2012 International Conference on Indoor Positioning and Indoor Navigation, 13-15th November 2012

CUPID algorithm for cooperative indoor multipath-aided localization

Samuel Van de Velde and Heidi Steendam
Department of Telecommunications and Information Processing,
Ghent University, Belgium, e-mail: {slvdveld, hs}@telin.ugent.be

***Abstract*—The main drawback today for range-based indoor localization is the requirement of a sufficient amount of fixed reference nodes within radio range of the user. However, these reference nodes, called anchors, are expensive and require professional maintenance. Using ultra-wideband in an indoor environment, the number of anchors can be reduced to one when reflections are taken into account. With the help of a floorplan it is possible to obtain a set of virtual anchors that can be associated with the reflections. In this paper, a *low-complex* two-step algorithm is proposed that is able to accurately estimate the user positions using a single anchor. In a first step, the algorithm tries to estimate a number of rigid structures using the noisy inter-node distances and tries to fit this structure in the room by exploiting the measured reflections. It is shown that the presented algorithm can provide positioning accuracy similar to multi-anchor localization algorithms, even in scenarios with many unwanted scatterers and non-line-of-sight.**

***Index Terms*—Indoor Localization, multipath, Cooperation, Ultra-Wideband**

I. Introduction

In recent years, position-aware computing has become an important part of mobile systems, both in indoor and outdoor environments. For the outdoor environment, the global positioning system (GPS) serves as the leading technology to provide this position-awareness. However, for the indoor environment, no single technology has proved to meet the various requirements yet. Many indoor systems utilize the same principle for localization as is used for GPS, where range measurements are made between some known reference nodes and the users. However, due to the harsh indoor environment with many obstacles that may block or reflect the transmitted signals, it becomes hard to reliable locate the user from the received signals. With the advent of ultra-wideband technology [1], with its fine delay resolution and ability to resolve multipath components, it has become possible to provide accurate line-of-sight (LOS) range estimates necessary for localization. However, due to the limited radio-range of these UWB signals and blockage of signals by obstacles, still a large number of reference nodes are required for accurate positioning. Placing and maintaining these reference nodes is expensive and time-consuming which may render the localization system cost-inefficient. This problem has been partially solved by adopting cooperation, in which all users collaborate in finding and refining their position. In [2], [3] several cooperative algorithms are presented where positioning is possible with a low number of reference nodes. When there are too few users or too much NLOS, these systems break down and some extra reference nodes remain necessary. As a result, researchers continued to search for solutions to keep the number of reference nodes low.

It has been observed in [4], [5], that not only the LOS range measurement is useful for localization but also the signal reflections. Using ultra-wideband, it is possible to distinguish the multipath components (MPCs). Such a MPC can be seen as a LOS component transmitted by a virtual anchor behind the corresponding reflecting surface. However, due to the non-identifiability of the measured MPCs, simple multilateration techniques are not applicable. In a probabilistic approach where reflections have a certain probability of coming from a certain virtual anchor, estimating the absolute position of a user becomes possible. In [6], an experimental study using real measurements has shown a proof-of-concept where the user is located with a single anchor. For a single user, the localization accuracy remains rather poor and therefore cooperation between users was added in [7]. While this algorithm performs very well, it is prohibitively complex and unsuitable for either larger networks or real-time localization.

In this paper, a *low-complex* algorithm for cooperative multipath-aided indoor localization called CUPID is presented.

II. Single anchor localization

Consider a wireless network of N users and a single anchor, situated in a room with known floor-plan. Let $\mathbf{x}_i \in \mathbb{R}^2$ be the position of user $i = 1..N$ and $\mathbf{x}_{N+1}$ the known position of the anchor. Using ultra-wideband, a measurement between two devices i and j is made by transmitting a narrow pulse from device i. At the receiver, a number of distorted copies of this pulse will be received, possibly including the line-of-sight component and some multipath components caused by reflections and scattering. In a synchronized network[1], the arrival time of the components can be converted to distances. We will assume that the receiver is able to identify the LOS component z^{ij}_{range} -if present- that represents the distance between devices i and j. In [8], [9], several methods are presented to determine the LOS nature of a measurement. We will model the LOS range estimates with a normal distribution with mean equal to the true distance and standard deviation equal to σ_{n}. The whole network is represented by an undirected graph $\mathcal{G} = (V, E)$ with vertices $V = \{\mathbf{x}_1, .., \mathbf{x}_N, \mathbf{x}_{N+1}\}$ representing the wireless devices and a set of edges E whenever a line-of-sight

[1]The assumption of synchronization can be omitted when a two-way ranging protocol is considered.

