Biometrika (2009), **xx**, x, *pp*. 1–17

Printed in Great Britain

Particle Learning in Nonlinear Models using Slice Variables

BY MICHAEL JOHANNES AND NICHOLAS G. POLSON AND SEUNG M. YAE

SUMMARY

This paper develops particle-based methods for sequential parameter learning and state filtering in nonlinear models. Sequential inference is notoriously difficult in nonlinear state space models. To overcome this, we use auxiliary slice variables to induce fixed-dimension conditional sufficient statistics and, given these, we adapt existing particle learning algorithms to update posterior beliefs about states and parameters. We provide three illustrations. First, a dynamic exponential model with Gaussian errors. Second, a stochastic growth model with nonlinear state evolution and t-distributed errors. Finally, a bivariate radar tracking problem which was originally analyzed in the nonlinear Monte Carlo filtering literature. In all cases, we illustrate the efficiency of our methodology.

Some key words: Nonlinear model; State Space Particle filtering; Sequential parameter learning; Slice variable.

1. INTRODUCTION

This paper develops particle methods for sequential parameter learning and state filtering in nonlinear state space models. Nonlinearity creates difficulties for common particle filtering algorithms (Storvik (2002), Fearnhead (2002), and Johannes and Polson (2007)) that require parameter posteriors to admit low-dimensional sufficient statistics, conditional on the observed data and latent states. Due to these problems, until now, the only alternative in nonlinear models is sequential importance sampling that, in the case of parameter learning, degenerates quickly as time progresses.

The goal of this paper is to demonstrate how to use slice variables to induce conditional sufficient statistics in nonlinear models. The slice variables induce intervals whose endpoints become the sufficient statistics for the parameter posteriors. Given this, we adapt sufficient-statistic based particle filtering algorithms to sequentially update the posterior distribution. Slice variables have been widely used to sample from non-standard distributions with MCMC, see Besag and Green (1993), Polson (1996), Damien, Wakefield and Walker (1999), and Neal (2003).

To demonstrate our approach, we consider three applications with increasing degrees of nonlinearities. The first model has mild nonlinearities with an exponential term in the observation equation, Gaussian errors, and a linear state evolution. The second model is a non-stationary growth model with nonlinear state evolution and t-errors, originally considered in a smoothing context by Kitagawa (1987) and Carlin, Polson and Stoffer (1992). The third model is a bivariate radar tracking problem from the original nonlinear Monte Carlo filtering literature, see Akashi and Kumamoto (1977). This model has a number of additional complications, including multiple latent states, switching errors, and deterministic and highly nonlinear state-dynamics.

For each application, we show that our algorithm is able to accurately track the parameters sequentially through time. Moreover, we show that our algorithms significantly improve the efficiency of existing particle filtering algorithms in nonlinear models. Thus, we extend the applicability of particle methods to a wide range of nonlinear models.

The rest of the paper is outlined as follows. Section 2 describes our algorithm and shows how auxiliary slice state variables can induce conditionally sufficient statistics for sequential parameter posterior learning. Section 3 describes results for our three empirical applications. Finally, Section 4 concludes.

2. Nonlinear Filtering and Parameter Learning

Consider a general nonlinear state space model where the system evolves from initial distribution $p(x_0, \theta)$ via

$$y_{t+1} = f(\theta, x_{t+1}, \eta_{t+1})$$
$$x_{t+1} = g\left(\theta, x_t, \eta^x_{t+1}\right),$$

where y_t is the observed data, x_t is the latent state, f and g are known analytical functions, η_{t+1} and η^x_{t+1} are errors, and θ is a vector of fixed static parameters. We allow for and later consider an example with deterministic state evolutions that occur when $\eta^x_{t+1} = 0$. Typically η_{t+1}, η^x_{t+1} are discrete or scale mixture of normal distributions. The observation equation induces the conditional density $p(y_{t+1}|x_{t+1}, \theta)$ with $p(x_{t+1}|x_t, \theta)$ defined similarly for the states.

A number of particle-based methods have been recently developed for sequential parameter inference in state space models, see for example, Carpenter et al. (1999), Chopin (2002, 2004), Storvik (2002), Fearnhead (2002), Andrieu, Doucet and Tadic (2003, 2005) and Johannes and Polson (2007). Particle methods provide an approximate sample from the sequence of joint distributions $p\left(\theta, x_t|y^t\right)$, where $y^t = (y_1, \ldots, y_t)$, by discretizing the support of the distribution into a finite set of $i = 1, ..., N$ particles $(\theta, x_t)^{(i)}$ with appropriate weights.

Particle methods have been widely used for state filtering in nonlinear models, assuming parameters are known. Doucet et al. (2001) provides a review. This was the original goal of Gordon et al. (1993), as well as earlier work on nonlinear Monte Carlo filtering in Handschin and Mayne (1969), Handschin (1970), Sorensen and Alspach (1971), Ackerson and Fu (1970). Given parameters, there are many refinements that improve the efficiency of the algorithms. These include Kitagawa (1996), Beadle and Djuric (1997), Pitt and Shephard (1999) and Chen and Liu (2000), as well as recent work on state smoothing, see Godsill et al. (2004) and Guo et al. (2005). This literature, however, has not addressed the problem of sequential parameter learning in nonlinear models, as parameters are assumed known. A related approach is Gilks and Berzuini (2001), who provide a hybrid particle filtering and MCMC algorithm for sequential learning. Chopin (2002) analyzes sequential parameter inference in nonlinear parametric models, without an explicit focus on state space models.

If a state space model admits fixed-dimension sufficient statistics for parameters, given latent state variables, Fearnhead (2002), Storvik (2002), Johannes and Polson (2007), and Carvalho, Johannes, Polson, and Lopes (2008) use particle based methods to sequentially learn parameters and states. These algorithms utilize the fact that $p(\theta|x^t, y^t) = p(\theta|s_t)$ and the fact that the sufficient statistics, s_t, can be recursively updated via a mapping $s_{t+1} = \mathcal{S}(s_t, x_{t+1}, y_{t+1})$. For nonlinear models, conditional sufficient statistics do not naturally exist. The purpose here is to show how slice state variables can induce conditional sufficient statistics. To develop our methodology, we first consider the simpler case of introducing slice variables into a general Bayesian parameter updating problem, without latent states. Then, we add latent state variables and discuss specific algorithms.

2·1. *Bayesian Sequential Learning*

Consider the problem of generating samples from a posterior distribution, $p\left(\theta|y^t\right)$, as new data sequentially arrives, assuming that $p\left(\theta|y^t\right)$ does not admit traditional sufficient statistics. By Bayes rule, the posterior sequentially updates via

$$p\left(\theta|y^{t+1}\right) = \frac{p\left(y_{t+1}|\theta\right) p\left(\theta|y^t\right)}{p\left(y_{t+1}|y^t\right)}.$$

The goal is to efficiently sample update samples from $p\left(\theta|y^t\right)$, as new data arrives. One approach to sampling from this distribution to draw $\theta^{(i)} \sim p\left(\theta|y^t\right)$, compute weights that are proportional to $p\left(y_{t+1}|\theta^{(i)}\right)$, and then re-sample the $\theta^{(i)}$'s given those weights. This is the original sequential importance sampling/resampling (SIR) idea of Smith and Gelfand (1992).

This tends to be an inefficient way of sampling from $p\left(\theta|y^{t+1}\right)$. The main reason is that the particles are never replenished just re-sampled. Due to this, the support of the particles rapidly declines over time leading to sample impoverishment. Additionally, there is limited scope for sampling refinements such as the auxiliary particle filter that "look-ahead" prior to resampling, because of the lack of sufficient statistics. To get around these extreme deficiencies, it is common to add an MCMC step, as in Gilks and Berzuini (2001), to improve performance, see Chopin (2002).

As an alternative, we use auxiliary slice variables u_{t+1} to induce conditional sufficient s_t statistics for θ, translating the parameter learning problem to one of filtering conditional sufficient statistics. The slice variables induce a slice interval, and the endpoints of these intervals are the conditional sufficient statistics. These sufficient statistics need to satisfy a recursive updating scheme $s_{t+1} = \mathcal{S}\left(s_t, u_{t+1}, y_{t+1}\right)$, so the methods developed in Storvik (2002), Fearnhead (2002), and Johannes and Polson (2007) apply. Given these sufficient statistics, the parameters can easily be updated or replenished.

The slice variable is defined as follows. Consider the joint posterior distribution,

$$p\left(\theta, u_{t+1}|y^t\right) = \frac{\mathbb{I}\left[0 < u_{t+1} < p\left(y_{t+1}|\theta\right)\right] p\left(\theta|y^t\right)}{p\left(y_{t+1}|y^t\right)}.$$

For this to be a valid, slice variables must satisfy a consistency condition, which states that the marginal from this joint distribution is the target distribution,

$$p(\theta|y^t) = \int_0^{p(y_{t+1}|\theta)} p\left(\theta, u_{t+1}|y^t\right) du_{t+1},$$

a basic consistency condition. Note that by definition more general slice regions such as

$$\mathbb{I}\left[c_p p\left(y_{t+1}|\theta\right) < u_{t+1} < C_p p\left(y_{t+1}|\theta\right)\right]$$

are admissible for any constants $0 < c_p < C_p$.

Given this slice region, we now consider sequential inference. Defining a vector of slice state variables $u^t = (u_1, \ldots, u_t)$, consider the conditional posterior distribution

$$\begin{aligned} p\left(\theta, u_{t+1}|u^t, y^{t+1}\right) &\propto p\left(u_{t+1}|y_{t+1}, \theta\right) p\left(\theta|u^t, y^t\right) \\ &\propto \mathbb{I}\left[u_{t+1} < p\left(y_{t+1}|\theta\right)\right] p\left(\theta|u^t, y^t\right) \\ &\propto \mathbb{I}\left[a\left(u_{t+1}, y_{t+1}\right) \leq \theta \leq A\left(u_{t+1}, y_{t+1}\right)\right] p\left(\theta|u^t, y^t\right), \end{aligned}$$

where we have assumed that that the slice region $u_{t+1} < p(y_{t+1}|\theta)$ can be characterized by simple endpoints a and A:

$$[u_{t+1} < p(y_{t+1}|\theta)] \Leftrightarrow [a(u_{t+1}, y_{t+1}) \leq \theta \leq A(u_{t+1}, y_{t+1})].$$

In the examples below, the endpoints can be computed analytically. However, the methodology is more general only requiring that the endpoints can be computed, possibly by numerical or other approximate means, see Neal (2003) for a thorough discussion of methods for computing slice regions.

