



**HAL**  
open science

# A multihypothesis set approach for mobile robot localization using heterogeneous measurements provided by the Internet of Things

Etienne Colle, Simon Galerne

## ► To cite this version:

Etienne Colle, Simon Galerne. A multihypothesis set approach for mobile robot localization using heterogeneous measurements provided by the Internet of Things. *Robotics and Autonomous Systems*, 2017, 96, pp.102–113. 10.1016/j.robot.2017.05.016 . hal-01621289

**HAL Id: hal-01621289**

**<https://hal.science/hal-01621289v1>**

Submitted on 23 Oct 2017

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# A multihypothesis set approach for mobile robot localization using heterogeneous measurements provided by the Internet of Things

Etienne Colle and Simon Galerne  
 IBISC, University of Evry, France  
[Etienne.Colle](mailto:Etienne.Colle@ibisc.univ-evry.fr) | [Simon.Galerie@ibisc.univ-evry.fr](mailto:Simon.Galerie@ibisc.univ-evry.fr)  
 Corresponding author [Etienne.Colle@ibisc.univ-evry.fr](mailto:Etienne.Colle@ibisc.univ-evry.fr)

## Abstract

Mobile robot localization consists in estimation of robot pose by using real-time measurements. The Internet of Things (IoT) adds a new dimension to this process by enabling communications with smart objects at anytime and anywhere. Thus data used by localization process can come both from the robot on-board sensors and from environment objects, mobile or not, able to sense the robot. The paper considers localization problem as a nonlinear bounded-error estimation of the state vector whose components are the robot coordinates. The approach based on interval analysis is able to answer the constraints of IoT by easily taking account a heterogeneous set and a variable number of measurements. Bounded-error state estimation can be an alternative to other approaches, notably particle filtering which is sensible to non-consistent measures, large measure errors, and drift of robot evolution model. The theoretical formulation of the set-membership approach and the application to the estimation of the robot localization are addressed first. In order to meet more realistic conditions the way of reducing the effect of environment model inaccuracies, evolution model drift, outliers and disruptive events such as robot kidnapping is introduced. By integrating these additional treatments to the set-membership approach we propose a bounded-error estimator using multihypothesis tracking. Simulation results show the contribution of each step of the estimator. Real experiments focus on global localization and specific treatments for synchronizing measurements and processing outliers.

*Keywords: Robot mobile localization · Interval analysis · Bounded-error estimator · Outlier, Model inaccuracy and drift · Multi-hypothesis tracking.*

## 1. Introduction

Localization of mobile robots estimates the position and the orientation of the vehicle (pose) related to a reference frame. Two different approaches have been distinguished. Local localization or pose tracking provides a new pose given a previous pose and proprioceptive information. However an accurate initial pose is needed. Global localization is designed to provide the robot pose without any a priori, given exteroceptive observations. Global localization is used for initializing the pose estimate of the robot at the beginning of the localization process or for solving the lost robot problem. Several localization strategies have been proposed combining proprioceptive and exteroceptive information. However the Internet of Things (IoT) has added new dimensions by enabling communications with and among smart objects, thus leading to the vision of “anytime, anywhere, anymedia, anything” [1]. So strategies have to adapt themselves in order to take into account the new context of IoT.

In current localization approach, strategies which track a single hypothesis or multiple hypotheses about robot pose are generally considered as a state estimation problem [2]. Extended Kalman Filter (EKF) has been early investigated for solving localization problem [3, 4, 5, 6]. However, as in many cases real-world disturbances do not satisfy statistical assumptions, the EKF is not guaranteed to converge. In addition, the tuning of parameters can be difficult and EKF needs an accurate initialization. Tracking multiple hypotheses by multiple Kalman filters allows significant improvement but at the expense of increased complexity [7]. EKF drawbacks have led to the development of other filters. The particle filtering (PF) is one of the most effective localization algorithms [8, 9]. PF presents advantage that non-Gaussian distributions and non-linear models can be incorporated. PF is known for its robustness and the ease of implementation if the state dimension is low, but might also be difficult to tune. In fact the efficiency of the filter depends mostly on the number of particles and on the way to re-allocate the weights of particles. The number of particles results of a compromise between robustness, accuracy and computing time. For Gustafson [9] who has surveyed PF theory and practice for positioning applications, PF is not practically useful when extending the models to more realistic cases. Three main cases are mentioned: i) High-dimensional state-space models, typically motion in more than three-dimensions space (six-dimensional pose), ii) More dynamic states (accelerations, unmeasured velocities), iii) Sensor bias and model drift. The Marginalized Particle Filter (MPF) brings an answer to the case of high-dimensional state-space models as long

as the nonlinearities only affect a small subset of the state. PF are also sensible to non-consistent measures, large measure errors [10] and model drift. Experiments point out that PF can locally converge towards a wrong solution due to erroneous measurements. Similar experiments with an EKF show the same phenomenon. EKF strongly underestimates its covariance matrix in presence of repeated biased measurements.

An alternative to PF could be the bounded-error state estimation. Experiments have demonstrated that the approach is operational on vehicle navigating in an indoor environment or in an outdoor environment [11]. In this approach, known as set-membership or set-theoretic estimation, all model and measurement errors are assumed to be bounded, with known bounds. The set-membership approach represents the solution of a problem by a closed set in which the real solution is guaranteed to be, in the sense that no solution consistent with the data and the hypotheses can be missed. Sabater *et al* [12] is one of the first papers in which set-theoretic approach has been applied to robotic localization for building a geometric representation of the environment. Set-membership localization based on angle measurements has been tackled by [13, 14] and based on telemetric data by [15]. Ceccarelli *et al.* [16] conclude that simulations and experimental tests show that the set-theoretic approach represents a valuable alternative to statistical SLAM whenever the bounded errors assumption holds.

We propose an approach based on set-membership estimator for determining the pose of a mobile robot or any mobile object in the context of the Internet of Things. Consideration of realistic conditions of use requires an approach capable of solving many constraints. In our scheme some of the constraints are solved by estimators, the other by complementary treatments.

The principle is as follows. Each Estimator predicts a hypothesis of robot pose and corrects it as soon as new measurement is available. Set-membership methods consider the inaccuracies of measurements and of environment model as bounded. Complementary treatments take into account specific cases:

- Global localization for initializing or re-initializing pose hypotheses
- Synchronisation of asynchronous measurements
- Detection and processing of disruptive events
  - o Presence of outliers among measurements
  - o Robot kidnapping which corresponds to a sudden change of robot pose
  - o Drift and inaccuracy of the robot evolution model
- Multihypothesis tracking

IoT adds as specificities a high heterogeneity of sensors and a variable density of observations depending on areas crossed by the robot, more or less poor in terms of sensor density.

Paper is organized as follow. Section 2 introduces the theoretical formulation of set-membership approach. Section 2 ends by the description of the components of the set-membership estimator and particularly those for processing inconsistent measurements and adapting to the context. Section 3 presents the multihypothesis Bounded-Error Estimator (BEE) able to handle outliers, asynchronous measurements, environment model inaccuracies and disruptive events. Section 4 illustrates the process of localization by BEE step by step from simulated data. Sect. 5 shows experimental results of main treatments: the global localization, the measurement synchronization and outlier processing.

## 2. Set approach for robot localization

In set membership approach, solution is generally of any form. Various set-theoretic methods have been developed, each based on a specific representation of measurements and solution: ellipsoid, polytope, parallelotope or zonotope. We have chosen a representation of variables by interval. Interval analysis allows an easier implementation and a real time computing [17].

### 2.1 Set inversion for estimating parameters

Interval analysis is based on the idea of enclosing real numbers in intervals and real vectors in boxes. The analysis by intervals consists in representing the real or integer numbers by intervals which contain them. This idea allowed algorithms whose results are guaranteed, for example for solving a set of non-linear equations [18, 19, 20].

An interval  $[x]$  is a set of  $IR$  which denotes the set of real interval

$$[x] = \{x \in IR \mid x^- \leq x \leq x^+, x^- \in IR, x^+ \in IR\} \quad (1)$$

$x^-$  and  $x^+$  are respectively the lower and upper bounds of  $[x]$ . The classical real arithmetic operations can be extended to intervals. Elementary functions also can be extended to intervals. Given  $f: IR \rightarrow IR$ , such as  $f \in \{cos, sin, arctan, sqr, sqrt, log, exp \dots\}$ , its interval inclusion  $[f]([x])$  is defined on the interval  $[x]$  as follow

$$[x] \rightarrow [f]([x]) = \{f(x) \mid x \in [x]\} \quad (2)$$

Colle and Galerne [21] presents a set-membership method which easily integrates a heterogeneous set of measurements provided by a large variety of sensors, from the roughest to the most complex one such as goniometric and range sensors, tactile tile with complex shape and dead reckoning measurements. Measurement and environment inaccuracies are taken into account. Simulation results as well as real experiments have shown the interest of the method in a cooperative environment context.

If the bounded-error assumption is verified, the result is guaranteed. In fact, little research has investigated the violation of bounded-error assumption for vehicle localization in membership approaches. The respect of the assumption that all the errors are bounded is difficult to prove but there are techniques to reject outliers [18, 19]. Lambert et al. [10] present a Bounded-Error State Estimation (BESE) for the localization of outdoor vehicle. Authors show that BESE approach is able to solve the localization problem with a better consistency than Bayesian approach such as particle filters in presence of outliers. Drevelle et al. [20] focused on the robustness of set-membership approach in presence of outliers for multi-sensory localization. However in all papers outlier rejection is done by relaxing some constraints, which can lead to a solution composed of non-connected sets. Each set corresponds to a pose hypothesis which is represented by a 3-dimension box  $([x_R], [y_R], [\theta_R])$  in interval analysis formalism.

