The Conditional Particle Filter

Sumeetpal S. Singh
CAMBRIDGE UNIVERSITY ENGINEERING DEPARTMENT

jointly with A. Lee, M. Vihola
older work with F. Lindsten, E. Moulines
older still with N. Chopin, B. Kuhlenschimdt

The Institute for Mathematical Sciences, NUS, 28 Aug. 2018
Running example (Yu & Meng 2011):

\[ X_{t+1} = \rho X_t + \sigma W_{t+1}, \quad W_{t+1} \sim \text{i.i.d. } N(0, 1) \]

\[ Y_t \mid X_t = x_t \sim \text{Poisson}(e^{x_t+\mu}) \]
Running example (Yu & Meng 2011):

\[ X_{t+1} = \rho X_t + \sigma W_{t+1}, \quad W_{t+1} \sim \text{i.i.d. } N(0, 1) \]

\[ Y_t | X_t = x_t \sim \text{Poisson}(e^{x_t + \mu}) \]

In general:

Figure: Evolution of the random variables of a HMM.
The posterior: \( p(\theta, x_{0:T} | y_{0:T}) \), \( \theta = (\mu, \rho, \sigma) \)
The posterior: \( p(\theta, x_0:T \mid y_0:T) \), \( \theta = (\mu, \rho, \sigma) \)

Gibbs sampler (one cycle): \( (\theta, x_0:T) \rightarrow (\theta', x'_0:T) \)
Inference Objective

The posterior: \( p(\theta, x_{0:T}|y_{0:T}) \), \( \theta = (\mu, \rho, \sigma) \)

Gibbs sampler (one cycle): \( (\theta, x_{0:T}) \rightarrow (\theta', x'_{0:T}) \)

\[
\begin{align*}
\sigma'|_(x_{0:T}, \mu, \rho) & \sim \text{Gamma} (\cdots) \\
\rho'|_(x_{0:T}, \mu, \sigma') & \sim \text{Normal} (\cdots) \\
\mu'|_(x_{0:T}, \sigma', \rho') & \sim \text{Normal} (\cdots) \\
x'_{0:T}|_(\sigma', \mu', \rho') & \sim p(x_{0:T}|\theta', y_{0:T})
\end{align*}
\]
Inference Objective

The posterior: \[ p(\theta, x_{0:T}|y_{0:T}), \quad \theta = (\mu, \rho, \sigma) \]

Gibbs sampler (one cycle): \( (\theta, x_{0:T}) \rightarrow (\theta', x'_{0:T}) \)

\[ \sigma'|_{(x_{0:T}, \mu, \rho)} \sim Gamma(\cdots) \]
\[ \rho'|_{(x_{0:T}, \mu, \sigma')} \sim Normal(\cdots) \]
\[ \mu'|_{(x_{0:T}, \sigma', \rho')} \sim Normal(\cdots) \]
\[ x'_{0:T}|_{(\sigma', \mu', \rho')} \sim p(x_{0:T}|\theta', y_{0:T}) \]

In general cannot sample from \( p(x_{0:T}|\theta', y_{0:T}) \)
Inference Objective

The posterior: \[ p(\theta, x_{0:T}|y_{0:T}), \quad \theta = (\mu, \rho, \sigma) \]

Gibbs sampler (one cycle): \[ (\theta, x_{0:T}) \rightarrow (\theta', x'_{0:T}) \]

\[ \sigma'|(x_{0:T}, \mu, \rho) \sim \text{Gamma}(\cdots) \]
\[ \rho'|(x_{0:T}, \mu, \sigma') \sim \text{Normal}(\cdots) \]
\[ \mu'|(x_{0:T}, \sigma', \rho') \sim \text{Normal}(\cdots) \]
\[ x'_{0:T}|(\sigma', \mu', \rho') \sim p(x_{0:T}|\theta', y_{0:T}) \]

