High integrity GNSS location zone characterization using interval analysis

Vincent Drevelle, Philippe Bonnifait, *Université de Technologie de Compiègne, CNRS Heudiasyc UMR 6599*

BIOGRAPHY

Vincent Drevelle (1985) graduated as a computer science engineer from the Université de Technologie de Compiègne (UTC), France, in 2007; then he obtained a master degree in system and information technologies in 2008. Since September 2008 he has been at Heudiasyc UMR 6599, France, as a Ph.D. student in Computer Science and Engineering at the UTC. His current research interests are in set-theoretic methods and interval analysis, for multi-sensor localization of vehicles.

Philippe Bonnifait (1969) graduated from the École Superieure d'Électronique de l'Ouest, France, in 1992 and obtained a MSc and his Ph.D. degrees in automatic control and computer science from the École Centrale de Nantes, France, in 1997. In December 2005, he obtained his Habilitation à Diriger des Recherches from the Université de Technologie de Compiègne (UTC). Since September 1998 he has been at Heudiasyc UMR 6599, France, and he is currently professor in the UTC's Computer Science and Engineering department. His current research interests are in Intelligent Vehicles and Advanced Driving Assistance Systems, with particular emphasis on dynamic positioning based on multisensor-fusion (GNSS, dead-reckoning and GIS).

ABSTRACT

Robust Set Inversion via Interval Analysis methods in a bounded error frame is used in this paper to compute three-dimensional location zones in real time, at a given confidence level. This approach differs significantly from the usual Gaussian error model paradigm since the satellite positions and the pseudo-ranges measurements are represented by intervals enclosing the true value with a particular confidence. The method computes a location zone recursively, using contraction and bisections of an arbitrarily big initial location box. Such an approach also enables to consider the presence of an arbitrary number of erroneous measurements using a q-relaxed solver, and allows integration of geographic and cartographic information such as digital elevation models or 3-dimensional maps. With enough data redundancy, inconsistent measurements can be detected and even rejected.The integrity risk of the location zone is only brought by the measurement bounds settings, since the solver is guaranteed. A way to set these bounds for a particular location zone confidence is proposed. An experimental validation using real L1 code measurements and a digital elevation model is also reported to illustrate the performance on the method with real data.

INTRODUCTION

When leading to safety decisions, positioning services not only need to provide an estimate of the user location, but also confidence indicators. This information enables the user to know if the position estimate is usable in a given context.

In practice, an upper bound on the positioning error, linked to an integrity risk, is required to determine if a navigation system is usable for a given task.

At the receiver level, standard approaches do Fault Detection (FD) to ensure the integrity of the position solution. This is known as Receiver Autonomous Integrity Monitoring (RAIM) (Brown and Chin, 1997; Walter and Enge, 1995). RAIM uses statistical consistency tests when measurements redundancy exists. In this process, FD characterizes the presence of pseudorange measurement errors that could lead to compute hazardous and misleading information of position.

The monitoring system has also to provide Horizontal and Vertical Protection Levels (HPL/VPL), which are upper bounds of the position error that must not be violated without being detected with a given integrity risk and in a given time to alert. A protection level is defined by two probabilities: the probability of missed detection and the probability of false alert. A false alert arises when a position failure is declared although there is no failure. A missed detection is the non indication of an alert when an unacceptable error occurs. These two parameters are set to meet the RNP requirements on integrity and availability. The protection level reflects the error detection capability of FD, given the current geometrical configuration (user position and satellite positions) and the expected measurement error characteristics. It is thus computed without the actual measurement values, as a mean to know if the positioning system is able to ensure the RNP in the given situation – *i.e.* if a position error greater than the alert limit could be detected. This way, RAIM is declared to be unavailable if the protection level exceeds the alert limit.

An estimate of the actual position error is based on the current measurements. It is often called Uncertainty Level (UL). UL describes the contribution of two kinds of disturbances: noise and missed detections. Usually, it is supposed that there is only one fault at a time. The minimal

detection bias is then applied to the measurement that affects the position error in the worst case, by a mechanism called *Max Slope* in the literature (Feng et al., 2006).

Localization can thus be considered as a set-theoretic (also called set-membership) problem that consists in computing a zone in which the user is located with a given confidence or risk, instead of just the coordinates of a point that should be near to the actual location. Indeed, an upper bound of the resulting position error is essential to decide if the navigation system can be used for a dedicated task. Avionics GPS receivers computes currently protection and uncertainty levels that characterize the position error; these are then compared to alert limits to decide of GPS availability as a primary navigation mean.

Set-theoretic methods are well suited to address localization problems, especially when dealing with uncertainty of position (Lévêque, 1998; Jaulin et al., 2002; Meizel et al., 2002). Some of them have shown interesting properties like guaranteed convergence and guaranteed characterization of every location hypothesis in a given search zone (multimodal ability). The main drawbacks of this kind of approach are of two kinds. First, they are often very conservative (or pessimistic) because of the measurement bounds that can be chosen excessively large and because of the wrapping effect (Jaulin et al., 2001b). Second, they are very sensitive to faults or outliers which can induce an empty set solution. This paper addresses those issues by using a technique that uses non guaranteed intervals and outliers: if a risk has been taken when setting error bounds, the risk of the solution set not including the true position can be bounded. From a practical point of view, the computed location zone (eventually the several ones) is not guaranteed, but an integrity risk is specified, depending upon the application. To achieve high integrity levels with good accuracy, a lot of data redundancy may be needed; otherwise, the location zone may become very large.

The paper is organized as follows. We first introduce set-membership localization with a simple pedagogical example. Then, guaranteed solvers for the set inversion localization problem, based on interval analysis and constraint propagation are presented. A robust solver is also introduced, and a method to set measurement error bounds according to a specified risk is explained. Finally, experimental results using real L1 GPS pseudo-ranges measurements are presented and analyzed, external altitude information is integrated to improve the solutions thanks to a digital elevation model.

SET-MEMBERSHIP POSITIONING

Pedagogical example with static beacons

Let's consider a time of flight positioning example having similarities with GNSS navigation. In a planar world, a robot and three beacons communicate altogether via a radio link. The robot can emit ultrasonic waves, while the beacons feature a receptor. To range itself to the beacons, the robot simultaneously emits a radio message and an ultrasound at time t_e. Beacons start timing, and stop when they receive the ultrasound at t_r. Since radio propagation time is negligible compared to sound travel time, the time of flight measurement is $t_r - t_e$. Knowing the speed of sound c_s, distance measurements to each beacon can easily be determined $d^i = c_s \cdot (t_r^i - t_e)$.

