



Stochastic Filtering with Networked Sensing

Hana Baili, Hichem Snoussi

► **To cite this version:**

Hana Baili, Hichem Snoussi. Stochastic Filtering with Networked Sensing. IEEE 70th Vehicular Technology Conference (VTC'09), Sep 2009, Anchorage, United States. CD-ROM Proceedings - 5 p. hal-00424403

HAL Id: hal-00424403

<https://hal-supelec.archives-ouvertes.fr/hal-00424403>

Submitted on 15 Oct 2009

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Stochastic Filtering with Networked Sensing

Hana Baili

Department of Signal Processing and Electronic Systems
Supélec
3 rue Joliot-Curie, Gif sur Yvette 91192 France
Email: hana.baili@supelec.fr

Hichem Snoussi

ISTIT/M2S
University of Technology of Troyes
12 rue Marie Curie, 10000 France
Email: snoussi@utt.fr

Abstract—This paper presents a solution to the problem of target tracking within a sensor network. This is based on modeling the target dynamics as a Markov process, and is reformulated as nonlinear filtering. When the target motion is a diffusion process, the optimal filtering involves a resolution of the backward Kolmogorov equation. Since an explicit solution for this partial differential equation does not exist in general, we recover the filtering by an alternative Monte-Carlo approach.

I. INTRODUCTION

A. Sensor networks: a technology and applications

Networked sensing is an interdisciplinary domain supporting many researches. As to application, we consider target tracking where sensing produces measurements such as the azimuth angle. The tracking problem consists in estimating in real time the target trajectory. This is based on a model for the target motion which either derives from physics laws (i.e. knowledge-based) or is set as a black-box prior. To improve the estimation there are in general multiple sensors, typically here, a sensor network. In reality sensors may measure different quantities, or may have different precisions on the same measurement. Besides, sensors are not synchronized so that the measurements proceed during irregular time intervals. As regards the technology, we are interested in a challenging technology of “smart” sensors whose role is not only measurement, but also data processing and decision making. Target tracking with networked sensing is mathematically reformulated in the next subsection.

B. Stochastic filtering

A possible model for the problem is the following. We have an \mathbf{R}^d -valued continuous-time process $X = \{X_t\}_{t \geq 0}$. We assume that X is a Markov process, more precisely, a diffusion process which evolves according to the stochastic differential equation (SDE)

$$dX_t = \sigma(X_t) dW_t + b(X_t) dt \quad \mathcal{L}(X_0) = \mu \quad (1)$$

with coefficients $\sigma(x) = (\sigma_j^i(x))$ and $b(x) = (b^1(x), \dots, b^d(x))$. $W = \{W_t\}_{t \geq 0}$ is a d -dimensional Wiener process and μ is a probability distribution on \mathbf{R}^d .

We have noisy nonlinear observations of X : an \mathbf{R}^q -valued discrete-time process $Y = \{Y_n\}_{n \in \{1, 2, \dots, N\}}$ indexed at N irregularly spaced instants t_1, t_2, \dots, t_N . There are different

ways in which the observations occur; in any case the observation process Y is related to the unobservable (state) process X via the conditional distribution

$$\mathbf{P}(Y_n \in \Gamma | \mathcal{F}_n^X \vee \sigma\{Y_1, \dots, Y_{n-1}\}),$$

for Γ a Borel-measurable set from \mathbf{R}^q . \mathcal{F}_n^X and $\sigma\{Y_1, \dots, Y_{n-1}\}$ are the σ -algebras generated respectively by $\{X_t, 0 \leq t \leq t_n\}$ and $\{Y_1, \dots, Y_{n-1}\}$.

We wish to compute the best (or optimal) estimate, in a mean square sense, of $f(X_t)$ given the observations $\{Y_1, \dots, Y_n\}$ up to time t , i.e., to compute the conditional expectation

$$\mathbf{E}[f(X_t) | Y_1, \dots, Y_n], \quad t_n \leq t, n \geq 1,$$

for all reasonable functions f on \mathbf{R}^d .

