Particle Methods for Parameter Estimation in Hidden Markov Models

Olivier Cappé

Centre National de la Recherche Scientifique (CNRS)
& Ecole Nationale Supérieure des Télécommunications (ENST)
Paris, France

Joint work with Eric Moulines (ENST) & Tobias Rydén (Lund)

Outline

Hidden Markov Model

The Hidden State Process $\{X_k\}_{k\geq 0}$ is a Markov chain on X with initial distribution $\nu(x)\lambda(dx)$ and transition kernel $q(x, x')\lambda(dx')$.

The Observed Process $\{Y_k\}_{k\geq 0}$ is such that the conditional distribution of $Y_{0:n}$* given $X_{0:n}$ has the product (conditional independence) form

$$\prod_{k=0}^{n} g(X_k, y_k)\mu(dy_k) \ .$$

We use $\nu(x\,;\theta)$, $q(x, x'\,;\theta)$, $g(x, y\,;\theta)$ to denote dependence with respect to the model parameter θ.

*$Y_{0:n}$ denotes the collection $Y_0, \ldots, Y_n$.

(General) State-Space Model

This notion is (in our views) fully equivalent to the HMM described in a functional form:

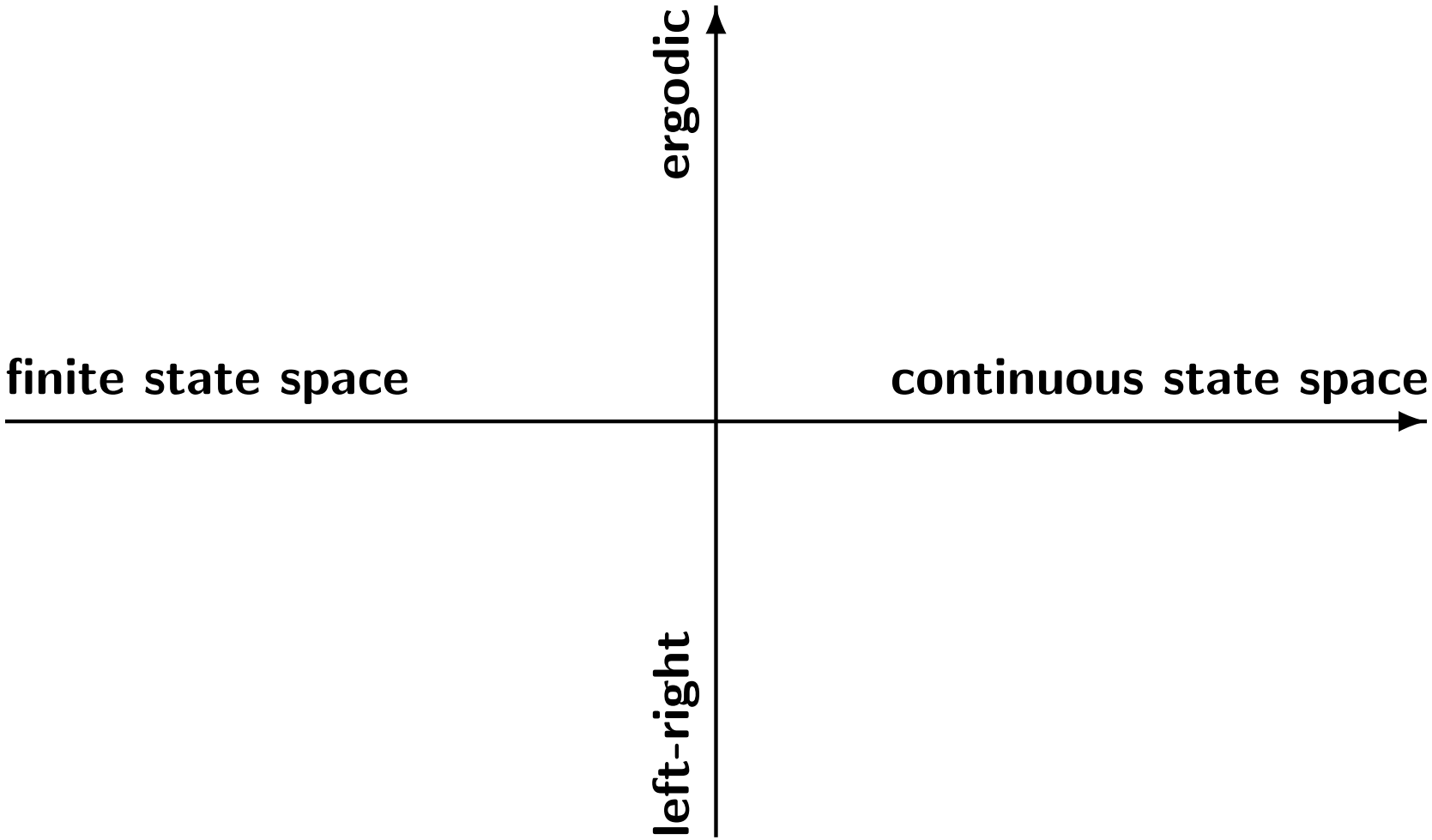
$$X_{k+1} = a(X_k, U_k) \ ,$$
$$Y_k = b(X_k, V_k) \ ,$$

where $\{U_k\}_{k\geq 0}$ and $\{V_k\}_{k\geq 0}$ are mutually independent i.i.d. sequences of random variables (also independent of X_0).

Remark

The term "state-space model" often refers to the case where a and b are linear functions of their arguments (and $\{U_k\}$, $\{V_k\}$, X_0 are jointly Gaussian).
Likewise, the term "HMM" is sometimes used (not in this talk!) more restrictively for the case where X is a finite set.

HMM Examples



HMM Examples

source coding ion channel modelling	**ergodic**	tracking, vision stochastic volatility g. l. state-space model
finite state space		**continuous state space**
speech recognition handwritting recognition	**left-right**	trajectory models

A Note on Non-HMMs

There are models that are not HMMs, for which (most of) what is said today applies, in particular Markov-switching (autoregressive) models (sometimes a.k.a. jump Markov models):

The Observed Process $\{Y_k\}_{k\geq 0}$ is such that the conditional distribution of Y_k given $\{X_k\}_{k\geq 0}$ and $Y_{0:k-1}$ is given by

$$g\left[\left(X_k, Y_{k-p:k-1}\right), y\right] \mu(dy) \ .$$

General Objectives

Except in particular cases, there is no "simple" (e.g., moment-based) estimator, hence the need to compute and study:

The (Log-)Likelihood

$$\ell_n(Y_{0:n}\,;\theta) \stackrel{\text{def}}{=} \log \mathsf{L}_n(Y_{0:n}\,;\theta)\ ,$$

The Score and Observed Information Matrix

$$\nabla_\theta \ell_n(Y_{0:n}\,;\theta)\ ,\ -\nabla^2_\theta \ell_n(Y_{0:n}\,;\theta)\ ,$$

The Intermediate Quantity of the Expectation-Maximization (EM)

$$\mathcal{Q}(\theta\,;\theta') = \mathsf{E}\left[\log p(X_{0:n}, Y_{0:n}\,;\theta) \middle| Y_{0:n}\,;\theta'\right]\ .$$

Two Main Approaches

1. Extended Chain / Sensitivity Equations
   [Gupta and Mehra, 1974], [Campillo and Le Gland, 1989], [Le Gland and Mevel, 1997], [Douc and Matias, 2002]

2. Fisher Equality [Baum et al., 1970], [Segal and Weinstein, 1989], [Douc et al., 2004]
   - Recursive Implementation [Zeitouni and Dembo, 1988], [Elliott et al., 1995], [C, 2001]

Additive Decomposition of the Log-Likelihood

$$\begin{aligned}\ell_n(Y_{0:n}\,;\theta) &= \sum_{k=0}^{n} \log \mathsf{L}_n(Y_k|Y_{0:k-1}\,;\theta) \\ &= \sum_{k=0}^{n} \log\Bigg[\int g(x, Y_k\,;\theta) \quad \underbrace{\mathsf{P}\left(X_k \in dx|\, Y_{0:k-1}\,;\theta\right)}_{\stackrel{\text{def}}{=}\phi_{k|k-1}(x\,;\theta)\lambda(dx) \quad \text{“predictor”}} \Bigg],\end{aligned} \tag{1}$$

where the predictor may be updated recursively

$$\phi_{k+1|k}(x'\,;\theta) = \int \frac{g(x, Y_k\,;\theta)\phi_{k|k-1}(x\,;\theta)}{\mathsf{L}_n(Y_k|Y_{0:k-1}\,;\theta)} q(x, x'\,;\theta)\,\lambda(dx) \tag{2}$$

The Extended Chain / Sensitivity Equations Approach

- $\ell_n(Y_{0:n}\,;\theta)$ is an additive functional of an extended Markov chain $\left\{X_k, Y_k, \phi_{k|k-1}(.\,;\theta)\right\}_{k\geq 0}$.
- Same is true for $\nabla_\theta \ell_n(Y_{0:n}\,;\theta)$ and $-\nabla^2_\theta \ell_n(Y_{0:n}\,;\theta)$ by formal differentiation of (1)–(2).
- In simple models at least (in particular when X is finite), may be effectively used for parameter estimation.

Discussion

Advantages

- Makes it possible to use known MC results.
- Naturally has a recursive form, which is good for large data sets, and may be used (...) for recursive estimation.

Discussion

Advantages

- Makes it possible to use known MC results.
- Naturally has a recursive form, which is good for large data sets, and may be used (...) for recursive estimation.

The Downside

- $\{X_k, Y_k, \phi_{k|k-1}(. \,; \theta)\}_{k \geq 0}$ is a very degenerate MC.
- Details of the computation of the score (and Hessian!) are often hairy.
- Necessitates the definition of quantities which are not of direct interest such as $\nabla_\theta \phi_{k+1|k}(x' \,; \theta)$ ("tangent predictor").

Additive Decomposition of the Gradient of the Log-Likelihood

$$\begin{aligned}\nabla_\theta \ell_n(Y_{0:n}\,;\theta) &= \mathsf{E}\left[\nabla_\theta \log p(X_{0:n}, Y_{0:n}\,;\theta) \middle| Y_{0:n}\,;\theta\right] \\ &= \sum_{k=0}^{n-1} \mathsf{E}\left[\nabla_\theta \log t_k(X_k, X_{k+1}\,;\theta) \middle| Y_{0:n}\,;\theta\right] \,, \qquad (3)\end{aligned}$$

where

$$t_k(x, x'\,;\theta) = \begin{cases} \nu(x\,;\theta) g(x, Y_0\,;\theta) q(x, x'\,;\theta) g(x', Y_1\,;\theta) \,, & \text{if } k = 0, \\ q(x, x'\,;\theta) g(x', Y_{k+1}\,;\theta) \,, & \text{otherwise.} \end{cases}$$

Discussion

- Involves the smoothing distributions rather than the prediction distributions.
- Cannot rely on usual† MC properties anymore but need to establish general forgetting properties, e.g., on behavior of

$$\left|\mathsf{E}\left[f(X_k, X_{k+1}) \middle| Y_{0:n}\,;\theta\right] - \mathsf{E}\left[f(X_k, X_{k+1}) \middle| Y_{0:n+p}\,;\theta\right]\right| \ ,$$

  as $n \to \infty$.
- Practically very simple once smoothing distributions have been evaluated;
- Allows for the use of a very basic optimization approach (Expectation-Maximization).

† Note that the conditional distribution of $\{X_k\}_{k\geq 0}$ given $Y_{0:n}$ is that of a non-homogeneous MC.

Recursive Implementation

Define τ_k such that, for any (bounded) function f,

$$\int f(x)\tau_k(x\,;\theta)\lambda(dx) = \mathsf{E}\left[f(X_k)\sum_{l=0}^{k-1}\nabla_\theta \log t_l(X_l, X_{l+1}\,;\theta) \middle| Y_{0:k}\,;\theta \right]$$

such that $\int \tau_k(x\,;\theta)\lambda(dx)$ is the quantity of interest.

- τ_{k+1} may be computed recursively from $\phi_{k|k}$, τ_k and Y_{k+1} as

$$\tau_{k+1}(x_{k+1}\,;\theta) = [c_{k+1}(\theta)]^{-1}\int \Big[\tau_k(x_k\,;\theta) + \phi_{k|k}(x_k\,;\theta)\,\nabla_\theta \log t_k(x_k, x_{k+1}\,;\theta)\Big] t_k(x_k, x_{k+1}\,;\theta)\lambda(dx_k)$$

- $\tau_k(\cdot\,;\theta) - \nabla_\theta \ell_k(Y_{0:k}\,;\theta)\phi_{k|k}(\cdot\,;\theta)$ is equal to the tangent filter $\nabla_\theta \phi_{k|k}(\cdot\,;\theta)$ (the recursion is fully equivalent to the sensitivity equations).

Observed Information Matrix

- Similar (conceptually) to the case of the score through the use of Louis identity (although details are somewhat more involved).

$$\nabla^2 \ell_n(Y_{0:n}\,;\theta) + \nabla \ell_n(Y_{0:n}\,;\theta)\big[\,-\,\big]^t = \sum_{k=0}^{n-1} \mathsf{E}\left[\nabla^2 \log t_k^\theta(X_k\,;\theta)\,\middle|\, Y_{0:n}\,;\theta\right] + \mathsf{E}\left[\sum_{k=0}^{n-1}\sum_{j=0}^{n-1} \nabla \log t_k^\theta(X_k, X_{k+1}\,;\theta)\big[\,-_j\,\big]^t \,\middle|\, Y_{0:n}\,;\theta\right]$$

- Also has a recursive implementation (although it does not reduce to an additive functional of the smoothing distributions).

Practical Implementation

Exact computation of the smoothing distributions is feasible

- in models where the state space X is finite [Stratonovich, 1960], [Baum et al., 1970]
- in Gaussian linear state-space models [Bryson and Frazier, 1963]; [Rauch et al., 1965] (and several papers in the time series literature of the early 1990s).

In all other cases (except but a few), only approximate numerical methods are available. We consider in the following the use of sequential Monte Carlo (a.k.a. particle filtering) methods.

Structure of the Joint Smoothing Distribution

The joint smoothing distribution

$$\phi_{0:k+1|k+1}(x_{0:k+1}\,;\theta) = [\mathsf{L}_{k+1}(Y_{0:k+1}\,;\theta)]^{-1} \prod_{l=0}^{k} t_l(x_l, x_{l+1}\,;\theta)$$

may be rewritten recursively as

$$\underbrace{\left[\frac{\mathsf{L}_{k+1}(Y_{0:k+1}\,;\theta)}{\mathsf{L}_k(Y_{0:k}\,;\theta)}\right]^{-1}}_{c_{k+1}(\theta)} \phi_{0:k|k}(x_{0:k}\,;\theta) t_k(x_k, x_{k+1}\,;\theta)\ ,$$

where the normalization constants are not computable.

Self-Normalized Importance Sampling

1. propose N independent "particle" trajectories $\{\xi^i_{0:k+1}\}^{1\leq i\leq N}$ under a Markovian scheme such that

$$\rho_{0:k+1}(\xi_{0:k+1}) = \rho_0(\xi_0)\prod_{l=1}^{k} r_l(\xi_l, \xi_{l+1})\ ;$$

2. compute importance weights

$$\omega^i_{k+1} = \frac{\phi_{0:k+1|k}(\xi^i_{0:k+1}\,;\theta)}{\rho_{0:k+1}(\xi^i_{0:k+1})}\ .$$

then,

$$\sum_{i=1}^{N} \frac{\omega^i_{k+1}}{\sum_{j=1}^{N}\omega^j_{k+1}} f(\xi^i_{0:k+1})$$

is an estimate of $\mathsf{E}\left[f(X_{0:k+1})|\,Y_{0:k+1}\,;\theta\right]$ for arbitrary (integrable) functions f.

