



iGPS: Global Positioning in Urban Canyons with Road Surface Maps

Vincent Drevelle, Philippe Bonnifait

► To cite this version:

Vincent Drevelle, Philippe Bonnifait. iGPS: Global Positioning in Urban Canyons with Road Surface Maps. IEEE Intelligent Transportation Systems Magazine, 2012, 4 (3), pp.6-18. 10.1109/MITS.2012.2203222 . hal-00805174

HAL Id: hal-00805174

<https://hal.science/hal-00805174>

Submitted on 27 Mar 2013

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.

iGPS: Global Positioning in Urban Canyons with Road Surface Maps

Vincent Drevelle and Philippe Bonnifait

Abstract—Interval Global Positioning with road Surface (iGPS) is a new method to obtain a robust and continuous positioning in urban areas by tightly-coupling precise 3-D drivable area maps with GPS pseudorange measurements. Map and GPS measurements are represented by geometric constraints, thus turning the localization problem into a constraint satisfaction problem whose solution is the confidence domain of position. Interval analysis is employed to solve the problem by using contractions and bisections of a prior position box. If more than 3 satellites are visible, the method is robust to erroneous pseudorange measurements. The system is also able to compute multiple position hypotheses where there are ambiguities. An experimental validation using real GPS pseudorange measurements and a precise 3-D map is reported to illustrate the performance of the method with real data in an urban area, and with reduced satellite visibility. Confidence domains are consistent with the ground truth throughout the 1 km trial, and a 6.5 m 95% accuracy is achieved with at least two satellites in view.

I. INTRODUCTION

Global Positioning Systems (GPS) can be used anywhere in the world and can provide highly accurate localization in open areas. Real time kinematic (RTK) offers centimeter-level positioning using phase measurements, but implies the use of a base station, and good satellite visibility. When these conditions are met, GPS can be sufficient to control autonomous vehicles in path-following tasks [1].

In urban environments, GPS signals are frequently blocked by surrounding buildings, and even reflected, causing multipath and non-line-of-sight measurement errors. This is incompatible with urban navigation tasks that require decimeter-level and high continuity positioning to keep vehicles within traffic lanes. Methods using exteroceptive sensors must therefore be used to control the vehicle trajectory and to perform obstacle avoidance either in a collaborative way with the infrastructure [2] or fully autonomously with embedded sensors [3].

However, even though GPS is not precise enough to perform robot control tasks, a robust global positioning method able to quantify position uncertainty may still be needed, in order to

- initialize a local positioning system (e.g. to limit the area in where initial particles of a particle filter will be generated, or to narrow the set of visual features to be searched and tracked by a vision based algorithm).
- perform integrity checks on the local positioning system: a reliable global positioning method can fire an alert if the vehicle drifts away from its planned trajectory.

Vincent Drevelle and Philippe Bonnifait are with Heudiasyc, CNRS UMR 7253, Université de Technologie de Compiègne, BP 20529, 60205 Compiègne Cedex, France, e-mail: vincent.drevelle@hds.utc.fr.



Figure 1: Heudiasyc's experimental vehicle: CARMEN in front of the 12th arrondissement *mairie* in Paris

- carry out high-level topological navigation: while reactive control is performed with local exteroceptive sensors such as cameras and LIDAR, navigation choices (e.g. path planning, turning at intersections, respecting the speed limit, stopping at the destination) require reliable global positioning information.

To counter GPS's poor precision and uncertain availability in urban environments, GPS is usually aided by inertial measurement units. For road vehicles, geographic information can be used as well. Much research has been done with standard maps [4], [5], but precise digital surface models [6] and lane-level maps [7] are now available to provide strong positional constraints.

Fusion of GPS measurements with geographical information for map-aided positioning has been an active area of research,

either by using the map to correct raw GPS measurements, like in the Road Reduction Filter [8], or by adding a constraint stage after the update of a Kalman filter, like the two-step projection method proposed in [9]. Particle filtering has also been studied to address the problem of positioning with a map constraint, like in [10] with a facet-based representation of the drivable space, in [11] with clothoidal surfaces and a representation of the connections between parallel driving lanes, and more recently in [12] where particles are weighted with respect to vehicle speed and functional road class, as well as map topology. A drawback of particle filtering is the risk of losing a solution if the number particles is not sufficient to evenly cover the position space. Moreover, strong constraints may lead to the death of a lot of particles, and the solution may tend to condensate on the edges of the road constraint.

In this paper we present Interval Global Positioning with Road Surface (iGPS), a robust approach for solving the positioning problem in known urban environments with geometric constraints. For this purpose, GNSS pseudorange measurements are tightly combined with a precise 3-D road surface model. Unlike Bayesian or optimization methods, which compute a point position estimate and quantify the error distribution around this point, the proposed method formulates the problem as a constraint satisfaction problem (CSP). Using interval analysis and contractors, the positioning confidence domain is computed as the set of positions compatible with all the constraints. It can even be a disconnected solution set, in the case of positioning ambiguity.

The paper is organized as follows. After an overview of interval analysis and contractors, a solver for constraint satisfaction problems (CSP) is introduced. The 3-D road constraint and GPS constraint are presented. A constraint for robust GPS positioning is also introduced. Finally, experimental results of iGPS with real GPS data and 3-D map are presented.

II. INTERVAL ANALYSIS AND CONTRACTORS

A. Interval analysis

Interval analysis [13] 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 is denoted \mathbb{IR} , and the set of n -dimensional boxes is \mathbb{IR}^n . In this paper, an interval or a box is denoted $[x] = [\underline{x}, \bar{x}]$, where vectors \underline{x} and \bar{x} are respectively the lower and upper bounds of $[x]$.

Interval arithmetic extends the use of classical real arithmetic operators $+$, $-$, \times and \div to intervals.

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

Similarly, 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 $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ 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 f if the image of $[x]$ by f is included in the image of $[x]$ by \mathbf{f} , i.e.

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

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: $[x] \sqcup [y] = \square([x] \cup [y])$.

B. Contraction and propagation

When the components of a vector x are linked by relations or constraints, a *constraint satisfaction problem* (CSP) can be defined. This consists in finding the solution set $X = \{x \in [x] | f(x) = 0\}$, where $[x]$ is the domain of the variables and $f(x) = 0$ represents the constraints, and can also represent inequalities by introducing slack variables [13].

A *contractor* \mathcal{C} for a CSP is an operator that computes a smaller domain $[x_c] = \mathcal{C}([x])$ without affecting the solution set, i.e. $X \subset [x_c] \subset [x]$. There are many ways to implement a contractor, one of which is the forward-backward contractor based on constraint propagation [14].

C. Set inversion and subpavings

The set inversion problem consists in determining the set X such that $f(X) = [y]$, where $[y]$ 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 $[x]$ is the union of non-empty and non-overlapping subboxes of $[x]$.

