

Particle filters for state space models with the presence of unknown static parameters

Geir Storvik*

Email: `geirs@math.uio.no`

Abstract

In this paper particle filters for dynamic state space models handling unknown static parameters are discussed. The approach is based on marginalizing the static parameters out of the posterior distribution such that only the state vector needs to be considered. Such a marginalization can always be applied. However, real-time applications are only possible when the distribution of the unknown parameters given both observations *and* the hidden state vector depends on some low-dimensional sufficient statistics. Such sufficient statistics are present in many of the commonly used state space models. Marginalizing the static parameters avoids the problem of impoverishment which typically occur when static parameters are included as part of the state vector. The filters are tested on several different models, with promising results.

Keywords

State space models, sequential updating, particle filters, global parameters, marginalization, sufficient statistics

To be published in IEEE Trans. of Signal Processing (2002)

*Geir Storvik is an Associate Professor at the University of Oslo and a part time researcher at the Norwegian Computing Center.

1 Introduction

Dynamic state space models [1, 2, 3] are useful for describing data in many different areas, for instance engineering [4], finance mathematics [5], environmental data [6], geophysical science [7] and disease data [8].

Using $p(\cdot|\cdot)$ for a generic conditional distribution, the general (discrete time) state space model is given by

$$\mathbf{x}_k \sim p(\mathbf{x}_k | \mathbf{x}_{k-1}; \boldsymbol{\theta}), \quad \text{system} \tag{1a}$$

$$\mathbf{z}_k \sim p(\mathbf{z}_k | \mathbf{x}_k; \boldsymbol{\theta}), \quad \text{observations} \tag{1b}$$

where \mathbf{z}_k contains the observations at time t_k , while $\{\mathbf{x}_k\}$ is an unobserved underlying stochastic process. In some cases \mathbf{x}_k may have a physical meaning while in other cases merely is included in order to describe the distribution of the observation process properly. $\boldsymbol{\theta}$ is a vector containing static parameters which in some cases can be specified but in many cases are unknown. Typically, some prior distribution is placed on \mathbf{x}_0 .

An important task when analyzing data by state space models is estimation of the underlying state process based on measurements from the observation process. The interest might be on \mathbf{x}_k itself (or a function of \mathbf{x}_k), or \mathbf{x}_k is a tool for making prediction on \mathbf{z}_k . Given data $\mathbf{z}_{1:N} = (\mathbf{z}_1, \dots, \mathbf{z}_N)$, estimation of $\mathbf{x}_{1:N}$ is usually referred to as *off-line* estimation, while *on-line* estimation is sequential estimation of \mathbf{x}_k based on $\mathbf{z}_{1:k}$ for $k = 1, 2, 3, \dots$. In this paper, our effort will be on the latter problem. In particular the focus will be on problems where new observations arrive frequently (hours/minutes/seconds) and real-time estimation/prediction is essential.

In a few cases, including linear Gaussian models and hidden Markov chains, the distribution of \mathbf{x}_k given the observations $\mathbf{z}_{1:k}$ can be computed exactly using recursive formula's. For situations where analytical solutions are impossible to obtain, stochastic simulation can be applied. Numerous papers have been written on construction of algorithms based on Markov Chain Monte Carlo (MCMC) dealing with general state space models (see Gamerman [9] and the references therein). Although such procedures may be effective for off-line estimation, there are problems with full MCMC in the case of on-line estimation. The MCMC algorithm needs to be restarted at each time point. Further, the dimension of the vector to be simulated increases with time.

An alternative to full MCMC at each time point is construction of simulation algorithms for sequential updating the posterior distributions. Such (slightly different) algorithms have been developed independently in many fields [10, 11, 12, 13, 14, 15] with different names (Bootstrap filter, Monte Carlo filter, Particle filter, Condensation algorithm). The excellent review by Doucet [16] even contains some references back to the late 60's. A collection of papers describing the state of the art in this field can be found in [17]. In this paper, such algorithms will be denoted *particle filters*.

The main idea behind particle filters is to represent the posterior distributions $p(\mathbf{x}_{1:k} | \mathbf{z}_{1:k})$ through a finite set of samples or particles which can be used to estimate any property of $p(\mathbf{x}_{1:k} | \mathbf{z}_{1:k})$ in an ordinary Monte Carlo estimation framework. When a new observation \mathbf{z}_k

arrives, the particles are updated in order to represent the new posterior $p(\mathbf{x}_{1:k+1}|\mathbf{z}_{1:k+1})$. Techniques for performing this updating include rejection sampling [10], importance sampling [18], sampling/importance resampling [11] and MCMC [14, 19]. A main computational problem with the general approach is that the dimension of the distribution increases with time. In many cases only $p(\mathbf{x}_k|\mathbf{z}_{1:k})$ is of interest. If earlier state variables can be *disengaged* [14], fast computation can be performed. Disengagement in this setting means that variables at previous time points can be neglected in the sequential computation algorithm.

Although particle filters have been successful in many simulation experiments and in analysis of real data, a main problem with such an approach is how to handle the presence of unknown *static* parameters. A common trick in engineering is to include the parameters as part of the state space vector \mathbf{x}_k . Berzuini et al. [14] put this approach into a formal Bayesian setting. However, the non-dynamics in the parameters makes the parameter samples degenerate into one or a few different values when k increases.

Gordon et al. [11], West [20], Bølviken et al. [21] and Liu and West [22] introduced diversity in the set of particles by adding random noise to the particles, which in this context is similar to approximate the non-dynamic parameters by some slowly changing dynamic ones. In addition to the problem of choosing the “diversity” procedure, this results in old observations being down-weighted and the parameter estimates obtained at a given time is mainly depending on the most recent observations.

This paper considers an alternative approach. Suppose that for *given* $\mathbf{x}_{1:k}$ and $\mathbf{z}_{1:k}$ the distribution of $\boldsymbol{\theta}$ is analytically tractable. In particular, the distribution of $\boldsymbol{\theta}$ is assumed to depend on $\mathbf{x}_{1:k}$ only through some low-dimensional sufficient statistics. In such cases, only samples of $\mathbf{x}_{1:k}$ given $\mathbf{z}_{1:k}$ is needed, since estimates of the posterior distribution for the static parameters can be obtained either through Rao-Blackwellization or by a simple additional simulation step. Furthermore, updating the particle set to a new particle set one time-step further on can be performed by simulation of the state vector *and* the parameters simultaneously. The approach can be considered as a marginalization of $\boldsymbol{\theta}$ from the posterior.

The main idea has been suggested several places before. Liu and Chen [18, section 5] call the procedure *Rao-Blackwellization*, but state that “when disengagement is implemented, Rao-Blackwellization is no longer directly applicable”. In this paper we demonstrate that by including the sufficient statistics for $\boldsymbol{\theta}$ into the state vector, the procedure *is* possible to combine with disengagement. Sufficient statistics were also applied in [23] for models with discrete valued state variables. No general treatment of this approach has however been given in the literature. The main contribution of this paper is to demonstrate the usefulness of marginalization in certain classes of state models when estimation of static parameters and dynamic state variables is performed simultaneously.

Although the required assumptions will restrict the set of models possible to process by this approach, many important (and widely used) models are included. In particular, models for which the underlying process is Gaussian and linear in the parameters involved (but not necessarily linear in the \mathbf{x} -process) can be handled by this approach. Further, the assumption about Gaussian noise in the system process can be relaxed to include

“partial non-Gaussian” processes as defined by Shephard [24]. Both T-distributions and mixture of Gaussians fall into this group of models. Also discrete-valued Markov models and mixtures of these and Gaussian-based models can be used. These system processes can be combined with any observational distribution which do not contain any additional unknown parameters. Many observational distributions *with* unknown parameters can be handled by this approach, but typically special treatment is needed in each case.

In Section 2 the general particle filters are reviewed. Section 3 introduces the particle filters for situations with unknown static parameters. Section 4 considers some particular classes of models which fit into this framework, while the filters are applied on different types of models in Section 5. Finally, a summary and discussion is given in Section 6.

2 Particle filters in general

This section discusses particle filters in situations where the static parameters are known. For the time being θ will be suppressed in the notation.