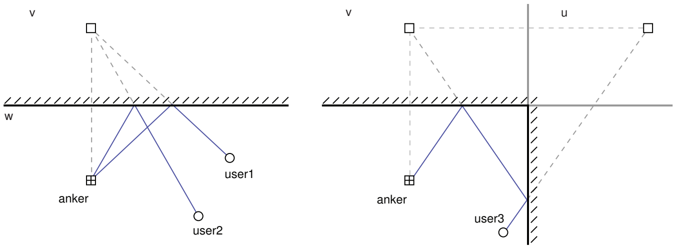

Figure 1. The reflections that the users receive can be seen as LOS components coming from the virtual anchors behind the wall. On the left we see a single reflection, on the right a double reflection.

range-measurement $z_{\mathrm{range},ij}$ is available between two devices i and j.

In traditional multi-anchor localization, the remaining multipath components are usually ignored and only the range measurements $\mathbf{z}_{\mathrm{range}}$ are used for estimating the user positions. It has been shown [4] however that the multipath components obtained from a measurement with the anchor still carry useful information when the floor-plan of the room is taken into account. For a user i, the multipath components received from the anchor are stored in the vector $\mathbf{z}_{\mathrm{mp},i} = \left\{z^i_{\mathrm{mp},m}\right\}$ with $m = 1..M_i$ and M_i the number of MPCs received by user i. A multipath measurement $z^i_{\mathrm{mp},m}$ can be seen as the distance between the user i and some virtual anchor (VA) behind the reflecting surface. In a typical indoor setting, the walls will cause most of the reflections and with a floor-plan available, a finite set of virtual anchors $\mathcal{V} = \{\mathbf{x}_{\mathrm{VA},p}\}$ can be obtained by mirroring the anchor position at each wall[2].

A clear picture of this is shown in Fig. 1 for both a single and a double reflection. Unfortunately, for a given multipath component, the corresponding virtual anchor is not known and because of this, simple multilateration techniques are not applicable for location estimation. Therefore, a probabilistic model where each multipath component has a certain probability of corresponding to a VA is an appropriate alternative.

Assuming the multipath components are independent from each other, we can write the likelihood function of the MPC measurement $\mathbf{z}^i_{\mathrm{mp}}$ given the position of user i as follows:

$$p\left(\mathbf{z}^i_{\mathrm{mp}} \mid \mathbf{x}_i\right) = \prod_{m=1}^{M_i} p\left(z^i_{\mathrm{mp},m} \mid \mathbf{x}_i\right). \tag{1}$$

The likelihood function for a single multipath measurement $\mathbf{z}^i_{\mathrm{mp},m}$ can be formulated as [4]:

$$\begin{aligned} &p\left(z^i_{\mathrm{mp},m} \mid \mathbf{x}_i\right) \\ &= \frac{P_{\mathrm{VA}}}{P_{\mathrm{v}}} \sum_{p=1}^{N_{\mathrm{VA}}} P_{\mathrm{v},p} \mathcal{N}(z^i_{\mathrm{mp},m}; \|\mathbf{x}_i - \mathbf{x}_{\mathrm{VA},p}\|, \sigma_p^2) + \frac{(1-P_{\mathrm{VA}})}{R_{\max}}. \end{aligned} \tag{2}$$

We made the following assumptions: A measurement $z^i_{\mathrm{mp},m}$ has probability P_{VA} of coming from a VA and $(1 - P_{\mathrm{VA}})$ of not being related to any VA in our finite set. In the latter case, $z^i_{\mathrm{mp},m}$ is uniformly distributed between 0 and the maximum range $R_{\max}$.[3] In the former case, we have a weighted sum of N_{VA} Gaussian rings centered around every virtual anchor (with position $\mathbf{x}_{\mathrm{VA},p}$) with radius $z^i_{\mathrm{mp},m}$ and standard deviation σ_p. The pth virtual anchor is visible with probability $P_{\mathrm{v},p}$. To ensure proper normalization, we introduce $P_{\mathrm{v}} = \sum_p^{N_{\mathrm{VA}}} P_{\mathrm{v},p}$.