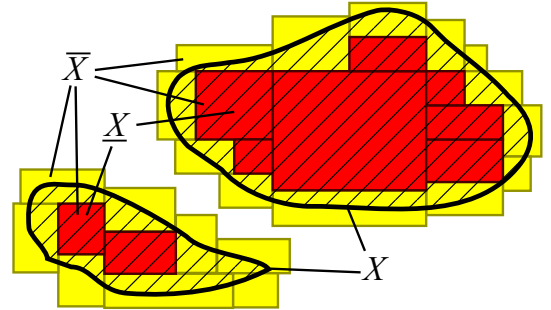


Figure 2: Bracketing of a disconnected set X between two subpavings \underline{X} (inner approximation) and \bar{X} (outer approximation).

Using interval analysis, the solution $X = f^{-1}([y])$ can be approximated between two subpavings \underline{X} and \bar{X} such that $\underline{X} \subseteq X \subseteq \bar{X}$ (Fig. 2). The SIVIA algorithm performs this type of set inversion by recursively bisecting an initial box [13].

Algorithm 1 implements a SIVIA that only computes an outer approximation \bar{X} of the solution set in a given domain $[x_0]$. It uses a list of boxes \mathcal{L} managed by the functions *push* and *pull*. If \mathcal{L} is a queue, the algorithm employs a breadth-first strategy. ϵ controls the sharpness of the subpaving \bar{X} . Boxes larger than ϵ after contraction are bisected.

Since we are seeking to characterize the positioning confidence domain, we only need to compute the outer subpaving \bar{X} of the set that fulfills positioning constraints. In this paper, we present a new kind of constraint based on 3-D drivable space.

Algorithm 1 SIVIA(in: $[x_0], \epsilon$)

```

1:  $\bar{X} := \emptyset$  // empty subpaving
2: push( $[x_0], \mathcal{L}$ )
3: while  $\mathcal{L}$  is not empty do
4:    $[x] := \text{pull}(\mathcal{L})$ 
5:    $[x] := \mathcal{C}([x])$  // contract the box
6:   if width( $[x]$ )  $< \epsilon$  then
7:      $\bar{X} := \bar{X} \cup [x]$ 
8:   else if  $[x] \neq \emptyset$  then
9:      $([x_1], [x_2]) := \text{bisect}([x])$ 
10:    push( $[x_1], \mathcal{L}$ ); push( $[x_2], \mathcal{L}$ )
11:   end if
12: end while
13: return  $\bar{X}$ 

```

III. 3-D ROAD CONSTRAINTS

In urban areas, a strong constraint on position is provided by the road network. 3-D roads can be used for contracting not only the East and North estimates through the use of road boundaries, but also altitude.

A. Drivable space

In this paper, the *drivable space* denotes the area in which the vehicle may move. For a car, the drivable space can be defined as the surface of the roadway, delimited by the curbs. Obstacles like poles or lane separators are also excluded from the drivable space. Finally, since we consider the position at the center of the vehicle, erosion is performed on the drivable space to take the size of the vehicle into account.

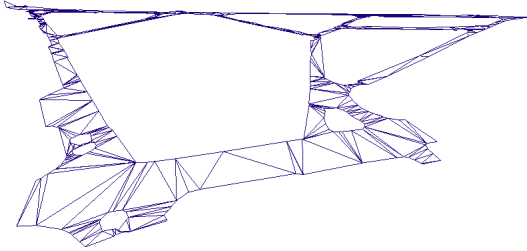


Figure 3: Wireframe view of the drivable space extracted from a database

The drivable space is represented in 3D by a triangular mesh, i.e. a surface made up of connected triangular facets (Fig. 3). Vertices are represented by their 3-D coordinates, while facets are defined by a list of three vertices. We assume that the vehicle moves only within the represented drivable space, which provides a very strong constraint on the position.

B. Positioning constraint on a single facet

Being located on a 3D triangular face can be expressed by four simple constraints: one constraint representing the facet plane, and three constraints for the edges.

Let us consider the facet of Fig. 4. The first constraint is given simply by the plane equation obtained from the 3

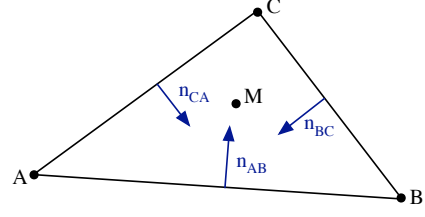


Figure 4: Facet ABC with its edge normals

vertices. The point M is on the plane defined by the points A, B, C , and therefore satisfies

$$(\overrightarrow{AB} \wedge \overrightarrow{AC}) \cdot \overrightarrow{AM} = 0 \quad (1)$$

Edge constraints can be expressed by enforcing the positivity of the dot products of vertex-to-point vectors and the corresponding edge normal vectors:

$$\begin{aligned} \overrightarrow{AM} \cdot \overrightarrow{n_{AB}} &\geq 0 \\ \overrightarrow{BM} \cdot \overrightarrow{n_{BC}} &\geq 0 \\ \overrightarrow{CM} \cdot \overrightarrow{n_{CA}} &\geq 0 \end{aligned} \quad (2)$$

Given a valid local tangent frame, the dot-product inequalities can be simplified by working only with 2D coordinates (assuming the road is never vertical). Moreover, if facets are small enough with respect to altitude variation, constraining the position inside the bounding box of the facet is sufficient to approximate the facet plane constraint (1). Algorithm 2 presents this contractor: initial contraction is performed with the bounding box of the facet, then dot-product constraints (2) are enforced simply by an inclusion test (line 6 of Alg. 2). If there is no point M in the box that satisfies (2), then the box is discarded. In this algorithm, interval vectors $[a], [b], [c]$ are the coordinates of the considered facet vertices.

Algorithm 2 facet_contract(in: $[x], [a], [b], [c]$)

```

1:  $[bbox] := [a] \sqcup [b] \sqcup [c]$ 
2:  $[x] := [x] \cap [bbox]$ 
3: if  $[x] = \emptyset$  then
4:   return ( $[x]$ )
5: end if
6: if  $([x_{1:2}] - [a_{1:2}]) \cdot \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} ([b_{1:2}] - [a_{1:2}]) \cap [0, +\infty] = \emptyset$ 
   or  $([x_{1:2}] - [b_{1:2}]) \cdot \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} ([c_{1:2}] - [b_{1:2}]) \cap [0, +\infty] = \emptyset$ 
   or  $([x_{1:2}] - [c_{1:2}]) \cdot \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} ([a_{1:2}] - [c_{1:2}]) \cap [0, +\infty] = \emptyset$  then
7:   return ( $\emptyset$ )
8: end if
9: return ( $[x]$ )

```

A more efficient contractor could be obtained with the Sutherland-Hodgman polygon clipping algorithm [15] used in computer graphics. Indeed, contracting a box $[x]$ with a facet ABC could be seen as clipping the facet with the box $[x]$, and then computing the bounding box of the resulting polygon. Nevertheless, the uncertainty of the map points is much more difficult to handle, particularly when the uncertainty is not the same for all the vertices.

