

Article

Outlier Detection in GNSS Pseudo-Range/Doppler Measurements for Robust Localization

Salim Zair *, Sylvie Le Hégarat-Masclé and Emmanuel Seignez

SATIE (Systems Applications of Information Energy Technologies) laboratory, University of Paris-Sud, 91405 Orsay, France; sylvie.le-hegarat@u-psud.fr (S.L.H.-M.); emmanuel.seignez@u-psud.fr (E.S.)

* Correspondence: salim.zair@u-psud.fr; Tel.: +33-1-69-15-78-66

Academic Editor: Davide Brunelli

Received: 24 January 2016; Accepted: 19 April 2016; Published: 22 April 2016

Abstract: In urban areas or space-constrained environments with obstacles, vehicle localization using Global Navigation Satellite System (GNSS) data is hindered by Non-Line Of Sight (NLOS) and multipath receptions. These phenomena induce faulty data that disrupt the precise localization of the GNSS receiver. In this study, we detect the outliers among the observations, Pseudo-Range (PR) and/or Doppler measurements, and we evaluate how discarding them improves the localization. We specify a *contrario* modeling for GNSS raw data to derive an algorithm that partitions the dataset between inliers and outliers. Then, only the inlier data are considered in the localization process performed either through a classical Particle Filter (PF) or a Rao-Blackwellization (RB) approach. Both localization algorithms exclusively use GNSS data, but they differ by the way Doppler measurements are processed. An experiment has been performed with a GPS receiver aboard a vehicle. Results show that the proposed algorithms are able to detect the ‘outliers’ in the raw data while being robust to non-Gaussian noise and to intermittent satellite blockage. We compare the performance results achieved either estimating only PR outliers or estimating both PR and Doppler outliers. The best localization is achieved using the RB approach coupled with PR-Doppler outlier estimation.

Keywords: Global Navigation Satellite Systems (GNSS); robust localization; a *contrario* decision; particle filter; Rao-Blackwellization

1. Introduction

The Global Navigation Satellite Systems (GNSS), such as the Global Positioning Systems (GPS), have been developed to provide an absolute location on an Earth-Centered Earth-Fixed (ECEF) [1]. These sensors became very popular for autonomous navigation [2] and applications of Intelligent Transportation Systems (ITS) thanks to the worldwide coverage of these constellations and the rather low cost of the receivers. Even if several works have proposed to combine GNSS data with other information sources, either sensors (e.g., Inertial Measurement Unit (IMU) [3]) or prior information (e.g., maps [4,5]), there is still a need to improve the performance of GNSS-only localization. Indeed, even in the perspective of fusion with other data, the accuracy of the GNSS estimation will impact the location result. Then, this study focuses on GNSS-only localization.

Early works estimated the receiver location based on GNSS Pseudo-Range (PR) data. Recently, the estimation of the instantaneous velocity that may be derived from Doppler measurements has been proposed. For instance, [6] introduces both the PR and Doppler measurements in the Extended Kalman Filter (EKF). These Doppler measurements may be particularly helpful in constrained environments where the number of usable observations may drop. Indeed, in space-constrained areas, the obstacles (buildings, trees) reflect the signals sent by the satellites, inducing Non-Line Of Sight (NLOS) and multipath receptions. The corrupted measurements are characterized by a positive bias that increases the estimated satellite-receiver distance in a faulty way, which is difficult to model and to correct. The

works in [7,8] have experimentally shown that Doppler measurements are affected, as well, although to a lesser extent than PR measurements, by multipaths and NLOS.

To detect the faulty data, a first approach is to analyze the signal measurements. For instance, [9] exploits the carrier to noise density measurement (C/N_0) in order to partition the observation set between LOS signals ($C/N_0 \simeq 45$ dB) and NLOS signals ($C/N_0 \leq 40$ dB). However, in urban canyons, the NLOS signals may be stronger than the LOS one [8]. A second approach is to look toward robust estimation, *i.e.*, methods that are able to cope with some faulty data. In GPS-based localization, we can cite the Receiver Autonomous Integrity Monitoring (RAIM) [10] or q -relaxation technique used in interval analysis [11]. Both approaches assume a bounded number of outliers. Assuming Gaussian noise, [12] proposed an EKF with outlier detection. The Particle Filter (PF), which has been proposed to resolve non-linear/non-Gaussian problems [13], was applied in [14] having discarded the outliers from the set of observations, whereas in [15], it was used to estimate both the corrupted bias on PR observations and the localization parameters. However, neither [15] nor [14] investigated the presence of outliers in Doppler measurements. Besides, [15] only considers simulated data.

The main contribution of this work is to propose a robust localization process that uses both PR and Doppler measurements. It is based on the adaptation of signal processing methods previously applied to other problems or data. It involves two parts: (i) the inlier/outlier partitioning characterized by the absence of a threshold; (ii) the filtering for a GPS-based localization characterized by its robustness to noise and to intermittent satellite blockage. The first point is achieved by formulating the problem in terms of minimization of a criterion, namely the Number of False Alarms (NFA). This criterion was introduced by [16,17] to measure the degree of surprise or contradiction of a structured observation relative to a noise (unstructured data) model, and it has been successfully applied to various problems in image processing [18–22]. In a previous work [23], we have defined and compared two NFA criteria, and we have shown that they are more efficient than classic statistical tests to partition the PR measurements between a consistent dataset (the inliers) against an inconsistent dataset (the outliers). However, in this first work, only PR measurements were considered. Then, this study develops the ideas and first results presented in conference paper [24], where a rather simple implementation of PF was considered. Based on temporal redundancy, the PF allows us to filter the noise present in the inlier data. However, for practical reasons, it cannot handle the state vector of large dimensionality. In this work, we propose to use a more sophisticated filter, namely the Rao–Blackwell Particle Filter (RBPF) [25]. Its principle is to split the state system into two subsystems, a linear part and a non-linear one, so that the linear part may be analytically solved, whereas the non-linear part is approximated using the importance sampling technique (like in PF). RBPF has already been applied successfully for navigation [26], tracking [27] and GPS multipath estimation [15]. In [26], RBPF was applied for GPS-based localization in urban canyons. However, the authors only consider PR measurements, whereas in this study, we propose to extend their filter to both PR and Doppler measurements and to couple it with the outlier detection using NFA criterion based on a *contrario* modeling. Concerning the application, we focus on land vehicle navigation in constrained environments. Then, to achieve similar localization performance in such environments as in open areas, our algorithm should be robust until about 40% of outliers. The “raw” data we consider (and among which outliers will be searched) are the PR and the Doppler measurements provided by the GNSS receiver (L_1 carrier). Even if they are already estimations from the pseudo-random codes, we call them “raw” by contrast to positioning values also provided by the GNSS receiver.

Section 2 introduces the notations and basic equations inherent to the localization problem using GNSS data. Section 3 describes the proposed method that involves a detection of outlier measurements in the dataset using the NFA criterion followed by the localization process using GPS raw data, either based on PF or on the RBPF algorithm. The experiment and related results are discussed in Section 4. Section 5 reports our conclusions and perspectives.

2. Problem Formulation

2.1. Observation Model

In this study, we consider two pieces of information provided by GNSS satellite S_j . The first one is the pseudo-range ρ_j that is related to the distance between the receiver and S_j . Denoting by upper-script \top the transpose operator, the receiver position is denoted $\mathbf{x}_r = (e_r, n_r, u_r)^\top$ in the ENU (East, North, Up) coordinate local system, and the S_j position is denoted $\mathbf{x}_{S_j} = (e_{S_j}, n_{S_j}, u_{S_j})^\top$. We choose the ENU frame for its wide use in land navigation (since it allows us to process the ‘up’ coordinates separately). Then, the pseudo-range depends on \mathbf{x}_r , \mathbf{x}_{S_j} , δ_t the time bias (difference) between the two unsynchronized clocks of the satellite and receiver, respectively, c the light velocity and random noise ϵ_j :

$$\begin{aligned}\rho_j &= \|\mathbf{x}_r - \mathbf{x}_{S_j}\| + c\delta_t + \epsilon_j, \\ &= \sqrt{(e_r - e_{S_j})^2 + (n_r - n_{S_j})^2 + (u_r - u_{S_j})^2} + c\delta_t + \epsilon_j\end{aligned}\quad (1)$$

Equation (1) is the simplest version of the PR observation equation. It does not represent multipath or NLOS receptions, so that they can be detected as deviations relatively to this model.

The second information piece is the Doppler measurement that is related to the receiver velocity $\dot{\mathbf{x}}_r = (\dot{e}_r, \dot{n}_r, \dot{u}_r)^\top$. Denoting $\dot{\mathbf{x}}_{S_j} = (\dot{e}_{S_j}, \dot{n}_{S_j}, \dot{u}_{S_j})^\top$, the S_i satellite velocity that is determined using broadcast ephemeris [28],

$$\dot{\rho}_j = (\dot{\mathbf{x}}_r - \dot{\mathbf{x}}_{S_j}) \cdot \mathbf{a}_j - c\dot{\delta}_t + \epsilon'_j \quad (2)$$

where $\dot{\rho}_j$, called the “PR rate”, is equal to $c \frac{D_j}{f_1}$, with $f_1 = 1.575$ GHz and D_j the Doppler observation (in Hz) provided by S_j , \mathbf{a}_j is to the unit vector collinear to the straight line through the receiver and satellite S_j ($\mathbf{a}_j = \frac{\mathbf{x}_{S_j} - \mathbf{x}_r}{\|\mathbf{x}_{S_j} - \mathbf{x}_r\|}$), “ \cdot ” denotes the dot product, $\dot{\delta}_t$ the clock drift and ϵ'_j random noise.

2.2. Localization Problem

For location estimation, different systems of equations may be considered depending on the used data: PR, Doppler measurements or both data.

Firstly, only using PR, at least four observations are required to estimate vector \mathbf{x}_r and time bias δ_t by solving the system of Equation (1).

Secondly, only using Doppler measurements, *theoretically* vector $\dot{\mathbf{x}}_r$, time drift $\dot{\delta}_t$ and vector \mathbf{x}_r could be estimated, since they all appear in Equation (2). However, two hindrances to this approach are: (i) per epoch, at least seven observations from different satellites would be required, which is incompatible with robustness to satellite blockage in constrained environments; (ii) Equation (2)’s sensitivity to \mathbf{x}_r is rather poor, since \mathbf{x}_r is involved only through \mathbf{a}_j . Thus, practically, PR measurements are also used to derive an estimation of \mathbf{x}_r and then an estimation of \mathbf{a}_j , denoted $\tilde{\mathbf{a}}_j$, which is introduced in Equation (2):

$$\dot{\rho}_j + \dot{\mathbf{x}}_{S_j} \cdot \tilde{\mathbf{a}}_j = \dot{\mathbf{x}}_r \cdot \tilde{\mathbf{a}}_j - c\dot{\delta}_t + \epsilon'_j \quad (3)$$

The third and last approach consist of considering simultaneously PR and Doppler measurements. The vector $\boldsymbol{\xi}_1 = (e_r, \dot{e}_r, n_r, \dot{n}_r, u_r, \dot{u}_r, \delta_t, \dot{\delta}_t)^\top$ gathers the parameters involved in Equations (1) and (2). Having linearized Equations (1) and (2), the resolution of the derived system (having at least eight equations) can be achieved using the Gauss–Newton iterative algorithm. Specifically, if \mathbf{X}_S and $\dot{\mathbf{X}}_S$ denote the matrices gathering the vectors \mathbf{x}_{S_j} and $\dot{\mathbf{x}}_{S_j}$, respectively, and the increment $\delta\hat{\boldsymbol{\xi}}_1^{(k)}$ to sum to previous estimate $\hat{\boldsymbol{\xi}}_1^{(k-1)}$ (k being the iteration number) is:

$$\delta\hat{\boldsymbol{\xi}}_1^{(k)} = \arg \min_{\delta\hat{\boldsymbol{\xi}}_1} \|(z - z^{(k-1)}) - H(\mathbf{X}_S, \dot{\mathbf{X}}_S, \hat{\boldsymbol{\xi}}_1^{(k-1)})\delta\hat{\boldsymbol{\xi}}_1\| \quad (4)$$

where $\mathbf{z} = (\rho_1 \cdots \rho_n, \dot{\rho}_1 \cdots \dot{\rho}_n)^\top$ is the vector of observations, $\mathbf{z}^{(k-1)}$ is the estimation of \mathbf{z} computed from previous (iteration $k - 1$) state vector $\xi_1^{(k-1)}$ and \mathbf{H} is the Jacobian matrix of the Equations (1) and (2) system (see [29] for more details).

