Sequential Monte Carlo:

What, How and Some Reasons Why

Adam M. Johansen

a.m.johansen@warwick.ac.uk
http://www2.warwick.ac.uk/fac/sci/statistics/staff/academic/johansen/talks

46th Gregynog Statistical Conference

16th April, 2010

Outline

- Background
- What?
- How?
- Why?
  - Bayesian Inference
  - Maximum Likelihood Parameter Estimation
  - Rare Event Simulation
  - Filtering (of Piecewise Deterministic Processes)

Introduction

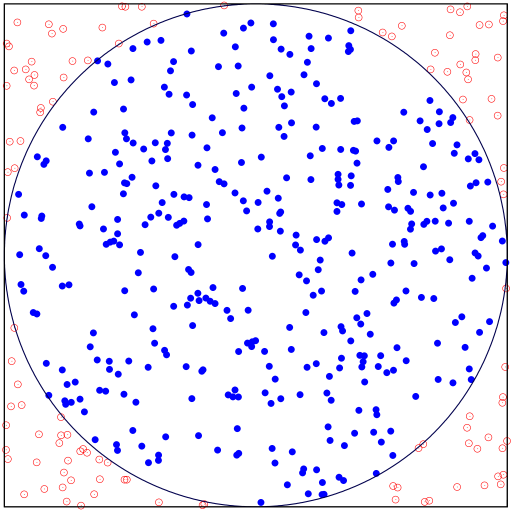
Why Sample from Distributions?

- Integration (Bayesian methods,. . .).
- Solving integral equations.
- Optimisation (SA,. . .).
- Characterisation of the distribution (SMC,. . .).
- Instead of evaluating a density (ABC).

General principle:

- Represent quantity of interest probabilistically.
- Use a sampling interpretation.

Estimating π



- Rain is uniform.
- Circle is inscribed in square.
- $A_{\text{square}} = 4r^2$.
- $A_{\text{circle}} = \pi r^2$.
- $p = \frac{A_{\text{circle}}}{A_{\text{square}}} = \frac{\pi}{4}$.
- 383 of 500 "successes".
- $\hat{\pi} = 4\frac{383}{500} = 3.06$.
- Also obtain confidence intervals.

The Monte Carlo Method

- Given a probability density, f,

$$I = \int_E \varphi(x) f(x) dx$$

- Simple Monte Carlo solution:
  - Sample $X_1, \ldots, X_N \stackrel{iid}{\sim} f$.
  - Estimate $\hat{I} = \frac{1}{N} \sum_{i=1}^{N} \varphi(X_i)$.
- Justified by the law of large numbers...
- and the central limit theorem.
- Can also be viewed as approximating $\pi(dx) = f(x)dx$ with

$$\hat{\pi}^N(dx) = \frac{1}{N} \sum_{i=1}^{N} \delta_{X_i}(dx).$$

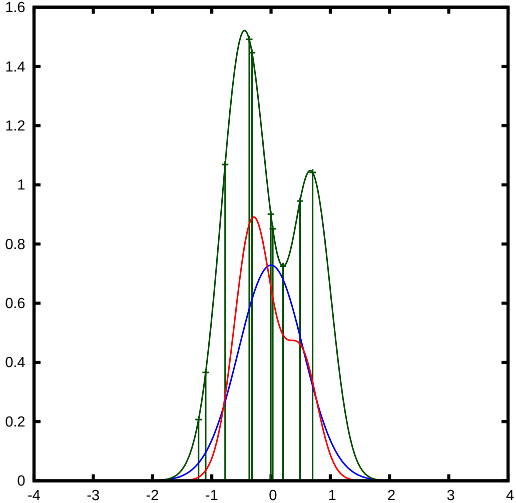
The Importance–Sampling Identity

- Given g, such that
  - $f(x) > 0 \Rightarrow g(x) > 0$
  - and $f(x)/g(x) < \infty$,

define $w(x) = f(x)/g(x)$ and:

$$
\begin{aligned}
I &= \int \varphi(x) f(x) dx \\
&= \int \varphi(x) f(x) g(x)/g(x) dx \\
&= \int \varphi(x) w(x) g(x) dx.
\end{aligned}
$$

Illustration of the Importance Sampling Identity



Importance Sampling

- This suggests the importance sampling estimator:
  - Sample $X_1, \ldots, X_N \overset{iid}{\sim} g$.
  - Estimate $\hat{I} = \frac{1}{N} \sum_{i=1}^{N} w(X_i)\varphi(X_i)$.
- Justified by the law of large numbers...
- and the central limit theorem.
- Can also be viewed as approximating $\pi(dx) = f(x)dx$ with

$$\hat{\pi}^N(dx) = \frac{1}{N} \sum_{i=1}^{N} w(X_i)\delta_{X_i}(dx).$$

Interesting Features of Importance Sampling

- Doesn't require samples from the distribution of interest.
- Variance of

$$\frac{1}{N}\left(\mathbb{E}_g[(w\varphi)^2] - \mathbb{E}_g[w\varphi]^2\right) = \frac{1}{N}\left(\mathbb{E}_f[w\varphi^2] - \mathbb{E}_f[\varphi]^2\right).$$

  Simple Monte Carlo has a variance of

$$\frac{1}{N}\left(\mathbb{E}_f[\varphi^2] - \mathbb{E}_f[\varphi]^2\right).$$

- Importance sampling can *reduce* the variance. If

$$g(x) = \frac{f(x)\varphi(x)}{\int f(x)\varphi(x)dx},$$

  then the variance is exactly 0.

Self-Normalised Importance Sampling

- Often, f is known only up to a normalising constant.
- As $\mathbb{E}_g(Cw\varphi) = C\mathbb{E}_f(\varphi)\ldots$
- If $v(x) = Cw(x)$, then

$$\frac{\mathbb{E}_g(v\varphi)}{\mathbb{E}_g(v\mathbf{1})} = \frac{C\mathbb{E}_f(\varphi)}{C\mathbb{E}_f(\mathbf{1})} = \mathbb{E}_f(\varphi).$$

- Estimate the numerator and denominator with the same sample:

$$\hat{I} = \frac{\sum_{i=1}^{N} v(X_i)\varphi(X_i)}{\sum_{i=1}^{N} v(X_i)}.$$

- Biased for finite samples, but consistent.
- Typically reduces variance.

Resampling

- We can produce unweighted samples from weighted ones.
- Given $\{W_i, X_i\}_{i=1}^N$ a consistent resampling $\{\tilde{X}_i\}_{i=1}^N$ is such that

$$\mathbb{E}\left[\frac{1}{N}\sum_{i=1}^N \varphi(\tilde{X}_i)\middle| \{W_i, X_i\}_{i=1}^N\right] = \sum_{i=1}^N W_i \varphi(X_i)$$

  for any continuous bounded φ.
- Simplest option: sample from empirical distribution

$$\tilde{X}_i \sim \sum_{i=1}^N W_i \delta_{X_i}(\cdot)$$

- Other approaches reduce the *additional* variance.

Markov Chain Monte Carlo

- A Markov chain with kernel $K(x, y)$ is f-invariant iff:

$$\int f(x)K(x, y)dx = f(y).$$

- MCMC simulates such a chain, $X_1, \ldots, X_N$.
- It's ergodic averages:

$$\frac{1}{N}\sum_{i=1}^{N}\varphi(X_i)$$

  approximate $\mathbb{E}_f[\varphi]$.
- Justified by ergodic theorems / central limit theorems.
- Difficulties include:
  - Constructing a good transition kernel.
  - Verifying convergence.