## 2.2 Components of the set-membership estimator

### 2.2.1 Principle of the set-membership estimator

Bounded-error estimation we propose is based on classical predictor-corrector filters which operate as follows. Pose hypotheses are updated using proprioceptive data applied to the evolution model of the robot. Over time these predicted hypotheses see their credibility vary depending on new observations. This classical scheme is completed in order to consider realistic conditions: localization initialization, data synchronization, outliers processing, model inaccuracy processing. Initialization is required at the beginning of the localization process, and re-initialization whenever the robot is lost. Internet of Things provides asynchronous information and notably measurements which has to be synchronized before localization estimation. At last the method considers possible the occurrence of disruptive events: the presence of outliers, the kidnapping of the robot which corresponds to a sudden change of pose, the inaccuracy of the environment model, and the drift and the inaccuracy of the robot evolution model. These events lead to a solution which could be false or reduced and even empty. In all these cases the result is no more guaranteed. Additional treatments have been added to moderate the effects of these events.

### 2.2.2 Localization initialization and re-initialization

Colle and Galerne [21] have combined forward-backward contraction with the SIVIA bisection algorithm [22] in order to take advantage of both technics. The aim of propagation techniques is to contract as much as possible the domain of the variables without losing any solution [23]. Forward-backward propagation selects the primitive constraints to be used for optimally contracting the size of the domains. It is based on the Waltz algorithm [24]. The contractor reduces the domain of each variable by processing each constraint one by one. This contractor is faster than set inversion done by SIVIA but does not always reach the optimal solution. We propose to combine the contraction to the bisection. When no more significant contraction can be performed the solution box is bisected and the same process is repeated on both sub-boxes. The forward-backward contraction and bisection (FBCB) is stopped as soon as the localization accuracy no more improves, contrarily to SIVIA which ends when reaching the required accuracy  $\epsilon$ . The result of FBCB is a 3-dimension box  $([x_R], [y_R], [\theta_R])$ .

### 2.2.3 Dead reckoning and measurement Synchronization

Pose tracking provides a new estimate, given a previous pose and proprioceptive data of dead reckoning.  $\Delta x_i$ ,  $\Delta y_i$ ,  $\Delta \theta_i$  are computed from the kinetic model of the mobile robot and the wheel speeds. The model is  $x_{Rn} = x_{Rn-1} + \Delta x_n$ ,  $y_{Rn} = y_{Rn-1} + \Delta y_n$ ,  $\theta_{Rn} = \theta_{Rn-1} + \Delta \theta_n$  at times  $n$  and  $n-1$  and the inclusion test is  $[x_n] \subset [x_{n-1}] + [\Delta x_n]$ . Proprioceptive information can be used in another way for data synchronization. Let the measurement equation  $[\lambda_{n-1}] = [f]([x_{n-1}])$  where  $\lambda_{n-1}$  is the measurement acquired at time  $n-1$  and  $x_{n-1}$  the state vector at time  $n-1$ . The measurement  $\lambda_{n-1}$  is synchronized by using  $[\Delta x_{n-1}]$  the robot displacement given by dead reckoning between times  $n-1$  and  $n$ ,  $[\lambda_n] = [f]([x_{n-1}], [\Delta x_{n-1}])$ .

### 2.2.4 Inaccuracy of environment model

Up to now we assumed that the coordinates of IoT were precisely known. In fact home sensors, RFID tags and mobile devices are approximately located in the reference frame but with a known inaccuracy. It is easy to consider the inaccuracy by replacing in the measurement equations, the scalar value of each coordinate  $x_i$  by an interval.  $[x_i]$ . FBCB algorithm is applied to measurement equations in which device coordinates are replaced by intervals. In this case, the algorithm reduces both the robot space vector  $([x_R], [y_R], [\theta_R])$  and the environment

model inaccuracies  $[x_i]$ . This is inherent to the process which decreases all interval domains of each variable, input variable as well as output variable.

### 2.2.5 Disruptive events

As said before disruptive events lead to a result which is no more guaranteed without adding additional treatments. During a predictor-corrector step disruptive events can be assumed whenever the intersection between predicted hypothesis and current measurement becomes empty. It is due to the fact that predicted hypothesis or measurement or both are erroneous. A new hypothesis has to be found by global localization (see sect. 2.2.2). Predicted hypothesis becomes wrong either because of robot kidnapping or during updating step. The latter evaluates the predicted pose by applying proprioceptive data to the robot evolution model. Both evolution model and proprioceptive data are inaccurate so dead reckoning is subject to cumulative errors. The second reason which leads to an empty intersection is outliers. Outlier processing requires a particular attention.

## 2.3 Outlier processing

An outlier is an observation which is distant from other observations. Determining whether or not a measurement is an outlier is not easy and generally context dependent. We consider that all sensors, not only robot but also IoT sensors, can generate outliers. The question is how make an estimator robust to outliers taking into consideration the context. In case of violation of the assumption of bounded-errors the set-membership approach no more guarantees the solution of robot localization. This is the case if the bounds of the interval associated to the measurement are ill-chosen or if a defective sensor provides an aberrant measurement. A simple way for setting bounds is to assign to a measure  $\lambda_i$  an interval  $[\lambda_i] = [\lambda_i - \Delta\lambda_i, \lambda_i + \Delta\lambda_i]$  where  $\Delta\lambda_i$  is defined according to the tolerance given by the sensor manufacturer. Nevertheless better knowledge on measurement process is better bound determination will be. Drevelle et al. [25] takes error model and risk into account. The choice of bounds associated with each measurement is accompanied by a risk. Given a prior measurement error distribution, the probability that the correct result be inside the solution set can be estimated. The method needs a prior measurement error distribution and is only applied to a homogeneous set of GPS measurements. A way for processing outliers consists in assuming that the set of measurements includes a given number of outliers. For avoiding that the intersection of measurements available at a given time be empty a small number of constraints, as small as possible, has to be relaxed. The q-relaxed intersection (see equation 3) computes the subset of the solution consistent with at least m-q measurements, where m is the number of measurements and q the number of outliers. In addition, in some cases an outlier can be identified. Indeed, a measurements that do not intersect the q-relaxed intersection could be suspected to be outlier.

$$\bigcap_{k \in \{1, \dots, n\}}^{[q]} [x_k] \quad (3)$$

where n is the number of available measurements.

The q-relaxed intersection makes robust the set membership approach in presence of inconsistent measurements. An inconsistent measurement is not deleted but its weight depends on its degree of inconsistency. The main weakness of constraint relaxation is that it generates a less precise and potentially discontinuous solution. In addition, determining the number q of supposed outliers presents a difficulty. Jaulin et al [26] proposed GOMNE (Guaranteed Outlier Minimal Number Estimator), an algorithm for minimizing the number  $q_{\min}$  of data considered as outliers. The principle consists in applying the q-relaxed algorithm to a set of m measurements, beginning with  $q = 0$ , and increasing q until a non-empty intersection is found. The algorithm gives the minimal number q of data inconsistent with the others. If there is no inconsistency GOMNE fails to detect outliers. A resort solution consists in incrementing  $q_{\min}$  in order to guard against the risk with a sufficient safety margin. Another drawback is the processing time as q-relaxed algorithm is run  $(q + 1)$  times.

## 2.4 Context adaptation

### 2.4.1 Multihypothesis tracking

The q-relaxed algorithm can provide a non-contiguous set of 3-dimension boxes. Each box  $[x]_i$  is considered as a hypothesis of the robot pose. Hypotheses are updated using proprioceptive data applied to the evolution model of the robot. A credibility  $C_i$  is associated at each predicted hypothesis  $[x]_{p,i}$ . Over time predicted hypothesis  $[x]_{p,i}$  see its credibility vary depending on new observations. Credibility is managed as follows. The credibility of a new hypothesis is initialized to 1. A new hypothesis is added to the list of predicted hypotheses. At each correction step the credibility of a predicted hypothesis is incremented whenever it is confirmed by measurements and decremented otherwise. When the credibility of a hypothesis becomes zero, the hypothesis is removed from the list. As there are no new measurements, hypotheses are updated using the robot evolution model. Their credibility does not change.

### 2.4.2 Variable number of measurements

The localization algorithm which will be described in section 3 is a multihypothesis tracking based on predictor-corrector filters, one by hypothesis. Updating process creates, keeps, reduces or kills hypotheses after comparing predictions and measurements at each iteration. In the Internet of Things, the robot goes through sensor-poor areas that alternate with sensor-dense areas. So the number of observations obtainable for localization varies during the robot trajectory. In the localization algorithm hypothesis updating is split into three stages which are activated depending on the number  $n$  of observations available at the current iteration. The correction step of filters is performed at each new observation. If there is no new observation predicted hypotheses are updated with proprioceptive data applied to the robot evolution model. If the number of observations is less than  $(3+q)$  predicted hypotheses are corrected by new measurements. If the number of observations is sufficient a global localization is completed using  $q$ -relaxed intersection, similarly if a disruptive event is detected. However in this case if the current number of observations is insufficient, global localization is achieved by adding past measurements after synchronizing.

## 3. Bounded-error estimator (BEE)

### 3.1 BEE Algorithm

Before describing BEE some clarifications have to be done:

- A pose hypothesis is a 3-dimension box  $([x], [y], [\theta])$  noted  $[x]$ .
- The result of FBCB algorithm which is called by algorithm BEE, is a list  $L_{Hc}$  of 3-dimension boxes  $[x]_{c,j}$ ,  $j = 1$  to  $N_{Hp}$  which enclose the pose of the robot. The list contains one element if  $q=0$  and perhaps more if  $q \neq 0$ . The parameters of FBCB ( $[x_0]$ ,  $L$ ,  $q$ ) are the initial box  $[x_0]$  in which the robot is supposed to be, the list  $L$  of  $m$  available measurements and the presumed number of outliers  $q$ .
- $L_m$  is the list of  $m$  synchronized measurements. It is a sliding history of the latest measurements. Past measurements are synchronized with respect to the current time using the corresponding proprioceptive data applied to the robot evolution model.

