Particle Markov Chain Monte Carlo

Arnaud Doucet
Institute of Statistical Mathematics
Tokyo, Japan

Objectives

- Two main classes of Monte Carlo methods are used nowadays for Bayesian computation: **Markov chain Monte Carlo** (MCMC) and **Sequential Monte Carlo** (SMC) aka Particle filters.

Objectives

- Two main classes of Monte Carlo methods are used nowadays for Bayesian computation: **Markov chain Monte Carlo** (MCMC) and **Sequential Monte Carlo** (SMC) aka Particle filters.
- It seems natural to combine MCMC and SMC.

Objectives

- Two main classes of Monte Carlo methods are used nowadays for Bayesian computation: **Markov chain Monte Carlo** (MCMC) and **Sequential Monte Carlo** (SMC) aka Particle filters.
- It seems natural to combine MCMC and SMC.
- We know how to use MCMC within SMC (Gilks & Berzuini, JRSS B, 2001).

Objectives

- Two main classes of Monte Carlo methods are used nowadays for Bayesian computation: **Markov chain Monte Carlo** (MCMC) and **Sequential Monte Carlo** (SMC) aka Particle filters.
- It seems natural to combine MCMC and SMC.
- We know how to use MCMC within SMC (Gilks & Berzuini, JRSS B, 2001).
- We show here how to use SMC within MCMC in a simple and principled way.

Basics of Markov chain Monte Carlo

- Assume you want to sample from an high dimensional probability density $\pi(z)$ where $z \in \mathcal{Z}$.

Basics of Markov chain Monte Carlo

- Assume you want to sample from an high dimensional probability density $\pi(z)$ where $z \in \mathcal{Z}$.
- Markov chain Monte Carlo (MCMC) basic idea: Build a Markov chain $\{Z(i)\}_{i \geq 1}$ such that

$$\mathcal{L}(Z(i)) \underset{i \to \infty}{\rightarrow} \pi(z) \text{ and } \frac{1}{L} \sum_{i=1}^{L} \varphi(Z(i)) \underset{L \to \infty}{\rightarrow} \int \varphi(z) \pi(z) \, dz$$

Basics of Markov chain Monte Carlo

- Assume you want to sample from an high dimensional probability density $\pi(z)$ where $z \in \mathcal{Z}$.
- Markov chain Monte Carlo (MCMC) basic idea: Build a Markov chain $\{Z(i)\}_{i\geq 1}$ such that

$$\mathcal{L}(Z(i)) \underset{i\to\infty}{\rightarrow} \pi(z) \text{ and } \frac{1}{L}\sum_{i=1}^{L}\varphi(Z(i)) \underset{L\to\infty}{\rightarrow} \int \varphi(z)\,\pi(z)\,dz$$

- Most famous MCMC method: the Metropolis-Hastings algorithm.

Metropolis-Hastings Algorithm

- Introduce a proposal density proposing a candidate z^* when the chain is in z : $q\left(z^* \middle| z \right)$.

At iteration i; $i \geq 1$

- Introduce a proposal density proposing a candidate z^* when the chain is in z : $q\left(z^* \middle| z\right)$.
- If $q\left(z^* \middle| z\right) = q\left(z^*\right)$, this is an independent proposal.

At iteration i; $i \geq 1$

Metropolis-Hastings Algorithm

- Introduce a proposal density proposing a candidate z^* when the chain is in z : $q\left(z^* \middle| z\right)$.
- If $q\left(z^* \middle| z\right) = q\left(z^*\right)$, this is an independent proposal.

<u>*At iteration i; $i \geq 1$*</u>

- Sample $Z^* \sim q\left(\cdot \middle| Z\left(i-1\right)\right)$.

Metropolis-Hastings Algorithm

- Introduce a proposal density proposing a candidate z^* when the chain is in z : $q(z^* | z)$.
- If $q(z^* | z) = q(z^*)$, this is an independent proposal.

At iteration i; $i \geq 1$

- Sample $Z^* \sim q(\cdot | Z(i-1))$.
- With probability

$$1 \wedge \frac{\pi(Z^*)}{\pi(Z(i-1))} \frac{q(Z(i-1) | Z^*)}{q(Z^* | Z(i-1))}$$

set $Z(i) = Z^*$ otherwise set $Z(i) = Z(i-1)$.

Metropolis-Hastings Algorithm

- Introduce a proposal density proposing a candidate z^* when the chain is in z : $q\left(z^* \mid z\right)$.
- If $q\left(z^* \mid z\right) = q\left(z^*\right)$, this is an independent proposal.

<u>*At iteration i; $i \geq 1$*</u>

- Sample $Z^* \sim q\left(\cdot \mid Z\left(i-1\right)\right)$.
- With probability

$$1 \wedge \frac{\pi\left(Z^*\right)}{\pi\left(Z\left(i-1\right)\right)} \frac{q\left(Z\left(i-1\right) \mid Z^*\right)}{q\left(Z^* \mid Z\left(i-1\right)\right)}$$

  set $Z\left(i\right) = Z^*$ otherwise set $Z\left(i\right) = Z\left(i-1\right)$.
- The Metropolis-Hastings works 'theoretically' under very weak assumptions. In practice the choice of $q\left(z^* \mid z\right)$ is absolutely crucial.

General State-Space Models

- Let $\{X_k\}_{k\geq 1}$ be a $\mathcal{X}-$valued Markov process defined by

$$X_1 \sim \mu(\cdot) \text{ and } X_k | (X_{k-1} = x_{k-1}) \sim f(\cdot| x_{k-1}).$$

General State-Space Models

- Let $\{X_k\}_{k\geq 1}$ be a $\mathcal{X}-$valued Markov process defined by

$$X_1 \sim \mu(\cdot) \text{ and } X_k | (X_{k-1} = x_{k-1}) \sim f(\cdot | x_{k-1}).$$

- We only have access to a process $\{Y_k\}_{k\geq 1}$ such that, conditional upon $\{X_k\}_{k\geq 1}$, the observations are statistically independent and

$$Y_k | (X_k = x_k) \sim g(\cdot | x_k).$$

Bayesian Inference in General State-Space Models

- Given a collection of observations $y_{1:T} := (y_1, ..., y_T)$, we are interested in inference about $X_{1:T} := (X_1, ..., X_T)$.

Bayesian Inference in General State-Space Models

- Given a collection of observations $y_{1:T} := (y_1, ..., y_T)$, we are interested in inference about $X_{1:T} := (X_1, ..., X_T)$.
- In this Bayesian framework, inference relies on the posterior density

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

where

$$p(x_{1:T}, y_{1:T}) = \underbrace{\mu(x_1) \prod_{k=2}^{T} f(x_k | x_{k-1})}_{\text{prior}} \underbrace{\prod_{k=1}^{T} g(y_k | x_k)}_{\text{likelihood}}$$

Bayesian Inference in General State-Space Models

- Given a collection of observations $y_{1:T} := (y_1, ..., y_T)$, we are interested in inference about $X_{1:T} := (X_1, ..., X_T)$.
- In this Bayesian framework, inference relies on the posterior density

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

where

$$p(x_{1:T}, y_{1:T}) = \underbrace{\mu(x_1) \prod_{k=2}^{T} f(x_k | x_{k-1})}_{\text{prior}} \underbrace{\prod_{k=1}^{T} g(y_k | x_k)}_{\text{likelihood}}$$

- Except for a few models including finite state-space HMM and linear Gaussian models (Kalman), this posterior density does not admit a standard form.