In general cannot sample from \[ p(x_{0:T}|\theta', y_{0:T}) \]

An old remedy is one at a time: \[ x_i|(x'_{0:i-1}, x_{i+1:T}, \theta') \]
Popularised by Gordon, Salmond and Smith (1993)
Particle Filtering

- Popularised by Gordon, Salmond and Smith (1993)
- (Sequential) Importance sampling method to approximate

\[ p(x_{0:T} | \theta, y_{0:T}) \]

using non-iid samples:

\[ \mathbb{E}(h(X_{0:T}) | \theta, y_{0:T}) \approx \sum_{i=1}^{N} h(X_{0:T}^{(i)}) W_T^{(i)} \]
Particle Filter execution for sampling $p(x_{0:T}|y_{0:T})$

Given $\sum_{i=1}^{N} \delta_{X^{(i)}} \approx p(x_{0:t}|y_{0:t})$, approximate $p(x_{0:t+1}|y_{0:t+1})$
Particle Filter execution for sampling $p(x_0:T|y_0:T)$

Given $\sum_{i=1}^{N} \delta x^{(i)}_{0:t} \approx p(x_0:t|y_0:t)$, approximate $p(x_0:t+1|y_0:t+1)$
Given $\sum_{i=1}^{N} \delta_{X_{0:t}^{(i)}} \approx p(x_{0:t} | y_{0:t})$, approximate $p(x_{0:t+1} | y_{0:t+1})$

Sample: $X_{t+1}^{(i)} \sim f(X_{t}^{(i)}, x_{t+1})$

Weight: $w_{t+1}^{(i)} = g(X_{t+1}^{(i)}, y_{t+1})$
Particle Filter execution for sampling $p(x_{0:T} | y_{0:T})$

Given $\sum_{i=1}^{N} \delta_{X_{0:t}}^{(i)} \approx p(x_{0:t} | y_{0:t})$, approximate $p(x_{0:t+1} | y_{0:t+1})$

Sample: $X_{t+1}^{(i)} \sim f(X_{t}^{(i)}, x_{t+1})$

Weight: $w_{t+1}^{(i)} = g(X_{t+1}^{(i)}, y_{t+1})$

$p(x_{0:t+1} | y_{0:t+1}) \approx \sum_{i=1}^{N} W_{t+1}^{(i)} \delta_{X_{0:t+1}}^{(i)}$
Given $\sum_{i=1}^{N} \delta_{X_{0:t}^{(i)}} \approx p(x_{0:t} | y_{0:t})$, approximate $p(x_{0:t+1} | y_{0:t+1})$

Sample: $X_{t+1}^{(i)} \sim f(X_{t}^{(i)}, x_{t+1})$

Weight: $w_{t+1}^{(i)} = g(X_{t+1}^{(i)}, y_{t+1})$

$p(x_{0:t+1} | y_{0:t+1}) \approx \sum_{i=1}^{N} W_{t+1}^{(i)} \delta_{X_{0:t+1}^{(i)}}$

Resample:

$p(x_{0:t+1} | y_{0:t+1}) \approx \sum_{i=1}^{N} \delta_{X_{0:t+1}^{(i)}}$
Given $\sum_{i=1}^{N} \delta_{X_{0:t}^{(i)}} \approx p(x_{0:t} | y_{0:t})$, approximate $p(x_{0:t+1} | y_{0:t+1})$

Sample: $X_{t+1}^{(i)} \sim f(X_{t}^{(i)}, x_{t+1})$

Weight: $w_{t+1}^{(i)} = g(X_{t+1}^{(i)}, y_{t+1})$

$p(x_{0:t+1} | y_{0:t+1}) \approx \sum_{i=1}^{N} W_{t+1}^{(i)} \delta_{X_{0:t+1}^{(i)}}$

Resample: 

