FILTERING VIA SIMULATION: AUXILIARY PARTICLE FILTERS

Michael K Pitt
Department of Mathematics, Imperial College, Queen's Gate, London, SW7 2BZ, UK
m.k.pitt@ic.ac.uk
And
Neil Shephard
Nuffield College, University of Oxford, Oxford OX1 1NF, UK
neil.shephard@nuf.ox.ac.uk
www.nuff.ox.ac.uk/users/shephard/

October 22, 1997

Abstract

This paper analyses the recently suggested particle approach to filtering time series. We suggest that the algorithm is not robust to outliers for two reasons: the design of the simulators and the use of the discrete support to represent the sequentially updating prior distribution. Both problems are tackled in this paper. We believe we have largely solved the first problem and have reduced the order of magnitude of the second.

In addition we introduce the idea of stratification into the particle filter which allows us to perform on-line Bayesian calculations about the parameters which index the models and maximum likelihood estimation. The new methods are illustrated by using a stochastic volatility model and a time series model of angles.

Some key words: Filtering, Markov chain Monte Carlo, Particle filter, Simulation, SIR, State space.

1 INTRODUCTION

1.1 The model

In this paper we model a time series y_t, $t = 1, ..., n$, as being conditionally independent given an unobserved sufficient state α_t, which is itself assumed to be Markovian. The task will be to use simulation to carry out on-line filtering — that is to learn about the state given contemporaneously available information. We do this by estimating the difficult to compute density (or probability distribution function) $f(\alpha_t|y_1, ..., y_t) = f(\alpha_t|Y_t)$, $t = 1, ..., n$. In a later section we will extend this model to the case where lagged observations affect both the measurement and transition equations and to where y_t depends on both α_t and α_{t-1}.

We assume parametric forms for both the 'measurement' density $f(y_t|\alpha_t)$ and the 'transition' density of the state $f(\alpha_{t+1}|\alpha_t)$. The state evolution is initialized by some density $f(\alpha_0)$.

Filtering can be thought of as the repeated application of a two stage procedure. First the current density has to be propagated into the future via the transition density $f(\alpha_{t+1}|\alpha_t)$ to produce the prediction density

$$f(\alpha_{t+1}|Y_t) = \int f(\alpha_{t+1}|\alpha_t) dF(\alpha_t|Y_t). \tag{1}$$

Second, we move to the filtering density via Bayes theorem

$$f(\alpha_{t+1}|Y_{t+1}) = \frac{f(y_{t+1}|\alpha_{t+1})f(\alpha_{t+1}|Y_t)}{f(y_{t+1}|Y_t)}, \quad f(y_{t+1}|Y_t) = \int f(y_{t+1}|\alpha_{t+1}) dF(\alpha_{t+1}|Y_t). \tag{2}$$

This implies the data can be processed in a single sweep, updating our knowledge about the states as we receive more information. When the integrals cannot be analytically solved then numerical methods have to be used. These typically require us to be able to evaluate both $f(y_t|\alpha_t)$ and $f(\alpha_{t+1}|\alpha_t)$. We will see that the most basic of the methods developed in this paper will only require that we can simulate from $f(\alpha_{t+1}|\alpha_t)$ and evaluate $f(y_t|\alpha_t)$. If we can evaluate $f(\alpha_{t+1}|\alpha_t)$ then this knowledge can be used to improve the efficiency of the procedures.

There have been numerous attempts to provide algorithms which approximate the filtering densities. Important recent work includes Kitagawa (1987), West (1992b), West (1992a), Gerlach, Carter, and Kohn (1996) and those papers reviewed in Harvey (1989, Section 3.7) and West and Harrison (1997, Ch. 13 and 15).

This paper uses simulation to perform filtering following an extensive recent literature. Our approach is to develop the particle filter which has recently been suggested independently by various authors. In particular it is used by Gordon, Salmond, and Smith (1993) on non-Gaussian state space models. The same algorithm, with extensions to the smoothing problem, has been independently proposed by Kitagawa (1996) for use in time series problems. It reappears and is then discarded by Berzuini, Best, Gilks, and Larizza (1997) in the context of a real time application of the sequential analysis of medical patients. It is again proposed by Isard and Blake (1996), in the context of robustly tracking motion in visual clutter, under the name of the "condensation" algorithm. Similar ideas are used on the blind deconvolution problem by Liu and Chen (1995). Some statistical refinements of this class of algorithm, generically called particle filters, is given in a paper by Carpenter, Clifford, and Fearnhead (1997). The idea of calling this class of algorithm 'particle filters' is due to Carpenter, Clifford, and Fearnhead (1997), although the use of the phrase 'particles' appears in Kitagawa (1996).

Our paper will discuss the particle filtering literature and extend it in a number of directions so that it can be used in a much broader context.

The outline of the sections of this paper is as follows. In Section 2 we analyse the statistical basis of particle filters and focus on its weaknesses. In Section 3 we introduce our main contribution, which is an auxiliary particle filter method. This is as simple and as general as the original particle filter, but it is much more efficient when we deal with difficult problems. Further, it can be extended conveniently in cases where the transition and measurement densities of the state space model are analytically tractable. In Section 4 we generalize the particle filter approach to what we call a fixed lag filtering algorithm, where we update the discrete distribution due to the arrival of blocks of data. In addition we discuss the use of stratification in this context. In Section 5 we apply the particle filters to estimating unknown parameters using a Bayesian and maximum likelihood approach. Section 6 applies this work to a stochastic volatility model. Section 7 extend the methods to allow for feedback. Section 8 concludes, pointing out the remaining weaknesses of our new method.

2 PARTICLE FILTERS

2.1 Discrete support makes filtering easier

Importantly, if the support of α_t is finite set of known discrete points, rather than continuous, then the problem of approximating the integrals required to evaluate the filtering densities disappears. In particular the prediction density, (1), becomes

$$f(\alpha_{t+1}|Y_t) = \sum_{\alpha_t} \Pr(\alpha_{t+1}|\alpha_t) f(\alpha_t|Y_t),$$

which weighs all propagated points by the transition density. Likewise the filtering density, (2), is

$$f(\alpha_{t+1}|Y_{t+1}) = \frac{f(y_{t+1}|\alpha_{t+1}) f(\alpha_{t+1}|Y_t)}{\sum_{\alpha_{t+1}} f(y_{t+1}|\alpha_{t+1}) f(\alpha_{t+1}|Y_t)},$$

which weighs each state by the likelihood. Hence discrete state space models are easier to handle than continuous ones.

2.2 The definition of particle filters

Particle filters are the class of simulation filters which recursively approximate the filtering random variable $\alpha_t|Y_t = (y_1, ..., y_t)'$ by the cloud of points or 'particles' $\alpha_t^1, ..., \alpha_t^M$, with discrete probability mass of $\pi_t^1, ..., \pi_t^M$ respectively. Hence a continuous variable is approximated by a discrete one with a random support. These discrete points are thought of as samples from $f(\alpha_t|Y_t)$. In the literature all the π_t^j are assumed to all equal $1/M$, so the samples can be thought of as a random sample, but here we will allow more flexibility. Throughout M is taken to be very large. Then we assume that as $M \to \infty$, the particles can be used to increasingly well approximate the density of $\alpha_t|Y_t$.

Particle filters treat the discrete support generated by the particles as the true filtering density (this is similar to the bootstrap which treats the empirical distribution function as the true data generation process (see, for example, Efron and Tibshirani (1993, pp. 35-37))). This allows us to produce an approximation to the prediction density, (1), via the results on discrete filters. We call

$$\widehat{f}(\alpha_{t+1}|Y_t) = \sum_{j=1}^{M} f(\alpha_{t+1}|\alpha_t^j)\pi_t^j, \tag{3}$$

the 'empirical prediction density'. This can be combined with the measurement density to produce, up to proportionality,

$$\widehat{f}(\alpha_{t+1}|Y_{t+1}) \propto f(y_{t+1}|\alpha_{t+1}) \sum_{j=1}^{M} f(\alpha_{t+1}|\alpha_t^j)\pi_t^j, \tag{4}$$

the 'empirical filtering density' as an approximation to the true filtering density (2). Generically particle filters then sample from this density to produce new particles $\alpha_{t+1}^1, ..., \alpha_{t+1}^M$ with weights $\pi_{t+1}^1, ..., \pi_{t+1}^M$. This procedure can be repeated allowing us to progress through the data. We will call a particle filter 'exact' if it produces independent and identically distributed samples from the empirical filtering density.

If the particle filter can be made to work it could be used in a number of different contexts. First, it can be used in on-line tracking problems, which are important in many branches of applied science. Second, it is sometimes useful to be able to estimate the one-step ahead density $f(y_{t+1}|Y_t)$ and so, via the prediction decomposition, the joint density of the observation. This can be carried out in its simplest form by computing

$$\sum_{j=1}^{M} \left\{ \frac{1}{K} \sum_{k=1}^{K} f(y_{t+1}|\alpha_{t+1}^{j,k}) \right\} \pi_t^j, \qquad \text{where} \qquad \alpha_{t+1}^{j,k} \sim \alpha_{t+1}|\alpha_t^j, \qquad k = 1, ..., K,$$

where the $\alpha_{t+1}^{j,1},...,\alpha_{t+1}^{j,K}$ are drawn from $\alpha_{t+1}|\alpha_t^j$. If attention focuses on this quantity it may be worthwhile setting K to be larger than one. In addition the use of importance sampling or antithetic and control variables could be useful in this context if we have sufficient knowledge of the transition density to be able to employ them.

Third, a particularly useful diagnostic measure of fit for non-Gaussian statistical problems is to compute

$$\widehat{\Pr}(y_{t+1} \leq yobs_{t+1}|Y_t) = \sum_{j=1}^{M} \left\{ \frac{1}{K} \sum_{k=1}^{K} \Pr(y_{t+1} \leq yobs_{t+1}|\alpha_{t+1}^{j,k}) \right\} \pi_t^j,$$

where $\alpha_{t+1}^{j,k} \sim \alpha_{t+1}|\alpha_t^j$, $k = 1,...,K$, and where $yobs_{t+1}$ denotes the observation made at time $t+1$, as each estimate is estimating a random variable which should be uniformly and independently distributed on the $0,1$ interval (Rosenblatt (1952)). This allows the development of a whole portfolio of exact diagnostic tests via the routine application of Monte Carlo test (see, for example, Ripley (1987, pp. 171-4)). The use of this distribution function is emphasized in the work of Dawid (1982) and Smith (1985), Shephard (1994) and Gerlach, Carter, and Kohn (1996) in the time series context. Shephard (1994), Geweke (1994) and Gerlach, Carter, and Kohn (1996) give $O(n^2)$ algorithms for estimating this probability using the output from a Markov chain Monte Carlo (MCMC) algorithm, although the algorithm by Gerlach, Carter, and Kohn (1996) is typically quite fast for many models even when n is moderately large.

Particle filters are at first sight less useful in performing parameter estimation, due to the availability of MCMC methods for solving the much easier problem (as it involves no iterative approximations) of simulating from the joint density of the parameters and states given the whole of the data $y_1,...,y_n$. Examples of this include, Albert and Chib (1993), McCulloch and Tsay (1993), McCulloch and Tsay (1994), Carter and Kohn (1994), Carter and Kohn (1996), Shephard (1994), Fruhwirth-Schnatter (1994), West (1995), Carpenter, Clifford, and Fearnhead (1996) and Shephard and Pitt (1997). For a review, see West and Harrison (1997, Ch. 15). However, particle filters do offer the hope of allowing estimation in models where evaluating the transition density is difficult or impossible, as well as allowing on-line parameter estimation. To the authors knowledge designing MCMC algorithms for such problems is an open question as the Metropolis rejection rate generally involves the transition density.

2.3 Sampling the empirical prediction density

One way of sampling from the empirical prediction density is to think of $\sum_{j=1}^{M} f(\alpha_{t+1}|\alpha_t^j)\pi_t^j$ as a 'prior' density $\widehat{f}(\alpha_{t+1}|Y_t)$ which is combined with the likelihood $f(y_{t+1}|\alpha_{t+1})$ to produce a posterior. Then we have already assumed that we can simulate from $f(\alpha_{t+1}|\alpha_t^j)$, so we can sample from $\widehat{f}(\alpha_{t+1}|Y_t)$ by choosing α_t^j with probability π_t^j and then drawing from $f(\alpha_{t+1}|\alpha_t^j)$. If we can also evaluate $f(y_{t+1}|\alpha_{t+1})$ up to proportionality this leaves us with three sampling methods to draw from $f(\alpha_{t+1}|Y_{t+1})$: sampling/importance resampling, acceptance sampling and Markov chain Monte Carlo. In the rest of this section we write the prior as $f(\alpha)$ and the likelihood as $f(y|\alpha)$, abstracting from subscripts and conditioning arguments, in order to briefly review these methods in this context.

2.3.1 Sampling/importance resampling (SIR)

This method (due to Rubin (1987), Rubin (1988) and Smith and Gelfand (1992)) can be used to simulate from a posterior density $f(\alpha|y)$, given an ability to:

1. simulate from the prior $f(\alpha)$;
2. evaluate (up to proportionality) the conditional likelihood $f(y|\alpha)$ which is assumed to vary smoothly with α.

The idea is to draw proposals $\alpha^1, ..., \alpha^R$ from $f(\alpha)$ and then associate with each of these draws the weights π_j where

$$w_j = f(y|\alpha^j), \qquad \pi_j = \frac{w_j}{\sum_{i=1}^R w_i}, \qquad j = 1, ..., R.$$

Then the weighted sample will converge, as $R \to \infty$, to a non-random sample from the desired posterior $f(\alpha|y)$ as $\sum_{i=1}^R w_i \xrightarrow{p} f(y)$. Intuitively we would expect that the speed of convergence will depend upon the variability of these weights, so if the variability of the weights is small convergence will be quite rapid. The non-random sample can be converted into a random sample by resampling the $\alpha^1, ..., \alpha^R$ using weights $\pi_1, ..., \pi_R$ to produce an independent and identically distributed sample of size M. This requires $R \to \infty$ and $R >> M$. A major attraction of the SIR method is that it can be run efficiently on a massively parallel computer with a large amount of memory as each part of the computations can be carried out separately. A disadvantage of the method is that it typically requires R to be large and so is quite demanding in terms of storage.

The use of this method has been suggested in the particle filter framework by Gordon, Salmond, and Smith (1993), Kitagawa (1996), Berzuini, Best, Gilks, and Larizza (1997) and Isard and Blake (1996).

To understand the efficiency of the SIR method it is useful to think of the SIR method as an approximation to the importance sampler of the moment

$$E_{f\pi}\left\{h(\alpha)\right\} = \int h(\alpha)\pi(\alpha)dF(\alpha),$$

by

$$\frac{1}{R}\sum_{j=1}^{R} h(\alpha^j)\pi(\alpha^j), \quad \alpha \sim f(\alpha), \quad \pi(\alpha) = \frac{f(y|\alpha)}{f(y)}.$$

Notice that this setup implies $E_f\left\{\pi(\alpha)\right\} = 1$. The approximation comes about due to the SIR's π_j having a scaling factor which is a sum rather than an integral. Liu (1996) has recently studied the variance of this type of importance sampler and suggested that when $h(\alpha)$ does not vary very quickly with α then the variance is approximately proportional to

$$\frac{1 + var_f\left\{\pi(\alpha)\right\}}{R} = \frac{E_f\left\{\pi(\alpha)^2\right\}}{R}.$$

Hence the SIR method will become very imprecise when the π_j become very variable. This will happen if the likelihood is highly peaked compared to the prior.

