Evaluation of a geometric positioning algorithm for hybrid wireless networks

Nicolas Amiot, Mohamed Laaraiedh, Bernard Uguen

Evaluation of a Geometric Positioning Algorithm for Hybrid Wireless Networks

Nicolas Amiot, Mohamed Laaraiedh, and Bernard Uguen
IETR, University of Rennes 1, France
{nicolas.amiot, mohamed.laaraiedh, bernard.uguen}@univ-rennes1.fr

***Abstract*—In this paper, we propose a geometric positioning method for hybrid wireless networks, based on a set membership method. Three common types of radio observables are considered for the position estimation: range, difference of ranges and received power. This paper details how to build geometric constraints from observables, and how to merge them to estimate the position. Given a realistic scenario, Monte Carlo simulation shows that the performance of the proposed method in terms of root mean squared error and cumulative density functions outperforms that of a numerically optimized maximum likelihood.**

***Index Terms*—Localization, set membership method, interval analysis, hybrid wireless networks, range, difference of ranges, received power, maximum Likelihood.**

I. Introduction

Recently, the number of available radio access techniques (RATs) on a single terminal has drastically increased and thus simplified hybrid positioning. In the same time, the emergence of mechanic sensors embedded in mobile devices has provided a new source of exploitable information. One challenge for positioning applications is to properly merge all those information. In this situation, the standard algorithms based on least square or maximization of a likelihood function usually used to perform the position estimation have two major drawbacks: They are not convenient for using both radio and non-radio observables, and their linearization process makes difficult to approach non convex regions. This last limitation is especially an issue in positioning problems where non convex regions are often encountered. To cope with this problem, algorithms based on a geometric approach as set-membership and interval analysis have recently brought a solution [1].

Contrary to classical algebraic methods, position estimation based on geometric method don't return a single position estimate but a set of intervals which contain the sought position. Recently used for addressing the problem of outdoor positioning, the geometrical approach has allowed to merge both GNSS observables and inertial sensors [2]. The achieved positioning accuracy and the limited computation complexity have demonstrated the great interest of the method. As well, those geometric methods have advantageously show their performance for positioning in wireless sensor networks using range observables [3] or received power observables [4].

Previous examples show that geometric positioning algorithms have always been envisaged with a single type of radio observable where a non-radio information can be added. To the best of our knowledge, this is the first attempt to apply a geometric method for hybrid positioning.

In this paper, the proposed geometric positioning algorithm for hybrid wireless networks is presented. The proposed algorithm allows to include the three most common radio observables: range, difference of ranges and observed power. The positioning accuracy is evaluated using Monte Carlo simulation and shows that the proposed method outperforms numerically optimized ML functions, for a given realistic scenario.

II. Assumed Scenario

The hybrid scenario of interest is described in Fig. 1. The blind node estimates its position β with the help of anchors providing three types of radio observables. The anchors at positions $\{A_P\}$ provide received power observations $\{P\}$, the anchors at positions $\{A_D\}$ provide difference of ranges observations $\{\Delta\}$ and the anchors at positions $\{A_R\}$ provide range observations $\{r\}$. These three types of anchors are drawn on the edges of three different squares, thus : $\{A_P\} \in \mathcal{H}_P$, $\{A_D\} \in \mathcal{H}_D$, $\{A_R\} \in \mathcal{H}_R$ with $\mathcal{H}_R \subset \mathcal{H}_D \subset \mathcal{H}_P$. The blind node is assumed to search its position in $\mathcal{H}_R$.

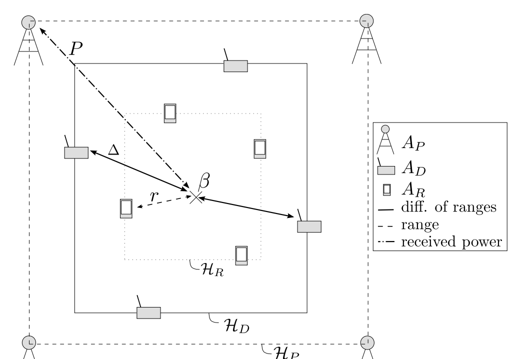


Fig. 1: A blind node at position β receives range observations $\{r\}$ from anchors at positions $\{A_R\}$, difference of ranges observations $\{\Delta\}$ from anchors at positions $\{A_D\}$ and received power observations $\{P\}$ from anchors at positions $\{A_P\}$.