The endpoints of the slice region are used to generate sufficient statistics that satisfy a recursive updating scheme. By induction, assume that the posterior at time t is of the form $p(\theta|u^t, y^t) \propto \mathbb{I}[a_t < \theta < A_t] p(\theta)$. Then the updated posterior is given by

$$p(\theta|u^{t+1}, y^{t+1}) \propto \mathbb{I}[a(u_{t+1}, y_{t+1}) < \theta < A(u_{t+1}, y_{t+1})] \mathbb{I}[a_t < \theta < A_t] p(\theta).$$

Combining the indicators,

$$\mathbb{I}[a(u_{t+1}, y_{t+1}) < \theta < A(u_{t+1}, y_{t+1})] \mathbb{I}[a_t < \theta < A_t] = \mathbb{I}[a_{t+1}, A_{t+1}]$$

where

$$\begin{aligned} a_{t+1} &= \max(a_t, a(u_{t+1}, y_{t+1})) \\ A_{t+1} &= \min(A_t, A(u_{t+1}, y_{t+1})). \end{aligned}$$

Defining $s_t = (a_t, A_t)$ generates fixed-dimension sufficient statistics and a recursive mapping $s_{t+1} = \mathcal{S}(a_t, A_t, u_{t+1}, y_{t+1})$, as required to update parameters.

Sampling from this distribution using particle methods is straightforward. At time t, the algorithm provides a cloud of particles, $(a_t, A_t)^{(i)}$ for $i = 1, ..., N$ from $p(a_t, A_t|y^t)$. The posterior distribution can then be approximated by the particle representation

$$p^N(\theta|y^t) = \frac{1}{N} \sum_{i=1}^{N} p\left(\theta|(a_t, A_t)^{(i)}\right)$$

and we can draw parameters, $\left\{\theta^{(i)}\right\}_{i=1}^{N}$ from $p\left(\theta|(a_t, A_t)^{(i)}\right)$. An "optimal" way to draw from the mixture approximation to $p(\theta|y^{t+1})$ would be to first computes weights $w^{(i)}$ proportional to $p\left(y_{t+1}|\theta^{(i)}\right)$ and re-sample the vector $(\theta, a_t, A_t)^{(i)}$ using those weights. Like the auxiliary particle filter of Pitt and Shephard (1999), this propagates only high-likelihood particles. Next, draw slice variables

$$u_{t+1}^{(i)} \sim p\left(u_{t+1}|\theta^{(i)}, y_{t+1}\right) \sim \mathcal{U}\left[0, p\left(y_{t+1}|\theta^{(i)}\right)\right].$$

Given $u_{t+1}^{(i)}$, compute the slice regions, $a\left(u_{t+1}^{(i)}, y_{t+1}\right)$ and $A\left(u_{t+1}^{(i)}, y_{t+1}\right)$, and the updated sufficient statistics, $a_{t+1}^{(i)}$ and $A_{t+1}^{(i)}$. Finally, update the parameters via $\theta^{(i)} \sim p\left(\theta|(a_{t+1}, A_{t+1})^{(i)}\right)$ to provide a particle approximation to the posterior $p(\theta|y^{t+1})$.

This provides an alternative to the SIR, "Bayes without tears," algorithm of Rubin (1988) and Smith and Gelfand (1992) that degenerates rapidly, as discussed above. Via slice variables, the new algorithm translates an intractable sampling problem into one that involves sufficient statistics and sampling from simple distributions. This is the main reason that slice samplers often generate large efficiency gains. At some level, this observation is identical to much of the

MCMC literature that introduces auxiliary state variables to translate an intractable sampling problem into a Gibbs sampling-type algorithm, as an alternative to Metropolis-Hastings.

This algorithm is more efficient than blind SIR due to the Rao-Blackwellization theorem. In this context, we estimate the parameter posterior as the Monte Carlo average

$$\begin{aligned} p^N\left(\theta|y^t\right) &= \frac{1}{N}\sum_{i=1}^N p\left(\theta|\left(a_t,A_t\right)^{(i)}\right) \\ &= \frac{1}{N}\sum_{i=1}^N \mathbb{I}\left[a_t^{(i)} \leq \theta \leq A_t^{(i)}\right] p(\theta) \\ &\rightarrow E\left[p\left(\theta|a_t,A_t\right)|y^t\right] = p\left(\theta|y^t\right), \end{aligned}$$

as N increases. This result is reminiscent of Feller's (1943) representation of a distribution as a mixture of uniforms.

2·2. *Bayesian sequential learning in nonlinear state space models*

Consider now the general problem of parameter and state learning in nonlinear state space models. The existing approaches of Storvik (2002), Fearnhead (2002), and Johannes and Polson (2007) assume the parameter posteriors admit fixed dimension sufficient statistics that recursively update. The fixed dimension sufficient statistics, s_t, are defined via a sequential version of Bayes rule for the conditional parameter posterior distribution:

$$p(\theta|s_{t+1}) = p\left(\theta|x_{t+1},x_t,s_t,y_{t+1}\right) \propto p(y_{t+1}|x_{t+1},\theta)p(x_{t+1}|x_t,\theta)p(\theta|s_t).$$

This functional relationship defines a recursive representation for the sufficient statistics via a mapping $\mathcal{S}$, where $s_{t+1} = \mathcal{S}\left(s_t,x_{t+1},y_{t+1}\right)$. If the likelihood and state evolution are linear and the shocks are Gaussian, it is easy to uncover the mapping.

However, when either the likelihood or state evolution is nonlinear, it is not generally possible to construct fixed dimensional conditional sufficient statistics, as the only sufficient statistics are the entire history $\left(x^t,y^t\right)$, whose dimension grows with time. This is why traditional particle filtering algorithms have difficulties learning in these models, as the dimensionality of the problem increases with time. As noted by a number of authors in the filtering setting, the Monte Carlo error present in sampling from $p\left(x_t|\theta,y^t\right)$ is much lower than when sampling from $p\left(x^t|\theta,y^t\right)$, see. e.g., Klaas et al. (2005), which is why it is easier to learn in models with sufficient statistics. We exploit this property in our approach.

Our approach works as follows. For notational simplicity, assume that the nonlinearities appear only in the likelihood, thus we will slice the nonlinearities from the likelihood. Then define the auxiliary variables by

$$p\left(\theta,u^t|x^t,y^t\right) \propto \prod_{s=1}^t \mathbb{I}\left[u_s \leq p\left(y_s|x_s,\theta\right)\right] \prod_{s=1}^t p\left(x_s|x_{s-1},\theta\right) p\left(\theta\right),$$

This insures the important consistency property

$$p\left(\theta|x^t,y^t\right) = \int p\left(\theta,u^t|x^t,y^t\right) du^t \propto \prod_{s=1}^t p\left(y_t|x_t,\theta\right) p\left(x_s|x_{s-1},\theta\right) p\left(\theta\right).$$

Next, we define the sufficient statistics. Assuming the parameter nonlinearities appear in the observation equation, the parameter posteriors for parameters in the state equation admit sufficient statistics, s_t^x, which are defined via

$$p\left(\theta|s_{t+1}^x\right) \propto p\left(x_{t+1}|x_t,\theta\right) p\left(\theta|s_t^x\right). \tag{1}$$

This is the standard definition and does not require auxiliary variables. The full vector of sufficient statistics, s_t, contains s_t^x, and the endpoints of the slice regions. We assume these endpoints

can be computed as before: $0 \leq u_{t+1} \leq p\left(y_{t+1}|x_{t+1},\theta\right)$ implies that

$$a_{t+1}^{*} = a\left(u_{t+1},y_{t+1},x_{t+1}\right) \leq \theta \leq A\left(u_{t+1},y_{t+1},x_{t+1}\right) = A_{t+1}^{*},$$

leading to a set of recursively defined conditional sufficient statistics (a_t, A_t). Given x_{t+1}, u_{t+1}, and y_{t+1}, Bayes rule implies that

$$\begin{aligned} p\left(\theta|u_{t+1},x_{t+1},x_t,s_t,y_{t+1}\right) &\propto p\left(y_{t+1},x_{t+1},u_{t+1}|x_t,\theta\right)p\left(\theta|s_t\right) \\ &\propto p\left(y_{t+1},u_{t+1}|x_{t+1},\theta\right)p\left(x_{t+1}|x_t,\theta\right)\mathbb{I}\left[a_t \leq \theta \leq A_t\right]p\left(\theta|s_t^x\right) \\ &\propto \mathbb{I}\left[a_{t+1}^{*} \leq \theta \leq A_{t+1}^{*}\right]\mathbb{I}\left[a_t \leq \theta \leq A_t\right]p\left(x_{t+1}|x_t,\theta\right)p\left(\theta|s_t^x\right). \end{aligned}$$

Combining indicators,

$$\mathbb{I}\left[a_{t+1}^{*} \leq \theta \leq A_{t+1}^{*}\right]\mathbb{I}\left[a_t \leq \theta \leq A_t\right] = \mathbb{I}\left[a_{t+1} \leq \theta \leq A_{t+1}\right]$$

where the endpoints are given by

$$a_{t+1} = \max\left(a_{t+1}^{*},a_t\right) \text{ and } A_{t+1} = \min\left(A_{t+1}^{*},A_t\right). \tag{2}$$

Thus, the posterior is

$$p\left(\theta|s_{t+1}\right) = \mathbb{I}_{[a_{t+1}\leq\theta\leq A_{t+1}]}p\left(\theta|s_{t+1}^x\right),$$

where $s_t = (a_t, A_t, s_t^x)$ are the sufficient statistics with recursion

$$s_{t+1} = \mathcal{S}\left(s_t,u_{t+1},x_{t+1},x_t,y_{t+1}\right)$$

defined via equation 2. We can now develop a general algorithm.

2·3. *Algorithms*

Our general algorithm builds on existing nonlinear state filtering algorithms by first propagating the states, re-sampling the triplet of states, parameters, and sufficient statistics, drawing auxiliary variables, updating sufficient statistics, and finally drawing parameters from their appropriate posterior distributions. As with any particle filtering algorithm, there are many potential variants based on the ordering of updating and resampling/propagating.

