# Particle Methods for Hidden Markov Models

Olivier Cappé

CNRS – Lab. Trait. Commun. Inform.

& ENST – département Trait. Signal Image

46 rue Barrault, 75634 Paris cedex 13, France

mailto:cappe@tsi.enst.fr

www.tsi.enst.fr/~cappe/

These lectures are based on the book Inference in Hidden Markov Models written with E. Moulines and T. Rydén (Springer-Verlag, to appear in 2005).

# Roadmap

1. What is a Hidden Markov Model?
2. Filtering and Smoothing Recursions
3. Monte Carlo, Importance Sampling and Sampling Importance Resampling
4. Sequential Importance Sampling
5. Sequential Importance Sampling with Resampling
6. More Sequential Monte Carlo Algorithms*
7. Approximation of Sums Functionals and Parameter Estimation*

# What is a Hidden Markov Model?

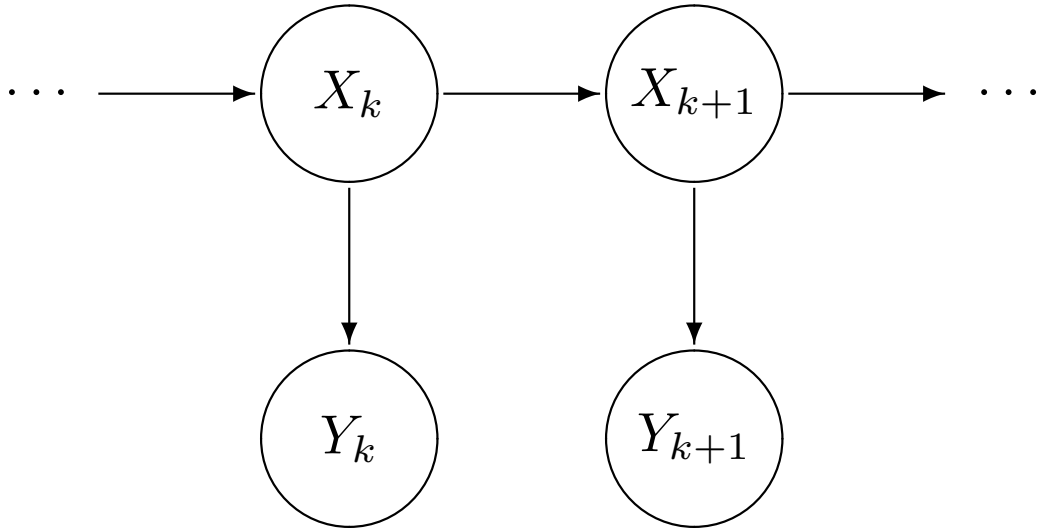
A hidden Markov model (abbreviated HMM) is a bivariate discrete-time process $\{X_k, Y_k\}_{k\geq 0}$, where

- $\{X_k\}_{k\geq 0}$ is an homogeneous Markov chain and,
- conditional on $\{X_k\}_{k\geq 0}$, $\{Y_k\}_{k\geq 0}$ is a sequence of independent random variables such that the conditional distribution of $Y_k$ only depends on $X_k$.

- The underlying Markov chain $\{X_k\}_{k\geq 0}$ is called the regime, or state.
- We denote the state space of the Markov chain $\{X_k\}_{k\geq 0}$ by $\mathsf{X}$ and the set in which $\{Y_k\}_{k\geq 0}$ takes its values by $\mathsf{Y}$.

# What is a Hidden Markov Model?

The dependence structure of an HMM can be represented by a graphical model as in




Graphical representation of the dependence structure of a hidden Markov model, where $\{Y_k\}_{k\geq 0}$ is the observable process and $\{X_k\}_{k\geq 0}$ is the hidden chain.

# What is a Hidden Markov Model?

Of the two processes $\{X_k\}_{k\geq 0}$ and $\{Y_k\}_{k\geq 0}$, only $\{Y_k\}_{k\geq 0}$ is actually observed; the Markov chain $\{X_k\}_{k\geq 0}$ is unobserved, or *hidden*.

- Hence, inference on the parameters of the model must be achieved using $\{Y_k\}_{k\geq 0}$ only.
- The other topic of interest is of course inference on the unobserved $\{X_k\}_{k\geq 0}$: given a model and some observations, can we estimate the value of the unobservable sequence of states?

These two major statistical objectives are indeed strongly connected !

# What is a Hidden Markov Model?

- the $Y$-variables are conditionally independent given $\{X_k\}_{k\geq 0}$, but $\{Y_k\}_{k\geq 0}$ is not an independent sequence because of the dependence in $\{X_k\}_{k\geq 0}$.
- $\{Y_k\}_{k\geq 0}$ is not a Markov chain either: the joint process $\{X_k, Y_k\}_{k\geq 0}$ is a Markov chain, but $\{Y_k\}_{k\geq 0}$ does not have the loss of memory property: the conditional distribution of $Y_k$ given $Y_0, \ldots, Y_{k-1}$ does depend on all the conditioning variables.

# What is a Hidden Markov Model?

There are numerous examples:

**where both $\mathsf{X}$ and $\mathsf{Y}$ are finite** coding, digital communications, bioinformatics

**where $\mathsf{X}$ is finite but $\mathsf{Y}$ is not** speech recognition, ion channel modelling (Gaussian HMMs)

**where both $\mathsf{X}$ and $\mathsf{Y}$ are continuous** linear state models, non-linear state space models
(ex. stochastic volatility model, bearings-only tracking)

**where $\mathsf{Y}$ is continuous and $\mathsf{X} = \mathsf{C} \times \mathsf{W}$ with $\mathsf{C}$ finite and $\mathsf{W}$ continuous** conditionally Gaussian linear state space models (AKA jump Markov models)

**non-HMMs that behave similarly** * switching autoregressions, Markov switching models

*Except for stability properties and theory of MLE which we don't consider today...

# Roadmap

1. What is a Hidden Markov Model?
2. Filtering and Smoothing Recursions
3. Monte Carlo, Importance Sampling and Sampling Importance Resampling
4. Sequential Importance Sampling
5. Sequential Importance Sampling with Resampling
6. More Sequential Monte Carlo Algorithms*
7. Approximation of Sums Functionals and Parameter Estimation*

# Notations for HMMs

**Hidden Markov Model**

1. $\{X_k\}_{k\geq 0}$ is a Markov chain on $\mathsf{X}$ with initial distribution $\nu$ and transition kernel $Q$
2. $\{Y_k\}_{k\geq 0}$ is such that for $f_0, \ldots, f_n \in \mathcal{F}_{\mathrm{b}}(\mathsf{Y})$,

$$\mathrm{E}\left[\prod_{k=0}^{n} f_k(Y_k) \middle| X_{0:n}\right] = \prod_{k=0}^{n} \int_{\mathsf{Y}} f_k(y)\, g(X_k, y)\mu(dy) \ ,$$

where $X_{0:n}$ denotes the collection $X_0, \ldots, X_n$ and $g$ is a transition density function (with respect to $\mu$) sometimes referred to as the *conditional likelihood function.*

We will also use the "simplified" notation

$$g_k(x) \stackrel{\mathrm{def}}{=} g(x, Y_k)$$

# Some More Notations: Usual Kernel Operations

$$Q(x, A) = \int_A Q(x, dx') \qquad \mathrm{P}[X_{k+1} \in A | X_k] = Q(X_k, A)$$

$$Q(x, f) = \int Q(x, dx') f(x') \qquad \mathrm{E}[f(X_{k+1}) | X_k] = Q(X_k, f)$$

(also denoted $(Qf)(x)$)

$$\nu Q(f) = \int \nu(dx) Q(x, dx') f(x')$$

Expectation after one step,

starting under $\nu$

$$Q^n(x_0, f) = Q^{n-1}(x_0, Qf)$$

Expectation after $n$ steps,

starting under $\delta_{x_0}$

- Markov transition kernels are such that $Q(x, \mathsf{X}) = 1$.
- Sometimes *unnormalized transition kernels*, such that $Q(x, A) \geq 0$ for all $A \in \mathcal{X}$ and $0 < Q(x, \mathsf{X}) < \infty$, are also used.

# Filtering and Smoothing Recursions

**To be answered** Given a HMM, how to evaluate the conditional distribution of the states $X_k$, given the observations $Y_0, \ldots, Y_n$? We introduce the generic notation

$$\phi_{\nu,k:l|n}$$

to denote the conditional distribution of $X_{k:l}$ given $Y_{0:n}$, where $\nu$ recalls the dependence with respect to the initial distribution (which will sometimes be omitted).

**The joint probability of the unobservable states and observations up to index** $n$ is such that, for any function $f \in \mathcal{F}_{\mathrm{b}}\left(\{\mathsf{X} \times \mathsf{Y}\}^{n+1}\right)$,

$$\mathrm{E}_\nu[f(X_0, Y_0, \ldots, X_n, Y_n)] = \int \cdots \int f(x_0, y_0, \ldots, x_n, y_n)$$
$$\times \nu(dx_0) g(x_0, y_0) \prod_{k=1}^{n} \{Q(x_{k-1}, dx_k) g(x_k, y_k)\} \mu_n(dy_0, \ldots, dy_n) \ .$$

# The Likelihood

Marginalizing with respect to the unobservable variables $X_0, \ldots, X_n$ yields

$$\mathrm{E}_\nu[f(Y_0, \ldots, Y_n)] = \int \cdots \int f(y_0, \ldots, y_n)\, \mathrm{L}_{\nu,n}(y_0, \ldots, y_n)\, \mu_n(dy_0, \ldots, dy_n) \ ,$$

for $f \in \mathcal{F}_\mathrm{b}\left(\mathsf{Y}^{n+1}\right)$, where

$$\mathrm{L}_{\nu,n}(y_0, \ldots, y_n) = \int \cdots \int \nu(dx_0) g(x_0, y_0) Q(x_0, dx_1) g(x_1, y_1) \cdots Q(x_{n-1}, dx_n) g(x_n, y_n) \ ,$$

is the likelihood of the observations.

# Joint Smoothing Distribution

**By Bayes' rule**

$$\phi_{\nu,0:n|n}(y_{0:n}, f) = \mathrm{L}_{\nu,n}(y_{0:n})^{-1} \int \cdots \int f(x_{0:n}) \times \nu(dx_0) g(x_0, y_0) \prod_{k=1}^{n} Q(x_{k-1}, dx_k) g(x_k, y_k)$$

for all functions $f \in \mathcal{F}_{\mathrm{b}}\left(\mathsf{X}^{n+1}\right)$.

**In the following,** we always use the implicit conditioning convention, writing

$$\phi_{\nu,0:n|n}(f) = \mathrm{L}_{\nu,n}^{-1} \int \cdots \int f(x_{0:n}) \nu(dx_0) g_0(x_0) \prod_{k=1}^{n} Q(x_{k-1}, dx_k) g_k(x_k)$$

where

$$\mathrm{L}_{\nu,n} = \int \cdots \int \nu(dx_0) g_0(x_0) \prod_{k=1}^{n} Q(x_{k-1}, dx_k) g_k(x_k) \ .$$

# Recursive Smoothing Formula

Comparing the expressions corresponding to $n$ and $n+1$ gives the following update equation for the joint smoothing distribution:

$$\phi_{\nu,0:n+1|n+1}(f_{n+1}) = \left(\frac{\mathrm{L}_{n+1}}{\mathrm{L}_n}\right)^{-1} \int \cdots \int f_{n+1}(x_{0:n+1}) \\ \phi_{\nu,0:n|n}(dx_0, \ldots, dx_{n-1}, dx_n)\, Q(x_n, dx_{n+1})\, g_{n+1}(x_{n+1})$$

for functions $f_{n+1} \in \mathcal{F}_{\mathrm{b}}\left(\mathsf{X}^{n+2}\right)$.

$\Longrightarrow$ Very simple structure but involves the normalization factor $c_{n+1} \stackrel{\mathrm{def}}{=} \mathrm{L}_{n+1}/\mathrm{L}_n$ which is not computable, except in simple cases such as when $\mathsf{X}$ finite*

*This claim is not obvious (see next slides...)

# Filtering Recursion

Marginalizing with respect to all variables but $x_n$ and $x_{n+1}$ gives the (marginal) filtering recursion:

$$c_{\nu,n+1} = \int \int \phi_{\nu,n|n}(dx) Q(x, dx') g_{n+1}(x') \ ,$$

$$\phi_{\nu,n+1|n+1}(f) = c_{\nu,n+1}^{-1} \int f(x) \int \phi_{\nu,n}(dx) Q(x, dx') g_{n+1}(x') \ ,$$

with initial condition

$$c_{\nu,0} = \nu(g_0) \ ,$$

$$\phi_{\nu,0|0}(f) = c_{\nu,0}^{-1} \int f(x) g_0(x) \, \nu(dx) \ .$$

**Remark** When $\mathsf{X}$ is finite (speech recognition, bioinformatics) the above is known as the normalized forward recursion (of forward-backward); the specialization of this relation to Gaussian linear state-space model is known as Kalman filtering.

# Prediction and Filtering Updates

It is sometimes convenient to break the previous recursion in two steps:

$$\phi_{\nu,n+1|n} = \phi_{\nu,n|n} Q \ . \qquad \text{prediction}$$

$$c_{\nu,n+1} = \phi_{\nu,n+1|n}(g_{n+1}) \ ,$$

$$\phi_{\nu,n+1|n+1}(f) = c_{\nu,n+1}^{-1} \int f(x)\, g_{n+1}(x) \phi_{\nu,n+1|n}(dx) \ . \qquad \text{filtering}$$

# Computation of the Log-Likelihood

$$\ell_{\nu,n} \stackrel{\text{def}}{=} \log \mathrm{L}_{\nu,n} = \sum_{k=0}^{n} \log \phi_{\nu,k|k-1}(g_k) \ .$$

This is non-trivial: we have replaced an $n+1$ dimensional integral by a product of $n+1$ integrals on $\mathsf{X}$! In finite state space HMMs, the filtering recursion makes it possible to evaluate the (log-)likelihood in $O\{(n+1) \times \mathrm{Card}^2(\mathsf{X})\}$ operations.

# Recap: Filtering and Smoothing

The recursion

$$\begin{aligned}
\phi_{\nu,n+1|n} &= \phi_{\nu,n|n} Q \ , \\
c_{\nu,n+1} &= \phi_{\nu,n+1|n}(g_{n+1}) \ , \\
\phi_{\nu,n+1|n+1}(f) &= c_{\nu,n+1}^{-1} \int f(x)\, g_{n+1}(x) \phi_{\nu,n+1|n}(dx) \ ,
\end{aligned}$$

with $\phi_{\nu,0|-1} \stackrel{\text{def}}{=} \nu$ computes the filtering and predictive distributions recursively, making it possible *(i)* to compute the likelihood $\mathrm{L}_{\nu,n+1}$ and, potentially, *(ii)* the joint smoothing distribution since

$$\begin{aligned}
\phi_{0:n+1|n+1}(f_{n+1}) = c_{\nu,n+1}^{-1} \int \cdots \int & f_{n+1}(x_{0:n+1}) \\
& \phi_{0:n|n}(dx_0, \ldots, dx_{n-1}, dx_n)\, Q(x_n, dx_{n+1})\, g_{n+1}(x_{n+1}) \ .
\end{aligned}$$

# Appendix: Finite-Dimensional Recursive Smoothing for a Sum

In particular, if $f_n(x_{0:n}) = \sum_{k=0}^{n} s(x_k)$, define the *signed* measure $\tau_{\nu,n}$ by

$$\tau_{\nu,n}(f) = \int \cdots \int f(x_n) \left( \sum_{k=0}^{n} s(x_k) \right) \phi_{\nu,0:n|n}(dx_0, \ldots, dx_n) \ ,$$

such that $\tau_{\nu,n}(\mathsf{X}) = \mathrm{E}_\nu[\sum_{k=0}^{n} s(X_k)|Y_{0:n}]$. Then,

$$\begin{aligned}
&\tau_{\nu,n+1}(f) \\
&\quad = c_{\nu,n+1}^{-1} \int \cdots \int f(x_{n+1}) \left( \sum_{k=0}^{n+1} s(x_k) \right) \\
&\qquad \phi_{\nu,0:n|n}(dx_0, \ldots, dx_{n-1}, dx_n)\, Q(x_n, dx_{n+1})\, g_{n+1}(x_{n+1}) \\
&\quad = \int f(x_{n+1}) \\
&\qquad \left( \phi_{\nu,n+1|n+1}(dx_{n+1})\, s(x_{n+1}) + c_{n+1}^{-1} \int \tau_{\nu,n}(dx_n)\, Q(x_n, dx_{n+1})\, g_{n+1}(x_{n+1}) \right) \ .
\end{aligned}$$

# Roadmap

1. What is a Hidden Markov Model?
2. Filtering and Smoothing Recursions
3. Monte Carlo, Importance Sampling and Sampling Importance Resampling
4. Sequential Importance Sampling
5. Sequential Importance Sampling with Resampling
6. More Sequential Monte Carlo Algorithms*
7. Approximation of Sums Functionals and Parameter Estimation*

# Monte-Carlo Integration

**Objective** Given a probability measure $\mu$, how to evaluate numerically $\mu(f) = \int_{\mathsf{X}} \mu(dx) f(x)$ for arbitrary $\mu$-integrable functions $f$?

**The Monte Carlo Answer**

1. Draw an independent sample $\xi^1, \ldots, \xi^N$ from the probability measure $\mu$.
2. Compute the sample average

$$N^{-1} \sum_{i=1}^{N} h(\xi^i) \ .$$

This technique is applicable only when direct sampling from the distribution $\mu$ is feasible.

# (Unnormalized) Importance Sampling: General Principle

It is also possible to sample from an instrumental (or importance) distribution $\nu$, applying a change-of-measure formula to account for the fact that the instrumental distribution differs from the target distribution $\mu$.

More formally, if the target probability measure $\mu$ is absolutely continuous with respect to to the instrumental probability measure $\nu$,

$$\mu \ll \nu \ .$$

For any $\mu$-integrable function $f$

$$\mu(f) = \int f(x)\, \mu(dx) = \int f(x)\, \frac{d\mu}{d\nu}(x) \nu(dx) \ ,$$

where $\frac{d\mu}{d\nu}$ is the Radon-Nikodym derivative of $\mu$ with respect to $\nu$, called the importance function (or importance ratio) in the context of importance sampling.

# (Unnormalized) Importance Sampling: the Algorithm

**Sampling** Draw an independent sample $\xi^1, \ldots, \xi^N$ from the distribution $\nu$.

**Weighting** Compute the importance weights

$$\omega^i = \frac{d\mu}{d\nu}(\xi^i) \ ,$$

for $i = 1, \ldots, N$.

**Weigthed Monte Carlo Approximation**

$$\tilde{\mu}^{\mathrm{IS}}_{\nu,N}(f) = N^{-1} \sum_{i=1}^{N} \omega^i f(\xi^i)$$

# (Unnormalized) Importance Sampling: Large Sample Performance

**Strong law of large numbers** The sequence $\tilde{\mu}^{\mathrm{IS}}_{\nu,N}(f)$ converges to $\mu(f)$, almost surely as $N \to \infty$.

**Central limit theorem** If $f$ is a real-valued measurable function satisfying

$$\nu\left((1+f^2)\left(\frac{d\mu}{d\nu}\right)^2\right) = \mu\left((1+f^2)\frac{d\mu}{d\nu}\right) < \infty\ ,$$

$\tilde{\mu}^{\mathrm{IS}}_{\nu,N}(f)$ is asymptotically normal $\sqrt{N}(\tilde{\mu}^{\mathrm{IS}}_{\nu,N}(f) - \mu(f)) \xrightarrow{\mathcal{D}} \mathrm{N}\left(0, \mathrm{Var}_\nu\left(\frac{d\mu}{d\nu} f\right)\right)$ where

$$\mathrm{Var}_\nu\left(\frac{d\mu}{d\nu} f\right) = \nu\left(\left\{f\frac{d\mu}{d\nu} - \mu(f)\right\}^2\right).$$

Deviations inequalities (exponential, $L^p$) or more sophisticated empirical process results are also available.

$\Longrightarrow$ Choosing $\nu$ such that $d\mu/d\nu$ stays as small as possible is very important in practice.

# Importance Sampling

In situations where $\frac{d\mu}{d\nu}$ is known only up to a scaling factor we can still use the importance sampling estimator, just changing the normalization factor

$$\widehat{\mu}^{\mathrm{IS}}_{\nu,N}(f) = \frac{\sum_{i=1}^{N} f(\xi^i)\frac{d\mu}{d\nu}(\xi^i)}{\sum_{i=1}^{N}\frac{d\mu}{d\nu}(\xi^i)},$$

- The (self-normalized) importance sampling estimator (sometimes also called Bayesian sampling estimator) is defined as a ratio of the unnormalized importance sampling estimators

$$\widehat{\mu}^{\mathrm{IS}}_{\nu,N}(f) = \frac{\tilde{\mu}^{\mathrm{IS}}_{\nu,N}(f)}{\tilde{\mu}^{\mathrm{IS}}_{\nu,N}(1)}\ .$$

- By the Strong Law of Large Numbers

$$\tilde{\mu}^{\mathrm{IS}}_{\nu,N}(f) \xrightarrow{\text{a.s.}} \mu(f)$$
$$\tilde{\mu}^{\mathrm{IS}}_{\nu,N}(1) \xrightarrow{\text{a.s.}} 1$$

showing that $\widehat{\mu}^{\mathrm{IS}}_{\nu,N}(f)$ is a strongly consistent estimator of $\mu(f)$.

# Importance Sampling (contd.)

Assuming in addition that $f$ is real-valued and satisfies

$$\nu\left((1+f^2)\left(\frac{d\mu}{d\nu}\right)^2\right) = \mu\left((1+f^2)\frac{d\mu}{d\nu}\right) < \infty\ ,$$

$$\sqrt{N}(\widehat{\mu}^{\mathrm{IS}}_{\nu,N}(f) - \mu(f)) \xrightarrow{\mathcal{D}} \mathrm{N}\left(0, \sigma^2(\nu, h)\right)\ ,$$

$$\sigma^2(\nu, f) = \mathrm{Var}_\nu\left(\frac{d\mu}{d\nu}\{f - \mu(f)\}\right) = \nu\left(\left(\frac{d\mu}{d\nu}\right)^2 (f - \mu(f))^2\right).$$

The estimator is errorless for constant functions and its performance is clearly dependent on the fact that $d\mu/d\nu$ stays small.

# Sampling Importance Resampling (SIR)

While importance sampling is originally designed to overcome difficulties with direct sampling from $\mu$ when approximating integrals like $\mu(f)$ it can also be used for approximate sampling from the distribution $\mu$.

The sampling importance resampling (SIR) method is a two-stages method:

**Sampling:** Draw an i.i.d. sample $\tilde{\xi}^1, \ldots, \tilde{\xi}^M$ from the instrumental distribution $\nu$.

**Weighting:** Compute the (normalized) importance weights $\omega^i = \frac{d\mu}{d\nu}(\tilde{\xi}^i) \Big/ \sum_{j=1}^M \frac{d\mu}{d\nu}(\tilde{\xi}^j)$ for $i = 1, \ldots, M$ .

**Resampling:**

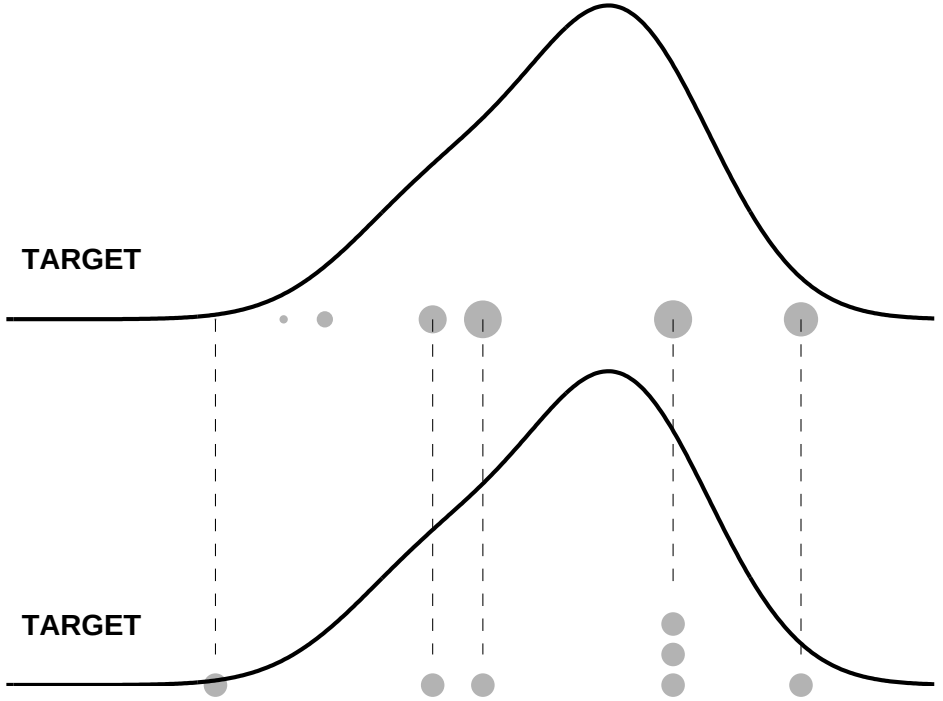
- Draw, conditionally independently given $(\tilde{\xi}^1, \ldots, \tilde{\xi}^M)$, $N$ discrete random variables $(I^1, \ldots, I^N)$ taking values in the set $\{1, \ldots, M\}$ with probabilities $(\omega^1, \ldots, \omega^M)$.
- Set, for $i = 1, \ldots, N$, $\xi^i = \tilde{\xi}^{I^i}$.