III. Constraints Description

The proposed geometric algorithm resolves the positioning problem by finding the region where all constraints are satisfied. In the following, a constraint designates both a simple mathematical expression which bounds a finite or infinite region of space, and its geometric representation. Note also that a constraint can be obtained from observables from very different nature such as a layout of a building [5], inertial motion data [2], or radio observables. In the following only radio observables are considered. Three types of constraint are built: the range constraint, the difference of ranges constraint and the power constraint.

A. Range Constraint

A range constraint is defined from a range observation r evaluated between an anchor at position A_R and the blind node at position β as such:

$$r = \|A_R - \beta\| + \delta_r, \tag{1}$$

where δ_r is the error in the range estimate. Given a probability model for δ_r, it is possible to determine a confidence interval for the range estimate r which yields a confidence region shaped as an annulus in two dimensions (2D) or a shell in three dimensions (3D), with center A_R.

B. Difference of Ranges Constraint

Given a difference of ranges Δ from anchors at position A_D and $A_{D'}$ to the blind node at position β, a difference of range constraint can be expressed as:

$$\Delta = \|A_D - \beta\| - \|A_{D'} - \beta\| + \delta_\Delta, \tag{2}$$

where δ_Δ is the error in the difference of ranges estimate. Given a probability model for δ_Δ, it is possible to determine a confidence interval for Δ which yields a confidence region contained between two hyperbolas in 2D or two hyperboloids in 3D with their common focal point $\bar{D}$:

$$\bar{D} = \begin{cases} D & \text{if } \Delta > 0 \\ D' & \text{if } \Delta < 0 \end{cases} \tag{3}$$

C. Power Constraint

Building spatial constraints requires to formalize mathematically a distance information. Hence, to build the power constraint we propose to model the log received power observation according to the standard path loss model :

$$P = P_0 - 10 n_p \log_{10}(d), \tag{4}$$

where P_0 is the power received at 1 meter and n_p is the path loss exponent. Thus, according to [6], the distance can be estimated as:

$$d = \exp(M - S^2) + \delta_P \tag{5}$$

with δ_P the error in distance estimate, and $M = \frac{\log(10)(P_0 - P)}{10 n_p}$ and $S = -\frac{\log(10)\sigma_X}{10 n_p}$, where σ_X^2 is the variance of the received power observation perturbation. Once this distance is obtained and given a probability model for δ_P it is possible to determine a confidence interval for the distance d which yields a confidence region shaped as an annulus in 2D or a shell in 3D with center A_P. Practically, the log received power information can be obtained from the received signal strength indicators.

D. Confidence Interval Determination

Due to the error in the observations, the constraint is associated to a limited region of the space. The extension of this region is proportional to the confidence interval chosen for the probability models of the considered observation. Practically, the probability model chosen for δ_r, δ_Δ and δ_P assumed to be zero mean Gaussian with σ_r^2, σ_Δ^2 and σ_P^2 their variances respectively. Thus, we can build a constraint interval $[I]$, and in particular, $[I_r]$, $[I_\Delta]$ $[I_P]$, the constraint interval of the range, of the difference of ranges constraint and of the power constraint respectively:

$$[I_r] = \quad [r - \gamma\sigma_r, r + \gamma\sigma_r] \tag{6}$$
$$[I_\Delta] = \quad [\Delta - \gamma\sigma_\Delta, \Delta + \gamma\sigma_\Delta] \tag{7}$$
$$[I_P] = \quad [d - \gamma\sigma_P, d + \gamma\sigma_P] \tag{8}$$
$$\tag{9}$$

with γ, an adjustment factor. Without prior information we first set $\gamma = 3$ to ensure a 99% confidence interval for all the constraints.

IV. Geometric Algorithm Description

Table I summarizes the 5 steps of the geometric algorithm. The detailed operation of this algorithm is presented step by step in this Section.

Table I: The Proposed Geometric Method: Algorithm Description

1) Build the constraints (Fig. 2),
2) Box the constraints (Fig. 3),
3) Merge the constraints to obtain a merged box (Fig. 4),
4) Approximate the merged box with a Kd-Tree algorithm to obtain an approximated region (Fig. 5),
5) Estimate the position from the approximated region (Fig. 6).