$p(x_{0:t+1} | y_{0:t+1}) \approx \sum_{i=1}^{N} \delta_{X_{0:t+1}^{(i)}}$
The final (Gibbs) step for states, $(\theta', x_{0:T}) \rightarrow (\theta', x'_{0:T})$ with

$$x'_{0:T}(\sigma', \mu', \rho') \sim \text{PF approx. of } p(x_{0:T} | \theta', y_{0:T})$$
The final (Gibbs) step for states, \((\theta', x_{0:T}) \rightarrow (\theta', x'_{0:T})\) with

\[x'_{0:T} | (\sigma', \mu', \rho') \sim \text{PF approx. of } p(x_{0:T} | \theta', y_{0:T})\]

Why not, since used extensively in EM and gradient methods to learn \(\theta\)

\[Q(\theta, \theta') = \mathbb{E} \left\{ \log p(x_{0:T}, y_{0:T} | \theta') \bigg| \theta, y_{0:T} \right\} \]
Particle Filtering (cont’d)

The final (Gibbs) step for states, \((\theta', x_{0:T}) \rightarrow (\theta', x'_{0:T})\) with

\[x'_{0:T} | (\sigma', \mu', \rho') \sim \text{PF approx. of } p(x_{0:T} | \theta', y_{0:T})\]

Why not, since used extensively in EM and gradient methods to learn \(\theta\)

\[Q(\theta, \theta') = \mathbb{E} \left\{ \log p(x_{0:T}, y_{0:T} | \theta') | \theta, y_{0:T} \right\}\]

In practise particle number \(N\) must grow linearly with \(T\)

(Many results on the error of Particle filter estimates, e.g. Del Moral’s book 2004, …)
Particle Filtering (cont’d)

The final (Gibbs) step for states, \((\theta', x_{0:T}) \rightarrow (\theta', x'_{0:T})\) with

\[
x'_{0:T}|(\sigma', \mu', \rho') \sim \text{PF approx. of } p(x_{0:T}|\theta', y_{0:T})
\]

Why not, since used extensively in EM and gradient methods to learn \(\theta\)

\[
Q(\theta, \theta') = \mathbb{E} \left\{ \log p(x_{0:T}, y_{0:T}|\theta') | \theta, y_{0:T} \right\}
\]

In practise particle number \(N\) must grow linearly with \(T\)
(Many results on the error of Particle filter estimates, e.g. Del Moral’s book 2004, ...)

The bias free (mathematically correct) way (ADH2010) is to use the \textit{conditional} Particle Filter
The final step for states, \((\theta', x_{0:T}) \rightarrow (\theta', x'_{0:T})\) with

\[ x'_{0:T} \sim \text{conditional Particle Filter (CPF)} \]
The Conditional Particle Filter (Andrieu, Doucet & Holenstein, 2010)

The final step for states, \((\theta', x_{0:T}) \rightarrow (\theta', x'_{0:T})\) with

\[ x'_{0:T} \sim \text{conditional Particle Filter (CPF)} \]

A CPF simulates a PF with \(N\) particles for \(T\) time steps as “usual” but with one particle set to \(X_{0:T}^{(1)} = x_{0:T}\)
The Conditional Particle Filter (Andrieu, Doucet & Holenstein, 2010)

- The final step for states, $(\theta', x_{0:T}) \rightarrow (\theta', x'_{0:T})$ with

  $$x'_{0:T} \sim \text{conditional Particle Filter (CPF)}$$

- A CPF simulates a PF with $N$ particles for $T$ time steps as “usual” but with one particle set to $X^{(1)}_{0:T} = x_{0:T}$
  - Then choose one particle randomly according to its weight
The final step for states, \((\theta', x_{0:T}) \rightarrow (\theta', x'_{0:T})\) with

\[x'_{0:T} \sim \text{conditional Particle Filter (CPF)}\]

A CPF simulates a PF with \(N\) particles for \(T\) time steps as “usual” but with one particle set to \(X_{0:T}^{(1)} = x_{0:T}\)

