

Research Article

The Particle Filter Sample Impoverishment Problem in the Orbit Determination Application

Paula Cristiane Pinto Mesquita Pardal,¹ Helio Koiti Kuga,² and Rodolpho Vilhena de Moraes³

 ¹University of São Paulo (USP)-EEL/LOB, Estrada Municipal do Campinho, s/n, 12602-810 Lorena, SP, Brazil
 ²National Institute for Space Research (INPE)-DMC, Avenida dos Astronautas, 1758 Jardim da Granja, 12227-010 São José dos Campos, SP, Brazil
 ³São Paulo Federal University (UNIFESP)-ICT/UNIFESP, Rua Talim, 330 Vila Nair, 12231-280 São José dos Campos, SP, Brazil

Correspondence should be addressed to Paula Cristiane Pinto Mesquita Pardal; paulapardal@usp.br

Received 11 February 2015; Accepted 6 May 2015

Academic Editor: Ruihua Liu

Copyright © 2015 Paula Cristiane Pinto Mesquita Pardal et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The paper aims at discussing techniques for administering one implementation issue that often arises in the application of particle filters: sample impoverishment. Dealing with such problem can significantly improve the performance of particle filters and can make the difference between success and failure. Sample impoverishment occurs because of the reduction in the number of truly distinct sample values. A simple solution can be to increase the number of particles, which can quickly lead to unreasonable computational demands, which only delays the inevitable sample impoverishment. There are more intelligent ways of dealing with this problem, such as roughening and prior editing, procedures to be discussed herein. The nonlinear particle filter is based on the bootstrap filter for implementing recursive Bayesian filters. The application consists of determining the orbit of an artificial satellite using real data from the GPS receivers. The standard differential equations describing the orbital motion and the GPS measurements equations are adapted for the nonlinear particle filter, so that the bootstrap algorithm is also used for estimating the orbital state. The evaluation will be done through convergence speed and computational implementation complexity, comparing the bootstrap algorithm results obtained for each technique that deals with sample impoverishment.

1. Introduction

The orbit of an artificial satellite is determined using real data from the Global Positioning System (GPS) receivers. In the orbit determination process of artificial satellites, the nature of the dynamic system and the measurements equations are nonlinear. As a result, it is necessary to manage a fully nonlinear problem in which the disturbing forces as well as the measurements are not easily modelled. In this orbit determination problem, the variables that completely specify a satellite trajectory in the space are estimated, with the processing of a set of pseudorange measurements related to the body.

A spaceborne GPS receiver is a powerful resource to determine orbits of artificial Earth satellites by providing many redundant measurements, which ultimately yields high degree of the observability to the problem. The Jason satellite is a nice example of using GPS for space positioning. Through an on-board GPS receiver, the pseudoranges (error corrupted distance from satellite to each of the tracked GPS satellites) can be measured and used to estimate the full orbital state.

The bootstrap filter is a particle filter whose central idea is to express the required probability density function (PDF) as a set of random samples, instead of a function over state space [1–3].

Numerous strategies have been developed for solving the particles degeneracy (or sample impoverishment) problem that often arises in particle filter applications like introduction of a risk-sensitive particle filter as an alternative approach to mitigate sample impoverishment based on constructing explicit risk functions from a general class of factorizable Herein, the main goal is to analyze the bootstrap filter behavior for the highly nonlinear orbit determination problem. Its simulation results are compared taking into account the sample impoverishment. A reference solution is a bootstrap particle filter (BPF) applied to orbit determination that has already been compared to the unscented Kalman filter solution for the same problem and works well for the analysis of the sample impoverishment issue [10].

2. Particle Filter

The particle filter was designed to numerically implement the Bayesian estimator [2]. The Bayesian approach consists of constructing the PDF of the state based on all the available information, and, for nonlinear or non-Gaussian problem, the required PDF has no closed form. The bootstrap filter represents the required PDF as a set of random samples, which works as an alternative to the function over state space. This filter is a recursive algorithm for propagation and update of these samples for the discrete time problem. The Bayes rule, the key update stage of the method, is implemented as a weighted bootstrap [1].