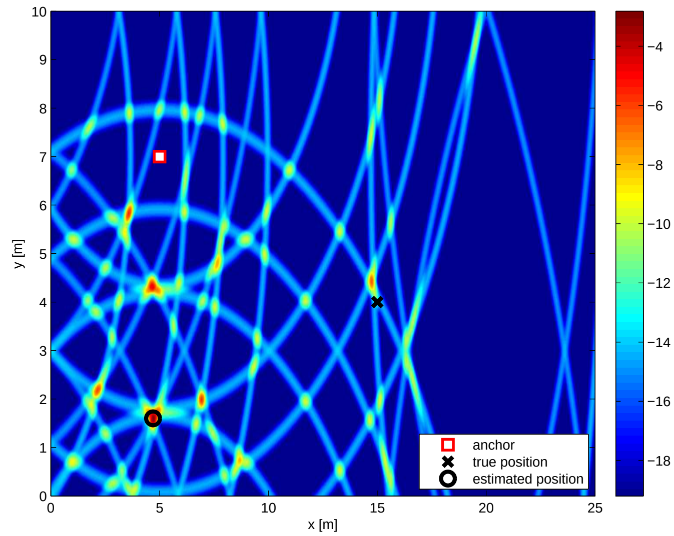

Figure 2. Plot of $\log p\left(\mathbf{z}^i_{\mathrm{mp}} \mid \mathbf{x}_i\right)$ where $\mathbf{z}^i_{\mathrm{mp}}$ contains 7 multipath components. In this plot the different intersecting Gaussian rings from (2) are clearly visible. Here the user $(15; 4)$ is not located in the absolute maximum of this function (4.78;1.54).

In Fig. 2, an example of the likelihood function $p\left(\mathbf{z}^i_{\mathrm{mp}} \mid \mathbf{x}_i\right)$ is shown for 7 multipath components. Because of the non-identifiability of the MPCs, this function has many local maxima and the true user position may even not be closely located near the absolute maximum of the function. Using this likelihood function alone for estimating the user position will result in bad performance. However, by introducing cooperation between multiple users, performance can be augmented.

The joint likelihood function of all observations $\{\mathbf{z}_{\mathrm{mp}}, \mathbf{z}_{\mathrm{range}}\}$ given the user positions $\mathbf{x}_{1:N} = \{\mathbf{x}_1, .., \mathbf{x}_N\}$ is given as follows:

$$\begin{aligned} &l(\mathbf{z}_{\mathrm{mp}}, \mathbf{z}_{\mathrm{range}} \mid \boldsymbol{x}_{1:N}) \\ &= \prod_{i=1}^{N} \underbrace{p\left(\mathbf{z}^i_{\mathrm{mp}} \mid \mathbf{x}_i\right)}_{\text{multipath}} \underbrace{\prod_{j \in \mathcal{N}(i)} p\left(z_{\mathrm{range},ij} \mid \mathbf{x}_i, \mathbf{x}_j\right)}_{\text{ranging}}. \end{aligned} \tag{3}$$

The maximum likelihood (ML) estimate for the user positions $\mathbf{x}_{1:N}$ is given by:

$$\hat{\mathbf{x}}_{1:N} = \arg\max_{\mathbf{x}_{1:N}} l\left(\mathbf{z}_{\mathrm{mp}}, \mathbf{z}_{\mathrm{range}} | \mathbf{x}_{1:N}\right). \tag{4}$$

[2]Note that, for the N users, the transmitter position is unknown and therefore no set of virtual anchors can be obtained.

[3]The maximum range $R_{\max}$ is the maximum distance between any virtual anchor, and one of the corners of the room.

In [7], the ML estimates are obtained by applying nonparametric belief propagation to a similar posterior distribution as (3). This iterative algorithm returns beliefs for every user that represent an approximation for the marginal distribution of the user position. However, these beliefs are generally very complex and a very large number[4] of particles are needed to accurately approximate it, resulting in a very computationally demanding algorithm.

Alternatively, point estimates for the user positions can be found using fast, low complexity optimization techniques by minimizing a suitable objective function related to the posterior (3). For multi-anchor cooperative localization this has proved to be very successful with algorithms such as weighted least squares [10], MDS [3] or SDP [11]. For the single-anchor problem however, the objective function is highly non-linear and non-convex and these algorithms cannot be directly applied. Because of this, one can only hope to find the maximum of (3) when sufficiently accurate initial estimates are provided.