C. Positioning constraint on the road surface

The whole road constraint is simply the union of the constraints from each facet of the map \mathcal{M} . Since the map may cover a very large area, the number of facets used in the road constraint may become very large too. However, it is not necessary to consider facets that lie outside the prior position box. A facet selection step is thus performed before computing the contracted boxes with each facet.

Algorithm 3 contracts a box with the entire road. The *extract_facets* function returns the set of facets whose bounding box intersects the prior box; this enables us to focus only on the interesting part of the map and to reduce computation time.

Algorithm 3 *road_contract*(in: $[x], \mathcal{M}$)

```

1:  $[xc] := \emptyset$ 
2:  $\mathcal{F} := \text{extract\_facets}(\mathcal{M}, [x])$ 
3: for each  $f$  in  $\mathcal{F}$  do
4:    $[xc] := [xc] \sqcup \text{facet\_contract}([x], f.[a], f.[b], f.[c])$ 
5: end for
6: return  $[xc]$ 

```

D. Taking topology into account

When a previous position is known, road topology can be used to reduce the initial facet candidates. Since the road network is very dense in urban areas, a basic facet selection method (like in section III-C) can select facets that are not reachable from the current location. Indeed, a facet which is not far from the current position, as the crow flies, can actually be unreachable if we consider that the vehicle has to follow the roads.

Let $[x_{k|k}]$ be the position at time t_k , $[x_{k+1|k}]$ the predicted location zone at time t_{k+1} , and s the distance traveled (possibly measured using an odometer) between t_k and t_{k+1} .

The first step is to bound the zone $[z]$ in which the vehicle can possibly have moved between t_k and t_{k+1} . It is straightforward to get a rough estimate for each component:

$$[z] = \frac{1}{2} ([x_{k|k}] + [x_{k+1|k}] + [-s, s]) \quad (3)$$

The physical interpretation of (3) is that the traveled distance constraint is applied independently on each dimension of $[x]$. This can be simplified when only travelled distance information is used for prediction. In this case, the vehicle evolution area is equal to the predicted position (Fig 5).

The facets that intersect $[z]$ are extracted from the map, to get the submap \mathcal{M}_z . Facets in \mathcal{M}_z that intersect $[x_{k|k}]$ are marked, then every facet that shares an edge with a marked facet in \mathcal{M}_z is iteratively marked. The obtained connected components are the roads on which the vehicle may have traveled between t_k and t_{k+1} . The final selection is performed by eliminating any marked facets that do not intersect $[x_{k+1|k}]$.

This procedure replaces line 2 of Alg. 3. Figure 5 shows how topological facet selection can reduce the number of facet candidates, compared to the selection of every facet compatible with the prior box. It permits to reduce road selection ambiguities, especially in dense road networks.

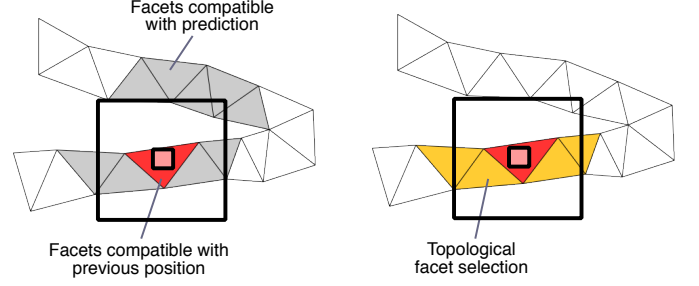


Figure 5: Simple facet selection vs topological facet selection. Simplified case with $[z] = [x_{k+1|k}]$ (large box). The small box is $[x_{k|k}]$.

IV. GPS PSEUDORANGE CONSTRAINTS

A. GPS observation model

GPS positioning with *pseudorange* measurements is a *Time of Arrival* method [16]. Pseudoranges are the distances to each visible satellite, offset by a unknown amount due to the offset between the receiver clock and the GPS time. GPS positioning using pseudoranges is thus a four-dimensional problem: along with the Cartesian coordinates (e, n, u) of the user in a local tangent frame to the Earth, the user's clock offset dt^u has to be estimated. Let $x = (e, n, u, d)$ with $d = c \cdot dt^u$.

Measured pseudoranges need to be corrected, to take into account relativity effects and the time taken by signals to cross the ionosphere and troposphere [17]. The propagation delay corrections applied to measured pseudoranges to get corrected pseudoranges ρ_i are imprecise because of model and parameter errors. Moreover, receiver measurement errors should also be taken into account. Corrected pseudorange measurements are thus represented as intervals $[\rho_i]$ whose bounds are determined given an integrity risk [18].

Satellite positions (e_i^s, n_i^s, u_i^s) are computed with broadcast ephemeris data. They are represented as boxes $[x^s] = ([e_i^s], [n_i^s], [u_i^s])$ to take ephemeris inaccuracy into account.

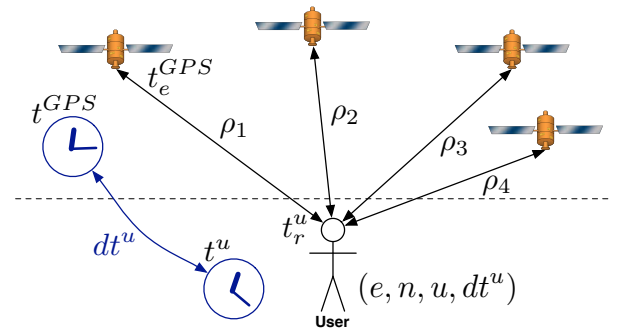


Figure 6: GPS positioning with pseudoranges

The location zone computation consists in characterizing the set X of all locations compatible with the measurements and the satellite position intervals. Each pseudorange introduces a constraint on the solution. The constraint induced by the i^{th} corrected pseudorange measurement is represented by the natural inclusion function of the GPS pseudorange observation

function:

$$[\rho_i] = \sqrt{([e] - [e_i^s])^2 + ([n] - [n_i^s])^2 + ([u] - [u_i^s])^2} + [d] \quad (4)$$

B. GPS contractor

A contractor for a single pseudorange constraint is given in Alg. 4. It is a forward-backward contractor [14], which performs constraint propagation in an optimal order, following the elementary constraints decomposition of the pseudorange measurement function (Fig. 7).

