Interval-based Fault Detection and Identification applied to Global Positioning

Vincent Drevelle, Philippe Bonnifait

Interval-based Fault Detection and Identification applied to Global Positioning

Vincent Drevelle [*,†] **Philippe Bonnifait** [*,†]

[*] *Heudiasyc, UMR CNRS 7253*
[†] *Université de Technologie de Compiègne*

Abstract: We present a real-time robust positioning system based on interval analysis and constraint propagation that computes position domains, and that is able to detect and reject erroneous measurements. GPS pseudorange measurements are represented by intervals assumed to contain the true value with a given confidence level. A 3-D map of the drivable space is also available to constrain the vehicle location.
By the use of a breadth-first exploration strategy and of measurement consistency counters, the algorithm can be stopped at any time of the evaluation, and can instantly return the solution subpaving and the list of identified erroneous measurements. The method has been evaluated with real raw GPS data (i.e. pseudodistances on visible satellites) in a urban environment with a low-cost high-sensitivity GPS receiver providing faulty multipath measurements.

1. INTRODUCTION

In this paper, we aim at estimating a vector of parameters from a set of measurements and prior information, while detecting and identifying erroneous measurements. These erroneous measurements, also called outliers, are observations that cannot be explained by the observation model, either because they correspond to extreme noise values, or because they result of phenomenons or faults that have not been modeled.

We focus on a snapshot localization problem, where time dependency of position is not considered. Therefore, only an observation model is considered with no evolution model.

In urban environments, GPS positioning is likely to suffer from erroneous measurements. In the case of GPS, faults may arise from satellite or control segment failures. Augmentation systems (like WAAS and EGNOS) are available to protect users against such system failures. Another source of outliers is the alteration of signal propagation by the user's environment. Indeed, for time-of-flight measurement, signals are assumed to follow a direct path from the satellite to the receiver. However, especially in urban environments, when the direct path is blocked by an obstacle (like a building or a tree), the receiver may acquire a reflected ranging signal. It is called *non line-of-sight* propagation. This measurement is then an outlier, since the observation model is based on the hypothesis of direct propagation.

Fault detection and identification (FDI) methods are implemented in some GPS receivers, to provide receiver autonomous integrity monitoring (RAIM). They are often based on range residuals (Walter and Enge [1995]) or parity space (Ding [2008]). RAIM is used in aeronautical navigation, to detect and identify erroneous measurements, and compute protection levels, which are bounds on the positioning error that may result from undetected faults.

Interval based methods have successfully been applied to model based diagnosis, as they enable to take model uncertainties into account and can handle nonlinear systems. Marx et al. [2010] use interval observers to detect change of operation mode, and Adrot et al. [2002] propose a parity space approach where the parity matrix depends on uncertain parameters.

Set-membership methods have already been used to compute uncertainty in GPS networks with zonotopes by Schön and Kutterer [2006] and for interval based height-aided GPS localization (Drevelle and Bonnifait [2011b]) with fault-detection results consistent with standard RAIM methods. In this paper, we compute a position domain in which the user is located using an interval based method that is robust to the presence of erroneous GPS measurements. The method performance is increased thanks to the use of a 3D map of the drivable space. Along with the computation of the localization confidence domain, the algorithm enables to detect and identify the faulty measurements. Our contribution is to perform FDI within the set-inversion process, in a quickly interruptible algorithm thanks to the use of *measurement compatibility counters*.

The paper is organized as follows. Interval analysis methods for robust bounded-error estimation are first presented. Then, a way to detect and identify outliers from robust set-inversion results is shown, and an algorithm than enables quick interruption of computation with fault detection result is presented. Finally, experimental results are reported with raw GPS measurements.

