New Online EM Algorithms for General Hidden Markov Models. Application to the SLAM Problem

Sylvain Le Corff, Gersende Fort, and Eric Moulines*

LTCI, CNRS and TELECOM ParisTech,
46 rue Barrault 75634 Paris Cedex 13, France

Abstract. In this contribution, new online EM algorithms are proposed to perform inference in general hidden Markov models. These algorithms update the parameter at some deterministic times and use Sequential Monte Carlo methods to compute approximations of filtering distributions. Their convergence properties are addressed in [9] and [10]. In this paper, the performance of these algorithms are highlighted in the challenging framework of Simultaneous Localization and Mapping.

Keywords: Online Expectation-Maximization, Hidden Markov models, Statistical inference, SLAM.

1 Introduction

The Expectation Maximization (EM, [6]) algorithm is a versatile tool for maximum-likelihood based parameter estimation in latent data models. However, when processing large data sets or data stream, EM becomes intractable since it requires the whole data set to be available at each iteration of the algorithm.

In this contribution, we are interested in *online-EM* algorithms designed to deal with data which are available sequentially in time. Online-EM algorithms have been recently proposed. [4,14] address the case of independent and identically distributed (i.i.d.) observations. More complex incomplete data models such as Hidden Markov Models (HMM) are of common use to represent time series in many fields such as statistics, information engineering, signal processing, financial econometrics... [3] provides online-EM algorithms for HMM with finite state space. These algorithms have been extended to general HMM by [3,5] in the case of exponential complete-data likelihood, and by [8] for non exponential and general HMM. Hereafter, we will write "exponential HMM" as a shorthand expression for "HMM with exponential complete-data likelihood".

These online-EM algorithms for HMM are iterative algorithms. Each iteration consists in two steps: *(i)* the E step computes the expectation of the complete log-likelihood under the conditional distribution of the hidden states given

* This work is partially supported by the French National Research Agency, under the programs ANR-07-ROBO-0002 and ANR-08-BLAN-0218.

F. Theis et al. (Eds.): LVA/ICA 2012, LNCS 7191, pp. 131–138, 2012.

the observations (available up to the current time) and the current parameter; *(ii)* the M step updates the parameter as a maximum of this mean complete log-likelihood. Unfortunately, the algorithms mentioned above rely on many approximations. For example, the algorithms by [3,5,8] for general HMM combine stochastic approximation methods, Sequential Monte Carlo (SMC) for the approximation of the filtering distributions and an approximation of the recursive mechanism used to compute particle approximations of the filtering distributions. Therefore, it is really difficult to address the consistency of the estimators and to assert the convergence of these EM-based algorithms.

In this contribution, we propose new online-EM algorithms for general (and non necessarily exponential) HMM. The first algorithm, called *Block Online EM* (BOEM), is designed for exponential HMM such that the filtering distributions can be computed explicitly. Examples of such models are finite HMM and linear Gaussian models. The second algorithm is a SMC approximation of BOEM (so called Particle-BOEM or P-BOEM) designed for HMM with intractable E step. For both algorithms, we also propose *averaged* versions which have better convergence rates. All these algorithms are described in Section 2. The convergence of these algorithms (BOEM, P-BOEM and their averaged versions) is out of the scope of this paper: in [9,10], we provide sufficient conditions for these algorithms to converge to the set of the stationary points of the limiting log-likelihood of the observations. The convergence rates are also derived and it is proved that the averaged versions converge at a faster rate.

We provide in Section 3 an application of the P-BOEM algorithm to non-exponential HMM: P-BOEM is used as a new tool to solve the Simultaneous Localization And Mapping (SLAM) problem. We compare our algorithm to the *OnlineEM SLAM* of [8] and to *MarginalSLAM* of [12]. This numerical section highlights the interest of our algorithm to solve the SLAM problem.

