Sequential Monte Carlo Methods for Bayesian Computation

A. Doucet

Feb. 2015

Monte Carlo Methods

- MCMC are the tools of choice in Bayesian computation for over 20 years whereas SMC have been widely used for 15 years in vision and robotics.
- Both MCMC and SMC are asymptotically (as you increase computational efforts) bias-free but computationally expensive.
- The development of new methodology combined to the emergence of cheap multicore architectures makes now SMC a powerful alternative/complementary approach to MCMC to address general Bayesian computational problems.

Some References and Resources

- A.D., J.F.G. De Freitas & N.J. Gordon (editors), *Sequential Monte Carlo Methods in Practice*, Springer-Verlag: New York, 2001.
- P. Del Moral, *Feynman-Kac Formulae: Genealogical and Interacting Particle Systems with Applications*, Springer-Verlag: New York, 2004.
- **Webpage with links to papers and codes**: http://www.stats.ox.ac.uk/~doucet/smc_resources.html
- Thousands of papers on the subject appear every year.

Organization of Lectures

- **State-Space Models** (approx.3 hours)
  - SMC filtering and smoothing
  - Maximum likelihood parameter inference
  - Bayesian parameter inference
- **Beyond State-Space Models** (approx. 1 hour)
  - SMC methods for generic sequence of target distributions
  - SMC samplers.

State-Space Models

- Let $\{X_t\}_{t\geq 1}$ be a latent/hidden $\mathcal{X}$-valued Markov process with

$$X_1 \sim \mu(\cdot) \text{ and } X_t|(X_{t-1}=x) \sim f(\cdot|x).$$

- Let $\{Y_t\}_{t\geq 1}$ be an $\mathcal{Y}$-valued Markov observation process such that observations are conditionally independent given $\{X_t\}_{t\geq 1}$ and

$$Y_t|(X_t=x) \sim g(\cdot|x).$$

- General class of time series models aka Hidden Markov Models (HMM) including

$$X_t = \Psi(X_{t-1}, V_t), \quad Y_t = \Phi(X_t, W_t)$$

where V_t, W_t are two sequences of i.i.d. random variables.

- **Aim**: Infer $\{X_t\}$ given observations $\{Y_t\}$ on-line or off-line.

State-Space Models

- State-space models are ubiquitous in control, data mining, econometrics, geosciences, system biology etc. Since Jan. 2014, more than 16,900 papers have already appeared (source: Google Scholar).
- **Finite State-space HMM**: $\mathcal{X}$ is a finite space, i.e. $\{X_t\}$ is a finite Markov chain

$$Y_t|(X_t=x) \sim g(\cdot|x)$$

- **Linear Gaussian state-space model**

$$X_t = AX_{t-1} + BV_t, \quad V_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I)$$
$$Y_t = CX_t + DW_t, \quad W_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I)$$

- **Switching Linear Gaussian state-space model:** $X_t = (X_t^1, X_t^2)$ where $\{X_t^1\}$ is a finite Markov chain,

$$X_t^2 = A(X_t^1)X_{t-1}^2 + B(X_t^1)V_t, \quad V_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I)$$
$$Y_t = C(X_t^1)X_t^2 + D(X_t^1)W_t, \quad W_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I)$$

State-Space Models

- **Stochastic Volatility model**

$$X_t = \phi X_{t-1} + \sigma V_t, \quad V_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1)$$
$$Y_t = \beta \exp(X_t/2) W_t, \quad W_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1)$$

- **Biochemical Network model**

$$\begin{aligned}
&\Pr\left(X_{t+dt}^1 = x_t^1+1, X_{t+dt}^2 = x_t^2 \middle| x_t^1, x_t^2\right) = \alpha x_t^1 dt + o(dt),\\
&\Pr\left(X_{t+dt}^1 = x_t^1-1, X_{t+dt}^2 = x_t^2+1 \middle| x_t^1, x_t^2\right) = \beta x_t^1 x_t^2 dt + o(dt),\\
&\Pr\left(X_{t+dt}^1 = x_t^1, X_{t+dt}^2 = x_t^2-1 \middle| x_t^1, x_t^2\right) = \gamma x_t^2 dt + o(dt),
\end{aligned}$$

with

$$Y_k = X_{k\Delta T}^1 + W_k \text{ with } W_k \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2).$$

- **Nonlinear Diffusion model**

$$dX_t = \alpha(X_t)dt + \beta(X_t)dV_t, \quad V_t \text{ Brownian motion}$$
$$Y_k = \gamma(X_{k\Delta T}) + W_k, \quad W_k \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2).$$

Inference in State-Space Models

- Given observations $y_{1:t} := (y_1, y_2, \ldots, y_t)$, inference about $X_{1:t} := (X_1, \ldots, X_t)$ relies on the posterior

$$p(x_{1:t}|y_{1:t}) = \frac{p(x_{1:t}, y_{1:t})}{p(y_{1:t})}$$

where

$$p(x_{1:t}, y_{1:t}) = \underbrace{\mu(x_1)\prod_{k=2}^{t} f(x_k|x_{k-1})}_{p(x_{1:t})}\underbrace{\prod_{k=1}^{t} g(y_k|x_k)}_{p(y_{1:t}|x_{1:t})},$$
$$p(y_{1:t}) = \int \cdots \int p(x_{1:t}, y_{1:t})\, dx_{1:t}$$

- When $\mathcal{X}$ is finite & linear Gaussian models, $\{p(x_t|y_{1:t})\}_{t\geq 1}$ can be computed exactly. For non-linear models, approximations are required: EKF, UKF, Gaussian sum filters, etc.
- Approximations of $\{p(x_t|y_{1:t})\}_{t=1}^{T}$ provide approximation of $p(x_{1:T}|y_{1:T})$.

Monte Carlo Methods Basics

- Assume you can generate $X_{1:t}^{(i)} \sim p\left(x_{1:t} \middle| y_{1:t}\right)$ where $i = 1, ..., N$ then MC approximation is

$$\widehat{p}\left(x_{1:t} \middle| y_{1:t}\right) = \frac{1}{N}\sum_{i=1}^{N} \delta_{X_{1:t}^{(i)}}\left(x_{1:t}\right)$$

- *Integration is straightforward.*

$$\begin{aligned}\int \varphi_t\left(x_{1:t}\right) p\left(x_{1:t} \middle| y_{1:t}\right) dx_{1:t} &\approx \int \varphi_t\left(x_{1:t}\right) \widehat{p}\left(x_{1:t} \middle| y_{1:t}\right) dx_{1:t} \\ &= \frac{1}{N}\sum_{i=1}^{N} \varphi\left(X_{1:t}^{(i)}\right)\end{aligned}$$

- *Marginalization is straightforward.*

$$\widehat{p}\left(x_k \middle| y_{1:t}\right) = \int \widehat{p}\left(x_{1:t} \middle| y_{1:t}\right) dx_{1:k-1} dx_{k+1:t} = \frac{1}{N}\sum_{i=1}^{N} \delta_{X_k^{(i)}}\left(x_k\right).$$

- **Basic and key property**: $\mathbb{V}\left[\frac{1}{N}\sum_{i=1}^{N} \varphi\left(X_{1:t}^{(i)}\right)\right] = \frac{C(t \dim(\mathcal{X}))}{N}$, i.e. rate of convergence to zero is independent of $\dim\left(\mathcal{X}\right)$ and t.

Monte Carlo Methods

- **Problem 1**: We cannot typically generate exact samples from $p\left(x_{1:t} \middle| y_{1:t}\right)$ for non-linear non-Gaussian models.
- **Problem 2**: Even if we could, algorithms to generate samples from $p\left(x_{1:t} \middle| y_{1:t}\right)$ will have at least complexity $\mathcal{O}\left(t\right)$.
- Typical solution to problem 1 is to generate approximate samples using MCMC methods but these methods are not recursive.
- **SMC Methods** solves *partially* Problem 1 and Problem 2 by breaking the problem of sampling from $p\left(x_{1:t} \middle| y_{1:t}\right)$ into a collection of simpler subproblems. First approximate $p\left(x_1 \middle| y_1\right)$ and $p\left(y_1\right)$ at time 1, then $p\left(x_{1:2} \middle| y_{1:2}\right)$ and $p\left(y_{1:2}\right)$ at time 2 and so on.
- Each target distribution is approximated by a cloud of random samples termed *particles* evolving according to *importance sampling* and *resampling* steps.

Standard Bayesian Recursion

- In most textbooks, you will find the following recursion for $\left\{p\left(x_t \middle| y_{1:t}\right)\right\}_{t\geq 1}$.
- **Prediction step**

$$\begin{aligned}p\left(x_t \middle| y_{1:t-1}\right) &= \int p\left(x_{t-1}, x_t \middle| y_{1:t-1}\right) dx_{t-1} \\ &= \int p\left(x_t \middle| y_{1:t-1}, x_{t-1}\right) p\left(x_{t-1} \middle| y_{1:t-1}\right) dx_{t-1} \\ &= \int f\left(x_t \middle| x_{t-1}\right) p\left(x_{t-1} \middle| y_{1:t-1}\right) dx_{t-1}.\end{aligned}$$

- **Bayes Updating step**

$$p\left(x_t \middle| y_{1:t}\right) = \frac{g\left(y_t \middle| x_t\right)\ p\left(x_t \middle| y_{1:t-1}\right)}{p\left(y_t \middle| y_{1:t-1}\right)}$$

where

$$p\left(y_t \middle| y_{1:t-1}\right) = \int g\left(y_t \middle| x_t\right)\ p\left(x_t \middle| y_{1:t-1}\right) dx_t$$

Bayesian Recursion on Path Space

- SMC approximate directly $\left\{p\left(x_{1:t} \middle| y_{1:t}\right)\right\}_{t\geq 1}$ not $\left\{p\left(x_t \middle| y_{1:t}\right)\right\}_{t\geq 1}$ and relies on

$$\begin{aligned}p\left(x_{1:t} \middle| y_{1:t}\right) &= \frac{p\left(x_{1:t}, y_{1:t}\right)}{p\left(y_{1:t}\right)} = \frac{g\left(y_t \middle| x_t\right)\ f\left(x_t \middle| x_{t-1}\right)}{p\left(y_t \middle| y_{1:t-1}\right)} \frac{p\left(x_{1:t-1}, y_{1:t-1}\right)}{p\left(y_{1:t-1}\right)} \\ &= \frac{g\left(y_t \middle| x_t\right)\ \overbrace{f\left(x_t \middle| x_{t-1}\right) p\left(x_{1:t-1} \middle| y_{1:t-1}\right)}^{\text{predictive } p\left(x_{1:t} \middle| y_{1:t-1}\right)}}{p\left(y_t \middle| y_{1:t-1}\right)}\end{aligned}$$

where

$$p\left(y_t \middle| y_{1:t-1}\right) = \int g\left(y_t \middle| x_t\right)\ p\left(x_{1:t} \middle| y_{1:t-1}\right) dx_{1:t}$$

- This can be alternatively written as

$$\begin{aligned}&\textbf{Prediction} \quad p\left(x_{1:t} \middle| y_{1:t-1}\right) = f\left(x_t \middle| x_{t-1}\right) p\left(x_{1:t-1} \middle| y_{1:t-1}\right), \\ &\textbf{Update} \quad p\left(x_{1:t} \middle| y_{1:t}\right) = \frac{g\left(y_t \middle| x_t\right) p\left(x_{1:t} \middle| y_{1:t-1}\right)}{p\left(y_t \middle| y_{1:t-1}\right)}.\end{aligned}$$

- SMC is a simple and natural simulation-based implementation of this recursion.

Monte Carlo Implementation of Prediction Step

- Assume you have at time $t-1$

$$\widehat{p}\left(x_{1:t-1}|\, y_{1:t-1}\right) = \frac{1}{N}\sum_{i=1}^{N}\delta_{X_{1:t-1}^{(i)}}\left(x_{1:t-1}\right).$$

- By sampling $\widetilde{X}_t^{(i)} \sim f\left(x_t|\, X_{t-1}^{(i)}\right)$ and setting $\widetilde{X}_{1:t}^{(i)} = \left(X_{1:t-1}^{(i)}, \widetilde{X}_t^{(i)}\right)$ then

$$\widehat{p}\left(x_{1:t}|\, y_{1:t-1}\right) = \frac{1}{N}\sum_{i=1}^{N}\delta_{\widetilde{X}_{1:t}^{(i)}}\left(x_{1:t}\right).$$

- Sampling from $f\left(x_t|\, x_{t-1}\right)$ is usually straightforward and can be done even if $f\left(x_t|\, x_{t-1}\right)$ does not admit any analytical expression; e.g. biochemical network models.

Importance Sampling Implementation of Updating Step

- Our target at time t is

$$p\left(x_{1:t}|\, y_{1:t}\right) = \frac{g\left(y_t|\, x_t\right) p\left(x_{1:t}|\, y_{1:t-1}\right)}{p\left(y_t|\, y_{1:t-1}\right)}$$

so by substituting $\widehat{p}\left(x_{1:t}|\, y_{1:t-1}\right)$ to $p\left(x_{1:t}|\, y_{1:t-1}\right)$ we obtain

$$\begin{aligned}\widehat{p}\left(y_t|\, y_{1:t-1}\right) &= \int g\left(y_t|\, x_t\right)\widehat{p}\left(x_{1:t}|\, y_{1:t-1}\right) dx_{1:t} \\ &= \frac{1}{N}\sum_{i=1}^{N} g\left(y_t|\, \widetilde{X}_t^{(i)}\right).\end{aligned}$$

- We now have

$$\widetilde{p}\left(x_{1:t}|\, y_{1:t}\right) = \frac{g\left(y_t|\, x_t\right)\widehat{p}\left(x_{1:t}|\, y_{1:t-1}\right)}{\widehat{p}\left(y_t|\, y_{1:t-1}\right)} = \sum_{i=1}^{N} W_t^{(i)}\delta_{\widetilde{X}_{1:t}^{(i)}}\left(x_{1:t}\right).$$

with $W_t^{(i)} \propto g\left(y_t|\, \widetilde{X}_t^{(i)}\right)$, $\sum_{i=1}^{N} W_t^{(i)} = 1$.

Multinomial Resampling

- We have a "weighted" approximation $\widetilde{p}\left(x_{1:t}|\, y_{1:t}\right)$ of $p\left(x_{1:t}|\, y_{1:t}\right)$

$$\widetilde{p}\left(x_{1:t}|\, y_{1:t}\right) = \sum_{i=1}^{N} W_t^{(i)}\delta_{\widetilde{X}_{1:t}^{(i)}}\left(x_{1:t}\right).$$

- To obtain N samples $X_{1:t}^{(i)}$ approximately distributed according to $p\left(x_{1:t}|\, y_{1:t}\right)$, resample N times with replacement

$$X_{1:t}^{(i)} \sim \widetilde{p}\left(x_{1:t}|\, y_{1:t}\right)$$

to obtain

$$\widehat{p}\left(x_{1:t}|\, y_{1:t}\right) = \frac{1}{N}\sum_{i=1}^{N}\delta_{X_{1:t}^{(i)}}\left(x_{1:t}\right) = \sum_{i=1}^{N}\frac{N_t^{(i)}}{N}\delta_{\widetilde{X}_{1:t}^{(i)}}\left(x_{1:t}\right)$$

where $\left\{N_t^{(i)}\right\}$ follow a multinomial with $\mathbb{E}\left[N_t^{(i)}\right] = NW_t^{(i)}$, $\mathbb{V}\left[N_t^{(1)}\right] = NW_t^{(i)}\left(1-W_t^{(i)}\right)$.

- This can be achieved in $\mathcal{O}(N)$.

Vanilla SMC: Bootstrap Filter (Gordon et al., 1993)

At time $t = 1$

- Sample $\widetilde{X}_1^{(i)} \sim \mu\left(x_1\right)$ then

$$\widetilde{p}\left(x_1|\, y_1\right) = \sum_{i=1}^{N} W_1^{(i)}\delta_{\widetilde{X}_1^{(i)}}\left(x_1\right),\ W_1^{(i)} \propto g\left(y_1|\, \widetilde{X}_1^{(i)}\right).$$

- Resample $X_1^{(i)} \sim \widetilde{p}\left(x_1|\, y_1\right)$ to obtain $\widehat{p}\left(x_1|\, y_1\right) = \frac{1}{N}\sum_{i=1}^{N}\delta_{X_1^{(i)}}\left(x_1\right)$.

At time $t \geq 2$

- Sample $\widetilde{X}_t^{(i)} \sim f\left(x_t|\, X_{t-1}^{(i)}\right)$, set $\widetilde{X}_{1:t}^{(i)} = \left(X_{1:t-1}^{(i)}, \widetilde{X}_t^{(i)}\right)$ and

$$\widetilde{p}\left(x_{1:t}|\, y_{1:t}\right) = \sum_{i=1}^{N} W_t^{(i)}\delta_{\widetilde{X}_{1:t}^{(i)}}\left(x_{1:t}\right),\ W_t^{(i)} \propto g\left(y_t|\, \widetilde{X}_t^{(i)}\right).$$

- Resample $X_{1:t}^{(i)} \sim \widetilde{p}\left(x_{1:t}|\, y_{1:t}\right)$ to obtain $\widehat{p}\left(x_{1:t}|\, y_{1:t}\right) = \frac{1}{N}\sum_{i=1}^{N}\delta_{X_{1:t}^{(i)}}\left(x_{1:t}\right)$.

SMC Output

- At time t, we get

$$\widetilde{p}\left(x_{1:t}|\, y_{1:t}\right) = \sum_{i=1}^{N} W_t^{(i)} \delta_{\widetilde{X}_{1:t}^{(i)}}\left(x_{1:t}\right),$$

$$\widehat{p}\left(x_{1:t}|\, y_{1:t}\right) = \frac{1}{N}\sum_{i=1}^{N} \delta_{X_{1:t}^{(i)}}\left(x_{1:t}\right).$$

- The marginal likelihood estimate is given by

$$\widehat{p}\left(y_{1:t}\right) = \prod_{k=1}^{t} \widehat{p}\left(y_k|\, y_{1:k-1}\right) = \prod_{k=1}^{t}\left(\frac{1}{N}\sum_{i=1}^{N} g\left(y_k|\, \widetilde{X}_k^{(i)}\right)\right).$$

- Computational complexity is $\mathcal{O}\left(N\right)$ at each time step and memory requirements $\mathcal{O}\left(tN\right)$.
- If we are only interested in $p\left(x_t|\, y_{1:t}\right)$ or $p\left(s_t\left(x_{1:t}\right)|\, y_{1:t}\right)$ where $s_t\left(x_{1:t}\right) = \Psi_t\left(x_t, s_{t-1}\left(x_{1:t-1}\right)\right)$ - e.g. $s_t\left(x_{1:t}\right) = \sum_{k=1}^{t} x_k^2$ - is fixed-dimensional then memory requirements $\mathcal{O}\left(N\right)$.