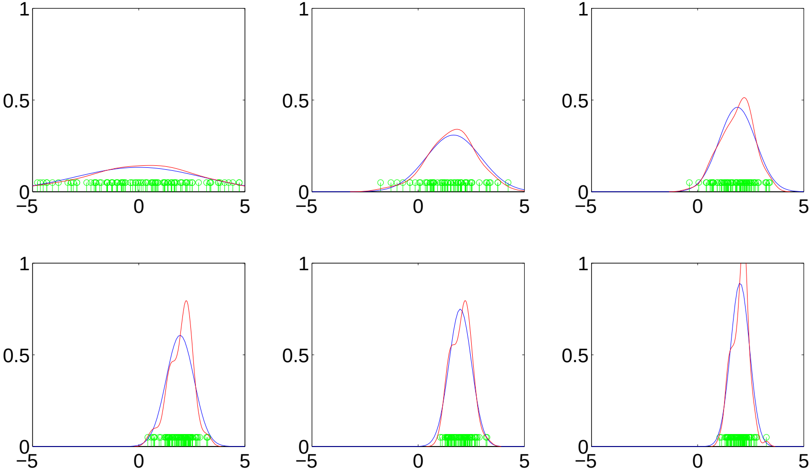
What?

What are sequential Monte Carlo methods?

"A class of methods for sampling from each of an 'arbitrary' sequence of distributions using importance sampling and resampling mechanisms."

Iteratively, efficiently and using the structure of the problem.

Or graphically. . .



How?

A Motivating Example: Filtering / Smoothing

- Let $X_1, \ldots$ denote the position of an object which follows Markovian dynamics:

$$X_n | \{X_{n-1} = x_{n-1}\} \sim f(\cdot | x_{n-1}).$$

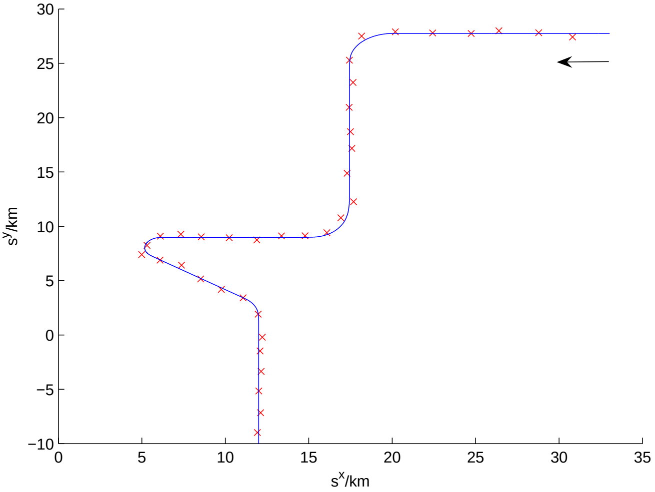
- Let $Y_1, \ldots$ denote a collection of observations:

$$Y_i | \{X_i = x_i\} \sim g(\cdot | x_i).$$

- Smoothing: estimate, as observations arrive, $p(x_{1:n} | y_{1:n})$.
- Filtering: estimate, as observations arrive, $p(x_n | y_{1:n})$.
- Formal Solution:

$$p(x_{1:n} | y_{1:n}) = p(x_{1:n-1} | y_{1:n-1}) \frac{f(x_n | x_{n-1}) g(y_n | x_n)}{p(y_n | y_{1:n-1})}$$

How... An Illustrative Example



But we could do importance sampling...

- If we sample $\{X_{1:n}^{(i)}\}$ at time n from $q_n(x_{1:n})$, define

$$w_n(x_{1:n}) \propto \frac{p(x_{1:n}|y_{1:n})}{q(x_{1:n})} = \frac{p(x_{1:n}, y_{1:n})}{q(x_{1:n})p(y_{1:n})}$$
$$\propto \frac{f(x_1)g(y_1|x_1)\prod_{m=2}^{n} f(x_m|x_{m-1})g(y_m|x_m)}{q_n(x_{1:n})}$$

- and set $W_n^{(i)} = w_n(X_{1:n}^{(i)})/\sum_j w_n(X_{1:n}^{(j)})$,
- then $\{W_n^{(i)}, X_n^{(i)}\}$ is a consistently weighted sample.
- This seems inefficient.

Sequential Importance Sampling I

- Importance weight

$$w_n(x_{1:n}) \propto \frac{f(x_1)g(y_1|x_1)\prod_{m=2}^{n} f(x_m|x_{m-1})g(y_m|x_m)}{q_n(x_{1:n})}$$

$$= \frac{f(x_1)g(y_1|x_1)}{q_n(x_1)} \prod_{m=2}^{n} \frac{f(x_m|x_{m-1})g(y_m|x_m)}{q_n(x_m|x_{1:m-1})}$$

- Given $\{W_{n-1}^{(i)}, X_{1:n-1}^{(i)}\}$ targetting $p(x_{1:n-1}|y_{1:n-1})$
- We could let $q_n(x_{1:n-1}) = q_{n-1}(x_{1:n-1})$ and sample each $X_n^{(i)} \sim q_n(\cdot|X_{n-1}^{(i)})$.

Sequential Importance Sampling II

- And update the weights:

$$
\begin{aligned}
w_n(x_{1:n}) =& w_{n-1}(x_{1:n-1}) \frac{f(x_n|x_{n-1})g(y_n|x_n)}{q_n(x_n|x_{n-1})} \\
W_n^{(i)} =& w_n(X_{1:n}^{(i)}) \\
=& w_{n-1}(X_{1:n-1}^{(i)}) \frac{f(X_n^{(i)}|X_{n-1}^{(i)})g(y_n|X_n^{(i)})}{q_n(X_n^{(i)}|X_{n-1}^{(i)})} \\
=& W_{n-1}^{(i)} \frac{f(X_n^{(i)}|X_{n-1}^{(i)})g(y_n|X_n^{(i)})}{q_n(X_n^{(i)}|X_{n-1}^{(i)})}
\end{aligned}
$$

- If $\int p(x_{1:n}|y_{1:n})dx_n \approx p(x_{1:n-1}|y_{1:n-1})$ this makes sense.
- We only need to store $\{W_n^{(i)}, X_{n-1:n}^{(i)}\}$.
- Same computation every iteration.

Importance Sampling on Huge Spaces Doesn't Work

- It's said that IS *breaks the curse of dimensionality*:

$$\sqrt{N}\left[\frac{1}{N}\sum_{i=1}^{N} w(X_i)\varphi(X_i) - \int \varphi(x) f(x) dx\right] \xrightarrow{d} \mathcal{N}(0, \mathsf{Var}_g(w\varphi))$$

- This is true.
- But it's not *enough*.
- $\mathsf{Var}_g(w\varphi)$ increases (often exponentially) with dimension.
- Eventually, an SIS estimator (of $p(x_{1:n}|y_{1:n})$) will fail.
- We're only concerned with $p(x_n|y_{1:n})$: a *fixed-dimensional* distribution.

Resampling Again: The SIR Algorithm

- Problem: variance of the weights builds up over time.
- Solution? Given $\{W_{n-1}^{(i)}, X_{1:n-1}^{(i)}\}$:
  1. Resample, to obtain $\{\frac{1}{N}, \widetilde{X}_{1:n-1}^{(i)}\}$.
  2. Sample $X_n^{(i)} \sim q_n(\cdot|\widetilde{X}_{n-1}^{(i)})$.
  3. Set $X_{1:n-1}^{(i)} = \widetilde{X}_{1:n-1}^{(i)}$.
  4. Set $W_n^{(i)} = f(X_n^{(i)}|X_{n-1}^{(i)})g(y_n|X_n^{(i)})/q_n(X_n^{(i)}|X_{n-1}^{(i)})$.
- And continue as with SIS.
- There is a cost, but this really works.

Cf. Doucet and Johansen, 2010 (4) for a review of "particle filtering" methods.

More Generally

- The problem in the previous example is really tracking a sequence of distributions.
- Key structural properties:
  - Size of space is increasing with time.
  - Consistency between existing part between distributions.
  - Most interested in what's new.
- Any problem of sequentially approximating a sequence of such distributions, p_n, can be addressed in the same way.