Sequential (Implementation of) Importance Sampling

If f depends only on x_{k+1} or is a sum functional of the states, storing the complete particle paths is not needed as

$$\omega_{k+1}^i = \omega_k^i \times \frac{t_k(\xi_k^i, \xi_{k+1}^i \,;\theta)}{r_k(\xi_k, \xi_{k+1})} \,,$$

and, in the example of the score,

$$\underbrace{\sum_{l=0}^{k} \nabla_\theta \log t_l(\xi_l^i, \xi_{l+1}^i \,;\theta)}_{\gamma_{k+1}^i} = \nabla_\theta \log t_l(\xi_k^i, \xi_{k+1}^i \,;\theta) + \underbrace{\sum_{l=0}^{k-1} \nabla_\theta \log t_l(\xi_l^i, \xi_{l+1}^i \,;\theta)}_{\gamma_k^i} \,.$$

Weight Degeneracy and Resampling

Regular resampling is needed to avoid weight degeneracy and to guarantee the long-term (as k increases) stability of the particle filter [Gordon et al., 1993], [Del Moral and Guionnet, 2001].

Resampling

Replace $\{\xi_k^i, \omega_k^i\}^{i=1,\dots,N}$ by $\{\tilde{\xi}_k^i, \tilde{\omega}_k^i\}^{i=1,\dots,\tilde{N}}$ such that the discrepancy between the resampled weights $\{\tilde{\omega}_k^i\}^{i=1,\dots,\tilde{N}}$ is reduced and $\sum_{i=1}^{\tilde{N}} \tilde{\omega}_k^i \delta_{\tilde{\xi}_k^i}$ is a good approximation to $\sum_{i=1}^{N} \omega_k^i \delta_{\xi_k^i}$.

In general the resampling is random and subject to the constraints

$$\begin{cases} \tilde{N} = N \\ \tilde{\omega}_k^i = 1/N \\ \mathsf{E}\left[\# \left\{ i, 1 \le i \le N : \tilde{\xi}_k^i = \xi_k^j \right\} \middle| \left\{ \omega_k^l \right\} \right] = N\omega_k^j \quad (\text{for } j = 1, \dots, N) \end{cases}$$

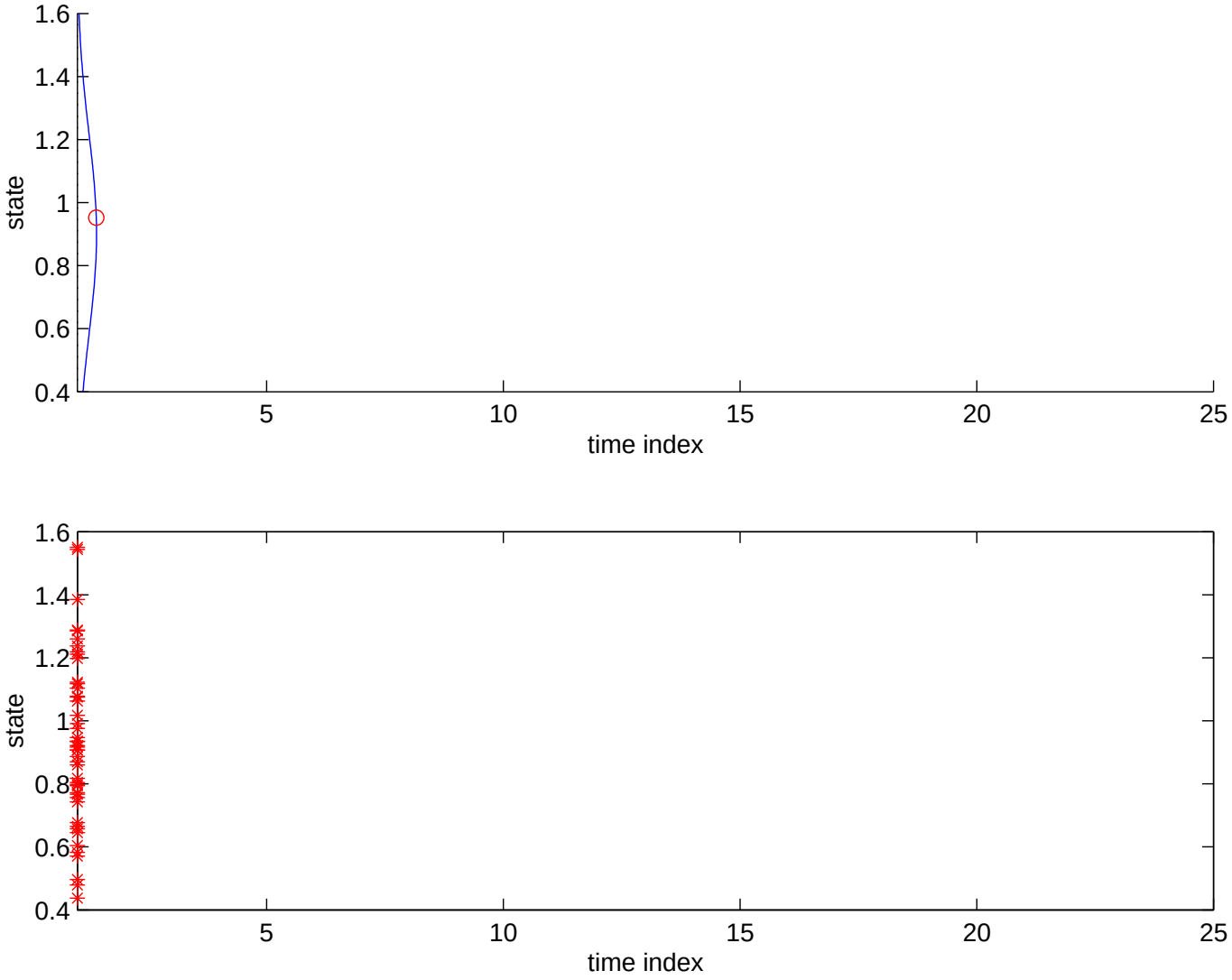
Illustration of the Boostrap Filter on a Toy Example

- Noisy AR(1) model

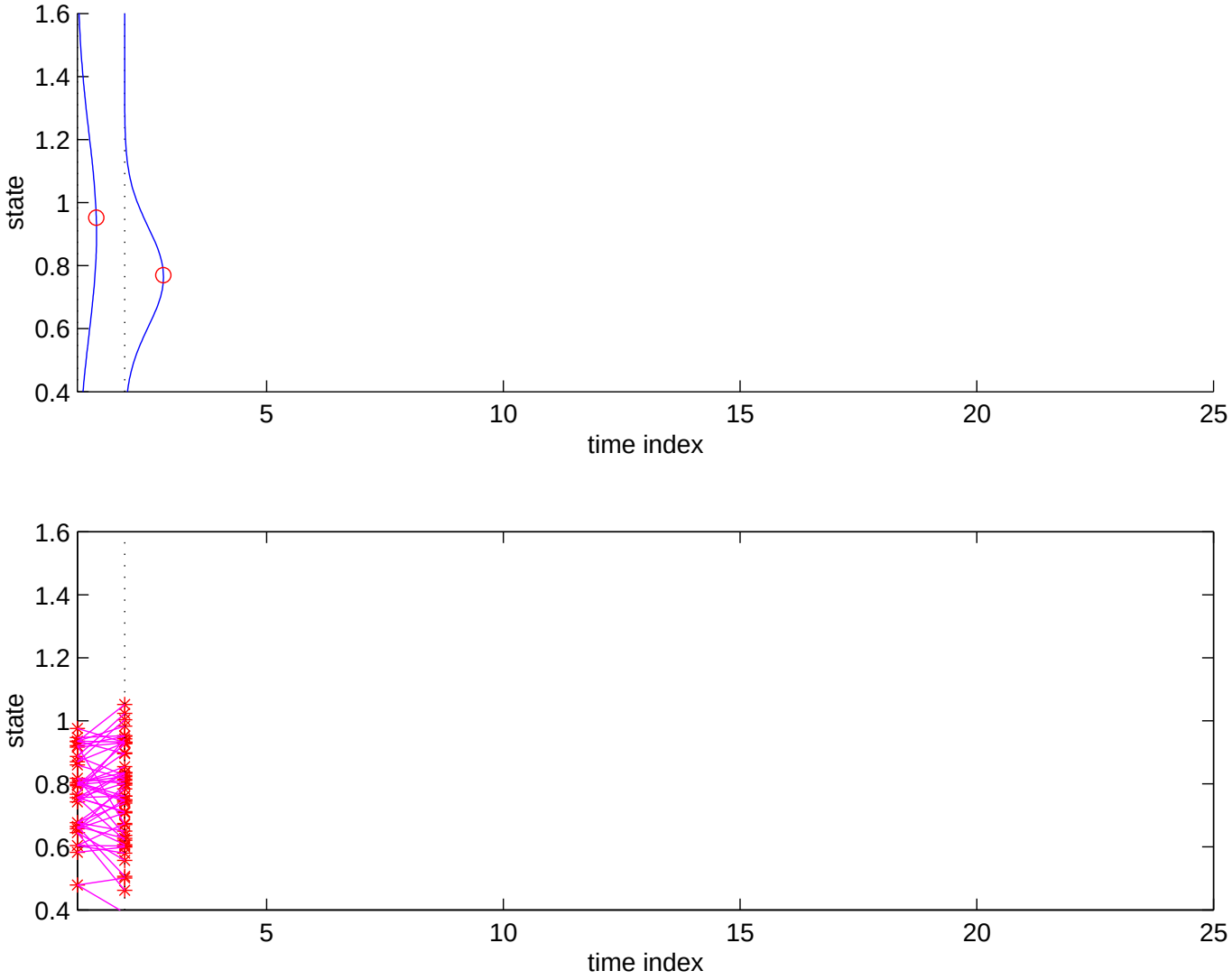
$$X_{k+1} - \mu = \phi(X_k - \mu) + \sigma U_k$$
$$Y_k = X_k + \eta V_k$$

$$\mu = 0.9, \phi = 0.95, \sigma^2 = 0.01, \eta^2 = 0.02 = (\sigma^2/(1-\phi^2))/5$$

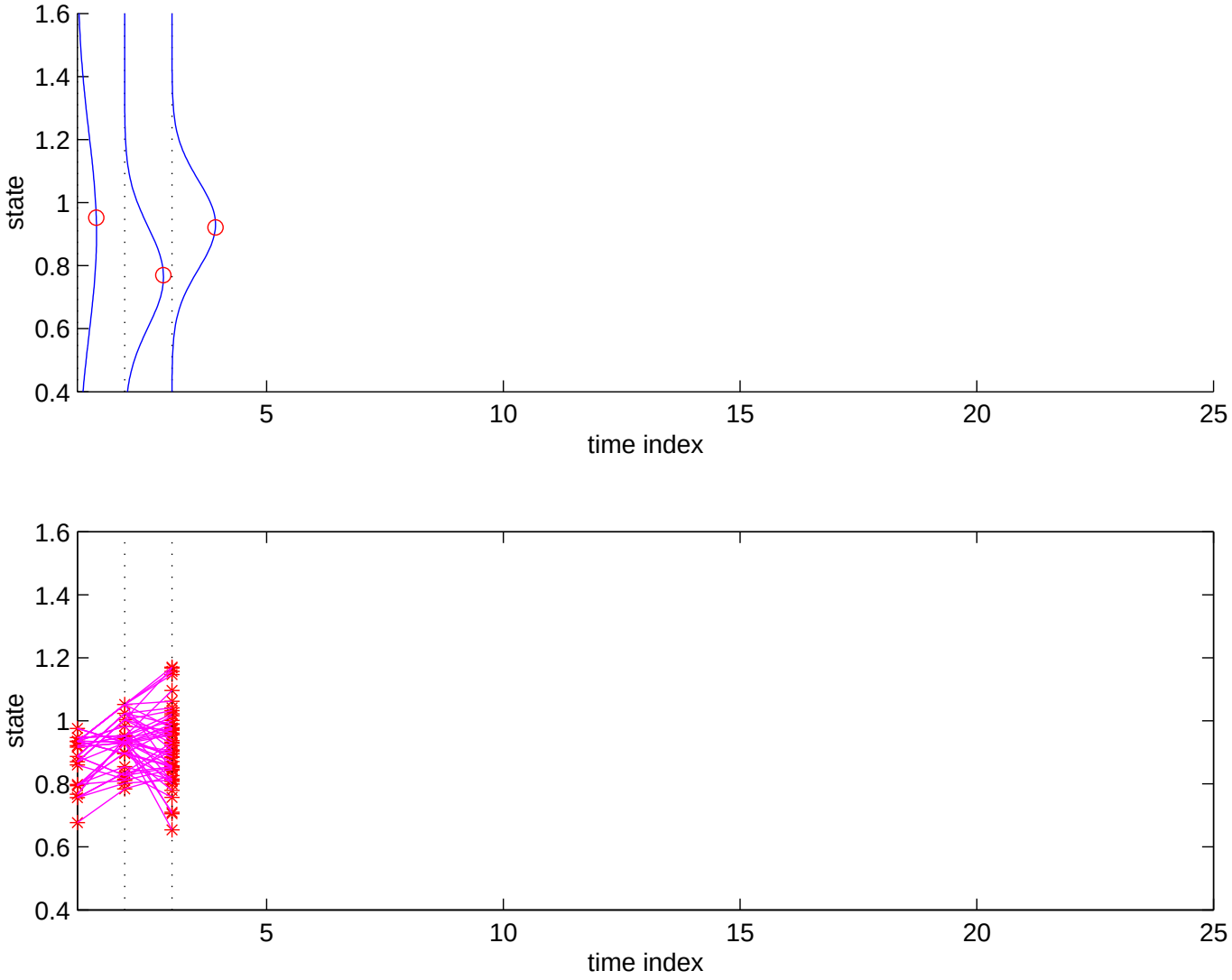
- To approximate the predictive distribution $\phi_{k+1|k}$, we use the bootstrap filter with $N = 50$ particles, plotting the full particle paths $\{\xi^i_{0:k}, \tilde{\xi}^i_{k+1}\}_{1 \le i \le N}$ for each time index.
- This example is used since we may also compute the actual filtering densities using Kalman filtering



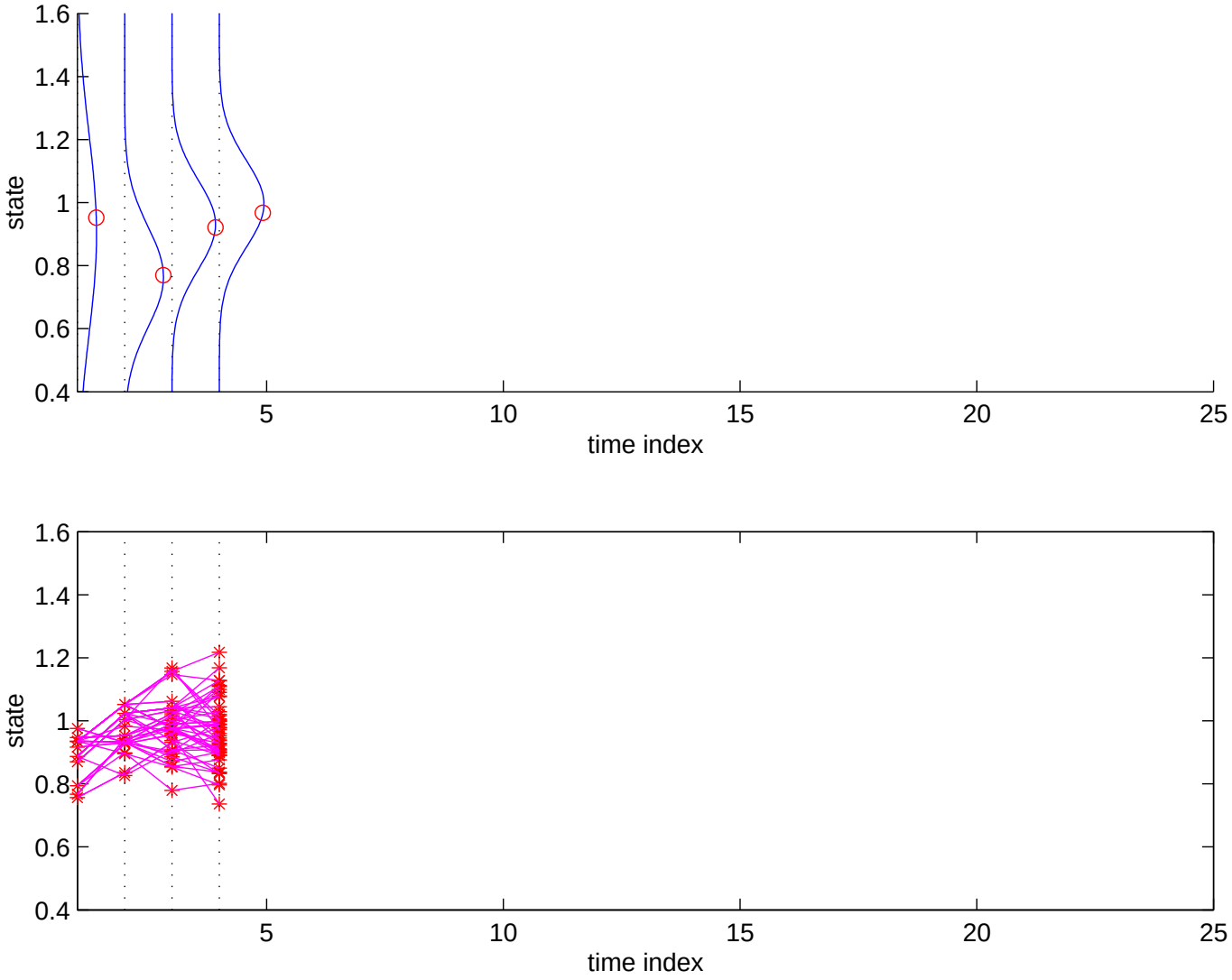
Predictive densities and evolution of the particle paths



