Hierarchical Particle Filter for Bearings-Only Tracking

T. Bréhard and J.-P. Le Cadre

IRISA/CNRS

Campus de Beaulieu

35042 Rennes Cedex, France

e-mail: thomas.brehard@gmail.com, lecadre@irisa.fr

Tel : 33-2.99.84.71.69   Fax : 33-2.99.84.71.71

http://www.irisa.fr/vista

Abstract

We address here the classical bearings-only tracking problem (BOT) for a single target, issue that belongs to the general class of nonlinear filtering problems. Recently, algorithm-based sequential Monte-Carlo methods (particle filtering) have been proposed. However, Fearnhead has observed that in practice this algorithm diverges. This problem is investigated further in this paper. We show that this phenomenon is due to the unobservability of the distance between the observer and the target. We propose a new algorithm named hierarchical particle filter which takes into account this aspect of the BOT. We demonstrate that this novel filter architecture largely overperforms the classical one. Moreover, these results are confirmed when considering highly maneuvering target scenarios. Finally, we propose a general architecture based on Monte-Carlo methods for filtering initialization, able to accommodate poor prior and complex constraints.

Keywords

Bearings-Only Tracking, Sequential Monte Carlo Methods, Hierarchical Filtering, Particle Filtering, Monte-Carlo Methods.

NOTATION

LP(C): Logarithmic Polar (Coordinates),

MP(C): Modified Polar (Coordinates),

BOT: Bearings-Only Tracking,

MCMC: Monte Carlo Markov Chain,

$\mathbf{x}_k$: is the relative target state in the Cartesian coordinate system,

$\mathbf{y}_k$: is the relative target state in the LPC system,

X^t: denotes the transpose of matrix X,

δ: Dirac δ-function,

η_k: time period,

Id_n: $n \times n$ identity matrix,

$\otimes$: Kronecker product,

F: $F = Id_4 + \eta_k B$ with $B = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \otimes Id_2$,

Q: $Q = \Sigma \otimes Id_2$ with $\Sigma = \begin{pmatrix} \alpha_3 & \alpha_2 \\ \alpha_2 & \alpha_1 \end{pmatrix}$. In practice, $\alpha_3 = \frac{\eta_k^3}{3}$, $\alpha_2 = \frac{\eta_k^2}{2}$ et $\alpha_1 = \eta_k$.

$\mathcal{R}_\beta$: $\mathcal{R}_\beta$ is a rotation matrix defined in the following way: $\mathcal{R}_\beta \triangleq \begin{bmatrix} -\sin\beta & \cos\beta \\ \cos\beta & \sin\beta \end{bmatrix}$.

INTRODUCTION

A common denominator to passive systems is that observations are reduced to bearings. This is true for systems as varied as passive sonar, Electronic Support Measurement (ESM) or Infrared systems (FLIR). Collected bearings are used to infer the target trajectory. This tracking problem has been of continuous interest for the past thirty years. The aim of Bearings-Only Tracking (BOT) is to estimate the target trajectory (a sequence of states) using noise-corrupted bearing measurements from a single observer. Target motion is classically described by a diffusion model[1] so that the filtering problem is composed of two stochastic equations. The first one represents the temporal evolution of the target state (position and velocity), called state equation. The second one links the bearing measurement to the target state at time k (measurement equation).

One of the characteristics of the problem is the nonlinearity of the measurement equation so that the classical Kalman filter is not convenient in this case. To address this problem, two methods have been proposed in the early 70's: the Extended-Kalman Filter (EKF) and the pseudo-measurement approach (PMF)[2]. Both methods are based on the transformation of the measurement equation. However, these methods still suffer from limitating drawbacks, especially when the problem is poorly observable. Thus, the EKF remains relatively unstable due to poor radial distance observability, while the pseudo-linear estimator in the same conditions suffers from bias. A large number of methods have been proposed for improving these algorithms. Among them, one can cite the Modified-Gain Extended Kalman Filter (MG-EKF) [3] , the Modified Polar coordinates Extended Kalman Filter (MP-EKF) [4] and the Range-Parameterized Extended Kalman

[1] see [1] for an exhaustive review on dynamic models

Filter (RP-EKF) [5] which are some EKF extensions aiming at defining more robust tracking algorithms. These three methods share a common objective which is to take into account the poor observability of the radial distance. On the other hand, in the early 90's, a new type of method, fundamentally adapted to nonlinear filtering issues, has been developed. This new approach, named sequential Monte Carlo method, has been initiated by Gordon et al. [6]. This method is a combination of various techniques, namely Monte Carlo integration, importance sampling and resampling techniques. While the first application of particle filtering to bearings-only tracking can be seen in [6], Arulampalam and Ristic [7] have demonstrated the superiority of such a method over more classical Kalman approaches (RP-EKF, MP-EKF). For an introduction to particle methods, we refer to Doucet et al. technical report [8] and to the excellent tutorial of Arulampalam et al. [9]. An interesting compilation of articles about particle filtering methods can be found in [10].

However, the problem remains far from being solved for this specific context. We shall demonstrate the limits of particle methods in section II. Indeed, there are some relatively simple scenarios for which the particle approach remains inconsistent. One can observe that at some point in time, the particle filter diverges. This has already been noticed by Fearnhead in [11]. How could we explain this phenomenon? Historically, the particle approach was applied using the nonlinear specificity of the BOT. But this is not its only specificity. Another fundamental issue is the unobservability [2] of the target-observer distance, which is called radial distance. We shall demonstrate that this classical specificity, which is not taken into account in the particle approach, could be the reason why the algorithm performs so badly. Numerous publications dealing with the problem of unobservability -in the BOT context- have been published in the 80's. In this respect, Aidala and Hammel [4] proposed, as early as in 1983 a seminal approach. The key idea was to use a novel coordinate system (namely Modified Polar Coordinates) to analyze and overcome the problem induced by poor radial observability. The fundamental interest of this system is its ability to separate the observable components from the unobservable components of the target.

The aim of this paper is to take benefit of the ideas developed by Aidala and Hammel by immersing them in a Particle Filtering framework. In this way, a specific coordinate system named Logarithmic Polar Coordinates (LPC) will be of constant use. This new coordinate system has been developed by Bréhard et al. in [12] for deriving closed-form formula for the *posterior* Cramér-Rao bound. We use here the LPC framework for analyzing the weak performances of the particle filter. Furthermore, it provides a natural way for solving the BOT problem via a *hierarchical* particle filter. We shall demonstrate in section III that this new filter is able to solve the divergence problems of the classical particle filter. These results will be confirmed in section V while using a more ”difficult” scenario, this one being characterized by a

[2] Or at best *poor observability* as long as strong observer maneuver does not occur.

maneuvering target.

Particle filter initialization is a classical problem, poorly evoked in the literature. It is generally assumed that the initial law is very informative -which is clearly far from reality- to cover the support of the initial distribution. A classical remedy is to consider grid-based methods, which require to draw a huge number of particles. Thus, the aim of section IV is to show how easy it is to gather our hierarchical particle filter architecture with a simple MCMC method for track initialization. In this new setup, prior is limited to radial distance and target speed lower and upper bounds. Our method can be referred to [13], [14], [15] and [11], which suggest using a sampling method based on the likelihood linked to the observation equation. Thus, the objective is to use a ”*batch*” method so as to initialize the tracking process in a ”clever” way. Here, a definite advantage of MCMC methods is their ability to explore the whole domain of solutions (here the posterior density), while taking benefit from informative constraints. For a complete presentation, we refer to [16]. The Markov chain can be simulated either by a classical Hastings-Metropolis algorithm or by Hit-and-Run sampling [17]. The advantage of the latter method is its ability to take into account complex constraints as target route, map, operational requirements, etc.

The paper is organized as follows. A general presentation of the BOT problem is presented in the first section. We shall demonstrate in section II, via simulations, that the classical bootstrap filter does not perform satisfactorily for poorly observable scenarios, even for perfect initialization. We define in section III the LPC and develop a more robust filter based on a hierarchical architecture, named hierarchical particle filter. We show in the same section the superiority of this approach compared to the more classical bootstrap filter. The problem of filter initialization is dealt within section IV. We propose a method based on the Hasting-Metropolis algorithm. This method in particular makes it possible to initialize the filter even when the knowledge on the initial state of the target is poor. Lastly, section V presents some complementary simulations. The initialization method developed in section IV is integrated within the hierarchical particle filter architecture and used for tracking a maneuvering target. We will show that the performances of the hierarchical particle filter largely exceeds that of a traditional particle filter.

I. The problem

Let us consider the following problem, described in figure 1. An object evolves in a x-y plane. One wishes to estimate the position and the speed of this object at every time period. We have a moving observer providing at each period a relative bearing measurement.

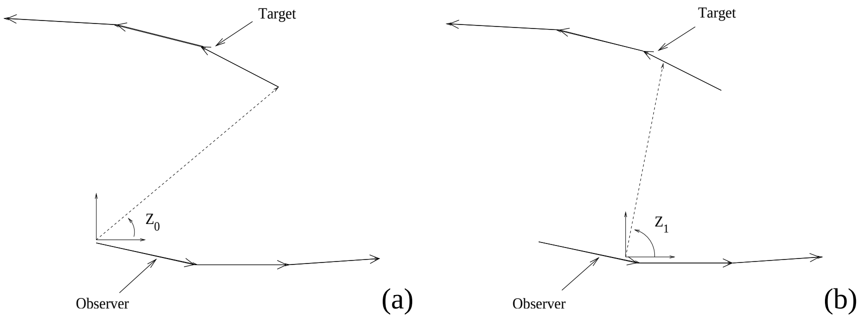

Fig. 1. An example of object trajectory. At the initial time period (a), the observer-target bearing measurement is z_0. Then, at the next time (b), it is z_1, etc..

Let us start by defining the target state in Cartesian coordinates at time k:

$$\mathbf{x}_k^{tgt} \triangleq \left[\begin{array}{cccc} r_x^{tgt}(k) & r_y^{tgt}(k) & v_x^{tgt}(k) & v_y^{tgt}(k) \end{array} \right]^t, \tag{1}$$

composed of the position and speed of the target in the $x-y$ plane. In the same way, we define the observer state at time k:

$$\mathbf{x}_k^{obs} \triangleq \left[\begin{array}{cccc} r_x^{obs}(k) & r_y^{obs}(k) & v_x^{obs}(k) & v_y^{obs}(k) \end{array} \right]^t. \tag{2}$$

Assuming that the observer state is known, the *relative* target state is:

$$\mathbf{x}_k \triangleq \mathbf{x}_k^{tgt} - \mathbf{x}_k^{obs} \triangleq \left[\begin{array}{cccc} r_x(k) & r_y(k) & v_x(k) & v_y(k) \end{array} \right]^t. \tag{3}$$

Throughout this paper, we will be concerned with the estimation (tracking) of this *relative* state vector.

The observation equation.

Denoting z_k the bearing measurement received at time k, the target state is connected to the angular measurement via the following equation:

$$z_k = h(\mathbf{x}_k) + w_k \tag{4}$$

where:

$$h(\mathbf{x}_k) \triangleq \operatorname{atan2}\left(r_y(k), r_x(k)\right) \tag{5}$$

where $atan2$ is the four quadrant inverse tangent. This equation is generally called measurement equation. It is supposed that w_k is a centered Gaussian noise of known variance σ_β^2.

The state equation.