### Algorithm description

Let

- $[x_0]$ : Initial state space
- $n$ : Number of measurements available at current time  $t$
- $L_c$ : List of  $n$  current measurements
- $m$ : Number of measurements synchronized on current time  $t$
- $L_m$ : List of  $m$  synchronized measurements
- $q$ : Number of supposed outliers
- $[x]_{p,i}$ : Predicted hypothesis  $i$
- $N_{Hp}$ : Number of predicted hypotheses
- $L_{Hp}$ : List of predicted hypotheses
- $C_{p,i}$ : Credibility of predicted hypothesis  $[x]_{p,i}$
- $[x]_{c,j}$ : Current hypothesis  $j$  computed from available measurements
- $N_{Hc}$ : Number of current hypotheses
- $L_{Hc}$ : List of current hypotheses

Algorithm #3 BEE	
1	<b>if</b> $n = 0$ then update predicted hypotheses with proprioceptive data applied to the robot evolution model (see algorithm n° 4)
2	<b>elseif</b> $0 < n < 3+q$ then correct predicted hypotheses by measurements (see algorithm n° 5)
3	<b>else</b> estimate global localization from measurements (see algorithm n° 6)
4	<b>endif</b>

Algorithm #4 Updating predicted hypotheses	
1	<b>for</b> $i = 1$ to $N_{Hp}$
2	$([x]_{p,i})_t = ([x]_{p,i})_{t-1} + [\Delta x]_{t,t-1}$
3	<b>endif</b>

Where  $([x]_{p,i})_t$  and  $([x]_{p,i})_{t-1}$  are respectively the predicted hypothesis  $i$  at  $t$  et  $t-1$ , and  $[\Delta x]_{t,t-1}$  the robot displacement given by dead reckoning between  $t-1$  and  $t$ .

<b>Algorithm # 5 Correction of predicted hypotheses by measurement(s)</b>	
1	counter = 0;
2	<b>for</b> i = 1 to $N_{Hp}$ , $[\mathbf{x}]_c = \text{FBCB}([\mathbf{x}]_{p,i}, L, q=0)$ ;
3	<b>if</b> $([\mathbf{x}]_c \cap [\mathbf{x}]_{p,i} < \text{vol\_min})$
4	counter = counter + 1;
5	$C_{p,i} = C_{p,i} - 1$ ;
6	<b>if</b> $C_{p,i} = 0$ , remove $[\mathbf{x}]_{p,i}$ from the list $L_{Hp}$ ;
7	$N_{Hp} = N_{Hp} - 1$ ;
8	<b>endif</b>
9	<b>else</b>
10	$[\mathbf{x}]_{p,i} = [\mathbf{x}]_c \cap [\mathbf{x}]_{p,i}$ ;
11	$C_{p,i} = C_{p,i} + 1$ ;
12	<b>endif</b>
13	<b>endfor</b>
14	<b>if</b> counter = $N_{Hp}$
15	$L_{Hp} = \text{FBCB}([\mathbf{x}_\theta], L_m, q)$ ;
16	<b>for</b> i = 1 to $N_{Hp}$ , $C_{p,i} = 1$ ;
17	<b>endfor</b>
18	<b>endif</b>

Line 2: The search space is limited to the predicted hypothesis  $[\mathbf{x}]_{p,i}$ . As the number of current measurements is insufficient, the number q of supposed outliers is zero. Only one hypothesis is returned by FBCB, so  $j=1$  and  $[\mathbf{x}]_{c,j}=[\mathbf{x}]_c$ .

Lines 14 to 17: if all the intersections between predicted hypotheses and current measurements are empty, the robot is supposed lost. A global localization is needed. The algorithm FBCB searches in all the space  $[\mathbf{x}_\theta]$ . As the number of measurements is insufficient the latest memorized measurements after synchronizing are used by FBCB. A number q of outliers can be taken into account.

Line 3: Test  $([\mathbf{x}]_c \cap [\mathbf{x}]_{p,i} < \emptyset)$  has been replaced by  $([\mathbf{x}]_c \cap [\mathbf{x}]_{p,i} < \text{vol\_min})$  in order to anticipate event “empty intersection”.

<b>Algorithm # 6 Global localization</b>	
1	$L_{Hc} = \text{FBCB}([\mathbf{x}_\theta], L_c, q)$ ; // Create a new list of current hypotheses
2	<b>for</b> j = 1 to $N_{Hc}$
3	counter = 0;
4	<b>for</b> i = 1 to $N_{Hp}$
5	<b>if</b> $([\mathbf{x}]_c \cap [\mathbf{x}]_{p,i} < \text{vol\_min})$
6	<b>else</b>
7	$[\mathbf{x}]_{p,i} = [\mathbf{x}]_{c,j} \cap [\mathbf{x}]_{p,i}$ ;
8	$C_{p,i} = C_{p,i} + 1$ ;
9	<b>endif</b>
10	<b>endfor</b>
11	<b>if</b> counter = $N_{Hp}$
12	add $[\mathbf{x}]_{p,i}$ to list $L_{Hp}$ ; // Create a new hypothesis
13	$C_{p,i} = 1$ ;
14	<b>endif</b>
15	<b>endfor</b>

Global localization is performed for initializing or re-initializing the process or when no predicted hypothesis is valid. We have assumed that the value of q is known.

### 3.2 Parameter setting

BEE algorithm can be adapted to the context and objectives by making it more or less pessimistic. Parameter setting depends on the knowledge we have about the various models used by the estimator. Table 1 specifies the problem source and which parameter is related to.

Problem sources	Models	Parameters
Inaccuracy of measurement	$[\lambda] = [\lambda - \Delta\lambda, \lambda + \Delta\lambda]$	$\Delta\lambda$
Inaccuracy of environment model	$[\mathbf{x}_T] = [\mathbf{x}_T - \Delta\mathbf{x}_T, \mathbf{x}_T + \Delta\mathbf{x}_T]$	$\Delta\mathbf{x}_T$
Inaccuracy of robot evolution model	$([\mathbf{x}]_{p,i})_t = ([\mathbf{x}]_{p,i})_{t-1} + [\Delta\mathbf{x}]_{t,t-1}$	$[\Delta\mathbf{x}]_{t,t-1}$
Detection of disruptive event	$vol\_min = [\mathbf{x}]_{c,j} \cap [\mathbf{x}]_{p,i}$	$vol\_min$
Number of outliers	q-relaxed intersection	$q$

**Table 1 :** Parameter setting

In table 1,  $\lambda$  is a measurement,  $\mathbf{x}_T$  the coordinates of a device of IoT (sensor or marker) which participates to the localization process, and  $([\mathbf{x}]_{p,i})_t$  and  $([\mathbf{x}]_{p,i})_{t-1}$  the predicted hypothesis  $i$  respectively at time  $t$  et  $t-1$ .

With regard to parameter setting the initial value of the first three parameters of Table 1 generally is given by the device maker ( $\Delta\lambda$  and  $[\Delta\mathbf{x}]_{t,t-1}$ ) and the home automation fitter ( $\Delta\mathbf{x}_T$ ). The last two parameters are tuned after a learning step in which parameters beginning at 0 are progressively increased until the reduction of the problem associated to.

#### 4. Simulation results

Section 4 analyzes the behavior of the localization algorithm in case of disruptive events: inconsistent measurements, drift and inaccuracy of evolution model, robot kidnapping. The idea is to present the action of each sub-treatment and then of the complete algorithm BEE. The algorithm is evaluated with simulated data in order to compare the behavior of the algorithm in known and repeatable conditions.

##### 4.1 Evaluation and criteria

Three types of criteria are used for analyzing behaviors of the algorithm: accuracy, confidence and consistency, and robustness. Localization accuracy might be assessed according to average or instantaneous error criteria. The average error as the mean square error RMSE (Root Mean Squared Error) reflects the overall performance of the estimator [28]. However it is also interesting to measure the accuracy on a temporal horizon to consider the behavior of the estimator in borderline cases. Instantaneous precision criteria usually are Euclidean error, and position and orientation errors.

An estimator must provide a given uncertainty around the estimate. It is the value of this uncertainty will determine the confidence and the consistency attributed to the estimator. Generally confidence and consistency are evaluated using probabilistic criteria as NCI (No Credibility Index), NEES (Normalized Squared Error Estimation) or NIS (Normalized Innovation Squared) [29]. They may also be studied using the uncertainty bands  $2\sigma$  or  $3\sigma$  or with the uncertainty ellipse describing the same standard deviations  $2\sigma$  and  $3\sigma$  (Lefebvre et al 2004). In this work, by analogy, a band is defined by the upper and lower bounds of the state estimate minus the corresponding reference state, one band by dimension of the pose hypothesis. Bands will be respectively applied to the instantaneous position and orientation errors. The consistency of the estimator is verified if the zero value is always included into the band. It means that the true value is included in the estimation. Confidence is given by the width of the band. The parameter setting of an estimator is the result of a compromise between confidence, consistency and accuracy. Robustness is studied in borderline cases by means of instantaneous criteria. Robustness can be put into default by the violation of the assumptions such as outliers and the drift of robot evolution model.