Importance Sampling in This Setting

- Given $p_n(x_{1:n})$ for $n = 1, 2, \ldots$.
- We could sample from a sequence $q_n(x_{1:n})$ for each n.
- Or we could let $q_n(x_{1:n}) = q_n(x_n|x_{1:n-1})q_{n-1}(x_{1:n-1})$ and re-use our samples.
- The importance weights become:

$$w_n(x_{1:n}) \propto \frac{p_n(x_{1:n})}{q_n(x_{1:n})} = \frac{p_n(x_{1:n})}{q_n(x_n|x_{1:n-1})q_{n-1}(x_{1:n-1})}$$

$$= \frac{p_n(x_{1:n})}{q_n(x_n|x_{1:n-1})p_{n-1}(x_{1:n-1})} w_{n-1}(x_{1:n-1})$$

Sequential Importance Sampling

<u>At time 1.</u>

For $i = 1 : N$, sample $X_1^{(i)} \sim q_1(\cdot)$.

For $i = 1 : N$, compute $W_1^{(i)} \propto w_1\left(X_1^{(i)}\right) = \frac{p_1\left(X_1^{(i)}\right)}{q_1\left(X_1^{(i)}\right)}$.

<u>At time n, $n \geq 2$.</u>

Sampling Step

For $i = 1 : N$, sample $X_n^{(i)} \sim q_n\left(\cdot \middle| X_{n-1}^{(i)}\right)$.

Weighting Step

For $i = 1 : N$, compute

$$w_n\left(X_{1:n-1}^{(i)}, X_n^{(i)}\right) = \frac{p_n\left(X_{1:n-1}^{(i)}, X_n^{(i)}\right)}{p_{n-1}\left(X_{1:n-1}^{(i)}\right) q_n\left(X_n^{(i)} \middle| X_{n-1}^{(i)}\right)}$$

and $W_n^{(i)} \propto W_{n-1}^{(i)} w_n\left(X_{1:n-1}^{(i)}, X_n^{(i)}\right)$.

Sequential Importance Resampling

At time n, $n \geq 2$.

Sampling Step

For $i = 1 : N$, sample $X_{n,n}^{(i)} \sim q_n\left(\cdot \middle| \widetilde{X}_{n-1}^{(i)}\right)$.

Resampling Step

For $i = 1 : N$, compute

$$w_n\left(\widetilde{X}_{n-1}^{(i)}, X_{n,n}^{(i)}\right) = \frac{p_n\left(\widetilde{X}_{n-1}^{(i)}, X_{n,n}^{(i)}\right)}{p_{n-1}\left(\widetilde{X}_{n-1}^{(i)}\right) q_n\left(X_{n,n}^{(i)} \middle| \widetilde{X}_{n-1}^{(i)}\right)}$$

and $W_n^{(i)} = \frac{w_n\left(\widetilde{X}_{n-1}^{(i)}, X_{n,n}^{(i)}\right)}{\sum_{j=1}^{N} w_n\left(\widetilde{X}_{n-1}^{(j)}, X_{n,n}^{(j)}\right)}$.

For $i = 1 : N$, sample $\widetilde{X}_n^{(i)} \sim \sum_{j=1}^{N} W_n^{(j)} \delta_{\left(\widetilde{X}_{n-1}^{(j)}, X_{n,n}^{(j)}\right)}\left(dx_{1:n}\right)$.

SMC Samplers: In Essence

- Let η_{n-1}, η_n be distributions over E.
- Let K_n and L_{n-1} be Markov kernels from E to E.
- Given a set of weighted samples $\{X_{n-1}^{(i)}, W_{n-1}^{(i)}\}_{i=1}^{N}$ such that

$$X_{n-1}^{(i)} \sim q_{n-1} \text{ and } W_{n-1}^{(i)} = \eta_{n-1}(X_{n-1}^{(i)})/q_{n-1}(X_{n-1}^{(i)}):$$

  - Sample $X_n^{(i)} \sim K_n\left(X_{n-1}^{(i)}, \cdot\right)$.
  - Calculate $W_n^{(i)} \propto W_{n-1}^{(i)} \frac{\eta_n(X_n^i)L_{n-1}(X_n^{(i)},X_{n-1}^{(i)})}{\eta_{n-1}(X_{n-1}^{(i)})K_n(X_{n-1}^{(i)},X_n^{(i)})}$
  - Now, $\{W_n^{(i)}, (X_{n-1}^{(i)}, X_n^{(i)})\}$ targets $\eta_n(x_n)L_{n-1}(x_n, x_{n-1})$ and marginally $\{W_n^{(i)}, X_n^{(i)}\}$ targets $\eta_n(x_n)$.

Del Moral et al., 2006 (3) suggest the SMC Sampler for a sequence of distributions $\eta_1, \eta_2, \ldots$

- Sample $X_n^{(i)} \sim K_n(X_{n-1}^{(i)}, \cdot)$.
- $\left\{(X_{n-1}^{(i)}, X_n^{(i)}), W_{n-1}^{(i)}\right\}_{i=1}^{N} \sim \eta_{n-1}(X_{n-1})K_n(X_{n-1}, X_n)$.
- Set weights $W_n^{(i)} = W_{n-1}^{(i)} \frac{\eta_n(X_n)L_{n-1}(X_n, X_{n-1})}{\eta_{n-1}(X_{n-1})K_n(X_{n-1}, X_n)}$.
- Thus:

$$\left\{(X_{n-1}^{(i)}, X_n^{(i)}), W_n^{(i)}\right\}_{i=1}^{N} \overset{targets}{\sim} \eta_n(X_n)L_{n-1}(X_n, X_{n-1})$$

and, marginally, $\left\{X_n^{(i)}, W_n^{(i)}\right\}_{i=1}^{(i)} \overset{targets}{\sim} \eta_n$.

- Optionally, resample to obtain an unweighted particle set.

SMC Samplers are SIR Algorithms

- Given a sequence of *target* distributions, η_n, on $E_n \ldots$,
- construct a synthetic sequence $\widetilde{\eta}_n$ on spaces $\bigotimes_{p=1}^{n} E_p$
- by introducing Markov kernels, L_p from E_{p+1} to E_p:

$$\widetilde{\eta}_n(x_{1:n}) = \eta_n(x_n) \prod_{p=1}^{n-1} L_p\left(x_{p+1}, x_p\right),$$

- These distributions
  - have the target distributions as time marginals,
  - have the correct structure to employ SMC techniques,
  - lead to precisely the SMC sampler algorithm.

SMC Outline

- Given a sample $\{X_{1:n-1}^{(i)}\}_{i=1}^{N}$ targeting $\widetilde{\eta}_{n-1}$,
- sample $X_n^{(i)} \sim K_n(X_{n-1}^{(i)}, \cdot)$,
- calculate

$$W_n(X_{1:n}^{(i)}) = \frac{\eta_n(X_n^{(i)})L_{n-1}(X_n^{(i)}, X_{n-1}^{(i)})}{\eta_{n-1}(X_{n-1}^{(i)})K_n(X_{n-1}^{(i)}, X_n^{(i)})}.$$

- Resample, yielding: $\{X_{1:n}^{(i)}\}_{i=1}^{N}$ targeting $\widetilde{\eta}_n$.
- Hints that we'd like to use

$$L_{n-1}(x_n, x_{n-1}) = \frac{\eta_{n-1}(x_{n-1})K_n(x_{n-1}, x_n)}{\int \eta_{n-1}(x'_{n-1})K_n(x'_{n-1}, x_n)}.$$

Things to remember when doing SMC

- Choose proposals which ensure weights are bounded.
- Logarithms are good:
  - Unnormalized weights may be very large or small.
  - Importance weights may be the ratio of two similar expressions.
- Efficient resampling algorithms are $\mathcal{O}(N)$.
- Parallelisation is possible, but resampling complicates things.

