Future Network and MobileSummit 2010 Conference Proceedings
Paul Cunningham and Miriam Cunningham (Eds)
IIMC International Information Management Corporation, 2010
ISBN: 978-1-905824-18-2

Static Positioning Using UWB Range Measurements

Mohammad Reza Gholami†, Erik G. Ström†, Francesco Sottile††, Davide Dardari§, Andrea Conti§, Sinan Gezici‡, Mats Rydström†, and Maurizio A. Spirito††
†Chalmers University of Tech. Dept. Signals & Systems Gothenburg, Sweden.
§University of Bologna at Cesena Via Venezia, 52 I-47023 Cesena (FC), Italy.
††Istituto Superiore Mario Boella Via P. C. Boggio 61, 10138 Turin, Italy.
‡Bilkent University Dept. of Electrical & Electronics Engineering, Ankara, Turkey.
{moreza, erik.strom, mats.rydstrom}@chalmers.se, ddardari@ieee.org, andrea.conti@gmail.com, francesco.sottile,maurizio.spirito@ismb.it, gezici@ee.bilkent.edu.tr

Abstract: The performance of several existing and partly new algorithms for positioning of sensor node based on distance estimate is compared when the distance estimates are obtained from a measurement campaign. The distance estimates are based on time-of-arrival measurements done by ultrawideband devices in an indoor office environment. Two different positioning techniques are compared: statistical and geometrical. In statistical category, distributed weighted-multidimensional scaling (dwMDS), least squares, and sum product algorithm are evaluated and in geometrical technique projections approach and outer approximation (OA) method are investigated. No method shows the best performance in all cases, while in many situations, sum product algorithm, dwMDS, nonlinear least square, projection approach, OA, and weighted least square work well.

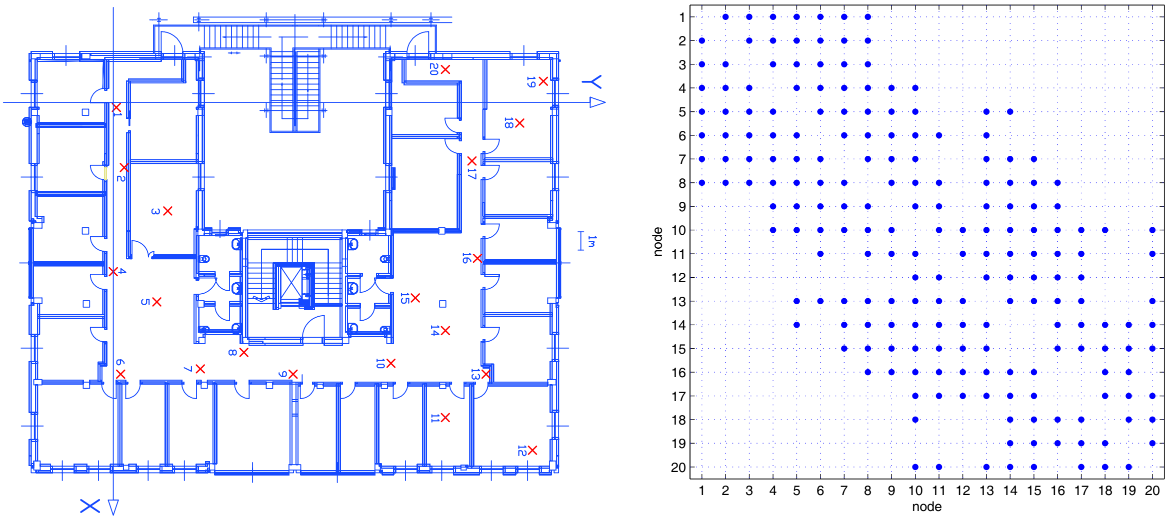
1. Introduction

Position information of the nodes that make up a wireless sensor network is required in most, if not all, applications. Preferably, the positioning should be carried out by the network itself to avoid a cumbersome manual node deployment.

We will here consider the problem of positioning one node using range (distance) estimates to a number of nodes at known positions (so-called anchor nodes or reference nodes). In general, the range estimates can be based on different types of measurements, e.g., received signal strength (RSS) or, as the case in this paper, time of arrival (TOA). The accuracy of the positioning depends on the quality of the range measurements, the geometry of the network, and the performance of the positioning algorithm. In particular, it is important that any assumptions on the range estimates posed by the positioning algorithms are satisfied to a reasonable degree. For example, a maximum likelihood approach requires knowledge of the joint probability density function (PDF) of the range estimates. In complex environments, e.g., indoor scenarios, the PDF might not be readily available, and we have to settle for other methods, such as least squares methods.