Algorithm 4

pseudorange_contract(in: $[x], [\rho], [x^s]$)

// Forward	// Backward
$[i9] := [u] - [u^s]$	$[d] := [d] \cap ([\rho] - [i1])$
$[i8] := \text{square}[i9]$	$[i1] := [i1] \cap ([\rho] - [d])$
$[i7] := [n] - [n^s]$	$[i2] := [i2] \cap \text{square}[i1]$
$[i6] := \text{square}[i7]$	$[i3] := [i3] \cap ([i2] - [i4])$
$[i5] := [e] - [e^s]$	$[i4] := [i4] \cap ([i2] - [i3])$
$[i4] := [i6] + [i8]$	$[i5] := [i5] \cap \text{square}^{-1}[i3]$
$[i3] := \text{square}[i5]$	$[i6] := [i6] \cap ([i4] - [i8])$
$[i2] := [i3] + [i4]$	$[i7] := [i7] \cap \text{square}^{-1}[i6]$
$[i1] := \sqrt{[i2]}$	$[i8] := [i8] \cap ([i4] - [i6])$
$[\rho] := [\rho] \cap ([i1] + [d])$	$[i9] := [i9] \cap \text{square}^{-1}[i8]$
	$[e] := [e] \cap ([i5] + [e^s])$
	$[n] := [n] \cap ([i7] + [n^s])$
	$[u] := [u] \cap ([i9] + [u^s])$

return $[x]$

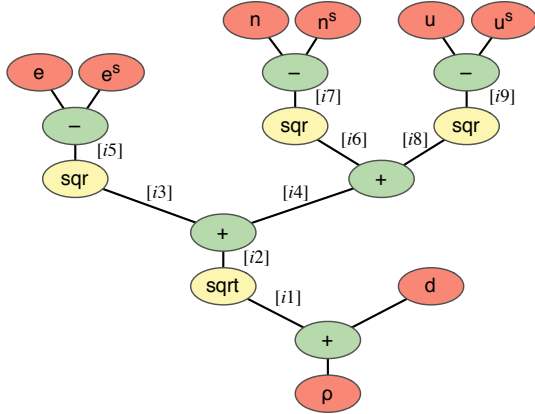


Figure 7: Elementary constraints decomposition of pseudorange measurement inclusion function (for the subtraction operators, the term on the right is the one which is subtracted)

The constraint for GPS positioning using pseudoranges consists in fulfilling the pseudorange constraints for all visible satellites. With m satellites in use, a GPS contractor is given in Alg. 5.

C. Robust GPS contractor

GPS measurements are prone to errors as shown in figure 8: although satellite failures are rare, measurement errors are

Algorithm 5

gps_contract(in: $[x], [\rho]_1, \dots, [\rho]_m, [x^s]_1, \dots, [x^s]_m$)

```

1: for  $i := 1$  to  $m$  do
2:    $[x] := [x] \cap \text{pseudorange\_contract}([x], [\rho]_i, [x^s]_i)$ 
3: end for
4: return  $[x]$ 

```

commonplace in urban environments, owing to multipath propagation, or non-line-of-sight measurements (when the direct signal is blocked and only reflected signals are measured). In these cases the error bounds set on pseudoranges are no longer consistent with the real pseudorange error.

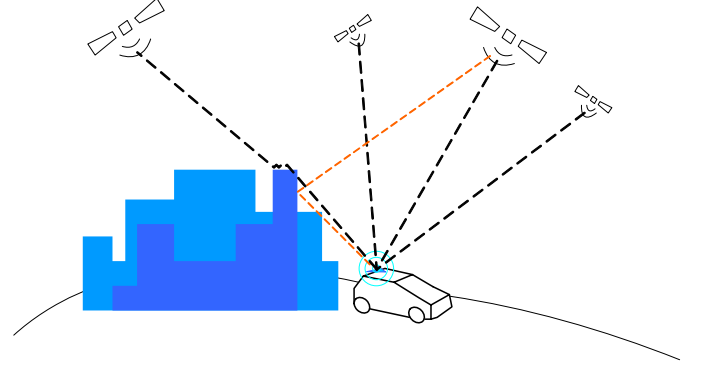


Figure 8: Illustration of satellite outage and multi-path in urban areas.

Inconsistent error bounds may lead either to an empty solution (in this case, the presence of an inconsistency is detected) which directly decreases the continuity of the positioning service, or to a misleading solution that affects integrity.

Instead of enforcing the constraints of all pseudoranges, the robust GPS constraint consists in relaxing a given number q of pseudorange constraints. This way, q erroneous measurements are tolerated, and the constraint corresponds to the set of solutions *at least compatible with $m - q$ measurements*. This is the so-called *q -relaxed intersection* [19] of the pseudorange constraints.

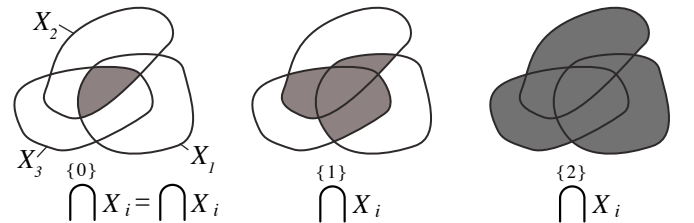


Figure 9: q -relaxed intersection of three sets for $q \in \{0, 1, 2\}$

Given m sets X_1, \dots, X_m of \mathbb{R}^n , the q -relaxed intersection $\bigcap_{\{q\}} X_i$ is the set of $x \in \mathbb{R}^n$ which belongs to at least $m - q$ of the X_i 's (Fig. 9).

The robust GPS contractor is given in Alg. 6. It should be noted that in contrast to the standard intersection of boxes, the q -relaxed intersection of several boxes is not generally a box.

As a consequence, the box-hull of the q -relaxed intersection is used at line 5 of Alg. 6.

Algorithm 6

```

gps_contract_qrel( $[x], [\rho]_1, \dots, [\rho]_m, [x]_1^s, \dots, [x]_m^s, q$ )
1: repeat
2:   for  $i := 1$  to  $m$  do
3:      $[xc](i) := \text{pseudorange\_contract}([x], [\rho]_i, [x]_i^s)$ 
4:   end for
5:    $[x] := \square \left( \bigcap_{i \in \{1, \dots, m\}} [xc](i) \right)$  hull of the  $q$ -relaxed  
intersection of  $m$   
boxes
6: until no more contraction can be done on  $[x]$ 
7: return  $[x]$ 

```

If a measurement is wrong and inconsistent with the other measurements, it is automatically excluded from the solution, and it can be identified as an outlier [18]. The fault detection and exclusion implemented in our system works by analyzing the consistency of individual measurements with the q -relaxed solution. The robust solution is equivalent to the union of every $m - q$ satellites combinations sub-solutions. When an erroneous measurement is present, the $m - q$ satellites sub-solutions tend to drift away, until they have no more common intersection. At this point, the presence of a fault can be detected. When all the sub-solutions involving a given measurement are empty, this measurement is identified as erroneous. Figure 10 gives an illustration with real data. Since the remaining sub-solutions do not include the faulty measurement, fault exclusion is automatic.