Standard MCMC Approaches

- We sample an ergodic Markov chain $\{X_{1:T}(i)\}$ of invariant density $p(x_{1:T} | y_{1:T})$ using a Metropolis-Hastings (MH) algorithm; e.g. an *independent* MH sampler where $q(x^*_{1:T} | x_{1:T}) = q(x^*_{1:T})$.

At iteration i; $i \geq 1$

Standard MCMC Approaches

- We sample an ergodic Markov chain $\{X_{1:T}(i)\}$ of invariant density $p(x_{1:T}|\, y_{1:T})$ using a Metropolis-Hastings (MH) algorithm; e.g. an *independent* MH sampler where $q(x^*_{1:T}|\, x_{1:T}) = q(x^*_{1:T})$.

At iteration i; $i \geq 1$

- Sample $X^*_{1:T} \sim q(\cdot)$.

Standard MCMC Approaches

- We sample an ergodic Markov chain $\{X_{1:T}(i)\}$ of invariant density $p(x_{1:T} | y_{1:T})$ using a Metropolis-Hastings (MH) algorithm; e.g. an *independent* MH sampler where $q(x^*_{1:T} | x_{1:T}) = q(x^*_{1:T})$.

<u>*At iteration i; $i \geq 1$*</u>

- Sample $X^*_{1:T} \sim q(\cdot)$.
- With probability

$$1 \wedge \frac{p(X^*_{1:T} | y_{1:T})}{p(X_{1:T}(i-1) | y_{1:T})} \frac{q(X_{1:T}(i-1))}{q(X^*_{1:T})}$$

set $X_{1:T}(i) = X^*_{1:T}$, otherwise set $X_{1:T}(i) = X_{1:T}(i-1)$.

Standard MCMC Approaches

- We sample an ergodic Markov chain $\{X_{1:T}(i)\}$ of invariant density $p(x_{1:T}|\, y_{1:T})$ using a Metropolis-Hastings (MH) algorithm; e.g. an *independent* MH sampler where $q(x^*_{1:T}|\, x_{1:T}) = q(x^*_{1:T})$.

<u>*At iteration i; $i \geq 1$*</u>

- Sample $X^*_{1:T} \sim q(\cdot)$.
- With probability

$$1 \wedge \frac{p(X^*_{1:T}|\, y_{1:T})}{p(X_{1:T}(i-1)|\, y_{1:T})} \frac{q(X_{1:T}(i-1))}{q(X^*_{1:T})}$$

  set $X_{1:T}(i) = X^*_{1:T}$, otherwise set $X_{1:T}(i) = X_{1:T}(i-1)$.
- Very simple but very inefficient.

Standard MCMC Approaches

- We sample an ergodic Markov chain $\{X_{1:T}(i)\}$ of invariant density $p(x_{1:T}|y_{1:T})$ using a Metropolis-Hastings (MH) algorithm; e.g. an *independent* MH sampler where $q(x^*_{1:T}|x_{1:T}) = q(x^*_{1:T})$.

<u>*At iteration i; $i \geq 1$*</u>

- Sample $X^*_{1:T} \sim q(\cdot)$.
- With probability

$$1 \wedge \frac{p(X^*_{1:T}|y_{1:T})}{p(X_{1:T}(i-1)|y_{1:T})} \frac{q(X_{1:T}(i-1))}{q(X^*_{1:T})}$$

set $X_{1:T}(i) = X^*_{1:T}$, otherwise set $X_{1:T}(i) = X_{1:T}(i-1)$.

- Very simple but very inefficient.
- For good performance, we need to select $q(x_{1:T}) \approx p(x_{1:T}|y_{1:T})$ but for highly nonlinear non-Gaussian models it is essentially *impossible* as soon as T is large.

Common Approaches and Limitations

- Standard practice consists of building an MCMC kernel updating each component X_k individually: this typically leads to *slow mixing algorithms*.

Common Approaches and Limitations

- Standard practice consists of building an MCMC kernel updating each component X_k individually: this typically leads to *slow mixing algorithms*.
- Many complex models (e.g. some Lévy-driven volatility models) are such that it is only possible to sample from the prior but impossible to evaluate it pointwise: how could you use MCMC for such models?

Sequential Monte Carlo aka Particle Filters

- SMC methods provide an alternative way to approximate $p\left(x_{1:T} \middle| y_{1:T}\right)$ and to compute $p\left(y_{1:T}\right)$.

Sequential Monte Carlo aka Particle Filters

- SMC methods provide an alternative way to approximate $p(x_{1:T} | y_{1:T})$ and to compute $p(y_{1:T})$.
- To sample from $p(x_{1:T} | y_{1:T})$, SMC proceeds sequentially by first approximating $p(x_1 | y_1)$ at time 1 then $p(x_{1:2} | y_{1:2})$ at time 2 and so on.

Sequential Monte Carlo aka Particle Filters

- SMC methods provide an alternative way to approximate $p\left(x_{1:T} \middle| y_{1:T}\right)$ and to compute $p\left(y_{1:T}\right)$.
- To sample from $p\left(x_{1:T} \middle| y_{1:T}\right)$, SMC proceeds sequentially by first approximating $p\left(x_1 \middle| y_1\right)$ at time 1 then $p\left(x_{1:2} \middle| y_{1:2}\right)$ at time 2 and so on.
- SMC methods approximate the distributions of interest via a cloud of N particles which are propagated using *Importance Sampling* and *Resampling* steps.

Vanilla SMC - The Bootstrap Filter

At time 1

- Sample $X_1^{(i)} \sim \mu(\cdot)$ and compute $W_1^{(i)} \propto g\left(y_1 \middle| X_1^{(i)}\right)$, $\sum_{i=1}^{N} W_1^{(i)} = 1$.

At time k, $k > 1$

Vanilla SMC - The Bootstrap Filter

At time 1

- Sample $X_1^{(i)} \sim \mu(\cdot)$ and compute $W_1^{(i)} \propto g\left(y_1 | X_1^{(i)}\right)$, $\sum_{i=1}^{N} W_1^{(i)} = 1$.
- Resample N times from $\widehat{p}(x_1 | y_1) = \sum_{i=1}^{N} W_1^{(i)} \delta_{X_1^{(i)}}(x_1)$.