To solve the problem, it is supposed that we have some information about the target trajectories. It is usual to model our uncertainty by the discretized white noise acceleration model:

$$\mathbf{x}_{k+1}^{tgt} = F\mathbf{x}_k^{tgt} + \sigma \mathbf{v}_k \ , \ \mathbf{v}_k \sim \mathcal{N}(0, Q) \ . \tag{6}$$

The process noise structure is represented by the Q matrix, its intensity is the σ scalar. The matrices F and Q are specified in the notational subsection. Depending on the application, a wide variety of target dynamics has been considered in the literature (see [1]). However, the model (6) is quite relevant for our applications. If the σ factor can be considered as mainly conceptual, it is more enlightening to consider (6) (with a convenient σ) as a majorizing model (see Hernandez et al. [18]). In addition, we will study the evolution of the target relatively to the observer. Also, we introduce the evolution equation of the observer:

$$\mathbf{x}_{k+1}^{obs} = F\mathbf{x}_k^{obs} - \mathbf{u}_k \ , \tag{7}$$

where $\mathbf{u}_k$ stands for the part of the observer state due to the control at time period control k. Combining equations (6) and (7), the relative evolution equation of the target is obtained:

$$\mathbf{x}_{k+1} = F\mathbf{x}_k + \mathbf{u}_k + \sigma \mathbf{v}_k \ . \tag{8}$$

The state covariance σ is unknown. However, we assume classically that $\sigma < \sigma_{max}$, so that we use in practice the following equation:

$$\mathbf{x}_{k+1} = F\mathbf{x}_k + \mathbf{u}_k + \sigma_{max} \mathbf{v}_k \ . \tag{9}$$

Equations (4) and (9) form the framework of the BOT filtering problem. One can notice right now that we are confronted to a nonlinear problem of filtering. The BOT objective then appears clearly. We have to estimate the law of the trajectory until time k noted $\mathbf{x}_{0:k} \triangleq (\mathbf{x}_0, \ldots, \mathbf{x}_k)$ knowing the observations $\mathbf{z}_{1:k} \triangleq (z_1, \ldots, z_k)$. The associated density, denoted $p(\mathbf{x}_{0:k}|\mathbf{z}_{1:k})$ is named *posterior* density. Obviously, from a filtering point of view, it can be sufficient to deal with the *posterior* density associated with the current state, i.e. $p(\mathbf{x}_k|\mathbf{z}_{1:k})$.

Lastly, let us finish this presentation of the BOT by mentioning the problem of unobservability of the distance between the observer and the target i.e. the range. As figure 2 shows it, to a given set of measurements $\mathbf{z}_{1:k}$, corresponds a set of trajectories. Three of them are represented on this figure. This type of ambiguity can be offset by an observer maneuver. However, there exist ambiguous maneuvers so that maneuver is a necessary but not a sufficient condition [19].

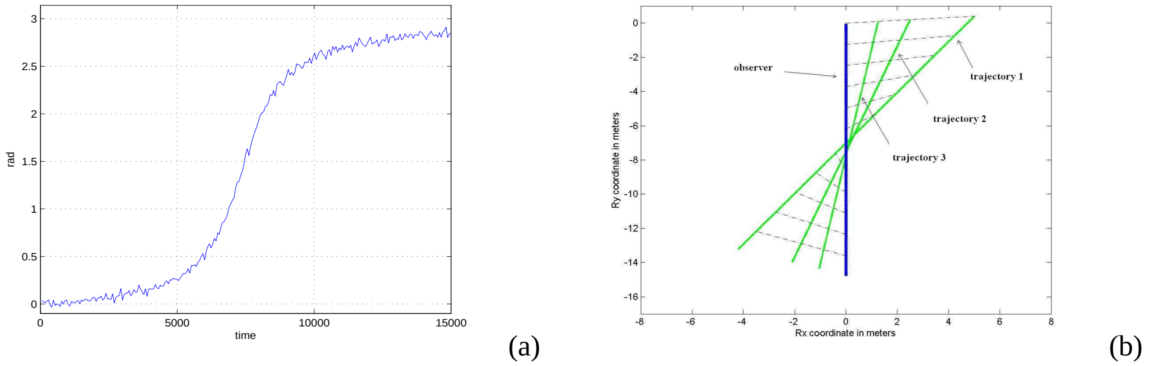

Fig. 2. An observability problem. To a given set of measurements (a) corresponds a set of acceptable trajectories for the target. Three of them are represented on figure (b).

II. Limitations of the particle filtering approach for the BOT

It is at the beginning of the 90's, that Gordon et al. [6] proposed an algorithm for the problem of stochastic filtering in the non-Gaussian nonlinear case, namely the bootstrap filter. We consider in this section various representative BOT target tracking scenarios, tending to show that the traditional particle filter suffers from severe drawbacks, even for relatively ”simple” scenarios. Let us specify the two performance criteria we will use thereafter. Firstly, we will look at the evolution of the Mean Square Error (MSE) in the course of time, defined in the following way:

Definition 1 (MSE) *At time k, the mean square error is defined by:*

$$MSE_k = \mathbb{E}\{(\mathbf{x}_k - \hat{\mathbf{x}}_k)^t(\mathbf{x}_k - \hat{\mathbf{x}}_k)\} \ . \tag{10}$$

However, a second index of performance, integrating the concept of confidence area, is also meaningful. The idea consists in estimating the probability that the state of the target is in the area of confidence associated with the *posterior* law. It will be said that the algorithm diverges if this probability is lower than $1-\alpha$. More precisely, this criterion is defined as follows:

Definition 2 (divergence) *At a given time period k, a divergence criterion is defined by:*

$$\mathcal{D}_k \triangleq \mathbb{P}\left((\mathbf{x}_k - \hat{\mathbf{x}}_k)^t\hat{\Sigma}_k^{-1}(\mathbf{x}_k - \hat{\mathbf{x}}_k) \leq \kappa_\alpha\right) < 1-\alpha \ , \tag{11}$$

where $\mathbf{x}_k$, $\hat{\mathbf{x}}_k$ and $\hat{\Sigma}_k$ are respectively the true target state, its estimate (tracking) and the variance of this estimate deduced form the posterior *distribution. Scalar κ_α denotes the fractile of the Chi square distribution with n_x degrees of freedom at the level α. n_x is the size of target state.*

This definition is based on the construction of a confidence area. Of course, a convenient approach would consist in using the Central Limit Theorem for sequential Monte-Carlo methods as derived in [20]. However, the estimation of the asymptotic covariance in this context is a difficult problem. To overcome this problem, one assumes $p(\mathbf{x}_k|\mathbf{z}_{1:k}) \approx \mathcal{N}\left(\hat{\mathbf{x}}_k, \hat{\Sigma}_k\right)$ so that $(\mathbf{x}_k - \hat{\mathbf{x}}_k)^t \hat{\Sigma}_k^{-1} (\mathbf{x}_k - \hat{\mathbf{x}}_k)$ follows a Chi square distribution with n_x degrees of freedom.

Three scenarios, of passive sonar type, inspired by the simulations suggested in the work of Ristic et al. [21] are studied. These scenarios are differentiated by various target behaviors. Scenario constants are summarized in table I. For the three scenarios, the target and observer trajectories are represented by fig. 3. Moreover, this same figure represents measurements of bearings received by the observer in the course of time. One can notice that the evolution of measurements on figures 3(b.1), 3(b.2) and 3(b.3) is very different according to the scenario considered. This has significant consequences on the variance of estimation of the trajectories. To figure the effects, let us recall the concept of local observability in the BOT context studied first in [22]. It has been shown (see [23]) that the determinant of the Fisher Information Matrix (FIM) locally admits the following approximation:

$$det(FIM_k) \approx \frac{c_k}{\sigma_\beta^8 r_k^8} (4\dot{\beta}_k^4 + 2\dot{\beta}_k \dddot{\beta}_k - 3\ddot{\beta}_k^2) \ , \tag{12}$$

where β_k is the relative angle between the target and the observer at time k. Terms $\dot{\beta}$, $\dot{\beta}$ and $\dot{\beta}$ denote the first, second and third order derivative of β. This formula indicates that the variance of estimation is lower for scenarios with ”large” bearing rates in the course of time. It is meaningful to rank the three scenarios. The first scenario is the most difficult (very weak bearing-rate), while the bearing-rate (and bearing-rate change) is rather large for the third scenario, the second one being of intermediate difficulty. This remark is illustrated by a comparison of the Posterior Cramér-Rao Bound [12] for each the three scenarios in fig.4.

A bootstrap filter is applied with the constants summed up in table II. The initial cloud of particles is sampled in the following way:

$$\mathbf{x}_0^{(i)} \quad = \quad \left[r_0^{(i)} \cos \beta_0^{(i)} \quad r_0^{(i)} \sin \beta_0^{(i)} \quad v_0^{(i)} \cos \beta_0^{(i)} \quad v_0^{(i)} \sin \beta_0^{(i)} \right]$$

where:

$$\left\{ \begin{array}{lcl} \beta_0^{(i)} & \sim & z_0 + \frac{\pi}{\sqrt{12}} \mathcal{N}(0,1) \ , \\ r_0^{(i)} & \sim & r_0 + 50 \, \mathcal{N}(0,1) \quad \text{(in meters)}, \\ v_0^{(i)} & \sim & v_0 + 0.01 \, \mathcal{N}(0,1) \quad \text{(in m/s)}. \end{array} \right. \tag{13}$$

Let us notice that to initialize this set of particles we use the values r_0 and v_0 i.e. the initial distance and relative speed. We choose here to initialize the particle filter by using the true initial state of the target in order to show that the problem of divergence of the bootstrap filter within the framework of the BOT is not due to a bad initialization of the algorithm.

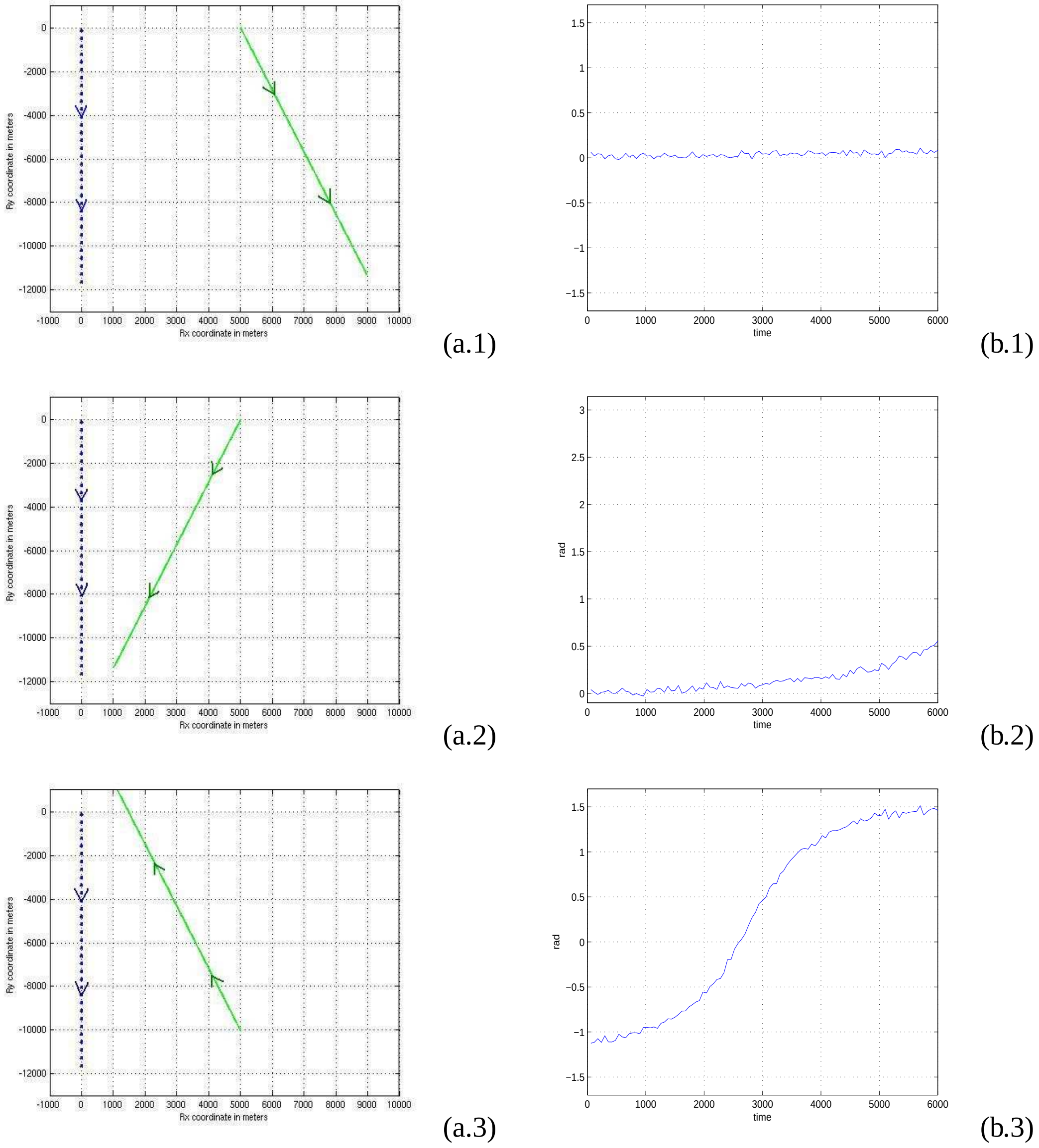

Fig. 3. Three scenarios for the BOT. Representation of the trajectories of the observer (dotted line) and the target (solid line) for scenarios 1 (a.1), 2 (a.2) and 3 (a.3) and of angles measurements obtained in the course of time, (b.1), (b.2) and (b.3).