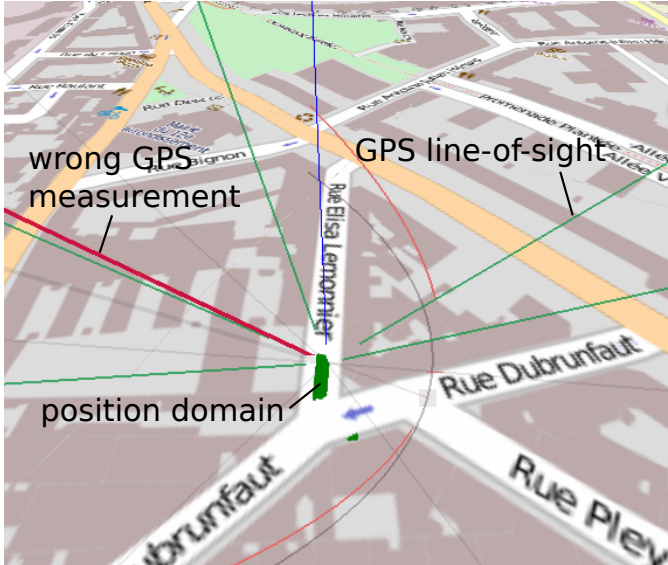


Figure 10: Detection of an erroneous GPS measurement with 6 satellites tracked by the receiver. The faulty pseudorange is shown in red (probably a non-line-of-sight signal)

When the robust GPS contractor is associated with the road contractor, the presence of a fault cannot be detected with less than 3 satellites in view, and at least 4 pseudoranges measurements are needed to identify the faulty measurement.

D. Receiver clock offset estimation/prediction

As previously stated, GPS positioning consists in finding not only the receiver's position, but also its clock offset dt^u . Over a short period of time, receiver time can be considered as a clock running at a constant speed. The receiver clock drift dt^u can thus be estimated to assist the prediction of dt^u . Once the clock offset predictor has been initialized, this enables positioning with a single satellite in view, as long as there is no ambiguity in the current road selection [20]. The clock offset prediction is thus a way to improve positioning availability in urban canyons.

The clock offset of many receivers is not continuous: it is often maintained within a specific range, which causes jumps in the dt^u value. In these cases, dt^u contraction fails because of inconsistency, and the receiver clock offset estimator is reset, which means in practice that the inconsistency is always detected, because typically a clock jump is about 1 ms and the confidence in dt^u is about 50 ns.

V. EXPERIMENTAL RESULTS

A. Experiment

To evaluate the iGPS positioning algorithm, data acquisition was performed on the *Stereopolis* vehicle from the French *Institut Geographique National* (IGN), for the *CityVIP* research program. It is a 1 km loop in the 12th arrondissement of Paris, a 5 minute drive around the local *mairie* (Fig. 11).



Figure 11: Trajectory around the 12th arrondissement *mairie* in Paris

The drivable space was provided by the IGN [6]. It was drawn up from precise aerial photographs (5 cm horizontal precision, 25 cm in altitude), and converted from its original Lambert93 projection to a local tangent frame. GPS positioning is also performed in the same local tangent frame. We use GPS pseudorange measurements acquired at 2 Hz with a high-end *Septentrio PolaRx2* receiver. Satellite positions and pseudorange corrections are computed with the open source *GPSTk* library [21]. Ground truth is provided by a post-processed *Applanix* inertial navigation system.

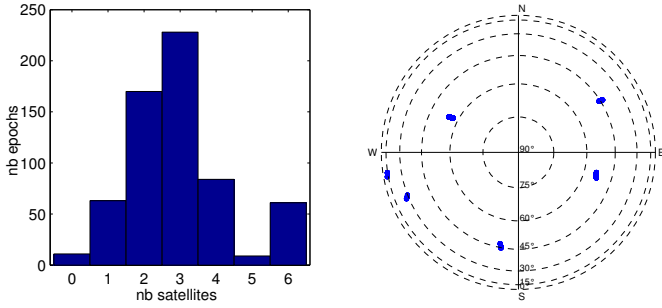


Figure 12: Histogram of the number of visible satellites and skyplot.

The acquired data are very challenging for GPS positioning, since there are fewer than 4 satellites in view during 77% of the trial, and fewer than 3 satellites in view in 40% of the measurement epochs (Fig. 12). In these conditions, the position computed by the *PolaRx2* receiver is unavailable for long periods of time in the narrow streets, and suffers from up to 10 m horizontal error and 35 m vertical error (Fig. 13).

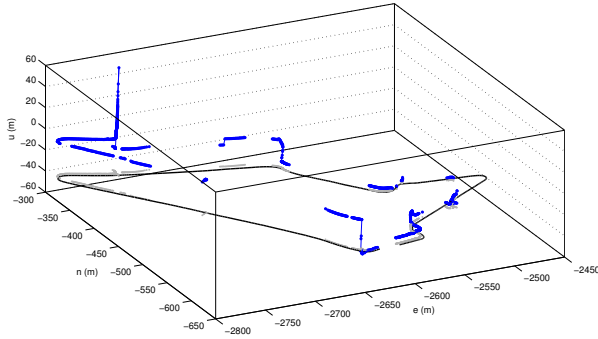


Figure 13: Position solution output by the *PolaRx2* receiver for the complete trial. Ground truth is in black.

B. Results of the iGPS positioning algorithm

Localization with iGPS is performed using the road contractor described in section III together with the robust GPS contractor described in section IV-C. We use the SIVIA algorithm (Alg. 1), with the contractor C_{road_gps} implemented in Alg. 7. This successively applies the road and GPS contractors until no more contraction occurs. The improved facet selection algorithm described in section III-D is employed. An odometer provides the traveled distance $s(t_k, t_{k+1})$ between two epochs. The traveled distance is also employed to compute a prior box $[x_{k+1}|_k]$ from the bounding box obtained in the previous step $[x_k|_k]$.

The SIVIA algorithm requires an initial box, in which the solution set will be searched. We use a large box that includes the whole test area. This is a great advantage of the interval based method over particle filtering, which requires an initial distribution of particles. For a large map, the required number of initial particles can be very large, which make initial localization time an memory consuming with particle

Algorithm 7 C_{road_gps} (in: $[x], \mathcal{M}, [\rho]_{1\dots m}, [x^s]_{1\dots m}, q$)

```

1: repeat
2:    $[x] := \text{road\_contract}([x], \mathcal{M})$ 
3:    $[x] := \text{gps\_contract\_qrel}([x], [\rho]_{1\dots m}, [x^s]_{1\dots m}, q)$ 
4: until no more contraction can be done on  $[x]$ 
5: return  $[x]$ 

```

filtering. On the contrary, initialization of the interval-based method only requires a single box that has to include possible locations. Thanks to the efficiency of CSP contraction, the initial box can be chosen arbitrarily large.

