Monte Carlo Filtering on Lie Groups

Alessandro Chiuso [1] and Stefano Soatto [2]

Abstract

We propose a nonlinear filter for estimating the trajectory of a random walk on a matrix Lie group with constant computational complexity. It is based on a finite-dimensional approximation of the conditional distribution of the state - given past measurements - via a set of fair samples, which are updated at each step and proven to be consistent with the updated conditional distribution. The algorithm proposed, like other Monte Carlo methods, can in principle track arbitrary distributions evolving on arbitrarily large state spaces. However, several issues concerning sample impoverishment need to be taken into account when designing practical working systems.

1 Introduction

In a wide range of modern engineering applications one is faced with having to estimate the state of dynamical systems evolving on Lie groups. For instance, in Computer Vision one is interested in estimating the pose of a moving object from its images. If the object is rigid, its pose can be described by a time-varying change of coordinates, that is a trajectory on the Lie group $SE(3)$, and the image can be described as a submersive map onto the image plane, modeled as the real projective plane (see [15] for more details). In Array Signal Processing one is interested in tracking the subspace spanned by the measurements of a sensor array. Using the geometry of Grassmann and Stiefel manifolds one can reduce this problem to the estimation of the state of a dynamical system on the n-dimensional orthogonal group $O(n)$ (see [18]). In Learning one is interested in tracking principal components that change in time as new training data are gathered. Again, this can be reduced to an estimation problem on $O(n)$ (see [6]). Many approaches to Rational Drug Design are formulated as optimization problems on the configuration of molecules, which has to obey kinematic constraints and can therefore be modeled as the n-dimensional product of Lie groups for some (large) n [9].

In all these cases, and many more, one has to infer the state of a system evolving on a matrix Lie group from a sequence of noisy measurements. If complete information on the dynamics of the state and on the input is *not* available, one can resort to a stochastic model, one of the very simplest being a first-order random walk. Therefore, we are interested in estimating the state of systems of the following type

$$\begin{cases} dg = e^{v} g dt \\ dv = d\beta \\ y(t) = h(g(t)) + w(t) \end{cases} \tag{1}$$

where g is a point on an n-dimensional matrix Lie group G and $\{\beta\}$ is a Brownian motion on the corresponding Lie algebra $\mathbf{g}$. The above equation is written in differential form as a shorthand for the corresponding Ito integral equation [14]. The measurements $y(t)$ live on the m-dimensional linear space $\mathbb{R}^m$ and are spoiled by noise $w(t)$.

1.1 Statement of the problem

In order to avoid the technicalities associated with stochastic calculus on manifolds [8], and since the measurements are typically gathered at discrete instants of time, we work in a discrete-time framework. Therefore, our prototypical model is of the following form

$$\begin{cases} g(t+1) = e^{v(t)} g(t) & g(0) = e \in G \\ v(t+1) = v(t) + \beta(t) & \beta(t) \sim \mathcal{N}(0,Q) \in \mathbf{g} \\ y(t) = h(g(t)) + w(t) & w(t) \sim \mathcal{N}(0,R) \in \mathbb{R}^m \end{cases}$$
$$w(t) \perp \beta(t) \perp v(0) \tag{2}$$

where $\{g(t)\}_{t=1,2,\ldots} \in G$ is a discrete-time trajectory on an n-dimensional matrix Lie group, $\{v(t)\}_{t=1,2,\ldots} \in \mathbf{g}$ is a velocity on the corresponding Lie algebra and $\{y(t)\}_{t=1,2,\ldots}$ is a sequence of measurements on a linear space. $\{\beta(t)\}$, $\{w(t)\}$ are Gaussian white zero-mean stochastic processes that we will assume to be mutually independent and also independent of the initial condition $v(0)$.

