Sequential Monte Carlo Methods for Dynamic Systems

Jun S. Liu and Rong Chen [1]

Abstract

A general framework for using Monte Carlo methods in dynamic systems is provided and its wide applications indicated. Under this framework, several currently available techniques are studied and generalized to accommodate more complex features. All of these methods are partial combinations of three ingredients: importance sampling and resampling, rejection sampling, and Markov chain iterations. We deliver a guideline on how they should be used and under what circumstance each method is most suitable. Through the analysis of differences and connections, we consolidate these methods into a generic algorithm by combining desirable features. In addition, we propose a general use of Rao-Blackwellization to improve performances. Examples from econometrics and engineering are presented to demonstrate the importance of Rao-Blackwellization and to compare different Monte Carlo procedures.

Keywords: Blind deconvolution; Bootstrap filter; Gibbs sampling; Hidden Markov model; Kalman filter; Markov chain Monte Carlo; Particle filter; Sequential imputation; State space model; Target tracking.

[1] Jun S. Liu is an assistant professor of Statistics, Department of Statistics, Stanford University, Stanford, CA 94305. Rong Chen is an associate professor of Statistics, Department of Statistics, Texas A&M University, College Station, TX 77843. Liu's research is partly supported by NSF grants DMS 95-01570 and DMS 95-96096, and the Terman fellowship from Stanford University. Chen's research is partly supported by NSF grant DMS 96-26113. We are grateful to Professor Wing Hung Wong for stimulating discussions that helped to formulate the general SIS framework, to Professors Wally Gilks and Neil Shephard for letting us read their enlightening manuscript before publication, to Professors John Rice and Mike West for pointing out related references, and to the associate editor and referees for many constructive suggestions.

1 INTRODUCTION

Dynamic modeling is an important statistical analysis tool and has attracted much attention from researchers in different fields. One most widely used dynamic model, the linear state space model, has long been an active subject in studying time series data and control systems (Harvey, 1989; West and Harrison, 1989). Despite their computational complexities, nonlinear/non-Gaussian state space models are also important in various applications. A partial list of references is given in Example 2 below.

Models of dynamic nature have also been used in various occasions, such as updating and learning in graphical models or the *probabilistic expert systems* (Spiegelhalter and Lauritzen 1990, Kong, Liu and Wong 1994), simulating protein structures (Leach 1996; Vasquez and Scherago 1985), genetics (Irwing, Cox and Kong 1994), and combinatorial optimizations (Wong and Liang 1997). An example of expert system updating can be found in Berzuini et al. (1997).

In this article, we study Monte Carlo computation methods for *real time* analysis of dynamic systems. Such a system can be abstractly defined as follows:

Definition 1 *A sequence of evolving probability distributions* $\pi_t(\boldsymbol{x}_t)$, *indexed by discrete time* $t = 0, 1, 2, \ldots$, *is called a probabilistic* **dynamic system**. *The state variable* $\boldsymbol{x}_t$ *can evolve in the following three ways: (i) Increasing dimension:* $\boldsymbol{x}_{t+1}$ *has one more component than* $\boldsymbol{x}_t$, *i.e.* $\boldsymbol{x}_{t+1} = (\boldsymbol{x}_t, x_{t+1})$, *where* x_{t+1} *can be a multidimensional component; (ii) Discharging:* $\boldsymbol{x}_{t+1}$ *has one fewer component than* $\boldsymbol{x}_t$, *i.e.* $\boldsymbol{x}_t = (\boldsymbol{x}_{t+1}, d_t)$, *and (iii) No change:* $\boldsymbol{x}_{t+1} = \boldsymbol{x}_t$.

Most of this article will be devoted to situation (i), whereas situations (ii) and (iii) can be handled similarly. Throughout the article, $\pi(\ )$ always refers to the target distribution of the dynamic system, and $p(\ )$ is a generic symbol for probability distributions.

In most applications, the difference between π_{t+1} and π_t is caused by the incorporation of new information in the analysis. Of interests in these systems are usually (a) prediction: $\pi_t(x_{t+1} \mid \boldsymbol{x}_t)$ (i.e., when π_t can be extended to a new component x_{t+1}, the best prediction of x_{t+1} *before* new information arrives is via π_t); (b)updating (smoothing): $\pi_{t+1}(\boldsymbol{x}_t)$ (i.e., the revision of previous state given new information); and (c) new estimation: $\pi_{t+1}(x_{t+1})$ (i.e., what we can say about x_{t+1} in light of new information). The following two examples are typical dynamical systems and they will be referred to repeatedly throughout this article.

Example 1: *Bayesian missing data problem.* Suppose $z_1, \ldots, z_n$ are iid from model $p(z \mid \theta)$, but some z are only partially observed. Let $z_i = (y_i, x_i)$ where y_i is the observed part and x_i the missing part. Let $\boldsymbol{y}_t = (y_1, \ldots, y_t)$ and $\boldsymbol{x}_t = (x_0, x_1, \ldots, x_t)$, where $x_0 = \theta$. The dynamic system in this case is $\pi_t(\boldsymbol{x}_t) = p(\boldsymbol{x}_t \mid \boldsymbol{y}_t)$. Of interest is usually the posterior distribution $\pi_n(x_0) = \int \pi_n(\boldsymbol{x}_n) dx_1 \cdots dx_n$. When θ (i.e., x_0) can be explicitly integrated out from $p(\boldsymbol{x}_t, \boldsymbol{y}_t) = p(y_1, \ldots, y_t, x_1, \ldots, x_t \mid \theta) p(\theta)$, such as in the case of multivariate normal data with missing components (Kong et al. 1994), a good approach is to draw $x_1, \ldots, x_n$ from $\pi_n(x_1, \ldots, x_n)$ and then use Rao-Blackwellization to approximate $\pi_n(\theta)$.

Example 2: *The State Space Model.* Such a model consists of two parts: (1) observation equation, which can be formulated as $y_t \sim f_t(\cdot \mid x_t, \phi)$; and (2) state equation, which can be represented by a Markov process as $x_t \sim q_t(\cdot \mid x_{t-1}, \theta)$. The y_t are observations and the x_t are referred to as the (unobserved) states. Of interest at any time t is the posterior distribution of $\boldsymbol{x}_t \equiv (\phi, \theta, x_1, \ldots, x_t)$. Hence the target distribution at time t is

$$\pi_t(\boldsymbol{x}_t) = \pi_t(\phi, \theta, x_1, \ldots, x_t) = p(\phi, \theta, x_1, \ldots, x_t \mid \boldsymbol{y}_t) \propto p(\theta, \phi) \prod_{s=1}^{t} f_s(y_s \mid x_s, \phi) q_s(x_s \mid x_{s-1}, \theta),$$

where the initial distribution $q_1(x_1 \mid x_0, \theta)$ is assumed known. When the parameters θ and ϕ

are given (such as in many engineering problems), $\boldsymbol{x}_t$ represents $(x_1, \ldots, x_t)$. In practice, the x's can be the unobserved true signals in signal processing (Liu and Chen 1995); the actual words in speech recognition (Rabiner 1989); the target characteristics (e.g., location, velocity etc.) in a multitarget tracking problem (Gordon et al. 1993, 1995; Avitzour 1995); the image characteristics in computer vision (Isard and Blake 1996); the gene indicator in a DNA sequence analysis (Churchill 1989); the underlying volatility in an economical time series (Pitt and Shephard 1997). The applications of dynamic state space model in DNA and protein sequence analysis are often referred to as the *hidden Markov models* (Krogh et al. 1994; Liu, Neuwald and Lawrence 1997).

Except for a few special cases, closed-form analysis of dynamical systems is usually formidable. Recently, there is a surge of interest in designing Monte Carlo methods for the analysis of these models. In fact, most of the references given in Example 2 use Monte Carlo or iterative methods. To implement Monte Carlo for a dynamic system, we need, at any time t, random samples either drawn from $\pi_t(\boldsymbol{x}_t)$ directly, or drawn from another distribution, say $g_t(\boldsymbol{x}_t)$, and weighted properly (importance sampling). Static methods, e.g., most of the popular MCMC schemes (Carlin et al. 1992, Carter and Kohn 1994), achieve this end by treating each π_t separately and repeating same kind of iterative processes. In other words, all of the results (i.e., random draws) obtained at time t are discarded when the system evolves from π_t to π_{t+1}.

However, when the system is slowly varying, (i.e. the L^2 distance between $\pi_t(\boldsymbol{x}_t)$ and $\pi_{t+1}(\boldsymbol{x}_t)$ is small), random samples obtained at time t can be 're-used' to help construct random samples at time $t+1$ so as to improve efficiency. Imagine that we have a sample $S_t = \{\boldsymbol{x}_t^{(j)}, j = 1, \ldots, m\}$, drawn from π_t. When the system evolves to π_{t+1}, it is desirable to keep those $\boldsymbol{x}_t^{(j)}$ and attach to each of them one or several $x_{t+1}^{(j)}$ drawn from some appropriate distribution

$g_{t+1}(\cdot \mid \boldsymbol{x}_t^{(j)})$. Let H_{t+1} denote the sample space of x_{t+1}. Then the foregoing idea is equivalent to drawing sample from the product space $S_t \bigotimes H_{t+1}$. Very often the evolved sample $\boldsymbol{x}_{t+1}^{(j)} = (\boldsymbol{x}_t^{(j)}, x_{t+1}^{(j)})$ needs to be *reweighted* or *resampled* to accommodate π_{t+1}. This is the basic principle behind almost all available sequential MC methods, e.g., Berzuini et al. (1997), Gordon et al. (1993), Hendry and Richard(1990), Kitagawa (1996), Kong et al. (1994), Liu and Chen (1995), MacEachern, Clyde and Liu (1998), Pitt and Shephard (1997), West (1992) etc.

To further elaborate on these ideas, in this article we describe a general framework for using sequential Monte Carlo methods in dynamic systems. Under this framework, we extend and unify previously more restrictive methods, study various reweighting and resampling techniques proposed, and discuss connections and comparisons of these approaches. A main message we want to communicate in this article is that the sequential importance sampling (SIS) setting provides us a good framework for understanding many existing methods and for further improving them (via Rao-Blackwellization, collapsing etc.).