The set $(I^1, \ldots, I^N)$ is thus a multinomial trial process. This resampling method is known as *multinomial resampling*.

# Sampling Importance Resampling (contd.)

The first stage sample $\tilde{\xi}^1, \ldots, \tilde{\xi}^M$ is really distributed under $\nu$.

In the resampling operation, the bad points, as measured by $d\mu/d\nu$, are discarded whereas the good points are selected (and perhaps duplicated) with high probability.



# SIR: Large Sample Behavior

It is not obvious in which sense $(\xi^1, \ldots, \xi^N)$ is (approximately) a sample from the target distribution $\mu$. Rewriting

$$\hat{\mu}^{\mathrm{SIR}}_{\nu,M,N}(f) = \frac{1}{N}\sum_{i=1}^{N} f(\xi^i) = \sum_{i=1}^{M} \frac{N^i}{N} f(\tilde{\xi}^i) \ ,$$

it is easily seen that the sample mean $\hat{\mu}^{\mathrm{SIR}}_{\nu,M,N}(f)$ of the SIR sample is, conditionally on the first-stage sample $(\tilde{\xi}^1, \ldots, \tilde{\xi}^M)$, equal to the importance sampling estimator $\widehat{\mu}^{\mathrm{IS}}_{\nu,M}(f)$:

$$\mathrm{E}\left[\hat{\mu}^{\mathrm{SIR}}_{\nu,M,N}(f) \,\middle|\, \tilde{\xi}^1, \ldots, \tilde{\xi}^M\right] = \widehat{\mu}^{\mathrm{IS}}_{\nu,M}(f) \ .$$

As a consequence the SIR estimator $\hat{\mu}^{\mathrm{SIR}}_{\nu,M,N}(f)$ is an unbiased estimate of $\mu(f)$, but its mean squared error is always larger than that of the importance sampling estimator due to the well-known variance decomposition

$$\begin{aligned}\mathrm{E}\left[(\hat{\mu}^{\mathrm{SIR}}_{\nu,M,N}(f) - \mu(f))^2\right] \\ = \mathrm{E}\left[(\hat{\mu}^{\mathrm{SIR}}_{\nu,M,N}(f) - \widehat{\mu}^{\mathrm{IS}}_{\nu,M}(f))^2\right] + \mathrm{E}\left[(\widehat{\mu}^{\mathrm{IS}}_{\nu,M}(f) - \mu(f))^2\right] \ .\end{aligned}$$

# SIR: Large Sample Behavior (contd.)

Going beyond this elementary result is not trivial because the second stage sample $\xi^1, \ldots, \xi^N$ is no more i.i.d. after resampling due to the normalization of the importance weights.

**Theorem 1** *Assume that $\mu \ll \nu$. Let $\{\xi^i\}_{1 \leq i \leq M}$ be i.i.d. random variables with distribution $\nu$. Then $\hat{\mu}^{\mathrm{SIR}}_{\nu,M,N}(f)$ is a (weakly) consistent estimate of $\mu(f)$ for $\mu$-integrable functions $f$ as $M, N \to \infty$.*

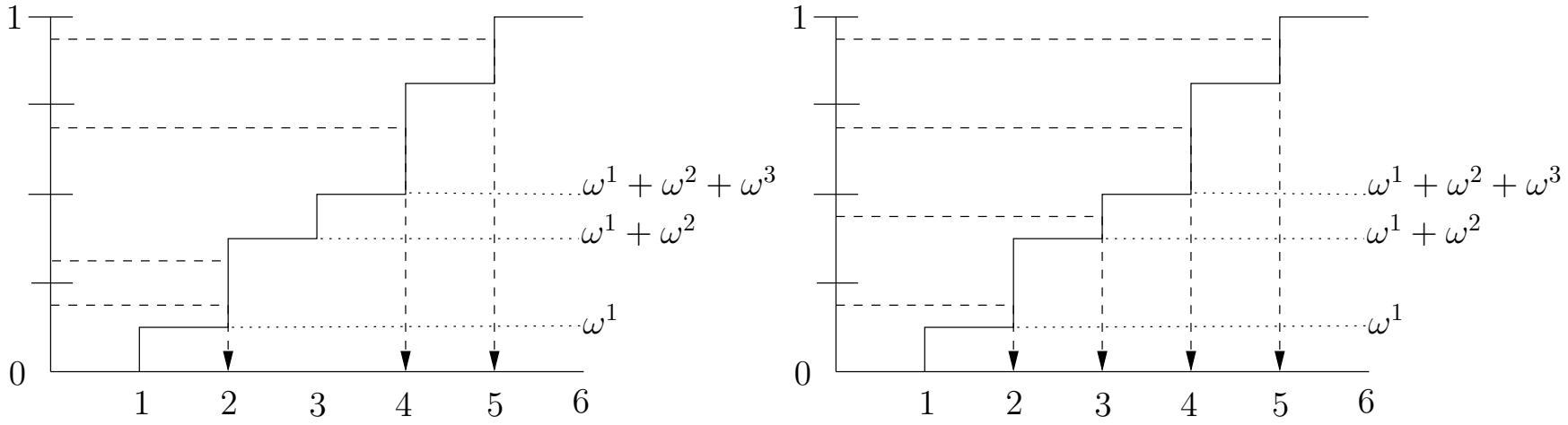
*Assume in addition that $\lim_{M,N\to\infty} M/N = \alpha$ for some $\alpha \geq 1$* and that $d\mu/d\nu$ and $f\, d\mu/d\nu$ are in $L^2(\mathsf{X}, \nu)$. Then $\hat{\mu}^{\mathrm{SIR}}_{\nu,M,N}(f)$ is asymptotically normal $\sqrt{N}(\hat{\mu}^{\mathrm{SIR}}_{\nu,M,N}(f) - \mu(f)) \xrightarrow{\mathcal{D}} \mathrm{N}\left(0, \tilde{\sigma}^2(f)\right)$ with*

$$\tilde{\sigma}^2(f) = \underbrace{\mathrm{Var}_\mu(f)}_{\text{variance of resampling}} + \alpha^{-1} \underbrace{\mathrm{Var}_\nu\left(\frac{d\mu}{d\nu}\{f - \mu(f)\}\right)}_{\text{variance of IS}}.$$

* Analysis of opposite case is possible but less interesting in practice.

# Alternative Resampling Schemes

There are other resampling schemes that guarantee that $\mathrm{E}\left[ N^i \mid \tilde{\xi}^1, \ldots, \tilde{\xi}^M \right] = N\omega^i$ for $i = 1, \ldots, N$ and that have lower conditional variance.




Principle of stratified sampling (left) and systematic sampling (right). Note: the latter does not always reduce the conditional variance.

Studying their large sample behavior is harder however.

# Roadmap

1. What is a Hidden Markov Model?
2. Filtering and Smoothing Recursions
3. Monte Carlo, Importance Sampling and Sampling Importance Resampling
4. Sequential Importance Sampling
5. Sequential Importance Sampling with Resampling
6. More Sequential Monte Carlo Algorithms*
7. Approximation of Sums Functionals and Parameter Estimation*

# Sequential Importance Sampling

- The principle of sequential Monte Carlo methods is to use Monte Carlo integration to approximate the filtering recursion in "general" HMMs (not finite HMMs or GLSSMs).
- The key remark, which can be traced back to (Handschin & Mayne, 1969) and (Handschin, 1970), is that the importance sampling method targeting the joint smoothing distribution $\phi_{0:n|n}$ can be implemented sequentially, due to the particular structure of $\phi_{0:n|n}$.
- The corresponding algorithm is known as sequential importance sampling (SIS).
- The SIS algorithm does reasonably well but is bound to become unreliable for larger values of $n$ (this limitation will be taken care of latter...)

# HMM Notations (Repeated)

Recall that an hidden Markov model is such that

$$X_{k+1} \sim Q(X_k, \cdot) \qquad \text{state equation}$$
$$Y_k \sim G(X_k, \cdot) \qquad \text{measurement equation}$$

where

- $\{X_k\}_{k \geq 0}$ is a Markov chain with transition kernel $Q$ and initial distribution $\nu$
- $G$ is a transition kernel from $(\mathsf{X}, \mathcal{X})$ to $(\mathsf{Y}, \mathcal{Y})$ and there exists a measure $\mu$ such that, for all $x \in \mathsf{X}$, $A \in \mathcal{Y}$,

$$G(x, A) = \int_A g(x, y) \mu(dy) \ .$$

To simplify the mathematical expressions, we use the notation $g_k$ to denote the function $g(\cdot, Y_k)$, considered as a function of its first argument.

# Smoothing (Repeated)

The posterior distribution $\phi_{0:n|n}$* of the states $X_{0:n}$ given the observations $Y_{0:n}$ may be computed recursively (in $n$) according to

$$\phi_{0|0}(f) = \frac{\int g_0(x_0)\nu(dx_0)f(x_0)}{\int g_0(x_0)\nu(dx_0)}$$

$$\phi_{0:n|n}(f_n) = \int \cdots \int f_n(x_{0:n})\, \phi_{\nu,0:n-1|n-1}(dx_{0:n-1})T^{\mathrm{u}}_{n-1}(x_{n-1}, dx_n) \ ,$$

where, for $k \geq 0$, $T^{\mathrm{u}}_n$ is the unnormalized transition kernel onto $(\mathsf{X}, \mathcal{X})$ given by

$$T^{\mathrm{u}}_k(x, A) = \left(\frac{\mathrm{L}_{k+1}}{\mathrm{L}_k}\right)^{-1} \int_A Q(x, dx')g_{k+1}(x'), \quad x \in \mathsf{X}, A \in \mathcal{X} \ .$$

* In this part we omit to indicate the dependence with respect to $\nu$ which not essential.

# Choice of the Instrumental Distribution

**Key Remark:** Both the simulation from the instrumental distribution and the computation of the importance weights can be carried out sequentially if a, possibly non-homogeneous, Markov chain is used as instrumental distribution.

More precisely,

- Let $\{R_k\}_{k\geq 0}$ denote a family of Markov transition kernels on $(\mathsf{X}, \mathcal{X})$ and $\rho_0$ a probability measure on $(\mathsf{X}, \mathcal{X})$.
- Assume that $\phi_{0|0} \ll \rho_0$ and for all $k \geq 0$ and all $x \in \mathsf{X}$, $T_k^{\mathrm{u}}(x, \cdot) \ll R_k(x, \cdot)$.
- The inhomogeneous Markov chain with initial distribution $\rho_0$ and transition kernels $\{R_k\}_{k\geq 0}$ defines the following distributions

$$\rho_{0:k}(f_k) = \int \cdots \int f_k(x_{0:k})\, \rho_0(dx_0) \prod_{l=0}^{k-1} R_l(x_l, dx_{l+1}) \ .$$

# Sequential Computation of the Importance Function

The importance function is then defined as

$$\frac{d\phi_{0:n|n}}{d\rho_{0:n}}(x_{0:n}) = \frac{d\phi_{0|0}}{d\rho_0}(x_0) \prod_{k=0}^{n-1} \frac{dT_n^{\mathrm{u}}(x_k, \cdot)}{dR_n(x_k, \cdot)}(x_{k+1}) \ ,$$

which can be computed sequentially in the sense that

$$\frac{d\phi_{0:k+1|k+1}}{d\rho_{0:k+1}}(x_{0:k+1}) = \frac{d\phi_{0:k|k}}{d\rho_{0:k}}(x_{0:k}) \frac{dT_n^{\mathrm{u}}(x_k, \cdot)}{dR_n(x_k, \cdot)}(x_{k+1}) \ ,$$

for $k \geq 0$.

# Sequential Importance Sampling Algorithm

**Initialization** Draw $\xi_0^1, \ldots, \xi_0^N$ independently from $\rho_0$ and compute the weights

$$\omega_0^i = \frac{d\phi_{0|0}}{d\rho_0}(\xi_0^i), \quad \text{for } i = 1, \ldots, N.$$

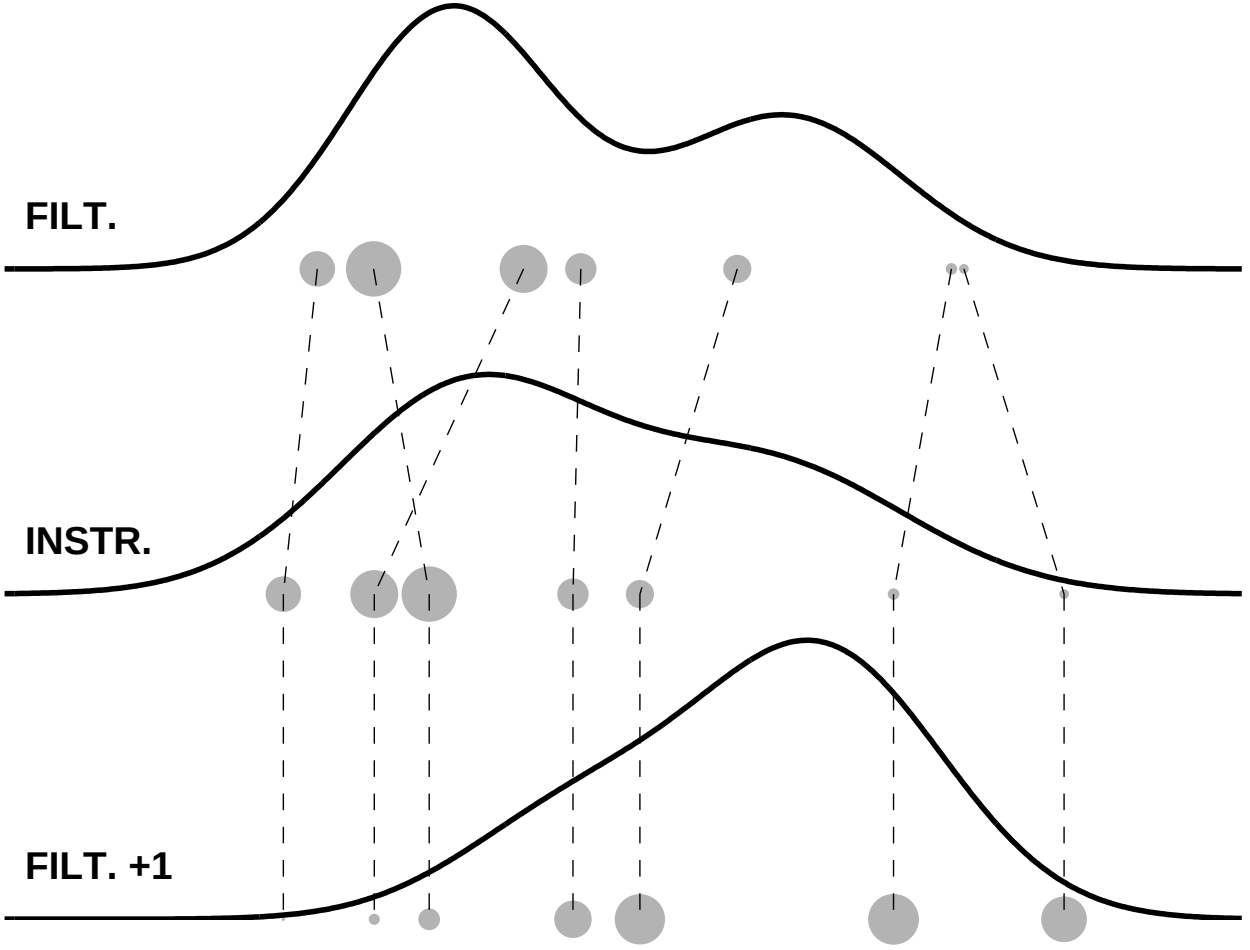
**Recursion** For $k = 0, \ldots$

For $i = 1, \ldots, N$

- Draw $\xi_{k+1}^i$ conditionally independently from $\{\xi_l^j, \xi_k^m\}_{l<k, 1\leq j\leq N, m<i}$ under the distribution $R_k(\xi_k^i, \cdot)$.
- Update the importance weight according to

$$\omega_{k+1}^i = \omega_k^i \frac{dT_k^{\mathrm{u}}(\xi_k^i, \cdot)}{dR_k(\xi_k^i, \cdot)}(\xi_{k+1}^i) \ .$$

The ratio $\omega_{k+1}^i / \omega_k^i$ is often referred to as the incremental weight; the points $\xi_k^i$ are called particles; the trajectories $\xi_{0:k}^i$ path particles.




One step of the SIS algorithm with just seven particles.

# Sequential Importance Sampling Approximation

At any time index $n$, the sequential importance sampling estimator of $\phi_{0:n|n}(f_n)$ is available as

$$\hat{\phi}^{\mathrm{IS}}_{0:n|n}(f_n) = \frac{\sum_{i=1}^N f_n(\xi^i_{0:n})\omega^i_n}{\sum_{i=1}^N \omega^i_n} \ .$$

**Remark** If we are just interested in functions $f_n(x_{0:n}) = f(x_n)$, storing the full trajectories of the particles is not required; each step of the algorithm involves $O(N)$ operations and requires just that $N + N\,\dim(\mathsf{X})$ real numbers be stored.

Likewise, for functions of the form $f_n(x_{0:n}) = f_k(x_{n-k:n})$ only the last $k+1$ elements of each path particle $\xi^i_{0:n}$ needs to be stored.

We will see latter that one may indeed consider more general functions $f_n$ as long as they have a specific structure...

# Choosing the Importance Kernel: (1) the Prior Kernel

As for non-sequential importance sampling, the performance of SIS depends crucially on the choice of the importance kernel $R_k$ (and, to a lesser extent, on that of $\rho_0$).

The most obvious solution is to use the prior kernel $R_n = Q$:

- The instrumental kernel at each iteration mimics the state dynamic, which is usually simple to sample from.
- The incremental weight

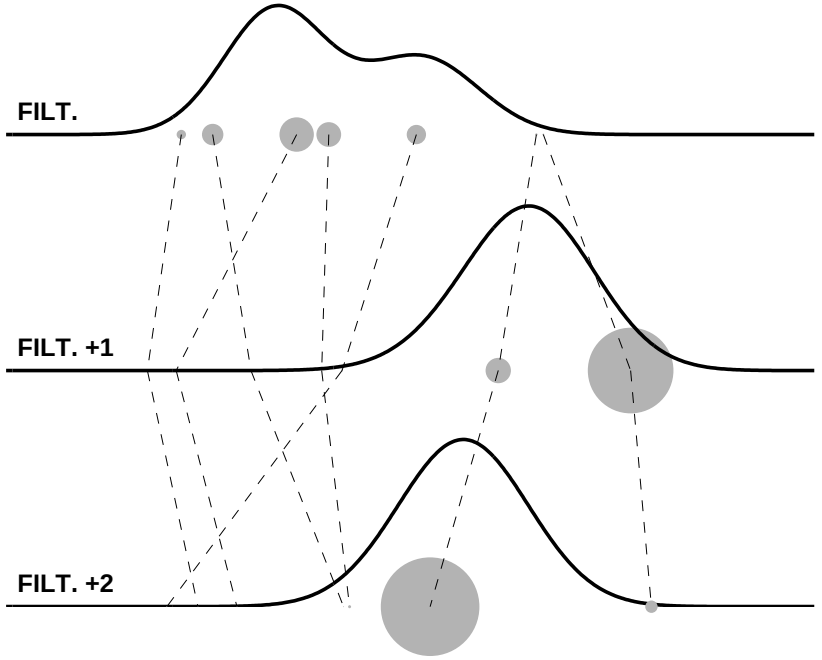
$$\frac{dT_k^{\mathrm{u}}(x,\cdot)}{dR_k(x,\cdot)}(x') = \frac{\mathrm{L}_k}{\mathrm{L}_{k+1}} g_{k+1}(x'), \quad \forall (x,x') \in \mathsf{X} \times \mathsf{X} \, .$$

  *does not depend* on $x \in \mathsf{X}$, hence computing the incremental weight simply amounts to evaluating the conditional likelihood function for the new particle positions*.

*Recall that the importance weights need to be evaluated up to a constant only, hence the non-computable factor $\mathrm{L}_k/\mathrm{L}_{k+1}$ may be omitted.

# Lack of Robustness of the Prior Kernel

The prior kernel is a reasonable option which is computationally very simple and is thus often hard to beat, especially in models where the state is not precisely identified by the observations. It is however very sensitive to the presence of "outliers":




Conflict between the prior and the posterior: at time $k+1$, the observation does not agree with the particle approximation of the predictive distribution. After reweighting step, the mass becomes concentrated on a single particle. Due to the multiplicative structure of the importance weight, recovering from this situation is almost often impossible.

# Choosing the Importance Kernel: (2) the "Optimal" Kernel

To circumvent the problem one needs to incorporate information both on the state dynamic and on the new observation.

Among all possible options, there is only one kernel which is such that the new weight $\omega_{k+1}^i$ is a deterministic function of the current particle $\xi_k^i$; this is the only choice for which the conditional variance of the new weights is equal to zero:

- Let $R_k(f) = T_k(x, f) \stackrel{\text{def}}{=} \gamma_k(x)^{-1} \int f(x')\, Q(x, dx') g_{k+1}(x')$ where

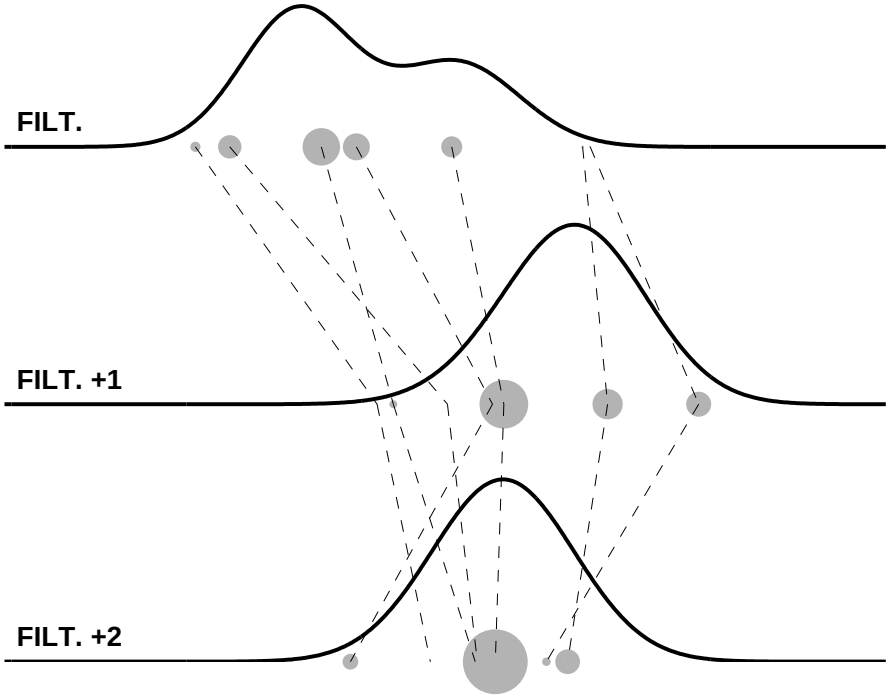
$$\gamma_k(x) \stackrel{\text{def}}{=} \int_{\mathsf{X}} Q(x, dx') g_{k+1}(x') \ .$$

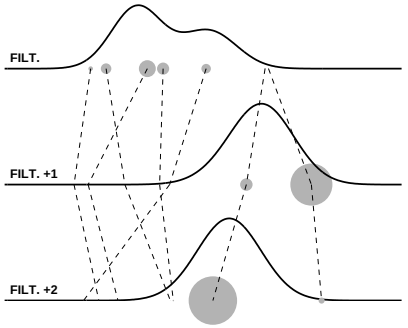
- Then

$$\frac{dT_k^{\mathrm{u}}(x, \cdot)}{dT_k(x, \cdot)}(x') = \frac{\mathrm{L}_k}{\mathrm{L}_{k+1}} \gamma_k(x), \quad \forall (x, x') \in \mathsf{X} \times \mathsf{X}.$$

Unfortunately, computing $\gamma_k$ is usually not feasible in models where implementing the filtering recursion is problematic!