The complete information on the state $g(t), v(t)$ given measurements from time 0 to time t is given by the conditional density

$$p(g(t), v(t)|Y_t) \tag{3}$$

where Y_t denotes the σ-field generated by the measurements $\{y(0), \ldots, y(t)\}$. Alternatively, one may be interested in the density of the predicted values of g and v:

$$p(g(t+1), v(t+1)|Y_t). \tag{4}$$

[1] Università di Padova, Dipartimento di Elettronica ed Informatica, via Gradenigo 6a, 35131 Padova - Italy and Washington University, Department of Electrical Engineering, One Brookings dr. 1127, St.Louis - MO 63130. Email: chiuso@dei.unipd.it

[2] Washington University, Department of Electrical Engineering, One Brookings dr. 1127, St.Louis - MO 63130 and University of California Los Angeles, Department of Computer Science, Los Angeles - CA 90095. Email soatto@ee.wustl.edu

We call the first one "filtering density" and the second one "prediction density". Assuming that we are given an initial estimate for the density $p(g(0), v(0)|Y_0)$ where $Y_0 = \{y(0)\}$, or $p(g(0), v(0)|Y_{-1})$ where $Y_{-1} = \emptyset$, we are interested in ways of efficiently computing the update of the conditional densities above.

1.2 Evolution and representation of the conditional density

Calling the state $x_t \doteq (g(t), v(t))$, which belongs to the direct product of the Lie group G and its Lie algebra $\mathbf{g}$, we first notice that the model (2) imposes two Markov-type constraints

$$\begin{aligned} p(x_{t+1}|x_t, Y_t) &= p(x_{t+1}|x_t) \qquad (5) \\ p(y_t|x_t, Y_{t-1}) &= p(y_t|x_t) \qquad (6) \end{aligned}$$

and therefore the one-step prediction can be obtained using Chapman-Kolmogorov equation:

$$p(x_{t+1}|Y_t) = \int_{TG} p(x_{t+1}|x_t)p(x_t|Y_t)d\mu(x_t) \qquad (7)$$

where TG is the tangent bundle of G, which is isomorphic to the direct product of the Lie group G and its Lie algebra $\mathbf{g}$, and μ is a base measure on TG. For ease of notation let us denote $\ell(x_{t+1}, x_t, Y_t) := p(x_{t+1}|x_t)\, p(y_t|x_t)\, p(x_t|Y_{t-1})$. One step before the new measurement $y(t)$ becomes available, the filtering density has the following expression, which is derived directly from Bayes' rule

$$p(x_{t+1}|Y_t) = \frac{\int_{TG} \ell(x_{t+1}, x_t, Y_t)d\mu(x_t)}{\int_{TG\times TG} \ell(x_{t+1}, x_t, Y_t)d\mu(x_t)d\mu(x_{t+1})} \qquad (8)$$

The denominator $p(y_t|Y_{t-1})$ is just a constant normalization factor, and we can call its inverse k. If we call $p(x_{t+1}|Y_t) \doteq \rho(x_t)$ we can write the above equation in shorthand as

$$\rho(x_{t+1}) = k \int_{TG} p(x_{t+1}|x_t)p(y_t|x_t)\rho(x_t)d\mu(x_t) \qquad (9)$$

which is updated starting from a given $\rho(x_0)$. From the model (2) it is immediate to derive expressions for $p(x_{t+1}|x_t)$ and $p(y_t|x_t)$. The crucial issue, then, is how to choose an initial $\rho(x_0)$ in such a way that the integral above and the normalization constant k are easy to compute.

This is no easy task. If we write the above equation in operator form as

$$\rho(x_{t+1}) = \mathcal{Z}(\rho(x_t)) \qquad (10)$$

then a desirable situation would be for ρ to belong to a finite-dimensional class of density functions that are invariant under $\mathcal{Z}$. That way, equation (10) could be updated by computing a finite number of parameters. Unfortunately, we do not know of any such finite-dimensional class and, indeed, strong results of non-existence of finite-dimensional distributions invariant to Kolmogorov forward operator (the continuous-time equivalent of $\mathcal{Z}$) have been proven ([13]).

Even worse, despite starting from a unimodal density $\rho(x_0)$ it is possible for $\rho(x_t)$ to become multi-modal due to the nonlinear nature of the model (2) and, possibly, to the multi-modality of the measurement noise $w(t)$ (see for instance Blake and Isard [1] for a case of multi-modal measurement noise in Computer Vision).

If we resign to the idea of not being able to find a finite-dimensional *representation* of ρ, we can certainly resort to looking for a finite-dimensional *approximation.* There is in the literature a wealth of finite-dimensional approximations of the optimal filter, which include numerical integration, series expansions, projection onto the manifold of exponential densities [2] and various choices of parametric classes of densities such as sums of Gaussians [17] or splines [16].

All the above techniques resort, in one form or another, to expressing the conditional density in terms of the sum of scaled basis elements (e.g. polynomials, splines, Gaussians, simple functions etc.).