2. ESTIMATION USING INTERVAL ANALYSIS

2.1 Interval analysis

Interval analysis (Jaulin et al. [2001]) involves intervals and their multidimensional extension, *interval vectors* (or *boxes*). In opposition to arbitrary sets, intervals and boxes

are easy to represent and manipulate. The set of real intervals $[x] = [\underline{x}, \overline{x}]$ is denoted $\mathbb{IR}$, and the set of n-dimensional boxes is $\mathbb{IR}^n$. In this paper, a box is denoted $[\mathbf{x}] = [\underline{\mathbf{x}}, \overline{\mathbf{x}}]$, where vectors $\underline{\mathbf{x}}$ and $\overline{\mathbf{x}}$ are respectively the lower and upper bounds of $[\mathbf{x}]$.

Interval arithmetic allows performing computations on intervals and boxes thanks to the interval extension of classical real arithmetic operators $+, -, \times$ and $\div$.

$$[x] + [y] = [\underline{x} + \underline{y}, \overline{x} + \overline{y}], \quad [x] - [y] = [\underline{x} - \overline{y}, \overline{x} - \underline{y}]$$

In the same way, elementary functions such as *tan, sin* and *exp* can be extended to intervals. This is done by returning the smallest interval covering the range of the input through the function.

The image of a box by a function $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^m$ is generally not itself a box, but an arbitrary set. This problem is solved using the so-called *inclusion functions*: The interval function $[\mathbf{f}]$ from $\mathbb{IR}^n$ to $\mathbb{IR}^m$ is an *inclusion function* for $\mathbf{f}$ if the image of $[\mathbf{x}]$ by $\mathbf{f}$ is included in the image of $[\mathbf{x}]$ by $[\mathbf{f}]$, *i.e.*

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

The *box hull* $\square S$ of a set S is the smallest box that includes S. Since the union of boxes is not generally a box, the *box union* operator $\sqcup$ returns the box hull of the union of two boxes: $[\mathbf{x}] \sqcup [\mathbf{y}] = \square([\mathbf{x}] \cup [\mathbf{y}])$.

2.2 Contraction and propagation

When the components of a vector x are linked by relations or constraints, one can define a *constraint satisfaction problem* (CSP). It consists in finding the solution set $X = \{\mathbf{x} \in [\mathbf{x}] | \mathbf{f}(\mathbf{x}) = \mathbf{0}\}$, where $[\mathbf{x}]$ is the domain of the variables and $\mathbf{f}(\mathbf{x}) = \mathbf{0}$ represents the constraints, and can also represent inequalities by introducing slack variables (Jaulin et al. [2001]).

A *contractor* $\mathcal{C}$ for a CSP is an operator that computes a smaller domain $[\mathbf{x}_c] = \mathcal{C}([\mathbf{x}])$ without affecting the solution set, *i.e.* $X \subset [\mathbf{x}_c] \subset [\mathbf{x}]$. There are many ways to implement a contractor, one of them is the forward-backward contractor based on constraint propagation (Benhamou et al. [1999]).

2.3 Set inversion and subpavings

The set inversion problem consists in determining the set X such as $\mathbf{f}(X) = [\mathbf{y}]$, where $[\mathbf{y}]$ is a known interval vector of m measurements. To approximate compact sets in a guaranteed way, *subpavings* can be used. A subpaving of a box $[\mathbf{x}]$ is the union of non-empty and non-overlapping subboxes of $[\mathbf{x}]$.

Using interval analysis, the solution $X = \mathbf{f}^{-1}([\mathbf{y}])$ can be approximated between two subpavings $\underline{X}$ and $\overline{X}$ such that $\underline{X} \subseteq X \subseteq \overline{X}$. The *SIVIA* algorithm allows performing such a set inversion, by recursively bisecting an initial box (Jaulin et al. [2001]).

Algorithm 1 implements a SIVIA that only computes an outer approximation $\overline{X}$ of the solution set in a given domain $[\mathbf{x}_0]$, since we are seeking to characterize the positioning confidence domain. It uses a list of boxes $\mathfrak{L}$ managed by the *push* and *pull* functions. If $\mathfrak{L}$ is a queue, the algorithm employs a breadth-first strategy. ε controls the sharpness of the subpaving $\overline{X}$. A contractor $\mathcal{C}([\mathbf{x}])$ is used to apply the constraint from measurements on each box. It may either reduce the size of a box without losing any solution, or return an empty box if the initial box is incompatible with the measurements. Boxes larger that ε after contraction are bisected and enqueued to be processed again.