Actually, it's rather easy in MatLab/R or similar.

SMCTC: C++ Template Class for SMC Algorithms

- Implementing SMC algorithms in C/C++ isn't hard.

- Software for implementing general SMC algorithms (9).
- C++ element largely confined to the library.
- Available (under a GPL-3 license from)

  `www2.warwick.ac.uk/fac/sci/statistics/staff/academic/johansen/smctc/`

  or type "smctc" into google.
- Example code included.

Why?

Bayesian Inference

See:

- Chopin, 2004 (1)
- Del Moral, Doucet and Jasra 2006 (2)
- Fan, Leslie and Wand 2008 (6)

and others.

Bayesian Inference and Decision Making

Given

- prior $p(\theta)$,
- likelihood $p(y|\theta)$ and data y,
- Bayesian inference depends upon

$$p(\theta|y) = p(y|\theta)p(\theta)/p(y)$$

Given a loss function $L(d, \theta)$ we're interested in minimising

$$\bar{L}(d) = \int L(d, \theta)p(\theta|y)d\theta$$

With $L_{\mathrm{SE}}(d, \theta) = (d - \theta)^2$:

$$d^\star_{\mathrm{SE}} = \int \theta p(\theta|y)d\theta.$$

Data Tempering — Online Bayesian Inference

- Given data, $y_{1,2,\dots}$ we have:

$$\begin{aligned}
\text{Prior:} \qquad \eta_0(\theta) &= p(\theta) \\
\eta_1(\theta) &= p(\theta|y_1) \propto p(y_1|\theta)p(\theta) \\
\eta_2(\theta) &= p(\theta|y_{1:2}) \propto p(y_{1:2}|\theta)p(\theta) \\
&\vdots \\
\text{Posterior:} \qquad \eta_t(\theta) &= p(\theta|y_{1:t}) \propto p(y_{1:t}|\theta)p(\theta)
\end{aligned}$$

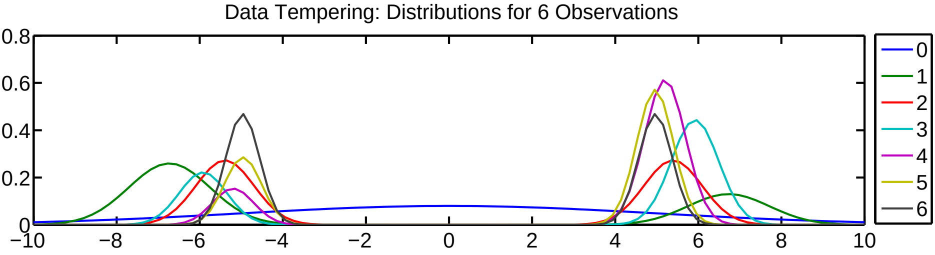
- $\eta_t(\theta) \propto \eta_{t-1}(\theta)p(y_t|\theta, y_{1:t-1})$ — ideal for online inference.
- We can be flexible with $\{\eta_n\}$.
- Appealing interpretability.

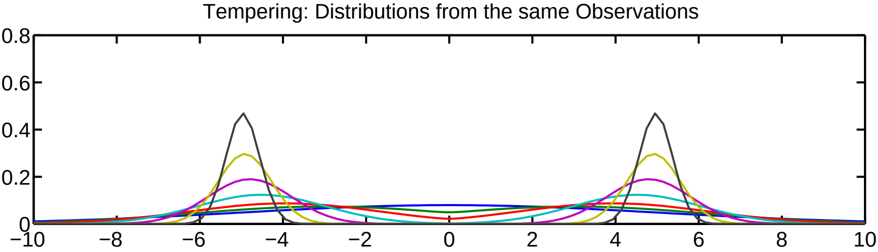
Tempering — Offline Bayesian Inference

- Given data, $y_{1,2,\ldots,t}$ we have:

$$
\begin{aligned}
\text{Prior:} \qquad \eta_0(\theta) =& p(\theta) = p(\theta)p(y_{1:t}|\theta)^0 \\
\eta_1(\theta) \propto& p(y_{1:t}|\theta)^{\gamma_1} p(\theta) \\
\eta_2(\theta) \propto& p(y_{1:t}|\theta)^{\gamma_2} p(\theta) \\
&\vdots \\
\text{Posterior:} \qquad \eta_P(\theta) =& p(\theta|x_{1:t}) \propto p(x_{1:n}|\theta)^1 p(\theta).
\end{aligned}
$$

- Choose $\{\gamma_n\}_{n=0}^P$ (non-decreasing, from 0 to 1).
- More regular than DT for offline inference.


Data Tempering: Distributions for 6 Observations
0.8
0.6
0.4
0.2
0
−10
−8
−6
−4
−2
0
2
4
6
8
10
0
1
2
3
4
5
6


Tempering: Distributions from the same Observations
0.8
0.6
0.4
0.2
0
−10
−8
−6
−4
−2
0
2
4
6
8
10

Example: Changepoint Detection[1]

- Given data, $y_{1:t}$ modelled by:

$$\begin{aligned} Y_t|\{S_{1:t-1} = s_{1:t-1}, Y_{1:t-1} = y_{1:t-1}\} &\sim g_\theta(\cdot; S_{t-r:t}, y_{1:t-1}) \\ S_t|\{S_{1:t-1} = s_{1:t-1}, Y_{1:t-1} = y_{1:t-1}\} &\sim f_\theta(\cdot; s_{t-1}) \end{aligned}$$

- *Changepoints* are:
  the beginning of a run of length $\geq k$ in $\{S_t\}$
- Given θ, the changepoint distribution is available explicitly.
- What about parameter uncertainty?

[1] Thanks to Christopher Nam and John Aston

An SMC approach to Parameter Uncertainty

- Let $\eta_0(\theta) = p(\theta)$ and $\eta_n(\theta) = p(\theta)p(y|\theta)^{\gamma_n}$.
- Use SMC to obtain a marginal approximation of $p(\theta|y)$:

$$\widehat{p}(\theta|y) = \sum_{i=1}^{n} W_T^{(i)} \delta_{\theta_T^{(i)}}(\theta)$$

- Look at the marginal of interest:

$$\begin{aligned} p(CP|y) &= \int p(CP|y,\theta)p(\theta|y)d\theta \\ &\approx \int p(CP|y,\theta)\widehat{p}(\theta|y)d\theta \\ &= \sum_{i=1}^{n} W_T^{(i)} p(CP|y,\theta_T^{(i)}) \end{aligned}$$

- A Monte Carlo estimate of the marginal distribution.

Parameter Estimation in Latent Variable Models

See Johansen, Doucet and Davy 2008 (11)

Maximum {Likelihood|*a Posteriori*} Estimation

- Consider a model with:
  - parameters, θ,
  - latent variables, x, and
  - observed data, y.
- Aim to maximise Marginal likelihood

$$p(y|\theta) = \int p(x, y|\theta) dx$$

or posterior

$$p(\theta|y) \propto \int p(x, y|\theta) p(\theta) dx.$$

- Traditional approach is Expectation-Maximisation (EM)
  - Requires objective function in closed form.
  - Susceptible to trapping in local optima.

A Probabilistic Approach

- A distribution of the form

$$\pi(\theta|y) \propto p(\theta)p(y|\theta)^{\gamma}$$

will become concentrated, as $\gamma \to \infty$ on the maximisers of $p(y|\theta)$ under weak conditions (Hwang, 1980).