Section 2 describes the general idea of the sequential importance sampling (SIS) method and several key implementation issues, such as the choice of sampling distribution, resampling, and Monte Carlo inference. Section 3 discusses several local Monte Carlo methods that are needed when SIS encounters certain difficulties. Section 4 proposes three methods for resampling from S_t and provides a generic algorithm that combines SIS and resampling. Section 5 brings in Rao-Blackwellization for improving estimation. Section 6 gives three examples to demonstrate the use of Rao-Blackwellization and to compare different procedures. Section 7 concludes with a brief summary.

2 SEQUENTIAL UPDATING IN DYNAMIC SYSTEM

One of the most successful methods for analyzing a complicated probabilistic system (such as a nonlinear state space model) is the Gibbs sampler (Carlin et al. 1992, Carter and Kohn, 1994, Gelfand and Smith 1990, Tanner and Wong 1987). However, the Gibbs sampler is less attractive when one's interest is in *real time* prediction and updating in a dynamic system. Another situation for the Gibbs sampler to be ineffective is when the states of the resulting samples are very "sticky", rendering the sampler very difficult to move (MacEachern et al. 1998). In this case it appears that intelligently choosing a dynamic system for sequential updating can be more efficient (Wong and Liang 1997). We first describe one of such sequential updating strategies, then discuss its several key implementation issues.

2.1 The Sequential Importance Sampling (SIS)

A useful way to represent a complicated high dimensional distribution, such as $\pi_t(\boldsymbol{x}_t)$, is by multiple Monte Carlo samples drawn from it. Multiple imputation (Rubin 1987) is a successful example of such a practice for survey data. In this article, we advocate a similar methodology to that of Rubin's for analyzing dynamic systems.

Definition 2 *A random variable X drawn from a distribution g is said to be* **properly weighted** *by a weighting function $w(X)$ with respect to the distribution π if for any integrable function h,*

$$E_g\{h(X)w(X)\} = E_\pi\{h(X)\}.$$

A set of random draws and weights $(x^{(j)}, w^{(j)})$, $j = 1, 2, \ldots,$ is said properly weighted *with respect to π if*

$$\lim_{m\to\infty} \frac{\sum_{j=1}^m h(x^{(j)})w^{(j)}}{\sum_{j=1}^m w^{(j)}} = E_\pi(h(X)) \tag{1}$$

for any integrable function h. In a practical sense we can think of π as being approximated by the discrete distribution supported on the $x^{(j)}$ with probabilities proportional to the weights $w^{(j)}$.

Let $S_t = \{\boldsymbol{x}_t^{(j)}, j = 1, \ldots, m\}$ denote a set of random draws that are properly weighted by the set of weights $W_t = \{w_t^{(j)}, j = 1, \ldots, m\}$ with respect to π_t. Let H_{t+1} be the sample space of X_{t+1}, and let g_{t+1} be a trial distribution. Then the SIS procedure consists of recursive applications of the following SIS steps:

SIS Step: for $j = 1, \ldots, m$:

(A) Draw X_{t+1}=$x_{t+1}^{(j)}$ from $g_{t+1}(x_{t+1}|\boldsymbol{x}_t^{(j)})$; attach it to $\boldsymbol{x}_t^{(j)}$ to form $\boldsymbol{x}_{t+1}^{(j)}$=$(\boldsymbol{x}_t^{(j)}, x_{t+1}^{(j)})$.

(B) Compute

$$u_{t+1}^{(j)} = \frac{\pi_{t+1}(\boldsymbol{x}_{t+1}^{(j)})}{\pi_t(\boldsymbol{x}_t^{(j)}) g_{t+1}(x_{t+1}^{(j)} \mid \boldsymbol{x}_t^{(j)})}; \quad \text{and let} \quad w_{t+1}^{(j)} = u_{t+1}^{(j)} w_t^{(j)}. \tag{2}$$

Here u_t is called an "incremental weight." It is easy to show that $(\boldsymbol{x}_{t+1}^{(j)}, w_{t+1}^{(j)})$ is a properly weighted sample of π_{t+1}. Thus, the SIS can be applied recursively for $t = 1, 2, \ldots$, to accommodate an ever-changing dynamical system.

The SIS method is also useful in non-Bayesian computation such as evaluating likelihood functions. Applications in this direction can be found in Hendry and Richard (1990) and Irwing et al. (1994). Briefly, suppose we are interested in evaluating the likelihood function $L(\theta) = p(y_1, \ldots, y_t;\ \theta)$ in the missing data problem (Example 1). Then for each fixed θ value, we apply the SIS procedure to impute $(x_1, \ldots, x_t)$ sequentially with $g_1(x_1) = p(x_1 \mid y_1; \theta)$ and

$$g_s(x_s \mid x_1, \ldots, x_{s-1}) = p(x_s \mid \boldsymbol{x}_{s-1}, \boldsymbol{y}_s; \theta), \quad s = 2, 3, \ldots$$

Kong et al. (1994) show that $\sum_{j=1}^m w_t^{(j)}/m$ is an unbiased estimate of $L(\theta)$. In Section 5 we show that Rao-Blackwellization (Casella and Robert 1996, Liu, Wong and Kong 1994) can be

applied to obtain a better estimate.

2.2 Choice of the Sampling Distribution g_{t+1} in SIS

The choice of the sampling distribution g_{t+1} is directly related to the efficiency of the proposed SIS method. For Bayesian missing data problems (example 1), Kong et al. (1994) suggest using

$$g_{t+1}(x_{t+1} \mid \boldsymbol{x}_t) = \pi_{t+1}(x_{t+1} \mid \boldsymbol{x}_t) = p(x_{t+1} \mid \boldsymbol{y}_{t+1}, \boldsymbol{x}_t),$$

with the incremental weight $u_{t+1} \propto p(y_{t+1} \mid \boldsymbol{y}_t, \boldsymbol{x}_t)$. Note that although the exact value is not easily known, u_{t+1} can sometimes be computed up to a normalizing constant, which is sufficient for estimation using formula (1). This choice of g_{t+1} is also used in Liu and Chen (1995). For the state space model (Example 2) with known x_0=(θ, ϕ), a similar trial distribution is

$$\begin{aligned} g_{t+1}(x_{t+1} \mid \boldsymbol{x}_t) &\propto f_{t+1}(y_{t+1} \mid x_{t+1}, \phi) q_{t+1}(x_{t+1} \mid x_t, \theta) \\ u_{t+1} &= \int f_{t+1}(y_{t+1} \mid x_{t+1}, \phi) q_{t+1}(x_{t+1} \mid x_t, \theta) dx_{t+1}. \end{aligned}$$

In the general dynamic system setting, we suggest g to be chosen as

$$g_{t+1}(x_{t+1} \mid \boldsymbol{x}_t) = \pi_{t+1}(x_{t+1} \mid \boldsymbol{x}_t), \quad t = 1, 2, \ldots, \tag{3}$$

with the incremental weight

$$u_{t+1} = \frac{\pi_{t+1}(\boldsymbol{x}_t)}{\pi_t(\boldsymbol{x}_t)}. \tag{4}$$

Note that u_{t+1} in (4) does not depend on the value of x_{t+1} and this feature is important to several issues discussed later. The reason that drawing x_{t+1} from $\pi_{t+1}(x_{t+1} \mid \boldsymbol{x}_t)$ is more desirable than from a more or less arbitrary function $g_{t+1}(x_{t+1} \mid \boldsymbol{x}_t)$ is clear from rewriting the incremental weight (2) as

$$u_{t+1} = \frac{\pi_{t+1}(\boldsymbol{x}_t)}{\pi_t(\boldsymbol{x}_t)} \frac{\pi_{t+1}(x_{t+1} \mid \boldsymbol{x}_t)}{g_{t+1}(x_{t+1} \mid \boldsymbol{x}_t)}.$$

Intuitively, the second ratio is needed to correct the discrepancy between $g_{t+1}(x_{t+1} \mid \boldsymbol{x}_t)$ and $\pi_{t+1}(x_{t+1} \mid \boldsymbol{x}_t)$ when they are different.

Other choices of g_{t+1} are also possible. For instance, if $\pi_t(\boldsymbol{x}_t)$ can be "extended" for x_{t+1}, one may use

$$g^*_{t+1}(x_{t+1} \mid \boldsymbol{x}_t) = \pi_t((x_{t+1}|\boldsymbol{x}_t). \tag{5}$$

For Example 1, this corresponds to $g^*_{t+1}(x_{t+1} \mid \boldsymbol{x}_t) = p(x_{t+1}|\boldsymbol{x}_t, \boldsymbol{y}_t)$. The corresponding incremental weight is $u_{t+1} \propto p(y_{t+1} \mid \boldsymbol{y}_t, \boldsymbol{x}_{t+1})$. For Example 2, choice (5) corresponds to

$$g^*_{t+1}(x_{t+1} \mid \boldsymbol{x}_t) = q_{t+1}(x_{t+1} \mid x_t, \theta)$$

and $u_{t+1} \propto f(y_{t+1} \mid x_{t+1})$. This is used in Avitzour (1995), Gordon et al. (1993, 1995), and Kitagawa (1996). Note that this trial distribution generates x_{t+1} using only the state equation.

Compared with (3), distribution (5) is usually easier to use but tends to result in greater Monte Carlo variation (Berzuini et al. 1997). In the state space model case, it is obvious that the choice (3) is more desirable than (5) because the former incorporates the most recent information in y_{t+1} whereas the latter does not. Using (3) has another advantage in estimation, which will be discussed in section 2.4. In many applications, however, it may not be easy to use (3). Section 3 provides methods for coping with this difficulty.

2.3 Resampling in SIS (SISR)

Suppose $S_t = \{\boldsymbol{x}_t^{(j)}, j = 1, \ldots, m\}$ is properly weighted by $W_t = \{w_t^{(j)}, j = 1, \ldots m\}$ with respect to π_t. Let us call each $\boldsymbol{x}_t^{(j)}$ a *"stream."* Instead of carrying the weight W_t as the system evolves, it is also legitimate, and sometimes beneficial (Liu and Chen, 1995), to insert a resampling step described as follows between SIS recursions, and such a procedure is referred to as the SIS with

resampling (SISR).