The general algorithm that we consider assumes that the nonlinearity is sliced from the likelihood. The algorithm is as follows: given particles $\left\{(\theta,x_t,s_t)^{(i)}\right\}_{i=1}^{N}$:

Algorithm: Particle Learning with slice variables

Step 1. Propagate state variables: draw $x_{t+1}^{(i)} \sim p(x_{t+1}|(x_t,\theta)^{(i)})$.

Step 2. Re-sample $(x_{t+1},s_t,\theta)^{(i)}$: for $i = 1,...,N$, draw $k(i) \sim Multi\left(w^{(i)}\right)$, where

$$w^{(i)} = \frac{p\left(y_{t+1}|(x_{t+1},\theta)^{(i)}\right)}{\sum_{j=1}^{N}p\left(y_{t+1}|(x_{t+1},\theta)^{(j)}\right)}$$

setting $(x_{t+1},s_t,\theta)^{(i)} = (x_{t+1},s_t,\theta)^{k(i)}$.

Step 3. Draw auxiliary variables:

$$u_{t+1}^{(i)} \sim p(u_{t+1}|\left(x_{t+1},\theta\right)^{(i)},y_{t+1}) \sim \mathcal{U}\left[0,p\left(y_{t+1}|\left(x_{t+1},\theta\right)^{(i)}\right)\right]$$

Step 4. Update conditional sufficient statistics, $s_{t+1}^{(i)}$:

$$s_{t+1}^{(i)} = \mathcal{S}\left(\left(s_t,x_{t+1},u_{t+1}\right)^{(i)},y_{t+1}\right)$$

Step 5. Update parameters: for $i = 1, ..., N$,

$$\theta^{(i)} \sim p(\theta|s_{t+1}^{(i)}).$$

There are a number of immediate comments.

- Although this algorithm focuses on slicing the likelihood, it is also possible to slice nonlinearities from the state equation. This requires no major changes, although the notation becomes quite complicated. In general, the approach can handle one slice variable per equation, with additional MCMC steps as in Gilks and Berzuini (2001). This is due to the fact that particle learning algorithms require that the parameters and states be updated sequentially. We consider examples below with sliced nonlinearities in the observation equation and another example with sliced nonlinearities in both equations.
- Given the recursive definition of a_t and A_t, it is clear that the slice intervals for each particle shrink when updated, see equation 2. However, since

$$p(\theta|y^t) = \int p\left(\theta|s_t\right) p\left(s_t|y^t\right) ds_t$$

and because the sufficient statistics are resampled, the slice regions need not shrink, although it is natural in most cases for them to shrink as more information arrives. Additionally, in many models, there is additional marginalization, either in states or parameters, that can be used to improve the performance, as discussed in Carvalho, Johannes, Lopes, and Polson (2008). This can be done when $p\left(y_{t+1}|x_t, \theta\right)$ is available (x_{t+1} can be marginalized out of the predictive likelihood) or when

$$p(x_{t+1}|x_t, s_t) = \int p(x_{t+1}|x_t, \theta)p(\theta|s_t)d\theta$$

is known. In the latter case, the parameters are integrated out of the state evolution.
- If the nonlinearities are in the state equation, it is possible to resample first, prior to propagation, provided the predictive likelihood, $p\left(y_{t+1}|x_t, \theta\right)$ is available.
- The main competitor is a modification of Storvik's (2002), which proceeds as follows. Given particles $\{(x_t, s_t, \theta)^{(i)}\}_{i=1}^N$, propagate states x_{t+1} with the evolution $p(x_{t+1}|(x_t, \theta)^{(i)})$, and then re-sample with weights proportional to $p\left(y_{t+1}|(x_{t+1}, \theta)^{(i)}\right)$. For sub-parameters that have a sufficient statistics, $s_{t+1} = S\left((s_t, x_{t+1}, \theta)^{(i)}, y_{t+1}\right)$ and then replenish particles with a draw from $\theta^{(i)} \sim p(\theta|s_{t+1}^{(i)})$. For the parameters that do not admit sufficient statistics, the particles are only re-sampled. This leads to severe degeneracies for the parameters that enter nonlinearly. In the particle filtering literature, the main tool for comparing algorithms is the effective sample size, see, e.g., Fearnhead (2002). This measures how many distinct particles are used at each time period in the particle approximation. The effective sample size is bounded above by 100%, which would coincide with an exact or direct sample from the particle approximation. It is common to use similar metrics such as "efficiency factors" in the MCMC literature. Ideally, we would like to compare the various particle filters to the "true" posterior, $p\left(\theta|y^t\right)$ and $p\left(x_t|y^t\right)$. However, these are not known in nonlinear models. Carvalho, Johannes, Lopes, and Polson (2008) show in dynamic linear models that particle filtering methods are as or more efficient, in terms of accuracy and computing time, as MCMC methods for computing smoothing distributions and parameter posteriors.
- The following convergence result holds.

THEOREM 1. *The particle-based estimate* $p^N\left(x_t,\theta|y^t\right)$ *is consistent for* $p\left(x_t,\theta|y^t\right)$ *namely*

$$\left\|p^N\left(x_t,\theta|y^t\right)-p\left(x_t,\theta|y^t\right)\right\|\rightarrow^{\mathcal{P}}0$$

with a $\sqrt{N}$ *convergence rate if*

$$\hat{C}_t=\int\frac{p^2(s_{t+1},s_t,x_{t+1}|y^{t+1})}{p(s_t,\theta|y^t)}d(s_{t+1},s_t,x_{t+1})<\infty.$$

Proof: See Appendix A. These results suggest the usual asymptotic of particle methods. They do not, however, provide a formal bound on the number of particle methods required for a given level of accuracy. Like MCMC methods, this implies that experimentation is needed to control Monte Carlo error. Unlike MCMC methods, there is no need to worry about Markov Chain convergence, as the draws are i.i.d. In practice, a common to evaluate the Monte Carlo error is to run the algorithm on a given dataset for multiple random seeds. If the results differ substantially, this implies that the Monte Carlo error is large. An open issue with these algorithms is how to increase N as the sample size increases. This is true not only of our nonlinear examples, but also more generally in the particle filtering literature.

We now turn to our empirical applications.

3. EMPIRICAL APPLICATIONS

In this section we consider three applications. We compare our slice particle filtering and learning algorithm with an extension of Storvik's algorithm adapted for nonlinear settings. We consider nonlinearity in the observation equation using an exponential growth model and nonlinearity in the state evolution using a non-stationary stochastic growth model. In the latter model, the state filtering distribution bifurcates, and we allow for heavy-tailed t-distributed errors. The third example is a bivariate radar tracking problem with nonlinearities in both the observation and evolution equations and a deterministic state evolution. This application was the original example in the Monte Carlo nonlinear filter literature, see Akashi and Kumamoto (1977).

3·1. *Exponential State Space Model*

Consider a state space model with an exponential term in the observation equation:

$$\begin{aligned}y_t&=\exp\left(-\gamma x_t\right)+\sigma\varepsilon_t^y\\x_t&=\alpha+\beta x_{t-1}+\sigma_x\varepsilon_t^x,\end{aligned}$$

where $\varepsilon_t^y,\varepsilon_t^x\sim\mathcal{N}(0,1)$. We assume a uniform prior for $\gamma\sim\mathcal{U}(a^\gamma,A^\gamma)$, $\sigma^2\sim\mathcal{IG}\left(b,B\right)$, and a conjugate normal/inverse Gamma prior for $\left(\alpha,\beta,\sigma_x^2\right)\sim\mathcal{N}\left(c,C\sigma_x^2\right)\mathcal{IG}\left(d,D\right)$. The state evolution and priors are standard.

To implement the algorithm given earlier, we need to simulating persistent states, re-sample the particles, update the auxiliary variables, update the sufficient statistics, and draw the parameters. The state simulation and re-sampling steps are straightforward, and the only detail requiring discussion is the definition of the auxiliary variables and sufficient statistics.

Defining the sufficient statistics for $\left(\alpha,\beta,\sigma_x^2\right)$ is straightforward: given a standard normal/inverse Gamma conjugate priors,

$$p\left(\alpha,\beta,\sigma_x^2|s_t^x\right)=\mathcal{N}\left(c_t,C_t\sigma_x^2\right)\mathcal{IG}\left(d_t,D_t\right),$$

where the recursive formulas for sufficient statistics (c_t, C_t, d_t, D_t) are straightforward to compute, see, for example, West and Harrison (1997) or Johannes and Polson (2007). The difficult part is the sufficient statistics for the parameters in the observation equation.

First, we can marginalize σ^2 in the likelihood. Given the inverse Gamma prior, the marginal likelihood is a t-distribution:

$$p\left(y_t|x_t, \gamma, b_{t-1}, B_{t-1}\right) \sim t_{2b_{t-1}}\left(\exp\left(-\gamma x_t\right), \sqrt{B_{t-1}/b_{t-1}}\right),$$

where the expressions for b_t and B_t are given below in the sufficient statistics for σ^2. The slice variable u_t is defined as

$$0 \leq u_t \leq p\left(y_t|x_t, \gamma, b_{t-1}, B_{t-1}\right) = \left(1 + \frac{\left(y_t - \exp\left(-\gamma x_t\right)\right)^2}{2B_{t-1}}\right)^{-(2b_{t-1}+1)/2},$$

with the normalisation constant absorbed into (c_p, C_p). Inverting the slice variable constraint defines a inequality region for the parameters:

$$[a_t^*, A_t^*] = \left[-\frac{1}{x_t}\ln\left(y_t + K\right), -\frac{1}{x_t}\ln\left(y_t - K\right)^+\right], \text{ where } K = \sqrt{2B_{t-1}\left(u_t^{-2(2b_{t-1}+1)^{-1}} - 2\right)}.$$

The sufficient statistics (a_t, A_t) are then defined recursively as $a_t = \max(a_{t-1}, a_t^*)$ and $A_t = \min(A_{t-1}, A_t^*)$.