2.3. Dynamic Model

In order to increase the robustness of the estimation, this latter may be done considering not only one epoch, but several epochs. Then, the data acquired at the different epochs should be linked through a model. In [30], the authors propose a polynomial dynamic model fitted on a time interval, including multiple epochs. Limiting ourselves to the first order, PR measurements are related using a dynamic model involving GNSS, the receiver location and speed, so that the ξ_1 vector already defined is suitable. However, considering also Doppler measurements, the acceleration should be introduced in the dynamic model, and the considered state vector becomes $\xi_2 = (e_r, \dot{e}_r, \ddot{e}_r, n_r, \dot{n}_r, \ddot{n}_r, u_r, \dot{u}_r, \delta_t, \dot{\delta}_t)^\top$.

State vectors, either ξ_1 or ξ_2 , at different instants are linearly linked through transition matrices $\mathbf{M}_{i,dt}$ of the considered dynamic models, defined as follows:

$$\begin{aligned} \mathbf{C}_{dt} &= \begin{pmatrix} 1 & dt \\ 0 & 1 \end{pmatrix}, \mathbf{0}_{2 \times 2} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \\ \mathbf{D}_{dt} &= \begin{pmatrix} \frac{dt^2}{2} & 0 \\ dt & 0 \end{pmatrix} \mathbf{I}_{2 \times 2} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \\ \mathbf{M}_{1,dt} &= \begin{pmatrix} \mathbf{C}_{dt} & \mathbf{0}_{2 \times 2} & \mathbf{0}_{2 \times 2} & \mathbf{0}_{2 \times 2} \\ \mathbf{0}_{2 \times 2} & \mathbf{C}_{dt} & \mathbf{0}_{2 \times 2} & \mathbf{0}_{2 \times 2} \\ \mathbf{0}_{2 \times 2} & \mathbf{0}_{2 \times 2} & \mathbf{C}_{dt} & \mathbf{0}_{2 \times 2} \\ \mathbf{0}_{2 \times 2} & \mathbf{0}_{2 \times 2} & \mathbf{0}_{2 \times 2} & \mathbf{C}_{dt} \end{pmatrix} \end{aligned} \quad (5)$$

$$\mathbf{M}_{2,dt} = \begin{pmatrix} \mathbf{C}_{dt} & \mathbf{D}_{dt} & \mathbf{0}_{2 \times 2} & \mathbf{0}_{2 \times 2} & \mathbf{0}_{2 \times 2} \\ \mathbf{0}_{2 \times 2} & \mathbf{I}_{2 \times 2} & \mathbf{D}_{dt}^\tau & \mathbf{0}_{2 \times 2} & \mathbf{0}_{2 \times 2} \\ \mathbf{0}_{2 \times 2} & \mathbf{0}_{2 \times 2} & \mathbf{C}_{dt} & \mathbf{0}_{2 \times 2} & \mathbf{0}_{2 \times 2} \\ \mathbf{0}_{2 \times 2} & \mathbf{0}_{2 \times 2} & \mathbf{0}_{2 \times 2} & \mathbf{C}_{dt} & \mathbf{0}_{2 \times 2} \\ \mathbf{0}_{2 \times 2} & \mathbf{0}_{2 \times 2} & \mathbf{0}_{2 \times 2} & \mathbf{0}_{2 \times 2} & \mathbf{C}_{dt} \end{pmatrix} \quad (6)$$

where the superscript τ denotes the anti-diagonal transpose operator (the transpose of the matrix with respect to the anti-diagonal). Denoting $\xi_{i,t}$, $i \in \{1, 2\}$, the state vector at t ,

$$\xi_{i,t+dt} = \mathbf{M}_{i,dt} \xi_{i,t} + o(dt^{i+1}) \quad (7)$$

where $o(dt^{i+1})$ is the error (approximation) of the considered dynamic model.

Using the dynamic Model Equation (7), we are now able to compute the expected measurements (PR or Doppler) at different instants. Specifically, denoting $\mathcal{T} = \{t_k = t + kdt, k \in \{0, \dots, n_{ep} - 1\}\}$ the set of epochs considered for the estimation of the solution, $\mathbf{X}_{S_i, t_k} = (e_{S_i, t_k}, n_{S_i, t_k}, u_{S_i, t_k})^\top$ the satellite S_i location at instant t_k and the expected pseudo-range $\tilde{\rho}_i(t_k)$ from S_i at t_k may be derived from ξ_1 :

$$\tilde{\rho}_i(t_k | \xi_{1,t}) = \|\mathbf{M}_{1,kdt} \xi_{1,t}\|_{1,3,5} - \mathbf{X}_{S_i, t_k} + c(\delta_t - \dot{\delta}_t kdt) \quad (8)$$

where the subscript in matrix notation $[\]_{l_1, l_2, l_3}$ denotes the restriction of the matrix or vector to rows l_1 , l_2 and l_3 and $\|v\|$ is the norm of vector v .

In a similar way, the Doppler measurement expected at t_k from S_i may be derived from ξ_2 :

$$\begin{aligned}
 a_{i,t_k} &= -\frac{[M_{2,kdt} \cdot \xi_{2,t}]_{1,4,7} - X_{S_i,t_k}}{\| [M_{2,kdt} \cdot \xi_{2,t}]_{1,4,7} - X_{S_i,t_k} \|} \\
 \tilde{\rho}_i(t_k | \xi_{2,t}) &= \left([M_{2,kdt} \cdot \xi_{2,t}]_{2,5,8} - \dot{X}_{S_i,t_k} \right) \cdot a_{i,t_k} - c\dot{\delta}_t
 \end{aligned} \quad (9)$$

Then, using classical regression, the state vector optimal values $\hat{\xi}_{1,t}$ and $\hat{\xi}_{2,t}$ are those minimizing the quadratic errors:

$$\hat{\xi}_{1,t} = \arg \min_{\xi_{1,t}} \sum_{t_k \in \mathcal{T}} \sum_{i \in \mathcal{I}(t_k)} [\tilde{\rho}_i(t_k | \xi_{1,t}) - \rho_i(t_k)]^2 \quad (10)$$

$$\hat{\xi}_{2,t} = \arg \min_{\xi_{2,t}} \sum_{t_k \in \mathcal{T}} \sum_{i \in \mathcal{I}(t_k)} \left[[\tilde{\rho}_i(t_k | \xi_{2,t}) - \rho_i(t_k)]^2 + \beta [\tilde{\rho}_i(t_k | \xi_{2,t}) - \dot{\rho}_i(t_k)]^2 \right] \quad (11)$$

where $\mathcal{I}(t_k)$ is the set of the indices of the satellites providing measurements at t_k and β is a weighting factor between the residues associated with PR and Doppler data, respectively.

In previous equations, the minimization is performed considering all of the measurements (PR and/or Doppler ones) available for the considered set of epochs \mathcal{T} . However, some of these measurements may correspond to outliers, and might then bias the estimation. In the following part of the paper, in addition to the acronym PR, we use the abbreviations “Dp” for “Doppler measurement” and “(PR,Dp)” for “both PR and Doppler measurements”.

3. Proposed Approach

In the presence of outliers, several strategies have been proposed. Robust methods aim at automatically mitigating the weight of these outliers in the estimation. For instance, PF or its variants belonging to the class of robust estimators can theoretically cope with outliers simply by giving a very small weight to the generated particles. However, if this filter has proven its efficiency against noise, we will see that too many outliers jeopardize the filter stability. Then, in the case of GPS data processing, some statistical tests have been proposed to detect the outliers, e.g., [31]. The most simple to cope with these outliers is simply to discard them from the data measurements (just as if the corresponding satellites were blocked). This is the strategy of the standard Fault Detection and Exclusion (FDE) technique implemented in the GPS receivers (even if they can only cope with at most one erroneous measurement [32]). More sophisticated strategies have also been proposed, e.g., [15,33], that aim at correcting the outliers. However, in this study, we do not consider such strategies, because we focus on the following basic main questions:

- For the localization problem, are Doppler measurements less subject to outliers than PR measurements?
- Does the presence of outliers also impact robust localization algorithms, such as PF or the Rao–Blackwell Particle Filter?
- In the affirmative case, is it worth detecting and discarding these outliers?

Then, in the localization algorithm, we add an outlier detection step that will select the data (among those available) involved in the location estimation. Specifically, considering filtering algorithms with two steps, prediction and estimation, the outlier detection step is inserted before the estimation step.

Outliers are searched either only in the PR dataset or also in the Dp dataset, depending on the assumption on Dp robustness:

- If Doppler measurements are assumed reliable like in [1], they are directly used to derive \dot{x}_r , and the outlier detection is performed only within the PR set.
- Otherwise, we assume like [7] that, even if the Doppler measurements are less distorted by NLOS reception than PR measurements (and thus, more reliable), both Doppler and PR observations are contaminated by multipaths. Then, the outlier detection is applied for (PR,Dp), so that only Dp inliers are used to derive \dot{x}_r (and only PR inliers are considered for the estimation step further).

3.1. Outlier Detection

The outlier detection is performed using the *a contrario* approach that we proposed in [23] extended to the case of (PR,Dp). The *a contrario* approach detects the inliers as observations that are “too regular to occur by chance”. “Chance” is measured through the Number of False Alarms (NFA), based on two items: a model, called the “naive” model, that represents the statistics of the outliers (the H_0 hypotheses in statistical decision theory) and a measurement that will allow the distinction of inlier and outlier sets under the “naive” model assumption. In [23], we have proposed and compared two “naive” models leading to two NFA criteria for partitioning the data between inliers and outliers. However, these models only deal with PR measurements. In this study, we extend to (PR,Dp) the first NFA criterion that experimentally leads to slightly better results than the second NFA criterion.

Before presenting the extended algorithm, let us specify the equations used and the notations. Assuming a value of ξ_i denoted $\tilde{\xi}_i$ and the satellite features (location and velocity), we are able to compute using Equations (8) and (9) the expected value of PR or Doppler measurement. Then, we can compare these expected ones to the actually observed ones. By definition, the residues are the differences between computed measurements (under the $\tilde{\xi}_i$ hypothesis) and the observed ones: $r_i(t_k)$ associated with the PR observation at t_k , $\rho_i(t_k)$, is:

$$r_i(t_k) = [\tilde{\rho}_i(t_k|\tilde{\xi}_i) - \rho_i(t_k)] \quad (12)$$

where $\tilde{\rho}_i(t_k|\tilde{\xi}_i)$ is computed using Equation (8), and $\dot{r}_i(t_k)$ associated with Doppler observation at t_k , $\dot{\rho}_i(t_k)$, is:

$$\dot{r}_i(t_k) = [\tilde{\dot{\rho}}_i(t_k|\tilde{\xi}_i) - \dot{\rho}_i(t_k)] \quad (13)$$

where $\tilde{\dot{\rho}}_i(t_k|\tilde{\xi}_i)$ is computed using Equation (9).