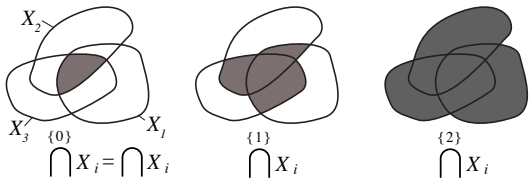


Fig. 1. q-relaxed intersection of three sets

Algorithm 1 SIVIA(in: $[\mathbf{x}_0], \mathcal{C}, \varepsilon$)

$\overline{X} := \emptyset$ *// empty subpaving*
push($[\mathbf{x}_0], \mathfrak{L}$)
while $\mathfrak{L}$ is not empty
  $[\mathbf{x}] :=$ pull($\mathfrak{L}$)
  $[\mathbf{x}] := \mathcal{C}([\mathbf{x}])$ *// contract the box*
  if width($[\mathbf{x}]$) $< \varepsilon$ then
    $\overline{X} := \overline{X} \cup [\mathbf{x}]$
  else if $[\mathbf{x}] \neq \emptyset$ then
    $([\mathbf{x}_1], [\mathbf{x}_2]) :=$ bisect($[\mathbf{x}]$)
    push($[\mathbf{x}_1], \mathfrak{L}$); push($[\mathbf{x}_2], \mathfrak{L}$)
  endif
end
return $\overline{X}$

2.4 Robust set-inversion

The presented set-inversion method is likely to return an empty solution set in the presence of erroneous measurements. Robustness to outliers can be achieved by computing the set of solutions that are compatible with *at least a given number of measurements* instead of the set of solutions compatible with all the measurements.

By using the *q-relaxed intersection* of m measurement constraints (Fig. 1), a set of solutions compatible with at least $m - q$ measurements is computed.

If a contractor $\mathcal{C}_i$ is available for every measurement, then a q-relaxed contractor $\mathcal{C}_{\{q\}}$ can be built. It is presented in Algorithm 2.

Figure 2 shows the results of the presented SIVIA algorithm with the q-relaxed contractor. Three constraints are provided by ranging beacons, one of which is faulty and gives an erroneous distance measurement. SIVIA is employed with the $\mathcal{C}_{\{1\}}$ contractor to compute an outer approximation of the 1-relaxed intersection of the three constraints. The obtained solution subpaving is consistent with ground truth.

Algorithm 2 $\mathcal{C}_{\{q\}}$: q-relaxed contraction (m measures)

for $i = 1$ **to** m { $[\mathbf{x}^i] := \mathcal{C}_i([\mathbf{x}])$ }

$$[\mathbf{x}_c] := \square \left(\bigcap_{i \in \{1,\ldots,m\}}^{\{q\}} [\mathbf{x}^i] \right)$$

return $[\mathbf{x}_c]$

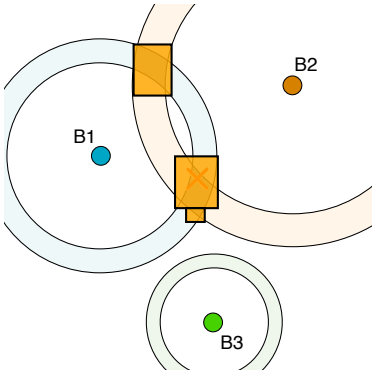


Fig. 2. Results of a 1-relaxed set inversion in a localization problem with 3 ranging beacons. The measurement from beacon B3 is erroneous. The solution subpaving covers the true solution-set.

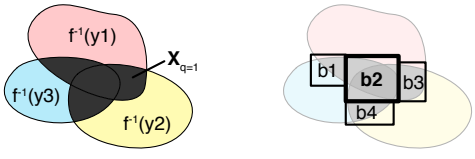
3. FAULT DETECTION AND IDENTIFICATION

The principle of the method is to compute a solution set and then to analyze the compatibility of the measurements with this set.