2 New Online EM Algorithms for General HMM

The goal is to fit a HMM model on $\mathbb{Y}$-valued observations $\{\mathbf{Y}_t, t \geq 0\}$ sequentially available. We denote by $\{m_\theta(x,x')\mathrm{d}\lambda(x'), \theta \in \Theta\}$ (resp. $\{g_\theta(x,y)\mathrm{d}\nu(y), \theta \in \Theta\}$) the family of transition kernels onto $\mathbb{X}$ of the hidden states (resp. the conditional distribution of the observation given the hidden state). For simplicity, we assume that $\mathbb{X} \subseteq \mathbb{R}^{n_x}$, $\mathbb{Y} \subseteq \mathbb{R}^{n_y}$ and $\Theta \subseteq \mathbb{R}^{n_\theta}$. The initial distribution χ of the hidden state is assumed to be known. We propose algorithms for the computation of a parameter $\theta_\star$ maximizing the limiting normalized log-likelihood of the observations on this class of model indexed by Θ. We consider the case when m_θ, g_θ describes an exponential HMM i.e. there exist $S : \mathbb{X} \times \mathbb{X} \times \mathbb{Y} \to \mathbb{R}^d$, $\psi : \Theta \to \mathbb{R}$ and $\phi : \Theta \to \mathbb{R}^d$ such that

$$\log\{m_\theta(x,x')g_\theta(x',y)\} = \phi(\theta) + \langle S(x,x',y); \psi(\theta)\rangle \ . \tag{1}$$

For $s \in \mathbb{R}^d$, define

$$\bar{\theta}(s) \stackrel{\mathrm{def}}{=} \mathrm{argmax}_{\theta \in \Theta}\ \phi(\theta) + \langle s; \psi(\theta)\rangle \ .$$

Given a set of observations $\mathbf{Y} = \{\mathbf{Y}_1, \cdots, \mathbf{Y}_T\}$, the $(n+1)$-th E step of the *batch EM* algorithm would consist in computing

$$\mathcal{S}_T^{\mathrm{EM}}(\theta_n) \stackrel{\mathrm{def}}{=} \frac{1}{T} \sum_{t=1}^{T} \Phi^0_{\theta_n,t,T}(S, \mathbf{Y}) \ , \tag{2}$$

where $\Phi^0_{\theta,t,T}(S, \mathbf{Y})$ denotes the expectation of the function S under the conditional distribution

$$\begin{aligned}&\Phi^r_{\theta,s,t}(h, \mathbf{y}) \\ &\stackrel{\mathrm{def}}{=} \frac{\int \chi(\mathrm{d}x_r)\{\prod_{i=r}^{t-1} m_\theta(x_i, x_{i+1}) g_\theta(x_{i+1}, \mathbf{y}_{i+1})\} \, h(x_{s-1}, x_s, \mathbf{y}_s) \, \mathrm{d}\lambda(x_{r+1:t})}{\int \chi(\mathrm{d}x_r)\{\prod_{i=r}^{t-1} m_\theta(x_i, x_{i+1}) g_\theta(x_{i+1}, \mathbf{y}_{i+1})\} \, \mathrm{d}\lambda(x_{r+1:t})} \ ;\end{aligned} \tag{3}$$

and the M-step would update the parameter by $\theta_{n+1} = \bar{\theta}(\mathcal{S}_T^{\mathrm{EM}}(\theta_n))$. A natural extension to deal with sequential data is to update the parameter when a new observation is available. Therefore, the T-th update of this *Online EM* is computed from T observations by the iterative formula $\theta_{T+1} = \bar{\theta}(\mathcal{S}_T^{\mathrm{EM}}(\theta_T))$.

The new ideas of our approach is to update the parameter when a block of observations have been (sequentially) processed: more precisely, every time a new observation is available, the conditional expectation of the complete log-likelihood given the observations from the beginning of the block is updated. Due to the exponential assumption (1), such an update only requires an update of the filtering distribution. Then, at some times, the parameter is updated according to the same rule as in the EM algorithm. Let $\{\tau_n, n \geq 0\}$ be positive integers, and set $T_n \stackrel{\mathrm{def}}{=} T_{n-1} + \tau_n = \sum_{i=1}^{n} \tau_i$, $T_0 \stackrel{\mathrm{def}}{=} 0$. τ_n is the length of block n and the parameter will be updated at times T_n.

Block Online EM (BOEM) is an iterative algorithm: given the parameter θ_n updated at time T_n,

block E step compute the BOEM statistic

$$\mathcal{S}^{\mathrm{BOEM}}_{T_n,\tau_{n+1}}(\theta_n) \stackrel{\mathrm{def}}{=} \frac{1}{\tau} \sum_{t=T_n+1}^{T_n+\tau_{n+1}} \Phi^{T_n}_{\theta_n,t,T_n+\tau_{n+1}}(S, \mathbf{Y}) \ .$$

M step At time T_{n+1}, update the parameter $\theta_{n+1} = \bar{\theta}\left(\mathcal{S}^{\mathrm{BOEM}}_{T_n,\tau_{n+1}}(\theta_n)\right)$.