# The Optimal Kernel is More Robust to Outliers




The optimal kernel proposes particles in the regions where the filtering density has most of its mass.

# Local Approximation of the Optimal Importance Kernel

The aim is to find a distribution which resembles sampling from the optimal kernel but for which the incremental weight is computable:

- Ideally, this distribution should be overdispersed (recall the $d\mu/d\nu$ factor!) but not wildly inaccurate.
- We can find such a distribution in two steps:
  1. locate the high-density region of the (multivariate) optimal distribution to ensure that our proposal does not entirely miss important regions;
  2. create an overdispersed approximation, so that the instrumental distribution dominates the optimal importance distribution.
- Of course, because we have to repeat the process for each particles, the overall procedure should be reasonably simple.

# Application to the Stochastic Volatility Model

Consider the (discrete-time) **stochastic volatility** model

$$X_{k+1} = \phi X_k + \sigma U_k \qquad |\phi| < 1 \ ,$$
$$Y_k = \beta \exp(X_k/2) V_k \ ,$$

where

1. $\{U_k\}_{k\geq 0}$ and $\{V_k\}_{k\geq 0}$ are independent standard Gaussian white noise processes.
2. $X_0 \sim \mathrm{N}(0, \sigma^2/(1-\phi^2))$.

In this model,

$$q(x, x') = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x' - \phi x)^2}{2\sigma^2}\right) \ ,$$
$$g_{k+1}(x') = \frac{1}{\sqrt{2\pi\beta^2}} \exp\left(-\frac{Y_{k+1}^2}{2\beta^2} \exp(-x') - \frac{1}{2}x'\right) \ ,$$

and the incremental weight $\gamma_k(x)$ is not available in closed form.

# Application to the Stochastic Volatility Model (contd.)

- The function $x' \mapsto \log(q(x,x')g_{k+1}(x'))$ is (strictly) log-concave and thus unimodal.
- The mode $m_k(x)$ of the optimal transition density is the unique solution of the non-linear equation,

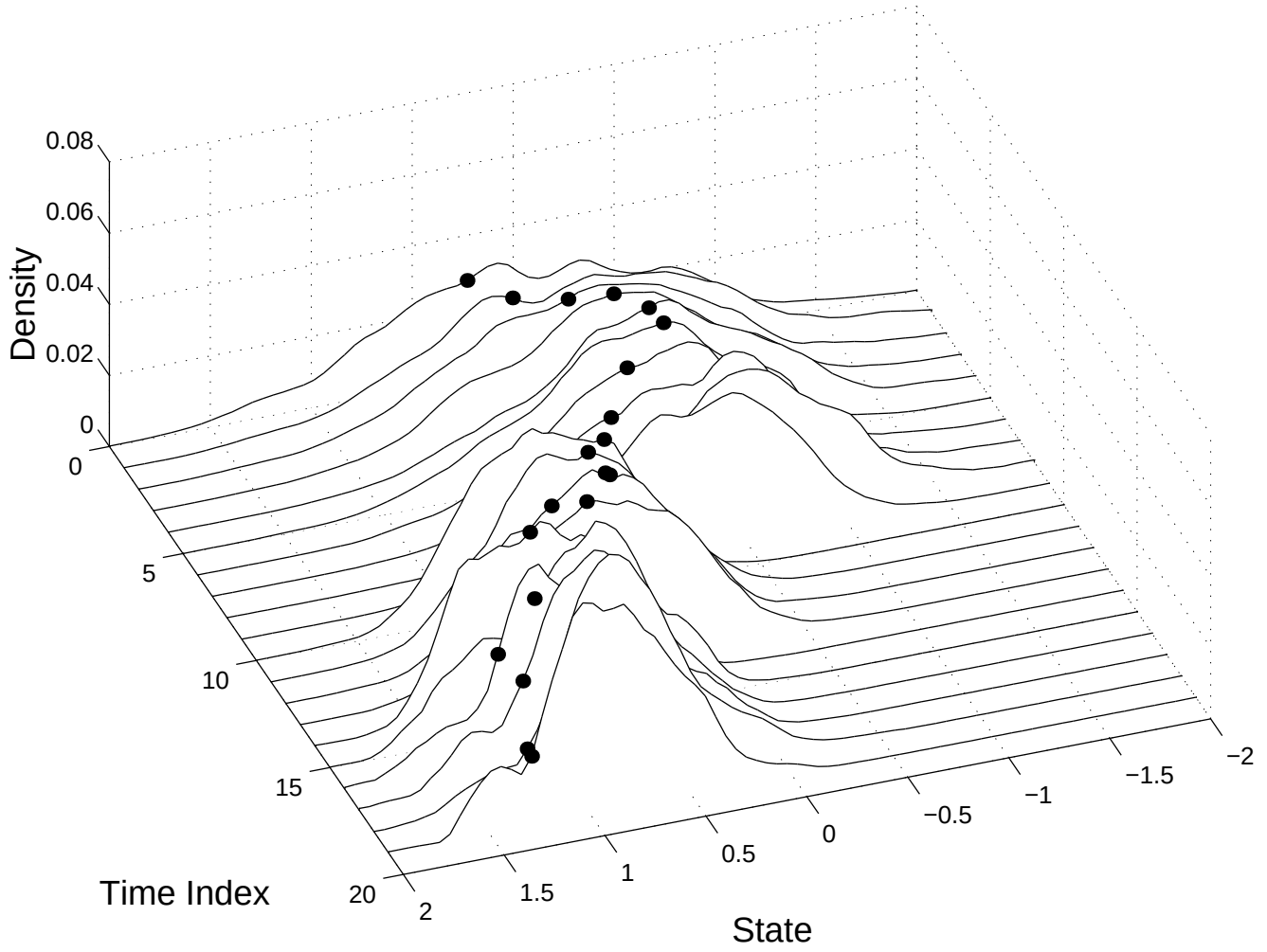
$$-\frac{1}{\sigma^2}(x' - \phi x) + \frac{Y_k^2}{2\beta^2}\exp(-x') - \frac{1}{2} = 0.$$

- The solution of this equation can be computed numerically.
- We use, for instance, as instrumental kernel a $t$-distribution with $\eta = 5$ degrees of freedom, the scale of which being set as the inverse of the negated second-order derivative of $x' \mapsto \log q(x,x')g_k(x')$ evaluated at the mode $m_k(x)$, which is given by:

$$\sigma_k^2(x) = \left(\frac{1}{\sigma^2} + \frac{Y_{k+1}^2}{2\beta^2}\exp[-m_k(x)]\right)^{-1}.$$

The incremental weight may easily be evaluated once $m_k(x)$ and $\sigma_k^2(x)$ have been computed (note that it now depends both on $x$ and $x'$). Recall also that we need to repeat these steps independently for each current particle position $x = \xi_k^i$.

# Application to the Stochastic Volatility Model (contd.)




Waterfall representation of the sequence of estimated filtering distribution with actual state (1,000 particles).

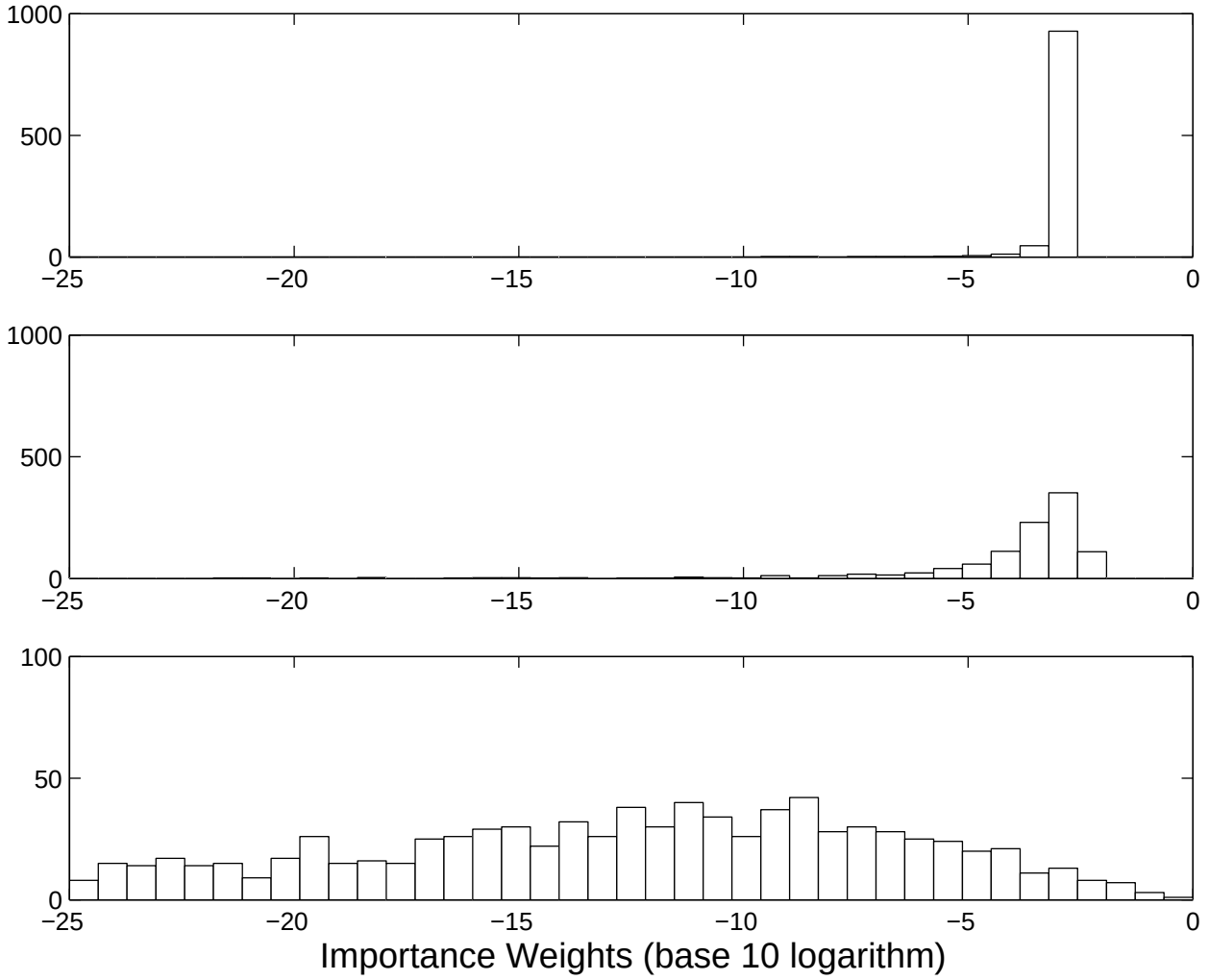
# Weight Degeneracy

- The normalized importance weights measure the pertinence of each particle: a relatively small importance weight implies that the associated particle is far from the main body of the posterior distribution and contributes poorly to the sequential importance sampling approximation.
- If there are too many such ineffective particles, the Monte-Carlo approximation becomes highly unreliable.

# Weight Degeneracy

- The normalized importance weights measure the pertinence of each particle: a relatively small importance weight implies that the associated particle is far from the main body of the posterior distribution and contributes poorly to the sequential importance sampling approximation.
- If there are too many such ineffective particles, the Monte-Carlo approximation becomes highly unreliable.
- Empirically, this phenomenon "always" happens when $n$ gets larger ($N$ being fixed).
- In simplistic models, it is possible to show that the asymptotic variance of the approximation $\hat{\phi}_n^{\mathrm{IS}}(f)$ increases exponentially as $n$ increases (see text).

# Application to the Stochastic Volatility Model (contd.)




Histograms of the base 10 logarithm of the normalized importance weights after (from top to bottom) 1, 10 and 100 iterations for the stochastic volatility model

# Numerical Indicator: (1) Coefficient of Variation

- A simple criterion is the coefficient of variation of the normalized weights,

$$\mathrm{CV}_N(\omega) = \left\{ \frac{1}{N} \sum_{i=1}^{N} \left\{ N \frac{\omega^j}{\sum_{j=1}^{N} \omega^j} - 1 \right\}^2 \right\}^{1/2},$$

$$\omega = (\omega^1, \ldots, \omega^N) \in (\mathbb{R}^+)^{\times N}.$$

- When the weights are all equal to $1/N$, then the coefficient of variation is equal to $0$. At the other extreme, when one normalized weight is equal to $1$ and all the others $0$, the coefficient of variation equals $\sqrt{N-1}$. Therefore, a large $\mathrm{CV}_N(\omega_k)$ indicates that there are many ineffective particles and that memory and computation will be wasted.

# Numerical Indicator: (2) Entropy

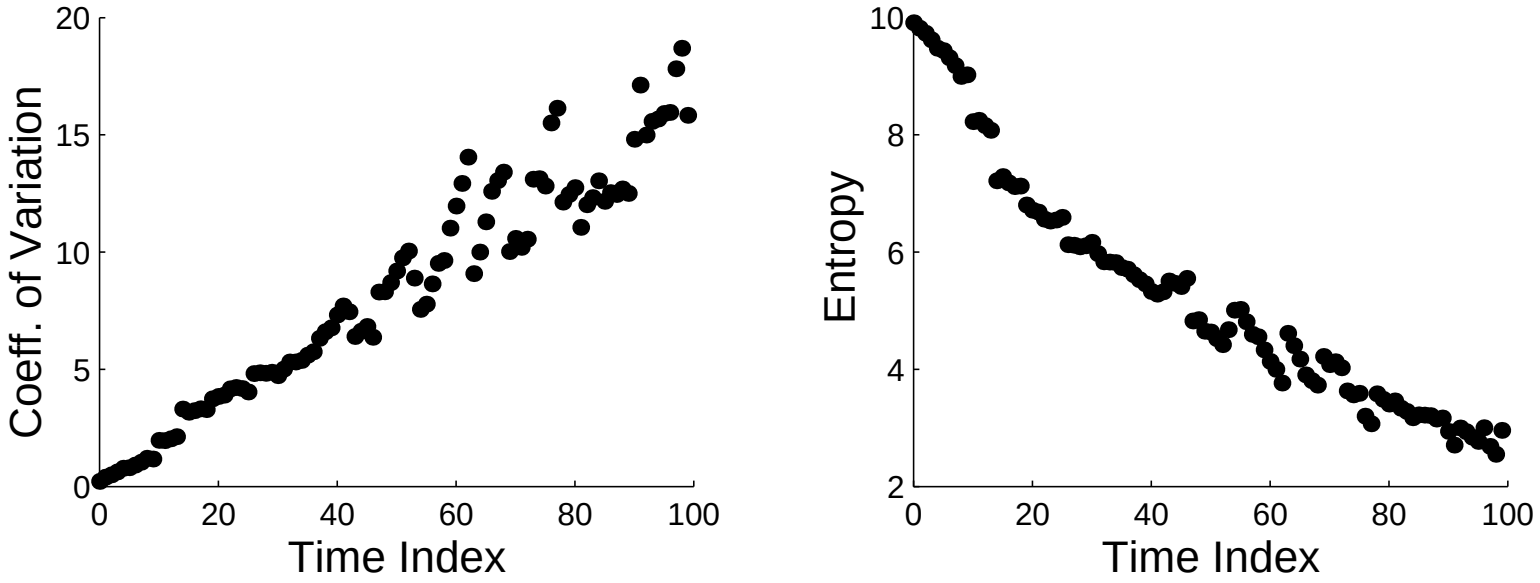
- Another possible measure of the weight imbalance is the Shannon Entropy of the importance weights, defined as

$$\mathrm{Ent}(\omega) = -\sum_{i=1}^{N} \frac{\omega^i}{\sum_{j=1}^{N} \omega^j} \log_2 \left( \frac{\omega^i}{\sum_{j=1}^{N} \omega^j} \right),$$

$$\omega = (\omega^1, \ldots, \omega^N) \in (\mathbb{R}^+)^{\times N}.$$

- When all the importance weights are $0$ except $1$, then the entropy is null. On the contrary, if all the weights are equal to $1/N$, then the entropy is maximal and equal to $\log_2(N)$.

# Application to the Stochastic Volatility Model (contd.)




Left: coefficient of variations of the weights; right: weight entropy, as a function of $n$

# Roadmap

1. What is a Hidden Markov Model?
2. Filtering and Smoothing Recursions
3. Monte Carlo, Importance Sampling and Sampling Importance Resampling
4. Sequential Importance Sampling
5. Sequential Importance Sampling with Resampling
6. More Sequential Monte Carlo Algorithms*
7. Approximation of Sums Functionals and Parameter Estimation*

# Resampling

- The solution, proposed by (Gordon, Salmond & Smith, 1993), to avoid the degeneracy of the importance weights is to regularly resample the particles according to their importance weights (thus equating all importance weights).

# Resampling

- The solution, proposed by (Gordon, Salmond & Smith, 1993), to avoid the degeneracy of the importance weights is to regularly resample the particles according to their importance weights (thus equating all importance weights).
- The basic idea of resampling is to

  (i) eliminate particles which have small importance weights,

  (ii) replicate particles which have large importance weights in proportion of their relevance.

# Resampling

- The solution, proposed by (Gordon, Salmond & Smith, 1993), to avoid the degeneracy of the importance weights is to regularly resample the particles according to their importance weights (thus equating all importance weights).
- The basic idea of resampling is to

  (i) eliminate particles which have small importance weights,

  (ii) replicate particles which have large importance weights in proportion of their relevance.
- Resampling concentrates the particles in regions of the state space which are pertinent and avoids exploration of highly improbable areas.

# Resampling

- This idea is clearly rooted in the sampling importance resampling (SIR) technique.

# Resampling

- This idea is clearly rooted in the sampling importance resampling (SIR) technique.
- However, contrary to standard (non-sequential) SIR, the main aim of the resampling step is not to draw (asymptotically correctly) an i.i.d. sample from a distribution but rather to avoid weight degeneracy.

# Resampling

- This idea is clearly rooted in the sampling importance resampling (SIR) technique.
- However, contrary to standard (non-sequential) SIR, the main aim of the resampling step is not to draw (asymptotically correctly) an i.i.d. sample from a distribution but rather to avoid weight degeneracy.
- The resampling step, while useful in fighting degeneracy, has a drawback: resampling introduces unnecessary noise into the algorithm, and this extra noise might be far from negligible.

# Resampling

- This idea is clearly rooted in the sampling importance resampling (SIR) technique.
- However, contrary to standard (non-sequential) SIR, the main aim of the resampling step is not to draw (asymptotically correctly) an i.i.d. sample from a distribution but rather to avoid weight degeneracy.
- The resampling step, while useful in fighting degeneracy, has a drawback: resampling introduces unnecessary noise into the algorithm, and this extra noise might be far from negligible.
- Intuitively, when the importance weights are nearly constant, resampling only reduce the number of distinct particles thus introducing an extra noise without much benefit on the weight degeneracy.

# Resampling

- This idea is clearly rooted in the sampling importance resampling (SIR) technique.
- However, contrary to standard (non-sequential) SIR, the main aim of the resampling step is not to draw (asymptotically correctly) an i.i.d. sample from a distribution but rather to avoid weight degeneracy.
- The resampling step, while useful in fighting degeneracy, has a drawback: resampling introduces unnecessary noise into the algorithm, and this extra noise might be far from negligible.
- Intuitively, when the importance weights are nearly constant, resampling only reduce the number of distinct particles thus introducing an extra noise without much benefit on the weight degeneracy.
- The one-step effect of resampling is thus negative but, on the long-term, resampling is required to guarantee a correct behavior of the algorithm.

# Sequential Importance Sampling with Resampling (SISR)

For time indices $k \geq 0$, do the following.

**Sampling:**

- Draw $(\tilde{\xi}^1_{k+1}, \ldots, \tilde{\xi}^N_{k+1})$ conditionally independently given $\{\xi^j_{0:k}\ , j = 1, \ldots, N\}$ from the instrumental kernel: $\tilde{\xi}^i_{k+1} \sim R_k(\xi^i_k, \cdot)$, $i = 1, \ldots, N$.
- Compute the updated importance weights

$$\omega^i_{k+1} = \omega^i_k \, g_{k+1}(\tilde{\xi}^i_{k+1}) \frac{dQ(\xi^i_k, \cdot)}{dR_k(\xi^i_k, \cdot)}(\tilde{\xi}^i_{k+1}) \ , \qquad i = 1, \ldots, N \ .$$

**Resampling (Optional):**

- Draw, conditionally independently given $\{(\xi^i_{0:k}, \tilde{\xi}^j_{k+1}),\ i, j = 1, \ldots, N\}$, the multinomial trial $(I^1_{k+1}, \ldots I^N_{k+1})$ with probabilities of success

$$\frac{\omega^1_{k+1}}{\sum_j^N \omega^j_{k+1}}, \ \ldots \ , \frac{\omega^N_{k+1}}{\sum_j^N \omega^j_{k+1}} \ .$$

- Reset the importance weights $\omega^i_{k+1}$ to a constant value for $i = 1, \ldots, N$.

# SISR contd.

If resampling is not applied, set for $i = 1, \ldots, N$, $I_{k+1}^i = i$.

**Trajectory update:** for $i = 1, \ldots, N$,

$$\xi_{0:k+1}^i = \left( \xi_{0:k}^{I_{k+1}^i} , \tilde{\xi}_{k+1}^{I_{k+1}^i} \right) .$$

Recall that storing the full particle path is usually not needed.

The SISR algorithm with systematic resampling and $R_k = Q$ (the prior kernel) is known as the bootstrap filter.