Resampling step: (i) sample a new set of streams (denoted as S_t') from S_t according to the weights $w_t^{(j)}$; and then (ii) assign equal weights to the streams in S_t'.

It is not immediately clear why one needs resampling at certain stage t. As much detailed theoretical discussion is given by Liu and Chen (1995), we only mention a few heuristics on the issue. Firstly, if the weights $w_t^{(j)}$ are constant (or near constant) for all t (such a case occurs when one can draw from π_t directly), resampling only reduces the number of distinctive streams and introduces extra Monte Carlo variation. This suggests that one should not perform resampling when the coefficient of variation, cv_t^2, for the $w_t^{(j)}$ is small. As argued in Kong et al (1994), the *'effective sample size'* is inversely proportional to $1+cv_t^2$. Secondly, Kong et al. (1994) show that as the system evolves cv_t^2 *increases* stochastically. When the weights get very skewed at time t, carrying many streams with very small weights is apparently a waste. Resampling can provide chances for the good (i.e., "important") streams to amplify themselves and hence "rejuvenate" the sampler to produce a better result for *future* states as system evolves, though it does not improve inferences on *current* state $\boldsymbol{x}_t$. Examples in Section 6 illustrate these heuristics.

The resampling schedule (i.e., when to resample) can be either deterministic or dynamic, and the sampling scheme can be either simple random sampling (with weights), residual sampling, or local Monte Carlo resampling (Section 4). The methods of Gordon et al. (1993), Hürzeler and Künsch(1995), Kitagawa (1996), Berzuini et al. (1997), and Pitt and Shephard (1997) can all be seen as SIS with special choices of g_{t+1} and with resampling at every stage.

2.4 Inference with Monte Carlo Samples

In dynamic systems, it is often of interest to obtain *on line* inference on the state variables, i.e. estimating $E_{\pi_t} h(\boldsymbol{x}_t)$ at time t. This is straightforward by using (1) when available is a sample $\{\boldsymbol{x}_t^{(j)}\}$ properly weighted by $w_t^{(j)}$. However, several issues concerning statistical efficiency of the estimates are worth mentioning. Casella (1997) provides a general treatment on this issue.

- Estimation should be done *before* a resampling step, since resampling introduces extra random variation in the *current* sample.

- Rao-Blackwellization can improve the accuracy of the estimation. For example, when weight (2) does not depend on x_{t+1}, such as in the case of using the optimal g_{t+1} in (3), the current state x_{t+1} should be estimated *before* it is drawn from g_{t+1}, by using

$$\hat{E}_{\pi_{t+1}} h(x_{t+1}) = \frac{\sum_{j=1}^m w_{t+1}^{(j)} E_{\pi_{t+1}}(h(x_{t+1}) \mid \boldsymbol{x}_t^{(j)})}{\sum_{j=1}^m w_{t+1}^{(j)}}, \tag{6}$$

  provided that $E_{\pi_{t+1}}(h(x_{t+1}) \mid \boldsymbol{x}_t^{(j)})$ can be calculated easily. In mixture normal state space models(Example 2 and Section 6.3) and other examples, this is indeed achievable.

- Delayed estimation (i.e. estimate of $E_{\pi_t} h(x_{t-k})$ at time t) usually is more accurate than concurrent estimation (estimate $E_{\pi_{t-k}} h(x_{t-k})$ at time $t-k$), since the estimation is based on more information. However, precaution needs to be taken with frequent resampling because resampling reduces distinct *past* samples.

2.5 Some Related Methods

The state space model as described in Example 2 has a special Markovian feature that the more general dynamic models do not possess. With given $x_0 = (\phi, \theta)$, Example 2 satisfies that

$$p(x_{t+1} \mid \boldsymbol{x}_t, \boldsymbol{y}_t, y_{t+1}) = p(x_{t+1} \mid x_t, \boldsymbol{y}_{t+1}) \propto f_{t+1}(y_{t+1} \mid x_{t+1}) p(x_t \mid \boldsymbol{y}_t).$$

That is, with given x_t, previous $\boldsymbol{x}_{t-1}$ and $\boldsymbol{y}_t$ can be "forgotten." As in a Kalman filter, the posterior distribution $p(x_t \mid \boldsymbol{y}_t)$ can be obtained recursively, at least in principle. The main difficulty is that analytical formulas for this recursive updating only exist for certain exponential family models (West and Harrison 1989) or finite discrete-state space model (Rabiner 1989).

Because of the popularity and simplicity of the state space model, several sequential Monte Carlo methods have been proposed to deal with nonlinear/non-Gaussian cases. In particular, Hendry and Richard (1990) note the potential use of the SIS in such models. West (1992) suggests to use a mixture distribution to approximate $p(x_t \mid \boldsymbol{y}_t)$ at each time t, and then proceed with an adaptive importance sampling strategy to produce a mixture approximation of $p(x_{t+1} \mid \boldsymbol{y}_t)$ at time $t+1$. Difficulties with this approach are that finding good mixture approximations for every t can be time-consuming and it can be difficult to implement when the dimensionality of x_t is high.

Gordon et al. (1993) and Kitagawa (1996) propose to use importance resampling to obtain a discrete approximation of $p(x_{t+1}|\boldsymbol{y}_{t+1})$, with a given set of samples drawn from $p(x_t|\boldsymbol{y}_t)$. They call such a procedure *bootstrap filter* or *particle filter*. The method has been successfully applied to multiple target tracking (Gordon et al. 1995, Avitzour 1995) and time series analysis (Kitagawa, 1996). Their method is essentially an SIS with g_{t+1} chosen as (5) and resampling at every t. Estimations were performed *after* resampling, which is less efficient. Hürzeler and

Künsch (1995) and Pitt and Shephard (1997) have proposed improved algorithms for the state space model. We discuss their approaches in detail in Section 3.

3 LOCAL MONTE CARLO METHODS FOR SIS

As we have discussed in Section 2.3, a favorable choice of the recursive sampling distribution is $g_{t+1}(x_{t+1} \mid \boldsymbol{x}_t) = \pi_{t+1}(x_{t+1} \mid \boldsymbol{x}_t)$. However, drawing x_{t+1} from $\pi_{t+1}(x_{t+1} \mid \boldsymbol{x}_t)$ may not be directly achievable and the incremental weight u_{t+1} may not be easy to compute. Under *such a premise*, a collection of methods have been developed to overcome the difficulty for the state space model. See, for example, Berzuini et al. (1997), Hürzeler and Künsch(1995), and Pitt and Shephard (1997). We propose here to extend their methods to our general SIS setting for simultaneously estimating the new weight w_{t+1} and drawing x_{t+1}. We refer to these methods as "local Monte Carlo methods" for SIS.

3.1 The Basic Idea

As usual, we let $S_t = \{\boldsymbol{x}_t^{(j)}, j = 1, \ldots, m\}$ and $W_t = \{w_t^{(j)}, j = 1, \ldots, m\}$. The central idea of this section is to regard π_t as being represented by the Monte Carlo sample S_t with weights W_t. Thus, at stage $t+1$, $\boldsymbol{x}_t$ can be treated as a random variable with this discrete *a priori* distribution. To simplify notations, we introduce a random variable J, who takes values in the set $\{1, \ldots, m\}$, to indicate the streams in S_t. Pitt and Shephard (1997) also use such a formulation, and call J the auxiliary variable.

Let the joint distribution of J and x_{t+1} be

$$p(J, x_{t+1}) \propto \frac{\pi_{t+1}(\boldsymbol{x}_t^{(J)}, x_{t+1})}{\pi_t(\boldsymbol{x}_t^{(J)})} w_t^{(J)}. \tag{7}$$

Then the marginal distribution of x_{t+1} from (7) is

$$\hat{\pi}_{t+1}(x_{t+1}) \propto \sum_{j=1}^{m} \frac{\pi_{t+1}(x_{t+1}, \boldsymbol{x}_t^{(j)})}{\pi_t(\boldsymbol{x}_t^{(j)})} w_t^{(j)}, \tag{8}$$

which would be a good approximation to the true marginal distribution $\pi_{t+1}(x_{t+1})$ provided that the Monte Carlo sample size m is large and the distribution of the w_t is not too skewed. The marginal distribution of J is

$$P(J=j) \propto w_t^{(j)} \int \frac{\pi_{t+1}(\boldsymbol{x}_t^{(j)}, x_{t+1})}{\pi_t(\boldsymbol{x}_t^{(j)})} dx_{t+1} = w_t^{(j)} \frac{\pi_{t+1}(\boldsymbol{x}_t^{(j)})}{\pi_t(\boldsymbol{x}_t^{(j)})} = w_t^{(j)} u_{t+1}^{(j)} = w_{t+1}^{(j)},$$

which is exactly the new weight at time $t+1$ for $\boldsymbol{x}_t^{(j)}$ according to (2) and (4).

Hence, if we have a method to draw a sample, $(j_1, x_{t+1}^{(j_1)}), \ldots, (j_\ell, x_{t+1}^{(j_\ell)})$, of (J, x_{t+1}) from (7), then the SIS step can be achieved by

(B) Estimate $w_{t+1}^{(j)}$ by $\hat{f}_j$ = frequency of $\{J=j\}$ in the sample.

(A) Form $\boldsymbol{x}_{t+1}^{(j)} = (\boldsymbol{x}_t^{(j)}, x_{t+1}^*)$ if $\hat{f}_j \neq 0$, where x_{t+1}^* is any value of x_{t+1} that is paired with $J=j$ in the sample.

Several methods for generating samples from (7) are described in the following subsections, and a few remarks are as follows.

Remark 1: As long as the estimates of the weights are unbiased, the new sample is properly weighted by $\hat{f}_j$ with respect to π_{t+1}. An accurate estimation of the weights is *not* necessary. Those $\boldsymbol{x}_t^{(j)}$ with $\hat{f}_j$=0 can be replaced by a random draw from those with $\hat{f}_j \neq 0$. If of interest is the estimation of the incremental weight $u_{t+1}^{(j)}$, one can set $w(\boldsymbol{y}) \equiv 1$ for $\boldsymbol{y} \in S_t$ in the above calculations.