Therefore, at every given time point t, the algorithm keeps track of the endpoints of the intervals as conditional sufficient statistics. The full parameter posteriors are

$$\begin{aligned}
p(\gamma|s_{t+1}) &\sim \mathcal{U}\left(a_{t+1}, A_{t+1}\right) \\
p(\sigma^2|s_{t+1}) &\sim \mathcal{IG}\left(b_{t+1}, B_{t+1}\right) \\
p\left(\sigma_x^2|s_{t+1}\right) &\sim \mathcal{IG}\left(d_{t+1}, D_{t+1}\right) \\
p(\alpha, \beta|\sigma_x^2, s_{t+1}) &\sim \mathcal{N}\left(c_{t+1}, \sigma_x^2 C_{t+1}\right)
\end{aligned}$$

for hyper-parameters defined recursively using the usual conjugate Bayesian theory.

To illustrate the algorithm, and its performance relative to Storvik's algorithm, we simulate artificial datasets for the following parameters: $\gamma = 1$, $\alpha = 0$, $\beta = 0.9$, $\sigma = 0.3$, and $\sigma_x = 0.2$. The priors parameters are given by $\gamma \sim \mathcal{U}(0, 4)$, $b = 10$, $B = 0.72$, $c = (0, 0.9)$, $C = diag\,(0.1, 0.2)$, and $d = 20$, $D = 0.76$. In state space models and especially in nonlinear models, there are additional identification issues that are present, so it is not possible to have extremely diffuse priors. Thus, the priors are mildly informative.

We simulated a time series of $T = 100$ observations from the model and implemented the particle filtering algorithm using $N = 100K$ particles. The particle size was chosen so that the Monte Carlo error was negligible. The results are summarized in Figures 1 and 2. The left hand panels of Figure 1 summarize the marginal posterior distribution at time $T = 100$ for each parameter (e.g., $p\left(\gamma|y^t\right)$) with the vertical bar indicating the true value, and the right hand panels provide a sequential summary of the posterior mean (black line), a 95% Bayesian probability region (solid grey lines), and the true values (straight thick solid grey line).

For each of the parameters, the posteriors are centered over the true values. Sequentially, the posterior gradually tighten around the true values, although there is a fair amount of variation across time. The algorithm learns some parameters, those in the observation equation, more rapidly than those in the state equation. The most difficult parameter to learn is σ_x, as this is the volatility of the latent process.

To compare our algorithm with Storvik's, Figure 2 reports, for time t, the number of distinct γ particles for the nonlinear parameters for the modification of Storvik's algorithm (solid line) and our algorithm (dashed line). As the figure indicates, the number of distinct γ particles in our algorithm is stable, whereas the effective sample size in Stovik's algorithm decreases rapidly, as expected. For time series of this size, the algorithm performs admirably.

3·2. *Univariate non-stationary growth model.*

Consider an extension of the non-stationary growth model that has been considered by Kitagawa (1987) and Carlin, Polson, and Stoffer (1992):

$$y_t = \frac{|x_t|^\alpha}{20} + \sigma\varepsilon_t^y$$
$$x_{t+1} = \beta_1 x_t + \frac{\beta_2 x_t}{1+x_t^2} + \beta_3\cos(1.2t) + \sigma_x\sqrt{\lambda_{t+1}}\varepsilon_{t+1}^x$$

where $\lambda_{t+1} \sim \mathcal{IG}(\nu/2, \nu/2)$ and ε_t^y and ε_t^x are independent standard normal. Extending previous models, we allow α to be an unknown parameter. Previous papers assumed $\alpha = 2$. By standard scale mixture theory, the marginal distribution of the state errors is $t-$distributed with ν degrees of freedom (which is assumed to be known). For prior distributions, we assume standard conjugate priors for the state parameters, $\beta_1, \beta_2, \beta_3, \sigma_x^2 \sim \mathcal{N}\left(b, \sigma_x^2 B\right)\mathcal{IG}(d, D)$, parameters $\sigma^2 \sim \mathcal{IG}(c, C)$, and a uniform prior for $\alpha \sim \mathcal{U}(a^\alpha, A^\alpha)$. In addition to the nonlinear term in the observation equation, this model adds two layers of complications from the previous example: t-distributed state errors and a nonlinear state equation that bifurcates.

The state evolution can be written as

$$x_{t+1} = Z_t\beta_x + \sigma_x\sqrt{\lambda_{t+1}}\varepsilon_{t+1}^x$$

where

$$Z_t = \left(x_t,\ \frac{x_t}{1+x_t^2},\ cos(1.2t)\right) \quad \text{and} \quad \beta_x = (\beta_1, \beta_2, \beta_3)'.$$

Conditional on λ_{t+1} and x_t, it is straightforward to derive the conditional sufficient statistic recursion for the state evolution parameters. The parameter α appears nonlinearly in the observation equation, and our algorithm will "slice" $|x_t|^\alpha$ from the observation equation, generating sufficient statistics for updating α.

Our algorithm requires simulating states, in this case λ_{t+1} and x_{t+1}, resampling the particles, drawing auxiliary variables, computing sufficient statistics, and drawing parameters. We simulate the states forward from a $t-$distribution and compute the resampling weights from $p(y_{t+1}|x_{t+1}, \alpha)$, again marginalizing out σ^2. By marginalizing out σ^2, the weights are flatter and are less likely to become unbalanced. We also draw λ_{t+1} from the appropriate inverse Gamma distribution. For the slice step, we again use the fact that $p(y_{t+1}|x_{t+1}, \alpha)$ is $t-$distributed. The region for nonlinear parameter α is defined using

$$[a_t^*, A_t^*] = \left[\frac{1}{\ln|x_t|}\ln 20\,(y_t + K), \frac{1}{\ln|x_t|}\ln 20\,(y_t - K)^+\right] \text{ and } K = \sqrt{2B_t\left(u_t^{-\frac{2}{2b_t+1}} - 1\right)}$$

To demonstrate the algorithm, we simulated data assuming that $\alpha = 2, \nu = 3$, $\beta_x = (0.5, 25, 8)$, and $\sigma_x^2 = 1$. For the priors, we assume $\alpha \sim \mathcal{U}(1, 4)$, $b = (0.5, 25, 8)$, $B = diag\,(0.025, 40, 2)$, $c = 5$, $C = 40$, $d = 10$, and $D = 9$. We provide two sets of results for this model.

The first set of results is a comparison of the effective sample sizes for our algorithm and the modification of Storvik's algorithm. Here, we simulate 50 samples of length 200 and use $N = 500$ particles. For each algorithm and path, we compute the effective sample size for a number of different signal to noise ratios, indexed by σ_x. The results indicate that our algorithm increases the effective sample size relative to Storvik's algorithm, although the improvement is greater when the evolution variance σ_x is small. In this calculation, the effective sample size is for the entire vector of parameters and states, instead of just a single parameter (as in Figure 2). Storvik's algorithm also utilizes sufficient statistics, which is why the overall effective sample is not even smaller.

σ_x^2	1	3	10
Slice approach	**74** (2.9)	**70** (3.6)	61 (4.7)
Blind	61(3.7)	61 (4.3)	61 (4.7)

Table 1. *Effective particle size of the filtering with sequential parameter learning. All numbers are expressed in the percentage of the physical particle size. Numbers in parenthesis are standard errors for the mean of effective particle size from* 50 *simulated time-series of length* 200 *from a univariate non-stationary growth model. Physical particle size is 500.* $\alpha = 2$*,* $\beta_1 = 0.5$*,* $\beta_2 = 25$*,* $\beta_3 = 8$ *and* $\sigma^2 = 10$*.*

Figure 3 displays a simulated sample path of the observations in the top panel and the true x_t's (thick grey line), posterior mean of x_t (solid black line), and the 95% Bayesian probability interval for x_t (thin grey lines) in the bottom panel. Notice the drastic changes in the x_t, as it bifurcates. Figure 4 summarizes the posterior distributions sequentially and at the end of the sample, as in Figure 1. Again, all of the posteriors are centered over the true values. Sequentially, the algorithm quickly learns α. Since small variation in α lead to drastic changes in the predictive distribution, the algorithm learns this parameter rapidly, despite the need to use slice variables. The algorithm learns σ quickly, as well as the regression coefficients in the state evolution, β_1, β_2, and β_3. The algorithm learns about σ_x more slowly. The second panel of Figure 2 displays the number of unique α particles, which is stable for our algorithm and steadily decreases for the modification of Stovik's algorithm.

3·3. *Bivariate Radar Tracking*

Consider a bivariate model for radar tracking of a vertically falling body. The distance, y_{t+1}, from the body to the radar is observed, and the true location of the body evolves according to the state evolution. The system is defined by

$$\begin{aligned} y_{t+1} &= \sqrt{x_{1,t+1}^2 + \alpha^2} + \sigma_{z_{t+1}} \varepsilon_{t+1} \\ x_{1,t+1} &= x_{1,t} - x_{2,t} \\ x_{2,t+1} &= x_{2,t} - \beta \exp\left(-\gamma x_{1,t}\right) x_{2,t}^2 + g + \sigma_x \varepsilon_{t+1}^x \end{aligned}$$

where $x_{1,t+1}, x_{2,t+1}$ and g are altitude (*m*), velocity (*m/sec downward*), and acceleration of gravity (*m/sec*^{2}) respectively. The error term in the observation takes one of two values, σ_0 and σ_1, with equal probability. The priors for the parameters are $\alpha \sim U\left(a^\alpha, A^\alpha\right)$, $\gamma \sim U\left(b, B\right)$, $\left(\beta, \sigma_x^2\right) \sim \mathcal{N}\left(c, \sigma_x^2 C\right) \mathcal{IG}\left(d, D\right)$, $\sigma_0^2 \sim \mathcal{IG}\left(e, E\right)$, and $\sigma_1^2 \sim \mathcal{IG}\left(f, F\right)$.

This model introduces a number of additional complications. The observation equation is highly nonlinear in both parameters states and has Markov switching errors. The model has two latent state variables. The dynamics for the first state variable are deterministic, and the dynamics for the second are nonlinear in both state variables.

