

Fast and robust anchor calibration in range-based wireless localization

Samuel Van de Velde*, Patrick Van Torre[†], and Heidi Steendam*

*Department of Telecommunications and Information Processing,
Ghent University, Belgium, e-mail: {slvdveld, hs}@telin.ugent.be

[†]Department of Information Technology (INTEC)
Ghent University, Belgium, e-mail: patrick.vanTorre@UGent.be

Abstract—In this paper we investigate the anchor calibration problem where we want to find the anchor positions when the anchors are not able to range between each other. This is a problem of practical interest because in many systems, the anchors are not connected in a network but are just simple responders to range requests. The proposed calibration method is designed to be fast and simple using only a single range-capable device. For the estimation of the inter-anchor distances, we propose a Total Least Squares estimator as well as a L_1 norm estimator. Real life experiments using publicly available hardware validate the proposed calibration technique and show the robustness of the algorithm to non-line-of-sight measurements.

Index Terms—anchor calibration, wireless localization, range-based

I. INTRODUCTION AND MOTIVATION

In the last decade, wireless indoor localization has received considerable attention in both academia and industry. By now, a number of technologies exist that provide accurate indoor localization either through angle-of-arrival, time-of-arrival, received signal strength (RSS), or some other type of measurement. Unfortunately, all the different technologies have one drawback in common, being the requirement of fixed infrastructure. This infrastructure generally consists of a number of fixed nodes, called anchors, for which the positions must be known by the user¹. Hence, all positions of the anchors should be measured first; this is referred to as the calibration of the localization system. However, this requires manual labor and is therefore costly, time consuming and may require expensive professional tools.

A commonly used approach to roughly localize (indoor) anchors is to collect a set of outdoor measurements from with an associated GPS tag. We call the device that makes measurements with the anchors the *calibration unit* (CU). The final anchor positions are then estimated through traditional localization algorithms where the GPS data points serve as the references to localize the actual anchor. In [1] a technique is proposed that locally looks for the direction of increasing RSS to provide a set of vectors that all point towards the anchor. In [2], the direction of the anchor is estimated by rotating the CU around the body of a person. However, due to the large error

in GPS coordinates, it is obvious that these methods can only provide very rough anchor information and estimation errors in the order of tens of meters have been reported with these approaches.

A more accurate solution is provided through SLAM (simultaneous localization and mapping) [3], where the calibration procedure is performed live whilst the localization system is running. Here the calibration unit is a robot that collects a sequence of different measurements such that it is possible to slowly but surely build up an increasingly accurate map of the environment. In SLAM, no GPS coordinates or any other absolute coordinates are used and consequently, only relative coordinates can be obtained for the anchors. Well-known methods such as EKF SLAM [4] or FAST SLAM [5] rely on motion models and require the knowledge of the so-called control vector that is controlling the CU: (i.e. 'move forward 20cm', 'turn 15° left', ..). As a result, SLAM is only applicable to automated devices and results in a (strongly) decreased performance in non-automated devices.

In this paper, we want to search for a calibration technique that requires, in contrast with the above-mentioned calibration techniques, no additional sensors or equipment. We restrict our attention to range-based calibration only. Consider a localization system with M range-capable anchors placed at some unknown positions $\mathbf{x}_i \in \mathbb{R}^\eta$ with $\eta = 2$ or $\eta = 3$ for 2D or 3D calibration, respectively. These anchors form a graph \mathcal{G} with as the vertices the anchors; two vertices are connected with an edge whenever the distance between the corresponding anchors $d_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|$ is available. For simplicity of presentation (but without loss of generality), we show in Fig. 1 the case of 3 anchors.

When only range-measurements are available, the most straightforward method to calibrate the anchors on a relative map, is to let every anchor make range measurements with the surrounding anchors. With the resulting range measurements, it is then possible to estimate the relative anchor positions using a cooperative localization algorithm such as MDS [6], WLS [7] or SDP [8]. Although this approach seems the most obvious one, this method has some shortcomings. First of all, this calibration method requires that the anchors are connected in a network, such that the ranging can be coordinated, and that the measurement data is transferred to a central processing

¹For some localization schemes, this is not true. For example, in RSS based fingerprinting the positions of the anchors (or access points) do not need to be known. However, these systems still require some other type of calibration.

node. However, most localization networks are user-oriented, and the anchors are simple responders that reply to a range request. This problem could be overcome by using additional equipment as e.g. a wifi device connected with the anchors, but this requires additional equipment, which is not favorable. A second problem is the presence of multi-path and non-line-of-sight measurements. Because the anchors are static, possible obstructions make it impossible to accurately measure the distance between two anchors. In the absence of one or more ranges, the resulting graph may no longer be redundantly rigid [9], and the localization of the anchors becomes impossible.