The well-known nonlinear least squares (NLS) method will therefore be used to benchmark a number of more novel algorithms that offer either lower computational

This work was supported in part by the European Commission in the framework of the FP7 Network of Excellence in Wireless COMmunications NEWCOM++ (contract no. 216715) and in part by the Swedish Research Council (contract no. 2007-6363)



(a) (b)

Figure 1: (a) Location of UWB static nodes (b) Connectivity matrix: the markers show which nodes were connected during the measurements.

complexity, robustness against positive bias in the distance estimates (which tends to occur in non line-of-sight situations), or better performance compared to standard NLS. Details about the algorithms are found in section 3.

The range estimates used to evaluate the algorithms come from a recent ultrawideband (UWB) measurement campaign planned and carried out under the auspices of NEWCOM++, an EU FP7 Network of Excellence [1]. UWB technology has the potential to deliver very accurate range estimates and thereby enabling accurate positioning. However, it is not clear how to best use the range estimates. We tackled the problem and presented some primary results in [2]. In this paper, we consider more algorithms for comparison after a simple pre-processing on data to remove obviously bad measurements, i.e., zeros and negative ones.

2. Range Measurements

The measurement campaign was performed on the second floor of the Department of Electronics, Information and Systems at the Cesena campus of the University of Bologna, Italy. Sensor node positions, numbered from 1 to 20, are indicated in the floor plan in Fig. 1a. See [2] for more details. The range estimate $\hat{d}_{i,j}$ between node i and j is simply modeled as

$$\hat{d}_{i,j} = d_j(\mathbf{x}_i) + w_{i,j}, \qquad i, j \in \{1, \ldots, 20\} \tag{1}$$

where $w_{i,j}$ is the ranging error, $d_j(\mathbf{x})$ is the distance from node j to $\mathbf{x}$, i.e., $d_j(\mathbf{x}) = \|\mathbf{x} - \mathbf{x}_j\|$, where $\|\cdot\|$ is the Euclidean norm and $\mathbf{x}_j = \begin{bmatrix} x_{j,1} & x_{j,2} \end{bmatrix}^T$ is the coordinates of node j, see Fig. 1a. In this paper, we consider the ranging error $w_{i,j}$ as a random variable with unknown distribution. In every position, an unknown node can communicate to a few anchor nodes based on connectivity matrix shown in Fig. 1b. In the sequel, the index i is (normally) used for the target node (i.e., a node whose position is to be estimated). We define $\mathcal{C}_i$ as the set of nodes that are connected to the ith node (which implies that $\hat{d}_{i,j}$ exists if and only if $j \in \mathcal{C}_i$). Let N_i be the number of nodes connected to the ith nodes.

3. Positioning Algorithms

In this section, some existing and partly novel methods are briefly reviewed. Since the PDF of measurement noise in (1) is not available, we consider two categories of suboptimal estimators: statistical and geometrical.

3.1 Statistical estimators

3.1.1 Nonlinear least squares

The nonlinear least squares (NLS) position estimate based on the ranging measurement (1) can be found as the solution to the non-convex optimization problem

$$\hat{\mathbf{x}}_i = \arg\min_{\mathbf{x}} \sum_{j \in \mathcal{C}_i} \|\hat{d}_{i,j} - d_j(\mathbf{x})\|^2. \tag{2}$$

We note, that if $w_{i,j}$ are identically distributed, zero-mean Gaussian random variables for all $j \in \mathcal{C}_i$, the NLS estimate is also the maximum likelihood estimate [3]. In this paper we will approximate the NLS estimate using the MATLAB routine `lsqnonlin` [4] randomly initialized in the deployment area (see Fig. 1a).