In practice, it is necessary to compute a solution with robustness. Otherwise, the solution can often be empty, which prevents fault identification. The robust method that we use in this work is the q-relaxed set inversion. It enables computing a solution set robust to a specified number of outliers.

Afterward, it is possible to try to identify the faulty measurements. Indeed, uncertainty affecting the measurements makes the identification not always possible. The method that we propose is to check if the boxes of the solution subpaving are compatible with each measurement.

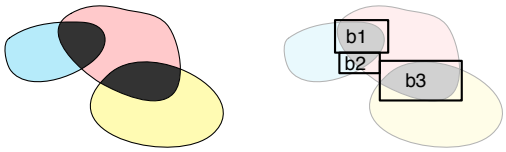
For each measurement, an inclusion test is done for every boxes of the subpaving. Detection and identification can be obtained using Boolean tests. The presence of an outlier is detected when the subpaving does not contain a box that can satisfy all the constraints. Outlier detection is performed when there is not any box than can satisfy the constraint of a given measurement. For instance, at the beginning of the process, all measurements are marked as outliers. As soon as a box compatible with the measurement is found, the measurement is unmarked. Once the whole subpaving has been tested, the measurements that remain marked are identified as outliers.



Box \ Measurement	$[y_1]$	$[y_2]$	$[y_3]$	$[\mathbf{y}]$
b1	1	0	1	0
b2	1	1	1	1
b3	1	1	0	0
b4	0	1	1	0
Result	1	1	1	1

Fig. 3. Detection and identification table with no erroneous measurement. The final row describes if the measurements are compatible with the computed subpaving.

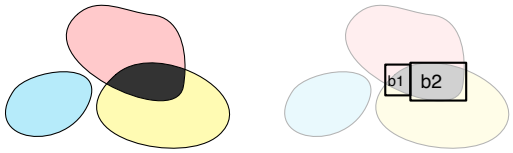
In Fig. 3, the compatibility of a box with $[\mathbf{y}]$ (last column) is computed using a logical AND between the elements of the corresponding row. For instance, box $b2$ is compatible with the three measurement intervals and box $b1$ is compatible with the single measurements $[y_1]$ and $[y_3]$, but not with $[y_2]$ nor all the measurements together $[\mathbf{y}]$. The results for the solution subpaving are obtained by applying a logical OR between the elements of each column. Since the lower-right element equals one, no erroneous measurement is detected.



Box \ Meas.	$[y_1]$	$[y_2]$	$[y_3]$	$[\mathbf{y}]$
b1	1	0	1	0
b2	1	0	1	0
b3	1	1	0	0
Result	1	1	1	0 *detection*

Fig. 4. Detection and identification table with an outlier detected.

In Fig. 4, there is no box in the subpaving that can fulfill the constraints of all the measurements since the lower-right element equals zero. The presence of an outlier is thus detected. However, since there is at least a box compatible with each measurement in the solution subpaving, it is not possible to identify the faulty measurement. The detection and identification table contains also extra information. For instance, if there is no more than one outlier at a time, one can conclude that the faulty measurement is either $[y_2]$ or $[y_3]$.



Box \ Meas.	$[y_1]$	$[y_2]$	$[y_3]$	$[\mathbf{y}]$
b1	1	1	0	0
b2	1	1	0	0
Result	1	1	0 *identification*	0 *detection*

Fig. 5. Outlier identified: $[y_3]$

In Fig. 5, the presence of an erroneous measurement is detected like in Fig. 4. Moreover, the outlier can be identified in this particular case. Indeed, there is no box in the subpaving that can satisfy the constraint from measurement $[y_3]$. The latter is thus identified as the faulty measurement. The identification is guaranteed as long as the number of true outliers remains lower or equal to the degree "q" of the q-relaxation.

4. OPTIMIZED IMPLEMENTATION

One of the main drawbacks of the previous method is that the processing is long to interrupt, since a post-processing has to be performed on the solution subpaving

to carry out fault identification. It is unfortunate, as the set inversion process can be interrupted at any time to quickly provide an approximate solution. The ability to provide a solution with a reduced and guaranteed latency is a basic requirement for real-time systems.