- **Key point:** Synthetic distributions of the form:

$$\bar{\pi}_{\gamma}(\theta, x_{1:\gamma}|y) \propto p(\theta) \prod_{i=1}^{\gamma} p(x_i, y|\theta)$$

admit the marginals

$$\bar{\pi}_{\gamma}(\theta|y) \propto p(\theta)p(y|\theta)^{\gamma}.$$

Maximum Likelihood via SMC

- Use a sequence of distributions $\eta_n = \pi_{\gamma_n}$ for some $\{\gamma_n\}$.
- Suggested in an MCMC context [Doucet et al., 2002 (5)].
  - Requires extremely slow "annealing".
  - Separation between distributions is large.
- SMC has two main advantages:
  - Introducing bridging distributions, for $\gamma = \lfloor\gamma\rfloor + \langle\gamma\rangle$, of:

$$\bar{\pi}_\gamma(\theta, x_{1:\lfloor\gamma\rfloor+1}|y) \propto p(\theta) p(x_{\lfloor\gamma\rfloor+1}, y|\theta)^{\langle\gamma\rangle} \prod_{i=1}^{\lfloor\gamma\rfloor} p(x_i, y|\theta)$$

    is straightforward.
  - Population of samples improves robustness.

Three Algorithms

- A generic SMC sampler can be written down directly...
- Easy case:
  - Sample from $p(x_n|y, \theta_{n-1})$ and $p(\theta_n|x_n, y)$.
  - Weight according to $p(y|\theta_{n-1})^{\gamma_n - \gamma_{n-1}}$.
- General case:
  - Sample existing variables from a η_{n-1}-invariant kernel:

$$(\theta_n, X_{n,1:\gamma_{n-1}}) \sim \mathcal{K}_{n-1}((\theta_{n-1}, X_{n-1}), \cdot).$$

  - Sample new variables from an arbitrary proposal:

$$X_{n,\gamma_{n-1}+1:\gamma_n} \sim q(\cdot|\theta_n).$$

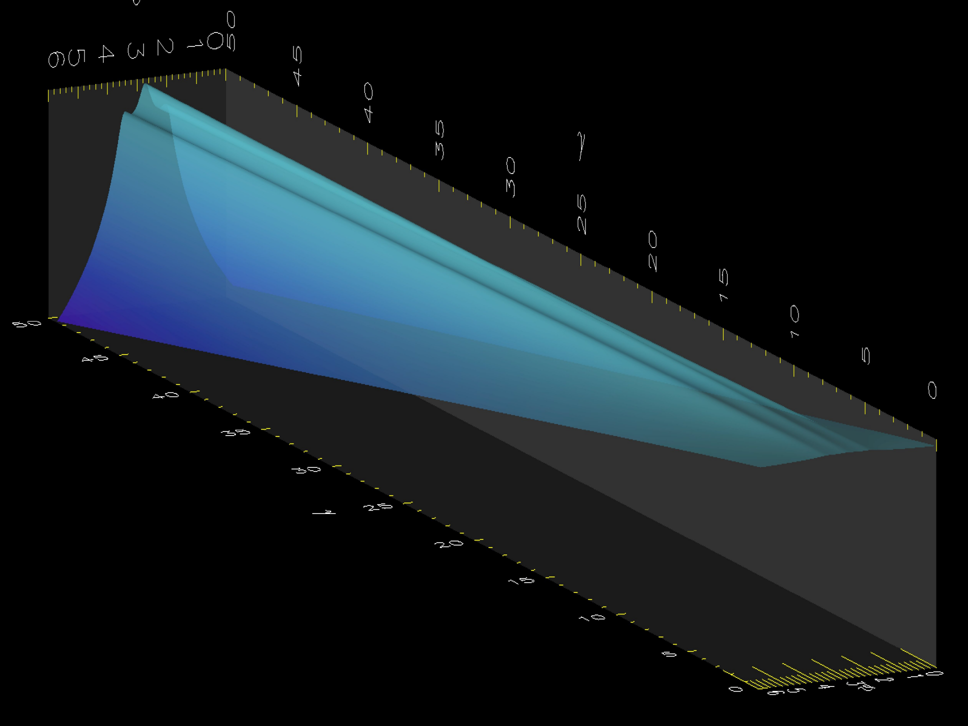
  - Use the composition of a time-reversal and optimal auxiliary kernel.
  - Weight expression does not involve the marginal likelihood.

Toy Example

- Student t-distribution of unknown location parameter θ with $\nu = 0.05$.
- Four observations are available, $y = (-20, 1, 2, 3)$.
- Log likelihood is:

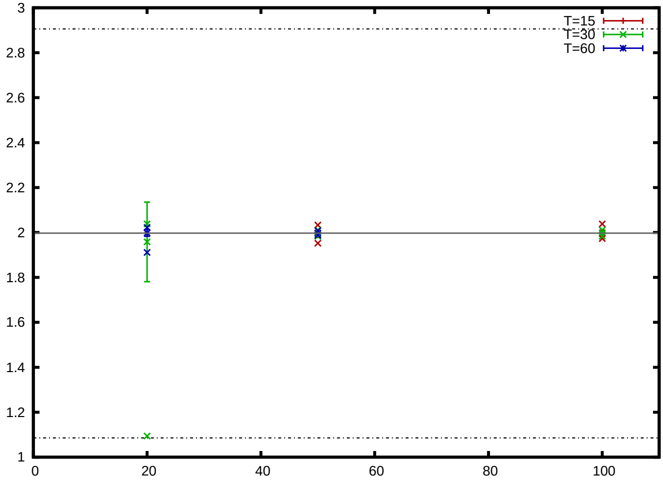
$$\log p(y|\theta) = -0.525 \sum_{i=1}^{4} \log\left(0.05 + (y_i - \theta)^2\right).$$

- Global maximum is at 1.997.
- Local maxima at $\{-19.993, 1.086, 2.906\}$.



y
0
5
10
15
20
25
30
35
40
45
50

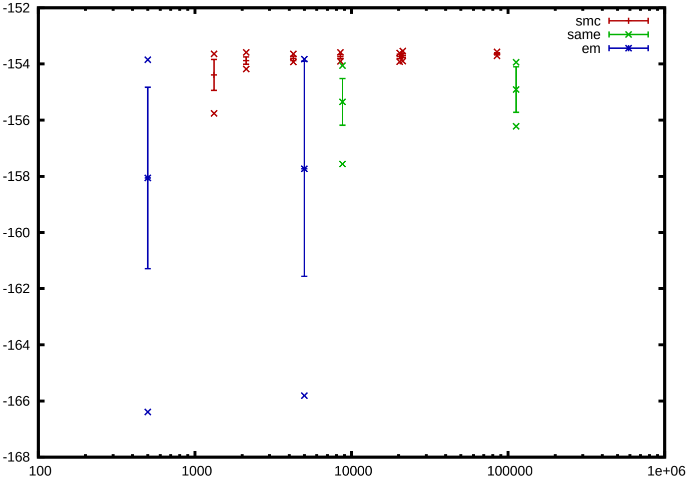
It actually works. . .



Example: Gaussian Mixture Model – MAP Estimation

- Likelihood $p(y|x, \omega, \mu, \sigma) = \mathcal{N}(y|\mu_x, \sigma_x^2)$.
- Marginal likelihood $p(y|\omega, \mu, \sigma) = \sum_{j=1}^{3} \omega_j \mathcal{N}(y|\mu_j, \sigma_j^2)$.
- Diffuse conjugate priors were employed.
- All full conditional distributions of interest are available.
- Marginal posterior can be calculated.

Example: GMM (Galaxy Data Set)



Rare Event Simulation

See Johansen, Doucet and Del Moral, 2006 (10).

The Trouble with Rare Events

- Consider a random variable, X, with density f.
- If $\{X \in \mathcal{T}\}$ is a *rare* event, $p = \mathbb{P}(\{X \in \mathcal{T}\}) < 10^{-6}$.
- With simple Monte Carlo simulation $X^{(i)} \sim \mathbb{P}$:

$$\mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N}\mathbb{I}_{\mathcal{T}}(X^{(i)})\right] = \mathbb{P}(\{X \in \mathcal{T}\}) = p$$

$$\mathsf{Var}\left[\frac{1}{N}\sum_{i=1}^{N}\mathbb{I}_{\mathcal{T}}(X^{(i)})\right] = p(1-p)/N$$

- But $\sqrt{p(1-p)/N}/p \approx \sqrt{1/Np}$.

