Introduction to Particle filtering

N. Kantas

Introduction

- Hidden Markov models are used in many disciplines
  - statistics, econometrics, engineering, neuroscience, medical & life sciences...
- Sequential Bayesian inference is natural for these models
  - known as (non-linear or stochastic) filtering
- Particle filtering (aka Sequential Monte Carlo)
  - in its most basic form is Sequential Importance Sampling & Resampling

Outline

- In other sessions
  - Hidden Markov Models and the filtering problem in discrete time
  - Kalman filtering and other deterministic methods
- Particle filtering
  - Sequential Importance Sampling for filtering
  - The need for resampling
  - Sequential Importance with Resampling

Introduction to Monte Carlo filtering

- Common problems with previous methods based on deterministic approximations:
  - Hard to quantify precision and performance
  - difficult to tune
  - very hard to be useful in higher dimensions than $2 - 3$
  - very often do not work, because underlying approximations are not valid.
- In the heart of filtering lies the problem of numerical integration
  - A different direction is to use simulation
  - take advantage of more computational power available.

HMMs and filtering on the path space

- Filtering recursions for the path space:
  - $\Pi_n(\cdot) = \mathbb{P}[X_{0:n} \in \cdot | Y_{0:n}]$
  - $\Pi_{n|n-1}(\cdot) = \mathbb{P}[X_{0:n} \in \cdot | Y_{0:n-1}]$.
- Reminder: posterior density:

$$p_\theta(x_{0:n} | y_{0:n}) = \frac{p_\theta(x_{0:n}, y_{0:n})}{p_\theta(y_{0:n})} \tag{1}$$

where

$$p_\theta(x_{0:n}, y_{0:n}) = \eta_\theta(x_0) \prod_{k=1}^{n} f_\theta(x_k | x_{k-1}) \prod_{k=0}^{n} g_\theta(y_k | x_k) \tag{2}$$

and the *marginal likelihood*, $p_\theta(y_{0:n})$, is given by

$$p_\theta(y_{0:n}) = \int p_\theta(x_{0:n}, y_{0:n})\, dx_{0:n}. \tag{3}$$

Recursions for joint filter and marginal likelihood

- We are also interested in the normalising constant.
  - marginal likelihood $p_\theta(y_{0:n})$
  - Here it also appears as product of normalising constants:

$$p_\theta(y_{0:n}) = \prod_{k=0}^{n} p_\theta(y_k | y_{0:k-1})$$

- Note here

$$p_\theta(x_{0:n} | y_{0:n}) = \frac{p_\theta(x_{0:n-1} | y_{0:n-1}) f_\theta(x_n | x_{n-1}) g_\theta(y_n | x_n)}{p_\theta(y_n | y_{0:n-1})}$$

SIS for filtering

- The density of interest at time n

$$\gamma_n(x_{0:n}) = \prod_{k=0}^{n} f_\theta\left(x_k \middle| x_{k-1}\right) g_\theta\left(y_k \middle| x_k\right)$$

  where for convenience we will write $f(x_0|x_{-1})$ to be $\eta(x_0)$
- Choose importance densities: $q_\theta\left(x_0 \middle| y_0\right)$ and $q_\theta\left(x_n \middle| y_n, x_{n-1}\right)$
- Compute importance weight as

$$w_n(x_{0:n}) = \frac{\gamma_n(x_{0:n})}{q_n(x_{0:n})} = \prod_{i=0}^{n} \omega_i(x_{i-1}, x_i) = \prod_{i=0}^{n} \frac{f_\theta\left(x_i \middle| x_{i-1}\right) g_\theta\left(y_i \middle| x_i\right)}{q_\theta\left(x_i \middle| y_i, x_{i-1}\right)}$$

SIS for filtering

- Define the incremental importance weights

$$\omega_0(x_0) = \frac{\eta_\theta(x_0)\, g_\theta(y_0 | x_0)}{q_\theta(x_0 | y_0)}, \tag{4}$$

$$\omega_n(x_{n-1:n}) = \frac{\gamma_n(x_n | x_{0:n-1})}{q_n(x_n | x_{0:n-1})} = \frac{f_\theta(x_n | x_{n-1})\, g_\theta(y_n | x_n)}{q_\theta(x_n | y_n, x_{n-1})} \text{ for } n \geq 1. \tag{5}$$

SIS filter

At time $n = 0$, For $i = 1, \ldots, N$

- Sample $X_0^i \sim q_\theta(x_0 | y_0)$.
- Compute the weights $W_0^i \propto \omega_0 \left(X_0^i\right)$, $\sum_{i=1}^N W_0^i = 1$.