TABLE I

CONSTANTS FOR THE THREE SCENARIOS

(1 knots $\approx 0.514 m/s$*)*

	Scenario 1	**Scenario 2**	**Scenario 3**
$r_x^{tgt}(0)$	$5\ km$	$5\ km$	$5\ km$
$r_y^{tgt}(0)$	$0\ km$	$0\ km$	$-10\ km$
$v_x^{tgt}(0)$	$1.3\ knots$	$-1.3\ knots$	$-1.3\ knots$
$v_y^{tgt}(0)$	$-3.7\ knots$	$-3.7\ knots$	$3.7\ knots$
$r_x^{obs}(0)$	$0\ km$	$0\ km$	$0\ km$
$r_y^{obs}(0)$	$0\ km$	$0\ km$	$0\ km$
$v_x^{obs}(0)$	$0\ knots$	$0\ knots$	$0\ knots$
$v_y^{obs}(0)$	$-4\ knots$	$-4\ knots$	$-4\ knots$
σ	$0.001 m/s$	$0.001 m/s$	$0.001 m/s$
σ_β	$1.5\ deg$	$1.5\ deg$	$1.5\ deg$
η_k	$60\ s$	$60\ s$	$60\ s$

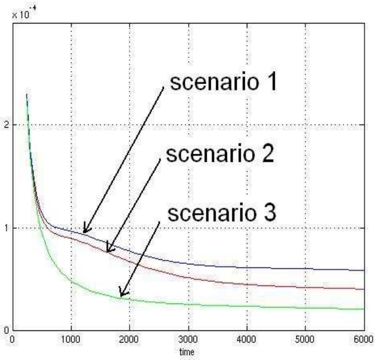

Fig. 4. Comparison of the trace of the *Posterior* Cramér-Rao bound for the three scenarios

TABLE II

BOOTSTRAP FILTER CONSTANTS

constant	**value**
N(number of particles)	10000
$N_{threshold}$	$\frac{N}{3}$
σ_{max}	0.01 m/s

The performance indices MSE_k and $\mathcal{D}_k$ defined by (10) and (11) are calculated by Monte-Carlo integration (the experiments are repeated 100 times). Figures 13(a.1), 13(a.2) and 13(a.3) present the evolution of $\mathcal{D}_k$ in the course of time. One can notice that $\mathcal{D}_k$ decreases quickly for the first scenario. In other words, at the end of a certain time, the probability that the true state of the target is in the area of confidence built by the bootstrap filter is zero. Let us precise this point now. Figure 5 shows examples of estimated trajectories obtained for scenarios 1 and 2. One can see that the final set of particles is not centered around the true position so that the confidence area built by the bootstrap filter does not contain the true trajectory any more. Moreover, the range is overestimated in the three cases. This conclusion is rather disturbing and not intuitive. Many tests were carried out using various constants for the bootstrap filter (the number of particles N and resampling threshold $N_{threshold}$), leading to the same conclusion.

This problem has been rarely evoked in the literature, Fearnhead's PhD thesis [11] being a noticeable exception. Indeed, "classical" scenarios include an observer maneuver at the beginning of the scenario so that the radial distance is observable. One can thus reasonably think that the divergence of the traditional particle filter is directly related to the radial distance unobservability. We propose in the following section to study the filtering problem associated to the BOT by using another coordinate system in order to develop a better understanding of this phenomenon.

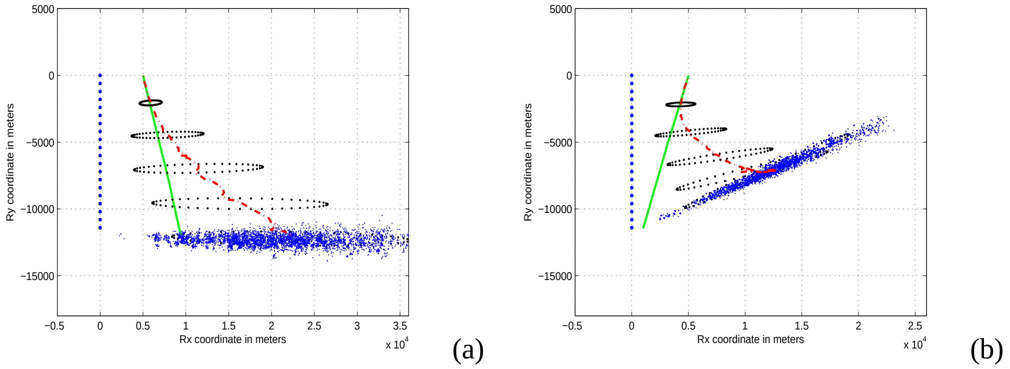

Fig. 5. Divergence of the bootstap filter. Representation of the estimated trajectory of the target (dashed line) as well as confidence areas at various moments. The true trajectory of the target is the solid line, the trajectory of the observer the dotted line. The set of particles at the final time is displayed. (a): scenario 1, (b): scenario 2

III. Particle filter approach for the BOT: solutions

As previously seen the robustness of the bootstrap filter is very disappointing, even for "simple" scenarios and perfect initialization. Not surprisingly, it will be shown now that it is the poor observability of the radial distance which is at the origin of this problem. Our approach consists in rewriting the filtering problem (4,8) within a new coordinate system, namely the logarithmic polar coordinates (LPC), defined in the following way:

Definition 3 (LPC) *The target state, in logarithmic polar coordinates, is defined as follows:*

$$\mathbf{y}_k = \begin{bmatrix} \beta_k & \rho_k & \dot{\beta}_k & \dot{\rho}_k \end{bmatrix}^t \tag{14}$$

where β_k and ρ_k are respectively the bearing and the logarithm of the relative distance between the target and the observer at time k; $\dot{\beta}_k$ et $\dot{\rho}_k$ are respectively the time-derivatives of β_k and ρ_k.

The transition formulas from Cartesian coordinates to logarithmic polar coordinates (LPC) are given below:

$$\mathbf{x}_k = f_{lp}^c(\mathbf{y}_k) = e^{\rho_k} \begin{bmatrix} \cos\beta_k \\ \sin\beta_k \\ -\dot{\beta}_k \sin\beta_k + \dot{\rho}_k \cos\beta_k \\ \dot{\beta}_k \cos\beta_k + \dot{\rho}_k \sin\beta_k \end{bmatrix} \tag{15}$$

and

$$\mathbf{y}_k = f_c^{lp}(\mathbf{x}_k) = \begin{bmatrix} \operatorname{atan2}\left(r_y(k), r_x(k)\right) \\ \ln\left(\sqrt{r_x^2(k) + r_y^2(k)}\right) \\ \frac{r_x(k)v_y(k) - r_y(k)v_x(k)}{r_x^2(k)+r_y^2(k)} \\ \frac{v_x(k)r_x(k)+r_y(k)v_y(k)}{r_x^2(k)+r_y^2(k)} \end{bmatrix} = \begin{bmatrix} \beta_k \\ \rho_k \\ \dot{\beta}_k \\ \dot{\rho}_k \end{bmatrix}. \tag{16}$$

This coordinate system is very close in spirit to the modified polar coordinates pioneered by Aidala and Hammel in [4]. The inverse of the radial distance has been replaced by the logarithm of the radial distance. We can notice that Aidala and Hammel used this coordinate system in [4] precisely to understand the divergence of the Kalman filter. For the particle filter, we will also use a similar approach. Rewriting the filtering problem (4,8) in logarithmic polar coordinates yields:

Proposition 1 (formulation of the problem of BOT) *The problem of bearings-only tracking described by the filtering problem (4,8) can be rewritten in the following form:*

$$\left\{ \begin{array}{lcl} \beta_{k+1} & = & \beta_k + \operatorname{atan2}\left(S_1(k), S_2(k)\right), \\ \rho_{k+1} & = & \rho_k + \frac{1}{2}\ln(S_1^2(k) + S_2^2(k)), \\ \begin{bmatrix} \dot{\beta}_{k+1} \\ \dot{\rho}_{k+1} \end{bmatrix} & = & \frac{1}{(S_1^2(k)+S_2^2(k))} \begin{bmatrix} S_2(k) & -S_1(k) \\ S_1(k) & S_2(k) \end{bmatrix} \begin{bmatrix} S_3(k) \\ S_4(k) \end{bmatrix}, \\ z_k & = & \beta_k + w_k \end{array} \right.$$

where:

$$\begin{bmatrix} S_1(k) \\ S_2(k) \\ S_3(k) \\ S_4(k) \end{bmatrix} = \begin{bmatrix} \eta_k \dot{\beta}_k \\ 1 + \eta_k \dot{\rho}_k \\ \dot{\beta}_k \\ \dot{\rho}_k \end{bmatrix} + \frac{1}{r_k} (Id_2 \otimes \mathcal{R}_{\beta_k}) \mathbf{u}_k + \frac{\sigma}{r_k} \mathbf{v}_k \ . \tag{17}$$

$\mathcal{R}_{\beta_k}$ is the 2D rotation matrix with angle β_k defined in the notation section. Moreover, by definition of the LPC, $r_k = e^{\rho_k}$ (r_k is the range at time k).

PROOF OF PROPOSITION 1: It consists in rewriting the dynamic system (4,8) by using the transitions formulas (15) and (16). The complete proof of this result is given in appendix A.□

Formulation (17) is particularly interesting. One can notice that the term $\frac{\sigma}{r_k}$ is a scale parameter governing the diffusion noise. In other words, as long as the observer does not maneuver, the BOT problem is a nonlinear filtering problem, with unknown and *time-varying* covariance. Moreover, this last term cannot be controlled via the measurement equation, while it is depending on the radial distance. We will now try to understand to what extent this structure is at the origin of the difficulties encountered by the bootstrap filter.