Example Consider $\alpha \sim N(0,1)$, $y|\alpha \sim N(\alpha, \sigma^2)$. Then $E\left(w_j^k\right)$ equals (using the moment generating function of a non-central chi-squared)

$$\begin{aligned} E_\alpha \exp\left\{-\frac{k}{2\sigma^2}(y-\alpha)^2\right\} &= \left(1+\frac{k}{\sigma^2}\right)^{-1/2} \exp\left\{-\frac{ky^2}{\sigma^2} / \left(1+\frac{k}{\sigma^2}\right)\right\} \\ &= \left(1+\frac{k}{\sigma^2}\right)^{-1/2} \exp\left\{-\frac{ky^2}{(\sigma^2+k)}\right\}. \end{aligned}$$

Hence

$$E\left\{\pi(\alpha)^2\right\} = E\left(\frac{w_j}{E(w_j)}\right)^2 = \frac{\left(1+\frac{1}{\sigma^2}\right)}{\left(1+\frac{2}{\sigma^2}\right)^{1/2}} \exp\left\{\frac{2y^2}{(\sigma^2+2)(\sigma^2+1)}\right\}$$

increases exponentially in y^2, while it increases without bound as $\sigma^2 \to 0$. This confirms the above impresion of the fragility of the SIR method to outliers and to highly peaked likelihoods. Of course, for many problems the prior will be much more spread out than the likelihood and so the second of these problems should not be typically important. However, the sensitivity to aberrant observations will be important.

2.3.2 Rejection sampling

The SIR method has some similarities with rejection sampling (see, for example, Ripley (1987, pp. 60-62) and Smith and Gelfand (1992)), which is based on simulating from $f(\alpha)$ and accepting with probability $\pi(\alpha) = f(y|\alpha)/f(y|\alpha_{\max})$, where $\alpha_{\max} = \arg\max_\alpha f(y|\alpha)$. Again the rejection becomes worse if the $var_f\left\{\pi(\alpha)\right\}$ is high. A fundamental difference is that rejection sampling produces a random sample regardless of the size of M, while the SIR's sample are dependent and only valid as $M \to \infty$. This means that the rejection sampler is generally preferable as it is easier to determine how much simulation to perform on a particular problem, but it requires us to compute $\alpha_{\max}$ which in high dimensional problems can be computationally demanding.

If the $var_f \{\pi(\alpha)\}$ is high it is sometimes possible and worthwhile to adapt both SIR and rejection sampling to improve their behaviour by taking some of the variability of the $f(y|\alpha)$ into the proposal density. A simple example of this for SIR is where $f(\alpha)$ is Gaussian and the $\log f(y|\alpha)$ is concave. In this case we can adapt the proposal. In particular, we might Taylor expand the $\log f(y|\alpha)$ to second order to give $\log g(\alpha)$ and then sample from the Gaussian density proportional to $f(\alpha)g(\alpha)$. Then $\pi(\alpha)$ becomes proportional to $f(y|\alpha)/g(\alpha)$. This may greatly reduce the variability of the SIR method. More generally, if we can (instead of step 1. given above)

1a. evaluate $f(\alpha)$;

1b. sample from $g(\alpha|y)$;

1c. evaluate $g(\alpha|y)$;

then we could draw $\alpha \sim g(\alpha|y)$ and let $\pi(\alpha) = f(y|\alpha)f(\alpha)/g(\alpha|y)$. Of course the choice of $g(\alpha|y)$ will be critical in this context and will usually be chosen to be close to $f(\alpha|y)$ but with fatter tails.

Adapting the rejection sampling or SIR in these ways, whilst ensuring coverage in the rejection case, can be useful in a number of problems where there is substantial knowledge of the form of the likelihood. However, such adaption is not always possible and so the SIR method is vulnerable to difficult problems.

Finally, adaption comes at quite a considerable cost in the context of filtering. In particular evaluating $f(\alpha) = \sum_{j=1}^{M} f(\alpha_{t+1}|\alpha_t^j)\pi_t^j$ means we have to be able to calculate $f(\alpha_{t+1}|\alpha_t^j)$. Further, even if we can do this calculating $f(\alpha)$ means we have to evaluate M densities which can be expensive if M is large.

2.3.3 Markov chain Monte Carlo

Another alternative to SIR is the use of a Markov chain Monte Carlo (MCMC) method (see Gilks, Richardson, and Spiegelhalter (1996) for a review). In this context the MCMC method accepts a move from a current state α^i to $\alpha^{i+1} \sim f(\alpha)$ with probability $\min\{1, f(y|\alpha^{i+1})/f(y|\alpha^i)\}$, otherwise it sets $\alpha^{i+1} = \alpha^i$. This procedure produces, after it is iterated until convergence, a stationary sequence whose marginal distribution is the required posterior $f(\alpha|y)$. Again if the likelihood is highly peaked there may be a large amount of rejection which will mean the Markov chain will have a great deal of dependence. This will mean it takes a large number of iterations until it converges to its equilibrium distribution and determining the point at which it has reached the equilibrium can be difficult. This suggests adapting, when this is possible, the MCMC method to draw from $g(\alpha|y)$ and then accept these draws with probability

$$\min\left\{1, \frac{f(y|\alpha^{i+1})f(\alpha^{i+1})}{f(y|\alpha^{i})f(\alpha^{i})}\frac{g(\alpha^{i}|y)}{g(\alpha^{i+1}|y)}\right\}.$$

Again the problem with this is that having to evaluate $f(\alpha)$ can be troublesome.

2.4 Particle filter's weaknesses

The propagation of samples through the empirical prediction density (3) and then sampling from the empirical filtering density (4) provides a general, simple and powerful approach to filtering time series.

The particle filter works well for standard problems where the model is a good approximation to the data and the conditional densities $f(y_t|\alpha_t)$ are reasonably flat in α_t, but when we have very severe outliers there are problems. In this context the fact that the particle filter is very difficult to adapt is an enormous weakness of the method.

To illustrate the potential problem we assume the observations arise from an autoregression observed with noise

$$\begin{array}{rclcl} y_t & = & \alpha_t + \varepsilon_t, & \varepsilon_t & \sim & NID(0,1) \\ \alpha_{t+1} & = & \phi\alpha_t + \eta_t, & \eta_t & \sim & NID(0,\sigma^2), \end{array} \qquad (5)$$

where $\phi = 0.9, \sigma^2 = 0.01$ and ε_t and η_t are independent white noise processes. The model is initialised by α_t's stationary prior. Set $n = 6$ and let the first five observations arise from the above autoregression observed with noise model (5) and then assign to the sixth observation the value 20. We observe the series

$$y = (-0.65201, -0.34482, -0.67626, 1.1423, 0.72085, 20.000)'.$$

The last observation is around twenty standard deviations away from that predicted by the model. We run a SIR (recall section 2.3.1) based particle filter on this problem using a variety of values of M and R, averaging over 125 replications and always taking $\pi^j = 1/M$. Table 1 displays the average simulation-based estimate of $E\left(\alpha_6|Y_6\right)$ and the true values computed using the Kalman filter. Hence the Table's focus is on the bias of the simulation procedure. The Table shows that the SIR based particle filter grossly underestimates the values of the states even when M and R are very large.

The particle filter based on SIR has two basic weaknesses. The first is well known and repeats the discussion of SIR given in section 2.3.1. When there is an outlier, the weights π_j will be very unevenly distributed and so it will require an extremely large value of R for the draws to be close to samples from the empirical filtering density. This is of particular concern if the measurement

		Particle			Auxiliary particle		
R	TRUTH	$M = 1,000$	10,000	50,000	1,000	10,000	50,000
50	.90743	.43523	.42183	.43504	.52630	.54516	.54920
250	.90743	.55188	.55829	.55579	.65437	.65274	.66682
2,000	.90743	.65164	.65384	.66269	.71899	.77279	.76714
10,000	.90743	.71235	.73396	.73382	.72653	.79637	.82569
25,000	.90743	.73433	.77206	.76071	.73043	.81076	.83324
100,000	.90743	.73083	.79238	.81929	.74424	.81975	.85721

Table 1: *SIR based particle and SIR based auxiliary particle algorithms. Recorded are the means of 125 independent replications of the SIR based particle and auxiliary particle filters run on the fixed y using a variety of values of M and R. Thus the Table demonstrates the bias of the methods.*

density $f(y_{t+1}|\alpha_{t+1})$ is highly sensitive to α_{t+1}. Notice this is not a problem of having too small a value of M. That parameter controls the accuracy of (3). Instead, the difficulty is, given that degree of accuracy, how to efficiently sample from (4)? Can SIR be improved upon in this problem? This paper will show that we can answer this question positively, with no added assumptions and little extra computational expense.

The second weakness holds in general for particle filters which have their π^j equal and who update the states one period at a time. As $R \to \infty$, so the weighted samples can be used to arbitrarily well approximate (4). However, the tails of (3) usually only poorly approximate the true tails of $\alpha_{t+1}|Y_t$ due to the use of the mixture approximation. As a result (4) can only ever poorly approximate the true $f(\alpha_{t+1}|Y_{t+1})$ when there is an outlier. Hence the second question is how do we improve the empirical prediction density's behaviour in the tails? Section 4 of this paper partially deals with this much harder problem.

3 AUXILIARY VARIABLE

3.1 The basics

A fundamental problem with existing particle filters is that its mixture structure means that it is difficult to improve the simulation performance of the SIR, rejection or MCMC sampling methods due to the expense of evaluating the empirical prediction density (3). We call the generic process of changing the sampling mechanism *adaption*.

The lack of adaptability makes particle filters less attractive for difficult problems where their naive application is less effective. Here we argue that many of these problems are reduced when we perform particle filtering in a higher dimension.

Our task will be to sample from the joint density $f(\alpha_{t+1}, k|Y_{t+1})$, where k is an index on the mixture in (3). Define

$$f(\alpha_{t+1}, k|Y_{t+1}) \propto f(y_{t+1}|\alpha_{t+1})f(\alpha_{t+1}|\alpha_t^k)\pi^k, \qquad k = 1, ..., M, \tag{6}$$

and we draw from this joint density and then discard the index we produce a sample from the empirical filtering density (4) as required. We call k an auxiliary variable as it is present simply to aid the task of the simulation. Generic particle filters of this type will be labelled auxiliary particle filters.

We can sample from $f(\alpha_{t+1}, k|Y_{t+1})$ using SIR, rejection or MCMC. We first of all deal with a very basic SIR. We approximate (6) by

$$g(\alpha_{t+1}, k|Y_{t+1}) \propto f(y_{t+1}|\mu_{t+1}^k)f(\alpha_{t+1}|\alpha_t^k)\pi^k, \qquad k = 1, ..., M,$$

where μ_{t+1}^k is the mean, the mode, a draw, or some other likely value associated with the density of $\alpha_{t+1}|\alpha_t^k$. The form of the approximating density is designed so that

$$g(k|Y_{t+1}) \propto \pi^k \int f(y_{t+1}|\mu_{t+1}^k)dF(\alpha_{t+1}|\alpha_t^k) = \pi^k f(y_{t+1}|\mu_{t+1}^k).$$

Thus we can sample from $g(\alpha_{t+1}, k|Y_{t+1})$ by simulating the index with probability λ_k, which is proportional to $g(k|Y_{t+1})$, and then sampling from the transition density given the mixture $f(\alpha_{t+1}|\alpha_t^k)$. We call the λ_k the first stage weights.

The implication is that we will simulate from particles which are associated with large predictive likelihoods. Having sampled the joint density of $g(\alpha_{t+1}, k|Y_{t+1})$ R times we perform a reweighting, putting on the draw (α_{t+1}^j, k^j) the weights proportional to the so-called second stage weights

$$w_j = \frac{f(y_{t+1}|\alpha_{t+1}^j)}{f(y_{t+1}|\mu_{t+1}^{k^j})}, \qquad \pi_j = \frac{w_j}{\sum_{i=1}^R w_i}, \qquad j = 1, ..., R.$$

The hope is that these second stage weights are much less variable than for the original SIR method. We might resample from this discrete distribution to produce a sample of size M.

This auxiliary variable based SIR requires only the ability to propagate and evaluate the likelihood, just as the original SIR suggestion of Gordon, Salmond, and Smith (1993). In practice, it runs slightly less quickly that the Gordon, Salmond, and Smith (1993) suggestion as we need to evaluate $g(k|Y_{t+1})$ and to perform two weighted bootstraps rather than one weighted and one unweighted bootstrap. However, the gains in sampling will usually dominate these small effects.

By making proposals which have high conditional likelihoods we reduce the costs of sampling many times from particles which have very low likelihoods and so will not be resampled at the second stage of the process. This improves the statistical efficiency of the sampling procedure and means that we can reduce the value of R substantially.

To measure the statistical efficiency of these procedures we argued in the first section that we could look at minimizing $E\left\{\pi(\alpha)^2\right\}$. Here we compare a standard SIR with a SIR based on our auxiliary variable. For simplicity we set $\pi^k = 1/M$ in both cases.

For a standard SIR based particle filter, for large M,

$$E\left\{\pi(\alpha)^2\right\} = \frac{\frac{1}{M}\sum_{k=1}^{M}\int f(y_{t+1}|\alpha_{t+1})^2 dF(\alpha_{t+1}|\alpha_t^k)}{\left\{\frac{1}{M}\sum_{k=1}^{M}\int f(y_{t+1}|\alpha_{t+1}) dF(\alpha_{t+1}|\alpha_t^k)\right\}^2} = \frac{M\sum_{k=1}^{M}\lambda_k^2 f_k}{\left(\sum_{k=1}^{M}\lambda_k f_k^*\right)^2},$$

where

$$f_k = \int\left\{\frac{f(y_{t+1}|\alpha_{t+1})}{f(y_{t+1}|\mu_{t+1}^k)}\right\}^2 dF(\alpha_{t+1}|\alpha_t^k) \quad \text{and} \quad f_k^* = \int\left\{\frac{f(y_{t+1}|\alpha_{t+1})}{f(y_{t+1}|\mu_{t+1}^k)}\right\} dF(\alpha_{t+1}|\alpha_t^k).$$

The same calculation for a SIR based auxiliary variable particle filter gives

$$E\left\{\pi_\alpha(\alpha)^2\right\} = \frac{\sum_{k=1}^{M}\lambda_k f_k}{\left(\sum_{k=1}^{M}\lambda_k f_k^*\right)^2},$$

which shows an efficiency gain if

$$\sum_{k=1}^{M}\lambda_k f_k < M\sum_{k=1}^{M}\lambda_k^2 f_k.$$

If f_k does not vary over k then the auxiliary variable particle filter will be more efficient as $\sum_{k=1}^{M}\lambda_k\frac{1}{M} = \frac{1}{M} \leq \sum_{k=1}^{M}\lambda_k^2$. More likely is that f_k will depend on k but only mildly as $f(\alpha_{t+1}|\alpha_t^k)$ will be typically quite tightly peaked (much more tightly peaked than $f(\alpha_{t+1}|Y_t)$) compared to the conditional likelihood.