At time $n \geq 1$, For $i = 1, \ldots, N$

- Sample $X_n^i \sim q_\theta(x_n | y_n, X_{n-1}^i)$ and set $X_{0:n}^i = \left(X_{0:n-1}^i, X_n^i\right)$.
- Compute the weights $W_n^i \propto W_{n-1}^i \omega_n \left(X_{n-1:n}^i\right)$, $\sum_{i=1}^N W_n^i = 1$.

SIS for filtering

At time n, the approximations of $p_\theta\left(\left. x_{0:n}\right| y_{0:n}\right)$ and $p_\theta\left(\left. y_n\right| y_{0:n-1}\right)$ after the sampling step are

$$\widehat{p}_\theta\left(\left. dx_{0:n}\right| y_{0:n}\right) = \sum_{i=1}^{N} W_n^i \delta_{X_{\mathbf{0}:n}^i}\left(dx_{0:n}\right),$$

$$\widehat{p}_\theta\left(y_{0:n}\right) = \frac{1}{N}\sum_{i=1}^{N} w_n\left(X_{0:n}^i\right) \text{ or } \prod_{p=0}^{n}\left(\frac{1}{N}\sum_{i=1}^{N}\omega_p\left(X_{p-1:p}^i\right)\right).$$

SIS for filtering: discussion

- Approach is quite old
  - J. E. Handschin and D. Q. Mayne (1966) Monte Carlo Techniques to Estimate the Conditional Expectation in Multistage Nonlinear Filtering, Int. J of Control
- Potential problems:
  - low weights will remain low for each particle
  - weight variance eventually explodes
  - mass concentrates to few particles
- Things will be better if we design q_n well. How to construct q_n?

Particle approximations with SIS

- Let also $\varphi : \mathcal{X}^{n+1} \to \mathbb{R}^d$ be a bounded measureable test function
- the integral of interest be

$$I_n = \int \varphi(x_{0:n}) p_\theta \left(x_{0:n} \middle| \, y_{0:n} \right) dx_{0:n}$$

- and its particle approximation

$$\begin{aligned} \hat{I}_n &= \int \varphi(x_{0:n}) \hat{p}_\theta \left(dx_{0:n} \middle| \, y_{0:n} \right) \\ &= \sum_{i=1}^{N} W_n^i \varphi \left(X_{0:n}^i \right) \end{aligned}$$

Choosing importance proposals

- Theoretical properties are as in SIS
  - bias and variance of $\widehat{\Pi}_n(\varphi)$ are $O(\frac{1}{N})$
- We are typically interested in the expectations of several test functions φ
- Interested to
  - minimise the rel. variance of the normalising constant $\hat{Z}_n$
  - or equivalently minimise the variance of the importance weights.
- This means $q_n(x_{0:n})$ should be **very similar or close** to $p_\theta\left(x_{0:n} \middle| y_{0:n}\right)$

"Bootstrap" proposal

- So how to choose importance densities: $q_\theta\left(x_0 \mid y_0\right)$ and $q_\theta\left(x_n \mid y_n, x_{n-1}\right)$?
- One simple and popular option is to use

$$q_\theta\left(x_n \mid y_n, x_{n-1}\right) = f_\theta\left(x_n \mid x_{n-1}\right)$$

  and hope that $w_n\left(x_{n-1:n}\right) = g(y_n|x_n)$ will not have very high variance.
  - "bootstrap proposal"
- This option will perform well when $p_\theta\left(x_{0:n} \mid y_{0:n}\right)$ does not change very fast with n
  - y_n is not very informative
  - dynamics of x_n not change very fast

Optimal proposal

- Previous option often is not very effective:
  - does not use any information from y_n
- It turns out that the **optimal** proposal which minimises the variance of the weights is:

  $$q_\theta^{opt}\left(x_n \mid y_n, x_{n-1}\right) = p_\theta\left(x_n \mid y_n, x_{n-1}\right)$$

  and this leads to weight ratio

  $$\omega_n^{opt}(x_{n-1}, x_n) = p_\theta(y_n | x_{n-1})$$

  but both $\omega_n^{opt}, q_n^{opt}$ are not possible to compute analytically.
- Instead use ideas from approximations of $p_\theta\left(x_n \mid y_{0:n}\right)$ from other methods: EKF, UKF, Laplace approximations.