The number q of tolerated wrong measurements in the robust GPS contractor is dynamically set as a function of the number m of visible satellites. With fewer than 4 satellites in view, we use a non-robust contractor ($q_{m<4} = 0$). With at least 4 satellites in view, the robust contractor is employed: $q_{m=4} = 1$, $q_{m>4} = 2$. Pseudorange intervals are set to ensure a risk of less than 10^{-4} , assuming normally distributed pseudorange errors with $\sigma = 1$ m. The choice of error bounds on pseudoranges is done as follows [18].

Let r be the risk associated with each pseudorange measurement:

$$r = \Pr(\rho_i \notin [\rho]_i), i \in \{1 \dots m\} \quad (5)$$

Let N_{ok} be the number of pseudorange intervals that are consistent with the truth. The probability of having exactly i good pseudoranges out of m is given by the binomial distribution. Thus, by summing over successive values of i , the probability of having at least $m - q$ good measurements is:

$$\Pr(N_{ok} \geq m - q) = \sum_{i=m-q}^m \binom{m}{i} (1-r)^i r^{m-i} \quad (6)$$

where $\binom{m}{i} = \frac{m!}{i!(m-i)!}$ is the binomial coefficient.

The robust q -relaxed GPS contractor enables to compute an outer approximation $\bar{X}(t)$ of the position confidence domain as a subpaving, which is guaranteed to be consistent with the true position $x(t)$ if the number of spurious pseudoranges is less than or equal to the number q of relaxed measurements. This way,

$$N_{ok} \geq m - q \Rightarrow x(t) \in \bar{X}(t) \quad (7)$$

which leads to

$$\Pr(x(t) \in \bar{X}(t)) \geq \Pr(N_{ok} \geq m - q) \quad (8)$$

$$\Pr(x(t) \notin \bar{X}(t)) \leq 1 - \Pr(N_{ok} \geq m - q) \quad (9)$$

$$R \leq 1 - \sum_{i=m-q}^m \binom{m}{i} (1-r)^i r^{m-i} \quad (10)$$

With the chosen strategy to set q as function of m , and with a risk $R = 10^{-4}$, (10) can be inverted to find the individual maximum risk r that can be assumed on each pseudorange interval. Assuming normally distributed pseudorange errors of variance σ^2 , a way to set the error bounds on each pseudorange measurement is then given by

$$[\rho] = [\rho - \alpha\sigma, \rho + \alpha\sigma] \text{ with } \alpha = -\Phi^{-1}\left(\frac{r}{2}\right)$$

where Φ represents the cumulative distribution function of the standard normal distribution. The values of r and α as a function of the number of pseudoranges m are given in Table I.

Table I: Pseudorange intervals risk r and error bounds for $R = 10^{-4}$ as a function of the number m of measurements. $[\rho] = [\rho - \alpha\sigma, \rho + \alpha\sigma]$

m	1	2	3	4	5	6
q	0	0	0	1	2	2
r	10^{-4}	$5 \cdot 10^{-5}$	$3.3 \cdot 10^{-5}$	$4.1 \cdot 10^{-3}$	0.022	0.017
α	3.89	4.06	4.15	2.87	2.29	2.38

The positioning algorithm is implemented in C++, and computes position confidence domains in real-time. It uses a multithreaded version of SIVIA that enables to process several boxes in parallel, thus taking advantage of multicore processors. In the presented results, a processing time of 250 ms was allotted for each sample.

Fig. 14 shows the bounds of the confidence domains with respect to ground truth. Confidence domains are consistent with ground truth throughout the trial. The center of gravity of the solution subpaving is used as a point position estimate for error computations.

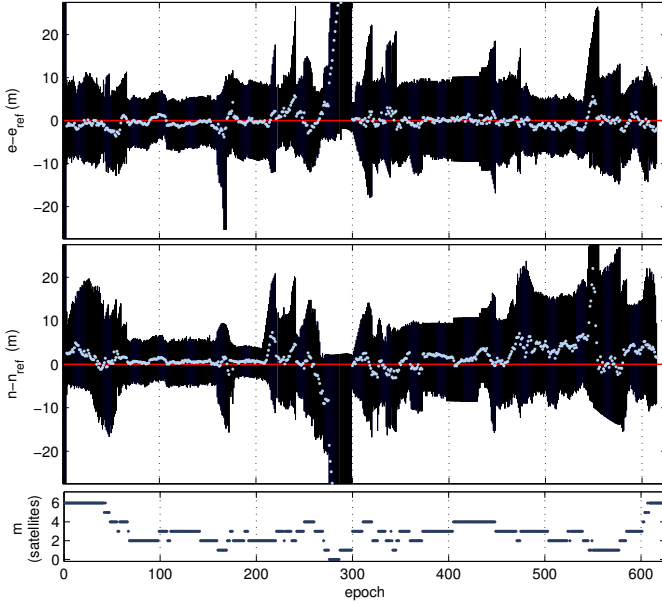


Figure 14: Bounds of the computed position subpaving. Zero ordinate is ground truth. Points are the center of gravity of the solution subpaving.

The test starts with good satellite visibility, before entering urban canyons with 2 or 3 satellites in view. At epoch #69 (Fig. 15a), accurate positioning is achieved with only two satellites in view, thanks to the road constraint. At epoch #220 (Fig. 15b), road selection is ambiguous at a crossroads. Ambiguous positioning occurs at epoch #553 (Fig. 15c) after a crossroads, since two satellites are not enough to select only the good road segment.

Around epoch #280, the system is unable to efficiently contract position with zero then only one satellite in view. Nevertheless, between epochs #560 and #580, the system is

able to contract position with only one satellite in view, thanks to the receiver clock offset prediction.