To assess the effectiveness of the SIR based auxiliary particle filter, the right hand sides of Table 1 replicate the earlier SIR studies on simulated Gaussian data using the auxiliary algorithm. Table 1, which reports the results of an experiment in which there is a very extreme outlier, suggests a very significant improvement due to the use of the auxiliary particle filter. In particular our simulations suggest that if we keep M fixed, that for the same value of R, auxiliary algorithm is an order of magnitude more efficient than SIR for outlier problems. As a result the auxiliary algorithm reduces the bias by an amount which would take many times the computational effort for the SIR to improve by the same degree.

Rejection sampling can also be used for the auxiliary particle filter so long as $\alpha_{t+1,\max} = \arg\max_{\alpha_{t+1}} f(y_{t+1}|\alpha_{t+1})$ can be found. The task is to draw from

$$f(\alpha_{t+1}, k|Y_{t+1}) \propto f(y_{t+1}|\alpha_{t+1})f(\alpha_{t+1}|\alpha_t^k)\pi^k \leq f(y_{t+1}|\alpha_{t+1,\max})f(\alpha_{t+1}|\alpha_t^k)\pi^k,$$

and so we can sample from the density by drawing k with probability π^k and then accepting $\alpha_{t+1} \sim f(\alpha_{t+1}|\alpha_t^k)$ with probability $f(y_{t+1}|\alpha_{t+1})/f(y_{t+1}|\alpha_{t+1,\max})$. This is likely to perform quite poorly for some problems as this ratio can be very small.

The MCMC variate of the auxiliary particle filter designs a Metropolis chain with an equilibrium distribution of the form $f(\alpha_{t+1},k|Y_{t+1})$. If we make proposals from $\alpha_{t+1}^{(i+1)},k^{(i+1)} \sim g(\alpha_{t+1},k|Y_{t+1})$, where $g(\alpha_{t+1},k|Y_{t+1})$ is some arbitrary density, then these moves are accepted with probability

$$\min\left\{1,\frac{f(y_{t+1}|\alpha_{t+1}^{(i+1)})f(\alpha_{t+1}^{(i+1)}|\alpha_t^{k^{(i+1)}})}{f(y_{t+1}|\alpha_{t+1}^{(i)})f(\alpha_{t+1}^{(i)}|\alpha_t^{k^{(i)}})}\frac{g(\alpha_{t+1}^{(i)},k^{(i)}|Y_{t+1})}{g(\alpha_{t+1}^{(i+1)},k^{(i+1)}|Y_{t+1})}\right\}.$$

In the special case where $g(\alpha_{t+1},k|Y_{t+1}) \propto g(k|Y_{t+1})f(\alpha_{t+1}|\alpha_t^k)$, this simplifies to

$$\min\left\{1,\frac{f(y_{t+1}|\alpha_{t+1}^{(i+1)})}{f(y_{t+1}|\mu_{t+1}^{k^{(i+1)}})}\frac{f(y_{t+1}|\mu_{t+1}^{k^{(i)}})}{f(y_{t+1}|\alpha_{t+1}^{(i)})}\right\},$$

which is extremely convenient as it involves just the evaluation of the measurement density. Hence this approach is particularly useful when it is not possible to evaluate the transition density.

3.2 Adaption

In this subsection we will show that these three sampling procedures are now easy to adapt.

3.2.1 Non-linear Gaussian model

In the Gaussian measurement case, the absorption of the measurement density into the transition equation is particularly convenient. Consider a non-linear transition density with $\alpha_{t+1}|\alpha_t \sim N\left\{\mu\left(\alpha_t\right),\sigma^2\left(\alpha_t\right)\right\}$ and $y_{t+1}|\alpha_{t+1} \sim N(\alpha_{t+1},1)$. Then

$$f(\alpha_{t+1},k|Y_{t+1}) \propto f(y_{t+1}|\alpha_{t+1})f(\alpha_{t+1}|\alpha_t^k) = g_k(y_{t+1})f(\alpha_{t+1}|\alpha_t^k,y_{t+1}),$$

where

$$f(\alpha_{t+1}|\alpha_t^k,y_{t+1}) = N(\mu_{p,k},\sigma_{pk}^2), \qquad \mu_{p,k} = \sigma_{p,k}^2\left\{\frac{\mu\left(\alpha_t^k\right)}{\sigma^2\left(\alpha_t\right)}+y_{t+1}\right\}, \qquad \sigma_{p,k}^{-2} = 1+\sigma^{-2}\left(\alpha_t^k\right).$$

This implies that the first stage weights are

$$g_k(y_{t+1}) \propto \exp\left\{\frac{\mu_{p,k}^2}{2\sigma_{p,k}^2}-\frac{\mu\left(\alpha_t^k\right)^2}{2\sigma^2\left(\alpha_t\right)}\right\}.$$

The Gaussian measurement density implies the second stage weights are all equal and so a weighted bootstrap at this stage is not required as all the draws would have equal weight. Consequently we say that the auxiliary particle filter has been fully adapted to the problem and it makes sense to take $R = M$. Of course in many problems full adaption is not realistic, but some form of adaption can be used and will improve the efficiency and reliability of the method. In many cases it is unnecessary, however it can be helpful when we come across very difficult problems.

Example: ARCH with error Consider the simplest Gaussian ARCH model (see, for example, Bollerslev, Engle, and Nelson (1994) for a review) observed with independent Gaussian error. So we have

$$y_t|\alpha_t \sim N(\alpha_t, \sigma^2), \qquad \alpha_{t+1}|\alpha_t \sim N(0, \beta_0 + \beta_1 \alpha_t^2).$$

This model is exactly adaptable. It has received a great deal of attention in the econometric literature as it has some attractive multivariate generalizations: see the work by Diebold and Nerlove (1989), Harvey, Ruiz, and Sentana (1992) and King, Sentana, and Wadhwani (1994). As far as we know no likelihood methods exist in the literature for the analysis of this type of model (and its various generalizations) although a number of very good approximations have been suggested.

Extended example: factor GARCH A more difficult example of this class of problem, following the work of King, Sentana, and Wadhwani (1994), is the bivariate factor GARCH model where

$$y_{t+1} = \gamma \varepsilon_{1t+1}\sigma_{1t+1} + \begin{pmatrix} \varepsilon_{2t+1}\sigma_{2t+1} \\ \varepsilon_{3t+1}\sigma_{3t+1} \end{pmatrix}, \quad \varepsilon_{it+1} \underset{iid}{\sim} NID(0,1),$$

and σ^2_{it+1} follows a GARCH(1,1) type volatility process $\sigma^2_{it+1} = \beta_{i0} + \beta_{i1}\sigma^2_{it} + \beta_{i2}\varepsilon_{it}$. If we write $\alpha_{t+1} = (\sigma^2_{1t+1}, \varepsilon_{1t+1}, ..., \sigma^2_{3t+1}, \varepsilon_{3t+1})'$, then

$$\begin{aligned} \widehat{f}(\alpha_{t+1}, k|Y_{t+1}) &\propto I\left\{ y_{t+1} = \gamma\varepsilon_{1t+1}\sigma_{1t+1} + \begin{pmatrix} \varepsilon_{2t+1}\sigma_{2t+1} \\ \varepsilon_{3t+1}\sigma_{3t+1} \end{pmatrix} \right\} f(\varepsilon_{1t+1}, ..., \varepsilon_{3t+1}|\alpha_t^k) \\ &= c(k) f(\varepsilon_{1t+1}, ..., \varepsilon_{3t+1}|\alpha_t^k, y_{t+1}). \end{aligned}$$

Here

$$c(k) \propto f(y_{1t+1} = \gamma\varepsilon_{1t+1}\sigma_{1t+1} + \varepsilon_{2t+1}\sigma_{2t+1}|\alpha_t^k) f(y_{2t+1} = \gamma\varepsilon_{1t+1}\sigma_{1t+1} + \varepsilon_{3t+1}\sigma_{3t+1}|\alpha_t^k, y_{1t+1}).$$

Thus we can draw k with probability proportional to $c(k)$ and then sample $\varepsilon_{1t+1}, ..., \varepsilon_{3t+1}$ from a constrained multivariate normal distribution. Hence in this example adaption is again exact. This argument generalizes to any factor GARCH model.

3.2.2 Log-concave measurement densities

Suppose that $f(\alpha_{t+1}|\alpha_t^k)$ is Gaussian, then we might extend the above argument by Taylor expanding $\log f(y_{t+1}|\alpha_{t+1})$ to a second order term, again around μ_{t+1}^k, to give the approximation

$$\begin{aligned} \log g(y_{t+1}|\alpha_{t+1}, \mu_{t+1}^k) &= \log f(y_{t+1}|\mu_{t+1}^k) + \left(\alpha_{t+1} - \mu_{t+1}^k\right) \frac{\partial \log f(y_{t+1}|\mu_{t+1}^k)}{\partial \alpha_{t+1}} \\ &+ \frac{1}{2}\left(\alpha_{t+1} - \mu_{t+1}^k\right)' \frac{\partial^2 \log f(y_{t+1}|\mu_{t+1}^k)}{\partial \alpha_{t+1} \partial \alpha_{t+1}'} \left(\alpha_{t+1} - \mu_{t+1}^k\right), \end{aligned}$$

then

$$g(\alpha_{t+1}, k|Y_{t+1}) \propto g(y_{t+1}|\alpha_{t+1}; \mu_{t+1}^k) f(\alpha_{t+1}|\alpha_t^k).$$

Rearranging, we can express this as

$$g(\alpha_{t+1}, k|Y_{t+1}) \propto g(y_{t+1}|\mu_{t+1}^k) g(\alpha_{t+1}|\alpha_t^k, y_{t+1}; \mu_{t+1}^k),$$

which means we could simulate the index with probability proportional to $g(y_{t+1}|\mu_{t+1}^k)$ and then draw from $g(\alpha_{t+1}|\alpha_t^k, y_{t+1}, \mu_{t+1}^k)$. The resulting reweighted sample's second stage weights are proportional to the hopefully fairly even weights

$$w_j = \frac{f(y_{t+1}|\alpha_{t+1}^j) f(\alpha_{t+1}|\alpha_t^{k_j})}{g(y_{t+1}|\mu_{t+1}^{k_j}) g(\alpha_{t+1}^j|\alpha_t^{k_j}, y_{t+1}, \mu_{t+1}^k)} = \frac{f(y_{t+1}|\alpha_{t+1}^j)}{g(y_{t+1}|\alpha_{t+1}^j; \mu_{t+1}^{k_j})}, \qquad \pi_j = \frac{w_j}{\sum_{i=1}^R w_i}, \qquad j = 1, ..., R.$$

Thus, we can exploit the special structure of the model, if available, to improve upon the auxiliary particle filter.

3.2.3 Stochastic volatility and rejection sampling

The same argument carries over when we use a first order Taylor expansion to construct $g(y_{t+1}|\alpha_{t+1}, \mu_{t+1}^k)$, but in this case we know that $g(y_{t+1}|\alpha_{t+1}, \mu_{t+1}^k) \geq f(y_{t+1}|\alpha_{t+1})$ for any value of μ_{t+1}^k due to the assumed log-concavity of the measurement density. Thus

$$f(\alpha_{t+1}, k|Y_{t+1}) \leq g(\alpha_{t+1}, k|Y_{t+1}) \propto g(y_{t+1}|\alpha_{t+1}; \mu_{t+1}^k) f(\alpha_{t+1}|\alpha_t^k) = g(y_{t+1}|\mu_{t+1}^k) g(\alpha_{t+1}|\alpha_t^k, y_{t+1}; \mu_{t+1}^k).$$

Thus we can perform rejection sampling from $f(\alpha_{t+1}, k|Y_{t+1})$ by simply sampling k with probability proportional to $g(y_{t+1}|\mu_{t+1}^k)$ and then drawing α_{t+1} from $g(\alpha_{t+1}|\alpha_t^k, y_{t+1}; \mu_{t+1}^k)$. This pair is then accepted with probability $f(y_{t+1}|\alpha_{t+1})/g(y_{t+1}|\alpha_{t+1}; \mu_{t+1}^k)$.

This argument applies to the non-linear time series model of evolving scale: the stochastic volatility (SV) model

$$y_t = \epsilon_t \beta \exp(\alpha_t/2), \quad \alpha_{t+1} = \phi \alpha_t + \eta_t, \tag{7}$$

where ϵ_t and η_t are independent Gaussian processes with variances of 1 and σ^2 respectively. Here β has the interpretation as the modal volatility, ϕ the persistence in the volatility shocks and σ_η^2 is the volatility of the volatility. This model has attracted much recent attention in the econometrics literature as a way of generalizing the Black-Scholes option pricing formula to allow volatility clustering in asset returns; see, for instance, Hull and White (1987), Harvey, Ruiz, and Shephard (1994) and Jacquier, Polson, and Rossi (1994). MCMC methods have been used on this model by, for instance, Jacquier, Polson, and Rossi (1994), Shephard and Pitt (1997) and Kim, Shephard, and Chib (1998).

For this model $\log f(y_{t+1}|\alpha_{t+1})$ is concave in α_{t+1} so that

$$\log g(y_{t+1}|\alpha_{t+1};\mu_{t+1}^k) = -\frac{1}{2}\alpha_{t+1} - \frac{y_t^2}{2\beta^2}\exp(-\mu_{t+1}^k)\left\{1-\left(\alpha_{t+1}-\mu_{t+1}^k\right)\right\}.$$

The implication is that

$$g(\alpha_{t+1}|\alpha_t^k, y_{t+1};\mu_{t+1}^k) = N\left[\mu_{t+1}^k + \frac{\sigma^2}{2}\left\{\frac{y_t^2}{\beta^2}\exp(-\mu_{t+1}^k)-1\right\},\sigma^2\right] = N(\mu_{t+1}^{p,k},\sigma^2).$$

Likewise

$$g(y_{t+1}|\mu_{t+1}^k) = \exp\left\{\frac{1}{2\sigma^2}\left(\mu_{t+1}^{p,k2}-\mu_{t+1}^{k2}\right)\right\}\exp\left\{-\frac{y_t^2}{2\beta^2}\exp(-\mu_{t+1}^k)\left(1-\mu_{t+1}^k\right)\right\}.$$

Finally the log-probability of acceptance is

$$-\frac{y_t^2}{2\beta^2}\left[\exp(-\alpha_{t+1}) - \exp(-\mu_{t+1}^k)\left\{1-\left(\alpha_{t+1}-\mu_{t+1}^k\right)\right\}\right].$$

Notice that as σ^2 falls to zero so the acceptance probability goes to one.

If an adapted SIR method had been applied here exactly the same calculations would have been applied except that no suggestion would be rejected and instead the second stage bootstrap weights would be the same as the acceptance rates.