A. Understanding the divergence of the traditional particle filter

We remind that at the initial time, the particles are uniformly dispersed in terms of radial distance between a minimal distance and a maximum distance. However it seems that, after a certain time, the particles corresponding to large radial distances are over-weighted, the particles corresponding to smaller distances then disappear during the state of resampling. However, this stage consists only in keeping the particles that have high weights. To understand the divergence problem, we thus propose to integrate a temporal aspect and to study the behavior of $p(z_{k+1}|\mathbf{y}_k)$. First, let us notice that thanks to Prop.1, we have:

$$z_{k+1} \quad = \quad \beta_k + \operatorname{atan2}\left(S_1(k) S_2(k)\right) + w_k \ , \tag{18}$$

quantities $S_1(k)$ and $S_2(k)$ being defined in Prop. 1. Though (18) is nonlinear, a convenient approximation obtained via a first order expansion is:

$$z_{k+1} \quad \approx \quad \beta_{k+1|k} + \frac{\sigma \eta_k}{r_k \ g(\dot{\beta}_k, \dot{\rho}_k)} v_k + w_k \ , \tag{19}$$

where:

$$\begin{cases} \beta_{k+1|k} & \triangleq \quad \beta_k + \text{atan2}\left(\eta_k \dot{\beta}_k, 1 + \eta_k \dot{\rho}_k\right) \ , \\ g(\dot{\beta}_k, \dot{\rho}_k) & \triangleq \quad \sqrt{(\eta_k \dot{\beta}_k)^2 + (1 + \eta_k \dot{\rho}_k)^2} \ , \\ v_k & \triangleq \quad \frac{1+\eta_k \dot{\rho}_k}{g(\dot{\beta}_k, \dot{\rho}_k)} \mathbf{v}_k^{(3)} - \frac{\eta_k \dot{\beta}_k}{g(\dot{\beta}_k, \dot{\rho}_k)} \mathbf{v}_k^{(4)} \end{cases}$$

Terms $\mathbf{v}_k^{(3)}$ and $\mathbf{v}_k^{(4)}$ denote component number 3 and 4 of vector $\mathbf{v_k}$. Using the statistical properties of $\mathbf{v}_k$ given by eq.(6), one can show that $v_k \sim \mathcal{N}(0, \alpha_3)$. Finally, we obtain via (19) an approximated expression for $p(z_{k+1}|\mathbf{y}_k)$ as follows:

$$p(z_{k+1}|\mathbf{y}_k) \propto \frac{1}{\phi_k} \, e^{-\frac{(z_{k+1} - \beta_{k+1|k})^2}{2\phi_k^2}} \tag{20}$$

where:

$$\phi_k^2 = \frac{\sigma^2 \eta_k^2 \alpha_3}{r_k^2 g^2(\dot{\beta}_k, \dot{\rho}_k)} + \sigma_\beta^2 \ .$$

Let us study the behavior of $p(z_{k+1}|\mathbf{y}_k)$ given by equation (20), as a function of r_k. One can show that this function has a maximum given by the following expression:

$$r_k = \frac{\sigma \eta_k \sqrt{\alpha_3}}{g(\dot{\beta}_k, \dot{\rho}_k) \sqrt{\left(z_{k+1} - \beta_{k+1|k}\right)^2 - \sigma_\beta^2}} \tag{21}$$

if $\left(z_{k+1} - \beta_{k+1|k}\right)^2 > \sigma_\beta^2$. Let us remark using (20) that

$$\mathbb{E}\{\left(z_{k+1} - \beta_{k+1|k}\right)^2 |\mathbf{y}_k\} = \phi_k^2 \tag{22}$$

so that

$$\mathbb{E}\{\left(z_{k+1} - \beta_{k+1|k}\right)^2 |\mathbf{y}_k\} > \sigma_\beta^2 \ . \tag{23}$$

Consequently, as $p(z_{k+1}|\mathbf{y}_k)$ has a maximum given by (21), one can expect to estimate the range. Therefore, the problem is that in classical tracking algorithms, the true covariance state σ is unknown and replaced by σ_{max} so that the maximum associated to $p(z_{k+1}|\mathbf{y}_k)$ is shifted. As $\sigma_{max} > \sigma$, r_k is then overestimated. This fact has been observed in the simulation results presented in the previous section.

Let us remark that σ_{max} is classically used to avoid the estimation of the covariance term. We have shown here that this technique can not be used in the bearings-only context. Now let us remark that in system (17), σ and r_k do not appear separately if the observer does not maneuver (i.e. $\mathbf{u}_k$ is zero). Consequently, the only term that can be estimated is the ratio

$$\boxed{\tilde{\sigma}_k \triangleq \frac{\sigma}{r_k}} \ . \tag{24}$$

Bréhard et al have shown in [24] that this term which is the ”natural” state covariance of system (17) can be estimated. We precise this point now by incorporating $\tilde{\sigma}_k$ in the filtering problem (17). First, the evolution of this time-varying parameter is given by the following equation:

$$\tilde{\sigma}_{k+1} = \tilde{\sigma}_k(S_1^2(k) + S_2^2(k))^{-\frac{1}{2}} \tag{25}$$

where $S_1^2(k)$ and $S_2^2(k)$ are given by eq.(17). Based on $\tilde{\sigma}_k$'s definition given by (24), equation (25) is derived from the evolution equation of ρ_k (see eq.(17)). Now, adding this evolution equation to the system (17), we obtain the new filtering problem:

$$\left\{ \begin{array}{rcl} \beta_{k+1} & = & \beta_k + \text{atan2}\left(S_1(k), S_2(k)\right), \\ \tilde{\sigma}_{k+1} & = & \tilde{\sigma}_k(S_1^2(k) + S_2^2(k))^{-\frac{1}{2}}, \\ \rho_{k+1} & = & \rho_k + \frac{1}{2}\ln(S_1^2(k) + S_2^2(k)), \\ \begin{bmatrix} \dot{\beta}_{k+1} \\ \dot{\rho}_{k+1} \end{bmatrix} & = & \frac{1}{(S_1^2(k)+S_2^2(k))} \begin{bmatrix} S_2(k) & -S_1(k) \\ S_1(k) & S_2(k) \end{bmatrix} \begin{bmatrix} S_3(k) \\ S_4(k) \end{bmatrix}, \\ z_k & = & \beta_k + w_k \end{array} \right. \tag{26}$$

where:

$$\begin{bmatrix} S_1(k) \\ S_2(k) \\ S_3(k) \\ S_4(k) \end{bmatrix} = \begin{bmatrix} \eta_k \dot{\beta}_k \\ 1 + \eta_k \dot{\rho}_k \\ \dot{\beta}_k \\ \dot{\rho}_k \end{bmatrix} + \frac{1}{r_k}(Id_2 \otimes \mathcal{R}_{\beta_k})\mathbf{u}_k + \tilde{\sigma}_k \mathbf{v}_k \; .$$

Let us remark that the filtering problem (26) can then be rewritten as follows:

$$\left\{ \begin{array}{rcl} \mathcal{Y}_{k+1} & = & f_1(\mathcal{Y}_k \; , \; \frac{1}{r_k}\mathbf{u}_k \; , \; \mathbf{v}_k) \; , \\ \rho_{k+1} & = & \rho_k + f_2(\mathcal{Y}_k \; , \; \frac{1}{r_k}\mathbf{u}_k \; , \; \mathbf{v}_k) \; , \\ z_k & = & \beta_k + w_k \end{array} \right.$$

where:

$$\boxed{\mathcal{Y}_k \triangleq \{\beta_k, \dot{\beta}_k, \dot{\rho}_k, \tilde{\sigma}_k\}} \; . \tag{27}$$

The definitions f_1 and f_2 are not presented here but are straightforwardly deduced from (26). The important property here is that as long as the observer does not maneuver (i.e. $\mathbf{u}_j = 0 \, \forall j \in \{1, \ldots, k\}$), $\{\mathcal{Y}_j\}_{j \in \{0,\ldots,k\}}$ is a Markovian process (i.e it doesn't depend on the sequence $\{r_j\}_{j \in \{1,\ldots,k\}}$). This idea is illustrated by figure 13. It is within this framework that we propose to solve this filtering problem by using a hierarchical approach.

B. A hierarchical approach

It has been shown, in the preceding section that the BOT is a nonlinear problem of filtering with an unknown diffusion parameter in the LPC framework given by eq.(26). This new filtering framework has a peculiar property illustrated by figure 13. One can notice that, as long as the observer does not maneuver, this filtering framework displays a particular structure perfectly adapted to a hierarchical estimation. The first stage consists in estimating $\mathcal{Y}_k \triangleq \{\beta_k, \dot{\beta}_k, \dot{\rho}_k, \tilde{\sigma}_k\}$. The second stage updates the *prior* knowledge ρ_k. Let us detail this structure.

Filter 1: estimation of $p(\mathcal{Y}_{0:k}|\mathbf{z}_{1:k})$

As long as the observer does not maneuver, based on (26), the filtering problem associated to $\mathcal{Y}_{0:k}$ can be written as follows:

$$\left\{\begin{array}{lcl} \beta_{k+1} & = & \beta_k + \operatorname{atan2}\left(S_1(k), S_2(k)\right), \\ \tilde{\sigma}_{k+1} & = & \tilde{\sigma}_k(S_1^2(k) + S_2^2(k))^{-\frac{1}{2}}, \\ \begin{bmatrix} \dot{\beta}_{k+1} \\ \dot{\rho}_{k+1} \end{bmatrix} & = & \frac{1}{(S_1^2(k)+S_2^2(k))} \begin{bmatrix} S_2(k) & -S_1(k) \\ S_1(k) & S_2(k) \end{bmatrix} \begin{bmatrix} S_3(k) \\ S_4(k) \end{bmatrix} \quad , \\ z_k & = & \beta_k + w_k \end{array}\right. \tag{28}$$

where:

$$\begin{bmatrix} S_1(k) \\ S_2(k) \\ S_3(k) \\ S_4(k) \end{bmatrix} = \begin{bmatrix} \eta_k \dot{\beta}_k \\ 1 + \eta_k \dot{\rho}_k \\ \dot{\beta}_k \\ \dot{\rho}_k \end{bmatrix} + \tilde{\sigma}_k \mathbf{v}_k \ .$$

In this context, notice that as long as the observer does not maneuver, $\{\mathcal{Y}_j\}_{j\in 0,\ldots,k}$ is a *Markovian* process (i.e it doesn't depend on the sequence $\{r_j\}_{j\in\{1,\ldots,k\}}$). One can thus obtain via a bootstrap filter the *posterior* distribution function $p(\mathcal{Y}_{1:k}|\mathbf{z}_{1:k})$. It is also possible to use a quasi-optimal particle algorithm as defined by Doucet in [8]. $\tilde{\sigma}_0$ can be initialized using the following method based on definition (24):

$$\tilde{\sigma}_0^{(i)} = \frac{\sigma^{(i)}}{r_0^{(i)}} \text{ where } \sigma^{(i)} \quad \sim \quad \sigma_{max}\mathcal{U}([0,1]) \tag{29}$$

and $r_0^{(i)}$ is sampled using the initial prior on radial distance.