To perform set-membership positioning, each measurement has to be represented as the set of possible values, taking uncertainty into account. Intervals are commonly used to express measurement inaccuracy. In the case of time of flight measurements based on ultrasound transducers, a proportional error is expected, due to the variation of the speed of sound. Measurements can be represented as intervals

$$[d^i] = [(1-\alpha)\cdot d^i, (1+\alpha)\cdot d^i] \quad (1)$$

Each measurement acts as a constraint on the robot location, setting bounds on the distance between the robot and the beacon. Thus, we have the membership relation:

$$\sqrt{(x_R - x_{B^i})^2 + (y_R - y_{B^i})^2} \in [d^i] \quad (2)$$

Given equation 2, each measurement constrains the robot location inside a ring, whose inner and outer radii are respectively the lower and outer bounds of the measurement interval $[d^i]$.

As there are three beacons, the measurement equation has to be verified for the three measurements. The robot is thus located in the intersection of the three rings (the black area in Fig. 1a).

It is important to keep in mind that as long as the measurement error is consistent with the chosen bounded error model, the true user location is inside the solution set.

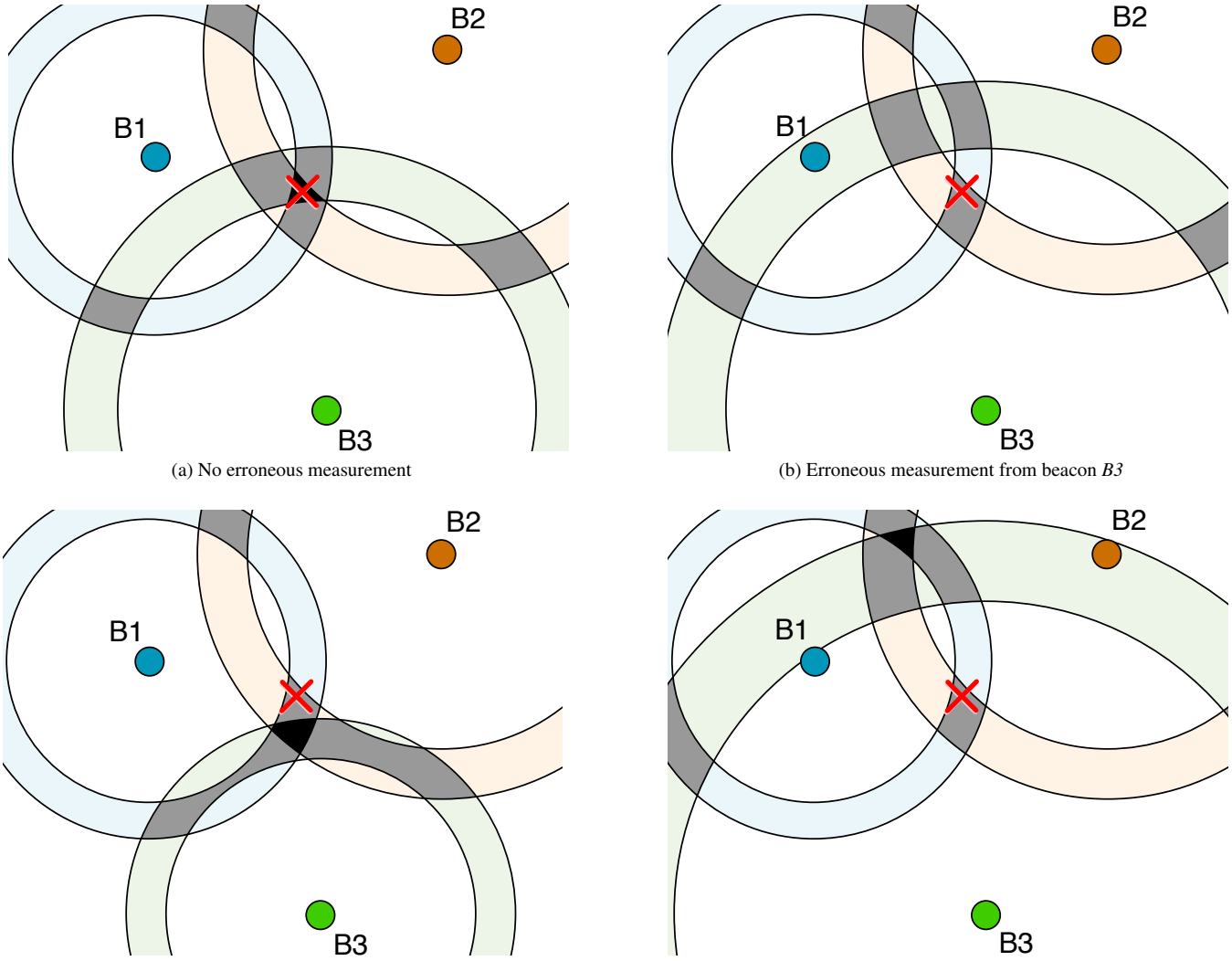
Influence of wrong measurements

Measurement systems are often prone to errors, so that a measurement may be inconsistent with the error-bounds. This kind of measurement is often called "outlier" or "fault". There are two possible consequences:

- The solution is the empty set (Fig. 1b). This case happens when the wrong measurement is inconsistent with the other measurements, so that there is no common intersection. On can thus conclude that there is something wrong with the measurements or the model.

- The solution is not empty, but does not contain the actual robot location (Fig. 1c and 1d). The set-membership method is then unable to detect the presence of an erroneous measurement, and the solution set is inconsistent with the truth.

To deal with erroneous measurements, a robust set-membership method has to be used. This is done by relaxing the number of constraints to be satisfied. In this example, allowing the presence of at most one erroneous measurement is achieved by considering the set of solutions compatible with at least two measurements (the gray and black surfaces in Fig. 1). In the general case, an arbitrary number q of erroneous measurements can be tolerated, using the q-relaxed intersection of the constraints (Jaulin,



(a) No erroneous measurement

(b) Erroneous measurement from beacon *B3*

(c) Non-detectable small error

(d) Non-detectable large error

Figure 1: Localization using 3 beacons. *Solution set in black. 1-relaxed solution set in black and grey. The actual solution is shown as a red cross.*

2009). Tolerating erroneous measurements leads to larger solution sets. A good balance between the error model tolerance and the number of tolerated outliers has to be found, in order to keep the solution set narrow. One should also ensure that the number of active constraints remains sufficient to compute a finite solution-set.