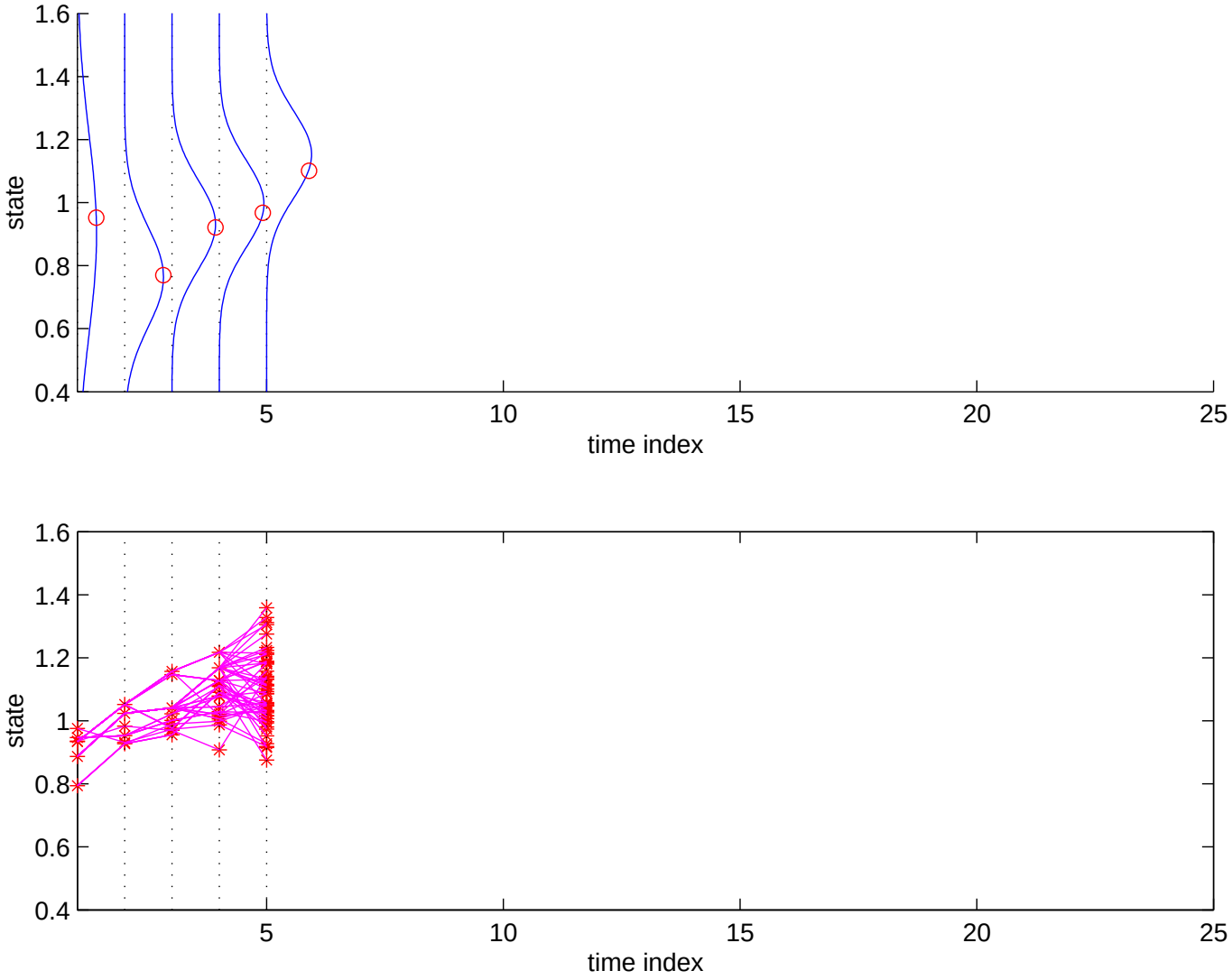
Predictive densities and evolution of the particle paths



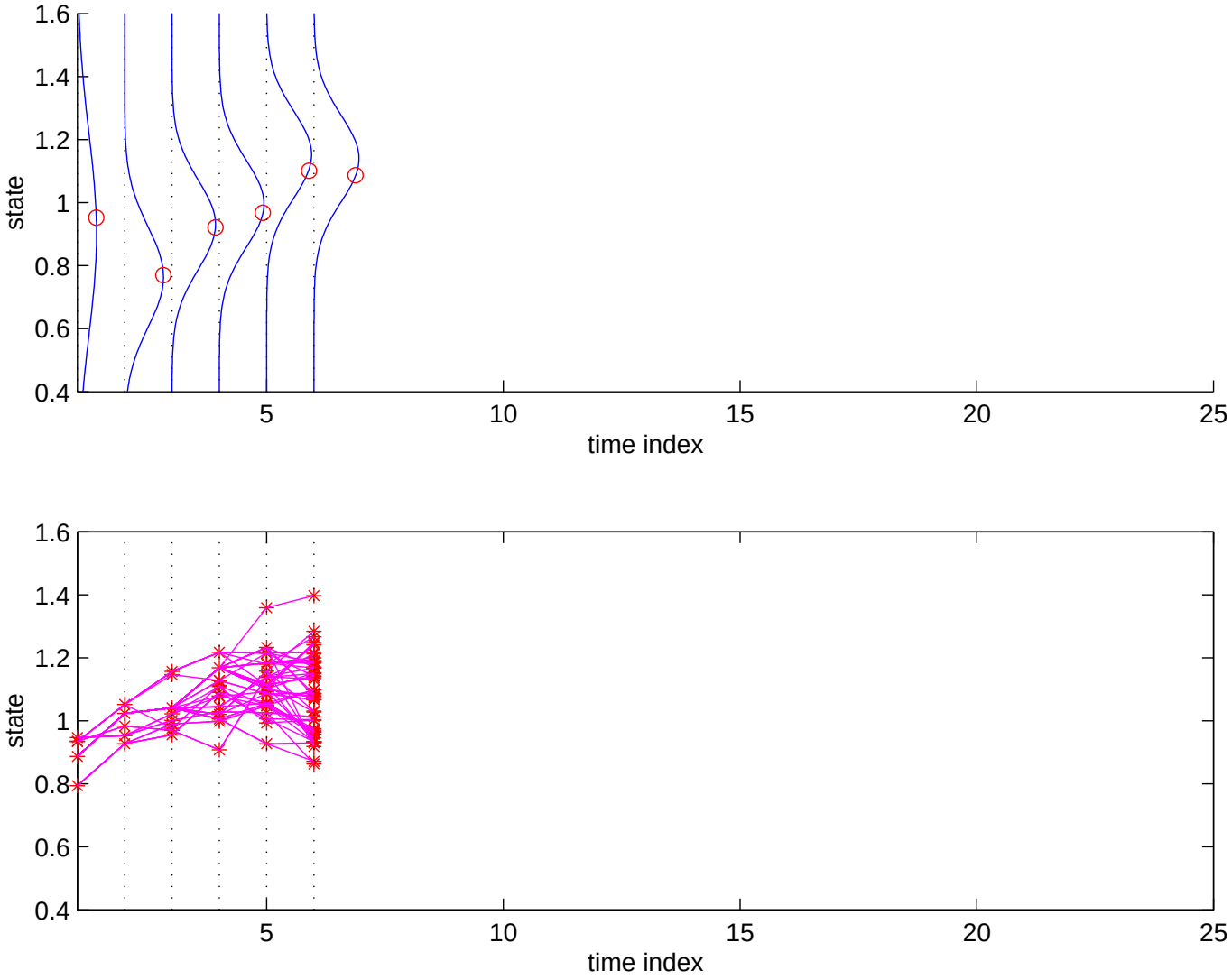
Predictive densities and evolution of the particle paths



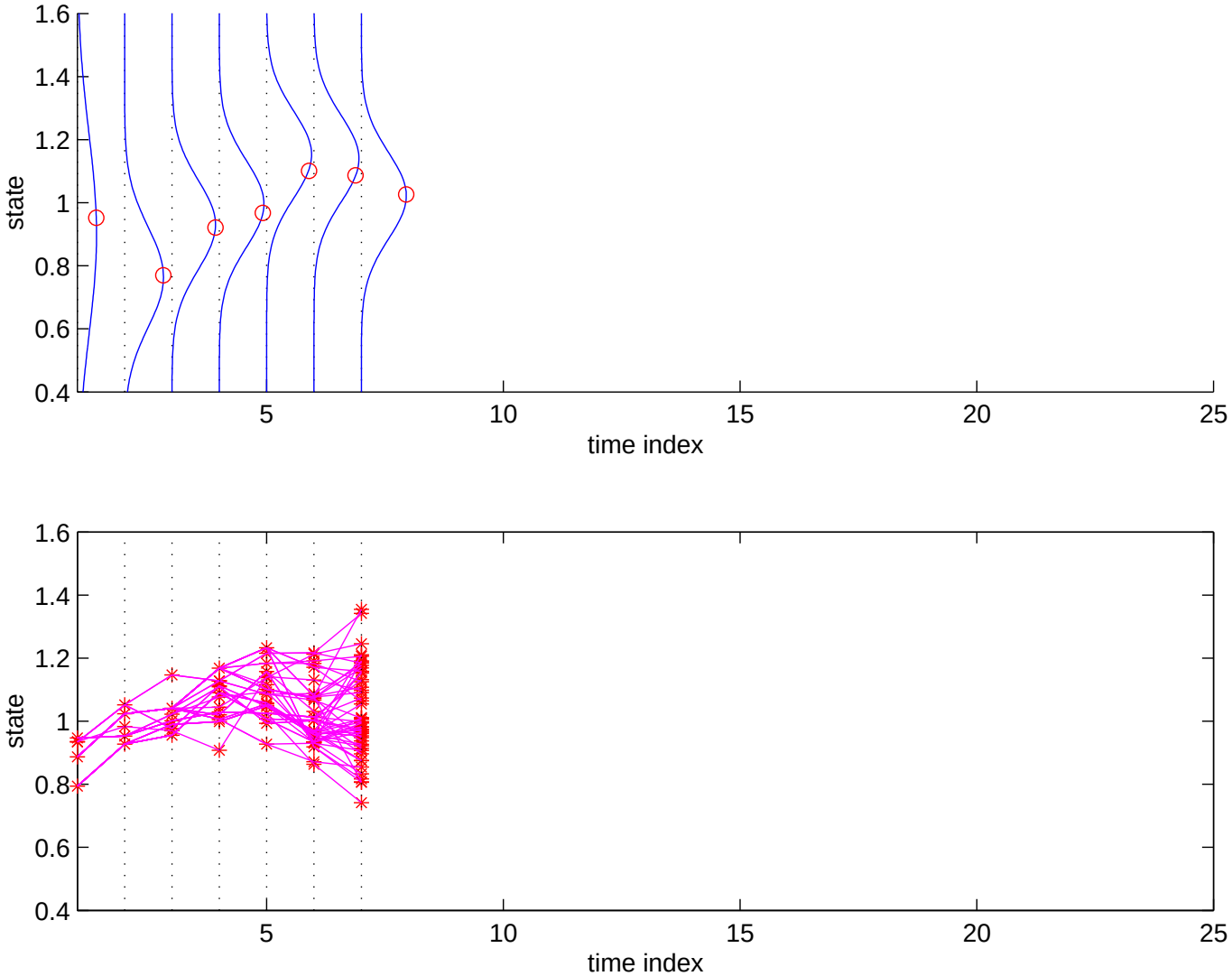
Predictive densities and evolution of the particle paths



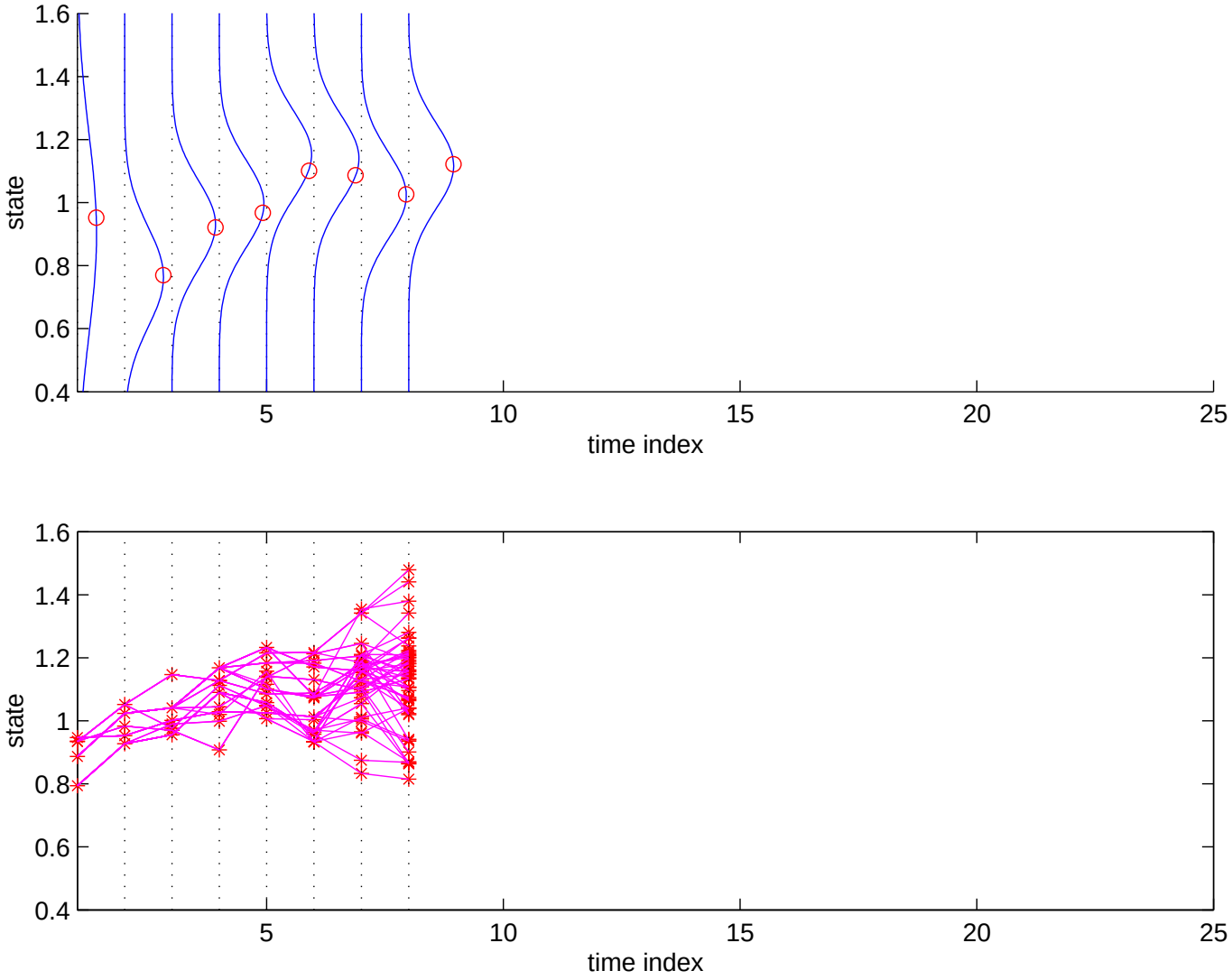
Predictive densities and evolution of the particle paths



