On-line Bayesian methods for estimation of non-linear non-Gaussian signals

Simon Godsill

Signal Processing Group, Cambridge University
sjg@eng.cam.ac.uk

September 26, 2002

Figure 1: Filtering densites evolving through time (blue line is true state value)

INTRODUCTION

In many signal processing applications it is required to estimate a latent or 'hidden' process (the 'state' of the system) from noisy, convolved or non-linearly distorted observations. Since data also arrive sequentially in many applications it is therefore desirable (or essential) to estimate the hidden process on-line, in order to avoid memory storage of huge datasets and to make inferences and decisions in real time. Some typical applications from the engineering perspective include:

- Tracking for radar and sonar applications
- Real-time enhancement of speech and audio signals
- Sequence and channel estimation in digital communications channels
- Medical monitoring of patient eeg/ecg signals
- Image sequence tracking

In this tutorial we will consider sequential estimation in such applications. Only when the system is linear and Gaussian can exact estimation be performed, using the classical Kalman filter. I will present a succinct derivation of the Kalman filter, based on Bayesian updating of probability models. In most applications, however, there are elements of non-Gaussianity and/or non-linearity which make analytical computations impossible. Here we must adopt numerical strategies. I will consider a powerful class of Monte Carlo filters, known generically as *particle filters*, which are well-adapted to general problems in this category. Worked examples will be given for several simple modelling scenarios.

OVERVIEW

- General summary of Bayesian methods
- State space models, filtering and smoothing
- The Kalman filter
- The extended Kalman filter
- The particle filter
- Developments and conclusions

BAYESIAN INFERENCE - OVERVIEW

Observations: $\mathbf{y}$

Quantity of interest: $\mathbf{x}$

Other parameters/unknowns: $\boldsymbol{\theta}$

Likelihood:

$$p(\,\mathbf{y}\,|\,\mathbf{x},\ \boldsymbol{\theta}\,)$$

Joint posterior for all unknowns (by Bayes Rule):

$$p(\,\mathbf{x},\ \boldsymbol{\theta}\,|\,\mathbf{y}\,)\ =\ \frac{p(\,\mathbf{y}\,|\,\mathbf{x},\ \boldsymbol{\theta}\,)\,p(\,\mathbf{x},\ \boldsymbol{\theta}\,)}{p(\mathbf{y})}$$

Marginal posterior for $\mathbf{x}$:

$$p(\,\mathbf{x}\,|\,\mathbf{y}\,)\ =\ \int_{\boldsymbol{\theta}} p(\,\mathbf{x},\ \boldsymbol{\theta}\,|\,\mathbf{y}\,)\,d\boldsymbol{\theta}$$

Marginal posterior is used for inference about the quantity of interest, $\mathbf{x}$.

STATE SPACE MODELS

First define the notations used. We will consider a very general class of time series models, the *state space model*. Almost all models of practical utility can be represented within this category, using a state vector of finite dimension. The sequential inference methods presented can readily be extended beyond the Markovian state space models given here, but for simplicity we retain the standard Markovian setup.

Note: from here on column vectors are denoted in standard typeface, e.g. x_t, and matrices are denoted by capitals, e.g. B. This avoids some cumbersome heavy typeface notations.

- Consider a time series with states x_t, $t \in \{0, 2, ..., T\}$.
- The states evolve in time according to a probability model. Assume a Markov structure, i.e.

$$p(x_{t+1}|x_0, x_1, ..., x_t) = f(x_{t+1}|x_t) \tag{1}$$

- The states are 'partially' observed through a likelihood function for observations $\{y_t\}$ which are assumed iid given the states, i.e.

$$p(y_{t+1}|x_0, x_1, ..., x_t, x_{t+1}, y_0, y_1, ..., y_t) = g(y_{t+1}|x_{t+1}) \tag{2}$$

Summarise as a 'state space' or 'dynamical' model:

$$
\begin{aligned}
x_{t+1} &\overset{\text{iid}}{\sim} f(x_{t+1}|x_t) && \text{State evolution density} \\
y_{t+1} &\overset{\text{iid}}{\sim} g(y_{t+1}|x_{t+1}) && \text{Observation density}
\end{aligned} \tag{3}
$$

Joint density can be expressed using the probability chain rule:

$$
p\left(x_{0:t}, y_{0:t}\right) = f(x_0) \prod_{i=1}^{t} f(x_t|x_{t-1}) \prod_{i=0}^{t} g(y_t|x_t)
$$

where $f(x_0)$ is the distribution of the initial state, $x_{0:t} \triangleq (x_0, \ldots, x_t)$ and $y_{0:t} \triangleq (y_0, \ldots, y_t)$.

