vol. 64, no. 4, pp. 1018-1033, Feb. 2016.

[7] T. L. N. Nguyen, Y. Shin, *A new approach for positioning based on AOA measurements*, International Conference on Computing, Management and Telecommunications (ComManTel), 2013.

[8] C. S. Chen; Y. J. Chiu; J. M. Lin; C. H. Lee, *Geometrical Positioning Schemes for MS Location Estimation*, International Symposium on Computer, Consumer and Control, 2012.

[9] J. Xu, M. Ma, C. L. Law *AOA Cooperative Position Localization*, IEEE GLOBECOM 2008.

[10] Y. Wang, K.C.Ho; G. Wang, *A Unified Estimator for Source Positioning and DOA Estimation Using AOA*, IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2018.

[11] D. Munoz, F. Bouchereau, C. Vargas, R. Enriquez, *Position Location Techniques and Applications*, Academic Press, 2009.

[12] H. M. Le, J-P. Rossi, D. Slock, *A Geometric Interpretation of Trilateration for RSS-based Localization*, 28th European Signal Processing Conference (EUSIPCO), 2020.

[13] S. Tomic, M. Beko, R. Dinis, *RSS-Based Localization in Wireless Sensor Networks Using Convex Relaxation: Noncooperative and Cooperative Schemes*, IEEE Transactions on Vehicular Technology, 2015.

[14] J. Shirahama, T. Ohtsuki, *RSS-Based Localization in Environments with Different Path Loss Exponent for Each Link*, VTC Spring 2008.

[15] D. Li, J. Huang, *RSS based method for sensor localization with unknown transmit power and uncertainty in path loss exponent*, 8th IEEE International Conference on Communication Software and Networks (ICCSN), 2016.

[16] S. Jung; C. Im; C. Eom; C. Lee, *A Localization Method with Insufficient RSS samples using Neural Network*, International Conference on Artificial Intelligence in Information and Communication (ICAIIC), 2020.

[17] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*, Prentice-Hall, 1993.

[18] Y. Viniotis, *Probability and Random Processes for Electrical Engineers*, New York: McGraw-Hill, 1998.

[19] Y. L. Tong, *The Multivariate Normal Distribution*, Springer, 1989, chapter 3.
[20] Å. Björck, *Numerical Methods for Least Squares Problems*, SIAM (Society for Industrial and Applied Mathematics), 1996, chapter 9.
[21] H. M. Le and D. Slock and J.-P. Rossi, *2D DDoA-based self-positioning for mobile devices*, European Signal Processing Conference (EUSIPCO), 2021.
[22] S. Haykin, *Neural Networks: A Comprehensive Foundation*, Prentice Hall, 1994.
[23] M.T. Hagan, M.B. Menhaj, *Training feedforward networks with the Marquardt algorithm*, IEEE Transactions on Neural Networks (Volume: 5, Issue: 6, Nov. 1994).
[24] J. Arif, N. R. Chaudhuri, S. Ray, B. Chaudhuri, *Online Levenberg-Marquardt algorithm for neural network based estimation and control of power systems*, International Joint Conference on Neural Networks, 2009.
[25] C. Lv, Y. Xing, J. Zhang, X. Na, Y. Li, T. Liu, D. Cao, F-Y. Wang, *Levenberg–Marquardt Backpropagation Training of Multilayer Neural Networks for State Estimation of a Safety-Critical Cyber-Physical System*, IEEE Transactions on Industrial Informatics (Volume: 14, Issue: 8, Aug. 2018).
[26] J. Gurland, R. C. Tripathi, *A Simple Approximation for Unbiased Estimation of the Standard Deviation*, The American Statistician Vol. 25, No. 4 (Oct., 1971), pp. 30-32.
[27] R. Sari, H. Zayyani, *RSS Localization Using Unknown Statistical Path Loss Exponent Model*, IEEE Communications Letters (Volume: 22, Issue: 9, Sept. 2018)
[28] H. F. Ates, S. M. Hashir, T. Baykas, B. K. Gunturk, *Path Loss Exponent and Shadowing Factor Prediction From Satellite Images Using Deep Learning*, IEEE Access (Volume: 7), 2019.

APPENDIX

A. Unbiased estimation of d_i^2

From (8), we have

$$\begin{aligned}\frac{-(\ln 10)\gamma_{\mathrm{RSS},i}}{5\alpha_i} &= 2(\ln 10)(\log d_i) + \frac{-(\ln 10)n_i}{5\alpha_i} \\ &= 2(\ln d_i) + \frac{-(\ln 10)n_i}{5\alpha_i}\end{aligned} \tag{25}$$

As $n_i \sim \mathcal{N}(0,\sigma_i^2)$, $\frac{-(\ln 10)\gamma_{\mathrm{RSS},i}}{5\alpha_i}$ has the mean value of $2(\ln d_i)$ and the variance of $\frac{(\ln 10)^2}{25\alpha_i^2}\sigma_i^2$.