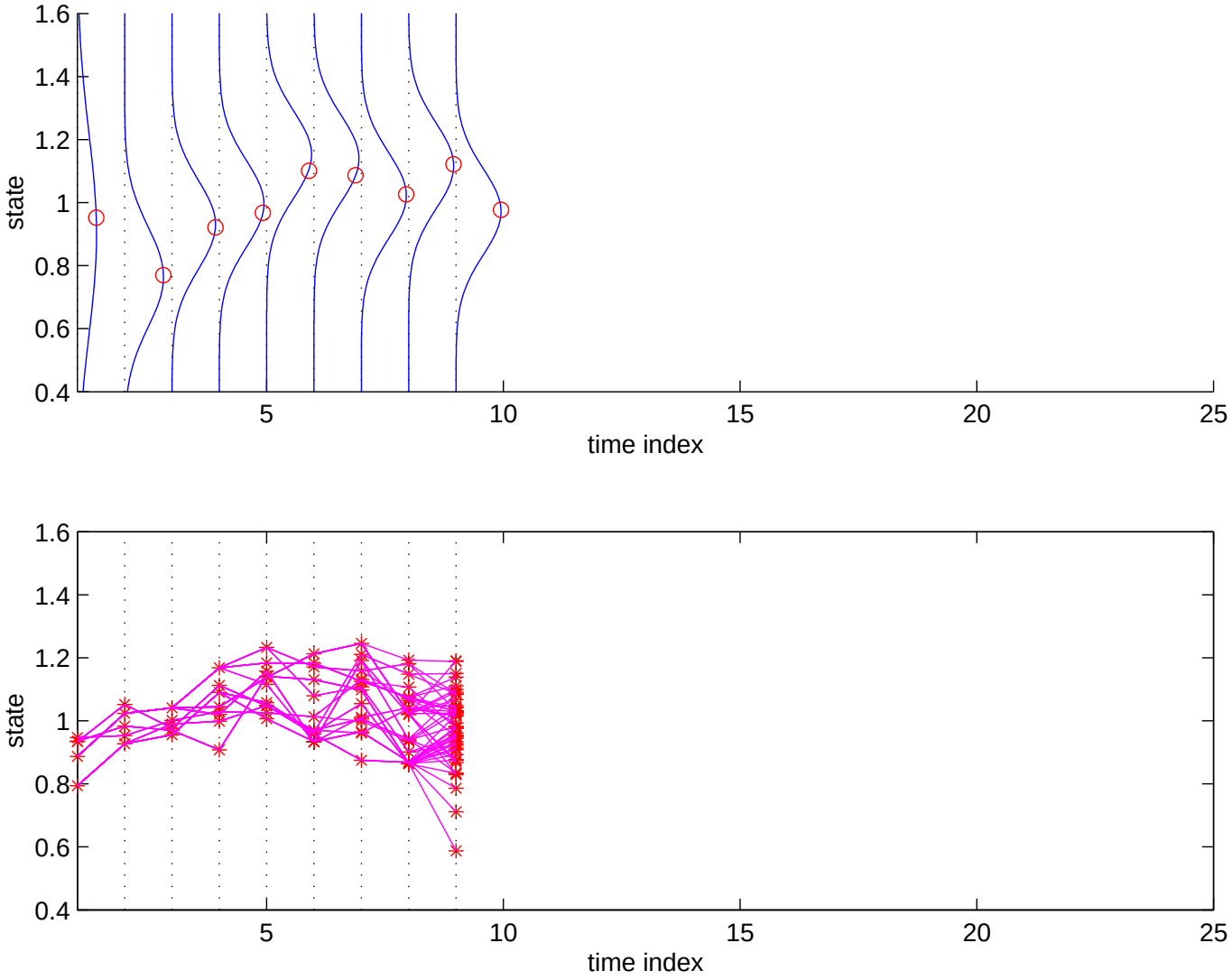
Predictive densities and evolution of the particle paths



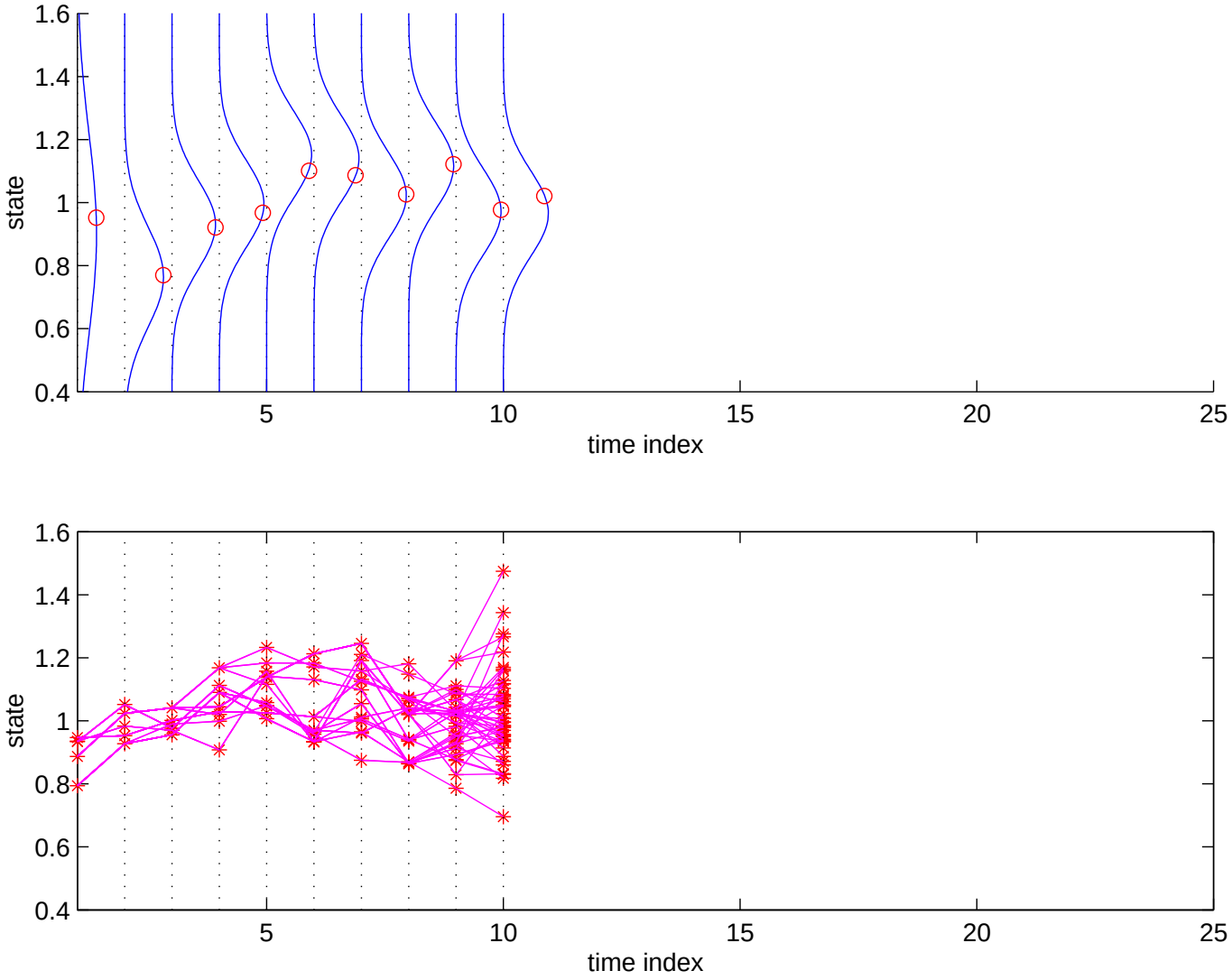
Predictive densities and evolution of the particle paths



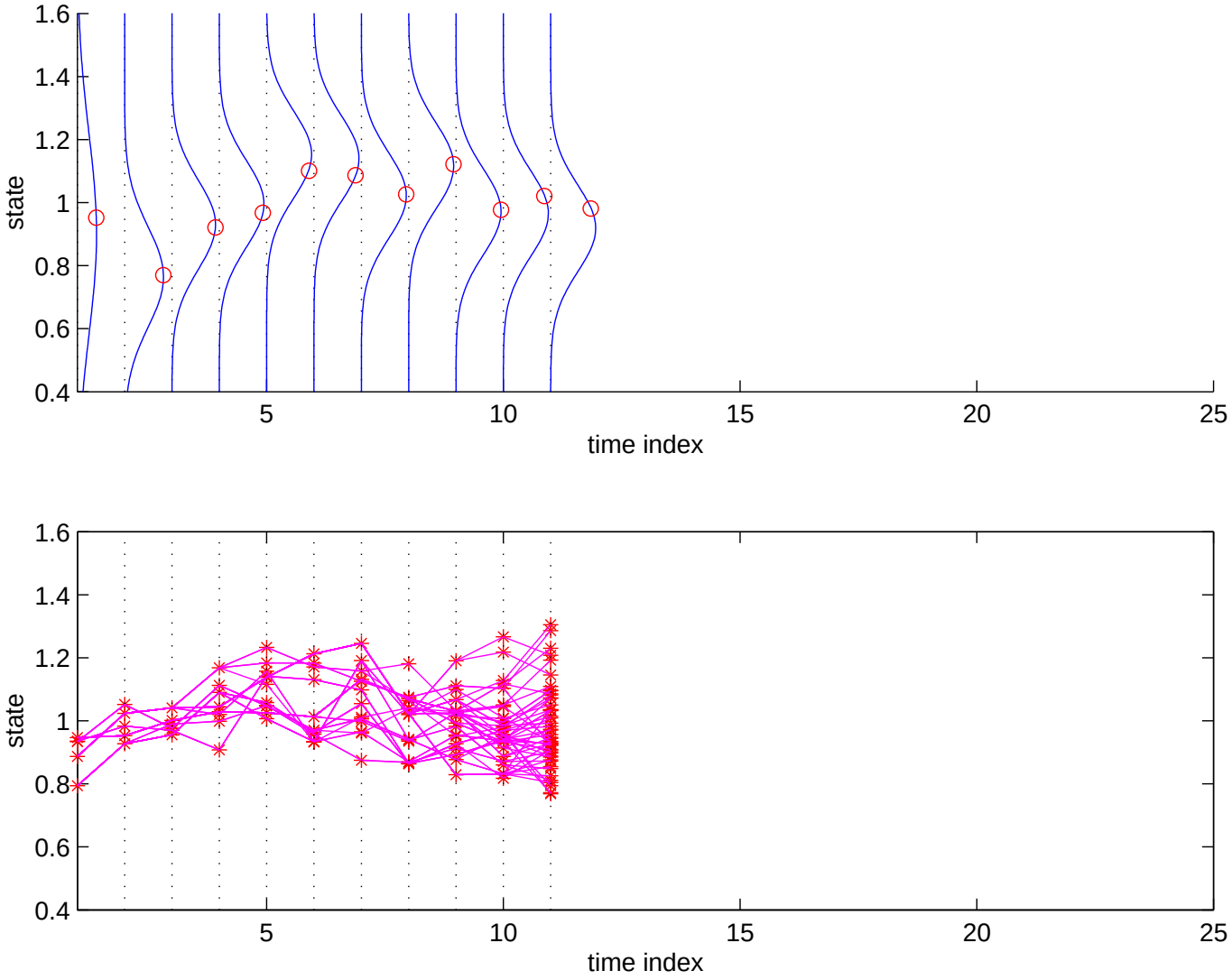
Predictive densities and evolution of the particle paths



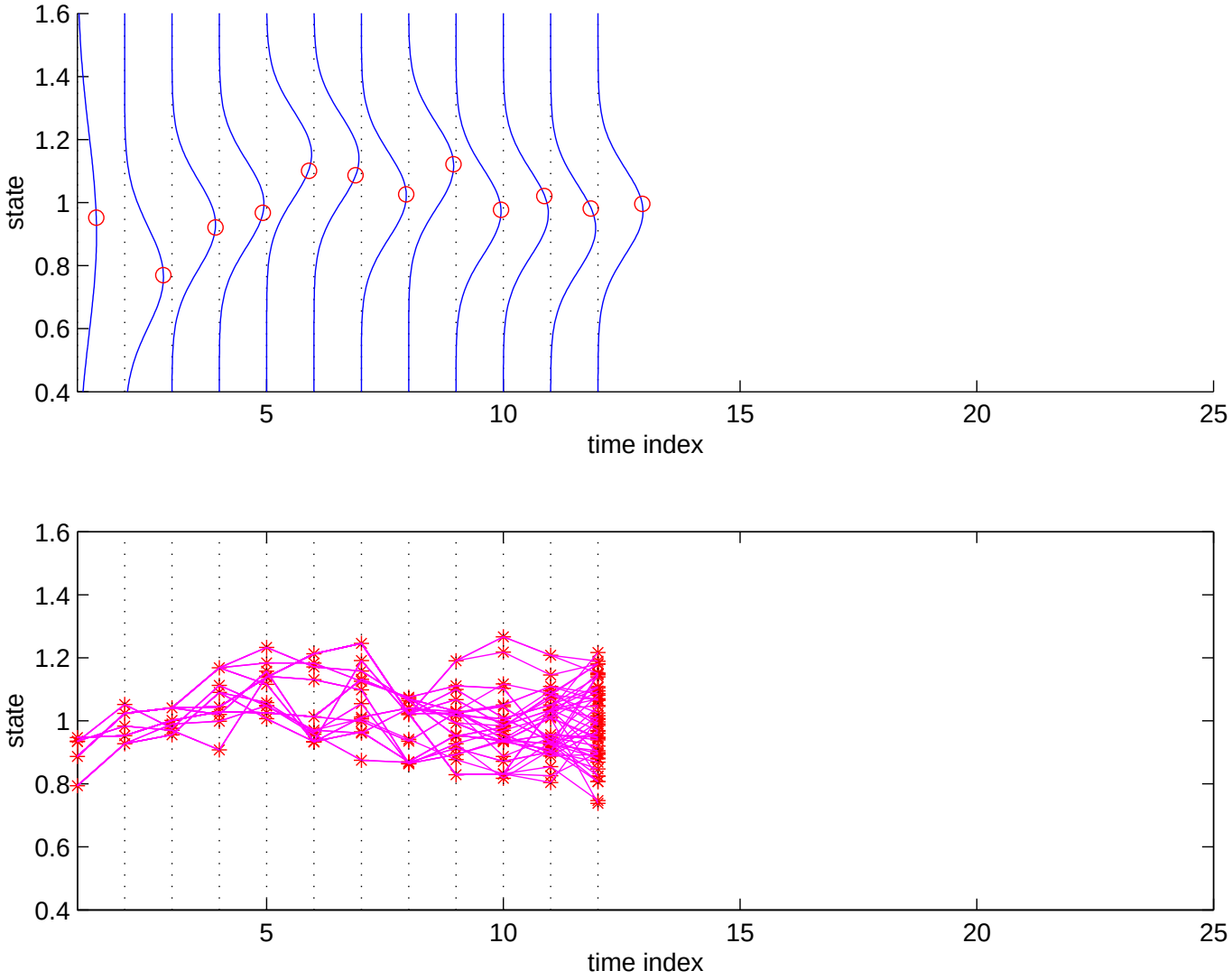
Predictive densities and evolution of the particle paths