2 Monte Carlo

To develop some intuition on the motivations for using a Monte Carlo filter let us for a moment assume that x_t were one dimensional and represent the density ρ using its integral, the distribution P defined by

$$dP(x_t) \doteq \rho(x_t)d\mu(x_t); \qquad (11)$$

μ is the base measure of the tangent bundle TG where x_t is defined (in other words, ρ is the Radon-Nykodym derivative of P).

Then one could represent the distribution P with staircase functions with steps of equal width (with respect to μ) but different heights. This results in well-known "numerical integration filters" (see [16] and references therein).

On the other hand, one could approximate the distribution P with a staircase function with steps of equal heights but different widths (with respect to μ). This results in a "counting" or "sampling" representation of the density ρ and in so-called Monte Carlo filters, which we describe below. It should be noted that it is possible to represent a distribution with steps of different height *and* width (w.r.t. μ); this results in the notion of "properly weighted sample" and in a scheme called "importance sampling", as discussed in remark 2.

There are several advantages in using a sampling representation: one is that - as long as we keep the number of samples constant - there is no need to compute the normalization factor k. We start, therefore, by defining a random sample from a distribution P (or from its density ρ)

Definition 1 *A sequence of measurable functions on TG, $\{x^{(i)}\}_{i=1\dots N}$ represents a random sample from P (or ρ), indicated by $x^{(i)} \sim P(x)$ ($x^{(i)} \sim \rho(x)$), if they are independent and identically distributed according to P. We indicate with $s^i \sim P(x)$* a realization *of the random sample (a sequence of points $\{s^i\}_{i=1\dots N}$ in the tangent bundle TG).*

Consider now an integrable function ϕ. Its integral

$$\Phi \doteq \int \phi(x) dP(x) \tag{12}$$

can be approximated in terms of the random samples

$$\hat{\Phi}_N \doteq \frac{1}{N} \sum_{x^{(i)} \sim P(x)} \phi(x^{(i)}) \tag{13}$$

since, as is trivially verified,

$$\int \phi(x) dP(x) = E_P[\hat{\Phi}_N]. \tag{14}$$

This suggests a method for approximating the integral Φ using a realization as

$$\tilde{\Phi}_N \doteq \frac{1}{N} \sum_{s^i \sim P(x)} \phi(s^i) \tag{15}$$

since $lim_{N\to\infty}\tilde{\Phi}_N \overset{a.s.}{=} E[\hat{\Phi}_N] = \Phi$, by the law of large numbers $\hat{\Phi}_N$ is an unbiased and strongly consistent estimator of Φ. We call $\hat{\Phi}_N$ the the Monte Carlo *estimator* (a function) and $\tilde{\Phi}_N$ the Monte Carlo *estimate* (a value). $\hat{\Phi}_N$ differs from Φ by a zero-mean error with variance

$$E[(\Phi - \hat{\Phi}_N)^2] = \frac{1}{N} \int (\phi(x) - \Phi)^2 dP(x) \tag{16}$$

for a scalar ϕ.

Remark 1 (Monte Carlo approximation) *The last equation expresses the remarkable fact that the Monte Carlo estimate of an integral improves linearly with the number of samples and* does not depend upon the dimension of *x (the base space). This makes it particularly appealing for filtering in high-dimensional state-spaces. However, the reader should be aware that the dimension of the state-space does adversely affect the quality of the sample, a phenomenon well-known to statisticians as "sample impoverishment", measured with the so-called "effective sample size" (ESS) and treated with a collection of heuristics [10]. Roughly speaking, the ESS, $\tilde{N}$, is the length of a "good" fair sample which would give the same accuracy as in (16).*

Remark 2 (Importance sampling) *When the density ρ is easy to compute but difficult to sample from, the integral Φ can be re-written as follows*

$$\begin{aligned} \textstyle\int \phi(x) dP(x) &= \textstyle\int \phi(x)\rho(x) d\mu(x) \\ &= \textstyle\int \phi(x) \frac{\rho(x)}{q(x)} q(x) d\mu(x) \end{aligned} \tag{17}$$

where $q(x)$ is a density having the same support as ρ. Monte Carlo integration applied to the above equation results in

$$\int \phi(x) dP(x) = \lim_{N\to\infty} \frac{1}{N} \sum_{s^i \sim P(x)} \phi(s^i) w_i \tag{18}$$

where $w_i \doteq \frac{\rho(s^i)}{q(s^i)}$. The set of pairs $\{s^i, w_i\}$ is called a properly weighted sample *from the distribution $P(x)$, and the sampling described in the previous equation is known as* importance sampling *[12]. We note here that all the considerations we present in this paper can be extended to distributions represented by properly weighted samples, although here we restrict the discussion to $q(x) = \rho(x)$ in order to avoid the heuristic issues involved in choosing an appropriate importance density $q(x)$.*