– Then choose one particle randomly according to its weight

In effect, the CPF is a Markov kernel:

\[X'_{0:T} \sim P_{\theta', N}(x_{0:T}, dx'_{0:T})\]
The Conditional Particle Filter (Andrieu, Doucet & Holenstein, 2010)

- The final step for states, \((\theta', x_{0:T}) \rightarrow (\theta', x'_{0:T})\) with
  
  \[x'_{0:T} \sim \text{conditional Particle Filter (CPF)}\]

- A CPF simulates a PF with \(N\) particles for \(T\) time steps as “usual” but with one particle set to \(X^{(1)}_{0:T} = x_{0:T}\)
  - Then choose one particle randomly according to its weight

- In effect, the CPF is a Markov kernel:
  
  \[X'_{0:T} \sim P_{\theta', N}(x_{0:T}, dx'_{0:T})\]

- Invariant measure is \(p(x_{0:T}|\theta', y_{0:T})\) for any \(N \geq 2\)
The Conditional Particle Filter (Andrieu, Doucet & Holenstein, 2010)

- The final step for states, \((\theta', x_{0:T}) \rightarrow (\theta', x'_{0:T})\) with
  \[x'_{0:T} \sim \text{conditional Particle Filter (CPF)}\]

- A CPF simulates a PF with \(N\) particles for \(T\) time steps as “usual” but with one particle set to \(X^{(1)}_{0:T} = x_{0:T}\)
  – Then choose one particle randomly according to its weight
- In effect, the CPF is a Markov kernel:
  \[X'_{0:T} \sim P_{\theta',N}(x_{0:T}, dx'_{0:T})\]

- Invariant measure is \(p(x_{0:T}|\theta', y_{0:T})\) for any \(N \geq 2\)
- Effective sampler? How should \(N\) grow with \(T\)?
CPF: $X_0$’s autocorrelation

Sampling $p(x_{0:399}|y_{0:399})$ with 200 particles (Chopin & S., 2013)

Statistic: ACF $X_0$
Coupling the CPF (Chopin & S., 2013)

The outputs of two CPFs $P_N(x_0:T, dx_0':T)$ and $P_N(\tilde{x}_0:T, dx_0':T)$ with different inputs $x_0:T$ and $\tilde{x}_0:T$ but implemented with common random numbers can be the same with (high) probability.

Thus if $(X_0:T, \tilde{X}_0:T) \sim \text{CPF}(x_0:T, \tilde{x}_0:T)$ then $X_0:T = P_N(x_0:T, dx_0':T)$ and $\tilde{X}_0:T = P_N(\tilde{x}_0:T, dx_0':T)$.

We have $P(X_0:T \neq \tilde{X}_0:T) \leq \epsilon$. 
The outputs of two CPFs

$$P_N(x_{0:T}, dx'_{0:T}) \quad P_N(\tilde{x}_{0:T}, d\tilde{x}'_{0:T})$$

with different inputs $x_{0:T}$ and $\tilde{x}_{0:T}$
but implemented with common random numbers
can be the same with (high) probability

Thus if $(X_{0:T}, \tilde{X}_{0:T}) \sim \text{CCPF}(x_{0:T}, \tilde{x}_{0:T})$ then

$$X_{0:T} = P_N(x_{0:T}, dx'_{0:T}) \quad \tilde{X}_{0:T} = P_N(\tilde{x}_{0:T}, d\tilde{x}'_{0:T})$$

