One-to-one non-linear transformation for RSS-based localization with unknown transmit power

Meysam Raees Danaee

Abstract
For most of the positioning algorithms, based on the received signal strength, the objective function of the estimator is linearized so that the conventional least square algorithm can be executed. This would necessitate using a data model containing many noisy elements which implicitly violates the least squares assumptions. To avoid this violation, an optimized data modelling would have to be set up as shown in this paper. The authors then trivially provide the solution of this new least square problem which is still a primary coarse position estimation because of ignoring the non-linear relationships existing among parameters of the estimation. To extract quantitative information from these non-linear relationships, the authors also propose two new signal space points for using Taylor-series expansion as effectively as possible in accordance of the new data model. The proposed algorithms work without any knowledge about the transmit power. The performance evaluation of the proposed idea indicates that the new suggested estimators perform significantly better than the other existing linear position estimators for a large range of noise variance and path-loss exponent values.

1 | INTRODUCTION

Wireless sensor networks (WSNs) provide the application level functionality in a variety of environments. To name just a few: reporting events from very close viewpoints (almost inside a phenomenon), functioning without direct commands, and acting as relays with the aim of providing hidden routers in non-line-of-sight (NLOS) scenarios (to reduce possible interference with other electromagnetic devices) [1, 2].

It is typical to assume that sensors are deployed uniformly at random in a large area. Therefore, their position is unknown already and should be estimated to provide reported events with clear tags. The large number of WSN nodes makes it costly to equip each node with the Global Positioning System (GPS) and some cheaper solutions should be sought to address WSN node positioning.

Their locations should be calculated by using exchanged signalling messages between network nodes and some pre-localized sensors [3, 4]. To that end, different types of signal can be used which may be divided into the four major categories: (i) Time-of-arrival (TOA)-based localization [5, 6]; (ii) Time-difference-of-arrival (TDOA) localization [7, 8]; (iii) Angle-of-arrival (AOA) localization [9, 10]; and (iv) Received-signal strength (RSS) [11, 12].

The RSS method requires less complex hardware circuitry which makes it a practical alternative option to be applied for many applications such as detecting the primary user emulation in cognitive radio networks [13] or positioning in underwater acoustic WSNs [14].

Although the simple hardware architecture is required for the RSS localization method, its measurement model is highly non-linear and some non-linear routines should be applied to find its global minimum solution. Unfortunately, using any numerical optimization method may result in getting stuck into a local rather than a global optimum solution. Two main approaches have been adopted in the literature to study numerical stability: (1) relaxing objective function and constraints based on the convex theory [15–18], and (2) applying modified least squares (LS) methods [19, 20].

In the first approach, the original objective function and the given problem constraints are relaxed to be expressed as a semi-definite program (SDP) [15, 16]. The reason is that efficient
algorithms which are utilized in the linear programs, for example interior point methods, can be generalized for solving SDPs [21]. This approach takes the advantage of convex optimization which is it can be guaranteed to find a global optimum with high accuracy, independently of the initial guess assumed. The drawback is, however, the inability to find a closed-form solution. Consequently, sometimes this leads to unreasonable computation burden. The complexity may be reduced by using another type of algorithm requiring a smaller number of constraints and variables for solving a convex optimization problem. This algorithm is called second-order cone program (SOCP) and applied in the context of localization problem by [17].

Another approach to avoid getting stuck in local solutions is to use LS methods which are very popular in localization problem, [19, 20, 22–24]. The LS methods have closed-form solutions and involve only light computation, compared to the SDP or SOCP methods. However, the LS methods are applicable only under linearity conditions. We need to use an approximated linearized form of the non-linear relation between the source node-to-receiver node distance and the coordinates of a source node for the RSS-based localization problem. It is inevitable that certain information is lost as a result of this approximation.

One of the possible solutions which has been offered yet for recovering such an information loss is to apply two successive LS filters where the first one obtains a coarse position coordinates and the second exploits the known relation between the parameter estimates for improving the position estimate [19, 20, 24].

In other words, these methods, including the methods applied in this paper, propose two-stage algorithms for node localization. The position estimation of a node is refined after two successive stages of the LS algorithms. For the first stage, the coarse estimate of a node position is obtained by using linear LS. The estimate is then made better in the next successive stage.

The present paper brings about two new improvements to such offered solution mentioned above. The first improvement tries to optimize the data modelling so that it will be in accordance with the pre-requisite of using LS methods. The pre-requisite is that only the observation vector should be noisy whereas the data matrix entries should be certain and noise-free. Still, this is not what happens in current data modelling of using the RSS measurements for the sake of localization. We use one-to-one non-linear transformation and define a kind of inverted-range variable so that it is only the observation vector whose components are noisy in the new proposed data model.

The second improvement deals with how we use the coarse estimate obtained from the first LS to present an effective Taylor series expansion. We propose new signal space points for using Taylor series expansion as effectively as possible in accordance of new data model. This way better refinement step is carried out and more information is utilized.

The proposed algorithms work in an environment when there is no knowledge about the transmitting power level. The simulation results confirm the significant accuracy improvement obtained by our proposed methods over some similar state-of-art approaches.

The paper is organized as follows. Section 2 examines the signal parameter model and then states the problem of interest in this paper. Section 3 introduces our two proposed methods related to a WSN node localization when the transmitted power is unknown. In that section, we describe their features in detail by illustrating various schematic diagrams. Those diagrams display the algorithm’s components. We also show a comparison table contrasting assumptions associated with each of these four algorithms. Then the performances of these proposed methods are evaluated on different scenarios and SNR’s in Section 4. Finally, the conclusion is given in Section 5.

2 SIGNAL MODEL AND PROBLEM FORMULATIONS

In the notation of this paper, scalars may be expressed in lower case italic (a, b, ...), matrices are denoted in upper case bold (A, B, ...), and vectors are shown in lower case bold italic letters (a, b, ...). The transpose operation is denoted by the superscript T. The identity matrix is denoted by I.

The coordinates of an unknown node are denoted by (x, y). There need to be at least L = 4 receivers, for node coordinate estimation in 2D space when the transmit power is unknown. These receivers are pre-localized sensors and equipped with GPS and thus are aware of their locations. They are called anchor or beacon nodes.

The lth anchor’s coordinates are shown as (xl, yl) for l = 1, 2, ..., L. The location of these anchor nodes undoubtedly affect the quality of the estimation performance. A comprehensive study of this topic may be found in [25] and [26], but in this paper we will not focus on optimizing a set of anchor nodes displacement.

The distance dl between a node and the lth receiver is calculated as:

\[ d_l = \sqrt{(x - x_l)^2 + (y - y_l)^2}. \] (1)

An unknown node communicates with all of the anchor nodes which are located within its radio range to find its location. The setup is shown in Figure 1.

The average RSS from the source node at the lth receiver is denoted by Pl and is expressed in decibel scale by

\[ 10 \log_{10} (P_l) = 10 \log_{10} (R_l) - 10 \alpha \log_{10} (d_l / d_0) + w_l, \] (2)

where \( \alpha \) is the path-loss exponent (PLE) of the environment, d0 is a reference distance from a node, Rl is the known and exact RSS measurement transmitted from a node and received by a receiver at the distance dl.

It is important to note, however, that, if the PLE values do not vary in different environments (the underlying assumption
in the present analyses), then the value of \( P_b \) would be identical in all receivers. \( P_b \) encompasses information about the source transmit power. \( w_l, \ l = 1, 2, ..., L \), are uncorrelated Gaussian random variables indicating the shadowing noises which are assumed to be zero-mean with known variances \( \sigma^2_l \). This paper takes \( \sigma^2_l = \sigma^2 \) for all sensors. It should also be noted that realistic modelling of the shadowing noise probability density functions instead of a Gaussian distribution is followed in [27] based on the Dempster–Shafer theory.