As an alternative, we could consider the use of calibration units placed at fixed positions. The motivation of this approach is that the presence of the calibration units (the dots in Fig. 1) results in a larger localization network such that the resulting larger graph becomes redundantly rigid, and anchor calibration is possible. However, a first disadvantage of this technique is that generally a large number of CU's is needed to obtain accurate calibration. A second disadvantage is that in many cases the anchors are placed in such a way that they form the largest possible convex hull. As a result, the CU's will always be placed inside this convex hull resulting in bad geometric properties to localize the anchors (see Fig. 1 where the large error ellipse is the result of this bad geometry)

In this paper, we present a novel technique for simple and fast anchor calibration, which is in some way related to the SLAM principle. However, in contrast to the SLAM method, we do not require additional sensors or equipment. In the proposed method, the system administrator moves around one single range-capable calibration unit. Hence, in contrast with the SLAM method, no predefined motion model is needed. Further, as the administrator is moving around, the probability of a network with bad geometric properties that hinder the localization of the anchors, is reduced. The proposed method is able to do calibration in 2D as well as in 3D. To assess the accuracy of the proposed method, live measurements were done with P410 UWB ranging devices [10]. In this live measurement campaign, the administrator was walking at a normal pace (ca. 1m/s), resulting in an average position error of 8cm. Hence the proposed method is well suited for practical anchor calibration.

II. ANCHOR CALIBRATION

The estimation of the anchor positions is obtained in two steps. In a first step, the distances between all anchors are estimated. For this step the system administrator has to be charged with the simple task of walking from an arbitrary point towards an anchor in a straight line, over a distance L_{path} , with a range capable device. During this procedure, the range capable device makes range measurements with every anchor. This must be repeated for every anchor such that all the distances between the anchors can be estimated. After this step, which is called the range collection, the anchor positions are estimated in the second step with a traditional cooperative localization algorithm.

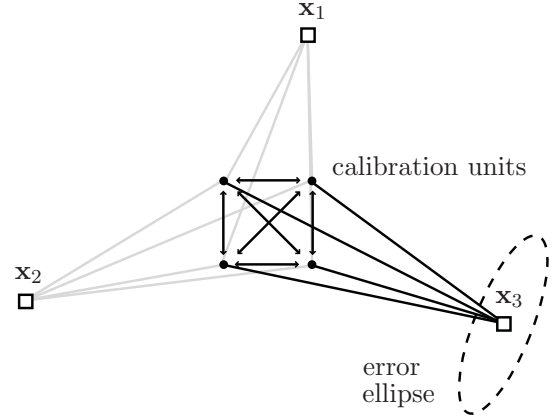


Figure 1. Range based anchor calibration methods. Anchors are represented by squares and calibration units by dots. The ellipse represents the uncertainty in the position of the anchor after calibration.

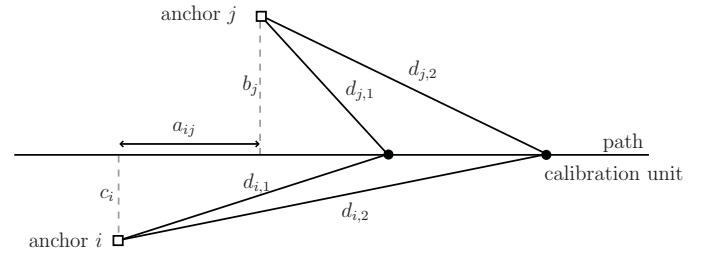


Figure 2. Anchor calibration in 2D. Anchors are represented by squares, the different positions of the calibration unit by a dot.

A. Range collection in 2D

The first step of the calibration method consists of finding the distances d_{ij} between all combinations of anchors i and j by performing some simple operations with the CU. It is in this spirit that we consider the scenario where the CU moves along an arbitrary straight line. Along this line, multiple range measurements have to be made with every anchor (if possible). The set of N_i^ℓ range measurements made along path ℓ with anchor i is denoted by $\mathbf{d}_i^\ell = [d_{i,1}^\ell, d_{i,2}^\ell, \dots, d_{i,N_i^\ell}^\ell]^T$. For each path ℓ and set of anchors (i, j) , we can introduce some additional parameters a_{ij}^ℓ , b_j^ℓ and c_i^ℓ which are defined in Fig. 2. For notational convenience we drop the path index ℓ .