2.1 Factor sampling

Consider now equation (9), where we represent the density ρ with a fair sample of length N. Using a realization from this sample, $s^i \sim \rho(x)$, we can approximate the right-hand side of equation (9) to an arbitrary degree as

$$\begin{aligned} \rho(x_{t+1}) &= k \textstyle\int_{TG} p(x_{t+1}|x_t) p(y_t|x_t) \rho(x_t) d\mu(x_t) \\ &\simeq \frac{k}{N} \sum_{s_t^i \sim \rho(x_t)} p(x_{t+1}|x_t = s_t^i) p(y_t|x_t = s_t^i)^{\cdot} \end{aligned} \tag{19}$$

The second term in the sum, once a measurement is available, is just a constant vector, which we call w_t^i

$$w_t^i \doteq p(y_t|x_t = s_t^i) \tag{20}$$

so that one can write

$$\rho(x_{t+1}) = \lim_{N\to\infty} \frac{k}{N} \sum_{s_t^i \sim \rho(x_t)} p(x_{t+1}|x_t = s_t^i) w_t^i. \tag{21}$$

The problem now remains of devising an algorithm to obtain a (realization of a) fair sample from $\rho(x_{t+1})$, $s_{t+1}^1, \dots, s_{t+1}^N$, given a (realization of a) fair sample $s_t^1, \dots, s_t^N$ from $\rho(x_t)$.

Remark 3 *Note that in (19) the right hand side is not a density for finite N, i.e. it is not necessarily normalized since*

$$\frac{k}{N} \sum_{s_t^i \sim \rho(x_t)} p(y_t|x_t = s_t^i) \simeq k p(y_t|Y_{t-1}) = 1$$

where equality holds only asymptotically. Note, however, that this is not a source of difficulty since normalization constants are easily accounted for.

2.2 Acceptance sampling

The algorithm we describe is a generalization to Lie groups of a sampling scheme for linear state-spaces that has been proposed independently by several authors in different disciplines. In Computer Vision, Blake and Isard call the algorithm “Condensation” [1]. Slight variations can be found in the Statistics literature under the name of “Sequential Importance Sampling” techniques [10].

a) for $j = 1 \dots N$, choose with replacement[1] s_t^j with probability w_t^i among $\{s_t^1, \dots, s_t^N\}$,

b) for each j draw *one* sample from $p(x_{t+1}|x_t = s_t^j)$; call it s_{t+1}^j

c) for $j = 1 \dots N$ set $w_{t+1}^j = p(y_{t+1}|x_{t+1} = s_{t+1}^j)$.

The following claim guarantees that this procedure results asymptotically in a (realization of a) fair sample of length N from $\rho(x_{t+1})$.

Claim 1 *Provided that $\{s_t^i\}_{i=1\dots N}$ is a realization of a fair sample from $\rho(x_t)$, and $\{w_t^i\}$ defined as in (20), the algorithm above generates, as $N \to \infty$, a realization $\{s_{t+1}^j\}$ of a fair sample from $\rho(x_{t+1})$.*

Proof: It suffices to show that:

$$P\{s_{t+1}^i \in \Delta\} = \int_\Delta p(x_{t+1}|Y_t)\, d\mu(x_{t+1}) \tag{22}$$

for any $\mu-$measurable set $\Delta \subset TG$. Based on step a) the following equation holds:

$$P\{s_{t+1}^i \in \Delta\} = \frac{\sum_i \int_\Delta p(x_{t+1}|x_t = s_t^i)\, w_t^i\, d\mu(x_{t+1})}{\sum_i p(y_t|x_t = s_t^i)} \tag{23}$$

If now we take into account that $\{s_t^i\}$ is a “fair sample” from $p(x_t|Y_{t-1})$, we can rewrite (asymptotically) the previous equation as:

$$P\{s_{t+1}^i \in \Delta\} = \frac{\int_{TG} \int_\Delta \ell(x_{t+1}, x_t, Y_t)\, d\mu(x_t) d\mu(x_{t+1})}{\int_{TG} p(y_t|x_t) p(x_t|Y_{t-1})\, d\mu(x_t)}, \tag{24}$$

where we recall that $\ell(x_{t+1}, x_t, Y_t) = p(x_{t+1}|x_t)\, p(y_t|x_t)\, p(x_t|Y_{t-1})$.