3.1.2 Linear least squares

To form a linear least squares problems, we need to find a signal model that is linear in unknown parameters [5]. One approach (call it LS-1) is to consider pairs of distance estimates as follows. We can form $M_i = N_i(N_i-1)/2$ distinct 2-element subsets (pairs) of $\mathcal{C}_i$. For the mth pair, $\{j,k\}$, we have that

$$b_m(\mathbf{x}_i) = [d_k^2(\mathbf{x}_i) - \|\mathbf{x}_k\|^2] - [d_j^2(\mathbf{x}_i) - \|\mathbf{x}_j\|^2] = 2(\mathbf{x}_j - \mathbf{x}_k)^T\mathbf{x} = \mathbf{a}_m^T\mathbf{x}, \tag{3}$$

which is seen to be a linear function of $\mathbf{x}_i$. We couple this signal model with the measurements

$$\hat{b}_m = (\hat{d}_{i,k}^2 - \|\mathbf{x}_k\|^2) - (\hat{d}_{i,j}^2 - \|\mathbf{x}_j\|^2). \tag{4}$$

An estimate of $\mathbf{x}_i$ can now be obtained by fitting the signal model (3) to the measurements (4). To this end, we form

$$\mathbf{b}_i(\mathbf{x}_i) = \begin{bmatrix} b_1(\mathbf{x}_i) & b_2(\mathbf{x}_i) & \cdots & b_{M_i}(\mathbf{x}_i)\end{bmatrix}^T = \begin{bmatrix}\mathbf{a}_1 & \mathbf{a}_2 & \cdots & \mathbf{a}_{M_i}\end{bmatrix}^T \mathbf{x}_i = \mathbf{A}_i\mathbf{x}_i. \tag{5}$$

Now the solution to the (5) can be obtained using the least squares criteria, i.e.,

$$\hat{\mathbf{x}}_i = \arg\min_{\mathbf{x}} \|\hat{\mathbf{b}}_i - \mathbf{b}_i(\mathbf{x}_i)\| = \mathbf{A}_i^\dagger \hat{\mathbf{b}}_i \ , \tag{6}$$

where $\hat{\mathbf{b}}_i = \begin{bmatrix}\hat{b}_1 & \hat{b}_2 & \cdots & \hat{b}_{M_i}\end{bmatrix}^T$ and $\mathbf{A}_i^\dagger$ is the left-hand pseudoinverse of $\mathbf{A}_i$. Assuming that $\mathbf{A}_i$ has full column rank, $\mathbf{A}_i^\dagger = (\mathbf{A}_i^T\mathbf{A}_i)^{-1}\mathbf{A}_i^T$.

In another LS technique (call it LS-2), instead of subtracting pairs of squared distances directly in (3), first the average of all distances is computed and then it is subtracted from all equations [6]. Therefore LS-2 can be formulated as $\hat{\mathbf{x}}_i = \tilde{\mathbf{A}}_i^\dagger \tilde{\mathbf{b}}_i$, where $\tilde{\mathbf{A}}_i^\dagger = (\tilde{\mathbf{A}}_i^T \tilde{\mathbf{A}}_i)^{-1} \tilde{\mathbf{A}}_i^T$ and the elements of vector $\tilde{\mathbf{b}}_i = [\tilde{b}_1, ..., \tilde{b}_{N_i}]^T$ and matrix $\tilde{\mathbf{A}}_i = \begin{bmatrix} \tilde{\mathbf{a}}_1 & \tilde{\mathbf{a}}_2 & \cdots & \tilde{\mathbf{a}}_{N_i} \end{bmatrix}^T$ are defined as follows

$$\tilde{b}_j = (\hat{d}_{i,j}^2 - \|\mathbf{x}_j\|^2) - \frac{1}{N} \sum_{l=1}^{N_i} (\hat{d}_{i,l}^2 - \|\mathbf{x}_l\|^2) \ , \tilde{\mathbf{a}}_j = 2\Big(\mathbf{x}_j - \frac{1}{N} \sum_{l=1}^{N_i} \mathbf{x}_l\Big)^T .$$

Both LS-1 and LS-2 use quadratic eliminations by subtracting squared of measurements from each other which make new sets of measurements be dependent. There is also another method of LS, which is called one step LS, that considers each measurement individually and constitutes new linear equations as follows

$$c_j(\mathbf{x}_i) = d_j^2(\mathbf{x}_i) - \|\mathbf{x}_j\|^2 = 2 \left[-\mathbf{x}_j^T \ 1\right] \left[\mathbf{x}_i \ \|\mathbf{x}_i\|^2\right]^T = \mathbf{g}_j^T \left[\mathbf{x}_i \ \|\mathbf{x}_i\|^2\right]^T , \tag{7}$$