In the absence of errors, we find the following relation between the distances $d_{i,k}$ and $d_{j,k}$ for $k = 1..N_i$, and the parameters a_{ij} , b_j and c_i :

$$\begin{aligned} b_j^2 &= d_{j,k}^2 - \left(\sqrt{d_{i,k}^2 - c_i^2} - a_{ij} \right)^2 \\ &= d_{j,k}^2 - d_{i,k}^2 + c_i^2 - a_{ij}^2 + 2a_{ij} \sqrt{d_{i,k}^2 - c_i^2}. \end{aligned} \quad (1)$$

If there are sufficient measurements, it is possible to estimate the parameters a_{ij}^2 , b_j^2 and c_i^2 from this non-linear set of equations. However, from Fig. 2 it can be observed that using these parameters, it is not possible to uniquely estimate the distance d_{ij} . This is due to the flip ambiguity of the anchors

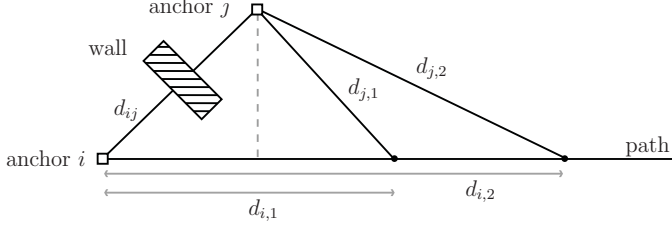


Figure 3. Anchor calibration in 2D with a path going straight to one anchor. A direct measurement of the distance d_{ij} between anchors i and j is not possible due to an obstruction (e.g. a wall).

around the path. However, if the calibration unit would move in a straight line towards an anchor, the flip ambiguity of this anchor disappears. The other anchor can now still flip without affecting distance d_{ij} . Hence it becomes possible to make an unambiguous estimation of d_{ij} . Moving towards anchor i corresponds to setting $c_i = 0$, resulting in the relation $d_{ij}^2 = a_{ij}^2 + b_j^2$. Furthermore, under this condition, equation (1) becomes linear in the unknown variables d_{ij}^2 and a_{ij} , resulting in

$$d_{ij}^2 - 2a_{ij}d_{i,k} = d_{j,k}^2 - d_{i,k}^2. \quad (2)$$

The scenario with $c_i = 0$ is depicted in Fig. 3. From this figure it becomes clear that, with this method, the distance d_{ij} can be measured even if the direct path between the two anchors i and j is obstructed.

Notice that we can rewrite equation (2) as follows

$$d_{j,k}^2 = (d_{i,k} - a_{ij})^2 - a_{ij}^2 + d_{ij}^2. \quad (3)$$

Hence, in order to estimate the unknown variables, we must fit a parabola to the measured noisy data pairs $(d_{i,k}, d_{j,k}^2)$ for $k = 1..N$. The intersection of this parabola with the y-axis gives the estimated squared distance \hat{d}_{ij}^2 .

In order to obtain the distances d_{ij} between all combinations of anchors (i, j) , we need to walk towards (at least) $M - 1$ different anchors. For the estimation of d_{ij} we can thus have more than one path available (for example one towards anchor i and another towards anchor j). Let us consider L available paths moving towards either anchor i or j . This results in $L + 1$ unknowns: d_{ij}^2 and a_{ij}^ℓ (one for each path) which we group in the parameter vector $\mathbf{x} = [d_{ij}^2, a_{ij}^1, a_{ij}^2, \dots, a_{ij}^L]^T$. The relationship of the parameter vector with the measurements is given by (2). Let us introduce the matrix $\mathbf{A}_{\rightarrow i}^\ell$ gathering all information for a single path ℓ going in the direction of anchor i , denoted by the subscript $\rightarrow i$:

$$\mathbf{A}_{\rightarrow i}^\ell = \begin{bmatrix} \mathbf{1}_{N_i^\ell \times 1} & \mathbf{0}_{N_i^\ell \times (\ell-1)} & -2\mathbf{d}_i^\ell & \mathbf{0}_{N_i^\ell \times (L-\ell)} \end{bmatrix}, \quad (4)$$

with $\mathbf{1}_{m \times n}$ and $\mathbf{0}_{m \times n}$ the all ones and all zeros matrix, respectively. Further, $\mathbf{b}_{\rightarrow i}^\ell$ is a N_i^ℓ dimensional vector with elements:

$$[\mathbf{b}_{\rightarrow i}^\ell]_n = \left([\mathbf{d}_j^\ell]_n \right)^2 - \left([\mathbf{d}_i^\ell]_n \right)^2. \quad (5)$$