For the case of unknown transmit power, the differential received signal strength (DRSS) measurements to cancel out the effect of \( P_b \). Without loss of generality, we assume that the 1st receiver is the reference receiver node. DRSS measurements \( R_d(l, 1) \) of the \( l \)th anchor node can be obtained as

\[
R_d(l, 1) = 10 \log_{10} (P_l) - 10 \log_{10} (P_1)
\]

\[
= -10 \alpha \log_{10} \left( \frac{d_l}{d_1} \right) + w_l - w_1
\]

\[
= -10 \alpha \ln \left( \frac{d_l}{d_1} \right) + w_l - w_1, \ l = 2, ..., L,
\]

where \( \{R_d(l, 1)\}, \ l = 2, ..., L \), are Gaussian random variables and \( \alpha = \delta / \ln 10 \). There is no estimator to attain the Cramer–Rao lower bound (CRLB) based on the measurement \( R_d(l, 1) \). Inspired by the idea developed in [28], it is trivial to show that the best unbiased estimator for \( (d_l/d_1) \) is the only unbiased estimate and is given by

\[
\left( \frac{d_l}{d_1} \right)_u = \exp \left( \frac{- \sigma^2}{100 \alpha^2} \right) \exp \left( - \frac{R_d(l, 1)}{10 \alpha} \right).
\]

It is very difficult, because of non-linear effects, to obtain node coordinates \( \{x, y\} \) from \( (d_l/d_1)_u \). A simpler idea is given in [29] and [20] and it is to take the ratio of estimated functions \( \hat{d}_l \) and \( \hat{d}_1 \) as the estimation of the ratio of two functions \( d_l \) and \( d_1 \). In other words, the below approximation is adopted

\[
\left( \frac{\hat{d}_l}{\hat{d}_1} \right)_u \approx \frac{d_l}{d_1},
\]

But this idea is applicable only when the noise power is much less than the signal power [20]. However, for the sake of simplicity, we follow the approximation given in (5). It results in

\[
\hat{d}_l = \hat{d}_1 \exp \left( -\frac{1}{100 \alpha^2} \left( \sigma^2 + 10 \alpha R_d(l, 1) \right) \right).
\]

By taking the square of the both sides of (6), we define \( \beta_{\hat{d}_l} \) by

\[
\beta_{\hat{d}_l} = \exp \left( -\frac{1}{50 \alpha^2} \left( \sigma^2 + 10 \alpha R_d(l, 1) \right) \right), \quad l = 2, ..., L.
\]

Substituting \( \hat{d}_l \) and \( \hat{d}_1 \) by their equivalent functions of the inputs in (1), we obtain

\[
(1 - \beta_{\hat{d}_l}) (x^2 + y^2) + \left( -2x_l + 2x_l \beta_{\hat{d}_l} \right) x
\]

\[
+ \left( -2y_l + 2y_l \beta_{\hat{d}_l} \right) y
\]

\[
= -x_l^2 + y_l^2 + \beta_{\hat{d}_l} (x_l^2 + y_l^2).
\]

In order to obtain a linear matrix model from (8), we define \( R \) to be a range variable equals \( x^2 + y^2 \). Tacking all the measurements into account, (8) takes a matrix form

\[
A \theta = b,
\]

where data matrix \( A \) and unknown parameter vector \( \theta \) are given by

\[
A = \begin{bmatrix}
-2x_2 \beta_{\hat{d}_2,1}^{-1} + 2x_1 & -2y_2 \beta_{\hat{d}_2,1}^{-1} + 2y_1 & \beta_{\hat{d}_2,1}^{-1} - 1 \\
-2x_3 \beta_{\hat{d}_3,1}^{-1} + 2x_1 & -2y_3 \beta_{\hat{d}_3,1}^{-1} + 2y_1 & \beta_{\hat{d}_3,1}^{-1} - 1 \\
\vdots & \vdots & \vdots \\
-2x_L \beta_{\hat{d}_L,1}^{-1} + 2x_1 & -2y_L \beta_{\hat{d}_L,1}^{-1} + 2y_1 & \beta_{\hat{d}_L,1}^{-1} - 1
\end{bmatrix},
\]

\[
\theta = \begin{bmatrix}
x & y & R^T
\end{bmatrix}^T,
\]

and observation vector \( b \) is given by

\[
b = \begin{bmatrix}
-(x_2^2 + y_2^2) \beta_{\hat{d}_2,1}^{-1} + (x_1^2 + y_1^2) \\
-(x_3^2 + y_3^2) \beta_{\hat{d}_3,1}^{-1} + (x_1^2 + y_1^2) \\
\vdots \\
-(x_L^2 + y_L^2) \beta_{\hat{d}_L,1}^{-1} + (x_1^2 + y_1^2)
\end{bmatrix}.
\]
The linear model (9) with elements defined from (10) to (12) is conventional in literature dealing with unknown transmit power, see [29] and [20] for instance.

However, we realize that it is possible to develop a linear model the other way around, let us say, better fitted to be used in the LS objective in the localization problem, when there is no knowledge about the transmit power.

The developed model is first described, followed by discussion about its superiority over using the model (9) when it comes to apply LS algorithms.

The developed model is an alternative representation of (9) and drawn with assuming the 1st reference receiver node is located at the origin. The new model is given by

$$\Lambda \xi = \zeta,$$  

where

$$\Lambda = \begin{bmatrix} -2x_2 & -2y_2 & x_2^2 + y_2^2 \\ -2x_3 & -2y_3 & x_3^2 + y_3^2 \\ \vdots & \vdots & \vdots \\ -2x_L & -2y_L & x_L^2 + y_L^2 \end{bmatrix},$$  

$$\xi = \begin{bmatrix} x/R \ y/R \ 1/R \end{bmatrix}^T,$$

and

$$\zeta = \begin{bmatrix} \beta_{2,1} - 1 \\ \beta_{3,1} - 1 \\ \vdots \\ \beta_{L,1} - 1 \end{bmatrix}.$$

The proof is presented in Appendix A.

Compared to (9), the model (13) has unique properties. First, the LS estimate is entirely consistent with the model (13). There is no uncertain term involved in matrix $\Lambda$. All noise-sources are concentrated in the vector $\zeta$. However, this is not true for the model (9). Not only the vector $b$ but also the matrix $A$ has noisy elements $\beta_{ii}^{-1}$, $i = 2, \ldots, L$. Second, the variables are inversely and non-linearly related to node coordinates. This is, though, the price that should be paid for making the model (13) be entirely consistent with the LS estimate. The linear LS estimates the parameter vector $\hat{\xi}$ and the node coordinates $x$ and $y$ are recovered from the estimated parameter vector, $\hat{\xi}$, by a non-linear one-to-one transformation. Using a non-linear one-to-one transformation to convert a non-linear LS problem into a linear LS problem is a well-known estimation method [30].

