

PARTICLE FILTERING APPLIED TO ROBUST MULTIVARIATE LIKELIHOOD OPTIMIZATION IN THE ABSENCE OF A CLOSED-FORM SOLUTION

*Pau Closas, Juan A. Fernández–Rubio**

Universitat Politècnica de Catalunya (UPC)
Dept. of Signal Theory and Communications
Campus Nord, Jordi Girona 1–3
08036 Barcelona, Spain.

Carles Fernández Prades

Centre Tecnològic
de Telecomunicacions de Catalunya (CTTC)
Av. Canal Olímpic s/n
08860 Castelldefels, Spain.

ABSTRACT

Sequential Monte Carlo (SMC) methods are studied to deal with multivariate optimization problems arising from Maximum Likelihood (ML) estimation approaches. We compare results to those obtained by other methods, showing faster convergence and improved robustness when local optimums are present in the cost function to optimize. This paper presents a SMC method to obtain ML estimates in general multivariate state-spaces where a closed-form solution cannot be obtained, reporting computer simulation results for a particular application.

1. INTRODUCTION

The need to estimate parameters of signals with known structure arises in many engineering problems. The use of pilot signals for synchronization in third generation wireless communication systems [1] or the propagation delay estimation in navigation systems [2] are examples of applications where the signal waveform is known by the receiver, and the information is enclosed in certain parameters (time delays, Doppler shifts) of such waveform. A large variety of solutions proposed in the literature are based in the Maximum Likelihood (ML) approach because it drives to asymptotically (in high signal-to-noise ratios) unbiased and efficient estimators. An extensive amount of work has been devoted on the case where the signals of interest are contaminated by white Gaussian noise, or on the case where the additive noise is nonwhite but has a known correlation and thus can be reduced to the former case by means of prewhitening techniques.

Focusing in time synchronization, which plays a central role in digital communications and radio–location systems, the Delay Locked Loop (DLL) and its variants are examples of hardware implementations of the ML approach. They are known to be optimum in the sense that they attain the Cramér–Rao Bound (CRB) when the signal model holds, that is, in case where only the desired signal and additive white Gaussian noise are present. However, the presence of multipath causes a bias in the time estimation delivered by the DLL [3], constituting a signal processing challenge that has deserved the attention of many researchers in the recent times. Although great achievements in multipath mitigation have been obtained for single–antenna receivers (see [4] for a review), they have inherent drawbacks like the fact that the estimator suffers from infinite variance when the time difference between the direct signal and the echo approach to zero. A promising approach which overcomes such draw-

back and effectively mitigates multipath is the use of antenna arrays and the exploitation of the spatial diversity that they provide, assuming that the additive noise has an unknown covariance matrix. This statistical assumption provides the ML estimator with a noticeable robustness against multipath [5].

Unfortunately, there is not a closed-form solution for this estimator but a cost function to be optimized. In general, the resulting ML cost function can be a multi-dimensional, highly non-linear function, which makes not feasible the use of gradient-based methods, such as the Steepest Descent or the Newton-Raphson algorithms which need a proper initialization point to converge to the optimal value in a finite time, i.e., they need to be initialized in the neighborhood of the global optimum. Thus, local minimums can cause gradient-based algorithms to fail. Alternative methods must be studied to deal with the optimization in a more suitable way. To this aim, SMC methods [6] have been investigated.

2. TECHNICAL BACKGROUND: ML SYNCHRONIZATION IN ANTENNA ARRAYS WITH UNKNOWN COVARIANCE MATRIX

We consider the following baseband signal model:

$$\mathbf{x}_k = \mathbf{H}\mathbf{s}_k(\boldsymbol{\tau}) + \mathbf{n}_k \quad (1)$$