In order to give the same weight to both kinds of measurements, PR and Doppler ones, $r_i(t_k)$ and $\dot{r}_i(t_k)$ are normalized by their standard deviation, σ_{PR} and σ_{Dp} , respectively, and gathered into vector \mathbf{R} :

$$\mathbf{R} = \left(\frac{r_1(t_{n_{ep}-1})}{\sigma_{PR}}, \dots, \frac{r_m(t)}{\sigma_{PR}}, \frac{\dot{r}_{m+1}(t_{n_{ep}-1})}{\sigma_{Dp}}, \dots, \frac{\dot{r}_M(t)}{\sigma_{Dp}} \right) \quad (14)$$

with M the cardinality of (PR,Dp) set.

As for Equations (10) and (11), several epochs are considered. This allows us to increase the number of available data, as well as the quality of the estimation, provided that the dynamic model (Equations (8) and (9)) used to ‘align’ the data acquired at different epochs is sufficiently accurate. The number of considered epochs, n_{ep} , is then a compromise between data availability and dynamic model approximation. In the following, \mathcal{S}_{PR} and \mathcal{S}_{Dp} denote the sets of available observations (PR and Doppler measurements, respectively) over the considered interval of epochs \mathcal{T} .

Let us now consider a subset of measurements noted \mathcal{D} in the whole set of observations $\{\mathcal{S}_{PR}, \mathcal{S}_{Dp}\}$. Given \mathcal{D} and \mathbf{R} (Equation (14)), $\delta_{\mathcal{D}}^2$ is defined as the sum of the squares of \mathbf{R} components for indices j belonging to \mathcal{D} (indeed, $\{\mathcal{S}_{PR}, \mathcal{S}_{Dp}\}$ measurements being indexed, \mathcal{D} also corresponds to a set of indices). Then, according to [23], $\delta_{\mathcal{D}}^2$ allows us to quantify the consistency of \mathcal{D} through the NFA measure (associated with the Gaussian naive model $\mathcal{N}(0, \sigma)$):

$$NFA_1(\mathcal{D}) = \eta_1 \frac{1}{\Gamma\left(\frac{|\mathcal{D}|}{2}\right)} \int_0^{\delta_{\mathcal{D}}^2/2\sigma^2} e^{-t} t^{\frac{|\mathcal{D}|}{2}-1} dt \quad (15)$$

where Γ is the Gamma function, $|\cdot|$ is the cardinality set operator and η_1 is a normalization term that allows us to control the average number of false alarms [17].

The χ^2 test using the SSE (Sum of Squared Error) is used in the classical RAIM (Receiver Autonomous Integrity Monitoring) method [10,34] to detect the presence of erroneous data. However, it requires an *a priori* parameter, namely the probability of false alarm P_{FA} , to threshold the values

of SSE. Conversely, using the NFA criterion, we are free from the fitting of a threshold parameter, since the solution is derived by optimization of the NFA function: the subset of inliers is the subset of measurements that allows us to reach the minimal value of NFA. Let us underline the difference between the parameter σ involved in the naive model and a threshold parameter: whereas a set of inliers obtained by thresholding will be very sensitive to the used threshold value, we have shown [23] that the subset \mathcal{D} that minimizes the NFA value is very robust to naive model parameter σ .

Algorithm 1 presents the extended version of Algorithm 1 of [23] that allows us to find the subset \mathcal{D} minimizing the NFA criterion. Here, the input parameters are the observation sets \mathcal{S}_{PR} and \mathcal{S}_{Dp} (possibly empty if Doppler measurements are not considered), the number of iterations N_{test} , the parameter σ of naive model \mathcal{M} , the standard deviations, σ_{PR} and σ_{Dp} , for the residue normalization and the binary parameter I_{Dp} that is equal to zero or one, depending on the kind of processed data: only PR data or (PR,Dp), respectively. The output parameters are the subset \mathcal{D} of the inliers and the estimation of $\tilde{\xi}_i$.

Following the *a contrario* RANSAC principle (e.g., [35]), the algorithm performs different estimations or tests (loop until N_{test}) in order to select the best one according to the NFA criterion. Then, for each test, it performs the three following steps. First, the data selection step consists of randomly drawing $d = 8$ elements in \mathcal{S}_{PR} (the set of PR observations) or, if I_{Dp} , $d = 10$ elements in \mathcal{S}_{PR} and \mathcal{S}_{Dp} (the set of Doppler measurements). The numbers eight and 10 correspond to the minimum number of observations to estimate $\tilde{\xi}_1$ or $\tilde{\xi}_2$ further. \mathcal{S}_{PR} and \mathcal{S}_{Dp} include any available observations performed during the considered interval of n_{ep} last epochs. According to [23], the random drawing of observations is biased in order to favor the drawing of favorable configurations of satellites. Since we use a sliding window over epochs, there is an overlapping between the sets of considered epochs for the estimation at two successive instants. Therefore, from the processing of the previous instant, we know the inliers corresponding to previous $n_{ep} - 1$ epochs. Then, like in [23], random drawing is constrained, such that: (i) there is at least one measurement per epoch; (ii) for epochs before the last one, the PR /Doppler measurements are chosen among the already detected inliers; (iii) the selection of different satellites is favored.

These d observations are used to derive a preliminary solution $\tilde{\xi}_1$ or $\tilde{\xi}_2$ (depending on the I_{Dp} value). To derive this solution, a regularization term may be added to Equation (8) or Equation (9), allowing both better conditioning of the problem and the receiver trajectory being smoother. Considering the regularization term, instead of Equation (10), we have to solve Equation (16):

$$\tilde{\xi}_1 = \arg \min_{\xi_1} \sum_{i \in \{1, \dots, d\}} (\tilde{o}_i(\xi_1) - o_i)^2 + \lambda_1^T \left[\text{abs} \left(\xi_1 - \xi_{1,t|t-1} \right) \right] \quad (16)$$

and instead of Equation (11), we have to solve Equation (17):

$$\tilde{\xi}_2 = \arg \min_{\xi_2} \sum_{i \in \{1, \dots, \frac{d}{2}\}} \left[(\tilde{o}_i(\xi_2) - o_i)^2 + \beta (\tilde{\dot{o}}_i(\xi_2) - \dot{o}_i)^2 \right] + \lambda_2^T \left[\text{abs} \left(\xi_2 - \xi_{2,t|t-1} \right) \right] \quad (17)$$

In Equation (16) and Equation (17), $\xi_{i,t|t-1}$, $i \in \{1, 2\}$, is the predicted vector state according to dynamic Model (7); $\text{abs}(v)$ returns the vector of the absolute values of v components; and λ_i , $i \in \{1, 2\}$, is the vector of the regularization parameters (λ weights the importance of the deviation between estimated $\tilde{\xi}_i$ and predicted state vector $\xi_{i,t|t-1}$). The Appendix specifies the derivation of $\xi_{i,t|t-1}$, $i \in \{1, 2\}$.

Algorithm 1: $\tilde{\xi}_i$ Estimation Based on the NFA Criterion with Inputs \mathcal{S}_{PR} , \mathcal{S}_{Dp} , σ , σ_{PR} , σ_{Dp} , I_{Dp} , t , N_{test} and Outputs $\tilde{\xi}$ and \mathcal{D} .

```

1  if  $I_{Dp} \neq 0$  then
2    |  $d \leftarrow 10$ ;  $M \leftarrow 2|\mathcal{S}_{PR}|$ ;
3  else
4    |  $d \leftarrow 8$ ;  $M \leftarrow |\mathcal{S}_{PR}|$ ;
5  end
6  Initialize the vectors  $\delta_{min}[i]$  and  $NFA[i]$  and the scalar  $NFA_{min}$  to  $+\infty$ ;
7  for 1 to  $N_{test}$  do
8    if  $I_{Dp} \neq 0$  then
9      From  $\mathcal{S}_{PR}$  and  $\mathcal{S}_{Dp}$ , respectively, draw randomly  $\frac{d}{2}$  observations  $(x_{S_j}, y_{S_j}, z_{S_j}, \rho_j)$  and  $\frac{d}{2}$ 
      observations  $(\dot{x}_{S_j}, \dot{y}_{S_j}, \dot{z}_{S_j}, \dot{\rho}_j)$  denoted by  $\{o_1, \dots, o_{\frac{d}{2}}\}$  and  $\{\dot{o}_1, \dots, \dot{o}_{\frac{d}{2}}\}$ , respectively;
10     From  $\{o_1, \dots, o_{\frac{d}{2}}, \dot{o}_1, \dots, \dot{o}_{\frac{d}{2}}\}$ , estimate  $\tilde{\xi}_2$  using Equation (11) or Equation (17) for a
      regularized solution;
11     Using Equations (12), (13) and (14), compute the vector of normalized residues, denoted
       $R$ ;
12   else
13     From  $\mathcal{S}_{PR}$ , draw randomly  $d$  observations  $(x_{S_j}, y_{S_j}, z_{S_j}, \rho_j)$ , denoted by  $\{o_1, \dots, o_d\}$ ;
14     From  $\{o_1, \dots, o_d\}$ , estimate  $\tilde{\xi}_1$  using Equation (10) or Equation (16) for a regularized
      solution;
15     Using Equation (12), compute the vector of residues, noted  $R$ ;
16   end
17   Sort  $R$  by increasing values into a vector  $R' = (r'_j)_{j \in \{1 \dots M\}}$  with  $\pi(\cdot)$  the index
      permutation function, such that  $\forall j \in \{1 \dots M\}, r'_j = R_{\pi(j)}$ ;
18   Initialize  $\delta_{\mathcal{D}}^2 = 0$ ;
19   for  $j = d + 1$  to  $M$  do
20      $\delta_{\mathcal{D}}^2 \leftarrow \delta_{\mathcal{D}}^2 + (r'_j)^2$ ;
21     if  $\delta_{\mathcal{D}}^2 < \delta_{min}[j]$  then
22        $\delta_{min}[j] = \delta_{\mathcal{D}}^2$ ;
23       Compute  $NFA[j]$  according to Equation (15);
24       if  $NFA[j] < NFA_{min}$  then
25         |  $NFA_{min} = NFA[j]$ ;  $\mathcal{D} = \{\pi(1) \dots \pi(j)\}$ ;
26       end
27     end
28   end
29 end
30 From the inlier set  $\mathcal{D}$ , estimate  $\hat{\xi}_1$  or  $\hat{\xi}_2$  using Equation (18).
```

The second part of the algorithm computes the non-null residues for all of the other (not drawn) observations, either only PR or (PR,Dp). Having increasingly sorted the vector of residues, the last part of the algorithm computes the minimum NFA values by varying the cardinality of \mathcal{D} (increasing from $d + 1$ to M). $\delta_{min}[i]$ is a vector that stores the values of the minimal quadratic errors (sum of the squares of the residues) for every cardinality of subset \mathcal{D} . Indeed, for a given cardinality of \mathcal{D} , the NFA value is minimum for minimum value of quadratic error $d_{\mathcal{D}}^2$ that is achieved considering the $|\mathcal{D}|$ lowest values of residues (hence, the sorting of R). Then, $NFA_{min}[i]$ is a vector that stores the NFA values corresponding to $\delta_{min}[i]$; NFA_{min} is the minimum among $NFA_{min}[i], \forall i \in \{d + 1, \dots, M\}$. The

inlier subset is the set \mathcal{D} achieving the NFA_{min} value. Finally, state vector $\tilde{\xi}_1$ or $\tilde{\xi}_2$ is estimated from \mathcal{D} and Equation (18):

$$\tilde{\xi}_i = \arg \min_{\xi_i} \sum_{j \in \mathcal{D}} R_j + \lambda_i^T \left[\text{abs} \left(\xi_i - \xi_{i,t|t-1} \right) \right] \quad (18)$$

where R_j is the residue provided by Equation (14).