Making use of Bayes rule, one sees that

$$\frac{\int_{TG} p(x_{t+1}|x_t)\, p(y_t|x_t)\, p(x_t|Y_{t-1})\, d\mu(x_t)}{\int_{TG} p(y_t|x_t) p(x_t|Y_{t-1})\, d\mu(x_t)} = p(x_{t+1}|Y_t) \tag{25}$$

that finally yields

$$P\{s_{t+1}^i \in \Delta\} = \int_\Delta p(x_{t+1}|Y_t)\, d\mu(x_{t+1}). \tag{26}$$

The fact that $\{s_{t+1}^i\}$ is a realization of a fair sample follows from the fact that random draws for different values of i are independent. On this fact see also [4] ■

We call this algorithm “acceptance sampling”. It has the advantage of requiring a constant number of operations in order to obtain a constant number of samples at each step of the update. It does, however, suffer from the phenomenon of “sample impoverishment”, which in the statistical literature is treated with a variety of remedies, including the “Rao-Blackwellization” procedure [10].

2.3 Other sampling strategies

Geweke and Tanizaki [7] describe a Monte Carlo algorithm based on “Rejection Sampling” for the filtering density $\rho(x_t) \doteq p(x_t|Y_t)$,

$$\rho(x_{t+1}) = kp(y_{t+1}|x_{t+1}) \int p(x_{t+1}|x_t) p(x_t|Y_t) d\mu(x_t) \tag{27}$$

which is based upon drawing samples from $p(x_{t+1}|x_t)$ and accepting them with probability proportional to $p(y_{t+1}|x_{t+1})$. This type of rejection techniques have the disadvantage that the time required to obtain a sample of size N changes at each step, and in certain cases this can be a problem if the estimates are to be used to perform a control action. Along these lines, see also Logothetis and Doucet [11].

The Statistics literature provides us with a wealth of heuristics to improve the performance on sampling algorithms of the type just described. Just to cite one example, [10] describes procedures such as “re-weighting”, “re-sampling”, “reallocation” and “Rao-Blackwellization”, “Rejection Control” and “Partial Rejection Control” that are believed to enhance the performance of Sequential Importance Sampling. Along these lines see also [5]

3 Experiments

We have implemented the algorithm described in section 2.2 for the model (2) for the cases of $G = SO(3), SE(3)$. Here we report some results of simulation studies for the simplest case of $SO(3)$. We have generated a random cloud of points in a volume of $1m^3$, rotating about an axis with constant velocity equal to $5^o/s$, and measured their projection on a coordinate plane spoiled by white zero-mean additive Gaussian noise of 30% std. The trajectory of a few measurements of shown in figure 1.

[1] Choosing a sample among $s^1, \dots, s^N$ with probability w^i is achieved by constructing the cumulative distribution F_N (a staircase function with steps at each i with height w^i), choosing a point r at random uniformly between 0 and $\sum w^i$, and selecting the largest sample s^j such that $F_N(j) \leq r$.

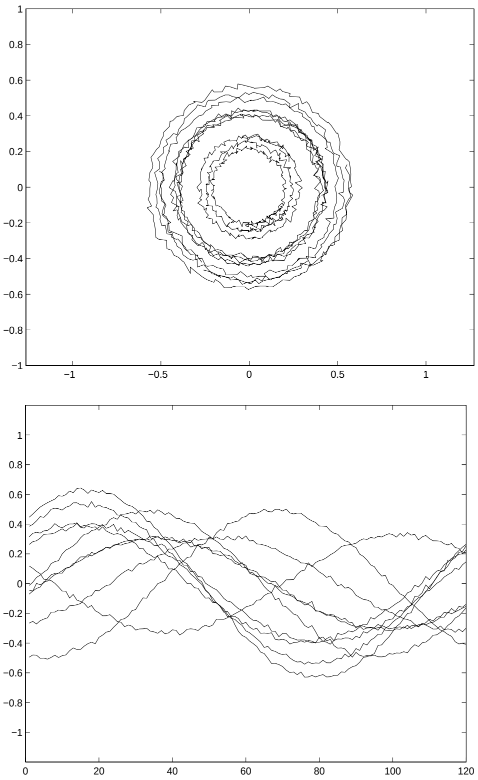

Figure 1: *(Top) top view of the rotating cloud, with spatial noise. (Bottom) measured projections for 10 points: abscissa is time.*