At time k, $k > 1$

Vanilla SMC - The Bootstrap Filter

At time 1

- Sample $X_1^{(i)} \sim \mu(\cdot)$ and compute $W_1^{(i)} \propto g\left(y_1 \middle| X_1^{(i)} \right)$, $\sum_{i=1}^N W_1^{(i)} = 1$.
- Resample N times from $\widehat{p}(x_1 | y_1) = \sum_{i=1}^N W_1^{(i)} \delta_{X_1^{(i)}}(x_1)$.

At time k, $k > 1$

- Sample $X_k^{(i)} \sim f\left(\cdot \middle| X_{k-1}^{(i)} \right)$, set $X_{1:k}^{(i)} = \left(X_{1:k-1}^{(i)}, X_k^{(i)} \right)$ and compute $W_k^{(i)} \propto g\left(y_k \middle| X_k^{(i)} \right)$, $\sum_{i=1}^N W_k^{(i)} = 1$.

Vanilla SMC - The Bootstrap Filter

At time 1

- Sample $X_1^{(i)} \sim \mu(\cdot)$ and compute $W_1^{(i)} \propto g\left(y_1 | X_1^{(i)}\right)$, $\sum_{i=1}^N W_1^{(i)} = 1$.
- Resample N times from $\widehat{p}(x_1 | y_1) = \sum_{i=1}^N W_1^{(i)} \delta_{X_1^{(i)}}(x_1)$.

At time k, $k > 1$

- Sample $X_k^{(i)} \sim f\left(\cdot | X_{k-1}^{(i)}\right)$, set $X_{1:k}^{(i)} = \left(X_{1:k-1}^{(i)}, X_k^{(i)}\right)$ and compute $W_k^{(i)} \propto g\left(y_k | X_k^{(i)}\right)$, $\sum_{i=1}^N W_k^{(i)} = 1$.
- Resample N times from $\widehat{p}(x_{1:k} | y_{1:k}) = \sum_{i=1}^N W_k^{(i)} \delta_{X_{1:k}^{(i)}}(x_{1:k})$.

SMC Output

- At time T, we obtain the following approximation of the posterior of interest

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

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

$$\widehat{p}\left(y_{1:T}\right) = \widehat{p}\left(y_1\right) \prod_{k=2}^{T} \widehat{p}\left(y_k \middle| y_{1:k-1}\right) = \prod_{k=1}^{T} \left(\frac{1}{N} \sum_{i=1}^{N} g\left(y_k \middle| X_k^{(i)}\right) \right).$$

SMC Output

- At time T, we obtain the following approximation of the posterior of interest

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

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

$$\widehat{p}\left(y_{1:T}\right)=\widehat{p}\left(y_{1}\right)\prod_{k=2}^{T}\widehat{p}\left(y_{k}\,|\,y_{1:k-1}\right)=\prod_{k=1}^{T}\left(\frac{1}{N}\sum_{i=1}^{N}g\left(y_{k}\,|\,X_{k}^{(i)}\right)\right).$$

- These approximations are asymptotically (i.e. $N\rightarrow\infty$) consistent under very weak assumptions.

Some Theoretical Results

- Under *mixing assumptions* (Del Moral, 2004), we have

$$\left\| \mathcal{L}\left(X_{1:T} \in \cdot\right) - p\left(\cdot \mid y_{1:T}\right) \right\| \leq C\frac{T}{N}$$

where $X_{1:T} \sim \widehat{p}\left(\cdot \mid y_{1:T}\right)$.

Some Theoretical Results

- Under *mixing assumptions* (Del Moral, 2004), we have

$$\left\| \mathcal{L}\left(X_{1:T} \in \cdot\right) - p\left(\cdot\middle|\, y_{1:T}\right)\right\| \leq C\frac{T}{N}$$

where $X_{1:T} \sim \widehat{p}\left(\cdot\middle|\, y_{1:T}\right)$.

- Under *mixing assumptions* (Del Moral & D., 2004) we also have

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

Some Theoretical Results

- Under *mixing assumptions* (Del Moral, 2004), we have

$$\left\| \mathcal{L}\left(X_{1:T} \in \cdot\right) - p\left(\cdot \mid y_{1:T}\right) \right\| \leq C\frac{T}{N}$$

where $X_{1:T} \sim \widehat{p}\left(\cdot \mid y_{1:T}\right)$.

- Under *mixing assumptions* (Del Moral & D., 2004) we also have

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

- Loosely speaking, the performance of SMC only degrade linearly with time instead of exponentially for naive approaches.

- 'Idea': Use the output of our SMC algorithm to define 'good' high-dimensional proposal distributions for MCMC.

Combining MCMC and SMC

- 'Idea': Use the output of our SMC algorithm to define 'good' high-dimensional proposal distributions for MCMC.
- **Problem**: Consider $X_{1:T} \sim \widehat{p}\left(\cdot \middle| y_{1:T}\right)$ then the unconditional distribution is

$$q\left(x_{1:T}\right) = \mathbb{E}\left(\widehat{p}\left(x_{1:T} \middle| y_{1:T}\right)\right)$$

  which does not admit an analytic expression and so we cannot use directly the MH algorithm to correct for the discrepancy between the proposal and the target.

Combining MCMC and SMC

- 'Idea': Use the output of our SMC algorithm to define 'good' high-dimensional proposal distributions for MCMC.
- **Problem**: Consider $X_{1:T} \sim \widehat{p}\left(\cdot \mid y_{1:T} \right)$ then the unconditional distribution is
$$q\left(x_{1:T} \right) = \mathbb{E}\left(\widehat{p}\left(x_{1:T} \mid y_{1:T} \right) \right)$$
which does not admit an analytic expression and so we cannot use directly the MH algorithm to correct for the discrepancy between the proposal and the target.
- Hence the need for a new methodology....

Particle MH Sampler

At iteration 1

- Run an SMC algorithm to obtain $\widehat{p}^{(1)}(x_{1:T} | y_{1:T})$ and $\widehat{p}^{(1)}(y_{1:T})$.

At iteration i; $i \geq 2$

Particle MH Sampler

<u>*At iteration* 1</u>

- Run an SMC algorithm to obtain $\widehat{p}^{(1)}\left(x_{1:T} \middle| y_{1:T} \right)$ and $\widehat{p}^{(1)}\left(y_{1:T} \right)$.
- Sample $X_{1:T}\left(1\right) \sim \widehat{p}^{(1)}\left(\cdot \middle| y_{1:T} \right)$.

<u>*At iteration* i; $i \geq 2$</u>

Particle MH Sampler

At iteration 1

- Run an SMC algorithm to obtain $\widehat{p}^{(1)}(x_{1:T} | y_{1:T})$ and $\widehat{p}^{(1)}(y_{1:T})$.
- Sample $X_{1:T}(1) \sim \widehat{p}^{(1)}(\cdot | y_{1:T})$.

At iteration i; $i \geq 2$

- Run an SMC algorithm to obtain $\widehat{p}^{*}(x_{1:T} | y_{1:T})$ and $\widehat{p}^{*}(y_{1:T})$.