EXAMPLE: LINEAR AR MODEL OBSERVED IN NOISE

$$z_t = \sum_{i=1}^{P} a_i z_{t-i} + e_t$$
$$y_t = z_t + w_t$$

with e_t and w_t independently distributed as zero mean Gaussians with variance σ_e^2 and σ_w^2, respectively (fixed and known).

a_i are the AR coefficients, of order P, also assumed here to be fixed and known.

We observe the noisy signal y_t.

The only unknown here is the signal z_t.

Form state vector as:

$$x_t = [z_t, z_{t-1}, \ldots, z_{t-P+1}]^T \tag{4}$$

Then a state space model in terms of the signal values is obtained as:

$$x_t = Ax_{t-1} + e_t \tag{5}$$

$$y_t = Bx_t + w_t \tag{6}$$

where:

$$A = \begin{bmatrix} a_1 & a_2 & \dots & a_P \\ 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \end{bmatrix} \tag{7}$$

$$B = \begin{matrix} 1 & 0 & 0 & \dots & 0 \end{matrix} \tag{8}$$

Alternatively, in terms of state evolution and observation densities:

$$\begin{aligned} f(x_{t+1}|x_t) &= \mathcal{N}(x_{t+1}|Ax_t, \sigma_e^2) \\ g(y_t|x_t) &= \mathcal{N}(y_t|Bx_t, \sigma_w^2) \end{aligned} \tag{9}$$

This is an example of the linear Gaussian state space model, an important special case that is used extensively to construct algorithms in the nonlinear non-Gaussian case (extended Kalman filters, Rao-Blackwellised particle filters, ...).

EXAMPLE: NON-LINEAR MODEL:

$$
\begin{aligned}
x_t &= A(x_{t-1}) + v_t \\
&= \frac{x_{t-1}}{2} + 25\frac{x_{t-1}}{1 + x_{t-1}^2} + 8\cos(1.2t) + v_t \\
y_t &= B(x_t) + w_t \\
&= \frac{(x_t)^2}{20} + w_t
\end{aligned}
$$

where $v_t \sim \mathcal{N}(0, \sigma_v^2)$ and $w_t \sim \mathcal{N}(0, \sigma_w^2)$.

This may be expressed in terms of density functions as:

$$
f(x_{t+1}|x_t) = \mathcal{N}(x_{t+1}|A(x_t), \sigma_v^2)
$$

$$
g(y_t|x_t) = \mathcal{N}(y_t|B(x_t), \sigma_w^2)
$$

INFERENCE TASKS

Observe data:

$$y_{0:t} \triangleq (y_0, \ldots, y_t)$$

Wish to infer the 'hidden states':

$$x_{0:t} \triangleq (x_0, \ldots, x_t)$$

Fundamental inference tasks:

- Filtering:

$$p(x_t | y_{0:t})$$

- Smoothing ('fixed lag'):

$$p(x_{t-p:t} | y_{0:t})$$

- Smoothing ('fixed interval'):

$$p(x_{0:T} | y_{0:T})$$

FILTERING

Wish to estimate $p(x_t|y_{0:t})$ itself or expectations of the form

$$\overline{h} = \mathbb{E}h(x_t) = \int h(x_t)p(x_t|y_{0:t})dx_t$$

e.g. $h(x_t) = x_t$ - posterior mean estimation (MMSE estimator)

Suppose we have $p(x_t|y_{0:t})$ but wish to find $p(x_{t+1}|y_{0:t+1})$. In principle we can use the *filtering recursions:*

$$\begin{aligned} p(x_{t+1}|y_{0:t}) &= \int p(x_t, x_{t+1}|y_{0:t})dx_t \\ &= \int p(x_t|y_{0:t})p(x_{t+1}|x_t, y_{0:t})dx_t \\ &= \int p(x_t|y_{0:t})f(x_{t+1}|x_t)dx_t \qquad (10) \end{aligned}$$

$$p(x_{t+1}|y_{0:t+1}) = \frac{g(y_{t+1}|x_{t+1})p(x_{t+1}|y_{0:t})}{p(y_{t+1}|y_{0:t})} \qquad (11)$$

Time	$t-1$		t		$t+1$	$\ldots$
Data	y_{t-1}		y_t		y_{t+1}	
Filtering	$p(x_{t-1}\|y_{0:t-1})$		$p(x_t\|y_{0:t})$		$p(x_t\|y_{0:t+1})$	
Prediction		$p(x_t\|y_{0:t-1})$		$p(x_{t+1}\|y_{0:t})$		

However, in the general case the integral is intractable and approximations must be used. (x_t high-dimensional, $f()$, $g()$ non-Gaussian, ...)