We have then initialized the filter with a density that is concentrated at the identity of $SO(3)$ for the trajectory, with a wide Gaussian density around 0 for the velocity. We have chosen to work with a number of samples ranging from a few hundreds to a few thousands. The outcome of the filter is difficult to visualize, since it consists of the conditional density at each instant. In figure 2 we show the conditional mean computed from the estimated conditional density and represented in canonical exponential coordinates. In figure 3 we show the conditional mean and covariance of the trajectory, for a sample of size 100, displayed as the 9 elements of an embedded representation of $SO(3)$.

In figure 4 we show the average error covariance after steady state for a number of samples ranging from 100 to 1000.

4 Generalizations

This paper extends the classical Sequential Importance Sampling algorithm to distributions of first-order random walks on matrix Lie groups. Extensions to higher-order random walks is straightforward, since the lifting maintains the linearity of the Brownian motion. Extensions to more general classes of distributions on the group, for instance determined by more complex dynamics, is yet an open problem, and of considerable practical importance in engineering applications. For instance, one may wish to devise algorithms for tracking the conditional density of $\{x_t\} \in G$ specified by the following model

$$\begin{cases} x_{t+1} \sim p(x_t, v_t) & \quad x_0 \sim p_0 \\ y_t \sim q(x_t, w_t) \end{cases} \tag{28}$$

where p is a density associated with a distribution on G and q is a density on the measurement space. $\{v_t\}$ and $\{w_t\}$ represent input and measurement noises. For models of this sort, a theory of observability is lacking, and so are quantitative results on the small-sample behavior of Monte Carlo algorithms.

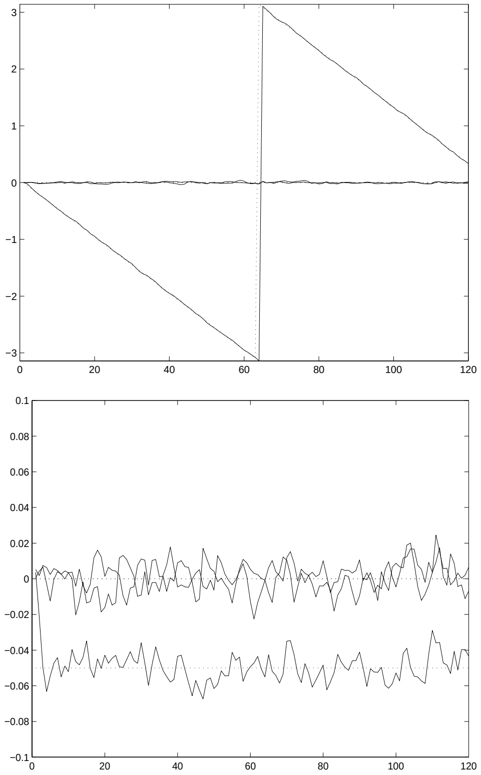

Figure 2: *(Top) Estimated conditional mean for exponential coordinates of pose (solid line) against ground truth (dotted line). (Bottom) Estimated conditional mean of velocity (dotted line) against ground truth (dotted line). Abscissa is time, ordinate is the value of the state*

Acknowledgments
This research was supported by NSF grant IIS-9876145 and ARO grant DAAD19-99-1-0139. The authors wish to thank J. O'Sullivan for his enlightening comments and A. Blake for numerous discussions on the topic and for his inspiring work on CONDENSATION.