II. FILTERING EQUATIONS: THE OPTIMAL FILTER VERSUS PARTICLE FILTERING

We assume that $\mathbf{P}(X_t \leq x | Y_1, \dots, Y_n)$, $t_n \leq t$, $n \geq 1$, possesses a density with respect to the Lebesgue measure ψ on \mathbf{R}^d :

$$\Pi_{(X_t | Y_1, \dots, Y_n)}(x) := \frac{d\mathbf{P}(X_t \leq x | Y_1, \dots, Y_n)}{\psi(dx)}.$$

A. Observation scheme

In a wireless sensor and micro-sensor network energy consumption is a key factor for the sensor lifetime. Optimizing energy consumption is a challenge requiring energy-efficient protocols. The following is a protocol which is adapted to the application. There is only one active sensor at the same time; the best sensor, for producing the next observation, is that one with the most probable position with respect to the last estimated target position density.

We single out one observation scheme: for any $n \geq 1$ we have

$$Y_n = h(X_n, C_n), \quad (2)$$

where h is a known function from $\mathbf{R}^d \times \mathbf{R}^d$ into \mathbf{R}^q , and $C = \{C_n\}_{n \in \{1, 2, \dots, N\}}$ is an \mathbf{R}^d -valued discrete-time process which represents the control. Specifically, let $X_t = (X_t^1, X_t^2)$ be the target position at time t in a plane equipped with

horizontal and vertical axes \vec{x}^1 and \vec{x}^2 , respectively. Let there be fixed locations x_i of sensors; each sensor could be the active sensor at an observation time t_n and the control C_n is nothing but the active sensor coordinates (C_n^1, C_n^2) .

In a limited sensing range we have for instance, for $n \geq 1$

$$\begin{cases} Y_n^1 = \sqrt{(X_n^1 - C_n^1)^2 + (X_n^2 - C_n^2)^2} + e_n^r \\ Y_n^2 = \arctan\left(\frac{X_n^2 - C_n^2}{X_n^1 - C_n^1}\right) + e_n^a \end{cases}$$

together with, for some t

$$C_n = \arg \max_{x_i} \Pi_{(X_t|Y_1, \dots, Y_{n-1})}(x) \quad t_{n-1} \leq t < t_n$$

subject to the initialization $t_0 = 0$ and $\Pi_{X_0} = \mu$. Here the sensor measurement Y^1 is the distance between the active sensor and the target, and the line they form leans at an angle Y^2 from the horizontal axis. We assume that the observation noises e^r and e^a are independent processes, white, Gaussian and centered with standard deviations σ^r and σ^a , respectively.

B. Conditional density characterization: the optimal filter

$\Pi_{(X_t|Y_1, \dots, Y_n)}(x)$, $t_n \leq t < t_{n+1}$, $n \geq 1$, solves the Chapman-Kolmogorov equation

$$\Pi_{(X_t|Y_1, \dots, Y_n)}(x) = \int_{u \in \mathbf{R}^d} p(t - t_n, u, x) \Pi_{(X_n|Y_1, \dots, Y_n)}(u) du, \quad (3)$$

where $p(t, u, x)$ is the density of the transition function $P(t, u, \Gamma)$ of the Markov process X . But for fixed x , $p(t, u, x)$ solves the backward Kolmogorov equation

$$\frac{\partial p}{\partial t} = \frac{1}{2} \sum_{i,j} a^{ij}(u) \frac{\partial^2 p}{\partial u^i \partial u^j}(u) + \sum_i b^i(u) \frac{\partial p}{\partial u^i}(u), \quad t > 0, \\ \lim_{t \downarrow 0} p(t, u, x) = \epsilon(u - x). \quad (4)$$

Here $a^{ij}(u) = \sum_k \sigma_k^i(u) \sigma_k^j(u)$ and $\epsilon(u - x)$ is a unit mass at position x .

At each observation instant t_n , $n \geq 1$, $\Pi_{(X_n|Y_1, \dots, Y_n)}(x)$ solves a ‘‘Bayes-like’’ rule, subject to $\Pi_{X_0} = \mu$:

$$\Pi_{(X_n|Y_1, \dots, Y_n)}(x) \propto \Pi_{(Y_n|X_n=x, Y_1, \dots, Y_{n-1})}(Y_n) \times \\ \int_{u \in \mathbf{R}^d} p(t_n - t_{n-1}, u, x) \Pi_{(X_{n-1}|Y_1, \dots, Y_{n-1})}(u) du \quad (5)$$