where $\mathbf{x}_k \in \mathbb{C}^{N \times 1}$ is the measurement vector at time k and N is the number of antennas, $\mathbf{H} \in \mathbb{C}^{N \times M}$ is the complex channel matrix and M is the number of Signals Of Interest (SOIs) with known waveform, $\mathbf{s}_k(\boldsymbol{\tau}) \in \mathbb{C}^{M \times 1}$ is the vector of M SOIs parameterized by a vector $\boldsymbol{\tau} \in \mathbb{R}^{M \times 1}$ of time delays (which is the vector parameter to be estimated, and it is considered constant in the observation window) and $\mathbf{n}_k \in \mathbb{C}^{N \times 1}$ is a term, usually referred to as measurement noise, that gathers noise and all the rest of unmodelled signals, as multipath or interferences. This term is considered to be Gaussian, with zero mean but with an unknown and arbitrary correlation matrix, an assumption that provides more flexibility to the system model than the more restrictive white noise assumption (which implies a diagonal covariance matrix). The probability density function (PDF) of a complex multivariate Gaussian vector \mathbf{x}_k , as expressed in (1), and considering a noise with arbitrary covariance matrix \mathbf{Q} , can be expressed as:

$$p(\mathbf{x}_k | \boldsymbol{\tau}) = \frac{\exp[-(\mathbf{x}_k - \mathbf{H}\mathbf{s}_k(\boldsymbol{\tau}))^H \mathbf{Q}^{-1}(\mathbf{x}_k - \mathbf{H}\mathbf{s}_k(\boldsymbol{\tau}))]}{\pi^N \det(\mathbf{Q})} \quad (2)$$

Considering a system with a memory of K snapshots, we will denote the block of measurement samples as $\mathbf{X}_k = \{\mathbf{x}_i, i = k - K + 1, \dots, k\}$.

*This work was financed by the Spanish/Catalan Science and Technology Commissions and FEDER funds from the European Commission: TIC2003-05482, TEC2004-04526 and 2001SGR-00268.

The basis function matrix \mathbf{S}_k is defined equally. Thus, extending the model to a collection of K snapshots, and using the following cross-correlation estimation matrix definitions:

$$\begin{aligned} \hat{\mathbf{R}}_{\mathbf{X}\mathbf{X}_k} &= \frac{1}{K} \mathbf{X}_k \mathbf{X}_k^H & \hat{\mathbf{R}}_{\mathbf{X}\mathbf{S}_k}(\boldsymbol{\tau}) &= \frac{1}{K} \mathbf{X}_k \mathbf{S}_k^H(\boldsymbol{\tau}) \\ \hat{\mathbf{R}}_{\mathbf{S}\mathbf{X}_k}(\boldsymbol{\tau}) &= \hat{\mathbf{R}}_{\mathbf{X}\mathbf{S}_k}^H(\boldsymbol{\tau}) & \hat{\mathbf{R}}_{\mathbf{S}\mathbf{S}_k}(\boldsymbol{\tau}) &= \frac{1}{K} \mathbf{S}_k(\boldsymbol{\tau}) \mathbf{S}_k^H(\boldsymbol{\tau}) \end{aligned} \quad (3)$$

the ML estimator of the time delays vector can be computed as (the full derivation can be found in [7]):

$$\begin{aligned} \hat{\boldsymbol{\tau}}_{MLE} &= \arg \min_{\boldsymbol{\tau}} \Lambda_{ML}(\boldsymbol{\tau}) = \arg \min_{\boldsymbol{\tau}} \ln \left(\det \left(\hat{\mathbf{W}} \right) \right) \\ &= \arg \min_{\boldsymbol{\tau}} \ln \left(\det \left(\hat{\mathbf{R}}_{\mathbf{X}\mathbf{X}_k} - \hat{\mathbf{R}}_{\mathbf{X}\mathbf{S}_k} \hat{\mathbf{R}}_{\mathbf{S}\mathbf{S}_k}^{-1} \hat{\mathbf{R}}_{\mathbf{S}\mathbf{X}_k}^H \right) \right) \end{aligned} \quad (4)$$

which is a nonlinear function without a closed-form solution. This multivariate problem can be solved by brute force, evaluating $\Lambda_{ML}(\boldsymbol{\tau})$ on a coarse grid to locate roughly the global minimum, and then resorting to a finer grid or some gradient-based iterative algorithm, but this solution can be prohibitively time-consuming. The main objective of this paper is the minimization of $\Lambda_{ML}(\boldsymbol{\tau})$ by means of an approximation of the posterior distribution and performing statistical estimates of the unknown synchronization vector $\boldsymbol{\tau}$.