We have

$$P(X_{0:T} \neq \tilde{X}_{0:T}) \leq \epsilon$$
The outputs of two CPFs

\[ P_N(x_0: T, d'x_0: T) \quad P_N(\tilde{x}_0: T, d'x_0: T) \]

with different inputs \( x_0: T \) and \( \tilde{x}_0: T \)

but implemented with common random numbers

can be the same with (high) probability

Thus if \((X_0: T, \tilde{X}_0: T) \sim CCPF(x_0: T, \tilde{x}_0: T)\) then

\[ X_0: T \overset{d}{=} P_N(x_0: T, d'x_0: T) \quad \tilde{X}_0: T \overset{d}{=} P_N(\tilde{x}_0: T, d'x_0: T) \]
The outputs of two CPFs

\[ P_N(x_{0:T}, d\tilde{x}'_{0:T}) \quad P_N(\tilde{x}_{0:T}, d\tilde{x}'_{0:T}) \]

with different inputs \( x_{0:T} \) and \( \tilde{x}_{0:T} \)
but implemented with common random numbers

can be the same with (high) probability

Thus if \( (X_{0:T}, \tilde{X}_{0:T}) \sim CCPF(x_{0:T}, \tilde{x}_{0:T}) \) then

\[ X_{0:T} \overset{d}{=} P_N(x_{0:T}, d\tilde{x}'_{0:T}) \quad \tilde{X}_{0:T} \overset{d}{=} P_N(\tilde{x}_{0:T}, d\tilde{x}'_{0:T}) \]

We have

\[ \mathbb{P}(X_{0:T} \neq \tilde{X}_{0:T}) \leq \epsilon \]
<table>
<thead>
<tr>
<th>Chopin+S. (2013)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Can construct a coupling $(P_N(x_0:T, \cdot), P_N(\tilde{x}_0:T, \cdot))$ with probability at least $1 - \epsilon$.</td>
</tr>
</tbody>
</table>

$$\| P_N^k(x_0:T, dx'_0:T) - p(dx'_0:T|y_0:T) \|_{tv} \leq \epsilon^k$$
Chopin & S. (2013)

Can construct a coupling \( (P_N(x_0; T, \cdot), P_N(\tilde{x}_0; T, \cdot)) \) with probability at least \( 1 - \epsilon \).

\[
\| P_N^k(x_0; T, dx'_0; T) - p(dx'_0; T | y_0; T) \|_{tv} \leq \epsilon^k
\]

Kuhlenschimdt + S. (2014)

\[
\| P_N^k(x_0; T, dx'_0; T) - p(dx'_0; T | y_0; T) \|_{tv} \leq \text{Const.} \times \left( \frac{T}{N} \right)^k
\]
Uniform Ergodicity

Chopin + S. (2013)

Can construct a coupling \( (P_N(x_0:T, \cdot), P_N(\tilde{x}_0:T, \cdot)) \) with probability at least \( 1 - \epsilon \).

\[
\| P_N^k(x_0:T, dx'_0:T) - \mathbb{P}(dx'_0:T | y_0:T) \|_{tv} \leq \epsilon^k
\]

Kuhlenschmidt + S. (2014)

\[
\| P_N^k(x_0:T, dx'_0:T) - \mathbb{P}(dx'_0:T | y_0:T) \|_{tv} \leq \text{Const.} \times \left( \frac{T}{N} \right)^k
\]

Andrieu, Lee, Vihola (2014); Douc, Lindsten, Moulines (2014)

\[ P_N(x_0:T, dx'_0:T) \geq \epsilon(N, T) \mathbb{P}(x'_0:T | y_0:T) \]

and

\[
\liminf_T \epsilon(N, T) > 0 \quad \text{provided } N \propto T
\]
These results say particles must increase linearly with $T$ costing $T^2$ per application of $P_{T,N}$.
These results say particles must increase linearly with $T$ costing $T^2$ per application of $P_{T,N}$.

Could CPF work with a fixed number of particles? Costing $NT$ per application of CPF or $P_{T,N}$. 
These results say particles must increase linearly with $T$ costing $T^2$ per application of $P_{T,N}$

Could CPF work with a fixed number of particles? Costing $NT$ per application of CPF or $P_{T,N}$.