Remark 2: Since the local MC methods provide samples of (J, x_{t+1}) with distribution (7), they achieve resampling effect automatically. See details in Section 4.

Remark 3: None of the methods described in this section are necessary when direct sampling from the optimal g_{t+1} of (3) is achievable.

3.2 Rejection Methods

Suppose we can draw x_{t+1} from a trial distribution $g^*_{t+1}(x_{t+1} \mid \boldsymbol{x}_t)$ which is not equal to (3). There are two rejection methods to sample (J, x_{t+1}) from (7): one is based on the joint distribution of (J, x_{t+1}) and the other based on the marginal of x_{t+1}. Let the "covering constant" be

$$c_{t+1} = \sup_{j, x_{t+1}} \frac{\pi_{t+1}(\boldsymbol{x}_t^{(j)}, x_{t+1})}{\pi_t(\boldsymbol{x}_t^{(j)}) g^*_{t+1}(x_{t+1} \mid \boldsymbol{x}_t^{(j)})}.$$

Rejection method 1:

- Draw $J = j$ with probability proportional to $w_t^{(j)}$;
- Given $J = j$, draw x_{t+1} from $g^*_{t+1}(x_{t+1} \mid \boldsymbol{x}_t^{(j)})$;
- Accept (j, x_{t+1}) with probability

$$p = \frac{\pi_{t+1}(\boldsymbol{x}_t^{(j)}, x_{t+1})}{c_{t+1}\ \pi_t(\boldsymbol{x}_t^{(j)}) g^*_{t+1}(x_{t+1} \mid \boldsymbol{x}_t^{(j)})}.$$

Rejection method 2: The first two steps are identical to Method 1. In Step 3:

- Accept x_{t+1} with probability

$$p = \frac{\sum_{j=1}^m w_t^{(j)} \pi_{t+1}(\boldsymbol{x}_t^{(j)}, x_{t+1}) / \pi_t(\boldsymbol{x}_t^{(j)})}{c_{t+1} \sum_{j=1}^m w_t^{(j)} g^*_{t+1}(x_{t+1} \mid \boldsymbol{x}_t^{(j)})}$$

Then the sample x_{t+1} accepted from using either of the methods follows (8). In method 2, we need to redraw J with probability

$$P(J = j \mid x_{t+1}) \propto \frac{\pi_{t+1}(\boldsymbol{x}_t^{(j)}, x_{t+1})}{\pi_t(\boldsymbol{x}_t^{(j)})} w_t^{(j)}. \tag{9}$$

Methods 1 and 2 are identical in the state space model case and are an essential part of Hürzeler and Künsch (1995). Generally, method 2 is a Rao-Blackwellization of method 1 (Casella and Robert, 1996) and can be more efficient.

3.3 Importance Resampling

Importance resampling method can also be used to generate approximate samples from (7).

- Draw $J = j$ with probability proportional to $w_t^{(j)}$;
- Given $J = j$, draw x_{t+1} from some $g_{t+1}^*(x_{t+1} \mid \boldsymbol{x}_t^{(j)})$;
- Assign to the sample (j, x_{t+1}) the weight

$$w(j, x_{t+1}) = \frac{\pi_{t+1}(\boldsymbol{x}_t^{(j)}, x_{t+1})}{\pi_t(\boldsymbol{x}_t^{(j)}) g_{t+1}^*(x_{t+1} \mid \boldsymbol{x}_t^{(j)})}.$$

The obtained sample (j, x_{t+1}) is properly weighted by the $w(j, x_{t+1})$ with respect to (7). At this point, one has three choices: (a) do resampling to achieve (7) approximately, as implemented in Gordon et al. (1993) and Kitagawa (1996); (b) estimate $P(J = j) \propto w_{t+1}^{(j)}$ directly using the weighted sample of (j, x_{t+1}); or (c) proceed with the newly sampled $(\boldsymbol{x}_t^{(j)}, x_{t+1})$ (with new weights $w(j, x_{t+1})$), as proposed by Pitt and Shephard (1997).

In addition, Pitt and Shephard (1997) suggest using an adjustment multiplier to improve efficiency. Briefly, one can instead draw $J = j$ with probability proportional to $w_t^{(j)} a_{t+1}^{(j)}$ and then adjust the weight accordingly. It is conceivable that by carefully choosing $a_{t+1}^{(j)}$ (a function of $\boldsymbol{x}_t^{(j)}$) and g_{t+1}^* one can achieve good efficiency. This idea can also be applied to rejection sampling and the following MCMC approach.

3.4 Hastings Independence Chain Approach

Alternatively, one can also use Hastings *independence chain* approach (Hastings 1970), as suggested by Berzuini et al. (1997) for the state space model. Here we prescribe a generalization of their method for dynamic systems. Detailed description of the general independence chain method is given in the appendix.

Suppose we can draw X_{t+1} from a trial distribution $g^*_{t+1}(x_{t+1} \mid \boldsymbol{x}_t)$. Then starting with an arbitrary $J^0 = j_0$, we iterate the following steps:

- Draw $J = j'$ with probability proportional to $w_t^{(j')}$.
- Draw $X_{t+1} = x'_{t+1}$ from $g^*_{t+1}(x_{t+1} \mid \boldsymbol{x}_t^{(j')})$ or from a reversible MCMC step with $g^*_{t+1}(\cdot \mid \boldsymbol{x}_t^{(j')})$ as its invariant distribution (see the proof of its correctness in Appendix).
- Set (J^{k+1}, x^{k+1}_{t+1}) equal to $(j', x^{'}_{t+1})$ with probability p_a, and equal to (J^k, x^k_{t+1}) with probability $1 - p_a$, where

$$p_a = \min\left\{1, \frac{p(j', x^{'}_{t+1}) w_t^{(J^k)} g^*_{t+1}(x^k_{t+1} \mid \boldsymbol{x}_t^{(J^k)})}{p(J^k, x^k_{t+1}) w_t^{(j')} g^*_{t+1}(x'_{t+1} \mid \boldsymbol{x}_t^{(j')})}\right\},$$

  where $p(J, x_{t+1})$ is defined in (7).

The resulting equilibrium distribution of (J, x_{t+1}) is exactly (7). Theoretical properties of the Hastings' chain are studied in Liu (1996) who shows that this method is comparable to rejection method in terms of statistical efficiency. The second rejection method described in the previous subsection can also take this MCMC twist. Its detail is omitted here.

The advantages of rejection methods are that no iterations are needed and the resulting sample is "exact," whereas the disadvantage is that c_{t+1} needs to be computed and the resulting scheme can be very inefficient. Liu (1996) provides more detailed comparisons of the three

methods. An interesting variation is to combine rejection and importance samplings as suggested by Liu, Chen and Wong (1997). When the difference between $\pi_{t+1}(x_{t+1} \mid \boldsymbol{x}_t)$ and $g^*_{t+1}(x_{t+1} \mid \boldsymbol{x}_t)$ is large, none of the methods is ideal. To alleviate the problem for state space model, Hürzeler and Künsch (1995) propose some smoothing techniques and Pitt and Shephard (1997) suggest using mode approximation to find a good adjustment multiplier.

3.5 Illustration with the State Space Model

Suppose one is interested in estimating the state space signal x_t *on line* (Example 2) with the parameters θ and ϕ given. For simplicity, we suppress ϕ and θ in all relevant formulas. Thus, the dynamic system for the state space model is $\pi_t(\boldsymbol{x}_t) \propto \prod_{s=1}^t f_s(y_s \mid x_s) q_s(x_s \mid x_{s-1})$, and

$$\pi_{t+1}(x_{t+1} \mid \boldsymbol{x}_t) \propto \frac{\pi_{t+1}(x_{t+1}, \boldsymbol{x}_t)}{\pi_t(\boldsymbol{x}_t)} = f_{t+1}(y_{t+1} \mid x_{t+1}) q_{t+1}(x_{t+1} \mid x_t).$$

Although sampling from $\pi_{t+1}(x_{t+1} \mid \boldsymbol{x}_t)$ can be difficult, one can usually draw from the state equation $g^*_{t+1}(x_{t+1} \mid \boldsymbol{x}_t) = q_{t+1}(x_{t+1} \mid x_t)$ easily. Rejection methods 1 and 2 are identical in this situation. Let $c_{t+1} = \sup_{x_{t+1}} f_{t+1}(y_{t+1}|x_{t+1})$. The procedure is

- Draw $J = j$ with probability proportional to $w_t^{(j)}$, then draw x_{t+1} from $q_{t+1}(x_{t+1} \mid x_t^{(j)})$.
- Accept (J, x_{t+1}) with probability $p = f_{t+1}(y_{t+1} \mid x_{t+1})/c_{t+1}$

All the samples of (J, x_{t+1}) drawn from this scheme follows the distribution

$$p(J = j, x_{t+1}) \propto w_t^{(j)} f_{t+1}(y_{t+1} \mid x_{t+1}) q_{t+1}(x_{t+1} \mid x_t^{(j)}).$$

Similarly, with $g^*_{t+1} = q_{t+1}(x_{t+1}|x_t)$, the importance resampling procedure becomes: (i) draw $J = j$ with probability proportional to $w_t^{(j)}$, (ii) draw x_{t+1} from $q_{t+1}(x_{t+1} \mid x_t^{(j)})$, and

(iii) assign weight to the sample (j, x_{t+1}) as $f_{t+1}(y_{t+1} \mid x_{t+1})$. The procedure of Gordon et al. (1995) and Kitagawa (1996) is exactly the above, with an additional step of resampling from the obtained sample using the assigned weight. In addition to the use of $g^*_{t+1} = q_{t+1}(x_{t+1}|x_t)$, Pitt and Shephard (1997) incorporate an adjustment multiplier $a^{(j)}_{t+1} = f_{t+1}(y_{t+1} \mid \mu^{(j)}_{t+1})$, where $\mu^{(j)}_{t+1}$ can be mode, mean, or other likely value of $x^{(j)}_{t+1}$. Thus, the resulting weight for the obtained sample is $w(j, x_{t+1}) = f_{t+1}(y_{t+1} \mid x^{(j)}_{t+1})/f_{t+1}(y_{t+1} \mid \mu^{(j)}_{t+1})$.