We present in this section a way to prepare as much as possible the fault identification process during the exploration of the solution space. The idea is to use *measurement compatibility counters*. As it slows down each iteration of the set inversion algorithm, this implementation reduces the number of bisections per second. However, as the bisection loop terminates, only few quick operations are needed to get the solution subpaving and the list of identified outliers.

RSIVIAEXTCOUNT (Alg. 3) computes a subpaving that covers the q-relaxed solution, and, in the meantime, keeps track of the compatibility of the subpaving with the measurements through the use of counters. Set-inversion can be interrupted at any time, and outlier detection and identification can then be done by checking the counters values. To reduce the number of inclusion tests, each box is augmented with a compatibility bit-field, which indicates which measurements are compatible with it.

When there is no solution (i.e. an empty set), there are more outliers in the measurements than q, which is the maximum number anticipated. Each counter equals zero in this case. If one wants to get a non empty result, it is possible to restart the estimation process with more relaxation (*e.g.* $q := q + 1$). This strategy is known as GOMNE (Guaranteed Outlier Minimal Number Estimator) presented in Jaulin et al. [1996].

Otherwise, the values of the counters are examined to check the presence of outliers. If the global counter c_{all} is equal to zero, then the presence of an outlier is detected, since there is no box in the solution that is compatible with all the measurements. One can notice that c_{all} corresponds to the lower-right element of a detection and identification table (see Fig. 4). It is then possible to check if an outlier has been identified, simply by studying the values of individual measurement compatibility counters c_i, which keep track of the number of boxes in the subpaving that are compatible with each measurement. If the counter c_i is null, then the measurement y_i is identified as an outlier, like in the last row of a detection and identification table (see Fig. 5). Figure 6 shows the counter values and outlier detection results after a 1-relaxed set inversion on simple examples.

This implementation of fault detection and identification inline with set-inversion not only enables reducing the latency when the computation is stopped, but also monitoring outliers during the computation. This is another advantage of this approach since it provides valuable information for online optimization of the set-inversion computation. Indeed, once a measurement has been identified as an outlier, computing a q-relaxed solution set with all the measurements is equivalent to compute a $(q-1)$-relaxed solution with all the measurements but the identified outlier. The robust set inversion algorithm can therefore dynamically discard measurements as they are identified as outliers and thus speed up the computation by reducing the number of constraints to satisfy.

Algorithm 3 RSIVIAEXTCOUNT
(in: $[\mathbf{x}], \{\mathcal{C}_i, i = 1 \dots m\}, [\mathbf{f}], [\mathbf{y}], q, \varepsilon$; out: $\overline{\mathbb{S}}_{\mathrm{x}}$)

```
// Initialize measurements compatibility counters
for i = 1 to m {
    c_i := 1
    c_all := 1
    [x].c_i := 1
}
L := {[x]}
S̄_x := ∅
while L ≠ ∅ and not timeout {
    // Pick a box from the list
    [x] := pull(L)
    // q-relaxed contraction with measurements
    for i = 1 to m {
    .   [x^i] := C_i([x])
    }
    [x_c] := □ ( ⋂_{i∈{1,...,m}}^{{q}} [x^i] )
    // Update inclusion-test results bit-field
    for i = 1 to m {
    .   if [f_i]([x_c]) ∩ [y_i] ≠ ∅ then [x_c].c_i := 1
    .   else [x_c].c_i := 0
    }
    // Update counters
    if [x_c] ≠ ∅ then {
    .   if w([x_c]) < ε then {
    .   .   // Box is too small
    .   .   S̄_x := S̄_x ∪ [x_c]
    .   .   // Update compatibility counters
    .   .   for i = 1 to m {
    .   .   .   c_i := c_i + [x_c].c_i − [x].c_i
    .   .   .   c_all := c_all + ∏_i [x_c].c_i − ∏_i [x].c_i
    .   .   }
    .   }
    .   else {
    .   .   // Bisect and enqueue suboxes
    .   .   ([x_1], [x_2]) = bisect([x_c])
    .   .   push([x_1], L); push([x_2], L)
    .   .   // Update compatibility counters (2 boxes)
    .   .   for i = 1 to m {
    .   .   .   c_i := c_i + 2 · [x_c].c_i − [x].c_i
    .   .   .   c_all := c_all + 2∏_i [x_c].c_i − ∏_i [x].c_i
    .   .   }
    .   }
    }
    else {
    .   for i = 1 to m {
    .   .   c_i := c_i − [x].c_i
    .   .   c_all := c_all − ∏_i [x]c_i
    .   }
    }
}
// Compute solution set
S̄_x := S̄_x ∪ L
if c_all = 0 then outlier detected
for i = 1 to m {
    if c_i = 0 then [y_i] is an outlier
}
```