A. Build the Constraints

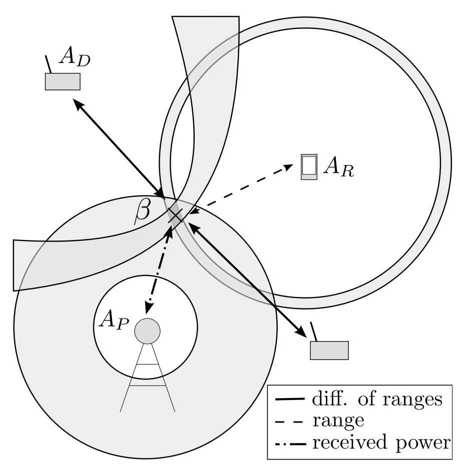


Fig. 2: The three constraints are built thanks to three different types of radio observables.

Fig. 2 shows the three types of radio constraints built from the radio observables as described in Section III.

B. Box the Constraints

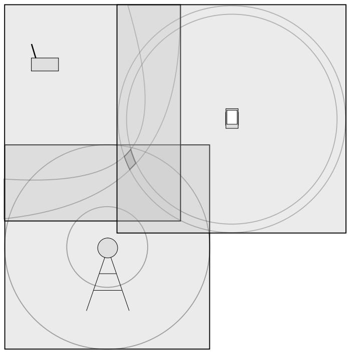


Fig. 3: The three constraints are boxed.

Then, for the ease of computation, the constraints need to be boxed as shown in Fig. 3. Boxing a constraint consists in projecting the constraint interval on each axis in order to obtain a box B defined as [7]:

$$B = [\mathbf{I}] = [I_1] \times [I_2] \times \ldots \times [I_k], \tag{10}$$

$$\text{with } [I_k] = proj_k([I]),$$

where k is the axis dimension, set at $k = 2$ for a 2D problem, and $proj_k([I])$ returns the projection of the interval $[I]$ on axis k. In order to find the smallest constraints intersection B_m, the adjustment factor value γ of each constraint is reduced until $\min_\gamma(B_m)$. Algorithm 1 describes the procedure.

Algorithm 1 iterative interval reduction

γ=3: to ensure a 99% confidence interval
$B_m = \bigcap_n^N B_n$: compute the box intersection of all constraints
while $B_m \neq \emptyset$ **do**
  $\gamma = \gamma - \alpha$: reducing γ and the confidence interval with $\alpha \in \mathbb{R}^+$
  $B_{ms} = B_m$: save the previous value of B_m
  $B_m = \bigcap_n^N B_n$: compute the box intersection of all constraints
end while
$B_m = B_{ms}$: keep the smallest non void intersection of constraints.

C. Merge the Constraints

Once all constraint boxes have been obtained, they are merged as illustrated in Fig. 4. This merging step allows to obtain a box which, if one assumes the absence of biases, necessarily encloses the blind node position. Practically, this merged box B_m is obtained by finding the intersection of all the boxed constraints B_n of constraint n:

$$B_m = \bigcap_n^N B_n \tag{11}$$

where N is the total number of constraints.

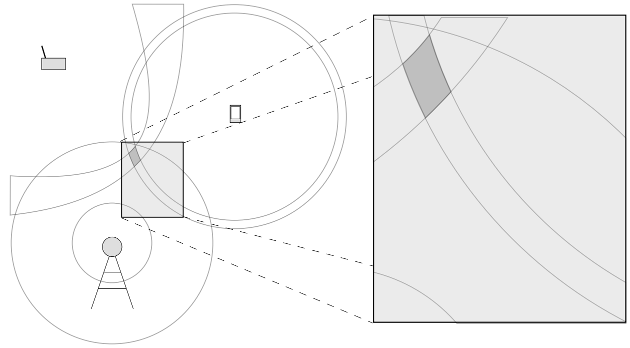
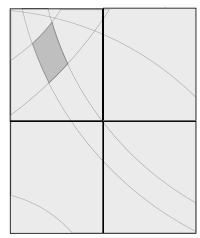
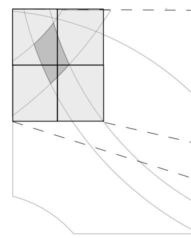


Fig. 4: The three boxed constraints are merged. It results in a merged box which contains the true blind node position.