3. SEQUENTIAL MONTE CARLO APPROXIMATION OF THE LIKELIHOOD

Sequential Monte Carlo (SMC) methods are a set of simulation based methods which adopt the discrete state-space approach to deal with the non-linear filtering problem, by means of recursively computing estimates of states given all available measurements \mathbf{X}_k up to time index k . As the parameters of interest are assumed to be piecewise constant during the observation interval, the state equation is $\boldsymbol{\tau}_k = \boldsymbol{\tau}_{k-1}$. However, it is desirable to introduce some perturbation in the state equation model, allowing the states to evolve as a discrete-time stochastic model:

$$\boldsymbol{\tau}_k = \boldsymbol{\tau}_{k-1} + \boldsymbol{\omega}_{k-1} \quad (5)$$

where $\boldsymbol{\omega}$, evaluated at time $k-1$, is the process noise considered white and with zero mean. The relation between measurements and states is modeled by equation (1). Both process noise $\boldsymbol{\omega}$ and measurement noise \mathbf{n} are assumed to be statistically independent. The initial *a priori* PDF of the state vector is assumed known, $p(\boldsymbol{\tau}_0)$.

The Optimal Bayesian filtering problem consists in computing filtered estimates of $\boldsymbol{\tau}_k$ taking into account all available measurement up to time. The solution is to recursively obtain the *a posteriori* PDF of states at time k given all available measurements, $p(\boldsymbol{\tau}_k | \mathbf{X}_k)$. It is shown in the literature [6, 8] that the desired density can be computed recursively in two stages: prediction and update. However, in a reduced set of cases a closed-form solution can be found and sub-optimal algorithms must be explored.

Particle Filter methods use the Sequential Importance Sampling (SIS) concept to characterize the posterior density dealing with nonlinearities/non-gaussianities of the model. Basically it involves the approximation of the posterior by a set of N_s random samples taken from an *importance density function*, $\boldsymbol{\tau}^i \sim \pi(\boldsymbol{\tau} | \mathbf{X})$, with associated importance weights w^i . The choosing of $\pi(\cdot)$ is a critical issue in any particle filter design. The posterior approximation is then

$$\hat{p}(\boldsymbol{\tau}_k | \mathbf{X}_k) = \sum_{i=1}^{N_s} w_k^i \delta(\boldsymbol{\tau}_k - \boldsymbol{\tau}_k^i) \quad (6)$$

being $\delta(\cdot)$ the Dirac's delta function. This approximation converges almost surely to the true posterior as $N_s \rightarrow \infty$ under weak assumptions according to the SLLN [6]. These assumptions hold if the support of the chosen importance density ($\bar{\pi}$) include the support of the posterior PDF (\bar{p}),

$$\begin{aligned} \bar{\pi} &= \left\{ \boldsymbol{\tau}_k \in \mathbb{R}^M \mid \pi(\boldsymbol{\tau}_k | \mathbf{X}_k) > 0 \right\} \\ \bar{p} &= \left\{ \boldsymbol{\tau}_k \in \mathbb{R}^M \mid p(\boldsymbol{\tau}_k | \mathbf{X}_k) > 0 \right\} \\ &\text{and } \bar{p} \subseteq \bar{\pi} \end{aligned} \quad (7)$$

The resampling step consists in replacing particles with low importance weights and multiply those with high importance weights. This is the so called Sampling Importance Resampling (SIR) step which, although constitutes the bottle neck in any parallel implementation of Particle Filters, prevents from the degeneracy phenomenon, which states that variance of importance weights can only increase over time. After a certain number of recursive steps, all but one particle will have negligible normalized weights. The general recursive solution obtained is

$$\begin{aligned} \boldsymbol{\tau}_k^i &\sim \pi(\boldsymbol{\tau}_k | \boldsymbol{\tau}_{k-1}^i, \mathbf{X}_k) \\ w_k^i &\propto w_{k-1}^i \frac{p(\mathbf{X}_k | \boldsymbol{\tau}_k^i) p(\boldsymbol{\tau}_k^i | \boldsymbol{\tau}_{k-1}^i)}{\pi(\boldsymbol{\tau}_k^i | \boldsymbol{\tau}_{k-1}^i, \mathbf{X}_k)} \end{aligned} \quad (8)$$