SMC on Path-Space - figures by Olivier Cappé

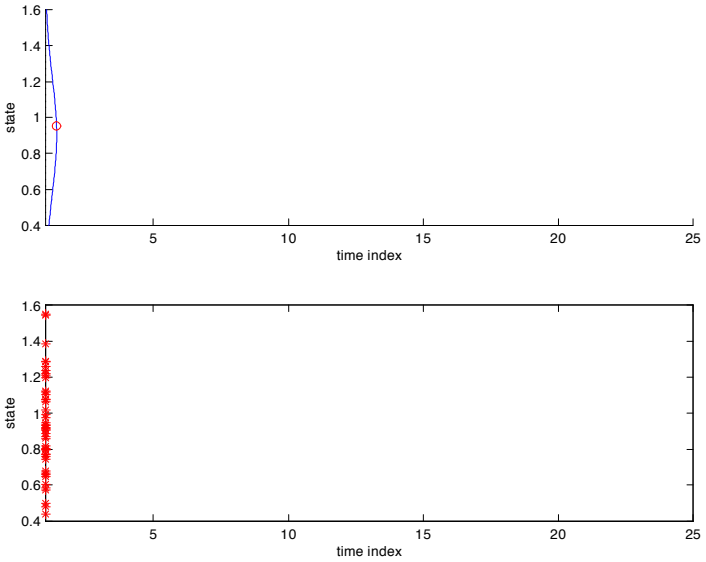


Figure: $p\left(x_1|\, y_1\right)$ and $\widehat{\mathbb{E}}\left[X_1|\, y_1\right]$ (top) and particle approximation of $p\left(x_1|\, y_1\right)$

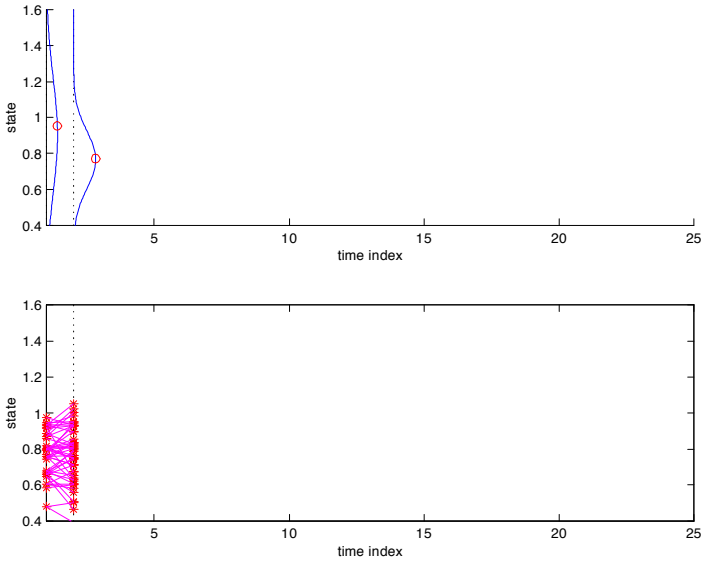


Figure: $p\left(x_1|\, y_1\right)$, $p\left(x_2|\, y_{1:2}\right)$and $\widehat{\mathbb{E}}\left[X_1|\, y_1\right]$, $\widehat{\mathbb{E}}\left[X_2|\, y_{1:2}\right]$ (top) and particle approximation of $p\left(x_{1:2}|\, y_{1:2}\right)$ (bottom)

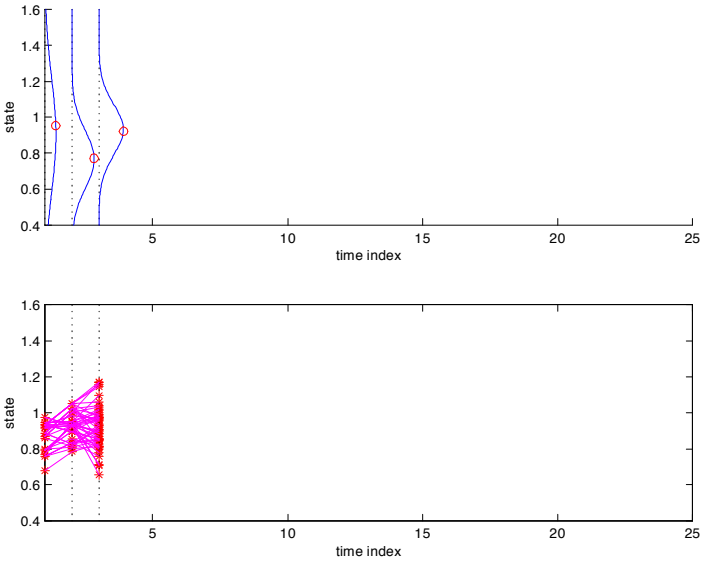


Figure: $p\left(x_t|\, y_{1:t}\right)$ and $\widehat{\mathbb{E}}\left[X_t|\, y_{1:t}\right]$ for $t = 1, 2, 3$ (top) and particle approximation of $p\left(x_{1:3}|\, y_{1:3}\right)$ (bottom)

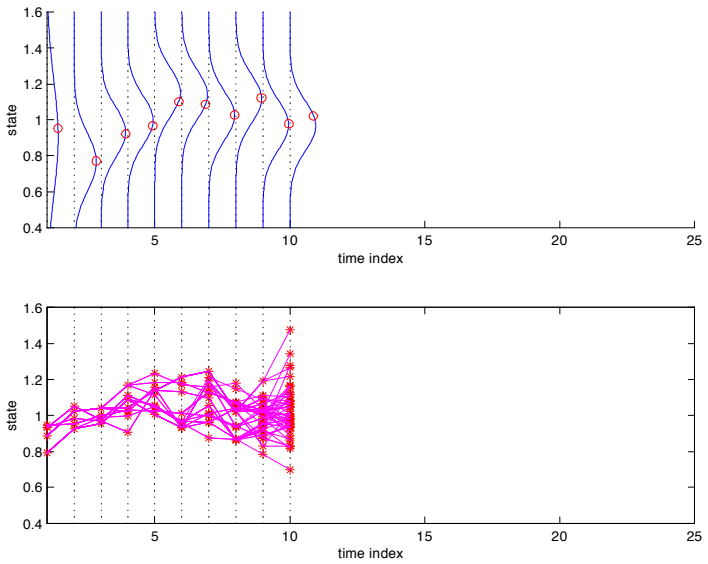


Figure: $p(x_t|y_{1:t})$ and $\widehat{\mathbb{E}}[X_t|y_{1:t}]$ for $t = 1, ..., 10$ (top) and particle approximation of $p(x_{1:10}|y_{1:10})$ (bottom)

Remarks

- Empirically this SMC strategy performs well in terms of estimating the marginals $\{p(x_t|y_{1:t})\}_{t\geq 1}$. This is what is only necessary in many applications thankfully.
- However, the joint distribution $p(x_{1:t}|y_{1:t})$ is poorly estimated when t is large; i.e. we have in the previous example

$$\widehat{p}(x_{1:11}|y_{1:24}) = \delta_{X^*_{1:11}}(x_{1:11}).$$

- **Degeneracy problem**. For any N and any k, there exists $t(k, N)$ such that for any $t \geq t(k, N)$

$$\widehat{p}(x_{1:k}|y_{1:t}) = \delta_{X^*_{1:k}}(x_{1:k});$$

$\widehat{p}(x_{1:t}|y_{1:t})$ is an unreliable approximation of $p(x_{1:t}|y_{1:t})$ as $t \nearrow$.

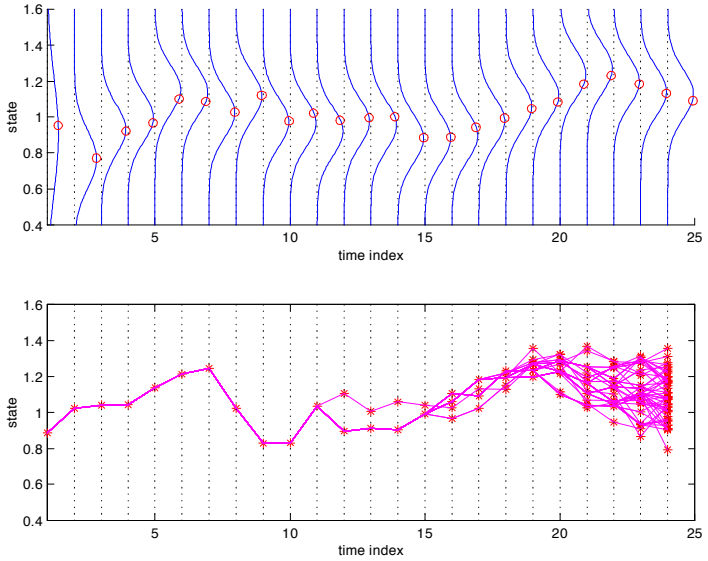


Figure: $p(x_t|y_{1:t})$ and $\widehat{\mathbb{E}}[X_t|y_{1:t}]$ for $t = 1, ..., 24$ (top) and particle approximation of $p(x_{1:24}|y_{1:24})$ (bottom)

Another Illustration of the Degeneracy Phenomenon

- For the linear Gaussian state-space model described before, we can compute exactly S_t/t where

$$S_t = \int \left(\sum_{k=1}^{t} x_k^2\right) p(x_{1:t}|y_{1:t})\, dx_{1:t}$$

using Kalman techniques.

- We compute the SMC estimate of this quantity using $\widehat{S}_t/t$ where

$$\widehat{S}_t = \int \left(\sum_{k=1}^{t} x_k^2\right) \widehat{p}(x_{1:t}|y_{1:t})\, dx_{1:t}$$

can be computed sequentially.

Another Illustration of the Degeneracy Phenomenon

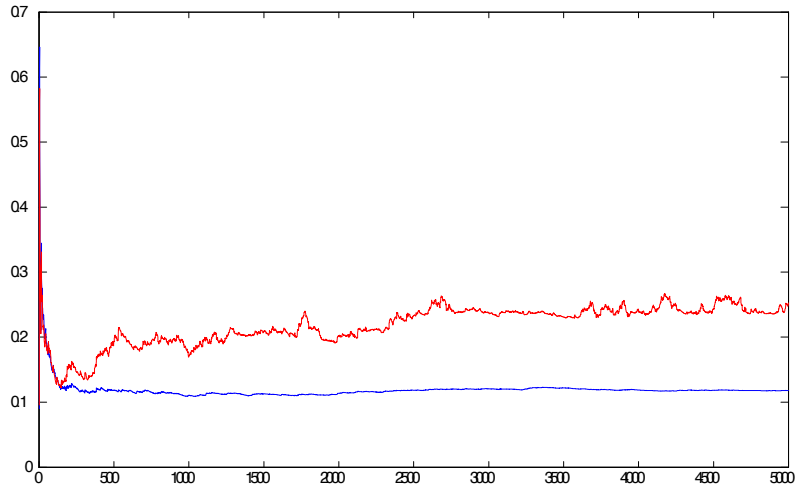


Figure: S_t/t obtained through the Kalman smoother (blue) and its SMC estimate $\widehat{S}_t/t$ (red).

Some Convergence Results for SMC

- Numerous convergence results for SMC are available; see Del Moral (2004,2013).
- Let $\varphi_t : \mathcal{X}^t \to \mathbb{R}$ and consider

$$\overline{\varphi}_t = \int \varphi_t(x_{1:t})\, p(x_{1:t}|y_{1:t})\, dx_{1:t},$$

$$\widehat{\varphi}_t = \int \varphi_t(x_{1:t})\, \widehat{p}(x_{1:t}|y_{1:t})\, dx_{1:t} = \frac{1}{N}\sum_{i=1}^{N} \varphi_t\left(X_{1:t}^{(i)}\right).$$

- We can prove that for any bounded function φ and any $p \geq 1$

$$\mathbb{E}\left[|\widehat{\varphi}_t - \overline{\varphi}_t|^p\right]^{1/p} \leq \frac{B(t)\, c(p)\, \|\varphi\|_\infty}{\sqrt{N}},$$

$$\lim_{N\to\infty} \sqrt{N}\left(\widehat{\varphi}_t - \overline{\varphi}_t\right) \Rightarrow \mathcal{N}\left(0, \sigma_t^2\right).$$

- **Very weak results**: $B(t)$ and σ_t^2 can increase with t and will for a path-dependent $\varphi_t(x_{1:t})$ as the degeneracy problem suggests.

Stronger Convergence Results

- Assume the following **exponentially stability assumption**: For any x_1, x_1'

$$\frac{1}{2}\int \left|p(x_t|y_{2:t}, X_1 = x_1) - p(x_t|y_{2:t}, X_1 = x_1')\right| dx_t \leq \alpha^t \text{ for } 0 \leq \alpha < 1.$$

- **Marginal distribution**. For $\varphi_t(x_{1:t}) = \varphi(x_{t-L:t})$, there exists $B_1, B_2 < \infty$ s.t.

$$\mathbb{E}\left[|\widehat{\varphi}_t - \overline{\varphi}_t|^p\right]^{1/p} \leq \frac{B_1\ c(p)\ \ \|\varphi\|_\infty}{\sqrt{N}},$$

$$\lim_{N\to\infty} \sqrt{N}\left(\widehat{\varphi}_t - \overline{\varphi}_t\right) \Rightarrow \mathcal{N}\left(0, \sigma_t^2\right) \text{ where } \sigma_t^2 \leq B_2,$$

  i.e. there is no accumulation of numerical errors over time.
- **L1 distance.** If $\overline{p}(x_{1:t}|y_{1:t}) = \mathbb{E}\left(\widehat{p}(x_{1:t}|y_{1:t})\right)$, there exists $B_3 < \infty$ s.t.

$$\int \left|\overline{p}(x_{1:t}|y_{1:t}) - p(x_{1:t}|y_{1:t})\right| dx_{1:t} \leq \frac{B_3\ t}{N};$$

  i.e. the bias only increases in t.

Stronger Convergence Results

- **Unbiasedness**. The marginal likelihood estimate is unbiased

$$\mathbb{E}\left(\widehat{p}(y_{1:t})\right) = p(y_{1:t}).$$

- **Central Limit Theorem**. There exists $B_5 < \infty$ s.t.

$$\lim_{N\to\infty} \sqrt{N}\log \widehat{p}(y_{1:t})/p(y_{1:t}) \Rightarrow \mathcal{N}\left(0, \overline{\sigma}_t^2\right) \text{ with } \overline{\sigma}_t^2 \leq B_5\ t.$$

- **Relative Variance Bound**. Under exponential stability assumptions, there exists $B_4 < \infty$

$$\mathbb{E}\left(\left(\frac{\widehat{p}(y_{1:t})}{p(y_{1:t})} - 1\right)^2\right) \leq \frac{B_4\ t}{N}$$

- **Another Central Limit Theorem**. Under exponential stability assumptions, for $N = \alpha^{-1} T$

$$\lim_{T\to\infty} \log \widehat{p}(y_{1:t})/p(y_{1:t}) \Rightarrow \mathcal{N}\left(-\frac{\alpha^2}{2}\sigma^2, \alpha^2\sigma^2\right).$$

Summary

- SMC provide consistent estimates under weak assumptions.
- Under stability assumptions, uniform in time stability of the SMC estimates of $\{p(x_t|y_{1:t})\}_{t\geq 1}$.
- Under stability assumptions, relative variance of the SMC estimate of $\{p(y_{1:t})\}_{t\geq 1}$ only increases linearly with t.
- Even under stability assumptions, one cannot expect to obtain uniform in time stability for SMC estimates of $\{p(x_{1:t}|y_{1:t})\}_{t\geq 1}$; this is due to the degeneracy problem.
- Is it possible to Q1: eliminate, Q2: mitigate the degeneracy problem?
- Answer: Q1: no, Q2: yes.

Is Resampling Really Necessary?

- Resampling is the source of the degeneracy problem and might appear wasteful.
- The resampling step is an unbiased operation

$$\mathbb{E}\left[\widehat{p}(x_{1:t}|y_{1:t})|\widetilde{p}(x_{1:t}|y_{1:t})\right]=\widetilde{p}(x_{1:t}|y_{1:t})$$

but clearly it introduces some errors "locally" in time. That is for any test function, we have

$$\mathbb{V}\left[\int\varphi(x_{1:t})\widehat{p}(x_{1:t}|y_{1:t})dx_{1:t}\right]\geq\mathbb{V}\left[\int\varphi(x_{1:t})\widetilde{p}(x_{1:t}|y_{1:t})dx_{1:t}\right]$$

- What about eliminating the resampling step?

Sequential Importance Samping: SMC Without Resampling

- In this case, the estimate of the posterior is

$$\widehat{p}_{\mathrm{SIS}}(x_{1:t}|y_{1:t})=\sum_{i=1}^{N}W_t^{(i)}\delta_{X_{1:t}^{(i)}}(x_{1:t})$$

where $X_{1:t}^{(i)}\sim p(x_{1:t})$ and

$$W_t^{(i)}\propto p\left(y_{1:t}|X_{1:t}^{(i)}\right)=\prod_{k=1}^{t}g\left(y_k|X_t^{(i)}\right).$$

- In this case, the marginal likelihood estimate is

$$\widehat{p}_{\mathrm{SIS}}(y_{1:t})=\frac{1}{N}\sum_{i=1}^{N}p\left(y_{1:t}|X_{1:t}^{(i)}\right)$$

- Relative variance of $p\left(y_{1:t}|X_{1:t}^{(i)}\right)=\prod_{k=1}^{t}g\left(y_k|X_t^{(i)}\right)$ is increasing exponentially fast...