In the independence chain approach with the same g^*_{t+1} as above, the rejection probability can be computed as

$$p_a = \min\left\{1, \frac{f_{t+1}(y_{t+1} \mid x'_{t+1})w_t^{(J_k)}}{f_{t+1}(y_{t+1} \mid x^k_{t+1})w_t^{(j')}}\right\},$$

and the rest can be carried out routinely.

4 RESAMPLING AND A GENERIC ALGORITHM

In many early work on Monte Carlo methods for the state space model, resampling has played a major role in evolving the system from time t to $t+1$ (e.g. Gordon et al. 1993; Kitagawa 1996). In this section we describe two resampling methods, discuss possible resampling schedules, and then prescribe a generic Monte Carlo algorithm for dynamic systems.

4.1 Resampling methods

4.1.1 Simple Random Sampling. In this procedure, one samples from S_t with replacement with probability proportional to the weights in W_t. Liu and Chen (1995) use this approach to modify the skewed importance weights resulting from the SIS.

In general, a resampling step is inserted between two SIS steps. But when the weight (2) does not depend on x_{t+1} (i.e., when sampling distribution (3) is used), the resampling step

should be inserted *inside* a SIS step. Specifically, an optimal SISR consists of: SIS step (B)— resampling step — SIS step (A). This generates more distinct samples of x_{t+1} than performing resampling *after* SIS steps (A) and (B).

4.1.2 Residual Sampling. The following scheme can replace the simple random sampling.

- Retain $k_j = [mw_t^{(*j)}]$ copies of $\boldsymbol{x}_t^{(j)}$ for each j, where $w_t^{(*j)}$ is the renormalized weight of $w_t^{(j)}$. Let $m_r = m - k_1 - \cdots - k_m$.
- Obtain m_r iid draws from S_t with probabilities proportional to $mw_t^{(*j)} - k_j, j = 1, \ldots, m$.
- Reset the weights to 1.

It is easily shown that the residual sampling dominates the SRS in having smaller Monte Carlo variance and favorable computation time, and it does not seem to have disadvantages in other aspects. A comparison of the two procedures is given in Section 6.2.

4.1.3 Local Monte Carlo Resampling. Since the local Monte Carlo methods described in Section 3 provide samples of (J, x_{t+1}) with distribution (7), it appears that these methods achieve resampling effect automatically. More precisely, let (J^k, x_{t+1}^k), k=1, $\ldots, m^*$ be a set of draws obtained from using either a rejection method or the Hastings method (after burning) in Section 3. Then the set of streams $S'_{t+1} = \{(\boldsymbol{x}_t^{(J^k)}, x_{t+1}^k), k = 1, \ldots, m^*\}$ is a desirable resample. The weights associated with the new streams are equal to 1. Note that m^* is not necessarily equal to m. This procedure avoids weight estimation.

4.2 Resampling schedule

As shown by our earlier arguments and later examples, resampling at every stage is neither necessary nor efficient. It is thus desirable to prescribe a schedule for the resampling step to take place. Two such schedules are available: deterministic versus dynamic. In a deterministic schedule one conducts resampling at time $t_0, 2t_0, \ldots$, where t_0 depends on difficulty of a particular problem and may require some experimentation. In a dynamic schedule, one gives a sequence of thresholds $c_1, c_2, \ldots$, and monitors the coefficient of variation of the weights cv_t^2. When $cv_t^2 > c_t$ occurs, one invokes resampling. A typical sequence of c_t can be $c_t = a + bt^\alpha$.

4.3 A Generic Monte Carlo Algorithm

We recommend the following algorithm for Monte Carlo computation in dynamic systems.

Main Algorithm

1. Check the weight distribution: perform one of the following two choices at time t:

   Dynamic: If the weight (or estimated weight) distribution is not too skewed, i.e., $cv_t^2(w) < c_t$, go to step 2. Otherwise go to step 3.

   Deterministic: If $t \neq kt_0$ for some integer k, go to step 2. Otherwise go to step 3.

2. Set $t = t + 1$. Invoke an SIS step (section 2.1). Sometimes one may need a local MC procedure (Section 3) to accomplish recursive sampling and weighting. Goto step 1.

3. Set $t = t + 1$. Invoke an SISR step (Section 2.3). Use residual sampling whenever possible. To avoid weight calculation, use local Monte Carlo resampling methods. Go to step 1.

A noticeable difference between our use of local Monte Carlo procedures and that of others

(Hürzeler and Künsch 1995; Berzuini et al. 1997; Pitt and Shephard 1997) is that we decouple the local MC outputs into two parts: estimating the new weights for the $\boldsymbol{x}_t$ and obtaining the draws of x_{t+1}. There are two advantages of doing such a decoupling: (i) obtaining an explicit weighting can tell us how different π_{t+1} and π_t are and how well the SIS works; and (ii) it provides a means to improve efficiency via the use of residual sampling and delayed resampling. Since the local MC procedures are merely used to achieve a good g_{t+1}, any other means that leads to this end should be considered whenever possible.

5 RAO-BLACKWELLIZATION

In all SISR procedures, the discrete representation of $\pi_{t+s}(\boldsymbol{x}_t)$, by a sample of $\boldsymbol{x}_t^{(j)}$ with the weight $w_{t+s}^{(j)}$, degenerates very rapidly as the number of resamplings increases between t and $t+s$. As a consequence, estimating a quantity of interest, such as $E_{\pi_{t+s}}(h(\boldsymbol{x}_t))$, can be very inaccurate.

Take Example 1 for instance. As SISR proceeds with t, the number of distinctive x_0 (i.e., θ) values decreases monotonically. This rapidly leads to a degenerate posterior distribution of θ. To alleviate this problem, we can apply a variant of the Rao-Blackwellization (Casella and Robert 1996; Kong et al. 1994; Liu et al. 1994).

Suppose, with the Bayesian missing data setting of Example 1, we have at time t the observed information $\boldsymbol{y}_t$ and multiple draws $(\theta^{(j)}, \boldsymbol{x}_t^{(j)})$, $j = 1, \ldots, m$, properly weighted by $w_t^{(j)}$. If the number of distinctive values of the $\theta^{(j)}$ is too small, we can fragment each stream $(\theta^{(j)}, \boldsymbol{x}_t^{(j)})$ by drawing $\theta^{(j1)}, \ldots, \theta^{(jk)}$ from the complete-data posterior distribution $p(\theta \mid \boldsymbol{x}_t^{(j)}, \boldsymbol{y}_t)$. When the posterior distribution of θ is continuous, we will have km distinctive θ values after Rao-

Blackwellization. The weight associated with each $(\theta^{(j\ell)}, \boldsymbol{x}_t^{(j)})$ is $w_t^{(j)}$. (This is the consequence of the fact that after a few steps of MCMC transition with respect to which the target distribution π_t is invariant, the sample is still properly weighted with respect to π_t. See MacEachern et al. (1998). To retain constant total number of streams m, one can either set $k = 1$, or resample m streams from the km streams according to their corresponding weight. Rao-Blackwellization as described above results in a sampling distribution that is closer to the target distribution since it uses more information.

If at time t we want $p(\theta \mid \boldsymbol{y}_t)$, we can use the Rao-Blackwellized estimate,

$$\hat{p}(\theta \mid \boldsymbol{y}_t) = \frac{\sum_{j=1}^m w_t^{(j)} p(\theta \mid \boldsymbol{x}_t^{(j)}, \boldsymbol{y}_t)}{\sum_{j=1}^m w_t^{(j)}},$$

instead of using the weighted histogram of the sampled $\theta^{(j)}$.

To compute the likelihood function $L(\theta|\boldsymbol{y}_t)$, we can first draw θ uniformly (if the parameter space is bounded, otherwise one need to combine the flat prior with some data) and apply the SIS to draw multiple copies $(\theta^{(j)}, \boldsymbol{x}_t^{(j)})$ with weight $w^{(j)}$. Then the Rao-Blackwellized estimate of the likelihood function is:

$$\hat{L}(\theta \mid \boldsymbol{y}_t) \approx \frac{\sum_{j=1}^m w^{(j)} p(\theta \mid \boldsymbol{x}_t^{(j)}, \boldsymbol{y}_t)}{\sum_{j=1}^m w^{(j)}},$$

where $p(\theta \mid \boldsymbol{x}_s^{(j)}, \boldsymbol{y}_s)$ is the complete-data posterior distribution of θ with flat prior.

Berzuini et al. (1997) notice that when some form of conditional independence is present (e.g., in a missing data problem when a parameter θ is involved conditional on which the missing data are independent of each other), one may sometimes *"disengage"* those early observation y's and early imputations. For instance, in Example 1 $p(x_{t+1}, y_{t+1}|\theta, \boldsymbol{x}_t, \boldsymbol{y}_t) = p(x_t, y_t|\theta)$. Hence all of the $\boldsymbol{x}_t$ and $\boldsymbol{y}_t$ can be "disengaged." Similarly in Example 2, the $\boldsymbol{x}_{t-1}$ and $\boldsymbol{y}_{t-1}$ can be

"disengaged." The advantage of doing this is obvious: it saves memory and may speed up computation.

However, when disengagement is implemented, Rao-Blackwellization is no longer directly applicable. A remedy is to represent the information contained in the disengaged components as a mixture distribution of the θ (via Rao-Blackwellization) and then proceed in combination with resampling. Numerical experiment on this method is under investigation.

6 EXAMPLES

6.1 Econometric Disequilibrium Model

Initially proposed by Fair and Jaffee (1972), this class of models has attracted much attention from econometrics researchers in past few decades. It provides a theoretical foundation for the philosophical arguments (generally called *Keynesian theory*, which is named after the economist J.M. Keynes who attacked the dominant paradigm of economics in 1930s) against the postwar mainstream approach to economics, the equilibrium methods. See Quandt (1982, 1988) for reviews and discussions. Here we only look at a special dynamic disequilibrium model in Hendry and Richards (1990). Almost all components other than the relevant lagged variables, such as prices and other environmental exogenous variables, are excluded for the sake of simplicity. We illustrate an improvement in estimation by using Rao-Blackwellization.