##### 4.2 Evaluation protocol

###### 4.2.1 Test bed

Global dimensions of the test bed are 10 m x 10 m. The room is equipped with two goniometric sensors which are fixed on the walls for perceiving the robot. These sensors are labeled C. The coordinates of building sensors are  $C_1(5;0)$  m and  $C_2(0;10)$  m. In addition the robot is equipped with a goniometric sensor. The onboard goniometric sensor detects two markers fixed on the walls. The markers coordinates are  $M_3(0;5)$  m and  $M_4(10;5)$  m.

A measurement  $\lambda$  is defined by an interval bounded by the lower and upper limits,  $[\lambda] = [\Delta\lambda - \Delta\lambda, \Delta\lambda + \Delta\lambda]$ : the precision of building goniometric sensor is  $\Delta\lambda = \pi/144$ , the precision of robot goniometric sensor is  $\Delta\lambda = \pi/36$ . The variables to be estimated are the components of the state vector  $\mathbf{x} = (x_R, y_R, \theta_R)^t$

FBCB recursive algorithm ends when the length  $\omega[\mathbf{x}]$  of the solution space is less than  $\varepsilon$ . Stop condition is  $(\varepsilon_x < 0.1 \text{ m})$  and  $(\varepsilon_y < 0.1 \text{ m})$  and  $(\varepsilon_\theta < \pi/36 \text{ rd})$ .

###### 4.2.2 Scenario



Robot follows a spiral trajectory in which robot coordinates vary continuously. Localization algorithm estimates ten poses  $(x_R, y_R, \theta_R)$  of the robot along the trajectory. The number of poses has been limited for the readability of results. Two disruptive events have been considered, the drift of robot evolution model and the presence of outliers. Results are shown under different forms:

- Figures which illustrate the estimate of robot pose by a box  $(x_R, y_R)$  along the trajectory;
- Figures which show instantaneous position error in x and y;
- RMSE and associated standard deviation in  $x_R, y_R$  and  $\theta_R$ .

Scenario supposes that for estimating robot localization:

- At poses 2, 4, 5, 8 and 10, no exteroceptive measurement is provided by IOT;
- At poses 3, 6 and 9 only ( $n < 3+q$ ) measurements are available;
- At poses 1 and 7 ( $n \geq 3+q$ ) measurements are available.

It is important to note that in the best case of the scenario the global location has at most 4 exteroceptive measurements.

### 4.2.3 Evolution model errors

Evolution model is used for updating predicted hypotheses. Few authors take into account the drift of proprioceptive data during the prediction step of the estimator. Without correcting drift the predicted hypothesis no more includes the true pose of the robot. Robot kidnapping leads to the same consequence. Two errors have been added to predicted hypotheses for simulating the defaults of the evolution model:

- Drift of box center coordinates

$[\mathbf{x}]_{d,i} = [\mathbf{x}]_{p,i} + 5\%[\Delta\mathbf{s}]$  where  $\Delta\mathbf{s}$  is the Euclidian distance between two poses.

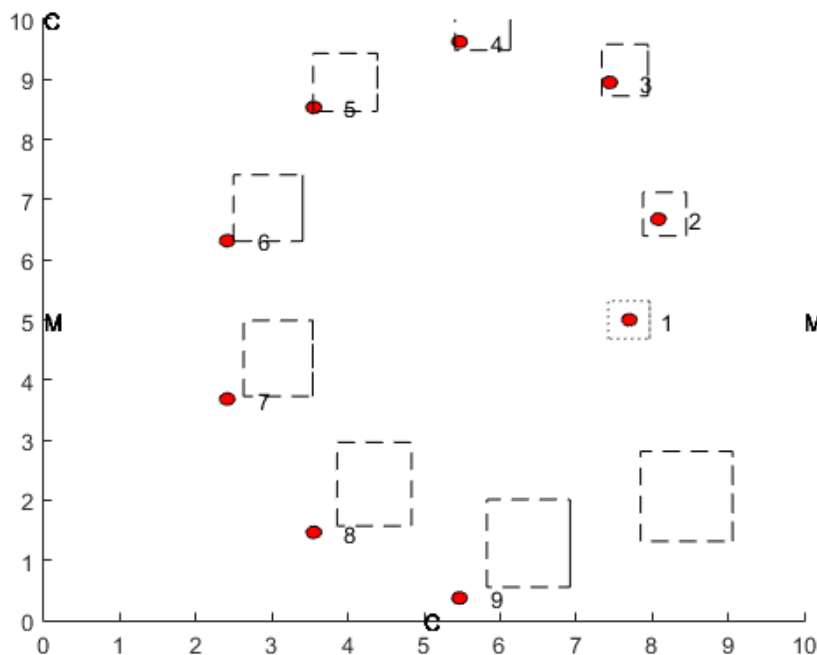
- Cumulative uncertainty by enlarging of hypothesis bounds at each iteration

For lower bounds of  $[\mathbf{x}]_{d,i}$

$[\mathbf{x}]_{d,i} = [\mathbf{x}]_{d,i} - 5\%[\Delta\mathbf{x}]$  where  $\Delta\mathbf{x}$  is the relative movement of the robot between two poses.

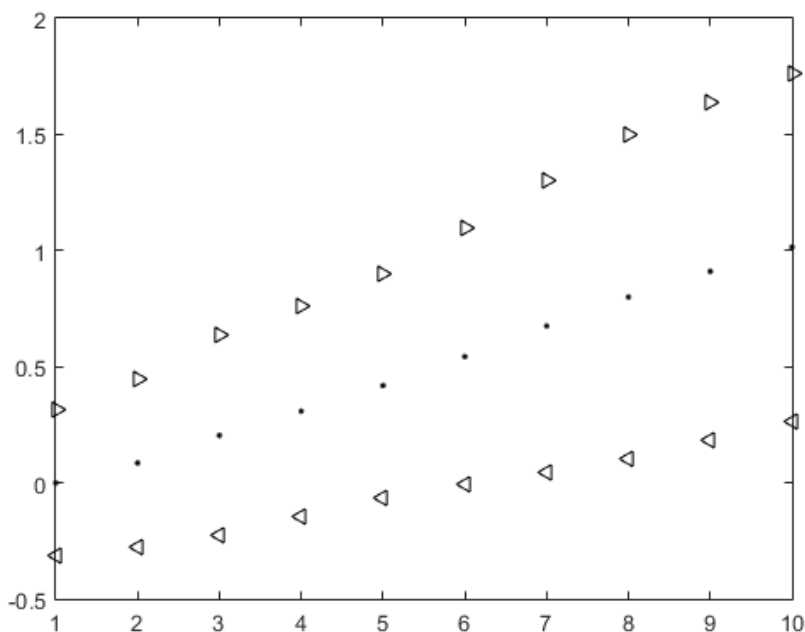
For upper bounds of  $[\mathbf{x}]_{d,i}$

$[\mathbf{x}]_{d,i} = [\mathbf{x}]_{d,i} + 5\%[\Delta\mathbf{x}]$



**Figure 1** : Predicted step. Predicted hypotheses (dashed rectangle) of the robot poses along the trajectory in x-y plane (meters). Labels C and M respectively stand for goniometric sensor and marker. Labels are located at the device coordinates. True robot position is represented by a point (red)

Figure 1 shows the effect of the evolution model defaults. At the beginning of the trajectory, the first pose is computed by the global localization algorithm (see algorithm #6). From the pose number 6, predicted hypothesis no longer includes the true position of the robot.

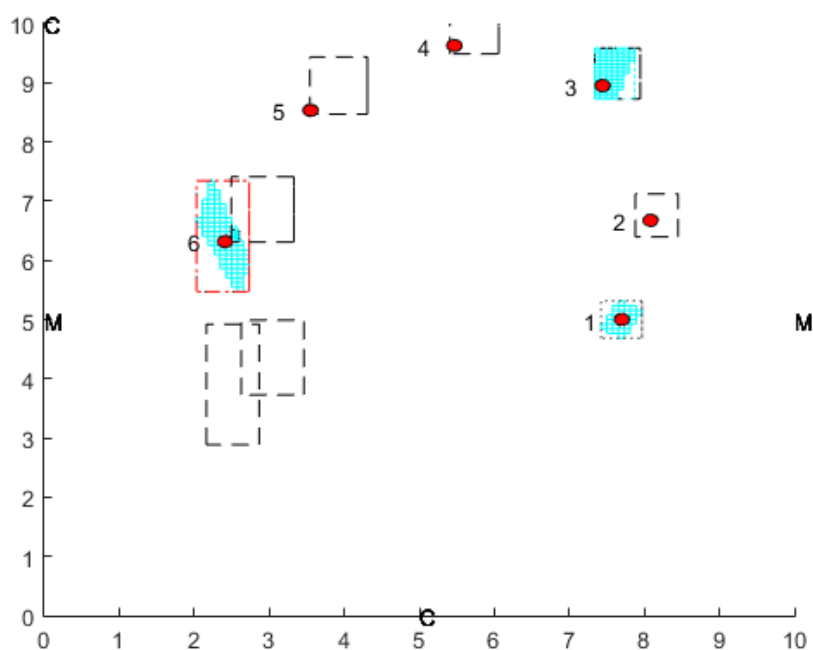


**Figure 2 :** Prediction step. Evolution of the instantaneous y error (black point) in meters with the uncertainty band (black triangle) depending on the number of the pose along the trajectory (see sect. 4.1).

In Figure 2 shows that the consistency of the predictor is not verified from the pose number 6 (see sect. 4.1). Indeed the reference value is no more part of the estimation.

#### 4.2.4 Correction of predicted hypothesis

Figure 3 shows how predicted hypothesis are corrected. Pose number 1 is supposed computed by the global localization (algorithm #6). Then if there are  $n$  measurements available such as  $0 < n < 3+q$ , it is possible to correct predicted hypotheses by new measurements (see algorithm #5). It is the case of pose number 3. Subpaving (blue) represents the effective intersection of measurements. In interval framework the result of localization is only represented by a dotted rectangle (cyan).