MOTIVATING EXAMPLES - EVOLVING DATASETS

- Tracking applications in radar, sonar, etc.
  - x_t - cartesian/polar coordinates of a target
  - y_t - noisy observations of a non-linear function of x_t. e.g. $y_t = \arg(x_t) + v_t$ ('Bearings only tracking').
  - $f(x_{t+1} | x_t)$ - determined by the dynamics of the target - e.g. constant acceleration.
  - Need to estimate target position x_t.
- Finance - stock prices, exchange rates arrive sequentially. Need to update portfolios on line as more data emerges.
- Medical Monitoring - on-line monitoring of eeg/ecg data for sick patients.
- Digital communications
- Speech recognition and processing.

LINEAR GAUSSIAN MODELS - THE KALMAN FILTER

(Harvey, 1989; Anderson and Moore, 1979)

- In cases where the state space model is linear and Gaussian, the classic Kalman filter can be applied. In this case we have:

$$\begin{aligned} f(x_{t+1}|x_t) &= \mathcal{N}(x_{t+1}|Ax_t, C) \\ g(y_t|x_t) &= \mathcal{N}(y_t|Bx_t, D) \end{aligned} \tag{12}$$

  where $\mathcal{N}(x|\mu, Q)$ is the Gaussian density function with mean vector μ and covariance matrix Q.

- We can write this equivalently as:

$$x_{t+1} = Ax_t + v_t \tag{13}$$

$$y_t = Bx_t + w_t \tag{14}$$

  where v_t and w_t are zero mean Gaussian vectors with covariance matrices C and D, respectively. v_t and w_t are independent over time and also independent of one another.

- We also require that the initial state be Gaussian distributed:

$$p(x_0) = \mathcal{N}(x_0|\mu_0, P_0)$$

- We first require $p(x_{t+1}|y_{0:t})$, the prediction step from the above filtering recursion:

$$p(x_{t+1}|y_{0:t}) = \int p(x_t|y_{0:t}) f(x_{t+1}|x_t) dx_t$$

- Suppose that we have already that at time t:

$$p(x_t|y_{0:t}) = \mathcal{N}(x_t|\mu_t, P_t)$$

- Now, from 13 we have

$$x_{t+1} = Ax_t + v_t$$

Thus from standard change of variables theory (linear Gaussian case) we have:

$$p(x_{t+1}|y_{0:t}) = \mathcal{N}(x_{t+1}|\mu_{t+1|t}, P_{t+1|t})$$

where:

$$\mu_{t+1|t} = A\mu_t,\ \ P_{t+1|t} = C + AP_tA^T \tag{15}$$

- Now, the correction step of the above filtering recursion is

$$p(x_{t+1}|y_{0:t+1}) = \frac{g(y_{t+1}|x_{t+1})p(x_{t+1}|y_{0:t})}{p(y_{t+1}|y_{0:t})} \tag{16}$$

- Substituting the above Gaussian forms into the numerator gives:

$$\begin{aligned}
p(x_{t+1}|y_{0:t+1}) &\propto \mathcal{N}(y_{t+1}|Bx_{t+1}, D)\mathcal{N}(x_{t+1}|\mu_{t+1|t}, P_{t+1|t}) \\
&\propto \exp(-\frac{1}{2}\{[y_{t+1} - Bx_{t+1}]^T D^{-1}[y_{t+1} - Bx_{t+1}]\}) \\
&\qquad \times \ \exp(-\frac{1}{2}\{[x_{t+1} - \mu_{t+1|t}]^T P_{t+1|t}^{-1}[x_{t+1} - \mu_{t+1|t}]\}) \\
&\propto \exp(-\frac{1}{2}\{[x_{t+1} - \mu_{t+1}]^T P_{t+1}^{-1}[x_{t+1} - \mu_{t+1}]\}) \\
&= \mathcal{N}(x_{t+1}|\mu_{t+1}, P_{t+1})
\end{aligned}$$

where

$$\mu_{t+1} = P_{t+1}(B^T D^{-1} y_{t+1} + P_{t+1|t}^{-1} \mu_{t+1|t}), \qquad \text{and} \qquad P_{t+1} = (B^T D^{-1} B + P_{t+1|t})^{-1}$$