where $1 \le j \le N_i$. A set of equations can be written in matrix form as $\mathbf{G}_i \boldsymbol{\theta}_i = \mathbf{c}_i$, where $\mathbf{G}_i = \begin{bmatrix} \tilde{\mathbf{g}}_1 & \tilde{\mathbf{g}}_2 & \cdots & \tilde{\mathbf{g}}_{N_i} \end{bmatrix}^T$, $\boldsymbol{\theta}_i = [\mathbf{x}_i \ \|\mathbf{x}_i\|^2]^T$, $\mathbf{c}_i = [\hat{c}_1, \cdots, \hat{c}_{N_i}]$, and $\tilde{\mathbf{g}}_j = 2 \left[-\mathbf{x}_j^T \ 1\right]$. Considering small noise in measurement (1), we can write

$$\hat{c}_j = \hat{d}_{i,j}^2 - \|\mathbf{x}_j\|^2 \simeq \|\mathbf{x}_j\|^2 - 2\mathbf{x}_j^T \mathbf{x}_i + \|\mathbf{x}_i\|^2 - 2d_j(\mathbf{x}_i) w_{i,j}. \tag{8}$$

Let $w_{i,j}$ be i.i.d. Considering the weighting matrix $\mathbf{W}_i = \frac{1}{4} \operatorname{diag}\{d_1^{-2}(\mathbf{x}_i) \ldots, d_{N_i}^{-2}(\mathbf{x}_i)\}$, the weighted LS (WLS) solution can be obtained as follows

$$\boldsymbol{\theta}_i = (\mathbf{G}_i^T \mathbf{W}_i \mathbf{G}_i)^{-1} \mathbf{G}_i^T \mathbf{W}_i \mathbf{c}_i. \tag{9}$$

Since in practice there is no access to the real distance $d_j^2(\mathbf{x}_i)$, the estimated one, i.e., $\hat{d}_{i,j}^2$, is used to compute the weighting matrix.

3.1.3 Distributed weighted-multidimensional scaling (dwMDS)

The dwMDS algorithm estimates unknown nodes coordinates by minimizing the following global cost function [7]

$$S = 2 \sum_{i=1}^{n} \sum_{j=1}^{n+m} \alpha_{i,j} \Big(\hat{d}_{i,j} - d_j\left(\mathbf{x}_i\right)\Big)^2 + \sum_{i=1}^{n} r_i \left\|\mathbf{x}_i - \overline{\mathbf{x}}_i\right\| , \tag{10}$$

where n is the number of nodes with unknown coordinates, m is the number of anchor nodes, $\alpha_{i,j}$ is the weight associated to the range measurement $\hat{d}_{i,j}$, $\overline{\mathbf{x}}_i$ is the a priori coordinates of unknown node i, and r_i is the a priori variance associated to $\overline{\mathbf{x}}_i$. In equation (10), unknown nodes and anchor nodes are indicated with indexes in the range $[1, n]$ and $[n+1, n+m]$, respectively. In this paper, since only a single node is estimated per time, n is one, and the set of anchor nodes indicated in (8) with index [n+1,m+n] coincides with the set $\mathcal{C}_i$. To be noted that the cost function (10) differs from the standard MDS objective function is that it adds a penalty term which accounts for prior knowledge about node locations. After simple manipulation, S can be rewritten as, $S = \sum_{i=1}^{n} S_i + c$, where the local cost functions S_i are associated for each unknown node (i.e. $1 \le i \le n$),

$$S_i = \sum_{j=1, j\neq i}^{n} \alpha_{i,j} \Big(\hat{d}_{i,j} - d_j\left(\mathbf{x}_i\right)\Big)^2 + \sum_{j=n+1}^{n+m} 2\alpha_{i,j} \Big(\hat{d}_{i,j} - d_j\left(\mathbf{x}_i\right)\Big)^2 + r_i \|\mathbf{x}_i - \overline{\mathbf{x}}_i\|^2, \tag{11}$$

and c is a constant independent of the nodes locations $\mathbf{x}_i$. The dwMDS algorithm uses a cooperative and distributed approach, in which each node iteratively updates its position estimate by minimizing the corresponding local cost function S_i, taking as input raging measurements and position estimates from its neighboring nodes. The local cost function is minimized by using quadratic majorizing function, which has the attractive property of generating a sequence of non-increasing cost function values. For more details please see [7].