Today many different versions of particle filters exist (see Doucet et al. [17]). Two different motivations are typically used in construction of a filter. One approach is based on importance sampling. In this case \mathbf{x}_k is simulated sequentially from some importance distribution $f_k(\mathbf{x}_k|\mathbf{x}_{1:k-1}, \mathbf{z}_{1:k})$ and the whole trajectory $\mathbf{x}_{1:k}$ is given importance weight

$$w_k = \frac{p(\mathbf{x}_{0:k}|\mathbf{z}_{1:k})}{f_0(\mathbf{x}_0) \prod_{l=1}^k f_l(\mathbf{x}_l|\mathbf{x}_{1:l-1}, \mathbf{z}_{1:l})}. \quad (2)$$

N such sequences are simulated in parallel, giving a weighted particle set $\mathbf{S}_k = \{(\mathbf{x}_{1:k}^{(i)}, w_k^{(i)}), i = 1, \dots, N\}$ at each time point t_k . Restrictions on the importance distributions are needed both for ease in simulation and in order to make the computation of the weights possible. See [18] for further discussion on this approach, which usually is named *sequential importance sampling* (SIS). A problem with this approach is that when time evolves the variance of the weights will increase [25] making the estimate (5) unstable. A common trick to avoid this is to *resample* from \mathbf{S}_k with probabilities proportional to $w_k^{(i)}$. Liu and Chen [18] give some heuristics on when to resample.

An alternative approach is based on the ordinary histogram approximation of the density $p(\mathbf{x}_{1:k}|\mathbf{z}_{1:k-1})$:

$$\hat{p}(\mathbf{x}_{1:k-1}|\mathbf{z}_{1:k-1}) = \frac{\sum_{i=1}^N w_{k-1}^{(i)} I(\mathbf{x}_{1:k-1} = \mathbf{x}_{1:k-1}^{(i)})}{\sum_{i=1}^N w_{k-1}^{(i)}} \quad (3)$$

where $I(A)$ is the indicator function for event A .

By Bayes rule and (1),

$$\begin{aligned} p(\mathbf{x}_{1:k}|\mathbf{z}_{1:k}) &\propto p(\mathbf{x}_{1:k-1}|\mathbf{z}_{1:k-1})p(\mathbf{x}_k|\mathbf{x}_{k-1})p(\mathbf{z}_k|\mathbf{x}_k) \\ &\approx \hat{p}(\mathbf{x}_{1:k-1}|\mathbf{z}_{1:k-1})p(\mathbf{x}_k|\mathbf{x}_{k-1})p(\mathbf{z}_k|\mathbf{x}_k). \end{aligned} \quad (4)$$

A new particle set \mathbf{S}_k can now be obtained by simulating from this approximative distribution.

A possible rejection sampling procedure [10] for simulation from (4) is to sample $\mathbf{x}_{1:k}$ from $\hat{p}(\mathbf{x}_{1:k}|\mathbf{z}_{1:k-1})$ and accept the sample with probability proportional to $p(\mathbf{z}_k|\mathbf{x}_k)$. This procedure can be repeated N times in order to obtain a new (unweighted) particle set $\mathbf{S}_k = \{\mathbf{x}_k^1, \dots, \mathbf{x}_k^B\}$. In practice the acceptance probability for this simple algorithm will be far too low, making the need for other approaches. In [19] constructions of more efficient proposal distributions are given and some other sampling approaches including sampling/importance resampling and MCMC are discussed. Note that the use of MCMC here is on a much smaller dimension than if a full MCMC scheme was to be applied.

A main problem with the particle filters described so far is that when simulating $\mathbf{x}_{1:k}$ at time k , the first $k-1$ components can only take the values given in \mathbf{S}_{k-1} . Gilks and Berzuini [26] and Carpenter et al. [27] introduced the possibilities of changing the whole vector $\mathbf{x}_{1:k}$ according to some Markov transition kernel having $p(\mathbf{x}_{1:k}|\mathbf{z}_{1:k})$ as stationary distribution. Such an approach *does* solve some of the degeneracy problems that can occur for the more standard particle filters, but not in general. The complexity of simulating from the Markov transition kernel will increase with time, giving similar problems as for full MCMC simulation.

Given the main framework, much freedom is available to the user on how to specify the algorithm. However, care has to be taken in order to make the algorithm work properly both at a fixed time point t_k for large enough N and for fixed N with time increasing. Storvik [28] demonstrated through some simulation experiments that the numerical errors introduced can accumulate linearly for some particle filters. This is particularly the case when unknown static parameters are present. Using more sophisticated filters can however remove this error accumulation. This is the aim of the filter that will be presented in the next section.

Assuming $f(\mathbf{x}_{1:k})$ is of interest, the posterior expectation $E_k^f = \mathbb{E}[f(\mathbf{x}_{1:k})|\mathbf{z}_{1:k}]$ is approximated by

$$\hat{E}_k^f = \sum_{i=1}^N w_k^{(i)} f(\mathbf{x}_{1:k}^{(i)}) / \sum_{i=1}^N w_k^{(i)}. \quad (5)$$

Recently, some theoretical results on particle filters and their associated Monte Carlo estimates have appeared. Berzuini et al. [14] established a central limit theorem for the estimator (5) for the sequential importance sampling approach with resampling at each stage. More general results on convergence is given in [29], which shows that most algorithms proposed will converge properly. These results are however based on increasing the number of particles N to infinity. In [3] it is proven that under certain conditions the error in the approximative distribution remains stable if N grows like k^2 as k increases. The order k^2 can probably be improved by introducing additional conditions or by constructing more efficient filters. Still however, to the author's knowledge, theoretical results on how errors propagate in time when N is fixed, is missing.

3 Particle filters including non-dynamic parameters

In this section an approach for particle filtering in the presence of unknown parameters will be discussed. The usual approach is to include the parameters as part of the state vector $(\mathbf{x}_k, \boldsymbol{\theta})$. Because of the non-dynamic feature of the parameters, samples of $\boldsymbol{\theta}$ at time t_k can only take the values given at time t_{k-1} . Since some of these values become very unlikely when new observations arrive, this will result in an impoverishment of the set of distinct $\boldsymbol{\theta}$ values.

The approach taken in this paper is based on a different idea. Assume that the posterior distribution of $\boldsymbol{\theta}$ given $\mathbf{z}_{1:k}$ and $\mathbf{x}_{1:k}$ depend on some sufficient statistics $\mathbf{T}_k = \mathbf{T}_k(\mathbf{x}_{1:k}, \mathbf{z}_{1:k})$, where \mathbf{T}_k is easy to update recursively.

Assume that an approximate particle set \mathbf{S}_{k-1} is available from the posterior distribution $p(\mathbf{x}_{1:k-1}|\mathbf{z}_{1:k-1})$. Again the particle set \mathbf{S}_{k-1} is to be updated to a new particle set \mathbf{S}_k at time t_k . Even though only the $\{\mathbf{x}_k\}$ process in addition to the sufficient statistics $\{\mathbf{T}_k\}$ will be stored, simulation simplifies if $\boldsymbol{\theta}$ is included as an ancillary variable in the simulation step. The approach is based on the following

$$\begin{aligned} p(\mathbf{x}_{1:k}, \boldsymbol{\theta}|\mathbf{z}_{1:k}) &= C p(\mathbf{x}_{1:k}, \boldsymbol{\theta}, \mathbf{z}_k|\mathbf{z}_{1:k-1}) \\ &= C p(\mathbf{x}_{1:k-1}|\mathbf{z}_{1:k-1}) p(\boldsymbol{\theta}|\mathbf{x}_{1:k-1}, \mathbf{z}_{1:k-1}) \times \\ &\quad p(\mathbf{x}_k|\mathbf{x}_{1:k-1}, \mathbf{z}_{1:k-1}, \boldsymbol{\theta}) p(\mathbf{z}_k|\mathbf{x}_k, \mathbf{z}_{1:k-1}, \boldsymbol{\theta}) \\ &= C p(\mathbf{x}_{1:k-1}|\mathbf{z}_{1:k-1}) p(\boldsymbol{\theta}|\mathbf{T}_{k-1}) p(\mathbf{x}_k|\mathbf{x}_{k-1}, \boldsymbol{\theta}) p(\mathbf{z}_k|\mathbf{x}_k, \boldsymbol{\theta}), \end{aligned} \quad (6)$$

where $C = [p(\mathbf{z}_k|\mathbf{z}_{1:k-1})]^{-1}$, a constant not depending on $\mathbf{x}_{1:k}$ or $\boldsymbol{\theta}$. Using the approximation in (3), simulation from (6) can be performed as before, but with the additional step that also $\boldsymbol{\theta}$ needs to be simulated. The simplest approach would be to simulate $\mathbf{x}_{1:k-1}$ from $\hat{p}(\mathbf{x}_{1:k-1}|\mathbf{z}_{1:k-1})$, $\boldsymbol{\theta}$ from $p(\boldsymbol{\theta}|\mathbf{T}_{k-1})$, \mathbf{x}_k from $p(\mathbf{x}_k|\mathbf{x}_{1:k-1}; \boldsymbol{\theta})$ and accept with probability proportional to $p(\mathbf{z}_k|\mathbf{x}_k; \boldsymbol{\theta})$. However, any simulation technique such as sampling/importance resampling or MCMC can be applied. Also the SIS approach can be used.