- This expression can be rearranged using the matrix inversion lemma to give:

$$\mu_{t+1} = \mu_{t+1|t} + K_t(y_{t+1} - B\mu_{t+1|t}), \qquad \text{and} \qquad P_{t+1} = (I - K_t B) P_{t+1|t}$$

where

$$K_t = P_{t+1|t} B^T (B P_{t+1|t} B^T + D)^{-1}$$

- Hence the whole Kalman filtering recursion can be summarised as:

$$\mu_{t+1|t} = A\mu_t \tag{17}$$

$$P_{t+1|t} = C + AP_tA^T \tag{18}$$

$$\mu_{t+1} = \mu_{t+1|t} + K_t(y_{t+1} - B\mu_{t+1|t}) \tag{19}$$

$$P_{t+1} = (I - K_tB)P_{t+1|t} \tag{20}$$

$$K_t = P_{t+1|t}B^T(BP_{t+1|t}B^T + D)^{-1} \tag{21}$$

THINGS YOU CAN DO WITH A KALMAN FILTER

The Kalman filter is a fundamental tool for tracking and on-line estimation problems:

- Estimate the system state sequentially using $\hat{x}_t = \mu_t$
- Obtain an uncertainty measure about the state using $\text{var}(\hat{x}_t) = P_t$
- Recursive least squares. With $C = 0$ we have the same model and updating rules as used in the RLS algorithm - hence RLS is a special case of Kalman.
- Fixed-lag smoothing: augment the state with past states: $x'_t = [x_t \, x_{t-1} \, \ldots \, x_{t-p}]$
- Fixed interval smoothing: the *Kalman smoother* operates backwards in time, estimating recursively $p(x_t|y_{0:T}), t < T$.

- Likelihood evaluation. A useful result is that the Kalman filter can sequentially evaluate the likelihood function, $p(y_{0:t})$. This is used for maximum likelihood or maximum *a posteriori* estimation, and also for Bayesian model choice problems and the Rao-Blackwellised particle filter. To see how this works, start from the Kalman prediction step:

$$p(x_{t+1}|y_{0:t}) = \mathcal{N}(x_{t+1}|A\mu_t, C + AP_tA^T)$$

Now, equation 14 expresses y_{t+1} in terms of Bx_{t+1} plus a random Gaussian disturbance w_t with covariance matrix D:

$$y_t = Bx_t + w_t$$

Hence we can obtain the conditional likelihood:

$$p(y_{t+1}|y_{0:t}) = \mathcal{N}(y_{t+1}|B\mu_{t+1|t}, D + BP_{t+1|t}B^T)$$

Finally, using the probability chain rule, we obtain the likelihood function:

$$p(y_{0:T}) = p(y_0)\prod_{t=0}^{T-1} p(y_{t+1}|y_{0:t})$$

NUMERICAL METHODS - OR THINGS YOU CAN'T DO WITH THE KALMAN FILTER

(Harvey, 1989; Anderson and Moore, 1979)

- The Kalman filter is optimal *only* for the linear Gaussian model. In other cases the Kalman filter will give the best *linear* estimator in a mean-square error sense, but this may not be good enough for highly non-linear or non-Gaussian models
- There are numerous methods for dealing with more general models, all based on numerical approximations to the filtering recursions of equations 10 and 11, e.g. the Gaussian sum filter (Sorenson and Alspach, 1971)
- Here we will consider two important examples in detail: the extended Kalman filter (EKF) and the Monte Carlo particle filter

THE EXTENDED KALMAN FILTER (EKF)

(Harvey, 1989; Anderson and Moore, 1979; Jazwinski, 1970)

The extended Kalman filter is the classical method for estimating non-linear state-space systems.

- Consider the following non-linear state-space model, which is the non-linear equivalent to equations 13 and 14:

$$x_{t+1} = A(x_t) + v_t \tag{22}$$

$$y_t = B(x_t) + w_t \tag{23}$$

where $A()$ and $B()$ are now non-linear functions.