The main idea of the BPF is intuitive and direct. At the beginning, N particles $x_{0,i}^+$ (i = 1, ..., N) are randomly generated, based on the known initial PDF $p(x_0)$. At each step of time k, the particles are propagated to the next step using the dynamics equation [2]. After receiving the measurement at time k, the PDF $p(y_k | x_{k-1}^i)$ is evaluated. That is, the conditional relative likelihood of each particle $x_{k,i}^-$ is calculated. If an *m*-dimensional measurement equation is given as $y_k = h(x_k) + v_k$ and v_k is a Gaussian random variable with a mean of zero and a variance of R, $v_k \sim N(0, R)$, then a relative likelihood q_i that the measurement is equal to a specific measurement y^* , given the premise that x_k is equal to the particle x_{k-1}^i , can be computed as follows [2]:

$$q_{i} = p\left(y_{k} = y^{*} \mid x_{k} = x_{k-1}^{i}\right) = p\left[v_{k} = y^{*} - h\left(x_{k-1}^{i}\right)\right]$$

$$\sim \frac{1}{(2\pi)^{m/2} |R|^{1/2}}$$

$$\cdot \exp\left(\frac{-\left[y^{*} - h\left(x_{k-1}^{i}\right)\right]^{\mathrm{T}} R^{-1} \left[y^{*} - h\left(x_{k-1}^{i}\right)\right]}{2}\right).$$
(1)

In (1), the symbol ~ means that the probability is directly proportional to the right side. So if the equation is used for all particles $x_{k,i}^-$, then the relative likelihood that the state is equal to each particle is correct. The relative likelihood values are normalized to ensure that the sum of all likelihood values is equal to one. Next, a new set of randomly generated particles $x_{k,i}^+$ is computed from the relative likelihood q_i .

In the resampling step, roughening was used, in order to prevent sample impoverishment. At this point, there is a set of particles $x_{k,i}^+$ that are distributed according to the PDF $p(x_k \mid y_k)$, and any desired statistical measure of it can be computed [2].

The particle filter, adjusted to the orbit determination problem, can be summarized as follows.

(1) The dynamic and the measurement equations are given as

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{f}_k \left(\mathbf{x}_k \right) + \mathbf{w}_k, \\ \mathbf{y}_k &= \mathbf{h}_k \left(\mathbf{x}_k \right) + \mathbf{v}_k, \end{aligned} \tag{2}$$

where \mathbf{w}_k and \mathbf{v}_k are independent white noise processes with known PDFs.

(2) *N* initial particles $\mathbf{x}_{0,i}^+$ (i = 1, ..., N) are randomly generated on the basis of the known initial state PDF $p(\mathbf{x}_0)$. *N* is a parameter chosen as a trade-off between computational cost and estimation accuracy [2].

(3) For
$$k = 1, 2, \ldots$$
,

(a) in the time propagation step, obtain a priori (predicted) particles $\mathbf{x}_{k,i}^{-}$, using the dynamics equation and the PDF of the process noise, both known:

$$\mathbf{x}_{k,i}^{-} = \mathbf{f}_{k-1}\left(\mathbf{x}_{k-1,i}^{+}\right) + \mathbf{w}_{k-1}^{i}, \quad i = 1, \dots, N,$$
 (3)

where each noise vector, $\mathbf{w}_{k-1}^{\prime}$, is randomly generated on the basis of the known PDF of \mathbf{w}_{k-1} ;

(b) compute the relative likelihood q_i of each particle $x_{k,i}^-$, conditioned on the measurement y_k , using the nonlinear measurement equation and the PDF of the measurement noise, as in (1); (c) normalize the relative likelihood values:

$$q_i = \frac{q_i}{\sum_{j=1}^N q_j};\tag{4}$$

(d) in the resampling step, generate a set of a posteriori (resampled) particles $\mathbf{x}_{k,i}^+$, on the basis of the relative likelihood q_i ;

(e) now, there is a set of particles $\mathbf{x}_{k,i}^+$ distributed according to the PDF $p(\mathbf{x}_k | \mathbf{y}_k)$, and mean and covariance statistical measures can be computed.

In the implementation of the bootstrap filter, there is only a small overlap between the prior and the likelihood.