Let $q_t'=(q_{1t}, q_{2t})$ be bivariate normal random variables with

$$E(q_{i(t+1)} \mid q_t) = \alpha_i q_{it}; \qquad Var(q_{t+1} \mid q_t) = I,$$

for $t = 0, \ldots, T-1$, where I is the identity matrix. The observed datum for this model are $y_t = \min\{q_{1t}, q_{2t}\}$, for $t = 1, \ldots, T$. For simplicity in presentation, the initial states q_{10} and

q_{20} were taken to be 0 and assumed known. Of interest is the likelihood function or posterior distribution of (α_1, α_2).

Let $\lambda_t = \max\{q_{1t}, q_{2t}\}$, let δ_t be i if $y_t = q_{it}$, and let $\theta = (\alpha_1, \alpha_2)$. If we write $x_t = (\lambda_t, \delta_t)$, the distribution involved in sequential sampling is $g_{t+1}(x_{t+1} \mid \boldsymbol{x}_t) = p(x_{t+1} \mid \theta, \boldsymbol{x}_t, \boldsymbol{y}_t, y_{t+1})$; and that involved in weight updating is $w_{t+1} \propto p(y_{t+1} \mid \theta, \boldsymbol{x}_t, \boldsymbol{y}_t)$. Detailed computations are given in the Appendix.

For each fixed θ, Hendry and Richard (1990) use the SIS to evaluate the likelihood $L(\theta \mid \boldsymbol{y}_T)$ based on equation (8) of Kong et al. (1994). Putting a flat prior on θ, we can treat the likelihood computation as a Bayesian computation and use the SIS method to simulate weighted samples of $(\theta, \boldsymbol{x})$ jointly. Rao-Blackwellization can be applied to improve the efficiency.

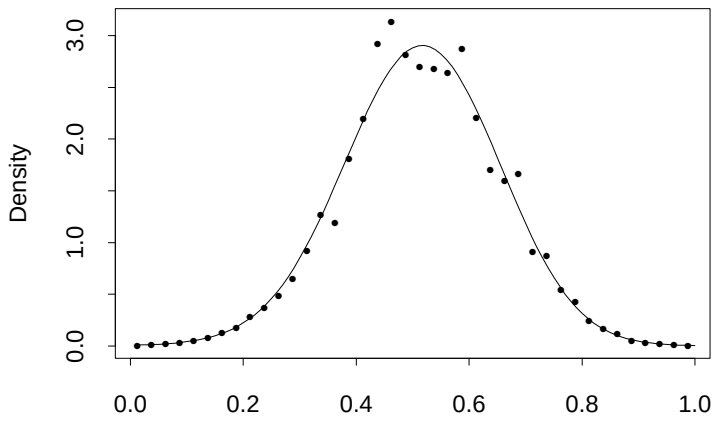

Figure 1: The posterior distribution of α_1 after the 50 observations with uniform prior. Line — result from Rao-Blackwellization; dots — result from the standard SIS.

We simulated 50 data observations $y_1, \ldots, y_{50}$ from the model with $\alpha_1 = \alpha_2 = 0.6$, and

initial value $q_{11} = q_{21} = 0$. Assuming that we know $\alpha_1 = \alpha_2 = \alpha$, we used the SIS method with $m = 10,000$ to obtain the likelihood function for α, as shown in Figure 1. It took 8.16 seconds on a Silicon Graphics workstation with R10000 microprocessor, and the cv^2 at the end of the SIS is 5.13. The smooth curve is the result from the Rao-Blackwellization.

6.2 Blind Deconvolution

The moving average system, $y_t = \sum_{i=1}^{q} \phi_i x_{t-i} + \varepsilon_t$, is often seen in digital communication. The input signal x_t takes value from a known set of discrete states and $\varepsilon_t \sim N(0, \sigma^2)$. By observing the blurred signals $y_1, \ldots, y_n$, it is of interest to reconstruct the x_t and to estimate the system coefficients ϕ_i. More references can be found in Liu and Chen (1995).

We took a simulated example from Chen and Li (1995) in which the system equation is

$$y_t = x_t + 0.8x_{t-1} - 0.4x_{t-2} + \varepsilon_t.$$

The input signals x_t were iid uniform on $\{0, 1, 3\}$. The signal-to-noise ratio was controlled at 15 dB, which gives a standard deviation 0.3 for the noise. The ϕ_i in this case can be easily integrated out with Normal prior and all the sampling and weighting calculations can be found in Liu and Chen (1995). A direct SIS without using a local MC procedure applies.

A total of 200 signal sequences were simulated, each with 200 sequential observations from the system. Our interest was in testing the simple SIS with different resampling schedules and with the two resampling methods (i.e., simple random sampling (s) versus residual sampling (r)). One thousand streams ($m = 1000$) were carried in the SIS procedure. We estimated the input signal x_t by MAP using the weight at time $t+3$. Table 1 shows the number of misclassification of signals in 200 simulations, each with 200 sequential signals. Here, resampling frequency t_0 means

procedures (s) or (r) were applied at $t_0, 2t_0, 3t_0, \ldots$ (so $t_0 = 200$ implies no resampling). For dynamic scheduling, resampling procedure is applied whenever the effective sample size (defined as $m/(1 + cv^2(w))$ is less than 3. In our example, this dynamic schedule leads to 5 to 15 times of resamplings in processing 200 signals.

Table 1 shows that resampling either too often (t_0 small) or too rare (t_0 big) tends to produce a large number of misclassifications. When resampling too often, e.g. $t_0 = 1$ or 5, there are marked frequency that the Monte Carlo method is never on the right track, resulting in disastrous estimations. In the reasonable range of t_0 (between 20 and 50), residual sampling method seems to be slightly better than the simple random sampling.

	Deterministic Resampling Schedule t_0												dynamic	
	1		5		20		50		100		200		schedule	
error	s	r	s	r	s	r	s	r	s	r	s	r	s	r
0-2	11	5	7	13	13	13	7	10	1	0	0	0	11	12
3-5	49	49	46	53	61	65	53	49	28	28	7	7	69	58
6-8	41	43	50	52	72	70	57	58	59	58	12	12	66	67
9-11	23	20	27	30	38	38	52	48	43	44	47	47	29	41
12-15	10	9	13	7	8	6	17	20	33	32	44	44	16	8
16-25	11	10	14	11	8	8	14	15	35	35	84	84	6	11
16-50	4	10	8	9	0	0	0	0	1	3	6	6	1	1
>50	51	54	35	25	0	0	0	0	0	0	0	0	2	2

Table 1: The numbers of misclassified signals (the first column) in a total of 200 simulations, each with 200 sequential signals, are demonstrated. All the columns except column 1 are results from different combinations of SIS strategies. Symbols "s" and "r" on row 3 represent *simple random sampling* and *residual sampling* respectively.

6.3 Target Tracking in Clutter

Tracking multiple targets in clutter is of interest to engineers and computer scientists. The problem has received much attention recently and many solutions have been proposed, among which the method of Gordon et al. (1995) and Avitzour (1995) is most closely related to the method described in this article. As has been mentioned earlier, their algorithm employ the sampling distribution (5). Here we use the example in Avitzour (1995) to show that using sampling distribution (3) can produce better tracking results.

The tracking problem in Avitzour (1995) can be formulated as a state space model with the state variable $x_t=(x_t^{(1)}, x_t^{(2)})$, where $x_t^{(1)}$ is the location of the target on a straight line and $x_t^{(2)}$ is the target velocity. The z_t are location observations. They evolve in the following way:

$$\begin{aligned} x_{t+1}^{(1)} &= x_t^{(1)} + x_t^{(2)} + \frac{1}{2}w(t+1), \\ x_{t+1}^{(2)} &= x_t^{(2)} + w(t+1), \\ z_{t+1} &= x_{t+1}^{(1)} + v(t+1), \end{aligned}$$

where $w(t) \sim N(0, q^2)$ and $v(t) \sim N(0, r^2)$ are independent. We further assume that we only have probability p_d to make the location observation z_t. The rate of false signal clutter is $\alpha\Delta$, with Δ being the width of $4r$ detection region. Therefore, the actual observation y_t is a vector of length m_t among which at most one is the true observation. The distribution of m_t is Bernoulli(p_d)+Poisson($\lambda\Delta$). The false signals are uniformly distributed in the detection region.

In this case, the sampling distribution $\pi_{t+1}(x_{t+1} \mid y_{t+1}, \boldsymbol{x}_t)$ in (3) can be easily shown to be a mixture of m_{t+1} normal distributions, with means and variances being functions of y_{t+1} and x_t. More details are given in the appendix. Resampling is conducted *before* each x_{t+1} is drawn, and concurrent estimation of $E_{\pi_{t+1}}(x_{t+1})$ is done using Rao-Blackwellization (6).

Another trick that we can play with this example is to integrate out the state variable x_t and use Monte Carlo to impute an indicator variable that tells which component of y_t is the true signal. With the true signal identified, it is trivial to estimate the true location of the target. This *collapsing procedure* produces an even better result.

Figure 2 shows the plots of tracking errors (estimated location − true location) of 50 simulated runs, with $r^2 = 1.0$, $q^2 = 0.1$, $p_d = 0.9$ and $\lambda = 0.1$. Five hundred streams (m=500) were used, with resampling done at every step. The top figure resulted from using the optimal sampling distribution (3) and the middle figure from using the collapsing procedure. The bottom figure shows the result from using a less optimal sampling distribution $g_{t+1} = q(x_{t+1} \mid x_t)$ as in Avitzour (1995). The top figure has 13 runs with absolute value of tracking errors exceeding 10 at least once, the middle figure has 16, and the bottom has 20. Similarly, the top figure has 4 runs exceeding the 20 limit, the middle figure has 4, the bottom has 8.