Proof of (5): Let f and g be arbitrary functions respectively on \mathbf{R}^d and \mathbf{R}^q . We have

$$\mathbf{E}[f(X_n)g(Y_n)|Y_1, \dots, Y_{n-1}] = \\ \mathbf{E}[\mathbf{E}[f(X_n)|Y_1, \dots, Y_n]g(Y_n)|Y_1, \dots, Y_{n-1}]. \quad (6)$$

The left hand side develops as follows:

$$\mathbf{E}[f(X_n)g(Y_n)|Y_1, \dots, Y_{n-1}] = \\ \mathbf{E}[f(X_n)\mathbf{E}[g(Y_n)|\mathcal{F}_n^X \vee \sigma\{Y_1, \dots, Y_{n-1}\}]|Y_1, \dots, Y_{n-1}] = \\ \mathbf{E}\left[f(X_n) \int_y g(y) \Pi_{(Y_n|\mathcal{F}_n^X \vee \sigma\{Y_1, \dots, Y_{n-1}\})}(y) dy \right. \\ \left. |Y_1, \dots, Y_{n-1}\right] = \\ \mathbf{E}\left[\mathbf{E}\left[f(X_n) \int_y g(y) \Pi_{(Y_n|X_n, Y_1, \dots, Y_{n-1})}(y) dy |X_{n-1}\right] \right. \\ \left. |Y_1, \dots, Y_{n-1}\right] = \\ \mathbf{E}\left[\int_x f(x) \int_y g(y) \Pi_{(Y_n|X_n=x, Y_1, \dots, Y_{n-1})}(y) dy \times \right. \\ \left. p(t_n - t_{n-1}, X_{n-1}, x) dx |Y_1, \dots, Y_{n-1}\right] = \\ \int_u \int_x f(x) \int_y g(y) \Pi_{(Y_n|X_n=x, Y_1, \dots, Y_{n-1})}(y) dy \times \\ p(t_n - t_{n-1}, u, x) dx \Pi_{(X_{n-1}|Y_1, \dots, Y_{n-1})}(u) du.$$

To calculate the right-hand side of (6) set

$$\mathbf{E}[f(X_n)|Y_1, \dots, Y_n] = G(Y_1, \dots, Y_n).$$

Then

$$\mathbf{E}[G(Y_1, \dots, Y_n)g(Y_n)|Y_1, \dots, Y_{n-1}] = \\ \mathbf{E}[\mathbf{E}[G(Y_1, \dots, Y_n)g(Y_n)|\mathcal{F}_n^X \vee \sigma\{Y_1, \dots, Y_{n-1}\}] |Y_1, \dots, Y_{n-1}] = \\ \mathbf{E}\left[\int_y G(Y_1, \dots, Y_{n-1}, y)g(y) \times \right. \\ \left. \Pi_{(Y_n|X_n, Y_1, \dots, Y_{n-1})}(y) dy |Y_1, \dots, Y_{n-1}\right] = \\ \mathbf{E}\left[\mathbf{E}\left[\int_y G(Y_1, \dots, Y_{n-1}, y)g(y) \times \right. \right. \\ \left. \left. \Pi_{(Y_n|X_n, Y_1, \dots, Y_{n-1})}(y) dy |X_{n-1}\right] |Y_1, \dots, Y_{n-1}\right] = \\ \mathbf{E}\left[\int_x \int_y G(Y_1, \dots, Y_{n-1}, y)g(y) \times \right. \\ \left. \Pi_{(Y_n|X_n=x, Y_1, \dots, Y_{n-1})}(y) dy \times \right. \\ \left. p(t_n - t_{n-1}, X_{n-1}, x) dx |Y_1, \dots, Y_{n-1}\right] = \\ \int_u \int_x \int_y G(Y_1, \dots, Y_{n-1}, y)g(y) \times \\ \Pi_{(Y_n|X_n=x, Y_1, \dots, Y_{n-1})}(y) dy \times \\ p(t_n - t_{n-1}, u, x) dx \Pi_{(X_{n-1}|Y_1, \dots, Y_{n-1})}(u) du.$$

Since g is arbitrary, we deduce that

$$G(Y_1, \dots, Y_{n-1}, Y_n) \propto$$

$$\int_u \int_x f(x) \Pi_{(Y_n | X_n = x, Y_1, \dots, Y_{n-1})}(Y_n) \times p(t_n - t_{n-1}, u, x) dx \Pi_{(X_{n-1} | Y_1, \dots, Y_{n-1})}(u) du.$$