There are some procedures that may be implemented for combating the consequent reduction in the number of truly distinct sample values, such as increasing the number of particles, roughening, and prior editing [1]. Here, they were implemented: a bootstrap particle filter with resampling (PF); a PF with roughening (PFR); and a PFR with prior editing (PFPE), in order to evaluate roughening and prior editing strategies for dealing with sample impoverishment. 2.1. Roughening. Roughening will be the first remedy for sample impoverishment to be discussed. It restrains the resampled particles spread (a posteriori particles) by adding random noise to them, which is similar to adding artificial process noise to the Kalman filter [2]. In roughening approach, the a posteriori particles are modified, after the resampling step, as follows:

$$\mathbf{x}_{k,i}^{+}(m) = \mathbf{x}_{k,i}^{+}(m) + \Delta \mathbf{x}(m), \quad m = 1,...,n,$$

$$\Delta \mathbf{x}(m) \sim \left(0, K\mathbf{M}(m) N^{-1/n}\right).$$
(5)

 $\Delta \mathbf{x}(m)$ is a zero-mean random variable (usually Gaussian); *K* is a constant tuning parameter; *N* is the number of particles; *n* is the state space dimension; and **M** is a vector of the maximum difference between the particle elements before roughening. The *m*th element of the **M** vector is given as

$$\mathbf{M}(m) = \max_{i,j} \left| \mathbf{x}_{k,i}^{+}(m) - \mathbf{x}_{k,j}^{+}(m) \right|, \quad m = 1, \dots, n, \quad (6)$$

where *k* is the step time and *i* and *j* are particle numbers.

The tuning parameter K choice is a compromise. Being too large, a value would blur the distribution, but being too small, it would produce tight clusters of points around the original particles [1]. In this paper, K = 0.1.

2.2. Prior Editing. Prior editing can be tried if roughening does not prevent sample impoverishment. Such approach edits the a posteriori particles from the prior time instant, $\mathbf{x}_{k-1,i}^+$ (after roughening), if the a priori particle from actual instant, $\mathbf{x}_{k,i}^-$, does not satisfy a coarse, pragmatic acceptance test [1]. Therefore, this procedure artificially boosts the number of samples of the prior editing in the neighborhood of the likelihood, for if an a priori particle is in a region of state space with small q_i , it is rejected. Then, the a priori rejected particle can be roughened as many times as required, according to (5), until it is in a region of significant q_i [2]. The prior editing was implemented as follows [1]:

- (a) Pass the resampled sample from previous instant, x⁺_{k-1}, through roughening and system model to generate the predicted sample from current instant, x⁻_k.
- (b) Calculate $v_{k,i} = z_k \mathbf{h}(\mathbf{x}_{k,i})$, the residual between the true and the predicted measurements, for the *i*th particle of the sample, considering that the actual instant observation is available.
- (c) If the magnitude of $v_{k,i}$ is higher than six standard deviations of the measurement noise, then it is highly unlikely that $\mathbf{x}_{k,i}^-$ is chosen as an a posteriori particle. In this case, $\mathbf{x}_{k,i}^-$ is rejected, and $\mathbf{x}_{k-1,i}^+$ is roughened again and passes one more time through dynamic model to generate a new a priori particle $\mathbf{x}_{k,i}^-$. As $\mathbf{x}_{k-1,i}^+$ has already passed through roughening and generated a rejected predicted particle, this procedure may be repeated while $\mathbf{x}_{k,i}^-$ is in a region of no negligible probability.

Due to the high computational cost involving prior editing, such approach was done only once. It is important to make it clear that, here, the *i*th particle is, in fact, an *n*-dimensional vector, while a sample is a $n \times N$ matrix, where the *n*th state variable is represented by N particles.

The accommodation of roughening and prior editing in the bootstrap particle filter algorithm can be schematized as Figure 1 shows.

3. Orbit Determination

The orbit determination is a process for obtaining values of the parameters that completely specify the motion of an orbiting body (as an artificial satellite), based on a set of observations of the body. It involves nonlinear dynamical and nonlinear measurement systems, which depends on the tracking system and the estimation technique [11, 12]. The dynamical system model consists of describing satellite orbital motion, which includes Earth's rotation effects and perturbation models and measurements models. These models depend on the state variables initial conditions, as well as a variety of parameters which affect both the dynamic motions as the measurement process [13]. Due to the complexity of the applied models, usually it is not possible to solve such models equations directly for any of these parameters from a given set of observations.

The observation may be obtained from the ground station networks using laser, radar, Doppler, or space navigation systems, as the GPS. The choice of the tracking system depends on a compromise between the goals of the mission and the available tools. In the case of the GPS, the advantages are global coverage, high precision, low cost, and autonomous navigation resources. The GPS may provide orbit determination with accuracy at least as good as the methods using ground tracking networks. The latter provides standard precision around tens of meters and the former can provide precision as tight as some centimetres. The GPS provides, at a given instant, a set of many redundant measurements, which makes the orbit position observable geometrically.