Optimal proposal for linear state space models

- Recall model

$$X_{n+1} = AX_n + BV_{n+1}, \quad Y_n = CX_n + D\mathcal{W}_n$$

with $V_n, \mathcal{W}_n$ both zero mean i.i.d with identity variance.

- $p_\theta\left(x_n \mid y_n, x_{n-1}\right)$ is Gaussian $\mathcal{N}(\mathfrak{m}_n(x_{n-1}), \mathcal{S})$
  - Covariance is

$$\mathcal{S}^{-1} = \left(BB^T\right)^{-1} + C^T\left(DD^T\right)^{-1}C$$

  - mean

$$\mathfrak{m}_n(x_{n-1}) = \mathcal{S}\left(\left(BB^T\right)^{-1}AX_{n-1} + C^T\left(DD^T\right)^{-1}y_n\right)$$

Monitoring the weights: the effective sampling size

- Once q is set, one may run the SIS filter.
- To monitor performance use the effective sample size

$$ESS_n = \frac{1}{\left(\sum_{i=1}^{N} (W_n^i)^2\right)}$$

  - $\max ESS_n = N$ when $W_n^i = N^{-1}$ (weights with zero variance)
  - If there exists i such that $W_n^i \approx 1$, and for $j \neq i$, $W_n^j \approx 0$, then $ESS_n \approx 1$.
  - the higher the ESS the better our approximation
    - unless all samples are very close to each other or stuck at a tail (locally flat density case)

Numerical Example

- Scalar linear Gaussian model

$$X_n = \rho X_{n-1} + \sigma_v V_n, \quad Y_n = cX_n + \sigma_w W_n,$$

  where $W_n, V_n \overset{\text{iid}}{\sim} \mathcal{N}(0,1)$, $X_0 \sim \mathcal{N}(0,1)$.
- $T = 100$, $\rho = 0.6, \tau = 1, \sigma = \sqrt{2}, c = 1$
- $N = 500$
- Check script SIStest.m

The need for resampling

- **Weight degeneracy** will appear for high n
  - low weights will remain low for each particle
  - mass concentrates to few or one particle
  - weight variance eventually explodes
- **What does resampling do?** At time n,
  - select particles with high weights, and remove particles with low weights.
  - spend the fixed computational budget on the most promising paths.
  - using stronger particles to generate new ones in the future can stabilise weights

Resampling

- This procedure comes under the name multinomial resampling
  - at time n let $o_n(i)$ denote number of offsprings of particle i.

- Sample

$$(o_n(1), \ldots, o_n(N)) \sim \mathcal{M}ultinomial(N; W_n^1, \ldots, W_n^N)$$

- Set $k = 0$;
- For $i = 1 : N$
  - For $j = 0 : o_n(i)$,
    - $\bar{X}^k_{0:n} = X^i_{0:n}$;
    - $k \leftarrow k + 1$
  - End For
- End For

Resampling

- All randomness added is due to sampling $(o_n(1), \ldots, o_n(N))$
- Particle approximation:

$$\begin{aligned} \overline{p}_\theta\left(dx_{0:n} \middle| y_{0:n}\right) &= \sum_{i=1}^{N} \frac{o_n(i)}{N} \delta_{X^i_{0:n}}\left(dx_{0:n}\right). \\ &= \frac{1}{N} \sum_{i=1}^{N} \delta_{\overline{X}^i_{0:n}}\left(dx_{0:n}\right) \end{aligned}$$

- Multinomial sampling unbiased as:

$$\mathbb{E}\left(o_n(i)\right) = NW_n^i, \quad \sum_{j=1}^{N} o_n(j) = N$$

Resampling

- Procedure of subsampling of $\widehat{p}_\theta\left(dx_{0:n} \mid y_{0:n}\right)$ to get $\bar{p}_\theta\left(dx_{0:n} \mid y_{0:n}\right)$ adds some noise to the relevant approximations.
- Desired property to preserve unbiasedness and accuracy in estimates:
$$\mathbb{E}\left(o_n(i)\right) = NW_n^i$$
- Resampling methods: multinomial resampling:
  - sampling with replacement $\mathbb{P}\left[a_n(i) = j\right] = W_n^j$, where $a_n(i)$ is index of ancestor of selected particle i after resampling.
    - cost is proportional to $N \log N$

Resampling

- Different methods: Systematic resampling

- Sample

$$U_1 \sim Uniform[0, \frac{1}{N}),$$

$$o_n(1) = \left\{k : \sum_{l=1}^{k-1} W_n^l \leq U_1 \leq \sum_{l=1}^{k} W_n^l\right\}$$