Another robust scheme called *Guaranteed Minimum Outlier Number Estimator* (Kieffer et al., 2000) (GOMNE) consists in adaptively relaxing the number of constraints: the strategy is to first compute a solution set considering all the constraints, then to iteratively relax the number of constraints to be satisfied until a non-empty solution set is obtained. The good point of GOMNE is to keep the solution set narrow as long as measurements are not incompatible. However, there is no guarantee that the true value will be located inside the solution returned by GOMNE, since a wrong measurement is not necessarily inconsistent with the other measurements (Fig. 1d).

SET INVERSION VIA INTERVAL ANALYSIS

In the previous example, the solution sets for each constraint were easy to represent as rings, and we supposed the ability to compute with exact representations of arbitrary sets. In real localization problems, the constraints are given by the measurements and an observation function, which can lead to arbitrary sets of solutions. We will thereby use interval analysis to perform a guaranteed set inversion.

Interval analysis

Exact representation of arbitrary sets is not tractable. An efficient representation is to consider intervals, and their multidimensional extension: *interval vectors* also called *boxes*. Let $\mathbb{IR}$ be the set of real intervals, and $\mathbb{IR}^n$ the set of n-dimensional boxes.

The classical real arithmetic operations ($+, -, \times$ and $\div$) can be extended to intervals. The interval extension of an operator returns the smallest interval containing all the results of the operation when the

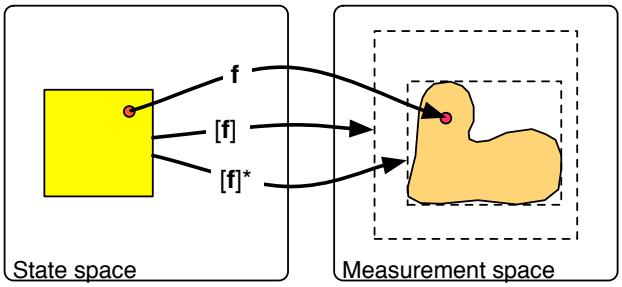


Figure 2: Inclusion functions. $[\mathbf{f}]$ *is an inclusion function for* $\mathbf{f}$. $[\mathbf{f}]^*$ *is the minimal inclusion function for* $\mathbf{f}$.

two operands cover their respective intervals. With a binary operator $\diamond$, we have

$$[x] \diamond [y] = [\{x \diamond y \in \mathbb{R} | x \in [x], y \in [y]\}].$$

In the same way, elementary functions such as $\tan, \sin, \exp \ldots$ extend to intervals. Given a function $f : \mathbb{R} \to \mathbb{R}$, its interval extension $[f]$ is given by

$$[f]([x]) = [\{f(x) | x \in [x]\}].$$

Let's consider a function $\mathbf{f}$ from $\mathbb{R}^n$ to $\mathbb{R}^m$. The interval function $[\mathbf{f}]$ from $\mathbb{IR}^n$ to $\mathbb{IR}^m$ is an *inclusion function* (Fig. 2) for $\mathbf{f}$ if the image of $[\mathbf{x}]$ by $[\mathbf{f}]$ includes the image of $[\mathbf{x}]$ by $\mathbf{f}$

$$\forall [\mathbf{x}] \in \mathbb{IR}^n, \mathbf{f}([\mathbf{x}]) \subset [\mathbf{f}]([\mathbf{x}]).$$

The minimal inclusion function for $\mathbf{f}$, $[\mathbf{f}]^*$ returns the smallest box that contains $\mathbf{f}([\mathbf{x}])$ — *i.e.,* the *interval hull* of $\mathbf{f}([\mathbf{x}])$.

If a function $f : \mathbb{R}^n \to \mathbb{R}$ can be expressed as a finite composition of operators $+, -, \times, \div$ and elementary functions (sin, cos, sqr...), an inclusion function $[f]$ for f is obtained by replacing each variable and each operator or function by their interval counterparts. This inclusion function is called the *natural inclusion function* of f. If f only involves continuous operators and functions, and if moreover each variable appears at most once in the expression of f, then the natural inclusion function is minimal.

The natural inclusion function for a vector function $\mathbf{f}$ is obtained by taking natural inclusion functions for each of its coordinate functions f_i.

To approximate compact sets in a guaranteed way, *subpavings* will be used. A subpaving of a box $[\mathbf{x}]$ is the union of non-empty and non-overlapping subboxes of $[\mathbf{x}]$. A guaranteed approximation of a compact set $\mathbb{X}$ can be done by bracketing it between an inner subpaving $\underline{\mathbb{X}}$ and an outer subpaving $\overline{\mathbb{X}}$ such as $\underline{\mathbb{X}} \subset \mathbb{X} \subset \overline{\mathbb{X}}$ (Fig. 3).

Set inversion

The set inversion problem consists in determining the set $\mathbb{X}$ such as $\mathbf{f}(\mathbb{X}) = \mathbb{Y}$ when $\mathbb{Y}$ is known.

A box $[\mathbf{x}]$ of $\mathbb{IR}^n$ will be said *feasible* if $[\mathbf{x}] \subset \mathbb{X}$ (all elements of the box are solutions of the problem) and *unfeasible* if $[\mathbf{x}] \cap \mathbb{X} = \emptyset$ (all elements of the box are not solutions of the problem), otherwise $[\mathbf{x}]$ is ambiguous.

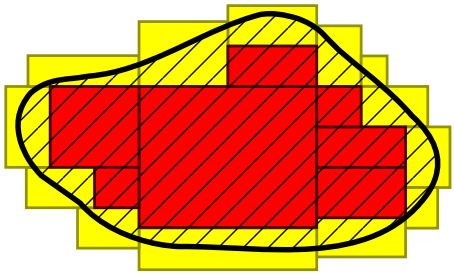


Figure 3: Bracketing of the hatched set between two subpavings. *Red boxes: inner subpaving, Red and yellow: outer subpaving*

Using an inclusion function $[\mathbf{f}]$ of the function $\mathbf{f}$ to be inverted, we can identify feasibility of boxes:

- If $[\mathbf{f}]([\mathbf{x}]) \subset \mathbb{Y}$ then $[\mathbf{x}]$ is feasible
- If $[\mathbf{f}]([\mathbf{x}]) \cap \mathbb{Y} = \emptyset$ then $[\mathbf{x}]$ is unfeasible
- Else $[\mathbf{x}]$ is indeterminate, meaning it can be feasible, unfeasible or ambiguous.