After some advances of technology, the single frequency GPS receivers provide a good basis to achieve fair precision at relatively low cost, still attaining the accuracy requirements of the mission operation. The GPS allows the receiver to determine its position and time, geometrically, anywhere at any instant, with data from at least four satellites. The principle of navigation by satellites is based on sending signals and data from the GPS satellites to a receiver located on board the satellite whose orbit needs to be determined. This receiver measures the travel time of the signal and then calculates the distance between the receiver and the GPS satellite. Those measurements of distances are called pseudoranges.

The instantaneous orbit determination using GPS satellites is based on the geometric method. In such method, the observer knows the set of GPS satellites position in a reference frame, obtaining its own position in the same reference frame.

3.1. Dynamics Model. In the case of orbit determination via GPS, the ordinary differential equations which represent the



FIGURE 1: Roughening and prior editing accommodation in the BPF algorithm.

dynamic model are, in its simplest form, given traditionally as follows:

$$\dot{\mathbf{r}} = \mathbf{v},$$

$$\dot{\mathbf{v}} = -\mu \frac{\mathbf{r}}{r^3} + \mathbf{a} + \mathbf{w}_{\nu},$$

$$\dot{b} = d,$$

$$\dot{d} = w_d,$$
(7)

wherein the variables are placed in the inertial reference frame. In (7), **r** is the vector of the position components (x, y, z); **v** is the velocity vector; **a** represents the modelled perturbing accelerations; **w**_v is the white noise vector with covariance **Q**; *b* is the user satellite GPS clock bias; *d* is the user satellite GPS clock drift; and w_d is the noise associated with the GPS clock. The GPS receiver clock offset was not taken into account, so as not to obscure the conclusions drawn in this paper due to introduction of clock offset models in the filters. Indeed, the receiver clock offset was beforehand obtained and used to correct the GPS measurements, so that the measurements are free from the error derived from receiver clock offset.

3.2. Forces Model. There are gravitational and nongravitational forces that affect the orbit of an Earth's artificial satellite. The main disturbing forces of gravitational nature are the nonuniform distribution of Earth's mass; ocean and terrestrial tides; and the gravitational attraction of the Sun and the Moon. And the principal nongravitational effects are Earth atmospheric drag; direct and reflected solar radiation pressure; electric drag; emissivity effects; relativistic effects; and meteorites impacts.

The disturbing effects are, in general, included according to the physical situation presented and to the accuracy that is intended for the orbit determination. Here, only a minimum set of perturbations was included which enable analyzing the performance of the particle filter [14]: geopotential [15]; direct solar radiation pressure [16, 17]; and third body point mass effect of the Sun and the Moon [18, 19]. *3.3. Observations Model.* The nonlinear equation of the observation model is

$$\mathbf{y}_{k} = \mathbf{h}_{k} \left(\mathbf{x}_{k}, t_{k} \right) + \boldsymbol{\nu}_{k}, \tag{8}$$

where, at instant t_k , \mathbf{y}_k is the vector of *m* observations; $\mathbf{h}_k(\mathbf{x}_k)$ is the nonlinear function of state \mathbf{x}_k , with dimension *m*; and \mathbf{v}_k is the observation errors vector, with dimension *m* and covariance \mathbf{R}_k . For the present application, the ion-free pseudorange measurements from the Jason-2 GPS receiver are used. Also, the receiver clock offset was computed before and used to correct the pseudorange measurements. Additionally, the nonlinear pseudorange measurement was modelled according to [20].

4. Results

The tests and the analysis for the bootstrap particle filter and two procedures for avoiding sample impoverishment (roughening and prior editing) are presented. To validate and to analyze the methods, real GPS data from the Jason-2 satellite are used. Ocean Surface Topography Mission (OSTM)/Jason-2 is a follow-on altimetry mission to the very successful TOPEX/Poseidon mission and Jason-1. It is a joint mission between NASA and CNES (French space agency), launched June 20, 2008. Jason-2 has a repeat period of approximately 10 days with 254 passes per cycle. Its nodal period is 6,745.72 sec (near 1.87 hours). Sometimes there may be anomalous or missing data. Occasionally Jason-2 must perform maneuvers to maintain orbit. When the satellite detects something abnormal, it will go into safe hold and will turn off all instruments and no data will be collected [21].

The filters estimated position and velocity are compared with Jason-2 precise orbit ephemeris (POE) from JPL/NASA. The test conditions consider real ion-free pseudorange data, collected by the GPS receiver on-board Jason-2, on October 22, 2010, presenting up to 12 GPS satellites tracked. The tests were limited to 5.5 hours of GPS data spaced 10 s. After that, there was an undesirable data gap which could spoil the test case. Such 5.5-hour arc (near 3 Jason-2 orbital periods) was considered sufficient for evaluating the bootstrap particle filter and roughening and prior editing approaches, in this orbit determination application.