- Perform a 1st order Taylor expansion of $A()$ and $B()$ around the points μ_t and $\mu_{t|t-1}$, respectively:

$$A(x_t) \approx A(\mu_t) + \left.\frac{\partial A(x_t)}{\partial x_t}\right|_{x_t=\mu_t}(x_t - \mu_t)$$

$$B(x_t) \approx B(\mu_{t|t-1}) + \left.\frac{\partial B(x_t)}{\partial x_t}\right|_{x_t=\mu_{t|t-1}}(x_t - \mu_{t|t-1})$$

- Substituting these approximations into the state-space model leads to a *linearized* set of equations which can be solved using the standard Kalman filter
- Limitatations - the approximation is still unimodal, hence for multimodal distributions the filter will fail
- Also the tracking performance and error covariance estimates will be sub-optimal

MONTE CARLO FILTERING

Consider numerical estimation of the following expectation:

$$\overline{h} = \mathbb{E}h(x_t) = \int h(x_t)p(x_t|y_{0:t})dx_t$$

- If the integral is intractable then we can resort to a Monte Carlo integration:

$$\widehat{\overline{h}} = 1/N \sum_{i=1}^{N} h(x_t^{(i)}), \quad \text{where } x_t^{(i)} \overset{\text{iid}}{\sim} p(x_t|y_{0:t}) \tag{24}$$

- More generally, when we cannot sample directly from $p(x_t|y_{0:t})$, we can sample from another distribution $q(x_t)$ ('importance function') having the same support as $p(x_t|y_{0:t})$. So we make N random draws from $q()$ instead of $p()$:

$$x_t^{(i)} \sim q(x_t), \qquad i = 1, \ldots, N$$

- Now we have to make a correction to ensure that the expectation estimate is good. It turns out that the required correction is proportional to the ratio $p()/q()$, which is termed the *importance weight*:

$$w_t^{(i)} \propto \frac{p(x_t^{(i)}|y_{0:t})}{q(x_t^i)}$$

- If we normalise the importance weights such that $\sum_{i=1}^N w_t^{(i)} = 1$ we can form an empirical approximation to the filtering density:

$$p(x_t|y_{0:t}) \approx \sum_{i=1}^N w_t^{(i)} \delta_{x_t^{(i)}}(x_t) \tag{25}$$

from which expectation estimates can be obtained as:

$$\widehat{\overline{h}} = \sum_{i=1}^{N} w^{(i)} h(x_t^{(i)}), \tag{26}$$

$$\text{where } x_t^{(i)} \overset{\text{iid}}{\sim} q(x_t),\ w_t^{(i)} \propto p(x_t^{(i)}|y_{0:t})/q(x_t^{(i)}),\ \sum_{i=1}^{N} w_t^{(i)} = 1 \tag{27}$$

i.e.

$$\begin{aligned}\overline{h} &= \mathbb{E}h(x_t) = \int h(x_t)p(x_t|y_{0:t})dx_t \\ &\approx \int h(x_t)\sum_{i=1}^{N} w_t^{(i)}\delta_{x_t^{(i)}}(x_t)dx_t = \sum_{i=1}^{N} w^{(i)} h(x_t^{(i)})\end{aligned}$$

- **Resampling** (this will prove important in the sequential setting). We now have the option of resampling the the particles so they have uniform weights:

$$\text{Set} \quad {x'}_t^{(i)} = x_t^{(i)} \ \text{ with probability } \ w_t^{(i)}$$

and set ${w'}_{t+1}^{(i)} = 1/N$.
While this is unnecessary in the static case, and would always increase the Monte Carlo variation of our estimators, it is a vital component of the sequential schemes which follow, limiting degeneracy of the importance weights over time. Note that resampling schemes can incorporate variance reduction strategies such as stratification in order to improve performance.

- We now have a means for approximating $p(x_t|y_{0:t})$ and also expectations of x_t.
- But, how do we adapt this to the sequential context? (Note that $p(x_t|y_{0:t})$ *cannot* in general be evaluated).