It is already proved in [18] that if q is a Gaussian distributed variable, the mean and variance of q will be $\mathrm{E}(e^q) = \exp\left(\mu + \frac{\sigma^2}{2}\right)$ and $\mathrm{var}(e^q) = \left(e^{\sigma^2} - 1\right)e^{2\mu+\sigma^2}$. Applying those results, the mean and variance of $\exp\left(\frac{-(\ln 10)\gamma_{\mathrm{RSS},i}}{5\alpha_i}\right)$ are computed as:

$$\mathrm{E}\left(\exp\left(\frac{-(\ln 10)\gamma_{\mathrm{RSS},i}}{5\alpha_i}\right)\right) = \exp\left(2\ln d_i + \frac{(\ln 10)^2\sigma_i^2}{50\alpha_i^2}\right) \tag{26}$$

and

$$\begin{aligned}&\mathrm{var}\left(\exp\left(\frac{-(\ln 10)\gamma_{\mathrm{RSS},i}}{5\alpha_i}\right)\right) \\ &= \left(\exp\left(\frac{(\ln 10)^2}{25\alpha_i^2}\sigma_i^2\right) - 1\right)\exp\left(4\ln d_i + \frac{(\ln 10)^2}{25\alpha_i^2}\sigma_i^2\right)\end{aligned} \tag{27}$$

Thus, from (26), we get

$$\mathrm{E}\left(\exp\left(\frac{-(\ln 10)\gamma_{\mathrm{RSS},i}}{5\alpha_i}\right)\right) = d_i^2\exp\left(\frac{(\ln 10)^2\sigma_i^2}{50\alpha_i^2}\right) \tag{28}$$

Consequently, the unbiased estimate of d_i^2 is

$$\widehat{d_i^2} = \exp\left(\frac{-(\ln 10)\gamma_{\mathrm{RSS},i}}{5\alpha_i} - \frac{(\ln 10)^2\sigma_i^2}{50\alpha_i^2}\right) \tag{29}$$

B. Estimation of var(d_i^2)

Furthermore, from (27), we have:

$$\begin{aligned}&\mathrm{var}\left(\exp\left(\frac{-(\ln 10)\gamma_{\mathrm{RSS},i}}{5\alpha_i}\right)\right) \\ &= d_i^4\exp\left(\frac{(\ln 10)^2}{25\alpha_i^2}\sigma_i^2\right)\left[\exp\left(\frac{(\ln 10)^2}{25\alpha_i^2}\sigma_i^2\right) - 1\right]\end{aligned} \tag{30}$$

From (25), we have

$$\frac{-(\ln 10)\gamma_{\mathrm{RSS},i}}{2.5\alpha_i} = 4(\ln d_i) + \frac{-(\ln 10)n_i}{2.5\alpha_i} \tag{31}$$

Hence, $\frac{-(\ln 10)\gamma_{\mathrm{RSS},i}}{2.5\alpha_i}$ has the mean value of $4(\ln d_i)$ and the variance of $\frac{(\ln 10)^2}{6.25\alpha_i^2}\sigma_i^2$

Similarly to the section A, we can compute the mean of $\exp\left(\frac{-(\ln 10)\gamma_{\mathrm{RSS},i}}{2.5\alpha_i}\right)$ by applying the results in [18].

$$\begin{aligned}\mathrm{E}\left(\exp\left(\frac{-(\ln 10)\gamma_{\mathrm{RSS},i}}{2.5\alpha_i}\right)\right) &= \exp\left(4\ln d_i + \frac{(\ln 10)^2\sigma_i^2}{12.5\alpha_i^2}\right) \\ &= d_i^4\exp\left(\frac{(\ln 10)^2\sigma_i^2}{12.5\alpha_i^2}\right)\end{aligned} \tag{32}$$

Thus, the unbiased estimate of d_i^4 is

$$\hat{d_i^4} = \exp\left(\frac{-(\ln 10)\gamma_{\mathrm{RSS},i}}{2.5\alpha_i} - \frac{(\ln 10)^2\sigma_i^2}{12.5\alpha_i^2}\right) \tag{33}$$

(9) gives us the unbiased estimate of d_i^2, so its variance can be computed by

$$\mathrm{var}(d_i^2) = \mathrm{var}\left(\exp\left(\frac{-(\ln 10)\gamma_{\mathrm{RSS},i}}{5\alpha_i}\right)\right)\exp\left(\frac{-(\ln 10)^2\sigma_i^2}{25\alpha_i^2}\right) \tag{34}$$

Using (30), we have

$$\begin{aligned}&\mathrm{var}(d_i^2) \\ &= d_i^4\exp\left(\frac{(\ln 10)^2}{25\alpha_i^2}\sigma_i^2\right)\left[\exp\left(\frac{(\ln 10)^2}{25\alpha_i^2}\sigma_i^2\right) - 1\right]\exp\left(\frac{-(\ln 10)^2\sigma_i^2}{25\alpha_i^2}\right) \\ &= d_i^4\left[\exp\left(\frac{(\ln 10)^2}{25\alpha_i^2}\sigma_i^2\right) - 1\right]\end{aligned} \tag{35}$$

Since the unbiased estimate of d_i^4 is given in (33), we have the equation of var(d_i^2):

$$\begin{aligned}&\mathrm{var}(d_i^2) \\ &= \exp\left(\frac{-(\ln 10)\gamma_{\mathrm{RSS},i}}{2.5\alpha_i} - \frac{(\ln 10)^2\sigma_i^2}{12.5\alpha_i^2}\right)\left[\exp\left(\frac{(\ln 10)^2}{25\alpha_i^2}\sigma_i^2\right) - 1\right]\end{aligned} \tag{36}$$