Particle MH Sampler

At iteration 1

- Run an SMC algorithm to obtain $\widehat{p}^{(1)}\left(x_{1:T} \middle| y_{1:T}\right)$ and $\widehat{p}^{(1)}\left(y_{1:T}\right)$.
- Sample $X_{1:T}(1) \sim \widehat{p}^{(1)}\left(\cdot \middle| y_{1:T}\right)$.

At iteration i; $i \geq 2$

- Run an SMC algorithm to obtain $\widehat{p}^{*}\left(x_{1:T} \middle| y_{1:T}\right)$ and $\widehat{p}^{*}\left(y_{1:T}\right)$.
- Sample $X_{1:T}^{*} \sim \widehat{p}^{*}\left(\cdot \middle| y_{1:T}\right)$.

Particle MH Sampler

At iteration 1

- Run an SMC algorithm to obtain $\widehat{p}^{(1)}\left(x_{1:T}\middle|\,y_{1:T}\right)$ and $\widehat{p}^{(1)}\left(y_{1:T}\right)$.
- Sample $X_{1:T}\left(1\right)\sim\widehat{p}^{(1)}\left(\cdot\middle|\,y_{1:T}\right)$.

At iteration i; $i\geq 2$

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

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

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

Main Result

- **Proposition**. For any $N \geq 1$ the PMH sampler is an independent MH sampler defined on the extended space $\{1, \ldots, N\} \times \mathcal{X}^{TN} \times \{1, \ldots, N\}^{(T-1)N+1}$ with a target density

$$\widetilde{p}\left(k, x_1^{1:N}, ..., x_T^{1:N}, i_1^{1:N}, \ldots, i_{T-1}^{1:N}\right)$$

which admits a conditional distribution for $X_{1:T}^K = (X_1^{I_1^K}, X_2^{I_2^K}, \ldots, X_{T-1}^{I_{T-1}^K}, X_T^K)$ equal to $p(x_{1:T} | y_{1:T})$. $\left\{I_1^K, I_2^K, ..., I_{T-1}^K\right\}$ corresponds to the ancestral lineage of the path $X_{1:T}^K$.

Main Result

- **Proposition**. For any $N \geq 1$ the PMH sampler is an independent MH sampler defined on the extended space $\{1,\ldots,N\} \times \mathcal{X}^{TN} \times \{1,\ldots,N\}^{(T-1)N+1}$ with a target density
$$\widetilde{p}\left(k, x_1^{1:N}, ..., x_T^{1:N}, i_1^{1:N}, \ldots, i_{T-1}^{1:N}\right)$$
which admits a conditional distribution for $X_{1:T}^K = (X_1^{I_1^K}, X_2^{I_2^K}, \ldots, X_{T-1}^{I_{T-1}^K}, X_T^K)$ equal to $p\left(x_{1:T} \middle| y_{1:T}\right)$. $\left\{I_1^K, I_2^K, ..., I_{T-1}^K\right\}$ corresponds to the ancestral lineage of the path $X_{1:T}^K$.
- **Proposition**. For any $N \geq 1$ we have
$$\left\|\mathcal{L}\left(X_{1:T}\left(i\right) \in \cdot\right) - p\left(\cdot \middle| y_{1:T}\right)\right\| \rightarrow 0 \text{ as } i \rightarrow \infty.$$
In addition, if $\sup_{x,y} \left|g\left(y \middle| x\right)\right| < C$, there exists $\rho \in [0,1)$ such that for any $i \geq 1$ and $N \geq 1$,
$$\left\|\mathcal{L}\left(X_{1:T}\left(i\right) \in \cdot\right) - p\left(\cdot \middle| y_{1:T}\right)\right\| \leq \rho^i \ .$$

Structure of the Artificial Extended Target Distribution

- How to sample from

$$\widetilde{p}\left(k, x_1^{1:N}, ..., x_T^{1:N}, i_1^{1:N}, \ldots, i_{T-1}^{1:N}\right)$$

Structure of the Artificial Extended Target Distribution

- How to sample from

$$\widetilde{p}\left(k, x_1^{1:N}, ..., x_T^{1:N}, i_1^{1:N}, \ldots, i_{T-1}^{1:N}\right)$$

- Sample $\left(I_1^K, I_2^K, ..., I_{T-1}^K, K\right)$ from a uniform distribution on $\{1, ..., N\}^T$.

Structure of the Artificial Extended Target Distribution

- How to sample from

$$\widetilde{p}\left(k, x_1^{1:N}, ..., x_T^{1:N}, i_1^{1:N}, \ldots, i_{T-1}^{1:N}\right)$$

- Sample $\left(I_1^K, I_2^K, ..., I_{T-1}^K, K\right)$ from a uniform distribution on $\{1, ..., N\}^T$.
- Sample $X_{1:T}^K = (X_1^{I_1^K}, X_2^{I_2^K}, \ldots, X_{T-1}^{I_{T-1}^K}, X_T^K)$ from $p\left(x_{1:T} \mid y_{1:T}\right)$. (We do not know how to do this, this is why we use MH).

Structure of the Artificial Extended Target Distribution

- How to sample from

$$\widetilde{p}\left(k, x_1^{1:N}, ..., x_T^{1:N}, i_1^{1:N}, \ldots, i_{T-1}^{1:N}\right)$$

- Sample $\left(I_1^K, I_2^K, ..., I_{T-1}^K, K\right)$ from a uniform distribution on $\{1, ..., N\}^T$.
- Sample $X_{1:T}^K = (X_1^{I_1^K}, X_2^{I_2^K}, \ldots, X_{T-1}^{I_{T-1}^K}, X_T^K)$ from $p\left(x_{1:T} \mid y_{1:T}\right)$. (We do not know how to do this, this is why we use MH).
- Run a conditional SMC algorithm compatible with $X_{1:T}^K$ and its ancestral lineage $\left(I_1^K, I_2^K, ..., I_{T-1}^K, K\right)$.

Conditional SMC Algorithm

At time 1

- For $i \neq I_1^K$, sample $X_1^{(i)} \sim \mu(\cdot)$ and compute $W_1^{(i)} \propto g\left(y_1 \middle| X_1^{(i)}\right)$, $\sum_{i=1}^N W_1^{(i)} = 1$.

Conditional SMC Algorithm

At time 1

- For $i \neq I_1^K$, sample $X_1^{(i)} \sim \mu(\cdot)$ and compute $W_1^{(i)} \propto g\left(y_1 \middle| X_1^{(i)}\right)$, $\sum_{i=1}^N W_1^{(i)} = 1$.
- Sample $N_1 \sim \mathcal{B}^+\left(N, W_1^{\left(I_1^K\right)}\right)$ and assign $N_1 - 1$ additional offspring to $X_1^{\left(I_1^K\right)}$.