3.1.4 Sum product algorithm

A general approach incorporating several Bayesian techniques is given by the sum-product algorithm over a wireless network (SPAWN) presented in [8]. It is a fully distributed and cooperative algorithm based on the principles of estimation theory and statistical inference within a framework based on the theory of factor graphs (FGs). A FG is mapped onto a time-varying network topology and spatio-temporal message schedule is employed resulting in a network FG and network message passing. SPAWN is considered here since it generalizes previously proposed localization algorithms. For a step by step description of SPAWN and for the performance evaluation over a large network by simulation, refer to [1]. The algorithm is composed essentially of two parts: prediction and correction. In the first one each node computes a message based on its internal metrics and its local mobility, while in the second it computes a message which considers also the measures obtained by communicating with other nodes. This message (belief) are then sent over the network broadcast. However, in a static scenario only the correction part is present and operates with belief initialized according to a uniform distribution inside the environment (which represents the possibility that at the start a node can be positioned anywhere).

The measures in the database before being processed have been filtered to eliminate those most affected by errors. The filtering method used is to select 100 consecutive measures, order them in terms of range, and consider the 20 central of this set. Then, the arithmetic mean on these 20 measures is utilized as ranging between the pair of nodes considered. The measures on ranging are affected by an error (bias) due to the presence of the walls between nodes [9]. To reduce the effect of bias, three deletion models were taken: Mean Bias (MB), WED, and WED with regression lines [10]. To apply these models, a priori statistical description of the environment has been considered, where for each walls configuration (LOS/NLOS with 1, 2, . . . , n walls) the error mean, standard deviation, and relative frequency has been calculated.

3.2 Geometric estimators

3.2.1 Projection onto convex sets

It is clear that the minimum of each term in the cost function in (2) is obtained when $\hat{d}_{i,j} = d_j(\mathbf{x}_i)$. Now, suppose we define the discs $\mathcal{D}_{i,j}$ as $\mathcal{D}_{i,j} = \{\mathbf{x} \in \mathbb{R}^2 : d_j(\mathbf{x}) \le \hat{d}_{i,j}\}, j \in \mathcal{C}_i$, it then is reasonable to define an estimate of $\mathbf{x}_i$ as a point in the intersection $\mathcal{D}_i$ of the discs $\mathcal{D}_{i,j}$,

$$\hat{\mathbf{x}}_i \in \mathcal{D}_i = \bigcap_{j \in \mathcal{C}_i} \mathcal{D}_{i,j} \,. \tag{12}$$

A method called projection onto convex sets (POCS) can be used to compute an estimate of the form (12) which was proposed for the positioning problem by Blatt and Hero in [11]. If the intersection is the empty set (which can occur due to measurement noise), the POCS estimate will be any point that minimizes the sum of the distance to the discs,

$$\hat{\mathbf{x}}_i = \arg\min_{\mathbf{x}} \sum_{j \in \mathcal{C}_i} \|\mathbf{x} - \mathcal{P}_{\mathcal{D}_{i,j}}(\mathbf{x})\|, \tag{13}$$

where $\mathcal{P}_{\mathcal{D}_{i,j}}(\mathbf{x})$ is the orthogonal projection of $\mathbf{x}$ onto $\mathcal{D}_{i,j}$.It has been shown that POCS has problems when the unknown node is outside the convex hull of the anchor nodes. On the other hand, POCS is quite robust against overestimated range estimates (which may occur in non-line-of-sight environments) as long as the unknown node is inside the convex hull of the anchor nodes.

3.2.2 Projection onto rings (POR)

In the case when the measurement noise in (1) is small, we can often improve POCS by replacing the disc $\mathcal{D}_{i,j}$ with a ring (or, more formally, an annulus) defined as

$$\mathcal{R}_{i,j} = \{\mathbf{x} \in \mathbb{R}^2 : \hat{d}_{i,j} - \epsilon_l \leq d_j(\mathbf{x}) \leq \hat{d}_{i,j} + \epsilon_u\}, \quad j \in \mathcal{C}_i, \tag{14}$$

where $\epsilon_l + \epsilon_u$ determines the width of the ring. The width is a tuning parameter of the resulting algorithm; it is reasonable to make the width dependent on the ranging error statistics in (1). Since we do not assume any knowledge of the error statistics, we simply choose $\epsilon_l = \epsilon_r = 0$ in the numerical results presented in Section 4. For details of POR method see [2].

3.2.3 Outer Approximation(OA)

As we saw before, the position of the unknown node can be found in the intersection of some discs. In this section, we consider a simple method to approximate the intersection. The method we consider here is not an optimal way to approximate the intersection, but it works well in most of the cases. In this method, we approximate the intersection of a number of discs by a disc. To implement the algorithm, we consider an iterative way. First for two nonempty discs, a new disc is found and then the non empty intersection of this new disc with another disc is computed in the same way. The procedure is continued until the whole intersection is covered by a disc.