One of the key points is the choose of a *good* importance density function $\pi(\cdot)$. This is to propose an importance density function close to the optimal, which is the posterior PDF. In our case, by setting it to the transitional prior defined by state equation (5), we obtain

$$\begin{aligned} \boldsymbol{\tau}_k^i &\sim \pi(\boldsymbol{\tau}_k | \boldsymbol{\tau}_{k-1}^i, \mathbf{X}_k) = p(\boldsymbol{\tau}_k | \boldsymbol{\tau}_{k-1}^i) \\ w_k^i &\propto p(\mathbf{X}_k | \boldsymbol{\tau}_k^i) \end{aligned} \quad (9)$$

which states that the computed weights will be proportional to the likelihood function, which is the target function. This is of great interest in problems where a closed-form solution cannot be obtained and a cost function has to be minimized.

As we deal with a sequential approximation to the ML solution, it is assumed that no prior information is available regarding the statistical evolution of states so that a non informative prior is used to characterize the evolution of states, that is the importance density function. A joint normal PDF is assumed for the state vector, whose covariance parameter is used to tune the statistical distribution as the number of iterations N_t increases. Then, for a given initial guess of states, N_s trial points are generated taking into account a normal PDF with mean the current estimate and covariance function of the initial ambiguity. A state estimate is given from the particle that optimizes the ML cost function from the set generated. From particle filtering theory, if $\hat{\boldsymbol{\tau}}_k$ denotes an estimation of the states at time index k , then the covariance matrix estimate is given by

$$\mathbf{C}_k^\tau \approx \sum_{i=1}^{N_s} w_k^i \left(\boldsymbol{\tau}_k^i - \hat{\boldsymbol{\tau}}_k \right) \left(\boldsymbol{\tau}_k^i - \hat{\boldsymbol{\tau}}_k \right)^T \quad (10)$$

which will be propagated to the next algorithm iteration jointly with the particle estimate. This estimation procedure converges to the minimum of the ML cost function adapting the covariance matrix at each iteration, where the algorithm updates the estimate if and only if $p(\mathbf{X}_k | \hat{\boldsymbol{\tau}}_k) < p(\mathbf{X}_{k-1} | \hat{\boldsymbol{\tau}}_{k-1})$.

4. DESCRIPTION OF ALGORITHMS

In order to assess the performance of the proposed approach, we have implemented and simulated the following algorithms:

4.1. Proposed SMC based algorithm

In order to optimize the multivariate likelihood function expressed in equation (4), which has proven to be robust against multipath [5, 7], we propose the use of Particle Filtering. Actually, this algorithm is applicable to any other multivariate optimization problem, regardless if the cost function is linear or not. A pseudocode description of the proposed optimization method is sketched in Table 1, where $\mathcal{N}(\mu, \sigma^2)$ denotes the normal probability density function of mean μ and variance σ^2 and, for a given input vector \mathbf{w}_k , the function $[V, I] = \min\{\mathbf{w}_k\}$ returns the minimum value (V) and its index (I). Notice that the algorithm only updates the particles for next iteration if the lowest weight from the set of points generated is lower than the value obtained from the previous iteration.

General SMC based optimization algorithm
<ul style="list-style-type: none"> • Initial State Estimate: $\hat{\boldsymbol{\tau}}_0$ • Initial State Uncertainty: \mathbf{C}_0^τ • Calculate $V_{past} = p(\mathbf{X}_k \hat{\boldsymbol{\tau}}_0)$ • FOR $k=1:N_t$ <ul style="list-style-type: none"> - FOR $i=1:N_s$ <ul style="list-style-type: none"> Generate $\boldsymbol{\tau}_k^i \sim \mathcal{N}(\hat{\boldsymbol{\tau}}_{k-1}^i, \mathbf{C}_{k-1}^\tau)$ Calculate $\tilde{w}_k^i = p(\mathbf{X}_k \boldsymbol{\tau}_k^i)$ - END FOR - Normalize $w_k^i = \tilde{w}_k^i / \sum_{j=1}^{N_s} \tilde{w}_k^j$ - $[V, I] = \min\{\mathbf{w}_k\}$ - IF $V_{past} \leq V$ <ul style="list-style-type: none"> $\hat{\boldsymbol{\tau}}_k = \hat{\boldsymbol{\tau}}_{k-1}$ $\mathbf{C}_k^\tau = \mathbf{C}_{k-1}^\tau$ - ELSE <ul style="list-style-type: none"> $\hat{\boldsymbol{\tau}}_k = \boldsymbol{\tau}_k^I$ Update $V_{past} = V$ Calculate \mathbf{C}_k^τ using eq. (10) - END IF • END FOR • Final estimates: $\hat{\boldsymbol{\tau}}_{SMC} = \hat{\boldsymbol{\tau}}_{N_t}$