Conditional SMC Algorithm

<u>*At time* 1</u>

- For $i \neq I_1^K$, sample $X_1^{(i)} \sim \mu(\cdot)$ and compute $W_1^{(i)} \propto g\left(y_1 | X_1^{(i)}\right)$, $\sum_{i=1}^{N} W_1^{(i)} = 1$.
- Sample $N_1 \sim \mathcal{B}^+\left(N, W_1^{\left(I_1^K\right)}\right)$ and assign $N_1 - 1$ additional offspring to $X_1^{\left(I_1^K\right)}$.
- Resample $N - N_1$ times from $\widehat{p}(x_1 | y_1) = \sum_{i=1, i \neq I_1^K}^{N} \widetilde{W}_1^{(i)} \delta_{X_1^{(i)}}(x_1)$ where $\widetilde{W}_1^{(i)} \propto W_1^{(i)}$

At time n, $n > 1$

- For $i \neq I_n^K$, sample $X_n^{(i)} \sim f\left(\cdot | X_{n-1}^{(i)}\right)$, set $X_{1:n}^{(i)} = \left(X_{1:n-1}^{(i)}, X_n^{(i)}\right)$ and compute $W_n^{(i)} \propto g\left(y_n | X_n^{(i)}\right)$, $\sum_{i=1}^N W_n^{(i)} = 1$.

At time n, $n > 1$

- For $i \neq I_n^K$, sample $X_n^{(i)} \sim f\left(\cdot \mid X_{n-1}^{(i)}\right)$, set $X_{1:n}^{(i)} = \left(X_{1:n-1}^{(i)}, X_n^{(i)}\right)$ and compute $W_n^{(i)} \propto g\left(y_n \mid X_n^{(i)}\right)$, $\sum_{i=1}^{N} W_n^{(i)} = 1$.
- Sample $N_1 \sim \mathcal{B}^+\left(N, W_n^{\left(I_n^K\right)}\right)$ and assign $N_1 - 1$ additional offspring to $X_n^{\left(I_n^K\right)}$.

At time n, $n > 1$

- For $i \neq I_n^K$, sample $X_n^{(i)} \sim f\left(\cdot | \, X_{n-1}^{(i)} \right)$, set $X_{1:n}^{(i)} = \left(X_{1:n-1}^{(i)}, X_n^{(i)} \right)$ and compute $W_n^{(i)} \propto g\left(y_n | \, X_n^{(i)} \right)$, $\sum_{i=1}^{N} W_n^{(i)} = 1$.
- Sample $N_1 \sim \mathcal{B}^+\left(N, W_n^{\left(I_n^K \right)} \right)$ and assign $N_1 - 1$ additional offspring to $X_n^{\left(I_n^K \right)}$.
- Resample $N - N_1$ times from $\widehat{p}\left(x_{1:k} | \, y_{1:k} \right) = \sum_{i=1, i \neq I_n^K}^{N} \widetilde{W}_n^{(i)} \delta_{X_{1:n}^{(i)}}\left(x_1 \right)$ where $\widetilde{W}_n^{(i)} \propto W_n^{(i)}$.

Waste Recycling

- The 'standard' estimate of $\int f\left(x_{1:T}\right) p\left(\left.x_{1:T}\right| y_{1:T}\right) dx_{1:T}$ for L MCMC iterations is

$$\frac{1}{L}\sum_{i=1}^{L} f\left(X_{1:T}\left(i\right)\right).$$

Waste Recycling

- The 'standard' estimate of $\int f(x_{1:T}) \, p(x_{1:T} | y_{1:T}) \, dx_{1:T}$ for L MCMC iterations is
$$\frac{1}{L} \sum_{i=1}^{L} f(X_{1:T}(i)).$$
- We generate N particles at each iteration i of the MCMC algorithm to decide whether to accept or reject one single candidate. This seems terribly wasteful.

Waste Recycling

- The 'standard' estimate of $\int f(x_{1:T}) \, p(x_{1:T} | y_{1:T}) \, dx_{1:T}$ for L MCMC iterations is
$$\frac{1}{L}\sum_{i=1}^{L} f(X_{1:T}(i)).$$
- We generate N particles at each iteration i of the MCMC algorithm to decide whether to accept or reject one single candidate. This seems terribly wasteful.
- It is possible to use all the particles by considering the estimate
$$\frac{1}{L}\sum_{i=1}^{L}\left(\sum_{k=1}^{N} W_T^{(k)}(i) \, f(X_{1:T}^{(k)}(i))\right).$$

Waste Recycling

- The 'standard' estimate of $\int f\left(x_{1:T}\right) p\left(x_{1:T} \middle| y_{1:T}\right) dx_{1:T}$ for L MCMC iterations is
$$\frac{1}{L}\sum_{i=1}^{L} f(X_{1:T}(i)).$$
- We generate N particles at each iteration i of the MCMC algorithm to decide whether to accept or reject one single candidate. This seems terribly wasteful.
- It is possible to use all the particles by considering the estimate
$$\frac{1}{L}\sum_{i=1}^{L}\left(\sum_{k=1}^{N} W_T^{(k)}(i)\, f(X_{1:T}^{(k)}(i))\right).$$
- We can even recycle the populations of particles rejected.

Nonlinear State-Space Model

- Consider the following model

$$\begin{aligned} X_k &= \frac{1}{2}X_{k-1} + 25\frac{X_{k-1}}{1+X_{k-1}^2} + 8\cos 1.2k + V_k, \\ Y_k &= \frac{X_k^2}{20} + W_k \end{aligned}$$

where $V_k \sim \mathcal{N}\left(0, \sigma_v^2\right)$, $W_k \sim \mathcal{N}\left(0, \sigma_w^2\right)$ and $X_1 \sim \mathcal{N}\left(0, 5^2\right)$.

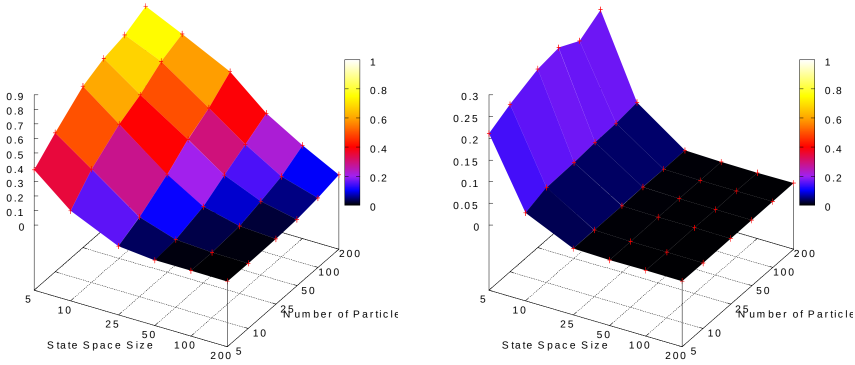
Nonlinear State-Space Model

- Consider the following model

$$
\begin{aligned}
X_k &= \frac{1}{2}X_{k-1} + 25\frac{X_{k-1}}{1+X_{k-1}^2} + 8\cos 1.2k + V_k, \\
Y_k &= \frac{X_k^2}{20} + W_k
\end{aligned}
$$

where $V_k \sim \mathcal{N}\left(0, \sigma_v^2\right)$, $W_k \sim \mathcal{N}\left(0, \sigma_w^2\right)$ and $X_1 \sim \mathcal{N}\left(0, 5^2\right)$.