- Consider updating the filtering distribution from t to $t+1$:

Step 0:

$$p(x_t, x_{t+1}|y_{0:t}) = p(x_t|y_{0:t})f(x_{t+1}|x_t)$$

Step 1:

$$p(x_{t+1}|y_{0:t}) = \int p(x_t, x_{t+1}|y_{0:t})dx_t$$

Step 2:

$$p(x_{t+1}|y_{0:t+1}) = \frac{g(y_{t+1}|x_{t+1})p(x_{t+1}|y_{0:t})}{p(y_{t+1}|y_{0:t})}$$

- We would like to mimic the three steps here by Monte Carlo operations.
- Suppose we start off with many 'particles' drawn from the filtering distribution $p(x_t|y_{0:t})$. We label these particles as

$$x_t^{(i)}, \quad i = 1, 2, \ldots, N \quad \text{with} \quad N >> 1$$

- These can be used to plot histogram estimates of $p(x_t|y_{0:t})$, form Monte Carlo estimates of expectations, ..., in fact perform almost any inference procedure we care to choose, provided N is 'sufficiently' large
- We can simulate **Step 0** above by taking each particle $x_t^{(i)}$ in turn and generating a new state from the state transition density according to:

$$x_{t+1}^{(i)} \sim f(x_{t+1}|x_t^{(i)})$$

- Each pair $(x_t^{(i)}, x_{t+1}^{(i)})$ is now a joint random sample from $p(x_t, x_{t+1}|y_{0:t})$.
- By construction, $x_{t+1}^{(i)}$ taken on its own is a random sample from the required *marginal distribution* $p(x_{t+1}|y_{0:t})$, **(Step 1)**

- **Step 2**. We now have samples from $p(x_{t+1}|y_{0:t})$. Step 2 gives us the appropriate importance weight:

$$
\begin{aligned}
w_{t+1} &\propto \frac{p(x_{t+1}|y_{0:t+1})}{q(x_{t+1})} \\
&\propto \frac{\frac{g(y_{t+1}|x_{t+1})p(x_{t+1}|y_{0:t})}{p(y_{t+1}|y_{0:t})}}{p(x_{t+1}|y_{0:t})} \\
&\propto g(y_{t+1}|x_{t+1})
\end{aligned}
$$

- We now have the option of
  1. retaining weighted particles, in which case the weights are accumulated over time as

$$w_{t+1} \propto w_t \, g(y_{t+1}|x_{t+1})$$

     Or:
  2. Resampling the particles so they have uniform weights:

$$\text{Set} \quad {x'}^{(i)}_{t+1} = x^{(i)}_{t+1} \ \text{ with probability } \ w^{(i)}_t$$

     and set ${w'}^{(i)}_{t+1} = 1/N$.

- A basic algorithm with (optional) resampling at every time step, the 'Bootstrap Filter', is thus (Gordon, Salmond and Smith, 1993)(Kitagawa, 1996):

For $t = 1, 2 \ldots, T$

For $i = 1, 2, \ldots, N$

$x_{t+1}^{(i)} \sim f(x_{t+1}^{(i)} | x_t^{(i)})$

$w_{t+1}^{(i)} \propto w_t^{(i)} \, g(y_{t+1} | x_{t+1}^{(i)})$

End

For $i = 1, 2, \ldots, N$

(Optional) Resample $x_{t+1}^{(i)}$ with probability $w_{t+1}^{(i)}$. Set $w_{t+1}^{(i)} = 1/N$

End

End

Figure 2: Bootstrap filter operation - nonlinear model

GENERAL SEQUENTIAL IMPORTANCE SAMPLING

We can do better in many cases than the basic bootstrap filter, by choosing a better importance function. Consider now the following modified updates:

Step 0':

$$q(x_t, x_{t+1}|y_{0:t+1}) = p(x_t|y_{0:t})q(x_{t+1}|x_t, y_{0:t+1})$$

Step 2':

$$p(x_t, x_{t+1}|y_{0:t+1}) = \frac{g(y_{t+1}|x_{t+1})f(x_{t+1}|x_t)p(x_t|y_{0:t})}{p(y_{t+1}|y_{0:t})}$$

We now consider $q(x_t, x_{t+1}|y_{0:t+1})$ to be an importance function for $p(x_t, x_{t+1}|y_{0:t+1})$.

The importance weight for **Step 2'** is hence modified to:

$$w_{t+1}^{(i)} \propto w_t^{(i)} \frac{g(y_{t+1}|x_{t+1}^{(i)})f(x_{t+1}^{(i)}|x_t^{(i)})}{q(x_{t+1}^{(i)}|x_t^{(i)})} \tag{28}$$

This is the general sequential importance (SIS) sampling method ((Liu and Chen, 1998), (Doucet, Godsill and Andrieu, 2000)).

Important special cases:

- $q(x_{t+1}|x_t) = f(x_{t+1}|x_t)$ - bootstrap filter (Gordon et al., 1993)(Kitagawa, 1996) - ‘prior’ sampling
- $q(x_{t+1}|x_t) = p(x_{t+1}|x_t, y_{t+1})$ - sequential imputations (Liu and Chen, 1995) - optimal importance function (Doucet et al., 2000).