The algorithmic details are as follows. First, due to the deterministic state evolution, we can substitute for $x_{1,t+1}$ in the observation equation to find the predictive distribution

$$p(y_{t+1}|x_{1,t}, z_{t+1}, \alpha) \sim \mathcal{N}\left(\sqrt{(x_{1,t} - x_{2,t})^2 + \alpha^2}, \sigma^2_{z_{t+1}}\right).$$

We can also marginalize out the state variable z_{t+1} and determine the predictive distribution:

$$p(y_{t+1}|x_{1,t}, x_{2,t}, \theta) \sim p\mathcal{N}\left(\sqrt{(x_{1,t} - x_{2,t})^2 + \alpha^2}, \sigma^2_0\right) + (1-p)\mathcal{N}\left(\sqrt{(x_{1,t} - x_{2,t})^2 + \alpha^2}, \sigma^2_1\right),$$

where p is the switching probability. Given the predictive, we first re-sample the states, parameters, and sufficient statistics using the predictive likelihood. Using the resampled parameters and states, we propagate the state particles using the deterministic evolution.

Next, the switching state can be updated using the "optimal" importance function taking into account y_{t+1}:

$$p(z_{t+1}|x_{1,t}, x_{2,t}, \alpha, y_{t+1}) \sim \mathcal{B}er\left(p_{t+1}\right) \quad \text{where} \quad p_{t+1} = \frac{p(y_{t+1}|x_{1,t}, z_{t+1} = 1, \alpha)}{p(y_{t+1}|x_{1,t}, x_{2,t}, \theta)} \frac{1-p}{p}.$$

Conditional on the Markov state, y_{t+1}, $x_{1,t}$, and $x_{2,t}$, it is possible to update the posteriors for σ_0 and σ_1, using standard conjugate updating. For example,

$$p(\sigma^2_1|y_{t+1}, z_{t+1} = 1) \sim \mathcal{IG}\left(e_{t+1}, E_{t+1}\right) \quad \text{and} \quad p(\sigma^2_0|y_{t+1}, z_{t+1} = 0) \sim \mathcal{IG}\left(f_{t+1}, F_{t+1}\right).$$

For the other parameters, we have to introduce slice regions, one in the observation equation and one in the second state evolution. For α, conditional on the Markov state, the slice region is defined as

$$0 \leq u^{\alpha}_{t+1} \leq \exp\left(-\frac{\left(y_{t+1} - \sqrt{x^2_{1,t+1} + \alpha^2}\right)^2}{2\sigma^2_{z_{t+1}}}\right),$$

where again the constants in the likelihood do not matter. This inequality is used to update u^{α}_{t+1} particles. Inverting this inequality, conditional on u_{t+1}, and following the updating rules in equation 2, generates the conditional posterior for α. For γ, we note that

$$p\left(x_{2,t+1}|x_{2,t}, x_{1,t}, \beta, \gamma, \sigma^2_x\right) \propto \exp\left(-\frac{\left(x_{2,t+1} - x_{2,t} + \beta \exp\left(-\gamma x_{1,t}\right) x^2_{2,t} - g\right)^2}{2\sigma^2_x}\right).$$

To define the slice variable, we integrate β and σ^2_x from $p\left(x_{2,t+1}|x_{2,t}, x_{1,t}, \beta, \gamma, \sigma^2_x\right)$ generating a t-distribution for the slice region for

$$0 \leq u^{\gamma}_{t+1} \leq p\left(x_{2,t+1}|x_{2,t}, x_{1,t}, \gamma, c_t, C_t, d_t, D_t\right) \propto \frac{1}{\sigma^{x_{2,t+1}}}\left(1 + \frac{(x_{2,t} - x_{2,t+1} + g - \mu^{x_{2,t+1}})^2}{2d_t(\sigma^{x_{2,t+1}})^2}\right)^{-(d_t + \frac{1}{2})},$$

where $\mu^{x_{2,t+1}} = Gc_t$, and

$$\sigma^{x_{2,t+1}} = \sqrt{\frac{D_t}{d_t}(1 + G^2 C_t)}, \text{ and } G = x^2_{2,t} \exp(-\gamma x_{1,t}).$$

The hyper-parameters c_t, C_t, etc. are the resampled sufficient statistics for β and σ^2_x. Given the slice variables, the rest of the parameter posteriors follow as in the previous examples.

We simulated the model with the same parameters for 10 seconds corresponding to a discretized total of 100 data points. The original approximate Monte Carlo nonlinear filter in Akashi

and Kumamoto (1997) considered only state filtering for the first 10 data-points. For the true parameters, we follow Akashi and Kumamoto (1977) and assume

$$\alpha = 30,000, \beta = 3.3 \times 10^{-3}, \gamma = 1.64 \times 10^{-4}.$$

We assume the following prior parameters: $a^\alpha = 2.4$, $A^\alpha = 3.6$, $b = 0.82 \times 10^{-4}$, $B = 3.28 \times 10^{-4}$, $c = 3.6 \times 10^{-3}$, $C = 10^{-8}$, $d = 5$, $D = 1600$, $e = 5$, $E = 3240000$, $f = 5$, $F = 3600$.

The top panel of Figure 5 displays the observed path, the middle panel summarizes inference on $x_{1,t}$, and the bottom panel summarizes inference on $x_{2,t}$. The algorithm does a good job of tracking the state variables. In particular, note that the posterior bands on $x_{2,t}$ shrink substantially over time. This is due to the decreased parameter uncertainty as the object falls over time. Figure 6 summarizes the parameters, using the same format as in the previous example. Again, the algorithm is able to track all of the parameters, but the posterior tail probabilities are more difficult to estimate, especially for γ. Finally, Figure 2 displays the unique particles for α and γ, with similar results as in the previous examples.

4. Conclusion

This paper develops a particle-based simulation method for nonlinear filtering and sequential parameter learning using slice variables. The slice variables induce conditional sufficient statistics for the nonlinear parameters, allowing the use sufficient statistic-based particle learning algorithms. In a number of examples, we document substantial efficiency gains to using slice variables when compared to existing methods.

These results suggest that slice variables can provide substantial efficiency gains. This, however, leaves open a number of questions that we intend to pursue in future research. First, the other main competitor for estimating these models is repeated application of MCMC methods for each time period. MCMC methods have difficulties in these models due to the requirement of single state updating for the state variables. It would be useful to compare these methods in terms of accuracy and computational time. Results in Carvalho, Johannes, Lopes, and Polson (2008) indicate that particle methods outperform MCMC methods in dynamic linear models. Second, it is important to study how many particles are required for accurate inference as the sample size increases. Theory provides a rough guide, but it would be useful to provide a detailed study of this tradeoff in a number of applications. We leave these for future applications.

References

Ackerson, G.A. and Fu, K.S. (1970). On State Estimation in Switching Environments. *IEEE Trans. Aut. Control*, AC-15, 10-17.

Akashi, H. and Kumamoto, H. (1977). Random Sampling Approach to State Estimation in Switching Environments. *Automatica*, 13, 429-434.

Andrieu, C., A. Doucet and Tadic, V. (2003). Online expectation-maximization type algorithms for parameter estimation in general state space models. *Proc. IEEE ICASSP*

Andrieu, C., A. Doucet and Tadic, V. (2005). Online simulation-based methods for parameter estimation in nonlinear non Gaussian state space models. *Proc. IEEE CDC*

Beadle, E.R. and Djuric, P.M. (1997). A fast-weighted Bayesian Bootstrap Filter for Nonlinear Model State Estimation. *IEEE Transactions on Aerospace and Electronic Systems, 33*, 338-363.

Besag, J. and Green, P. (1993) Spatial Statistics and Bayesian Computation. *Journal of Royal Statistical Society, B*, 55, 25-37.

Carlin, Brad, Polson, Nicholas and Stoffer, David (1992). A Monte Carlo approach to nonnormal and nonlinear state-space modeling. *Journal of American Statistical Assoc*, 87, 493-500.

Carpenter, James, Peter Clifford., and Paul Fearnhead (1999). An Improved Particle Filter for Nonlinear Problems. *IEE Proceedings – Radar, Sonar and Navigation*, 146, 2-7.

Carvalho, C., M. Johannes, H.F. Lopes and N.G. Polson (2008). Particle Learning and Smoothing. *Working Paper*.

Chen, R. and Liu, J. (2000). Mixture Kalman Filters. *Journal of Royal Statistical Society, B*, 62, 493-508.

Chopin, Nicolas (2002). A Sequential Particle Filter method for static models. *Biometrika*, 89, 539-552.

Chopin, Nicolas (2004). Central Limit Theorems for Sequential Monte Carlo methods. *Annals of Statistics*, 32, 2385-2411.

Damien, P., Wakefield, J. and Walker, S. (1999). Gibbs sampling for Bayesian nonconjugate and hierarchical models using auxiliary variables. *Journal of Royal Statistical Society, B,* 61, 331-344.

Doucet, Arnaud, Nando de Freitas, and Neil Gordon (2001). *Sequential Monte Carlo Methods in Practice*, New York: Springer-Verlag, Series Statistics for Engineering and Information Science.

Fearnhead, Paul (2002). MCMC, sufficient statistics and particle filter. *Journal of Computational and Graphical Statistics*, 11, 848-862.

Feller, W. (1943). On a general class of contagious distribution. *Annals of Mathematical Statistics*, 14, 389-400.

Gilks, W. and C. Berzuini (2001). Following a moving target: Monte Carlo inference for dynamic Bayesian models. *Journal of Royal Statistical Society, B*, 63, 127-146.

Godsill, S.J., Doucet, A., and West, M. (2004). Monte Carlo Smoothing for Nonlinear Time Series. *Journal of the American Statistical Association*, 99, 156-168.

Gordon, Neil, D. Salmond and Adrian Smith (1993). Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proceedings*, F-140, 107-113.

Guo, P., X. Wang and R. Chen (2005). New SMC methods for nonlinear dynamic systems. *Statistics and Computing*, 15, 135-147.

Handschin, J.E. (1970). Monte Carlo Techniques for Prediction and Filtering in Nonlinear Stochastic Processes. *Automatica*, 6, 555-563.

Handschin, J.E. and Mayne, D.Q. (1969). Monte Carlo Techniques to Estimate the Conditional Expectation in multi-stage Nonlinear Filtering. *Int. J. Control*, 9, 547-559.

Johannes, M. and Polson, N.G. (2007). Particle Filtering and Parameter Learning. *Working Paper*, University of Chicago.

Kitagawa, G. (1987). Non-Gaussian State Space Modeling of Nonstationary Time Series (with discussion). *Journal of the American Statistical Association*, 82, 1032-1063.

Kitagawa, G. (1996). Monte Carlo Filter and Smoother for Non-Gaussian Nonlinear State Space Models. *Journal of Computational and Graphical Statistics*, 5, 1, 1-25.