Note that the quantity $\mathcal{S}^{\mathrm{BOEM}}_{T_n,\tau_{n+1}}(\theta_n)$ corresponds to the intermediate quantity (2) computed with the observations $(\mathbf{Y}_{T_n+1}, \cdots, \mathbf{Y}_{T_n+\tau_{n+1}})$. This algorithm is fully online if the E-step can be processed online: the observations along block n have to be used once and the algorithm should not ask for a storage of the data. To that goal, the key property is to observe that (see e.g. [3])

$$\mathcal{S}^{\mathrm{BOEM}}_{T,\tau}(\theta) = \phi^\theta_{T,\tau}(R^\theta_{T,\tau}) \tag{4}$$

where $\phi^{\theta}_{T,t}$ is the filtering distribution at time t w.r.t. the parameter θ and the observations $(\mathbf{Y}_{T+1}, \cdots, \mathbf{Y}_{T+t})$, and the functions $R^{\theta}_{T,t} : \mathbb{X} \to \mathbb{R}^d$, $1 \leq t \leq \tau$, satisfy the following equation

$$R^{\theta}_{T,t}(x) = \frac{1}{t}\mathrm{B}^{\theta}_{T,t}\left(x, S(\cdot, x, Y_{T+t})\right) + \frac{t-1}{t}\mathrm{B}^{\theta}_{T,t}\left(x, R^{\theta}_{T,t-1}\right) , \tag{5}$$

where B^{θ}_{t} denotes the backward smoothing kernel at time t: $\mathrm{B}^{\theta}_{T,t}(x, \mathrm{d}x') \propto m_{\theta}(x', x)\phi^{\theta}_{T,t-1}(\mathrm{d}x')$. By convention, $R^{\theta}_{T,0} = 0$.

When the expectation under the filtering distribution $\phi^{\theta}_{T,t}$ is intractable, it can be replaced by a particle approximation. This yields to the *Particle-BOEM* (P-BOEM) algorithm.

block Particle E step compute the P-BOEM statistic $\mathcal{S}^{\mathrm{P-BOEM}}_{N_{n+1},T_n,\tau_{n+1}}(\theta_n)$, defined as a SMC approximation of $\mathcal{S}^{\mathrm{BOEM}}_{T_n,\tau_{n+1}}(\theta_n)$ computed with N_{n+1} particles.

M step. At time T_{n+1}, update the parameter $\theta_{n+1} = \bar{\theta}\left(\mathcal{S}^{\mathrm{P-BOEM}}_{N_{n+1},T_n,\tau_{n+1}}(\theta_n)\right)$.

Here again, the Particle E step has to be computed online; this can be done by applying the algorithm of [3] (see also [5]), which consists in replacing the filtering distributions in Eqs (4) and (5), by a particle approximation.

Eq. (5) shows that the sufficient statistic along block n follows a stochastic approximation dynamic. It is known that the convergence of such algorithms can be improved by replacing the updated quantity with its *averaged one* (see [9]). In our case, this yields to the *averaged BOEM* algorithm: each block E step and M step of BOEM are followed by

averaged block E step. compute the statistic

$$\widetilde{\mathcal{S}}^{\mathrm{BOEM}}_{n+1} \stackrel{\mathrm{def}}{=} \frac{T_n}{T_{n+1}}\widetilde{\mathcal{S}}^{\mathrm{BOEM}}_{n} + \frac{\tau_{n+1}}{T_{n+1}}\mathcal{S}^{\mathrm{BOEM}}_{T_n,\tau_{n+1}}(\theta_n) = \frac{1}{T_{n+1}}\sum_{j=1}^{n}\tau_{j+1}\mathcal{S}^{\mathrm{BOEM}}_{T_j,\tau_{j+1}}(\theta_j) .$$

averaged block M step. Update the parameter $\tilde{\theta}_{n+1} = \bar{\theta}\left(\widetilde{\mathcal{S}}^{\mathrm{BOEM}}_{n+1}\right)$.

The same averaged steps can be done for the E and M P-BOEM steps, thus yielding to the *averaged P-BOEM.* The convergence properties of both BOEM and P-BOEM and their averaged versions have been derived in [9] and in [10]. These algorithms are seen as perturbations of a limiting EM recursion and it can be proved that they inherit the asymptotic behavior of this limiting EM. This has to be compared to the online EM of [5] which introduces many approximations and which theoretical analysis remains quite challenging.