Starting from an arbitrarily big prior searching box $[\mathbf{x}_0]$, the Set Inversion Via Interval Analysis (Jaulin and Walter, 1993) algorithm (SIVIA) works by testing feasibility of boxes. If a box is feasible, it is added to the inner subpaving $\underline{\mathbb{X}}$ of solution. If a box is unfeasible, it is discarded, since it is proven that it contains no solution. Finally, an indeterminate box is bisected into two sub-boxes, which are enqueued in the list $\mathfrak{L}$ of boxes waiting to be examined. Indeterminate boxes whose width is too small (less than ε) are added to the subpaving of indeterminate boxes $\Delta\mathbb{X}$. Thus the outer subpaving is $\overline{\mathbb{X}} = \underline{\mathbb{X}} \cup \Delta\mathbb{X}$.

Algorithm 1 SIVIA(in: $[\mathbf{x}_0], \mathbb{Y}$; out: $\underline{\mathbb{X}}$, $\Delta\mathbb{X}$)

```
1:  push([x_0], L)
2:  while L ≠ ∅ do
3:     [x] = pull(L)
4:     if [f]([x]) ⊂ Y then
5:        X = X ∪ [x]
6:     else if [f]([x]) ∩ Y = ∅ then
7:        discard [x]
8:     else if w([x]) < ε then
9:        ΔX = ΔX ∪ [x]
10:    else
11:       ([x_1], [x_2]) = bisect([x])
12:       push([x_1], L); push([x_2], L)
13:    end if
14: end while
```

There are several ways to implement the list of boxes $\mathfrak{L}$ used in the algorithm. Implementing $\mathfrak{L}$ as a stack provides a minimal memory occupation that can be bounded as a function of n and ε. It is a depth first search, requiring the end of computation to get a usable result. By setting $\mathfrak{L}$ as a queue, search over the state space is done breadth first, allowing an homogeneous width of subpaving over

the whole solution set. This enables the computation to be stopped at any time to get a result, thus being compatible with real time applications. The main drawback is memory occupation, which is a lot larger than when using a stack.

SIVIA is described in algorithm 1, where the *push* and *pull* functions are respectively used to add and extract a box from the list.

The number of needed bisections gets exponentially bigger as the dimension of the problem increases, and the computational burden quickly becomes intractable. To counteract this effect, *contractors* have to be used. A contractor is a fonction that shrinks a box without loosing any solution. It allows to speed-up computation without loosing guarantee of the solution. A simple way to build a contractor is using a constraint propagation algorithm (Jaulin et al., 2001b).

With contractors, the generated subpavings are no longer regular, and size of boxes is unpredictable. A *sorted list* implementation of the list of boxes may be used, to bisect the largest boxes first, thus leading to rather homogeneous box sizes.

q-relaxed set inversion

Adding robustness can be done by relaxing a given number q of constraints. The solver will then compute a subpaving of the state space consistent with at least $m-q$ measurements. The *Robust Set Inverter via Interval Analysis* (RSIVIA) solver(Jaulin et al., 2001b) enables guaranteed computation of a q-relaxed solution set.

The component functions f_i of $\mathbf{f}$ are considered independently, with their inclusion functions $[f_i]$. If feasibility of $[\mathbf{x}]$ is achieved with at least $m-q$ components of $\mathbb{Y}$ via component inclusion functions $[f_i]$, $[\mathbf{x}]$ will be considered as feasible. Else, if infeasibility of $[\mathbf{x}]$ is concluded for more than q components, $[\mathbf{x}]$ will be unfeasible. Otherwise, $[\mathbf{x}]$ will be indeterminate.

Fig. 4 shows the main steps of RSIVIA, applied to a fixed beacon localization problem. The prior box is first contracted independently with each measurement to get three boxes approximating the intersection of each constraint with the prior box (Fig. 4a). The grayed zones in Fig. 4b represents the 1-relaxed intersection of the three boxes. The hatched box, which is the box union of the grayed boxes, becomes a new initial box. We have thus contracted the initial box, without loosing any solution. Contraction with each measurement is done again, starting with the new box (Fig. 4c). Then, the initial box is reduced again so as to enclose the q-relaxed intersection of contracted boxes. These steps are repeated (Fig. 4d and 4e) until no more contraction can be performed (Fig. 4f). A bisection is then done, and the contraction process is applied to the two sub-boxes (Fig. 4g).

A fast contractor for the q-relaxed intersection can be implemented using axis projection of constraints. Each dimension is considered separately. Constraint propagation is applied to the input box, thus obtaining a smaller box constrained by only one measurement. The obtained upper and lower bounds for the considered axis are added to a list of bounds and associated values. Each opening bracket (lower bound) is associated with the $+1$ value while each upper bound is associated with the -1 value. Such a list of bounds is constructed for each dimension, and populated by applying constraint propagation of each available measurements to the input box.

For each axis, bounds are sorted in ascending order and a counter is set to 0. Then, the bounds associated to the axis are examined from the lowest to the highest. Each time a bound is encountered, its associated value is added to the counter. The first bound that makes the counter hit $m-q$ is set as the axis' lower bound. The last bound that makes the counter fall below $m-q$ is set a the upper bound for the considered axis. The contracted box is the Cartesian product of the contracted intervals obtained on each dimension. The contraction process is iterated until no more reduction of the box size is obtained.

This contractor is not optimal – *i.e* it does not reach the interval hull of the q-relaxed solution set in the general case – but its complexity is low. After contraction, the box is bisected and enqueued so as to be further processed to get a thinner result.

Setting measurement error bounds as a function of a given confidence

When using a q-relaxed guaranteed solver such as RSIVIA, the probability of the true solution being inside the computed solution set can be computed, given a prior measurement error distribution and a maximum number of outliers (denoted q) (Drevelle and Bonnifait, 2009).