- We first compare PMCMC to CBMC (Frenkel et al., 1992) for $\sigma_v^2 = 10$ and $\sigma_w^2 = 0.5$ in terms of acceptance rate of the proposal for various N and T. Note that PMCMC is marginally more expensive to implement.



Acceptance probabilities for PMH (left) and CBMC (right) as a function of T and N. Each point was computed using 250,000 runs.

A More Realistic Problem

- In numerous scenarios, $\mu(x)$, $f(x'|x)$ and $g(y|x)$ are not known exactly but depend on an unknown parameter θ and we note $\mu_\theta(x)$, $f_\theta(x'|x)$ and $g_\theta(y|x)$.

A More Realistic Problem

- In numerous scenarios, $\mu(x)$, $f(x'|x)$ and $g(y|x)$ are not known exactly but depend on an unknown parameter θ and we note $\mu_\theta(x)$, $f_\theta(x'|x)$ and $g_\theta(y|x)$.
- We set a prior $p(\theta)$ on θ and we are interested in

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

A More Realistic Problem

- In numerous scenarios, $\mu(x)$, $f(x'|x)$ and $g(y|x)$ are not known exactly but depend on an unknown parameter θ and we note $\mu_\theta(x)$, $f_\theta(x'|x)$ and $g_\theta(y|x)$.
- We set a prior $p(\theta)$ on θ and we are interested in

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

- Many SMC methods have been proposed to estimate $p(\theta, x_{1:T}|y_{1:T})$: they all suffer from severe drawbacks.

Marginal Metropolis-Hastings Algorithm

- To sample from $p(\theta, x_{1:T} | y_{1:T})$, an efficient MCMC strategy would consist of using the following so-called marginal MH algorithm which uses

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

as

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

Marginal Metropolis-Hastings Algorithm

- To sample from $p\left(\theta, x_{1:T}\middle| y_{1:T}\right)$, an efficient MCMC strategy would consist of using the following so-called marginal MH algorithm which uses

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

as

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

- **Problem**: We do not know $p\left(y_{1:T}\middle|\theta\right) = \int p\left(x_{1:T}, y_{1:T}\middle|\theta\right) dx_{1:T}$ analytically and we do not know how to sample from $p\left(x_{1:T}\middle| y_{1:T},\theta\right)$.

Marginal Metropolis-Hastings Algorithm

- To sample from $p\left(\theta, x_{1:T} \middle| y_{1:T}\right)$, an efficient MCMC strategy would consist of using the following so-called marginal MH algorithm which uses

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

as

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

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

"Ideal" Marginal MH Sampler

<u>*At iteration* 1</u>

- Set $\theta(1)$.

<u>*At iteration* i; $i \geq 2$</u>

“Ideal” Marginal MH Sampler

At iteration 1

- Set $\theta(1)$.
- Sample $X_{1:T}(1) \sim p(\cdot|\, y_{1:T}, \theta(1))$.

At iteration i; $i \geq 2$

"Ideal" Marginal MH Sampler

At iteration 1

- Set $\theta(1)$.
- Sample $X_{1:T}(1) \sim p(\cdot|\, y_{1:T}, \theta(1))$.

At iteration i; $i \geq 2$

- Sample $\theta^* \sim q(\cdot|\, \theta(i-1))$.

"Ideal" Marginal MH Sampler

<u>*At iteration 1*</u>

- Set $\theta(1)$.
- Sample $X_{1:T}(1) \sim p(\cdot|\, y_{1:T}, \theta(1))$.

<u>*At iteration i; $i \geq 2$*</u>

- Sample $\theta^* \sim q(\cdot|\, \theta(i-1))$.
- Sample $X^*_{1:T} \sim p(\cdot|\, y_{1:T}, \theta^*)$.

"Ideal" Marginal MH Sampler

At iteration 1

- Set $\theta(1)$.
- Sample $X_{1:T}(1) \sim p(\cdot|\, y_{1:T}, \theta(1))$.

At iteration i; $i \geq 2$

- Sample $\theta^* \sim q(\cdot|\, \theta(i-1))$.
- Sample $X^*_{1:T} \sim p(\cdot|\, y_{1:T}, \theta^*)$.
- With probability

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

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

Particle Marginal MH Sampler

At iteration 1

- Set $\theta(1)$ and run an SMC algorithm to obtain $\widehat{p}(x_{1:T} \mid y_{1:T}, \theta(1))$ and $\widehat{p}(y_{1:T} \mid \theta(1))$.

At iteration i; $i \geq 2$

Particle Marginal MH Sampler

At iteration 1

- Set $\theta(1)$ and run an SMC algorithm to obtain $\widehat{p}(x_{1:T} | y_{1:T}, \theta(1))$ and $\widehat{p}(y_{1:T} | \theta(1))$.
- Sample $X_{1:T}(1) \sim \widehat{p}(\cdot | y_{1:T}, \theta(1))$.

At iteration i; $i \geq 2$

Particle Marginal MH Sampler

At iteration 1

- Set $\theta(1)$ and run an SMC algorithm to obtain $\widehat{p}(x_{1:T} | y_{1:T}, \theta(1))$ and $\widehat{p}(y_{1:T} | \theta(1))$.
- Sample $X_{1:T}(1) \sim \widehat{p}(\cdot | y_{1:T}, \theta(1))$.

At iteration i; $i \geq 2$

- Sample $\theta^* \sim q(\cdot | \theta(i-1))$ and run an SMC algorithm to obtain $\widehat{p}(x_{1:T} | y_{1:T}, \theta^*)$ and $\widehat{p}(y_{1:T} | \theta^*)$.

Particle Marginal MH Sampler

At iteration 1

- Set $\theta(1)$ and run an SMC algorithm to obtain $\widehat{p}(x_{1:T} | y_{1:T}, \theta(1))$ and $\widehat{p}(y_{1:T} | \theta(1))$.
- Sample $X_{1:T}(1) \sim \widehat{p}(\cdot | y_{1:T}, \theta(1))$.

At iteration i; $i \geq 2$

- Sample $\theta^* \sim q(\cdot | \theta(i-1))$ and run an SMC algorithm to obtain $\widehat{p}(x_{1:T} | y_{1:T}, \theta^*)$ and $\widehat{p}(y_{1:T} | \theta^*)$.
- Sample $X^*_{1:T} \sim \widehat{p}(\cdot | y_{1:T}, \theta^*)$.

Particle Marginal MH Sampler

At iteration 1

- Set $\theta(1)$ and run an SMC algorithm to obtain $\widehat{p}(x_{1:T} | y_{1:T}, \theta(1))$ and $\widehat{p}(y_{1:T} | \theta(1))$.
- Sample $X_{1:T}(1) \sim \widehat{p}(\cdot | y_{1:T}, \theta(1))$.