3 Experiments

In this section, the performance of the algorithms presented in Section 2 are illustrated through Monte Carlo experiments. The SLAM problem has been addressed in different works [2]. When both the robot motion and the robot perception are perturbed by Gaussian noises, EKF-based algorithms proposed to approximate the joint distribution of the map and the robot pose. It has been successfully applied to numerous SLAM problems. Despite encouraging experimental results, the EKF-based SLAM algorithms do not converge due to the required Taylor expansion and the necessity to approximate a joint distribution between the pose and the map which is a static parameter, see [1,7]. On the other hand, the most famous SLAM solution proposed is the FastSLAM algorithm and its different variants, see [13]. In this case, the model is not linearized and the motion noise is not necessarily Gaussian. In the FastSLAM framework, the joint distribution of the robot trajectories and the map is approximated. The robot path is estimated with sequential Monte Carlo methods and, for each particle representing a trajectory, landmark positions are estimated using EKF steps. A linearization step is required to perform the update of each landmark position. Once again, experimental results and the possibility to keep a map estimate for each possible trajectory made these methods successful. However, the issue of the joint estimation of the static parameter and the robot path still remain: in this case it comes from the well known path degeneracy issue when computing joint distribution with SMC methods. As a map estimate is associated to each particle, after successive resampling steps, all the particles share the same estimation for old landmarks.

To overcome this difficulty, [12] introduced the *MarginalSLAM* algorithm and [8] the *OnlineEM SLAM*. The SLAM problem is seen as an inference task in HMM. The map parameterizes a latent data model and is estimated in the maximum likelihood sense. The localization procedure is answered by SMC methods. In [12], the map is estimated by a stochastic gradient algorithm (see e.g. [11]). In [8], this estimation procedure is replaced by an online EM based algorithm. In this paper, we propose to use the P-BOEM algorithm to sequentially estimate the map and to produce weighted particles to solve the localization problem. As said in Section 1, the convergence properties of P-BOEM have been addressed in [10], justifying the use of this algorithm to give a solution to the SLAM problem.

The robot evolves in a 2-dimensional landmark based map: its pose $x_t \stackrel{\text{def}}{=} \{x_{t,i}\}_{i=1}^3$ consists in cartesian coordinates $x_{t,1}$ and $x_{t,2}$ and a heading direction $x_{t,3}$. At each time step, the robot motion is controlled by deterministic commands: a velocity v_t and a heading direction ψ_t. The evolution of the robot pose can be written:

$$x_t = f(x_{t-1}, \hat{v}_t, \hat{\psi}_t) \; , \tag{6}$$

where $(\hat{v}_t, \hat{\psi}_t) \sim \mathcal{N}_2(0, Q)$. Q is assumed to be known. From now on, f is the kinematic model of the front wheel of a bicycle (see e.g. [1]):

$$f(x_{t-1}, \hat{v}_t, \hat{\psi}_t) = x_{t-1} + \begin{pmatrix} \hat{v}_t d_t \cos(x_{t-1,3} + \hat{\psi}_t) \\ \hat{v}_t d_t \sin(x_{t-1,3} + \hat{\psi}_t) \\ \hat{v}_t d_t \frac{\sin(\hat{\psi}_t)}{B} \end{pmatrix} \; ,$$

where d_t is the time period between two successive poses and B is the robot wheelbase.

Each landmark is represented by a vector θ_j. It is assumed that the total number of landmarks q and the association between observations and landmarks are known. The robot is equipped with range and bearing sensors: it observes the distance and the angular position of all landmarks in its neighborhood denoted by $\mathcal{A}_t$ at time t. The observation $y_{t,i} \in \mathbb{R}^2$ of the landmark i is written $y_{t,i} = h(x_t, \theta_{.,i}) + \delta_{t,i}$, where h is defined by

$$h(x, \tau) = \begin{pmatrix} \sqrt{(\tau_1 - x_1)^2 + (\tau_2 - x_2)^2} \\ \arctan \frac{\tau_2 - x_2}{\tau_1 - x_1} - x_3 \end{pmatrix} .$$