III. CUPID ALGORITHM

A. Two step localization

In this paper, we propose a novel low-complexity, two-step algorithm called CUPID (Cooperative UWB Positioning Indoors) for estimating the user positions. The algorithm is based on the observation that a related problem to (4) can be solved efficiently, namely the problem of finding relative coordinates by matching the range measurements $\mathbf{z}_{\text{range}}$. By solving this related problem a relative map is obtained, subject to three degrees of freedom[5], i.e. a translation, rotation and reflection. In traditional multi-anchor localization, these relative coordinates can be converted to absolute coordinates by matching the anchor positions. For this operation, at least three anchors are necessary: one for each degree of freedom. If only a single anchor is available however, two degrees of freedom still remain. Now, the node in the relative map, corresponding to the anchor, can be seen as a pivot around which the relative map can be rotated or reflected. Estimating the absolute coordinates can thus be done by finding the rotation and reflection for the relative map that maximizes the likelihood function (3). Finding this transformation to absolute coordinates is a low-dimensional problem that can be efficiently solved, in contrast to the original problem (4) that requires the maximization over all possible user positions. The CUPID algorithm consists of these two steps: one for estimating the relative coordinates, and one for estimating the transformation to obtain absolute coordinates.

In the first step we ignore the multipath components and only take into account the range information $z_{\text{range},ij}$ for $i, j = 1, .., N+1$ corresponding to the LOS components between the different users and/or anchor. We reformulate this problem as a weighted least squares (WLS) problem such as in [10]. The cost function to be minimized is given as follows:

$$f(\mathbf{x}_{1:N+1}) = \sum_{i}^{N+1} \sum_{j}^{N+1} w_{ij} e_{ij}^2, \tag{5}$$

with $e_{ij} = (z_{\text{range},ij})^2 - (\mathbf{x}_i - \mathbf{x}_j)^T(\mathbf{x}_i - \mathbf{x}_j)$ the error in squared Euclidean distances. When all pairwise distance estimates are available and all weights are equal, this reverts to a convex problem with a unique solution. When some distance estimates are missing, which will be the case for NLOS measurements, this is no longer true. For now we will assume all range-measurements are available. How to deal with NLOS measurements will be discussed in the next section.

Minimization of (5) can be efficiently done using the iterative Newton's method. Each iteration k, the minimum is updated according to: $\mathbf{x}_{1:N+1}^{(k+1)} = \mathbf{x}_{1:N+1}^{(k)} + \Delta\mathbf{x}_{1:N+1}^{(k)}$. Here the superscript indicates the iteration index and $\Delta\mathbf{x}_{1:N+1}^{(k)} = -\left[\nabla^2 f(\mathbf{x}_{1:N+1}^{(k)})\right]^{-1} \nabla f(\mathbf{x}_{1:N+1}^{(k)})$ 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. The gradient of $f(\mathbf{x}_{1:N+1})$ is a $2 \times (N+1)$ matrix with components:

$$\nabla f(\mathbf{x}_{1:N+1}) = \left[\begin{array}{cccc} \frac{\partial f(\mathbf{x}_{1:N+1})}{\partial \mathbf{x}_1} & \frac{\partial f(\mathbf{x}_{1:N+1})}{\partial \mathbf{x}_2} & \dots & \frac{\partial f(\mathbf{x}_{1:N+1})}{\partial \mathbf{x}_{N+1}} \end{array} \right], \tag{6}$$

where the partial derivatives are given by:

$$\frac{\partial f(\mathbf{x}_{1:N+1})}{\partial \mathbf{x}_i} = -8 \sum_{j}^{N+1} w_{ij} e_{ij} (\mathbf{x}_i - \mathbf{x}_j). \tag{7}$$

The $2(N+1) \times 2(N+1)$ Hessian matrix is constructed by 2×2 blocks:

$$\left[\nabla^2 f(\mathbf{x}_{1:N+1})\right]_{ij} = \left[\frac{\partial^2 f(\mathbf{x}_{1:N+1})}{\partial \mathbf{x}_i \partial \mathbf{x}_j}\right], \tag{8}$$

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

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

With the WLS algorithm, relative coordinates are obtained, which corresponds to three degrees of freedom. The relative coordinates can be transformed to absolute coordinates by the following rigid (distance respecting) transformation:

$$\mathbf{x}_i = \mathbf{T}\mathbf{x}_i^{\text{rel}} + \mathbf{t}. \tag{9}$$

Here $\mathbf{T}$ is a 2×2 orthogonal matrix and $\mathbf{t} \in \mathbb{R}^2$ represents a translation. If three or more anchors are available, the transformation parameters $\mathbf{T}$ and $\mathbf{t}$ can be estimated by mapping the relative coordinates of the anchors to the known absolute

[4]In [7], 15000 particles were used to represent the beliefs.

[5]When some range measurements are missing, the corresponding map can have more degrees of freedom. See section III-B.