The above parameter combination is slightly different from that of Avitzour (1995), with smaller clutter density but larger state equation variance. With their configuration, the results are similar but the differences between different procedures are smaller.

7 SUMMARY

In this paper we propose a general framework for *on-line* Monte Carlo computations for

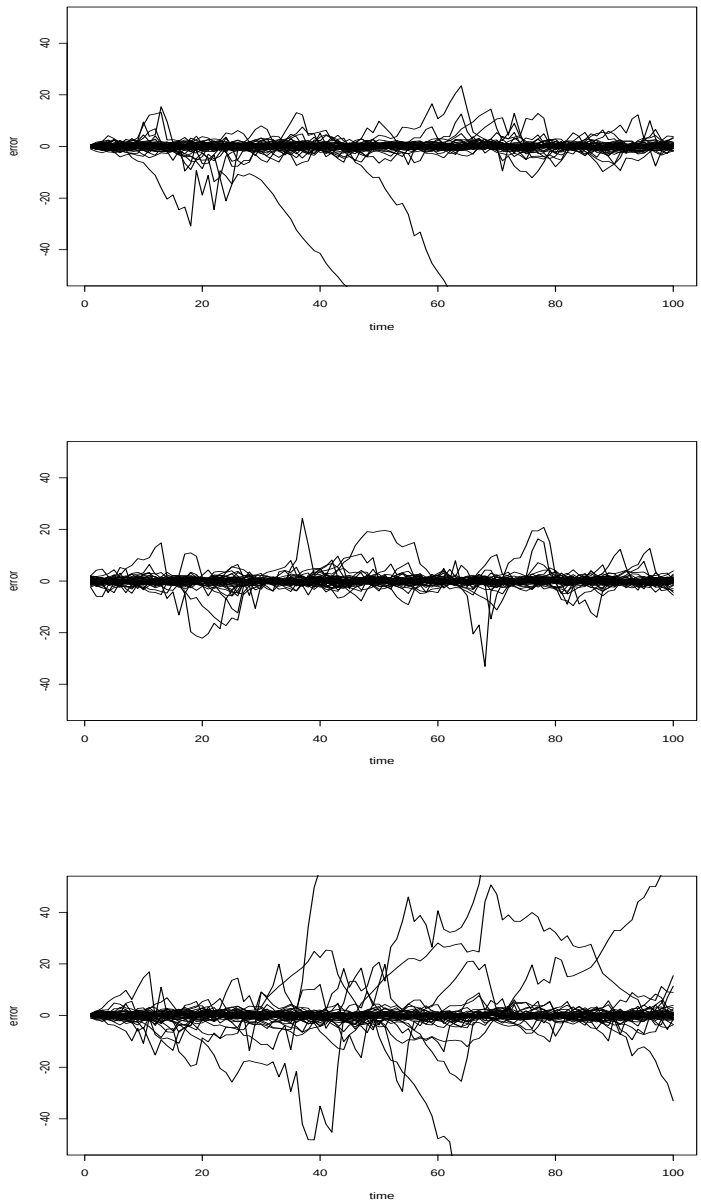

Figure 2: The tracking results from using three different sequential Monte Carlo methods. We used m=500 and resampled at every step. The y-axis is the distance between the estimated and true positions of the target. Top: results from using g_{t+1} prescribed by (3); middle: results from using the collapsing procedure; bottom: results from using (5).

dynamic systems. It is clear that almost all the available sequential Monte Carlo procedures can be unified under this framework. This general setting provides a common ground for understanding and improving various similar methods developed for specific models. It also provides a general guidance on how such procedures should be used in practice and how different 'tricks' developed for specific problems can be combined to achieve maximum efficiency. In particular, we discussed several key issues in implementing sequential Monte Carlo methods, namely, choices of the recursive sampling distribution g_{t+1}, advantages and disadvantages of resampling and their scheduling, efficient use of Monte Carlo samples, and Rao-Blackwellization.

Besides the obvious application of the sequential MC in the state space models, there are many other problems that can be formulated as a dynamic system and solved using techniques described in this article. For example, the SIS procedure can be built into a MCMC scheme to produce a more efficient transition proposal chain. The Hastings' rejection procedure described in Section 8.1 can be used in combination. This type of Monte Carlo methods (sometimes called "configuration-biased Monte Carlo") have been tested effective for simulating biopolymers (Leach, 1996). See Irwing et al. (1994), Kong et al. (1994), Wong and Liang (1997) and others for more examples. We hope that the results reported here can stimulate more interest and effort from other researchers on this type of problems.

8 APPENDIX

8.1 The Invariant Distributions of the Hastings Independence Chain

This scheme is first discussed by Hastings (1970, Section 2.5) as one way to do importance sampling. Tierney (1991) generalizes the discussion under the heading "independence chains,"

and is called "Metropolized independence sampling" by Liu (1996). The general scheme can be stated as follows.

Suppose that $\pi(x)$ is known up to a normalizing constant, and we are able to draw independent samples from $g(x)$. A Markov chain $\{X_1, X_2, \ldots\}$ can be constructed with the transition function

$$K(x,y) = \begin{cases} g(y)\min\{1, \frac{w(y)}{w(x)}\}, & \text{if } y \neq x, \\ 1 - \int_{z \neq x} g(z)\min\{1, \frac{w(z)}{w(x)}\}dz, & \text{if } y = x, \end{cases}$$

where $w(x) = \pi(x)/g(x)$ is called the *importance ratio* (or *importance weight*). Intuitively, the transition from $X_n = x$ to $X_{n+1} = y$ is accomplished by generating an independent sample from $g(\cdot)$, and then "thinning" it down based on a comparison of the corresponding importance ratios $w(y)$ and $w(x)$. It can be shown that π is an invariant distribution of the constructed Markov chain. Note that the above scheme is only a special example, that more serious Metropolis-Hastings algorithms most commonly make dependent local moves instead of independent global jumps. An eigen-analysis of this chain is provided in Liu (1996).

Suppose we can not directly sample from $g(x)$ but have a reversible MCMC procedure (most single-step MCMC scheme satisfies this condition), with transition function $A(x,y)$, that has $g(x)$ as its invariant distribution. Then we have $g(x)A(x,y) = g(y)A(y,x)$. Hence, if we conduct a Metropolis step with $A(x,y)$ as the proposal chain and $\pi(x)$ as the target distribution, the rejection probability can be computed as

$$p_a = \min\left\{1, \frac{\pi(y)A(y,x)}{\pi(x)A(x,y)}\right\} = \min\left\{1, \frac{\pi(y)g(x)}{\pi(x)g(y)}\right\} = \min\left\{1, \frac{w(y)}{w(x)}\right\}.$$

Hence, the procedure described in Section 3.4 is still valid, but can no longer be called "independence chain" approach.

8.2 Computations Involved in the Example (section 6.1)

For computing the weights, we have to compute

$$p(y_t \mid \theta, \boldsymbol{x}_{t-1}, \boldsymbol{y}_{t-1}) = \phi(y_t - \alpha_1 q_{1(t-1)})[1 - \Phi(y_t - \alpha_2 q_{2(t-1)})] + \phi(y_t - \alpha_2 q_{2(t-1)})[1 - \Phi(y_t - \alpha_1 q_{1(t-1)})],$$

and for imputing missing data, we need

$$\begin{aligned} p(\delta_t = 1 \mid \theta, \boldsymbol{x}_{t-1}, \boldsymbol{y}_{t-1}, y_t) &= \frac{\phi(y_t - \alpha_1 q_{1(t-1)})[1 - \Phi(y_t - \alpha_2 q_{2(t-1)})]}{p(y_t \mid \theta, \boldsymbol{x}_{t-1}, \boldsymbol{y}_{t-1})} \\ p(\lambda_t \mid \delta_t = 1, y_t, \theta, \boldsymbol{x}_{t-1}, \boldsymbol{y}_{t-1}) &= \frac{\phi(\lambda_t - \alpha_2 q_{2(t-1)})}{1 - \Phi(y_t - \alpha_2 q_{2(t-1)})} \\ p(\lambda_t \mid \delta_t = 2, y_t, \theta, \boldsymbol{x}_{t-1}, \boldsymbol{y}_{t-1}) &= \frac{\phi(\lambda_t - \alpha_1 q_{1(t-1)})}{1 - \Phi(y_t - \alpha_1 q_{1(t-1)})} \end{aligned}$$

Suppose that the prior distribution for θ is $p_0(\alpha_1, \alpha_2)$, then given complete observations, the posterior

$$p(\theta \mid q_1, \ldots, q_T) \propto p_0(\theta) \exp\left\{ - \sum_{t=2}^{T} \frac{(q_{1t} - \alpha_1 q_{1(t-1)})^2 + (q_{2t} - \alpha_2 q_{2(t-1)})^2}{2} \right\}.$$

Without loss of generality, we take $p_0(\theta)$ to be uniform on $[0, 1]^2$. Then the posterior distribution is simplified as

$$p(\theta \mid q_1, \ldots, q_T) \propto \exp\left\{ - \frac{(\alpha_1 - a_1)^2}{2b_1^2} - \frac{(\alpha_2 - a_2)^2}{2b_2^2} \right\}, \quad 0 \le a_1, a_2 < 1,$$

where

$$\begin{aligned} b_1 &= (\sum_{t=2}^{T} q_{1(t-1)}^2)^{-1/2}, & a_1 &= b_1^2 \sum_{t=2}^{T} q_{1t} q_{1(t-1)} \\ b_2 &= (\sum_{t=2}^{T} q_{2(t-1)}^2)^{-1/2}, & a_2 &= b_2^2 \sum_{t=2}^{T} q_{2t} q_{2(t-1)}. \end{aligned}$$

To sample from the truncated normal distribution $X \sim \phi(x)$ with $X > c$, where $\pi(x)$ is standard normal density, we use the following strategy. When $c < 0$, we simply conduct a

simple normal random number generation, and do rejection until we have a sample satisfying $X > c$. For $c > 0$, especially when c is big, we use exponential random variable with the rejection method.