The noise vectors $\{\delta_{t,i}\}_{t,i}$ are i.i.d Gaussian $\mathcal{N}_2(0, R)$, where R is assumed to be known. In this example, the complete-data log-likelihood is not exponential. The marginal log-likelihood is written (up to an additive constant independent from θ),

$$\sum_{i \in \mathcal{A}_t} \ln g_\theta(x_t, y_{t,i}) \propto \sum_{i \in \mathcal{A}_t} \left[y_{t,i} - h(x_t, \theta_i)\right]^\star R^{-1} \left[y_{t,i} - h(x_t, \theta_i)\right] .$$

P-BOEM cannot be directly applied: therefore, at the beginning of each block, the function $\tau \mapsto h(x, \tau)$ is approximated by its first order Taylor expansion at all the current landmark estimates. This kind of first order approximations is of common use in the SLAM literature (e.g. in EKF-SLAM or in FastSLAM). In our case, this leads to a quadratic approximation of the likelihood of the observation and to an approximate exponential-HMM (see [8]).

Observations are sampled using $R = \operatorname{diag}(\sigma_r^2, \sigma_b^2)$, where $\sigma_r = 0.5$m and $\sigma_b = \frac{\pi}{60}$rad. The robot path is sampled with a given set of controls and using $Q = \operatorname{diag}(\sigma_v^2, \sigma_\phi^2)$ where $\sigma_v = 0.5\text{m.s}^{-1}$ and $\sigma_\psi = \frac{\pi}{60}$rad. In this experiment, the proposed algorithm is compared to the *MarginalSLAM* and to the *OnlineEM SLAM*. The block size sequence is slowly increasing $\{\tau_n \propto n^{1.1}\}_{n \geq 1}$ to allow a sufficiently large number of updates. The number of particles is constant on each block and fixed at 50. For the SMC step, new particles are sampled using the prior model, this method is known in the SMC literature as the Bootstrap filter. The step-size sequence used in the *MarginalSLAM* and in the *OnlineEM SLAM* for the stochastic approximation step are chosen such that $\gamma_n \propto n^{-0.8}$.

For each run the weighted mean of the particles and the estimated map are stored. Figure 1 displays the estimated path given by the *MarginalSLAM* and the P-BOEM SLAM for one of the 50 Monte Carlo runs. The path estimate given by the P-BOEM is clearly better than the one given by the *MarginalSLAM*.

Figure 2 displays boxplots of the landmark estimation error over 50 Monte Carlo runs for the *MarginalSLAM* and the P-BOEM SLAM. Both algorithms give similar results for the estimation of the landmarks observed at the beginning of the experiment. However, when considering the other landmarks, P-BOEM SLAM shows better results. Figure 3 compares the result given by the P-BOEM and the Online EM SLAM. As noted in [10], both algorithms have a similar behaviors.

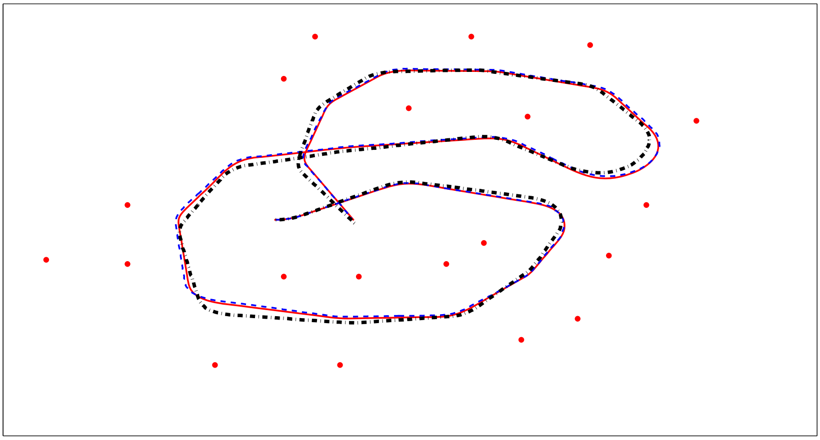

Fig. 1. True trajectory (bold line) and true landmark positions (balls) with the estimated path given by the P-BOEM SLAM (dashed line) and by the *MarginalSLAM* (dashed and dotted line)