Repeated application over time (without resampling) leads to degeneracy of the weights - all the mass becomes concentrated on a few i - hence estimates are poor.

The resampling procedure (choosing $x_{t+1}^{(i)}$ with probability $w_{t+1}^{(i)}$) alleviates this - SIR (**?**).

eg. Measure degeneracy by estimating the variance of $w_{t+1}^{(i)}$ - since reduction in effective sample size is approximately $(1+\text{var}(w_{t+1}^{(i)}))$.

The algorithm is now modified to:

For $t = 0, 2..., T$

For $i = 1, 2, ..., N$

$x_{t+1}^{(i)} \sim q(x_{t+1}^{(i)} | x_t^{(i)}, y_{0:t+1})$

$w_{t+1}^{(i)} \propto w_t^{(i)} \frac{g(y_{t+1}|x_{t+1}^{(i)}) f(x_{t+1}^{(i)}|x_t^{(i)})}{q(x_{t+1}^{(i)}|x_t^{(i)}), y_{0:t+1}}$

End

For $i = 1, 2, ..., N$

(Optional) Resample $x_{t+1}^{(i)}$ with probability $w_{t+1}^{(i)}$. Set $w_{t+1}^{(i)} = 1/N$

End

End

RECENT DEVELOPMENTS

Some (biased!) pointers to recent advances in algorithms:

- **Auxiliary particle filters** - (Pitt and Shephard, 1999). The idea here is to presample the time t particles according to their importance at time $t+1$ - see (Godsill and Clapp, 2001) for further discussion.
- **Rao-Blackwellised particle filters** - (Doucet et al., 2000). The idea here is to marginalise any tractable parameters (e.g. conditionally linear- Gaussian or Hidden Markov Model (HMM)). The system is no longer Markovian and weights are modified accordingly. Similar idea to the 'Mixture Kalman filter' (Liu and Chen, 2000).
- **MCMC particle filters** (Gilks and Berzuini, 2000)(MacEachern, Clyde and Liu, 1999). See also annealed particle filters (Godsill and Clapp, 2001).
- **Particle smoothers** - these sample from the entire state trajectory $p(x_{0:t}|y_{0:t})$. The can be implemented in a particle filtering/ backwards particle smoothing framework, see (Fong, Godsill, Doucet and West, 2002)(Hürzeler and Künsch, 2000)(Kitagawa, 1996)

MONTE CARLO FILTERING - HISTORY

- Automatic control problems - (Handschin and Mayne, 1969; Handschin, 1970; Akashi and Kumamoto, 1977; Akashi and Kumamoto, 1975; Zaritskii, Svetnik and Shimelevich, 1975)
- Statistical developments - (West, 1993)(Mueller, 1992)(**?**)(Liu and Chen, 1995)(Liu and Chen, 1998)
- Recent review material: (Doucet et al., 2000; Doucet, De Freitas and Gordon, 2001)

MONTE CARLO FILTERING - THEORY

The theory of particle filtering is now quite well developed, although it remains a research topic still for mathematicians. For example, the empirical particle measure converges almost surely to $p(x_t|y_{0:t})$ for all $t > 0$ as $N \longrightarrow \infty$ under quite general conditions on the state space model. Moreover, rates of convergence to zero have been established for mean squared error with respect to this filtering density. Hence particle filters are rigorously validated as a means for tracking the distribution of, and estimating the value of a hidden state over time. Some recent advances in convergence analysis can be found in Del Moral (1996), Del Moral (1998), Crisan, Del Moral and Lyons (1999), Crisan and Lyons (1999), Crisan and Doucet (2000), Crisan (2001).

References

Akashi, H. and Kumamoto, H.: 1975, Construction of discrete-time nonlinear filter by Monte Carlo methods with variance-reducing techniques, *Systems and Control* **19**, 211–221. In Japanese.

Akashi, H. and Kumamoto, H.: 1977, Random sampling approach to state estimation in switching environments, *Automatica* **13**, 429–434.

Anderson, B. D. and Moore, J. B.: 1979, *Optimal Filtering*, Prentice-Hall, New Jersey.

Crisan, D.: 2001, Parficle filters - a theoretical perspective, *in* A. Doucet, J. F. G. De Freitas and N. J. Gordon (eds), *Sequential Monte Carlo Methods in Practice*, New York: Springer-Verlag.