SIS For Stochastic Volatility Model

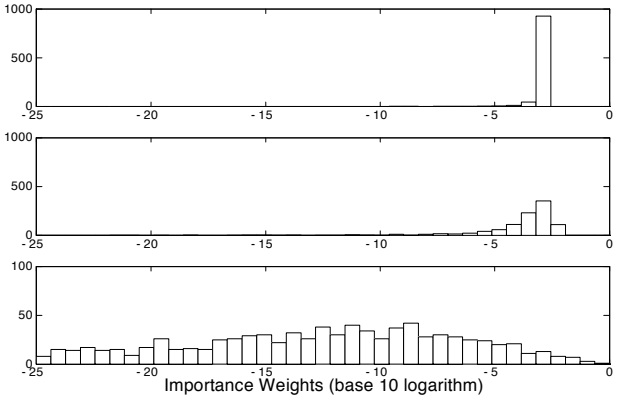


Figure: Histograms of $\log_{10}\left(W_t^{(i)}\right)$ for $t=1$ (top), $t=50$ (middle) and $t=100$ (bottom).

- The algorithm performance collapse as t increases as expected.

Central Limit Theorems

- For both SIS and SMC, we have a CLT for the estimates of the marginal likelihood

$$\sqrt{N}\left(\frac{\widehat{p}_{\mathrm{SIS}}\left(y_{1:t}\right)}{p\left(y_{1:t}\right)}-1\right) \Rightarrow \mathcal{N}\left(0, \sigma_{t,\mathrm{SIS}}^2\right),$$
$$\sqrt{N}\left(\frac{\widehat{p}_{\mathrm{SMC}}\left(y_{1:t}\right)}{p\left(y_{1:t}\right)}-1\right) \Rightarrow \mathcal{N}\left(0, \sigma_{t,\mathrm{SMC}}^2\right).$$

- The variance expressions are

$$\begin{aligned}\sigma_{t,\mathrm{SIS}}^2 &= \int \frac{p^2(x_{1:t}|y_{1:t})}{p(x_{1:t})} dx_{1:t} - 1 = \frac{\int p^2(y_{1:t}|x_{1:t}) p(x_{1:t}) dx_{1:t}}{p^2(y_{1:t})} - 1 \\ \sigma_{t,\mathrm{SMC}}^2 &= \int \frac{p^2(x_1|y_{1:t})}{\mu(x_1)} dx_1 + \sum_{k=2}^t \int \frac{p^2(x_{1:k}|y_{1:t})}{p(x_{1:k-1}|y_{1:k-1}) f(x_k|x_{k-1})} dx_{1:k} - t \\ &= \frac{\int g^2(y_1|x_1)\mu(x_1)dx_1}{p^2(y_1)} + \sum_{k=2}^t \frac{\int p^2(y_{k:t}|x_k) p(x_k|y_{1:k-1}) dx_k}{p^2(y_{k:t}|y_{1:k-1})} - t\end{aligned}$$

- SMC "breaks" the integral over $\mathcal{X}^t$ into t integrals over $\mathcal{X}$.

A Toy Example

- Consider the case where $f\left(x'|x\right) = \mu\left(x'\right) = \mathcal{N}\left(x'; 0, \sigma^2\right)$ and $g\left(y|x\right) = \mathcal{N}\left(y; 0, 1-\frac{1}{\sigma^2}\right)$ where $\sigma^2 > 1$.
- Assume we observe $y_1 = \cdots = y_t = 0$ then we have

$$\mathbb{V}\left(\frac{\widehat{p}_{\mathrm{SIS}}\left(y_{1:t}\right)}{p\left(y_{1:t}\right)}\right) = \frac{\sigma_{t,\mathrm{SIS}}^2}{N} = \frac{1}{N}\left[\left(\frac{\sigma^4}{2\sigma^2-1}\right)^{t/2} - 1\right],$$
$$\mathbb{V}\left(\frac{\widehat{p}_{\mathrm{SMC}}\left(y_{1:t}\right)}{p\left(y_{1:t}\right)}\right) \approx \frac{\sigma_{t,\mathrm{SMC}}^2}{N} = \frac{t}{N}\left[\left(\frac{\sigma^4}{2\sigma^2-1}\right)^{1/2} - 1\right].$$

- If select $\sigma^2 = 1.2$ then it is necessary to use $N \approx 2 \times 10^{23}$ particles to obtain $\frac{\sigma_{t,\mathrm{SIS}}^2}{N} = 10^{-2}$ for $t = 1000$.
- To obtain $\frac{\sigma_{t,\mathrm{SMC}}^2}{N} = 10^{-2}$, SMC requires only $N \approx 10^4$ particles: improvement by 19 orders of magnitude!

Better Resampling Schemes

- Better resampling steps can be designed such that $\mathbb{E}\left[N_t^{(i)}\right] = NW_t^{(i)}$ but $\mathbb{V}\left[N_t^{(i)}\right] < NW_t^{(i)}\left(1 - W_t^{(i)}\right)$; residual resampling, minimal entropy resampling etc. (Cappé et al., 2005).
- *Residual Resampling*. Set $\widetilde{N}_t^{(i)} = \left\lfloor NW_t^{(i)} \right\rfloor$, sample $\overline{N}_t^{1:N}$ from a multinomial of parameters $\left(N, \overline{W}_t^{(1:N)}\right)$ where $\overline{W}_t^{(i)} \propto W_t^{(i)} - N^{-1}\widetilde{N}_t^{(i)}$ then set $N_t^{(i)} = \widetilde{N}_t^{(i)} + \overline{N}_t^{(i)}$.
- *Systematic Resampling*. Sample $U_1 \sim \mathcal{U}\left[0, \frac{1}{N}\right]$ and define $U_i = U_1 + \frac{i-1}{N}$ for $i = 2, ..., N$, then set $N_t^i = \left|\left\{U_j : \sum_{k=1}^{i-1} W_t^{(k)} \leq U_j \leq \sum_{k=1}^{i} W_t^{(k)}\right\}\right|$ with the convention $\sum_{k=1}^{0} := 0$.

Measuring Variability of the Weights

- To measure the variation of the weights, we can use the Effective Sample Size (ESS)

$$ESS = \left(\sum_{i=1}^N \left(W_t^{(i)}\right)^2\right)^{-1}$$

- We have $ESS = N$ if $W_t^{(i)} = 1/N$ for any i and $ESS = 1$ if $W_t^{(i)} = 1$ and $W_t^{(j)} = 1$ for $j \neq i$.
- Liu (1996) showed that for simple importance sampling for φ "regular enough"

$$\mathbb{V}\left(\sum_{i=1}^N W_t^{(i)} \varphi\left(X_t^{(i)}\right)\right) \approx \mathbb{V}_{p(x_{1:t}|y_{1:t})}\left(\frac{1}{ESS}\sum_{i=1}^{ESS} \varphi\left(X_t^{(i)}\right)\right);$$

i.e. the estimate is roughly as accurate as using an iid sample of size ESS from $p\left(x_{1:t}|y_{1:t}\right)$.

Dynamic Resampling

- Resampling at each time step can be harmful: only resample when necessary.
- **Dynamic Resampling**: If the variation of the weights as measured by ESS is too high, e.g. $ESS < N/2$, then resample the particles.
- We can also use the entropy

$$Ent = -\sum_{i=1}^{N} W_t^{(i)} \log_2\left(W_t^{(i)}\right)$$

- We have $Ent = \log_2(N)$ if $W_t^{(i)} = 1/N$ for any i. We have $Ent = 0$ if $W_t^{(i)} = 1$ and $W_t^{(j)} = 1$ for $j \neq i$.

A General Bayesian Recursion

- Introduce an arbitrary proposal distribution $q(x_t|y_t, x_{t-1})$; i.e. an approximation to $p(x_t|y_t, x_{t-1})$.
- We have seen that

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

so clearly

$$p(x_{1:t}|y_{1:t}) = \frac{w(x_{t-1}, x_t, y_t)\, q(x_t|y_t, x_{t-1})\, p(x_{1:t-1}|y_{1:t-1})}{p(y_t|y_{1:t-1})}$$

where

$$w(x_{t-1}, x_t, y_t) = \frac{g(y_t|x_t)\ f(x_t|x_{t-1})}{q(x_t|y_t, x_{t-1})}$$

- This suggests a more general SMC algorithm.

Improving the Sampling Step

- **Bootstrap filter**. Sample particles blindly according to the prior without taking into account the observation
  $\rightsquigarrow$ Very inefficient for vague prior/peaky likelihood.
- **Optimal proposal/Perfect adaptation**. Implement the following alternative update-propagate Bayesian recursion

$$\begin{aligned} \text{Update} \quad & p(x_{1:t-1}|y_{1:t}) = \frac{p(y_t|x_{t-1})p(x_{1:t-1}|y_{1:t-1})}{p(y_t|y_{1:t-1})} \\ \text{Propagate} \quad & p(x_{1:t}|y_{1:t}) = p(x_{1:t-1}|y_{1:t})\, p(x_t|y_t, x_{t-1}) \end{aligned}$$

where

$$p(x_t|y_t, x_{t-1}) = \frac{f(x_t|x_{t-1})\, g(y_t|x_{t-1})}{p(y_t|x_{t-1})}$$

$\rightsquigarrow$ Much more efficient when applicable; e.g.
$f(x_t|x_{t-1}) = \mathcal{N}(x_t; \varphi(x_{t-1}), \Sigma_v)$, $g(y_t|x_t) = \mathcal{N}(y_t; x_t, \Sigma_w)$.

A General SMC Algorithm

Assume we have N weighted particles $\left\{W_{t-1}^{(i)}, X_{1:t-1}^{(i)}\right\}$ approximating $p(x_{1:t-1}|y_{1:t-1})$ then at time t,

- Sample $\widetilde{X}_t^{(i)} \sim q\left(x_t|y_t, X_{t-1}^{(i)}\right)$, set $\widetilde{X}_{1:t}^{(i)} = \left(X_{1:t-1}^{(i)}, \widetilde{X}_t^{(i)}\right)$ and

$$\widetilde{p}(x_{1:t}|y_{1:t}) = \sum_{i=1}^{N} W_t^{(i)} \delta_{\widetilde{X}_{1:t}^{(i)}}(x_{1:t}),$$

$$W_t^{(i)} \propto W_{t-1}^{(i)} \frac{f\left(\widetilde{X}_t^{(i)} \middle| X_{t-1}^{(i)}\right) g\left(y_t|\widetilde{X}_t^{(i)}\right)}{q\left(\widetilde{X}_t^{(i)} \middle| y_t, X_{t-1}^{(i)}\right)}.$$

- If ESS$< N/2$ resample $X_{1:t}^{(i)} \sim \widetilde{p}(x_{1:t}|y_{1:t})$ and set $W_t^{(i)} \leftarrow \frac{1}{N}$ to obtain $\widehat{p}(x_{1:t}|y_{1:t}) = \frac{1}{N}\sum_{i=1}^{N} \delta_{X_{1:t}^{(i)}}(x_{1:t})$.

Building Proposals

- Our aim is to select $q(x_t|y_t, x_{t-1})$ as "close" as possible to $p(x_t|y_t, x_{t-1})$ as this minimizes the variance of

$$w(x_{t-1}, x_t, y_t) = \frac{g(y_t|x_t)\ f(x_t|x_{t-1})}{q(x_t|y_t, x_{t-1})}.$$

- **Example - EKF proposal**: Let

$$X_t = \varphi(X_{t-1}) + V_t,\ \ Y_t = \Psi(X_t) + W_t,$$

with $V_t \sim \mathcal{N}(0, \Sigma_v)$, $W_t \sim \mathcal{N}(0, \Sigma_w)$. We perform local linearization

$$Y_t \approx \Psi(\varphi(X_{t-1})) + \left.\frac{\partial \Psi(x)}{\partial x}\right|_{\varphi(X_{t-1})} (X_t - \varphi(X_{t-1})) + W_t$$

and use as a proposal.

$$q(x_t|y_t, x_{t-1}) \propto \widehat{g}(y_t|x_t)\ f(x_t|x_{t-1}).$$

- Any standard suboptimal filtering methods can be used: Unscented Particle filter, Gaussan Quadrature particle filter etc.

Implicit Proposals

- Proposed recently by Chorin (2012). Let

$$F(x_{t-1}, x_t) = \log g(y_t|x_t) + \log\ f(x_t|x_{t-1})$$

and

$$x_t^* = \arg\max F(x_{t-1}, x_t) = \arg\max\ p(x_t|y_t, x_{t-1})$$

- We sample $Z \sim \mathcal{N}(0, I_{n_x})$, then we solve in X_t

$$F(x_{t-1}, x_t^*) - F(x_{t-1}, X_t) = \frac{1}{2} Z^{\mathrm{T}} Z, \quad Z \sim \mathcal{N}(0, I_{n_x})$$

so if there is a unique solution

$$\begin{aligned} q(x_t|y_t, x_{t-1}) &= p_Z(z) \left|\det \partial z/\partial x_t\right| \\ &\propto \frac{\exp(-F(x_{t-1}, x_t^*))}{\left|\det \partial x_t/\partial z\right|} g(y_t|x_t)\ f(x_t|x_{t-1}) \end{aligned}$$

- The incremental weight is

$$\frac{g(y_t|x_t)\ f(x_t|x_{t-1})}{q(x_t|y_t, x_{t-1})} \propto \left|\det \partial x_t/\partial z\right| \exp(F(x_{t-1}, x_t^*))$$

Block Sampling Proposals

- **Problem**: we only sample X_t at time t so, even if you use $p(x_t|y_t, x_{t-1})$, the SMC estimates could have high variance if $\mathbb{V}_{p(x_{t-1}|y_{1:t-1})}[p(y_t|x_{t-1})]$ is high.
- **Block sampling idea**: allows yourself to sample again $X_{t-L+1:t-1}$ as well as X_t in light of y_t. Optimally we would like at time t to sample

$$X_{t-L+1:t}^{(i)} \sim p\left(x_{t-L+1:t}|y_{t-L+1:t}, X_{t-L}^{(i)}\right)$$

and

$$\begin{aligned} W_t^{(i)} &\propto W_{t-1}^{(i)} \frac{p\left(X_{1:t}^{(i)}\middle| y_{1:t}\right)}{p\left(X_{1:t-L}^{(i)}\middle| y_{1:t-1}\right) p\left(X_{t-L+1:t}^{(i)}\middle| y_{t-L+1:t}, X_{t-L}^{(i)}\right)} \\ &\propto W_{t-1}^{(i)} p\left(y_t|y_{t-L+1:t-1}, X_{t-L}^{(i)}\right) \end{aligned}$$

- When $p(x_{t-L+1:t}|y_{t-L+1:t}, x_{t-L})$ and $p(y_t|y_{t-L+1:t-1}, x_{t-L})$ are not available, we can use analytical approximations of them and still have consistent estimates (D., Briers & Senecal, 2006).

Block Sampling Proposals

- Computational cost is increased from $\mathcal{O}(N)$ to $\mathcal{O}(LN)$ so is it worth it?
- Consider the ideal scenario where

$$\begin{aligned} X_t &= X_{t-1} + V_t \\ Y_t &= X_t + W_t \end{aligned}$$

where $X_1 \sim \mathcal{N}(0, 1)$ and $V_t, W_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, 1)$.

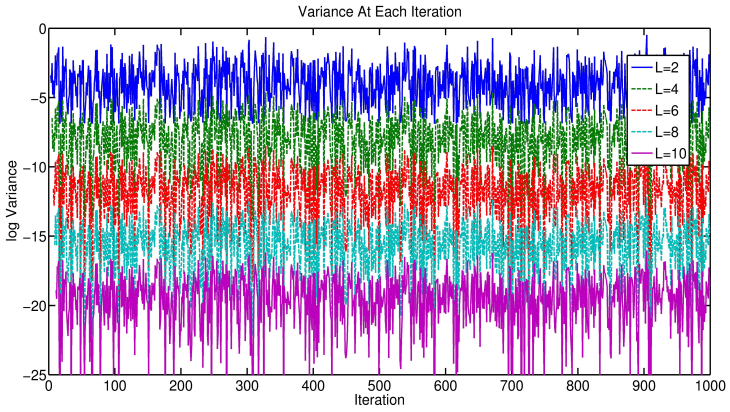
- In this case, we have

$$|p(y_t|y_{t-L+1:t-1}, x_{t-L}) - p(y_t|y_{t-L+1:t-1}, x'_{t-L})| < c|x_{t-L} - x'_{t-L}|/2^L$$

where the rate of exponential convergence depends upon the signal-to-noise ratio if more general Gaussian AR are considered.

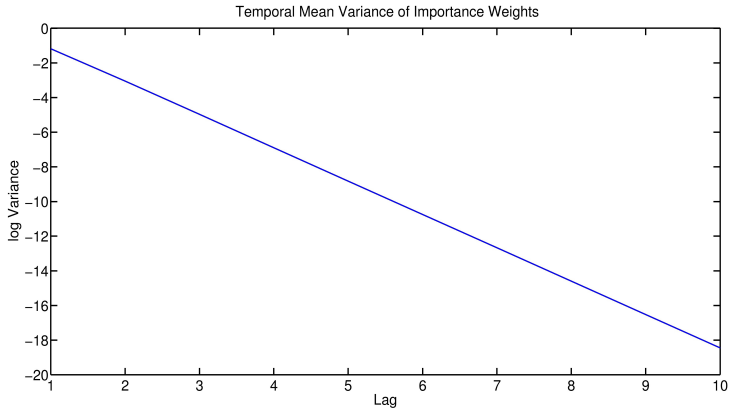
- We can obtain an analytic expression of the variance of the (normalized) weight.

Block Sampling Proposals



Variance of incremental weight w.r.t. $p\left(x_{1:t-L}\middle|\, y_{1:t-1}\right)$.

Block Sampling Proposals



Time averaged variance of of incremental weight w.r.t. $p\left(x_{1:t-L}\middle|\, y_{1:t-1}\right)$.