Algorithm 1 has a linear complexity with N_{test} . For one iteration, the complexity mainly comes from state vector estimation (Algorithm A1, Appendix). The complexity of this latter depends on d : matrix inversion and matrix multiplication are in $O(d^3)$. Then, the complexity of the sorting of \mathbf{R} is in $O(M \log(M))$. For NFA(PR,Dp), $d = 10$, and M varies in [12, 33] considering a temporal window of three epochs. Therefore, to control the computation time, one should fit the parameter N_{test} .

Finally, note that, even if Algorithm 1 provides estimations of GNSS receiver localization parameters, the proposed coupling between Algorithm 1 and the robust localization algorithm (PF/RBPF presented in the next section) is only done in terms of data selection. Indeed, in Algorithm 1, the provided estimation only aims at evaluating the consistency of a subset of data, whereas PF/RBPF allows for non-linear/non-Gaussian data filtering that exploits some classic *a priori* parameter on the smoothness of the trajectories. Such an independence between the detection step (Algorithm 1) and the filtering step (PF/RBPF) increases the robustness of the global localization algorithm.

3.2. Localization Algorithm

The particle filter, also called the Sequential Monte Carlo (SMC) method, is a numerical method that consists of approximating the posterior probability $p(x_t|z_t)$ (probability of the state x_t given the set of observations z_t) using a sufficient number of particles x_t^i . A particle represents a state vector solution, and the associated weight w_t^i represents its likelihood. Such a representation based on samples/particles allows us to approximate and deal with any statistical distribution of error, especially non-parametric ones and non-Gaussian ones.

3.2.1. SIR-PF

The Sequential Importance Resampling (SIR) particle filter [13], also known as the “bootstrap filter”, is the most popular method to solve the non-linear filtering problem.

For SIR-PF, the number of the required particles is directly linked to the dimensionality of the state vector. In order to keep a reasonable number of particles (bounded to a few thousands), we assume that either the altitude is constant, as is often in urban environments, or it is known as in our case from the output $\tilde{\xi}$ of Algorithm 1, so that it has not been introduced in the state vector. For the same reasons, velocity is also excluded from the state vector (conversely to the RBPF state vector presented in the next section). Then, the SIR-PF particles are $x_t^i = (e_t^i, n_t^i, \delta_t^i)^T$, where i denotes the particle index and t is the epoch.

At each epoch, the SIR-PF iterates the three steps “prediction”, “estimation” and “resampling”.

Prediction Step

This step, sometimes called PF time update, aims at providing an estimation of the state vector at the next time step. Note that if here, we place it at the beginning of iteration at time t , it can equivalently be placed at the end of iteration at $t - 1$.

To predict the next position of the particle, we need an estimation of the velocity \dot{x}_r . Since, \dot{x}_r is not part of the state vector, it should be provided by external data. Using GPS-only data, we consider Doppler measurements to derive \dot{x}_r : Doppler measurements at time $t - 1$ provide PR rates from which we derive the receiver velocity $\dot{x}_r = (\dot{e}_r, \dot{n}_r, \dot{u}_r)$ using Equation (3). In order to comply with common notations in the transportation and navigation community, \dot{x}_r can be equivalently represented in terms of norm and orientation: $V_{t-1}^i = \text{sqr}t(\dot{e}_r^2 + \dot{n}_r^2)$ and $\theta_{t-1}^i = \arctan(\frac{\dot{n}_r}{\dot{e}_r})$, respectively. Then, we predict the next state at t of the i -th particle according to:

$$\begin{cases} e_t^i &= e_{t-1}^i + V_{t-1}^i \cos(\theta_{t-1}^i) dt + v(\sigma_e) \\ n_t^i &= n_{t-1}^i + V_{t-1}^i \sin(\theta_{t-1}^i) dt + v(\sigma_n) \\ \delta_t^i &= \delta_{t-1}^i + \dot{\delta}_{t-1} dt + v(\sigma_{\delta_t}) \end{cases} \quad (19)$$

where the time step dt is equal to one and v is the prediction noise associated with each component of the state vector. Indeed, as a stochastic approach, PF is based on stochastic simulations provided here by the addition (to the deterministic predictions state vectors) of a Gaussian noise with zero mean and standard deviation $(\sigma_e, \sigma_n, \sigma_{\delta_t})$.

Note that in our case, the velocity used for prediction is estimated from (\dot{e}_r, \dot{n}_r) at $t - 1$. Instead of using Doppler measurements at $t - 1$, we could have used those acquired at t . However, since the prediction is between $t - 1$ and t , it will not provide necessarily a more accurate prediction. In comparison with the RBPF (presented in Section 3.2.2), let us underline that the velocity estimation is performed as an external process to the SIR-PF itself, since velocity is not a part of the state vector.

Estimation Step

This step, sometimes called PF measurement update, aims at correcting the prediction step estimate according to the observations. Since velocity is not represented in the state vector \mathbf{x}_t^i , the posterior probability of our SIR-PF is only computed relatively to the PR measurements. It is denoted $p(\boldsymbol{\rho}_t | \mathbf{x}_t^i)$ with $\boldsymbol{\rho}_t$ the vector of ρ_i observed at t .

The update process of weights w_t^i is a weighting function of their previous values [36] by the observation likelihood function $p(\boldsymbol{\rho}_t | \mathbf{x}_t^i)$: $w_t^i \propto w_{t-1}^i p(\boldsymbol{\rho}_t | \mathbf{x}_t^i)$. In most cases, because of computational constraints, the likelihood function $p(\boldsymbol{\rho}_t | \mathbf{x}_t^i)$ is approximated by a multivariate Gaussian density. Finally, normalization of the weights is performed so that $\sum_{i=1}^k w_t^i = 1$.

Having updated the weights, the ‘optimal’ state vector $\hat{\mathbf{x}}_t$ is derived as the weighted sum of all particles:

$$\hat{\mathbf{x}}_t = \sum_{i=1}^k w_t^i \mathbf{x}_t^i \quad (20)$$

Resampling

This step aims at preventing the degeneracy of the algorithm, in particular to avoid that computer resources are consumed by “unlikely” particles. During this step, a threshold is computed [36] to partition the set of the particles according to their weight [13]. Having removed the particles that present lower weight than the considered threshold, the remaining particles are duplicated in order to keep a constant number of particles, and all of the weights are reinitialized to a constant value (reciprocal of the total number of particles).

3.2.2. Rao-Blackwellised PF

In previous PF, the velocity was estimated directly from Doppler measurements (being ‘outside’ of the PF estimation step, it does not take into account previous estimations of the PF prediction step). This boils down to assuming no noise on Doppler measurements. In order to avoid such an assumption and to be more realistic, we extend the state vector from $(e, n, \delta_t)^T$ to $(e_r, n_r, \delta_t, \dot{e}_r, \dot{n}_r, \dot{\delta}_t, \ddot{e}_r, \ddot{n}_r)^T$, i.e., its dimensionality increases from three to eight.

However, standard PF would require a very important number of particles to explore the whole space of solutions, and the PF would become intractable. On the other hand, the Rao-Blackwellization approach [37,38] was proposed both to reduce the complexity and to better approximate the solution in case of convex functions. It is based on the idea that splitting the state vector allows us to decrease the approximate error by exploiting linear substructures [25]. A classic case corresponds to the splitting of the initial state vector into two sub-vectors, one being estimated analytically and the other one by

importance sampling (e.g., PF). Thus, the number of particles required for precise estimation remains tractable thanks to the lower dimensionality of the non-linear subsystem [25,38].

Considering our problem, we split the system of eight components describing the prediction step equations into two sub-systems, a linear and a non-linear one, as follows. The equations involving PR observations (Equation (1)) are non-linear leading to a non-linear system for deriving GPS position. On the other hand, the velocity estimation knowing the position of the receiver and the Doppler measurements is achieved solving a linear system (Equation (3)). Thus, we define the two state vectors $\mathbf{x}_{pf} = (e_r, n_r, \delta_t)^\top$ and $\mathbf{x}_{kf} = (\dot{e}_r, \dot{n}_r, \dot{\delta}_t, \ddot{e}_r, \ddot{n}_r)^\top$.

The posterior probability of the RBPF is factorized:

$$p(\mathbf{x}_{kf,t}, \mathbf{x}_{pf,t} | \mathbf{z}_t) = p(\mathbf{x}_{kf,t} | \mathbf{x}_{pf,t}, \mathbf{z}_t) p(\mathbf{x}_{pf,t} | \mathbf{z}_t) \quad (21)$$

where \mathbf{z}_t still denotes the set of observations. The first term is solved analytically using EKF, and the second term is estimated by Monte Carlo sampling using PF. Then, in RBPF, we can keep the same number of particles as in Section 3.2.1, while considering also the receiver velocity in the state vector and filtering it. The proposed model for RBPF is triangular:

$$\begin{pmatrix} \mathbf{x}_{pf,t} \\ \mathbf{x}_{kf,t} \end{pmatrix} = \begin{pmatrix} \mathbf{I}_{3 \times 3} & \mathbf{A}_{pf,dt} \\ \mathbf{0}_{5 \times 3} & \mathbf{A}_{kf,dt} \end{pmatrix} \begin{pmatrix} \mathbf{x}_{pf,t-1} \\ \mathbf{x}_{kf,t-1} \end{pmatrix} + \begin{pmatrix} \mathbf{Q}_{pf} \\ \mathbf{Q}_{kf} \end{pmatrix} \quad (22)$$

where $\mathbf{I}_{n \times n}$ is the square identity matrix of dimensionality n , $\mathbf{0}_{m \times n}$ is the rectangular zero matrix of dimensionality $m \times n$, \mathbf{Q}_{pf} and \mathbf{Q}_{kf} are the covariance matrices of the noise, which is assumed zero mean Gaussian (for notation shortness, we omitted the time dependency for covariance matrices) and $\mathbf{A}_{pf,dt}$ and $\mathbf{A}_{kf,dt}$ are the transition matrices defined as follows:

$$\mathbf{A}_{pf,dt} = \begin{pmatrix} dt & 0 & 0 & \frac{dt^2}{2} & 0 \\ 0 & dt & 0 & 0 & \frac{dt^2}{2} \\ 0 & 0 & dt & 0 & 0 \end{pmatrix} \quad (23)$$

$$\mathbf{A}_{kf,dt} = \begin{pmatrix} 1 & 0 & 0 & dt & 0 \\ 0 & 1 & 0 & 0 & dt \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad (24)$$

The non-linear part is processed using the same PF presented in Section 3.2.1 to estimate the state vector of each particle $\mathbf{x}_{pf(i),t}$ and its associated weight w_t^i . The linear part is processed using an EKF applied to the state vector $\mathbf{x}_{kf(i),t}$ of each particle recursively. EKF involves two main steps:

Prediction Step

This step occurs between the prediction step and the estimation step of the SIR-PF. We define intermediate variables,

$$\mathbf{N}_t = \mathbf{A}_{pf,dt} \mathbf{P}_{t-1|t-1} \mathbf{A}_{pf,dt}^\top + \mathbf{Q}_{pf} \quad (25)$$

$$\mathbf{L}_t = \mathbf{A}_{kf,dt} \mathbf{P}_{t-1|t-1} \mathbf{A}_{pf,dt}^\top \mathbf{N}_t^{-1} \quad (26)$$

$$\mathbf{y}_t = \mathbf{x}_t^{pf} - \mathbf{x}_{t-1}^{pf} \quad (27)$$

where y_t is interpreted as an error measurement and L_t and N_t are intermediate matrices modeling the impact of the non-linear system on the linear estimation. Then,

$$\hat{x}_{kf,t|t-1} = A_{kf,dt} \hat{x}_{kf,t-1|t-1} + L_t (y_t - A_{pf,dt} \hat{x}_{kf,t-1|t-1}) \quad (28)$$

$$P_{t|t-1} = A_{kf,dt} P_{t-1|t-1} A_{kf,dt}^T + Q_{kf} - L_t N_t L_t^T \quad (29)$$

where $P_{t-1|t-1}$ is the covariance matrix of x_t^{kf} . Note that if the A_t^{pf} matrix is null, previous equations boil down to Kalman's filter prediction step. Note that, since the prediction step presented in Section 3.2.1 is involved in Equation (27), the current prediction step occurs after the prediction of the non-linear part of RBPF.