### 3 | LOCATION ESTIMATORS FOR ONE-TO-ONE NON-LINEAR TRANSFORMATION

#### 3.1 | One-stage LS

If the variance of shadowing noise $\sigma^2$ is more than zero, then there cannot exist any $\hat{\xi}$ as a solution to satisfy the system of equations defined in (13). It seems to be more appropriate if (13) is shown as an overdetermined system of equations as follows:

$$\Lambda \hat{\xi} \approx \zeta.$$

The main idea, laid down in the LS estimation approach, is to correct the inner perturbation included in $\hat{\xi}$ by adding $\Delta \hat{\xi}$ as a correction term to the observation vector $\hat{\xi}$ so that after this treatment the corrected observation vector gives a uniquely-solvable system of equations as follows:

$$\Lambda \xi = \zeta + \Delta \xi.$$  

However, the requirement imposed by the LS estimator is that the correction term $\Delta \hat{\xi}$ should be as little as possible in the Frobenius norm sense. This may be expressed as the following optimization problem:

$$\left\{ \hat{\xi}_{LS}, \Delta \hat{\xi}_{LS} \right\} = \arg \min_{\xi, \Delta \xi} \| \Delta \xi \|_F, \text{ subject to} \Lambda \xi = \zeta + \Delta \xi.$$  

The well-known solution to the optimization problem defined in (19) is

$$\hat{\xi}_{LS} = \left( \Lambda^T \Lambda \right)^{-1} \Lambda^T \xi.$$  

The estimation of vector $\hat{\xi}$ is actually half the story. Its entries are non-linearly related to the node coordinates $(x, y)$. However, we reflect the fact that minimization in the original space can be done in a non-linear transformed space according to [30, Chap. 8]. Conditional on using a one-to-one transformation, the results can be converted by the corresponding inverse transformation to yield the desired estimation in the original space. This means that the result of estimation in the new space, let us say $\hat{\xi}_{LS}$, can easily be converted in order to estimate the node coordinates $(x, y)$. Using the inverse transformation of the one presented in (15) is sufficient to reverse the map to the original space of $(x, y)$ variables. This is given by

$$x = \frac{\hat{\xi}_1}{\hat{\xi}_3}, y = \frac{\hat{\xi}_2}{\hat{\xi}_3}.$$  

On the side of this, the estimation of node coordinates obtained by one-step LS is denoted by $\hat{\xi}_{IR\text{-OSLS}}, \hat{\xi}_{IR\text{-OSLS}},$.
where IR-OSBL is the abbreviation for Inverted Range One-Stage Least Squares estimation approach. They are given by

\[
\begin{align*}
\hat{x}_{\text{IR-OSLS}} &= \frac{\hat{x}_1}{\hat{x}_3}, \quad \hat{x}_{\text{IR-OSLS}} = \frac{\hat{x}_2}{\hat{x}_3}.
\end{align*}
\] (22)

### 3.2 One-stage BLUE

The one-stage LS estimation method developed in Section 3.1 does not take statistical information available in \( \xi \) into consideration. Contrary to the LS approach, the best linear unbiased estimator (BLUE) utilizes the covariance matrix of observation vector \( \zeta \) so that each of its elements has the minimum variance of all unbiased estimators constructed by any function which is linear in the elements of \( \zeta \).

The covariance matrix of observation vector \( \zeta \) is denoted by \( \mathbf{C} \) and is given by

\[
\mathbf{C}(i, i) = \left( \frac{d_{i+1}}{d_i} \right)^4 e^{a^2/50a^2} \left( e^{a^2/50a^2} - 1 \right),
\] (23)

and for off-diagonal \((i, j)\)th \((i \neq j)\) entries of \( \mathbf{C} \)

\[
\mathbf{C}(i, j) = \left( \frac{d_{i+1}}{d_i} \right)^2 \left( \frac{d_{j+1}}{d_j} \right)^2 e^{a^2/50a^2} \left( e^{a^2/50a^2} - 1 \right), \quad i \neq j.
\] (24)

The proof is presented in Appendix B.

Having \( \mathbf{C} \) at our disposal, the estimation of \( \xi \) obtained by BLUE is given by

\[
\hat{\xi}_{\text{BL}} = (\Lambda^T \mathbf{C}_{\xi}^{-1} \Lambda)^{-1} \Lambda^T \mathbf{C}_{\xi}^{-1} \xi,
\] (25)

where

\[
\text{Cov}(\hat{\xi}_{\text{BL}}) = (\Lambda^T \mathbf{C}_{\xi}^{-1} \Lambda)^{-1}.
\] (26)

As a result, the estimation of node coordinates obtained by one-step BLUE is denoted by \( \hat{x}_{\text{IR-OSBL}}, \hat{y}_{\text{IR-OSBL}} \) where IR-OSBL is the abbreviation for Inverted Range One-Stage Best Linear Unbiased Estimator. They are given by

\[
\hat{x}_{\text{IR-OSBL}} = \frac{\hat{x}_{\text{BL}1}}{\hat{x}_{\text{BL}3}}, \quad \hat{y}_{\text{IR-OSBL}} = \frac{\hat{y}_{\text{BL}2}}{\hat{y}_{\text{BL}3}},
\] (27)

where \( \hat{x}_{\text{BL}i} \) denotes the \( i \)th entry of \( \hat{x}_{\text{BL}} \).

### 3.3 Two-stage BLUE

In order to obtain the 2D coordinates of a node \((x, y)\) for the two methods which are introduced in Sections 3.1 and 3.2, it is only required to divide the first and second elements of the vector \( \xi \) by its third element, respectively.

It is clear that there are relationships among entries of the vector \( \xi \) as follows:

\[
\xi_1^2 + \xi_2^2 = \xi_3.
\] (28)

The effect of (28) is not taken into account when obtaining the estimated results for the one-stage BLUE and one-stage LS.

In this section, we want to improve the localization accuracy by exploiting the information offered by (28) in another stage following the one-stage BLUE. As the above described methods, the localization is performed without considering the non-linear dependency exists between estimate parameters for the first stage. During the second stage, it is tried to include as much available knowledge of this dependency as possible. This is done to further improve the position estimation obtained from the first stage and update it. For this reason, we called our improved localization method two-stage BLUE.

In other words, two-stage BLUE can be thought as one-stage BLUE followed by an accuracy improving stage which is a stage that uses information available in the entries of vector \( \xi \) to improve the localization accuracy.

There are two ways to develop the idea of two-stage BLUE. A first approach may be taken by using the Taylor series for quadratic forms of the entries of \( \hat{\xi}_{\text{BL}} \). The Taylor series expansion apply the rough position estimate result as its nominal point to extract information which has been attributed to an inherent non-linearity in the estimation parameters.

The quadratic form expansion is common in localization literature, e.g. see [16, 19, 20]. The reason for the popularity of this approach is using quadratic form is simple and well-behaved to apply Taylor series. Since the extracted information by this method is inversely related to the desire position variables, one extra step is required to convert this information into usable format. This extra conversion itself is a source of information loss or, better yet, a cause of entailing an invalid information.

Another approach to two-stage BLUE algorithm is to use the Taylor series for variables which have been already directly proportional to the position variables and no information is lost in the subsequent conversion process. The former approach extract information from variables directly proportional to \( \hat{\xi}_{\text{BL}} \) entries and the latter approach deals with variables inversely proportional to \( \hat{\xi}_{\text{BL}} \) entries.