3.2.4 Limited dependent processes

A less trivial example of exact adaption is a special cases of limited dependent processes, where the observations are deterministic functions of the states. A simple example of this is a Probit time series where $y_t = I(\alpha_t > 0)$, where α_t is Gaussian and univariate and $I(.)$ denotes an indicator function. Then if $y_{t+1} = 1$ we have, exactly,

$$\Pr(\alpha_{t+1}, k|Y_{t+1}) \propto w^k \Pr\left(\alpha_{t+1}|\alpha_t^k, \alpha_{t+1} > 0\right), \quad w^k = \Pr\left(\alpha_{t+1} > 0|\alpha_t^k\right).$$

Hence we choose k with probability proportional to w^k and then draw from a truncated distribution conditional on k. If y_{t+1} is negative then the weights w^k would be $\Pr\left(\alpha_{t+1} < 0|\alpha_t^k\right)$ while the truncated draw would be from $\Pr\left(\alpha_{t+1}|\alpha_t^k, \alpha_{t+1} < 0\right)$. This style of argument carries over to ordered Probit and censored models where we observe, for example, $\min(0, \alpha_t)$.

Adaption can be very important in these types of models for naively implemented particle and auxiliary variable filters are generally vulnerable to tightly peaked measurement densities. In the censored model, where $y_{t+1} = \min(0, \alpha_{t+1})$, the measurement density is degenerate when $y_{t+1} > 0$ and so the particle filter will degenerate to give all of its mass on the simulation which is closest (but because they are simulated from $\Pr\left(\alpha_{t+1}|\alpha_t^k\right)$ not equal) to y_{t+1}. Adaption overcomes this problem instantly.

3.2.5 Disequilibrium models

Adaption is also essential for the following problem. Suppose $\alpha_{t+1}|\alpha_t$ is Gaussian, α_{t+1} is bivariate and that we observe $y_{t+1} = \min(\alpha_{t+1})$. Such models are called disequilibrium models in economics (recent work in this area includes Laroque and Salanie (1993) and Lee (1995)). Then

$$\Pr(\alpha_{t+1}, k|Y_{t+1}) \propto \Pr(y_{t+1}|\alpha_{t+1}) \Pr(\alpha_{t+1}|\alpha_t^k).$$

Then we have that w^k should be proportional to the probability of $\alpha_{t+1}|\alpha_t^k$ having its minimum exactly at y_{t+1}. This can be shown to be exactly

$$w^k = f_{\alpha^k_{1,t+1|\alpha_t}}(y_{t+1})\left\{1 - \Pr\alpha_{2,t+1|\alpha_t}(y_{t+1})\right\} + f_{\alpha_{2,t+1|\alpha_t}}(y_{t+1})\left\{1 - \Pr\alpha_{1,t+1|\alpha_t}(y_{t+1})\right\},$$

while having selected k we sample

$$\alpha_{1,t+1} = y_{t+1} \quad \text{with probability} \quad \lambda_{t+1} = \frac{f_{\alpha_{1,t+1|\alpha_t^k}}(y_{t+1})\left\{1 - \Pr\alpha_{2,t+1|\alpha_t^k}(y_{t+1})\right\}}{w^k},$$

and then from

$$\alpha_{2,t+1}|\alpha_{1,t+1} = y_{t+1}, \alpha_t^k, \alpha_{2,t+1} > y_{t+1}.$$

Likewise $\alpha_{2,t+1} = y_{t+1}$ with probability $1 - \lambda_{t+1}$.

3.2.6 Mixtures of normals

Suppose $f(\alpha_{t+1}|\alpha_t)$ is Gaussian, but the measurement density is a discrete mixture of normals $\sum_{j=1}^{P} \lambda_j f_j(y_{t+1}|\alpha_{t+1})$. Then we can perfectly sample from $f(\alpha_{t+1}, k|Y_{t+1})$ by working with

$$f(\alpha_{t+1}, k, j|Y_{t+1}) \propto \lambda_j f_j(y_{t+1}|\alpha_{t+1}) f\left(\alpha_{t+1}|\alpha_t^k\right) = w_{j,k} f_j(\alpha_{t+1}|\alpha_t^k, y_{t+1}).$$

Then we sample from $f(\alpha_{t+1}, k, j|Y_{t+1})$ by selecting the index k, j with probability proportional to $w_{j,k}$ and then drawing from $f_j(\alpha_{t+1}|\alpha_t^k, y_{t+1})$. The disadvantage of this approach is that the complete enumeration and storage of $w_{j,k}$ involves PM calculations. This approach can be trivially extended to cover the case where $f(\alpha_{t+1}|\alpha_t)$ is a mixture of normals. MCMC smoothing methods for state space models with mixtures have been studied by, for example, Carter and Kohn (1994), Carter and Kohn (1996) and Shephard (1994). The special case of an autoregression with additive and innovative outliers, as well as mean shifts, can also be put in this framework. Again there is an extensive MCMC literature on this topic starting with Albert and Chib (1993) and then further developed and popularised by McCulloch and Tsay (1993) and McCulloch and Tsay (1994). Filtering via MCMC methods is developed in Gerlach, Carter, and Kohn (1996), although their methods are $O(n^2)$.

3.3 Existing literature

The above auxiliary variable particle filter seems to be a new idea. However, there are some similarities with a recent paper by Berzuini, Best, Gilks, and Larizza (1997) (BBGL). In Section 4 of BBGL, the method of Gordon, Salmond, and Smith (1993) reappears and is then studied in some detail before being rejected as being inefficient. Then in Section 5 BBGL do not propose sampling the index k with equal weight (as in SIR), or sampling with weights proportional to $f(y_{t+1}|\mu_{t+1}^k)$ (as in SIR in the context of a auxiliary particle filter). Instead they use a uniform distribution as a proposal density for a MCMC algorithm which updates k given the current value of the state α_{t+1}. Their MCMC algorithm is completed by using a MCMC suggestion for the state α_{t+1} given the mixture. This delivers a simulation whose equilibrium distribution, $f(\alpha_{t+1}, k|Y_t)$, is the same as advocated above.

We believe our approach is superior for a number of reasons. Making uniform proposals for moving k means that BBGL make enormous numbers of draws which are irrelevant in cases where the measurement density is highly peaked. In particular, it can take a long time until a single sensible value of k is chosen. Our method is much simpler and immediately achieves the desired goal of sampling the important indices. Further, the index k and state α_{t+1} may be quite highly correlated given past information. Hence the MCMC algorithm may converge very slowly. Indeed, in the special case of a model with no measurement error this sampler will never converge. We avoid this by integrating out the index. In most cases we avoid MCMC altogether and just use SIR on the auxiliary particle filter, which is helpful as this is then simpler to monitor.

3.4 Example: a time series of angles

3.4.1 The model

In this section we will compare the performance of the particle and auxiliary particle filter methods for an angular time series model; the bearings-only model. Typically, the model is used for the problem of tracking a ship by using only angular information, hence the term "bearings-only". We are provided with no information about range.

We consider the simple scenario described by Gordon, Salmond, and Smith (1993). The observer is considered stationary at the origin of the $x-z$ plane. A simple model for the ship is obtained by assuming that the ship gradually accelerates or decelerates randomly over time. We use the following discretisation of this system provided by Carpenter, Clifford, and Fearnhead

(1996), where $\alpha_t = (x_t, vx_t, z_t, vz_t)'$,

$$\alpha_{t+1} = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \alpha_t \quad + \sigma_\eta \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix} u_t, \quad u_t \sim \mathsf{NID}(0, \mathsf{I}). \tag{8}$$

In obvious notation x_t, z_t represent the ship's horizontal and vertical position at time t and vx_t, vz_t represent the corresponding velocities. The state evolution is thus a VAR(1) of the form $\alpha_{t+1} = T\alpha_t + Hu_t$. The model indicates that the source of state evolution error is due to the accelerations being white noise. The velocities therefore follow a random walk. The initial state describes the ship's starting positions and velocities $\alpha_1 \sim \mathsf{NID}(a_1, P_1)$. This prior together with the state evolution of (8) describes the overall prior for the states.

The measurements consist of the true angle of the ship corrupted by a wrapped Cauchy error term. Hence we have,

$$y_t | \mu_t \sim \mathsf{WC}(\mu_t, \rho), \quad \mu_t = \tan^{-1}(z_t/x_t). \tag{9}$$

The density for the two parameter wrapped Cauchy distribution; $\mathsf{WC}(\mu, \rho)$, see Fisher (1993, p. 46), is of the following form,

$$f(y_t|\mu) = \frac{1}{2\pi} \frac{1-\rho^2}{1+\rho^2 - 2\rho\cos(y_t - \mu)}, \quad 0 \leq y_t < 2\pi, \quad 0 \leq \rho \leq 1, \tag{10}$$

where μ is the mean direction and ρ is termed the mean resultant length. We choose the wrapped Cauchy as this allows very heavy tails which models the aberrant measurements which occasionally arise from radar systems.

3.4.2 Particle filters

The proposal density for the auxiliary particle filter (projected just one step ahead) is exactly correct for particular expansion values in the mixture as

$$f(\alpha_{t+1}, k|Y_{t+1}) \propto f(y_{t+1}|\alpha_{t+1}) f(\alpha_{t+1}|k) = f(y_{t+1}|\widehat{\alpha}_{t+1}^k) f(\alpha_{t+1}|k), \tag{11}$$

where $\widehat{\alpha}_{t+1}^k = T\widehat{\alpha}_t^k$, $\widehat{\alpha}_t^k$ representing the component of the mixture k at the time t. Hence the auxiliary approximation to the joint empirical filtering density is *exact* for the bearings-only model. This equality arises because the sufficient state elements for the measurement density are the position elements $\alpha_{t+1,1}, \alpha_{t+1,3}$. However, since these elements are projected without noise through the state equation then $(\alpha_{t+1,1}, \alpha_{t+1,3})' = (\widehat{\alpha}_{t+1,1}^k, \widehat{\alpha}_{t+1,3}^k)'$.

For the bearings-only model the equality is extremely useful. Since we can exactly sample from the empirical filtering density we can dispense with the reweighting procedure. Hence we have only one weighted bootstrap, of size M, to perform at each time step. By contrast it can immediately be seen that for fixed M the particle filter approach requires $R \to \infty$ in order to obtain the same accuracy.

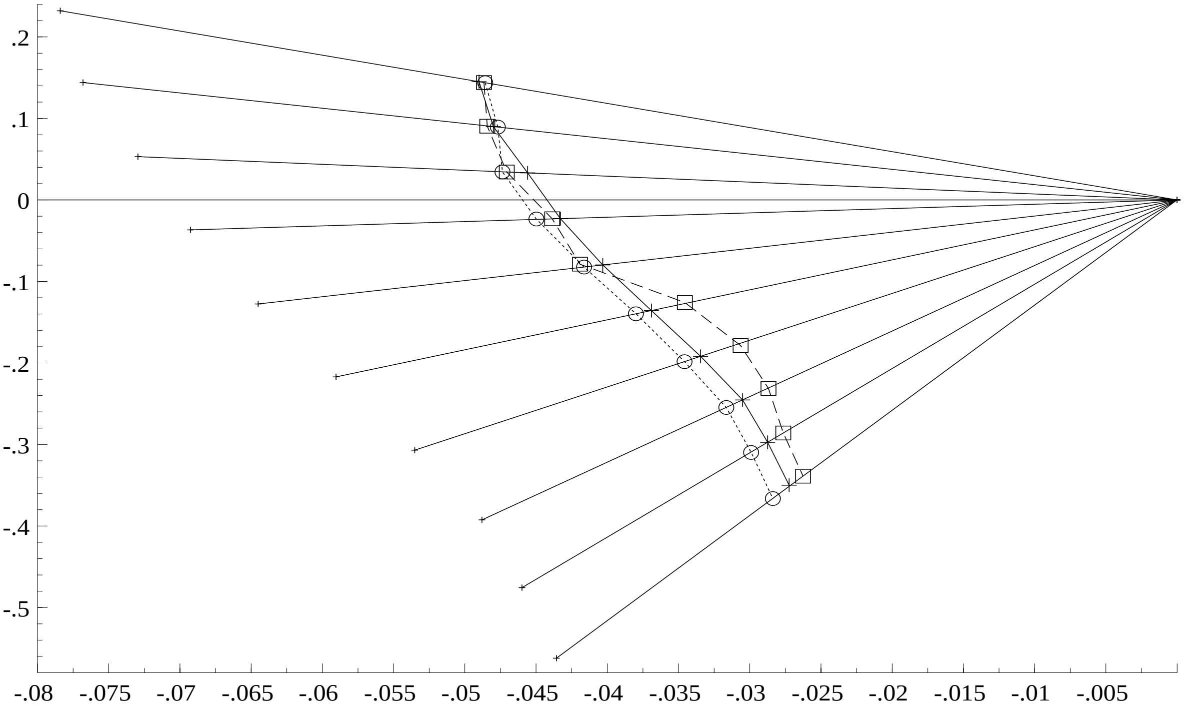

Figure 1: *Plot of the angular measurements from origin, the true trajectory (solid line, crosses), the particle filtered mean trajectory (dashed line, boxes) and the auxiliary particle mean trajectory (dotted line, circles). Ship moving South-East.* $T = 10$*,* $M = 300$*,* $R = 500$*.*

3.4.3 The simulated scenario

In order to assess the relative efficiency of the particle filter and auxiliary method we have closely followed the setup described by Gordon, Salmond, and Smith (1993). They consider $\sigma_\eta = 0.001$ and $\sigma_\varepsilon = 0.005$, where $z_t|\mu_t \sim NID(\mu_t, \sigma_\varepsilon^2)$. We choose $\rho = 1 - \sigma_\varepsilon^2$ (yielding the same circular dispersion) for our wrapped Cauchy density. The actual initial starting vector of this is taken to be $\alpha_1 = (-0.05, 0.001, 0.2, -0.055)'$. By contrast with Gordon, Salmond, and Smith (1993), we wish to have an extremely accurate and tight prior for the initial state. This is because we want the variance of quantities arising from the filtered posterior density to be small enabling reasonably conclusive evidence to be formulated about the relative efficiency of the auxiliary method to the standard method. We therefore take $a_1 = \alpha_1$ and have a diagonal initial variance P_1 with the elements $0.001 \times (0.5^2, 0.005^2, 0.3^2, 0.01^2)$ on the diagonal.

Figure 1 illustrates a realization of the model for the above scenario with $T = 10$. The ship is moving in a South-Easterly direction over time. The trajectories given by the posterior filtered means from the particle method ($M = 300$, $R = 500$) and the auxiliary method ($M = 300$) are both fairly close to the true path despite the small amount of simulation used.