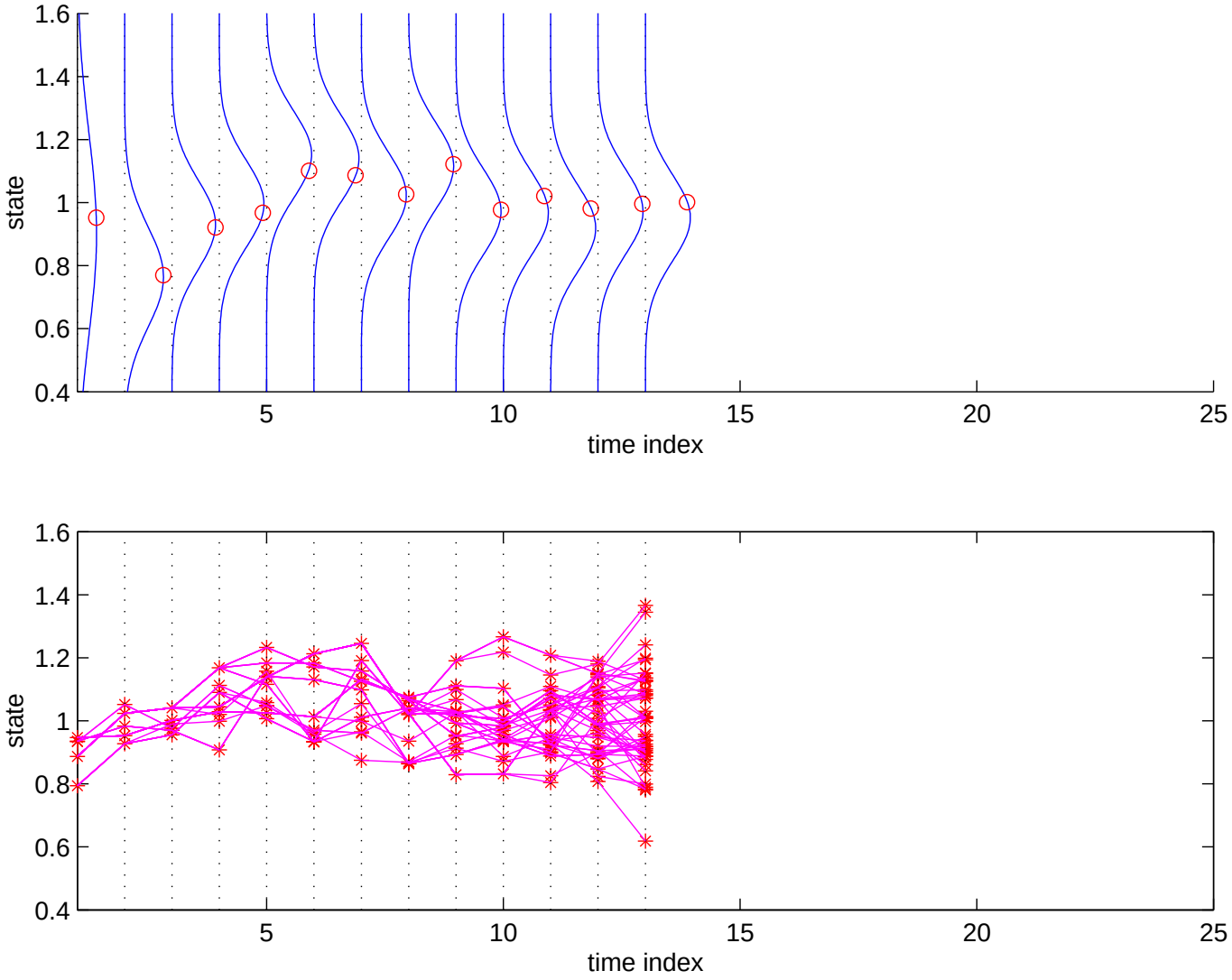
Predictive densities and evolution of the particle paths



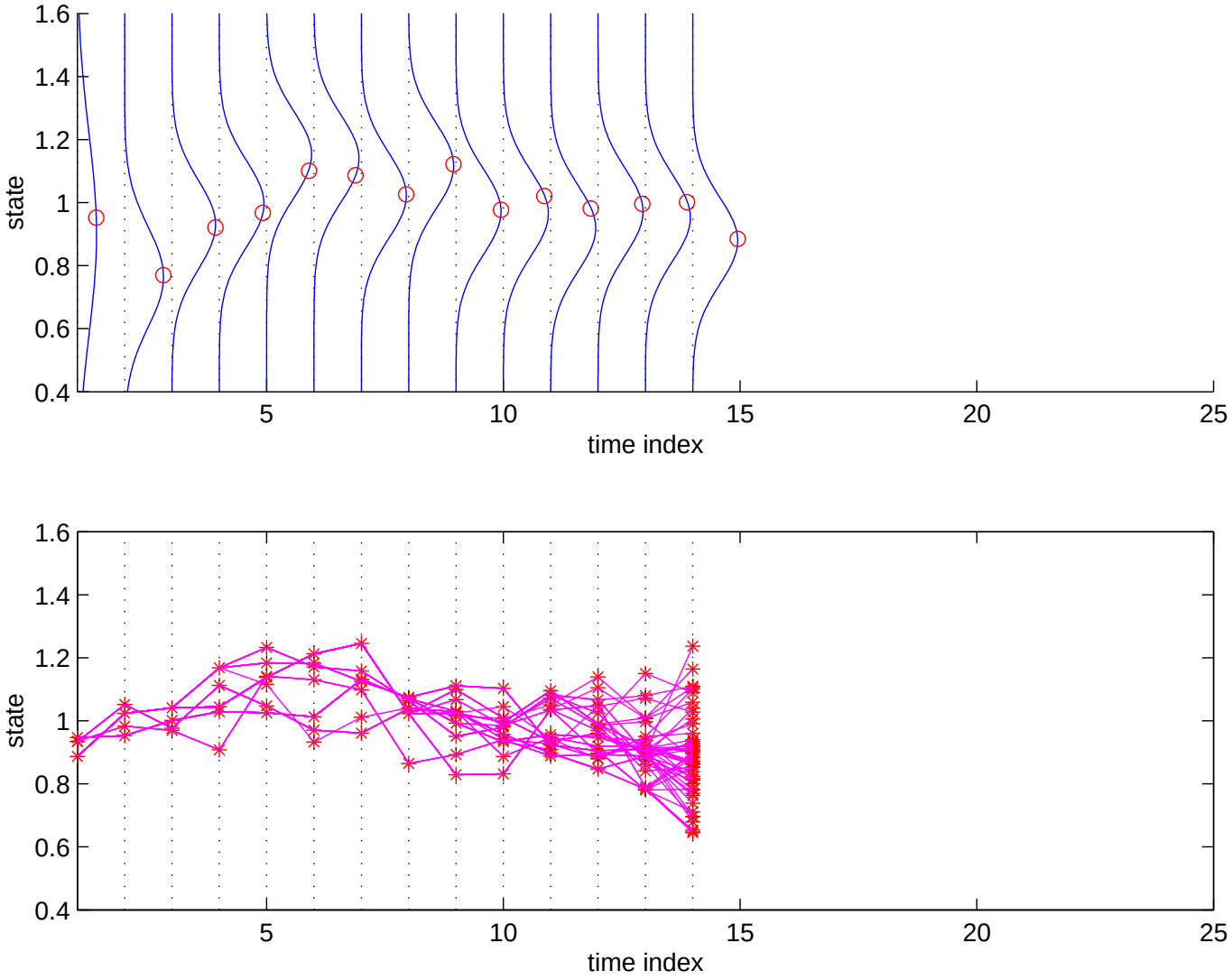
Predictive densities and evolution of the particle paths



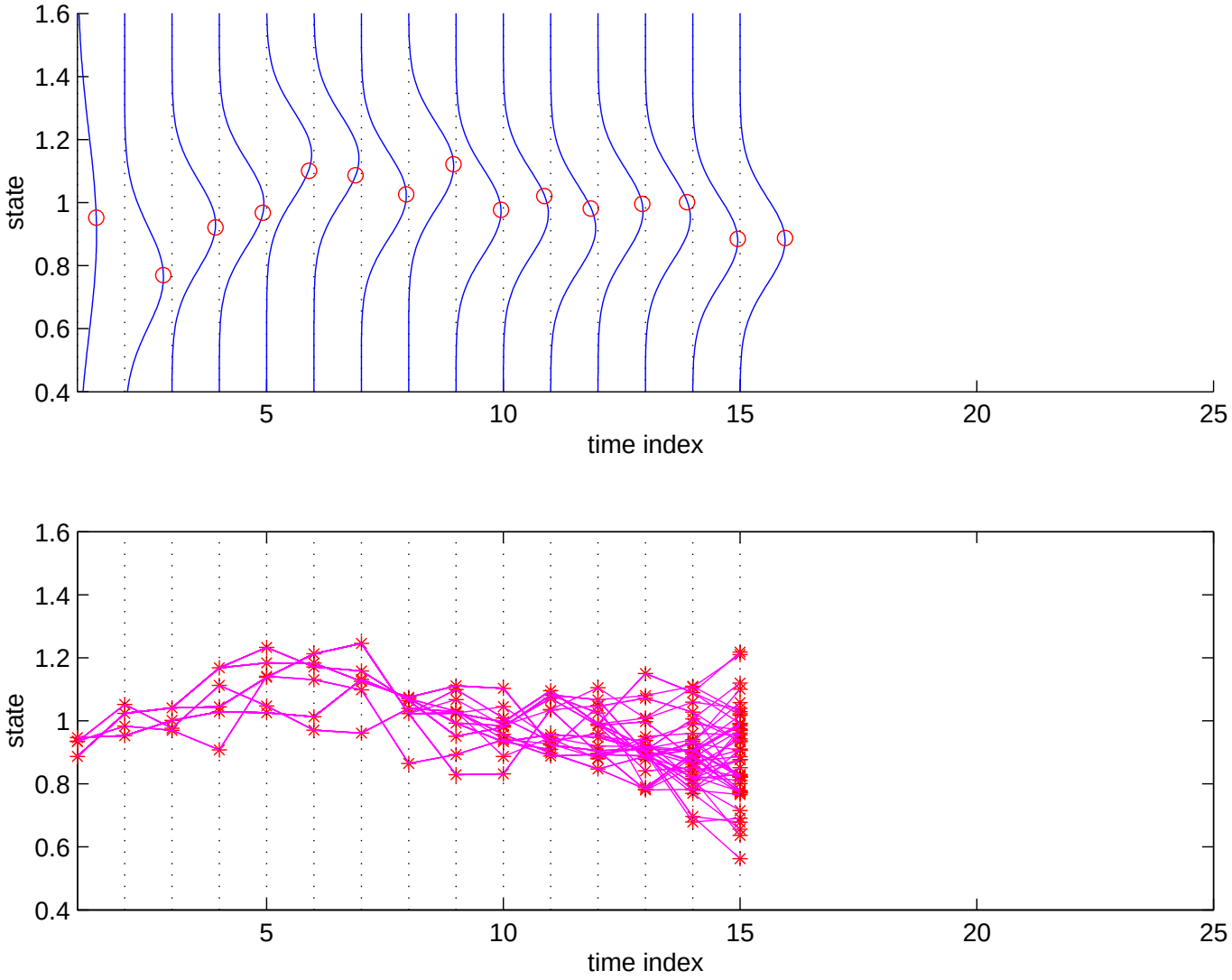
Predictive densities and evolution of the particle paths



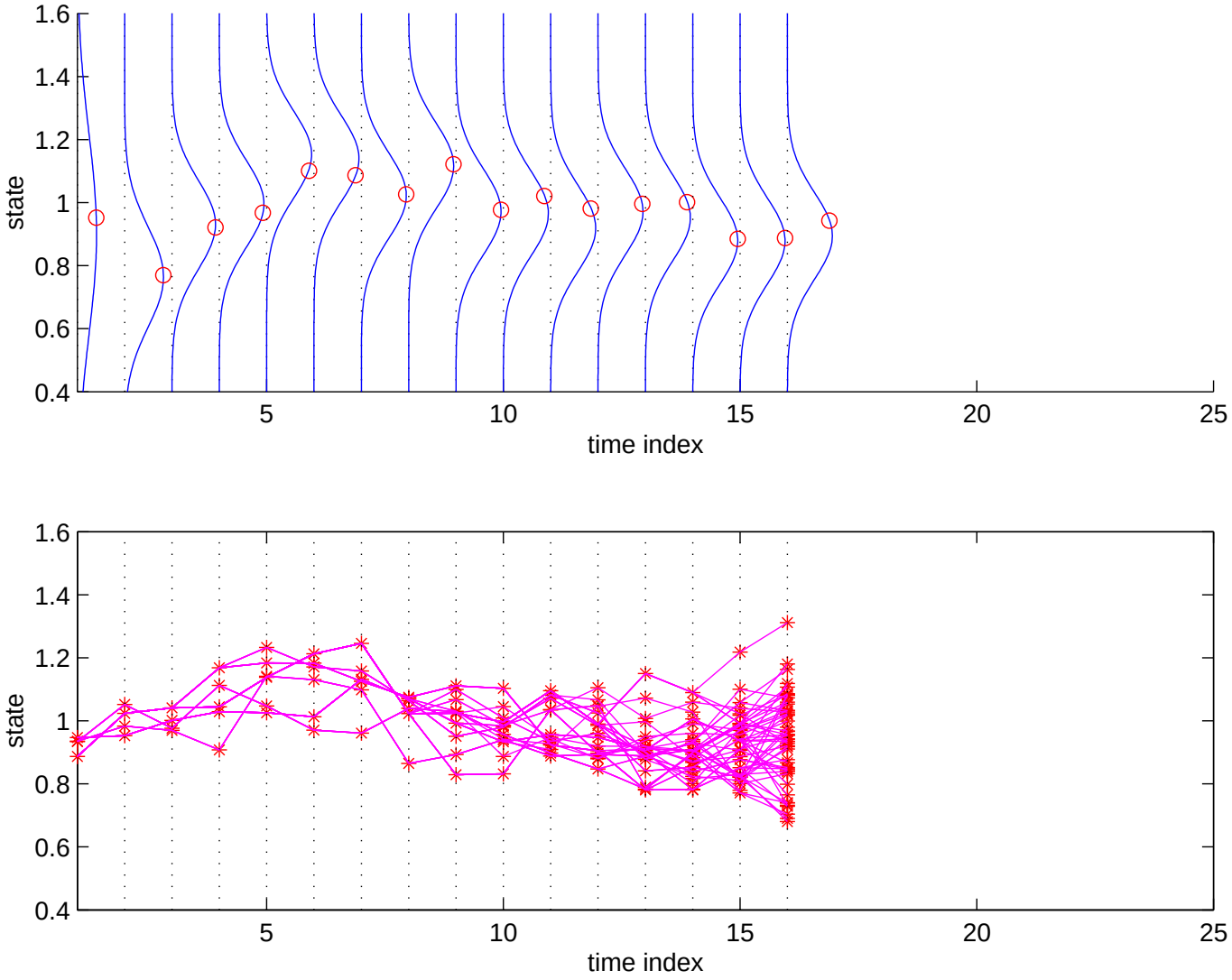
Predictive densities and evolution of the particle paths