Knowing the probability density function f_{e_y} of the measurement error e_y and the error bounds $[a,b]$, a measurement y_{meas} is represented by the interval $[y_{meas}] = [y_{meas}+a, y_{meas}+b]$. One can compute the probability $p = P(y \in [y_{meas}])$ of the true y being inside $[y_{meas}]$.

$$p = P(y \in [y_{meas}]) = \int_a^b f_{e_y}(\alpha) d\alpha \tag{3}$$

Let n_{ok} be the number of measurements that respect the error bounds. The probability of having exactly k good measurements out of m is given by binomial law:

$$P(n_{ok} = k) = \frac{m!}{k!(m-k)!} p^k (1-p)^{m-k} \tag{4}$$

Thus, by summing equation 4 over successive k values, the probability of having at least $m-q$ good measurements is

$$P(n_{ok} \geq m-q) = \sum_{k=m-q}^{m} \frac{m!}{k!(m-k)!} p^k (1-p)^{m-k} \tag{5}$$

A guaranteed algorithm like RSIVIA computes a conservative approximation $\overline{\mathbb{X}}$ of the solution set $\mathbb{X}$. Moreover, if the hypotheses made on the measurements are verified, the solution set is consistent with the truth. This way,

$$n_{ok} \geq m-q \Rightarrow x \in \mathbb{X} \Rightarrow x \in \overline{\mathbb{X}}$$

which leads to

$$P(x \in \overline{\mathbb{X}}) \geq P(x \in \mathbb{X}) \geq P(n_{ok} \geq m-q) \tag{6}$$

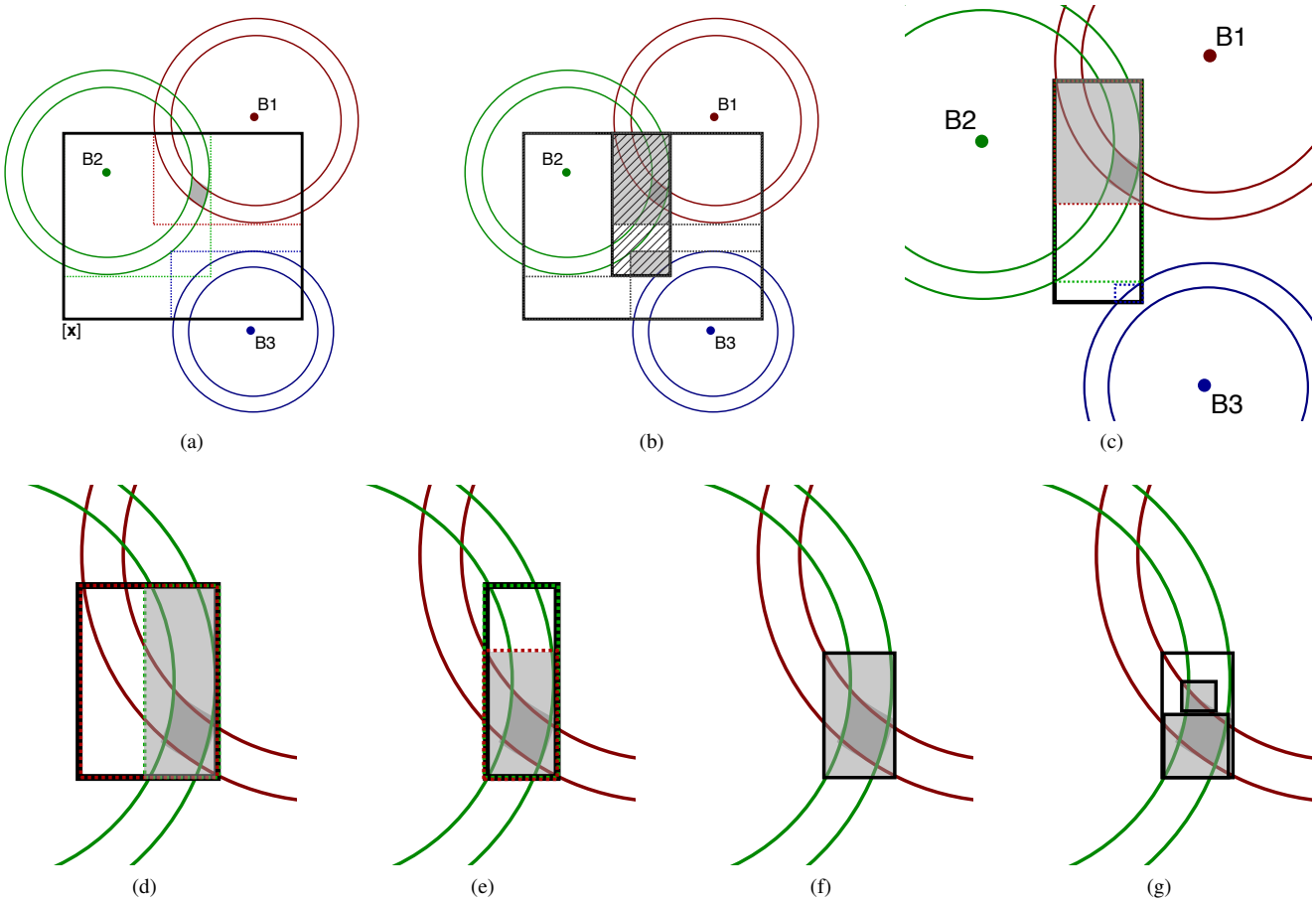


Figure 4: Steps of RSIVIA on a 3 beacon localization problem

Setting $P(n_{ok} \geq m-q)$ to a confidence level P_{conf} will ensure at least this confidence level for the computed solution. Since the number of measurements m and the number of tolerated outliers q are known, p can be determined from Eq. 5 for a given $P(n_{ok} \geq m-q)$. Measurement bounds can then be chosen so as to verify Eq. 3.

There is one degree of freedom left to choose the lower and upper bounds a and b of the measurement error interval. One can thus decide to choose them so as to minimize the width of $[a,b]$ – *i.e.* to minimize $a-b$.