Table 1. Proposed SMC based algorithm

4.2. Markov chain Monte Carlo based algorithm

Markov chain Monte Carlo (MCMC) methods [9] are based on a sampling from the target distribution using a Markov chain, such as

$$\boldsymbol{\tau}_k = \alpha \boldsymbol{\tau}_{k-1} + \boldsymbol{\nu}_k \quad (11)$$

where $|\alpha_u| < 1$ for $u = 1 \dots M$ and the elements of $\boldsymbol{\nu}_k$ are distributed as $\mathcal{N}(0, \sigma^2)$. We have implemented a Metropolis–Hastings algorithm, which pseudocode is sketched in Table 2 where

$$\beta(\boldsymbol{\tau}' | \boldsymbol{\tau}) = \min \left(1, \frac{\Lambda_{ML}(\boldsymbol{\tau}') p(\boldsymbol{\tau} | \boldsymbol{\tau}')}{\Lambda_{ML}(\boldsymbol{\tau}) p(\boldsymbol{\tau}' | \boldsymbol{\tau})} \right). \quad (12)$$

Metropolis-Hastings (MH) algorithm
<ul style="list-style-type: none"> • Initial State Estimate: $\hat{\boldsymbol{\tau}}_0$ • FOR $k=1:N_t$ <ul style="list-style-type: none"> - Generate a candidate $\boldsymbol{\tau}' \sim p(\boldsymbol{\tau}_k \boldsymbol{\tau}_{k-1})$. - Generate $\xi \sim \mathcal{U}[0, 1]$. - IF $\xi \leq \beta(\boldsymbol{\tau}' \boldsymbol{\tau}_{k-1})$ <ul style="list-style-type: none"> $\boldsymbol{\tau}_k = \boldsymbol{\tau}'$ - ELSE <ul style="list-style-type: none"> $\boldsymbol{\tau}_k = \boldsymbol{\tau}_{k-1}$ - END IF • END FOR • Final estimates: $\hat{\boldsymbol{\tau}}_{MH} = \hat{\boldsymbol{\tau}}_{N_t}$

Table 2. MCMC based algorithm

4.3. Steepest descent algorithm

The steepest descent is an algorithm for finding a local minimum of a function in an iterative way. It does not ensure finding the global minimum, and therefore it must be initialized within a convergence region by means of an acquisition (coarse estimation) method. Once it is initialized in a suitable synchronization vector, $\boldsymbol{\tau}_0$, the algorithm moves towards the direction of the local downhill gradient:

$$\boldsymbol{\tau}_{k+1} = \boldsymbol{\tau}_k - \lambda_k \nabla_{\boldsymbol{\tau}_k} \Lambda_{ML}(\boldsymbol{\tau}_k) \quad (13)$$

When applied to a set of K snapshots of signal model (1), and thus using the likelihood function obtained in (4), the elements of the gradient vector $\nabla_{\boldsymbol{\tau}_k} \Lambda_{ML}(\boldsymbol{\tau}_k)$ are:

$$\frac{\partial \Lambda_{ML}}{\partial \boldsymbol{\tau}_k^{(u)}} = -\frac{1}{K} \text{Tr} \left\{ \hat{\mathbf{W}}_k^{-1} \mathbf{X}_k \frac{\partial \mathbf{P}_{\mathbf{S}_k^H}}{\partial \boldsymbol{\tau}_k^{(u)}} \mathbf{X}_k^H \right\}. \quad (14)$$