- For $k = 2 : N$,

$$U_k = U_1 + \frac{k-1}{N},$$

$$o_n(k) = \left\{j : \sum_{l=1}^{j-1} W_n^l \leq U_k \leq \sum_{l=1}^{j} W_n^l\right\}$$

End for

- cost is proportional to N

Resampling

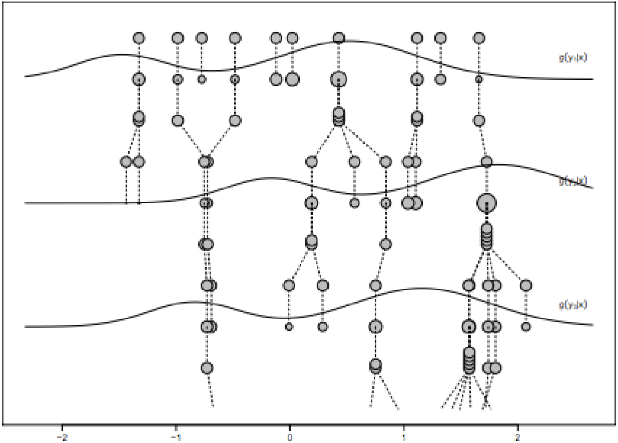
- Different methods: Residual resampling
  - improve on multinomial resampling to ensure that $o_n(i) \geq \lfloor NW_n^i \rfloor$, with $c = \lfloor x \rfloor$ being the highest integer such that $c \leq x$. $(floor())$
  - Procedure
    - Let $\tilde{N}_n^i = \lfloor NW_n^i \rfloor$ and $\tilde{N}_n = \sum_{i=1}^{N} \lfloor NW_n^i \rfloor$
    - Set $W_n^i \propto W^i - \frac{\tilde{N}_n^i}{N}$ and sample

$$(\tilde{o}_n(1), \ldots, \tilde{o}_n(N)) \sim \mathcal{M}ultinomial(N - \tilde{N}_n; \tilde{W}_n^1, \ldots, \tilde{W}_n^N)$$

    - For $k = 1 : N$, compute $o_n(k) = \tilde{o}_n(k) + \tilde{N}_n^k$

Illustration of method

N=10 particles



Sequential Importance Sampling and Resampling (SIR)

At time $n = 0$, For all $i \in \{1, ..., N\}$:

- Sample $X_0^i \sim q_\theta(x_0 | y_0)$.
- Compute the weights $w_0\left(X_0^i\right)$ and set $W_0^i \propto w_0\left(X_0^i\right)$, $\sum_{i=1}^N W_0^i = 1$.
- Resample $\left\{W_0^i, X_0^i\right\}$ to obtain N equally-weighted particles $\left\{\frac{1}{N}, \overline{X}_0^i\right\}$.

At time $n \geq 1$, For all $i \in \{1, ..., N\}$:

- Sample $X_n^i \sim q_\theta(x_n | y_n, \overline{X}_{n-1}^i)$ and set $X_{0:n}^i \leftarrow \left(\overline{X}_{0:n-1}^i, X_n^i\right)$.
- Compute the weights $\omega_n\left(X_{n-1:n}^i\right)$ and set $W_n^i \propto \omega_n\left(X_{n-1:n}^i\right)$, $\sum_{i=1}^N W_n^i = 1$.
- Resample $\left\{W_n^i, X_{0:n}^i\right\}$ to obtain N new equally-weighted particles $\left\{\frac{1}{N}, \overline{X}_{0:n}^i\right\}$.

Sequential Importance Sampling and Resampling (SIR)

- Incremental weights are as in Sequential Importance Sampling (SIS)
- The importance weights are

$$\omega_n(x_0) = w_n(x_0) = \frac{\eta_\theta(x_0)\, g_\theta(y_0 | x_0)}{q_\theta(x_0 | y_0)}, \tag{6}$$

$$\omega_n(x_{n-1:n}) = \frac{\gamma_n(x_n | x_{0:n-1})}{q_n(x_n | x_{0:n-1})} = \frac{f_\theta(x_n | x_{n-1})\, g_\theta(y_n | x_n)}{q_\theta(x_n | y_n, x_{n-1})} \text{ for } n \geq 1. \tag{7}$$