#### 3.3.1 Two-stage BLUE for functions directly proportional to \( \hat{\xi}_{\text{BL}} \)

For the first approach of using two-stage BLUE, the information is extracted from the entries of vector \( \xi \) by using a Taylor series. We expand \( x^2/R^2 \) and \( y^2/R^2 \) around the nominal operating points obtained from (25). These points are, respectively,
\[ \left( \frac{\hat{\xi}_{BL,1}}{R} \right)^2 = \left( \hat{\xi}_{BL,1} \right)^2 + 2 \left( \frac{\hat{\xi}_{BL,1}}{R} \right) \frac{\partial \left( \frac{\hat{\xi}_{BL,1}}{R} \right)}{\partial \frac{\hat{\xi}_{BL,1}}{R}} \hat{\xi}_{BL,1} + \text{H.O.T.} \]

\[ = \left( \hat{\xi}_{BL,1} \right)^2 + 2 \left( \frac{\hat{\xi}_{BL,1}}{R} \right) \hat{\xi}_{BL,1} + \text{H.O.T.}, \]

where H.O.T. denotes the higher-order terms.

Neglecting higher-order terms, the equation system obtained from using the truncated Taylor series expansions up to the first order is

\[
\begin{bmatrix}
\hat{\xi}_{BL,1}^2 \\
\hat{\xi}_{BL,2}^2 \\
\hat{\xi}_{BL,3}^2 \\
\end{bmatrix} \approx \begin{bmatrix}
1 & 0 & \frac{\hat{\xi}_{BL,1}^2}{R^2} \\
0 & 1 & \frac{\hat{\xi}_{BL,2}^2}{R^2} \\
0 & 0 & 1 \\
\end{bmatrix} 
\begin{bmatrix}
\hat{\xi}_{BL,1} \\
\hat{\xi}_{BL,2} \\
\hat{\xi}_{BL,3} \\
\end{bmatrix} 
- \begin{bmatrix}
2\frac{\hat{\xi}_{BL,1}^2}{R^2} & 0 & 0 \\
0 & 2\frac{\hat{\xi}_{BL,2}^2}{R^2} & 0 \\
0 & 0 & 1 \\
\end{bmatrix} \begin{bmatrix}
\frac{\hat{\xi}_{BL,1}}{R} \\
\frac{\hat{\xi}_{BL,2}}{R} \\
\frac{\hat{\xi}_{BL,3}}{R} \\
\end{bmatrix}.
\]

Our adherence to (42) shows that we implicitly adopt a flexible attitude towards (39), implying that the only noisy term of the vector \( [\hat{\xi}_{BL,1}^2, \hat{\xi}_{BL,2}^2, \hat{\xi}_{BL,3}^2]^T \) is

\[
\begin{bmatrix}
2\frac{\hat{\xi}_{BL,1}^2}{R^2} & 0 & 0 \\
0 & 2\frac{\hat{\xi}_{BL,2}^2}{R^2} & 0 \\
0 & 0 & 1 \\
\end{bmatrix} \begin{bmatrix}
\frac{\hat{\xi}_{BL,1}}{R} \\
\frac{\hat{\xi}_{BL,2}}{R} \\
\frac{\hat{\xi}_{BL,3}}{R} \\
\end{bmatrix}.
\]

In other words, by totally ignoring H.O.T.,

\[
\text{Cov} \begin{bmatrix}
\hat{\xi}_{BL,1}^2 \\
\hat{\xi}_{BL,2}^2 \\
\hat{\xi}_{BL,3}^2 \\
\end{bmatrix} \approx \text{Cov} \begin{bmatrix}
2\frac{\hat{\xi}_{BL,1}^2}{R^2} & 0 & 0 \\
0 & 2\frac{\hat{\xi}_{BL,2}^2}{R^2} & 0 \\
0 & 0 & 1 \\
\end{bmatrix}.
\]

Another approximation is obtained by picking up the vector \( [2\hat{\xi}_{BL,1}, 2\hat{\xi}_{BL,2}, 1]^T \) instead of its statistical moment in (32). That way, the covariance matrix is approximated by

\[
\text{Cov} \begin{bmatrix}
\hat{\xi}_{BL,1}^2 \\
\hat{\xi}_{BL,2}^2 \\
\hat{\xi}_{BL,3}^2 \\
\end{bmatrix} \approx \begin{bmatrix}
2\frac{\hat{\xi}_{BL,1}^2}{R^2} & 0 & 0 \\
0 & 2\frac{\hat{\xi}_{BL,2}^2}{R^2} & 0 \\
0 & 0 & 1 \\
\end{bmatrix}
\cdot \text{Cov} \begin{bmatrix}
\frac{\hat{\xi}_{BL,1}}{R} \\
\frac{\hat{\xi}_{BL,2}}{R} \\
\frac{\hat{\xi}_{BL,3}}{R} \\
\end{bmatrix}.
\]

By substituting the result obtained in accordance with (26) for (33), the covariance matrix is given by

\[
\text{Cov} \begin{bmatrix}
\hat{\xi}_{BL,1}^2 \\
\hat{\xi}_{BL,2}^2 \\
\hat{\xi}_{BL,3}^2 \\
\end{bmatrix} \approx \begin{bmatrix}
2\frac{\hat{\xi}_{BL,1}^2}{R^2} & 0 & 0 \\
0 & 2\frac{\hat{\xi}_{BL,2}^2}{R^2} & 0 \\
0 & 0 & 1 \\
\end{bmatrix}
\cdot \left( \begin{bmatrix}
\hat{\xi}_{BL,1} \\
\hat{\xi}_{BL,2} \\
\hat{\xi}_{BL,3} \\
\end{bmatrix} \right)^{-1}.
\]

The assumptions made from (42) to (34) allow us to use the BLUE to estimate the parameter vector of (42) as follows:

\[
\begin{bmatrix}
\frac{\hat{\xi}_{BL,1}}{R} \\
\frac{\hat{\xi}_{BL,2}}{R} \\
\frac{\hat{\xi}_{BL,3}}{R} \\
\end{bmatrix} = \left( \begin{bmatrix}
1 & 0 \\
0 & 1 \\
1 & 1 \\
\end{bmatrix} \right)^{-1} \cdot \text{Cov}^{-1} \begin{bmatrix}
\hat{\xi}_{BL,1}^2 \\
\hat{\xi}_{BL,2}^2 \\
\hat{\xi}_{BL,3}^2 \\
\end{bmatrix}.
\]

The estimation obtained by using BLUE in (35) can be used to refine the location estimate of the one-stage BLUE in (25). The method is, therefore, called Inverted Range Two-Stage Best Linear Unbiased Estimator 1 (IR-TSBL1) where 1 is added to distinguish this approach from another way of using two-stage BLUE algorithm which will be described in
Section 3.3.2. Therefore, the obtained coordinates are shown hereafter by $\hat{x}_{IR\text{-TSBL1}}, \hat{y}_{IR\text{-TSBL1}}$ and given by

$$\hat{x}_{IR\text{-TSBL1}} = \text{sgn}(\hat{x}_{BL1}) \sqrt{\frac{(x/R)^2}{(x/R)^2 + (y/R)^2}},$$



$$\hat{y}_{IR\text{-TSBL1}} = \text{sgn}(\hat{x}_{BL2}) \sqrt{\frac{(y/R)^2}{(x/R)^2 + (y/R)^2}},$$

where the sgn function extracts the sign of a real number.

3.3.2 Two-stage BLUE for functions inversely proportional to $\hat{x}_{BL1}$

For the second approach of using two-stage BLUE, we may expand the vector of non-linear functions $[x^2, y^2, R]$ in powers of entries of the vector parameter $[x/R, y/R, 1/R]$ by means of the Taylor series around the results obtained in the first stage of estimating variables $x/R, y/R, 1/R$. The reason for choosing the vector of non-linear functions as $[x^2, y^2, R]$ is that its third entry can be eliminated since it is the summation of the first and the second vector entries. This leads to sharing out a whole of second stage new information only between estimations of $x^2$ and $y^2$ (and as a result between estimations of $x$ and $y$).