We can now write the system of equations as $\mathbf{A}\mathbf{x} = \mathbf{b}$ where \mathbf{A} and \mathbf{b} are obtained by stacking the different $\mathbf{A}_{\rightarrow i}^\ell$ and $\mathbf{b}_{\rightarrow i}^\ell$. This leads to an overdetermined system for which we can find a solution in a number of ways. The most straightforward estimation method is the well known Least Squares (LS) method for which the estimate is given by:

$$\hat{\mathbf{x}}_{LS} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b}. \quad (6)$$

In Least Squares estimation, the L_2 norm $\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2$ is minimized, which implicitly assumes that the measurement errors are confined to the vector \mathbf{b} . However, because both $d_{i,k}$ and $d_{j,k}$ are subject to errors, the matrix \mathbf{A} is in fact also a random matrix and the LS assumption breaks. Because of this we expect reduced performance when applying standard Least Squares. To remedy this, we propose to employ Total Least Squares (TLS) which accounts for errors in both \mathbf{A} and \mathbf{b} . In TLS, the Frobenius norm $\|[\mathbf{E} \ \mathbf{f}]\|_F$ is minimized where \mathbf{E} and \mathbf{f} are the errors in \mathbf{A} and \mathbf{b} respectively. It is shown in [11] that the TLS solution to $\mathbf{A}\mathbf{x} = \mathbf{b}$ can be obtained by singular value decomposition (SVD) of the augmented matrix $[\mathbf{A} \ \mathbf{b}] = \mathbf{U}\Sigma\mathbf{V}^*$. The TLS estimate is given by $\hat{\mathbf{x}}_{TLS} = -\mathbf{V}_{xy}\mathbf{V}_{yy}^{-1}$ with \mathbf{V}_{xx} the upper right $(L+1) \times 1$ block of \mathbf{V} and \mathbf{V}_{yy} the lower right element of \mathbf{V} .

It is known that applying LS or TLS works well when the errors are normally distributed and small. However from the experiments we conducted (see subsection III-B), we observed that the measurements contained a lot of outliers due to non-line-of-sight (NLOS) which cannot be handled well by these estimators. As an alternative, we minimized the L_1 norm $\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_1$ instead of the L_2 norm in LS and TLS². Minimizing the L_1 norm is more robust to outliers but does not result in a closed form expression for the estimate. Rather, it results in solving the following Linear Program in $\mathbf{x} \in \mathbb{R}^{L+1}$ and $\mathbf{s} \in \mathbb{R}^{(N_i^1 + \dots + N_i^L)}$

$$\begin{aligned} & \text{minimize} && \mathbf{1}^T \mathbf{s} \\ & \text{subject to} && \mathbf{A}\mathbf{x} - \mathbf{b} \preceq \mathbf{s} \\ & && \mathbf{A}\mathbf{x} - \mathbf{b} \succeq -\mathbf{s} \end{aligned}$$

where \preceq and \succeq are elementwise inequality signs. The performance of the above estimators is compared in section III-B.

B. Range collection in 3D

Next we explore the possibility of using the same simple calibration procedure for anchors in a three-dimensional space. Similarly as in 2D, this is possible by moving the CU in a straight line towards the anchor. However, in general, this straight line is no longer parallel to the floor. If we would use a trajectory parallel to the floor, the possible anchor positions are subject to a rotational ambiguity around the line of the movement. Hence, as the possible anchors lie on a circle around the path, it is not possible to obtain an unambiguous distance d_{ij} . However, in case that the heights of the anchors

²The Frobenius norm is the ‘‘entrywise’’ L_2 norm for matrices.

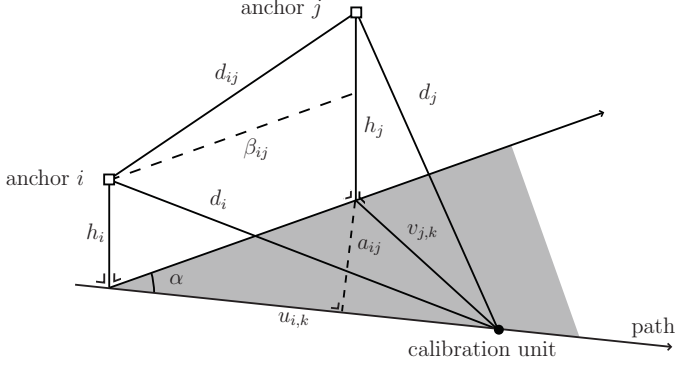


Figure 4. Anchor calibration in 3D. The gray plane is a plane parallel to the floor, where the CU is moving.

are known, the calibration method can still work with a few adjustments.