Suppose that the exponential distribution $\lambda_0 e^{-\lambda_0 x}$ is to be used as an envelop function, then we need to find the minimum constant b so that

$$\frac{\phi(x+c)}{1-\Phi(c)} \leq b\lambda_0 e^{-\lambda_0 x}, \quad x \geq 0$$

This gives us the best solution for b:

$$b = \frac{\exp\{(\lambda_0^2 - 2\lambda_0 c)/2\}}{\sqrt{2\pi}\lambda_0(1-\Phi(c))}.$$

The acceptance rate for using this exponential distribution is then $1/b$. To achieve minimum rejection rate, we further find that the best choice for λ_0 is

$$\lambda_0 = (c + \sqrt{c^2+4})/2.$$

With this choice of λ_0 and b, we implement the rejection method of von Neumann (1951). The rejection rate for this scheme decreases as c increases, and this rate is very small for moderate to large c (e.g., for $c = 0, 1, 2$, the rejection rates are 0.24, 0.12, and 0.07).

8.3 Computations for Target Tracking (section 6.3)

Let $y_{t+1} = (y_{t+1(1)}, \ldots, y_{t+1(m_{t+1})})$ be the observed signals at time $t+1$. Then

$$\begin{aligned} f(y_{t+1} \mid x_{t+1}) &= (1-p_d)\left[\frac{1}{\Delta}^{m_{t+1}} \frac{e^{-\lambda\Delta}(\lambda\Delta)^{m_{t+1}}}{m_{t+1}!}\right] \\ &\quad + p_d\left[\frac{1}{m_{t+1}}\sum_{i=1}^{m_{t+1}} \frac{1}{\Delta^{m_{t+1}-1}}\phi_r(y_{t+1(i)} \mid x_{t+1})\right]\frac{e^{-\lambda\Delta}(\lambda\Delta)^{m_{t+1}-1}}{(m_{t+1}-1)!} \\ &= \left[p_d\sum_{i=1}^{m_{t+1}}\phi_r(y_{t+1(i)} \mid x_{t+1}) + (1-p_d)\lambda\right]\frac{e^{-\lambda\Delta}\lambda^{m_{t+1}-1}}{m_{t+1}!} \end{aligned}$$

for $t = 0, 1, 2, \ldots$, where ϕ_r refers to the normal density with variance r^2. Furthermore

$$\begin{aligned}\phi_r(y_{t+1(i)} \mid x_{t+1}) q(x_{t+1} \mid \boldsymbol{x}_t) &= \frac{1}{\pi q r} \exp\left\{ -\frac{(y_{t+1(i)} - x_{t+1}^{(1)})^2}{2r^2} - \frac{(x_{t+1}^{(1)} - x_t^{(1)} - x_t^{(2)})^2}{2(q/2)^2} \right\} \\ &= \frac{c}{\sqrt{2\pi}\sigma_{t+1}} \exp\left\{ -\frac{(x_{t+1}^{(1)} - \mu_{t+1})^2}{2\sigma_{t+1}^2} \right\}\end{aligned}$$

where

$$\sigma_{t+1}^2 = \frac{r^2 q^2}{q^2 + 4r^2}, \quad \mu_{t+1} = \left[\frac{y_{t+1(i)}}{r^2} + \frac{x_t^{(1)} + x_t^{(2)}}{(q/2)^2} \right] \sigma_{t+1}^2$$

and

$$c = \frac{2\sigma_{t+1}}{\sqrt{2\pi} r q} \exp\left\{ -\frac{1}{2} \left[\frac{y_{t+1(i)}^2}{r^2} + \frac{x_t^{(1)} + x_t^{(2)}}{(q/2)^2} - \frac{\mu_{t+1}^2}{\sigma_{t+1}^2} \right] \right\}$$

Hence, $f(y_{t+1} \mid x_{t+1}) q(x_{t+1} \mid \boldsymbol{x}_t)$ is a mixture of m_{t+1} normal distribution.

REFERENCES

Avitzour, D. (1995), A stochastic simulation Bayesian approach to multitarget tracking, *IEE Proceedings on Radar, Sonar and Navigation*, **142**, 41-44.

Berzuini, C., Best, N.G., Gilks, W.R., and Larizza, C. (1996), "Dynamic conditional independence models and Markov chain Monte Carlo methods," *J. Amer. Statist. Assoc.*, to appear.

Carlin, B.P, Polson, N. G. and Stoffer, D. S. (1992), "A Monte Carlo approach to nonnormal and nonlinear state-space modeling," *Journal of the American Statistical Association*, **87** 493-500.

Carter, C.K. and Kohn, R. (1994), "On Gibbs sampling for state space models," *Biometrika*, **81**, 541-553.

Casella, G. (1997), "Statistical inference and Monte Carlo algorithms," *Test*, **5**, 249-344.

Casella, G. and Robert, C.P. (1996), "Rao-Blackwellization of sampling schemes," *Biometrika* **83**, 81-94.

Chen, R. and Li, T. (1995) "Blind Restoration of Linearly Degraded Discrete Signals by Gibbs Sampler," *IEEE Transactions on Signal Processing*, **43**, 2410-2413.

Churchill, G.A. (1989), "Stochastic Models for Heterogeneous DNA Sequences," *Bulletin of Mathematical Biology* **51**, 79-94.

Fair, R.C. and Jaffee, D.M. (1972), "Methods of estimation for markets disequilibrium," *Econometrica* **40**, 497-514.

Gelfand, A.E. and Smith, A.F.M. (1990), "Sampling-based Approaches to Calculating Marginal Densities," *Journal of the American Statistical Association,* **85**, 398–409.

Hastings, W.K. (1970), "Monte Carlo sampling methods using Markov chains and their applications," *Biometrika*, **57**, 97-109.

Gordon, N.J., Salmon, D.J. and Smith, A.F.M. (1993), "A novel approach to nonlinear/non Gaussian Bayesian state estimation," *IEE Proceedings on Radar and Signal Processing*, **140**, 107-113.

Gordon, N.J., Salmon, D.J. and Ewing, C.M. (1995), "Bayesian state estimation for tracking and guidance using the bootstrap filter," *AIAA Journal of Guidance, Control and Dynamics*, **18**, 1434-1443.

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

Hendry, D.F. and Richard, J-F.(1991), "Likelihood evaluation for dynamic latent variables models." In *Computational Economics and Econometrics,* Ch.1. Amman,H.M.,Belsley,D.A. and Pau,L.F. (eds). Dordrecht: Kluwer

Hürzeler, M. and Künsch, H.R. (1995), "Monte Carlo approximations for general state space models." *Research Report 73*, ETH, Zürich.

Isard, M. and Blake, A. (1996), "Contour tracking by stochastic propagation of conditional density", in *Computer Vision - ECCV' 96*, B. Buxton and R. Cipolla (eds), Springer: New York.

Irwing, M., Cox, N. and Kong, A. (1994), "Sequential imputation for multilocus linkage analysis," *Proceedings of the National Academy of Science, USA*, **91**, 11684-11688.

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

Kong, A., Liu, J.S., and Wong, W.H. (1994), "Sequential imputations and Bayesian missing data problems," *J. Amer. Statist. Assoc.*, **89**, 278-288.

Krogh, A., Brown, M., Mian, S., Sjolander, K. and Haussler, D (1994), "Protein Modeling Using Hidden Markov Models," *Journal of Molecular Biology* **235**, 1501-1531.

Leach, A.R. (1996), *Molecular Modelling: Principles and Applications.* Addison Wesley Longman: Singapore.

Liu, J.S. (1996), "Metropolized independent sampling with comparisons to rejection sampling and importance sampling," *Statistics and Computing*, **6**, 113-119.

Liu, J.S. and Chen, R. (1995), "Blind deconvolution via sequential imputations," *Journal of the American Statistical Association*, **90**, 567-576.

Liu, J.S., Chen, R., and Wong, W.H. (1996), "Rejection control for sequential importance sampling," *Technical Report*, Department of Statistics, Stanford University.

Liu, J.S., Neuwald, A.F., and Lawrence, C.E. (1997), "Markov structures in biological sequence alignments," *Technical Report*, Stanford University.

Liu, J.S., Wong, W.H., and Kong, A. (1994), "Covariance structure of the Gibbs sampler with applications to the comparisons of estimators and augmentation schemes," *Biometrika*, **81**, 27-40.

MacEachern, S.N., Clyde, M.A., and Liu, J.S. (1998), "Sequential Importance Sampling for Nonparametric Bayes Models: The Next Generation," *Canadian Journal of Statistics*, in press.

von Neumann, J. (1951), "Various techniques used in connection with random digits," *National Bureau of Standards Applied Mathematics Series*, **12**, 36-38.

Pitt, M.K. and Shephard, N. (1997), "Filtering via simulation: auxiliary particle filters," *preprint: www.nuff.ox.ac.uk/users/shephard.*

Quandt, R.E. (1982), "Econometric disequilibrium models (with discussions)," *Econometric Review* **1**, 1-96.

— (1988), *The Econometrics of Disequilibrium*, New York: Basil Blackwell.

Rabiner, L.R. (1989), "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *Proceedings of the IEEE*, **77**, 257-286.

Rubin, D.B. (1987), *Multiple Imputation for Nonresponse in Surveys*, New York: Wiley.

Spiegelhalter, D.J. and Lauritzen, S.L. (1990), "Sequential Updating of Conditional Probabilities on Directed Graphical Structures," *Network*, **20**, 579-605.

Tanner, M. A. and Wong, W.H. (1987), "The calculation of posterior distributions by data augmentation (with discussion)," *Journal of the American Statistical Association* **82**, 528-550.

Tierney, L. (1994), "Markov chains for exploring posterior distributions (with discussion)," *Annals of Statistics* **22**, 1701-1762.

Vasquez, M. and Scherago, H.A. (1985), "Use of Build-up and Energy-Minimization Procedures to Compute Low-Energy Structures of the Backbone of Enkephalin," *Biopolymers*, **24**, 1437-1447.

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

West, M. and Harrison, J. (1989), *Bayesian forecasting and dynamic models*, New York: Springer-Verlag.

Wong, W.H. and Liang, F. (1997), "Dynamic Importance Weighting in Monte Carlo and Optimization," *Proceedings of the National Academy of Science*, to appear.