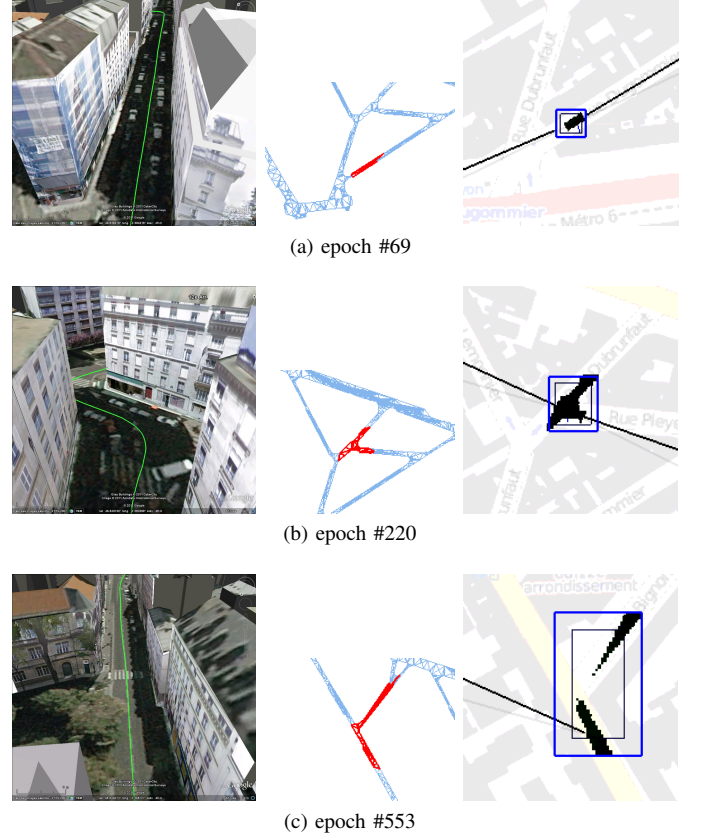


Figure 15: Left: view of surrounding buildings. Center: drivable space. Candidate facets are in red. Right: position solution subpaving (in black). The blue box is the prior predicted position, and the lines are GPS satellites lines of sight.

Although the computed confidence domains are easy to interpret visually when plotted on a map, a point position estimate is often needed to communicate with other systems. The center of gravity of the subpaving is a good position estimate when the solution set is connected [22]. With at least 2 satellites in view, positioning error is less than 6.5 m 95% of the time (Fig. 16). In the same conditions, the radius of the computed confidence domain does not exceed 16 m 95% of the time.

Along with the confidence domain radius, it is also interesting to see how the true position is distributed inside the computed domains. Figure 17 shows the distribution of the relative location of the ground truth inside the confidence domain bounds. Since the domain's width changes at each sample, we have normalized the domain between 0 (lower bound) and 1 (upper bound). It shows that the true position is not always at the center of the domain, but distributed over it. The confidence domains are not excessively pessimistic since the true position is quite well distributed over the full width of the interval. With the chosen risk of 10^{-4} , one could expect that the extreme values (located at the boundaries of the domain) would be rare. However, there is a significant number

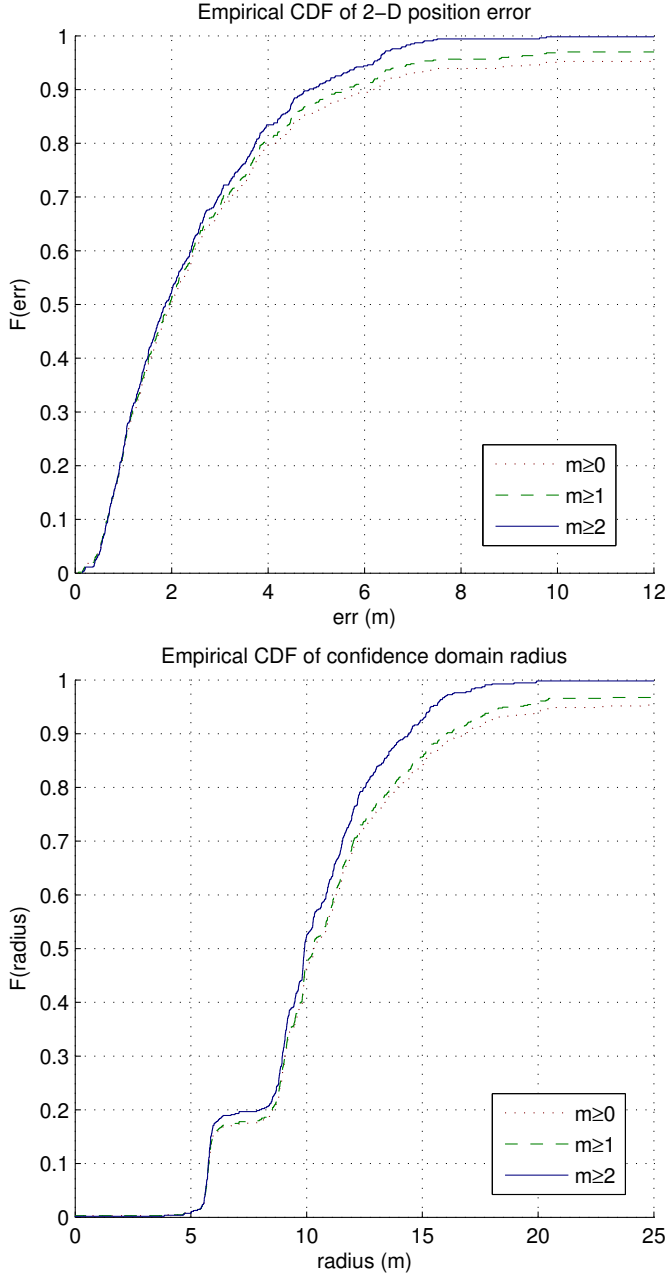


Figure 16: Empirical cumulative distribution functions of the 2-D position error of the center of gravity of the solution, and of the confidence domain radius. CDFs are plotted for all epochs ($m \geq 0$), and for epochs with at least one ($m \geq 1$) or two ($m \geq 2$) satellites in view.

of true positions located near the confidence domain boundaries. They are mainly due to under-constrained cases, where the non-isotropy of the map constraint distorts the confidence domain in a direction or causes multiple hypotheses.

VI. CONCLUSIONS

The iGPS global localization method has been presented in this paper. It uses bounded-error GPS measurements and a precise 3-D map of the drivable space as geometric constraints on position. These data are merged in a branch and prune

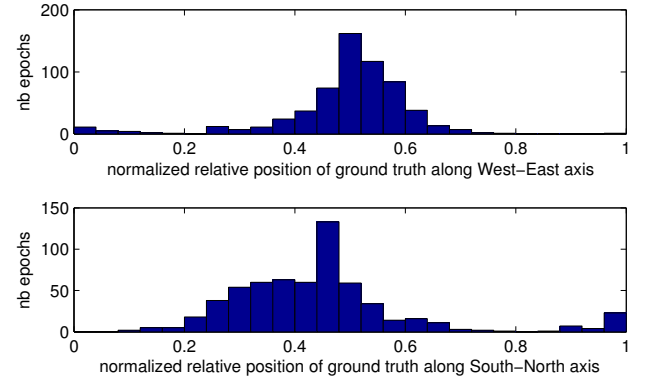


Figure 17: Distribution of the relative location of the true position inside the confidence domain bounds. 0 corresponds to the lower bound, 1 corresponds to the upper bound.

algorithm, which combines a robust GPS pseudorange contractor and a road contractor that takes topology into account, to characterize the user location zone.