N. Whiteley (RSS discussion of PMCMC, 2010) suggested an extra *backward* step that tries to modify (recursively, backward in time) the ancestry of the selected trajectory.
These results say particles must increase linearly with $T$ costing $T^2$ per application of $P_{T,N}$.

Could CPF work with a fixed number of particles? Costing $NT$ per application of CPF or $P_{T,N}$.

N. Whiteley (RSS discussion of PMCMC, 2010) suggested an extra backward step that tries to modify (recursively, backward in time) the ancestry of the selected trajectory.

Highly successful in practise but no theoretical verification.
Blocked Gibbs sampler for $p(x_0:T \mid y_0:T)$

- Group states $x_0:T$ into $m$ overlapping blocks

\[ J_1 \quad J_2 \quad J_3 \quad J_4 \quad J_5 \]

When sampling block $J_i = r:s$, sample from $p(x_r:s \mid x_{r-1}, y_r:s, x_{s+1})$ while holding remaining states unchanged.

Cycle through the blocks in any order, sequentially, odd-even etc.

Effectively sampling $p(x_0:T \mid y_0:T)$ using the Markov kernel $P(x_0:T, dx'_{0:T}) = P_o P_e$ where

\[ P_o = P_{J_1} P_{J_3} \cdots P_{J_m} \]

\[ P_e = P_{J_2} P_{J_4} \cdots P_{J_{m-1}} \]
Blocked Gibbs sampler for $p(x_{0:T}|y_{0:T})$

- Group states $x_{0:T}$ into $m$ overlapping blocks.
- When sampling block $J_i = r:s$, sample from

$$p(x_{r:s}|x_{r-1}, y_{r:s}, x_{s+1})$$

while holding remaining states unchanged.
Blocked Gibbs sampler for $p(x_{0:T}|y_{0:T})$

- Group states $x_{0:T}$ into $m$ overlapping blocks.
- When sampling block $J_i = r:s$, sample from
  \[ p(x_r:s|x_{r-1}, y_r:s, x_{s+1}) \]
  while holding remaining states unchanged.
- Cycle through the blocks in any order, sequentially, odd-even etc.
Blocked Gibbs sampler for $p(x_0:T | y_0:T)$

- Group states $x_0:T$ into $m$ overlapping blocks.
- When sampling block $J_i = r:s$, sample from
  
  $$p(x_{r:s} | x_{r-1}, y_{r:s}, x_{s+1})$$

  while holding remaining states unchanged.
- Cycle through the blocks in any order, sequentially, odd-even etc.
- Effectively sampling $p(x_0:T | y_0:T)$ using the Markov kernel $P(x_0:T, dx_0:T)$.
Blocked Gibbs sampler for $p(x_{0:T} | y_{0:T})$

- Group states $x_{0:T}$ into $m$ overlapping blocks
- When sampling block $J_i = r : s$, sample from

$$p(x_r:s | x_{r-1}, y_r:s, x_{s+1})$$

while holding remaining states unchanged.
- Cycle through the blocks in any order, sequentially, odd-even etc.
- Effectively sampling $p(x_{0:T} | y_{0:T})$ using the Markov kernel

$$P(x_{0:T}, dx'_{0:T})$$

$$P = P_o P_e$$ where

$$\begin{align*}
P_o &= P_{J_1} P_{J_3} \cdots P_{J_m} \\
P_e &= P_{J_2} P_{J_4} \cdots P_{J_{m-1}}
\end{align*}$$
If \( \{J_1, \ldots, J_m\} \) be an arbitrary cover of \( \{1, \ldots, T\} \)

If \( \mathcal{P} \) is the blocked Gibbs kernel of one complete sweep then

\[
\|p(dx_0:T|y_0:T) - \mu\mathcal{P}^k\|_{\text{tv}} \leq (T + 1)\lambda^k
\]

where \( \lambda \) is \( T \)-independent (S., Lindsten and Moulines, 2015)
If \( \{ J_1, \ldots, J_m \} \) be an arbitrary cover of \( \{ 1, \ldots, T \} \)