Filter 2: estimation of $p(\rho_k|\mathbf{z}_{1:k})$

The objective of the second filter is to estimate $p(\rho_k|\mathbf{z}_{1:k})$ in a recursive way or more precisely to generate a sample $\{\rho_k^{(j)}\}_{j\in\{1,\ldots,M\}}$ according to this law. First, combining the diffusion equation of ρ_k and $\tilde{\sigma}_k$ given by (26), we obtain:

$$\rho_k = \rho_{k-1} + \ln\left(\frac{\tilde{\sigma}_{k-1}}{\tilde{\sigma}_k}\right) \ . \tag{30}$$

Iterating, the last equation can be rewritten

$$\rho_k = \rho_0 + \Delta_{\mathcal{Y}_{0:k}} \tag{31}$$

where:

$$\Delta_{\mathcal{Y}_{0:k}} = \sum_{j=1}^{k} \ln\left(\frac{\tilde{\sigma}_{j-1}}{\tilde{\sigma}_j}\right) \ .$$

Now using eq.(31), $p(\rho_k|\mathbf{z}_{1:k})$ can be rewritten

$$p(\rho_k|\mathbf{z}_{1:k}) = \int \delta_{\rho_0+\Delta_{\mathcal{Y}_{0:k}}}(\rho_k) p(\rho_0, \mathcal{Y}_{0:k}|\mathbf{z}_{1:k}) d\{\mathcal{Y}_{0:k}, \rho_0\} \ . \tag{32}$$

Let us remark that

$$p(\rho_0|\mathcal{Y}_{0:k}, \mathbf{z}_{1:k}) = \frac{p(\mathbf{z}_{1:k}|\mathcal{Y}_{0:k}, \rho_0) p(\rho_0|\mathcal{Y}_{0:k})}{p(\mathbf{z}_{1:k}|\mathcal{Y}_{0:k})} = p(\rho_0|\mathcal{Y}_{0:k}) \ . \tag{33}$$

Incorporating (33) in (32), we obtain

$$p(\rho_k|\mathbf{z}_{1:k}) = \int \delta_{\rho_0+\Delta_{\mathcal{Y}_{0:k}}}(\rho_k) p(\rho_0|\mathcal{Y}_{0:k}) p(\mathcal{Y}_{0:k}|\mathbf{z}_{1:k}) d\{\mathcal{Y}_{0:k}, \rho_0\} \ . \tag{34}$$

Then, the hierarchical structure of the filter described by figure 13 implies that

$$p(\rho_0|\mathcal{Y}_{0:k}) = \frac{p(\mathcal{Y}_{0:k}|\rho_0) p(\rho_0)}{p(\mathcal{Y}_{0:k})} = p(\rho_0) \ . \tag{35}$$

This property is only valid as long as the observer does not perform any maneuver. Incorporating (35) into (34), we obtain:

$$p(\rho_k|\mathbf{z}_{1:k}) = \int \delta_{\rho_0+\Delta_{\mathcal{Y}_{0:k}}}(\rho_k) \underbrace{p(\rho_0)}_{\star} \underbrace{p(\mathcal{Y}_{0:k}|\mathbf{z}_{1:k})}_{\star\star} d\{\mathcal{Y}_{0:k}, \rho_0\} \ . \tag{36}$$

One can notice that all the elementary terms composing the equation (36) are known. Indeed, $\star$ is the prior distribution function of the range at the initial time, $\star\star$ is the result of filter 1 at the moment k. It is thus possible to perform a simulation according to this distribution by Monte-Carlo integration in the following way:

$$\left\{ \begin{array}{lcl} p(\mathcal{Y}_{0:k}|\mathbf{z}_{1:k}) & \approx & \sum_{i=1}^{N_1} w_k^{(i)} \delta_{\mathcal{Y}_{0:k}^{(i)}}(\mathcal{Y}_{0:k}) \ , \\ p(\rho_0) & \approx & \sum_{j=1}^{N_2} \frac{1}{N_2} \delta_{\rho_0^{(j)}}(\rho_0) \ . \end{array} \right. \tag{37}$$

The first approximation is the result at the time period k of filter 1 which is a particle filter. The second one is obtained from $p(\rho_0)$. Combining (36) and (37), we can thus generate a sample of $\{\rho_k^{(k)}\}$, associated with the $p(\rho_k|\mathbf{z}_{1:k})$ density, as follows:

$$\rho_k \sim \sum_{i=1}^{N_1}\sum_{j=1}^{N_2}\frac{w_k^{(i)}}{N_2}\delta_{\rho_0^{(j)}+\Delta_{\mathcal{Y}_{0:k}^{(i)}}}(\rho_k)\ . \tag{38}$$

Final algorithm: the hierarchical filter

We have built a hierarchical particle estimation algorithm. The various stages of the algorithm are summarized by algorithm 1. This algorithm is based on the particular structure of the problem, illustrated by figure 13. The latter in particular makes it possible to demonstrate formula (35). However, as shown on fig. 13, as soon as the ρ_k is observable, the structure of the problem also changes. The estimation of components $\mathcal{Y}_k \triangleq \{\beta_k, \dot{\beta}_k, \dot{\rho}_k, \tilde{\sigma}_k\}$ and ρ_k can then be carried out jointly using a traditional particle filter. to be completely exhaustive, a statistical test should be developed to precise when ρ_k is observable. Then, the algorithm should be able through the scenariot o switch from the hierarchical particle filter to the traditional particle filter when ρ_k is unobservable and vice versa. This point is out of scope of the paper but should investigated further. The global algorithm thus has the final structure described in figure 6. Let us notice finally that from an algorithmic point of view, this algorithm is equivalent in terms of complexity to a traditional particle algorithm.

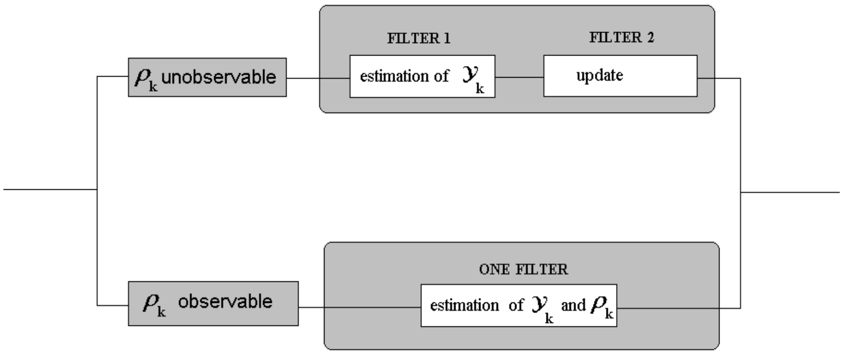

Fig. 6. Hierarchical particle filter. Estimation at time k. When ρ_k is unobservable, estimation is proceeded using a hierarchical particle filter. When ρ_k is observable, a classical particle filter is used.

C. Simulations

We will now compare the performance of this new algorithm with bootstrap filter ones, considering again the three scenarios considered in section 2. We refer to fig.3, constants are summarized in table I. The bootstrap particle filter is applied using the constants presented in table II. The hierarchical particle filter uses the constants given in table III. The set of particles associated to $\{\beta_k, \rho_k, \dot{\beta}_k, \dot{\rho}_k\}$ are initialized in the same way using (13) for the two filters. Concerning the hierarchical particle filter, $\tilde{\sigma}_0$ can be initialized using the following method based on definition (24):

$$\tilde{\sigma}_0^{(i)} = \frac{\sigma^{(i)}}{r_0^{(i)}} \tag{39}$$

where:

$$\begin{cases} r_0^{(i)} & \sim \quad r_0 + 50\,\mathcal{N}(0,1) \qquad \text{(in meters)}, \\ \sigma^{(i)} & \sim \quad \sigma_{max}\mathcal{U}([0,1]) \qquad\qquad . \end{cases}$$

Results are displayed in fig.14. We represent the evolution versus time of the performance indices MSE_k and $\mathcal{D}_k$ defined by (10) and (11). One can notice that the hierarchical filter gives much more satisfactory results in terms of divergence, the true state of the target being *always* in the area of confidence associated with the hierarchical estimate. Moreover, one can notice that, in terms of quadratic error, the performances of this new algorithm largely exceeds that of the bootstrap particle filter.

TABLE III

HIERARCHICAL PARTICLE FILTER CONSTANTS

constant	**value**
N_1(number of particles for filter 1)	10000
N_2(number of particles for filter 2)	10000
$N_{threshold}$	$\frac{N}{3}$
σ_{max}	0.01 m/s

IV. INITIALIZING THE HIERARCHICAL BOT PARTICLE FILTER

This section deals with the initialization of (hierarchical) sequential Monte-Carlo methods in the BOT context. Though this problem is rarely mentioned in the literature it is of a crucial importance, especially for the partially observed case. It is generally admitted that the initial distribution $p(\mathbf{y_0})$ is ”sufficiently” informative. However, in practice, our *prior* knowledge is frequently very poor. A straightforward application of particle filtering would require a huge number of particles, allowing to cover the support of the initial density distribution. Even if this approach can possibly work, it seems unrealistic. To avoid these drawbacks, first let us recall that we can take benefit (in the BOT context) from operational constraints. These constraints may be minimal and maximal range, or speed. We will see that MCMC methods are a natural way for using these constraints in our context [16]. Moreover, it is well admitted to have recourse to a *batch* type method for tracking initialization (see e.g. [13]). This means that we assume that the target trajectory is deterministic throughout the initialization. One can then build in an analytical way the density of the initial state given a "*batch*" of measurements $\mathbf{z}_{1:\kappa}$. The objective of the initialization method is then to generate a set of N particles with the following posterior distribution:

$$p(\mathbf{y}_0|\mathbf{z}_{1:\kappa}) \tag{40}$$

via the likelihood $p(\mathbf{z}_{1:\kappa}|\mathbf{y}_0)$ and priors (constraints). This idea is represented by fig. 7 and will consists in generating a Markovian process which will evolve on the support of the distribution. A simple and feasible solution is given by the Hasting-Metropolis algorithm [25]. The idea of the algorithm itself consists in defining a Markov Chain whose stationary distribution is precisely $p(\mathbf{y}_0|\mathbf{z}_{1:\kappa})$. For a very readable presentation of this (and related) algorithm, we refer to [26]. A basic ingredient is simply the Bayes formula, here yielding:

$$p(\mathbf{y}_0|\mathbf{z}_{1:\kappa}) \propto p(\mathbf{z}_{1:\kappa}|\mathbf{y}_0)p(\mathbf{y}_0) \; . \tag{41}$$

In the right-hand side of the above expression $p(\mathbf{y}_0)$ stands for the prior, here limited to a uniform density on the constraint domain. Moreover, if the diffusion noise is zero during κ time-periods, an analytic expression $p(\mathbf{z}_{1:\kappa}|\mathbf{y}_0)$ can be derived:

$$p(\mathbf{z}_{1:\kappa}|\mathbf{y}_0) \propto e^{-\frac{(\mathbf{z}_{1:\kappa}-f(\mathbf{y}_0))^t(\mathbf{z}_{1:\kappa}-f(\mathbf{y}_0))}{2\sigma_\beta^2}} \; , \tag{42}$$

where σ_β^2 is the variance of the measurement noise,

$$f(y_0) = \begin{pmatrix} \beta_0 + \operatorname{atan2}\left(1 + \eta_k \frac{\dot{r}_0}{r_0} + e^{-\rho_0}(\sin\beta_0 U_1^1 + \cos\beta_0 U_1^2), \eta_k \dot{\beta}_0 + e^{-\rho_0}(\cos\beta_0 U_1^1 - \sin\beta_0 U_1^2)\right) \\ \vdots \\ \beta_0 + \operatorname{atan2}\left(1 + \kappa\eta_k \frac{\dot{r}_0}{r_0} + e^{-\rho_0}(\sin\beta_0 U_\kappa^1 + \cos\beta_0 U_\kappa^2), \kappa\eta_k \dot{\beta}_0 + e^{-\rho_0}(\cos\beta_0 U_\kappa^1 - \sin\beta_0 U_\kappa^2)\right) \end{pmatrix} \tag{43}$$

and U_j^1 et U_j^2 are the two components of the U_j "control" vector defined by

$$\begin{cases} F_k = \overbrace{F \times \ldots \times F}^{k \;\; times} \; , \\ \mathbf{u}_k = \sum_{j=0}^{k-1} F_{k-j}\mathbf{u}_j \; . \end{cases} \tag{44}$$

A proof of (42) can be found in the Appendix B.

From a practical point of view, at each step of the algorithm, physical constraints are checked when a new state is proposed. As long as the state suggested does not fit to the constraints, a new state is proposed. According to the acceptance probability, the candidate state is then accepted or rejected. This probability is computed using the likelihood $p(\mathbf{z}_{1:\kappa}|\mathbf{y}_0)$ given by (42). The algorithm stops when we have a set of N particles. Let us remark that it is important for this first particle to be on the support of the distribution of the related density $p(\mathbf{y}_0|\mathbf{z}_{1:\kappa})$. This is achieved thanks to a standard Gauss-Newton algorithm (see [27] and [28]). Finally, let us emphasize that integration of complex constraints can be achieved via an extension of the Hasting-Metropolis algorithm, named Hit-and-Run sampler [17]. Finally, remark that the choice of an optimal batch size κ is an open problem. This constant should be fixed depending on the nature of the scenario and the number of particles used in the particle filter.