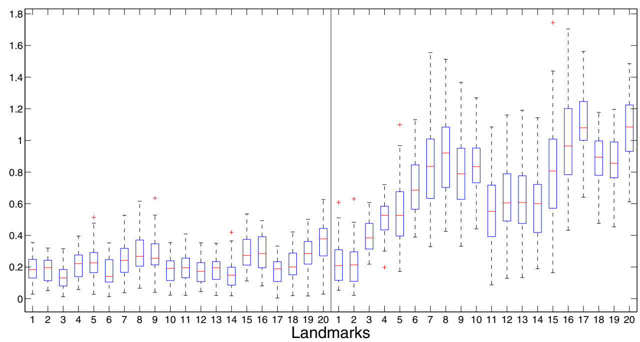

Fig. 2. Distance between the estimate at the end of the loop ($T = 1800$) and the true position using the P-BOEM SLAM (left) and the Marginal SLAM (right)

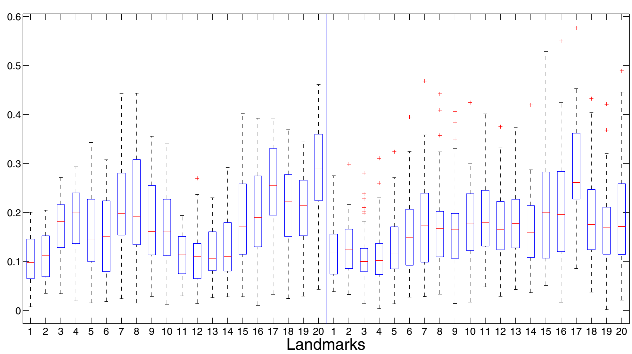

Fig. 3. Distance between the estimate at the end of the loop ($T = 1800$) and the true position using the P-BOEM SLAM (left) and the *OnlineEM SLAM* (right)

4 Conclusion

New algorithms for online Maximum-Likelihood based inference in exponential HMM have been proposed. These new online-EM procedures have been applied to solve the SLAM problem which is a case of non-exponential HMM. The experiments show that the our algorithm provides better result than the *Marginal-SLAM* algorithm when estimating the map online. The results are quite similar to those given by the online EM algorithm of [5]. Nevertheless, the asymptotic behavior of our algorithms has been addressed showing that they answer to the Maximum-Likelihood estimation problem. On the contrary, it remains quite challenging to analyze the convergence properties of the online EM of [5].

References

1. Bailey, T., Nieto, J., Guivant, J., Stevens, M., Nebot, E.: Consistency of the EKF-SLAM algorithm. In: IEEE International Conference on Intelligent Robots and Systems, pp. 3562–3568 (2006)
2. Burgard, W., Fox, D., Thrun, S.: Probabilistic robotics. MIT Press, Cambridge (2005)
3. Cappé, O.: Online EM algorithm for Hidden Markov Models. To Appear in J. Comput. Graph. Statist. (2011)
4. Cappé, O., Moulines, E.: Online Expectation Maximization Algorithm for Latent Data Models. J. Roy. Statist. Soc. B, 593–613 (2009)
5. Del Moral, P., Doucet, A., Singh, S.S.: Forward smoothing using sequential Monte Carlo. arXiv:1012.5390v1 (2011)
6. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. J. Roy. Statist. Soc. B, 1–38 (1977)
7. Julier, S.J., Uhlmann, J.K.: A counter example to the theory of simultaneous localization and map building. In: IEEE International Conference on Robotics and Automation, pp. 4238–4243 (2001)
8. Le Corff, S., Fort, G., Moulines, E.: Online EM algorithm to solve the SLAM problem. In: IEEE Workshop on Statistical Signal Processing (2011)
9. Le Corff, S., Fort, G.: Online Expectation Maximization based algorithms for inference in Hidden Markov Models. arXiv:1108.3968 (2011)
10. Le Corff, S., Fort, G.: Convergence of a particle-based approximation of the Block Online Expectation Maximization algorithm. arXiv:1111.1307 (2011)
11. Le Gland, F., Mevel, L.: Recursive estimation in HMMs. In: IEEE Conference on Decision and Control, pp. 3468–3473 (1997)
12. Martinez-Cantin, R.: Active map learning for robots: insights into statistical consistency. PhD thesis (2008)
13. Montemerlo, M., Thrun, S., Koller, D., Wegbreit, B.: FastSLAM 2.0: An Improved Particle Filtering Algorithm for Simultaneous Localization and Mapping that Provably Converges. In: IJCAI (2003)
14. Titterington, D.M.: Recursive parameter estimation using incomplete data. J. Roy. Statist. Soc. B, 257–267 (1984)