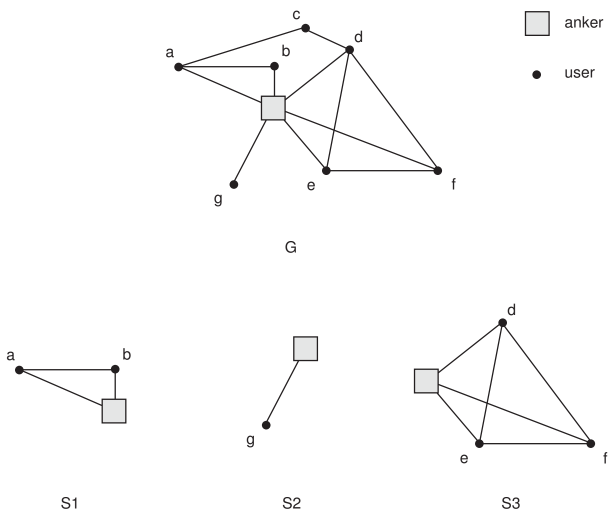

Figure 3. Graph $\mathcal{G}$ is not uniquely realizable (it is both not 3-edge-connected and not redundantly rigid) but can be decomposed into three uniquely realizable subgraphs $\mathcal{S}_j$.

coordinates of the anchors. This operation can be achieved by Procrustes analysis [12]. If only a single anchor is available, two degrees of freedom still remain which corresponds to the unknown orthogonal projection $\mathbf{T}$. Using the multipath components it is possible to estimate the orthogonal projection $\mathbf{T}$ with a ML estimator. The second step in the algorithm consists of estimating the transformation parameters and can be done as follows:

$$\hat{\mathbf{t}} = \mathbf{x}^{\mathrm{a}} - \mathbf{x}^{\mathrm{rel}}_{N+1} \tag{10}$$

$$\hat{\mathbf{T}} = \underset{\mathbf{T}}{\mathrm{argmax}}\, l\left(\mathbf{z}_{\mathrm{mp}}, \mathbf{z}_{\mathrm{range}} | \mathbf{T}\mathbf{x}^{\mathrm{rel}}_{1:N} + \hat{\mathbf{t}}\right) \tag{11}$$

Maximization of the function $l\left(T\mathbf{x}^{\mathrm{rel}}_{1:N} + \hat{\mathbf{t}} | \mathbf{z}_{\mathrm{mp}}, \mathbf{z}_{\mathrm{coop}}\right)$ parametrized over $\mathbf{T}$ is best performed by a grid search, because it has multiple local maxima similar to (2). The complexity of this grid search remains low due to the small parameter space of $\mathbf{T}$. This can be explained as follows: we can write the matrix $\mathbf{T}$ in terms of the angle of rotation $\theta \in [0, 2\pi)$ and a parameter $\alpha = [1, -1]$ that indicates if a reflection is present.

$$\mathbf{T} = \begin{bmatrix} \cos\theta & -\alpha\sin\theta \\ \sin\theta & \alpha\cos\theta \end{bmatrix}. \tag{12}$$

Hence, for each of the two values for α, i.e. ± 1, a one-dimensional search over θ must be performed.

B. NLOS and rigid graphs

The success of the aforementioned algorithm relies on the fact that the relative map, estimated from the first step, is an accurate representation of the true relative coordinates because this map is used for estimating the transformation to absolute coordinates. Using a bad estimate for the relative coordinates is likely to have a detrimental effect on the estimation of the projection parameters of the second step. Due to the high

Algorithm 1 General CUPID algorithm

1. **Find** all rigid subgraphs $\mathcal{S}_i$ of $\mathcal{G}$
2. **for** $\forall\, \mathcal{S}_i$ that include the anchor:
2a Perform WLS to estimate relative coordinates
2b Find orthogonal transformation to obtain absolute coordinates using (10-11)
3. **end for**
4. For nodes without estimate, perform ML

accuracy of UWB range measurements, it is possible to obtain a very accurate relative map with the important exception of non-line-of-sight (NLOS) measurements. In case of NLOS, the range measurement can have a large unknown positive bias that will deform the relative map when used for estimation. A possible solution to deal with NLOS measurements is to discard them, i.e. by setting the corresponding weights w_{ij} equal to zero. However, due to the missing links, the relative map may have more than three degrees of freedom. In such a case it becomes impossible to uniquely estimate the relative coordinates of the users and we say that the graph $\mathcal{G}$ in not uniquely realizable. An example of this is given in Fig. 3 where the graph $\mathcal{G}$ is not uniquely realizable. For this graph, an infinite amount of possible realizations exist. For example: node 7 can freely be rotated around the anchor node, nodes 5 and 6 can be flipped around the line through the anchor and node 4, etc...