The important part about this approach is that the parameter $\boldsymbol{\theta}$ simulated at time t_k does not depend on values simulated at previous time points. This avoids the problem with impoverishment.

In principle, the existence of a low-dimensional sufficient statistic for $\boldsymbol{\theta}$ is not necessary, because only evaluation or simulation from $p(\boldsymbol{\theta}|\mathbf{x}_{1:k}, \mathbf{z}_{1:k})$ is needed, as noted by Liu and Chen [18]. However, in order to make the filter run fast and not have increasing complexity as time evolves, the need for $p(\boldsymbol{\theta}|\mathbf{x}_{1:k}, \mathbf{z}_{1:k})$ only to depend on $(\mathbf{x}_{1:k}, \mathbf{z}_{1:k})$ through \mathbf{T}_k becomes apparent.

Following [25], a SIS with resampling (SISR) algorithm which includes static parameters is defined as follows:

Importance sampling: For $i = 1, \dots, N$,

- sample $\boldsymbol{\theta} \sim f_{k,1}(\boldsymbol{\theta}|\mathbf{x}_{0:k-1}^{(i)}, \mathbf{z}_{1:k})$,

- sample $\tilde{\mathbf{x}}_k^{(i)} \sim f_{k,2}(\mathbf{x}_k | \mathbf{x}_{0:k-1}^{(i)}, \mathbf{z}_k, \boldsymbol{\theta})$ and define $\tilde{\mathbf{x}}_{1:k}^{(i)} \triangleq (\mathbf{x}_{0:k-1}^{(i)}, \tilde{\mathbf{x}}_k^{(i)})$,
- evaluate the importance weights

$$\tilde{w}_k^{(i)} = w_{k-1}^{(i)} \frac{p(\boldsymbol{\theta} | \mathbf{T}_{k-1}^{(i)}) p(\tilde{\mathbf{x}}_k^{(i)} | \mathbf{x}_{k-1}^{(i)}, \boldsymbol{\theta}) p(\mathbf{z}_k | \tilde{\mathbf{x}}_k^{(i)}, \boldsymbol{\theta})}{f_{k,1}(\boldsymbol{\theta} | \mathbf{x}_{0:k-1}^{(i)}, \mathbf{z}_{1:k}) f_{k,2}(\tilde{\mathbf{x}}_k^{(i)} | \mathbf{x}_{k-1}^{(i)}, \mathbf{z}_k, \boldsymbol{\theta})}.$$

Resampling: For $i = 1, \dots, N$,

- sample an index j_i from $\{1, \dots, N\}$ with probabilities proportional to $\tilde{w}_k^{(j_i)}$,
- put $\mathbf{x}_{1:k}^{(i)} = \tilde{\mathbf{x}}_{1:k}^{(j_i)}$, $\mathbf{T}_k^{(i)} = T(\mathbf{T}_{k-1}^{(j_i)}, \mathbf{x}_k^{(i)}, \mathbf{z}_k)$ and $w_k^{(i)} = N^{-1}$.

Here $f_{k,1}(\boldsymbol{\theta} | \mathbf{x}_{0:k-1}^{(i)}, \mathbf{z}_{1:k})$ and $f_{k,2}(\tilde{\mathbf{x}}_k^{(i)} | \mathbf{x}_{k-1}^{(i)}, \mathbf{z}_k, \boldsymbol{\theta})$ are proposal distributions for $\boldsymbol{\theta}$ and \mathbf{x}_k , respectively. Typically $f_{k,1}(\boldsymbol{\theta} | \mathbf{x}_{0:k-1}^{(i)}, \mathbf{z}_{1:k}) = f_{k,1}(\boldsymbol{\theta} | \mathbf{T}_{k-1}^{(i)}, \mathbf{z}_k)$ in order to make both simulation and computation fast. $T(\cdot)$ is a function updating the sufficient statistics.

As for the ordinary SIS algorithm, resampling can be performed at each time step or according to some specific rules [12]. Stratified sampling [27] and other variance reduction methods can also easily be incorporated.

In the case of resampling at each time step and with $f_{k,1}(\boldsymbol{\theta} | \mathbf{x}_{0:k-1}^{(i)}, \mathbf{z}_{1:k}) = p(\boldsymbol{\theta} | \mathbf{T}_{k-1}^{(i)})$ the algorithm can be seen as a special case of the *resample-move* algorithm due to [26] where the move step at time $t - 1$ corresponds to sampling $\boldsymbol{\theta}$ from $p(\boldsymbol{\theta} | \mathbf{x}_{0:k-1}^{(i)}, \mathbf{z}_{1:k-1})$.

Samples of $\boldsymbol{\theta}$ based on $\mathbf{z}_{1:k}$ are directly available through the algorithm. In order to estimate $\boldsymbol{\theta}$, a better approach is to use Rao-Blackwellization, as described in [25].

4 Gaussian-based system processes

In this section some particular classes of models will be discussed. Only models for system processes will be considered, because these can be discussed in general terms. When unknown parameters are present in the observational distribution, special treatment usually is needed. This is therefore discussed through specific examples in Section 5. In many cases, the parameters involved in the observation process are given from other sources. In such cases, no extra treatment is needed.

We will concentrate on Gaussian-based models. This is both because such models are commonly applied as system processes and because sufficient statistics are easily calculated for this class of models.

4.1 Gaussian system processes

A particular useful class of models is obtained when the underlying state process is Gaussian, but where the observation distribution is arbitrary (though following (1b)). Assume

$$\mathbf{x}_k = \mathbf{F}_k' \boldsymbol{\beta} + \boldsymbol{\varepsilon}_k, \quad \boldsymbol{\varepsilon}_k \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{Q}), \quad (7)$$

where $\mathbf{F}_k = \mathbf{F}(\mathbf{x}_{k-1})$ is a matrix with elements possibly non-linear functions of \mathbf{x}_{k-1} . The unknown parameters are in this case β and σ (\mathbf{Q} is assumed known, the general case can also be handled but becomes much more complex).

Assume β and σ^2 to have priors $\beta \sim \mathcal{N}(\beta_0, \sigma^2 \mathbf{C}_0)$ and $\sigma^2 \sim \text{IG}(\frac{\nu_0}{2}, \frac{d_0}{2})$ where IG is the inverse Gamma distribution. Then a trivial extension of the standard theory [2] yields

$$[\beta | \mathbf{x}_{1:k}, \mathbf{z}_{1:k}, \sigma^2] \sim \mathcal{N}(\mathbf{m}_k, \sigma^2 \mathbf{C}_k) \quad (8a)$$

$$[\sigma^2 | \mathbf{x}_{1:k}, \mathbf{z}_{1:k}] \sim \text{IG}(\frac{\nu_k}{2}, \frac{d_k}{2}) \quad (8b)$$

where the sufficient statistics $\mathbf{m}_k, \mathbf{C}_k, d_k$ and ν_k are updated according to the equations

$$\mathbf{D}_k = \mathbf{F}_k' \mathbf{C}_{k-1} \mathbf{F}_k + \mathbf{Q}, \quad (9a)$$

$$\mathbf{C}_k = \mathbf{C}_{k-1} - \mathbf{C}_{k-1} \mathbf{F}_k \mathbf{D}_k^{-1} \mathbf{F}_k' \mathbf{C}_{k-1}, \quad (9b)$$

$$\mathbf{m}_k = \mathbf{m}_{k-1} + \mathbf{C}_{k-1} \mathbf{F}_k \mathbf{D}_k^{-1} [\mathbf{x}_k - \mathbf{F}_k' \mathbf{m}_{k-1}], \quad (9c)$$

$$d_k = d_{k-1} + (\mathbf{x}_k - \mathbf{F}_k' \mathbf{m}_{k-1})' \mathbf{D}_k^{-1} (\mathbf{x}_k - \mathbf{F}_k' \mathbf{m}_{k-1}), \quad (9d)$$

$$\nu_k = \nu_{k-1} + q. \quad (9e)$$

where q is the dimension of \mathbf{x}_k . The particle filter approach described in Section 3 can therefore easily be applied.