3.4.4 Monte Carlo comparison

The two methods are now compared using a Monte Carlo study of the above scenario with $T = 20$. The number of scenario replications, REP say, is set to 20. The "true" filtered mean is calculated for each replication by using the auxiliary method with $M = 80,000$. Within each replication the mean squared error for the particle method for each component of the state over time is evaluated by running the method, with a different random number seed, S times and recording the average of the resulting squared difference between the resulting particle' estimated mean and the "true" filtered mean. Hence for replication i, state component j, at time t we calculate

$$MSE^P_{i,j,t} = \frac{1}{S}\sum_{s=1}^{S}(\overline{\alpha}^i_{t,j,s} - \widetilde{\alpha}^i_{t,j})^2,$$

where $\overline{\alpha}^i_{t,j,s}$ is the particle mean for replication i, state component j, at time t, for simulation s and $\widetilde{\alpha}^i_{t,j}$ is the "true" filtered mean replication i, state component j, at time t. The log mean squared error for component j at time t is obtained as

$$LMSE^P_{j,t} = \log \frac{1}{REP}\sum_{i=1}^{REP} MSE^P_{i,j,t}.$$

The same operation is performed for the auxiliary method to deliver the corresponding quantity $LMSE^{AM}_{j,t}$. For this study we use $M = 2000$, $REP = 20$ and $S = 20$. Figure 2 shows the relative performance of the two methods for each component of the state vector over time. For each component j, the quantity $LMSE^{AM}_{j,t} - LMSE^P_{j,t}$ is plotted against time. The four plots in each box indicate the different values of R, for the particle method, which are $M/4, M, M \times 4$, and $M \times 16$. Values close to 0 indicate that the two methods are broadly equivalent in performance whilst negative values indicate that the auxiliary method performs better than the standard particle filter.

The graphs give the expected result with the auxiliary particle filter typically being more precise, but with the difference between the two methods falling as R increases. Note the computational burden for the auxiliary particle filter is proportional to M, while for the particle filter is roughly proportional to $M + R$.

4 GENERALIZATIONS

4.1 Fixed lag filtering

The auxiliary particle filter method can also be used when we update the estimates of the states not by a single observation but by a block of observations. Again suppose that we approximate the density of $\alpha_t|Y_t = (y_1, ..., y_t)'$ by a distribution with discrete support at the

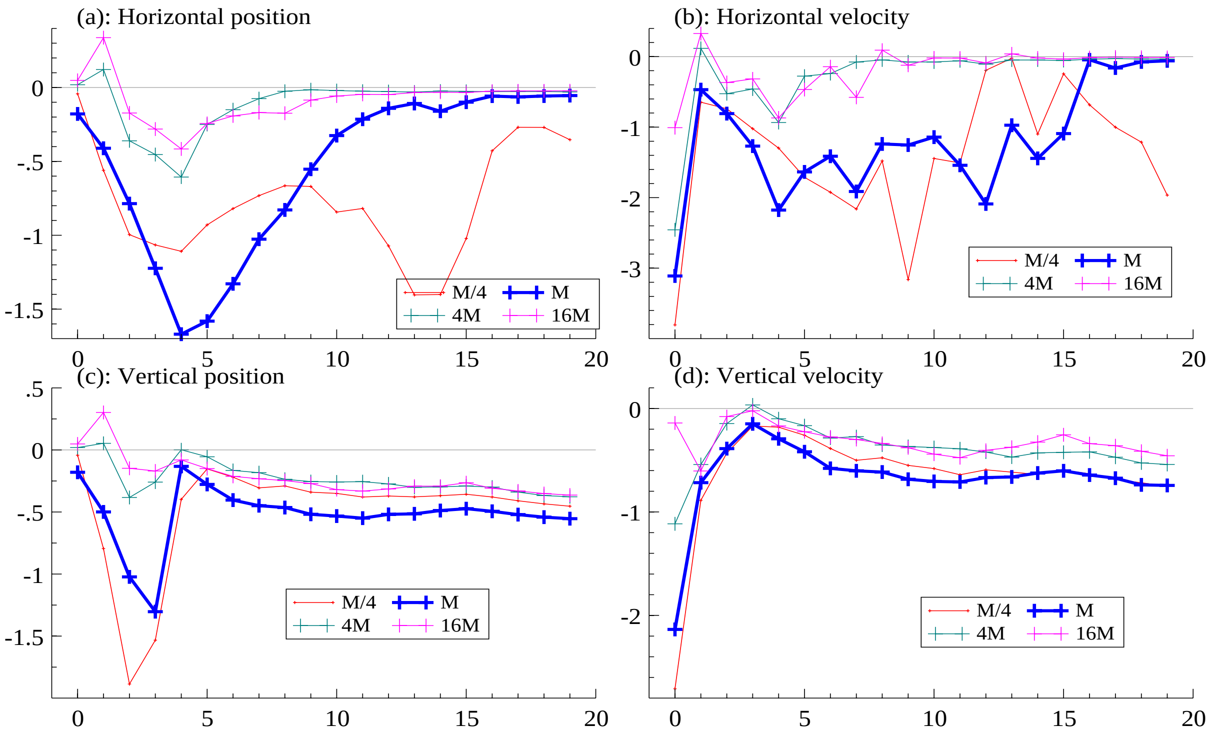

Figure 2: *Plot of the relative mean square error performance (on the log-scale) of the particle filter and the auxiliary based particle filter for the bearings only tracking problem. Numbers below zero indicate a superior performance by the auxiliary particle filter. In these graphs* R *takes on the values* $M/4$, M, $4M$ *and* $16M$. *Throughout SIR is used as the sampling mechanism. Figure (a):* $\alpha_{t1} = x_t$, *Figure (c):* $\alpha_{t3} = z_t$, *while Figure (b):* $\alpha_{t2} = vx_t$ *and Figure (d):* $\alpha_{t4} = vz_t$.

points $\alpha_t^1, ..., \alpha_t^M$, with mass $\pi_t^1, ..., \pi_t^M$. Then the task will be to update this distribution to provide a sample from $\alpha_{t+1}, ..., \alpha_{t+p}|Y_{t+p}$. At first sight this result seems specialized as it is not often that we have to update after the arrival of a block of observations. However, as well as solving this problem it also suggests a way of reducing the bias caused by using the empirical prediction density as an approximation to $f(\alpha_{t+1}|Y_t)$. Suppose that instead of updating p future observations simultaneously, we store $p-1$ observations and update those observations together with an empirical prediction density for $f(\alpha_{t-p+2}|Y_{t-p+1})$. This would provide us with draws from $f(\alpha_{t+1}|Y_{t+1})$ as required. We call this fixed lag filtering. The hope is that the influence of the empirical prediction density will be reduced as it will have been propagated p times through the transition density. This may reduce the influence of outliers on the auxiliary method.

This can be carried out by using a straightforward particle filter using SIR, rejection sampling or MCMC, or by building in an auxiliary variable so that we sample from $\alpha_{t+1}, ..., \alpha_{t+p}, k|Y_{t+p}$. Typically the gains from using the auxiliary approach is greater here for as p grows so naive implementations of the particle filter will become less and less efficient due to not being able to adapt the sampler to the measurement density.

To illustrate this general setup consider the use of an auxiliary particle filter where we take

$$\begin{aligned} g(k|Y_{t+p}) &\propto \int f(y_{t+p}|\mu^k_{t+p})...f(y_{t+1}|\mu^k_{t+1})dF(\alpha_{t+p}|\alpha_{t+p-1})...dF(\alpha_{t+1}|\alpha^k_t) \\ &= f(y_{t+p}|\mu^k_{t+p})...f(y_{t+1}|\mu^k_{t+1}), \end{aligned}$$

and then sampling the index k with weights proportional to $g(k|Y_{t+p})$. Having selected the index k^j we then propagate the transition equation p steps to produce a draw $\alpha^j_{t+1},...,\alpha^j_{t+p}$, $j = 1,...,R$. These are then reweighted according to the ratio

$$\frac{f(y_{t+p}|\alpha^j_{t+p})...f(y_{t+1}|\alpha^j_{t+1})}{f(y_{t+p}|\mu^{kj}_{t+p})...f(y_{t+1}|\mu^{kj}_{t+1})}.$$

This approach has three main difficulties. First it requires us to store p sets of observations and $p \times M$ mixture components. This is more expensive than the previous method as well as being slightly harder to implement. Second, each auxiliary variable draw now involves $3p$ density evaluations and the generation of p simulated propagation steps. Third, the auxiliary variable method is based on approximating the true density of $f(k, \alpha_{t-p+1},...,\alpha_t|Y_t)$, and this approximation is likely to deteriorate as p increases. This suggests that the more sophisticated adaptive sampling schemes, discussed above, may be particularly useful at this point. Again however, this complicates the implementation of the algorithm.

We tried the fixed lag versions of SIR based particle and auxiliary particle filters on the difficult outlier problem previously discussed in Section 2.4 and report the results in detail in Table 2. The results all take $p = 2, 3$ and suggest an order of magnitude improvement in the auxiliary method, and a fall in the efficiency of SIR based particle filters as p increases due to the poor sampling behaviour of the algorithm. The fixed lag auxiliary filter is now 50 to 500 times as efficient, in terms of the reducing the bias, as the plain particle filter for the $p = 3$ case. This suggests that the fixed lag approach has some uses and should be followed up when we deal with difficult problems.

4.2 Stratification

4.2.1 The basics

Suppose that instead of having an equally weighted empirical prediction density, we had a stratified density based on S sets of samples of size $\{M_i, i = 1,...,S\}$. We write the individual elements as $\left\{\alpha^{j,i}_t, i = 1,...,S, j = 1,...,M_i\right\}$. The stratified empirical prediction density is then going to be of the form

$$\sum_{i=1}^{S}\pi^i_t\sum_{j=1}^{M_i} f(\alpha_{t+1}|\alpha^{j,i}_t), \qquad \sum_{i=1}^{S}\pi^i_t = 1.$$

		M					
$p = 2$		particle			auxiliary particle		
R	TRUTH	1,000	10,000	50,000	1,000	10,000	50,000
50	.9074	.4492	.4359	.4455	.6025	.5838	.5909
250	.9074	.5595	.5498	.5632	.6935	.7189	.7150
2,000	.9074	.6675	.6619	.6790	.7842	.8005	.8254
10,000	.9074	.7291	.7479	.7440	.8055	.8515	.8565
25,000	.9074	.7612	.7771	.7641	.7965	.8584	.8751
100,000	.9074	.7840	.8051	.8146	.8087	.8642	.8766
400,000	.9074	.7984	.8251	.8442	.8023	.8588	.8847
$p = 3$							
R	TRUTH	1,000	10,000	50,000	1,000	10,000	50,000
50	.9074	.4116	.4176	.4093	.6040	.6190	.6072
250	.9074	.5369	.5302	.5104	.7150	.7263	.7202
2,000	.9074	.6442	.6632	.6623	.7800	.8108	.8196
10,000	.9074	.7096	.7198	.7276	.8119	.8553	.8674
25,000	.9074	.7495	.7479	.7603	.8281	.8683	.8819
100,000	.9074	.7765	.7848	.7972	.8326	.8764	.8923
400,000	.9074	.7944	.8257	.8299	.8464	.8819	.8946

Table 2: *Fixed lag SIR based particle and auxiliary particle filtering algorithms, using* $p = 2, 3$. *Recorded are the means of 125 independent replications of the filters run on the fixed y using a variety of values of M and R. Thus the Table demonstrates the bias of the methods.*

We will propagate R_i times from the $i - th$ strata. The associated propagation probabilities will be $1/R_i$ for a particle based SIR or

$$p^{j,i} = \frac{f(y_{t+1}|\mu^{j,i}_{t+1})}{\sum_{k=1}^{R_i} f(y_{t+1}|\mu^{k,i}_{t+1})},$$

for an SIR applied to a auxiliary particle filter.

We write the propagated samples' one-step ahead weights as $l^{j,i} = f(y_{t+1}|\alpha^{j,i}_{t+1})/p^{j,i}$ and we resample within each strata. Then we can estimate the strata probabilities as

$$\pi^i_{t+1} = \sum_{j=1}^{R_i} l^{j,i} / \left\{ \sum_{k=1}^{S} \sum_{j=1}^{R_i} l^{j,k} \right\}.$$

Clearly stratification has the difficulty that, in effect, we are performing a SIR sampling method within each strata and so we are likely to need R_i and M_i to be quite large for all values of i. Of course, the advantage is that the strata can be chosen so that the associated weights $l^{1,i}, ..., l^{R_i,i}$ maybe quite even within the strata which may reduce the amount of simulation we require.

One possible generalization of this idea is that we can estimate the strata probabilities, $\{\pi^i_{t+1}\}$, not as a by-product of the SIR operation but as an independent statistical calculation.

We can estimate replace π_{t+1}^i given above by an alternative estimator which is proportional to

$$\pi_t^i \frac{1}{N_i} \sum_{k=1}^{N_i} f(y_{t+1}|\alpha_{t+1}^{k,i}), \tag{12}$$

where the $\alpha_{t+1}^{k,i}$ are generated by first sampling k between $1, ..., M_i$ with equal probability and then drawing from $\alpha_{t+1}|\alpha_t^{k,i}$. This has the advantage that we can then use rejection sampling or MCMC within a strata in a straightforward manner.

Stratification has also been independently proposed in the context of particle filters by Carpenter, Clifford, and Fearnhead (1997) although their motivation is rather different than ours.

4.2.2 Posterior density of parameters

A fundamental characteristic of the particle filter is that it uses a discrete support for the states which randomly changes as time progresses through the propagation mechanism of the transition equation. Points of support with high measurement densities flourish and multiply, while points with low support die out.

Theoretically a state could represent any quantity which we might wish to learn about and so we might be tempted to perform particle filtering on unknown parameters or states which do not change over time. Unfortunately it is well known that this performs very poorly for points of support with low likelihoods are quickly discarded even though they will be very important when the sample size is larger.

A radical alternative is to use stratification for this problem. The idea will be to generate a set of parameter points $\theta_1, ..., \theta_S$ and then to draw the random states associated with each of the parameter values. A sensible way to proceed would be in the first strata to draw M_1 lots of states $\alpha_t^{1,1}, ..., \alpha_t^{M_1,1}$ from $\alpha_0|\theta_1$. The same operation can be conducted for each parameter value. Then we proceed as before.

In this situation the strata probabilities have an interesting interpretation. Clearly π_t^i is an estimate of a constant times $\pi_{t-1}^i f(y_t|Y_{t-1}, \theta_i)$, from (12). That is the strata weights are the normalized likelihood functions for the parameter values. This allows either Bayesian or maximum likelihood calculations.

As we process more information the strata probabilities will increasingly become more concentrated on a few points of support. In this case we can start to discard points of support with extremely low strata probabilities (typically $\log \pi_t^i > -100$) and we can replace them by other values of θ which are close to the points of support which have high strata probabilities. These methods will be discussed at more length in the next section.

5 ESTIMATORS OF PARAMETERS

5.1 Motivation

Suppose that we have a Gaussian autoregression observed with noise (5) and we wished to perform parameter estimation. The Kalman filter yields the exact likelihood in this case, but it might be useful to look at using particle filters to estimate θ in this case as we can use the Kalman filter to check the results. Let $n = 550$, $\beta = 0.5$, $\phi = 0.975$ and $\sigma^2 = 0.02$, where

$$\begin{array}{rcllcl} y_t & = & \alpha_t + \varepsilon_t, & \varepsilon_t & \sim & NID(0, 4.9) \\ \alpha_{t+1} - \beta & = & \phi(\alpha_t - \beta) + \eta_t, & \eta_t & \sim & NID(0, \sigma^2), \end{array}$$

and ε_t and η_t are independent white noise processes. Suppose initially that ϕ and σ^2 is known. Then we will employ a particle filter to estimate the relative log-likelihood as β varies for this problem. Initially we carry out this calculation for 280 different values of β which are drawn from a $N(0, 10)$ distribution.