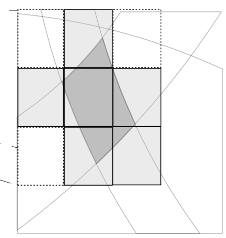
D. Approximate the Merged Box



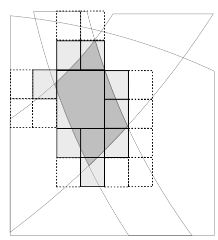
(a) First quadtree iteration on merged box,



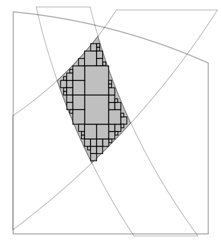
(b) Upper left box is solely remained,



(c) Zoom on second quadtree iteration. Doted boxes are removed,



(d) Third quadtree iteration,



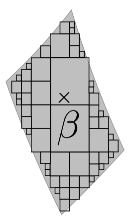
(e) Final approximated region is composed of a cluster of boxes.

Fig. 5: The quadtree approximation allows to enclose the blind node position.

Once the smallest merged box has been obtained, the blind node position needs to be approximated as shown in Fig. 5. For that purpose, this merged box is approximated by a recursive Kd-Tree algorithm (a.k.a. quadtree algorithm in 2D, or octree algorithm in 3D). The Kd-Tree algorithm puts a single box at the input and returns 2^n sets of boxes $\{B\}$, where n is the space dimension. Practically, this partitioning method consists in splitting all intervals of a box into two complementary intervals for each dimension of the box. Each box returned by the Kd-Tree algorithm is intersected with each boxed constraint. If the intersection is not void, the box is candidate for a new Kd-Tree iteration, otherwise the box is removed. This whole process is repeated until at least μ sets of enclosed boxes $\{B_e\}$ are obtained, whereupon the process is stopped. Algorithm 2 describes the complete procedure of the implemented Kd-Tree algorithm.

Algorithm 2 interval approximation by boxes

$\{B_e\} = \{B_m\}$: initialization with the merged box B_m
while card($\{B\}$) $<= \mu$ **do**
  for b in $\{B\}$ **do**
    $\{Q\} = KdTree(b)$: for each box b from the merged box B_m, apply a Kd-Tree
    for q in $\{Q\}$ **do**
      if $q \cap \forall B_n$ **then**
        $\{B\} = \{B\} + q$: the box q obtained from the Kd-Tree is remained if q intersect all constraints boxes
      end if
    end for
  end for
end while



(a) Example of position estimation in the approximated region

(b) Screenshot from demonstrator of the approximated region, the true position (green ball) and estimated position (black ball)

Fig. 6: The position is estimated by computing the center of mass of all boxes.

E. Estimate the Position

Fig. 6 illustrates the position estimation step. It consists in estimating the true position from the center of mass of the set of enclosing boxes $\{B_e\}$. Fig. 6b is a screenshot of an estimated blind node position using our demonstrator. Note that in case where the set of enclosing boxes $\{B_e\}$ are disjoints, the position estimate would take advantage of an advanced estimation procedure based on hypothesis testing decision as described in [8].

V. Results and Discussions

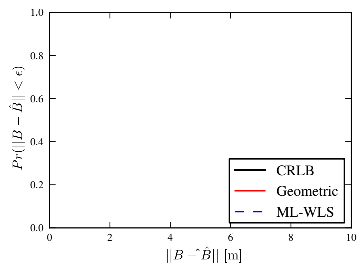
A. Simulation setup

The performance of the proposed geometric method is compared to a ML approximation and to the Cramer-Rao lower bound (CRLB) via Monte Carlo simulation based on the realistic scenario described in Section II. The ML approximation uses a Nelder-Mead simplex optimizer initialized with a weighted least square solution (ML-WLS) [9]. Multidimensional likelihood functions corresponding to the given scenarios are described in [10]. As mentioned in Section III-D, the perturbation of the observation of the range constraint, the difference of range constraint and the power constraint are supposed zero mean Gaussian, with their respective variances σ_r^2, σ_Δ^2 and σ_P^2. The parameter settings in Table II have been chosen compliant with the WHERE2 measurement campaign [11]. Finally, the last version of the entire framework used to perform those simulations can be obtained on the github website https://github.com/niamiot/RGPA.