being $\mathbf{P}_{\mathbf{S}_k^H} = \mathbf{S}_k^H (\mathbf{S}_k \mathbf{S}_k^H)^{-1} \mathbf{S}_k$ the projection matrix over the subspace spanned by the columns of \mathbf{S}_k . Its derivative with respect to the elements of vector $\boldsymbol{\tau}$ can be computed as

$$\frac{\partial \mathbf{P}_{\mathbf{S}_k^H}}{\partial \boldsymbol{\tau}^{(u)}} = \mathbf{P}_{\mathbf{S}_k^H}^\perp \frac{\partial \mathbf{S}_k^H}{\partial \boldsymbol{\tau}^{(u)}} (\mathbf{S}_k^H)^\dagger + \left(\mathbf{P}_{\mathbf{S}_k^H}^\perp \frac{\partial \mathbf{S}_k^H}{\partial \boldsymbol{\tau}^{(u)}} (\mathbf{S}_k^H)^\dagger \right)^H \quad (15)$$

where $\mathbf{P}_{\mathbf{S}_k^H}^\perp = \mathbf{I} - \mathbf{P}_{\mathbf{S}_k^H}$ and the superindex $(\cdot)^\dagger$ stands for the pseudo-inverse. The step size vector λ_k has been set heuristically. The algorithm is outlined in Table 3.

Steepest Descent (SD) algorithm
<ul style="list-style-type: none"> • Initial State Estimate: $\hat{\boldsymbol{\tau}}_0$ • FOR $k=1:N_t$ <ul style="list-style-type: none"> - Compute $\nabla_{\boldsymbol{\tau}_k} \Lambda_{ML}(\boldsymbol{\tau}_{k-1})$ by means of equations (15) and (14). - Compute $\boldsymbol{\tau}_k$ as in (13). • END FOR • Final estimates: $\hat{\boldsymbol{\tau}}_{SD} = \hat{\boldsymbol{\tau}}_{N_t}$

Table 3. Gradient based algorithm

4.4. Space–Alternating Generalized Expectation–Maximization algorithm

The Space–Alternating Generalized EM algorithm (SAGE) [10] is a general method for solving ML estimators that are too complex

for direct solution, and allows a complex multivariate optimization problem to be simplified into a number of decoupled optimization problems. In short, each iteration re-estimates not the whole vector τ but only a subset of the elements, keeping fixed the estimates of the other elements. The implementation is outlined in table 4, where

$$\tilde{\tau}_k^i = \left[\hat{\tau}_k^{(1)}, \dots, \hat{\tau}_k^{(i-1)}, \hat{\tau}_k^{(i+1)}, \dots, \hat{\tau}_{k-1}^{(M)} \right]^T \quad (16)$$

is an $(M - 1) \times 1$ vector where all the model parameters are fixed to the latest estimation except the i th element of τ_k on which the optimization is performed.

SAGE algorithm	
•	Initial State Estimate: $\hat{\tau}_0$
•	FOR k=1: N_t
-	FOR i=1:M
	E-step: Compute $\Lambda_{ML}(\tau_k^{(i)}; \tilde{\tau}_k^i)$
	M-step: $\hat{\tau}_k^{(i)} = \arg \min_{\tau} \Lambda_{ML}(\tau_k^{(i)}; \tilde{\tau}_k^i)$
-	END FOR
•	END FOR
•	Final estimates: $\hat{\tau}_{SAGE} = \hat{\tau}_{N_t}$

Table 4. Sequential estimation algorithm

5. SIMULATION RESULTS

We have studied the performance of algorithms described in section 4. The considered signal structure has been the L1 C/A GPS signal [2], filtered in the bandpass at 2 MHz, downshifted to an intermediate frequency of $f_{IF} = 4.309$ MHz, sampled at $f_s = 5.714$ MHz (applying incommensurable sampling and exploiting the intentional aliasing, usual techniques in GPS receivers [7]), and taking 1 ms of data. 7 satellites and a secondary path of one of them have been simulated. Taking as a reference $\hat{\tau}_{opt}$ the value obtained by the brute-force, grid search method applied to equation (4), we compare the convergence rate of the four described algorithms. As showed in figure 1, the SMC based algorithm achieves the fastest convergence rate and the minimum error floor, closely followed by the SAGE algorithm. Whereas the SMC algorithm has non-increasing curves, the SAGE algorithm does not. However, the computational burden of the proposed algorithm increases proportionally to the number of particles used, being of interest for high-precision applications. Notice that both the SD and the MH algorithms have convergence rates much slower than those exhibit by SMC and SAGE algorithms, at the expense of increasing the computational burden.