In [13], two conditions are given such that a graph $\mathcal{G}$ has a unique realization in two dimensions[6]:

- $\mathcal{G}$ is 3-vertex-connected
- $\mathcal{G}$ is redundantly rigid (and thus also rigid)

The first condition makes sure there are no flip ambiguities. The second condition means that whenever an edge in the graph is removed, the graph remains rigid, which means it cannot undergo any flexible variation. This last condition also implies rigidity of the graph.

It is clear that we need a way to deal with relative maps that are not uniquely realizable, because they give bad results for the CUPID algorithm. If there are more degrees of freedom, we could use more complex transformations than (9) that take into account these extra degrees of freedoms. However, this would increase complexity of the algorithm dramatically which was the main goal of the algorithm in the first place. We will therefore use another approach. In Fig. 3, we can see that the original graph contains some parts that are well constrained. By searching the graph $\mathcal{G}$ such as is explained in [13] we can extract a set of uniquely localizable subgraphs $\{\mathcal{S}_i\}$, as is shown in Fig. 3. Because each subgraph is uniquely realizable, the relative coordinates of the corresponding nodes can be accurately estimated[7]. For each such subgraph that

[6]Under the assumption that no three nodes are collinear.

[7]Note that finding the unique realization of the map is possible, but remains a very hard task [14]. In order to obtain better estimates for the relative map we use the non-cooperate ML estimate of the user position by solving (4) with the neighborhood of every node i restricted to the anchor, i.e., by setting $\mathcal{N}_i = \{N+1\}$ for $\forall i$.

Table I
COMPARISON IN TIME COMPLEXITY(IN SECONDS) BETWEEN THE CUPID ALGORITHM AND BP.

	$N = 2$	$N = 3$	$N = 4$	$N = 5$	$N = 10$
CUPID	0.1	0.2	0.3	0.4	0.9
BP, $R_1 = 5000$	2.7	31	68	135	809
BP, $R_1 = 15000$	22	200	382	572	2405

contains the anchor node, we can apply the two-step algorithm from the previous section to obtain its absolute coordinates. After all subgraphs are processed, some nodes will have multiple positions estimates and some will have none. For simplicity we take the mean whenever multiple estimates are available and for nodes without a position estimate we use the non-cooperative ML estimate.

An overview of the CUPID algorithm adapted to NLOS measurements is given in Alg. 1.

IV. RESULTS

A. Complexity analysis

The main goal of this paper is to present a low-complex algorithm to solve the cooperative, multipath-aided localization problem. In [7], an algorithm was presented based on belief propagation (BP). The complexity of the algorithm follows from the computation of the messages between every neighboring user. In [7], each messages is represented with $R_1 = 15000$ particles and the computational complexity of a single message is $\mathcal{O}(R_1^2)$. The total complexity for a network with N users is thus of the order $\mathcal{O}\left(MR_1^2\right)$ where $M = C_2^N$ is the binomial coefficient representing the maximum number of links between the cooperating users.

In the CUPID algorithm, one Newton minimization and two one-dimensional searches are performed which require very low computational power. The main complexity here is determined by the calculation of the initial non-cooperative ML estimates which is done using a grid search using R_2 points. For N users, the complexity is of the order $\mathcal{O}(NR_2)$ with $R_2 = 10000$. In table I the computation time on an Intel i7 mobile processor in seconds is presented for both algorithms. It is clear that for any number N of cooperative users the CUPID algorithm is much faster than the algorithm using belief propagation.

B. Accuracy

We now present some numerical results for the CUPID algorithm. Results were obtained by means of Monte Carlo simulation using the parameters from table II.

In Fig 4, the cumulative distribution function is presented for the LOS scenario. It can be seen that both accuracy and robustness increase when the number of cooperative users increases. Here, we say that a localization algorithm is robust whenever there are only a few estimates having a very large error (for indoor localization we can say that an error exceeding 1m is large). Without cooperation, we see that only 64% of all estimates have a localization error smaller than 30cm as compared to the cooperative case with 10 users

Table II
SIMULATION PARAMETERS

Parameter	Note	Value
σ_p, σ_n	Ranging std. deviation	0.1m for $\forall\, p$
$P_{\mathrm{v},p}$	Visibility for VAs:	
	single reflections	0.5 for $1 \leq p \leq 4$
	double reflections	0.3 for $5 \leq p \leq 8$
P_{VA}	Prob. that a $z^i_{\mathrm{mp},m}$ is from a VA	0.4
$w \times h$	room dimensions	10 m $\times$ 25 m
$(x_\mathrm{a},\, y_\mathrm{a})$	anchor location	(5 m, 7 m)