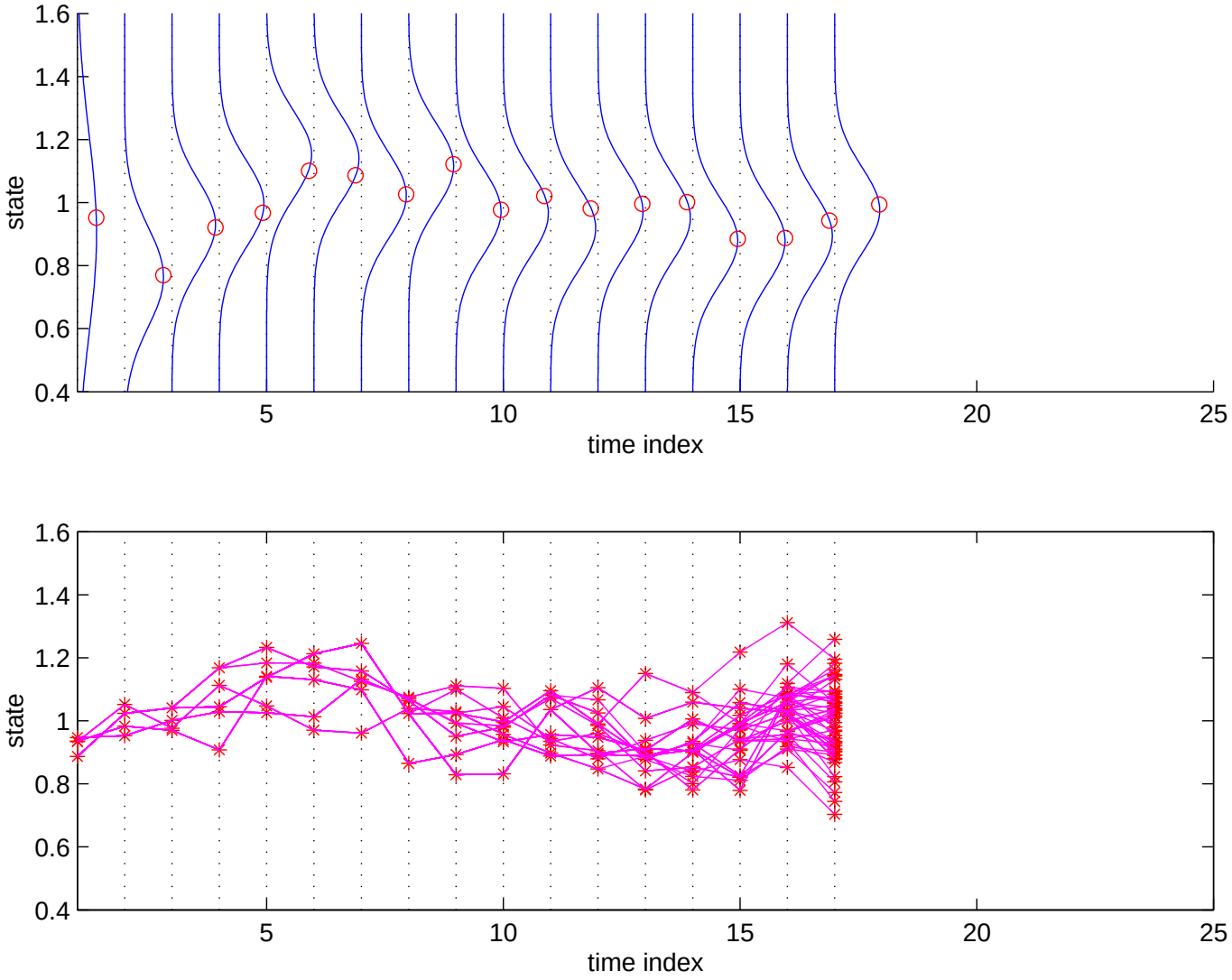
Predictive densities and evolution of the particle paths



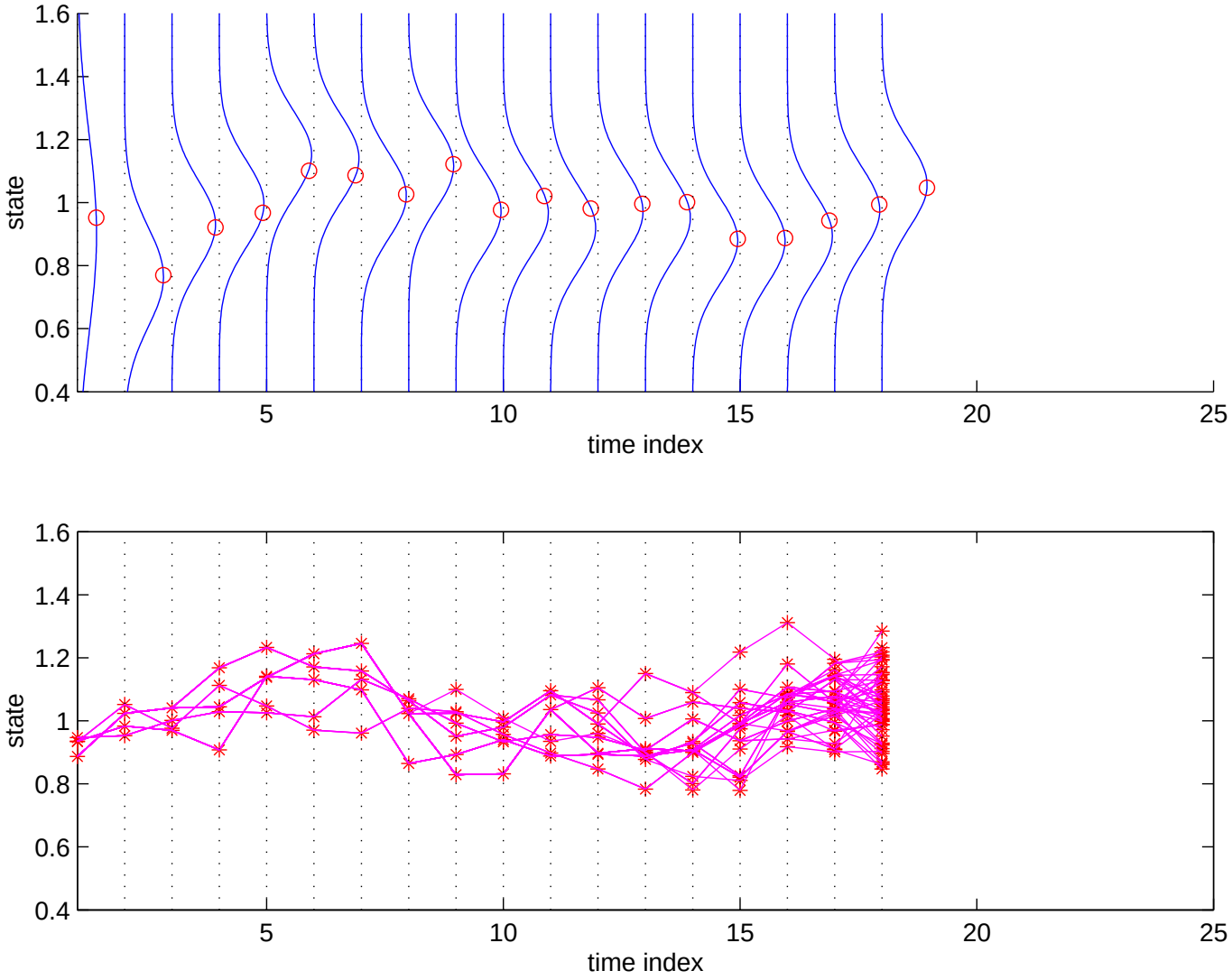
Predictive densities and evolution of the particle paths



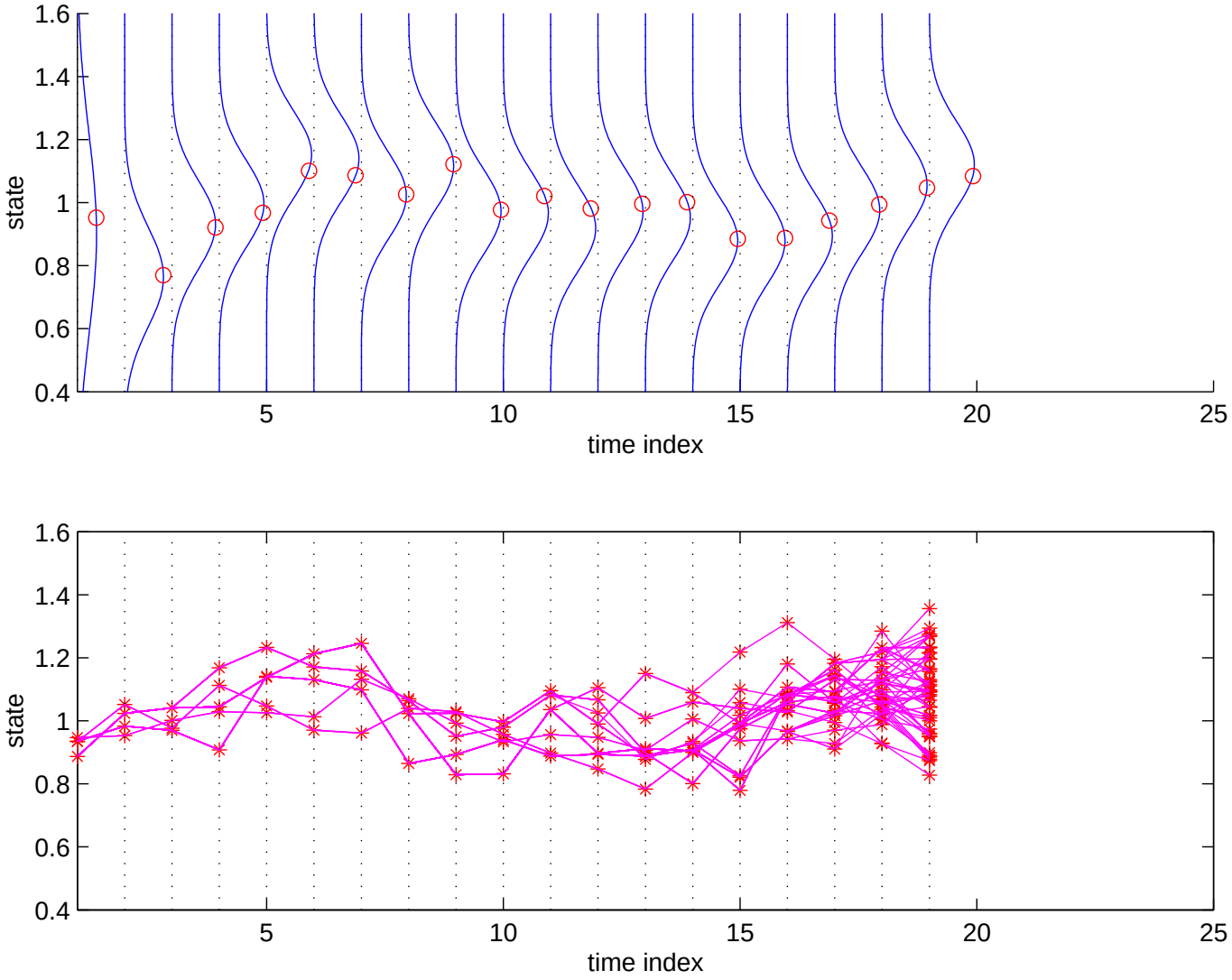
Predictive densities and evolution of the particle paths



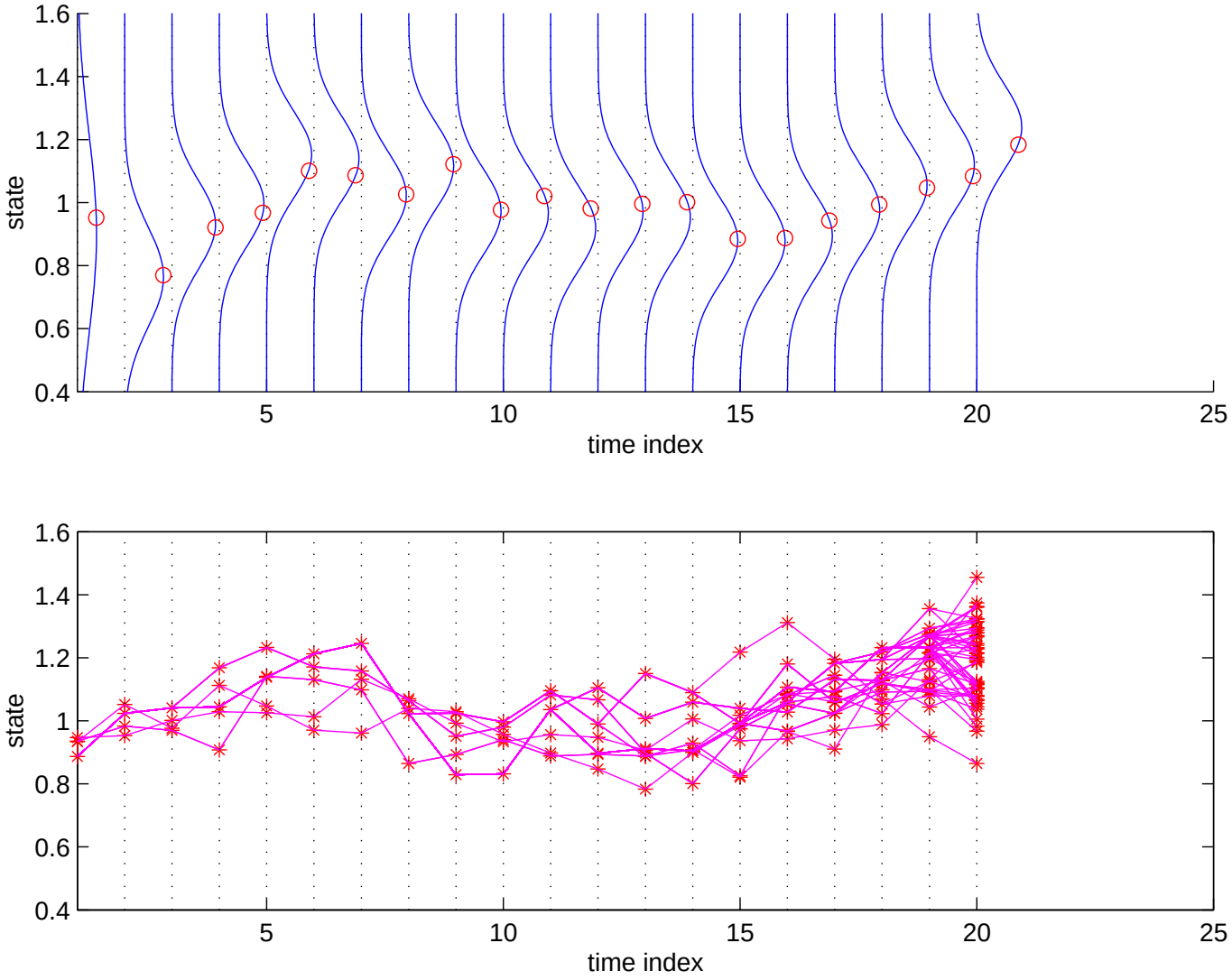
Predictive densities and evolution of the particle paths



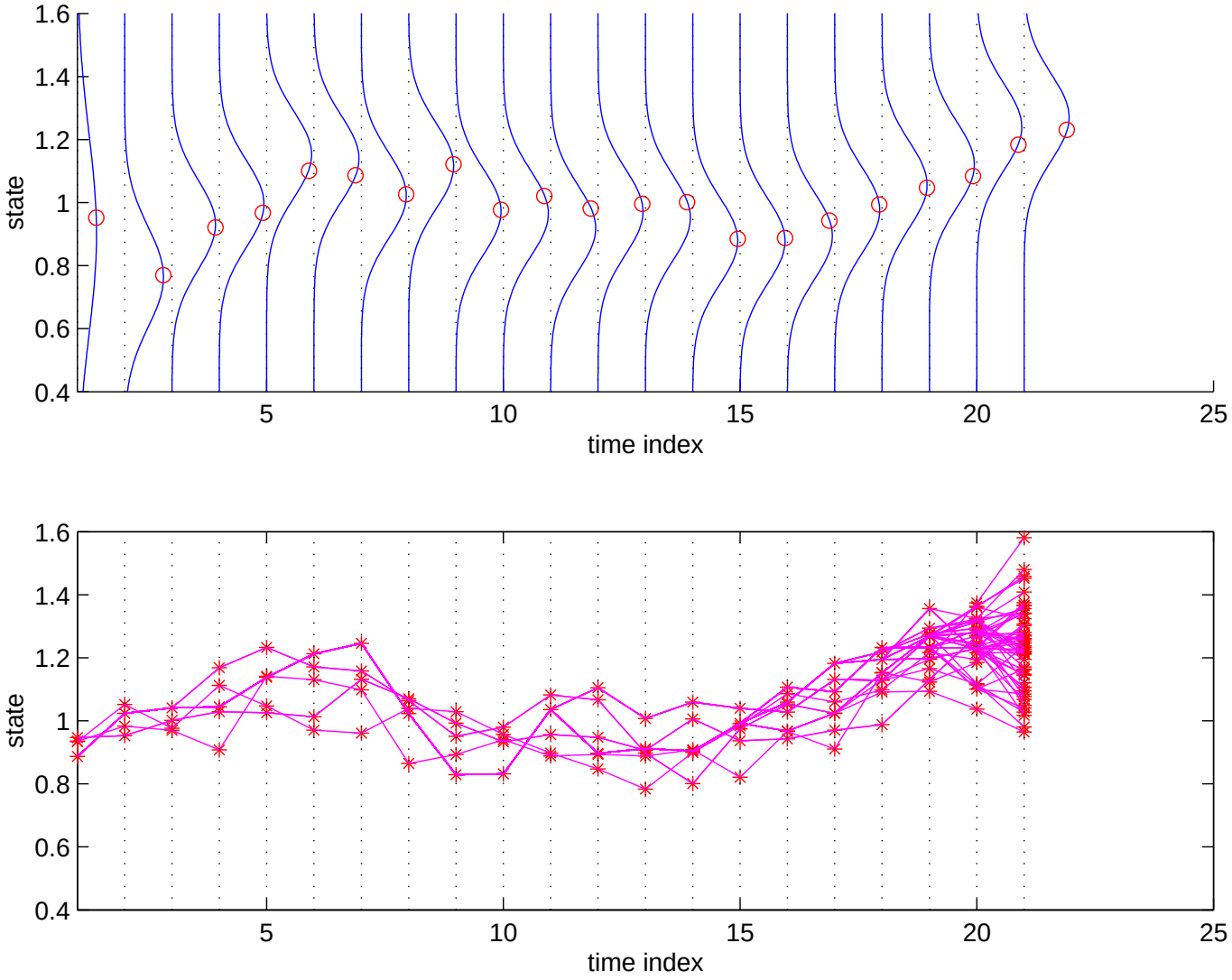
Predictive densities and evolution of the particle paths