The force model comprises perturbations due to geopotential up to order and degree 50×50 ; direct solar radiation pressure; and Sun-Moon gravitational attraction. This model of forces is suitable for implementation in orbit determination because of the low computational cost added compared to the improvement in the results accuracy [22–24]. The pseudorange measurements were corrected to the first order with respect to ionosphere.

This work is not a search for results accuracy. It aims at analyzing the application of a bootstrap filter to the orbit determination problem. The analysis is based on comparing the errors in position (translated to the orbital radial, normal, and along-track RNT components) among three solutions:

 The bootstrap particle filter with resampling, applied without any sample impoverishment procedure (named "PF" in the results).

- (2) The bootstrap particle filter applied with roughening (named "PFR" in the results).
- (3) The bootstrap particle filter applied with roughening and prior editing (named "PFPE" in the results).

The RNT system interpretation is straightforward: the radial component "R" points to the nadir direction; the normal "N" is perpendicular to orbital plane; and the transversal (along-track) "T" is orthogonal to "R" and "N," and it is also the velocity component. Thus, it is possible to analyze what happens with the orbital RNT components and the orbit evolution as well. There is also interest for processing time, in order to analyze the computational efforts face to the accuracy achieved by each algorithm.

Regarding time of processing, t_{CPU} , it was expected that the time required for the bootstrap particle filter algorithm was increasing in two scenarios: if the number of particles rises and if roughening and prior editing were added to the algorithm. According to the estimator nature, each element of the state was replaced by an array of N particles, where N is a trade-off between computational cost and estimation accuracy. Here, tests were done for 17, 100, 300, and 700 particles.

As said before, prior editing recomputes any particle that does not match a specific criterion. Therefore, a relevant test is to observe the instant when each algorithm starts rejecting particles which will be presented. The goal is to verify whether, as a procedure is included, it delays the rejection process. It will also be analyzed whether the number of particles affects the particle rejection, that is, whether its increase may work as an approach for avoiding sample impoverishment.

If N is set (so the analysis is per line in Table 1), it is noticeable that t_{CPU} , CPU time, measured for PF and PFR is very similar, with a maximum 3% of difference. However, the algorithm that includes prior editing, PFPE, is significantly more costly, reaching 39% of increase. This was expected for PFPE, because of the particles that do not pass the acceptance test and need to return to prior instant. Regarding the effects of increasing the number of particle, for the same algorithm (so the analysis is per column in Table 1), the raise in processing time was 83% from 17 to 100 particles, 67% from 100 to 300, and 59% from 300 to 700. Considering that, for 17, 100, 300, and 700 particles, the increase in *N* is 6, 3, and 2.3 times, respectively; then, t_{CPU} increase is not directly proportional to the number of particles used. Therefore, the CPU time and the increase in the number of particles show that the time of processing is related to the chosen number of particles only when the number of particles changes, but it has no relation with the estimator implemented.

For computing time of processing and for all the simulations shown, a computer Intel Core i5-2430M processor of 2.40 GHz, with 2.70 GB of RAM, was used. The program was coded in FORTRAN 77 within operating system Windows XP and compiler Compaq Visual Fortran version 6.1.

When some particles do not reach prior editing criterion (i.e., the magnitude of $v_{k,i}$ is higher than six standard deviation of the measurement noise), they need to be edited in the prior instant of time. The first instant when any particle

Ν PF PFR PFPE 17 44 s 45 s 1 min 23 s 100 4 min 16 s 4 min 19 s 7 min 07 s $t_{\rm CPU}$ 300 12 min 49 s 13 min 05 s 20 min 56 s 700 31 min 09 s 31 min 27 s 50 min 31 s

TABLE 1: Time of processing.

TABLE 2: Instant of the first rejection of particles occurrence.

	N	PF	PFR	PFPE
t _{PE}	17	16 min 00 s	26 min 40 s	26 min 40 s
	100	10 min 00 s	45 min 50 s	45 min 50 s
	300	9 min 20 s	45 min 50 s	45 min 50 s
	700	9 min 00 s	45 min 50 s	45 min 50 s

needed to be edited, t_{PE} , was computed, for each algorithm and *N*, as can be verified in Table 2.