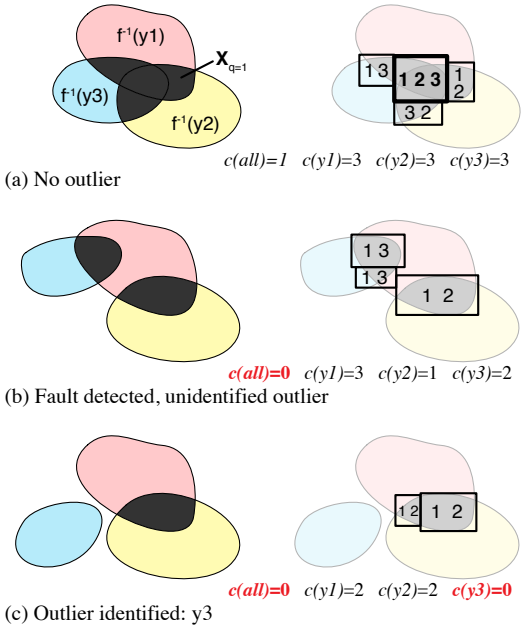


Fig. 6. Outlier detection and identification with compatibility counters

5. EXPERIMENTAL RESULTS

In this section, we report experimental GPS positioning results obtained with a real-time implementation of the algorithm implemented in C++. Raw time-of-flight measurements (pseudoranges) from a low-cost high-sensitivity *uBlox LEA4T* GPS receiver are used to constrain the user position. Measurement noise is taken into account by representing pseudoranges with intervals. A 3D map of the drivable space (provided by the french *Institut Geographique National*) is also used a constraint on position. Positioning is done with the previously presented robust set-inversion algorithm. The contractors for GPS and 3D map are respectively based on interval constraint propagation and on a polygon clipping algorithm, and can be found in Drevelle and Bonnifait [2011a].

The dataset consists in three loops around the *mairie* of the 12$^{\text{th}}$ arrondissement in Paris, covering 3 kilometers in 16 minutes (Fig. 7). It is very challenging for GPS since the narrow streets with high buildings prevent good reception of the satellites' signals, and are source of erroneous measurements due to multipath and non-line-of-sight signal propagation. A common practice to reduce the number of erroneous GPS measurements is to filter out observations whose signal-to-noise ratio (SNR) is too low. In order to test the presented algorithm in the presence of a lot of outliers, we set the SNR filtering threshold to a very low value of 20 dBHz (the standard being at 37 dBHz). Pseudorange measurements were modeled by ± 3 m intervals.

The robust set-inversion algorithm used in this experiment implements the GOMNE strategy: computation starts with no constraint relaxation, and the number q of relaxed GPS measurements is increased until a non-empty solution is found. Since an empty solution-set with a q-relaxed set inversion indicates that there are at least $q+1$ outliers in the measurements vector, the algorithm is able to

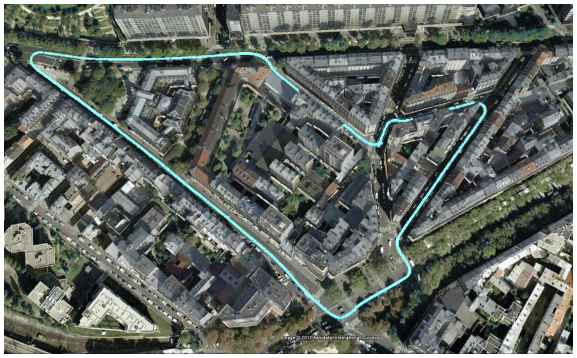


Fig. 7. Path followed around the *mairie* of the 12$^{\text{th}}$ arrondissement in Paris.