Estimation Step

This step occurs between the estimation step and the resampling of the SIR-PF. It is the classical correction step of the extended Kalman filter.

$$\hat{x}_{kf,t|t} = \hat{x}_{kf,t|t-1} + K_t (\dot{\rho}_t - C_t \hat{x}_{kf,t|t-1}) \quad (30)$$

$$P_{t|t} = P_{t|t-1} - K_t C_t P_{t|t-1} \quad (31)$$

$$K_t = P_{t|t-1} C_t^T (C_t P_{t|t-1} C_t^T + R_t)^{-1} \quad (32)$$

where C_t is the observation matrix of Doppler measurements derived from Equation (3).

This analytical correction of the $\hat{x}_{kf,t|t}$ subvector is independent from the estimation of $\hat{x}_{pf,t|t}$ that is performed according to the estimation step presented in Section 3.2.1 (Equation (20)).

One of the objectives of this study was to check the interest of removing outliers from the datasets, either PR or (PR,Dp). This can be achieved by comparing the localization results obtained using outlier detection coupled with PF or RBPF.

4. Experiment and Results

In order to test our localization method, we have acquired data in constrained environments: an urban canyon and forest, characterized by NLOS reception. Figure 1b shows the receiver trajectory in the South of Paris (France). It is 5 km long for an experiment duration of 11 min.

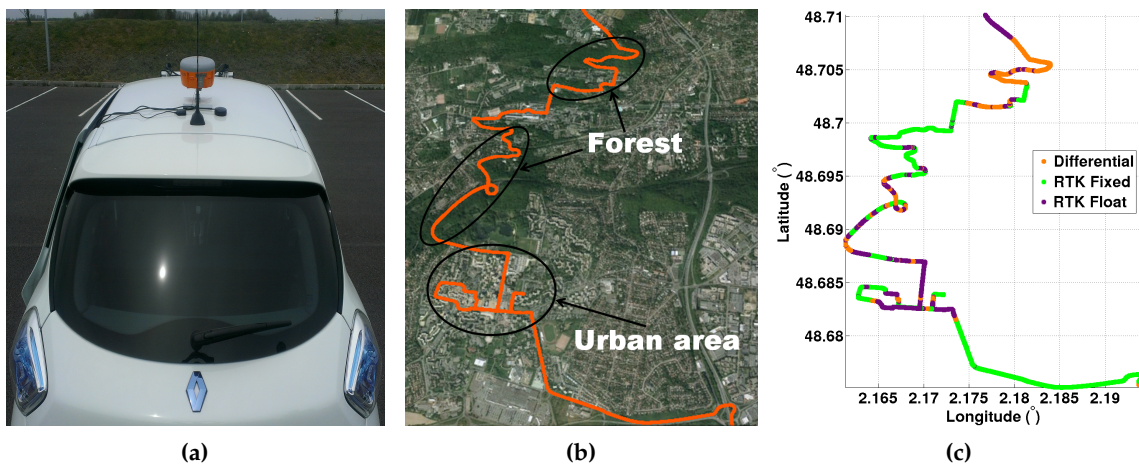


Figure 1. (a) Experimental platform with the three GPS visible on the roof of the car; (b,c) trajectory of the experiment, either (b) plotted on Google Earth[®] or (c) labeled in terms of the quality of the Real-Time Kinematic (RTK) solution (“ground truth”).

4.1. Platform and Parameters

Figure 1a shows the used experimental vehicle ZOE that is equipped with two low cost GPS and one high cost GPS. The two low cost GPS are GARMIN 18x and UBLOX EVK-5T, which are single-frequency receivers delivering the positioning solution at 1 Hz. The high cost GPS is an APS-3 multi-frequency and multi-constellation receiver (L1/L2/L2C GPS, GLONASS and satellite-based augmentation system (SBAS)) that belongs to the GPS class RTK (Real-Time Kinematic). This latter has a sampling frequency equal to 1 Hz, and its location accuracy is equal to 1 cm, according to factory specifications in the case of the “fixed solution”. This solution was available during 41% of the experiment (cf. Figure 1c), whereas the two other solutions, ‘RTK float’ and ‘differential’, whose precision may drop until 40 cm, were available during 39% and 20%, respectively. APS-3 is used for two purposes: to establish the ground truth and to get the raw data used in the localization algorithms. However, the considered raw data are not post-processed by APS-3.

The GARMIN 18x has an accuracy (measured by the root mean square error) equal to 5 m in location and 0.05 m/s in velocity. Finally, the UBLOX EVK-5T acquires only PR measurements (no Doppler measurements) and is specified to have a location accuracy of 3 m in the static case and an open area.

The configuration of the satellites during the experiment is shown in Figure 2. The number of available satellites varies between four and 11 with an average equal to nine.

For the used algorithms, the parameters are:

- In Algorithm 1, $n_{ep} = 3$, $\lambda_1^T = (200 \ 20 \ 200 \ 20 \ 10000 \ 20 \ 0 \ 0)$ and $\lambda_2^T = (20 \ 2 \ 20 \ 20 \ 2 \ 20 \ 10000 \ 10000 \ 0 \ 0)$;
- In EKF, SIR-PF and RBPF, the PR precision is $\sigma_{PR} = 5$ m, and the Dp precision is $\sigma_{Dp} = 2$ Hz;
- In SIR-PF and RBPF, the number of particles is set to 3000.

4.2. Localization Results

The global performance of the localization is represented in terms of the cumulative distribution curve: the better the result, the greater the area below the cumulative distribution curve. In this study, we consider eleven localization processes. Two of them are GPS solutions themselves: either the UBLOX or the GARMIN GPS. The GARMIN and UBLOX EVK-5T solutions are plotted just as references, since it would be unfair to compare high cost and low cost GPS. However, we note that the GARMIN solution seems rather interesting, and even if the GARMIN algorithm is unknown, we may guess that it uses preprocessing of the measurements. For instance, if it uses the satellite elevation mask (discarding the satellites having an elevation lower than 15°), according to Figure 2, the satellites S_4 , S_{10} , S_{11} , S_{31} and S_{32} will not be used, which corresponds to frequent outliers, as we will see further.

The other processes correspond to different versions of the extended Kalman filter, the particle filter and the Rao-Blackwellised PF: without removing any outliers, by coupling it with the PR outlier detection or with the (PR,Dp) outlier detection. In the three filters (EKF, PF and RBPF), the initial solution is provided either by the least mean square solution or by the output of Algorithm 1 when there is an outlier detection step. For comparison, we also implement a recent robust outlier method called ORKF (Outlier Robust Kalman Filter) [39]. It is similar to the EKF, except that the covariance of the observation noise is estimated recursively inside the estimation step (releasing the assumption on the measurement precision).

Figure 3 and Table 1 allow us to draw the following conclusions:

- Among the implemented algorithms, the Particle Filter (PF) provides rather disappointing results with an error lower than 6 m in only 55% of cases. This relatively bad performance of PF, against EKF for instance, is probably due to the fact that the velocity is not part of the state vector; it is not at all filtered, conversely to the case of the EKF.
- The ORKF has better performance than the simplest version of PF and the classical EKF, and similar performance to EKF + NFA (PR) and EKF + NFA (PR + Dp) when the errors are less than 6 m.

- By removing the PR outliers at the entry of the filters, EKF + NFA (PR) and PF + NFA (PR) allow for much better localization than the ‘all-data’ EKF, PF or even ORKF for errors lower than 6 m. Besides, if EKF + NFA (PR) still performs better than PF + NFA (PR) for errors lower than 6 m, the gap has narrowed, and in terms of errors lower than 3 m, PF + NFA (PR) outperforms EKF + NFA (PR).
- By removing also the Dp outliers, PF + NFA (PR,Dp) provides better results than the previous methods. For instance, its 95th percentile corresponds to an error lower than 9 m, whereas PF + NFA (PR) percentile error is 11.5 m. This clearly illustrates the interest of removing also the Doppler outliers, especially as they are not filtered (by the estimation step of PF). Conversely, in the case of the EKF where velocities are filtered, the effect of removing Dp outliers is less clear: it appears just for errors lower than 9 m.
- By removing the PR outliers, RBPF + NFA (PR) has the same performance in localization as the PF + NFA (PR,Dp) version (see Table 1). This can be explained by the fact that, by filtering the velocity estimation, RBPF is rather robust to outliers in Doppler measurements. It also outperforms EKF + NFA (PR).
- Finally, removing also the Dp outliers, RBPF + NFA (PR,Dp) outperforms all of the other results. According to Table 1, if the performance for PR + NFA (PR,Dp) and the two RBPFs is close under 3 m, a higher level of confidence is achieved by RBPF + NFA (PR,Dp) for errors lower than 6 m and 9 m.

Table 1. Percentiles of positioning errors. NFA, Number of False Alarms; PR, Pseudo-Range; Dp, Doppler measurement; RBPF, Rao–Blackwell Particle Filter; ORKF, Outlier Robust Kalman Filter.

Localization Method	% Error < 3 m	% Error < 6 m	% Error < 9 m
UBLOX	20.9%	47.15%	64.92%
GARMIN	28.6%	72.97%	90.72%
EKF	37.26%	71.66%	80.75%
EKF + NFA (PR)	40.94%	81.82%	91.73%
EKF + NFA (PR,Dp)	37.13%	74.88%	96.49%
ORKF	40.83%	74.77%	83.22%
PF	21.1%	55.02%	75.12%
PF + NFA (PR)	44.6%	77.19%	89.85%
PF + NFA (PR,Dp)	59.95%	87.05%	94.61%
RBPF + NFA (PR)	61%	85.78%	93.38%
RBPF + NFA (PR,Dp)	61.96%	90.11%	98.28%

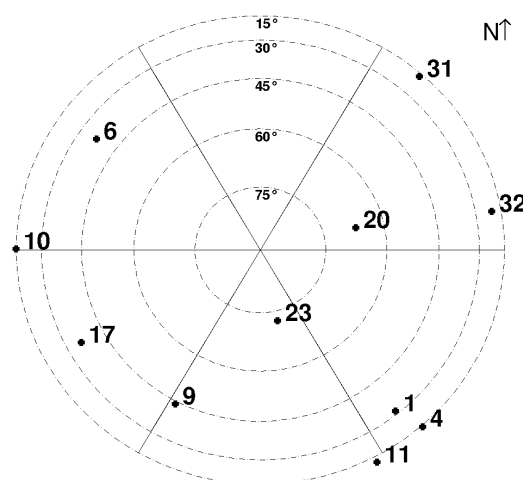


Figure 2. Skyplot configuration during the experimental data acquisition in the urban area.