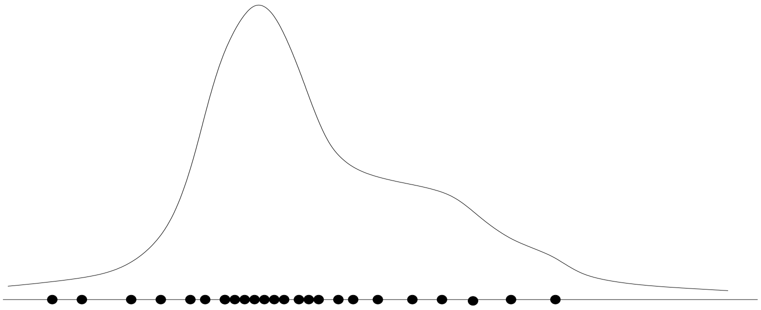

Fig. 7. Initialization of the particle filter.

V. Tracking a maneuvering target

We consider in this section another application of the hierarchical particle filter developed in section III. The scenario considered is the tracking of a maneuvering target. It is a question here of showing the robustness of the approach. Moreover, we also suppose that the knowledge on the initial state of the target is poor so that the method of initialization based on MCMC methods suggested in section IV is applied. The scenario is presented in the following subsection, while simulation results are detailed in subsection B.

A. Scenario

We consider the scenario described by figure 8. A plane moves in straight line at the speed of 350 m/s. It considers every 0.02 sec. a relative angular measurement between its position and the position of a missile in approach. The trajectory of the latter is governed by a guidance law, for which the observer does not have any prior. In practice, this guidance law was generated by proportional navigation (see [29]). The constants of the scenarios are summarized in table IV. Our knowledge on the initial state of the target is relatively weak. We simply have lower and upper bounds for target speed and range. Two different initial priors given in table V are considered in simulations. The second one introduces a bias on the radial distance at the inital time. We use the initialization method developed in section IV. The batch size κ is fixed to 40. Figure 9 illustrates the influence of this parameter on the performance of the hierarchical particle filter. We compare the performances of the hierarchical particle filter (algorithm described by figure 6)) proposed in section III with constants given in table VII to those of the bootstrap filter with constants given in table VI. Both algorithms use the same initialization.

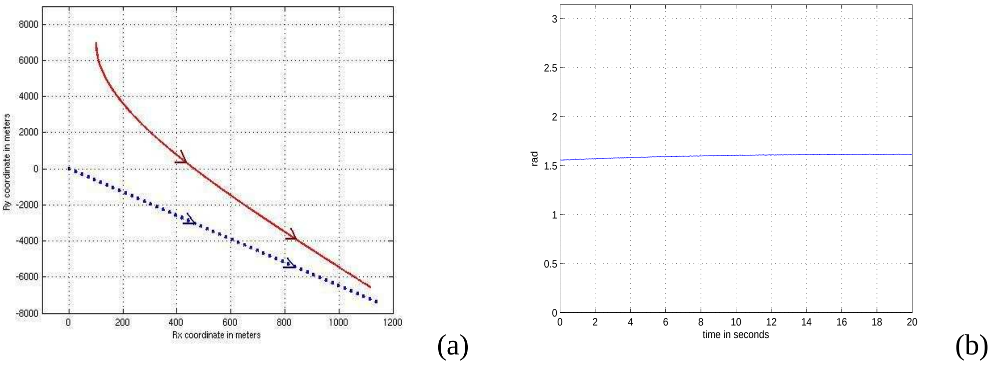

Fig. 8. "*maneuvering target*" scenario. Representation of the trajectories of the observer (dotted line) and the target (solid line) (a) and evolution of angles measurements in the course of time (b).

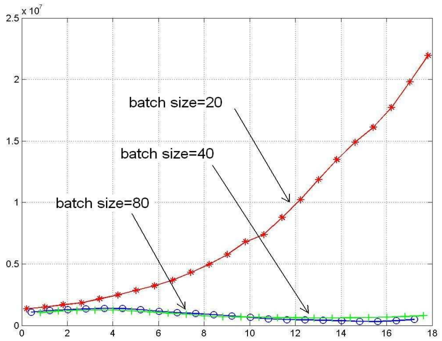

Fig. 9. Comparison of the performances of the LPC filter for "*maneuvering target*" scenario for different batch sizes. Evolution of the mean square error through time for different batch sizes.

TABLE IV

CONSTANTS OF THE “*maneuvering target*” SCENARIO

	Scenario
duration	18 s
$r_x^{tgt}(0)$	$0\ km$
$r_y^{tgt}(0)$	$7\ km$
$v_x^{tgt}(0)$	$0\ m/s$
$v_y^{tgt}(0)$	$-680\ m/s$
$r_x^{obs}(0)$	$0\ km$
$r_y^{obs}(0)$	$0\ km$
$v_x^{obs}(0)$	$58\ m/s$
$v_y^{obs}(0)$	$-376\ m/s$
σ_β	$0.1\ deg$
η_k	$0.02\ s$

TABLE V

INITIAL *Prior* INFORMATION ON THE “*maneuvering target*” SCENARIO

	prior 1	*prior 2*
r_{min}	$4\ km$	$3\ km$
r_{max}	$9\ km$	$8\ km$
v_{min}	$600\ m/s$	$600\ m/s$
v_{max}	$900\ m/s$	$900\ m/s$

TABLE VI

BOOTSTRAP FILTER CONSTANTS

constant	**value**
N(number of particles)	5000
$N_{threshold}$	$\frac{N}{3}$
σ_{max}	$20\ m/s$

TABLE VII

HIERARCHICAL PARTICLE FILTER CONSTANTS

constant	**value**
N_1(number of particles for filter 1)	5000
N_2(number of particles for filter 2)	5000
$N_{threshold}$	$\frac{N}{3}$
σ_{max}	20 m/s

B. *Simulations*

The estimated trajectories as well as confidence ellipsoids at three time periods for the hierarchical particle filter and the bootstrap filter using initial prior 1 (given in table V) are represented on figures 10. We note that after a certain time, the bootstrap algorithm diverges which is not the case of the hierarchical filter. One can notice that both filters perform conveniently during the first part of the scenario. However (see fig.10(a)), one can notice that the bootstrap filter estimate diverges after a certain time. Thus, at the end of the scenario, the radial distance is largely over-estimated. Thus, the associated confidence ellipsoid does not include the true state of the target any more. Conversely, the trajectory obtained by the hierarchical filter (see fig.10(b)) performs quite satisfactorily. Figure 11 represents a comparison of the estimated trajectories as well as confidence ellipsoids computed with the hierarchical partical filter and the bootstrap filter with initial prior 2 (given in table V). At the initial time, both filters suffers from a bias on the radial distance due to the initial prior. One more time, we observe the divergence of the bootstrap filter when the hierarchical particle filter performs satisfactory. Specially, the confidence ellipsoid contains the true target state even if the estimate is biased. Finally, we present in fig.12 the temporal evolution of the mean quadratic error evaluated on the basis of 100 trials. Again, the hierarchical filter widely overperforms the bootstrap filter.

VI. CONCLUSION

We have shown that the performance of the ”classical” bootstrap filter developed by Gordon, Salmond and Smith [6] could be relatively disappointing in a partially observed context. This corroborates Fearnhead’s work [11]. More precisely, in cases where the observer performs a maneuver relatively late, a systematic divergence of the filter is observed. A rewriting of the problem using a new frame of reference, named logarithmic polar coordinates, was used to understand this phenomenon. Classical trackers like the bootstrap filter fix to a maximal value the unknown variance appearing in the state equation. We suggested in this paper

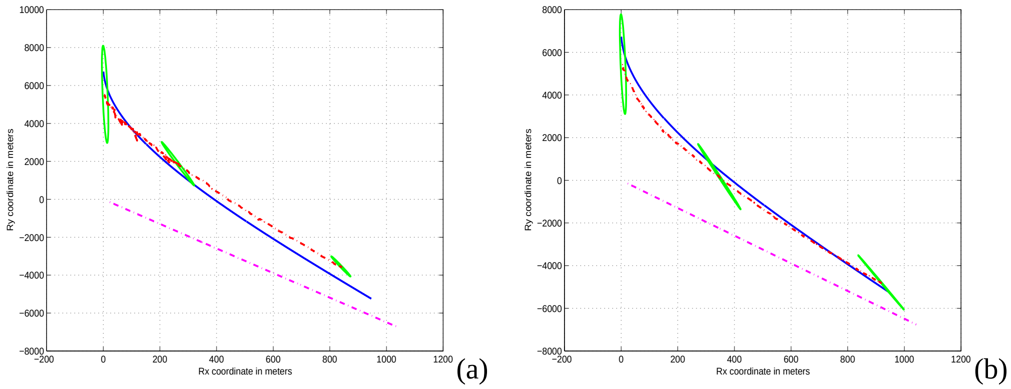

Fig. 10. Estimates and areas of confidences of the filters for the ”maneuvering target ” scenario using prior 1 given in tab.V. Representation of the trajectories of the observer (dotted line) and the target (solid line). Estimated trajectories for the two algorithms: the bootstrap filter (a) and the hierarchical particle filter (b). Areas of confidence at the initial time step and at time 7 s and 18 s.

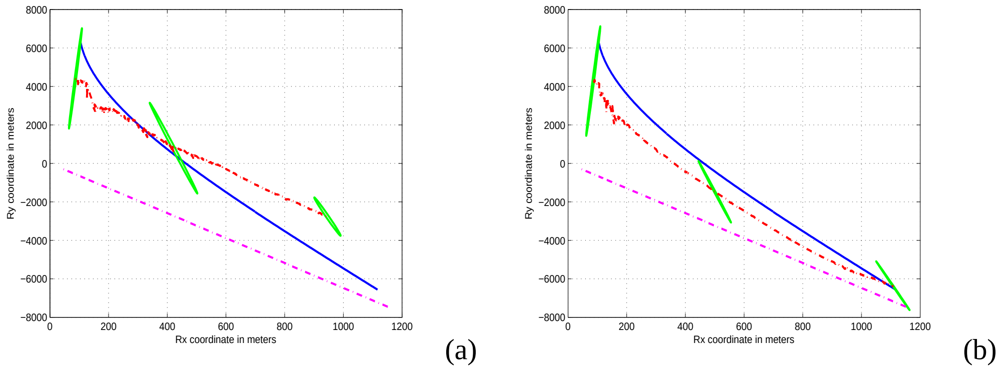

Fig. 11. Estimates and areas of confidences of the filters for the ”maneuvering target ” scenario using prior 2 given in tab.V. Representation of the trajectories of the observer (dotted line) and the target (solid line). Estimated trajectories for the two algorithms: the bootstrap filter (a) and the hierarchical particle filter (b). Areas of confidence at the initial time step and at time 7 s and 18 s.

that this choice induces an overestimation of the radial distance. To avoid this problem, a hierarchical method of estimation based on the LPC framework was proposed. We showed the superiority of this hierarchical particle filter in terms of divergence and mean square error. In addition these results were confirmed for a more difficult scenario characterized by a maneuvering target.

The robustness of this new algorithm would be helpful to solve the multi-target tracking problems [30], [31]. Moreover, it can also be integrated within the framework of maneuvering target [32], [33] and distributed tracking [34].