**Figure 3 :** Correction step. Hypotheses of the robot poses along the trajectories in x-y plane (meters): Predicted hypothesis is represented by dashed rectangle (black), Corrected hypothesis by dotted rectangle (cyan) and new hypothesis by dashed-dot

rectangle (red). Labels C and M respectively stand for goniometric sensor and marker. Labels are located at the device coordinates. True robot position is represented by a large point (red).

At pose 6, the intersection between predicted hypothesis and current measurements is empty, not in x-y plane but in  $\theta$  axis (axis not visible on Fig. 3). The robot is supposed lost. Global localization FBCB is applied to all the space  $[x_0]$  using the latest memorized measurements after synchronizing (see sect. 2.2.3). In the example the size of the stack which memorized past data is four cells. FBCB computes a new hypothesis represented by is the dashed-dot rectangle (red) which again contains the true position of the robot (see Fig. 3). From pose number 6 there are two hypotheses to be managed. As seen in section 2.4.1 a credibility is attributed to each hypothesis.

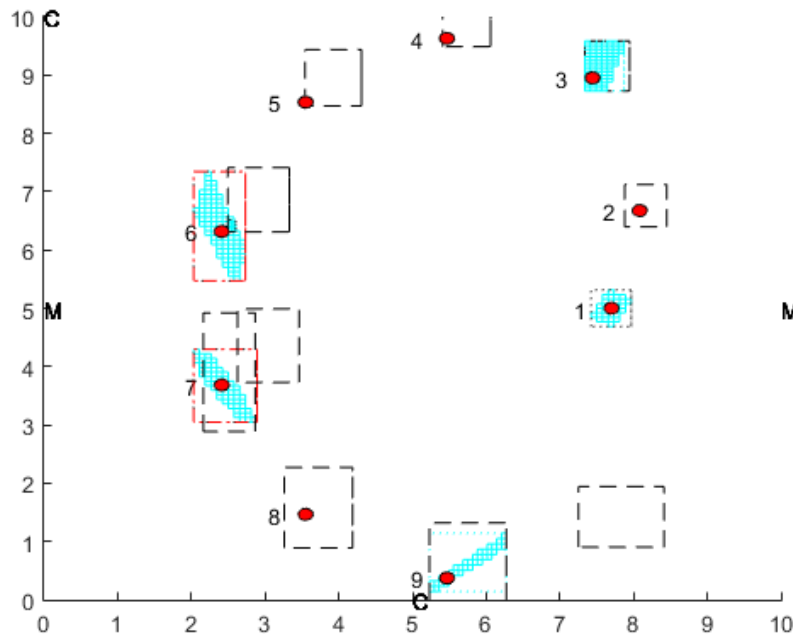
Table 2 shows the evolution of credibility attributed to hypotheses of Figure 3. Between poses 1 and 5, the initial hypothesis has been confirmed once at pose number 3, its credibility has been incremented once. So its value is 2 at pose 5. At pose 6 there are two hypotheses. The credibility attributed to the new hypothesis is 1 and the credibility of the old hypothesis is decremented because not confirmed by new measurements.

Pose number	Hypothesis number	Credibility of the first hypothesis	Credibility of the new hypothesis
Pose 1	1	1	
Pose 2	1	1	
Pose 3	1	2	
Pose 4	1	2	
Pose 5	1	2	
Pose 6	2	1	1

**Table 2 :** Evolution of hypotheses credibility for the first six poses of the trajectory

#### 4.2.5 BEE algorithm

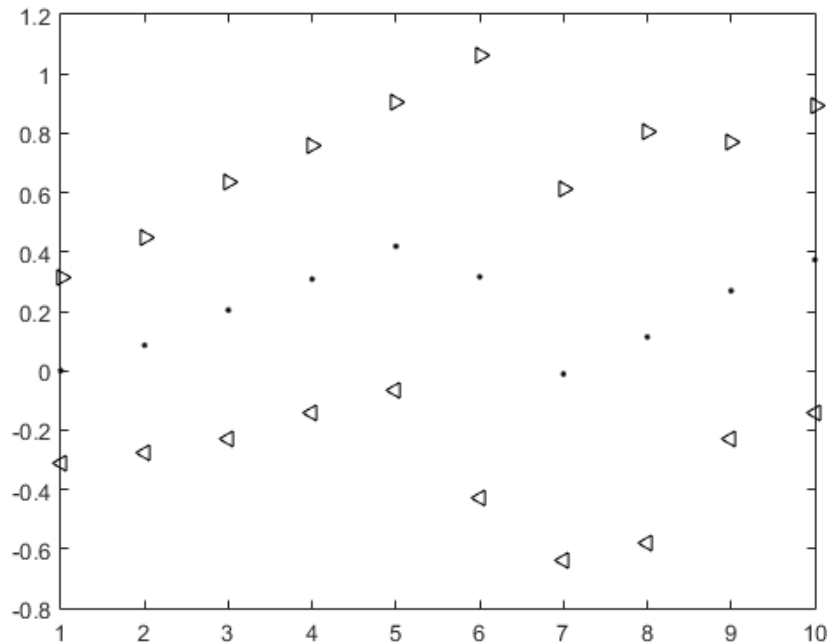
Following results concern the whole algorithm BEE which is a multihypothesis set-membership estimator. BEE processing depends on the number  $n$  of external measurements available at current time (Fig. 4). If  $n \geq 3+q$ , a global localization is computed (poses 1 and 7). If  $0 < n < 3+q$ , predicted hypotheses are corrected by external measurements (poses 3 and 9). If  $n=0$ , predicted hypotheses are updated using proprioceptive information and robot evolution model. (Poses number 2, 4, 5, 8 and 10).



**Figure 4 :** BEE algorithm. Hypotheses of the robot poses along the trajectories in x-y plane (meters): Predicted hypothesis is represented by dashed rectangle (black), Corrected hypothesis by dotted rectangle (cyan) and new hypothesis by dashed-dot rectangle (red). Labels C and M respectively stand for goniometric sensor and marker. Labels are located at the device coordinates. True robot position is represented by a large point (red).

In Figure 4 new hypotheses at poses 6 and 7, represented by dashed-dot red rectangles, correspond to two different cases. At pose 6, the intersection between predicted hypothesis and measurements is empty (not in x-y plane but in  $\theta$  axis) and the number  $n$  of available measurements is less than  $(3+q)$  (Algorithm #5). At pose 7, the number  $m$  of available measurements is greater than or equal to  $3+q$  (see Algorithm #6). At pose 7 there are 3 hypotheses. The credibility attributed to the new hypothesis is 1 and the credibility of old hypotheses are decremented because not confirmed by new measurements. As the credibility of old hypotheses becomes zero, they are removed from the list of hypotheses.

In multihypothesis tracking the calculation of the evaluation criteria takes into account all the hypotheses valid at the current time. For example at pose 6 criteria are computed using the two hypotheses which consequently degrades the precision of results. See the uncertainty band in Figure 5.

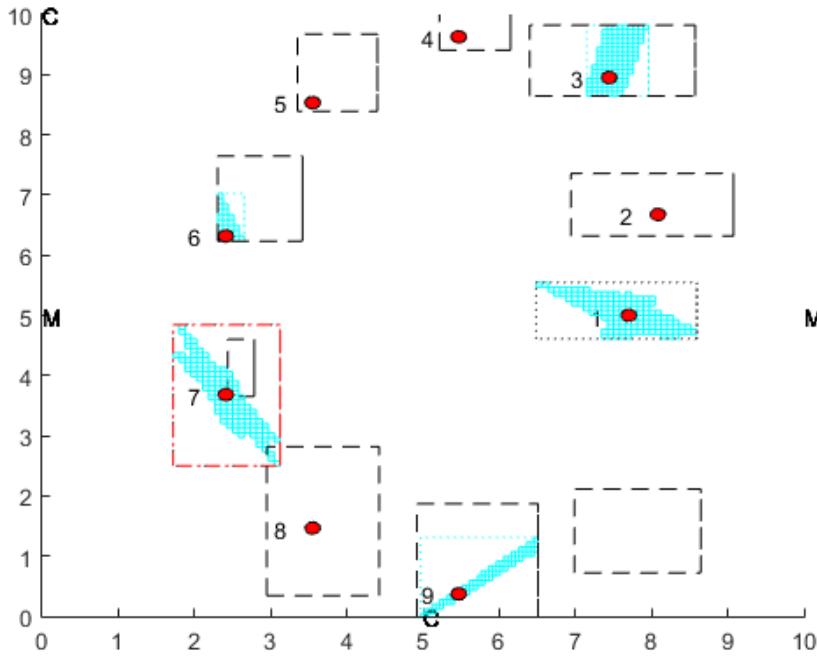


**Figure 5 :** BEE algorithm. Evolution of the instantaneous y error (black point) in meters with the uncertainty band (black triangle) depending on the number of the pose along the trajectory.

When compared with Figure 2 there are no more inconsistency.

#### 4.2.6 BEE algorithm with outliers

The  $q$ -relaxed intersection makes robust the set membership approach in the presence of inconsistent measurements. In Figure 6 BEE algorithm supposes the presence of one outlier ( $q = 1$ ). The main consequence is the degradation of the precision (increasing rectangle size). However robot remains always inside pose hypotheses. There are no inconsistency.



**Figure 6 :** BEE algorithm with one supposed outlier. Hypotheses of the robot poses along the trajectories in x-y plane (meters): Predicted hypothesis is represented by dashed rectangle (black), Corrected hypothesis by dotted rectangle (cyan) and new hypothesis by dashed-dot rectangle (red). Labels C and M respectively stand for goniometric sensor and marker. Labels are located at the device coordinates. True robot position is represented by a large point (red).