The particle filter will be stratified, with each point of support for β being given a separate strata. We discard points of support for β which at some stage have $\left\{\pi_t^j\right\}$ which is less than $\exp(-50)$. Naturally a large number of points of support die during the iterative procedure. Typically around a half will survive this experiment. Throughout we will initialize the states by using their unconditional density, so that $\alpha_0|\beta \sim N\left\{\beta, \sigma^2/\left(1-\phi^2\right)\right\}$.

In this experiment we will use a straightforward SIR based particle filter with $M = R/2$ in each strata and take $R = 150, 500, 1000, 3000$. The resulting $\{\pi_n^j\}$, which are proportional to the estimated likelihood function, are graphed in Figure 3. This figure is very ragged reflecting the many sources of randomness in this process. It can be used to estimate the posterior moments of $\beta|y$ by weighting each strata by the prior density of the value of β indexing the strata.

5.2 Smoothing via common random numbers

Some of the roughness of the estimates of the relative log-likelihood function is caused by the use of different random numbers in the different strata. There would seem to be little point in this if we want to compare the estimated log-likelihood in the different strata. Hence we carry out the same experiment but with common random numbers for each strata — using a common seed across each strata, but a different one for each time period. The results are given in Figure 4.

The Figure indicates that using common random numbers does not really change the estimators very much. The estimated log-likelihood is still very random across strata. The reason for this is that the bootstrapping operation is extremely rough so that small changes in the resampling weights can mean that a very different state is sampled. This can effect the estimated

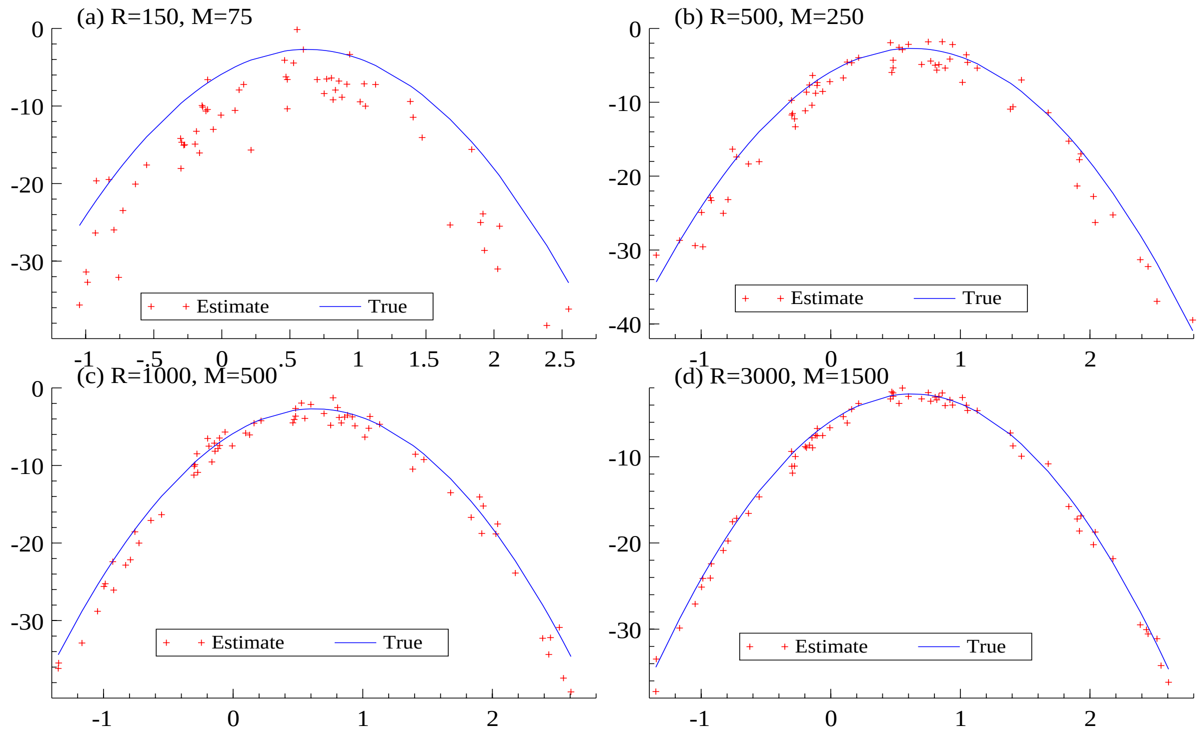

Figure 3: *Estimators of the relative log-likelihood computed via simulation estimators of the prediction decomposition. The graphs plot the estimated log-likelihood against the β, which means the true value is 0.5. Each value of β is used as a strata.*

log-likelihood dramatically.

5.3 Smoothing via sorting and recycling

An alternative to recycling the random numbers is to try to smooth the likelihood via the use of sorting. The discontinuities result as different α_t^j are randomly chosen, when θ changes, in the bootstrap. As contiguous α_t^j can be very different so these changes can make quite a big impact on the estimated likelihood function. We can reduce the impact by simply sorting the α_t^j (and their weights if using an auxiliary variable) before bootstrapping. Then small changes in θ may change the α_t^j which is selected, but the impact on the estimated log-likelihood will be small.

This approach has three problems. First, sorting is quite expensive and so this addition to the algorithm considerably slows it. Second, although this reduces the discontinuities, they are not removed. Third, in very complicated problems it may not be immediately obvious how to sort the states (or signals) in order to reduce their sensitivity.

Usually we have found the pure particle filter is far smoother than its auxiliary variable generalization. This is because the second stage resampling weights of the auxiliary variable are more sensitive to θ than the standard particle filter.

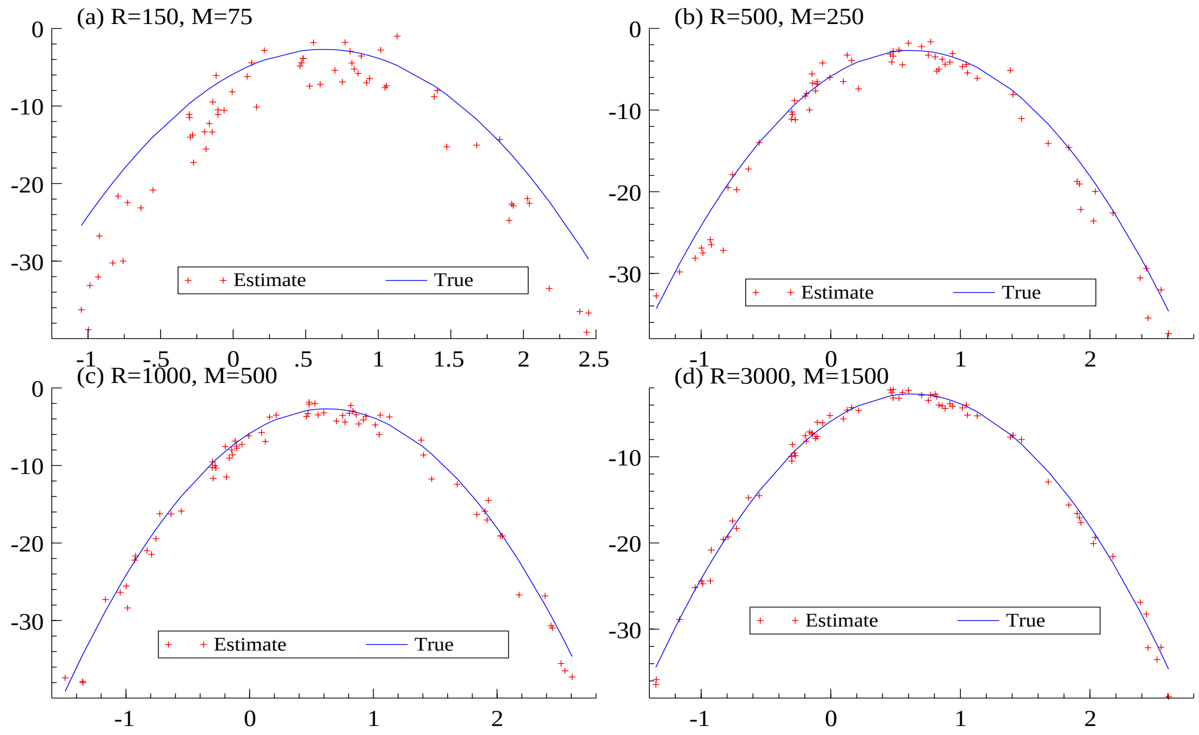

Figure 4: *Estimators of the relative log-likelihood computed via simulation estimators of the prediction decomposition using common random numbers in each strata. The graphs plot the estimated log-likelihood against the β, which means the true value is 0.5. Each value of β is used as a strata and common random numbers are used in each strata.*

To illustrate this we repeat the above experiment but now with common random numbers and using sorting. The resulting estimates of the log-likelihood are given in Figure 5. These curves are now extremely smooth — although they are still not differentiable for all parameter points.

These estimated log-likelihood functions are sufficiently smooth to enable us to use a standard Nelder-Meed simplex numerical optimization routine to approximate the maximum likelihood estimator and its sampling variation. The parameterization has been selected so that the optimization can be carried out in a relatively small number of steps.

In this experiment we take the same model but with $n = 955$ and now use $M = 2500$ and $R = 5000$. Throughout we use constant random numbers and sort the states for each value of t. The reported standard errors, given in Table 3 are computed using an outer-product estimator of the covariance of the score, using quite a coarse numerical derivative on the parameterization $\beta/2$, $\{\log\phi/(1-\phi)\}/4$ and $\sigma^{1/2}$. The Table also reports the implied 95% confidence intervals for the parameters of interest, β, ϕ and σ.

The results are suggestive that the MLE estimator is quite easily approximated by the simulation version. To confirm this we can estimate the score a number of times using new sets

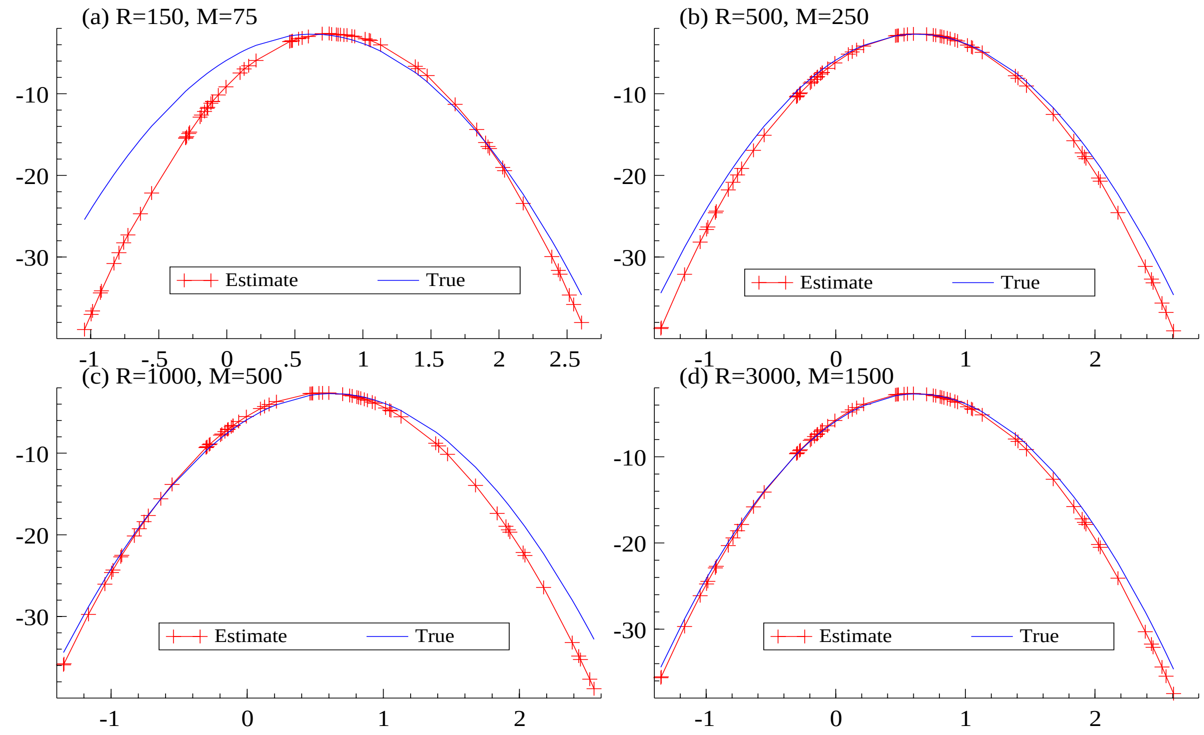

Figure 5: *Estimators of the relative log-likelihood computed via simulation estimators of the prediction decomposition. The graphs plot the estimated log-likelihood against the β, which means the true value is 0.5. Each value of β is used as a strata and common random numbers are used in each strata. Further the value of the states are sorted during each propogation step.*

of random numbers. This will allow us to estimate the variability of the score as a function of the simulation process and so estimate the contribution the simulation makes to the parameter uncertainty.

In this experiment we estimate the score 10 independent (with fixed data but with new simulations each time) times and report in Table 4 the average score and the associated covariance matrix. The values reported are very small.

5.4 Moment conditions

The particle filters deliver draws from $\alpha_{t+1}|Y_t$ and so can be used to unbiasedly estimate the moments (assuming they exist) of $y_{t+1}|Y_t$ for every value of t. Hence we can construct a series of moment conditions

$$y_{t+1}^k - E\left(y_{t+1}^k|Y_t;\theta\right),\ k=1,...$$

replacing the expected value by an unbiased simulated version of this expectation (notice the simulation error does not effect this argument). Such estimated moment conditions hold true over the whole data and are independent over t when the parameter is taken at the true value.

	Sim MLE	Covariance			
$\log(\phi/(1-\phi))/4$	0.88646	0.024742	-0.077219	-0.00052404	
$\sigma^{1/2} \times 10$	3.7348	-0.077219	0.39342	0.0022645	
$\beta/2$	0.23631	-0.00052404	0.0022645	0.0076095	
	Sim MLE	upper .025	lower .975	Initial values	MLE
ϕ	0.972	0.910	0.991	0.9	0.972
σ	0.139	0.063	0.246	0.141	0.139
β	0.473	0.131	0.815	0.5	0.478

Table 3: *Simulation estimation of the ML estimator of the parameters: one version is with the transformed parameters, the other with the parameters of interest. The confidence intervals are calculated as being two sided and contain the truth with 0.95 probability. Initial values denote the initial values used in the optimizarion. MLE denotes the true maximum likelihood estimator of these parameters.*

	Average score	Covariance of score		
$\log(\phi/(1-\phi))/4$	-0.00018226	2.1655e-006	4.9657e-007	2.1291e-007
$\sigma^{1/2} \times 10$	-0.00012770	4.9657e-007	3.0412e-007	1.7816e-007
$\beta/2$	-0.00049014	2.1291e-007	1.7816e-007	4.6521e-007

Table 4: *Simulation of the score at the simulated MLE, varying the random numbers in the simulation. The estimate of the average score and the covariance of the score is carried out using 10 replications.*

Hence a second set of estimated moment constraints is that

$$\left\{y_{t+1} - E\left(y_{t+1}|Y_t;\theta\right)\right\}\left\{y_{t+s} - E\left(y_{t+s}|Y_{t+s-1};\theta\right)\right\},\ s = 2, ...$$

All of these moments could be used as an input into a generalized method of moment (GMM) estimation procedure. For a discussion of GMM see, for example, Hansen (1982) and an introduction given in Hamilton (1994, Ch. 14).