The choice of distributions for priors on β and σ^2 are crucial in order to obtain the analytical tractable forms (8a) and (8b), but should be sufficient in most cases. Priors with little information can be obtained choosing a large \mathbf{C}_0 and ν_0 and d_0 small. More complex priors are possible to obtain by using mixtures of Gaussians (for β) and mixtures of inverse gamma distributions (for σ^2). Choosing such mixtures as priors will change the posteriors (8a) and (8b) to mixtures of Gaussians and inverse gamma distributions, respectively.

4.2 Partial non-Gaussian state space

Shephard [24] introduced a class of non-Gaussian time-series models allowing the noise process to be T-distributed or a mixture of Gaussians. The particle filter approach presented in this paper is also applicable for this type of models. For illustration purpose, only T-distributed noise will be considered.

Rewrite model (7) as

$$\mathbf{x}_k = \mathbf{F}_k' \beta + \check{\mathbf{e}}_k / \sqrt{\omega_k}, \quad \check{\mathbf{e}}_k \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{Q}) \text{ and } \omega_k \sim \chi_\nu^2, \quad (10)$$

with $\check{\mathbf{e}}_k$ and ω_k being independent. The posterior distributions of β and σ^2 given both $\mathbf{x}_{1:k}$ and $\omega_{1:k}$ become equal to (8), but the updating formula's (9) are slightly changed by replacing \mathbf{Q} in (9a) by \mathbf{Q}/ω_k .

The state vector needs in this case to be extended to include ω_k . Simulation *conditional on* ω_k can be performed as in the Gaussian case with small modifications. Simulation of

ω_k given all the other variables is also easy,

$$\omega_k \sim \mathcal{G}\left(\frac{\nu + q}{2}, \frac{1}{2}\left(1 + \frac{1}{\sigma^2}(\mathbf{x}_k - \mathbf{F}'_k \boldsymbol{\beta})' \mathbf{Q}^{-1}(\mathbf{x}_k - \mathbf{F}'_k \boldsymbol{\beta})\right)\right),$$

where q is the dimension of \mathbf{x}_k . Direct simulation of all variables involved is however not longer possible, but a blocked Gibbs sampler approach switching between sampling ω_k and a block containing all the other variables can be applied.

5 Experiments

In this section some examples of dynamic models will be considered in order to evaluate the performance of the particle filter when some static parameters are unknown. For each example, a SISR filter including static parameters using $f_{k,1}(\boldsymbol{\theta}|\mathbf{x}_{0:k-1}^{(i)}, \mathbf{z}_{1:k}) = p(\boldsymbol{\theta}|\mathbf{T}_{k-1})$ and $f_{k,2}(\mathbf{x}_k|\mathbf{x}_{0:k-1}, \mathbf{z}_{1:k}, \mathbf{z}_k, \boldsymbol{\theta}) = p(\mathbf{x}_k|\mathbf{x}_{k-1}, \boldsymbol{\theta})$ is applied. Resampling is performed at each time step. The result is that the weights $w_k^{(i)}$ in (2) reduces to $p(\mathbf{z}_k|\tilde{\mathbf{x}}_k^{(i)}, \boldsymbol{\theta})$. More efficient filters, where the proposal distribution depends on the new observation \mathbf{z}_k , can however be constructed similar to the standard SISR filters.

In each case, the estimated posterior distributions is compared to those obtained from a full MCMC run at each time step using a huge number of iterations. For the example in Section 5.1, a comparison is made towards other methods used in the literature.

In all examples, the number of particles has been chosen such that a reasonable agreement with the results obtained by the MCMC runs was obtained. Guidelines for specifying N in practice is still missing, the rule of thumb being try and fail.

5.1 A linear partial Gaussian process

The first example is a simple linear model where the observation noise is assumed to follow a T -distribution. The model can be written as

$$x_k = ax_{k-1} + \sigma \varepsilon_k, \tag{11a}$$

$$z_k = x_k + \tau v_k, \tag{11b}$$

where $\{\varepsilon_k\}$ and $\{v_k\}$ are independent zero-mean white noise processes, the first being Gaussian with variance equal to one, while the other follows a T -distribution with ν degrees of freedom. Such models can be used to allow for outliers in the observations [24]. The unknown static parameters are in this case $\boldsymbol{\theta} = (a, \sigma^2, \tau^2)$ (ν is assumed known).

Since the system process follows the model discussed in Section 4.1, the sufficient statistics for (a, σ^2) can be updated according to the equations in (9). Given the model formulation above, no sufficient statistic for τ is available. Rewrite, however, the observation model as

$$z_k = x_k + \tau \check{v}_k / \sqrt{\delta_k},$$

where \check{v}_k is a Gaussian variable, while δ_k is an independent χ_ν^2 -variable. The distribution for z_k is of course unaltered, but the point is that a sufficient statistic for τ given $z_{1:k}, x_{1:k}$ and $\delta_{1:k}$ is available. In particular,

$$[\tau^2 | x_{1:k}, z_{1:k}, \delta_{1:k}] \sim \text{IG}(\frac{\eta_k}{2}, \frac{e_k}{2})$$

where

$$\begin{aligned} e_k &= e_{k-1} + \delta_k(z_k - x_k)^2, \\ \eta_k &= \eta_{k-1} + 1. \end{aligned}$$

Similar to the approach discussed in Section 4.2, (x_k, δ_k) is used as state vector in the particle filter. Note that simulation from the prior of δ_k is simply to draw from a χ_ν^2 distribution, since there is no dynamic structure in this variable.

Data was simulated from this model with $a = 0.9, \sigma^2 = 1, \tau^2 = 1$ and $\nu = 5$. For σ^2 and τ^2 , an inverse Gamma distribution with both shape and scale parameters equal to 0.5 was used. The prior distribution for a was chosen to be $N(0, \sqrt{10})$. In Figure 1, posterior means and quantiles obtained from the SISR filter using $N = 2000$ particles is plotted together with the same quantities calculated using full MCMC at each time step. For the $\{x_k\}$ process and the estimates of a at each time step, both posterior means and quantiles are almost identical. Some discrepancy is present for the σ estimates, but the results are still acceptable. Also the estimates for τ are very good.

Figure 1 around here.

Other filters have been proposed in the literature for handling static parameters. We have compared the filters discussed in this paper with two other approaches. Both approaches include θ into the state vector. The first approach retains the assumption that θ is static (as in Kitagawa [30]). In the second approach, following Bølviken et al. [21], and Liu and West [22], diversity is introduced into the parameter. In both cases, variance parameters are included on a log scale in order to ensure positivity. For further reference we will denote the filter based on marginalization and sufficient statistics by F_0 , while the filters based on including θ into the state vector will be denoted F_1 and F_2 , F_1 including diversity into the parameter.

None of F_1 and F_2 worked properly with the priors used for $\theta = (a, \sigma, \tau)$. The main problem was to get proper estimates of τ . In order to give some comparison, the prior for τ^2 was therefore changed to an inverse Gamma distribution with shape and scale parameters equal to 10. This gives a more informative prior, making estimation of τ easier. For all filters $N = 5000$ was used. Figure 2 compares these approaches. F_0 gave in this case almost identical results compared to MCMC. For F_1 , the problems in estimating τ is clearly seen. Reasonable estimates of σ is obtained, but the quantiles are very different from the ones obtained by MCMC. The problems in estimating the static parameters also influences the inference for x_k in that the credibility intervals become wider. The problem with impoverishment is clearly seen for algorithm F_2 . At $k = 40$ only a single value of θ from the N particles initially drawn has survived. This value is then impossible to change at later time points.

Figure 2 around here.