Fighting Degeneracy Using MCMC Steps

- The design of "good" proposals can be complicated and/or time consuming so, after the resampling step, a few particles might inherit many offspring.
- A standard way to limit degeneracy is known as the Resample-Move algorithm (Gilks & Berzuini, 2001); i.e. using MCMC kernels as a principled way to "jitter" the particle locations.
- A MCMC kernel $K_t\left(x'_{1:t}\middle|\, x_{1:t}\right)$ of invariant distribution $p\left(x_{1:t}\middle|\, y_{1:t}\right)$ is a Markov transition kernel with the property that

$$p\left(x'_{1:t}\middle|\, y_{1:t}\right) = \int p\left(x_{1:t}\middle|\, y_{1:t}\right) K_t\left(x'_{1:t}\middle|\, x_{1:t}\right) dx_{1:t},$$

i.e. if $X_{1:t} \sim p\left(x_{1:t}\middle|\, y_{1:t}\right)$ and $X'_{1:t}\middle|\, X_{1:t} \sim K_t\left(x'_{1:t}\middle|\, X_{1:t}\right)$ then marginally $X'_{1:t} \sim p\left(x_{1:t}\middle|\, y_{1:t}\right)$.

Fighting Degeneracy Using MCMC Steps

- *Example 1: Gibbs moves*. Set $X'_{1:t-L} = X_{1:t-L}$ then sample X'_{t-L+1} from $p\left(x_{t-L+1}\middle|\, y_{t-L+1}, x'_{t-L}, x_{t-L+2}\right)$, sample X'_{t-L+2} from $p\left(x_{t-L+2}\middle|\, y_{t-L+2}, x'_{t-L+1}, x_{t-L+3}\right)$ and so on until we sample X'_t from $p\left(x_t\middle|\, y_t, x'_{t-1}\right)$; that is

$$K_t\left(x'_{1:t}\middle|\, x_{1:t}\right) = \delta_{x_{1:t-L}}\left(x'_{1:t-L}\right) \prod_{k=t-L+1}^{t-1} p\left(x'_k\middle|\, y_k, x'_{k-1}, x_{k+1}\right) \times p\left(x'_t\middle|\, y_t, x'_{t-1}\right)$$

- *Example 2: Metropolis-Hastings moves*. Set $X'_{1:t-L} = X_{1:t-L}$ then sample X^*_{t-L+1} from $q\left(x'_{t-L+1:t}\middle|\, x_{t-L}, x_{t-L+1:t}\right)$ and set $X'_{t-L+1} = X^*_{t-L+1}$ with proba.

$$1 \wedge \frac{p\left(x^*_{t-L+1:t}\middle|\, y_{t-L+1}, x_{t-L}\right)}{p\left(x_{t-L+1:t}\middle|\, y_{t-L+1}, x_{t-L}\right)} \frac{q\left(x_{t-L+1:t}\middle|\, x_{t-L}, x^*_{t-L+1:t}\right)}{q\left(x^*_{t-L+1:t}\middle|\, x_{t-L}, x_{t-L+1:t}\right)},$$

otherwise set $X'_{t-L+1} = X_{t-L+1}$.
- Contrary to MCMC, we typically do not use ergodic kernels in SMC.

Example: Bearings-only-tracking

- Target modelled using a standard constant velocity model

$$X_t = AX_{t-1} + V_t$$

where $V_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \Sigma)$. The state vector $X_t = \begin{pmatrix} X_t^1 & X_t^2 & X_t^3 & X_t^4 \end{pmatrix}^{\mathrm{T}}$ contains location and velocity components.

- One only receives observations of the bearings of the target

$$Y_t = \tan^{-1}\left(\frac{X_t^3}{X_t^1}\right) + W_t$$

where $W_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}\left(0, 10^{-4}\right)$; i.e. the observations are almost noiseless.

- We compare Bootstrap filter, SMC-EKF with $L = 5, 10$, MCMC moves $L = 5, 10$ using dynamic resampling.

Degeneracy for Various Proposals

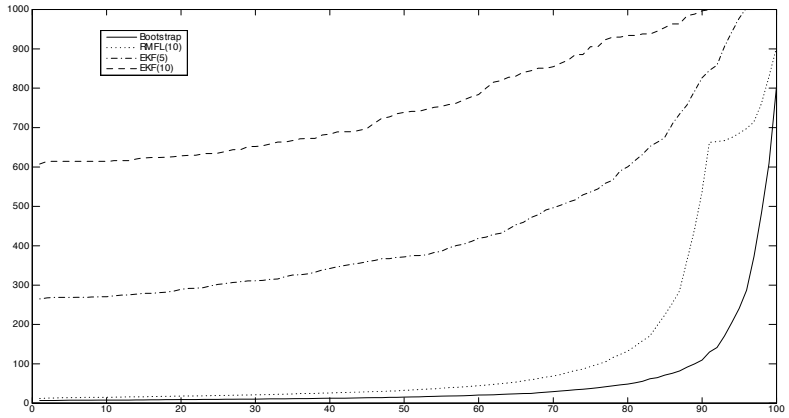


Figure: Average number of unique particles $X_t^{(i)}$ approximating $p(x_t | y_{1:100})$; time on x-axis, average number of unique particles on y-axis.

Summary

- SMC provide consistent estimates under weak assumptions.
- We can estimate $\{p(x_t | y_{1:t})\}_{t \geq 1}$ satisfactorily but our approximations of $\{p(x_{1:t} | y_{1:t})\}_{t \geq 1}$ degenerates as t increases because of resampling steps.
- Resampling is crucial.
- We can mitigate but not eliminate the degeneracy problem by the design of "clever" proposals.
- Smoothing methods to estimate $p(x_{1:T} | y_{1:T})$ can come to the rescue.

Smoothing in State-Space Models

- **Smoothing problem**: given a fixed time T, we are interested in $p(x_{1:T} | y_{1:T})$ or some of its marginals, e.g. $\{p(x_t | y_{1:T})\}_{t=1}^{T}$.
- Smoothing is crucial to parameter estimation.
- Direct SMC approximations of $p(x_{1:T} | y_{1:T})$ and its marginals $p(x_k | y_{1:T})$ are poor if T is large.
- SMC provide "good" approximations of marginals $\{p(x_t | y_{1:t})\}_{t \geq 1}$. This can be used to develop efficient smoothing estimates.
  - $\rightsquigarrow$ Fixed-lag smoothing
  - $\rightsquigarrow$ Forward-backward smoothing
  - $\rightsquigarrow$ (Generalized) two-filter smoothing

Fixed-Lag Smoothing

- The fixed-lag smoothing approximation relies on

$$p\left(x_t \middle| y_{1:T}\right) \approx p\left(x_t \middle| y_{1:t+\Delta}\right) \text{ for } \Delta \text{ large enough.}$$

  and quantitative bounds can be established under stability assumptions.
- This can be exploited by SMC methods (Kitagawa & Sato, 2001)
- **Algorithmically**: stop resampling $\left\{X_t^{(i)}\right\}$ beyond time $t+\Delta$ (Kitagawa & Sato, 2001).
- Computational cost is $\mathcal{O}(N)$ but non-vanishing bias as $N \longrightarrow \infty$ (Olsson & al., 2008).
- Picking Δ is difficult: Δ too small results in $p\left(x_t \middle| y_{1:t+\Delta}\right)$ being a poor approximation of $p\left(x_t \middle| y_{1:T}\right)$. Δ too large improves the approximation but degeneracy creeps in.

Forward Backward Smoothing

- Forward Backward (FB) decomposition states

$$\begin{aligned} p\left(x_{1:T} \middle| y_{1:T}\right) &= p\left(x_T \middle| y_{1:T}\right) \prod_{t=1}^{T-1} p\left(x_t \middle| y_{1:T}, x_{t+1:T}\right) \\ &= p\left(x_T \middle| y_{1:T}\right) \prod_{t=1}^{T-1} p\left(x_t \middle| y_{1:t}, x_{t+1}\right) \end{aligned}$$

  where

$$p\left(x_t \middle| y_{1:t}, x_{t+1}\right) = \frac{f\left(x_{t+1} \middle| x_t\right) p\left(x_t \middle| y_{1:t}\right)}{p\left(x_{t+1} \middle| y_{1:t}\right)}.$$

- Conditioned upon $y_{1:T}$, $\{X_t\}_{t=1}^T$ is a backward Markov chain of initial distribution $p\left(x_T \middle| y_{1:T}\right)$ and inhomogeneous Markov transitions $\left\{p\left(x_t \middle| y_{1:t}, x_{t+1}\right)\right\}_{t=1}^{T-1}$.

Forward Filtering Backward Sampling

- To obtain a sample from $p\left(x_{1:T} \middle| y_{1:T}\right)$,
  - **Forward filtering**: compute and store $\left\{p\left(x_t \middle| y_{1:t}\right)\right\}_{t=1}^T$
  - **Backward sampling**: sample $X_T \sim p\left(x_T \middle| y_{1:T}\right)$ then for $t = T-1, ..., 1$ sample $X_t \sim p\left(x_t \middle| y_{1:t}, X_{t+1}\right)$.
- SMC to obtain an approximate sample from $p\left(x_{1:T} \middle| y_{1:T}\right)$
  - **Forward filtering**: compute and store $\left\{\widehat{p}\left(x_t \middle| y_{1:t}\right)\right\}_{t=1}^T$.
  - **Backward sampling**: sample $X_T \sim \widehat{p}\left(x_T \middle| y_{1:T}\right)$ then for $t = T-1, ..., 1$ sample $X_t \sim \widehat{p}\left(x_t \middle| y_{1:t}, X_{t+1}\right)$ where

$$\begin{aligned} \widehat{p}\left(x_t \middle| y_{1:t}, X_{t+1}\right) &\propto f\left(X_{t+1} \middle| x_t\right) \widehat{p}\left(x_t \middle| y_{1:t}\right) \\ &\propto \sum_{i=1}^N f\left(X_{t+1} \middle| X_t^{(i)}\right) \delta_{X_t^{(i)}}(x_t) \end{aligned}$$

- Direct implementation $\mathcal{O}(NT)$ (Godsill, D. & West, 2004). Rejection sampling possible if $f\left(x_{t+1} \middle| x_t\right) \leq C\left(x_{t+1}\right)$ (Douc et al., 2011) and cost $\mathcal{O}(NT)$.

Forward Filtering Backward Smoothing

- Assume you want to compute the marginal smoothing distributions $\left\{p\left(x_t \middle| y_{1:T}\right)\right\}_{t=1}^T$ instead of sampling from them.
- **Forward filtering Backward smoothing** (FFBS).

$$\begin{aligned} \overbrace{p\left(x_t \middle| y_{1:T}\right)}^{\text{smoother at } t} &= \int p\left(x_t, x_{t+1} \middle| y_{1:T}\right) dx_{t+1} \\ &= \int p\left(x_{t+1} \middle| y_{1:T}\right) p\left(x_t \middle| y_{1:t}, x_{t+1}\right) dx_{t+1} \\ &= \int \overbrace{p\left(x_{t+1} \middle| y_{1:T}\right)}^{\text{smoother at } t+1} \underbrace{\frac{f\left(x_{t+1} \middle| x_t\right) \overbrace{p\left(x_t \middle| y_{1:t}\right)}^{\text{filter at } t}}{p\left(x_{t+1} \middle| y_{1:t}\right)}}_{\text{backward transition } p\left(x_t \middle| y_{1:t}, x_{t+1}\right)} dx_{t+1}. \end{aligned}$$

- For finite state-space HMM, it is surprisingly and unfortunately not the recursion usually implemented (Rabiner et al., 1989).

SMC Forward Filtering Backward Smoothing

- **Forward filtering**: compute and store $\{\widehat{p}(x_t|y_{1:t})\}_{t=1}^T$ using your favourite SMC.
- **Backward smoothing**: For $t = T-1, ..., 1$, we have $\widehat{p}(x_t|y_{1:T}) = \sum_{i=1}^N W_{t|T}^{(i)} \delta_{X_t^{(i)}}(x_t)$ with $W_{T|T}^{(i)} = 1/N$ and

$$\widehat{p}(x_t|y_{1:T}) = \underbrace{\widehat{p}(x_t|y_{1:t})}_{\frac{1}{N}\sum_{i=1}^N \delta_{X_t^{(i)}}(x_t)} \int \underbrace{\widehat{p}(x_{t+1}|y_{1:T})}_{\sum_{j=1}^N W_{t+1|T}^{(j)} \delta_{X_{t+1}^{(j)}}(x_{t+1})} \frac{f(x_{t+1}|x_t)}{\int f(x_{t+1}|x_t)\widehat{p}(x_t|y_{1:t})dx_t} dx_{t+1}$$
$$= \sum_{i=1}^N W_{t|T}^{(i)} \delta_{X_t^{(i)}}(x_t)$$

where

$$W_{t|T}^{(i)} = \sum_{j=1}^N W_{t+1|T}^{(j)} \frac{f\left(X_{t+1}^{(j)}|X_t^{(i)}\right)}{\sum_{l=1}^N f\left(X_{t+1}^{(j)}|X_t^{(l)}\right)}.$$

- Computational complexity is $\mathcal{O}(TN^2)$.

Two-Filter Smoothing

- An alternative to FB smoothing is the Two-Filter (TF) formula

$$p(x_t, x_{t+1}|y_{1:T}) \propto \overbrace{p(x_t|y_{1:t})}^{\text{forward filter}} f(x_{t+1}|x_t) \overbrace{p(y_{t+1:T}|x_{t+1})}^{\text{backward filter}}$$

- The backward information filter satisfies $p(y_T|x_T) = g(y_T|x_T)$ and

$$p(y_{t:T}|x_t) = \int p(y_t, y_{t+1:T}, x_{t+1}|x_t)\, dx_{t+1}$$
$$= g(y_t|x_t) \int p(y_{t+1:T}|x_{t+1})\, f(x_{t+1}|x_t)\, dx_{t+1}$$

- Various particle methods have been proposed to approximate $\{p(y_{t:T}|x_t)\}_{t=1}^T$ but rely implicitly on $\int p(y_{t:T}|x_t)\, dx_t < \infty$ and try to come up with a backward dynamics; e.g. solve

$$X_{t+1} = \varphi(X_t, V_{t+1}) \Leftrightarrow X_t = \varphi^{-1}(X_t, V_{t+1}).$$

This is incorrect.

Generalized Two-Filter Smoothing

- **Generalized Two-Filter smoothing** (Briers, D. & Maskell, 2004-2010)

$$p(x_t, x_{t+1}|y_{1:T}) \propto \frac{\overbrace{p(x_t|y_{1:t})}^{\text{forward filter}} f(x_{t+1}|x_t) \overbrace{\overline{p}(x_{t+1}|y_{t+1:T})}^{\text{backward filter}}}{\underbrace{\overline{p}(x_{t+1})}_{\text{artificial prior}}}$$

where

$$\overline{p}(x_{t+1}|y_{t+1:T}) \propto p(y_{t+1:T}|x_{t+1})\, \overline{p}(x_{t+1}).$$

- By construction, we now have integrable $\overline{p}(x_{t+1}|y_{t+1:T})$ which we can approximate using a backward SMC algorithm targeting $\{\overline{p}(x_{t+1:T}|y_{t+1:T})\}_{t=T}^1$ where

$$\overline{p}(x_t|y_{t:T}) \propto \overline{p}(x_t) \prod_{k=t+1}^T f(x_k|x_{k-1}) \prod_{k=t}^T g(y_k|x_k).$$

SMC Generalized Two-Filter Smoothing

- **Forward filter**: compute and store $\{\widehat{p}(x_t|y_{1:t})\}_{t=1}^T$ using your favourite SMC.
- **Backward filter**: compute and store $\{\widehat{\overline{p}}(x_t|y_{t:T})\}_{t=1}^T$ using your favourite SMC.
- **Combination step**: for any $t \in \{1, ..., T\}$ we have

$$\widehat{p}(x_t, x_{t+1}|y_{1:T}) \propto \widehat{p}(x_t|y_{1:T}) \frac{f(x_{t+1}|x_t)}{\overline{p}(x_{t+1})} \widehat{\overline{p}}(x_{t+1}|y_{t+1:t})$$
$$\propto \sum_{i=1}^N \sum_{j=1}^N \frac{f\left(\overline{X}_{t+1}^{(j)} \middle| X_t^{(i)}\right)}{\overline{p}\left(\overline{X}_{t+1}^{(j)}\right)} \delta_{X_t^{(i)}, \overline{X}_{t+1}^{(j)}}(x_t, x_{t+1}).$$

- Cost $\mathcal{O}(N^2 T)$ but $\mathcal{O}(NT)$ through importance sampling (Briers, D. & Singh, 2005; Fearnhead, Wyncoll & Tawn, 2010) and fast computational methods (Klaas et al., 2005).