Importance Sampling of Rare Events

- In principle, if we sample from:

$$g(x) = \frac{f(x)\mathbb{I}_{\mathcal{T}}(x)}{\int f(x')\mathbb{I}_{\mathcal{T}}(x')dx'}$$

- And use weighting:

$$w(x) = \frac{f(x)}{g(x)} = f(x)\frac{\int f(x')\mathbb{I}_{\mathcal{T}}(x')dx'}{f(x)\mathbb{I}_{\mathcal{T}}(x)}$$
$$\overset{a.e.}{=} \int f(x')\mathbb{I}_{\mathcal{T}}(x')dx'$$

- We get the answer with zero variance using 1 sample.

Static Rare Events

Consider *static rare events*:

- Do the first $P+1$ elements of a Markov chain lie in a $\mathcal{T}$?
- We are interested in

$$\mathbb{P}_{\mu_0}\left(x_{0:P} \in \mathcal{T}\right)$$

  and

$$\mathbb{P}_{\mu_0}\left(x_{0:P} \in dx_{0:P} \left| x_{0:P} \in \mathcal{T}\right.\right)$$

- We assume that the rare event is characterised as a level set of a suitable potential function:

$$V: \mathcal{T} \rightarrow [\hat{V}, \infty), \text{ and } V: E_{0:P} \setminus \mathcal{T} \rightarrow (-\infty, \hat{V}).$$

Static Rare Events: Our Approach

- Initialise by sampling from the law of the Markov chain.
- Iteratively obtain samples from a sequence of distributions which moves "smoothly" towards the target.
- Proposed sequence of distributions:

$$\eta_n(dx_{0:P}) \propto \mathbb{P}_{\mu_0}(dx_{0:P}) g_{n/T}(x_{0:P})$$

$$g_\theta(x_{0:P}) = \left(1 + \exp\left(-\alpha(\theta)\left(V(x_{0:P}) - \hat{V}\right)\right)\right)^{-1}$$

- Estimate the normalising constant of the final distribution and correct via importance sampling.

Path Sampling [See ⋆⋆ or Gelman and Meng, 1998]

- Given a sequence of densities $p(x|\theta) = q(x|\theta)/z(\theta)$:

$$\frac{\mathrm{d}}{\mathrm{d}\theta} \log z(\theta) = \mathbb{E}_\theta \left[\frac{\mathrm{d}}{\mathrm{d}\theta} \log q(\cdot|\theta) \right] \qquad (\star)$$

  where the expectation is taken with respect to $p(\cdot|\theta)$.
- Consequently, we obtain:

$$\log \left(\frac{z(1)}{z(0)} \right) = \int_0^1 \mathbb{E}_\theta \left[\frac{\mathrm{d}}{\mathrm{d}\theta} \log q(\cdot|\theta) \right]$$

- In our case, we use our particle system to approximate *both* integrals.

Approximate the path sampling identity to estimate the normalising constant:

$$\hat{Z}_1 = \frac{1}{2}\exp\left[\sum_{n=1}^{T}\left(\alpha(n/T) - \alpha((n-1)/T)\right)\frac{\hat{E}_{n-1} + \hat{E}_n}{2}\right]$$

$$\hat{E}_n = \frac{\sum_{j=1}^{N} W_n^{(j)} \frac{V\left(X_n^{(j)}\right) - \hat{V}}{1+\exp\left(\alpha_n\left(V\left(X_n^{(j)}\right)-\hat{V}\right)\right)}}{\sum_{j=1}^{N} W_n^{(j)}}$$

Estimate the rare event probability:

$$p^\star = \hat{Z}_1 \frac{\sum_{j=1}^{N} W_T^{(j)}\left(1+\exp(\alpha(1)(V\left(X_T^{(j)}\right) - \hat{V}))\right)\mathbb{I}_{(\hat{V},\infty]}\left(V\left(X_T^{(j)}\right)\right)}{\sum_{j=1}^{N} W_T^{(j)}}.$$

Example: Gaussian Random Walk

- A toy example: $M_t(R_{t-1}, R_t) = \mathcal{N}(R_t | R_{t-1}, 1)$.
- $\mathcal{T} = \mathbb{R}^P \times [\hat{V}, \infty)$.
- Proposal kernel:

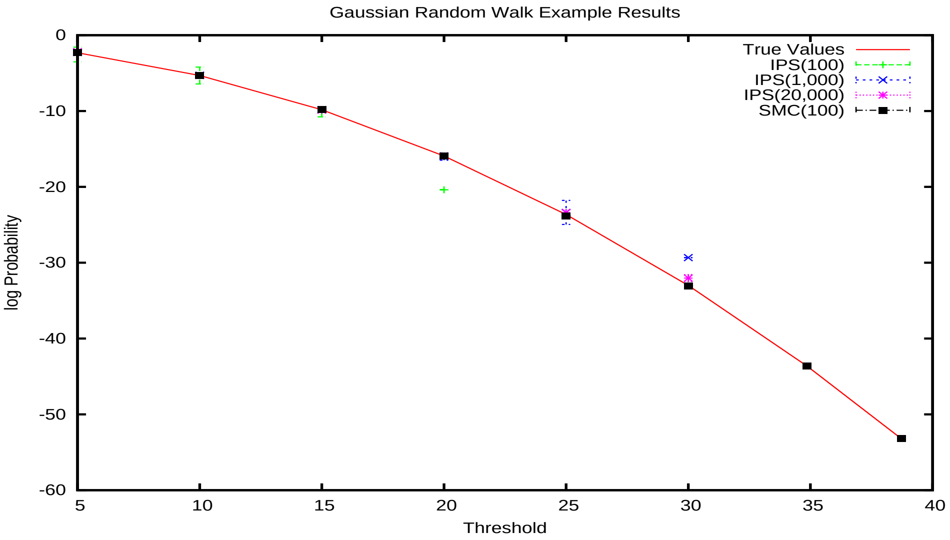
$$K_n(X_{n-1}, X_n) = \sum_{j=-S}^{S} \alpha_{n+1}(X_{n-1}, X_n) \prod_{i=1}^{P} \delta_{X_{n-1,i}+ij\delta}(X_{n,i}),$$

  where the weighting of individual moves is given by

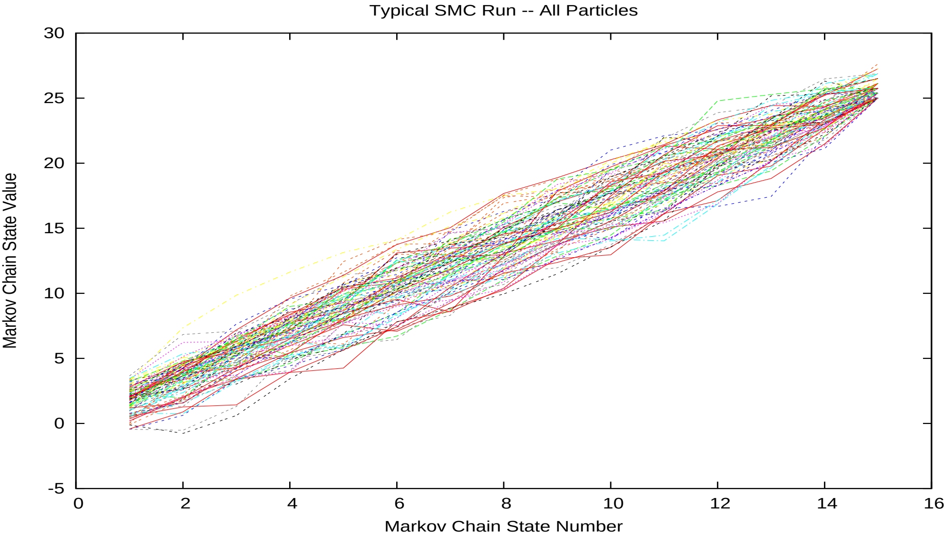
$$\alpha_n(X_{n-1}, X_n) \propto \eta_n(X_n).$$

- Linear annealing schedule.
- Number of distributions $T \propto \hat{V}^{3/2}$ (T=2500 when $\hat{V} = 25$).