By inspection of Fig. 4, we can reformulate the problem by introducing the following variables. Consider the orthogonal projection of the anchors i and j on the plane in which the CU is moving, parallel to the floor. The distances $u_{i,k}$ and $v_{j,k}$ are the distances between the CU at position k and the projections of anchors i and j on this plane. Hence, $u_{i,k}^2 = d_{i,k}^2 - h_i^2$ and $v_{j,k}^2 = d_{j,k}^2 - h_j^2$. In this plane we recognize the 2D scenario and it follows from (2) that the distance between the projected anchors equals $\beta_{ij}^2 = v_{j,k}^2 - u_{i,k}^2 + 2u_{i,k}a_{ij}$. Taking this into account, we obtain the distance between anchors i and j :

$$d_{ij}^2 = (h_j - h_i)^2 + \beta_{ij}^2 \quad (7)$$

$$= d_{j,k}^2 - d_{i,k}^2 + 2h_i^2 - 2h_i h_j + 2a_{ij} \sqrt{d_{i,k}^2 - h_i^2} \quad (8)$$

In the special case where all anchors are at the same height, this can be simplified to

$$d_{ij}^2 = d_{j,k}^2 - d_{i,k}^2 + 2a_{ij} \sqrt{d_{i,k}^2 - h^2} \quad \text{if } h_i = h_j = h. \quad (9)$$

Expression (8) results in a set of linear equations from which the squared distance d_{ij}^2 can be estimated in the same way as described in the 2D case. By walking towards the M different anchors, in a straight line parallel to the floor, it is possible to obtain the complete distance graph which is fully connected.

C. Anchor localization

In the last step of the calibration procedure, the actual anchor positions are estimated. Because we have a full distance matrix (i.e. all pairs of distances are estimated), we can simply use a cooperative localization algorithm that delivers the relative coordinates of the anchors. We formulate the localization problem as a weighted least squares (WLS) problem such as in [7] because this can easily be adapted to 3D localization with known heights. The cost function to be minimized equals:

$$f(\mathbf{x}_{1:M}) = \sum_i^M \sum_j^M w_{ij} e_{ij}^2, \quad (10)$$

with $e_{ij} = \hat{d}_{ij}^2 - (\mathbf{x}_i - \mathbf{x}_j)^T (\mathbf{x}_i - \mathbf{x}_j)$ the error in squared Euclidean distance. Because all pairwise distance estimates are available and all weights are equal, this reverts to a convex problem with a unique solution (implying that no special care must be taken to select an initial estimate). Minimization of (10) can efficiently be done using the iterative Newton's method. Each iteration k , the estimate is updated according to: $\mathbf{x}_{1:M}^{(k+1)} = \mathbf{x}_{1:M}^{(k)} + \Delta \mathbf{x}_{1:M}^{(k)}$, where the superscript indicates the iteration index and $\Delta \mathbf{x}_{1:M}^{(k)} = -[\nabla^2 f(\mathbf{x}_{1:M}^{(k)})]^{-1} \nabla f(\mathbf{x}_{1:M}^{(k)})$ is called the Newton step. For the calculation of the Newton step, a closed form expression for the gradient and Hessian of the cost function is required. For the two-dimensional case, the gradient of $f(\mathbf{x}_{1:M})$ is a $2 \times M$ matrix with components:

$$\nabla f(\mathbf{x}_{1:M}) = \begin{bmatrix} \frac{\partial f(\mathbf{x}_{1:M})}{\partial \mathbf{x}_1} & \frac{\partial f(\mathbf{x}_{1:M})}{\partial \mathbf{x}_2} & \cdots & \frac{\partial f(\mathbf{x}_{1:M})}{\partial \mathbf{x}_M} \end{bmatrix}, \quad (11)$$

where the partial derivatives are given by:

$$\frac{\partial f(\mathbf{x}_{1:M})}{\partial \mathbf{x}_i} = -8 \sum_j^M w_{ij} e_{ij} (\mathbf{x}_i - \mathbf{x}_j). \quad (12)$$

The $2M \times 2M$ Hessian matrix is constructed by 2×2 blocks:

$$[\nabla^2 f(\mathbf{x}_{1:M})]_{ij} = \begin{bmatrix} \frac{\partial^2 f(\mathbf{x}_{1:M})}{\partial \mathbf{x}_i \partial \mathbf{x}_j} \end{bmatrix}, \quad (13)$$

where $\frac{\partial^2 f(\mathbf{x}_{1:M})}{\partial \mathbf{x}_i \partial \mathbf{x}_j}$ are 2×2 matrices defined as:

$$\frac{\partial^2 f(\mathbf{x}_{1:M})}{\partial \mathbf{x}_i \partial \mathbf{x}_j} = \begin{cases} -8 \sum_l^M w_{il} (e_{il} \mathbf{I} - 2(\mathbf{x}_i - \mathbf{x}_l)(\mathbf{x}_i - \mathbf{x}_l)^T) & i = j \\ -8w_{ij} (-e_{ij} \mathbf{I} + 2(\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T) & i \neq j \end{cases}$$

For three-dimensional localization with known heights, the above algorithm can be used by adding the height to the parameter vector $\mathbf{x}_{1:M}$ and setting the derivatives to the z-coordinate equal to zero.