For this example, a comparison of computer times for the different algorithms were performed. Figure 3 shows computer times used for each time point for the particle filter F_0 (solid line), filter F_1 (dotted line) and the MCMC algorithm (dashed line). Times used for F_0 and F_1 are almost identical. The number of iterations used for MCMC was fixed for each time step, giving the linear increase in computation time. This number was chosen such that there was reasonable agreement with results obtained using huge number of MCMC iterations. A block Gibbs sampler (with the whole state vector (x_1, \dots, x_n) sampled simultaneously) was used for the MCMC algorithm. All computer times were obtained from C-programs run on a Dual Pentium II 350 MHz computer with 128 MB RAM and 512 kB Cache. The differences in computer time between algorithms F_0 and F_1 are negligible. The advantage in computer time using the particle filters compared to MCMC is clearly seen. In practice, with the number of time points increasing, also the number of iterations in the MCMC algorithm typically need to be increased, giving even higher computational cost for the MCMC algorithm. Better MCMC algorithms using Metropolis-Hastings steps could possibly decrease computation time, but would not change the main picture that while the computational cost for particle filters remains constant with time, it increases (linearly or more) for MCMC algorithms.

Figure 3 around here.

5.2 A dynamic generalized linear model

West et al. [31] considered a general class of dynamic Bayesian models. They studied the case where the underlying system process is linear and the distribution for the observation \mathbf{z}_k conditioned on the underlying state vector \mathbf{x}_k is in the exponential family. We will consider applications where the observed data are (possibly multivariate) binary, making the logistic model an obvious choice. Such models have been used in e. g. ecology [32] where a number of observers indicate whether the population at the current time is either high or low. Here, only a simplified version will be considered:

$$x_k \sim N(ax_{k-1}, \sigma^2), \quad (12a)$$

$$z_k \sim \text{Binom}(r, \text{logit}\{\alpha + \beta x_k\}), \quad (12b)$$

for $k = 1, 2, \dots$. The unknown static parameters are $\boldsymbol{\theta} = (a, \sigma^2, \alpha, \beta)$.

Data were simulated according to the model with $a = 0.9, \sigma^2 = 1$ and $\alpha = \beta = 0.5$. Assume first the parameters in the observation process, α and β , are known, while a priori $a \sim N(0, \sqrt{10})$ and $\sigma^2 \sim IG(0.5, 0.5)$. In this particular case the recursions given in (9) can be applied to update the sufficient statistics. Figure 4 shows posterior means and quantiles obtained from the SIS filter using $N = 2000$ particles. The same quantities calculated using full MCMC at each time step are also plotted for comparison. For all estimates, both posterior means and quantiles are almost identical.

Figure 4 around here.

Turn now to the case when also α and β are unknown. In this case it will be advantageous to reparameterize the model such that all the parameters become part of the system process. This can be done by defining $\check{x}_k = \alpha + \beta x_k$. Then the model can be written as

$$\check{x}_k \sim N(\alpha + a(\check{x}_{k-1} - \alpha), \check{\sigma}^2), \quad (13a)$$

$$z_k \sim \text{Binom}(r, \text{logit}\{\check{x}_k\}), \quad (13b)$$

where $\check{\sigma} = \beta\sigma$. Note that β and σ are only identifiable through their product. This means that the $\{x_k\}$ process can only be recovered up to an unknown scale factor. In the ecology example referred to above this is not a serious concern since interest is primarily in the seasonal patterns of the process.

We will assume a priori $a \sim N(0, \sigma_a^2)$, $\check{\sigma}^2 \sim IG(\frac{\nu_0}{2}, \frac{d_0}{2})$ and $\alpha \sim N(0, \sigma_\alpha^2)$. In this case, the model does not fit into (7) because a and α appear in the model through their product. Also direct simulation from $p(a, \check{\sigma}^2, \alpha | T_k)$ is not possible. It is however easy to show that

$$\begin{aligned} [\check{\sigma}^2 | a, \alpha, \check{x}_{1:k}, z_{1:k}] &\sim IG(\nu_k/2, d_k/2) \\ [a | \check{\sigma}^2, \alpha, \check{x}_{1:k}, z_{1:k}] &\sim N(m_k^a, c_k^a) \\ [\alpha | \check{\sigma}^2, a, \check{x}_{1:k}, z_{1:k}] &\sim N(m_k^\alpha, c_k^\alpha), \end{aligned}$$

where

$$\begin{aligned} \nu_k &= \nu_0 + k \\ d_k &= d_0 + \sum_{l=1}^k \check{x}_l^2 - 2a(1-a)\alpha \sum_{l=1}^k \check{x}_l \check{x}_{l-1} + a^2 \sum_{l=1}^k \check{x}_{l-1}^2 - \\ &\quad 2(1-a)\alpha \left[\sum_{l=1}^k \check{x}_l - a \sum_{l=1}^k \check{x}_{l-1} \right] + (k-1)(1-a)^2 \alpha^2 \\ m_k^a &= \frac{\sigma_a^2 (\sum_{l=1}^k \check{x}_l \check{x}_{l-1} - \alpha \sum_{l=1}^k (\check{x}_l + \check{x}_{l-1}) + (k-1)\alpha^2)}{\check{\sigma}^2 + \sigma_a^2 (\sum_{l=1}^k \check{x}_{l-1}^2 - 2\alpha \sum_{l=1}^k \check{x}_{l-1} + (k-1)\alpha^2)} \\ c_k^a &= \frac{\sigma_a^2 \check{\sigma}^2}{\check{\sigma}^2 + \sigma_a^2 (\sum_{l=1}^k \check{x}_{l-1}^2 - 2\alpha \sum_{l=1}^k \check{x}_{l-1} + (k-1)\alpha^2)} \\ m_k^\alpha &= \frac{\sigma_\alpha^2 (1-a) (\sum_{l=1}^k \check{x}_l - a \sum_{l=1}^k \check{x}_{l-1})}{\check{\sigma}^2 + (k-1)\sigma_\alpha^2 (1-a)^2} \\ c_k^\alpha &= \frac{\sigma_\alpha^2 \check{\sigma}^2}{\check{\sigma}^2 + (k-1)\sigma_\alpha^2 (1-a)^2}, \end{aligned}$$

showing that

$$\mathbf{T}_k = \left(\sum_{l=1}^k \check{x}_l, \sum_{l=1}^k \check{x}_{l-1}, \sum_{l=1}^k \check{x}_l^2, \sum_{l=1}^k \check{x}_{l-1}^2, \sum_{l=1}^k \check{x}_l \check{x}_{l-1} \right)$$

is a sufficient statistic for $(a, \check{\sigma}^2, \alpha)$ given $(\check{x}_{1:k}, z_{1:k})$. A SISR algorithm which samples $(a, \check{\sigma}^2, \alpha)$ approximately from $p(a, \check{\sigma}^2, \alpha | T_k)$ using a few Gibbs sampling steps was therefore

applied. \check{x}_k was simulated from $p(\check{x}_k|a, \check{\sigma}^2, \alpha, \check{x}_{k-1})$. The weights $\tilde{w}_k^{(i)}$ were put to $p(z_k|\check{x}_k)$, ignoring the error introduced by using the approximative Gibbs sampling algorithm for simulating the static parameters.

In Figure 5, posterior means and quantiles obtained from the SISR filter using $N = 2000$ particles and 5 Gibbs sampler steps is plotted together with the same quantities calculated using full MCMC at each time step. The free software BUGS [33] was used for the MCMC runs. Also in this case, there is a nice agreement between the estimates obtained by the particle filter and the ones given by the full MCMC runs.

Figure 5 around here.

5.3 A Gamma-Poisson model

Consider the model

$$x_k \sim N(ax_{k-1}, \sigma^2) \tag{14a}$$

$$y_k \sim \text{Poisson}(x_k^2) \tag{14b}$$

Note that marginally $x_k^2 \sim \text{Gamma}(\frac{1}{2}, \frac{2\sigma^2}{1-a^2})$, so this can also be described as a Gamma-Poisson process where a controls the autocorrelation of the Gamma-process, while $\beta = \frac{2\sigma^2}{1-a^2}$ defines the scale of the Gamma variable. Gamma-Poisson processes has been considered in e.g. [8].

Figure 6 shows the results obtained by using a SISR filter on this example where data were simulated from the true model using $a = 0.9$ and $\sigma = 1$. An inverse Gamma distribution with shape and scale parameters equal to 0.5 was used as prior for σ^2 , while $a \sim N(0, \sqrt{10})$ a priori. Because the posterior distribution of x_k becomes bimodal in this case, the ordinary Monte Carlo estimate (5) for x_k would not be sensible to use. Rather the whole distribution should be reported. In order to evaluate the performance of the *algorithm* in this case, estimates and quantiles of $|x_k|$ are shown instead.

The filter was applied with $N = 2000$ particles. Constructing efficient MCMC algorithms for this model is difficult. Instead, a comparison with a filter using $N = 50000$ particles was performed. Again the results are quite satisfactory.