6. CONCLUSIONS

This paper addresses the problem of multivariate optimization of likelihood functions using Sequential Monte Carlo methods. An algorithm is proposed to deal with general optimization of multivariate, possibly non-linear functions. The algorithm shows high convergence rate compared to other simulated methods, though an increase in the computational burden must be held up. As an application example, the optimization of the ML cost function obtained in a navigation problem has been studied. Several methods are compared in terms of convergence rate, and we have provided the pseudo-code description of all of them.

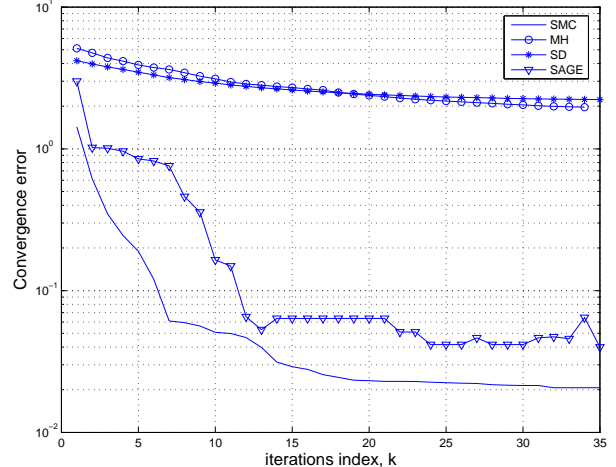


Fig. 1. Convergence of the Sequential Monte Carlo (SMC), Metropolis-Hastings (MH), Steepest Descent (SD) and Space-Alternating Generalized Expectation-Maximization (SAGE) algorithms in the multivariate likelihood optimization.

7. REFERENCES

- [1] E. Dahlman, B. Gudmundson, M. Nilsson, and A. Skold, "UMTS/IMT-2000 based on wideband CDMA," *IEEE Communications Magazine*, vol. 36, no. 9, pp. 70–80, Sept. 1998.
- [2] B.W. Parkinson and J.J. Spilker, Eds., *Global Positioning System: Theory and Applications*, vol. 1, vol. 163 of *Progress in Astronautics and Aeronautics*, American Institute of Aeronautics, Inc., Washington DC, 1996.
- [3] T. Felhauer, "Comparision of EML and DOT discriminator DLL multipath performance in GPS/GLONASS navigation receivers," *Electronics Letters*, vol. 33, no. 3, pp. 179–181, January 1997.
- [4] Lawrence L. Weill, "Multipath mitigation. How good can it get with new signals?," *GPS World*, pp. 106–113, June 2003.
- [5] Gonzalo Seco, Juan A. Fernández-Rubio, and Carles Fernández-Prades, "ML estimator and Hybrid Beamformer for multipath and interference mitigation in GNSS receivers," *IEEE Transactions on Signal Processing*, vol. 53, no. 3, pp. 1194–1208, March 2005.
- [6] A. Doucet, N. de Freitas, and N. Gordon, *Sequential Monte Carlo in Practice*, Springer, 2001.
- [7] C. Fernández-Prades, *Advanced Signal Processing Techniques for Global Navigation Satellite System Receivers*, Ph.D. thesis, Universitat Politècnica de Catalunya, Barcelona, Spain, 2006.
- [8] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter: Particle Filters for tracking applications*, Artech House, Boston, 2004.
- [9] A. Doucet and X. Wang, "Monte Carlo methods for signal processing: A review in the statistical signal processing context," *IEEE Signal Processing Magazine*, vol. 22, no. 6, pp. 152–170, November 2005.
- [10] J. A. Fessler and A. O. Hero, "Space-alternating generalized expectation-maximization algorithm," *IEEE Transactions on Signal Processing*, vol. 42, no. 10, pp. 2664–2677, October 1994.