Particle approximations

At time n, the approximations of $p_\theta\left(x_{0:n} \middle| y_{0:n} \right)$ and $p_\theta\left(y_n \middle| y_{0:n-1} \right)$ after the sampling step are

$$\widehat{p}_\theta\left(dx_{0:n} \middle| y_{0:n} \right) = \sum_{i=1}^{N} W_n^i \delta_{X_{\mathbf{0}:n}^i}\left(dx_{0:n} \right), \tag{8}$$

$$\widehat{p}_\theta\left(y_n \middle| y_{0:n-1} \right) = \frac{1}{N} \sum_{i=1}^{N} \omega_n\left(X_{n-1:n}^i \right). \tag{9}$$

Hence an estimate of the marginal likelihood is given by

$$\widehat{p}_\theta\left(y_{0:n} \right) = \widehat{p}_\theta\left(y_0 \right) \prod_{k=1}^{n} \widehat{p}_\theta\left(y_k \middle| y_{0:k-1} \right). \tag{10}$$

After the resampling step, an alternative approximation of $p_\theta\left(x_{0:n} \middle| y_{0:n} \right)$ is

$$\overline{p}_\theta\left(dx_{0:n} \middle| y_{0:n} \right) = \frac{1}{N} \sum_{i=1}^{N} \delta_{\overline{X}_{\mathbf{0}:n}^i}\left(dx_{0:n} \right). \tag{11}$$

Numerical Example

- Scalar linear Gaussian model

$$X_n = \rho X_{n-1} + \sigma_v V_n, \quad Y_n = cX_n + \sigma_w W_n,$$

  where $W_n, V_n \overset{\text{iid}}{\sim} \mathcal{N}(0,1)$, $X_0 \sim \mathcal{N}(0,1)$.
- $T = 100$, $\rho = 0.6, \tau = 1, \sigma = \sqrt{2}, c = 1$
- $N = 500$
- Check scripts PFtest.m, demo.m

Discussion

- Improvement in terms of performance are clear when resampling is used.
  - Particle filter perform very well when integral of interest is

$$I_n = \int \varphi(x_n) p_\theta \left(x_{0:n} \middle| y_{0:n} \right) dx_{0:n}$$

- Figure with final particles indicates that if

$$I_n = \int \varphi(x_{0:m}) p_\theta \left(x_{0:n} \middle| y_{0:n} \right) dx_{0:n}$$

  with $m < n$ then particle approximations lack diversity
  - number of unique particles decreases as we go backwards in time.

Discussion

- $p_\theta\left(x_{0:m} \middle| y_{0:n} \right)$ will eventually be approximated by a single unique particle as $n - m$ increases.
  - This is known in the literature as the **degeneracy** problem due to resampling.
  - Fundamental weakness of SMC: given a fixed number of particles N, it is impossible to approximate full $p_\theta\left(x_{0:n} \middle| y_{0:n}\right)$ "well" when n is large (Typically as soon as $n \approx N$).

Convergence results: L_p bounds

- Let $\epsilon_{\theta,n}(dx_{0:n}) = \widehat{p}_\theta(dx_{0:n}|\, y_{0:n}) - p_\theta(dx_{0:n}|\, y_{0:n})$. If $\omega_0(x_0)$ and $\omega_n(x_{n-1:n})$ are upper bounded, for any bounded test function $\varphi_n : \mathcal{X}^{n+1} \to \mathbb{R}$, there exists constants $C_{\theta,n,p} < \infty$ such that for any $p > 0$,

$$\mathbb{E}_N\left[\left|\int \varphi_n(x_{0:n})\epsilon_{\theta,n}(dx_{0:n})\right|^p\right]^{\frac{1}{p}} \leq \frac{C_{\theta,n,p}\overline{\varphi_n}}{N^{1/2}}, \qquad (12)$$

  where $\overline{\varphi_n} = \sup_{x_{0:n}\in\mathcal{X}^{n+1}} |\varphi_n(x_{0:n})|$
- Weak result:
  - as typically $C_{\theta,n,p}$ grows exponentially/ polynomially with n.
  - Not surprising: the dimension of the target density $p_\theta(x_{0:n}|\, y_{0:n})$ we are approximating is increasing with n.