A continual problem with this approach is that the moment conditions are again not smooth in θ even with fixed random numbers and employing sorting.

6 APPLIED EXAMPLE: STOCHASTIC VOLATILITY

6.1 Application

In this section we will analyse the weekday closes (difference of the log of the series) on the Pound Sterling/US Dollar exchange rate from 1/10/81 to 28/6/85. The sample size is $n = 946$. This dataset has been previously analysed using quasi-likelihood methods in Harvey, Ruiz, and Shephard (1994) and by Bayesian MCMC by Kim, Shephard, and Chib (1998), whose result are

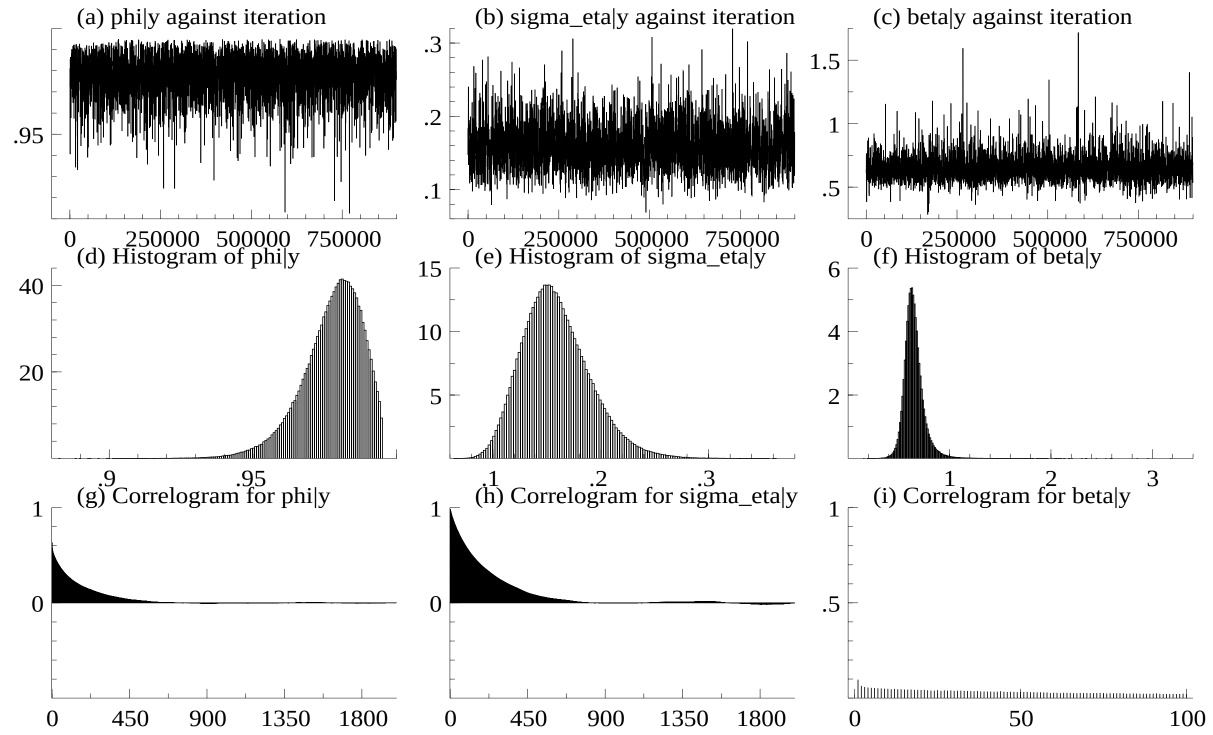

Figure 6: *Single move Gibbs sampler for the Sterling series. Graphs (a)-(c): simulations against iteration. Graphs (d)-(f): histograms of marginal distribution. Graphs (g)-(i): corresponding correlograms for simulation. In total 1,000,000 iterations were drawn, discarding the first 50,000.*

summarized in Table 5 using simulations graphed in Figure 6. In that paper the prior for the parameters were independent, with $\phi \sim\ 2Beta(20, 1.5) - 1.0$, $\sigma_\eta^2\ \sim 0.01 \times 5/\chi_5^2$ and a diffuse prior on $\log \beta$. Here we replace the diffuse prior by a Gaussian distribution with a mean of zero and variance of ten as particle filters cannot deal with diffuse conditions.

	Mean	MC S.E.	Inefficiency	Covariance & *Correlation*		
$\phi\|y$	0.97762	0.00013754	163.55	0.00011062	*-0.684*	*0.203*
$\sigma_\eta\|y$	0.15820	0.00063273	386.80	-0.00022570	0.00098303	*-0.129*
$\beta\|y$	0.64884	0.00036464	12.764	0.00021196	-0.00040183	0.0098569

Table 5: *Daily returns for Sterling: summaries of Figure 6. The Monte Carlo S.E. of simulation is computed using a bandwidth of 2,000, 4,000 and 2,000 respectively. Italics are correlations rather than covariances of the posterior. Inefficiency denotes an estimate of the simulation efficiency of this MCMC procedure compared to a hypothetical sampler which produces iid draws using the same computer time.*

Here we will perform an on-line Bayesian analysis of this problem as well as a simulated maximum likelihood analysis. Throughout we use the simplest of particle filters based on a SIR algorithm. The simulation efficiency of this procedure could be very significantly improved by using the auxiliary variables rejection algorithm which is available for the SV model.

6.2 On-line Bayesian estimation and diagnostic checking

In this section we estimate the SV model on-line using a SIR based filtering algorithm with stratification. We stratify to keep 840 different values of the parameters. These parameters are initially drawn from the prior density. In each strata we take $R = 1785$ and $M = 892$. Common random numbers were used across strata and sorting was employed. Up to $t = 100$ we replace draws from the prior which had relatively log-likelihoods which were less than -1×10^{40}. At the end of the sample we have 216 remaining points of support.

The resulting simulation estimates of the posterior means and covariances are given in Table 6. The results are very slightly different from Table 5 for the MCMC algorithm.

	Mean	MC S.E.	Ineff	Covariance & *Correlation*			lower .025	upper .975
$\phi \| y$	0.97466	0.00169	32.4	0.0000742	*-0.63186*	*0.28177*	0.96454	0.99516
$\sigma_\eta \| y$	0.15988	0.00467	22.1	-0.0001566	0.000828	*-0.26633*	0.091702	0.19530
$\beta \| y$	0.63775	0.01483	21.2	0.0002268	-0.000716	0.008725	0.53857	0.85807

Table 6: *SIR based stratified sampler to perform Bayesian calculations. The lower and upper points are estimates of the upper and lower 0.025 and 0.975 quantiles of the posterior density. MC S.E. denotes a bootstrap estimator of the standard error in estimating via simulation the posterior mean. Ineff denotes the inefficiency factor, which is an estimate of the simulation efficiency of the SIR based sampler (using 840 strata drawn from the prior) compared to a hypothetical sampling which produces iid draws from the posterior distribution.*

The simulation error induced by using this procedure is estimated in the following way. Suppose we use, in total, S strata then we estimate the posterior moments as

$$\widehat{\theta} = \sum_{i=1}^{S} \pi_i \theta_i, \qquad \text{such that} \qquad \sum_{i=1}^{S} \pi_i = 1,$$

where π_i represents the relative likelihood for the simulated prior value θ_i. Then the simulation error is estimated via a bootstrap conducted on the discrete population $(\theta_1, \pi_1), ..., (\theta_S, \pi_S)$. The i-th replication of the bootstrap draws from the S strata with equal probability and with replacement to produce a bootstrap sample $(\theta^{1,i}, \pi^{1,i}), ..., \left(\theta^{S,i}, \pi^{S,i}\right)$ and records the corresponding mean

$$\widehat{\theta}^i = \sum_{j=1}^{S} \pi^{j,i} \theta^{j,i} / \sum_{j=1}^{S} \pi^{j,i}, \qquad i = 1, ..., B.$$

The estimated of the simulation error is then the standard deviation of the bootstrap replications $\widehat{\theta}^1, ..., \widehat{\theta}^B$.

The inefficiency factor attempts to measure the statistical efficiency of the simulation method compared to a hypothetical sampler which is able to produce iid samples from the posterior density. The stratification sampler we are using draws S samples from the prior and then produces an estimated Monte Carlo variance via the above bootstrap argument. Call the simulation variance $Var(E\theta|y)$, then we compute the inefficiency factor by looking at the ratio of $S \times Var(E\theta|y)$ to the posterior variance.

The advantage of this approach is that it produces on-line estimates of the parameters and so allows us to see how the parameter estimates are influenced by various stretches of data. Figure 7 displays the evolution of the posterior means of the parameters together with the upper and lower 2.5 percentage of the distribution. Inevitably these quantiles are rather roughly estimated and so vary quite dramatically at times.

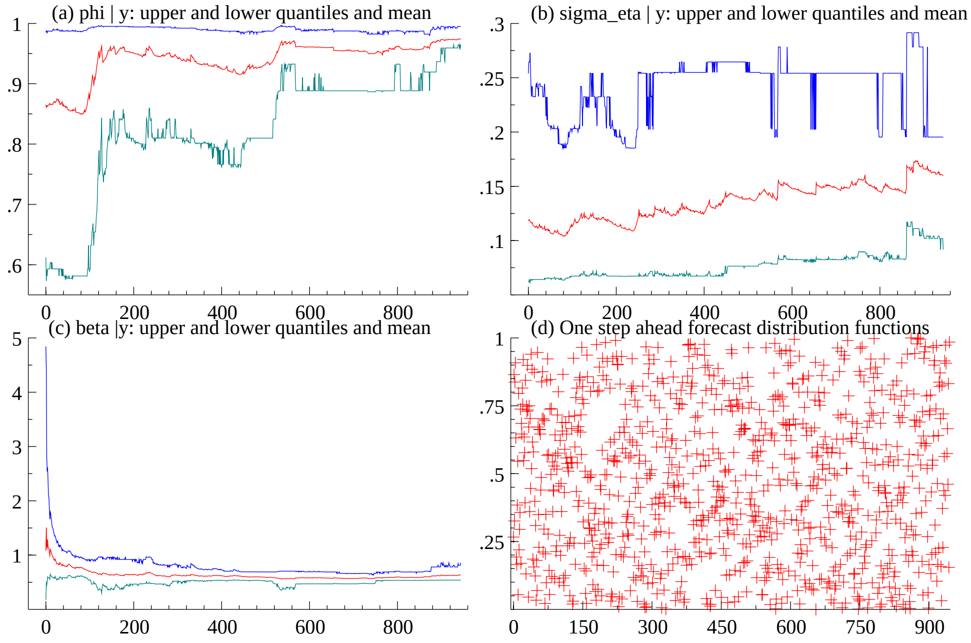

Figure 7: *On-line Bayesian estimation of the SV model. (a)-(c) Graphs the estimated posterior mean and upper and lower 2.5 percentage points of the posterior distribution function against observation number. (a) is for $\phi|y$, (b) has σ_η and (c) has $\beta|y$. Graph (d) plots the estimated distribution function of the one step ahead prediction distribution against observation number. These shoulc be $UID(0,1)$ if the model and prior are true.*

The estimated distribution functions are used in a series of statistical graphs to assess the validity of the model and prior densities. Let us write u_t as the distribution functions, then we will compute a histogram of their marginal distribution and an associated QQ-plot. We will also map them to normals via the inverse of the Gaussian distribution function. Then their correlogram will be plotted. In addition the same operation will be performed on the reflected uniforms, which work on $2|u_t - 0.5|$. These reflected uniforms assess whether there is predictability in the size of discrepancy of the uniforms from their means. The corresponding correlogram should pick up failures in the scale of the forecast density. It is the natural extension of an ARCH based test which is used for linear models. The results of these calculations are plotted in Figure 8.

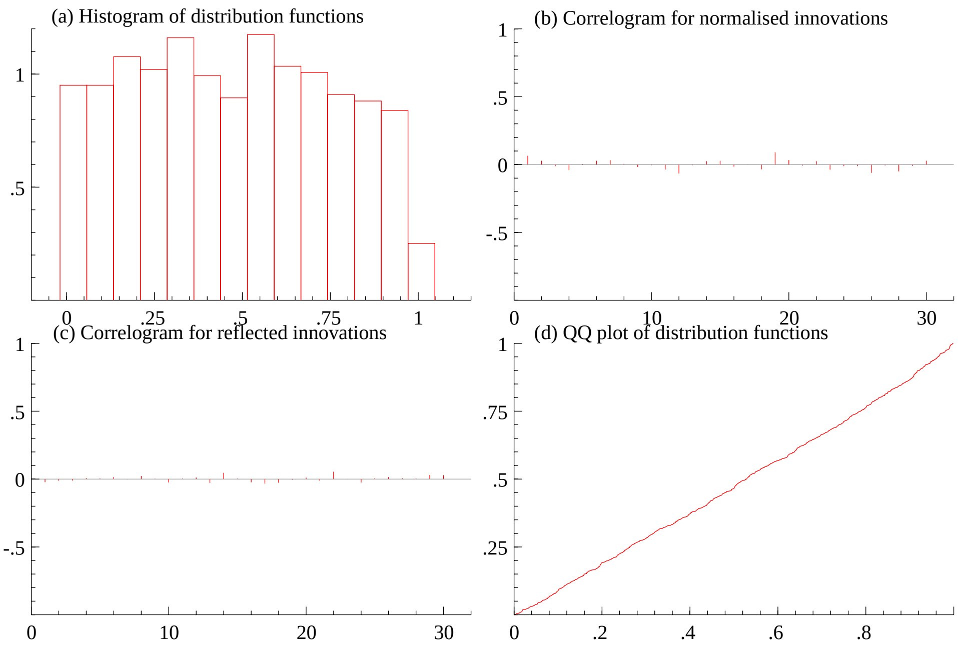

Figure 8: *Diagnostics for on-line Bayesian estimation of the SV model. Graph (a) is the marginal histogram of the estimated distribution functions. These should be uniform if the model and prior are true. Graph (b) works on the inverse Gaussian distribution function evaluated at the estimated distribution function. These should be white noise if the the model and prior is true and so Graph (b) plots their correlogram. Graph (c) does the same operation but on the reflected uniforms. Graph (d) is the QQ plot of the estimated distribution function and so contains the same information as Graph (a).*

These graphs show the model fits this data quite well with a little failure in Graph (b). However, this is a failure of the mean of the model, not the volatility part which would have been picked up in Graph (c). The histogram and QQ-plot suggest the model and prior fit the data quite well.