In Table 2, the rejection of particles was first detected in the PF algorithm, despite the number of particles used. This was expected, since no procedure in order to combat the number of truly distinct samples reduction was used in this algorithm. And the instant when the first rejection occurred in PFR and PFPE concurred for all changing in the number of particles. This is considerably consistent, since prior editing procedure depends first on roughening implementation. Regarding the increase in the number of particles (from 17 to 100), it is noticeable that, when procedures for avoiding sample impoverishment are adopted (in PFR and PFPE cases), the instant when the first rejection of particles is detected is delayed in 42%. Nevertheless, for the other cases analyzed (300 and 700 particles), such instant remains the same. This suggests that increasing N as an approach for minimizing the sample impoverishment issue is a little efficient, with a very high computational burden, as seen in Table 1. For PF, the results were not conclusive. It seems that the higher the number of particles, the faster their impoverishment if nothing is done for avoiding sample impoverishment.

As said before, the number of particles is chosen as a trade-off between computational cost and estimation accuracy. The results in Table 3 present mean and standard deviation of the errors in RNT components evaluated for N = 17; 100; 300; and 700. If only these statistics are analyzed, it is clear that the estimation accuracy improves as the number of particles increases. The largest standard deviation occurred for PFPE (N = 17). In the along-track component, a divergence occurrence in all algorithms (N =17) was detected, which can be verified by the high mean and standard deviation values obtained in the three cases. Such divergence disappears when N is increased. When N =700, the statistics and the errors behavior were very similar in the results obtained by PFR and PFPE versions. By the information presented, it is possible to conclude that if the number of particles is very small, any sample impoverishment avoidance procedure will not be effective. Even more, the PFPE approach as used here (particles edition executed only



FIGURE 2: Errors in RNT coordinates for PFPE simulation (N = 17).



FIGURE 3: Errors in RNT coordinates for PFR simulation (N = 300).

once) does not present any significant improvement facing the PFR procedure, no matter the number of particles used. And taking into account the higher computational cost for only one edition of particles, if the PFPE is implemented as many times as necessary, the computation burden is enough to contraindicate this procedure use, even if the results are improved.

In order to show the behavior of the errors in terms of RNT coordinates, Figures 2, 3, and 4 are presented. Figure 2 shows the worst result, the solution obtained for PFPE algorithm considering N = 17, where the divergence in along-track coordinate is shown. Figure 3 shows a PFR solution for N = 300, while Figure 4 presents the N = 700 solution. It was chosen to introduce PFR solutions because this is the algorithm with better performance during orbit determination process. The errors decrease considerably in all implementations, for 100 or more particles (see Table 3),

Estimator	Ν	Mean ± standard deviation (m)		
Estimator		R	Ν	Т
	17	-5.940 ± 36.276	-1.821 ± 17.869	120.504 ± 212.955
DE	100	-10.289 ± 14.711	-0.238 ± 9.961	94.695 ± 119.586
L L,	300	-1.065 ± 9.367	0.139 ± 7.333	-9.177 ± 50.723
	700	-1.031 ± 8.089	$\textbf{0.038} \pm \textbf{7.047}$	-1.792 ± 34.949
	17	27.666 ± 64.094	6.011 ± 31.725	37.537 ± 343.973
DED	100	-6.876 ± 19.706	-2.935 ± 12.435	2.562 ± 11.078
FFK	300	-6.105 ± 16.806	-2.335 ± 5.478	1.066 ± 6.198
	700	-5.246 ± 12.361	-1.322 ± 4.574	-0.051 ± 3.486
	17	-101.686 ± 63.379	-1.322 ± 75.400	1394.298 ± 902.146
DEDE	100	-9.833 ± 15.272	-1.813 ± 7.972	1.290 ± 9.093
FFFE	300	-5.839 ± 15.008	-1.890 ± 7.615	0.602 ± 3.953
	700	-5.250 ± 12.675	-1.669 ± 4.164	-0.243 ± 4.043

TABLE 3: Mean and standard deviation statistics.



FIGURE 4: Errors in RNT coordinates for PFR simulation (N = 700).

despite an incongruous behavior close to 1 and 5 hours of processing, from which the filters recover later. This anomalous behavior is detected in all simulations results. In the graphics, blue curves correspond to radial component (R); red to normal (N); and green to along-track (T). Increasing N from 300 to 700 was more meaningful to along-track component, as its mean and standard deviation had higher improvement (decreasing in values) and less significant to normal coordinate.