We expand $x^2$, $y^2$, and $R$ around the nominal operating points, $\hat{x}_{BL1}, \hat{x}_{BL2},$ and $\hat{x}_{BL3}$, obtained from (25). For example, $x^2 = (\hat{x}_{BL1}/\hat{x}_{BL3})^2$ is the function of only $\hat{x}_{1}$ and $\hat{x}_{3}$ thus the Taylor series expansions is

$$x^2 = \left(\frac{\hat{x}_{BL1}}{\hat{x}_{BL3}}\right)^2 + \left(\hat{x}_{1} - \frac{\hat{x}_{BL1}}{\hat{x}_{BL3}}\right) \frac{\partial (\hat{x}_{1}/\hat{x}_{3})^2}{\partial \hat{x}_1} \bigg|_{\hat{x}=\hat{x}_{BL1}}$$

$$+ \left(\hat{x}_{3} - \hat{x}_{BL3}\right) \frac{\partial (\hat{x}_{1}/\hat{x}_{3})^2}{\partial \hat{x}_3} \bigg|_{\hat{x}=\hat{x}_{BL1}} + \text{H.O.T.}$$

$$= \left(\frac{\hat{x}_{BL1}}{\hat{x}_{BL3}}\right)^2 + \left(\hat{x}_{1} - \frac{\hat{x}_{BL1}}{\hat{x}_{BL3}}\right) \cdot 2 \cdot \hat{x}_{1} \cdot \left(\frac{1}{\hat{x}_{3}}\right)^2$$

$$+ \left(\hat{x}_{3} - \hat{x}_{BL3}\right) \cdot (-2) \cdot \left(\frac{\hat{x}_{3}}{\hat{x}_{3}}\right) + \text{H.O.T.},$$

where H.O.T. denotes the higher-order terms. The other two entries $y^2$, and $R$ are expanded in the same way. The results are given by

$$y^2 = \left(\frac{\hat{x}_{BL2}}{\hat{x}_{BL3}}\right)^2 + \left(\hat{x}_{2} - \frac{\hat{x}_{BL2}}{\hat{x}_{BL3}}\right) \cdot 2 \cdot \hat{x}_{2} \cdot \left(\frac{1}{\hat{x}_{3}}\right)^2$$

$$+ \left(\hat{x}_{3} - \frac{\hat{x}_{BL3}}{\hat{x}_{BL3}}\right) \cdot (-2) \cdot \left(\frac{\hat{x}_{3}}{\hat{x}_{3}}\right)^3 + \text{H.O.T.},$$

and

$$R = \frac{1}{\hat{x}_{BL3}} + \left(\hat{x}_{3} - \frac{\hat{x}_{BL3}}{\hat{x}_{BL3}}\right) \cdot \frac{-1}{\hat{x}_{3}} + \text{H.O.T.}.$$

Form (41) to (48), the first-stage estimation results, $\hat{x}_{BLi}, i = 1, 2, 3$, are shown in the shortened form as $\hat{x}_{i}, i = 1, 2, 3$, substantially to save space.

Neglecting higher-order terms, the equation system obtained from using the truncated Taylor series expansions up to the first order is

$$\begin{bmatrix} x^2 \\ y^2 \\ R \end{bmatrix} \approx \begin{bmatrix} \hat{x}_{1}/\hat{x}_{3} \\ \hat{x}_{2}/\hat{x}_{3} \\ 1/\hat{x}_{3} \end{bmatrix}$$

$$+ \begin{bmatrix} \frac{2\hat{x}_{1}}{(\hat{x}_{3})^2} & 0 & -\frac{2(\hat{x}_{1})^2}{(\hat{x}_{3})} \\ 0 & \frac{2\hat{x}_{2}}{(\hat{x}_{3})^2} & -\frac{2(\hat{x}_{2})^2}{(\hat{x}_{3})} \\ 0 & 0 & -1/(\hat{x}_{3})^2 \end{bmatrix} \begin{bmatrix} \frac{x - \hat{x}_{1}}{\hat{x}_{1}} \\ \frac{y - \hat{x}_{2}}{\hat{x}_{2}} \\ \frac{1}{\hat{x}_{3}} - \hat{x}_{3} \end{bmatrix}.$$
where we have actually defined a new virtual linear model in which the new virtual observation vector is $[\xi_x/\xi_3, \xi_y/\xi_3, 1/\xi_3]^T$ and the new virtual unknown parameter vector is $[\xi^2, \xi^2]^{T}$. Also our adherence to (43) shows that we implicitly adopt a flexible attitude implying that the only noisy constituent term of the new virtual observation vector, $[\xi_x/\xi_3, \xi_y/\xi_3, 1/\xi_3]^T$, is

\[
\begin{bmatrix}
\frac{2\xi_1}{(\xi_3)} & 0 & -2\frac{(\xi_3)^2}{(\xi_3)} \\
0 & \frac{2\xi_2}{(\xi_3)} & -2\frac{(\xi_3)^2}{(\xi_3)} \\
0 & 0 & -1\frac{(\xi_3)^2}{(\xi_3)}
\end{bmatrix} \begin{bmatrix}
\frac{X}{\xi_3} - \frac{\xi_1}{\xi_3} \\
\frac{1}{\xi_3} - \frac{\xi_1}{\xi_3} \\
\frac{1}{\xi_3} - \frac{\xi_1}{\xi_3}
\end{bmatrix}.
\]  

(44)

In other words, by totally ignoring H.O.T.,

\[
\text{Cov}
\begin{bmatrix}
\frac{2\xi_1}{(\xi_3)} & 0 & -2\frac{(\xi_3)^2}{(\xi_3)} \\
0 & \frac{2\xi_2}{(\xi_3)} & -2\frac{(\xi_3)^2}{(\xi_3)} \\
0 & 0 & -1\frac{(\xi_3)^2}{(\xi_3)}
\end{bmatrix} \approx \text{Cov}
\begin{bmatrix}
\frac{2\xi_1}{(\xi_3)} & 0 & -2\frac{(\xi_3)^2}{(\xi_3)} \\
0 & \frac{2\xi_2}{(\xi_3)} & -2\frac{(\xi_3)^2}{(\xi_3)} \\
0 & 0 & -1\frac{(\xi_3)^2}{(\xi_3)}
\end{bmatrix}
\]

(45)

Another approximation is obtained by picking up the random matrix

\[
\begin{bmatrix}
\frac{2\xi_1}{(\xi_3)^2} & 0 & -2\frac{(\xi_3)^2}{(\xi_3)} \\
0 & \frac{2\xi_2}{(\xi_3)^2} & -2\frac{(\xi_3)^2}{(\xi_3)} \\
0 & 0 & -1\frac{(\xi_3)^2}{(\xi_3)}
\end{bmatrix}
\]  

(46)

Instead of its statistical moments in (45). That way, the covariance matrix is approximated by

\[
\begin{bmatrix}
\frac{2\xi_1}{(\xi_3)^2} & 0 & -2\frac{(\xi_3)^2}{(\xi_3)} \\
0 & \frac{2\xi_2}{(\xi_3)^2} & -2\frac{(\xi_3)^2}{(\xi_3)} \\
0 & 0 & -1\frac{(\xi_3)^2}{(\xi_3)}
\end{bmatrix}
\]

(47)