Let us notice that all this study was carried out using the logarithmic polar coordinates but could have

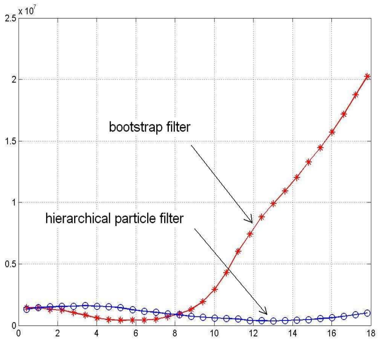

Fig. 12. Comparison of the mean square error through time of the bootstrap filter and the hierarchical particle filter for “*maneuvering target*” scenario

just as easily been carried out by using the modified polar coordinates suggested by Aidala and Hammel [4]. However, this new frame of reference takes all its interest within the framework of performance analysis [12]. Indeed, the LPC make it possible to calculate an exact bound for the error covariance matrix, which is impossible when using the MPC or the Cartesian coordinates. We have considered in this article a target evolving in a two dimensional environment. One can then naturally put the question of the extension of this work to the case of a three dimensional setting which practical interest is obvious. The essential point is to find the extension of the logarithmic polar coordinates in this context. Let us notice that there are some identified systems that could be interesting. One can quote the modified spherical coordinates evoked by Allen and Blackman in [35] and the Cartesian system standardized by the distance proposed by Grossman [36] which is not without pointing out the modified polar coordinates.

Appendix A: proof of proposition 1

The aim of this section is to show that the equation of evolution of the target expressed in LPC has a very particular form. One uses the equation (9). Let us start by expressing the term $F\mathbf{x}_k$ in logarithmic polar coordinates using (15):

$$F\mathbf{x}_k = \begin{bmatrix} r_x(k) + \eta_k v_x(k) \\ r_y(k) + \eta_k v_y(k) \\ v_x(k) \\ v_y(k) \end{bmatrix} = \begin{bmatrix} r_k \mathcal{R}_{\beta_k} \begin{bmatrix} \eta_k \dot{\beta}_k \\ 1 + \eta_k \dot{\rho}_k \end{bmatrix} \\ r_k \mathcal{R}_{\beta_k} \begin{bmatrix} \dot{\beta}_k \\ \eta_k \dot{\rho}_k \end{bmatrix} \end{bmatrix} \tag{45}$$

where:

$$\mathcal{R}_{\beta_k} = \begin{bmatrix} -\sin\beta_k & \cos\beta_k \\ \cos\beta_k & \sin\beta_k \end{bmatrix} .$$

Inserting this result in equation (9), we obtain:

$$\mathbf{x}_{k+1} = \begin{bmatrix} r_k \mathcal{R}_{\beta_k} \begin{bmatrix} \eta_k \dot{\beta}_k \\ 1 + \eta_k \dot{\rho}_k \end{bmatrix} \\ r_k \mathcal{R}_{\beta_k} \begin{bmatrix} \dot{\beta}_k \\ \dot{\rho}_k \end{bmatrix} \end{bmatrix} + \mathbf{u}_k + \sigma \mathbf{v}_k \ . \tag{46}$$

If we factorize $r_k \mathcal{R}^t_{\beta_k}$, the previous expression can be rewritten in the following way:

$$\mathbf{x}_{k+1} = \begin{bmatrix} r_k \mathcal{R}_{\beta_k} \begin{bmatrix} S_1(k) \\ S_2(k) \end{bmatrix} \\ r_k \mathcal{R}_{\beta_k} \begin{bmatrix} S_3(k) \\ S_4(k) \end{bmatrix} \end{bmatrix} \tag{47}$$

where:

$$\begin{bmatrix} S_1(k) \\ S_2(k) \\ S_3(k) \\ S_4(k) \end{bmatrix} = \begin{bmatrix} \eta_k \dot{\beta}_k \\ 1 + \eta_k \dot{\rho}_k \\ \dot{\beta}_k \\ \dot{\rho}_k \end{bmatrix} + \frac{1}{r_k} (Id_2 \otimes \mathcal{R}_{\beta_k}) \mathbf{u}_k + \frac{\sigma}{r_k} \mathbf{v}_k \ .$$

Expression (47) is obtained by remarking that $(Id_2 \otimes \mathcal{R}_{\beta_k})\mathbf{v}_k$ has the same statistical properties than $\mathbf{v}_k$. Now, we study the term y_{k+1} using:

$$y_{k+1} = f^{lp}_c(\mathbf{x}_{k+1}) = \begin{bmatrix} \operatorname{atan2}\left(r_y(k+1), r_x(k+1)\right) \\ \ln\left(\sqrt{r_x^2(k+1) + r_y^2(k+1)}\right) \\ \frac{r_x(k+1)v_y(k+1) - r_y(k+1)v_x(k+1)}{r_x^2(k+1) + r_y^2(k+1)} \\ \frac{r_x(k+1)v_x(k+1)) + r_y(k+1)v_y(k+1)}{r_x^2(k+1) + r_y^2(k+1)} \end{bmatrix} . \tag{48}$$

Finally, putting (47) in (48), we obtain

$$\left\{ \begin{array}{lcl} \beta_{k+1} & = & \beta_k + \operatorname{atan2}\left(S_1(k), S_2(k)\right), \\ \rho_{k+1} & = & \rho_k + \frac{1}{2}\ln(S_1^2(k) + S_2^2(k)), \\ \begin{bmatrix} \dot{\beta}_{k+1} \\ \dot{\rho}_{k+1} \end{bmatrix} & = & \frac{1}{(S_1^2(k) + S_2^2(k))} \begin{bmatrix} S_2(k) & -S_1(k) \\ S_1(k) & S_2(k) \end{bmatrix} \begin{bmatrix} S_3(k) \\ S_4(k) \end{bmatrix} \end{array} \right.$$

where:

$$\begin{bmatrix} S_1(k) \\ S_2(k) \\ S_3(k) \\ S_4(k) \end{bmatrix} = \begin{bmatrix} \eta_k \dot{\beta}_k \\ 1 + \eta_k \dot{\rho}_k \\ \dot{\beta}_k \\ \dot{\rho}_k \end{bmatrix} + \frac{1}{r_k}(Id_2 \otimes \mathcal{R}_{\beta_k})\mathbf{u}_k + \frac{\sigma}{r_k}\mathbf{v}_k \tag{49}$$

APPENDIX B: PROOF OF EQ.(42)

The idea consists in simplifying the problem by supposing that the target follows a uniform rectilinear motion:

$$\mathbf{x}_{k+1} = F\mathbf{x}_k + \mathbf{u}_k \ . \tag{50}$$

This entails to suppose that the noise w_k is null in the equation of evolution (9) associated with the problem of BOT presented in section 1. In other words, the target has a deterministic behavior. All the interest of this approach lies then in the fact that one can parameterize the trajectory by a low number of parameters. One generally chooses to parameterize it by a target state at one moment of reference. In the following, the state of reference is the initial state x_0. The target's state at time k is then obtained starting from this initial state using the following relation:

$$\mathbf{x}_k = F_k \mathbf{x}_0 + U_k$$

where:

$$\begin{cases} F_k = \overbrace{F \times \ldots \times F}^{k \ times} \ , \\ U_k = \sum_{j=0}^{k-1} F_{k-j} \mathbf{u}_j \ . \end{cases} \tag{51}$$

By using the equation of observation (4) and the equation (51), one obtains the following result:

$$\begin{pmatrix} z_1 \\ \vdots \\ z_\kappa \end{pmatrix} = \begin{pmatrix} h(F_1\mathbf{x}_0 + U_1) \\ \vdots \\ h(F_\kappa \mathbf{x}_0 + U_\kappa) \end{pmatrix} + \begin{pmatrix} w_1 \\ \vdots \\ w_\kappa \end{pmatrix} \ . \tag{52}$$

The key point of the demonstration consists in rewriting (52) by using the modified logarithmic polar coordinates via the formulas of transitions (15) and (16).

REFERENCES

[1] X. Rong Li and V. Jilkov. A Survey of Maneuvering Target Tracking Part I: Dynamics Models. IEEE *Transactions on Aerospace and Electronic Systems*, 39(4):1333–1364, October 2003.

[2] D.W. Whitcombe. Pseudo State Measurements Applied to Recursive Theory Filtering. In *Proceedings of the Third Symposium on Nonlinear Estimation Theory and its Applications*, September 1972.

[3] T. L. Song and J. L. Speyer. A Stochastic Analysis of a Modified Gain Extended Kalman Filter with Applications to Estimation with Bearings Only Measurements. IEEE *Transactions on Automatic Control*, 30(10):940–949, October 1985.

[4] V.J. Aidala and S.E. Hammel. Utilization of Modified Polar Coordinates for Bearing-Only Tracking. IEEE *Transactions on Automatic Control*, 28(3):283–294, March 1983.

[5] N. Peach. Bearing-Only Tracking using a Set of Range-Parametrised Extented Kalman Filters. IEEE *Proc.-Control Theory Appl.*, 142(1):73–80, January 1995.

[6] N. Gordon, D. Salmond, and A. Smith. Novel Approach to Non-Linear/Non-Gaussian Bayesian State Estimation. *Proc. Ins. Elect. Eng.*, 140(2):107–113, April 1993.

[7] S. Arulampalam and B. Ristic. Comparaison of the Particle Filter with Range-Parameterised and Modified Polar EKFs for Angle-Only Tracking. In *Conference on Signal and Data Processing of Small Targets*, volume 4048, pages 288–299, SPIE Annual International Symposium on Aerosense, 2000.

[8] A. Doucet, S. Godsill, and C. Andrieu. On Sequential Monte Carlo Sampling Methods for Bayesian Filtering. Technical report, Cambridge University Engineering Department, 2000.

[9] M.S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A Tutorial on Particle Filters for Online Non-Linear/Non-Gaussian Bayesian Tracking. IEEE *Transactions on Signal Processing*, 50(2), February 2002.

[10] A. Doucet, N. De Freitas, and N. Gordon. *Sequential Monte Carlo Methods in Practice*. Springer, 2001.

[11] P. Fearnhead. *Sequential Monte-Carlo Methods in Filter Theory*. PhD thesis, University of Oxford, 1998.

[12] T. Bréhard and J-P. Le Cadre. Closed-form Posterior Cramér-Rao Bound for Bearings-Only Tracking. *to appear in* IEEE *Transactions on Aerospace and Electronic Systems*, 2006.

[13] H. Weiss and J. B. Moore. Improved Extended Kalman Filter Design for Passive Tracking. IEEE *Transactions on Automatic Control*, 25(4):807–811, 1980.

[14] C. Berzuini, N. Best, W. Gilks, and C. Larizza. Dynamic Conditional Independence Models and Markov Chain Monte Carlo Methods. *Journal of the American Statisitical Association*, 92(440):1403–1412, 1997.

[15] W.R. Gilks and C. Berzuini. Following a Moving Target - Monte Carlo Inference for Dynamic Bayesian Models. *Journal of the Royal Statistical Society*, 63:127–148, 2001.

[16] F. Bavencoff, J.-M. Vanpeperstraete, and J.-P. Le Cadre. Constrained Bearings-Only Target Motion Analysis via Monte Carlo Markov Chain Methods. IEEE *Transactions on Aerospace and Electronic Systems*, 42(3), 2006.

[17] M.-H. Chen and B.W. Schmeiser. General Hit-and-Run Monte-Carlo Sampling for Evaluating Multidimensional Integrals. *Operations Research Letters 19*, pages 161–169, 1996.

[18] M. Hernandez. Performance for GMTI Tracking. In *6th Int. Conf. on Information Fusion, Cairns,Queensland, Australia*, July 2003.

[19] C. Jauffret and D. Pillon. Observability in Passive Target Motion Analysis. IEEE *Transactions on Aerospace and Electronic Systems*, 32(4):1290–1300, october 1996.

[20] N. Chopin. Central limit theorem for sequential monte-carlo methods and its application to bayesian inference. *The Annals of Statistics*, 32(6), 2004.

[21] B. Ristic, S. Arulampalam, and N. Gordon. *Beyond the Kalman Filter, Particle Filters for Tracking Applications*. Artech House Publishers, 2004.