Convergence Results

- **Exponentially stability assumption**. For any x_1, x_1'

$$\frac{1}{2}\int \left|p\left(x_t \middle| y_{2:t}, X_1 = x_1\right) - p\left(x_t \middle| y_{2:t}, X_1 = x_1'\right)\right| dx_t \leq \alpha^t \text{ for } |\alpha| < 1.$$

- Here $\widehat{\varphi}_T$ denotes SMC estimates obtained using direct, fixed-lag FB or TF method.
- **Marginal distribution**. If $\varphi_T\left(x_{1:T}\right) = \varphi\left(x_t\right)$, we have for the standard path-based SMC estimate

$$\lim_{N\to\infty} \sqrt{N}\left(\widehat{\varphi}_T - \overline{\varphi}_T\right) \Rightarrow \mathcal{N}\left(0, \sigma_T^2\right), \underline{A}\left(T-t+1\right) \leq \sigma_T^2 \leq \overline{A}\left(T-t+1\right)$$

whereas for FB and TF estimates there exists B independent of T s.t.

$$\lim_{N\to\infty} \sqrt{N}\left(\widehat{\varphi}_T - \overline{\varphi}_T\right) \Rightarrow \mathcal{N}\left(0, \sigma_T^2\right) \text{ where } \sigma_T^2 \leq B$$

Convergence Results for Smoothed Additive Functionals

- Consider now the case where $\varphi_T\left(x_{1:T}\right) = \sum_{t=1}^T \varphi\left(x_t\right)$, so that

$$\begin{aligned}\overline{\varphi}_T &= \int \varphi_T\left(x_{1:T}\right) p\left(x_{1:T} \middle| y_{1:T}\right) dx_{1:T} \\ &= \sum_{t=1}^T \int \varphi\left(x_t\right) p\left(x_t \middle| y_{1:T}\right) dx_t\end{aligned}$$

- This type of functionals is crucial when performing ML parameter estimation.
- We have for the standard path-based SMC estimate (Poyiadjis, D. & Singh, 2010)

$$\lim_{N\to\infty} \sqrt{N}\left(\widehat{\varphi}_T - \overline{\varphi}_T\right) \Rightarrow \mathcal{N}\left(0, \sigma_T^2\right) \text{ where } \underline{A}T^2 \leq \sigma_T^2 \leq \overline{A}T^2.$$

For the FB and TF estimates (Del Moral, D. & Singh, 2009), we have

$$\lim_{N\to\infty} \sqrt{N}\left(\widehat{\varphi}_T - \overline{\varphi}_T\right) \Rightarrow \mathcal{N}\left(0, \sigma_T^2\right) \text{ where } \sigma_T^2 \leq CT$$

Comparison Direct Method vs FB and TF

- Assume the model is stable and we are interested in approximating $\overline{\varphi}_T = \int \varphi\left(x_t\right) p\left(x_t \middle| y_{1:T}\right) dx_t$ using SMC.

Method	Fixed-lag	Direct SMC	FB/TF
# particles	N	N	N
cost	$\mathcal{O}\left(TN\right)$	$\mathcal{O}\left(TN\right)$	$\mathcal{O}\left(TN^2\right), \mathcal{O}\left(TN\right)$
Variance	$\mathcal{O}\left(1/N\right)$	$\mathcal{O}\left(\left(T-t+1\right)/N\right)$	$\mathcal{O}\left(1/N\right)$
Bias	$\circ$	$\mathcal{O}\left(1/N\right)$	$\mathcal{O}\left(1/N\right)$
MSE=Bias^2+Var	$\circ^2 + \mathcal{O}\left(1/N\right)$	$\mathcal{O}\left(\left(T-t+1\right)/N\right)$	$\mathcal{O}\left(1/N\right)$

- FB/TF provide uniformly "good" approximations of $\left\{p\left(x_t \middle| y_{1:T}\right)\right\}_{t=1}^T$ whereas direct method provide only "good" approximation for $|T-t|$ "small".
- "Fast" implementations FB and TF of computational complexity $\mathcal{O}\left(NT\right)$ outperform other approaches as MSE is $\mathcal{O}\left(1/N\right)$ whereas it is $\mathcal{O}\left(\left(T-t+1\right)/N\right)$ for direct SMC.

Comparison Direct Method vs FB and TF

- Assume we are interested in approximating $\overline{\varphi}_T = \sum_{t=1}^T \int \varphi\left(x_t\right) p\left(x_t \middle| y_{1:T}\right) dx_t$ using SMC.

Method	Fixed-lag	Direct SMC	FB/TF
# particles	N	N	N
cost	$\mathcal{O}\left(TN\right)$	$\mathcal{O}\left(TN\right)$	$\mathcal{O}\left(TN^2\right), \mathcal{O}\left(TN\right)$
Var.	$\mathcal{O}\left(T/N\right)$	$\mathcal{O}\left(T^2/N\right)$	$\mathcal{O}\left(T/N\right)$
Bias	$T\circ$	$\mathcal{O}\left(T/N\right)$	$\mathcal{O}\left(T/N\right)$
MSE=Bias^2+Var	$T^2\circ^2 + \mathcal{O}\left(T/N\right)$	$\mathcal{O}\left(T^2/N\right)$	$\mathcal{O}\left(T^2/N^2\right)$

- "Naive" implementations FB and TF have MSE of same order as direct method for fixed computational complexity but MSE is bias dominated for FB/TF whereas it is variance dominated for Direct SMC.
- "Fast" implementations FB and TF of computational complexity $\mathcal{O}\left(NT\right)$ outperform other approaches as MSE is $\mathcal{O}\left(T^2/N^2\right)$ whereas it is $\mathcal{O}\left(T^2/N\right)$ for direct SMC.

Experimental Results

- Consider a linear Gaussian model

$$X_t = 0.8X_{t-1} + 0.5V_t, \quad V_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1)$$
$$Y_t = X_t + W_t, \quad W_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1).$$

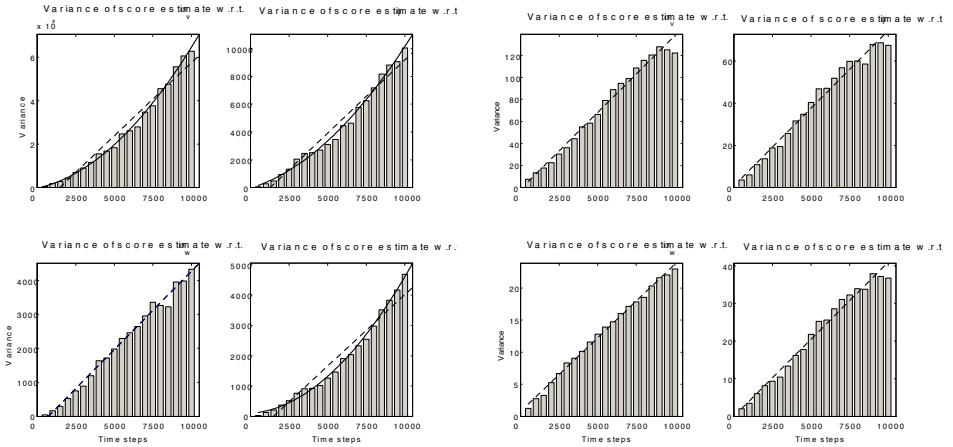
- We simulate 10,000 observations and compute SMC estimates of

$$\int \varphi_T(x_{1:T}) \; p(x_{1:T} | y_{1:T}) \, dx_{1:T}$$

for 4 different additive functionals
$\varphi_t(x_{1:t}) = \varphi_{t-1}(x_{1:t-1}) + \varphi(x_{t-1}, x_t, y_t)$ including
$\varphi^1(x_{t-1}, x_t, y_t) = x_{t-1}x_t$, $\varphi^2(x_{t-1}, x_t, y_t) = x_t^2$. [Ground truth can be computed using Kalman smoother.]

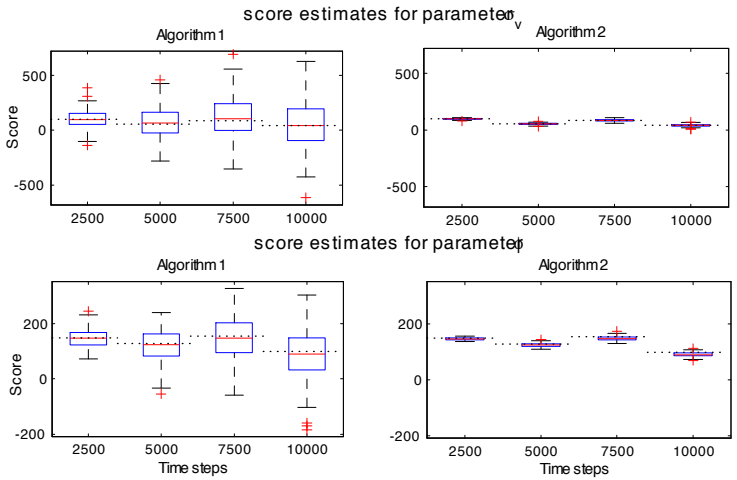
- We use SMC over 100 replications on the same dataset to estimate the empirical variance.

Empirical Variance for Direct vs FB



Direct (left) vs FB (right); the vertical scale is different

Boxplots of SMC Estimates for Direct vs FB



Direct (left) vs FB (right)

Summary

- SMC smoothing techniques allow us to "solve" the degeneracy problem.
- SMC fixed-lag smoothing is the simplest one but has non-vanishing bias difficult to quantify.
- SMC FB and SMC TF algorithms provide uniformly "good" approximations of marginal smoothing distributions contrary to direct method.
- In terms of MSE, only "fast" implementations of SMC FB/TF provide a gain in terms of MSE.
- For direct implementation SMC FB/TF, MSE is of the same order but SMC FB/TF is bias dominated and direct SMC is variance dominated.

ML Parameter Estimation in State-Space Models

- In most scenarios of interest, the state-space model contains an unknown static parameter $\theta \in \Theta$ so that

$$X_1 \sim \mu_\theta(x_1) \text{ and } X_t | (X_{t-1} = x_{t-1}) \sim f_\theta(x_t | x_{t-1}).$$

- The observations $\{Y_t\}_{t \geq 1}$ are conditionally independent given $\{X_t\}_{t \geq 1}$ and

$$Y_t | (X_t = x_t) \sim g_\theta(y_t | x_t).$$

- In many applications, we actually only care about θ and would like to estimate it off-line or on-line.

Examples

- **Stochastic Volatility model**

$$X_t = \phi X_{t-1} + \sigma V_t, \quad V_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1)$$
$$Y_t = \beta \exp(X_t/2) W_t, \quad W_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1)$$

where $\theta = (\phi, \sigma^2, \beta)$.

- **Biochemical Network model**

$$\Pr\left(X^1_{t+dt}=x^1_t+1, X^2_{t+dt}=x^2_t \middle| x^1_t, x^2_t\right) = \alpha x^1_t dt + o(dt),$$
$$\Pr\left(X^1_{t+dt}=x^1_t-1, X^2_{t+dt}=x^2_t+1 \middle| x^1_t, x^2_t\right) = \beta x^1_t x^2_t dt + o(dt),$$
$$\Pr\left(X^1_{t+dt}=x^1_t, X^2_{t+dt}=x^2_t-1 \middle| x^1_t, x^2_t\right) = \gamma x^2_t dt + o(dt),$$

with

$$Y_k = X^1_{k\Delta T} + W_k \text{ with } W_k \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2)$$

where $\theta = (\alpha, \beta, \gamma)$.

Likelihood Function Estimation

- Let $y_{1:T}$ being given, the log-(marginal) likelihood is given by

$$\ell(\theta) = \log p_\theta(y_{1:T}).$$

- For any $\theta \in \Theta$, one can estimate $\ell(\theta)$ using standard SMC. methods, variance $\mathcal{O}(T/N)$.
- Direct maximization of $\ell(\theta)$ difficult as SMC estimate $\widehat{\ell}(\theta)$ is not a smooth function of θ even for fixed random seed.
- For $\dim(X_t) = 1$, we can obtain smooth estimate of log-likelihood function by using a smoothed resampling step (e.g. Pitt, 2002-2011); i.e. piecewise linear approximation of $\Pr(X_t < x | y_{1:t})$.
- For $\dim(X_t) > 1$, we can obtain estimates of $\ell(\theta)$ highly positively correlated for neigbouring values in Θ (e.g. Lee, 2008).

Gradient Ascent

- To maximise $\ell(\theta)$ w.r.t θ, use at iteration $k+1$

$$\theta_{k+1} = \theta_k + \gamma_k \nabla \ell(\theta)|_{\theta=\theta_k}$$

where $\nabla \ell(\theta)|_{\theta=\theta_k}$ is the so-called score vector.

- $\nabla \ell(\theta)|_{\theta=\theta_k}$ can be estimated using finite differences but more efficiently using Fisher's identity (e.g. Cappé et al., 2005)

$$\nabla \ell(\theta) = \int \nabla \log p_\theta(x_{1:T}, y_{1:T}) \; p_\theta(x_{1:T} | y_{1:T}) \, dx_{1:T}$$

where

$$\nabla \log p_\theta(x_{1:T}, y_{1:T}) = \nabla \log \mu_\theta(x_1) + \sum_{t=2}^T \nabla \log f_\theta(x_t | x_{t-1}) + \sum_{t=1}^T \nabla \log g_\theta(y_t | x_t).$$

- An alternative is to use IPA (Coquelin, Deguest & Munos, 2009).

Example: SV Model

- Remember that

$$X_t = \theta X_{t-1} + \sigma V_t, \quad V_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1)$$
$$Y_t = \beta \exp(X_t/2)\, W_t, \quad W_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1)$$

  where we assume here that (σ^2, β) are known so that $\theta = \phi$.
- In this scenario

$$\log f_\theta(x_t | x_{t-1}) = -\frac{1}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}(x_t - \theta x_{t-1})^2,$$
$$\nabla \log f_\theta(x_t | x_{t-1}) = \frac{x_{t-1}(x_t - \theta x_{t-1})}{\sigma^2} = \frac{x_{t-1}x_t}{\sigma^2} - \frac{\theta x_{t-1}^2}{\sigma^2},$$

  hence

$$\nabla \ell(\theta) = \frac{\mathbb{E}_\theta\left(\sum_{t=2}^T X_{t-1}X_t \middle| y_{1:T}\right)}{\sigma^2} - \frac{\theta \mathbb{E}_\theta\left(\sum_{t=2}^T X_{t-1}^2 \middle| y_{1:T}\right)}{\sigma^2}.$$

Gradient Ascent using SMC

- An obvious SMC approximation is given by

$$\theta_{k+1} = \theta_k + \gamma_k \widehat{\nabla \ell(\theta)}\Big|_{\theta=\theta_k}$$

  where $\widehat{\nabla \ell(\theta)}\Big|_{\theta=\theta_k}$ is estimated by your favourite SMC smoothing technique.
- As $\nabla \ell(\theta)$ is a smoothed additive functional, all previously presented SMC methods and results do apply; see previous numerical results.
- Similarly, it is possible to estimate the observed information matrix $-\nabla^2 \ell(\theta)$ using SMC based on Louis identity (e.g. Cappé et al., 2005) to implement a Newton-Raphson algorithm (Poyadjis, D. & Singh, 2010).

ML Parameter Estimation using EM

- The Expectation-Maximization (EM) algorithm is a celebrated alternative to gradient ascent technique.
- To maximise $\ell(\theta)$ w.r.t θ, the EM uses

$$\theta_{k+1} = \arg\max_\theta \; Q(\theta_k, \theta).$$

  where

$$Q(\theta_k, \theta) = \int \log p_\theta(x_{1:T}, y_{1:T})\; p_{\theta_k}(x_{1:T}|y_{1:T}) dx_{1:T}$$

  and we know that

$$\ell(\theta_{k+1}) \geq \ell(\theta_k).$$

- If $p_\theta(x_{1:T}, y_{1:T})$ is in the exponential family then we have

$$\theta_{k+1} = \Lambda\left(T^{-1} \simeq_T^{\theta_k}\right)$$

  where

$$\simeq_T^\theta = \int \left(\sum_{t=2}^T \varphi(x_{t-1}, x_t, y_t)\right) p_\theta(x_{1:T}|y_{1:T}) dx_{1:T}$$

Example: SV Model

- Remember that

$$X_t = \theta X_{t-1} + \sigma V_t, \quad V_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1)$$
$$Y_t = \beta \exp(X_t/2)\, W_t, \quad W_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1)$$

  where we assume here that (σ^2, β) are known so that $\theta = \phi$.
- In this scenario

$$\log f_\theta(x_t | x_{t-1}) = -\frac{1}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}(x_t - \theta x_{t-1})^2$$
$$= -\frac{1}{2}\log(2\pi\sigma^2) - \frac{x_t^2}{2\sigma^2} - \frac{\theta^2 x_{t-1}^2}{2\sigma^2} + \frac{\theta x_{t-1}x_t}{\sigma^2}$$

  so that

$$\theta_{k+1} = \frac{\mathbb{E}_{\theta_k}\left(\sum_{t=2}^T X_{t-1}X_t \middle| y_{1:T}\right)}{\mathbb{E}_{\theta_k}\left(\sum_{t=2}^T X_{t-1}^2 \middle| y_{1:T}\right)}.$$

EM using SMC

- SMC approximation of the EM is direct.
- As EM requires computing smoothed additive functionals $\simeq_T^\theta = \int \left(\sum_{t=2}^T \varphi\left(x_{t-1}, x_t, y_t\right)\right) p_\theta(x_{1:T}|y_{1:T}) dx_{1:T}$, all previously presented SMC smoothing methods and results do apply.
- There is obviously no more guarantee that $\ell(\theta_{k+1}) \geq \ell(\theta_k)$ for finite N but many positive experimental results; e.g. (Schon, Wills & Ninness, 2011).

ML Parameter Estimation using Online Gradient

- In many applications, we would like to estimate the parameter on-line.
- *Recursive maximum likelihood* (Titterington, 1984; LeGland & Mevel, 1997) proceeds as follows

$$\theta_{t+1} = \theta_t + \gamma_t \nabla \log p_{\theta_{1:t}}\left(y_t | y_{1:t-1}\right)$$

where $p_{\theta_{1:t}}\left(y_t | y_{1:t-1}\right)$ is computed using θ_k at time k and $\sum_t \gamma_t = \infty$, $\sum_t \gamma_t^2 < \infty$. Under regularity conditions, this converges towards a local maximum of the (average) log-likelihood.

- Note that

$$\nabla \log p_{\theta_{1:t}}\left(y_t | y_{1:t-1}\right) = \nabla \log p_{\theta_{1:t}}\left(y_{1:t}\right) - \nabla \log p_{\theta_{1:t-1}}\left(y_{1:t-1}\right)$$

is given by the difference of two pseudo-score vectors where

$$\nabla \log p_{\theta_{1:t}}\left(y_{1:t}\right) := \int \left(\sum_{k=2}^t \nabla \log f_\theta\left(x_k | x_{k-1}\right)|_{\theta_k} + \nabla \log g_\theta\left(y_k | x_k\right)|_{\theta_k}\right) p_{\theta_{1:t}}\left(x_{1:t} | y_{1:t}\right) dx_{1:t}$$

ML Parameter Estimation using SMC Online Gradient

- SMC approximation follows

$$\theta_{t+1} = \theta_t + \gamma_t \widehat{\nabla \log} p_{\theta_{1:t}}\left(y_t | y_{1:t-1}\right)$$

where

$$\widehat{\nabla \log} p_{\theta_{1:t}}\left(y_t | y_{1:t-1}\right) = \widehat{\nabla \log} p_{\theta_{1:t}}\left(y_{1:t}\right) - \widehat{\nabla \log} p_{\theta_{1:t-1}}\left(y_{1:t-1}\right)$$

is given by the difference of SMC estimates of pseudo-score vectors (Poyadjis, D. & Singh, 2011).