Figure 6 around here.

6 Discussion

The particle filter is a powerful method for processing a huge range of dynamic models. This paper discusses an approach based on the particle filter for tackling unknown parameters.

The approach has been tested on several different models, all giving estimates almost identical to the ones obtained by running full MCMC at each time point. Running full MCMC is not practical in real time processing, because the number of variables to be simulated increases in time. In contrast, the particle filters discussed in this paper only

need a fixed amount of computation at each time point, making real time processing possible.

In order to make this approach work in real time, a crucial assumption is that the posterior distribution for the parameters depends on the underlying system process only through some sufficient statistics that can be updated recursively. For many models commonly applied, such sufficient statistics exist. In some other cases, the state vector can be extended by an additional variable, for which the extended model fits into the framework. When sufficient statistics are not available, the approach can still be applied, but the computational complexity will increase with time.

In this paper, only Gaussian based system processes combined with general observation distributions have been considered, in which case the sufficient statistics involved can be updated using Kalman type filters. The approach should however be possible to apply also for many other types of models for the system process. In particular, cases where the underlying state vector is a discrete valued Markov model can be handled using hidden Markov chain algorithms for updating the sufficient statistics [23]. Also mixtures of discrete valued Markov models and Gaussian based models can be handled. Any kind of distribution can be approximated by Gaussian mixtures by making the number of mixtures large enough. The computation time will however increase with the number of mixtures.

Acknowledgement

This work was finalized during the author's sabbatical at the Department of Statistics and Demography at the University of Odense. The author would like to thank the department for being able to use all their facilities during this period. I am also grateful to Bård Storvik and Bent Natvig and the anonymous referees for valuable comments on this paper.

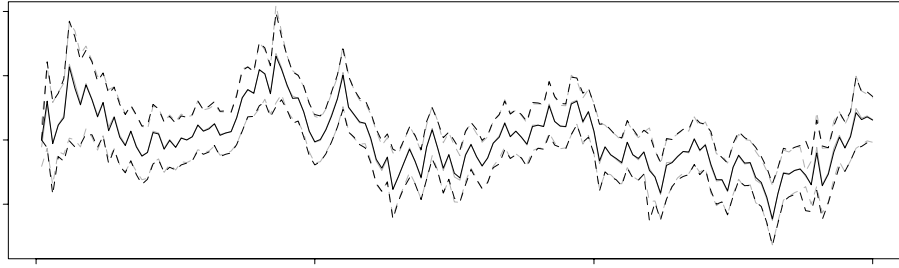
References

- [1] A. C. Harvey, *Forecasting, Structural Time Series Models and the Kalman Filter*, Cambridge: Cambridge University Press, 1989.
- [2] M. West and J. Harrison, *Bayesian forecasting and dynamic models*, Springer Series in Statistics. Springer-Verlag, New York, second edition, 1997.
- [3] H. R. Künsch, "State space and hidden Markov models," in *Complex Stochastic Systems*, O. E. Barndorff-Nielsen, D. R. Cox, and C. Klüppelberg, Eds., number 87 in Monographs on Statistics and Applied Probability, pp. 109–173. Chapman and Hall, London, 2001.
- [4] B. D. O. Anderson and J. B. Moore, *Optimal filtering*, Electrical Engineering Series. Prentice Hall, 1979.

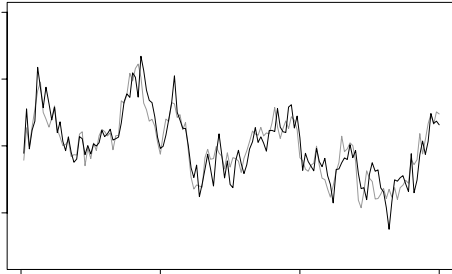
- [5] N. Shephard, “Statistical aspects of ARCH and stochastic volatility,” in *Time Series Models with Econometric, Finance and Other Applications*, D. R. Cox, D. V. Hinkley, and O. E. Barndorff-Nielsen, Eds., pp. 1–67. Chapman and Hall, London, 1996.
- [6] C. K. Wikle, M. Berliner, and N. Cressie, “Hierarchical Bayesian space-time models,” *Environmental and Ecological Statistics*, vol. 5, no. 2, 1998.
- [7] M. Ghil and K. Ide, “Data assimilation in meteorology and oceanography: Theory and practice,” *J. Meteor. Soc. Japan, Special Issue*, vol. 75, no. 1B, pp. 111–496, 1997.
- [8] B. Jørgensen, S. Lundbye-Christensen, P. X. K. Song, and L. Sun, “A longitudinal study of emergence room visits and air pollution for Prince George British Columbia,” *Statist. Med.*, vol. 15, pp. 823–836, 1996.
- [9] D. Gamerman, “Markov chain Monte Carlo for dynamic generalised models,” *Biometrika*, vol. 85, no. 1, pp. 215–227, 1998.
- [10] P. Müller, “Monte Carlo integration in general dynamic models,” *Contemporary Mathematics*, vol. 115, pp. 145–163, 1991.
- [11] N. Gordon, D. Salmond, and A. F. M. Smith, “Novel approach to nonlinear/non-Gaussian Bayesian state estimation,” *IEE Proceedings-F*, vol. 140, no. 2, pp. 107–113, 1993.
- [12] A. Kong, J. S. Liu, and W. H. Wong, “Sequential imputations and Bayesian missing data problems,” *Journal of the American Statistical Association*, vol. 89, no. 425, pp. 278–288, 1994.
- [13] G. Kitagawa, “Monte Carlo filter and smoother for non-Gaussian nonlinear state space models,” *J. Computational and Graphical Statistics*, vol. 5, pp. 1–25, 1996.
- [14] C. Berzuini, N. G. Best, W. R. Gilks, and C. Larizza, “Dynamic conditional independence models and Markov chain Monte Carlo methods,” *Journal of the American Statistical Association*, vol. 92, pp. 1403–141, 1997.
- [15] M. Isard and A. Blake, “Condensation - conditional density propagation for visual tracking,” *Int. J. of Computer Vision*, vol. 29, no. 1, pp. 5–28, 1998.
- [16] A. Doucet, “On sequential simulation-based method for Bayesian filtering,” Tech. Rep. TR.310, CUED/F-INFENG, 1998.
- [17] A. Doucet, N. de Freitas, and N. Gordon, *Sequential Monte Carlo in Practice*, Springer-Verlag, 2001.
- [18] J. S. Liu and R. Chen, “Sequential Monte Carlo methods for dynamic systems,” *Journal of the American Statistical Association*, vol. 93, no. 443, pp. 1032–1044, 1998.

- [19] M. K. Pitt and N. Shephard, “Filtering via simulation: Auxiliary particle filters,” *Journal of the American Statistical Association*, vol. 94, no. 446, pp. 590–599, 1999.
- [20] M. West, “Mixture models, Monte Carlo, Bayesian updating and dynamic models,” *Computing Science and Statistics*, vol. 24, pp. 325–333, 1993.
- [21] E. Bølviken, P. J. Acklam, N. Christophersen, and J. M. Størdal, “Monte Carlo filters for non-linear state estimation,” Tech. Rep. 14, University of Oslo, department of mathematics, section of statistics, 1997.
- [22] J. Liu and M. West, “Combined parameter and state estimation in simulation-based filtering,” in *Sequential Monte Carlo in Practice*, A. Doucet, N. de Freitas, and N. Gordon, Eds., pp. 197–223. Springer-Verlag, 2001.
- [23] T. C. Clapp and S. J. Godsill, “Fixed-lag smoothing using sequential importance sampling,” in *Bayesian Statistics 6*, J. M. Bernardo, J. O. Berger, P. Dawid, and A. F. M. Smith, Eds., pp. 743–751. Oxford University Press, 1999.
- [24] N. Shephard, “Partial non-Gaussian state space,” *Biometrika*, vol. 81, no. 1, pp. 115–31, 1994.
- [25] A. Doucet, S. Godsill, and C. Andrieu, “On sequential Monte Carlo sampling methods for Bayesian filtering,” *Statistics and Computing*, vol. 10, pp. 197–208, 2000.
- [26] W. R. Gilks and C. Berzuini, “Following a moving target – Monte Carlo inference for dynamic Bayesian models,” *Journal of Royal Statistical Society, Series B*, vol. 63, pp. 1–20, 2001.
- [27] T. R. Carpenter, P. Clifford, and P. Fearnhead, “Building robust simulation-based filters for evolving data sets,” Tech. Rep., Department of Statistics, University of Oxford, UK, 1998.
- [28] G. Storvik, “Some further topics on Monte Carlo methods for dynamic Bayesian problems,” in *Models and inference in HSSS: Recent developments and perspectives*. 2002, To appear.
- [29] D. Crisan and A. Doucet, “Convergence of sequential Monte Carlo methods,” Submitted for publication, 2000.
- [30] G. Kitagawa, “A self-organized state-space model,” *Journal of the American Statistical Association*, vol. 93, no. 443, pp. 1–1215, 1998.
- [31] M. West, J. Harrison, and H. S. Migon, “Dynamic generalized linear models and Bayesian forecasting,” *Journal of the American Statistical Association*, vol. 80, pp. 73–97, 1985.