[22] S.C. Nardone and V.J. Aidala. Observability Criteria for Bearings-Only Target Motion Analysis. IEEE *Transactions on Aerospace and Electronic Systems*, 17(2):161–166, March 1981.

[23] J.-P. Le Cadre. Properties of Estimability Criteria for Target Motion Analysis. *IEE Proc. Radar, Sonar and Navigation*, 145(2):92–99, April 1998.

[24] T. Bréhard and J-P. Le Cadre. A New Approach for the Bearings-Only Problem: Estimation of the Variance-to-Range Ratio. In *7th International Conference on Information Fusion*, Stockholm, Sweden, 2004.

[25] W. K. Hasting. Monte Carlo Sampling Methods Using Markov Chains and Their Application. *Biometrika*, 57:97–109, 1970.

[26] D. MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, 2003.

[27] S.C. Nardone, A.G. Lindgren, and K.F. Gong. Fundamental Properties and Performance of Conventional Bearing-Only Target Motion Analysis. IEEE *Transactions on Automatic Control*, 29(9):775–787, September 1984.

[28] J-P. Le Cadre and C. Jauffret. On the Convergence of Iterative Methods for Bearings-Only Tracking. IEEE *Transactions on Aerospace and Electronic Systems*, 35(3), July 1999.

[29] M. Li and C-C. Lim. Observability-Enhanced Proportianal Navigation Guidance with Bearings-Only Measurements. *J. Austral. Math. Soc. Ser. B*, 40:457–512, 1999.

[30] C. Hue, J-P. Le Cadre, and P. Pérez. Sequential Monte Carlo Methods for Multiple Target Tracking and Data Fusion. *IEEE Trans. on Signal Processing*, 50(2):309–325, February 2002.

[31] M. Vihola. Random Set Particle Filter for Bearings-Only Multitarget Tracking. In *SPIE, Vol. 5809, pp. 301-312, Mar. 28-30 2005, Orlando, Florida*, 2005.

[32] S. S. Blackman and S. H. Roskowski. Application of IMM Filtering to Passive Ranging. In *Conference on Signal and Data Processing of Small Targets*, volume 3809, pages 270–281, SPIE, 1999.

[33] B. Ristic and S. Arulampalam. Tracking a Manoeuvring Target using Angle-Only Measurements : Algorithms and Performance. *Signal Processing*, 83:1223–1238, February 2002.

[34] T. Bréhard and J.-P. Le Cadre. Distributed Target Tracking for Nonlinear Systems: Application to Bearings-Only Tracking. In *Proc. Int. Conf. Information Fusion (FUSION'05)*, Philadelphia, USA, 2005.

[35] R.R. Allen and S.S. Blackman. Implementation of an Angle-Only Tracking Filter. In *SPIE, Signal and Data Processing of Small Targets 1991, Oliver E. Drummond; Ed*, pages 292–303, August 1991.

[36] W. Grossman. Bearing-Only Tracking: A Hybrid Coordinate System Approach. *Journal of Guidance, Control and Dynamics*, 17(3), 1994.

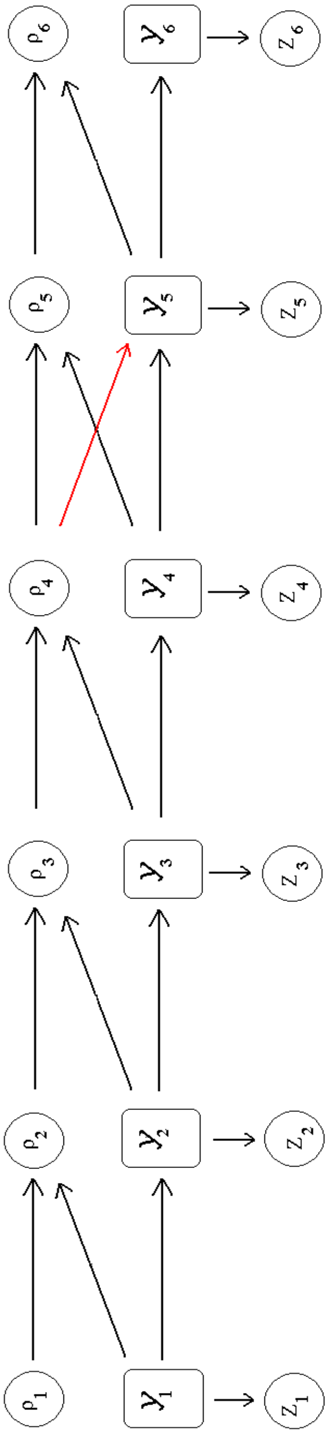

Fig. 13. Representation of the Markovian process associated with the dynamic system (26).The observer maneuvers only at time 4. This operation causes to add a conditioning represented by the red arrow. As long as the observer does not maneuver, the diffusion of components $\mathcal{Y}_k$ does not depend on ρ_0.

Algorithm 1 Hierarchical particle filter

```
• FILTER 1: sequential importance sampling for $\mathcal{Y}_k$:
  1. sampling :
     for $i = 1...N_1$, sample$\mathcal{Y}_k^{(i)} \sim p(\mathcal{Y}_k|\mathcal{Y}_{k-1}^{(i)})$,
  2. compute un-normalized weights :
     for $i = 1...N_1$, compute $\widetilde{w}_k^{(i)} = p(z_k|\mathcal{Y}_k^{(i)})w_k^{(i)}$,
  3. normalize weights :
     for $i = 1...N_1$, compute $w_k^{(i)} = \frac{\widetilde{w}_k^{(i)}}{\sum_{j=1}^{N} \widetilde{w}_k^{(j)}}$.
• FILTER 2: sampling for $\rho_k$:
  for $j = 1...N_2$, sampling $\rho_0^{(j)} \sim p(\rho_0)$
  for $j = 1...N_2$, sampling $\rho_k^{(j)} \sim \sum_{i=1}^{N_1} \widetilde{w}_k^{(i)} \delta_{\rho_0^{(j)}+\Delta_{\mathcal{Y}_{0:k}^{(i)}}}(\rho_k)$ ,
• Monte Carlo estimation
    the clouds $\{\mathcal{Y}_k^{(i)}, w_k^{(i)}\}_{i=1...N_1}$ and $\{{\rho_k}^{(j)}\}_{j=1...N_2}$ allow to approximate the posterior
  distribution function
    $p(y_k|\mathbf{z}_{1:k}) \simeq \sum_{i=1}^{N_1}\sum_{j=1}^{N_2} \frac{w_k^{(i)}}{N_2}\, \delta_{\mathcal{Y}_k^{(i)}}(\mathcal{Y}_k)\delta_{\rho_k^{(i)}}(\rho_k)$
    and for $g$ Lebesgue integrable
    $\mathbb{E}\{g(y_k)|\mathbf{z}_{1:k}\} \simeq \sum_{i=1}^{N_1}\sum_{j=1}^{N_2} g(\mathcal{Y}_k^{(i)}, \rho_k^{(j)})\, \frac{w_k^{(i)}}{N_2}$
• resampling :
  1. compute $N_{eff} = \frac{N_1}{\sum_{i=1}^{N_1}(w_k^{(i)})^2}$,
  2. if $N_{eff} < N_{threshold}$,
  draw N particles $\widetilde{\mathcal{Y}}_{0:k}^{(i)}$ in $\{\mathcal{Y}_{0:k}^{(i)}\}_{i=1...N_1}$ proportionally to weights $\{w_k^{(i)}\}_{i=1...N_1}$ and set
  $w_k^{(i)} = 1/N_1$, $\mathcal{Y}_{0:k}^{(i)} = \widetilde{\mathcal{Y}}_{0:k}^{(i)}$ for $i = 1...N_1$.
```

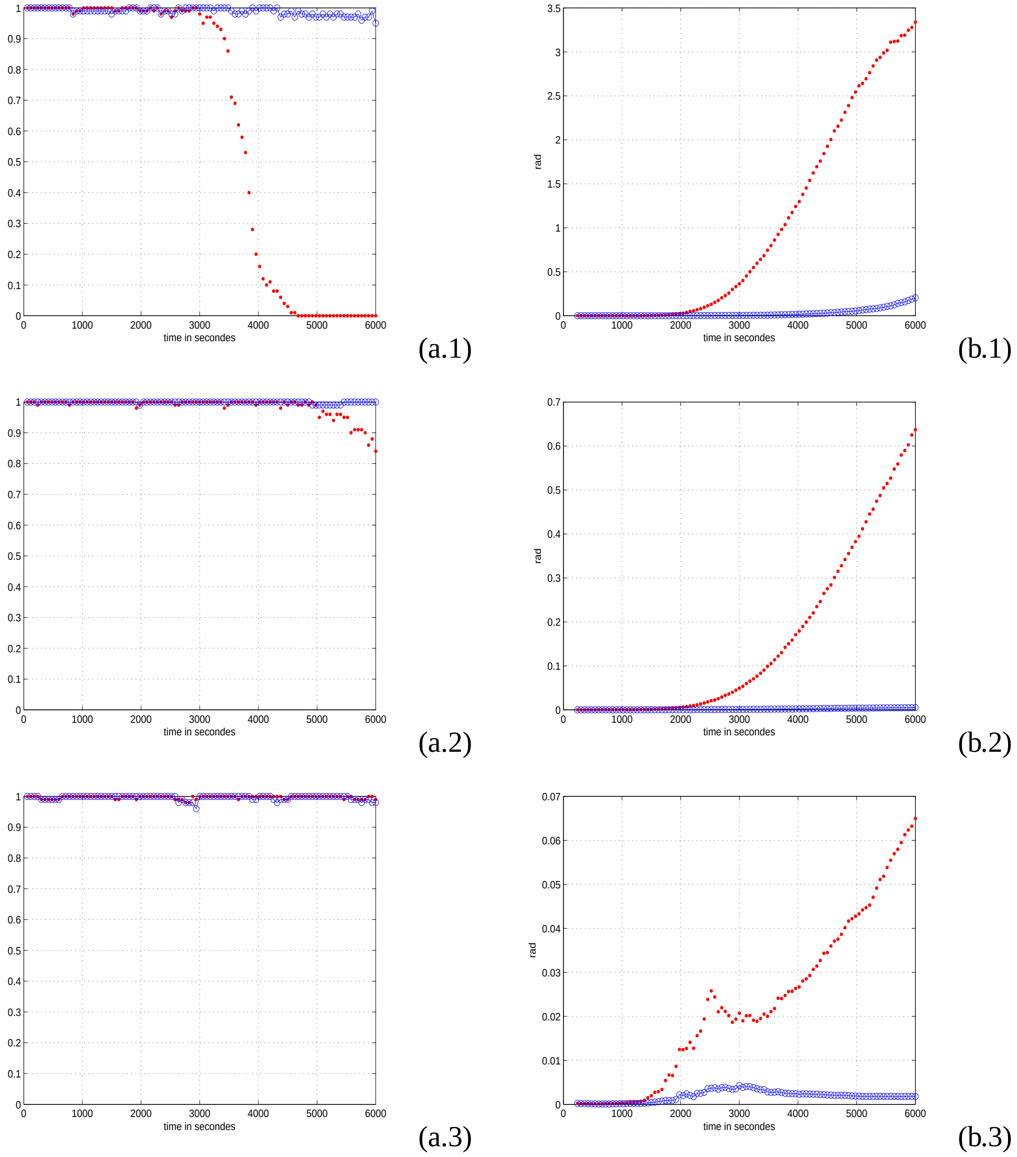

Fig. 14. Comparison of the performances in the course of time of the hierarchical filter (solid line) and of the bootstrap filter (dotted line) for different scenarios. **On the left side**, evolution of the divergence (probability that the state of the target is in the area of confidence) (a.1): scenario 1, (a.2) scenario 2, (a.3) scenario 3. **On the right side**, comparison of the evolution of the mean square error in the course of time for different scenarios (b.1): scenario 1, (b.2) scenario 2, (b.3) scenario 3.