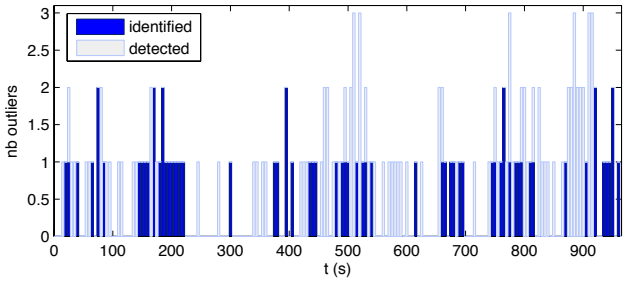


Fig. 8. Number of detected and identified outliers at each epoch.

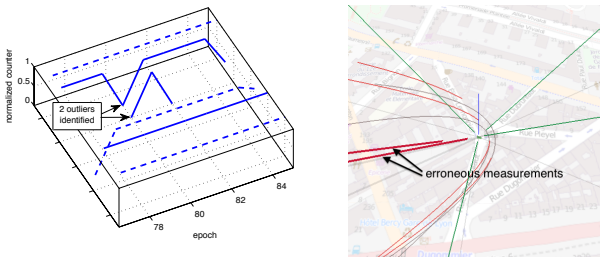


Fig. 9. Two outliers identified at epoch #80. Left: measurement compatibility counters values (normalized by the number of boxes). Right: 3D view of solution. The solution subpaving is in green. Green lines are lines-of-sight to the used satellites. Thick red lines represent identified outliers.

report a lower bound of the number of outliers. This is the number of detected outliers shown in Fig. 8. Outliers identification is performed with the compatibility counters presented in the previous section. Since the use of intervals to characterize the solution as a subpaving, the results are dependent on the coordinate system. Computations are done in a "East, North, Up" local tangent frame, that enables very efficient altitude contraction with the map constraint.

Fig. 8 shows that the algorithm detects at most 3 simultaneous outliers during the trial, and is able to identify at most two outliers. This is illustrated by Fig. 9, where two outliers are successfully detected and identified at epoch #80 (t=400 s).

Due to the lack of redundancy, fault detection may fail, and even lead to wrong exclusion. This happens at epoch #89

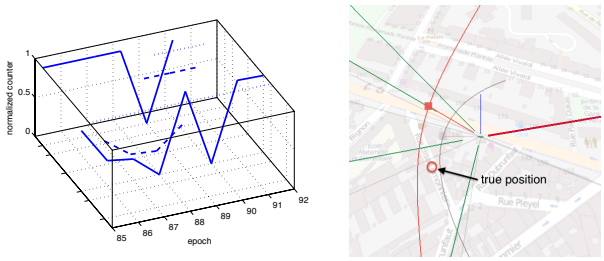


Fig. 10. Fault detection and identification failure at epoch #89. The circle represents ground truth position.

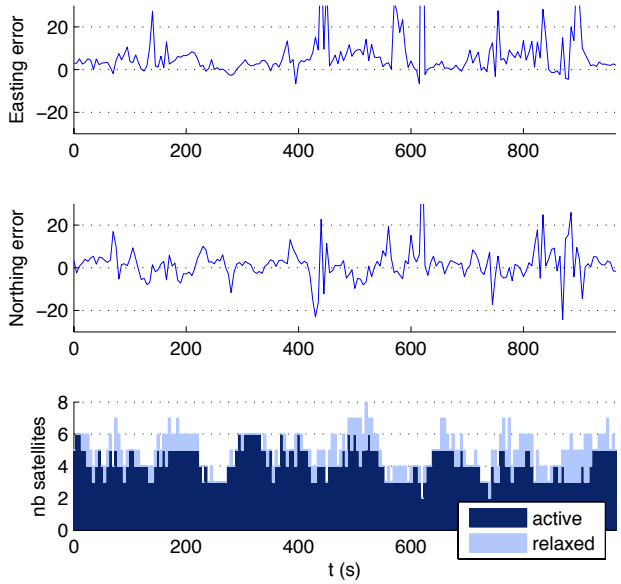


Fig. 11. Northing and easting error (in meters) when using the subpaving's center of gravity as a position estimate. The lower row shows the number of active and relaxed GPS constraints (number of detected faults) at each epoch.