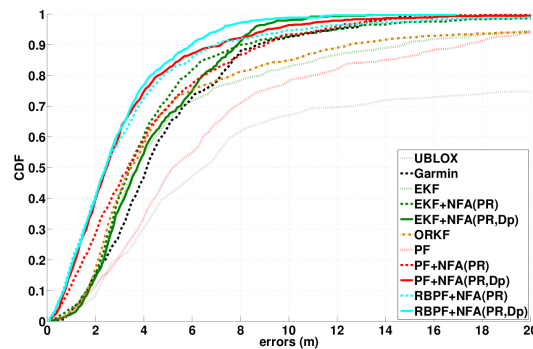


Figure 3. Cumulative distribution function of errors achieved by the four versions of KF, the five versions of particle filters and the two GPS solutions for our experiment of 11 min and 40 s.

Table 2 shows the global precision of the localization. Precision was evaluated through three indicators: the $Norm_1$ norm, the $Norm_2$ and the mean and standard deviation of errors. $Norm_1$ and $Norm_2$ can be computed on east and north coordinates: precisely, denoting ϵ_i the error of the position at instant i along a given direction (east or north), $Norm_1 = \sum_{i=1}^n |\epsilon_i|$ is the average of the absolute value of the errors, and $Norm_2 = \sqrt{\sum_{i=1}^n \epsilon_i^2}$ is the root of average of the squared errors. Denoting E_i the Euclidean distance between estimated and ‘ground truth’ positions at instant i , μ_{loc} and σ_{loc} are the mean and the standard deviation of E_i values. The results are consistent with Figure 3: Among the implemented algorithms, when using all-data, EKF and ORKF show good performance, and when removing the outliers (either PR or (PR,Dp)), RBPF outperforms the other approaches. The best results are obtained for NFA (PR,Dp) coupled with RBPF, even if the interest of removing outliers can also be noticed in the case of EKF or PF. Finally, to quantify the improvement due to the NFA outlier detection, we run RBPF with an elevation mask removing satellites below 15° (as is usually done on most GNSS receiver devices). The results are: $Norm_1 = (2.82, 3.10)$, $Norm_2 = (4.68, 4.38)$ and $(\mu_{loc}, \sigma_{loc}) = (4.64, 4.42)$. As expected, localization is less accurate than RBPF + NFA (PR,Dp) or even RBPF + NFA (PR), showing that the satellite elevation criterion does not exactly fit the outlier detection.

Table 2. Localization error (in m) on (east, north) coordinates, $Norm_1$ and $Norm_2$ of error, error mean and standard deviation: comparison of the four versions of KF, the five versions of particle filters and the two GPS solutions on our 11 min 40 s experiment.

Error Measure	Localization Algorithm	Data		
		All-Data	NFA (PR) Inliers	NFA (PR,Dp) Inliers
L_1	UBLOX	(11.92,10.20)	-	-
	GARMIN	(3.35,2.76)	-	-
	EKF	(3.76,4.50)	(2.63,3.18)	(3.31,2.24)
	ORKF	(3.55,4.31)	-	-
	PF	(6.68,6.72)	(2.61,2.83)	(1.82,2.41)
	RBPF	-	(1.84,2.69)	(1.62,2.17)
L_2	UBLOX	(20.44,18.60)	-	-
	GARMIN	(4.73,3.35)	-	-
	EKF	(5.77,7.47)	(3.43,5.00)	(3.92,3.09)
	ORKF	(5.55,7.79)	-	-
	PF	(9.09,9.49)	(3.48,3.86)	(2.95,3.51)
	RBPF	-	(3.37,3.53)	(2.51,3.20)
$(\mu_{loc}, \sigma_{loc})$	UBLOX	(16.72,22.02)	-	-
	GARMIN	(4.91,3.08)	-	-
	EKF	(6.40,6.96)	(4.59,3.96)	(4.37,2.42)
	ORKF	(6.13,7.36)	-	-
	PF	(10.43,7.99)	(4.25,3.41)	(3.37,3.11)
	RBPF	-	(3.53,3.56)	(2.96,2.25)

Table 3 shows the localization error computed on the three subparts of the trajectory corresponding to the three RTK solution qualities. The localization results are those obtained using RBPF with removal of outliers, either in the PR dataset or in the (PR,Dp) one (we focus on the best results), and the considered errors are computed as previously in terms of $Norm_1$, $Norm_2$ on east and north coordinates and the mean and standard deviation of the distance between estimation and ground truth. From Table 3, we observe a ‘correlation’ between the quality of the localization result and the RTK quality: localization is more precise on the RTK fixed part than on the RTK float part, and the differential part presents the worst localization results. There are two interpretations of such a fact: (i) the imprecision of the ground truth in the case of RTK float or the differential solutions introduces a supplementary error that slightly degrades the *estimated* precision of the localization; (ii) the RTK fixed solution occurs mainly in open areas (whereas the RTK float solution also occurs in an urban environment and the differential solution in the forest part; cf. Figure 1b and 1c) where localization is generally good. Indeed, looking at the localization precision distribution *versus* the RTK solution for other methods, we also note that the results are more precise on the RTK fixed part of the trajectory.

Table 3. Proposed method localization error (in m) on (east, north) coordinates, $Norm_1$ and $Norm_2$ of error, error mean and standard deviation *versus* the quality of RTK solution used as the ground truth.

	Solution Quality	RBPF + NFA (PR)	RBPF + NFA (PR,Dp)
L_1	RTK fixed	(1.44,2.08)	(1.27,1.74)
	RTK float	(2.21,3.03)	(1.91,2.55)
	Differential	(2.16,4.16)	(2.03,2.62)
L_2	RTK fixed	(2.56,3.14)	(1.76,2.45)
	RTK float	(3.19,4.35)	(2.56,3.26)
	Differential	(3.70,6.02)	(2.75,3.85)
$(\mu_{loc}, \sigma_{loc})$	RTK fixed	(2.74,2.99)	(2.38,1.85)
	RTK float	(4.05,3.57)	(3.44,2.31)
	Differential	(5.08,4.96)	(3.68,3.00)

4.3. Validation of the Outlier Estimation

In this section, we aim at checking the efficiency of Algorithm 1 in terms of outlier detection. The tricky part is the derivation of a ‘ground truth’ in terms of outliers. First of all, note that the definition of an outlier itself depends on the adopted point of view: from the statistical point of view, an outlier is a measurement *considerably dissimilar or inconsistent with the remainder of the data* [40], whereas from the physical point of view and according to the considered application, an outlier is then a measurement affected by multipath or NLOS reception. In this study, we adopt the statistical definition, and we derive an estimation of the biases, like in [7], as follows.

Among the (PR,Dp) set, we want to derive the subset of observations that behave as outliers from the statistical point of view. The only “ground truth” we have is the receiver position that is provided by the APS-3 GPS + GLONASS RTK. The construction of a “ground truth” about outliers from this ground truth about receiver localization proceeds in two steps: (i) firstly, estimation of the biases between observed measurements and expected ones; (ii) secondly, analysis of the biases to classify them as induced by outliers or by inliers.

4.3.1. Bias Estimation for Qualitative Analysis

For the first step, we have to estimate the ‘expected’ measurements from the receiver localization ground truth. This latter allows us to derive the Euclidean distance between the satellite S_j and the receiver position, d_j . However, we still need to estimate the clock bias $\tilde{\delta}_t$. In [7], the equation $c\tilde{\delta}_t = \frac{1}{N} \sum_{j=1}^N (\rho_j - d_j)$ was used. However, the mean estimator is not robust to outlier presence nor to

the fact that the oscillator embedded on GPS receivers is not stable nor accurate. Then, we rather use the M estimator [41] as a simple solution among robust estimator class:

Assuming e the vector of residues of clock bias estimation ($e_j = \rho_j - d_j - c\hat{\delta}_t$), $\alpha(e_j)$ is the weight coefficient defined by $\alpha(e_j) = \frac{1}{|e_j|}$, and the optimal clock bias $c\hat{\delta}_t$ and the PR bias estimate $\tilde{\Delta}m_j$ are:

$$\begin{cases} c\hat{\delta}_t = \frac{\sum_{j=1}^N \alpha(e_j)(\rho_j - d_j)}{\sum_{j=1}^N \alpha(e_j)} \\ \tilde{\Delta}m_j = \rho_j - d_j - c\hat{\delta}_t \end{cases} \quad (33)$$

In a similar way, we derive the biases $\tilde{\Delta}m_j$ on Doppler measurements knowing both the velocity and location of the GPS receiver. Figures 4 and 5 allow us to check qualitatively the consistency between the large biases (either in PR or Doppler measurements) and NFA detected outliers. Specifically, they show the temporal variation of the estimated biases for PR and Doppler observations of each satellite, and the values detected as outliers by the NFA algorithm are pointed out (by a red marker). We also observe that the estimated signal in Equation (33) is probably affected by atmospheric and electronic noises that differ from one satellite to another. This satellite specificity induces different biases even between consistent curves (e.g., see the S_{17} , S_{20} and S_{23} curves in Figure 4). In the Dp case, the estimation is less affected by atmospheric noise, so that the peaks in Figure 5 corresponding to potential NLOS reception or multipaths reception appear clearly.

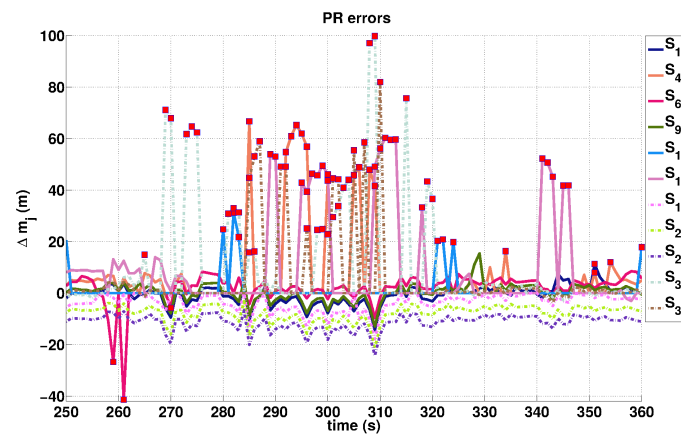


Figure 4. $\tilde{\Delta}m_j$ estimations on PR measurements acquired by the different satellites (numbered between 1 and 32). Red markers point out PR outliers detected by NFA.

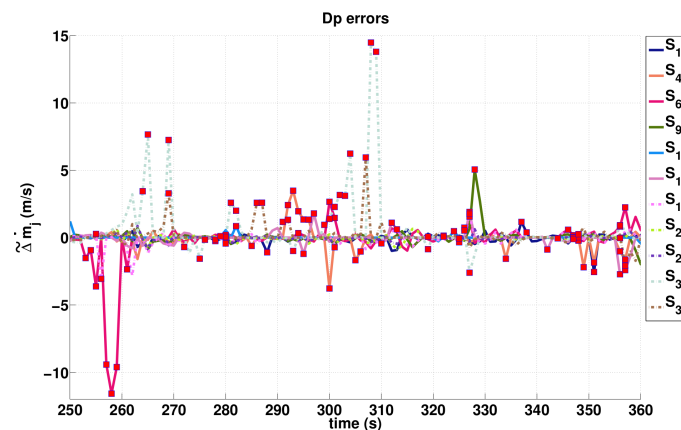


Figure 5. $\tilde{\Delta}m_j$ estimations on Doppler measurements acquired by the different satellites (numbered between 1 and 32). Red markers point out Dp outliers detected by NFA.