But

$$G(Y_1, \dots, Y_n) = \int_x f(x) \Pi_{(X_n | Y_1, \dots, Y_n)}(x) dx,$$

and f is arbitrary; this obviously leads to (5). \diamond

C. The approximate filter: a Monte-Carlo particle filter

The true filter (3-5) which is the optimal solution in a mean square sense for continuous-discrete-time nonlinear filtering is computationally intractable. In fact, $p(t, u, x)$ is the solution of the backward Kolmogorov equation (4) and we have no explicit form for this solution in general. This section presents an approximation for the posterior distribution $\mathbf{P}(X_t \in \Gamma | Y_1, \dots, Y_n)$, $t_n \leq t$, $n \geq 1$, as a weighted sum of random Dirac's measures:

$$\mathbf{P}(X_t \in \Gamma | Y_1, \dots, Y_n) \approx \sum_{k=1}^K w_k \epsilon_{\xi_k}(\Gamma), \quad (7)$$

where the particles ξ_k are independent identically distributed random variables with “the same” law as X_t ; here these particles are samples drawn from the Euler discretization of the SDE (1). Then, for any function f on \mathbf{R}^d , we have:

$$\mathbf{E}[f(X_t) | Y_1, \dots, Y_n] \approx \sum_{k=1}^K w_k f(\xi_k), \quad t_n \leq t, n \geq 1.$$

As to the weights $\{w_k\}$, they are updated only as and when an observation Y_n proceeds, each one according to the likelihood of its corresponding particle, i.e. at each observation instant t_n

$$w_k = \frac{\Pi_{(Y_n | X_n = \xi_k, Y_1, \dots, Y_{n-1})}(Y_n)}{\sum_{\ell=1}^K \Pi_{(Y_n | X_n = \xi_\ell, Y_1, \dots, Y_{n-1})}(Y_n)},$$

where obviously the $\{\xi_k\}$'s are now samples with “the same” law as X_n .

Now besides sampling and approximation there may be *resampling* at each observation instant t_n : the set of particles is updated for removing particles with small weights and duplicating those with important weights. We simulate K new iid random variables according to

$$\sum_{k=1}^K w_k \epsilon_{\xi_k}.$$

(This is *importance* resampling.) The new particles $\xi_k^{(new)}$ have new weights

$$w_k^{(new)} \propto \Pi_{(Y_n | X_n = \xi_k^{(new)}, Y_1, \dots, Y_{n-1})}(Y_n),$$

and give a new approximation:

$$\mathbf{P}(X_n \in \Gamma | Y_1, \dots, Y_n) \approx \sum_{k=1}^K w_k^{(new)} \epsilon_{\xi_k^{(new)}}(\Gamma).$$

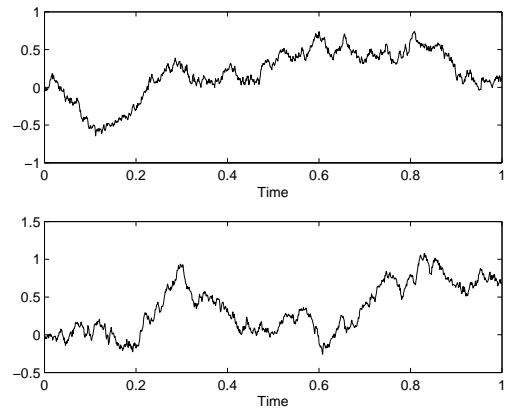
Finally, These new particles are used to initialize the Euler discretization scheme for the next sampling and posterior distribution approximation.

III. DEMONSTRATION

Consider the following diffusion as the target motion:

$$dX_t = \alpha \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} X_t dt + dW_t, \quad \mathcal{L}(X_0) = \mu,$$

with $\alpha = -0,5$. The following figure shows particular trajectories of the state components during one unit of time.



To evaluate the quality of filtering, a particular path of X is simulated; **Figure 1** shows the points on this simulated path which are the true target positions at some well determined instants. These instants of interest are called filtering instants. Naturally, observation instants t_n form a subset of the filtering instants. So Figure 1 is useful as a reference. It shows the sensing range and locations; here 100 sensors are deployed and distributed randomly about a range of 8×8 (units of surface), but their locations are fixed throughout.

Figure 2 shows a run on an arbitrary couple of observation times. The circle is the true target position and the triangle is the posterior estimate for that position. The star is the active sensor which produces the observation, calculates the target position estimate and decides which sensor is the next to be activated. This figure is to illustrate the active sensor selection before each observation.