Formalizing tests of the model and prior are in principle completely straightforward given the nature of the estimated distribution functions under the null hypothesis. Let s_y denote any test statistic which is a function of the data only through the estimate distribution functions. Under the null s_y is exactly pivotal and so we can use it as the basis of an exact Monte Carlo test of the hypothesis that the model and prior are true (see, for example, Ripley (1987, pp. 171-4)). This takes on the remarkably simple form of simulating $u_1^j, ..., u_n^j$ as iid draws from $UID(0,1)$ and then using these as an input into the statistic s, to deliver the observed value s_{u^j}. Carrying this out M times delivers a population $s_{u^1}, ..., s_{u^M}$ which gives us a basis for judging the size of the observed value s_y which resulted from the data. Under the null s_y comes from the same population as $s_{u^1}, ..., s_{u^M}$ and so can be used as a basis for a formal test.

This setup is both remarkably simple and extends to the case where we think of computing the exact distribution of many tests simultaneously. Indeed it would seem that we can control the overall size of all simultaneous tests in this framework.

6.3 Maximum likelihood estimation

We also used the SIR based particle filter to perform maximum simulated likelihood estimation of the SV model for this data set. The results are given in Table 7 and are based on taking $M = 2500$ and $R = 5000$. The results are broadly in line with the Bayesian estimation except that σ is estimated to be quite a lot higher in this experiment. This could be due to the prior density for the Bayesian estimator which had quite a lot of mass below 0.1. The confidence intervals are of the same type of size as observed for the Bayesian analysis. Again Table 7 displays the simulation covariance of the scores to give a guide as to the uncertainty associated with this simulation based estimator. The covariance is very small and so we feel confident that the estimator is very close to the maximum likelihood estimator.

	Sim MLE	Covariance		
$\log(\phi/(1-\phi))/4$	0.88912	0.010242	-0.023987	0.00024964
$\sigma^{1/2} \times 10$	4.1244	-0.023987	0.11019	-0.0023843
$\log(\beta)$	-0.47843	0.00024964	-0.0023843	0.0088938
	Sim MLE	upper .025	lower .975	Initial values
ϕ	0.97177	0.941	0.987	0.972
σ	0.170	0.121	0.228	0.141
β	0.620	0.515	0.746	0.606
Simulation error for the score		Covariance		
$\log(\phi/(1-\phi))/4$		3.0465e-006	2.0024e-007	8.4064e-007
$\sigma^{1/2} \times 10$		2.0024e-007	1.0137e-006	1.0890e-006
$\log(\beta)$		8.4064e-007	1.0890e-006	2.9953e-006

Table 7: *Simulation estimation of the ML estimator of the SV parameters: one version is with the transformed parameters, the other with the parameters of interest. The confidence intervals are calculated as being two sided and contain the truth with 0.95 probability. Initial values denote the initial values used in the optimizarion. Simulation error for the score is the covariance of 10 independent simulations of the score evaluated at the simulated ML estimator.*

7 EXTENSIONS

7.1 Allowing feedback

In some problems the standard model is not sufficiently rich and has to be extended. In particular it is often helpful to allow the measurement and transition equations to depend on past observations. If we write the information up to time t as Y_t, then the measurement density becomes $f(y_t|\alpha_t, \alpha_{t-1}, Y_{t-1})$, while the transition density is $f(\alpha_{t+1}|\alpha_t, Y_t)$. We call this a feedback model.

In principle the particle and auxiliary particle filter methods are not complicated by this extension, although now it is necessary to store the history of the time series. All that changes is that we now must be able to simulate from $f(\alpha_{t+1}|\alpha_t, Y_t)$ and evaluate $f(y_t|\alpha_t, \alpha_{t-1}, Y_{t-1})$.

A simple example of this framework is where $f(y_t|\alpha_t, Y_{t-1})$ is a switching autoregression with outliers and level shifts, with α_t being a discrete state Markov chain so that $f(\alpha_{t+1}|\alpha_t, Y_t) = f(\alpha_{t+1}|\alpha_t)$. Such models have attracted quite a lot of interest following the work of, for example, Hamilton (1989), Albert and Chib (1993), McCulloch and Tsay (1993), McCulloch and Tsay (1994) and Gerlach, Carter, and Kohn (1996).

7.2 Simulation based models

In this paper we have shown how to perform filtering even in cases where we can only simulate from $f(\alpha_{t+1}|\alpha_t)$ and evaluate the measurement density $f(y_t|\alpha_t)$. In some problems it may be unrealistic to assume that we can compute $f(y_t|\alpha_t)$, but in such situations it might still be possible to simulate from $y_t|\alpha_t$. In these cases we could draw a large sample $y_t^1, ..., y_t^S$ from $y_t|\alpha_t$ and then use some non-parametric density estimator to form an estimate $f(\widehat{y_t|\alpha_t})$. This could then replace the true density in the SIR algorithm outlined above for the particle filter. The use of common random numbers and a smooth density estimator could be very useful in this situation as this could reduce the impact of the simulation error in estimating the likelihood ratios important in the SIR algorithm.

8 CONCLUSION

This paper has studied the weaknesses of the very attractive particle filtering method proposed by Gordon, Salmond, and Smith (1993). The SIR implementation of this method is not robust to outliers for two different reasons: sampling efficiency and the unreliability of the empirical prediction density in the tails of the distribution. We introduce an auxiliary variable into the particle filter to overcome the first of these problems, providing a powerful framework which is as simple as SIR, but more flexible and reliable. We study the fixed lag filtering algorithm to tackle the second problem. Our experiments suggest that this produces a significant improvement in the algorithm, however it still cannot deal with some problems.

The combination of the two improvements produce an algorithm which is a very significant improvement over the existing technology. Further, it can be tailored to the particular problem at hand. We believe that except in some very exceptional problems, the auxiliary variable particle fixed lag filtering algorithm can be used reliably.

9 APPENDIX

9.1 Multinomial sampling

The following algorithm is discussed in Carpenter, Clifford, and Fearnhead (1997). Suppose x takes on the values $0, ..., I-1$ with probability of $\pi_0, ..., \pi_{I-1}$. Then the task will be to draw a sample of size R from this discrete distribution in $O(R)$ computations. We carry this out by sampling an ordered version of these variables, so that $x_0 \leq x_1 \leq ... \leq x_{R-2} \leq x_{R-1}$. In the applications discussed in this paper it is not necessary to shuffle these variables.

Drawing order variables will be carried out by first drawing order uniforms (see, for example, Ripley (1987, p. 96)). Let $u_0, ..., u_{R-1} \sim UID(0,1)$, then

$$u_{(R-1)} = u_{R-1}^{1/R}, \quad u_{(k)} = u_{(k+1)} u_k^{1/(k+1)}, \quad k = R-2, R-3, ..., 1, 0,$$

where $u_{(0)} < u_{(1)} < ... < u_{(R-2)} < u_{(R-1)}$. This is most easily carried out in logarithms.

Then we calculate the ordered x using the following trivial algorithm

```
s=0,k=0,j=0;
for (i=0; i<I; i++)
{
   s+=$\pi_i$;
   while ($u_{(j)} \leq s$   &&   $j < R$)
   {
      $x_j = i$;
      j+=1;
   }
}
```

ACKNOWLEDGEMENTS

We would like to thank the ESRC for their financial support through the project ‘Estimation via Simulation in Econometrics’ and the EU through their grant ‘Econometric inference using Simulation Techniques’. All the computations reported in this paper were carried out using the

Ox language of Doornik (1996). Neil Shephard would like to thank Robert Kohn, Peter Clifford, Angelo Melino and David Firth for various helpful conversations.

References

Albert, J. H. and S. Chib (1993). Bayesian inference via Gibbs sampling of autoregressive time series subject to Markov mean and variance shifts. *J. Business and Economic Statist. 11*, 1–15.

Berzuini, C., N. G. Best, W. R. Gilks, and C. Larizza (1997). Dynamic graphical models and Markov chain Monte Carlo methods. *J. Am. Statist. Assoc. 92*. Forthcoming.

Bollerslev, T., R. F. Engle, and D. B. Nelson (1994). ARCH models. In R. F. Engle and D. McFadden (Eds.), *The Handbook of Econometrics, Volume 4*, pp. 2959–3038. North-Holland.

Carpenter, J. R., P. Clifford, and P. Fearnhead (1996). Sampling strategies for Monte Carlo filters of non-linear systems. *IEE Colloquium Digest 243*, 6/1–6/3.

Carpenter, J. R., P. Clifford, and P. Fearnhead (1997). Efficient implementation of particle filters for non-linear systems. 4th Interim Report, DRA contract WSS/U1172, Department of Statistics, Oxford University.

Carter, C. K. and R. Kohn (1994). On Gibbs sampling for state space models. *Biometrika 81*, 541–53.

Carter, C. K. and R. Kohn (1996). Markov chain Monte Carlo in conditionally Gaussian state space models. *Biometrika 83*, 589–601.

Dawid, A. P. (1982). The well-calibrated Bayesian (with discussion). *J. Am. Statist. Assoc. 77*, 605–613.

Diebold, F. X. and M. Nerlove (1989). The dynamics of exchange rate volatility: a multivariate latent factor ARCH model. *J. Appl. Econometrics 4*, 1–21.

Doornik, J. A. (1996). *Ox: Object Oriented Matrix Programming, 1.10.* London: Chapman & Hall.

Efron, B. and R. J. Tibshirani (1993). *An Introduction to the Bootstrap.* New York: Chapman & Hall.

Fisher, N. I. (1993). *Statistical Analysis of Circular Data.* Cambridge: Cambridge University Press.

Fruhwirth-Schnatter, S. (1994). Data augmentation and dynamic linear models. *J. Time Series Analysis 15*, 183–202.

Gerlach, R., C. Carter, and R. Kohn (1996). Diagnostics for time series analysis. Unpublished paper: Australian Graduate School of Management, University of New South Wales.

Geweke, J. (1994). Bayesian comparison of econometric models. Unpublished paper: Federal Reserve Bank of Minneapolis.

Gilks, W. K., S. Richardson, and D. J. Spiegelhalter (1996). *Markov Chain Monte Carlo in Practice.* London: Chapman & Hall.

Gordon, N. J., D. J. Salmond, and A. F. M. Smith (1993). A novel approach to non-linear and non-Gaussian Bayesian state estimation. *IEE-Proceedings F 140*, 107–33.

Hamilton, J. (1989). A new approach to the economic analysis of nonstationary time series and the business cycle. *Econometrica 57*, 357–384.

Hamilton, J. (1994). *Time Series Analysis.* Princeton: Princeton University Press.

Hansen, L. P. (1982). Large sample properties of generalized method of moments estimators. *Econometrica 50*, 1029–54.

Harvey, A. C. (1989). *Forecasting, Structural Time Series Models and the Kalman Filter.* Cambridge: Cambridge University Press.

Harvey, A. C., E. Ruiz, and E. Sentana (1992). Unobserved component time series models with ARCH disturbances. *J. Econometrics 52*, 129–158.

Harvey, A. C., E. Ruiz, and N. Shephard (1994). Multivariate stochastic variance models. *Rev. Economic Studies 61*, 247–64.

Hull, J. and A. White (1987). The pricing of options on assets with stochastic volatilities. *J. Finance 42*, 281–300.

Isard, M. and A. Blake (1996). Contour tracking by stochastic propagation of conditional density. *Proceedings of the European Conference on Computer Vision, Cambridge 1*, 343–356.

Jacquier, E., N. G. Polson, and P. E. Rossi (1994). Bayesian analysis of stochastic volatility models (with discussion). *J. Business and Economic Statist. 12*, 371–417.

Kim, S., N. Shephard, and S. Chib (1998). Stochastic volatility: likelihood inference and comparison with ARCH models. *Rev. Economic Studies 65.* Forthcoming.

King, M., E. Sentana, and S. Wadhwani (1994). Volatility and links between national stock markets. *Econometrica 62*, 901–933.

Kitagawa, G. (1987). Non-Gaussian state space modelling of non-stationary time series. *J. Am. Statist. Assoc. 82*, 503–514.

Kitagawa, G. (1996). Monte Carlo filter and smoother for non-Gaussian nonlinear state space models. *Journal of Computational and Graphical Statistics 5*, 1–25.

Laroque, G. and B. Salanie (1993). Simulation-based estimation of models with lagged latent variables. *J. Appl. Econometrics 8*, S119–S133.

Lee, L. (1995). Simulation estimation of dynamic switching regesion and dynamic disequilibrium models - some Monte Carlo results. Unpublished paper: Department of Economics, The Hong Kong University of Science and Technology.

Liu, J. (1996). Metropolized independent sampling with comparison to rejection sampling and importance sampling. *Statistics and Computing 6*, 113–119.

Liu, J. and R. Chen (1995). Blind deconvolution via sequential imputation. *J. Am. Statist. Assoc. 90*, 567–76.

McCulloch, R. E. and R. S. Tsay (1993). Bayesian inference and prediction for mean and variance shifts in autoregressive time series. *J. Am. Statist. Assoc. 88*, 968–978.

McCulloch, R. E. and R. S. Tsay (1994). Bayesian analysis of autoregressive time series via Gibbs sampling. *J. Time Series Analysis 15*, 235–250.

Ripley, B. D. (1987). *Stochastic Simulation.* New York: Wiley.

Rosenblatt, M. (1952). Remarks on a multivariate transformation. *Annals of Mathematical Statistics 23*, 470–2.

Rubin, D. B. (1987). A noniterative sampling/importance resampling alternative to the data augmentation algorithm for creating a few imputations when the fraction of missing information is modest: the SIR algorithm. Discussion of Tanner and Wong (1987). *J. Am. Statist. Assoc. 82*, 543–546.

Rubin, D. B. (1988). Using the SIR algorithm to simulate posterior distributions. In J. M. Bernardo, M. H. DeGroot, D. V. Lindley, and A. F. M. Smith (Eds.), *Bayesian Statistics 3*, pp. 395–402. Oxford: Oxford University Press Press.

Shephard, N. (1994). Partial non-Gaussian state space. *Biometrika 81*, 115–31.

Shephard, N. and M. K. Pitt (1997). Likelihood analysis of non-Gaussian measurement time series. *Biometrika 84*, 653–67.

Smith, A. F. M. and A. E. Gelfand (1992). Bayesian statistics without tears: a sampling-resampling perspective. *American Statistican 46*, 84–88.

Smith, J. Q. (1985). Diagnostic checks of non-standard time series models. *Journal of Forecasting 4*, 283–91.

Tanner, M. A. and W. H. Wong (1987). The calculation of posterior distributions by data augmentation (with discussion). *J. Am. Statist. Assoc. 82*, 528–50.

West, M. (1992a). Mixture models, Monte Carlo, Bayesian updating and dynamic models. *Computer Science and Statistics 24*, 325–333.

West, M. (1992b). Modelling with mixtures. In J. M. Bernardo, J. O. Berger, A. P. Dawid, and A. F. M. Smith (Eds.), *Bayesian Statistics 4*, pp. 503–524. Oxford: Oxford University Press.

West, M. (1995). Bayesian inference in cyclical component dynamic linear models. *J. Am. Statist. Assoc. 90*, 1301–1312.

West, M. and J. Harrison (1997). *Bayesian Forecasting and Dynamic Models* (2 ed.). New York: Springer-Verlag.