Crisan, D., Del Moral, P. and Lyons, T.: 1999, Discrete filtering using branching and interacting particle systems, *Markov proc. & rel. flds.* **5**(3), 293–318.

Crisan, D. and Doucet, A.: 2000, Convergence of sequential Monte Carlo methods, *Technical Report CUED/F-INFENG/TR.381*, Cambridge University Engineering Department.

Crisan, D. and Lyons, T.: 1999, A particle approximation of the solution of the Kushner-Stratonovich equation, *Prob. theor. & rel. flds.* **5**(3), 293–318.

Del Moral, P.: 1996, Non-linear filtering: interacting particle solution, *Markov proc. & rel. flds.* **2**(4), 555–580.

Del Moral, P.: 1998, Measure valued processes and interacting particle systems, *Ann. Appl. Prob.* **8**(2), 438–495.

Doucet, A., De Freitas, J. F. G. and Gordon, N. J. (eds): 2001, *Sequential Monte Carlo Methods in Practice*, New York: Springer-Verlag.

Doucet, A., Godsill, S. J. and Andrieu, C.: 2000, On sequential Monte Carlo sampling methods for Bayesian filtering, *Statistics and Computing* **10**, 197–208.

Fong, W., Godsill, S. J., Doucet, A. and West, M.: 2002, Monte carlo smoothing with application to speech enhancement, *IEEE Trans. on Signal Processing* . Special issue on Monte Carlo Methods.

Gilks, W. R. and Berzuini, C.: 2000, Following a moving target - Monte Carlo inference for dynamic Bayesian models, *J. R. Statist. Soc. B* **63**, 1–20.

Godsill, S. J. and Clapp, T. C.: 2001, Improvement strategies for Monte Carlo particle filters, *in* A. Doucet, J. F. G. De Freitas and N. J. Gordon (eds), *Sequential Monte Carlo Methods in Practice*, New York: Springer-Verlag.

Gordon, N. J., Salmond, D. J. and Smith, A. F. M.: 1993, Novel approach to nonlinear/non-Gaussian Bayesian state estimation, *IEE Proceedings-F* **140**(2), 107–113.

Handschin, J. E.: 1970, Monte Carlo techniques for prediction and filtering of non-linear stochastic processes, *Automatica* **6**, 555–563.

Handschin, J. E. and Mayne, D. Q.: 1969, Monte Carlo techniques to estimate the conditional expectation in multi-stage non-linear filtering, *Int. J. Control* **9**, 547–559.

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

Hürzeler, M. and Künsch, H. R.: 2000, Monte Carlo approximations for general state space models, *J. Comp. Graph. Statist.* **7**(2), 175–193.

Jazwinski, A.: 1970, *Stochastic processes and filtering theory*, New York: Academic Press.

Kitagawa, G.: 1996, Monte Carlo filter and smoother for nonlinear non-Gaussian state space models, *J. Comp. Graph. Statist.* **5**, 1–25.

Liu, J. S. and Chen, R.: 1995, Blind deconvolution via sequential imputations, *J. Am. Statist. Assoc.* **90**(430), 567–576.

Liu, J. S. and Chen, R.: 1998, Sequential Monte Carlo methods for dynamical systems, *J. Am. Statist. Assoc.* **93**, 1032–44.

Liu, J. S. and Chen, R.: 2000, Mixture Kalman filters, *J. R. Statist. Soc. B* **62**, 493–508.

MacEachern, S. N., Clyde, M. and Liu, J. S.: 1999, Sequential importance sampling for nonparametric Bayes models: the next generation, *Can. J. Stats.* **27**, 251–267.

Mueller, P.: 1992, Posterior integration in dynamic models, *Computing Science and Statistics* **24**, 318–324.

Pitt, M. K. and Shephard, N.: 1999, Filtering via simulation: auxiliary particle filters, *J. Am. Statist. Assoc.* **94**, 590–9.

Sorenson, H. W. and Alspach, D. L.: 1971, Recursive Bayesian estimation using Gaussian sums, *Automatica* **7**, 465–479.

West, M.: 1993, Mixture models, Monte Carlo, Bayesian updating and dynamic models, *Computing Science and Statistics* **24**, 325–333.

Zaritskii, V. S., Svetnik, V. B. and Shimelevich, L. I.: 1975, Monte Carlo techniques in problems of optimal data processing, *Automation and Remote Control* **12**, 95–103.