#### 4.2.7 Criteria comparison

Table 3 shows the improvement of performances given by BEE algorithm by comparing prediction step both with BEE algorithm without outlier and BEE algorithm with one supposed outlier among four measurements. Performances are evaluated by computing RMSE (Root Mean Squared Error), standard deviation and the number of inconsistencies.

Criteria	Algorithm # 4 Prediction		Algorithm BEE		Algorithm BEE	
	Without supposed outlier				With one supposed outlier	
	RMSE	Standard deviation	RMSE	Standard deviation	RMSE	Standard deviation
<b>Euclidian distance (m)</b>	0,842	0,491	0,349	0,194	0,355	0,175
<b>x (m)</b>	0,593	0,348	0,238	0,132	0,208	0,183
<b>y (m)</b>	0,597	0,348	0,254	0,153	0,287	0,155
<b><math>\theta</math> (rd)</b>	0,141	0,092	0,049	0,039	0,071	0,066
<b>Number of inconsistency in x</b>	5		0		0	
<b>Number of inconsistency in y</b>	4		0		0	
<b>Number of inconsistency in <math>\theta</math></b>	3		0		0	

**Table 3 :** Average error and the standard deviation of each step of BEE

Accuracy is improved and there is no more inconsistency even in presence of one outlier ( $q=1$ , one outlier among four measurements corresponds to 25% of outliers). From the viewpoint of run time, the frequency of BEE algorithm is close to 5 Hz with Matlab software running on Computer Intel ® Core 2(TM) i7-4710MQ CPU 2.5 GHz. Frequency respects the real time constraint of indoor robotics. If we assumed 25% of outliers, the frequency of the algorithm is substantially the same but the localization accuracy decreases.

## 5. Experimental Results

The section presents the results of three experiments with a real robot in a real environment. First evaluation presents results given by the absolute location algorithm FBCB when the robot is moving. The second evaluation shows that adding past measurements synchronized with current ones can improve the localization precision.

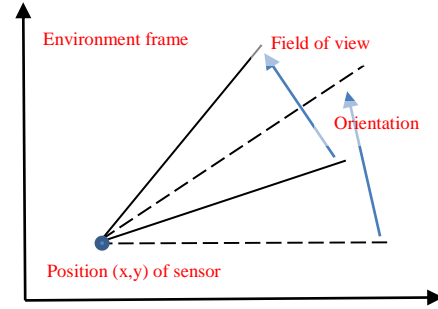
The last evaluation illustrates the principle of outlier processing. FBCB algorithm, synchronization outlier processing are three important components of the multihypothesis set estimator.

## 5.1 Test bed and scenario

### 5.1.2 Test bed

The global dimensions of the test bed are 10 m x 10 m. The room is equipped with different types of sensors. **Table 4** gives the sensor specification of the testbed: measurement precision, sensor position, sensor orientation and field of view. Three sensors are fixed on the walls of the room for perceiving the robot: a video camera (room camera) and two presence detectors. Video camera and presence detectors operate as goniometric sensors. Robot uses two onboard goniometric sensors: a **RFID reader which detects RFID Tag** and a pan tilt video camera (robot camera) **which detects visual markers**. Several **RFID tags** and visual markers are fixed on the wall. **Tags and Markers** are detected by the goniometric sensors of the robot (RFID reader and robot camera).

Sensors	Presence detector	Room Camera	Robot camera	RFID Reader
Precision $\Delta\lambda$ (°)	$\pm 45$	$\pm 2.5$	$\pm 2.5$	$\pm 10$
Field of view (°)	$\pm 45$	$\pm 22.5$	$\pm 90$	$\pm 90$
Orientation $\theta$ (°)	225	50	$\theta_{Robot}$	$\theta_{Robot}$
Position $(x_j, y_j)$ (m)	(-4.6, 4.5)	(-2.1, 1.5)	$(x_R, y_R)$	$(x_R, y_R)$



**Table 4 :** Testbed sensor specifications

For example the pan-tilt camera of the robot can detect a visual marker in a field of view of 180 degrees with an accuracy of 2.5 degrees. The orientation of the pan-tilt camera is the robot orientation  $\theta_{Robot}$ .

Table 5 describes testbed equipment.



Testbed equipment	
Two drive-wheel mobile robot (Droids company)	Environment
On board pan tilt video camera	2D circular barcode visual markers
On board RFID reader (Balogh)	Active RFID tags (Balogh)
On board dead-reckoning	Two 90° infrared detector (Legrand)
	Room video camera

**Table 5 :** Testbed equipment

Dead-reckoning uses the robot kinematic model to compute the robot displacement  $\Delta\mathbf{x}_{n-1}$  (see sect. 2.2.3)

**Goniometric measurements can be described by two types of equation.** In one hand, if a robot sensor detects an environment marker  $M_i$ , the measurement depends on the marker (or tag) coordinates  $M_i (x_i, y_i)$  and the state vector.

$$\lambda_i = \text{tg}^{-1}\left(\frac{y_R - y_i}{x_R - x_i}\right) - \theta_R \quad (4)$$

In the other hand, if the robot is detected by an environment sensor  $C_j$ , the measurement depends on the sensor coordinates and orientation  $C_j (x_j, y_j, \theta_j)$  and the state vector.

$$\lambda_j = \text{tg}^{-1}\left(\frac{y_R - y_j}{x_R - x_j}\right) - \theta_j \quad (5)$$

The state vector  $\mathbf{x} = (x_R, y_R, \theta_R)^T$  is then to be estimated from the  $M$  observations  $\boldsymbol{\lambda} = (\lambda_1, \dots, \lambda_M)$  with the associated bounded errors  $[\boldsymbol{\lambda}] = ([\lambda_1], \dots, [\lambda_M])$  and the known data  $\mathbf{x}_i = (x_i, y_i)^T$  and  $\mathbf{x}_j = (x_j, y_j, \theta_j)^T$ .

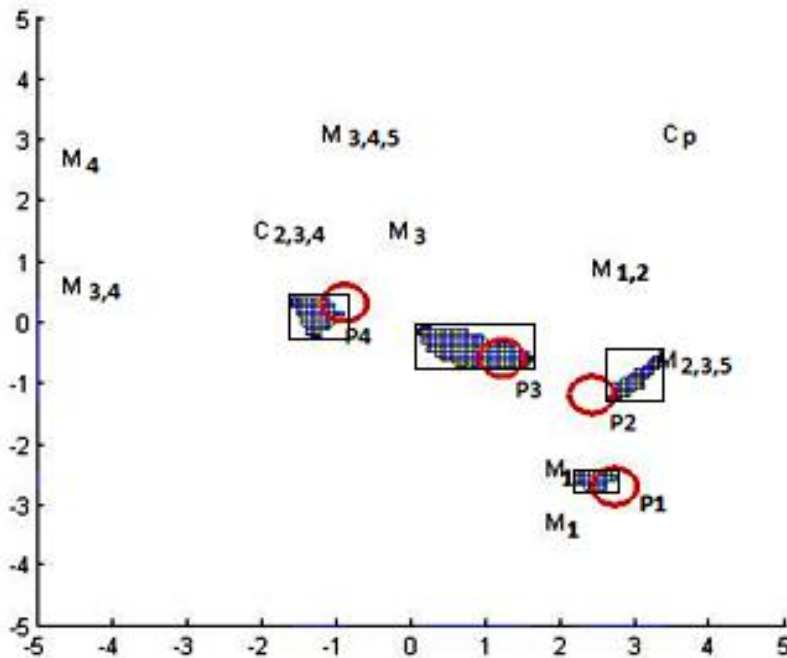
### 5.1.2 Scenario

Robot follows a planned trajectory in which robot coordinates vary continuously. Localization algorithm estimates four poses  $(x_R, y_R, \theta_R)$  of the robot along the trajectory. The number of poses has been limited for the readability of results. Robot computes localization using measures available at this pose (see Fig.7).

Labels  $P_i$  for  $i = 1$  to 4 stands for pose number  $i$ . Labels  $C$  stand for room goniometric sensors (room camera and IR detector). These sensors detect robot. Labels  $M$  stand for room markers (RFID tag and visual marker). These markers are detected by robot on-board sensors (pan-tilt camera and RFID reader). Label  $C_p$  stands for presence detector. The index attached to a label  $C$  or  $M$  indicates the number of the pose for which the calculation of localization has used this measure. For example the measures of the room video camera  $C_{2,3,4}$  has been exploited for computing robot localization at poses  $P_1, P_2$  and  $P_4$ .