4.3.2. Bias Classification for Quantitative Analysis

To evaluate quantitatively the efficiency of outlier detection, we have to label the previously estimated biases either as (induced by) “outlier” or as “inlier”. Such a labeling was done only for the $\tilde{\Delta}m_j$ (due to the objective difficulty of labeling the $\tilde{\Delta}m_j$) by a human operator as follows. For every epoch t of the experiment, a bias $\tilde{\Delta}m_j$ is labeled “outlier” if, at t , it appears neither consistent with the average bias of the considered satellite nor with the other satellite biases. Practically, a thresholding step relative to the median value of all $\tilde{\Delta}m_j$ at t is first applied (the test of consistency with other satellites), then followed by visual inspection of the selected biases. For instance, at time 351 s, even if S_{23} presents a rather important $\tilde{\Delta}m_j$ value, only S_1 and S_4 are labeled as outliers. Even if previous labeling includes a subjective part, we assume that it is *statistically* unbiased, and we use it to analyze *statistically* outlier detection results.

From previous bias labeling, on the one hand, and inlier set \mathcal{D} provided by Algorithm 1, we compute the numbers of True Positives (TP , $PR \in \mathcal{D}$ with bias label “inlier”), false alarms called False Positives (FP , $PR \in \mathcal{D}$ with bias label “outlier”), misdetections called False Negatives (FN , $PR \notin \mathcal{D}$ with bias label “inlier”) and True Negatives (TN , $PR \notin \mathcal{D}$ with bias label “outlier”). From these statistics, the accuracy $\frac{(TP+TN)}{(TP+TN+FP+FN)}$ and precision $\frac{(TP)}{(TP+FP)}$ are derived. The sample set includes 3498 PR samples corresponding to a sub-part of the experiment (of 7 mn) where biases $\tilde{\Delta}m_j$ were labeled. Table 4 shows the obtained values. The two presented coupling ways between particle filter and outlier detection either restricted to the PR measurements or for (PR,Dp) are called “NFA (PR)” and “NFA (PR,Dp)”, respectively. By comparing these two approaches, we note that the performance of both of them is very high. Besides, they appear very close given the statistical imprecision and the labeling process.

Table 4. Performance of Algorithm 1 for outlier detection among PR measurements or (PR,Dp).

	TP	FP	FN	TN	Accuracy	Precision
NFA (PR)	3131	39	49	279	97.5	98.7
NFA (PR,Dp)	3112	91	45	250	96.1	97.2

4.3.3. Correlation between PR and Doppler Outliers

Having, at least qualitatively, positively assessed the outlier detection, we can interpret its result also in terms of the occurrence of Doppler outliers.

In terms of global statistics and correlation between PR and Doppler outliers, during the experiment, NFA (PR) excludes 9.83% of available PR observations, whereas NFA (PR,Dp) discards less PR observations (8.37%), but discards 2.85% of available Doppler observations. Among the Doppler outliers, 54% are also PR outliers. Thus, one can deduce that, according to these statistics, in constrained environments, Doppler measurements present three-times less outliers than PR measurements, but they, nevertheless, suffer from NLOS or multipath phenomena.

5. Conclusions

In this paper, a new approach able to cope with a significant number of outliers was presented for GNSS positioning. Based on a *contrario* modeling, the Number of False Alarms (NFA) criterion allows us to partition the pseudo-range and Doppler measurements between inliers and outliers. Then, detected outliers are removed from the dataset to achieve robust estimation of receiver position and velocity. Two models based on particle filtering have been considered for the localization process. The first model (PF) only filters the receiver position, whereas the second model (RBPF) is a more complete filter that handles receiver position and velocity and using both PR and Doppler observations in its estimation step. Tests have been performed in the case of a receiver on board a vehicle traveling in urban canyons and forest areas. Results show that, by excluding erroneous measurements and filtering the noise of the observations, more accurate localization is achieved.

Future work will deal with the optimization of the time processing and memory. The *a contrario* approach may be parallelized, since the N_{test} loop in Algorithm 1 may be run independently, and the comparison of the results to get the solution minimizing NFA may be only done at the end of the algorithm. Besides, the prediction part of the RBPF can also be processed simultaneously with the outlier detection. We will also investigate a stronger coupling between particle filtering and *a contrario* estimation. The proposed *a contrario* detection algorithm will not only provide the partition between inliers and outliers, but it could also provide an estimate of the state vector (used to interpret the measurements) that may be combined with the particle filter estimate in a fusion process. Finally, we aim at using a more sophisticated observation model instead of basic Equation (1), e.g., including the atmospheric effect, to analyze the detected outliers and, when possible, to correct them.

Appendix

Equation (18) is solved using Algorithm A1. It is presented in the case of ξ_2 estimation, but the case of ξ_1 may be derived as a specific case. The input data are the observations, for the previous solution $\xi_{(2,t|t-1)}$ and the regularization parameter vector λ . Like in most practical applications, the regularization parameter is fitted (or learned) using a first set of data. In our case, we do not regularize clock bias and clock drift, δ_t and $\dot{\delta}_t$, so that the corresponding λ coefficients are set to zero. This algorithm involves the computation of the Jacobian that is as follows.

Let us define two non-linear functions:

$$f_i(\mathbf{X}) = \sqrt{(E_r - e_{S_i})^2 + (N_r - n_{S_i})^2 + (U_r - u_{S_i})^2} + c\Delta t \quad (\text{A1})$$

$$g_i(\mathbf{X}) = (\dot{E}_r - \dot{e}_{S_i})a_{e_i} + (\dot{N}_r - \dot{n}_{S_i})a_{n_i} + (\dot{U}_r - \dot{u}_{S_i})a_{u_i} - c\dot{\delta}t \quad (\text{A2})$$

where $\mathbf{X}^T = (e_r, n_r, u_r, \delta t, \dot{e}_r, \dot{n}_r, \dot{u}_r, \dot{\delta}t, \ddot{e}_r, \ddot{n}_r)$ is the unknown state vector. Then:

$$\begin{cases} E_r = e_r + \dot{e}_r dt + \ddot{e}_r \frac{dt^2}{2}, \\ N_r = n_r + \dot{n}_r dt + \ddot{n}_r \frac{dt^2}{2}, \\ U_r = u_r + \dot{u}_r dt, \\ \dot{E}_r = \dot{e}_r + \ddot{e}_r dt, \\ \dot{N}_r = \dot{n}_r + \ddot{n}_r dt, \\ \Delta t = \delta t + \dot{\delta}t dt, \\ a_{e_i} = \frac{e_{S_i} - E_r}{R_i}, \\ a_{n_i} = \frac{n_{S_i} - N_r}{R_i}, \\ a_{u_i} = \frac{u_{S_i} - U_r}{R_i} \end{cases}$$

Where dt is the time update and $R_i = \sqrt{(E_r - e_{S_i})^2 + (N_r - n_{S_i})^2 + (U_r - u_{S_i})^2}$ is the distance receiver/satellite for all $S_i, i \in \{1 \dots n\}$.

The Jacobian is:

$$J = \begin{pmatrix} \frac{\partial f_1(X)}{\partial e_r} & \frac{\partial f_1(X)}{\partial n_r} & \dots & \frac{\partial f_1(X)}{\partial \ddot{n}_r} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n(X)}{\partial e_r} & \frac{\partial f_n(X)}{\partial n_r} & \dots & \frac{\partial f_n(X)}{\partial \ddot{n}_r} \\ \frac{\partial g_1(X)}{\partial e_r} & \frac{\partial g_1(X)}{\partial n_r} & \dots & \frac{\partial g_1(X)}{\partial \ddot{n}_r} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial g_n(X)}{\partial e_r} & \frac{\partial g_n(X)}{\partial n_r} & \dots & \frac{\partial g_n(X)}{\partial \ddot{n}_r} \end{pmatrix} \quad (\text{A3})$$

where:

$$\left\{ \begin{array}{l} \frac{\partial f_i(\mathbf{X})}{\partial e_r} = \frac{E_r - e_{s_i}}{R_i}, \frac{\partial f_i(\mathbf{X})}{\partial n_r} = \frac{N_r - n_{s_i}}{R_i} \\ \frac{\partial f_i(\mathbf{X})}{\partial u_r} = \frac{U_r - u_{s_i}}{R_i}, \frac{\partial f_i(\mathbf{X})}{\partial \delta t} = c \\ \frac{\partial f_i(\mathbf{X})}{\partial \ddot{e}_r} = dt \frac{\partial f_i(\mathbf{X})}{\partial e_r}, \frac{\partial f_i(\mathbf{X})}{\partial \ddot{n}_r} = dt \frac{\partial f_i(\mathbf{X})}{\partial n_r} \\ \frac{\partial f_i(\mathbf{X})}{\partial \ddot{u}_r} = dt \frac{\partial f_i(\mathbf{X})}{\partial u_r}, \frac{\partial f_i(\mathbf{X})}{\partial \ddot{\delta t}} = cdt, \\ \frac{\partial f_i(\mathbf{X})}{\partial \ddot{e}_r} = \frac{dt^2}{2} \frac{\partial f_i(\mathbf{X})}{\partial e_r}, \frac{\partial f_i(\mathbf{X})}{\partial \ddot{n}_r} = \frac{dt^2}{2} \frac{\partial f_i(\mathbf{X})}{\partial n_r} \end{array} \right.$$

and:

$$\left\{ \begin{array}{l} \frac{\partial g_i(\mathbf{X})}{\partial e_r} = (\dot{E}_r - \dot{e}_s) \frac{\partial a_{e_i}}{\partial e_r} + (\dot{N}_r - \dot{n}_s) \frac{\partial a_{n_i}}{\partial e_r} + (\dot{U}_r - \dot{u}_s) \frac{\partial a_{u_i}}{\partial e_r} \\ \frac{\partial g_i(\mathbf{X})}{\partial n_r} = (\dot{E}_r - \dot{e}_s) \frac{\partial a_{e_i}}{\partial n_r} + (\dot{N}_r - \dot{n}_s) \frac{\partial a_{n_i}}{\partial n_r} + (\dot{U}_r - \dot{u}_s) \frac{\partial a_{u_i}}{\partial n_r} \\ \frac{\partial g_i(\mathbf{X})}{\partial u_r} = (\dot{E}_r - \dot{e}_s) \frac{\partial a_{e_i}}{\partial u_r} + (\dot{N}_r - \dot{n}_s) \frac{\partial a_{n_i}}{\partial u_r} + (\dot{U}_r - \dot{u}_s) \frac{\partial a_{u_i}}{\partial u_r} \\ \frac{\partial g_i(\mathbf{X})}{\partial \delta t} = 0 \\ \frac{\partial g_i(\mathbf{X})}{\partial \ddot{e}_r} = a_{e_i}, \frac{\partial g_i(\mathbf{X})}{\partial \ddot{n}_r} = a_{n_i} \\ \frac{\partial g_i(\mathbf{X})}{\partial \ddot{u}_r} = a_{u_i}, \frac{\partial g_i(\mathbf{X})}{\partial \ddot{\delta t}} = -cdt \\ \frac{\partial g_i(\mathbf{X})}{\partial \ddot{e}_r} = a_{e_i} dt, \frac{\partial g_i(\mathbf{X})}{\partial \ddot{n}_r} = a_{n_i} dt \end{array} \right.$$

The derivatives of the unit vector are given by:

$$\left\{ \begin{array}{l} \frac{\partial a_{e_i}}{\partial e_r} = \frac{(E_r - e_s)^2 - R_i^2}{R_i^3} \\ \frac{\partial a_{n_i}}{\partial n_r} = \frac{(N_r - n_s)^2 - R_i^2}{R_i^3} \\ \frac{\partial a_{u_i}}{\partial u_r} = \frac{(U_r - u_s)^2 - R_i^2}{R_i^3} \\ \frac{\partial a_{e_i}}{\partial n_r} = \frac{\partial a_{n_i}}{\partial e_r} = \frac{(E_r - e_s)(N_r - n_s)}{R_i^3} \\ \frac{\partial a_{e_i}}{\partial u_r} = \frac{\partial a_{u_i}}{\partial e_r} = \frac{(E_r - e_s)(U_r - u_s)}{R_i^3} \\ \frac{\partial a_{u_i}}{\partial n_r} = \frac{\partial a_{n_i}}{\partial u_r} = \frac{(U_r - u_s)(N_r - n_s)}{R_i^3} \end{array} \right.$$

The derivative of \mathbf{a} relatively to the velocity and the acceleration are assumed null.