# Illustration of the Boostrap Filter on a Toy Example

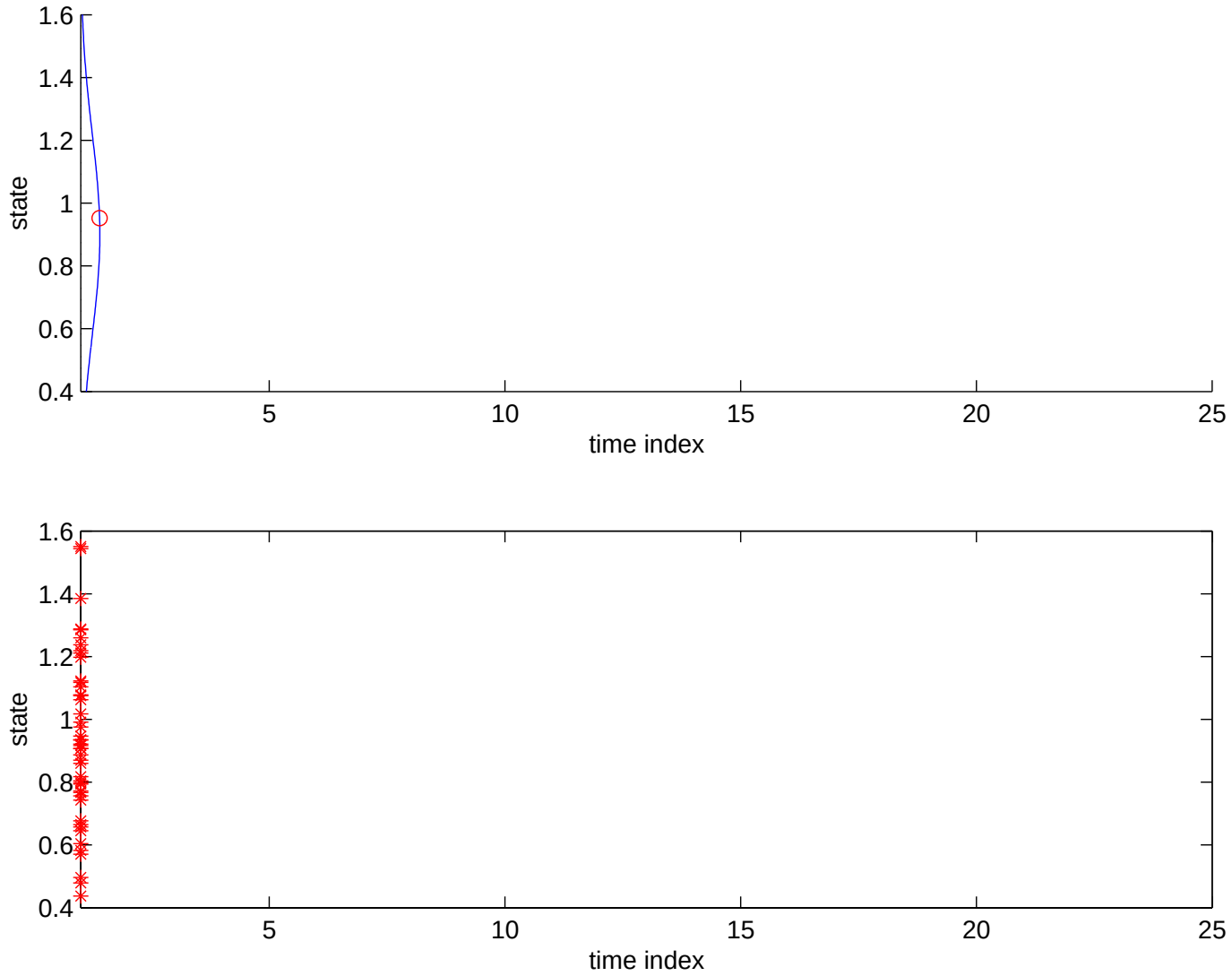
**Noisy AR(1) model**

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

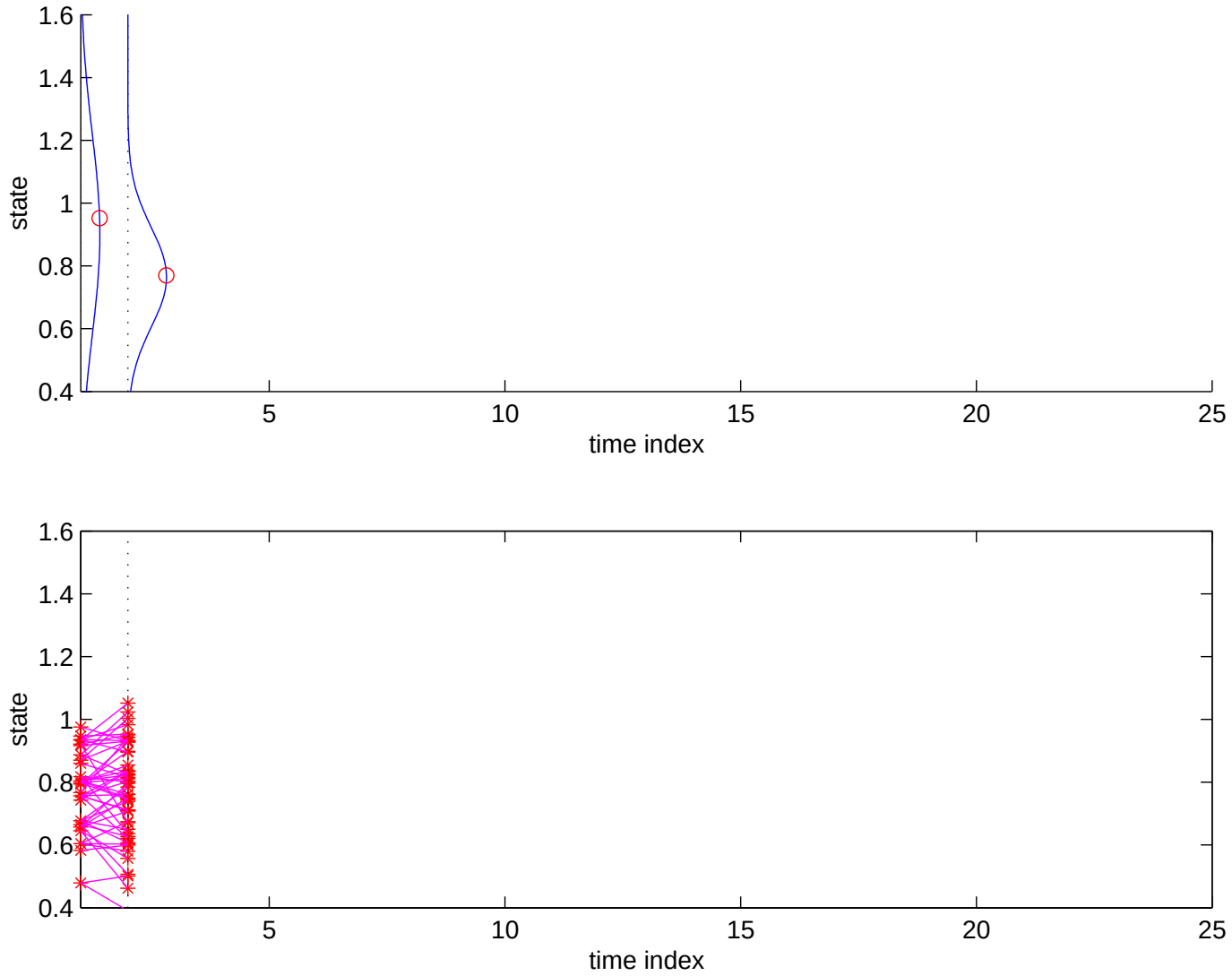
$\mu = 0.9, \phi = 0.95, \sigma^2 = 0.01, \eta^2 = 0.02 = (\sigma^2/(1-\phi^2))/5$

To approximate the predictive distribution $\phi_{k+1|k}$, we use the bootstrap filter with $N = 50$ particles, plotting the full particle paths $\{\xi^i_{0:k}, \tilde{\xi}^i_{k+1}\}_{1 \leq i \leq N}$ for each time index.

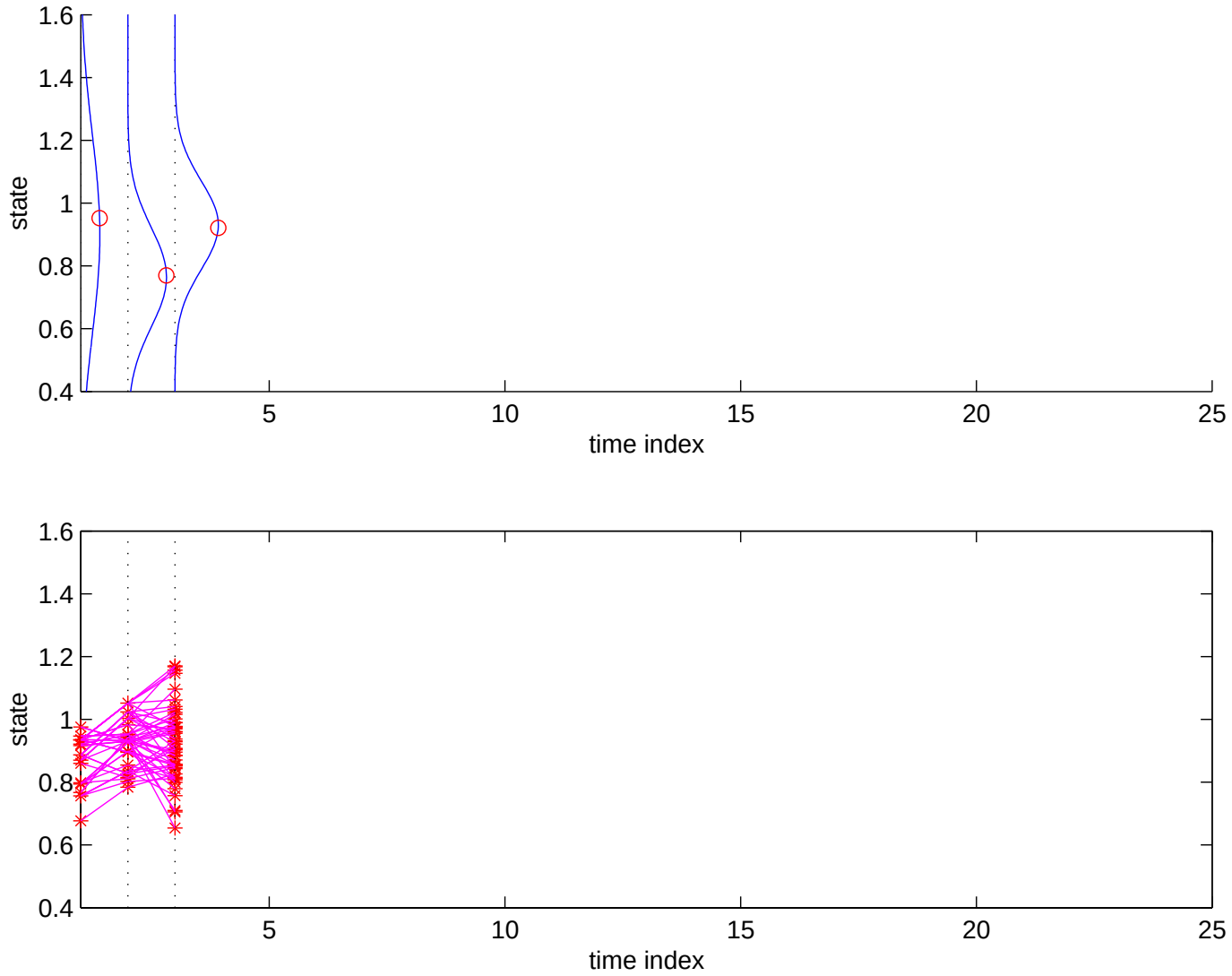
This example is used since we may also compute the actual filtering densities using Kalman filtering