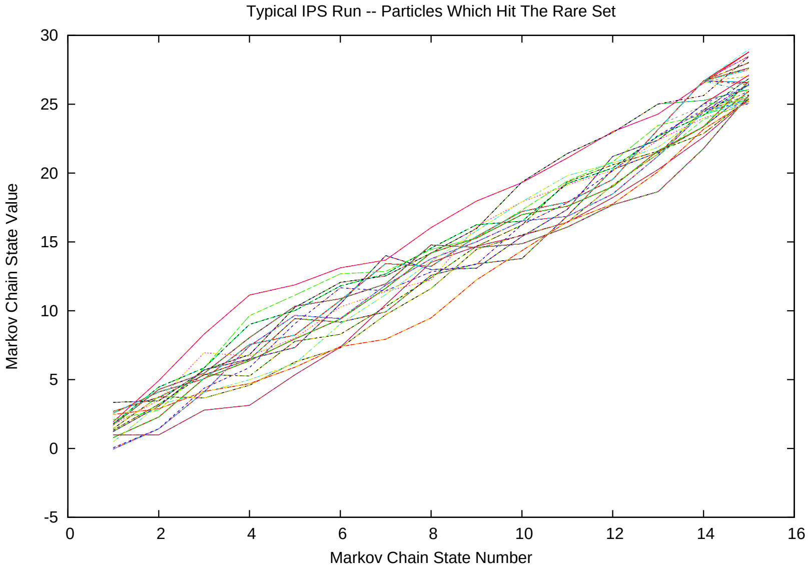
Rare Events



Gaussian Random Walk Example Results
True Values
IPS(100)
IPS(1,000)
IPS(20,000)
SMC(100)
log Probability
Threshold
0
-10
-20
-30
-40
-50
-60
5
10
15
20
25
30
35
40


Typical SMC Run -- All Particles
Markov Chain State Value
Markov Chain State Number
30
25
20
15
10
5
0
-5
0
2
4
6
8
10
12
14
16

Rare Events


Typical IPS Run -- Particles Which Hit The Rare Set
Markov Chain State Value
Markov Chain State Number
30
25
20
15
10
5
0
-5
0
2
4
6
8
10
12
14
16

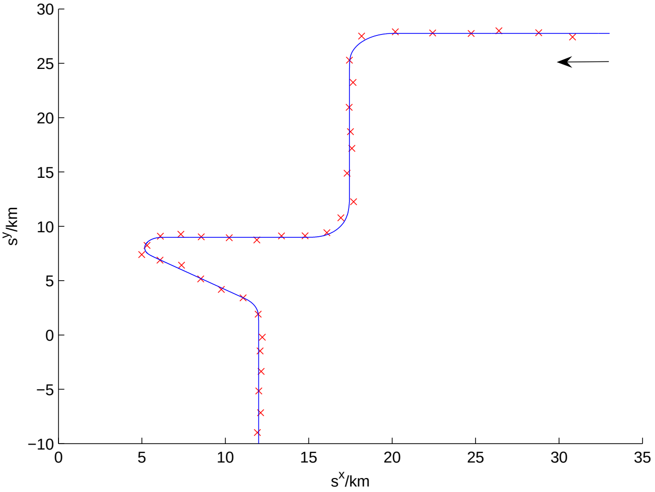
Filtering of Piecewise Deterministic Processes

See Whiteley, Johansen and Godsill, 2007;2010 (12, 13)

Motivation: Observing a Manoeuvring Object

- For $t \in \mathbb{R}_0^+$, consider object with
  - position s_t,
  - velocity v_t and
  - acceleration a_t
- Let $\zeta_t = (s_t, v_t, a_t)$
- From initial condition ζ_0, state evolves until random time τ_1, at which acceleration jumps to a new random value, yielding ζ_{τ_1}
- From ζ_{τ_1}, evolution until τ_2, state becomes ζ_{τ_2}, etc.
- At each Observation time, $(t_n)_{n \in \mathbb{N}}$, a noisy measurement of the object's position is made.

Filtering



An Abstract Formulation

- Pair Markov chain $(\tau_j, \theta_j)_{j \in \mathbb{N}}$, $\tau_j \in \mathbb{R}^+$, $\theta_j \in \Theta$

$$p(d(\tau_j, \theta_j)|\tau_{j-1}, \theta_{j-1}) = q(d\theta_j|\theta_{j-1}, \tau_j, \tau_{j-1}) f(d\tau_j|\tau_{j-1}),$$

- Count the jumps $\nu_t := \sum_j \mathbb{I}_{[\tau_j \leq t]}$
- Deterministic evolution function $F : \mathbb{R}_0^+ \times \Theta \to \Theta$, s.t. $\forall \theta \in \Theta$,

$$F(0, \theta) = \theta$$

- Signal process $(\zeta_t)_{t \in \mathbb{R}_0^+}$,

$$\zeta_t := F(t - \tau_{\nu_t}, \theta_{\nu_t})$$

Filtering 1

- This describes a Piecewise Deterministic Process.
- It's partially observed via observations $(Y_n)_{n \in \mathbb{N}}$, e.g.,

$$Y_n = G(\zeta_{t_n}) + V_n$$

  and likelihood function $g_n(y_n|\zeta_{t_n})$
- Filtering: given observations, $y_{1:n}$, estimate ζ_{t_n}.
- How can we approximate $p(\zeta_{t_n}|y_{1:n})$, $p(\zeta_{t_{n+1}}|y_{1:n+1})$, ... ?

Filtering 2

- Sequence of spaces $(E_n)_{n \in \mathbb{N}}$,

$$E_n = \biguplus_{k=0}^{\infty} \{k\} \times \mathbb{T}_{n,k} \times \Theta^{k+1},$$

$$\mathbb{T}_{n,k} = \{\tau_{1:k} : 0 < \tau_1 < \tau_2 < \ldots < \tau_k \leq t_n\}.$$

- Define $k_n := \nu_{t_n}$ and $X_n = (\zeta_0, k_n, \tau_{1:k_n}, \theta_{1:k_n}) \in E_n$
- Sequence of posterior distributions $(\eta_n)_{n \in \mathbb{N}}$

$$\eta_n(x_n) \propto q(\zeta_0) \prod_{j=1}^{k_n} f(\tau_j|\tau_{j-1}) q(\theta_j|\theta_{j-1}, \tau_j, \tau_{j-1}) \times \prod_{p=1}^{n} g_p(y_p|\zeta_{t_p}) S(\tau_{k_n}, t_n)$$

SMC Filtering

- Recall $X_n = (\zeta_0, k_n, \tau_{1:k_n}, \theta_{1:k_n})$ specifies a path $(\zeta_t)_{t \in [0, t_n]}$
- If forward kernel K_n only alters the recent components of x_{n-1} and adds new jumps/parameters in $E_n \setminus E_{n-1}$, online operation is possible

$$p(d\zeta_{t_n} | y_{1:n}) \approx \sum_{i=1}^{N} W_n^{(i)} \delta_{F(t_n - \tau_{k_n}^{(i)}, \theta_{k_n}^{(i)})}(d\zeta_{t_n})$$