Although each point inside of final disc normally can be considered as an estimate, it is clear that some points result in lower error of estimation. Here we try to select some points that are highly likely to be close to the unknown node position. To get a criteria, we consider the proximity order of an unknown node to the anchor nodes. Consider a grid of $P \times P$ points inside the final disc. Now we sort the distance measurements in anchor nodes, and we pick points inside the disc which their distances to the anchor nodes satisfy the same order of proximity. Finally the average of those points determines the position estimate.

4. Numerical Results

In this section, the performance of different algorithms described in section 3 are compared when applied to the practical measurement. Performance is measured in terms of root mean-squared error (RMSE) and cumulative distribution function (CDF) of the position error. Measured distances are first pre-processed to remove obviously bad measurements, such as zero and negative ones.

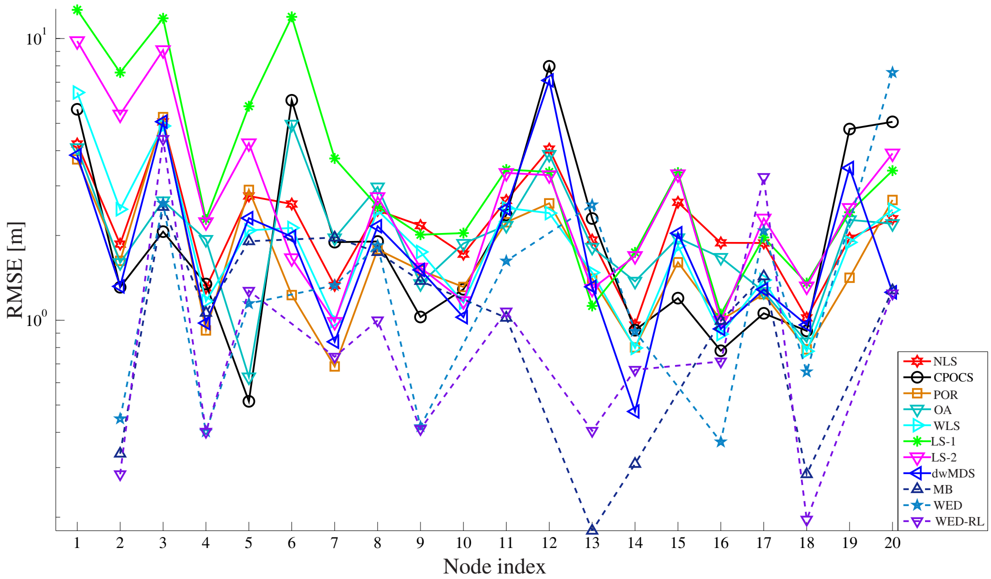


Figure 2: RMSE of different algorithms for different nodes

For the outer approximation, a grid of 9 points inside the final disc were considered. Concerning numerical results about the dwMDS, we set ranging weights $\alpha_{i,j} = 1$ for every pair of nodes, and a priori knowledge $r_i = 0$ for every unknown node and the corresponding initial estimation equal to the average coordinates of the neighboring anchor nodes. We would expect better performance, assuming for instance different $\alpha_{i,j}$ values for LOS and NLOS cases.

Fig. 2 shows RMSE for all algorithms for different positions of nodes. As it is seen, the RMSE fluctuates and no algorithm is uniformly best. For some positions some algorithms show relatively bad performance. For instance, both LS-1 and LS-2 for positions 1, 2, and 3 show poor performance compared to others. Among LS methods, WLS has good performance and outperforms LS-1 and LS-2 in most cases. The reason for bad performance perhaps is due to the nonlinear pre-processing needed by these algorithms. The NLS shows relatively acceptable performance, but not the best. The reason for that probably is convergence to local minima. The POCS and POR algorithms show different performance and it is seen that in overall POR method is slightly better than POCS. The POCS method is very effective to remove outliers while the POR is not able to localize well in that case. It is seen that POCS has problems when the unknown node is outside the convex hull of the anchor nodes, which is the case for nodes 1, 6, 12, 19, and 20. We also see that the OA method shows good performance. We can see the dwMDS also works well in most of the positions. Finally it is seen that the SPAWN methods work well for most of the cases and in general they show better performance compared to other methods.