At iteration i; $i \geq 2$

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

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

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

Results

- **Proposition**. Assume the "ideal" MH sampler of target density $p\left(\theta, x_{1:T} \mid y_{1:T}\right)$ and proposal density $q\left(\theta^* \mid \theta\right) p\left(x^*_{1:T} \mid y_{1:T}, \theta^*\right)$ is irreducible and aperiodic then for any $N \geq 1$ the PMMH sampler generates a sequence $\{\theta(i), X_{1:T}(i)\}$ such that

$$\left\| \mathcal{L}\left(\left(\theta(i), X_{1:T}(i)\right) \in \cdot\right) - p(\cdot \mid y_{1:T}) \right\| \rightarrow 0 \text{ as } i \rightarrow \infty .$$

Results

- **Proposition**. Assume the "ideal" MH sampler of target density $p\left(\theta, x_{1:T} \middle| y_{1:T}\right)$ and proposal density $q\left(\theta^* \middle| \theta\right) p\left(x_{1:T}^* \middle| y_{1:T}, \theta^*\right)$ is irreducible and aperiodic then for any $N \geq 1$ the PMMH sampler generates a sequence $\{\theta(i), X_{1:T}(i)\}$ such that

$$\left\| \mathcal{L}\left(\left(\theta(i), X_{1:T}(i)\right) \in \cdot\right) - p(\cdot| y_{1:T}) \right\| \rightarrow 0 \text{ as } i \rightarrow \infty .$$

- **Proposition**. Let $\mathcal{U} = \{\theta : \forall M > 0, q_\theta\left(\{\widehat{p}\left(y_{1:T} \middle| \theta\right) > M\}\right) > 0\}$. If $p\left(\mathcal{U} \middle| y_{1:T}\right) > 0$ then the MCMC kernel cannot be geometrically ergodic.

Results

- **Proposition**. Assume the "ideal" MH sampler of target density $p\left(\theta, x_{1:T} \middle| y_{1:T}\right)$ and proposal density $q\left(\theta^* \middle| \theta\right) p\left(x_{1:T}^* \middle| y_{1:T}, \theta^*\right)$ is irreducible and aperiodic then for any $N \geq 1$ the PMMH sampler generates a sequence $\{\theta(i), X_{1:T}(i)\}$ such that

$$\left\| \mathcal{L}\left(\left(\theta(i), X_{1:T}(i)\right) \in \cdot\right) - p(\cdot | y_{1:T}) \right\| \to 0 \text{ as } i \to \infty .$$

- **Proposition**. Let $\mathcal{U} = \{\theta : \forall M > 0, q_\theta\left(\{\widehat{p}\left(y_{1:T} \middle| \theta\right) > M\}\right) > 0\}$. If $p\left(\mathcal{U} \middle| y_{1:T}\right) > 0$ then the MCMC kernel cannot be geometrically ergodic.
- **Proposition**. If the 'ideal' MH kernel is uniformly ergodic and $\sup_{x,y,\theta} \left|g_\theta\left(y \middle| x\right)\right| < C$ then the MCMC kernel is uniformly ergodic.

Stochastic kinetic Lokta-Volterra model

- The Lotka-Volterra model describes the evolution of two species Z_t^1 (prey) and Z_t^2 (predator) which are non-negative integer-valued processes.

Stochastic kinetic Lokta-Volterra model

- The Lotka-Volterra model describes the evolution of two species Z_t^1 (prey) and Z_t^2 (predator) which are non-negative integer-valued processes.
- In a small time interval $(t, t+dt]$, there are three possible transitions for the Markov jump process $Z_t = \left(Z_t^1, Z_t^2\right)$

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

corresponding respectively to prey reproduction, predator reproduction and predator death.

Stochastic kinetic Lokta-Volterra model

- The Lotka-Volterra model describes the evolution of two species Z_t^1 (prey) and Z_t^2 (predator) which are non-negative integer-valued processes.
- In a small time interval $(t, t+dt]$, there are three possible transitions for the Markov jump process $Z_t = \left(Z_t^1, Z_t^2\right)$

$$\Pr\left(Z_{t+dt}^1 = z_t^1 + 1, Z_{t+dt}^2 = z_t^2 \middle| z_t^1, z_t^2\right) = \alpha z_t^1 dt + o\left(dt\right),$$

$$\Pr\left(Z_{t+dt}^1 = z_t^1 - 1, Z_{t+dt}^2 = z_t^2 + 1 \middle| z_t^1, z_t^2\right) = \beta z_t^1 z_t^2 dt + o\left(dt\right),$$

$$\Pr\left(Z_{t+dt}^1 = z_t^1, Z_{t+dt}^2 = z_t^2 - 1 \middle| z_t^1, z_t^2\right) = \gamma z_t^2 dt + o\left(dt\right),$$

  corresponding respectively to prey reproduction, predator reproduction and predator death.
- We assume we observe the process in some noise at some discrete times $n\Delta$

$$Y_n = Z_{n\Delta} + W_n, \, W_n \overset{\text{i.i.d.}}{\sim} \mathcal{N}\left(0, \sigma^2\right)$$

PMCMC for Bayesian inference in Lokta-Volterra model

- We are interested here in making inference about the parameters $\theta = (\alpha, \beta, \gamma)$ where

$$\alpha \sim \mathcal{G}(a, b), \ \beta \sim \mathcal{G}(a, b), \gamma \sim \mathcal{G}(a, b).$$

PMCMC for Bayesian inference in Lokta-Volterra model

- We are interested here in making inference about the parameters $\theta = (\alpha, \beta, \gamma)$ where

$$\alpha \sim \mathcal{G}(a, b), \;\; \beta \sim \mathcal{G}(a, b), \; \gamma \sim \mathcal{G}(a, b).$$

- (Wilkinson, 2006; Boys et al., 2008) develop sophisticated MCMC algorithms to fit this model

PMCMC for Bayesian inference in Lokta-Volterra model

- We are interested here in making inference about the parameters $\theta = (\alpha, \beta, \gamma)$ where

$$\alpha \sim \mathcal{G}(a, b), \ \beta \sim \mathcal{G}(a, b), \gamma \sim \mathcal{G}(a, b).$$

- (Wilkinson, 2006; Boys et al., 2008) develop sophisticated MCMC algorithms to fit this model
- We use PMCMC with an off-the-shelve particle filter.

A Block Gibbs Sampler

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

A Block Gibbs Sampler

- To sample from $p(\theta, x_{1:T} | y_{1:T})$, an alternative MCMC strategy consists of using the following block Gibbs sampler.
- *At iteration i*
  Sample $X_{1:T}(i) \sim p(\cdot | y_{1:T}, \theta(i-1))$.
  Sample $\theta(i) \sim p(\cdot | y_{1:T}, X_{1:T}(i))$.