Algorithm A1: Estimation Based on the non-Linear Solver and Including the Regularization Term, Inputs: Matrix \mathcal{S} , Whose Lines are the Observations $(x_{S_i}, y_{S_i}, z_{S_i}, \dot{x}_{S_i}, \dot{y}_{S_i}, \dot{z}_{S_i}, \dot{\rho}_i)$, $\xi_{(2,t|t-1)}$, λ and Outputs $\tilde{\xi}_2$; the Same Algorithm to Estimate $\tilde{\xi}_1$ Omitting Dp Measurements $\tilde{\rho}$.

```

1 Set  $\epsilon$  to the precision threshold;
2 if  $\lambda_2 = 0$  then
3   | Estimate  $\tilde{\xi}_2$  using the Gauss-Newton algorithm;
4 else
5   | Initialize  $\tilde{\xi}_2$  and  $\tilde{\xi}_2$  to  $\xi_{(2,t|t-1)}$ ;
6   | Define  $\phi$ , the vector of  $\frac{d}{2}$  PR measurements and  $\frac{d}{2}$  Dp measurements from  $\mathcal{S}$ ;
7   | while  $|\tilde{\xi}_2 - \xi_2| > \epsilon$  do
8     |  $\tilde{\xi}_2 \leftarrow \tilde{\xi}_2; \tilde{\phi} \leftarrow [\tilde{\rho} \ \tilde{\rho}]^T$ ;
9     | Calculate the Jacobian  $J(\tilde{\xi}_2, \mathcal{S})$ ;
10    |  $\tilde{\xi}_2 \leftarrow (J^T J + \lambda_2 I)^{-1} J^T (\tilde{\phi} - \phi) + \tilde{\xi}_2$ ;
11  | end
12 end

```

Author Contributions: Salim Zair and Sylvie Le Hégarat-Masclé conceived the RBPF algorithm with the outlier detection part and wrote the paper; Salim Zair and Emmanuel Seigneze designed and performed the experiments; All the authors contributed reagents/materials/analysis tools and analyzing the data;

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Kaplan, E.; Hegarty, C. *Understanding GPS: Principles and Applications*; Artech House: Norwood, MA, USA, 2005.
2. Skog, I.; Handel, P. In-car positioning and navigation technologies: A survey. *IEEE Trans. Intell. Transp. Syst.* **2009**, *10*, 4–21.
3. Chiang, K.W.; Duong, T.T.; Liao, J.K. The Performance Analysis of a Real-Time Integrated INS/GPS Vehicle Navigation System with Abnormal GPS Measurement Elimination. *Sensors* **2013**, *13*, 10599–10622.
4. Peyraud, S.; Bataille, D.; Renault, S.; Ortiz, M.; Mougel, F.; Meizel, D.; Peyret, F. About non-Line-of-Sight Satellite Detection and Exclusion in a 3D Map-Aided Localization Algorithm. *Sensors* **2013**, *13*, 829–847.
5. Lu, W.; Seigneze, E.; Rodriguez, F.A.; Reynaud, R. Lane marking based vehicle localization using particle filter and multi-kernel estimation. In Proceedings of the 13th International Conference on Control Automation Robotics Vision (ICARCV), Singapore, 10–12 December 2014; pp. 601–606.
6. Mao, X.; Wada, M.; Hashimoto, H. Nonlinear filtering algorithms for GPS using pseudorange and Doppler shift measurements. In Proceedings of the IEEE 5th International Conference on Intelligent Transportation Systems, Singapore, 3–6 September 2002; pp. 914–919.
7. Le Marchand, O.; Bonnifait, P.; Ibanez-Guzmán, J.; Betaille, D.; Peyret, F. Characterization of GPS multipath for passenger vehicles across urban environments. *ATTI dell'Ist. Ital. Navig.* **2009**, *189*, 77–88.
8. Xie, P.; Petovello, M. Measuring GNSS Multipath Distributions in Urban Canyon Environments. *IEEE Trans. Instrum. Meas.* **2015**, *64*, 366–377.
9. Obst, M.; Wanielik, G. Probabilistic non-line-of-sight detection in reliable urban GNSS vehicle localization based on an empirical sensor model. In Proceedings of the IEEE Intelligent Vehicles Symposium (IV), Gold Coast, Australia, 23–26 June 2013; pp. 363–368.
10. Brown, R.G. A baseline GPS RAIM scheme and a note on the equivalence of three RAIM methods. *Navigation* **1992**, *39*, 301–316.
11. Seigneze, E.; Kieffer, M.; Lambert, A.; Walter, E.; Maurin, T. Real-time bounded-error state estimation for vehicle tracking. *Int. J. Robot. Res.* **2009**, *28*, 34–48.
12. Hewitson, S.; Wang, J. GNSS Receiver Autonomous Integrity Monitoring with a Dynamic Model. *J. Navig.* **2007**, *60*, 247–263.
13. Arulampalam, M.S.; Maskell, S.; Gordon, N.; Clapp, T. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Trans. Signal Process.* **2002**, *50*, 174–188.
14. Marais, J.; Nahimana, D.F.; Viandier, N.; Duflos, E. GNSS accuracy enhancement based on pseudo range error estimation in an urban propagation environment. *Expert Syst. Appl.* **2013**, *40*, 5956–5964.
15. Giremus, A.; Tourneret, J.Y.; Calmettes, V. A particle filtering approach for joint detection/estimation of multipath effects on GPS measurements. *IEEE Trans. Signal Process.* **2007**, *55*, 1275–1285.
16. Desolneux, A.; Moisan, L.; Morel, J.M. Meaningful alignments. *Int. J. Comput. Vis.* **2000**, *40*, 7–23.
17. Desolneux, A.; Moisan, L.; Morel, J.M. *From Gestalt Theory to Image Analysis: A Probabilistic Approach*; Springer Science & Business Media: New York, NY, USA, 2007; Volume 34.
18. Almansa, A.; Desolneux, A.; Vamech, S. Vanishing point detection without any a priori information. *IEEE Trans. Pattern Anal. Mach. Intell.* **2003**, *25*, 502–507.
19. Muse, P.; Sur, F.; Cao, F.; Gousseau, Y.; Morel, J.M. An a-contrario Decision Method for Shape Element Recognition. *Int. J. Comput. Vis.* **2006**, *69*, 295–315.
20. Burrus, N.; Bernard, T.M.; Jolion, J.M. Image segmentation by a-contrario simulation. *Pattern Recognit.* **2009**, *42*, 1520–1532.
21. Robin, A.; Moisan, L.; Le Hégarat-Masclé, S. An a-contrario approach for subpixel change detection in satellite imagery. *IEEE Trans. Pattern Anal. Mach. Intell.* **2010**, *32*, 1977–1993.
22. Ammar, M.; Le Hégarat-Masclé, S.; Vasiliu, M.; Reynaud, R. An a-contrario approach for object detection in video sequence. *Int. J. Pure Appl. Math.* **2013**, *89*, 173–201.

23. Zair, S.; Le Hégarat-Masclé, S.; Seigniez, E. A-contrario modeling for robust localization using raw GNSS data. *IEEE Trans. Intell. Transp. Syst.* **2016**, doi:10.1109/TITS.2015.2502279.
24. Zair, S.; Le Hégarat-Masclé, S.; Seigniez, E. Coupling Outlier Detection with Particle Filter for GPS-Based Localization. In Proceedings of the 2015 IEEE 18th International Conference on Intelligent Transportation Systems (ITSC), Las Palmas de Gran Canaria, Spain, 15–18 September 2015; pp. 2518–2524.
25. Schön, T.; Gustafsson, F.; Nordlund, P.J. Marginalized particle filters for mixed linear/nonlinear state-space models. *IEEE Trans. Signal Process.* **2005**, *53*, 2279–2289.
26. Rabaoui, A.; Viandier, N.; Duflos, E.; Marais, J.; Vanheeghe, P. Dirichlet process mixtures for density estimation in dynamic nonlinear modeling: Application to GPS positioning in urban canyons. *IEEE Trans. Signal Process.* **2012**, *60*, 1638–1655.
27. Gustafsson, F.; Gunnarsson, F.; Bergman, N.; Forssell, U.; Jansson, J.; Karlsson, R.; Nordlund, P.J. Particle filters for positioning, navigation, and tracking. *IEEE Trans. Signal Process.* **2002**, *50*, 425–437.
28. Zhang, J.; Zhang, K.; Grenfell, R.; Deakin, R. GPS satellite velocity and acceleration determination using the broadcast ephemeris. *J. Navig.* **2006**, *59*, 293–305.
29. Li, L.; Zhong, J.; Zhao, M. Doppler-Aided GNSS Position Estimation with Weighted Least Squares. *IEEE Trans. Veh. Technol.* **2011**, *60*, 3615–3624.
30. Zhou, Z.; Li, B. GNSS windowing navigation with adaptively constructed dynamic model. *GPS Solut.* **2015**, *19*, 37–48.
31. Knight, N.L.; Wang, J. A comparison of outlier detection procedures and robust estimation methods in GPS positioning. *J. Navig.* **2009**, *62*, 699–709.
32. Le Marchand, O.; Bonnifait, P.; Bañez-Guzmán, J.; Peyret, F.; Betaille, D. Performance Evaluation of Fault Detection Algorithms as Applied to Automotive Localisation. In Proceedings of the European Navigation Conference-GNSS 2008, Toulouse, France, 22–25 April 2008.
33. Cheng, C.; Tourneret, J.Y.; Pan, Q.; Calmettes, V. Detecting, estimating and correcting multipath biases affecting GNSS signals using a marginalized likelihood ratio-based method. *Signal Process.* **2016**, *118*, 221–234.
34. Hewitson, S.; Wang, J. GNSS receiver autonomous integrity monitoring (RAIM) performance analysis. *GPS Solut.* **2006**, *10*, 155–170.
35. Rabin, J.; Delon, J.; Gousseau, Y.; Moisan, L. MAC-RANSAC: A robust algorithm for the recognition of multiple objects. In Proceedings of the Fifth International Symposium on 3D Data Processing, Visualization and Transmission (3DPTV 2010), Paris, France, 25 March 2010.
36. Doucet, A.; de Freitas, N.; Gordon, N. *An Introduction to Sequential Monte Carlo Methods*; Springer: New York, NY, USA, 2001.
37. Casella, G.; Robert, C.P. Rao-Blackwellisation of sampling schemes. *Biometrika* **1996**, *83*, 81–94.
38. Doucet, A.; Godsill, S.; Andrieu, C. On sequential Monte Carlo sampling methods for Bayesian filtering. *Stat. Comput.* **2000**, *10*, 197–208.
39. Agamennoni, G.; Nieto, J.I.; Nebot, E.M. An outlier-robust Kalman filter. In Proceedings of the 2011 IEEE International Conference on Robotics and Automation (ICRA), Shanghai, China, 9–13 May 2011.
40. Ramaswamy, S.; Rastogi, R.; Shim, K. *Efficient Algorithms for Mining Outliers from Large Data Sets*; ACM SIGMOD Record; ACM: New York, NY, USA, 2000; Volume 29, pp. 427–438.
41. Huber, P.J. Robust estimation of a location parameter. *Ann. Math. Stat.* **1964**, *35*, 73–101.