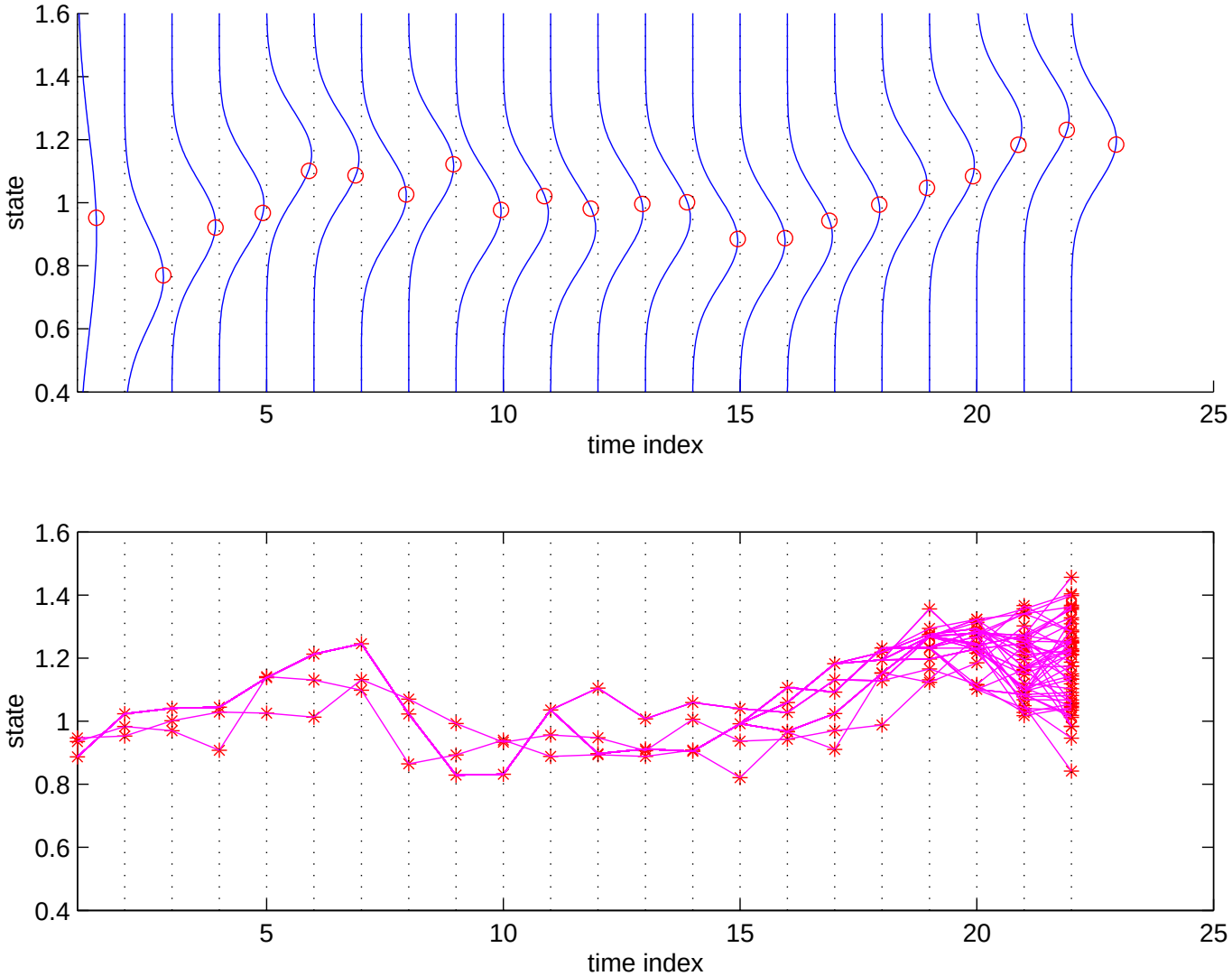
Predictive densities and evolution of the particle paths



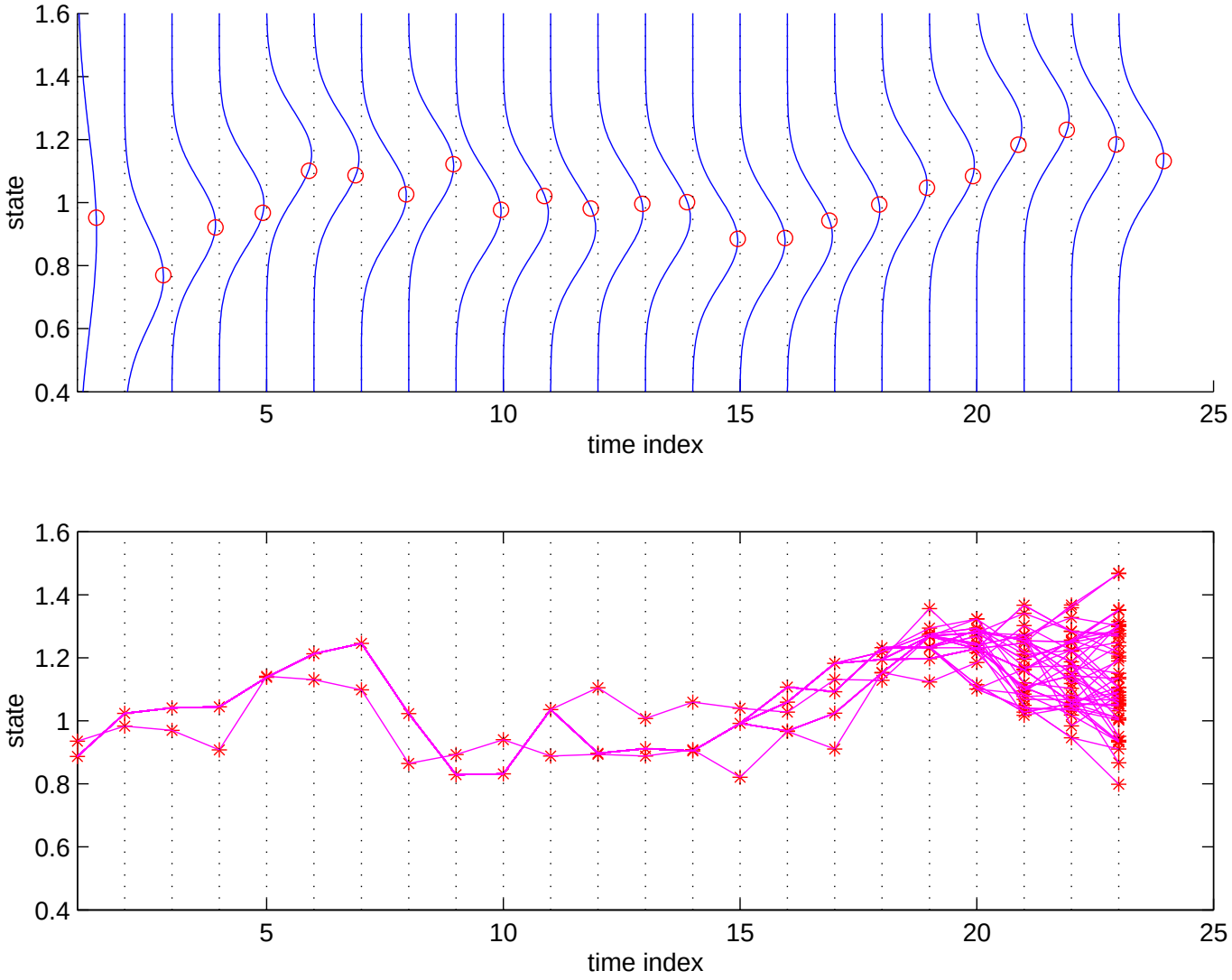
Predictive densities and evolution of the particle paths



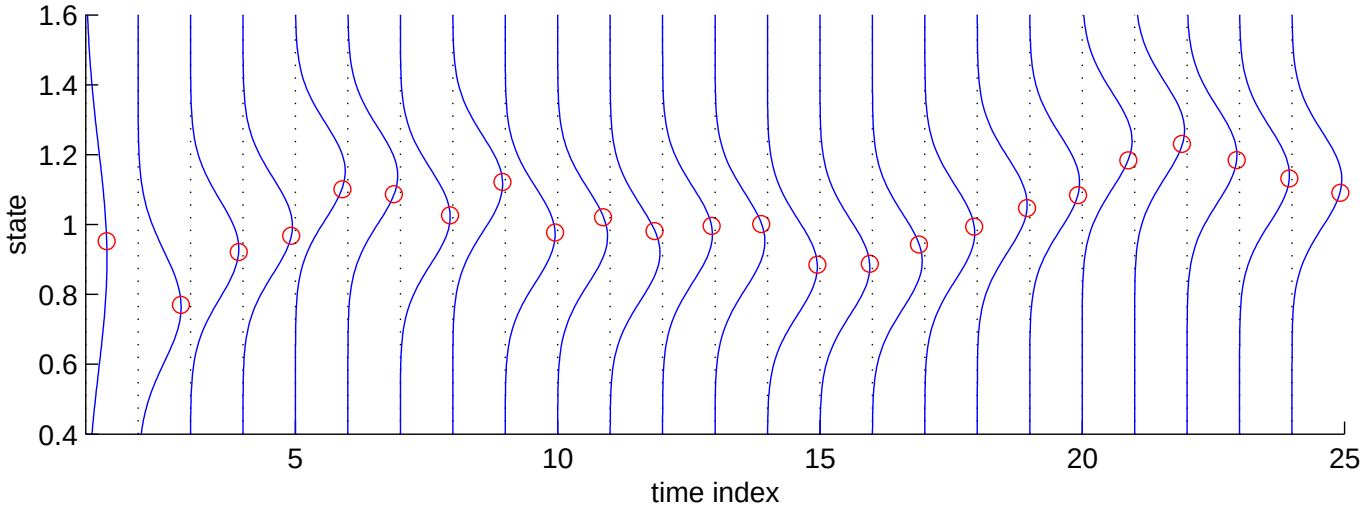
Predictive densities and evolution of the particle paths

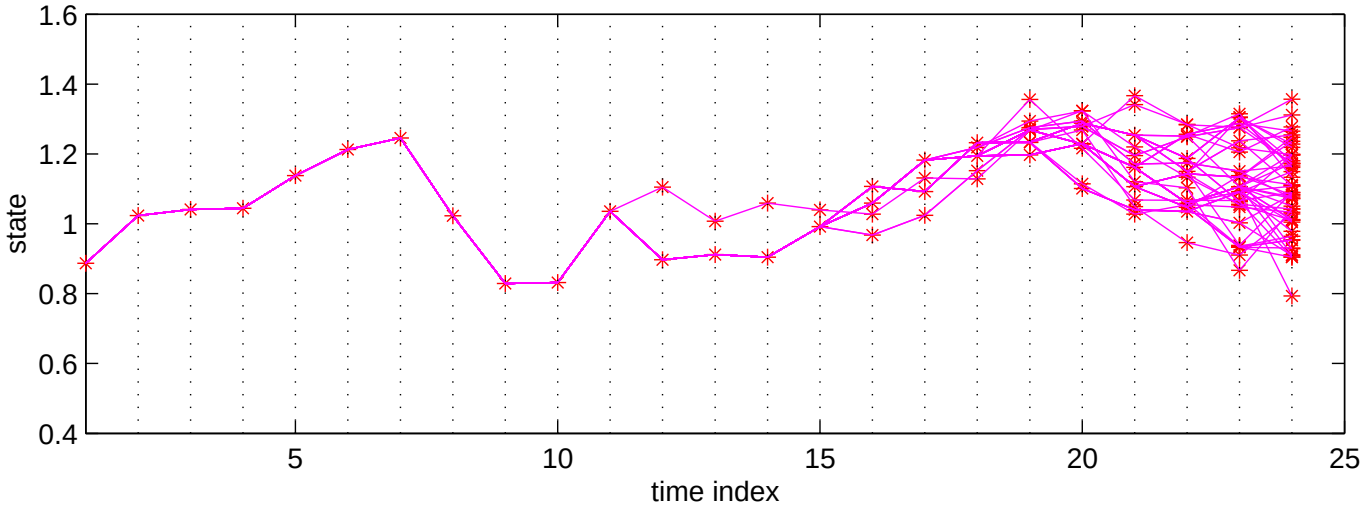


Predictive densities and evolution of the particle paths



Predictive densities and evolution of the particle paths





Predictive densities and evolution of the particle paths

More Advanced Smoothing Approaches

Several approaches have been investigated to improve smoothing estimates, many of which are based on non-sequential processing (typically forward-backward approaches).
A simple but effective approach proposed in [C et al., 2005] consists in devising particle methods that approximate

$$\sum_{l=0}^{k-1} \mathsf{E}\left[\nabla_\theta \log t_l(X_l, X_{l+1}\,;\theta) \middle| Y_{0:(l+d)\wedge k}\,;\theta\right]$$

(for a sufficiently large delay d) rather than

$$\sum_{l=0}^{k-1} \mathsf{E}\left[\nabla_\theta \log t_l(X_l, X_{l+1}\,;\theta) \middle| Y_{0:k}\,;\theta\right] \ .$$

Some Heuristic Arguments on Why the Fixed-Lag Approximation is Preferable

- The Induced Bias Is Small In models where the particle filter is stable (approximation error does not grow with time index n for fixed-dimensional functions), $\mathsf{E}\left[f(X_l, X_{l+1}) \middle| Y_{0:(l+d)\wedge n}\right]$ and $\mathsf{E}\left[f(X_l, X_{l+1}) \middle| Y_{0:n}\right]$ are rapidly getting comparable as d grows due to the forgetting properties of the smoothing distributions.
- The Variance Is Reduced For a given number N of particles, $\mathsf{E}\left[f(X_l, X_{l+1}) \middle| Y_{0:(l+d)\wedge n}\right]$ may be approximated more reliably than $\mathsf{E}\left[f(X_l, X_{l+1}) \middle| Y_{0:n}\right]$.

Stochastic Volatility Model

- The canonical discrete-time model is

$$X_{k+1} = \phi X_k + \sigma U_k \qquad U_k \sim \mathsf{N}(0,1)$$
$$Y_k = \beta \exp(X_k/2) V_k \qquad V_k \sim \mathsf{N}(0,1)$$

where

1. $\{Y_k\}_{k\geq 0}$, the observations, are the log-returns;
2. $\{X_k\}_{k\geq 0}$ is the log-volatility which is assumed to be a stationary Gaussian AR(1) process;
3. $\{U_k\}_{k\geq 0}$ and $\{V_k\}_{k\geq 0}$ are independent zero-mean unit variance white Gaussian noise.

- The parameter β is a scaling factor, ϕ is the persistence (memory) in the log-volatility and $\sigma/\sqrt{1-\phi^2}$ its scale.