In the case of a centered Gaussian measurement error $e_y \sim \mathcal{N}(0,\sigma_y)$, with Φ representing the cumulative distribution function of the standard normal distribution, the measurement interval should be set to

$$[y_{meas}] = [y_{meas} - K\sigma_y, y_{meas} + K\sigma_y] \tag{7}$$

In this case, K is simply given by

$$K = -\Phi^{-1}\left(\frac{1-p}{2}\right) \tag{8}$$

COMPUTATION OF GPS LOCATION ZONES

Set-membership GPS localization

GPS localization using pseudoranges is a four-dimensional problem: along with space coordinates (x,y,z) of the user, the user clock offset dt_u has to be estimated. With ρ_i being the corrected pseudo-ranges, the GPS code observation model is :

$$\begin{pmatrix} \rho_1 \\ \rho_2 \\ \vdots \\ \rho_m \end{pmatrix} = \begin{pmatrix} \sqrt{(x-x_1^s)^2+(y-y_1^s)^2+(z-z_1^s)^2} + cdt_u \\ \sqrt{(x-x_2^s)^2+(y-y_2^s)^2+(z-z_2^s)^2} + cdt_u \\ \vdots \\ \sqrt{(x-x_m^s)^2+(y-y_m^s)^2+(z-z_m^s)^2} + cdt_u \end{pmatrix}$$

Satellite positions (x_i^s, y_i^s, z_i^s) are known with uncertainty due to the inaccuracy of the broadcast ephemeris information. For each satellite, we consider a box $[\mathbf{x}_i^s] = ([x_i^s],[y_i^s],[z_i^s])$ whose bounds are chosen to contain the true satellite position at a given confidence level.

Measured pseudo-ranges are compensated from relativistic effects, ionosphere and troposphere propagation delays using EGNOS to get corrected pseudoranges ρ_i. These corrections are imprecise due to model and parameters errors. Moreover, the receiver makes also measurement errors. Therefore, we model the pseudo-range measurements as intervals $[\rho_i]$ whose bounds will be determined given an integrity risk.

The location zone computation consists in characterizing the set $\mathbb{X}$ of all locations compatible with the

measurements and the satellite positions intervals:

$$\mathbb{X} = \Big\{ (x,y,z,cdt_u) \in \mathbb{R}^4, \forall i = 1 \dots m, \\ \exists \rho_i \in [\rho_i], \exists (x_i^s, y_i^s, z_i^s) \in [\mathbf{x}_i^s], \\ \rho_i = \sqrt{(x-x_i^s)^2 + (y-y_i^s)^2 + (x-y_i^s)^2} + cdt_u \Big\}$$

The solution set is the set of locations for which a pseudorange and a satellite position can be found inside the measurement and satellite position intervals for every satellite.

To add robustness, we will compute a location zone compatible with at least $m-1$ measurements. RSIVIA is used to solve the relaxed set inversion problem. By recursively contracting and bisecting an arbitrarily big initial box, this algorithm returns a subpaving of the state-space (user position and clock offset) guaranteed to include the solution set — *i.e.,* an outer approximation of the solution set by a set of boxes. As long as the initial hypothesis is valid, *i.e.* as long as only at most one measurement exceeds the error bounds, the true receiver position is guaranteed to be inside the computed localization zone. The bounds are chosen so that the risk of more than one wrong measurement (*i.e.* the risk that at least two measurements don't respect the error bounds) is of 10^{-7}.

The constraint induced by the i^{th} pseudorange mesurement is represented by the natural inclusion function for the observation function:

$$[f_i]([\mathbf{x}]) = \sqrt{([x]-[x_i^s])^2 + ([y]-[y_i^s])^2 + ([z]-[z_i^s])^2} + c[dt_u] \quad (9)$$

The square and square root functions are replaced by their interval extensions. Since each variable appears only one time in the expression of f_i, the natural inclusion function obtained by replacing each operator of f_i by its interval counterpart is the minimal inclusion function for f_i.

A contractor can be built using constraint propagation. Since the decomposition in elementary constraints contains no cycle (Fig. 5), an improvement of the Waltz constraint propagation algorithm (Waltz, 1972) can be used: the FALL-CLIMB algorithm (Jaulin et al., 2001a), witch allows to propagate constraints in an optimal order. The input and output variables (shown in red in Fig. 5) are set to their initial interval values. Unknown intermediate values are set to $[-\infty,+\infty]$. A forward propagation is done, falling from the leafs down to the root of the tree. Then backward propagation is done, climbing the tree from the root up to the leafs. When a binary operator is encounterd, contration is done on its two operands.

Using the $+$ operator as an example, let us contract the $[a]+[b]=[c]$ constraint:

- forward:
$$[c] \leftarrow ([a]+[b]) \cap [c]$$
- backward:
$$[a] \leftarrow ([c]-[b]) \cap [a]$$
$$[b] \leftarrow ([c]-[a]) \cap [b]$$

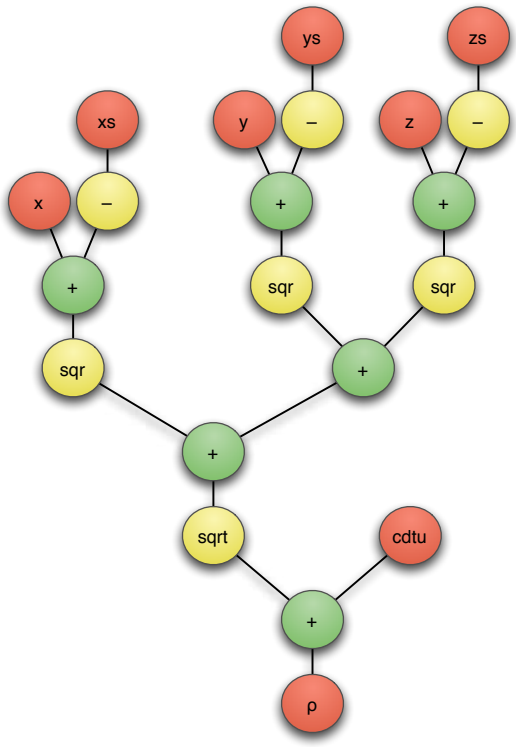


Figure 5: Elementary constraints decomposition of pseudorange measurement inclusion function

Figure 6: Our experimental vehicles: CARMEN and STRADA

We use a real-time oriented implementation of the set-inversion algorithm, based on an sorted list, that gives usable results even if computation time is bounded. This allows to get a good characterization of the localization zone in less than one second.