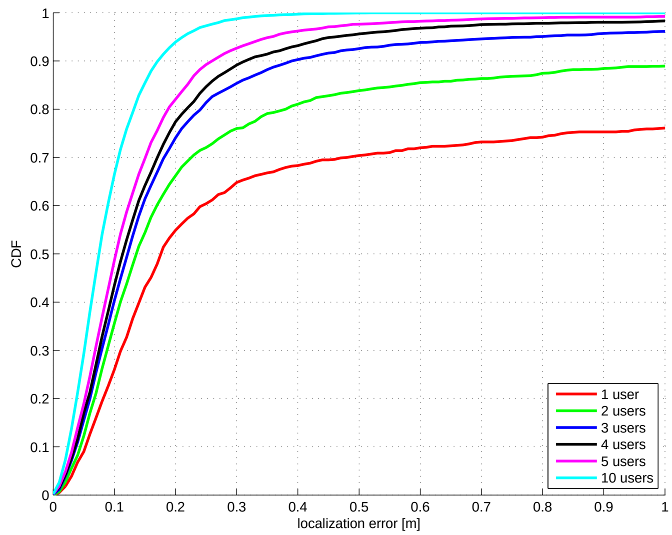

Figure 4. Cumulative distribution function for the estimation error obtained by the CUPID algorithm under LOS conditions for a different number of cooperating users.

where 99% of all estimates have a localization error smaller than 30cm. Furthermore, it is observed that, starting from 3-4 cooperating users, the algorithm becomes very robust, i.e., less than 4% of the estimates have a large error.

In Fig. 5 we show the effect of NLOS by setting the probability of LOS at 80%. If we directly apply the CUPID algorithm in such a scenario, performance becomes very bad. This can be seen for $N = 3$ and $N = 5$ (dashed line). If we use the theory from section III-B and apply the CUPID on the different uniquely localizable subgraphs and use the non-cooperative ML estimates as initial estimates, we obtain much better performance (solid line).

In order to appraise the resulting performance of the CUPID algorithm, a comparison is made with the existing algorithm from [7] that uses belief propagation to solve the same multipath problem. Furthermore, both these algorithms are compared against a traditional multianchor localization algorithm that uses three anchors and only range information. For the multianchor estimation we use the WLS algorithm from step 1 and perform procrustes analysis afterward to obtain absolute coordinates. For all algorithms there is again only 80% LOS. In Fig. 6 the probability of a localization error larger than 1m is shown for a varying number of cooperative users. It is observed that if the number of cooperative users is low, robust localization is not possible: between 10 to 30% of

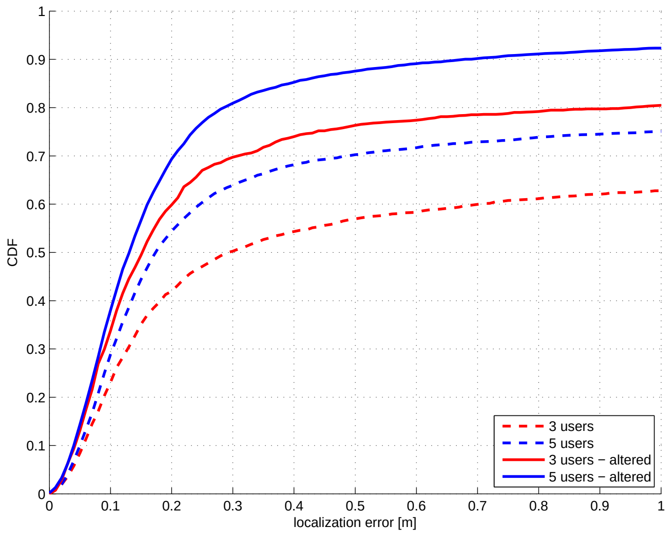

Figure 5. Cumulative density function under NLOS conditions for the estimation error obtained by the CUPID algorithm used on the whole graph $\mathcal{G}$ (dashed line, resulting in bad performance) or the different subgraphs (solid line).