Convergence results: uniform L_p bounds for HMMs

- Many state-space models possess the so-called *exponential forgetting* property. For any $x_0, x_0' \in \mathcal{X}$ and observation record $y_{0:n}$,

$$\int \left| p_\theta\left(x_n \middle| y_{0:n}, x_0\right) - p_\theta\left(x_n \middle| y_{0:n}, x_0'\right) \right| dx_n \leq C\lambda^n, \qquad (13)$$

  where $\lambda \in [0,1)$ and C is a constant.
- Then: for an integer $L > 0$ and any bounded test function $\varphi_L : \mathcal{X}^L \to \mathbb{R}$, there exists constants $D_{\theta,L,p} < \infty$ such that for any $p > 0$

$$\mathbb{E}_N\left[\left|\int \varphi_L(x_{n-L+1:n})\epsilon_{\theta,L}\left(dx_{n-L+1:n}\right)\right|^p\right]^{\frac{1}{p}} \leq \frac{D_{\theta,L,p}\overline{\varphi_L}}{N^{1/2}}, \qquad (14)$$

  where $\epsilon_{\theta,L}\left(dx_{n-L+1:n}\right) = \int_{\mathcal{X}^{n-L+1}} \epsilon_{\theta,n}\left(dx_{0:n}\right)$

On the Monte Carlo variance

- CLT also applies for SMC and asymptotic variance much lower than SIS
- SMC Algorithm results in **unbiased estimation of the marginal likelihood**

$$\mathbb{E}_N[\hat{p}_{\theta'}(y_{0:T})] = p_{\theta'}(y_{0:T})$$

  - Non-trivial result (due to Del Moral 1995)
- $\widehat{p}_\theta(y_{0:n})$ has a relative (non-asymptotic) variance that increases linearly with n
  - relative variance is variance of $\frac{\widehat{p}_\theta(y_{0:n})}{p_\theta(y_{0:n})}$
  - Cerou, Del Moral & Guyader 2011

Smoothed additive functionals

- On the contrary, even if (13) holds, then the asymptotic variance of the SMC estimate of the additive functional

$$I_n = \int \left[\sum_{k=0}^{n} \varphi\left(x_k\right) \right] p_\theta\left(x_{0:n} \middle| y_{0:n} \right) dx_{0:n}, \tag{15}$$

which is

$$\widehat{I}_n = \int \left[\sum_{k=0}^{n} \varphi\left(x_k\right) \right] \widehat{p}_\theta\left(dx_{0:n} \middle| y_{0:n} \right), \tag{16}$$

satisfies (Poyiadjis et al 2009)

$$\mathbb{V}ar\left(\widehat{I}_n\right) \geq D_\theta \frac{n^2}{N}. \tag{17}$$

- This motivates the use of dedicated smoothing algorithms (especially for parameter estimation).

Advanced particle filters: a large family

- Large ecosystem of PFs (e.g. [Doucet et. al. 01])
  - auxiliary PF [Pitt & Sheppard 99]
    - score weight & use approximation of optimal proposal
  - resample move PF [Gillks & Berzuini 99]
    - use MCMC steps after resampling
  - adaptive PFs
  - regularised PFs [Le Gland, Outjane & Musso 00]
    - smooth Dirac measure with kernels
  - block sampling & fixed lag smoothing [Briers & Doucet 05], [Johansen 15]
    - propagate at each time $(X_{n-L+1}, \ldots, X_n)$
  - tempering and PF [Godsill & Clapp 01]
  - ABC-style filters [Jasra et. al 11, Campillo & Rossi 09]
    - perturbed observations or likelihood without densities
  - conditional SMC [Andrieu et. al. 10], twisted PFs [Whiteley & Lee 14], space time PF [Beskos et. al. 16] ....

Discussion

- There are more elaborate particle filtering algorithms
  - they work better than vanilla version
    - in terms of variance of estimators, ESS, accuracy etc.
- Weight degeneracy can be still present
  - when $q(x_n|y_n, x_{n-1})p(x_{0:n-1}|y_{0:n-1})$ is very different to $p(x_{0:n}|y_{0:n})$
  - high dimensions, informative likelihoods,...
- Path degeneracy can be addressed using smoothing algorithms
  - with some extra computational cost
- Very interesting theory [Del Moral 06, 13]

Reading List

- Doucet et. al. 1998
  - http://www.stats.ox.ac.uk/~doucet/doucet_godsill_andrieu_sequentialmontecarloforbayesfiltering.pdf
  - or Technical Report version http://www.cs.ubc.ca/~arnaud/doucet_tr310_sequentialmontecarlofiltering.pdf
- Tutorial on filtering: http://www.stats.ox.ac.uk/~doucet/doucet_johansen_tutorialPF2011.pdf

SMC Portals with papers:

- Arnaud Doucet: http://www.stats.ox.ac.uk/~doucet/smc_resources.html
- Pierre Del Moral: http://web.maths.unsw.edu.au/~peterdel-moral/simulinks.html