Note the accuracy of target tracking on **Figure 3** where we plot errors on the horizontal and vertical coordinates for 1000 filtering instants (which are all observation times as well).

The following is the remainder of implementation details of the Monte-Carlo particle filter.

- number of particles: $K = 1000$;
- standard deviations of observation noises: $\sigma^r = 0.1$ (unit of length), $\sigma^a = 1$ (degree);
- time step of the Euler scheme: 0.01 (unit of time);
- probability distribution on \mathbf{R}^2 , of the initial state X_0 :

$$\mu = \mathbf{N} \left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 10^{-4} & 0 \\ 0 & 10^{-4} \end{pmatrix} \right).$$

$\mathbf{N}(\underline{\mathbf{m}}, \Sigma)$ denotes the Gaussian law with mean the vector $\underline{\mathbf{m}}$ and covariance matrix Σ .

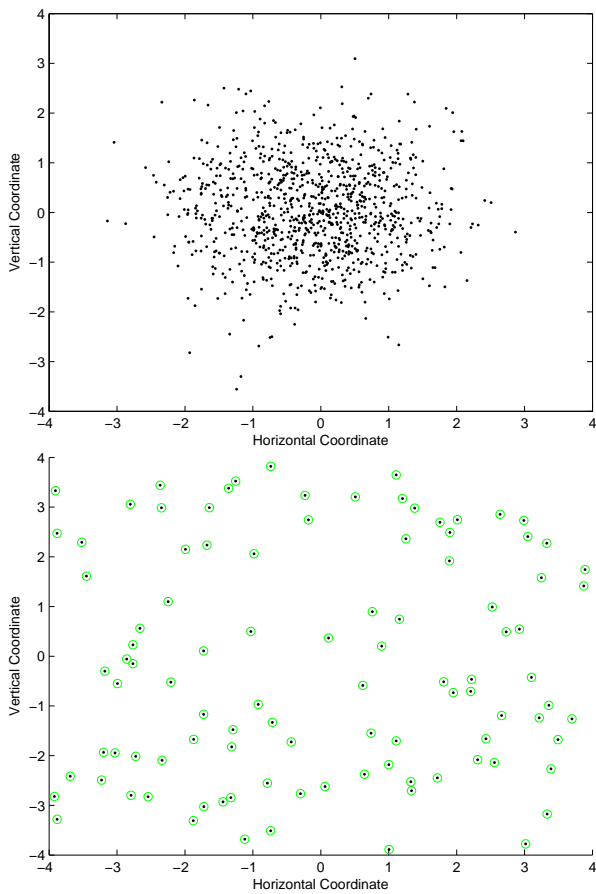


Fig. 1. From top to bottom: target positions at 1000 units of time, locations for sensors.

IV. CONCLUSION: SUMMARY AND PERSPECTIVES

Online filtering within a sensor network is considered when applied to target tracking, for the very special case of a diffusion (as the target motion). Observation times may be irregularly spaced but are assumed to be rigorously determined. A possible energy-efficient scheduling protocol for the sensor network goes as follows. At any observation time only one sensor is activated according to a criteria