Klaas, M., N. de Freitas, and A. Doucet (2005). Toward practical N^2 Monte Carlo: The Marginal Particle Filter. Proceedings of UAI.

Neal, R. (2003). Slice sampling. *Annals of Statistics,* 31, 705-767.

Pitt, M. and N. Shephard (1999). Filtering via Simulation: Auxilary Particle Filters. *Journal of the American Statistical Association*, 94, 590-599.

Polson, N.G. (1996). Convergence of MCMC Algorithms. In: *Bayesian Statistics 5* (Bernardo et al eds), 297-321.

Rubin, D., (1988), Using the SIR algorithm to simulate Posterior distributions. In *Bayesian Statistics 3* (eds Bernardo et al), 395-402. Oxford University Press.

Smith, Adrian, and Alan Gelfand, (1992). Bayesian statistics without tears: a sampling-resampling perspective, *American Statistician* 46, 84-88.

Sorensen, H.W. and Alspach, D.L. (1971). Recursive Bayesian Estimation using Gaussian Sums. *Automatica*, 7, 465-479.

Storvik, G, (2002), Particle filters in state space models with the presence of unknown static parameters, *IEEE Transactions on Signal Processing,* 50, 281-289.

West, M. and J. Harrison, (1997). *Bayesian Forecasting and Dynamic Models*. New York, Springer-Verlag.

Appendix A: We need to show that our particle approximation $p^N(x_t, \theta|y^t)$ converges to the posterior $p(x_t, \theta|y^t)$. First, write $p(x_t, \theta|y^t) = \int p(x_t, \theta|s_t)p(s_t|y^t)ds_t$ as a mixture. Then, we show that the marginal sufficient statistic posterior converges, namely $p^N(s_t|y^t) \to p(s_t|y^t)$, as follows

$$p^N\left(s_{t+1}|y^{t+1}\right) = \frac{1}{N}\sum_{i=1}^{N} w\left((x_{t+1}, s_t)^{(i)}\right) p(s_{t+1}|(s_t, x_{t+1})^{(i)}, y_{t+1})$$

where $w\left((x_{t+1}, s_t)^{(i)}\right) = p(y_{t+1}|(x_{t+1}, s_t)^{(i)}) / \sum_{i=1}^{N} p(y_{t+1}|(x_{t+1}, s_t)^{(i)})$. The distribution $p(y_{t+1}|x_{t+1}, s_t) = p(y_{t+1}|x_{t+1}, \theta)p(\theta|s_t)d\theta$ and $p(s_{t+1}|s_t, x_{t+1}, y_{t+1})$ is defined by the mapping $s_{t+1} = \mathcal{S}\left(s_t, x_{t+1}, y_{t+1}\right)$.

Dividing numerator and denominator by $p(y_{t+1}|y^t)$ we can analyze,

$$p^N\left(s_{t+1}|y^{t+1}\right) = \frac{\frac{1}{N}\sum_{i=1}^{N} p\left(s_{t+1}|\left(s_t, x_{t+1}\right)^{(i)}, y_{t+1}\right)\frac{p(y_{t+1}|(x_{t+1}, s_t)^{(i)})}{p(y_{t+1}|y^t)}}{\frac{1}{N}\sum_{i=1}^{N}\frac{p(y_{t+1}|(x_{t+1}, s_t)^{(i)})}{p(y_{t+1}|y^t)}}$$

Taking expectations gives a denominator that converges to

$$E\left(\frac{1}{N}\sum_{i=1}^{N}\frac{p(y_{t+1}|(x_{t+1}, \theta)^{(i)})}{p(y_{t+1}|y^t)}\right) = 1$$

and a numerator that converges to

$$\begin{aligned}
&E\left(\frac{1}{N}\sum_{i=1}^{N} p(s_{t+1}|(s_t, x_{t+1})^{(i)}, y_{t+1})\frac{p(y_{t+1}|(x_{t+1}, s_t)^{(i)})}{p(y_{t+1}|y^t)}\right) \\
&= \int p(s_{t+1}|s_t, x_{t+1}, y_{t+1})\frac{p(y_{t+1}|x_{t+1}, s_t)p(s_t, x_{t+1}|y^t)}{p(y_{t+1}|y^t)}d(s_t, x_{t+1}) \\
&= \int p(s_{t+1}|s_t, x_{t+1}, y_{t+1})p(s_t, x_{t+1}|y^{t+1})d(s_t, x_{t+1}) \\
&= p(s_{t+1}|y^{t+1})
\end{aligned}$$

Combining, we have $p^N(s_{t+1}|y^{t+1}) \to p(s_{t+1}|y^{t+1})$, hence $\left\|p^N(s_t|y^t) - p(s_t|y^t)\right\| \to 0$, where $\|\cdot\|$ is the L^1 norm.

For convergence of the joint posterior $p^N(x_t, \theta|y^t)$ we use the mixture representation $p^N(x_t, \theta|y^t) = \int p(x_t, \theta|s_t)p^N(s_t|y^t)ds_t$ to obtain

$$\begin{aligned}
\left\|p^N(x_t, \theta|y^t) - p(x_t, \theta|y^t)\right\| &= \left|\int p(x_t, \theta|s_t)(p^N(s_t|y^t) - p(s_t|y^t))d\theta dx_t ds_t\right| \\
&\leq \int p(x_t, \theta|s_t)\left|p^N(s_t|y^t) - p(s_t|y^t)\right| d\theta dx_t ds_t \\
&= \int \left|p^N(s_t|y^t) - p(s_t|y^t)\right| ds_t \\
&= \left\|p^N(s_t|y^t) - p(s_t|y^t)\right\| \to 0.
\end{aligned}$$

Thus, we conclude that $\left\|p^N(x_t, \theta|y^t) - p(x_t, \theta|y^t)\right\| \to 0$.

For the variance we can use the inequality $var(\phi) \leq E(\phi^2)$ to deduce that

$$Var\left(p^N\left(s_{t+1}|y^{t+1}\right)\right) \leq \frac{1}{N}\int\left(p(s_{t+1}|s_t,x_{t+1},y_{t+1})\frac{p(y_{t+1}|x_{t+1},s_t)}{p(y_{t+1}|y^t)}\right)^2 p(s_t,x_{t+1}|y^t)d(x_{t+1},s_t)$$
$$= \frac{1}{N}\int\frac{p^2(s_{t+1},s_t,x_{t+1}|y^{t+1})}{p(s_t,x_{t+1}|y^t)}d(s_{t+1},s_t,x_{t+1}) = \frac{\widehat{C}_{t+1}}{N}$$

where we have used Bayes rule $p(x_{t+1},s_t|y^{t+1})/p(x_{t+1},s_t|y^t) = p(y_{t+1}|x_{t+1},s_t)/p(y_{t+1}|y^t)$.

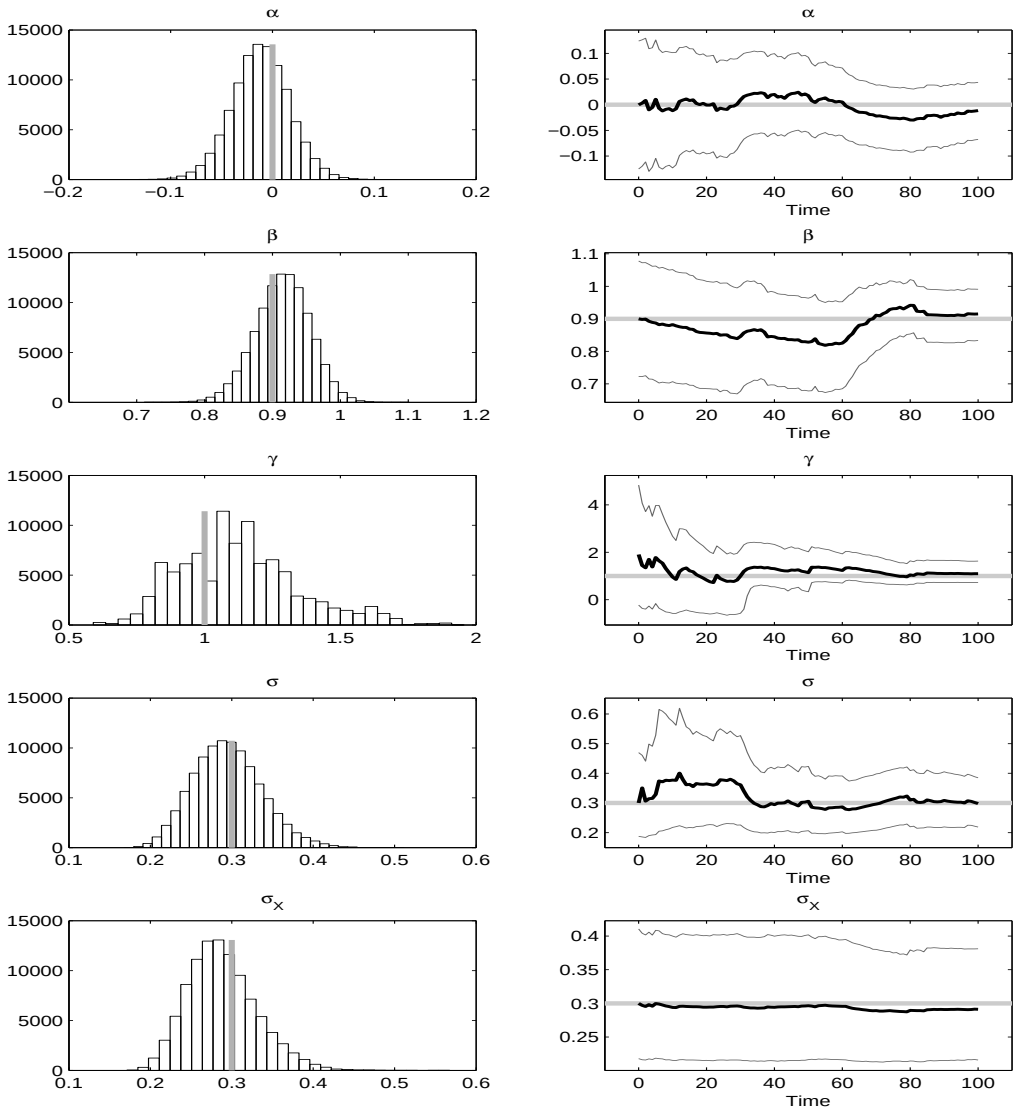

Fig. 1. Posterior distribution and sequential learning of the parameters in the exponential state space model. The gray vertical bar in the histogram represents the true value of the parameters. The thick gray horizontal line in the sequential learning plot denotes true value of the parameters. The thick black solid line and the thin gray lines show the posterior mean and 95% Bayesian probability region at each time respectively. Particle size is $100k$.