By substituting the result obtained in accordance with (26) for (47), the covariance matrix is given by

\[
\begin{bmatrix}
\frac{2\xi_1}{(\xi_3)^2} & 0 & -2\frac{(\xi_3)^2}{(\xi_3)} \\
0 & \frac{2\xi_2}{(\xi_3)^2} & -2\frac{(\xi_3)^2}{(\xi_3)} \\
0 & 0 & -1\frac{(\xi_3)^2}{(\xi_3)}
\end{bmatrix}
\cdot (\Lambda^T C^{-1}_S \Lambda)^{-1}
\]

(48)

The assumptions made from (43) to (48) allow us to use the BLUE to estimate the new virtual unknown parameter vector of (43), $[\xi^2, \xi^2]^T$, as follows:

\[
\begin{bmatrix}
\hat{\xi}^2
\end{bmatrix} = \begin{bmatrix}
1 & 0
\end{bmatrix}^T \cdot \text{Cov}^{-1}
\begin{bmatrix}
\frac{2\xi_1}{(\xi_3)^2} & 0 & -2\frac{(\xi_3)^2}{(\xi_3)} \\
0 & \frac{2\xi_2}{(\xi_3)^2} & -2\frac{(\xi_3)^2}{(\xi_3)} \\
0 & 0 & -1\frac{(\xi_3)^2}{(\xi_3)}
\end{bmatrix}
\cdot \begin{bmatrix}
1 & 0
\end{bmatrix}^{-1}
\]
According to the data model described in (3), the MLE is found as a sub-problem of the maximum likelihood minimization problem into a convex algorithm we have chosen is the method introduced in (18), which first approximates the non-convex ML problem into a convex problem which can guarantee the localization objective function the global minimum. It then relaxes the convex problem to the SDP problem by dropping the rank constraint. However, it should be noted that their proposed method requires the true value of transmit power, \( P_t \), in advance.

According to the RSS measurement model defined in (2), the SDP [18] is expressed as

\[
\text{minimize} \sum_{l=1}^{L} \left( n_l \left( \frac{P_l}{P_t} \right) \left( \frac{\hat{x}}{\hat{y}} \right) + v_l \left( \frac{P_l}{P_t} \right) \left( \frac{\hat{x}}{\hat{y}} \right) \right) + v_l \left( \frac{P_l}{P_t} \right) \left( \frac{\hat{x}}{\hat{y}} \right) \right)
\]

subject to \( \hat{b}_l = \begin{bmatrix} x_l \\ y_l \\ -1 \end{bmatrix}, \) \( \begin{bmatrix} n_l & 1 \\ 1 & n_l \end{bmatrix} \geq 0_{2 \times 2}, \) \( Z_{1,2,1,2} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \)

\( Z \geq 0_{3 \times 3}, \)

and finally, \( Q_{(L-1)\times(L-1)} \) is the covariance matrix of the correlated shadowing noise for the DRSS data model where its element at the \( i \)th row and \( j \)th column can be readily shown to be as

\[
Q_{i,j} = \begin{cases} 2\sigma^2, & \text{if } i = j \\ \sigma^2, & \text{otherwise.} \end{cases}
\]

The problem defined in (52) cannot be solved analytically and a numerical method should be applied instead. Therefore, we use \texttt{lsqnonlin}, which is a MATLAB routine with all its default settings. However, any numerical solution to the MLE is notoriously way too sensitive about the initial point. In our simulations, the \texttt{lsqnonlin} routine is always initialized with the true coordinates of the source in order to reduce the effects of this drawback as much as possible. Although the true coordinates are never available to an estimator in advance.

We also incorporate the localization results of a convex relaxation technique, namely, SDP, into the simulation results. The convex algorithm we have chosen is the method introduced in [18] which first approximates the non-convex ML problem into a convex problem which can guarantee the localization objective function the global minimum. It then relaxes the convex problem to the SDP problem by dropping the rank constraint. However, it should be noted that their proposed method requires the true value of transmit power, \( P_t \), in advance.

according to the RSS measurement model defined in (2), the SDP [18] is expressed as

\[
\min Z_{x,y} \sum_{l=1}^{L} \left( n_l \left( \frac{P_l}{P_t} \right) \left( \frac{x_l}{y_l} \right) + v_l \left( \frac{P_l}{P_t} \right) \left( \frac{x_l}{y_l} \right) \right) + v_l \left( \frac{P_l}{P_t} \right) \left( \frac{x_l}{y_l} \right)
\]

subject to \( \hat{b}_l = \begin{bmatrix} x_l \\ y_l \\ -1 \end{bmatrix}, \) \( \begin{bmatrix} n_l & 1 \\ 1 & n_l \end{bmatrix} \geq 0_{2 \times 2}, \) \( Z_{1,2,1,2} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \)

\( Z \geq 0_{3 \times 3}, \)

and finally, \( Q_{(L-1)\times(L-1)} \) is the covariance matrix of the correlated shadowing noise for the DRSS data model where its element at the \( i \)th row and \( j \)th column can be readily shown to be as

\[
Q_{i,j} = \begin{cases} 2\sigma^2, & \text{if } i = j \\ \sigma^2, & \text{otherwise.} \end{cases}
\]

The problem defined in (52) cannot be solved analytically and a numerical method should be applied instead. Therefore, we use \texttt{lsqnonlin}, which is a MATLAB routine with all its default settings. However, any numerical solution to the MLE is notoriously way too sensitive about the initial point. In our simulations, the \texttt{lsqnonlin} routine is always initialized with the true coordinates of the source in order to reduce the effects of this drawback as much as possible. Although the true coordinates are never available to an estimator in advance.

We also incorporate the localization results of a convex relaxation technique, namely, SDP, into the simulation results. The convex algorithm we have chosen is the method introduced in [18] which first approximates the non-convex ML problem into a convex problem which can guarantee the localization objective function the global minimum. It then relaxes the convex problem to the SDP problem by dropping the rank constraint. However, it should be noted that their proposed method requires the true value of transmit power, \( P_t \), in advance.

According to the RSS measurement model defined in (2), the SDP [18] is expressed as

\[
\min Z_{x,y} \sum_{l=1}^{L} \left( n_l \left( \frac{P_l}{P_t} \right) \left( \frac{x_l}{y_l} \right) + v_l \left( \frac{P_l}{P_t} \right) \left( \frac{x_l}{y_l} \right) \right) + v_l \left( \frac{P_l}{P_t} \right) \left( \frac{x_l}{y_l} \right)
\]

subject to \( \hat{b}_l = \begin{bmatrix} x_l \\ y_l \\ -1 \end{bmatrix}, \) \( \begin{bmatrix} n_l & 1 \\ 1 & n_l \end{bmatrix} \geq 0_{2 \times 2}, \) \( Z_{1,2,1,2} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \)

\( Z \geq 0_{3 \times 3}, \)

and finally, \( Q_{(L-1)\times(L-1)} \) is the covariance matrix of the correlated shadowing noise for the DRSS data model where its element at the \( i \)th row and \( j \)th column can be readily shown to be as

\[
Q_{i,j} = \begin{cases} 2\sigma^2, & \text{if } i = j \\ \sigma^2, & \text{otherwise.} \end{cases}
\]
should be located at origin as described before. The sensors displacement is shown in Figure 2 where sensors are depicted by solid circles.

These simulations are run by using MATLAB R2014a on PC, Win 10 Pro, X 64-based processor, RAM 8 GB with i5-4460 CPU@ 3.20 GHz. The number of Monte Carlo runs is 10,000 for all simulations. To generate the RSS measurements, the propagation model (2) is used. To generate DRSS measurements $R_d(l, l_i)$, we first form RSS $P_l$ by using the signal parameter model described in Section II and then obtain DRSS measurements from (3).