A Block Gibbs Sampler

- To sample from $p\left(\theta, x_{1:T} \mid y_{1:T}\right)$, an alternative MCMC strategy consists of using the following block Gibbs sampler.
- *At iteration i*
  Sample $X_{1:T}(i) \sim p\left(\cdot \mid y_{1:T}, \theta(i-1)\right)$.
  Sample $\theta(i) \sim p\left(\cdot \mid y_{1:T}, X_{1:T}(i)\right)$.
- **Problems**: We do not know how to sample from $p\left(x_{1:T} \mid y_{1:T}, \theta\right)$ so we typically use instead an MH one-at-a time algorithm.

Particle Gibbs Sampler

At iteration 1

- Set $\theta(1)$ and $X_{1:T}(1)$, $I(1)$ randomly where $I(1)$ is the ancestral lineage of $X_{1:T}(1)$.

At iteration i; $i \geq 2$

Particle Gibbs Sampler

At iteration 1

- Set $\theta(1)$ and $X_{1:T}(1)$, $I(1)$ randomly where $I(1)$ is the ancestral lineage of $X_{1:T}(1)$.

At iteration i; $i \geq 2$

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

Particle Gibbs Sampler

At iteration 1

- Set $\theta(1)$ and $X_{1:T}(1)$, $I(1)$ randomly where $I(1)$ is the ancestral lineage of $X_{1:T}(1)$.

At iteration i; $i \geq 2$

- Sample $\theta(i) \sim p(\cdot | y_{1:T}, X_{1:T}(i-1))$.
- Run a conditional SMC algorithm for $\theta(i)$ consistent with $X_{1:T}(i-1), I(i-1)$.

Particle Gibbs Sampler

At iteration 1

- Set $\theta(1)$ and $X_{1:T}(1)$, $I(1)$ randomly where $I(1)$ is the ancestral lineage of $X_{1:T}(1)$.

At iteration i; $i \geq 2$

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

Results

- **Proposition**. Assume that the 'ideal' Gibbs sampler is irreducible and aperiodic then for any $N \geq 2$ the particle Gibbs sampler generates a sequence $\{\theta(i), X_{1:T}(i)\}$ such that

$$\left\| \mathcal{L}\left(\left(\theta(i), X_{1:T}(i)\right) \in \cdot\right) - p(\cdot | y_{1:T}) \right\| \rightarrow 0 \text{ as } i \rightarrow \infty.$$

- We now consider the case where $\theta = \left(\sigma_v^2, \sigma_w^2\right)$ is unknown and random with vague inverse-Gamma priors.

- We now consider the case where $\theta = \left(\sigma_v^2, \sigma_w^2\right)$ is unknown and random with vague inverse-Gamma priors.
- We compare the Particle Gibbs sampler to an MH one-at-a time using the same proposal distribution.

- We now consider the case where $\theta = \left(\sigma_v^2, \sigma_w^2\right)$ is unknown and random with vague inverse-Gamma priors.
- We compare the Particle Gibbs sampler to an MH one-at-a time using the same proposal distribution.
- We update N times the latent process in the MH one-at-a time before sampling θ to perform a reasonably fair comparison with PMCMC.

- We now consider the case where $\theta = \left(\sigma_v^2, \sigma_w^2\right)$ is unknown and random with vague inverse-Gamma priors.
- We compare the Particle Gibbs sampler to an MH one-at-a time using the same proposal distribution.
- We update N times the latent process in the MH one-at-a time before sampling θ to perform a reasonably fair comparison with PMCMC.
- In this example, the MH one-at-a time does not provide reliable results and is very sensitive to initialization.

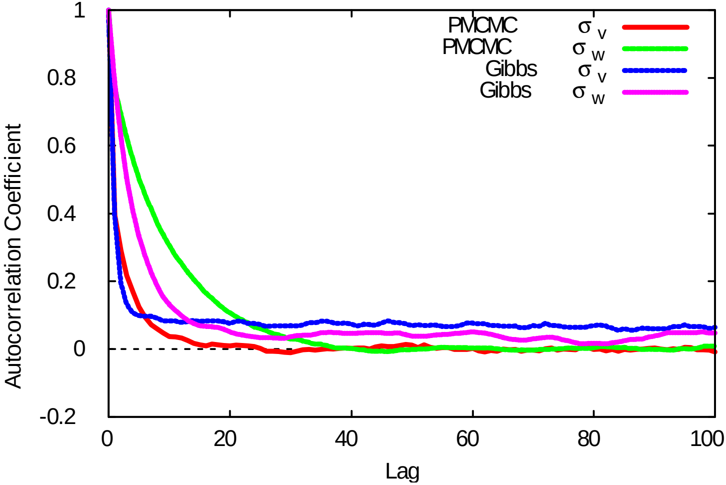
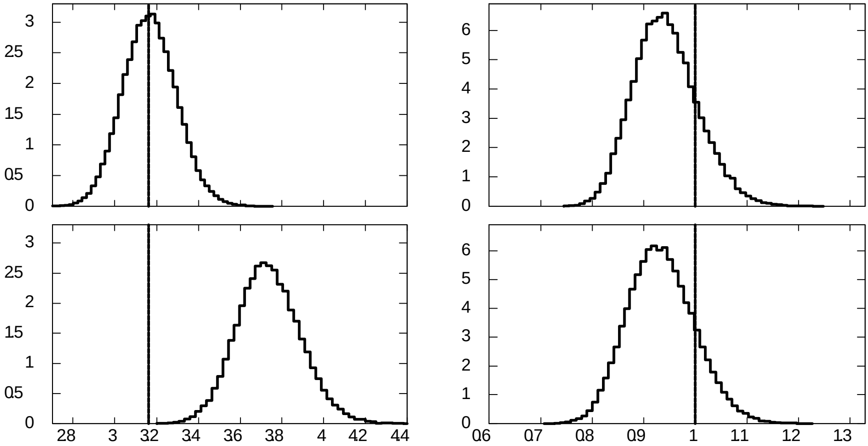


Figure: Standard Gibbs and EAV algorithm for $N = 500$



Estimates of $p(\sigma_V | y_{1:T})$ (left) and $p(\sigma_W | y_{1:T})$ (right) for particle Gibbs (top) and Gibbs (bottom)

Extensions

- This methodology is by no means limited to state-space models and in particular there is no need for any Markovian assumption.

Extensions

- This methodology is by no means limited to state-space models and in particular there is no need for any Markovian assumption.
- Wherever SMC have been used, PMCMC can be used: self-avoiding random walks, contingency tables, mixture models etc.

Extensions

- This methodology is by no means limited to state-space models and in particular there is no need for any Markovian assumption.
- Wherever SMC have been used, PMCMC can be used: self-avoiding random walks, contingency tables, mixture models etc.
- Sophisticated SMC algorithms can also be used including 'clever' importance distributions and resampling schemes.

Discussion

- PMCMC methods allow us to design ‘good’ high dimensional proposals based only on low dimensional proposals.

Discussion

- PMCMC methods allow us to design 'good' high dimensional proposals based only on low dimensional proposals.
- PMCMC allow us to perform Bayesian inference for dynamic models for which only forward simulation is possible.