III. RESULTS

A number of simulation tests as well as real-live tests were performed in order to investigate the performance of the proposed calibration method. Furthermore we wanted to study the impact of some parameters such as the length of the path to walk or the number of measurements made along each path. In order to evaluate the estimation errors on the anchor positions it is necessary to convert the obtained relative coordinates to absolute coordinates. This requires finding a suitable rigid transformation (i.e. a rotation, reflection and translation), which is achieved by performing the Procrustes algorithm [12]. This algorithm returns a rigid transformation that minimizes the squared error between the relative positions and the true (absolute) positions of the anchors. The resulting MSE corresponds to the error made in the calibration.

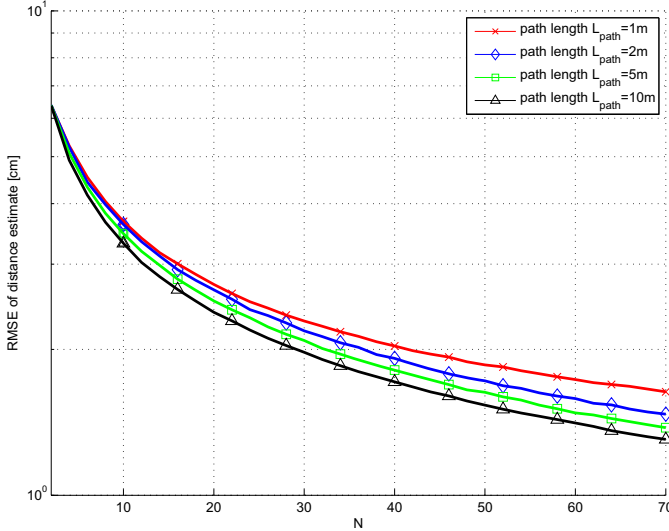


Figure 5. Root Mean squared error (RMSE) of the distance estimates between the anchors as a function of the number of measurements N .

A. Simulation results

In our simulation setup we placed 4 anchors in the corners of a square $10\text{m} \times 10\text{m}$ room. For each anchor, a random starting point within the room at a distance L_{path} from the anchor was selected to start the path. In Fig. 5 the root mean squared error (RMSE) of the distance estimates are shown as a function of the number of measurements N and the path length L_{path} such that the distance between two measurements equals L_{path}/N . In these simulations, the range measurements were modeled as Gaussian variables with zero mean and standard deviation $\sigma_n = 5\text{cm}$. The estimations were obtained using the Total Least Squares algorithm using a single path for each anchor. From this figure it follows that with a reasonable number of measurements and path length, the proposed algorithm is able to estimate the distance with centimeter accuracy. Furthermore, it is observed that the RMSE decreases with N as well as L_{path} . The fact that a longer path provides more accurate results can be understood by considering that the latent variable a_{ij} also plays an important role in the estimation, and that this variable is more accurately estimated when the measured points cover a longer distance.

Next we want to investigate the effect of the direction of the path on the estimation of the distances. If we vary the angle α of the path of the CU with the line between the two anchors (see Fig. 4), we observe from Fig. 6 that depending on the length of the path, some angles provide better estimation accuracy than others. However, it always holds true that the worst performance is obtained when $\alpha = 180^\circ$. The best result is generally found around $\alpha \approx 80^\circ$.

As a general rule of thumb, the CU should follow paths that are inside of the convex hull of the anchors as this minimizes the probability of unfavorable angles.