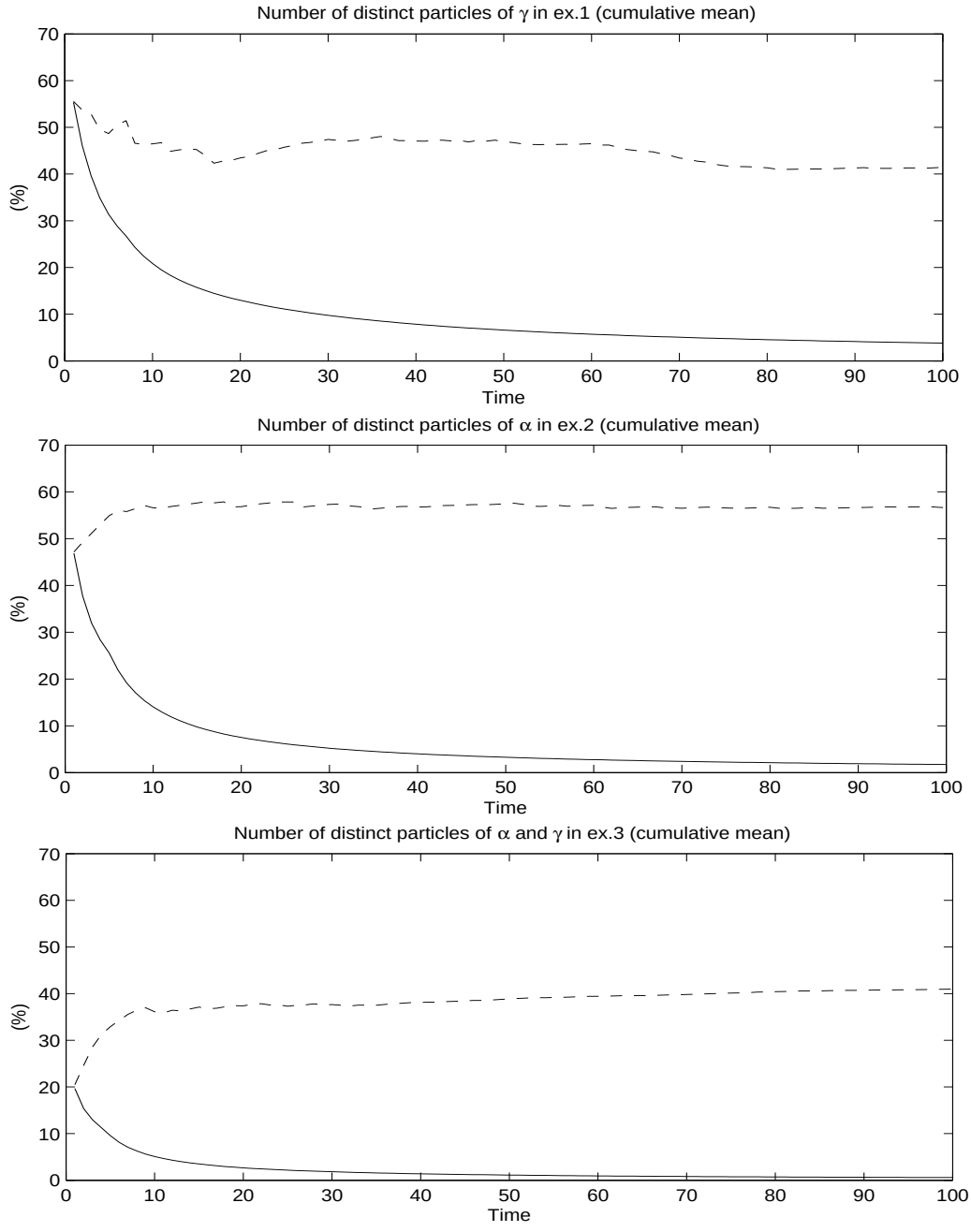

Fig. 2. The number of distinct particles of non-linear parameters expressed by the cumulative average and the percentiles. The dashed line represents JPY method and the solid line is standard method without sufficient statistics.

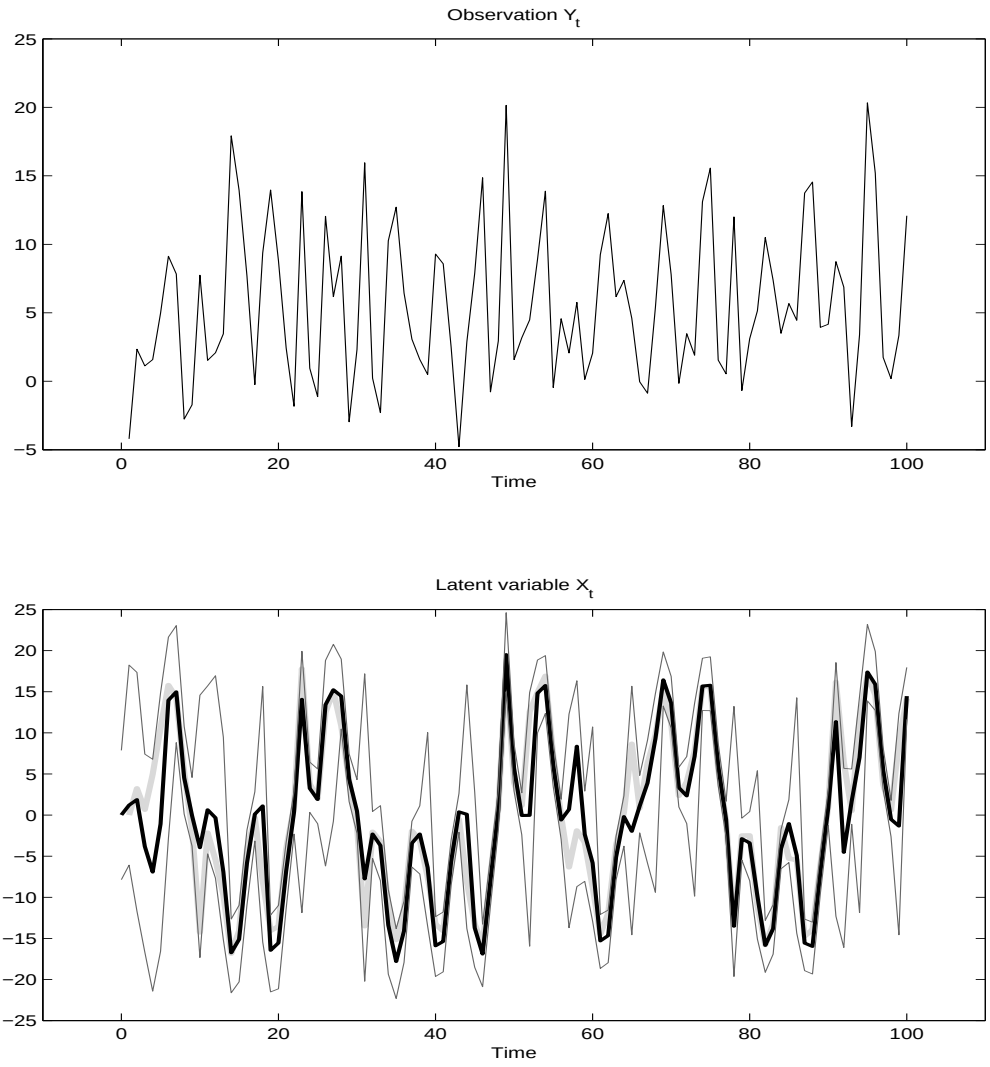

Fig. 3. (Top) Time series of observation y_t in the non-stationary growth model. (Bottom panel) Filtered state variable x_t (thick black line) with the 95% Bayesian probability region (thin gray line) and true simulated x_t (Thick gray line).