- Asymptotic variance of $\widehat{\nabla \log} p_{\theta_{1:t}}\left(y_t | y_{1:t-1}\right)$ is uniformly bounded for FB estimate (Del Moral, D. & Singh, 2011) whereas it increases linearly with t for direct SMC method.
- **Major Problem**: If we use FB, this is not an online algorithm anymore as it requires a backward pass of order $\mathcal{O}\left(t\right)$ to approximate $\nabla \log p_{\theta_{1:t}}\left(y_{1:t}\right)$ ...

Variance of the Gradient Estimate for Direct vs FB

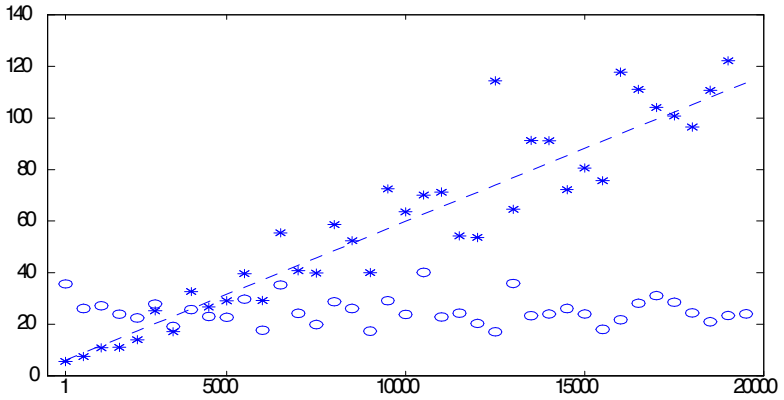


Figure: Empirical variance of the gradient estimate for standard versus FB approximations (SV model)

Online SMC ML Estimation using Direct Approximation

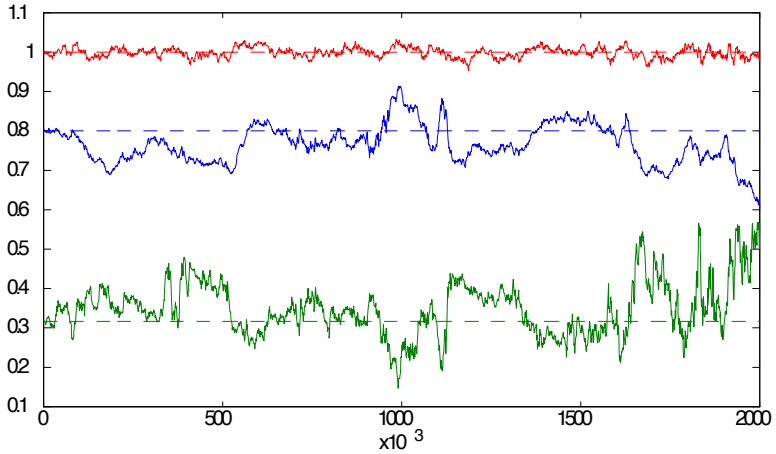


Figure: $N = 10,000$ particles, online parameter estimates for SV model.

SMC ML Estimation for SV Model using FB

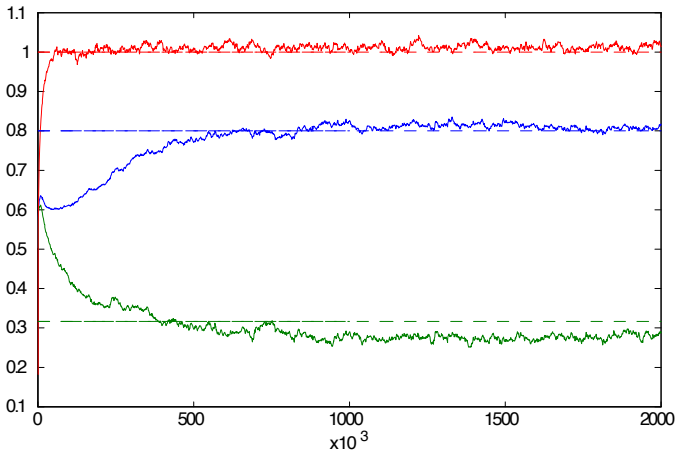


Figure: $N = 50$ particles, online parameter estimates for SV model.

Forward only Smoothing

- For the time being, we do not have an online implementation as a backward pass of length t is required at time t.
- It is possible to completely bypass the backward pass to compute using FB

$$\simeq_t^\theta = \int_t \simeq_t (x_{1:t}) \; p_\theta (x_{1:t} | y_{1:t}) \, dx_{1:t}$$

where

$$\simeq_t (x_{1:t}) = \sum_{k=1}^{t} \simeq (x_{k-1:k}, y_k)$$

using a dynamic programming trick for the "backward" Markov chain of transition densities $\{p_\theta (x_k | y_{1:k}, x_{k+1})\}$.

- Let us introduce the "value" function

$$V_t^\theta (x_t) := \int \simeq_t (x_{1:t}) \; p_\theta (x_{1:t-1} | y_{1:t-1}, x_t) \, dx_{1:t-1}$$

then

$$\simeq_t^\theta = \int V_t^\theta (x_t) \, p_\theta (x_t | y_{1:t}) \, dx_t.$$

Forward only Smoothing

- *Forward smoothing recursion*

$$V_t^\theta (x_t) = \int \left[V_{t-1}^\theta (x_{t-1}) + \simeq (x_{t-1:t}, y_t) \right] p_\theta (x_{t-1} | y_{1:t-1}, x_t) \, dx_{t-1}$$

- Proof is trivial

$$\begin{aligned} V_t^\theta (x_t) &= \int \varphi_t (x_{1:t}) \; p_\theta (x_{1:t-1} | y_{1:t-1}, x_t) \, dx_{1:t-1} \\ &= \int \left[\varphi_{t-1} (x_{1:t-1}) + \simeq (x_{t-1:t}, y_t) \right] \; p_\theta (x_{1:t-2} | y_{1:t-2}, x_{t-1}) \\ &\qquad \times p_\theta (x_{t-1} | y_{1:t-1}, x_t) \, dx_{1:t-1} \\ &= \int (\underbrace{\int \varphi_{t-1} (x_{1:t-1}) \, p_\theta (x_{1:t-2} | y_{1:t-2}, x_{t-1}) \, dx_{1:t-2}}_{V_{t-1}^\theta (x_{t-1})} \\ &\qquad + \simeq (x_{t-1:t}, y_t)) \; p_\theta (x_{t-1} | y_{1:t-1}, x_t) \, dx_{t-1} \end{aligned}$$

- Appears implicitly in Elliott, Aggoun & Moore (1996), Ford (1998) and rediscovered a few times... Presentation follows here (Del Moral, D. & Singh, 2009).

SMC Forward only Smoothing

- At time $t-1$, we have $\widehat{p}_\theta\left(x_{t-1}|\, y_{1:t-1}\right)=\frac{1}{N}\sum_{i=1}^N \delta_{X_{t-1}^{(i)}}\left(x_{t-1}\right)$ and $\left\{\widehat{V}_{t-1}^\theta\left(X_{t-1}^{(i)}\right)\right\}_{1\leq i\leq N}$.
- At time t, compute $\widehat{p}_\theta\left(x_t|\, y_{1:t}\right)=\sum_{i=1}^N W_t^{(i)}\delta_{X_t^{(i)}}\left(x_t\right)$ and set

$$\widehat{V}_t^\theta\left(X_t^{(i)}\right)=\int\left[\widehat{V}_{t-1}^\theta\left(x_{t-1}\right)+\simeq\left(x_{t-1:t},y_t\right)\right]\widehat{p}_\theta\left(x_{t-1}|\, y_{1:t-1},X_t^{(i)}\right)dx_{t-1}$$
$$=\frac{\sum_{j=1}^N f_\theta\left(X_t^{(i)}|X_{t-1}^{(j)}\right)\left[\widehat{V}_{t-1}^\theta\left(X_{t-1}^{(j)}\right)+\simeq\left(X_{t-1}^{(j)},X_t^{(i)},y_t\right)\right]}{\sum_{j=1}^N f_\theta\left(X_t^{(i)}|X_{t-1}^{(j)}\right)},$$
$$\widehat{\simeq}_t^\theta=\frac{1}{N}\sum_{i=1}^N\widehat{V}_t^\theta\left(X_t^{(i)}\right).$$

- This estimate is exactly the same as the SMC FB estimate, computational complexity $\mathcal{O}\left(N^2\right)$.

ML Parameter Estimation using SMC Online Gradient

- At time $t-1$, we have $\widehat{p}_{\theta_{1:t-1}}\left(x_{t-1}|\, y_{1:t-1}\right)$, $\left\{\widehat{V}_{t-1}^{\theta_{1:t-1}}\left(X_{t-1}^{(i)}\right)\right\}$and $\widehat{\nabla\log}\, p_{\theta_{1:t-1}}\left(y_{1:t-1}\right)=\int\widehat{V}_{t-1}^{\theta_{1:t-1}}\left(x_{t-1}\right)\widehat{p}_{\theta_{1:t-1}}\left(x_{t-1}|\, y_{1:t-1}\right)dx_{t-1}$ and obtained θ_t.
- At time t, use SMC to compute $\widehat{p}_{\theta_{1:t}}\left(x_t|\, y_{1:t}\right)$ and

$$\widehat{V}_t^{\theta_{1:t}}\left(X_t^{(i)}\right)=\int\left[\widehat{V}_{t-1}^{\theta_{1:t-1}}\left(x_{t-1}\right)+\simeq\left(x_{t-1:t},y_t\right)\right]\widehat{p}_{\theta_{1:t}}\left(x_{t-1}|\, y_{1:t-1},X_t^{(i)}\right)dx_{t-1}$$
$$\simeq\left(x_{t-1:t},y_t\right)=\nabla\log f_\theta\left(x_t|\, x_{t-1}\right)|_{\theta_t}+\nabla\log g_\theta\left(y_t|\, x_t\right)|_{\theta_t}$$

and

$$\widehat{\nabla\log}\, p_{\theta_{1:t}}\left(y_{1:t}\right)=\int\widehat{V}_t^{\theta_{1:t}}\left(x_t\right)\widehat{p}_{\theta_{1:t}}\left(x_t|\, y_{1:t}\right)dx_t$$

- Parameter update

$$\theta_{t+1}=\theta_t+\gamma_t\left(\widehat{\nabla\log}\, p_{\theta_{1:t}}\left(y_{1:t}\right)-\widehat{\nabla\log}\, p_{\theta_{1:t-1}}\left(y_{1:t-1}\right)\right)$$

Online ML Parameter Estimation through EM

- Batch EM uses

$$\simeq_T^{\theta_k}=\int\left(\sum_{t=2}^T\varphi\left(x_{t-1:t},y_t\right)\right)p_{\theta_k}(x_{1:T}|y_{1:T})dx_{1:T},$$
$$\theta_{k+1}=\Lambda\left(T^{-1}\simeq_T^{\theta_k}\right)$$

- Online EM uses

$$\simeq_{t+1}^{\theta_{1:t}}=\gamma_{t+1}\int\varphi\left(x_{t:t+1},y_{t+1}\right)p_{\theta_{1:t}}(x_t,x_{t+1}|y_{1:t+1})dx_{t:t+1}$$
$$+\left(1-\gamma_{t+1}\right)\sum_{k=1}^t\left(\prod_{l=k+2}^t\left(1-\gamma_l\right)\right)\gamma_{k+1}$$
$$\times\int\varphi\left(x_{k-1:k},y_k\right)p_{\theta_{1:t}}(x_{k-1},x_k|y_{1:t+1})dx_{k-1:k}$$

then set $\theta_{t+1}=\Lambda\left(\simeq_{t+1}^{\theta_{1:t}}\right)$ for $\{\gamma_t\}_{t\geq1}$ satisfying $\sum_t\gamma_t=\infty$ and $\sum_t\gamma_t^2<\infty$; e.g. $\gamma_t=t^{-\alpha}$ with $0.5<\alpha\leq1$.

- Under regularity conditions, this converges towards a local maximum of the (average) log-likelihood (well not yet proven for HMM)

Online ML Parameter Estimation through SMC EM

- At time $t-1$, we have $\widehat{p}_{\theta_{1:t-1}}\left(x_{t-1}|\, y_{1:t-1}\right)$, $\left\{\widehat{V}_{t-1}^{\theta_{1:t-1}}\left(X_{t-1}^{(i)}\right)\right\}$ and obtained θ_t.
- At time t, use SMC to compute $\widehat{p}_{\theta_{1:t}}\left(x_{t-1}|\, y_{1:t-1}\right)$ and

$$\widehat{V}_t^{\theta_{1:t}}\left(X_t^{(i)}\right)=\int\left[\left(1-\gamma_t\right)\widehat{V}_{t-1}^{\theta_{1:t-1}}\left(x_{t-1}\right)+\gamma_t\simeq\left(x_{t-1:t},y_t\right)\right]$$
$$\times\widehat{p}_{\theta_{1:t}}\left(x_{t-1}|\, y_{1:t-1},X_t^{(i)}\right)dx_{t-1},$$
$$\simeq_t^{\theta_{1:t}}=\int\widehat{V}_t^{\theta_{1:t}}\left(x_t\right)\widehat{p}_{\theta_{1:t}}\left(x_t|\, y_{1:t}\right)dx_t$$

- Parameter update

$$\theta_{t+1}=\Lambda\left(\simeq_t^{\theta_{1:t}}\right)$$

Application to SV Model

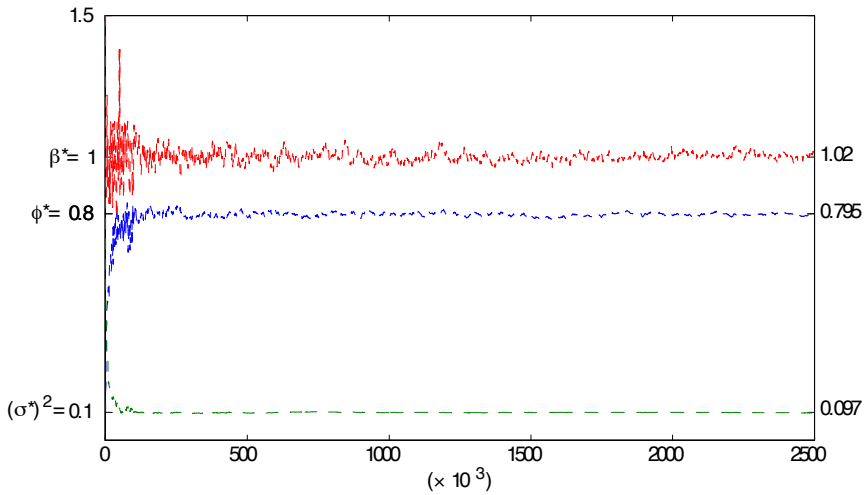


Figure: Online EM algorithm with $N = 200$ initialized at $(\phi, \sigma^2, \beta^2) = (0.1, 1, 2)$; the true values are $(\phi, \sigma^2, \beta^2) = (0.8, 0.1, 1)$.

Experimental Comparisons of Direct vs Forward Smoothing for online EM

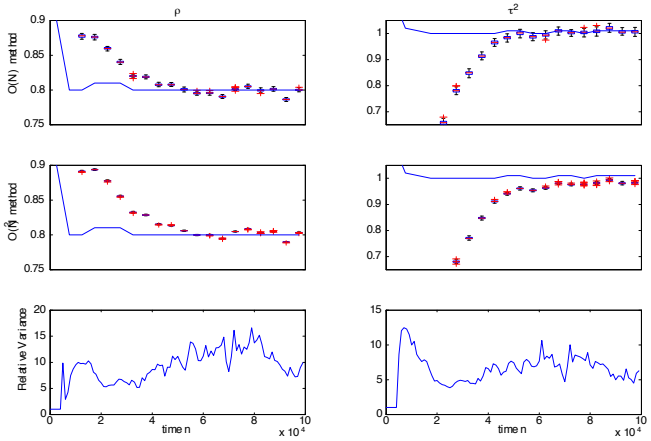


Figure: Parameter estimates for online EM obtained over 50 runs compared to ground truth: direct (left) vs forward smoothing (right).

Direct SMC vs Forward Smoothing for Online EM

- For online gradient techniques, forward smoothing is stable contrary to the direct method.
- Structure of online EM is significantly different.
- We have seen previously that the MSE for smoothed additive functionals is of the same order for direct and FB estimates.
- Direct method is variance dominated, FB is bias dominated.
- We compare experimentally both methods on a simple linear Gaussian model over 100 runs.

Summary

- SMC smoothing techniques can be used to perform off-line and on-line ML parameter estimation.
- FB estimates for smoothed additive functionals can be computed using a forward only procedure.
- Forward smoothing allows us to implement a degeneracy free on-line gradient ascent algorithm.
- For on-line EM, forward smoothing and direct methods have both pros and cons with no clear winner.
- Bias reduction approaches are currently under study.