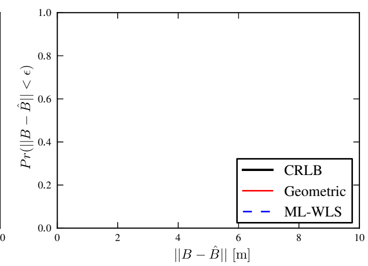
Table II: Parameters settings

Parameter	Value	
$\mathcal{H}_P$	$[-1,1] \times [-1,1]$	km^2
$\mathcal{H}_D$	$[-100,100] \times [-100,100]$	m^2
$\mathcal{H}_R$	$[-10,10] \times [-10,10]$	m^2
σ_r	2.97	m
σ_Δ	3.55	m
σ_X	4.34	dB
n_p	2.64	
P_0	-40	dB

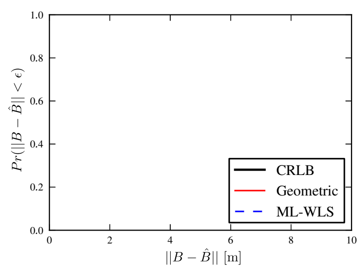
B. Comparison of Performances



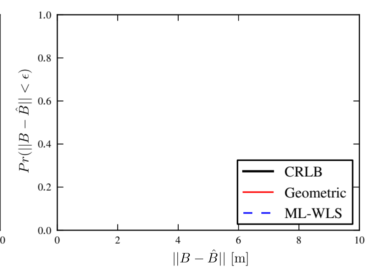
(a) Powers + diff. of ranges



(b) Powers + ranges



(c) Ranges + diff. of ranges



(d) Full Hybrid

Fig. 7: CDFs of positioning error using the proposed geometric method, ML-WLS, and CRLB applied on hybrid positioning technique.

We compare the performances of the three algorithms in terms of cumulative density function (CDF), root mean square error (RMSE) and computation speed for the hybrid cases for four hybrid configurations:

- Powers + difference of ranges, using 4 received power and 3 difference of ranges observables (Fig. 7a),
- Powers + ranges, using 4 received power and 4 ranges observables (Fig. 7b),
- Ranges + difference of ranges, using 4 ranges and 3 difference of ranges observables (Fig. 7c),
- Full hybrid, using 4 received power, 4 ranges and 3 difference of ranges observables (Fig. 7d).

The non-hybrid cases using a unique type of observation (only range, difference of ranges, or received power) are not considered here. From the empirical CDF shown in Fig. 7 (a-d) it appears that the proposed geometric method prevails on

Table III: RMSE vs Method

Hybrid mode	Geometric (m)	ML-WLS (m)	CRLB (m)
Power + diff. of ranges	2.46	2.91	2.23
Power + ranges	2.51	2.81	1.78
Ranges + diff. of ranges	1.68	1.93	0.96
Full Hybrid	1.65	1.92	0.95

ML-WLS. This increased accuracy of positioning is especially significant on Fig.7a and Fig.7b. Those two cases using received power observables allow a $1\ \mathrm{m}$ gain for all blind nodes. Other cases based only on time based observables as shown in Fig. 7c, or using all type of observables as shown in Fig. 7d, also show a better accuracy in terms of position estimation. Those results are confirmed by the RMSE values shown in Table III. The most significant improvement is observed for the hybrid scheme mixing powers and ranges. In average, the proposed geometric method ensures a $30\ \mathrm{cm}$ increase of positioning accuracy for blind nodes drawn in a $20 \times 20\ \mathrm{m}^2$ room.

Obviously these improvements come out at the cost of extra computation complexity. In spite of providing a complete complexity study, Fig. 8 shows some preliminary results based on an average of computation speed for each method. On those histograms, it can be observed that the proposed method is generally slower than the ML-WLS excepted when the received power observables and range observables are used. It also shows that the difference of ranges constraint is the worst in term of speed. A further investigation would be to improve the speed of the difference of ranges constraint. Moreover, the comparison between both methods is unfair, because the ML-WLS numerical optimization is based on an optimized compiled Fortran code, whereas the proposed geometric method is based on an interpreted code in Python. Considering that difference of implementation, a geometrical method as fast as the ML-WLS could be feasible. Moreover, the geometrical method is highly parallel and involves only elementary operations and could probably be very efficiently implemented in dedicated hardware.