(t=445 s) where a single fault is detected and identified with the measurements compatibility counters, but leads to an erroneous solution set (Fig. 10). In fact, at that particular epoch, four out of the five received signals have very low SNRs, and even the remaining stronger signal would have normally been filtered out by a standard SNR threshold of 37 dBHz. With 4 wrong measurements out of 5, it is not possible to compute a solution consistent with the ground truth.

The center of gravity of the solution subpaving can be used as point estimate for position. Fig 11 represents positioning error in the north and east coordinates, with respect to the position solution of an *Applanix* inertial navigation system. The mean horizontal positioning error during the whole trial is 9.47 m, with a median of 5.78 m. Large errors in Fig 11 are mainly due to the lack of measurements (*e.g.* at t=630 s) which leads to disconnected solution sets representing multiple localization hypotheses. Another source of error is the presence of too many outliers at particular epochs since the SNR threshold for measurement rejection was set very low (*e.g.* at t=445 s).

6. CONCLUSION

An interval-based position estimation method with fault detection and identification has been presented. It enables computing a localization domain from a few GPS measurements with outliers and a 3D map. An optimized implementation with measurement compatibility counters has been introduced, and tested on a sequence of real GPS data in a very difficult urban environment. The method is able to compute a position and to successfully detect and identify outliers when GPS measurement redundancy is high enough. However, it may fail to detect a fault when there are not enough GPS measurements (at least 3 satellites are required, then detection performance depends if the error moves the computed position out of the drivable space or not), or identify a good measurement as a fault when there are more erroneous measurements than good observations.

Improvements to the fault detection capability of the method could be obtained by the computation of an inner solution subpaving along with the outer subpaving, or by checking particular points like box centers. Finally, this paper has proposed a purely static approach ; A dynamic approach could improve the overall performance by taking time dependency into account.

REFERENCES

O. Adrot, H. Janati-Idrissi, and D. Maquin. Fault detection based on interval analysis. In *15th IFAC World Congress on Automatic Control*, page CDROM, Barcelonne, Espagne, July 2002.

F. Benhamou, F. Goualard, L. Granvilliers, and J.-F. Puget. Revising hull and box consistency. In *Int. Conf. on Logic Programming*, pages 230–244. MIT press, 1999.

S.X. Ding. *Model-based fault diagnosis techniques: design schemes, algorithms, and tools*. Springer Verlag, 2008.

V. Drevelle and P. Bonnifait. Global positioning in urban areas with 3-D maps. In *IEEE Intelligent Vehicle Symposium*, pages 764–769, Baden-Baden, Germany, 2011a.

Vincent Drevelle and Philippe Bonnifait. A set-membership approach for high integrity height-aided satellite positioning. *GPS Solutions*, 15(4):357–368, 2011b. doi: 10.1007/s10291-010-0195-3.

L. Jaulin, É. Walter, and O. Didrit. Guaranteed robust nonlinear parameter bounding. In *CESA'96 IMACS Multiconference (Symposium on Modelling, Analysis and Simulation)*, volume 2, pages 1156–1161, Lille, 1996.

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

B. Marx, D. Maquin, and Ragot J. State estimation and diagnosis of uncertain system based on an interval approach. In *Conference on Control and Fault-Tolerant Systems, SysTol'10*, Nice, France, October 6-8 2010.

S. Schön and H. Kutterer. Uncertainty in gps networks due to remaining systematic errors: the interval approach. *Journal of Geodesy*, 80(3):150–162, 2006.

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