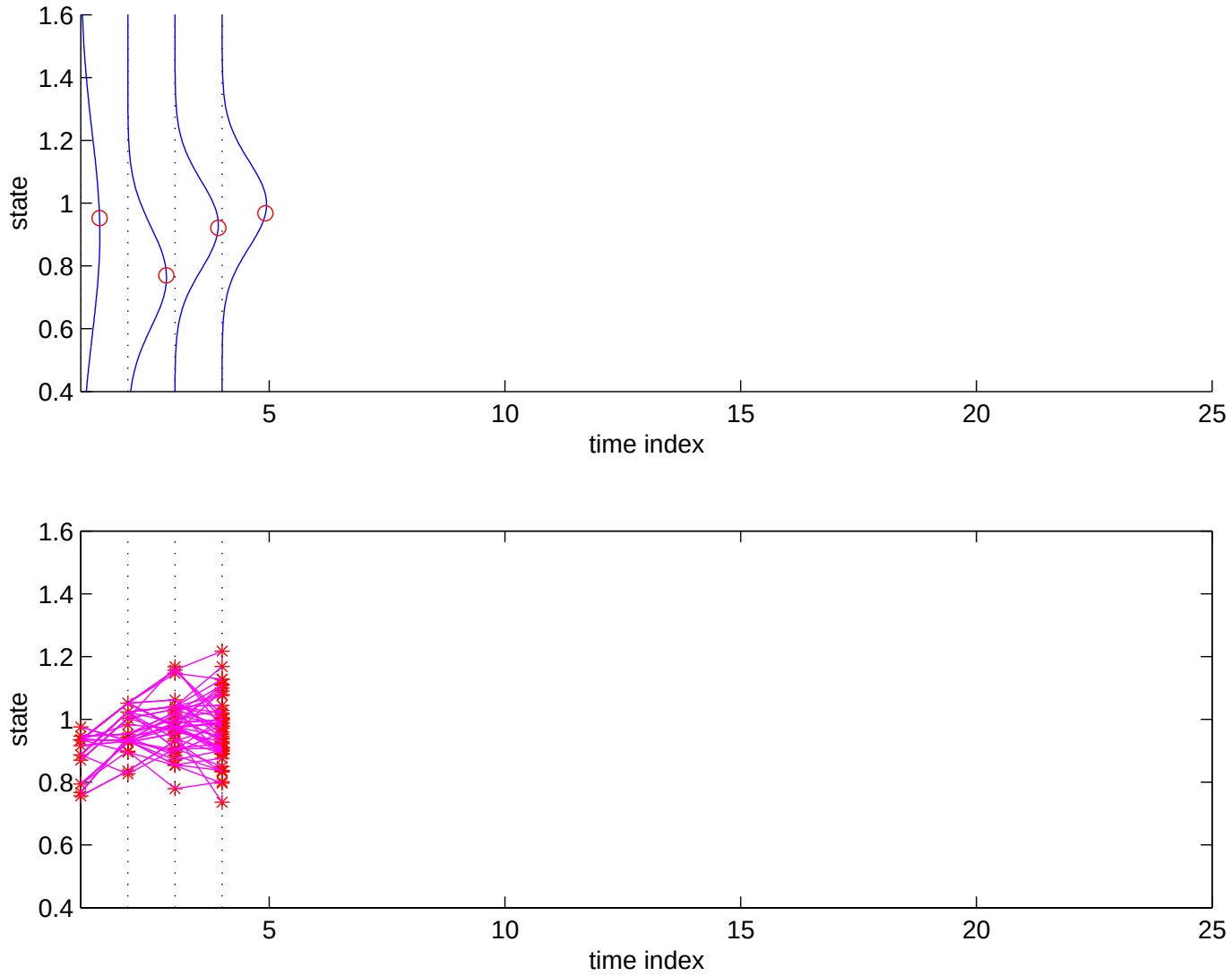

Predictive densities and evolution of the particle paths



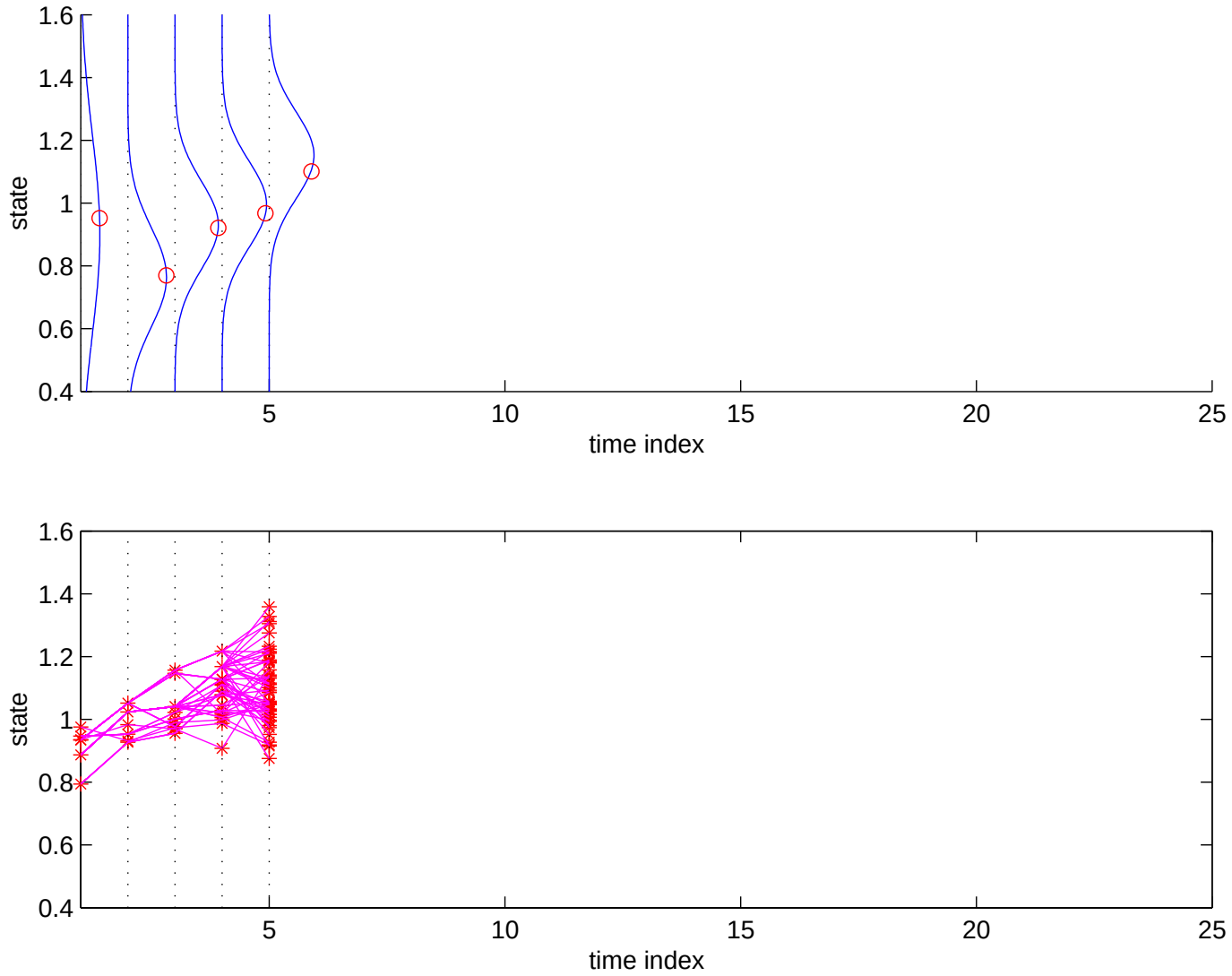

Predictive densities and evolution of the particle paths



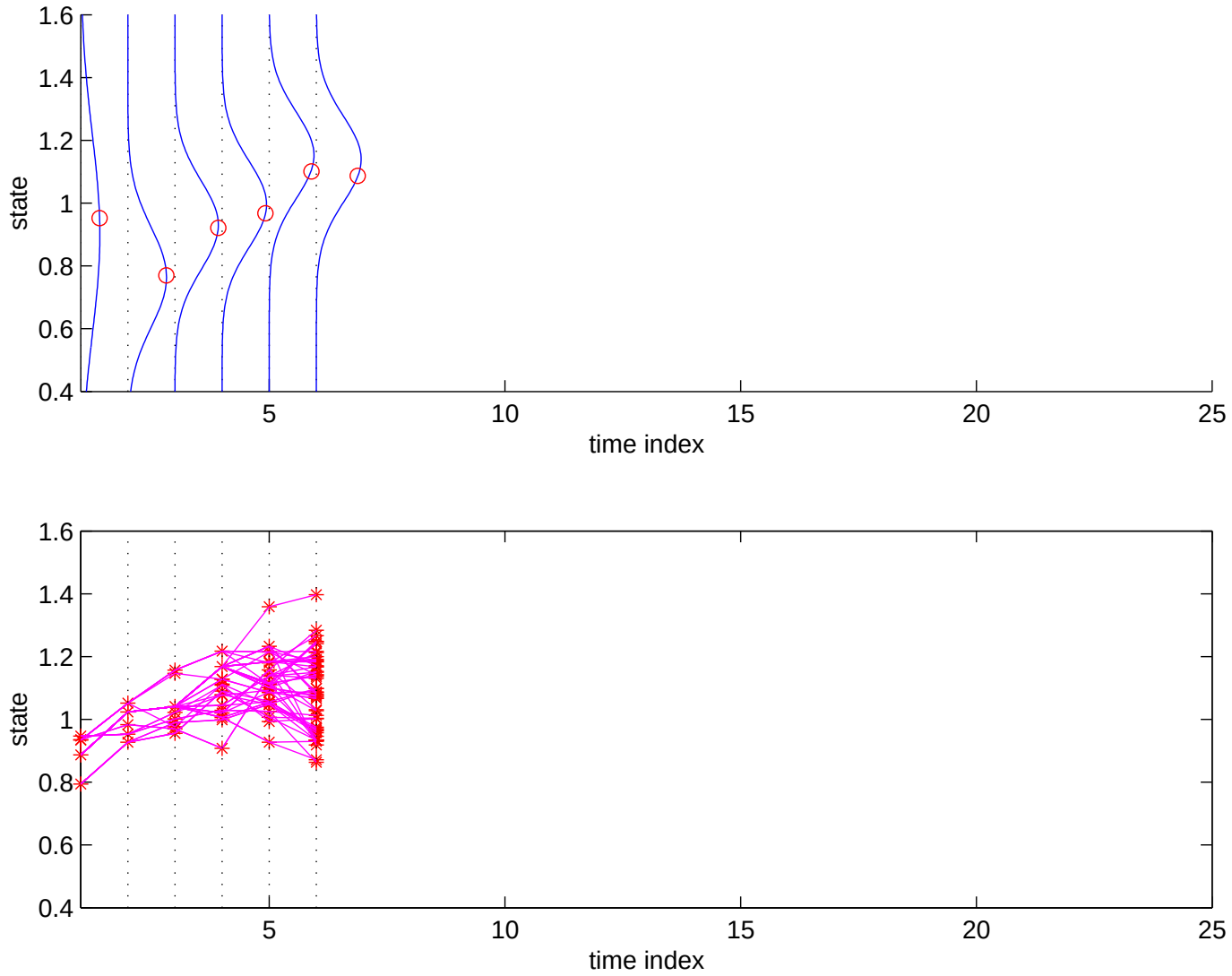

Predictive densities and evolution of the particle paths



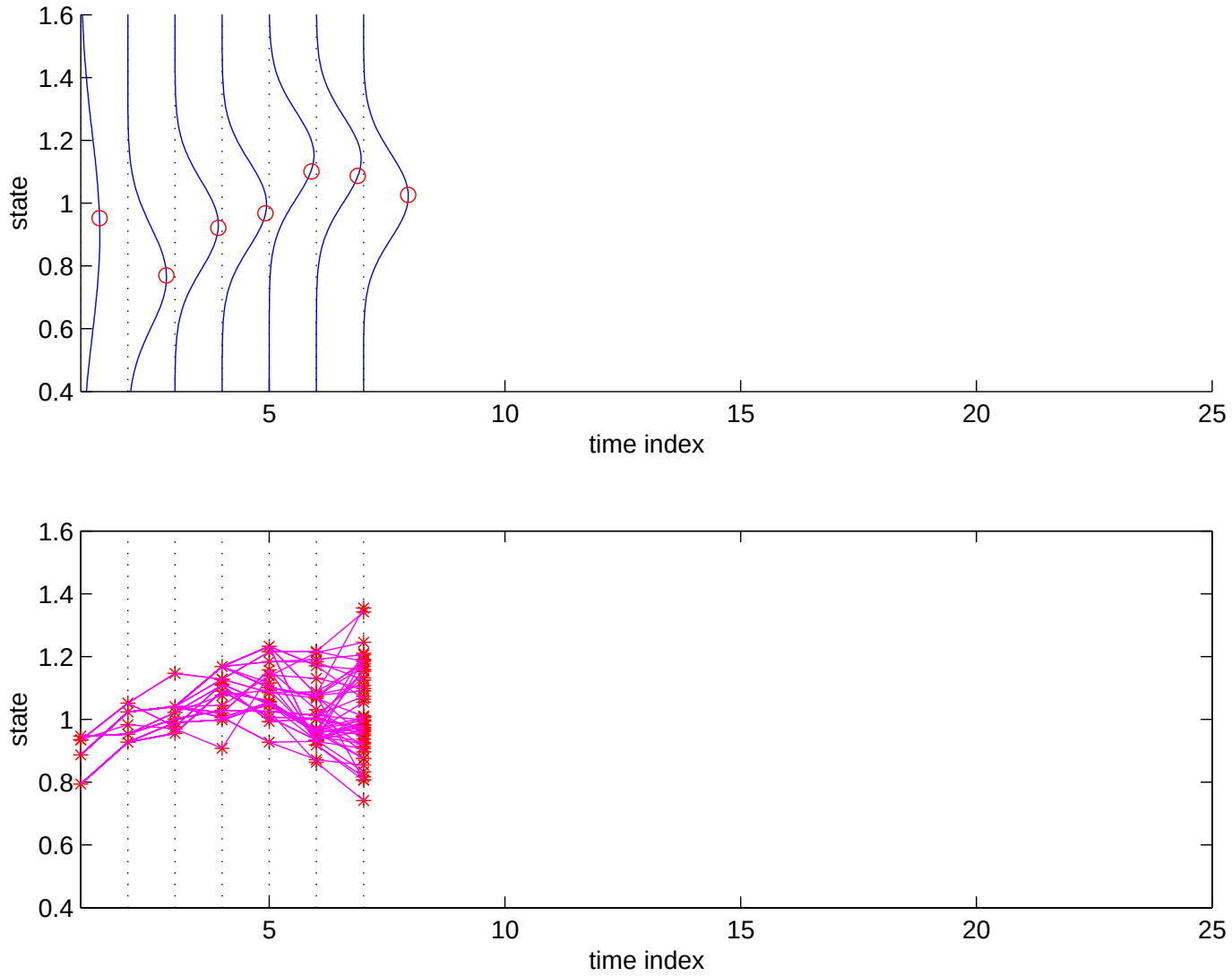

Predictive densities and evolution of the particle paths



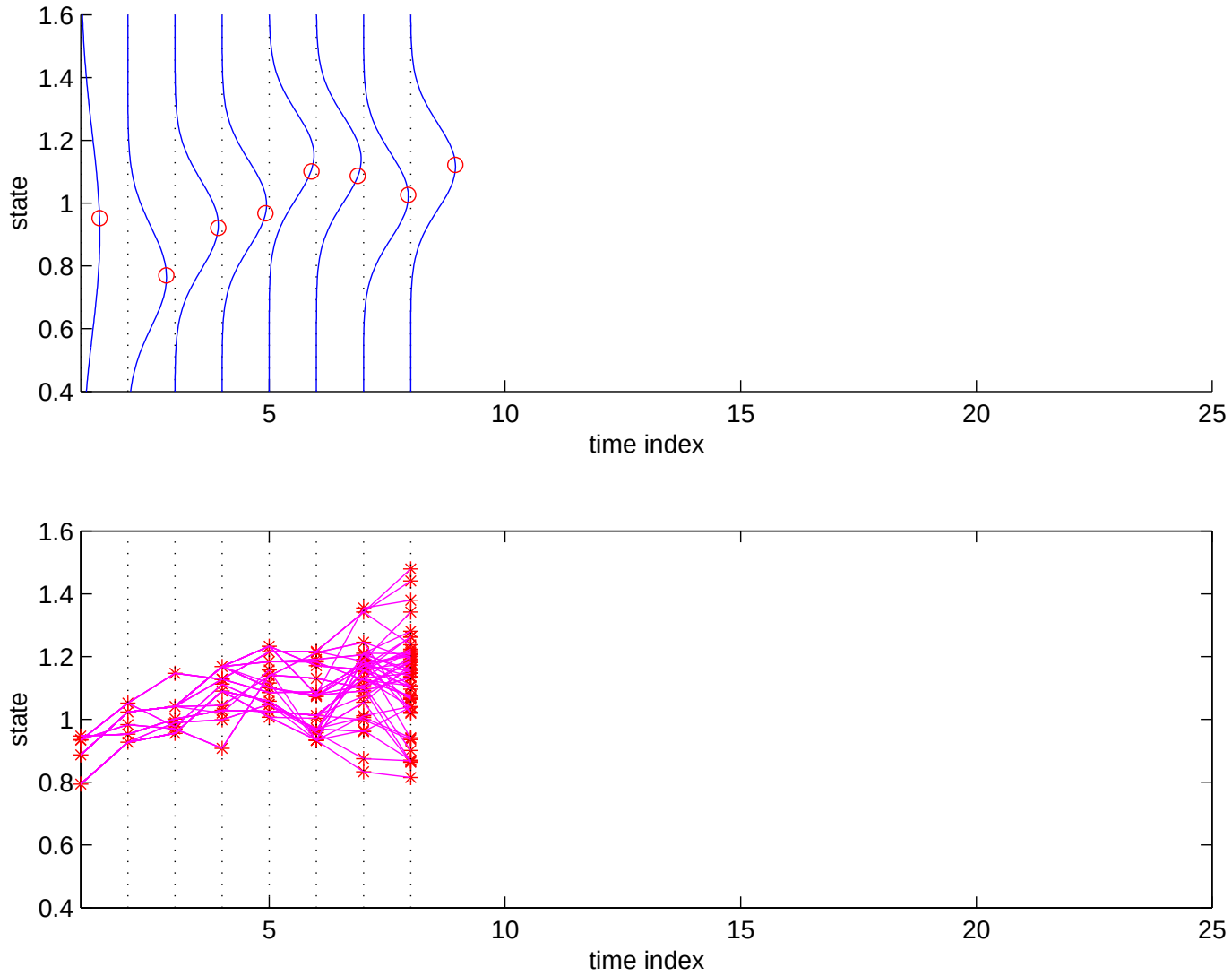

Predictive densities and evolution of the particle paths



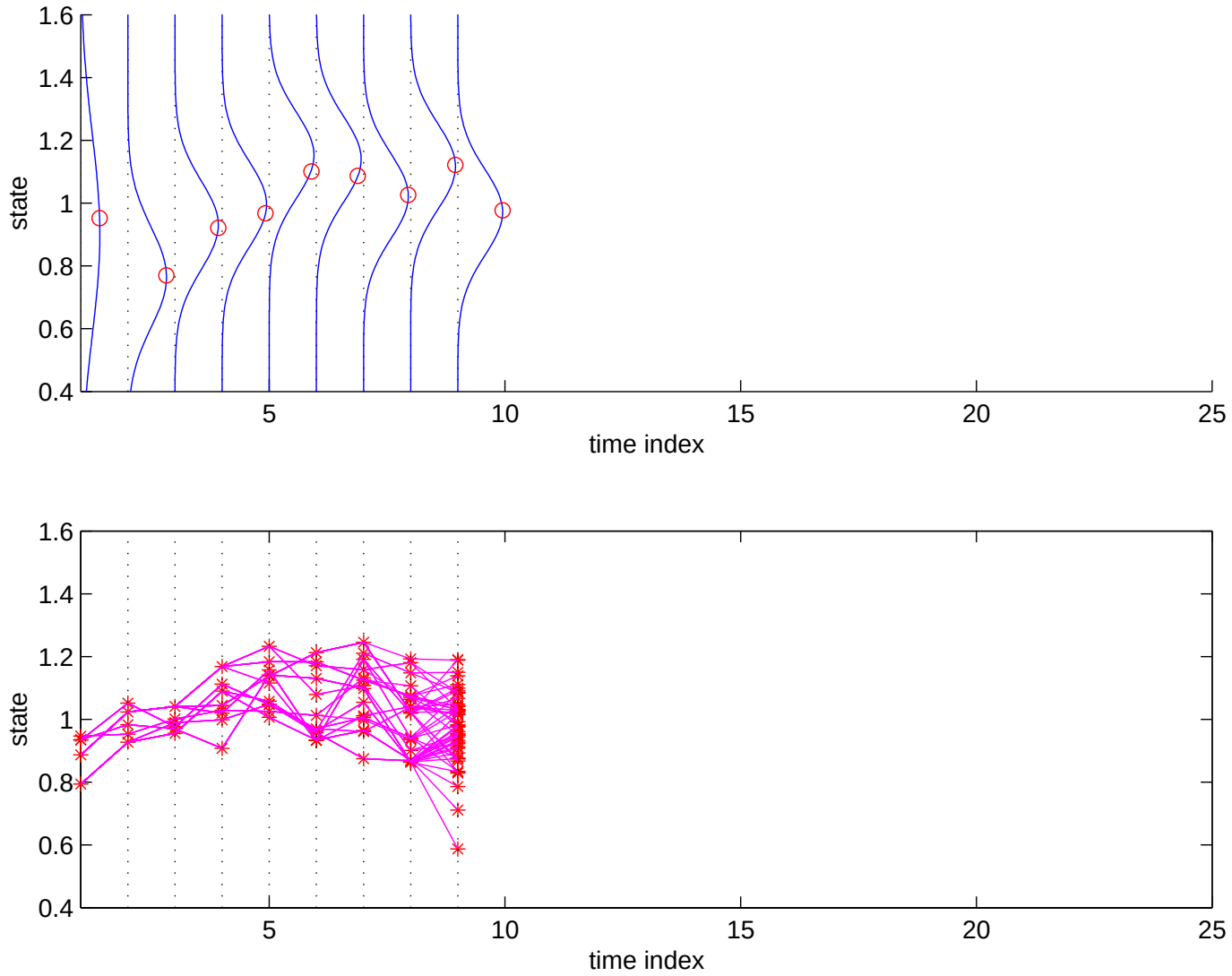

Predictive densities and evolution of the particle paths



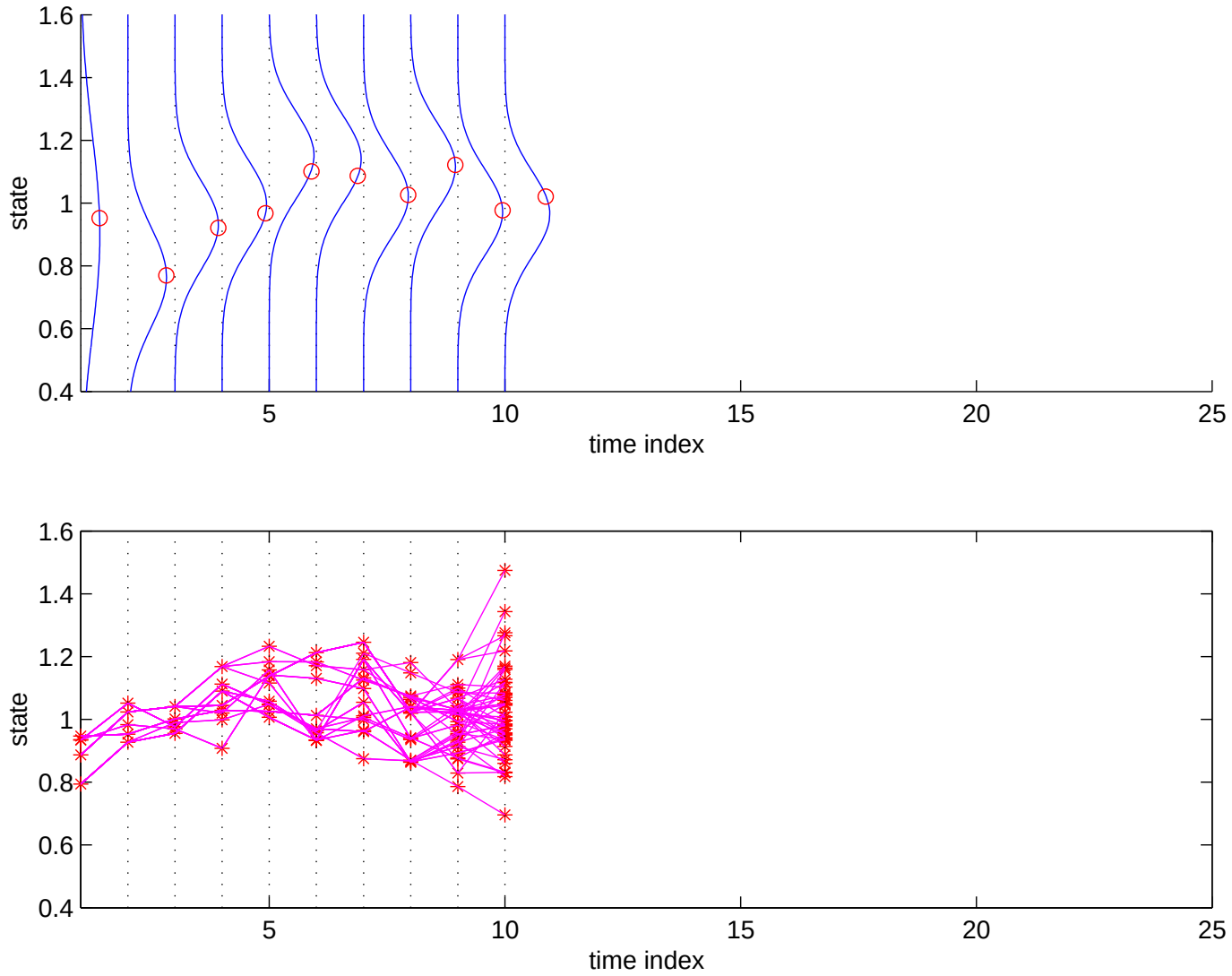

Predictive densities and evolution of the particle paths



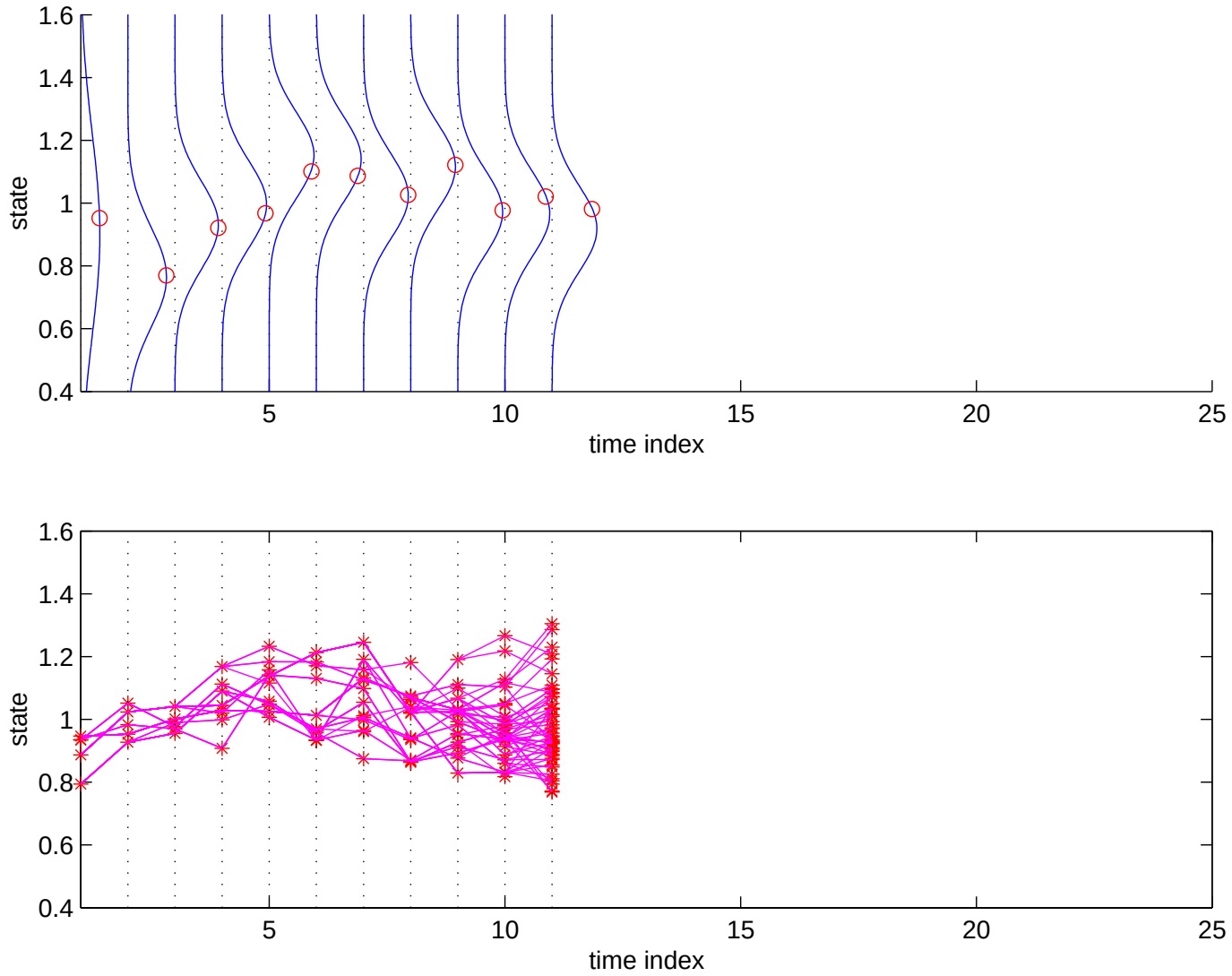

Predictive densities and evolution of the particle paths



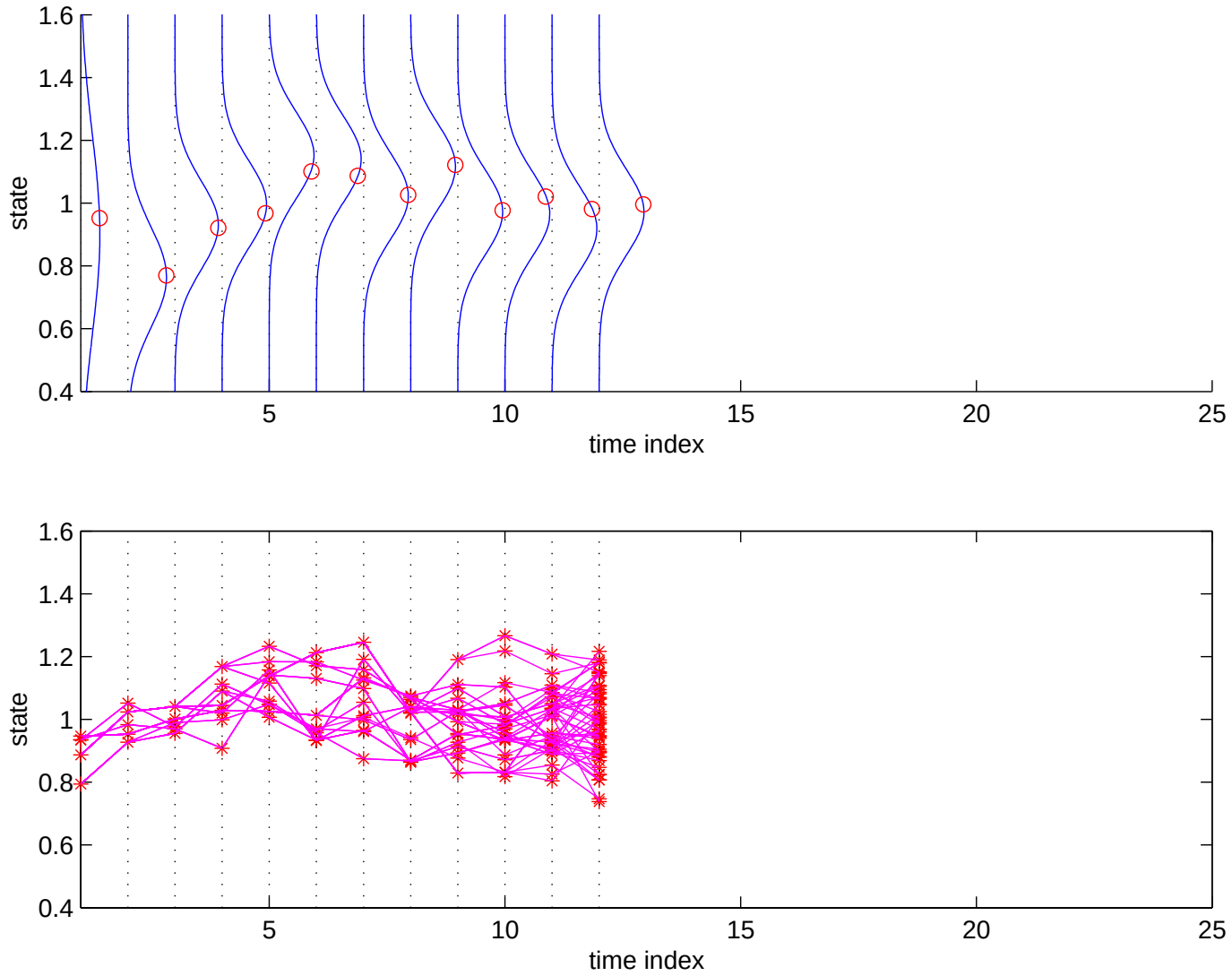

Predictive densities and evolution of the particle paths



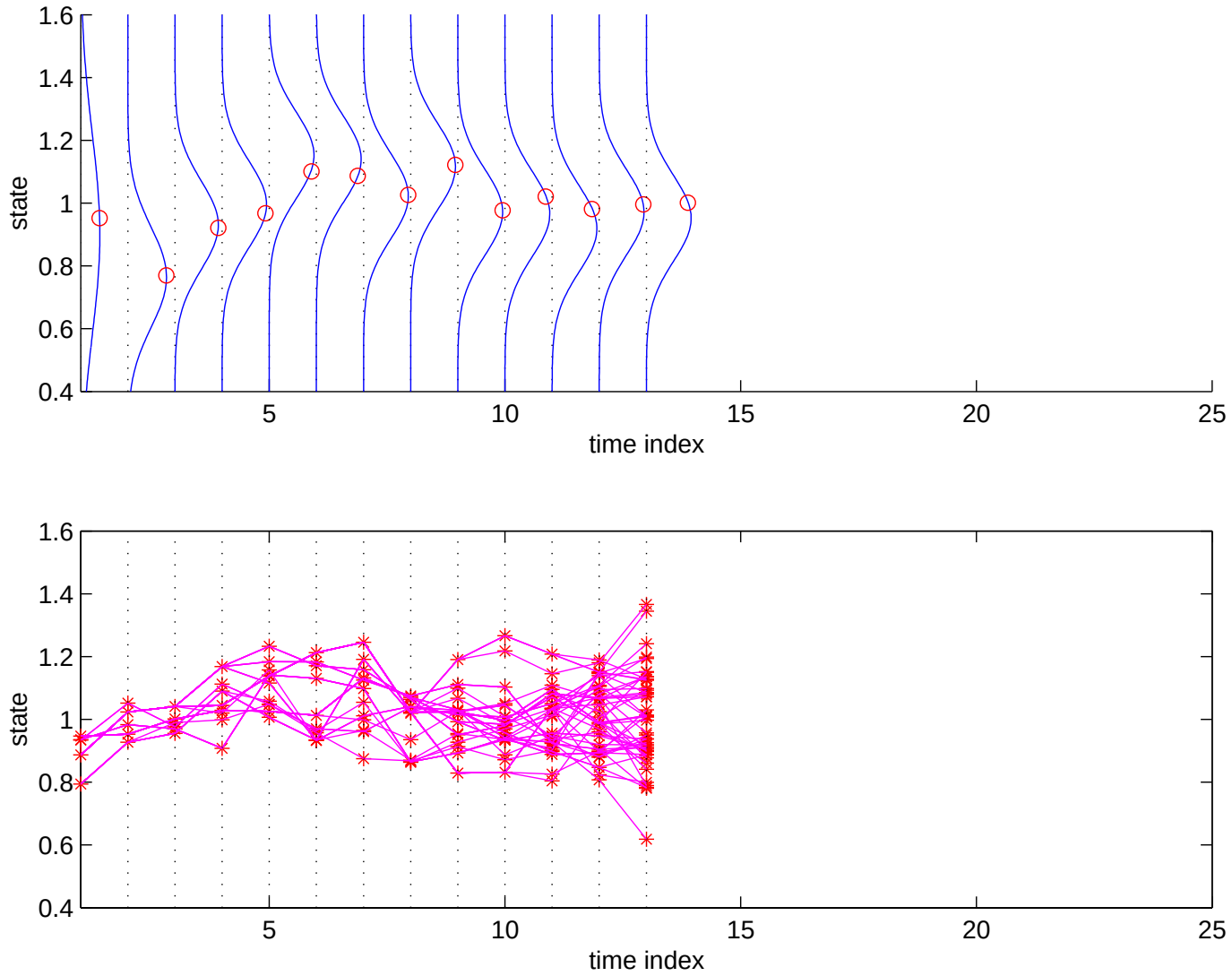

Predictive densities and evolution of the particle paths



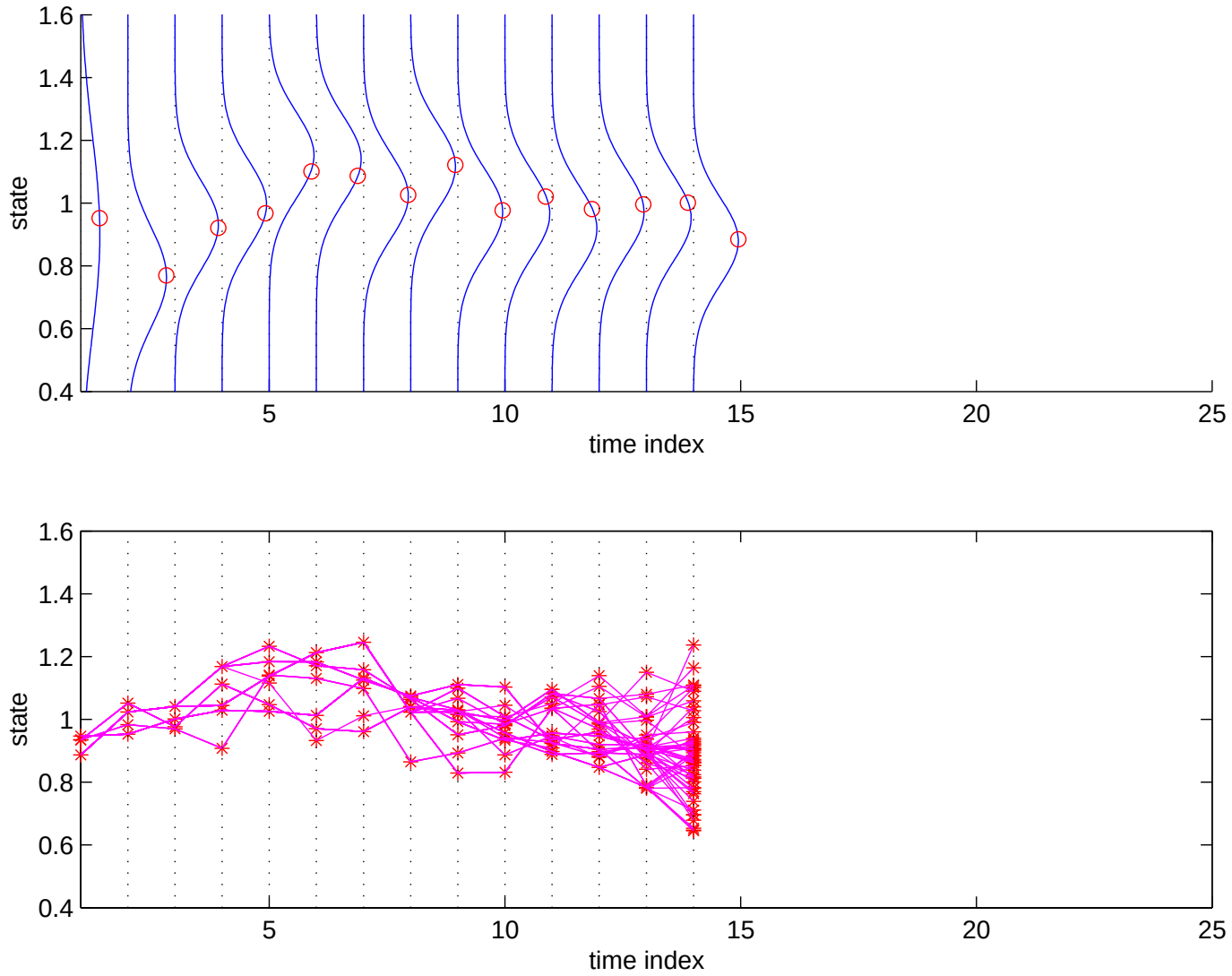

Predictive densities and evolution of the particle paths



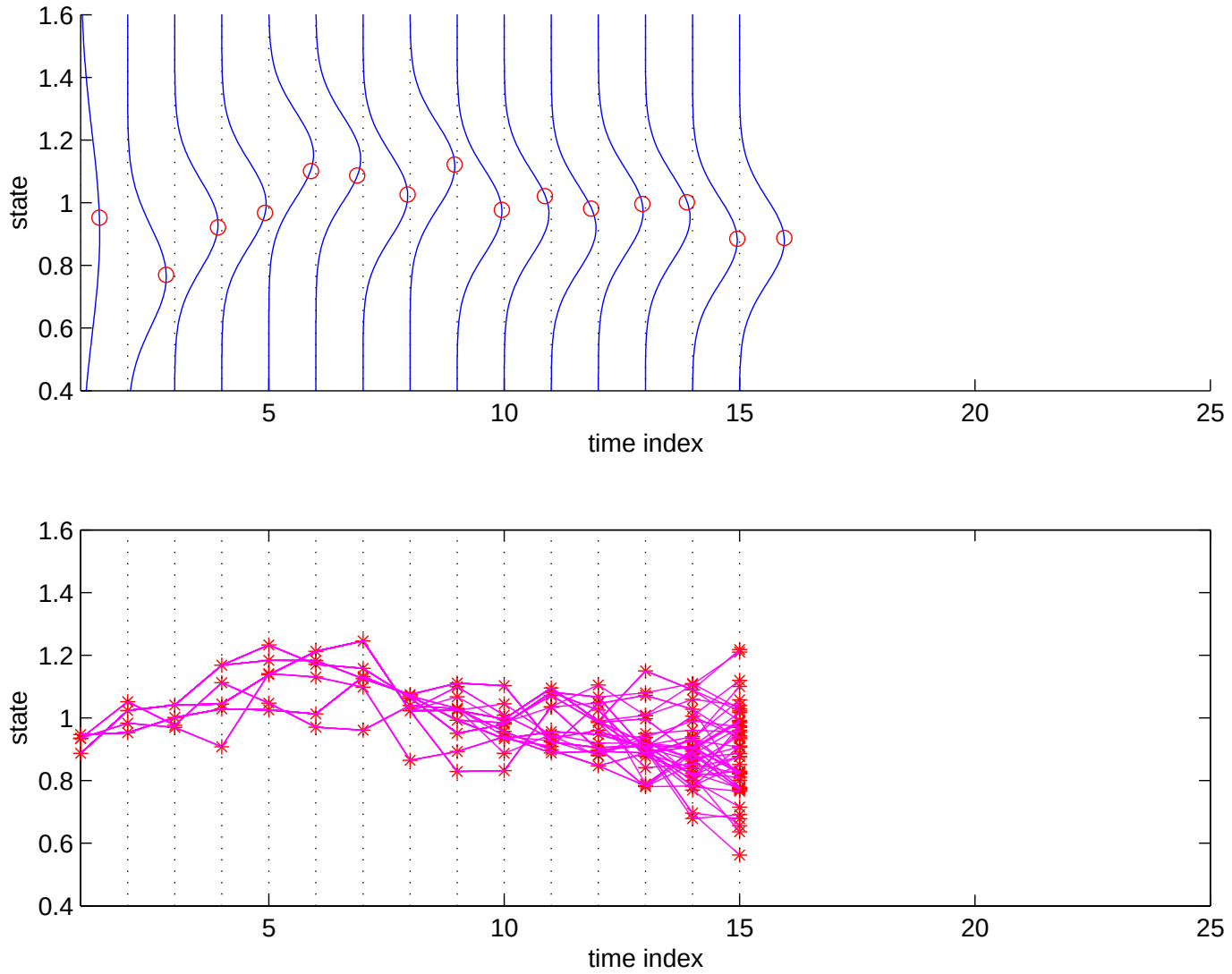

Predictive densities and evolution of the particle paths