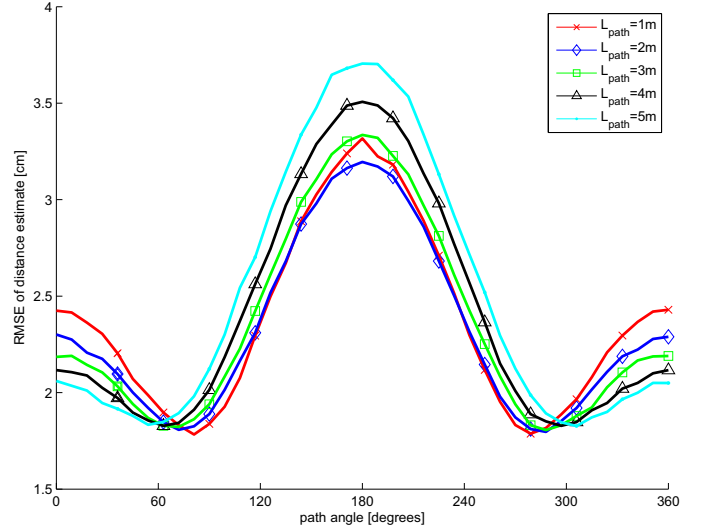


Figure 6. RMSE of the distance estimates as a function of the path angle α using $N = 30$.

B. Experimental results

To validate the proposed anchor calibration technique, real life experiments were performed with the P410 RCM ultra-wideband (UWB) modules [10] from Timedomain Corp. These modules perform ranging using a two-way time of arrival (TW-TOA) protocol providing 5 cm accuracy under line-of-sight conditions. These modules simply respond to ranging requests. In our test setup we placed 4 UWB devices at chest-height in the corners of a rectangular $5\text{m} \times 9\text{m}$ room to serve as anchors. The actual anchor positions were obtained by measuring the orthogonal distances to the walls with a laser-based distance meter (up to 2mm accurate). The distance measurements were always with respect to the phase center of the UWB antenna. One additional UWB device served as the calibration unit and was connected to a microcontroller that initiated ranging with the anchors and saved the resulting measurements. A button connected to this microcontroller was used to indicate the starting and ending point of a path. In total 15 calibration runs were made, where in each run, the system administrator walked once towards every anchor with the CU in his hands. With the P410 module, range measurements take around 40ms such that on a path of 4m, walking at a speed of 1m/s, a total of 25 measurements can be made to each anchor. Due to varying path lengths and walking speed in our tests, the number of measurements for each path was between 20 and 28. Further, the P410 module provides a flag to indicate if a measurement is subject to non-line-of-sight (NLOS). Although we used this flag to discard NLOS measurements, we have noticed that still some heavily biased range-measurements slipped through.

In Fig. 7, the paths resulting from a single calibration run are shown. Here the positions of the CU are estimated using the range measurements and for each path a line is drawn indicating the average alignment of the path. Even though the system administrator walked towards the anchors without any

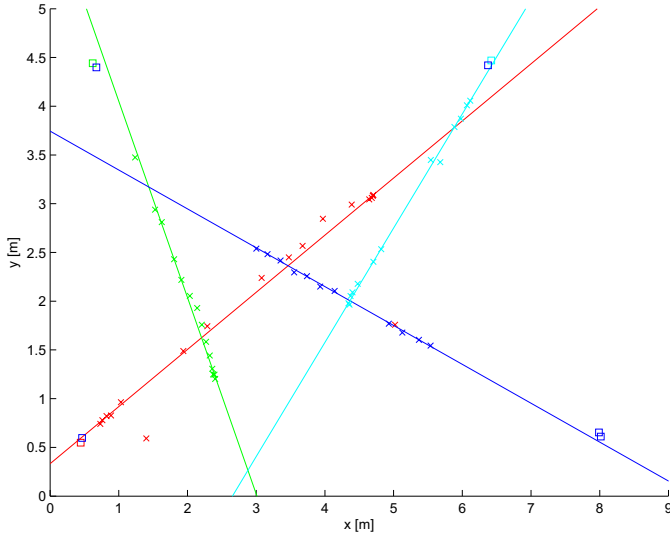


Figure 7. Paths of the calibration unit when performing anchor calibration with 4 anchors (shown as squares). The crosses indicate points where the CU has made a measurement with all anchors. The solid lines show the average alignment of each path.

guidance, it can be seen that the average alignments of the paths are indeed (almost) towards their corresponding anchor. Furthermore, we observe in Fig. 7 that the anchor positions estimated with the proposed method are very close to the actual anchor positions. Notice that this figure also provides good visual information about the quality of the calibration. For example curving paths or widely spread points around the path both indicate that the quality of the calibration is poor and that the process should be redone.

In Fig. 8, the cumulative distribution function of the estimation error on the distance is shown for the different algorithms proposed in the paper. It can be seen that the L1-norm cost function performs better than both the Total Least Squares and the Least Squares. Even though the UWB device gives an indication of a possible NLOS condition, it is by no means correct all the time and the more robust L1-norm is capable of dealing with these outliers, resulting in better performance.