GPS only location zone computation

Test data were recorded using a *Septentrio PolaRx* receiver and the experimental vehicle *Strada* of Fig. 6. The ground truth solution is provided by a post-processed *Trimble 5700* receiver with a local base. The sequence covers 1800 m of street and road near the lab in Compiègne, and lasts about 170 seconds. The trajectory is shown in Fig 7, where

Figure 7: Trajectory. *Boxes are bounding boxes of non robust solution sets. Reference trajectory is in black.*

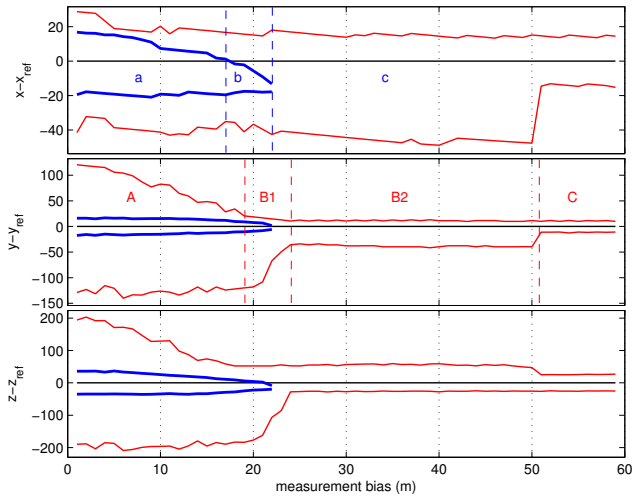
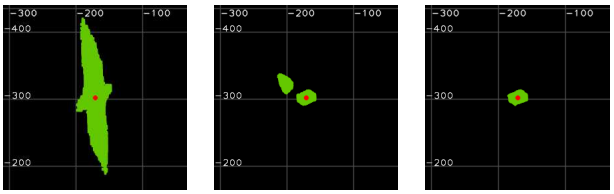


Figure 8: Location zone bounds with respect to ground truth. An increasing bias is applied to the first pseudorange. *Non robust set inversion in thick blue, robust set inversion in red*

the bounding boxes of solution sets for each measurement epoch are reprensented. These solutions are computed using a non robust SIVIA, with a contractor based on constraint propagation.

EGNOS augmentation system is used to get corrected pseudoranges, with associated measurement error variances, assuming an overbounding Gaussian distribution.

When using a non robust solver, inconsitency between



(a) no bias (b) 30 m bias (c) 50 m bias

Figure 9: XY projection of the 1-relaxed solution with one biased measurement

measurements can be detected when the solution set is empty. However, the measurement error can be too small to be detected, while being large enough to make the computed location zone inconsistent with ground truth.

In Fig. 8, a bias ramp is added to the first pseudorange of the set of six available pseudoranges. Non-robust solution set (blue lines) remains consistent with ground truth until the bias reaches 18 meters (a). With a measurement bias ranging from 18 to 23 meters (b), ground truth doesn't belong anymore to the solution set but there is no way to detect it. This is the weak point of the non-robust solver. Starting from a 24-meter bias, the solution set becomes empty (c), which proves inconsistency between measurements and the model.

Using a robust 1-relaxed solver, the presence of one wrong measurement does not compromise the integrity of the computed location zone. The evolution of location zone bounds with respect to the pseudorange bias in Fig. 8 (red lines) can be explained in three phases. From a 0 to 19 meter bias (A), a non-empty six-satellite solution can be computed. Since the solver is robust to one faulty measurement, all the five-satellite solutions are also included (Fig. 9a). The location zone gets tighter as the inconsistency grows. Starting from a 19-meter bias, the presence of an erroneous measurement can be detected, as the six-satellite solution is the empty set. The solution set is thus only made of five-satellite solutions (B1). Then only two five-satellite solutions are non empty (Fig. 9b), until a 50 meter bias (B2). Finally, when the bias exceeds 50 meters (C), only one five-satellite solution is non empty, thus excluding the faulty measurement from the location zone computation (Fig. 9c). If needed, the wrong measuremnt can be identified by testing compatibility of the six measurements with a box of the solution sub-paving.

Use of geographical information

When using a robust set inversion algorithm, the computed location zones tends to get wider. This is a direct consequence of considering the union of not well conditioned satellite subsets solutions. To counteract this phenomenon, more redundancy is needed. Since GPS satellites visibility can be reduced, especially in urban environment, other sources of information have to be used.

Altitude measurements can easily be obtained by the use of Digital Elevation Models (DEM) for ground applications, or altimeters for planes.

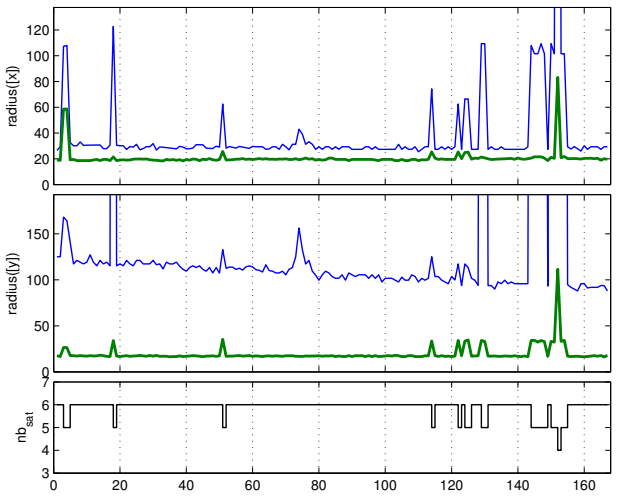


Figure 10: 1-relaxed location zone radius with *(thick green line)* and without *(blue line)* DEM information

Fusion with GNSS pseudorange measurements can be done either by setting the prior search box according to the measured altitude and its uncertainty, or by implementing a new constraint in the set-inversion.

Given a DEM with $\pm\Delta v$ vertical accuracy (altitude measurement) and $\pm\Delta h$ horizontal accuracy (planimetric error), a contraction operator can be defined (Alg. 2). It will be applied to each box in the set inversion algorithm, thus enforcing the altitude constraint.

Algorithm 2 DEMCONTRACT(in: $[\mathbf{x}_{in}]$, DEM; out: $[\mathbf{x}_{out}]$)

1: $[\mathbf{wnd}] = \begin{pmatrix} [x_{in}] \\ [y_{in}] \end{pmatrix} + \begin{pmatrix} [-\Delta h, +\Delta h] \\ [-\Delta h, +\Delta h] \end{pmatrix}$
2: $h_{min} = \min_{[\mathbf{wnd}]}(DEM) - \Delta v$
3: $h_{max} = \max_{[\mathbf{wnd}]}(DEM) + \Delta v$
4: $\mathbf{x}_{out} = \begin{pmatrix} [x_{out}] \\ [y_{out}] \\ [z_{out}] \end{pmatrix} = \begin{pmatrix} [x_{in}] \\ [y_{in}] \\ [z_{in}] \cap [h_{min}, h_{max}] \end{pmatrix}$

The DEM horizontal error is taken into account by searching for the altitude extrema in a larger area than the horizontal plane extent of the input box. Then, the vertical error interval of the DEM is added to the altitude amplitude found in the search zone. The output box is obtained by cropping the vertical component of the input box.