It is assumed that the unknown node is able to communicate with all anchors. The PLE $\alpha$ is set to 3 and the reference distance is $d_0 = 1$ m, unless otherwise stated. The performance metric is the (RMSE), which is calculated as follows:

$$\text{RMSE} = \sqrt{\frac{1}{10,000} \sum_{i=1}^{10,000} (\hat{x}_i - x)^2 + (\hat{y}_i - y)^2},$$ (60)

where $\hat{x}_i$ and $\hat{y}_i$ are the estimates of $x$ and $y$ obtained for the $i$th Monte Carlo run, respectively.

For the first scenario, an unknown node is supposed to be located close to the reference node which has to be located at (0,0). The unknown node is located at (2,8). Figure 3 shows RMSEs of the proposed methods and their counterparts. Any localization result for an algorithm which ended up having the RMSE value more than 50 m was considered failure. Therefore, its RMSE is not plotted in Figure 3.

Figures 4 and 5 depict enlarged views of Figure 3 for the range of small and high noise std values, respectively. As Figures 3–5 show, the proposed algorithms IR-TSBL1 and IR-TSBL2 outperform other algorithms. The IR-TSBL1 and IR-TSBL2 algorithms perform almost similarly for low noise levels. However, the localization obtained by using the IR-TSBL-2 maintains more stability for environments with noise variance more than 3. This is mainly because the IR-TSBL-2 eliminates the need to convert information into an appearance which is suitable to be directly expressed in the unknown node coordinates. Consequently, further information would be avoided to be lost in the IR-TSBL-2 method.

The two non-linear iterative estimators, MLE and SDP [18], present the best performance excluding the IR-TSBL-2 and IR-TSBL-1 methods. The SDP [18] method performs better for higher shadowing noise levels in particular. However, it should be noted that any convex relaxation technique demand excessive computational burdens.

Table 1 gives a clear demonstration of required computational complexity in terms of run-time on PC, Win 10 Pro, X 64-based processor, RAM 8 GB with i5-4460 CPU@ 3.20 GHz and by using MATLAB R2014a. Although the SDP [18] method can achieve almost similar performance to the IR-TSBL-2 and IR-TSBL-1 methods for high levels of shadowing noise, its computation run-time is approximately 4600 times larger than the computation run-time of the IR-TSBL-2 and IR-TSBL-1 methods.
For the next simulation, we chose an unknown target farther away from the reference node. It is located at (30,30), while the other simulation parameters are not changed.

The performance of different algorithm for the new target position has been shown in Figure 6. The overall result confirms smooth curves for all simulated algorithms. All algorithms perform more stable, compared to the simulation scenario of Figure 3, particularly against a range of large noise levels.

It is interesting to note that based on the simulation results shown in Figure 6, it can be said that in order to achieve less localization error, it is better to select a node as the reference node from the set of anchor nodes that is farthest from an unknown target node. Of course, this is impossible because the location of an unknown target node is not known in advance. By the way, the reason behind the smooth curves in Figure 6 may be explained by the approximation we made in (5). This approximation sounds more solid when $d_1$ is larger. As it is clear in Figures 3 and 5, the smallness of $d_1$ only becomes troublesome when the variance of noise is too large.

Due to these smooth behaviours of different algorithms, there is no need to provide enlarged views of Figure 6 anymore.
Another thing which can be inferred from Figure 6 is the huge gap exists between performance of our two proposed methods compared to their counterparts. This confirm the superiority of appropriate modelling when it comes to use LS estimation methods.

To investigate how the performance of different algorithms is affected by various PLE values, the RMSE values of the considered algorithms are shown in Table 2, for the range of PLEs, $\hat{\alpha} \in [1, 5]$. For all the results presented in Table 2, the target node is located at (30,30), and the measurement noise variance is $\sigma^2 = 5$ dB.

As it can be inferred from Table 2, the lowest RMSEs belong to the two proposed algorithms IR-TSBL1 and IR-TSBL2.

The noticeable feature of Table 2 is that the RMSE is decreasing with the increase in the PLE values. In particular, this applies to all the considered algorithms. For example, the RMSE of IR-TSBL2 algorithm drops from 4.54 for $\hat{\alpha} = 1$ to 0.94 for $\hat{\alpha} = 5$, showing a reduction of more than 11 dB. One reason for this could be that an increment in the PLE value relieves the adverse effect of the measurement noise in $\beta_2$. To see why this is, consider the term $(\sigma^2/(50\alpha^2))$ in (7). As the squared value of PLE, $\alpha^2$, increases the total value of $(\sigma^2/(50\alpha^2))$ decrease for the fixed measurement noise variance, $\sigma^2$.

In order to observe the effect of different unknown target positions on the performance of different algorithms we have adopted a different method in the next simulation scenario. In the new simulation, the true position of the unknown target is changing from one Monte Carlo run to another. In other words, the RMSE obtained for a given noise variance, is computed by
averaging 10,000 RMSE values while each value is computed assuming the target is located at different Cartesian coordinates inside the square region of $[0, 50 \text{ m}] \times [0, 50 \text{ m}]$. The simulation parameters are: $d_0 = 1 \text{ m}$, $\alpha = 3$, and $\sigma^2 = 5 \text{ dB}$.

Simulation results of this scenario are shown in Figure 7. It is normal for performance curves to exhibit more volatile behaviour. Because of the shifting of the target location, the recorded result of the performance related to that position has been averaged from a much smaller number of samples than 10,000 different performances, so the variance shows a higher estimate. By the way, also in this case, we can confirm from Figure 7 that the new algorithms IR-TSBL1 and IR-TSBL2 outperform the other methods with a good margin.

In order to better examine the localization accuracy of the proposed algorithms with other top-notch methods such as A-BLUE [16] and TS-WLS [20], the 3D RMSE results and their colourful 2D contour maps are shown from Figure 5(a) to Figure 11(b). For each algorithm, the RMSE is computed at each of the 2D Cartesian coordinates grid points located all over the square region of $[0,50 \text{ m}] \times [0,50 \text{ m}]$. Grid points are separated by 2 m in each dimension. For each grid point, the resulted RMSE is averaged over 10,000 Monte Carlo runs, thus providing a more realistic view of localization performance per each candidate target position than the results provided in Figure 7. The simulation parameters are: $d_0 = 1 \text{ m}$, $\alpha = 3$, and $\sigma^2 = 5 \text{ dB}$.

Figure 9 which shows the performance of IR-TSBL2 exhibits more robust localization results over all grid points with least RMSE values. Its colourful 2D contour map exhibits broader regional coverage of the low contours in the whole area. This may put IR-TSBL2 forward as a strong candidate for target node localization using RSS measurements with unknown transmit power.
5 | CONCLUSION

This paper first applied a one-to-one non-linear transformation to the position coordinates of an unknown target so that the new data model would be best fitted to LS problems. We have then introduced two new RSS-based positioning algorithms, IR-TSBL-1 and IR-TSBL-2, for unknown source transmit power. The first algorithm, IR-TSBL-1, presents the Taylor series expansion for quadratic forms of the entries of estimation results obtained from a primary coarse position. The second estimator, IR-TSBL-2, uses the Taylor series for variables which have been already directly proportional to the position variables. The performance evaluation of the proposed algorithms demonstrate their superiority compared to the other existing linear position estimators for a large range of noise variance and PLE values. However, the localization obtained by using the IR-TSBL-2 is more stable compared to IR-TSBL-1 for environments with large noise levels. This is due to the fact that the IR-TSBL-2 algorithm avoids losing information by reducing the number of required non-linear transformations.