If \( \mathcal{P} \) is the blocked Gibbs kernel of one complete sweep then

\[
\left\| \rho(dx_0:T|y_0:T) - \mu \mathcal{P}^k \right\|_{tv} \leq (T + 1) \lambda^k
\]

where \( \lambda \) is \( T \)-independent (S., Lindsten and Moulines, 2015)

Once you decide on a block size and overlap proportion, works for any time-series length \( T \)

Rate quickens, \( \lambda \rightarrow 0 \), as block overlap increases.
If \( \{J_1, \ldots, J_m\} \) be an arbitrary cover of \( \{1, \ldots, T\} \)

If \( \mathcal{P} \) is the blocked Gibbs kernel of one complete sweep then

\[
\left\| p(dx_0:T|y_0:T) - \mu \mathcal{P}^k \right\|_{tv} \leq (T + 1)\lambda^k
\]

where \( \lambda \) is \( T \)-independent (S., Lindsten and Moulines, 2015)

- Once you decide on a block size and overlap proportion, works for any time-series length \( T \)
- Rate quickens, \( \lambda \to 0 \), as block overlap increases.
- Recall \( \mathcal{P} = P_{J_1} P_{J_2} \cdots P_{J_m} \). Idea is to approximate each \( P_{J_i} \) with CPF.
Uniform ergodicity: fixed $N$ and any $T$!

- Approximate each block kernel $P_{J_i}$ with CPF $P_{J_i,N}$:

  $$(\text{ideal}) \mathcal{P} = P_{J_1} P_{J_2} \cdots P_{J_m} \quad (\text{CPF}) \mathcal{P}_N = P_{J_1,N} P_{J_2,N} \cdots P_{J_m,N}$$
Approximate each block kernel $P_{J_i}$ with CPF $P_{J_i,N}$:

(ideal) $\mathcal{P} = P_{J_1} P_{J_2} \cdots P_{J_m}$ \hspace{1cm} (CPF) $\mathcal{P}_N = P_{J_1,N} P_{J_2,N} \cdots P_{J_m,N}$

If $\mathcal{P}_N$ is the blocked pGibbs kernel of one complete sweep then

$$ || \rho(dx_{0:T}|y_{0:T}) - \mu \mathcal{P}_N^k ||_{tv} \leq (T + 1) \lambda_N^k $$

(S., Lindsten and Moulines, 2015)
Approximate each block kernel $P_{J_i}$ with CPF $P_{J_i,N}$:

(ideal) $\mathcal{P} = P_{J_1} P_{J_2} \cdots P_{J_m}$  
(CPF) $\mathcal{P}_N = P_{J_1,N} P_{J_2,N} \cdots P_{J_m,N}$

If $\mathcal{P}_N$ is the blocked pGibbs kernel of one complete sweep then

$$||p(dx_0:T | y_0:T) - \mu \mathcal{P}^k_N||_{tv} \leq (T + 1) \lambda_N^k$$

(S., Lindsten and Moulines, 2015)

Rate is

$$\lambda_N = \sqrt{\lambda} + \text{Const.} \times \max \text{ block size} \times \frac{1}{N}$$
The outputs of two CPFs

\[ P_N(x_0:T, dx'_0:T) \quad P_N(\tilde{x}_0:T, d\tilde{x}'_0:T) \]

with different inputs \( x_0:T \) and \( \tilde{x}_0:T \)
but implemented with common random numbers

can be the same with (high) probability

Thus if \( (X_0:T, \tilde{X}_0:T) \sim \text{CCPF}(x_0:T, \tilde{x}_0:T) \) then

\[ X_0:T \overset{d}{=} P_N(x_0:T, dx'_0:T) \quad \tilde{X}_0:T \overset{d}{=} P_N(\tilde{x}_0:T, d\tilde{x}'_0:T) \]