Bayesian Parameter Inference in State-Space Models

- Assume we have

$$X_t | (X_{t-1} = x_{t-1}) \sim f_\theta (x_t | x_{t-1}),$$
$$Y_t | (X_t = x_t) \sim g_\theta (y_t | x_t),$$

  where θ is an *unknown* static parameter with prior $p(\theta)$.
- Given data $y_{1:t}$, inference relies on

$$p(\theta, x_{1:t} | y_{1:t}) = p(\theta | y_{1:t}) p_\theta (x_{1:t} | y_{1:t})$$

  where

$$p(\theta | y_{1:t}) \propto p_\theta (y_{1:t}) p(\theta).$$

- SMC methods apply as it is a standard model with extended state $Z_t = (X_t, \theta_t)$ where

$$f(z_t | z_{t-1}) = \underbrace{\delta_{\theta_{t-1}}(\theta_t)}_{\text{practical problems}} f_{\theta_t}(x_t | x_{t-1}), \; g(y_t | z_t) = g_{\theta_t}(y_t | x_t).$$

SMC with MCMC Step for Parameter Estimation

- Given at time $t-1$, the approximation

$$\widehat{p}(\theta, x_{1:t-1} | y_{1:t-1}) = \frac{1}{N} \sum_{i=1}^{N} \delta_{\left(\theta_{t-1}^{(i)}, X_{1:t-1}^{(i)}\right)} (\theta, x_{1:t-1}),$$

  we update the approximation as follows at time t.
- Sample $\widetilde{X}_t^{(i)} \sim f_{\theta_{t-1}^{(i)}} \left(\cdot | X_{t-1}^{(i)}\right)$, set $\widetilde{X}_{1:t}^{(i)} = \left(X_{1:t-1}^{(i)}, \widetilde{X}_t^{(i)}\right)$ and

$$\widetilde{p}(\theta, x_{1:t} | y_{1:t}) = \sum_{i=1}^{N} W_t^{(i)} \delta_{\left(\theta_{t-1}^{(i)}, \widetilde{X}_{1:t}^{(i)}\right)} (\theta, x_{1:t}),$$
$$W_t^{(i)} \propto g_{\theta_{t-1}^{(i)}} \left(y_t | \widetilde{X}_t^{(i)}\right).$$

- Resample $X_{1:t}^{(i)} \sim \widetilde{p}(x_{1:t} | y_{1:t})$ then sample $\theta_t^{(i)} \sim p\left(\theta | y_{1:t}, X_{1:t}^{(i)}\right)$ to obtain $\widehat{p}(\theta, x_{1:t} | y_{1:t}) = \frac{1}{N} \sum_{i=1}^{N} \delta_{\left(\theta_t^{(i)}, X_{1:t}^{(i)}\right)} (\theta, x_{1:t})$.

Cautionary Warning

- For fixed θ, $\mathbb{V}\left[\widehat{p}_\theta (y_{1:t})\right] / p_\theta^2 (y_{1:t})$ is in $\mathcal{O}(t/N)$.
- In a Bayesian context, the problem is even more complex as $p(\theta | y_{1:t}) \propto p_\theta (y_{1:t}) p(\theta)$ and we have $\theta_t = \theta$ for all t so the latent process does not enjoy mixing properties.
- A seemingly attractive idea consists of using MCMC steps on θ; e.g. (Andrieu, De Freitas & D.,1999; Fearnhead, 2002; Gilks & Berzuini 2001; Storvik, 2002; Carvalho et al., 2010) so as to introduce some "noise" on the θ component of the state.
- When $p(\theta | y_{1:t}, x_{1:t}) = p(\theta | s_t (x_{1:t}, y_{1:t}))$ where $s_t (x_{1:t}, y_{1:t})$ is a fixed-dimensional of sufficient statistics, the algorithm is particularly elegant but still implicitly relies on SMC approximation of $p(x_{1:t} | y_{1:t})$ so degeneracy will creep in.
- As $\dim(Z_t) = \dim(X_t) + \dim(\theta)$, such methods are not recommended for high-dimensional θ, especially with vague priors.

A Toy Example

- Linear Gaussian state-space model

$$X_t = \theta X_{t-1} + \sigma_V V_t, \; V_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1)$$
$$Y_t = X_t + \sigma_W W_t, \; W_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1).$$

- We set $p(\theta) \propto 1_{(-1,1)}(\theta)$ so

$$p(\theta | y_{1:t}, x_{1:t}) \propto \mathcal{N}\left(\theta; m_t, \sigma_t^2\right) 1_{(-1,1)}(\theta)$$

  where

$$\sigma_t^2 = S_{2,t}^{-1}, \; m_t = S_{2,t}^{-1} S_{1,t}$$

  with

$$S_{1,t} = \sum_{k=2}^{t} x_{k-1} x_k, \; S_{2,t} = \sum_{k=2}^{t} x_{k-1}^2$$

SMC with MCMC Step for Parameter Estimation

- At time $t-1$, $\left(\theta_{t-1}^{(i)}, X_{t-1}^{(i)}, S_{t-1}^{(i)}\right)$ we have

$$\widehat{p}\left(\theta, x_{t-1}, s_{t-1} \middle| y_{1:t-1}\right) = \frac{1}{N}\sum_{i=1}^{N} \delta_{\left(\theta_{t-1}^{(i)}, X_{t-1}^{(i)}, S_{t-1}^{(i)}\right)}\left(\theta, x_{t-1}, s_{t-1}\right).$$

- Sample $\widetilde{X}_t^{(i)} \sim f_{\theta_{t-1}^{(i)}}\left(\cdot \middle| X_{t-1}^{(i)}\right)$, set $\widetilde{S}_{1,t}^{(i)} = S_{1,t-1}^{(i)} + X_{t-1}^{(i)}\widetilde{X}_t^{(i)}$, $\widetilde{S}_{2,t}^{(i)} = S_{2,t-1}^{(i)} + \left(X_{t-1}^{(i)}\right)^2$, $W_t^{(i)} \propto g_{\theta_{t-1}^{(i)}}\left(y_t \middle| \widetilde{X}_t^{(i)}\right)$ and

$$\widetilde{p}\left(\theta, x_t, s_t \middle| y_{1:t}\right) = \sum_{i=1}^{N} W_t^{(i)} \delta_{\left(\theta_{t-1}^{(i)}, \widetilde{X}_t^{(i)}, \widetilde{S}_t^{(i)}\right)}\left(\theta, x_t, s_t\right).$$

- Resample $\left(X_t^{(i)}, S_t^{(i)}\right) \sim \widetilde{p}\left(x_t, s_t \middle| y_{1:t}\right)$ then sample $\theta_t^{(i)} \sim \mathcal{N}\left(\theta; \left(S_{2,t}^{(i)}\right)^{-1} S_{1,t}^{(i)}, \left(S_{2,t}^{(i)}\right)^{-1}\right) 1_{(-1,1)}(\theta)$ to obtain $\widehat{p}\left(\theta, x_t, s_t \middle| y_{1:t}\right) = \frac{1}{N}\sum_{i=1}^{N} \delta_{\left(\theta_t^{(i)}, X_t^{(i)}, S_t^{(i)}\right)}\left(\theta, x_t, s_t\right)$.

Illustration of the Degeneracy Problem

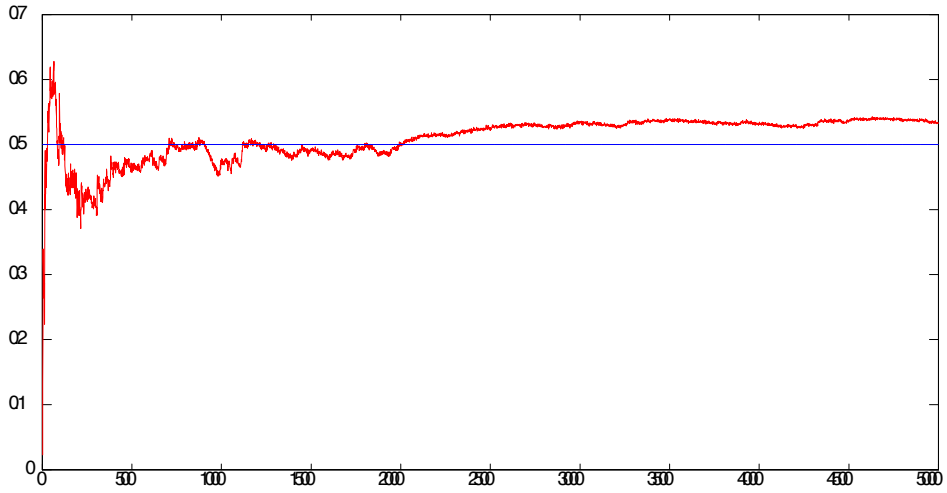


Figure:

Another Toy Example

- Linear Gaussian state-space model

$$X_t = \rho X_{t-1} + V_t, \quad V_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1)$$
$$Y_t = X_t + \sigma W_t, \quad W_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1).$$

- We set $\rho \sim \mathcal{U}_{(-1,1)}$ and $\sigma^2 \sim \mathcal{IG}(1,1)$.
- We use particle filter with perfect adaptation and Gibbs moves with $N = 10000$; particle learning (Andrieu, D. & De Freitas, 1999; Carvalho et al., 2010)
- We compare to the ground truth obtained using Kalman filter on states and grid on parameters.

Another Illustration of Degeneracy for Particle Learning

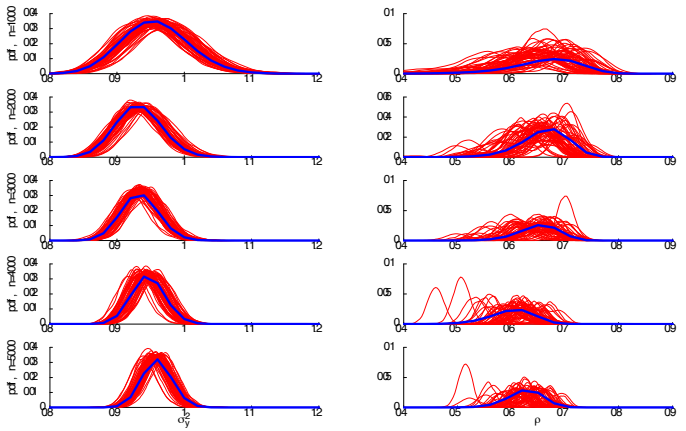


Figure: Estimates of $p\left(\rho \middle| y_{1:t}\right)$ and $p\left(\sigma^2 \middle| y_{1:t}\right)$ over 50 runs (red) vs ground truth (blue) for $t = 10^3, 2.10^3, ..., 5.10^3$ for $N = 10^4$.

Online Bayesian Parameter Estimation

- All proposed procedures for online Bayesian parameter estimation are deficient.
- Some artificial dynamics can be introduced but then we do not approximate $\{p(\theta, x_{1:t}|y_{1:t})\}_{t\geq 1}$; e.g. (Liu & West, 2001; Flury & Shephard, 2010).
- Methods based on MCMC steps are elegant but do suffer from the degeneracy problem and provide unreliable approximations.

Offline Bayesian Parameter Estimation

- Given a collection of observations $y_{1:T} := (y_1, ..., y_T)$, T being fixed, inference relies on the posterior density

$$p(\theta, x_{1:T}|y_{1:T}) = p(\theta|y_{1:T})\, p_\theta(x_{1:T}|y_{1:T}) \propto p(\theta, x_{1:T}, y_{1:T})$$

where

$$p(\theta, x_{1:T}, y_{1:T}) \propto p(\theta)\, \mu_\theta(x_1) \prod_{t=2}^{T} f_\theta(x_t|x_{t-1}) \prod_{t=1}^{T} g_\theta(y_t|x_t) \ .$$

- We show how to address this problem using particle MCMC (Andrieu, D. & Holenstein, *JRSS* B, 2010).

Common MCMC Approaches and Limitations

- **MCMC Idea**: Simulate an ergodic Markov chain $\{\theta(i), X_{1:T}(i)\}_{i\geq 0}$ of invariant distribution $p(\theta, x_{1:T}|y_{1:T})$... infinite number of possibilities.
- Typical strategies consists of updating iteratively $X_{1:T}$ conditional upon θ then θ conditional upon $X_{1:T}$.
- To update $X_{1:T}$ conditional upon θ, use MCMC kernels updating subblocks according to $p_\theta(x_{t:t+K-1}|y_{t:t+K-1}, x_{t-1}, x_{t+K})$.
- Standard MCMC algorithms are inefficient if θ and $X_{1:T}$ are strongly correlated.
- Strategy impossible to implement when it is only possible to sample from the prior but impossible to evaluate it pointwise.

Metropolis-Hastings (MH) Sampling

- To bypass these problems, we want to update jointly θ and $X_{1:T}$.
- Assume that the current state of our Markov chain is $(\theta, x_{1:T})$, we propose to update simultaneously the parameter and the states using a proposal

$$q((\theta^*, x^*_{1:T})|(\theta, x_{1:T})) = q(\theta^*|\theta)\ q_{\theta^*}(x^*_{1:T}|y_{1:T}).$$

- The proposal $(\theta^*, x^*_{1:T})$ is accepted with MH acceptance probability

$$1 \wedge \frac{p(\theta^*, x^*_{1:T}|y_{1:T})}{p(\theta, x_{1:T}|y_{1:T})} \frac{q((x_{1:T}, \theta)|(x^*_{1:T}, \theta^*))}{q((x^*_{1:T}, \theta^*)|(x_{1:T}, \theta))}$$

- **Problem**: Designing a proposal $q_{\theta^*}(x^*_{1:T}|y_{1:T})$ such that the acceptance probability is not extremely small is very difficult.

"Idealized" Marginal MH Sampler

- Consider the following so-called marginal Metropolis-Hastings (MH) algorithm which uses as a proposal

$$q\left(\left(x_{1:T}^{*}, \theta^{*}\right) \middle| \left(x_{1:T}, \theta\right)\right) = q\left(\theta^{*} \middle| \theta\right) p_{\theta^{*}}\left(x_{1:T}^{*} \middle| y_{1:T}\right).$$

- The MH acceptance probability is

$$1 \wedge \frac{p\left(\theta^{*}, x_{1:T}^{*} \middle| y_{1:T}\right)}{p\left(\theta, x_{1:T} \middle| y_{1:T}\right)} \frac{q\left(\left(x_{1:T}, \theta\right) \middle| \left(x_{1:T}^{*}, \theta^{*}\right)\right)}{q\left(\left(x_{1:T}^{*}, \theta^{*}\right) \middle| \left(x_{1:T}, \theta\right)\right)} = 1 \wedge \frac{p_{\theta^{*}}\left(y_{1:T}\right) p\left(\theta^{*}\right)}{p_{\theta}\left(y_{1:T}\right) p\left(\theta\right)} \frac{q\left(\theta \middle| \theta^{*}\right)}{q\left(\theta^{*} \middle| \theta\right)}$$

- In this MH algorithm, $X_{1:T}$ has been essentially integrated out.

Implementation Issues

- **Problem 1**: We do not know $p_{\theta}\left(y_{1:T}\right) = \int p_{\theta}\left(x_{1:T}, y_{1:T}\right) dx_{1:T}$ analytically.
- **Problem 2:** We do not know how to sample from $p_{\theta}\left(x_{1:T} \middle| y_{1:T}\right)$.
- **"Idea"**: Use SMC approximations of $p_{\theta}\left(x_{1:T} \middle| y_{1:T}\right)$ and $p_{\theta}\left(y_{1:T}\right)$.

Sequential Monte Carlo aka Particle Filters

- Given θ, SMC methods provide approximations of $p_{\theta}\left(x_{1:T} \middle| y_{1:T}\right)$ and $p_{\theta}\left(y_{1:T}\right)$.
- At time T, we obtain the following approximation of the posterior of interest

$$\widehat{p}_{\theta}\left(x_{1:T} \middle| y_{1:T}\right) = \frac{1}{N} \sum_{k=1}^{N} \delta_{X_{1:T}^{(k)}}\left(x_{1:T}\right)$$

and an approximation of $p_{\theta}\left(y_{1:T}\right)$ is given by

$$\widehat{p}_{\theta}\left(y_{1:T}\right) = \widehat{p}_{\theta}\left(y_{1}\right) \prod_{t=2}^{T} \widehat{p}_{\theta}\left(y_{t} \middle| y_{1:t-1}\right) = \prod_{t=1}^{T} \left(\frac{1}{N} \sum_{k=1}^{N} g_{\theta}\left(y_{t} \middle| X_{t}^{(k)}\right) \right)$$

if we use $f_{\theta}\left(x_{t} \middle| x_{t-1}\right)$ as a proposal.

Reminder...

- Under *mixing assumptions*, we have

$$\frac{\mathbb{V}\left[\widehat{p}_{\theta}\left(y_{1:T}\right)\right]}{p_{\theta}^{2}\left(y_{1:T}\right)} \leq D_{\theta} \frac{T}{N}.$$

- Under *mixing assumptions*, we also have

$$\int \left|\mathbb{E}\left[\widehat{p}_{\theta}\left(x_{1:T} \middle| y_{1:T}\right)\right] - p_{\theta}\left(x_{1:T} \middle| y_{1:T}\right)\right| dx_{1:T} \leq C_{\theta} \frac{T}{N}$$

so if I run an SMC method to obtain $\widehat{p}_{\theta}\left(x_{1:T} \middle| y_{1:T}\right)$ then $X_{1:T} \sim \widehat{p}_{\theta}\left(x_{1:T} \middle| y_{1:T}\right)$, unconditionally $X_{1:T} \sim \mathbb{E}\left[\widehat{p}_{\theta}\left(\cdot \middle| y_{1:T}\right)\right]$.

- **Problem**: We cannot compute analytically the particle filter proposal $q_{\theta}\left(x_{1:T} \middle| y_{1:T}\right) = \mathbb{E}\left[\widehat{p}_{\theta}\left(x_{1:T} \middle| y_{1:T}\right)\right]$ as it involves an expectation w.r.t all the variables appearing in the particle algorithm...

"Idealized" Marginal MH Sampler

At iteration i

- Sample $\theta^* \sim q\left(\theta \middle| \theta\left(i-1\right)\right)$.
- Sample $X_{1:T}^* \sim p_{\theta^*}\left(x_{1:T} \middle| y_{1:T}\right)$.
- With probability

$$1 \wedge \frac{p_{\theta^*}\left(y_{1:T}\right) p\left(\theta^*\right)}{p_{\theta(i-1)}\left(y_{1:T}\right) p\left(\theta\left(i-1\right)\right)} \frac{q\left(\theta\left(i-1\right) \middle| \theta^*\right)}{q\left(\theta^* \middle| \theta\left(i-1\right)\right)}$$

set $\theta\left(i\right) = \theta^*$, $X_{1:T}\left(i\right) = X_{1:T}^*$ otherwise set $\theta\left(i\right) = \theta\left(i-1\right)$, $X_{1:T}\left(i\right) = X_{1:T}\left(i-1\right)$.