Finally, we estimated the 2D anchor positions using the WLS algorithm for each calibration run. As inter-anchor distances we used the L1-norm estimated distances as these yield the best estimation. For 15 calibration runs, we obtained an average positioning error of 8cm.

IV. CONCLUSIONS

In this paper we propose a fast method to accurately estimate the position of the anchors using only range information. The calibration procedure simply requires that the system administrator walks towards the anchors. With this approach, the complete distance matrix for the anchors can be estimated and the relative positions of the anchors can be obtained. This procedure provides a practical alternative for anchor calibration whenever the anchors cannot range themselves (either by hardware restrictions or due to obstructions causing NLOS). Both simulations and real-life experiments validate

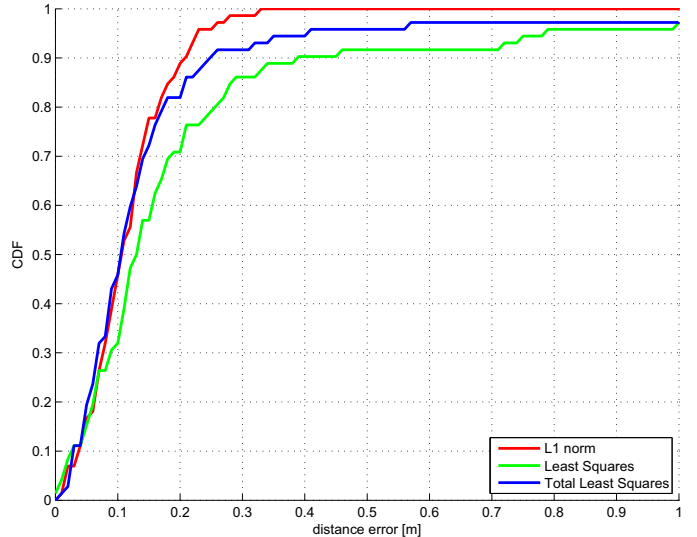


Figure 8. Cumulative Distribution Function for the distance error.

the proposed calibration method and using publicly available hardware, an average error of 8 cm was found.

ACKNOWLEDGMENTS

The first author gratefully acknowledges the financial support from the Belgian National Fund for Scientific Research (FWO Flanders).

REFERENCES

- [1] D. Han, D. G. Andersen, M. Kaminsky, K. Papagiannaki, and S. Seshan, "Access point localization using local signal strength gradient," in *Passive and Active Network Measurement*. Springer, 2009, pp. 99–108.
- [2] Z. Zhang, X. Zhou, W. Zhang, Y. Zhang, G. Wang, B. Y. Zhao, and H. Zheng, "I am the antenna: accurate outdoor ap location using smartphones," in *Proceedings of the 17th annual international conference on Mobile computing and networking*. ACM, 2011, pp. 109–120.
- [3] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: part i," *Robotics & Automation Magazine, IEEE*, vol. 13, no. 2, pp. 99–110, 2006.
- [4] R. Smith, M. Self, and P. Cheeseman, "Estimating uncertain spatial relationships in robotics," in *Autonomous robot vehicles*. Springer, 1990, pp. 167–193.
- [5] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, "Fastslam: A factored solution to the simultaneous localization and mapping problem," in *AAAI/IAAI*, 2002, pp. 593–598.
- [6] W. S. Torgerson, "Multidimensional scaling: I. theory and method," *Psychometrika*, vol. 17, no. 4, pp. 401–419, 1952.
- [7] G. Destino and G. de Abreu, "Sensor localization from wls optimization with closed-form gradient and hessian," *In proc. IEEE GLOBECOM '06*, pp. 1–6, Dec 2006.
- [8] P. Biswas and Y. Ye, "Semidefinite programming for ad hoc wireless sensor network localization," *In proc. IEEE IPSN '04, Berkeley, CA*, pp. 46–54, 2004.
- [9] D. Goldenberg, A. Krishnamurthy, W. Maness, Y. Yang, A. Young, A. Morse, and A. Savvides, "Network localization in partially localizable networks," vol. 1, pp. 313–326 vol. 1, Mar 2005.
- [10] "P410 data sheet," *Time Domain Corp., Huntsville, AL, USA*, <http://www.timedomain.com/p410.php>.
- [11] G. H. Golub and C. F. Van Loan, "An analysis of the total least squares problem," *SIAM Journal on Numerical Analysis*, vol. 17, no. 6, pp. 883–893, 1980.
- [12] C. Goodall, "Procrustes methods in the statistical analysis of shape," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 285–339, 1991.