To improve efficiency, the DEM contraction step may be bypassed when the input box is very large (because of the huge amount of DEM data to process), and also when the input box is small compared to the DEM grid resolution. Another optimisation is to use a coarse model for large boxes, then to switch to the full precision model for smaller input boxes.

We use a 25 meter horizontal grid resolution DEM with 1 meter altitude precision covering the lab's neighbourhoud (*BD Topo* charted by the French Institut Géographique National).

In this trial, with the guarantee of immunity to one

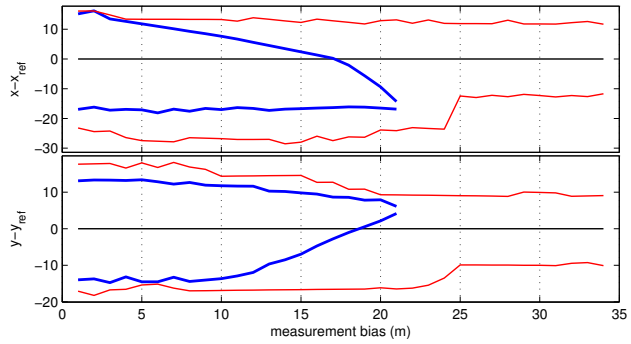


Figure 11: Location zone bounds with respect to ground truth. An increasing bias is applied on the first pseudorange. DEM information is used. *Non robust set inversion in thick blue, robust set inversion in red*

erroneous measurement (RSIVIA with $q = 1$), the altitude information from the DEM allows to reduce the horizontal location zone radius by a factor 5 on the y axis (Fig. 10). The reduction of horizontal location uncertainty is even greater in poor geometrical configurations, especially with five or less satellites. The altitude constraint enables robust snapshot localization with as low as four satellites.

The benefits of altitude information on horizontal uncertainty are especially noticeable when only few satellites are visible, or when robustness to a large number of faulty measurements is needed.

Fig. 11 shows the influence of a biased measurement on the horizontal position bounds, when merging DEM information with GPS pseudorange measurements. It is the same six-satellite dataset as in Fig. 8, allowing to compare results with and without the use of a DEM. While the non robust approach is not improved a lot by the use of a DEM, great improvement can be seen in the 1-relaxed solver behaviour. The location zone always remains consistent with ground truth, since there is only one erroneous measurement, but it is tighter thanks to the added constraint. The erroneous measurement is totally excluded from the solution after a 25 meter bias (where 50 meters were necessary without altitude information).

CONCLUSION

A method to characterize a location zone using robust set-membership solvers has been presented in this paper. Bounds are set on measurements, taking error model and risk into account. Those bounded-error measurements translate into constraints in the location domain. Using interval analysis, the constraints satisfaction problem can then be solved, and a defined number of constraints can even be discarded in order to keep solution integrity in the presence of outliers.

An experimental validation has been achieved to compute location zones, using GPS pseudorange measurements corrected with EGNOS, and a Digital Elevation Model. It has been implemented in real time as a parallelized C++ program. It showed that additional altitude information from the DEM enabled more precise positionning while

tolerating GPS outliers, especially with a small number of visible satellites.

Future work will be focused on dynamic car localization, using a kinematic model of vehicle. Embedded proprioceptive sensors will be used to constrain the location zone, especially during GPS outages in difficult environment such as urban canyons. In such areas, GPS measurements can be very limited and contaminated by non-line-of-sight (NLOS) reflected signals; a data horizon scheme will be experimented to deal with the small number of good GPS measurements. Three-dimensional maps of the navigable space will also be studied to a stronger constraint to the location zone, particularly when GPS geometrical satellite configuration does not enables a tight location zone characterization. It can be either a polyline representation, with altitude and road width attributes, or a full surface model of the navigable space. In both cases, the planimetric and altimetric errors will have to be considered.

ACKNOWLEDGMENTS

The dataset and the reference trajectory used in this work were provided by Clément Fouque (*Heudiasyc UMR 6599, Université de Technologie de Compiègne*).

REFERENCES

Brown, R. and Chin, G. (1997). GPS RAIM: calculation of thresholds and protection radius using Chi-square methods – a geometric approach. *Global Positioning System*, pages 155–179.

Drevelle, V. and Bonnifait, P. (2009). Integrity zone computation using interval analysis. In *European Navigation Conference GNSS*.

Feng, S., Ochieng, W., Walsh, D., and Ioannides, R. (2006). A measurement domain receiver autonomous integrity monitoring algorithm. *GPS Solutions*, 10(2):85–96.

Jaulin, L. (2009). Robust set-membership state estimation; application to underwater robotics. *Automatica*, 45(1):202–206.

Jaulin, L., Kieffer, M., Braems, I., and Walter, E. (2001a). Guaranteed nonlinear estimation using constraint propagation on sets. *International Journal of Control*, 74(18):1772–1782.

Jaulin, L., Kieffer, M., Didrit, O., and Walter, É. (2001b). *Applied Interval Analysis*. Springer-Verlag.

Jaulin, L., Kieffer, M., Walter, E., and Meizel, D. (2002). Guaranteed robust nonlinear estimation with application to robot localization. *IEEE Transactions on systems, man and cybernetics; Part C Applications and Reviews*, 32(4):374–382.

Jaulin, L. and Walter, E. (1993). Set inversion via interval analysis for nonlinear bounded-error estimation. *Automatica*, 29(4):1053–1064.

Kieffer, M., Jaulin, L., Walter, É., and Meizel, D. (2000). Robust autonomous robot localization using interval analysis. *Reliable Computing*, 6(3):337–362.

Lévêque, O. (1998). *Méthodes Ensemblistes pour la Localisation de Véhicules*. PhD thesis, Université de Technologie de Compiègne.

Meizel, D., Leveque, O., Jaulin, L., and Walter, E. (2002). Initial localization by set inversion. *IEEE Transactions on Robotics and Automation*, 18(6):966–971.

Walter, T. and Enge, P. (1995). Weighted RAIM for precision approach. In *Proceedings of ION GPS*, volume 8, pages 1995–2004. Institute of Navigation.

Waltz, D. L. (1972). *Generating Semantic Description from Drawings of Scenes with Shadows*. PhD thesis, Massachusetts Institute of Technology.