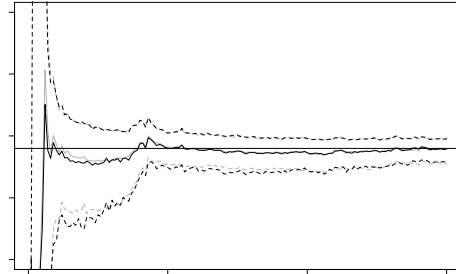
- [32] E. Bølviken and G. Storvik, “Deterministic and stochastic particle filters in state space models,” in *Sequential Monte Carlo in Practice*, A. Doucet, N. de Freitas, and N. Gordon, Eds. Springer-Verlag, 2001.
- [33] D. Spiegelhalter, A. Thomas, N. Best, and W. Gilks, “Bugs 0.5: Bayesian inference using Gibbs sampling, manual (version ii),” MRC Biostatistics Unit, Institute of Public Health, Robinson Way, Cambridge CB2 2SR, 1996.



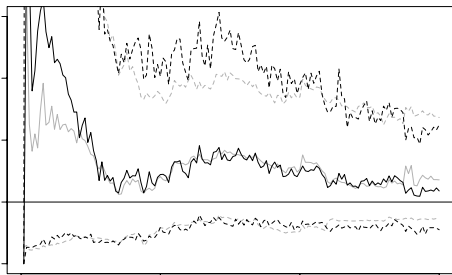
Estimates of x_k



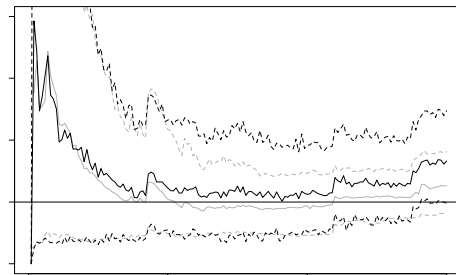
Estimates and true values of x_k



Estimates of a



Estimates of σ



Estimates of τ

Figure 1: *Model (11). Posterior expectations (solid lines) and 0.025 and 0.975 quantiles (dashed lines) based on a particle filter (black) and full MCMC (grey). In the middle left panel, the estimates of $\{x_k\}$ are given in black while the true values are given in grey.*

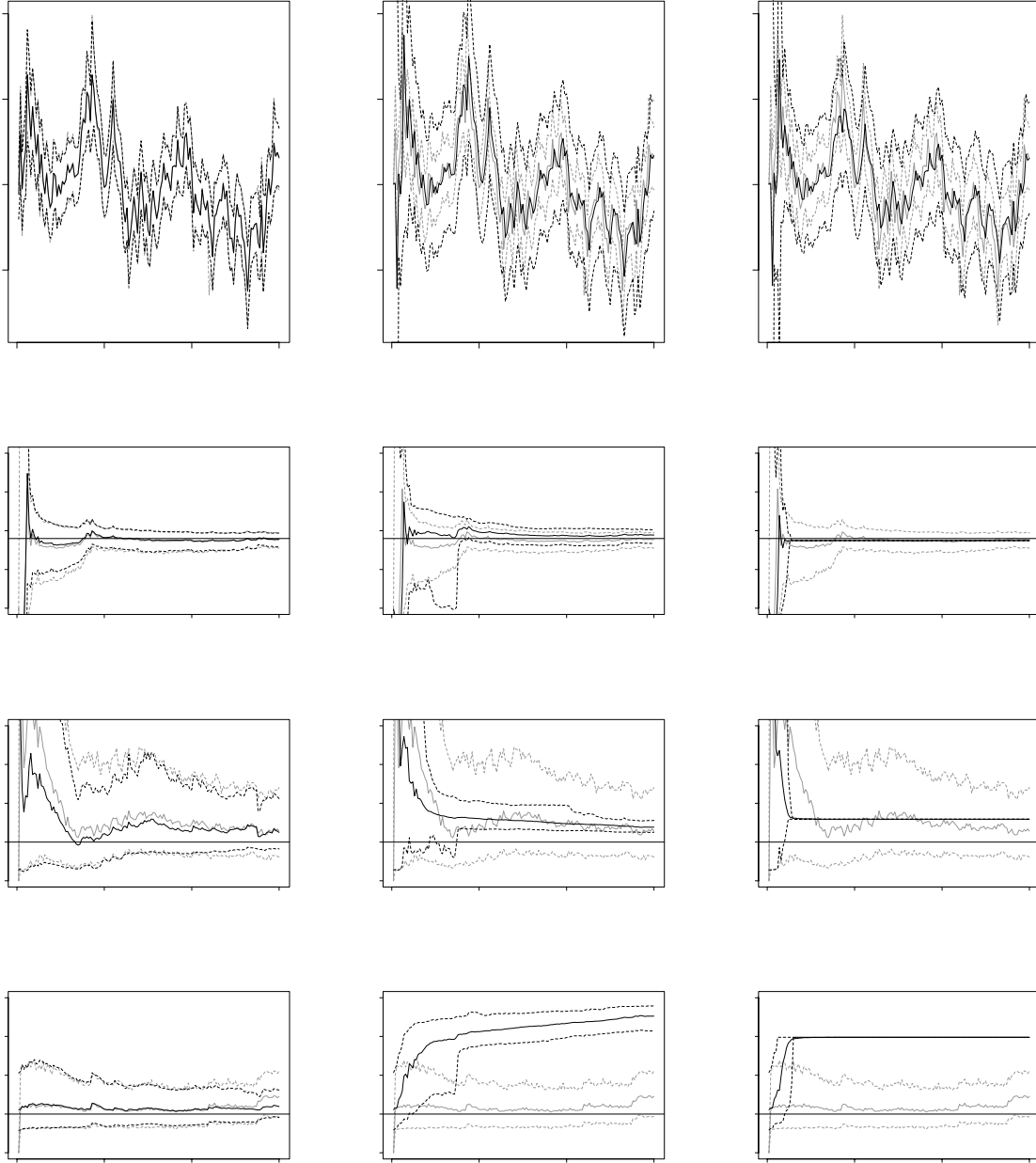


Figure 2: *Model (11) with an informative prior on τ . Posterior expectations (solid lines) and 0.025 and 0.975 quantiles (dashed lines) from particle filters (black) and MCMC (grey). First row shows estimates for x_k , second row for a , third row for σ and last row for τ . First column is algorithm F_0 , second is F_1 and third is F_2 .*

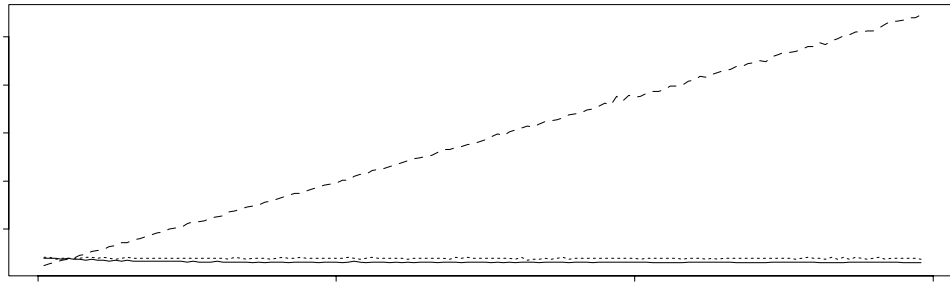
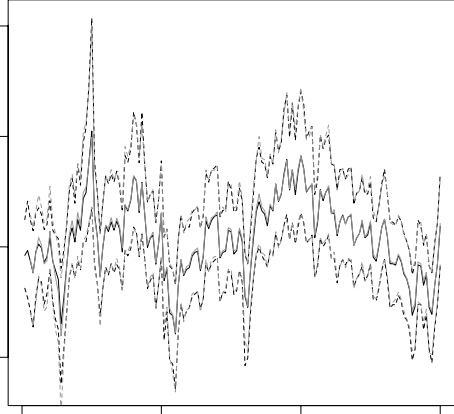
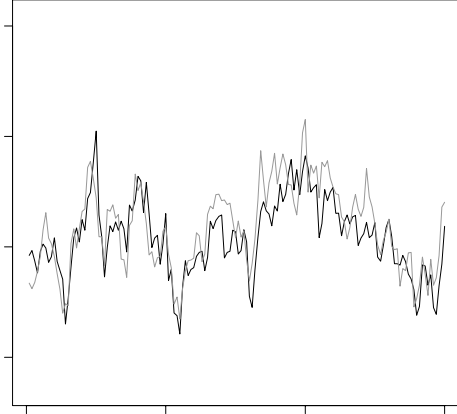