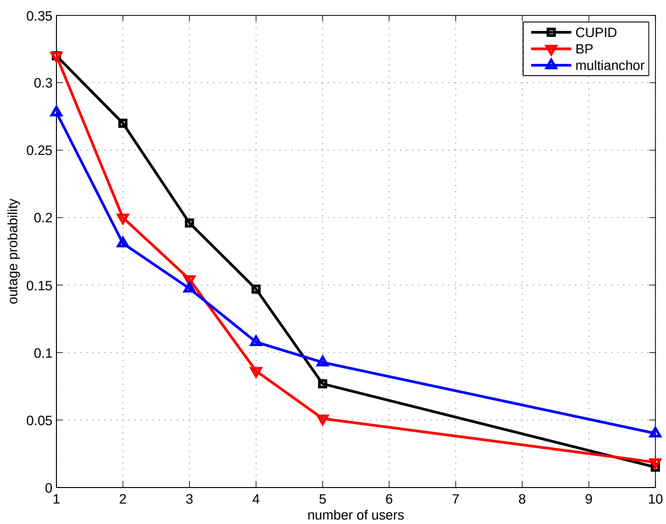

Figure 6. Plot of the outage probablity $P(\text{error} > 1\text{m})$.

all users are miss-located. It is seen that for a low number of cooperating users, the CUPID algorithm generally requires one more user than BP to obtain a similar outage probability. This is more or less in line with the anticipation because the BP algorithm does a joint maximization at the expense of higher complexity. However, when the number of users increases, the difference in performance for both algorithms becomes negligible.

V. Conclusions and Future Work

In this paper, the problem of single-anchor cooperative localization is considered. This problem generally leads to very computationally complex algorithms if solved jointly. In this paper a low-complex algorithm called CUPID is presented for cooperatively estimating the user positions using a single anchor. In a first step the relative coordinates of the cooperating users are estimated using inter-node range measurements. These relative coordinates are then transformed to absolute coordinates by also taking multipath components into account. In case of LOS this algorithm performs very well. When NLOS links are present however, the relative map representing the cooperating users may be no longer uniquely realizable and performance loss is expected. A solution is to apply the CUPID algorithm on different uniquely realizable subgraphs in order to maintain a good position estimate. Future work includes adding tracking and testing the algorithm on real-life data.

Acknowledgments

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

References

[1] S. Gezici, G. Giannakis, H. Kobayashi, A. Molisch, H. Poor, and Z. Sahinoglu, "Localization via ultra-wideband radios: a look at positioning aspects for future sensor networks," *IEEE Signal Processing Magazine*, vol. 22, pp. 70–84, Jul 2005.

[2] H. Wymeersch, J. Lien, and M. Z. Win, "Cooperative localization in wireless networks," *In proc. of the IEEE*, vol. 97, pp. 427–450, Feb 2009.

[3] Y. Shang, W. Rumi, Y. Zhang, and M. Fromherz, "Localization from connectivity in sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 15, pp. 961–974, Nov 2004.

[4] P. Meissner, T. Gigl, and K. Witrisal, "UWB sequential monte carlo positioning using virtual anchors," in *In proc. of the International Conference on Indoor Positioning and Indoor Navigation (IPIN2010)*, (Zurich), pp. 1 – 10, IEEE, Sep 2010.

[5] Y. Shen and M. Z. Win, "On the use of multipath geometry for wideband cooperative localization," *In proc. IEEE GLOBECOM '09*, pp. 1–6, Nov 2009.

[6] P. Meissner and K. Witrisal, "Multipath-assisted single-anchor indoor localization in an office environment," in *19th International Conference on Systems, Signals and Image Processing, IWSSIP 2012*, (Vienna), pp. 22–25, IEEE, Apr 2012.

[7] S. Van de Velde, H. Wymeersch, P. Meissner, W. Klaus, and H. Steendam, "Cooperative multipath-aided indoor localization," *In proc. IEEE Wireless Communications and Networking Conference (WCNC) '12*, pp. 3107 –3111, Apr 2012.

[8] S. Gezici, H. Kobayashi, and H. Poor, "Nonparametric nonline-of-sight identification," in *In proc. of Vehicular Technology Conference, 2003. VTC 2003-Fall.*, vol. 4, pp. 2544 – 2548, Oct 2003.

[9] H. Wymeersch, S. Marano, W. M. Gifford, and M. Z. Win, "A machine learning approach to ranging error mitigation for UWB Localization," *IEEE Trans. Commun.*, Jun 2012.

[10] 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.

[11] 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.

[12] J. Gower and G. B. Dijksterhuis, "Procrustes problems," *Oxford University Press*.

[13] 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.

[14] T. Eren, O. Goldenberg, W. Whiteley, Y. Yang, A. Morse, B. Anderson, and P. Belhumeur, "Rigidity, computation, and randomization in network localization," in *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, vol. 4, pp. 2673 – 2684, Mar 2004.