Complete-Data Sufficient Statistics

To implement the EM algorithm (or gradient-based methods), one need to evaluate $\mathsf{E}\left[S_i(X_{0:n}) \middle| Y_{0:n}\,;\theta\right]$, for $0 \le i \le 4$ with

$$S_0(x_{0:n}) = x_0^2\,, \qquad S_1(x_{0:n}) = \sum_{k=0}^{n-1} x_k^2\,, \qquad S_2(x_{0:n}) = \sum_{k=1}^{n} x_k^2\,,$$

$$S_3(x_{0:n}) = \sum_{k=1}^{n} x_k x_{k-1}\,, \qquad S_4(x_{0:n}) = \sum_{k=0}^{n} Y_k^2 \exp(-x_k)\,.$$

The EM update formulas are simple functions of these quantities, e.g.,

$$\beta^* = \sqrt{\frac{s_4}{n+1}}\,.$$

Reliability of the Approximations

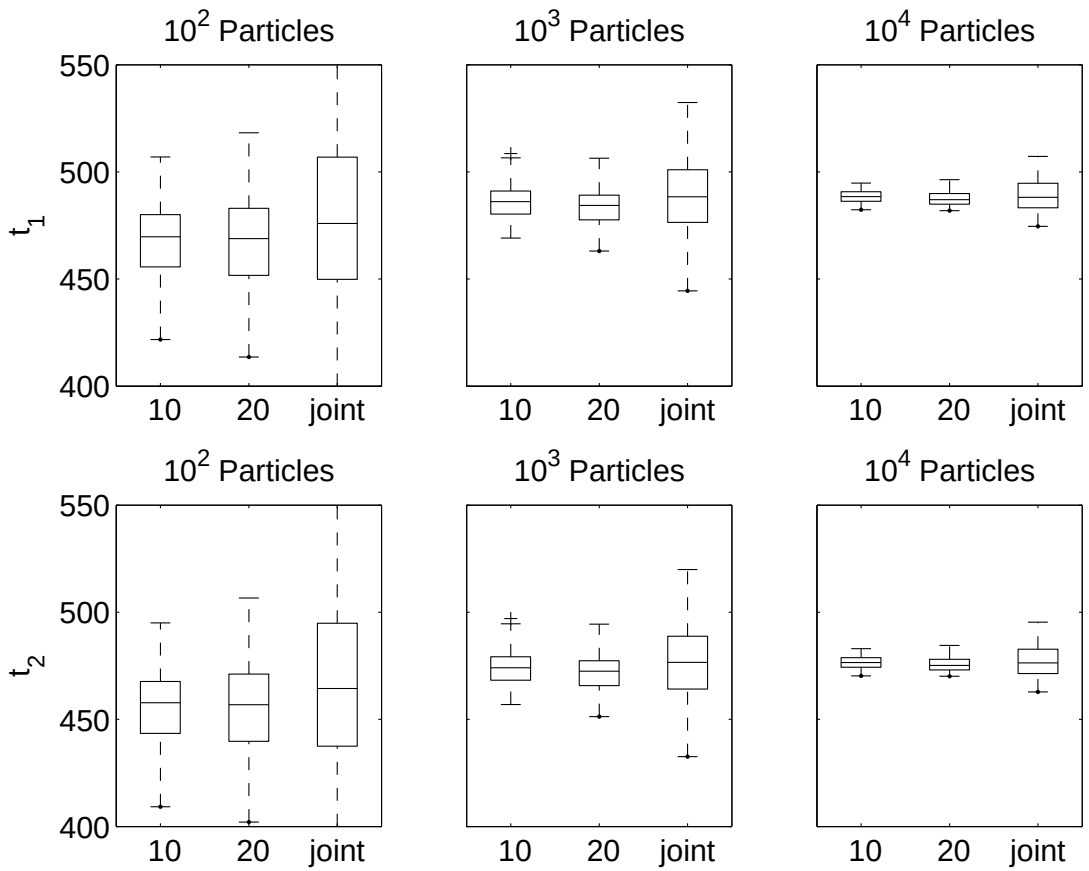


Figure: Particle estimators for $t_{n,1}(x_{0:n}) = \sum_{k=0}^{n-1} x_k^2$ (top) and $t_{n,3}(x_{0:n}) = \sum_{k=1}^{n} x_k x_{k-1}$ (bottom) for $n = 945$: from left to right, increasing particle population sizes of $N = 10^2$, 10^3, and 10^4; on each graph, fixed-lag smoothing approximation for smoothing delays $d = 10$, 20, and full-path "joint" particle approximation.

Stochastic EM (SEM) [Celeux and Diebolt, 1992]

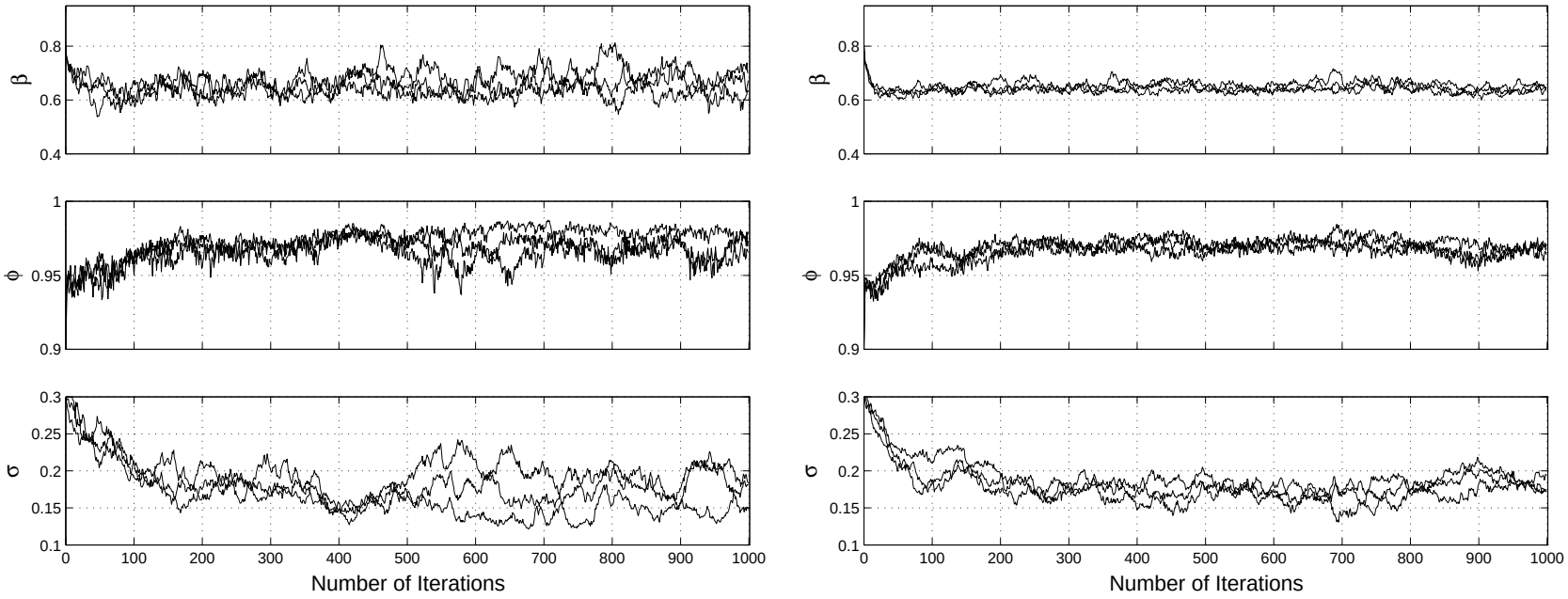


Figure: Three superimposed SEM trajectories with $N = 50$ particles for, **left:** "joint", and, **right:** "fixed-lag" ($\delta = 20$) approximations ($X_{k+1} = \phi X_k + \sigma U_k$, $\beta \exp(X_k/2) V_k$, $n = 945$, exchange rate data).

Monte Carlo EM (MCEM)[‡] [Wei and Tanner, 1991]

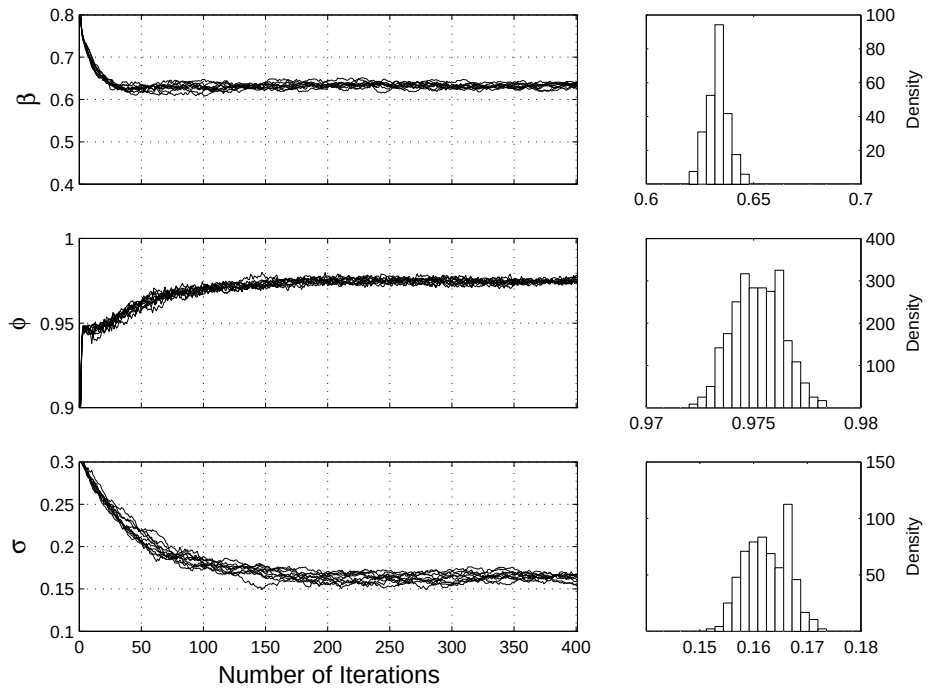


Figure: Parameter estimates with MCEM and fixed-lag approximation (delay $d = 20$): 400 iterations; the number of particles is 250 for the first 100 EM iterations, 500 for iterations 101 to 200, and then increases proportionally to the squared iteration number.

[‡] Other Option: Stochastic-Approximation EM (SAEM) [Delyon et al., 1999].

How Do We Know It Is Correct? Comparison with Brute Force Strategy using MCMC Simulations

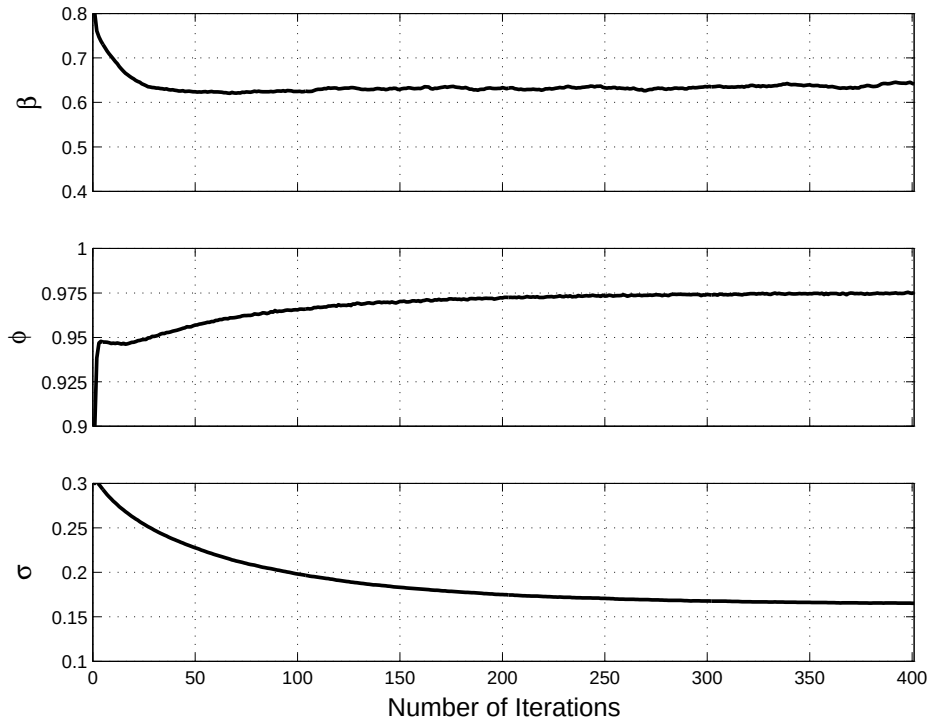


Figure: Trajectory of the MCEM algorithm for the stochastic volatility model and GBP/USD exchange rate data. In the E-step, an MCMC algorithm was used to impute the missing data. The plots show 400 EM iterations with 25,000 MCMC sweeps in each iteration.

Open issues

- Stability of approximations of $\sum_{k=0}^{n-1} \mathsf{E}\left[s(X_k, X_{k+1}) \middle| Y_{0:n}\,; \theta\right]$ as n increase (with proper normalization in n).
- Convergence of parameter estimates based on particle approximation to $\sum_{k=0}^{n-1} \mathsf{E}\left[s(X_k, X_{k+1}) \middle| Y_{0:(k+d)\wedge k}\,; \theta\right]$.
- Suitability for recursive estimation (empirically, yes, based on SAEM)?

Some References I

Baum, L. E., Petrie, T. P., Soules, G. and Weiss, N. (1970) A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains.
Ann. Math. Statist., **41**, 164–171.

Bryson, A. and Frazier, M. (1963) Smoothing for linear and nonlinear dynamic systems.
Tech. Rep. TDR 63-119, Aero. Sys. Div. Wrigth-Patterson Air Force Base.

Campillo, F. and Le Gland, F. (1989) MLE for patially observed diffusions: Direct maximization vs. the EM algorithm.
Stoch. Proc. App., **33**, 245–274.

Some References II

Cappé, O. (2001) Recursive computation of smoothed functionals of hidden Markovian processes using a particle approximation.
Monte Carlo Methods Appl., **7**, 81–92.

Cappé, O., Moulines, E. and Rydén, T. (2005) *Inference in Hidden Markov Models*.
Springer.
URL `http://www.tsi.enst.fr/~cappe/ihmm/`.

Celeux, G. and Diebolt, J. (1992) A stochastic approximation type EM algorithm for the mixture problem.
Stochastics Stochastics Rep., **41**, 119–134.

Some References III

Del Moral, P. and Guionnet, A. (2001) On the stability of interacting processes with applications to filtering and genetic algorithms.
Annales de l'Institut Henri Poincaré, **37**, 155–194.

Delyon, B., Lavielle, M. and Moulines, E. (1999) On a stochastic approximation version of the EM algorithm.
Ann. Statist., **27**.

Douc, R. and Matias, C. (2002) Asymptotics of the maximum likelihood estimator for general hidden Markov models.
Bernoulli.

Some References IV

Douc, R., Moulines, E. and Rydén, T. (2004) Asymptotic properties of the maximum likelihood estimator in autoregressive models with Markov regime.
Ann. Statist., **32**, 2254–2304.

Elliott, R. J., Aggoun, L. and Moore, J. B. (1995) *Hidden Markov models: Estimation and Control.*
Springer.

Gordon, N., Salmond, D. and Smith, A. F. (1993) Novel approach to nonlinear / non-Gaussian Bayesian state estimation.
IEE Proc. F, Radar Signal Process., **140**, 107–113.

Some References V

Gupta, N. and Mehra, R. (1974) Computational aspects of maximum likelihood estimation and reduction in sensitivity function calculations.
IEEE Trans. Automat. Control, **19**, 774–783.

Le Gland, F. and Mevel, L. (1997) Recursive estimation in HMMs.
In *Proc. IEEE Conf. Decis. Control*, 3468–3473.

Rauch, H., Tung, F. and Striebel, C. (1965) Maximum likelihood estimates of linear dynamic systems.
AIAA Journal, **3**, 1445–1450.

Some References VI

- Segal, M. and Weinstein, E. (1989) A new method for evaluating the log-likelihood gradient, the Hessian, and the Fisher information matrix for linear dynamic systems. *IEEE Trans. Inform. Theory*, **35**, 682–687.
- Stratonovich, R. L. (1960) Conditional Markov processes. *Theory Probab. Appl.*, **5**, 156–178.
- Wei, G. C. G. and Tanner, M. A. (1991) A Monte-Carlo implementation of the EM algorithm and the poor man's Data Augmentation algorithms. *J. Am. Statist. Assoc.*, **85**, 699–704.

Some References VII

Zeitouni, O. and Dembo, A. (1988) Exact filters for the estimation of the number of transitions of finite-state continuous-time Markov processes.
IEEE Trans. Inform. Theory, **34**.