Particle Marginal MH Sampler

At iteration i

- Sample $\theta^* \sim q\left(\theta \middle| \theta\left(i-1\right)\right)$ and run an SMC algorithm to obtain $\widehat{p}_{\theta^*}\left(x_{1:T} \middle| y_{1:T}\right)$ and $\widehat{p}_{\theta^*}\left(y_{1:T}\right)$.
- Sample $X_{1:T}^* \sim \widehat{p}_{\theta^*}\left(x_{1:T} \middle| y_{1:T}\right)$.
- With probability

$$1 \wedge \frac{\widehat{p}_{\theta^*}\left(y_{1:T}\right) p\left(\theta^*\right)}{\widehat{p}_{\theta(i-1)}\left(y_{1:T}\right) p\left(\theta\left(i-1\right)\right)} \frac{q\left(\theta\left(i-1\right) \middle| \theta^*\right)}{q\left(\theta^* \middle| \theta\left(i-1\right)\right)}$$

set $\theta\left(i\right) = \theta^*$, $X_{1:T}\left(i\right) = X_{1:T}^*$ otherwise set $\theta\left(i\right) = \theta\left(i-1\right)$, $X_{1:T}\left(i\right) = X_{1:T}\left(i-1\right)$.

Validity of the Particle Marginal MH Sampler

- **Proposition**. Assume that the 'idealized' marginal MH sampler chain is ergodic then, under very weak assumptions, the PMMH sampler chain is ergodic and admits $p(\theta, x_{1:T} \mid y_{1:T})$ **whatever being** $N \geq 1$.
- It is easy to show the simpler result that the PMMH admits $p(\theta \mid y_{1:T})$ as invariant distribution **whatever being** $N \geq 1$.
- Let U denote all the r.v. introduce to build the SMC estimate then write $\widehat{p}_\theta\left(y_{1:T}\right) = \widehat{p}_\theta\left(y_{1:T}, U\right)$ and from unbiasedness

$$\int \widehat{p}_\theta\left(y_{1:T}, u\right) q_\theta\left(u\right) du = p_\theta\left(y_{1:T}\right).$$

An Incomplete But Trivial Proof

- The PMMH targets the distribution

$$\widetilde{\pi}\left(\theta, u\right) \propto p\left(\theta\right) \widehat{p}_\theta\left(y_{1:T}, u\right) q_\theta\left(u\right)$$

which satisfies

$$\widetilde{\pi}\left(\theta\right) = p(\theta \mid y_{1:T}).$$

- The PMMH sampler uses as a proposal

$$q\left(\left(\theta^*, u^*\right) \middle| \left(\theta, u\right)\right) = q\left(\theta^* \middle| \theta\right) q_{\theta^*}\left(u^*\right)$$

and

$$\frac{\widetilde{\pi}(\theta^*,u^*)}{\widetilde{\pi}(\theta,u)} \frac{q((\theta,u)|(\theta^*,u^*))}{q((\theta^*,u^*)|(\theta,u))} = \frac{p(\theta^*)\widehat{p}_{\theta^*}(y_{1:T},u^*)q_{\theta^*}(u^*)}{p(\theta)\widehat{p}_\theta(y_{1:T},u)q_\theta(u)} \frac{q(\theta|\theta^*)q_\theta(u)}{q(\theta^*|\theta)q_{\theta^*}(u^*)} = \frac{p(\theta^*)\widehat{p}_{\theta^*}(y_{1:T},u^*)}{p(\theta)\widehat{p}_\theta(y_{1:T},u)} \frac{q(\theta|\theta^*)}{q(\theta^*|\theta)}$$

- **Trivial but deep result:** if you plug any unbiased likelihood estimate within a MCMC scheme, you do not perturb the invariant distribution.

Explicit Structure of the Target Distribution

- Let first consider the case where $T = 1$.
- *Proposal distribution*

$$\widetilde{q}\left(\left(\theta^*, k, x_1^{(1:N)}\right)\middle|\,\theta\right) = q\left(\theta^*|\,\theta\right) \underbrace{\prod_{m=1}^{N} \mu_{\theta^*}\left(x_1^{(m)}\right)\ W_1^{(k)}}_{q_{\theta^*}(u)}$$

- *Target distribution*

$$\tilde{\pi}\left(\theta, k, x_1^{(1:N)}\right) \propto\ p\left(\theta\right) \underbrace{\frac{1}{N}\sum_{m=1}^{N} g_\theta\left(y_1|\,x_1^{(m)}\right)}_{\widehat{p}_\theta(y_1)} \prod_{m=1}^{N} \mu_\theta\left(x_1^{(m)}\right)\ W_1^{(k)}$$

- We have already shown

$$\frac{\tilde{\pi}\left(\theta^*, k, x_1^{(1:N)}\right)}{\widetilde{q}^N\left(\left(\theta^*, k, x_1^{(1:N)}\right)\middle|\,\theta\right)} = \frac{p\left(\theta^*\right)}{q\left(\theta^*|\,\theta\right)} \frac{\widehat{p}_{\theta^*}\left(y_1\right)}{p_{\theta^*}\left(y_1\right)}$$

Explicit Structure of the Target Distribution

- The target is given by

$$\tilde{\pi}\left(\theta, k, x_1^{(1:N)}\right) \propto p\left(\theta\right) \left(\sum_{m=1}^{N} g_\theta\left(y_1|\,x_1^{(m)}\right)\right) \prod_{m=1}^{N} \mu_\theta\left(x_1^{(m)}\right)\ W_1^{(k)}$$

but $W_1^{(k)} = g_\theta\left(y_1|\,x_1^{(k)}\right) / \left(\sum_{m=1}^{N} g_\theta\left(y_1|\,x_1^{(m)}\right)\right)$.

- Hence, we can actually rewrite the target as

$$\tilde{\pi}^N\left(\theta, k, x_1^{(1:N)}\right) = \frac{p\left(\theta, x_1^{(k)}\middle|\,y_1\right)}{N} \prod_{m=1; m\neq k}^{N} \mu_\theta\left(x_1^{(m)}\right).$$

- This shows that we are able to sample from $p\left(\theta, x_1|\,y_1\right)$ and not only its marginal $p\left(\theta|\,y_1\right)$.

Sampling from the Target Distribution

- To sample from this target distribution
  - Sample K from a uniform distribution on $\{1, ..., N\}$.
  - Sample $\left(\theta, X_1^{(K)}\right)$ from $p\left(\theta, x_1|\,y_1\right)$. (We do not know how to do this, this is why we use MCMC).
- Sample $X_1^{(m)} \sim \mu_\theta\left(x_1\right)$ for $m \neq K$.

Explicit Structure of the Target Distribution

- This construction can be extended to the case $T > 1$.
- To sample from this target distribution
  - Sample indexes from a uniform distribution on $\{1, ..., N\}^T$ corresponding to an ancestral line.
  - Sample θ and $X_{1:T}$ for this ancestral line from $p\left(\theta, x_{1:T}|\,y_{1:T}\right)$. (We do not know how to do this, this is why we use MCMC).
- Run a conditional SMC algorithm compatible with $X_{1:T}$ and its ancestral lineage; see (Andrieu, D. & Holenstein, 2010).

Conditional SMC

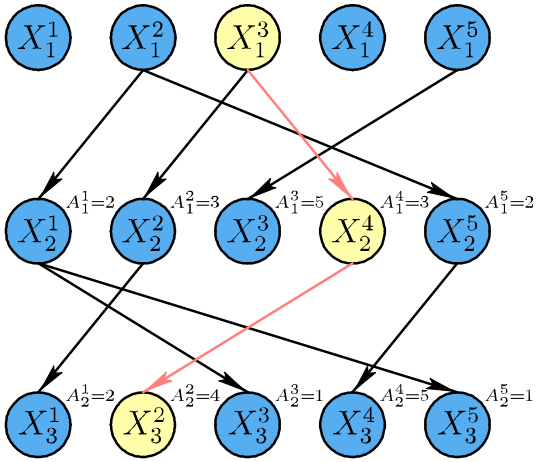


Figure: Example of $N-1=4$ ancestral lineages generated by a conditional SMC algorithm for $N=5$, $T=3$ conditional upon $X_{1:3}^2$ and $B_{1:3}^2$

Particle Gibbs Sampler

At iteration i

- Sample $\theta(i) \sim p(\theta | y_{1:T}, X_{1:T}(i-1))$.
- Run a conditional SMC algorithm for $\theta(i)$ consistent with $X_{1:T}(i-1)$ and its ancestral lineage.
- Sample $X_{1:T}(i) \sim \widehat{p}(x_{1:T} | y_{1:T}, \theta(i))$ from the resulting approximation (hence its ancestral lineage too).

- **Proposition**. Assume that the 'ideal' Gibbs sampler chain is ergodic then under very weak assumptions the particle Gibbs sampler chain is ergodic and admits $p(\theta, x_{1:T} | y_{1:T})$ as an invariant distribution **for any** $N \geq 2$.

"Idealized" Gibbs Sampler

- To sample from $p(\theta, x_{1:T} | y_{1:T})$, an MCMC strategy consists of using the following block Gibbs sampler.

At iteration i

- Sample $X_{1:T}(i) \sim p_{\theta(i-1)}(x_{1:T} | y_{1:T})$.
- Sample $\theta(i) \sim p(\theta | y_{1:T}, X_{1:T}(i))$.

- **Problem**: We do not know how to sample from $p_\theta(x_{1:T} | y_{1:T})$.
- Naive particle approximation where $X_{1:T}(i) \sim \widehat{p}(x_{1:T} | y_{1:T}, \theta(i))$ is substituted to $X_{1:T}(i) \sim p(x_{1:T} | y_{1:T}, \theta(i))$ is obviously incorrect.

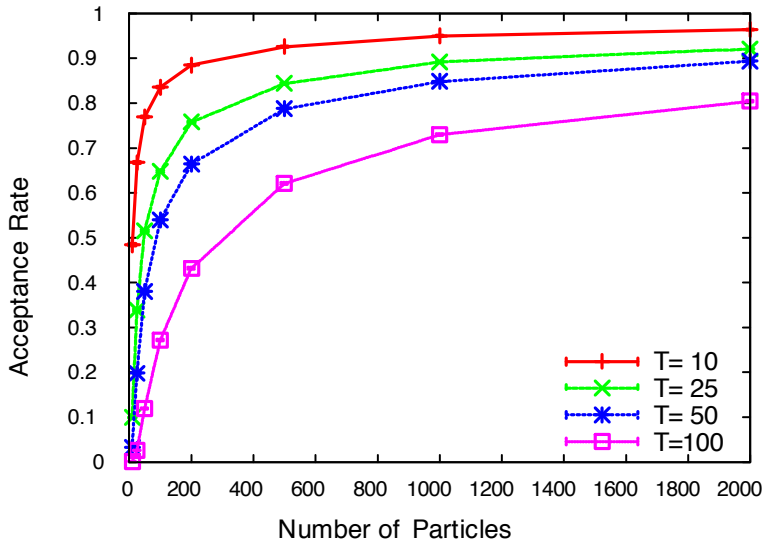
Nonlinear State-Space Model

- Consider the following model

$$X_t = \frac{1}{2}X_{t-1} + 25\frac{X_{t-1}}{1+X_{t-1}^2} + 8\cos 1.2t + V_t,$$
$$Y_t = \frac{X_t^2}{20} + W_t$$

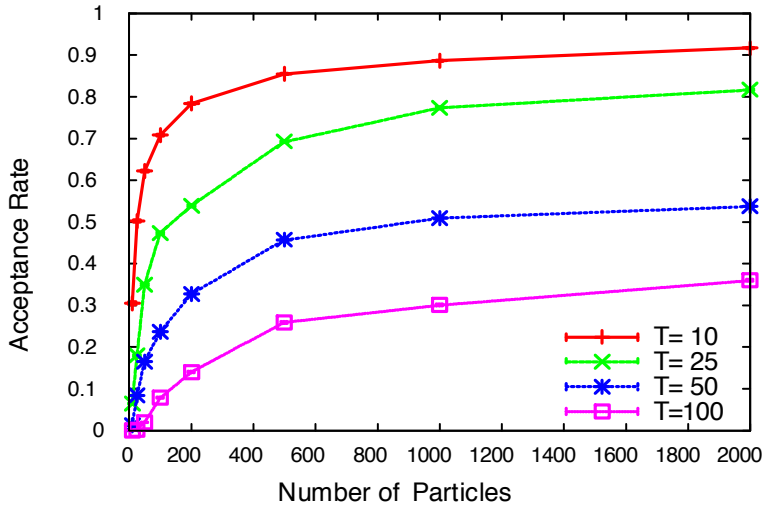
  where $V_t \sim \mathcal{N}(0, \sigma_v^2)$, $W_t \sim \mathcal{N}(0, \sigma_w^2)$ and $X_1 \sim \mathcal{N}(0, 5^2)$.
- Use the prior for $\{X_t\}$ as proposal distribution.
- For a fixed θ, we evaluate the expected acceptance probability as a function of N.

Average Acceptance Probability



Average acceptance probability when $\sigma_v^2 = \sigma_w^2 = 10$

Average Acceptance Probability



Average acceptance probability when $\sigma_v^2 = 10$, $\sigma_w^2 = 1$

Inference for Stochastic Kinetic Models

- Two species X_t^1 (prey) and X_t^2 (predator)

$$\begin{aligned}
&\Pr\left(X_{t+dt}^1 = x_t^1 + 1, X_{t+dt}^2 = x_t^2 \middle| x_t^1, x_t^2\right) = \alpha\, x_t^1\, dt + o\left(dt\right), \\
&\Pr\left(X_{t+dt}^1 = x_t^1 - 1, X_{t+dt}^2 = x_t^2 + 1 \middle| x_t^1, x_t^2\right) = \beta\, x_t^1\, x_t^2\, dt + o\left(dt\right), \\
&\Pr\left(X_{t+dt}^1 = x_t^1, X_{t+dt}^2 = x_t^2 - 1 \middle| x_t^1, x_t^2\right) = \gamma\, x_t^2\, dt + o\left(dt\right),
\end{aligned}$$

with

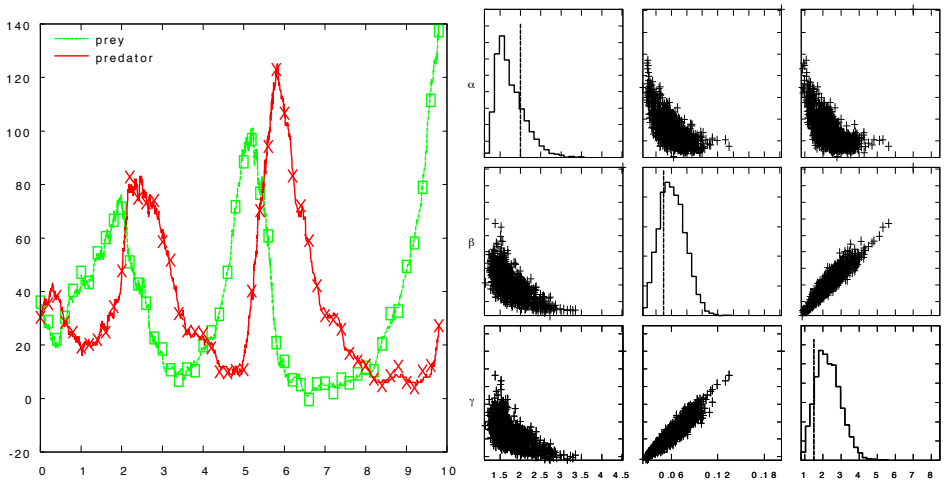
$$Y_k = X_{k\Delta T}^1 + W_k \text{ with } W_k \overset{\text{i.i.d.}}{\sim} \mathcal{N}\left(0, \sigma^2\right).$$

- We are interested in the kinetic rate constants $\theta = (\alpha, \beta, \gamma)$ a priori distributed as (Boys et al., 2008; Kunsch, 2011)

$$\alpha \sim \mathcal{G}(1, 10), \quad \beta \sim \mathcal{G}(1, 0.25), \quad \gamma \sim \mathcal{G}(1, 7.5).$$

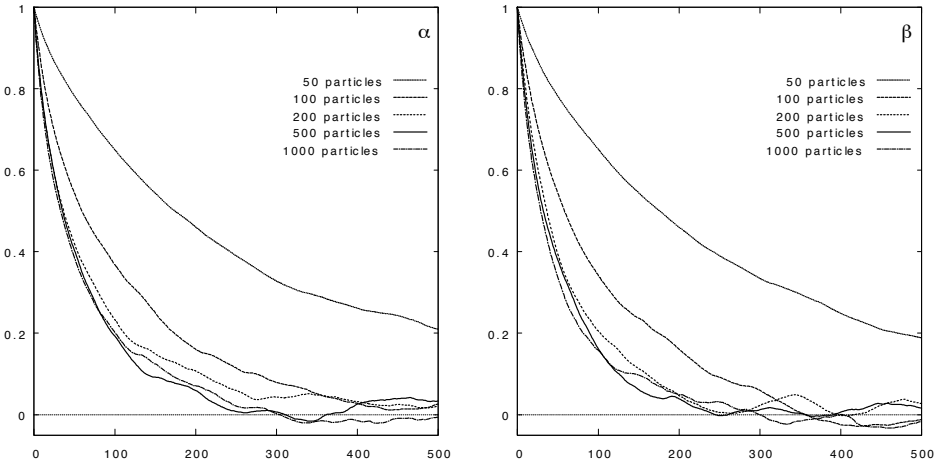
- MCMC methods require reversible jumps, Particle MCMC requires only forward simulation.

Experimental Results



Simulated data Posterior distributions

Autocorrelation Functions



Autocorrelation of α (left) and β (right) for the PMMH sampler for various N.

Summary

- Offline Bayesian parameter inference is feasible by using SMC proposals within MCMC.
- This approach does not suffer from degeneracy problem and N scales roughly linearly with T.
- Particle MCMC allow us to perform Bayesian inference for dynamic models for which only forward simulation is possible.
- Computationally intensive but several implementations on GPU already available and applications in control, ecology, econometrics, biochemical systems, epidemiology, water resources research etc.
- Selection of N is a key issue and some guidelines are available (D., Pitt, Deligiannidis & Kohn, 2014).