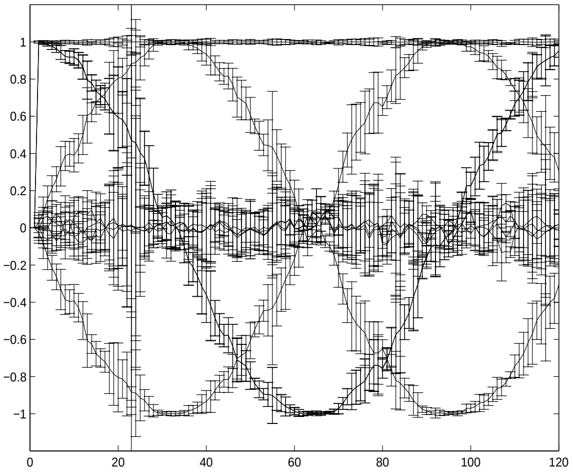

Figure 3: *Estimated state, embedded representation: estimated conditional mean and covariance for 100 points. Abscissa is time, ordinate is the estimated state.*

References

[1] A. Blake and M. Isard. *Active contours.* Springer Verlag, 1998.

[2] D. Brigo, B. Hanzon, and F. LeGland. A differential geometric approach to nonlinear filtering: the projection filter. *IEEE Trans. on Automatic Control*, 68:181–188, 1998.

[3] E. Celledoni. Lie group methods and the approximation of the exponential map, *Foundations of Computational Mathematics Workshop on Geometric Integration and Computation on Manifolds*, July 26-28, 1999

[4] J. MacCormick. Probabilistic modelling and stochastic algorithms for visual localisation and tracking. *PhD Thesis, University of Oxford*, 2000.

[5] J. MacCormick and M. Isard Partitioned Sampling, Articulated Objects, and Interface-Quality Hand Tracking. *Proc. of sixth European Conference on Computer Vision*, pp. 3–19, Dublin, Ireland, June 2000.

[6] A. Edelman, T. Arias, and S. Smith. The geometry of algorithms with orthogonality constraints. *SIAM J. on Matrix Analysis and Applications*, 2000.

[7] J. Geweke and H. Tanizaki. On markov chain monte carlo methods for nonlinear and non gaussian state-space models. *commun. Stat. Simul. C*, 28:867–894, 1999.

[8] H. Kunita. *Stochastic differential equations on manifolds.* Cambridge University Press, 1991.

[9] S. LaValle, P. Finn, L. Kavraki, and J-C Latombe. A randomized kinematics-based approach to pharmacophore-constrained conformational search and database screening. Dept. of computer science, Stanford Universityd, 1999.

[10] J. Liu, R. Chen, and T. Logvinenko. A theoretical framework for sequential importance sampling and resampling. Technical report, Stanford University, Department of Statistics, January 2000.

[11] A. Logothetis and A. Doucet. Markov chain monte carlo methods for tracking a maneuvering target in clutter. In *IEEE Conf. on Decision and Control*, 1999.

[12] A. Marshall. The use of multi-stage sampling schemes in monte carlo computations. *Symposium on Monte Carlo Methods*, pages 123–140, 1956.

[13] S. K. Mitter. Existence and non-existence of finite-dimensional filters. *Mathematical theory of optimization*, pages 1–10, 1981.

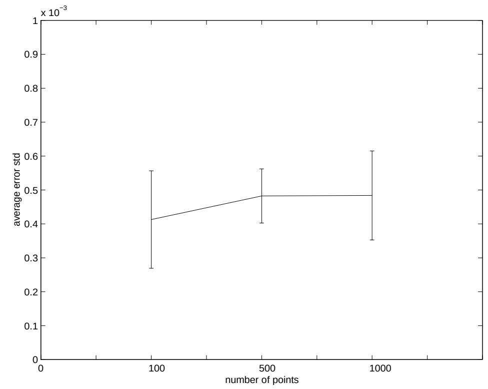

Figure 4: *Mean and std of the variance of the estimation error over 10 trials for 100, 500 and 1000 points. There is no significant change in the performance of the filter in increasing the number of points beyond 100, for this specific example.*

[14] B. Oksendal. *Stochastic Differential Equations.* Springer Verlag, 1995.

[15] S. Soatto, R. Frezza, and P. Perona. Motion estimation via dynamic vision. *IEEE Transactions on Automatic Control*, 41(3):393–414, March 1996.

[16] H. Sorenson. On the development of practical nonlinear filters. pages 63–79.

[17] H. Sorenson and D. Alspach. Recursive bayesian estimation using gaussian sums. *Automatica*, 7:465–479, 1971.

[18] A. Srivastava. A bayesian approach to geometric subspace estimation. *IEEE Trans. on Signal Processing*, 2000.