To give more insight, we consider the position error CDFs of different methods except SPAWN methods. In the following, we will discuss the CDFs for nodes 1, 12, 14, and 20 to point out some interesting features.

Fig. 3a shows the error CDFs for node 1. We note that POR, NLS, dwMDS, and OA are approximately similar and different from other methods (it is also evident from

Fig. 2). The LS-1 has the worst performance, while the WLS improves the performance for most of the measurements. The unknown node is outside of the convex hull, and then the error in POCS method comes mostly from convex hull issue.

The CDFs for node 12 in Fig. 3b show very similar performance for the NLS approache, LS-1, LS-2, and WLS approaches; Here POR outperforms other methods in most of the case, roughly speaking 98%. The POCS method has frequent and large errors, explaining its large RMSE. This relative poor POCS performance is expected since node 12 lies outside the convex hull of its anchor nodes (nodes 10, 11, and 13–17), see Figs. 1a and 1b. The dwMDS in this case shows poor performance and the OA has acceptable performance compared to POCS and dwMDS.

For node 14, which is in a good position, we see from Fig. 3c that most of algorithms show good performance and the dwMDS has the best performance among all methods. In overall, for this position as it was expected the performance of algorithms are quite good. The reason for that is the line of sight situation is more satisfied in this position compared to the other 3 positions. The interesting observation relates to improvement due to weighting in least square approach.

Finally we consider CDFs for node 20 in Fig. 3d. Like node 12, it is located outside of convex hull, therefore we expect POCS shows bad performance. The performance of dwMDS is well as can be seen from RMSE plotted in Fig. 2. Again weighting improves the performance of LS. We also see that the POR and OA show good performance after dwMDS.

5. Conclusions

Several positioning algorithms have been compared in terms of root mean-square error (RMSE) and cumulative density function (CDF) of position error. The algorithms attempt to position a single node given distance estimates to a number of nodes at known positions (anchor nodes). The distance estimates were obtained from an indoor measurement campaign employing ultrawideband devices with built-in time-of-arrival ranging capabilities. A simple pre-processing on data is done which remove only zero and negative distance estimated by UWB device. From the numerical results, it can be concluded that weighing least square totaly outperforms ordinary least squares. Distributed weighted-Multidimensional scaling and projection onto ring approaches preform well in most cases. The projection onto convex set (POCS) show good performance when target is inside of the convex hull. However, POCS performs relatively bad for nodes that are outside the convex hull of the anchor nodes. The sum product algorithms outperform other methods in most of the cases. Position error CDFs can lead to other rankings of the algorithms.

References

[1] D. Dardari and F. Sottile, “WPR.B database: Annex of progress report II on advanced localization and positioning techniques: Data fusion and applications,” tech. rep., Deliverable DB.3 Annex, 216715 Newcom++ NoE, WPR.B, December 2009.

[2] M. R. Gholami, E. G. Ström, and M. Rydström, “Indoor sensor node positioning using uwb range measurents,” in *17th European Signal Processing Conference(Eusipco)*, pp. 1943–1947, 2009.