According to Figures 3 and 4, the results for N = 300 are as competitive as for N = 700. And if computational burden of implementing 700 particles is taken into account, 300 particles are sufficient for the evaluation proposed in this paper.

Despite the undesirable data gap near 5.5 hours of GPS data, in order to properly evaluate the fittest algorithm (PFR) regarding number of particles, two other graphics were generated. Figures 5 and 6 present Δr and Δv , the



FIGURE 5: Δr (m) obtained in the PFR simulation.



FIGURE 6: Δv (m/s) obtained in the PFR simulation.

errors in position and velocity, respectively, for 24 hours of implementation. The results obtained for N = 300 (green curves) are compared with N = 17 (dark blue curves). According to the results, it is clear that a higher number of particles are important for improving the results, since the amplitude of errors diminished considerably when N = 300

was used. And despite the troublesome data, for 300 particles case, PFR can recover and continue the estimation process, without divergence, which is an indication of the particle filter robustness. In each figure, Δr and Δv represent, respectively, the absolute value of the errors in position and in velocity, in the ECEF (Earth-Centered Earth-Fixed) reference frame, also known as ECR (Earth-Centered Rotational).

5. Conclusions

A Bayesian bootstrap particle filter was applied for the satellite orbit determination problem, using a set of GPS measurements. The development was evaluated taking into account performance and computational burden. The bootstrap filter results, implemented with resampling, were compared with two other versions, which include roughening and prior editing, aiming at avoiding sample impoverishment.

With regard to time of processing, results showed that PFPE algorithm requires greater time than PF, which is as competitive as PFR. Since the number of particles is chosen as a trade-off between computational cost and estimation accuracy, the CPU time is related to the variation in number of particles chosen.

When the number of particles used is analyzed, it is also possible to conclude that if it is very small, any sample impoverishment avoidance procedure will not be effectual. And since PFPE approach executes particles edition only once here, it does not present any significant improvement facing the PFR procedure, no matter the number of particles used. Additionally, if the computational costs between PFR and PFPE (as implemented) are compared, the computation burden is enough to contraindicate PFPE implementation as many times as necessary, even if the results are improved.

Results confirm that the greater the number of particles, the better the estimation accuracy. The best result was achieved for N = 700, in the three versions of the estimator, although a higher computational effort was demanded in this case. Therefore, when results are compared, it is possible to assure that 300 particles are enough to achieve the accuracy level aimed in this paper.

In order to obtain a better bootstrap particle filter performance, especially in terms of estimation accuracy, adjustments in the many filter variants might be done for improving its efficiency. Such adjustments are directly related to the knowledge about the filter. Other strategies can also be tried for solving implementation issues such as sample impoverishment. Another approach for improving particle filtering is to combine it with another filter such as extended Kalman filter or unscented Kalman filter. In this approach, each particle is updated at the measurement time using the extended or the unscented filter, and then resampling is performed using the measurements.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The authors wish to express their consideration to the National Institute for Space Research (INPE) that kindly provided everything necessary for this paper to be developed. The authors are also grateful to São Paulo Research Foundation (FAPESP), for the support under Contract no. 2012/21023-6, and to National Council for Scientific and Technological Development (CNPq), for the support under Contract no. 303119/2010-1.