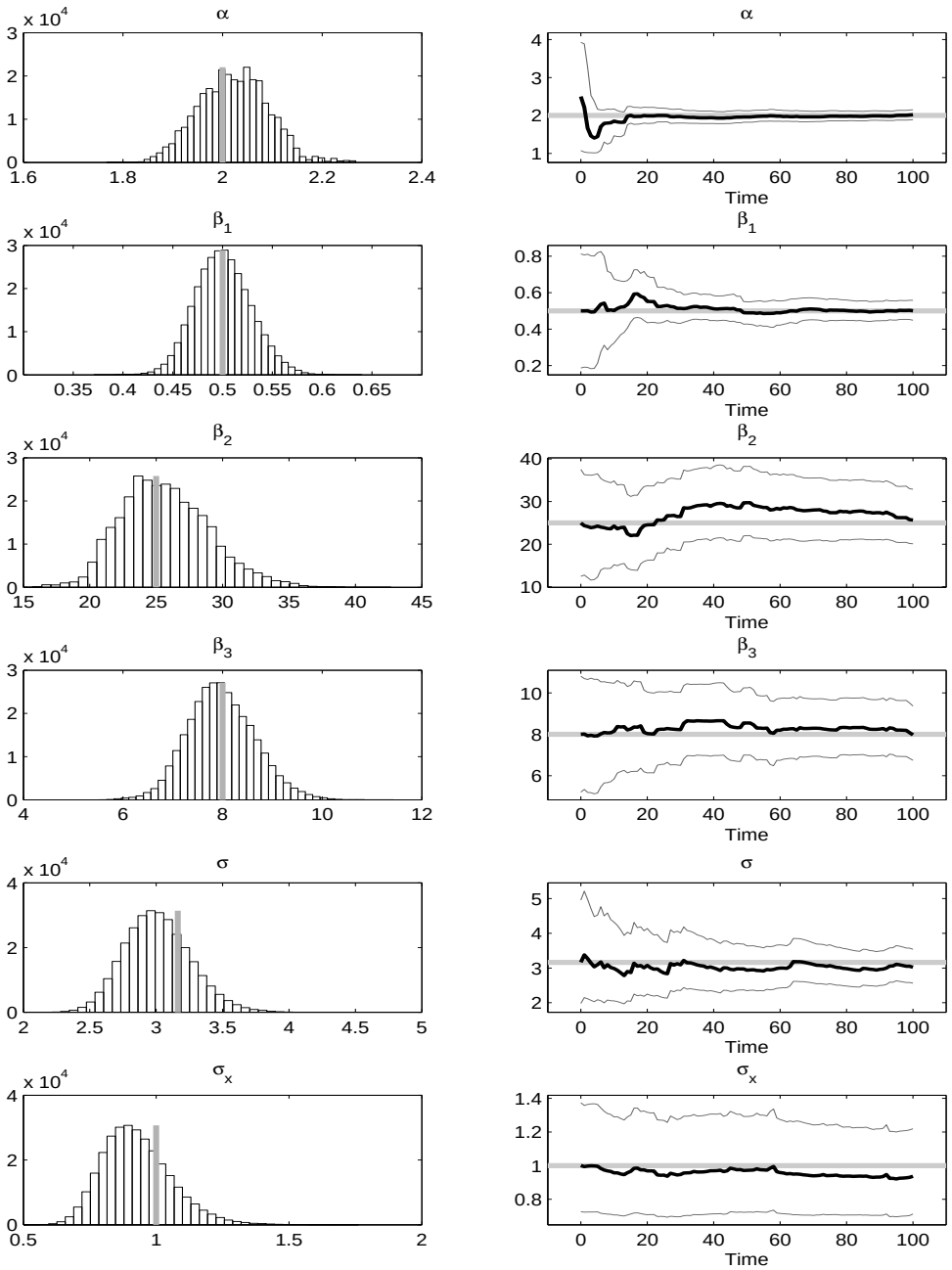

Fig. 4. Posterior distribution and sequential learning of the parameters in the non-stationary growth model. The lines are defined as in Figure 1. Particle size is $300k$.

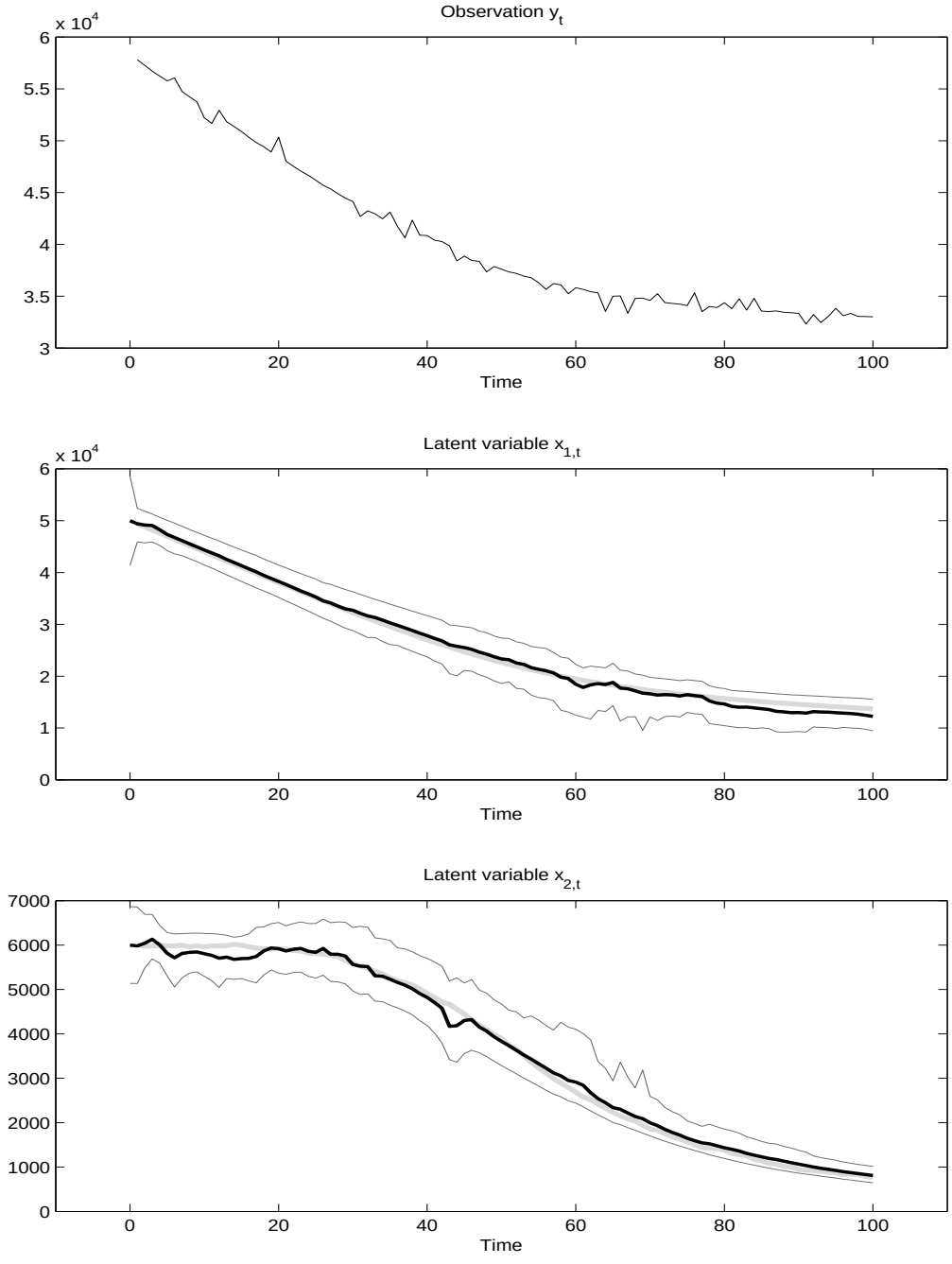

Fig. 5. (Left column) Filtered state variable $x_{1,t}$ (Altitude) and $x_{2,t}$ (Velocity). Particle size is $500k$. $\sigma_0 = 900$, $\sigma_1 = 30$, $x_{1,1} = 50,000$, $x_{2,1} = 6,000$ and $g = 9.8$. Model and data for 10 seconds are discretized into total 100 data points. The lines in the plots represent the same meanings as in Figure 1.

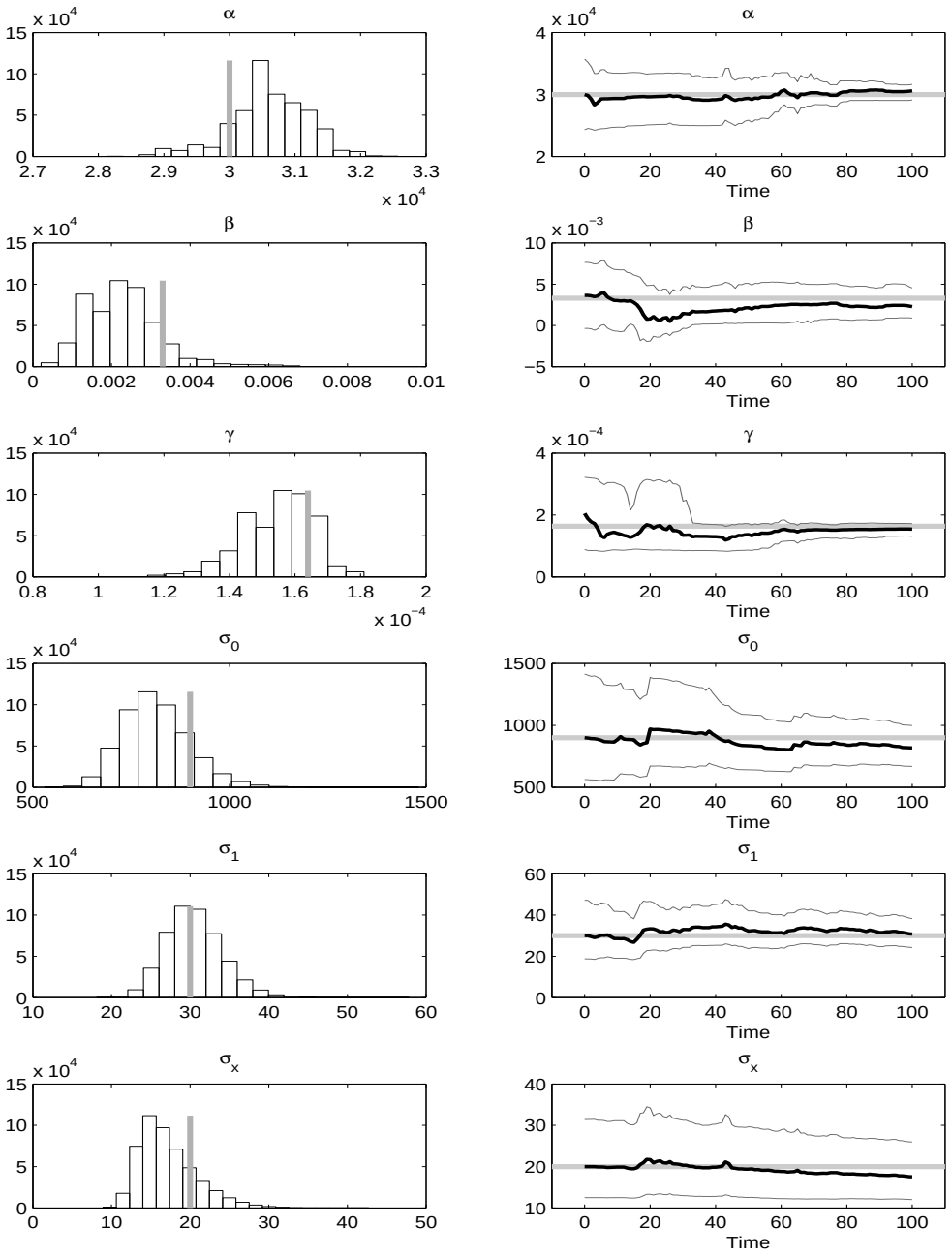

Fig. 6. (Left column) Parameter learning in the radar tracking problem. $\sigma_0 = 900$, $\sigma_1 = 30$, $x_{1,1} = 50,000$, $x_{2,1} = 6,000$ and $g = 9.8$. Model and data for 10 seconds are discretized into total 100 data points. The lines in the plots represent the same meanings as in Figure 1. Particle size is $500k$.