Experimental validation was performed with real GPS data in Paris. With at least 2 satellites in view, the method provides consistent error bounds with 16-meter average radius location zones, and a 6.5 m horizontal accuracy (95%).

Since the method neither estimates nor predict the full vehicle pose, confidence domains computed with fewer than 2 satellites in view are very large. Integrating a vehicle evolution model will provide a prediction of the vehicle pose, resulting in tighter confidence domains when GPS visibility is reduced. It should also allow the use of a robust GPS contractor with fewer than four satellites in view.

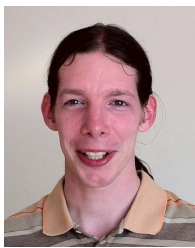
ACKNOWLEDGMENTS

This work has been supported by ANR (French National Agency) CityVIP project under grant ANR-07_TSFA-013-01. The authors would particularly like to thank the IGN staff.

REFERENCES

- [1] B. Thuilot, R. Lenain, P. Martinet, and C. Cariou, "Accurate GPS-based guidance of agricultural vehicles operating on slippery grounds," in *Focus on Robotics Research*, J. Liu, Ed. Nova Science Publishers, 2006, pp. 185–239, Chapter 6 ISBN: 1-59454-594-4.
- [2] H. Zhao, J. Cui, H. Zha, K. Katabira, X. Shao, and R. Shibasaki, "Sensing an intersection using a network of laser scanners and video cameras," *Intelligent Transportation Systems Magazine, IEEE*, vol. 1, no. 2, pp. 31–37, summer 2009.
- [3] I. Miller, S. Lupashin, N. Zych, P. Moran, B. Schimpf, A. Nathan, and E. Garcia, "Cornell university's 2005 DARPA grand challenge entry," *Journal of Field Robotics*, vol. 23, no. 8, pp. 625–652, 2006.
- [4] A. Lahrech, C. Boucher, and J.-C. Noyer, "Accurate vehicle positioning in urban areas," *IECON Proceedings (Industrial Electronics Conference)*, vol. 5, pp. 486–490, 2005.
- [5] S. Syed and M. E. Cannon, "Map-aided GPS navigation," *GPS World*, vol. 16, no. 11, pp. 39–44, 2005.
- [6] N. Paparoditis, C. Thom, and H. Jibrini, "Surface reconstruction in urban areas from multiple views of aerial digital frame cameras," *International Archives of Photogrammetry and Remote Sensing*, vol. 33, no. Suppl. B3, pp. 43–50, 2000.
- [7] R. Toledo-Moreo, D. Betaille, and F. Peyret, "Lane-level integrity provision for navigation and map matching with gnss, dead reckoning, and enhanced maps," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 11, no. 1, pp. 100–112, march 2010.

- [8] G. Taylor, G. Blewitt, and J. Wiley, *Intelligent positioning: GIS-GPS unification*. John Wiley, 2006.
- [9] S. J. Julier and J. J. LaViola, "On kalman filtering with nonlinear equality constraints," *Signal Processing, IEEE Transactions on*, vol. 55, no. 6, pp. 2774–2784, june 2007.
- [10] F. Chausse, J. Laneurit, and R. Chapuis, "Vehicle localization on a digital map using particles filtering," in *IEEE Intelligent Vehicles Symposium, 2005*. Las Vegas, Nevada, U.S.A.: IEEE, 2005, pp. 243–248.
- [11] A. Selloum, D. Bétaille, E. Le Carpentier, and F. Peyret, "Robustification of a map aided location process using road direction," in *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on*. IEEE, 2010, pp. 1504–1510.
- [12] A. Peker, O. Tosun, and T. Acarman, "Particle filter vehicle localization and map-matching using map topology," in *Intelligent Vehicles Symposium (IV), 2011 IEEE*. IEEE, 2011, pp. 248–253.
- [13] L. Jaulin, M. Kieffer, O. Didrit, and É. Walter, *Applied Interval Analysis*. Springer-Verlag, 2001.
- [14] F. Benhamou, F. Goualard, L. Granvilliers, and J.-F. Puget, "Revising hull and box consistency," in *Int. Conf. on Logic Programming*. MIT press, 1999, pp. 230–244.
- [15] I. E. Sutherland and G. W. Hodgman, "Reentrant polygon clipping," *Communications of the ACM*, vol. 17, no. 1, pp. 32–42, 1974.
- [16] A. Leick, *GPS satellite surveying, Third edition*. Wiley, 2004.
- [17] E. Kaplan and C. Hegarty, *Understanding GPS: Principles and Applications Second Edition*. Artech House Publishers, 2006.
- [18] V. Drevelle and P. Bonnifait, "A set-membership approach for high integrity height-aided satellite positioning," *GPS Solutions*, vol. 15, no. 4, pp. 357–368, 2011.
- [19] L. Jaulin, "Robust set-membership state estimation; application to underwater robotics," *Automatica*, vol. 45, no. 1, pp. 202–206, 2009.
- [20] C. Fouque and P. Bonnifait, "Multi-hypothesis map-matching on 3d navigable maps using raw gps measurements," in *13th IEEE International Conference on Intelligent Transportation Systems*, Madeira Island Portugal, 09 2010, pp. 1498–1503.
- [21] B. Tolman, R. B. Harris, T. Gaussiran, D. Munton, J. Little, R. Mach, S. Nelsen, and B. Renfro, "The GPS Toolkit: Open Source GPS Software," in *Proceedings of the 16th International Technical Meeting of the Satellite Division of the Institute of Navigation*, Long Beach, California, September 2004.
- [22] M. Kieffer, L. Jaulin, É. Walter, and D. Meizel, "Robust autonomous robot localization using interval analysis," *Reliable Computing*, vol. 6, no. 3, pp. 337–362, 08 2000.



sensor localization of vehicles.

Vincent Drevelle is a teaching assistant in the Computer Science and Engineering department of the the Université de Technologie de Compiègne (UTC), France. He obtained his Ph.D. degree in systems and information technologies from the UTC in 2011, after he graduated as a computer science engineer and he obtained a master degree in from the UTC. Since 2008, he has been with Heudiasyc UMR 7253, a joint research unit backed by CNRS and UTC. His current research interests are in set-theoretic methods and interval analysis, for multi-



database for Advanced Driving Assistance Systems.

Philippe Bonnifait is a professor in the Computer Science and Engineering department of the Université de Technologie de Compiègne (UTC), France. He obtained his Ph.D. degree in automatic control and computer science from the École Centrale de Nantes in 1997. Since 1998, he has been with Heudiasyc UMR 7253, a common research laboratory between UTC and CNRS. His research interests are in Intelligent Vehicles. He is currently working on high integrity positioning and map-matching to retrieve contextual information stored in a navigable