REFERENCES

1. Qiu, T. et al.: Robustness optimization scheme with multi-population co-evolution for scale-free wireless sensor networks. IEEE/ACM Trans. Networking 27(5), 1028–1042 (2019).
2. Khalaf, O., Sabah, A.: An overview on wireless sensor networks and finding optimal location of nodes. Period. Eng. Nat. Sci. 7(3), 1096–1101 (2019).
3. Kulkarni, V., Desai, V., Kulkarni, R.: A comparative investigation of deterministic and metaheuristic algorithms for node localization in wireless sensor networks. Wireless Netw. 25(5), 2789–2803 (2019).
4. Promwong, S., Thongkam, J.: Evaluation of weighted impulse radio for ultra-wideband localization. Wireless Pers. Commun. 115(4), 1–10 (2020).
5. Pérez, et al.: Probabilistic time of arrival localization. IEEE Signal Process. Lett. 26(11), 1683–1687 (2019).
6. Rezat, M., Ricciato, E.: On the impact of time-of-departure knowledge on the accuracy of time-of-arrival localization. Comput. Networks 176, 107265 (2020).
7. Zhao, Y., et al.: Sensor selection for TDOA-based localization in wireless sensor networks with non-line-of-sight condition. IEEE Trans. Veh. Technol. 68(10), 9935–9950 (2019).
8. Vashishtha, A., Law, C.: F. DTDSSA based localization for wireless sensor networks with clock drift compensation. IEEE Sens. J. 20(5), 2648–2658 (2019).

How to cite this article: Danaee MR. One-to-one non-linear transformation for RSS-based localization with unknown transmit power. *IET Commun.* 2021;15:627–641. https://doi.org/10.1049/cmu2.12094

APPENDIX A: DERIVATION OF NEW DIFFERENTIAL RECEIVED SIGNAL STRENGTH (DRSS)-BASED MODEL

The parameter model in (13) is derived in this Appendix. According to the model defined in (9), the reduced model
associated with taking \((x_1,y_1) = (0,0)\) is given by
\[
\begin{pmatrix}
-2\kappa_2\beta_2^{-1} & -2\kappa_2\beta_2^{-1} & \beta_2^{-1} - 1 \\
-2\kappa_3\beta_3^{-1} & -2\kappa_3\beta_3^{-1} & \beta_3^{-1} - 1 \\
\vdots & \vdots & \vdots \\
-2\kappa_L\beta_L^{-1} & -2\kappa_L\beta_L^{-1} & \beta_L^{-1} - 1 \\
\end{pmatrix} \begin{bmatrix} \kappa_x \kappa_y \end{bmatrix} = \begin{bmatrix} -(\kappa_2^2 + \kappa_3^2)\beta_2^{-1} \\
-(\kappa_2^2 + \kappa_3^2)\beta_3^{-1} \\
\vdots \\
-(\kappa_2^2 + \kappa_3^2)\beta_L^{-1} \end{bmatrix}.
\]
(A.1)

The \(i\)th equation, after multiplying the \(i\)th row of (A.1) with \(\beta_{i+1,i}\), is given by
\[-2\kappa_{i+1}\kappa_x - 2\kappa_{i+1} + (1 - \beta_{i+1,i})R = -(\kappa_{i+1}^2 + \kappa_{i+1}^2). \] (A.2)

If Equation (A.2) is divided by \(R\), it will be seen that after some trivial algebra we obtain
\[-2\kappa_{i+1}\kappa_x/R + (-2\kappa_{i+1})/R + (\kappa_{i+1}^2 + \kappa_{i+1}^2)/R = (\beta_{i+1,i} - 1). \]
(A.3)

The matrix form generated by ranging the variable \(i\) from 1 to \(L - 1\) (\(L\) as the number of receiver nodes) is given by (13).

**APPENDIX B: DERIVATION OF COVARIANCE MATRIX OF OBSERVATION VECTOR**

The covariance matrix of the observation vector \(\xi\) in (13) is derived in this Appendix. Recall that for the observation vector \(\xi\), the covariance matrix \(\text{Cov}(\xi)\) is computed as
\[
\text{Cov}(\xi) = \mathbb{E}\left\{ (\xi - \mathbb{E}[\xi]) \cdot (\xi - \mathbb{E}[\xi])^T \right\}. \] (B.1)

According to (16), if \(\text{Cov}(\xi)\) is denoted by \(\text{C}_\xi\), then it is possible to write,
\[
\text{C}_\xi = \text{Cov}\left( [\beta_{2,1}, \beta_{3,1}, \ldots, \beta_{L,1}]^T \right). \] (B.2)

Based on (3) and (7), we may write
\[
\beta_{i,i} = A\beta_{i+1,i}, \] (B.3)
where \(A, B_i\), and \(m\) may be defined as
\[
A = e^{-\frac{\kappa_x^2}{2\sigma^2}}, B_i = e^{\kappa_x(1)}, m = -\frac{1}{2\sigma^2}. \] (B.4)

The diagonal elements of \(\text{C}_\xi\), \(\text{C}_\xi(i,i)\), are given by
\[
\text{C}_\xi(i,i) = \text{var}(\beta_{i+1,i}) = \mathbb{E}\left\{ (\beta_{i+1,i} - \mathbb{E}[\beta_{i+1,i}]) \right\}, i \in \{1, \ldots, L - 1\}. \] (B.5)

In order to compute \(\text{C}_\xi(i,i)\), we first have to compute \(\mathbb{E}[\beta_{i+1,i}]\),
\[
\mathbb{E}[\beta_{i,i}] = A\mathbb{E}[\beta_{i+1,i}] \] (B.6)

Since
\[
R_d(l, 1) = \ln \left( \frac{d}{d_1} \right)^{\frac{1}{2}} + w_y - w_1, \] (B.7)
then \(R_d(l, 1)\) is a Gaussian distributed random variable where
\[
R_d(l, 1) \sim \mathcal{N} \left( \ln \left( \frac{d}{d_1} \right)^{\frac{1}{2}}, 2\sigma^2 \right). \] (B.8)

As a result, for any \(l\), the below expectation can be calculated based on the Gaussian distribution moment-generating function
\[
\mathbb{E} \left( e^{R_d(l, 1)} \right) = \left( \frac{d}{d_1} \right)^{\frac{1}{2}} e^{\frac{1}{2} \sigma^2}. \] (B.9)

According to (B.9),
\[
\mathbb{E}[\beta_{i,i}] = \left( \frac{d}{d_1} \right)^{\frac{1}{2}} \sigma^2/30\alpha^2, \] (B.10)
where we also define \(D_l = \mathbb{E}[\beta_{i,i}]\). Thus the diagonal entries \((i,i)\)th of \(\text{C}_\xi\) are given by
\[
\text{C}_\xi(i,i) = \mathbb{E}\left\{ \beta_{i+1,i} \right\} - \mathbb{E}\left\{ \beta_{i+1,i} \right\} = A^2\mathbb{E}\left\{ \beta_{i+1,i} \right\} - D_{i+1}\]
(B.11)

The off-diagonal entries \((i,j)\)th \((i \neq j)\) of \(\text{C}_\xi\) are given in the same way by
\[
\text{C}_\xi(i,j) = A^2\mathbb{E}\left\{ \beta_{i+1,i} \right\} - \mathbb{E}\left\{ \beta_{i+1,i} \right\} - D_{i+1}D_{j+1} \]
(B.12)