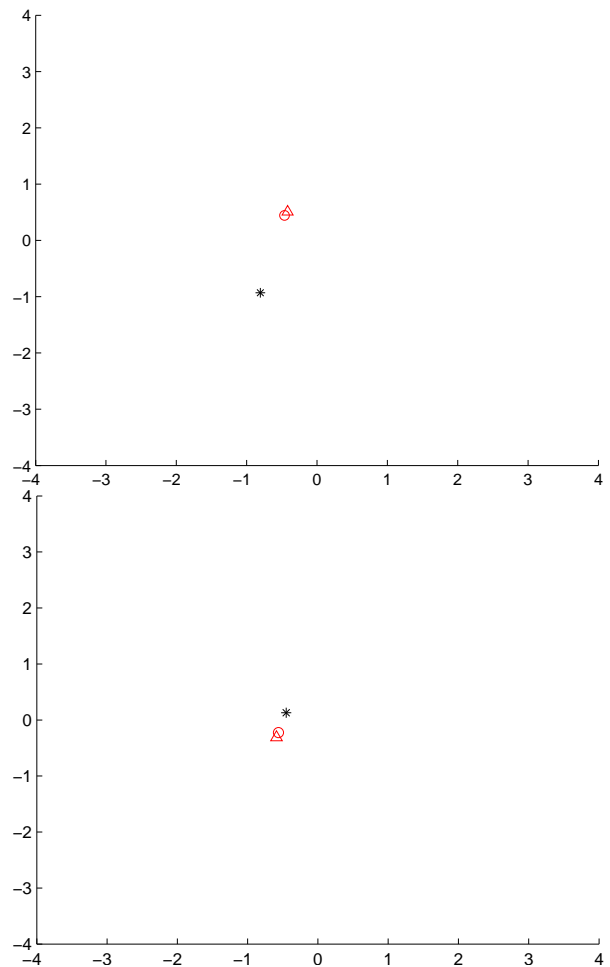


Fig. 2. Star: active sensor, circle: true target position, triangle: posterior estimate.

involving the last filtering distribution, estimated before that time. An alternative filter instead of the optimal one is proposed since the latter is not computationally feasible. It is based on samples drawn by discretization of a stochastic differential equation. Here we use the well known Euler scheme since there isn't a significant gain with more sophisticated discretization schemes. Throughout the demonstration the filter is validated on a synthetic example: a 2-dimensional Ornstein-Uhlenbeck process.

In reality there is an unknown number of targets in the scene, so we have to estimate this number when evolving in time, and in case of multiple targets, we also have to associate measurements to sensors. Besides, there is another question to answer: false alarms. In fact, we do not know whether a given measurement is a clutter measurement or a measurement from one of the targets. We shall address these existing problems by proposing a stochastic hybrid differential system for modeling; this will be developed elsewhere.

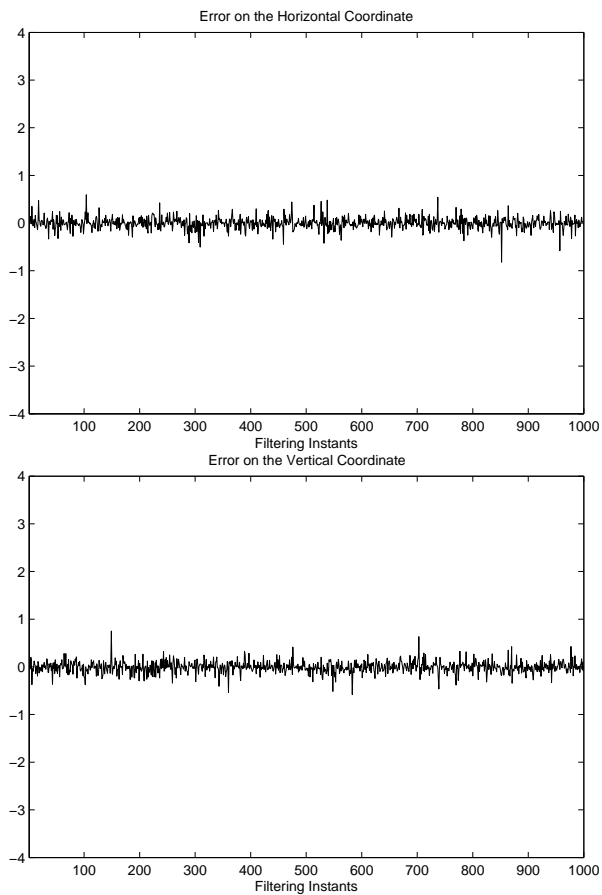


Fig. 3.

REFERENCES

- [1] H. Baili (2006). *Uncertainty Management for Estimation in Dynamical Systems*. Proceedings of IEEE Asia Pacific Conference on Circuits and Systems, p:1994-1997.
- [2] E. Lahalle, H. Baili, and J. Oksman (2008). *Online Estimation of Time-Varying Volatility using a Continuous-Discrete LMS Algorithm*. EURASIP Journal on Advances in Signal Processing Volume 2008 (2008), Article ID 532760, 8 pages doi:10.1155/2008/532760.
- [3] A. H. Jazwinski (1970). *Stochastic Processes and Filtering Theory*. Academic Press, New York.
- [4] P. E. Kloeden and E. Platen (1999). *Numerical Solution to Stochastic Differential Equations*. Springer, New York.
- [5] B. Øksendal (2003). *Stochastic Differential Equations: An Introduction with Applications*. Springer, New York, 6th edition.