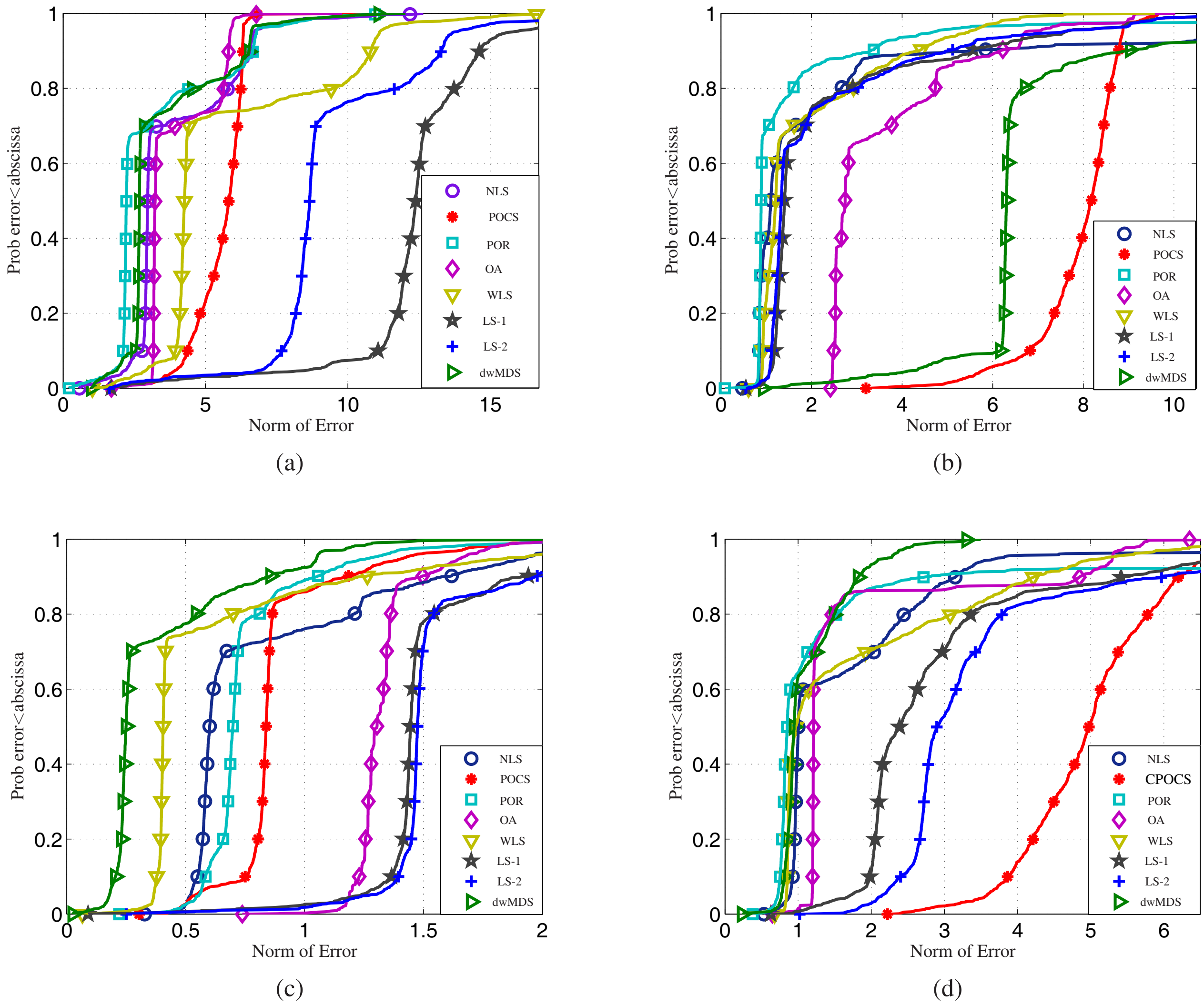


Figure 3: Numerical results; (a), (b), (c), and (d) show the position error CDFs for node 1, 12, 14, and 20, respectively.

[3] S. Kay, *Fundemental of Statistical Processing*. Prentice-Hall, 1993.

[4] The Mathworks Inc., 2010. `http://www.mathworks.com/access/helpdesk/help/toolbox/optim/ug/lsqnonlin.html`.

[5] C. Meesookho, U. Mitra, and S. Narayanan, "On energy-based acoustic source localization for sensor networks," *IEEE Trans. on Sig. Proc.*, vol. 56, pp. 365–377, Jan. 2008.

[6] S. Gezici, I. Guvenc, and Z. Sahinoglu, "On the performance of linear least-squares estimation in wireless positioning systems," in *IEEE International Conference on Communications*, pp. 4203–4208, May 2008.

[7] J. A. Costa, N. Patwari, and A. O. Hero, III, "Distributed weighted-multidimensional scaling for node localization in sensor networks," *ACM Trans. Sen. Netw.*, vol. 2, no. 1, pp. 39–64, 2006.

[8] H. Wymeersch, J. Lien, and M. Win, "Cooperative localization in wireless networks," *Proceedings of the IEEE*, vol. 97, pp. 427–450, February 2009.

[9] D. Dardari, A. Conti, U. J. Ferner, A. Giorgetti, and M. Z. Win, "Rang- ing with ultrawide bandwidth signals in multipath environments," *Proceedings of the IEEE*, vol. 97, pp. 427–450, February 2009.

[10] D. Dardari, A. Conti, J. Lien, A. Giorgetti, and M. Z. Win, "The effect of cooperation on localization systems using uwb experimental data," *EURASIP J. Appl. Signal Process*, pp. 1–11, 2008.

[11] A. O. Hero and D. Blatt, "Sensor network source localization via projection onto convex sets (POCS)," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, vol. 3, pp. 689–692, Mar. 2005.