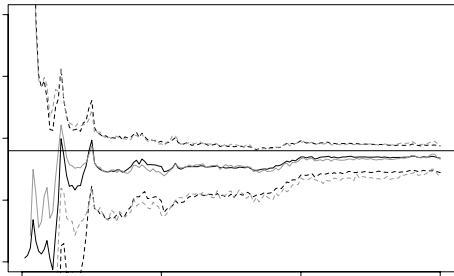
Figure 3: Comparison of times using particle filters (F_0 solid line and F_1 dotted line) and an MCMC algorithm (dashed line). The scale on the y axis is in seconds.



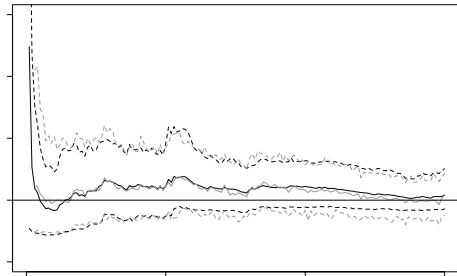
Estimates of x_k



Estimates and true values of x_k

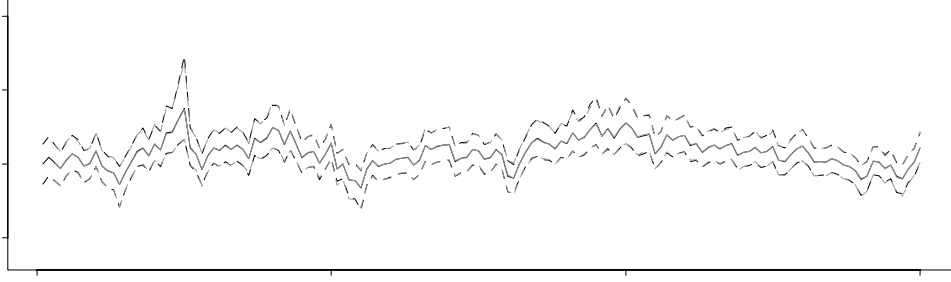


Estimates of a

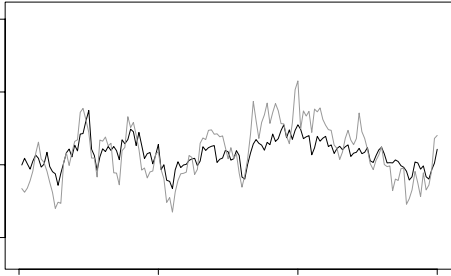


Estimates of σ

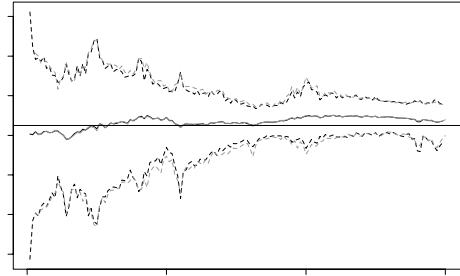
Figure 4: *Model (12) with α and β known. Posterior expectations (solid lines) and 0.025 and 0.975 quantiles (dashed lines) based on a particle filter (black) and full MCMC (grey). In the upper right panel, the estimates of $\{x_k\}$ are given in black while the true values are given in grey.*



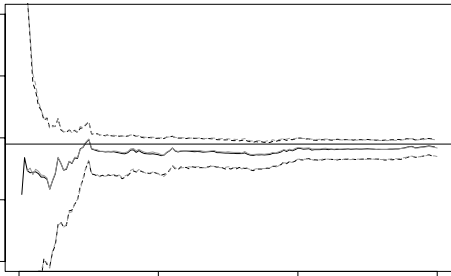
Estimates of x_k



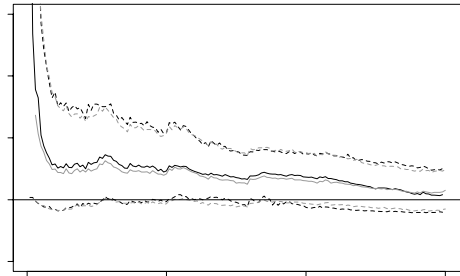
Estimates and true values of x_k



Estimates of α

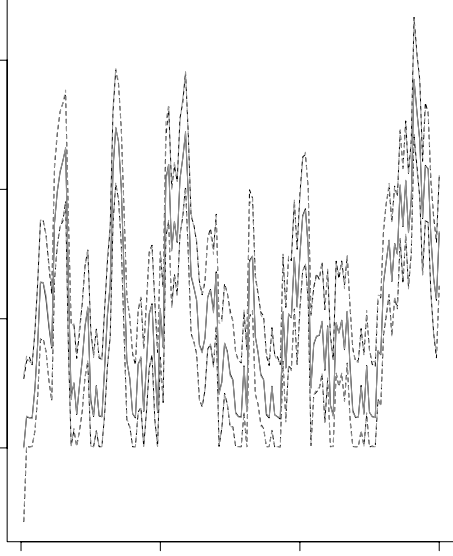


Estimates of a

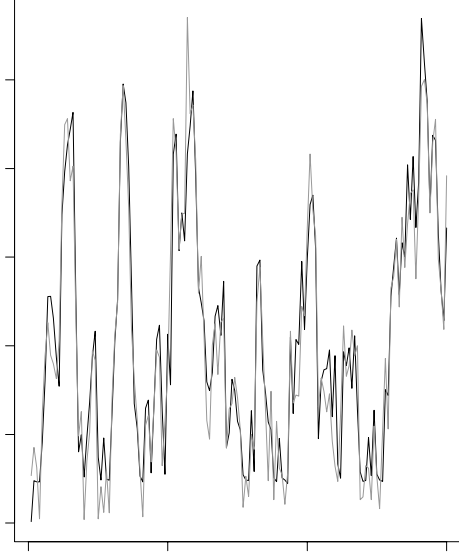


Estimates of $\tilde{\sigma}$

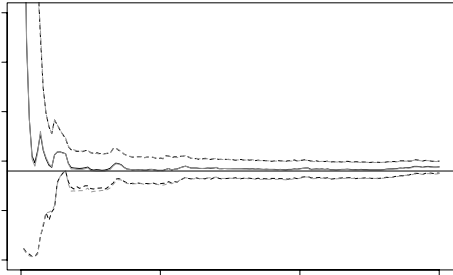
Figure 5: *Model (12) with α and β unknown. Posterior expectations (solid lines) and 0.025 and 0.975 quantiles (dashed lines) based on a particle filter (black) and full MCMC (grey). In the middle left panel, the estimates of $\{x_k\}$ are given in black while the true values are given in grey.*



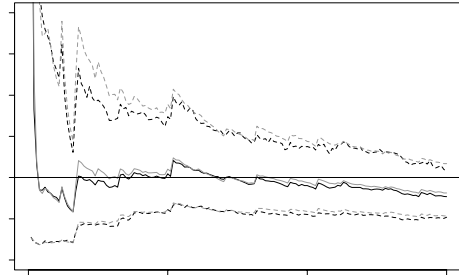
Estimates of $|x_k|$



Estimates and true values of $|x_k|$



Estimates of a



Estimates of σ

Figure 6: *Model (14). Posterior expectations (solid lines) and 0.025 and 0.975 quantiles (dashed lines) based on a particle filter using $N = 2000$ particles (black) and $N = 50000$ particles (grey). In the upper right panel, the estimates of $\{x_k\}$ are given in black while the true values are given in grey.*