### 5.2 Global localization by FBCB

Localization is calculated by FBCB for four poses  $(x_R, y_R, \theta_R)$  during the trajectory followed by the robot. For each pose, measures are collected by a gateway that handles exchanges with the ambient environment. Localization result is a 3D-box as seen before  $([x_R], [y_R], [\theta_R])$  which encloses the intersection area of measurements (blue subpaving). The true robot position is depicted by an ellipse (red)

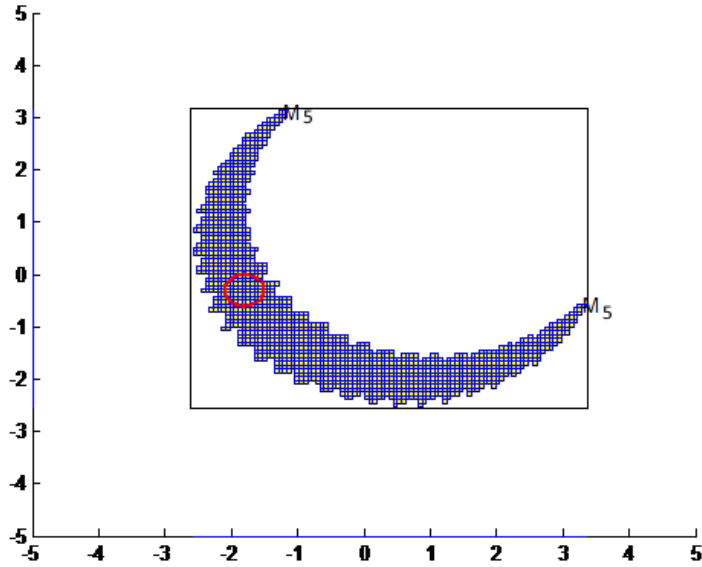


**Figure 7 :** The first four positions of the robot in  $x$ - $y$  plane  $P_i (x_i, y_i)$  in meters . Rectangles are the results given by FBCB. Subpavings (blue) represent the effective intersection of measurements. The true robot position is depicted by an ellipse (red) which suggests the fact that the ground truth is known with a certain uncertainty ( $\pm 0.25$  m).

Figure 7 shows the estimate of the position of the robot  $(x_R, y_R)$  in meters for the first four successive poses ( $P_1, P_2, P_3, P_4$ ). At pose  $P_1$  FCBC algorithm computes localization using three goniometric measures acquired by robot. In Figure 7 the three markers detected by the robot are labelled  $M_1$  or  $M_{1,2}$ . As the intersection between estimate (box) and true position (ellipse) is not empty, localization estimation is consistent. At pose  $P_2$  FCBC algorithm computes localization using two goniometric measures acquired by robot ( $M_{1,2}$  or  $M_{1,2,3}$ ) and a goniometric measure acquired by the room camera ( $C_{2,3,4}$ ). Although the intersection between the estimate and the robot position is small, it is not zero. Localization estimation remains consistent.

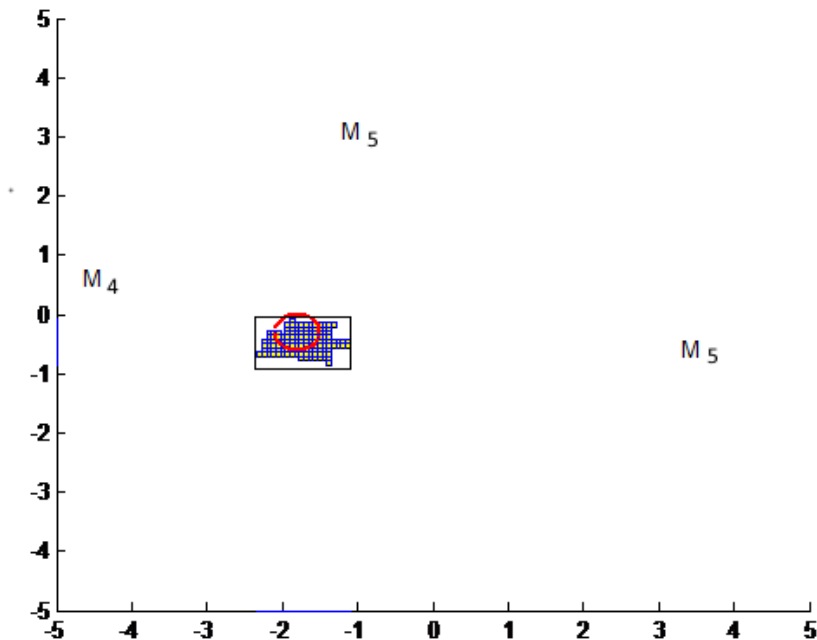
### 5.3 Global localization with data synchronization

When missing measures, the localization accuracy becomes very low as in the case of pose number five (see Fig. 8a).



**Figure 8a :** The fifth position of the robot in x-y plane (meters). The localization result is inaccurate because computed with not enough measurements, only two measurements. Subpaving (blue) represents the real intersection of the two measurements framed by a black rectangle.

Synchronization of past data presented in section 2.2.3 improves the accuracy by adding a measure acquired at pose number four (see Fig. 8b). Data synchronization takes into account the default of the model of robot evolution.



**Figure 8b :** The fifth position of the robot in x-y plane (meters). Adding a third measurement acquired at pose number 4 but synchronized with dead-reckoning data improves the accuracy of the localization.

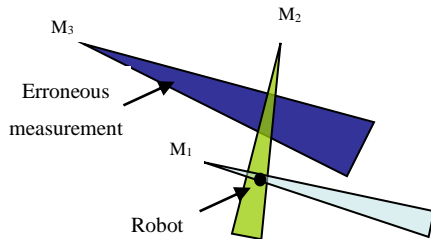
#### 5.4 Robot Localization with outlier processing

We have experimentally reproduced the case of Figure 9a in which the measurement  $M_3$  is an outlier.

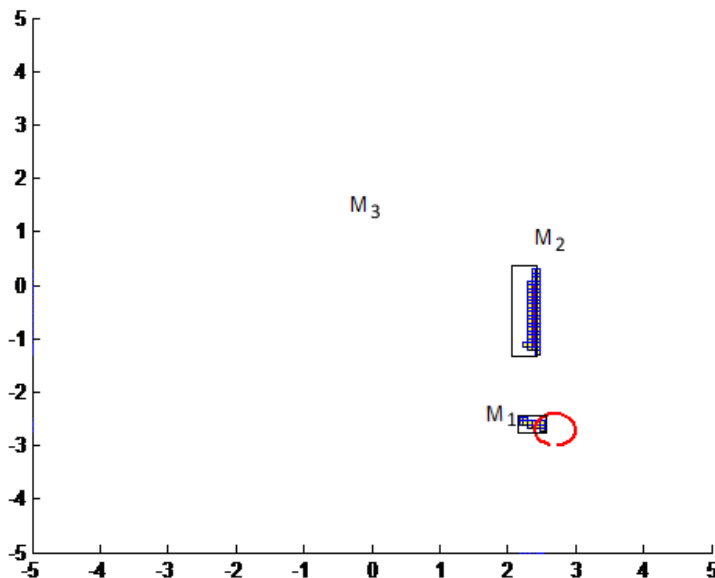


In the picture robot detects three markers. Cones represent goniometric measures acquired by the robot taking into account measurement inaccuracy.

Due to the presence of this incorrect measurement there is no common intersection between the three measurements. At best the measures intersect two by two. There are in this case two areas. Only one of the two areas corresponds to the true position of the robot (black point). As it is impossible to decide which area is correct, outlier processing has to keep both. This is why multihypothesis tracking is required.



**Figure 9a :** One among three measurements is an outlier. The true position is represented by a black point.



**Figure 9b :** One among three measurements is an outlier. A solution is found by relaxing one constraint ( $q = 1$ ), as explained in section 2.3. The result is a discontinuous set of two rectangles (black). One of them includes the true robot position represented by an ellipse (red).

Without outlier processing, set-membership method would give an empty solution. In Figure 9b the  $q$ -relaxed intersection guarantees the localization result if the effective number of outliers is less or equal to  $q$ . **Two areas have been detected by outlier processing. One corresponds to the true position of the robot.**

The main weakness of constraint relaxation is that it can generate a possibly discontinuous set as shown in Figure 9b. This obliges either to enclose all subsets in one more pessimistic box or to handle a multihypothesis approach, one hypothesis by subset of solution as done by BEE algorithm.

## 5.2 Discussion

### 5.4.1 Discussion about parameter setting

As shown by simulation and experimental results, BEE algorithm can be adapted to the application context. Parameter setting depends on the knowledge we have about the various models used by the estimator: measurement model, environment model and robot evolution model. Parameter setting also depends on the application constraints, particularly computing time, localization accuracy and robustness against inconsistent measurements. Table 1 lists the problem source and the parameter related to (see sect. 3.2). The difficulty of setting parameters depends on the source of the problem. For instance reducing the inaccuracy of environment model requires only setting only one parameter  $\Delta x_T$  but for every sensors or markers of the IoT that participates in the process of localization. At the opposite the effect of inconsistent measurements will be reduced by tuning

three parameters  $\Delta\lambda$ ,  $q$  and  $vol\_min$ . Upcoming works will investigate the influence of parameters on the quality of the localization as well as on execution time. The objective is to find the best compromise depending on the context of application and the knowledge of the process.

#### 5.4.2 Discussion about approach

Some limitations need to be highlighted regarding additional treatments. Synchronization requires knowledge of a robot evolution model. On the other hand the number  $q$  of supposed outliers is not boundless. Processing time and inaccuracy increase with  $q$ . Experimental simulations have shown that 25 % of outliers seems a good compromise between accepted risk and localization performances.

The approach is an innovative assemblage of bricks. Among them the following treatments are new: synchronization (see sect. 2.2.3), inaccuracy of environment model (see sect. 2.2.4), disruptive events (see sect. 2.2.35). Some ones have been modified: Set-membership estimator (see sect. 2.2.3), FBCB (see sect. 2.2.2).

We have completed the treatment of outliers borrowed from the literature by a multihypothesis tracking. Without such a tracking, outlier processing cannot be operational.

The approach could be extended to a multirobot environment. More generally it requires that all markers as well as all mobile objects of the environment be identified in order to determine which object is seen by a sensor.

In summary, the multihypothesis set-membership estimator takes into account measurement inaccuracy  $\Delta\lambda$  and a large variety of measurements (Table 6).