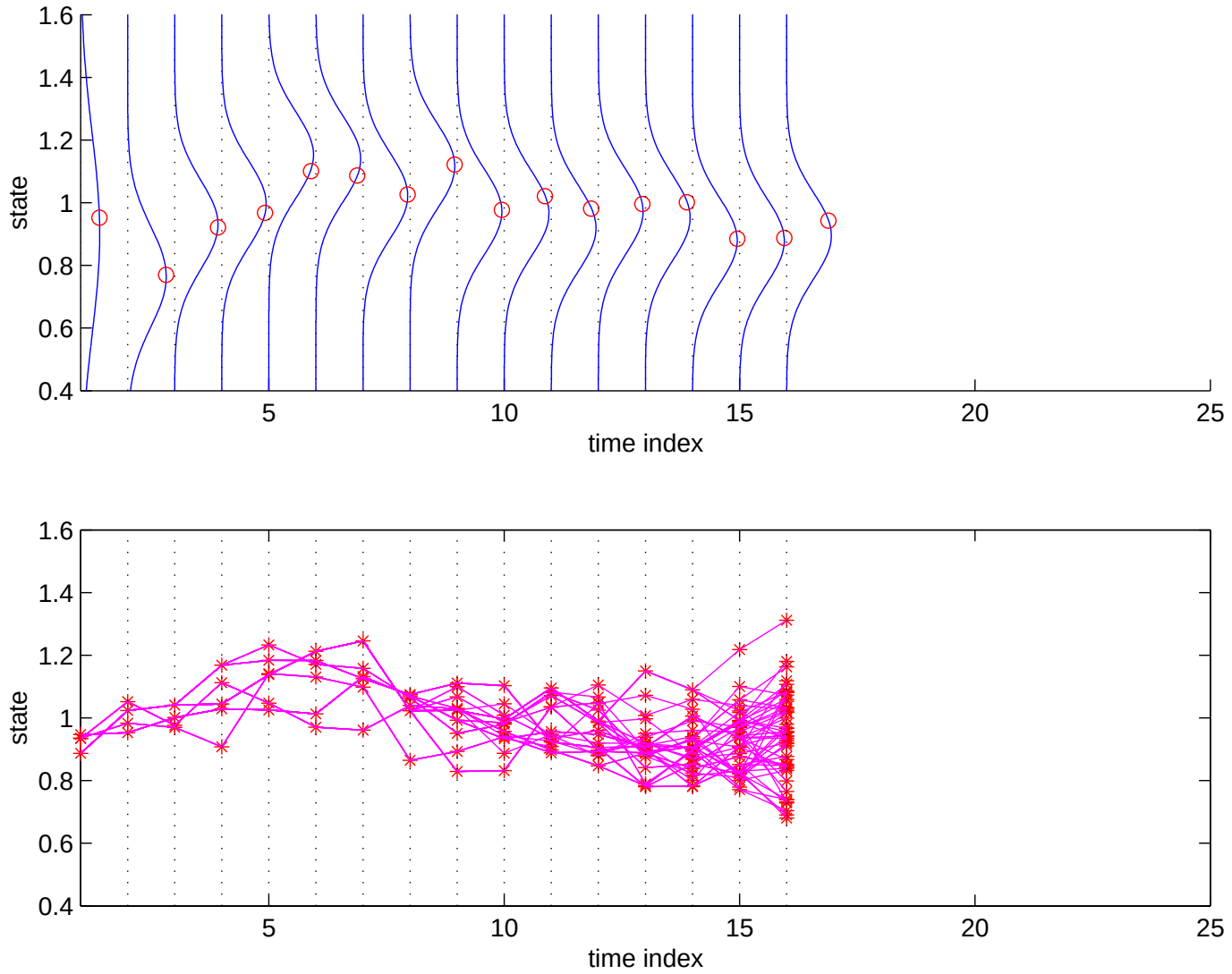

Predictive densities and evolution of the particle paths



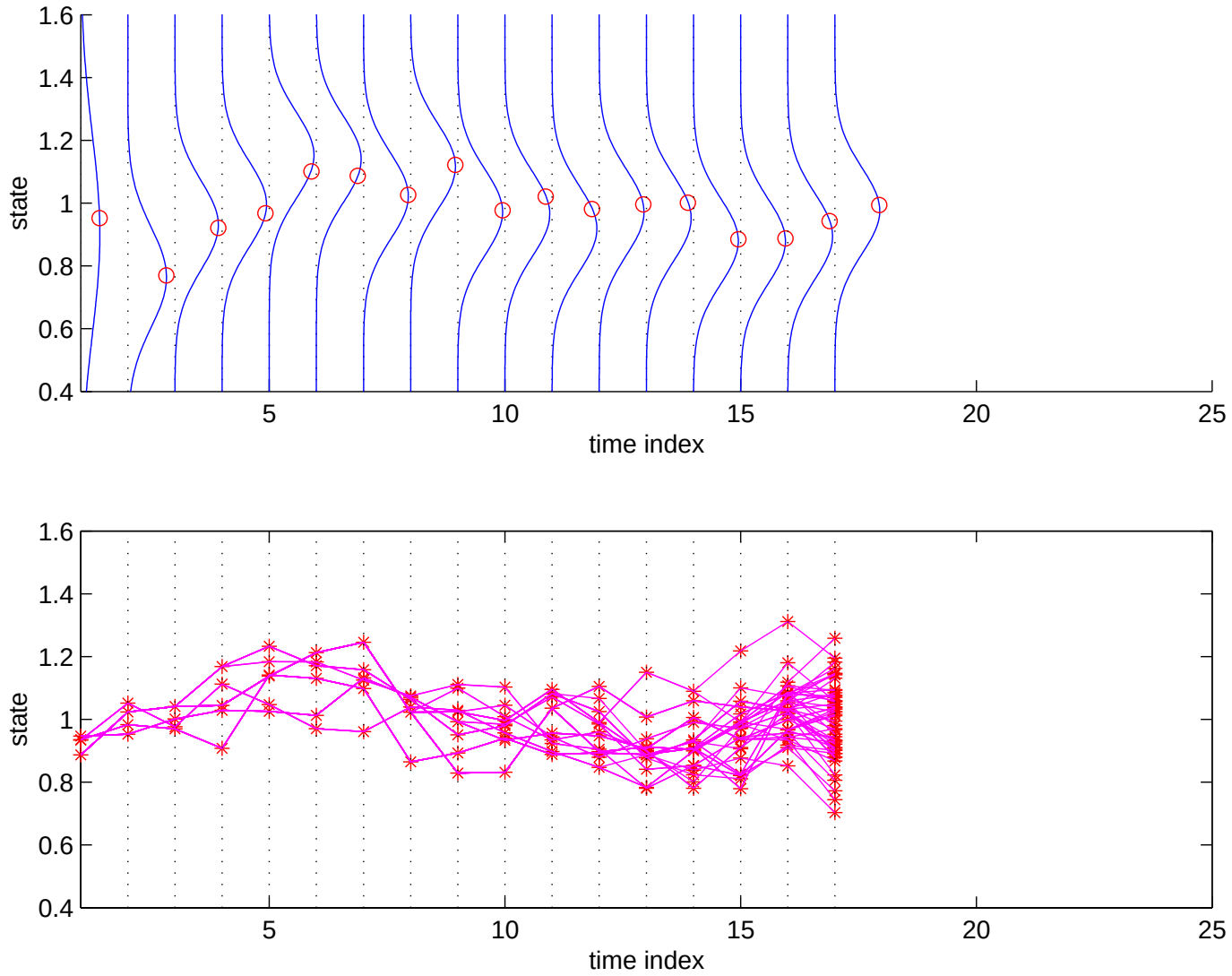

Predictive densities and evolution of the particle paths



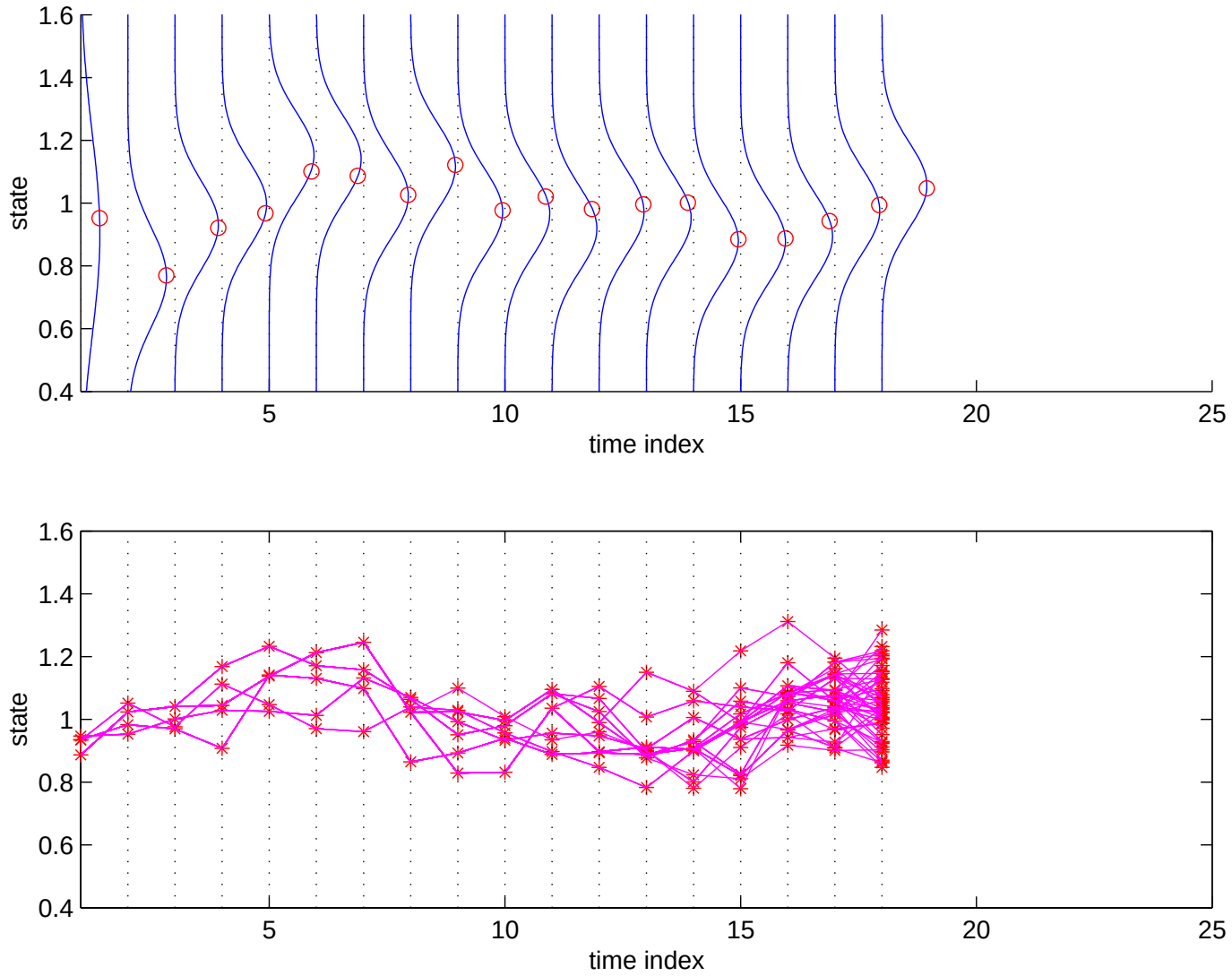

Predictive densities and evolution of the particle paths



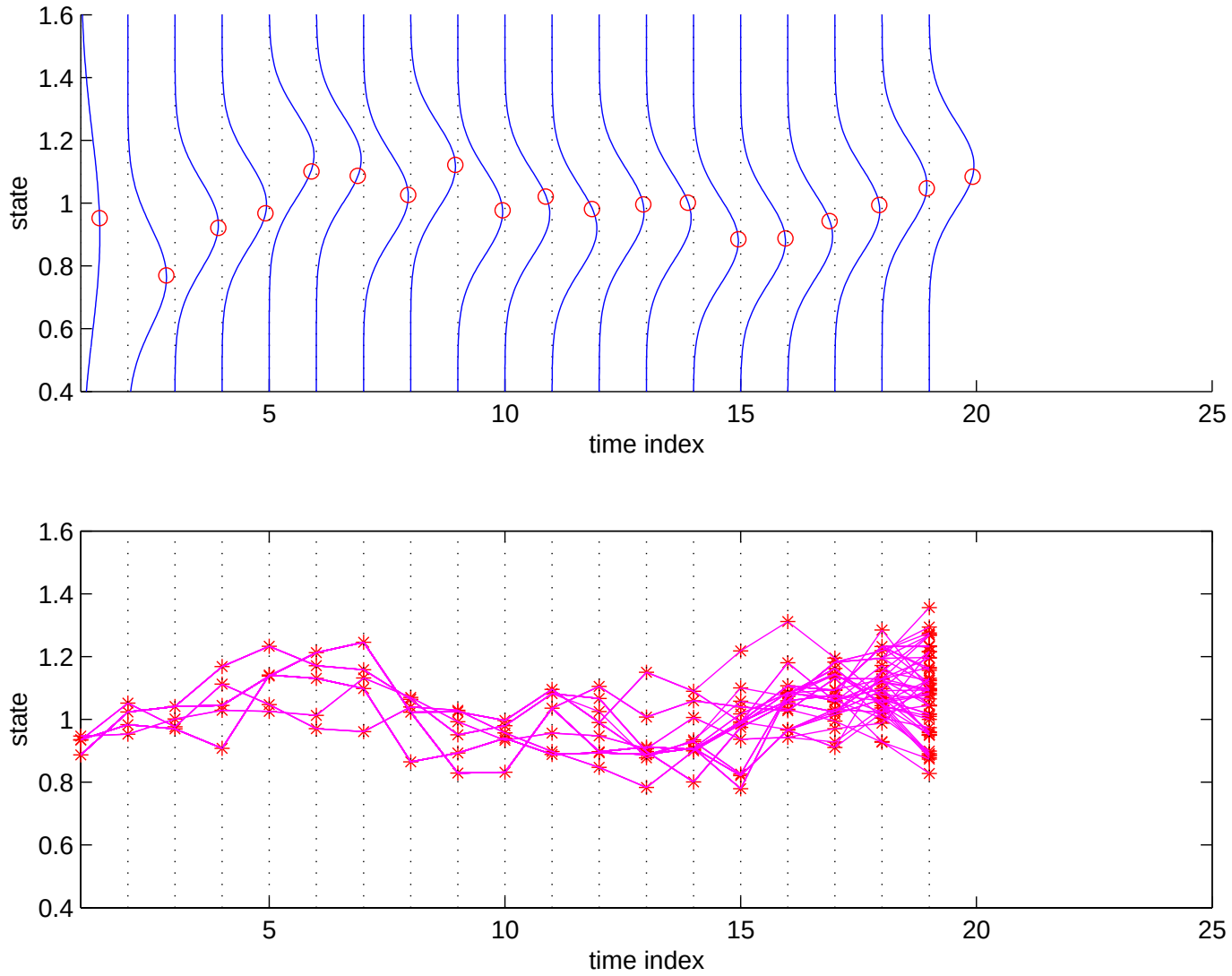

Predictive densities and evolution of the particle paths



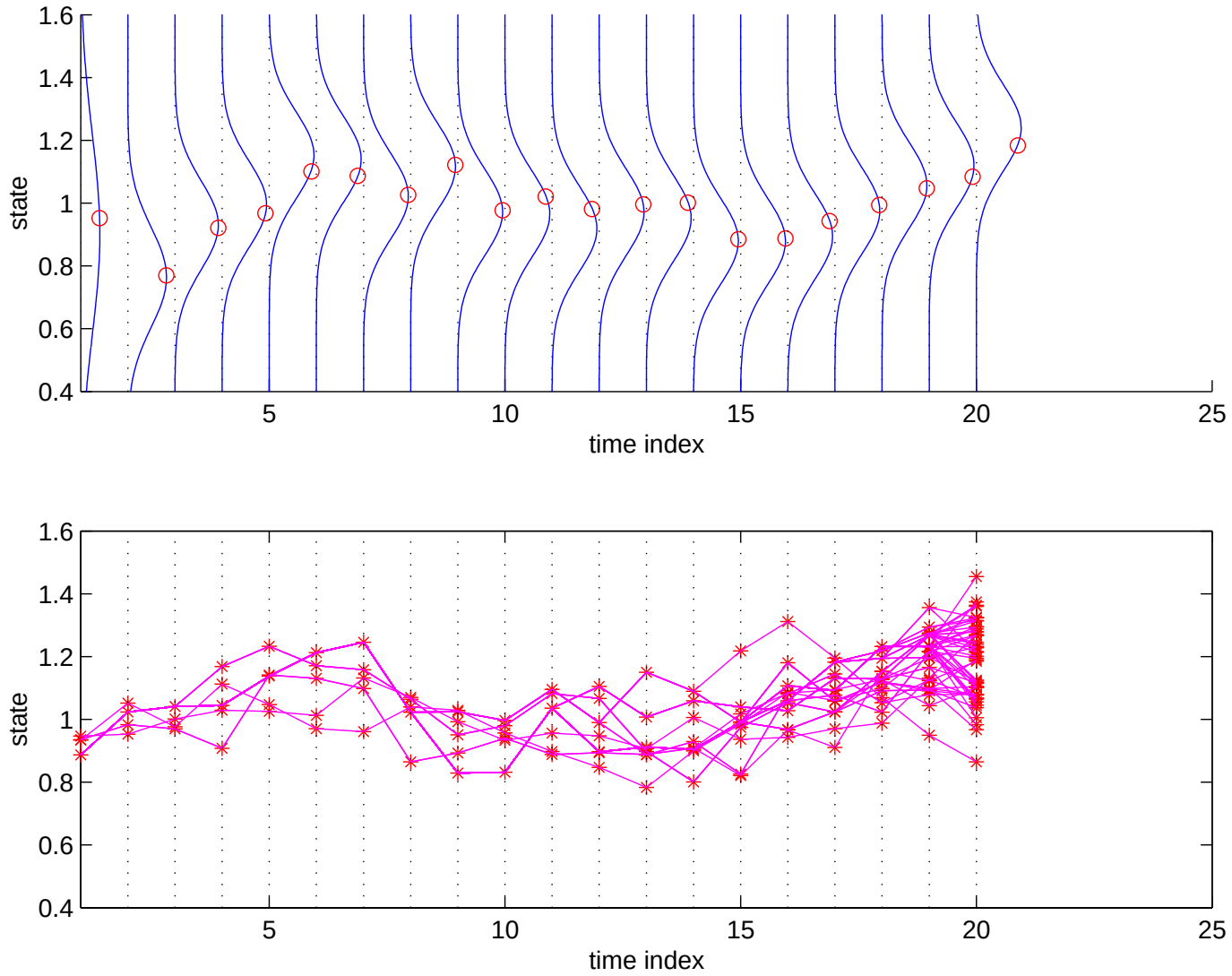

Predictive densities and evolution of the particle paths



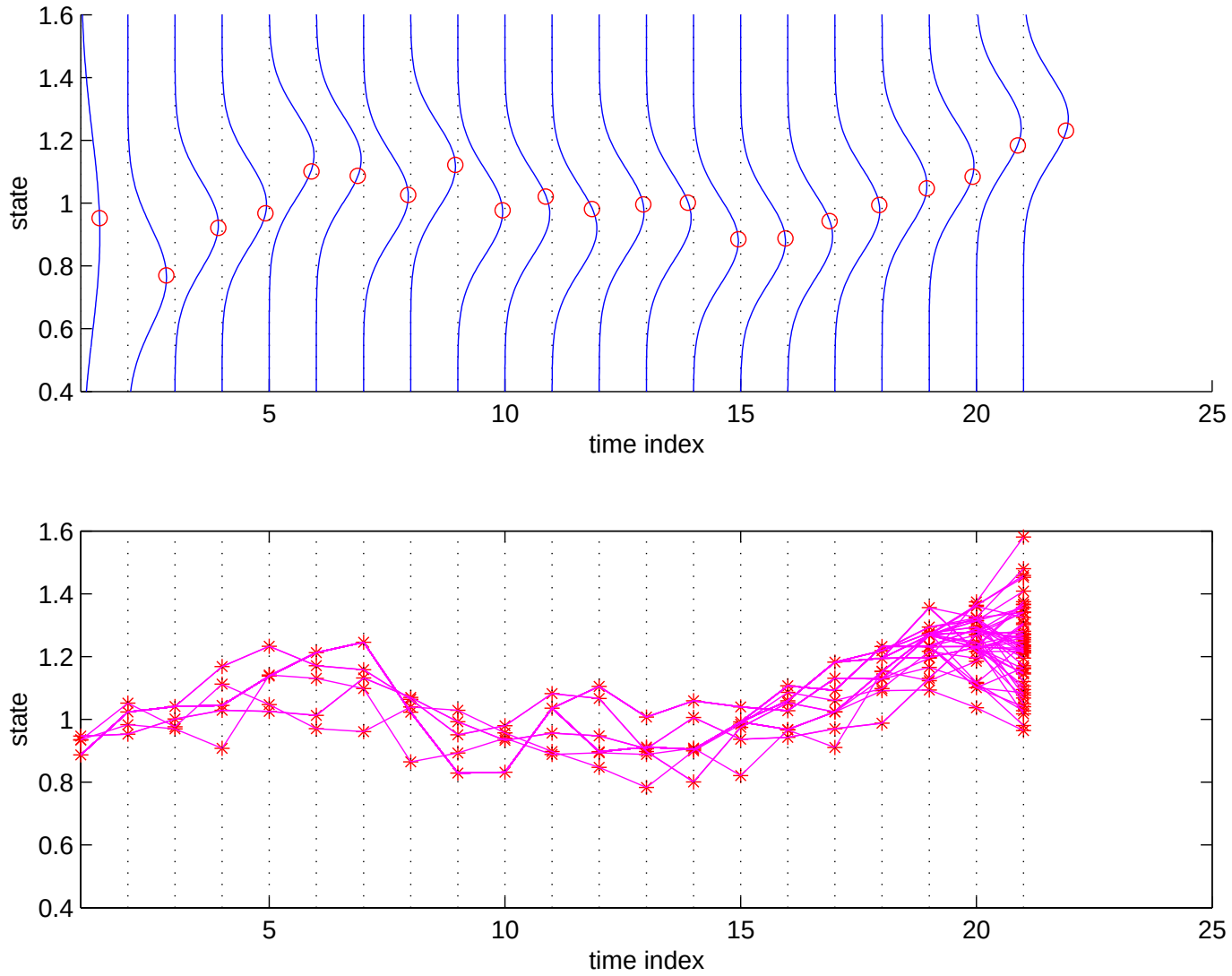

Predictive densities and evolution of the particle paths



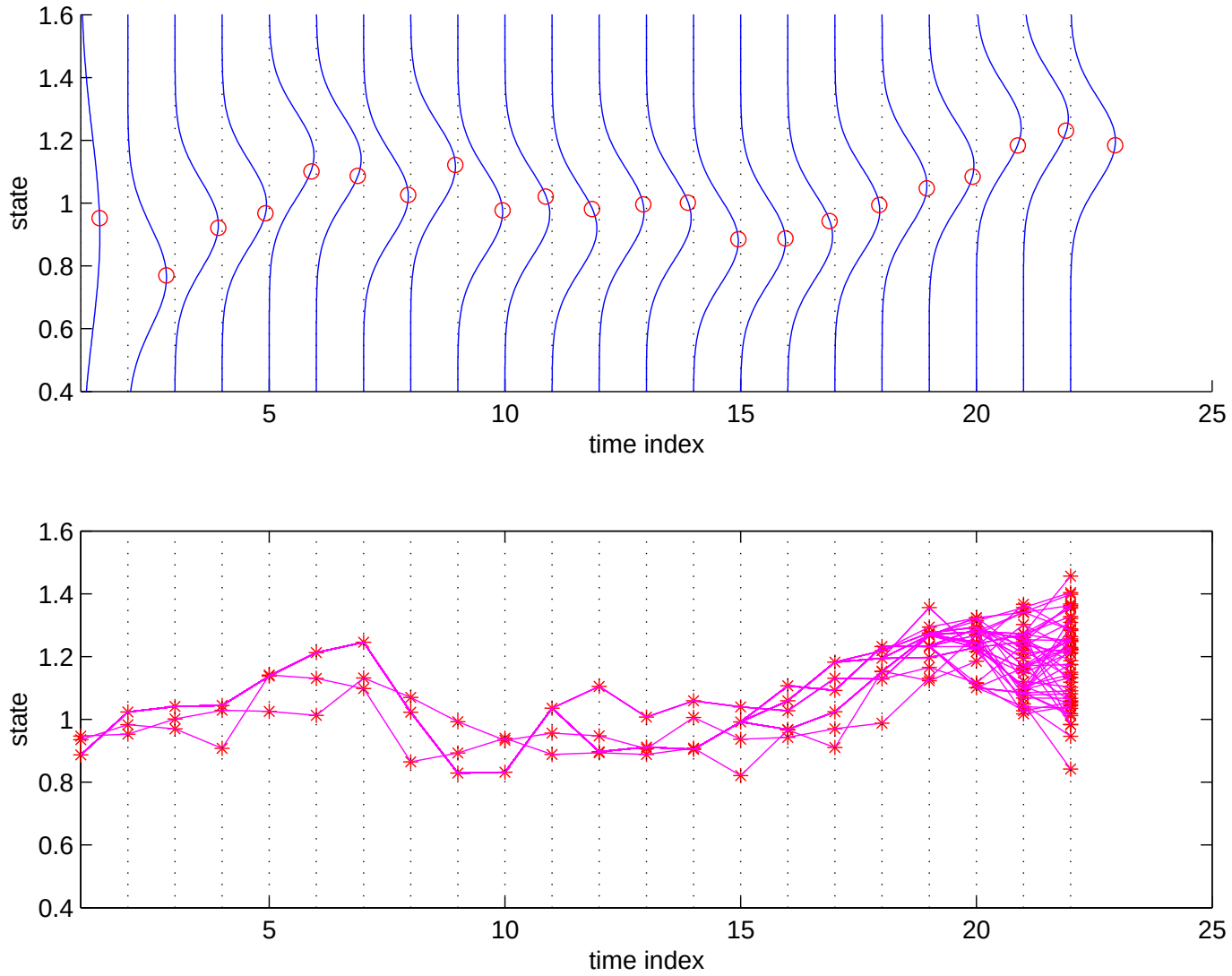

Predictive densities and evolution of the particle paths



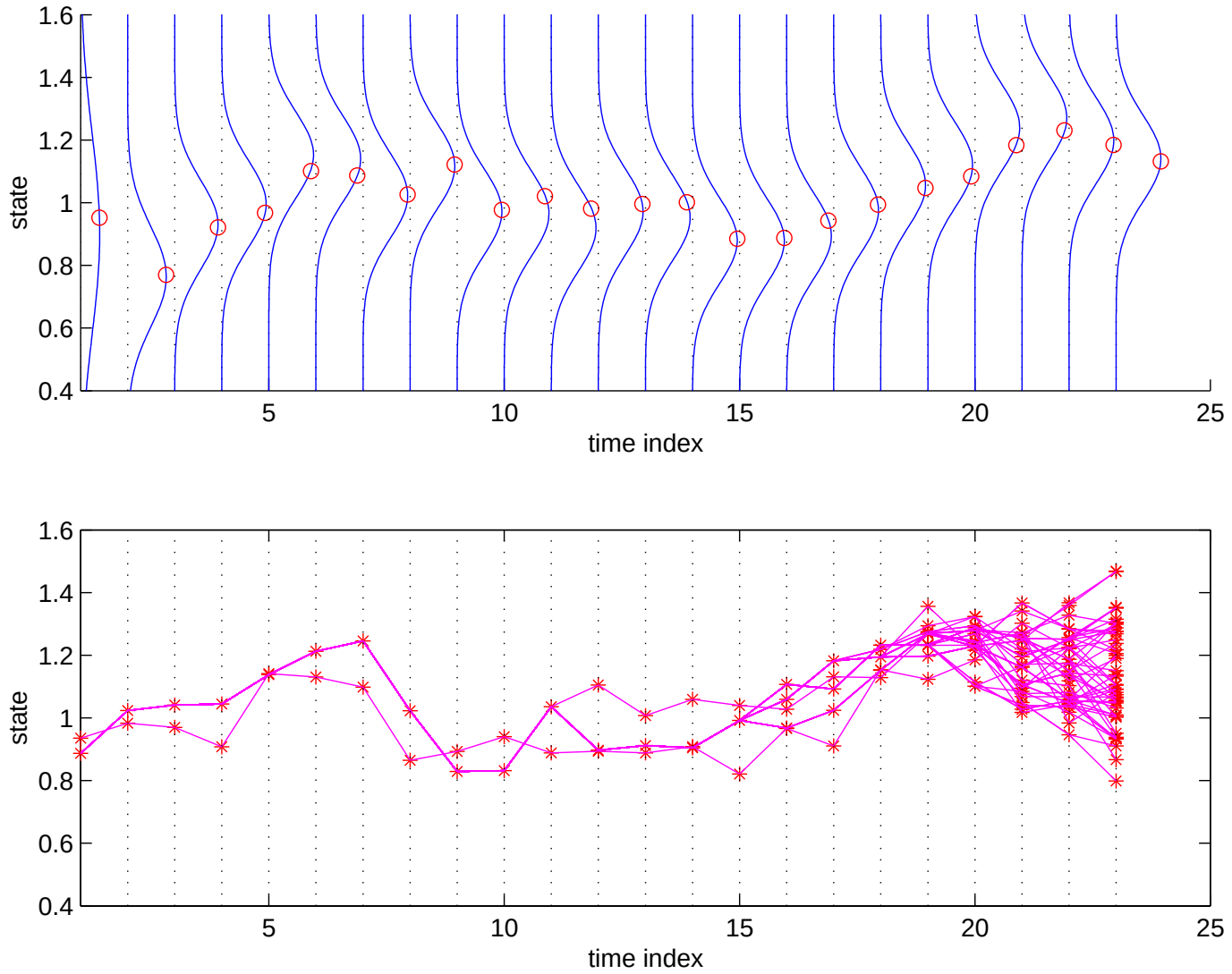

Predictive densities and evolution of the particle paths



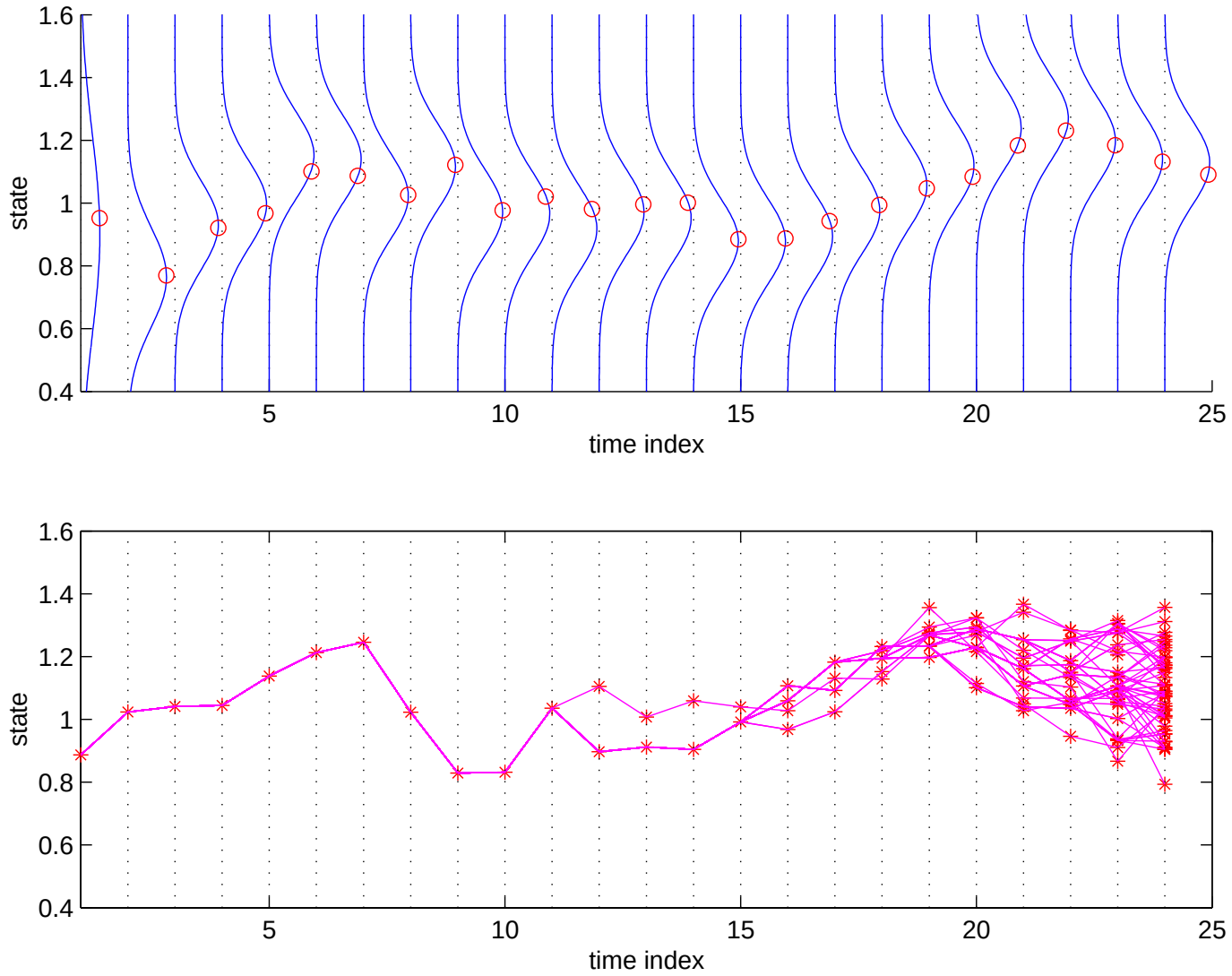

Predictive densities and evolution of the particle paths




Predictive densities and evolution of the particle paths

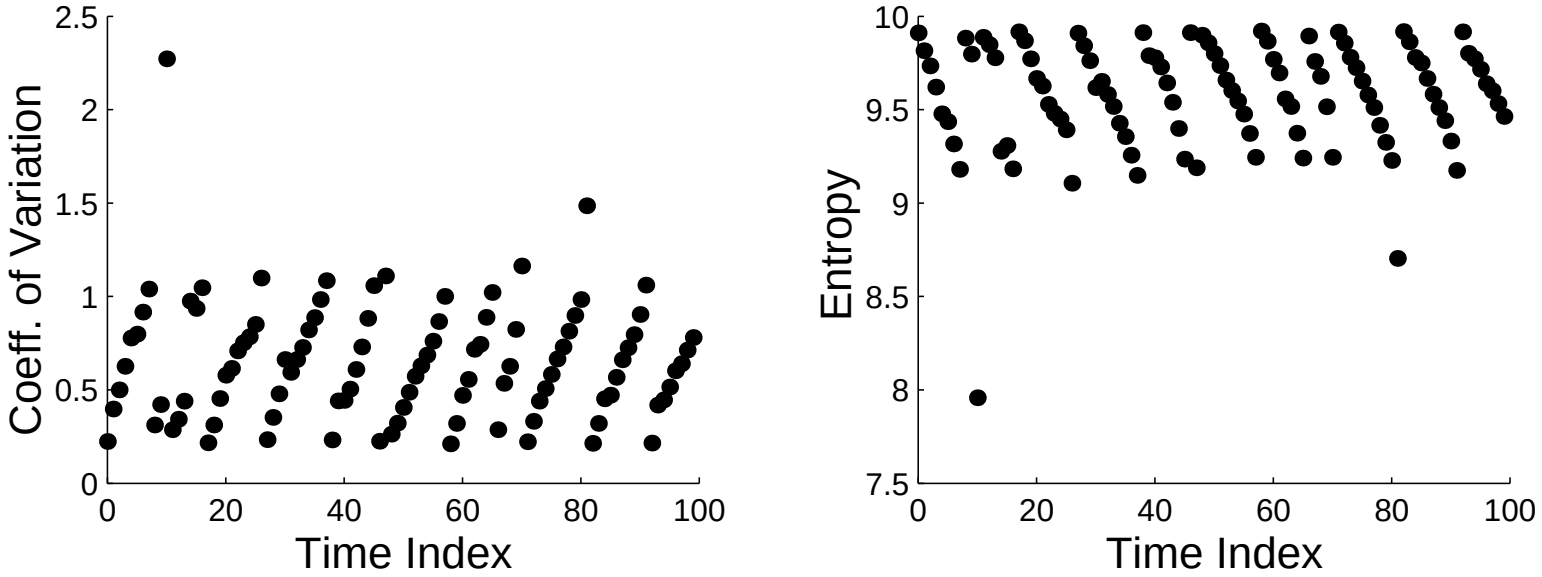



Predictive densities and evolution of the particle paths



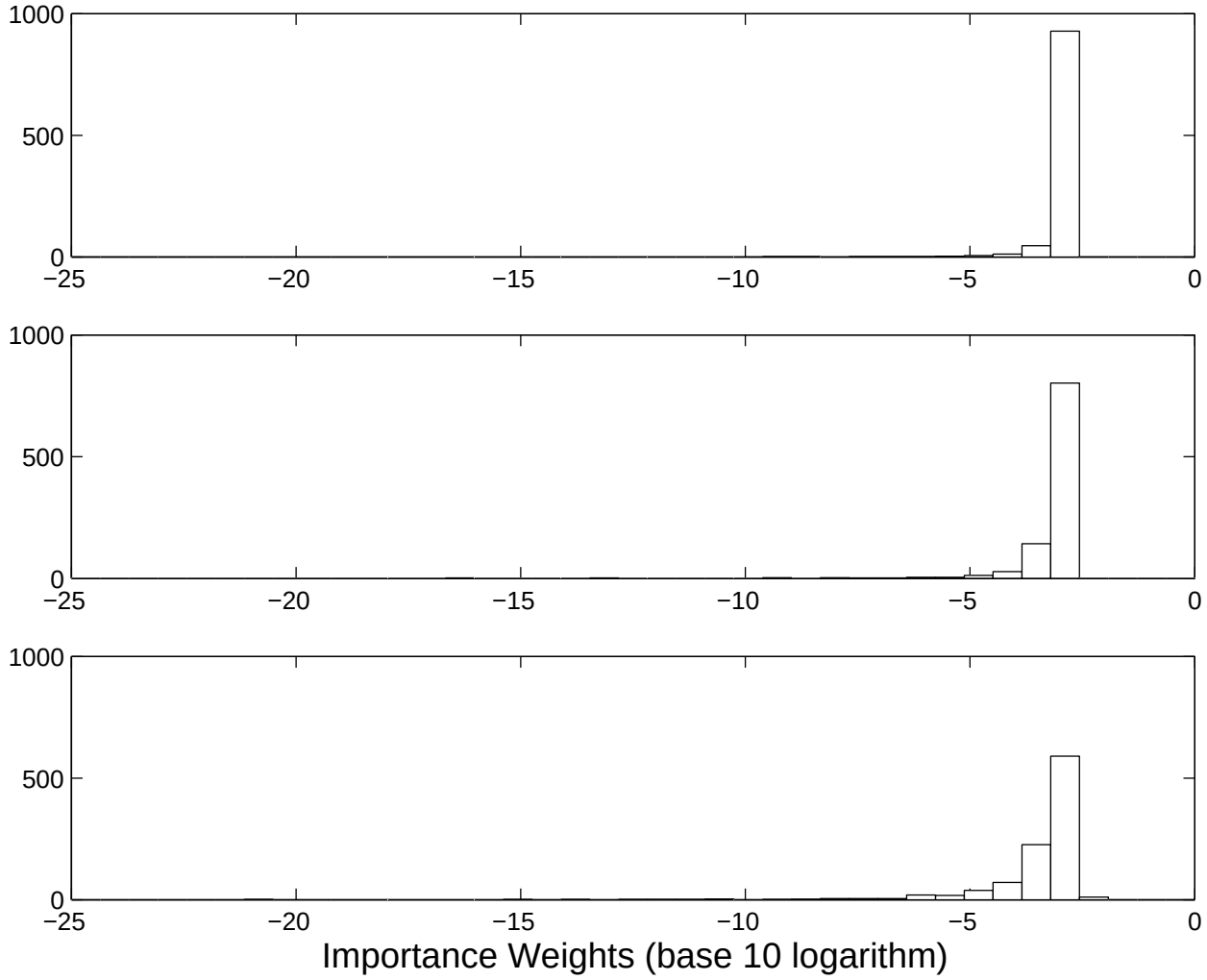

Predictive densities and evolution of the particle paths

# Application to the Stochastic Volatility Model (contd.)




Coefficient of variation (left) and entropy of the normalized importance weights as a function of the number of iterations when using resampling triggered by $\mathrm{CV}_N(\omega) > 1$.

# Application to the Stochastic Volatility Model (contd.)




Histograms of the base 10 logarithm of the normalized importance weights after (from top to bottom) 1, 10 and 100 iterations when using resampling triggered by $\mathrm{CV}_N(\omega) > 1$.

# Roadmap

1. What is a Hidden Markov Model?
2. Filtering and Smoothing Recursions
3. Monte Carlo, Importance Sampling and Sampling Importance Resampling
4. Sequential Importance Sampling
5. Sequential Importance Sampling with Resampling
6. More Sequential Monte Carlo Algorithms*
7. Approximation of Sums Functionals and Parameter Estimation*

# Alternatives to SISR

- The resampling step in the SISR algorithm can be seen as a method to sample approximately under $\phi_{0:k+1|k+1}$ given the current particle approximation $\hat{\phi}_{0:k|k}$.
- This alternative way of thinking about resampling suggests several sequential Monte Carlo variants.

# Sequential Monte Carlo Reinterpreted

Recall that each update consists of two steps.