References

- N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel approach to nonlinear/non-gaussian Bayesian state estimation," *IEE Proceedings, Part F: Radar and Signal Processing*, vol. 140, no. 2, pp. 107–113, 1993.
- [2] D. Simon, Optimal State Estimation, Wiley-Interscience/John Wiley & Sons, Hoboken, NJ, USA, 2006.
- [3] A. Doucet, N. de Freitas, and N. Gordon, Eds., Sequential Monte Carlo Methods in Practice, Springer, New York, NY, USA, 2001.
- [4] U. Orguner and F. Gustafsson, "Risk-sensitive particle filters for mitigating sample impoverishment," *IEEE Transactions on Signal Processing*, vol. 56, no. 10, pp. 5001–5012, 2008.
- [5] S. Park, J. P. Hwang, E. Kim, and H.-J. Kang, "A new evolutionary particle filter for the prevention of sample impoverishment," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 4, pp. 801–809, 2009.
- [6] N. M. Kwok, G. Fang, and W. Zhou, "Evolutionary particle filter: re-sampling from the genetic algorithm perspective," in *IEEE IRS/RSJ International Conference on Intelligent Robots and Systems (IROS '05)*, pp. 2935–2940, IEEE, August 2005.
- [7] T. Li, T. P. Sattar, and S. Sun, "Deterministic resampling: unbiased sampling to avoid sample impoverishment in particle filters," *Signal Processing*, vol. 92, no. 7, pp. 1637–1645, 2012.
- [8] F. Daum and J. Huang, "Particle degeneracy: root cause and solution," in *Signal Processing, Sensor Fusion, and Target Recognition XX*, vol. 8050 of *Proceedings of SPIE*, Orlando, Fla, USA, April 2011.
- [9] H. Driessen and Y. Boers, "Efficient particle filter for jump Markov nonlinear systems," *IEE Proceedings—Radar, Sonar and Navigation*, vol. 152, no. 5, pp. 323–326, 2005.
- [10] P. C. P. M. Pardal, H. K. Kuga, and R. V. Moraes, "Orbit Determination Using Nonlinear Particle Filter and GPS Measurements," in *Proceedings of the AAS/AIAA Astrodynamics Specialist Conference*, vol. 150 of *Advances in the Astronautical Sciences*, pp. 1077–1092, August 2014.
- [11] P. S. Maybeck, *Stochastic Models, Estimation, and Control*, vol. 2, Academic Press, New York, NY, USA, 1982.
- [12] R. G. Brown and P. Y. C. Hwang, "A Kalman filter approach to precision GPS geodesy," *Navigation*, vol. 30, no. 4, pp. 338–349, 1983.
- [13] O. Montenbruck and E. Gill, Satellite Orbits: Models, Methods, and Applications, Springer, Berlin, Germany, 2001.
- [14] P. C. P. M. Pardal, H. K. Kuga, and R. Vilhena de Moraes, "Non linear sigma point kalman filter applied to orbit determination using GPS measurements," in *Proceedings of the 22nd International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS 2009)*, pp. 2301–2308, Savannah, Ga, USA, September 2009.

- [15] W. M. Kaula, *Theory of Satellite Geodesy*, Blasdell, Walthmam, Mass, USA, 1966.
- [16] P. W. Binning, "GPS, dual frequency, SA free satellite navigation. Navigation technology for the 3rd millennium," in *Proceedings* of the 52nd Annual Meeting of the Institute of Navigation, pp. 803–812, Cambridge, Mass, USA, June 1996.
- [17] J. A. Marshall and S. B. Luthcke, "Modeling radiation forces acting on Topex/Poseidon for precision orbit determination," *Journal of Spacecraft and Rockets*, vol. 31, no. 1, pp. 99–105, 1994.
- [18] A. F. B. A. Prado and H. K. Kuga, Eds., Fundamentos de Tecnologia Espacial, INPE, São José dos Campos, Brazil, 2001.
- [19] T. Guan, Special cases of the three body problem, http://inside .mines.edu/fs_home/tohno/teaching/PH505_2011/Paper_Tianyu -anGuan.pdf.
- [20] A. P. M. Chiaradia, H. K. Kuga, and A. F. B. A. Prado, "Single frequency GPS measurements in real-time artificial satellite orbit determination," *Acta Astronautica*, vol. 53, no. 2, pp. 123– 133, 2003.
- [21] "OSTM/Jason-2 Products Handbook," 2009, ftp://podaac.jpl .nasa.gov/allData/ostm/preview/L2/GPS-OGDR/docs/userhandbook.pdf.
- [22] P. C. P. M. Pardal, H. K. Kuga, and R. V. de Moraes, "Comparing the extended and the sigma point kalman filters for orbit determination modeling using GPS measurements," in *Proceedings* of the 23rd International Meeting of the Satellite Division of the Institute of Navigation (ION GNSS '10), Portland, Ore, USA, September 2010.
- [23] P. C. P. M. Pardal, H. K. Kuga, and R. Vilhena de Moraes, "Unscented Kalman filter robustness assessment for orbit determination using GPS signals," *Advances in the Astronautical Sciences*, vol. 148, pp. 117–134, 2013.
- [24] P. C. P. M. Pardal, H. K. Kuga, and R. V. de Moraes, "Analyzing the unscented kalman filter robustness for orbit determination through global positioning system signals," *Journal of Aerospace Technology and Management*, vol. 5, no. 4, pp. 395–408, 2013.



The Scientific World Journal





Decision Sciences







Journal of Probability and Statistics



Hindawi Submit your manuscripts at





International Journal of Differential Equations





International Journal of Combinatorics





Mathematical Problems in Engineering



Abstract and Applied Analysis



Discrete Dynamics in Nature and Society







Journal of Function Spaces



International Journal of Stochastic Analysis



Journal of Optimization