Type of measurement	Measurement	Measurement model
Goniometric (Fig.1)	angle $\lambda_i$ or $\lambda_j$	$f_i(\mathbf{x}) = \text{tg}^{-1}\left(\frac{y_R - y_j}{x_R - x_j}\right) - \theta_i$ or $f_j(\mathbf{x}) = \text{tg}^{-1}\left(\frac{y_R - y_i}{x_R - x_i}\right) - \theta_R$
Range (Fig.3a)	range $d_i$	$g(\mathbf{x}) = \sqrt{(x_R - x_j)^2 + (y_R - y_j)^2}$
Goniometric and range	angle $\lambda_i$ or $\lambda_j$ and range $d_i$	$(f_i(\mathbf{x}) \text{ or } f_j(\mathbf{x}))$ and $g(\mathbf{x})$
Dead reckoning	$\Delta x_i, \Delta y_i, \Delta \theta_i$	$x_{Rn} = x_{Rn-1} + \Delta x_n, y_{Rn} = y_{Rn-1} + \Delta y_n, \theta_{Rn} = \theta_{Rn-1} + \Delta \theta_n$ at time $n$ and $n-1$ , See [15]
Tactile tile (Fig.3c)	Center Coordinates : $C_c(x_i, y_i)$	$x_i = x_R, y_i = y_R$
Door crossing detector	$C_{ci}(x_i, y_i)$ (Same case as a narrow tactile tile)	$x_i = x_R, y_i = y_R$
Complex shape	$\{C_{ci}(x_i, y_i)\}$ for $i=1$ to $n$ The shape is divided into a set of tactile tiles	for $i=1$ to $n, x_i = x_R, y_i = y_R$

**Table 6 :** Types of measurement taken into account by the approach

The approach can correct environment model inaccuracy that is to say the ill knowledge of marker coordinates (or mobile object coordinates) in the same process. Generally drift of robot evolution model is not treated in literature. We have integrated that eventuality in the approach. In addition the approach is able to progressively include asynchronous measurements.

## 6 Conclusion

Development of the Internet of Things allows to consider fruitful cooperation between ambient environment and robots. Indeed data used by localization process can come both from the robot on-board sensors and from environment objects able to sense the robot. Set-membership approach based on bounded errors and interval analysis offers a suitable framework for heterogeneous sensor fusion. Measurements are bounded between lower and upper limits. In the paper set approach is applied to estimate the robot localization. In interval analysis framework each hypothesis of robot pose is represented by a 3-dimension box ( $[x_R], [y_R], [\theta_R]$ ). Box dimensions only depend on the measurement context (measurement accuracy, sensor configuration...). Contrary to most localization algorithm, set approach is able to give a solution as soon as one measurement is available. This property is interesting when the distribution of the sensors in the environment is sparse. Set-membership estimator provides a guaranteed solution if the assumption of bounded error is respected. The violation of this assumption can be avoided by changing the bounds of the interval which encloses the measurement. Another way consists in relaxing  $q$  constraints with  $q$  the number of supposed outliers. However relaxation can lead to a solution composed of non-contiguous sets, each considered as a pose hypothesis. So a multihypothesis tracking is needed. In addition disruptive events are detected when intersection between predicted and corrective hypotheses is empty. Prediction step of an estimator depends on the robot evolution model. Drift and inaccuracy

of the robot evolution model between two iterations are often overlooked by the authors. The multihypothesis set-membership estimator is described focusing on additional treatments which makes estimator able to process constraints of real applications:

- Measurement inaccuracy
- Environment model inaccuracy
- Drift and inaccuracy of robot evolution model
- Asynchronous measurement.

Estimator behavior depends on the number of available measurements  $n$  with respect to the reference value  $(3 + q)$  wherein  $q$  is the number of outliers. The number  $n$  might be increased by observations which might be collected at different times thanks to the synchronization mechanism.

From the viewpoint of run time, the frequency of BEE algorithm, close to 5 Hz, respects the real time constraint of indoor robotics. For instance if the maximum speed of the robot is  $1 \text{ m.s}^{-1}$  the robot displacement between two absolute localizations is only 0.2 m.

Future works aims at investigating influence of parameters on the behavior of the localization algorithm in order to find the best compromise depending on the context and the more or less precise knowledge of the application. In addition we plan to adjust bounds which enclosed all variables, measurements and objet coordinates by a learning step seen as a constraint satisfaction problem.

## References

- [1] Atzori, L., Iera, A., Morabito G. (2010). The Internet of Things: A survey, Computer Networks, Elsevier.
- [2] Paull, L., Saeedi, S., Seto, M. & Li H. (2014). AUV Navigation and Localization-ng A review. *IEEE Oceanic Engineering Journal*, 39(1), 131-149.
- [3] Cox, I. J. (1991). Blanche - an experiment in guidance and navigation of an autonomous mobile robot. *IEEE Transactions on Robotics and Automation* 7(3), 193-204.
- [4] Dissanayake, M., Newman, P., Clark, S., Durrant-Whyte, H. F. & Csorba M. (2001). A solution to the simultaneous localization and map building (SLAM) problem. *IEEE Transactions on Robotics and Automation* 17(3), 229-241.
- [5] Leonard, J. J., Durrant-Whyte, H. F. & Cox, I. J. (1992). Dynamic map building for an autonomous mobile robot *IEEE Transactions on Robotics and Automation* 11 (4), 89-96.
- [6] Castellanos, J. M., Neira, J. & Tardos, J. D. (2004). Limits to the consistency of EKF based SLAM. In *Proceedings of the 5th IFAC Symposium on Intelligent Autonomous Vehicles* (pp. 1244-1249).
- [7] Piasecki, M. (1995). Global localization for mobile robots by multiple hypothesis tracking. *Robotics and Autonomous Systems*, 6, 93-104.
- [8] Thrun, S., Fox, D., Burgard, W. & Dellaert F. (2000). Robust Monte Carlo localization for mobile robots. *Artificial Intelligence*, 128(1-2), 99-141.
- [9] Gustafson, F. (2010). Particle filter theory and practice with positioning applications. *IEEE Aerospace and Electronic Systems Magazine*, 7(25), 53-81.
- [10] Lambert, E., A., Gruyer, D., Vincke B. & Seignez, E. (2009) Consistent Outdoor Vehicle Localization by Bounded-Error State Estimation. In *Proceeding of International Conference on Robots and Systems (IROS)* (pp. 1211-1216).
- [11] Seignez, E., Kieffer, E.M., Lambert, A., Walter, E. & Maurin, T. (2009). Realtime bounded-error state estimation for vehicle tracking. *International Journal of Robotics Research*, 28(1), 34-48.
- [12] Sabater, A., Thomas, F. (1991). Set membership approach to the propagation of uncertain geometric information. In *Proceeding of IEEE International Conference on Robotics and Automation (ICRA)* (pp. 2718-2723).
- [13] Hanebeck, U. D., Schmidt, G. (1996). Set theoretical localization of fast mobile robots using an angle measurement technique. In *Proceeding of IEEE International Conference on Robotics and Automation (ICRA)* (pp. 1387-1394).
- [14] Garulli, A. & Vicino, A. (2001). Set membership localization of mobile robots via angle measurements. *IEEE Transactions on Robotics and Automation* 17(4), 450-463.
- [15] Lévêque, O., Jaulin, L., Meizel, D. & Walter, E. (1997). Vehicule localization from inaccurate telemetric data: a set of inversion approach. In *proceeding IFAC Symposium on robot Control*, 1, (pp. 179-186).
- [16] Ceccarelli, N., Di Marco M., Garulli, A., Giannitrapani, A. & Vicino, A. (2006). Set membership localization and map building for mobile robots. In *Current Trends in Nonlinear Systems and Control*, pp 289-308. Springer
- [17] Moore, R.E. (1979). Method and applications of internal analysis, SIAM, Philadelphia.
- [18] Jaulin, L., Kieffer, M., Walter, E., Meizel, D. (2002). Guaranteed Robust Nonlinear Estimation With Application to Robot Localization, *IEEE Transaction SMC, Part C Applications and Review*, 32(4), 254-267.
- [19] Jaulin, L. (2009). Robust set-membership state estimation; application to underwater robotics, *Automatica*, 45(1), 202-206.
- [20] Drevelle, V. & Bonnifait P. (2010). Robust positioning using relaxed constraint-propagation. In *Proceeding of International Conference on Intelligent Robots and Systems (IROS)* (10, pp. 4843-4848).
- [21] Colle, E. & Galerne, S. (2013). Mobile robot localization by multiangulation using set inversion. *Robotics and Autonomous Systems*, 61(1), 39-48.
- [22] Jaulin, L. & Walter, E. (1993). Set inversion via interval analysis for nonlinear bounded-error estimation. *Automatica*, 29(4), 1053-1064.
- [23] Benhamou, F. Goualard, F., Granvilliers, L. & Puget, J. F. (1999). Revising hull and box consistency. In *Int. Conf. on Logic Programming*, MIT press, 230-244.

- [25] Waltz, D. L. (1972). Generating Semantic Description from Drawings of Scenes with Shadows. *PhD thesis*, Massachusetts Institute of Technology.
- [26] Jaulin L., Walter E. and Didrit. O. (1996). Guaranteed robust nonlinear parameter bounding. In Proceeding of Symposium on Modelling, Analysis and Simulation (CESA'96 IMACS) (pp. 1156-1161).
- [27] Jaulin, L. (2011). Set-membership localization with probabilistic errors. *Robotics and Autonomous Systems*, 59(6), 489-495,
- [28] Li., X. R. & Zhao Z. (2006) Evaluation of estimation algorithms part i: incomprehensive measures of performance. *IEEE Transactions on Aerospace and Electronic Systems*, 42(4) 1340-1358.
- [29] Bar-Shalom Y. & (1998). Estimation and Tracking: Principles, Techniques, and Software. Artech House, Boston, MA, 1993. Reprinted by YBS Publishing.