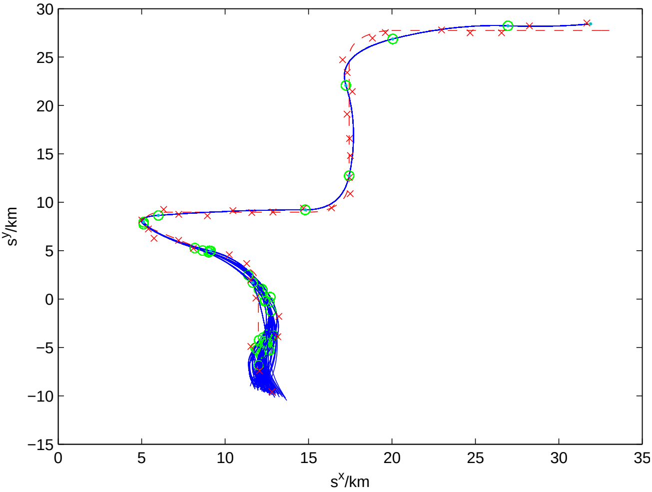
- A mixture proposal

$$K_n(x_{n-1}, x_n) = \sum_m \alpha_{n,m}(x_{n-1}) K_{n,m}(x_{n-1}, x_n),$$

SMC Filtering

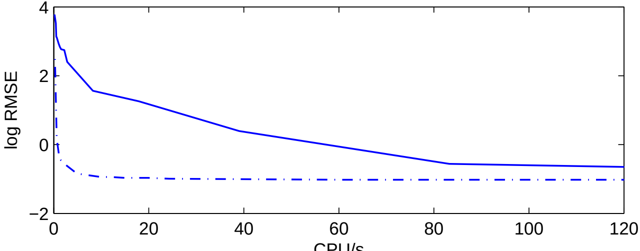
- When K_n corresponds to extending x_{n-1} into E_n by sampling from the prior, obtain the algorithm of (Godsill et al., 2007).
- This is inefficient as involves propagating multiple copies of particles after resampling
- A more efficient strategy is to propose births and to perturb the most recent jump time/parameter, (τ_k, θ_k)
- To minimize the variance the importance weights, we would like to draw from $\eta_n(\tau_k, \theta_k | x_{n-1} \setminus (\tau_k, \theta_k))$, or sensible approximations thereof.

Filtering



N	Godsill et al. 2007 RMSE / km	Godsill et al. 2007 CPU / s	Whiteley et al. 2007 RMSE / km	Whiteley et al. 2007 CPU / s
50	42.62	0.24	0.88	1.32
100	33.49	0.49	0.66	2.62
250	22.89	1.23	0.54	6.56
500	17.26	2.42	0.51	12.98
1000	12.68	5.00	0.50	26.07
2500	6.18	13.20	0.49	67.32
5000	3.52	28.79	0.48	142.84

RMSE and CPU time (200 runs).



Convergence

- This framework allows us to analyse algorithm of Godsill et al. 2007
- $\mu_n(\varphi) := \int \varphi(\zeta_{t_n}) p(d\zeta_{t_n} | y_{1:n})$ and $\mu_n^N(\varphi)$ the corresponding SMC approximation
- Under standard regularity conditions

$$\sqrt{N}(\mu_n^N(\varphi) - \mu_n(\varphi)) \Rightarrow \mathcal{N}(0, \sigma_n^2(\varphi))$$

- Under rather strong assumptions*

$$\mathbb{E}\left[|\mu_n^N(\varphi) - \mu_n(\varphi)|^p\right]^{1/p} \leq \frac{c_p(\varphi)}{\sqrt{N}}$$

*which include: $(\zeta_{t_n})_{n\in\mathbb{N}}$ is uniformly ergodic Markov, likelihood bounded above and away from zero uniformly in time

Conclusion

In Conclusion

- Monte Carlo Methods have uses beyond the calculation of posterior means.
- SMC provides a viable alternative to MCMC.
- SMC is effective at:
  - ML and MAP estimation;
  - rare event estimation;
  - filtering outside the standard particle filtering framework.
  - ...
  - Other published applications include: approximate Bayesian computation, Bayesian estimation in GLMMs, options pricing and estimation in partially observed marked point processes, filtering of diffusions, air traffic control, optimal design.

References I

[1] N. Chopin. A sequential particle filter method for static models. *Biometrika*, 89(3): 539–551, 2002.

[2] P. Del Moral, A. Doucet, and A. Jasra. Sequential Monte Carlo methods for Bayesian Computation. In *Bayesian Statistics 8*. Oxford University Press, 2006.

[3] P. Del Moral, A. Doucet, and A. Jasra. Sequential Monte Carlo samplers. *Journal of the Royal Statistical Society B*, 63(3):411–436, 2006.

[4] A. Doucet and A. M. Johansen. A tutorial on particle filtering and smoothing: Fiteen years later. In D. Crisan and B. Rozovsky, editors, *The Oxford Handbook of Nonlinear Filtering*. Oxford University Press, 2010. To appear.

[5] A. Doucet, S. J. Godsill, and C. P. Robert. Marginal maximum a posteriori estimation using Markov chain Monte Carlo. *Statistics and Computing*, 12:77–84, 2002.

[6] Y. Fan, D. Leslie, and M. P. Wand. Generalized linear mixed model analysis via sequential Monte Carlo sampling. *Electronic Journal of Statistics*, 2:916–938, 2008.

[7] S. J. Godsill, J. Vermaak, K.-F. Ng, and J.-F. Li. Models and algorithms for tracking of manoeuvring objects using variable rate particle filters. *Proceedings of IEEE*, 95(5): 925–952, 2007.

[8] C.-R. Hwang. Laplace's method revisited: Weak convergence of probability measures. *The Annals of Probability*, 8(6):1177–1182, December 1980.

[9] A. M. Johansen. SMCTC: Sequential Monte Carlo in C++. *Journal of Statistical Software*, 30(6):1–41, April 2009.

[10] A. M. Johansen, P. Del Moral, and A. Doucet. Sequential Monte Carlo samplers for rare events. In *Proceedings of the 6th International Workshop on Rare Event Simulation*, pages 256–267, Bamberg, Germany, October 2006.

References II

[11] A. M. Johansen, A. Doucet, and M. Davy. Particle methods for maximum likelihood parameter estimation in latent variable models. *Statistics and Computing*, 18(1):47–57, March 2008.

[12] N. Whiteley, A. M. Johansen, and S. Godsill. Efficient Monte Carlo filtering for discretely observed jumping processes. In *Proceedings of IEEE Statistical Signal Processing Workshop*, pages 89–93, Madison, WI, USA, August 26th–29th 2007. IEEE.

[13] N. Whiteley, A. M. Johansen, and S. Godsill. Monte Carlo filtering of piecewise-deterministic processes. *Journal of Computational and Graphical Statistics*, 2010. To appear.

Path Sampling Identity

Given a probability density, $p(x|\theta) = q(x|\theta)/z(\theta)$:

$$\begin{aligned}
\frac{\partial}{\partial\theta}\log z(\theta) =& \frac{1}{z(\theta)}\frac{\partial}{\partial\theta}z(\theta) \\
=& \frac{1}{z(\theta)}\frac{\partial}{\partial\theta}\int q(x|\theta)dx \\
=& \int \frac{1}{z(\theta)}\frac{\partial}{\partial\theta}q(x|\theta)dx \qquad (\star\star) \\
=& \int \frac{p(x|\theta)}{q(x|\theta)}\frac{\partial}{\partial\theta}q(x|\theta)dx \\
=& \int p(x|\theta)\frac{\partial}{\partial\theta}\log q(x|\theta)dx = \mathbb{E}_{p(\cdot|\theta)}\left[\frac{\partial}{\partial\theta}\log q(x|\theta)\right]
\end{aligned}$$

wherever $\star\star$ is permissible. Back to $\star$.