**Prediction step:** compute the one-step ahead predictive distribution from the filtering distribution.

$$\phi_{0:k+1|k} = \phi_{0:k|k} Q$$

**Correction step (Bayes):** compute the filtering distribution from the predictive distribution by taking into account the new observation $Y_{k+1}$:

$$\phi_{0:k+1|k+1}(f_{k+1}) = \frac{\int \cdots \int f_{k+1}(x_{0:k+1})\, g_{k+1}(x_{k+1}) \phi_{0:k+1|k}(dx_{0:k+1})}{\int g_{k+1}(x_{k+1})\, \phi_{k+1|k}(dx_{0:k+1})} \,.$$

$$\phi_{0:k|k} \overset{\text{prediction}}{\longrightarrow} \phi_{0:k+1|k} \overset{\text{correction}}{\longrightarrow} \phi_{0:k+1|k+1}$$

# Sequential Monte Carlo Reinterpreted

Replace $\phi_{0:k|k}$ by the *empirical filtering distribution*.

$$\hat{\phi}_{0:k|k} = \sum_{i=1}^{N} \frac{\omega_k^i}{\sum_{j=1}^{N} \omega_k^j} \delta_{\xi_{0:k}^i} \ .$$

Applying the prediction and then the correction step to this approximation yields

$$\hat{\phi}_{0:k|k} \overset{\text{prediction}}{\longrightarrow} \tilde{\phi}_{0:k+1|k} = \sum_{i=1}^{N} \frac{\omega_k^i}{\sum_{j=1}^{N} \omega_k^j} \delta_{\xi_{0:k}^i} Q(\xi_k^i, \cdot)$$

$$\overset{\text{correction}}{\longrightarrow} \tilde{\phi}_{0:k+1|k+1}(f_{k+1}) = \frac{\sum_{i=1}^{N} \omega_k^i \int f_{k+1}(\xi_{0:k}^i, x)\, g_{k+1}(x) Q(\xi_k^i, dx)}{\sum_{i=1}^{N} \omega_k^i \int g_{k+1}(x) Q(\xi_k^i, dx)} \ .$$

The distribution $\tilde{\phi}_{0:k+1|k+1}$ is sometimes called the *empirical filtering distribution*. It is in some sense the best approximation to $\phi_{0:k+1|k+1}$ based on the knowledge of $\hat{\phi}_{0:k|k}$. It is obvioulsy not in general a distribution supported by a finite set of points!

# Sequential Monte Carlo Reinterpreted

The empirical filtering distribution is a mixture distribution

$$\tilde{\phi}_{0:k+1|k+1} = \sum_{i=1}^{N} \frac{\omega_k^i \gamma_k(\xi_k^i)}{\sum_{j=1}^{N} \omega_k^j \gamma_k(\xi_k^j)} \int f_{k+1}(\xi_{0:k}^i, x)\, T_k(\xi_k^i, dx) \ ,$$

where

$$\gamma_k(x) = \int Q(x, dx') g_{k+1}(x') \ ,$$

$$T_k(x, A) = \frac{\int_A Q(x, dx') g_{k+1}(x')}{\gamma_k(x)} \ .$$

Direct sampling from this distribution is usually not possible (because sampling from $T_k$ and evaluating $\gamma_k$ aren't either).

# Auxiliary Sampling

But we may in general use importance sampling or SIR, proposing new points $\tilde{\xi}^1_{k+1}, \ldots, \tilde{\xi}^N_{k+1}$ under the mixture

$$\rho_{0:k+1}(f_{k+1}) = \sum_{i=1}^{N} \frac{\omega_k^i \ \tau_k^i}{\sum_{j=1}^N \omega_k^j \ \tau_k^j} \int f(\xi_{0:k}^i, x) \ R_k(\xi_k^i, dx) \ ,$$

where $\tau_k^1, \ldots, \tau_k^N$ are user-selected adjusment weigths and $R_k$ is a kernel which is easy to sample from. In doing so, we first need to draw mixture component indicators $I_k^1, \ldots, I_k^N$.

It is easilly checked that the importance weigths are then given by

$$\omega_{k+1}^i = \frac{g_{k+1}(\tilde{\xi}_{k+1}^i)}{\tau_k^{I_k^i}} \frac{dQ(\xi_k^{I_k^i}, \cdot)}{dR_k(\xi_k^{I_k^i}, \cdot)}(\tilde{\xi}_{k+1}^i) \ .$$

This strategy named auxiliary sampling and proposed by (Pitt & Shephard, 1999) is often usefull in practice when combined with clever ways of setting $\{\tau_k^i\}_{i=1,\ldots,N}$ and $R_k$.

# IID Sampling

It is interesting to consider what happens in cases where sampling from $T_k$ and evaluating $\gamma_k$ is feasible (i.e. when $\tau_k^i = \gamma_k(\xi_k^i)$ and $R_k = T_k$):

**Weight computation:** For $i = 1, \ldots, N$, compute the (unnormalized) importance weights

$$\alpha_k^i = \gamma_k(\xi_k^i) \ .$$

**Selection:** Draw $I_{k+1}^1, \ldots, I_{k+1}^N$ conditionally i.i.d. given $\{\xi_{0:k}^i\}_{1 \leq i \leq N}$, with probabilities $\mathrm{P}(I_{k+1}^1 = j)$ proportional to $\alpha_k^j$, $j = 1, \ldots, N$.

**Sampling:** Draw $\tilde{\xi}_{k+1}^1, \ldots, \tilde{\xi}_{k+1}^N$ conditionally independently given $\{\xi_{0:k}^i\}_{1 \leq i \leq N}$ and $\{I_{k+1}^i\}_{1 \leq i \leq N}$, with distribution $\tilde{\xi}_{k+1}^i \sim T_k(\xi_k^{I_{k+1}^i}, \cdot)$. Set $\xi_{0:k+1}^i = (\xi_{0:k}^{I_{k+1}^i}, \tilde{\xi}_{k+1}^i)$ and $\omega_{k+1}^i = 1$ for $i = 1, \ldots, N$.

# IID Sampling

- Compared with the SISR Algorithm for the particular choice $R_k = T_k$, the IID sampling algorithm differs only by the order in which the sampling (or mutation) and selection operations are performed.

# IID Sampling

- Compared with the SISR Algorithm for the particular choice $R_k = T_k$, the IID sampling algorithm differs only by the order in which the sampling (or mutation) and selection operations are performed.

- The SISR Algorithm prescribes that each trajectory be first extended by setting $\xi^i_{0:k+1} = (\xi^i_{0:k}\,,\,\tilde{\xi}^i_{k+1})$ where $\tilde{\xi}^i_{k+1}$ is drawn from $T_k(\xi^i_k, \cdot)$. Then resampling is performed in the population of extended trajectories according to their importance weights.

# IID Sampling

- Compared with the SISR Algorithm for the particular choice $R_k = T_k$, the IID sampling algorithm differs only by the order in which the sampling (or mutation) and selection operations are performed.

- The SISR Algorithm prescribes that each trajectory be first extended by setting $\xi^i_{0:k+1} = (\xi^i_{0:k}\ ,\ \tilde{\xi}^i_{k+1})$ where $\tilde{\xi}^i_{k+1}$ is drawn from $T_k(\xi^i_k, \cdot)$. Then resampling is performed in the population of extended trajectories according to their importance weights.

- In contrast, the IID sampling algorithm first selects the trajectories based on the weights $\alpha^i_k$ and then simulate an independent extension for each selected trajectory. The new particles $\tilde{\xi}^1_{k+1}, \ldots, \tilde{\xi}^N_{k+1}$ are conditionally independent given the current generation of particles $\{\xi^i_k\}_{i=1,\ldots,N}$.

# IID Sampling

- Compared with the SISR Algorithm for the particular choice $R_k = T_k$, the IID sampling algorithm differs only by the order in which the sampling (or mutation) and selection operations are performed.

- The SISR Algorithm prescribes that each trajectory be first extended by setting $\xi^i_{0:k+1} = (\xi^i_{0:k}\ ,\ \tilde{\xi}^i_{k+1})$ where $\tilde{\xi}^i_{k+1}$ is drawn from $T_k(\xi^i_k, \cdot)$. Then resampling is performed in the population of extended trajectories according to their importance weights.

- In contrast, the IID sampling algorithm first selects the trajectories based on the weights $\alpha^i_k$ and then simulate an independent extension for each selected trajectory. The new particles $\tilde{\xi}^1_{k+1}, \ldots, \tilde{\xi}^N_{k+1}$ are conditionally independent given the current generation of particles $\{\xi^i_k\}_{i=1,\ldots,N}$.

- This is of course only possible because the optimal importance kernel $T_k$ is used as instrumental kernel which renders the incremental weights independent of the position of the particle at index $k+1$ and thus allow for early selection. This way of proceeding is provably better than SISR with $R_k = T_k$ (see text).

# Roadmap

1. What is a Hidden Markov Model?
2. Filtering and Smoothing Recursions
3. Monte Carlo, Importance Sampling and Sampling Importance Resampling
4. Sequential Importance Sampling
5. Sequential Importance Sampling with Resampling
6. More Sequential Monte Carlo Algorithms*
7. Approximation of Sums Functionals and Parameter Estimation*

# EM and Friends in HMMs—A Long Story Made Short

- If the HMM has some unknown parameters $\theta$, likelihood-based parameter inference, be it through Expectation-Maximization (EM) or gradient-based approaches, (only) requires to be able to compute quantities of the form

$$\mathrm{E}\left[\sum_{k=0}^{n-1} s_i(X_k, X_{k+1}) \,\middle|\, Y_{0:n}\,;\theta\right] \quad ,$$

  for some model-dependent functions $s_i$.

# EM and Friends in HMMs—A Long Story Made Short

- If the HMM has some unknown parameters $\theta$, likelihood-based parameter inference, be it through Expectation-Maximization (EM) or gradient-based approaches, (only) requires to be able to compute quantities of the form

$$\mathrm{E}\left[\sum_{k=0}^{n-1} s_i(X_k, X_{k+1}) \middle| Y_{0:n}\,;\theta\right] \quad ,$$

  for some model-dependent functions $s_i$.

- If exact computation is not feasible, we may use approximate Monte-Carlo evaluation in combination with variants of the former methods (MCEM, SAME, SAEM, stochastic gradient, etc.)

# EM and Friends in HMMs—A Long Story Made Short

- If the HMM has some unknown parameters $\theta$, likelihood-based parameter inference, be it through Expectation-Maximization (EM) or gradient-based approaches, (only) requires to be able to compute quantities of the form

$$\mathrm{E}\left[\sum_{k=0}^{n-1} s_i(X_k, X_{k+1}) \,\middle|\, Y_{0:n}\,;\theta\right] \quad ,$$

  for some model-dependent functions $s_i$.

- If exact computation is not feasible, we may use approximate Monte-Carlo evaluation in combination with variants of the former methods (MCEM, SAME, SAEM, stochastic gradient, etc.)

**Are sequential Monte Carlo methods appropriate for this task?**

# A Very simple Example

In the stochastic volatility model, one need to evaluate $\mathrm{E}\left[S_i(X_{0:n})\middle| Y_{0:n}\right]$, for $0 \leq i \leq 4$ with

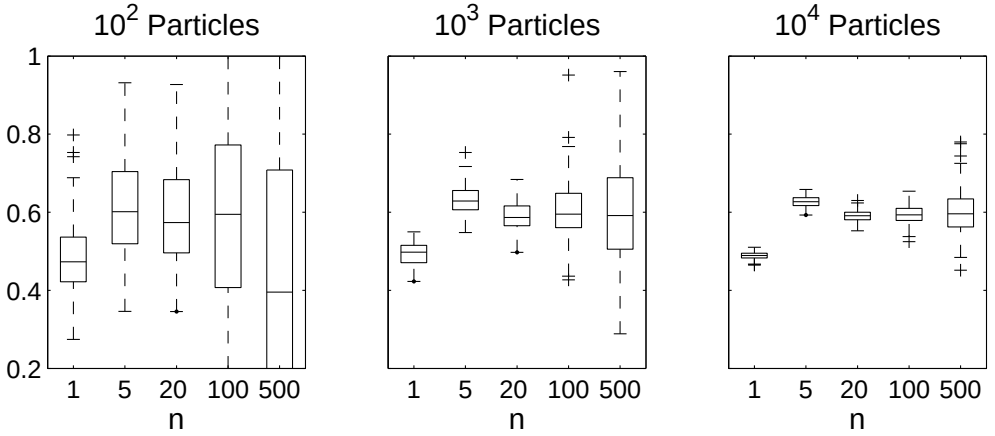
$$S_0(x_{0:n}) = x_0^2 \ , \qquad S_1(x_{0:n}) = \sum_{k=0}^{n-1} x_k^2 \ , \qquad S_2(x_{0:n}) = \sum_{k=1}^{n} x_k^2 \ ,$$

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

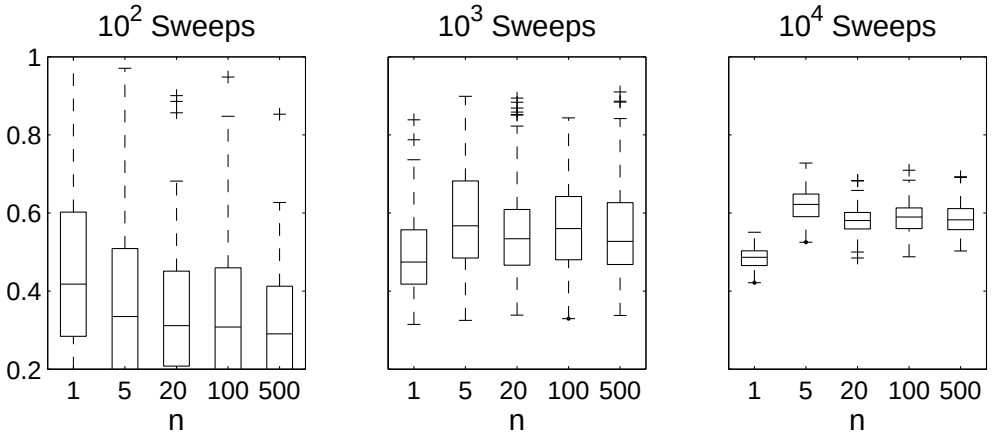
Lets first look at the simple case of $S_0$:

$\sum_{i=1}^N \omega_n^i \left\{\xi_{0:n}^i(0)\right\}^2$ is the sequential Monte Carlo estimate of $\mathrm{E}\left(X_0^2\middle| Y_{0:n}\right)$.

# Smoothing for $X_0^2$ contd.

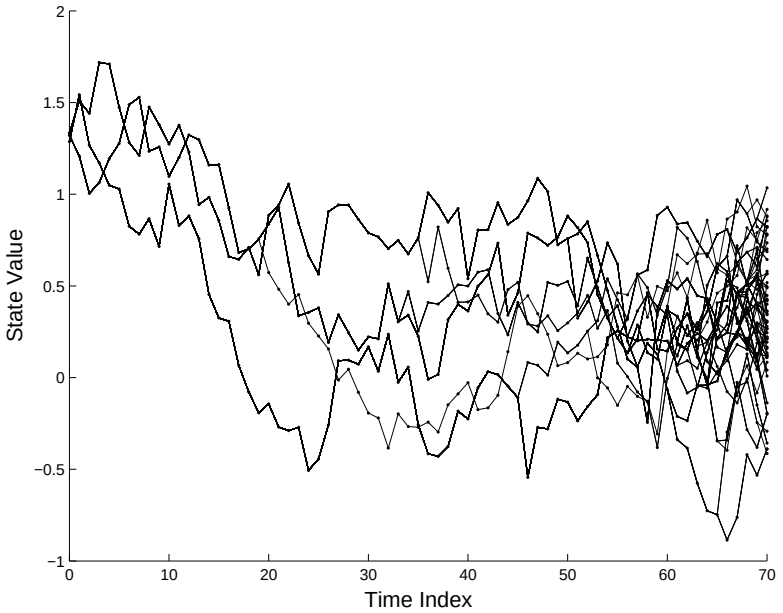



Box and whisker plots of particle estimates of $\int x^2 \, \phi_{0|n}(dx)$ for $n = 1$, 5, 20, 30 and 500, and particle population sizes $N = 10^2, 10^3$ and $10^4$.




Same figure as above for MCMC estimates of $\int x^2 \, \phi_{0|n}(dx)$, where $N$ refers to the number of MCMC sweeps though the data, using an MCMC sampler (single site MH).

# Smoothing for $X_0^2$ contd.




Particle trajectories at time $n = 70$ for the stochastic volatility model with $N = 100$ particles and systematic resampling.

$\Longrightarrow$ Using $k < n$ is beneficial! Properly setting $k$ corresponds to a bias-variance tradeoff: $k \uparrow$ bias decreases* , $k \downarrow$ variance decreases.

*Under mixing assumption on the hidden chain, it may be shown that $\phi_{0|k}$ and $\phi_{0|n}$ are indeed close when $k$ is large (see text).

# Smoothing for More General Functions

**Why is it a fixed-dimensional problem?**

For importance sampling, obviously

$$\sum_{i=1}^{N} \omega_{n+1}^{i} \underbrace{\sum_{k=0}^{n} s\left(\xi_{0:n+1}^{i}(k), \xi_{0:n+1}^{i}(k+1)\right)}_{\gamma_{n+1}^{i}} = \sum_{i=1}^{N} \omega_{n+1}^{i} \sum_{k=0}^{n} s\left(\xi_{k}^{i}, \xi_{k+1}^{i}\right)$$

$$= \sum_{i=1}^{N} \omega_{n+1}^{i} \left\{ s\left(\xi_{n}^{i}, \xi_{n+1}^{i}\right) + \underbrace{\sum_{k=0}^{n-1} s\left(\xi_{k}^{i}, \xi_{k+1}^{i}\right)}_{\gamma_{n}^{i}} \right\}.$$

# Smoothing for More General Functions

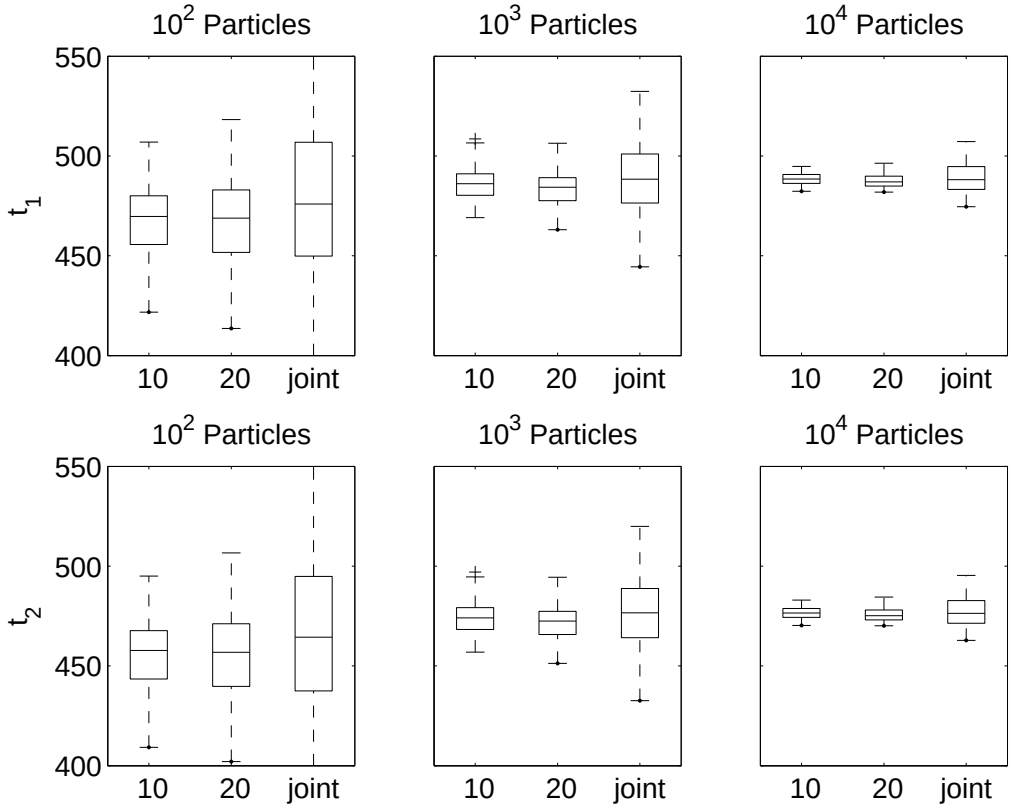
When resampling occurs

$$\sum_{i=1}^{N} \frac{1}{N} \underbrace{\sum_{k=0}^{n} s\left(\xi^i_{0:n+1}(k), \xi^i_{0:n+1}(k+1)\right)}_{\gamma^i_{n+1}}$$

$$= \sum_{i=1}^{N} \frac{1}{N} \left\{ s\left(\xi^{I^i_{n+1}}_{0:n}(n), \xi^i_{n+1}\right) + \underbrace{\sum_{k=0}^{n-1} s\left(\xi^{I^i_{n+1}}_{0:n}(k), \xi^{I^i_{n+1}}_{0:n}(k+1)\right)}_{\gamma_n^{I^i_{n+1}}} \right\}.$$

$\Longrightarrow$ We never need to store more than $\omega^i_n, \xi^i_n, \gamma^i_n$ for $i = 1, \ldots, N$.

# Smoothing in the Stochastic Volatility Model




Box and whisker plots of particle estimators of the expectations of the two statistics $t_{n,1}(x_{0:n}) = \sum_{k=0}^{n-1} x_k^2$ (top) and $t_{n,2}(x_{0:n}) = \sum_{k=1}^{n} x_k x_{k-1}$ (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 $k = 10$ and $20$ and full path "joint" particle approximation. The plots are based on 100 independent replications.

# Fixed Lag-Smoothing

The principle is to replace $\phi_{l|n}$ by $\phi_{l|(l+k)\wedge n}$ for a fixed delay $k$:

$\sum_{l=0}^{n} \mathrm{E}\left[ s(X_l, X_{l+1}) \middle| Y_{0:(l+k)\wedge(n+1)} \right]$ may be approximated by

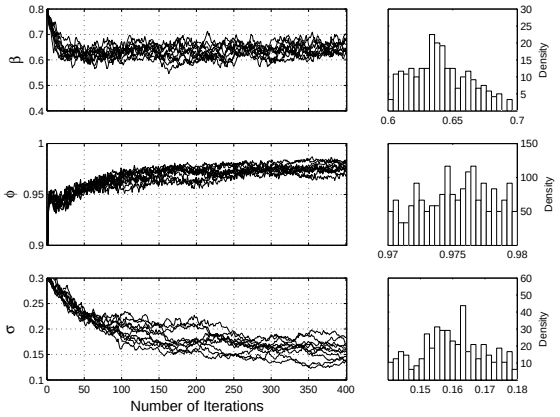
$$\sum_{i=1}^{N} \frac{1}{N} \sum_{l=0}^{n} s\left( \xi^i_{0:(l+k)\wedge(n+1)}(l), \xi^i_{0:(l+k)\wedge(n+1)}(l+1) \right) =$$

$$\sum_{i=1}^{N} \frac{1}{N} \left\{ s\left( \xi^{I^i_{n+1}}_{0:n}(n), \xi^i_{n+1} \right) + \sum_{l=n-k+2}^{n-1} \underbrace{s\left( \xi^{I^i_{n+1}}_{0:n}(l), \xi^{I^i_{n+1}}_{0:n}(l+1) \right)}_{\text{terms that need to be kept individually}} \right\}$$

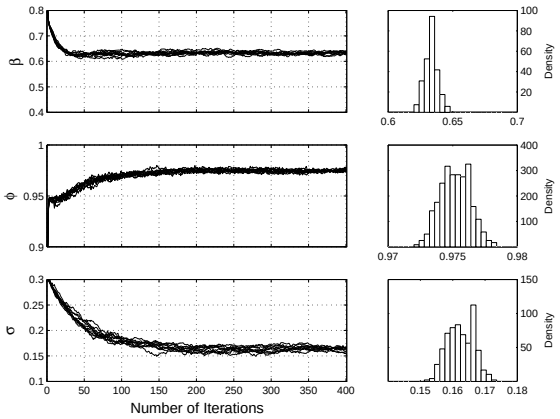
$$+ \underbrace{\sum_{i=1}^{N} \frac{1}{N} \left\{ s\left( \xi^{I^i_{n+1}}_{0:n}(n+1-k), \xi^{I^i_{n+1}}_{0:n}(n+2-k) \right) + \sum_{l=0}^{n-k} s\left( \xi^i_{0:l+k}(l), \xi^i_{0:l+k}(l+1) \right) \right\}}_{\text{cumulated contribution } \Gamma_{n+1}} .$$

We now need to store $\omega^i_n, \xi^i_{0:n}(n-k+1:n)$ and $\Gamma_n$ for $i = 1, \ldots, N$.

# These Ideas May be Used for Parameter Estimation...




400 iterations of the MCEM algorithm employing SISR approximation of the joint smoothing distributions. The number of particles was 250 for the first 100 EM iterations, 500 for iterations 101 to 200, and then increased proportionally to the squared iteration number.




Same as the above except that the parameter estimates were computed using an MCEM algorithm employing SISR approximation of fixed-lag smoothing distributions with delay $k = 20$.

# Roadmap

1. What is a Hidden Markov Model?
2. Filtering and Smoothing Recursions
3. Monte Carlo, Importance Sampling and Sampling Importance Resampling
4. Sequential Importance Sampling
5. Sequential Importance Sampling with Resampling
6. More Sequential Monte Carlo Algorithms*
7. Approximation of Sums Functionals and Parameter Estimation*

That's all; thank you for your attention!