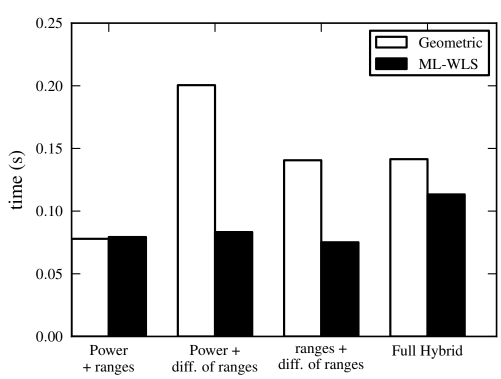


Fig. 8: Speed computation comparison between the proposed geometric method and a ML-WLS using a numerical optimizer.

VI. Conclusion

This paper has presented and evaluated a geometrical method for the positioning problem in case of hybrid observables. We have considered three cases of radio observables: range, difference of ranges and received power. Those radio observables are used to build constraints, which are merged to obtain a position estimate. Monte Carlo simulation have been computed in a realistic hybrid scenario. This simulation shows that the performance of the proposed geometrical method in terms of RMSE and CDF globally outperforms numerically optimized ML functions. In average, a $30\ \mathrm{cm}$ improvement of position accuracy has been observed for a blind node drawn randomly in a $20 \times 20\ \mathrm{m}^2$ room. As well, compared to the Nelder-Mead simplex optimized ML, the method shows promising results in term of computation speed, considering the current stage of its development. Our current work consists in the development of a dynamic and cooperative version of the algorithm. We also investigate the impact of additional non-radio constraints on the position estimation.

Acknowledgment

The work presented in this paper has been performed in the framework of the FP7 project ICT-248894 WHERE2 (Wireless Hybrid Enhanced Mobile Radio Estimators - Phase 2) which is funded by the European Union.

References

[1] L. Jaulin, "A nonlinear set membership approach for the localization and map building of underwater robots," *Robotics, IEEE Transactions on*, vol. 25, pp. 88–98, feb. 2009.

[2] V. Drevelle and P. Bonnifait, "High integrity GNSS location zone characterization using interval analysis," *ION GNSSSavannah, GA : United States*, 2009.

[3] F. Mourad, H. Snoussi, M. Kieffer, and C. Richard, "Robust interval-based localization algorithms for mobile sensor networks.," *IJDSN*, vol. 2012, 2012.

[4] J. Leger and M. Kieffer, "Guaranteed robust distributed estimation in a network of sensors," in *Acoustics Speech and Signal Processing, ICASSP, 2010 IEEE International Conference on*, pp. 3378–3381, march 2010.

[5] M. Raspopoulos, B. Denis, M. Laaraiedh, J. Domingez, L. de Clelis, D. Slock, G. Agapiou, J. Stephan, and S. Stavrou, "Location-dependent information extraction forpositioning," *ICL-GNSS conference*, 2012.

[6] M. Laaraiedh, S. Avrillon, and B. Uguen, "Hybrid data fusion techniques for localization in UWB networks," in *Positioning, Navigation and Communication, 2009. WPNC 2009. 6th Workshop on*, pp. 51–57, march 2009.

[7] L. Jaulin, M. Kieffer, O. Didrit, and E. Walter, *Applied Interval Analysis*. Springer, 2001.

[8] N. Amiot, T. Pedersen, M. Laaraiedh, and B. Uguen, "A hybrid positioning method based on hypothesis testing," *Wireless Communications Letters, IEEE*, vol. PP, no. 99, pp. 1–4, 2012.

[9] M. Laaraiedh, S. Avrillon, and B. Uguen, "Enhancing positioning accuracy through direct position estimators based on hybrid RSS data fusion," *IEEE VTC Spring*, 2009.

[10] M. Laaraiedh, *Contribution on Hybrid Localization Techniques For Heterogeneous Wireless Networks*. PhD thesis, University of Rennes 1, Dec. 2010.

[11] ICT-WHERE-Project, "Deliverable 4.1: Measurements of location-dependent channel features," tech. rep., October 2008.