We have

\[ \mathbb{P}(X_0:T \neq \tilde{X}_0:T) \leq \epsilon \]
1: Set $X_0:T[1] \leftarrow \text{CPF}(x_0:T)$, $\tilde{X}_0:T[1] = x_0:T$, $x_0:T$ arbitrary.
2: for $n = 2, 3, \ldots$ do
3: \[(X_0:T[n], \tilde{X}_0:T[n]) \leftarrow \text{CCPF}(X_0:T[n-1], \tilde{X}_0:T[n-1])\]
4: if $X_0:T[n] = \tilde{X}_0:T[n]$ then output
\[
Z = h(X_0:T[1]) + \sum_{k=2}^{n} h(X_0:T[k]) - h(\tilde{X}_0:T[k])
\]
5: end for

- **Unbiased estimation:**
\[
\mathbb{E}(h(Z)) = \int h(x_0:T)p(x_0:T|y_0:T)dx_0:T
\]

- Jacob, Lindsten, Schon (2017) use the CCPF within the scheme of Glynn & Rhee (2014),
Coupling for Unbiased estimation

- Works because (i) \( \tilde{X}_{0:T}[k] \overset{d}{=} X_{0:T}[k - 1] \),
  (ii) coupling time is finite and
  (iii) the coupling CCPF is for ergodic kernels

\[
\| P^k_N - p(x_{0:T} | y_{0:T}) \|_{tv} \xrightarrow{k \to \infty} 0
\]
Coupling for Unbiased estimation

- Works because (i) $\tilde{X}_{0:T}[k] \overset{d}{=} X_{0:T}[k - 1]$, (ii) coupling time is finite and (iii) the coupling CCPF is for ergodic kernels

$$
\| P_N^k - p(X_{0:T} \mid Y_{0:T}) \|_{tv} \xrightarrow{k \to \infty} 0
$$

- Under weak assumptions

Lee, S., Vihola (2018)

There exists a constant $c$ such that for any $N \geq 2$ the coupling time

$$
P(\tau \geq k) \leq \left( \frac{c}{c + N} \right)^k
$$
Coupling for Unbiased estimation

- Works because (i) $\tilde{X}_{0:T}[k] \overset{d}{=} X_{0:T}[k - 1]$,  
  (ii) coupling time is finite and  
  (iii) the coupling CCPF is for ergodic kernels

$$\|P_N^k - p(x_{0:T} \mid y_{0:T})\|_{tv} \xrightarrow{k \to \infty} 0$$

- Under weak assumptions:
  
  Lee, S., Vihola (2018)

  There exists a constant $c$ such that for any $N \geq 2$ the coupling time

  $$\mathbb{P}(\tau \geq k) \leq \left(\frac{c}{c + N}\right)^k$$

  Under stronger assumptions, the coupling time is stable provided

  $$N \propto 2^T$$
The problem here is we rely one *one-shot* coupling:
if \((X_0:T, \tilde{X}_0:T) \sim \text{CCPF}(x_0:T, \tilde{x}_0:T)\) then

\[
P(X_0:T \neq \tilde{X}_0:T) \leq \frac{c}{N + c}
\]
The Coupled Conditional Backward Particle Filter, or CCBPF (Lee, S., Vihola, 2018)

- The problem here is we rely one one-shot coupling:
  if \((X_0:T, \tilde{X}_0:T) \sim \text{CCPF}(x_0:T, \tilde{x}_0:T)\) then

  \[
P(X_0:T \neq \tilde{X}_0:T) \leq \frac{c}{N + c}
  \]

- Is there a version which will work with a fix number of particles \(N\) irrespective of \(T\)?
The problem here is we rely one *one-shot* coupling:
if \((X_0:T, \tilde{X}_0:T) \sim \text{CCPF}(x_0:T, \tilde{x}_0:T)\) then

\[
P(X_0:T \neq \tilde{X}_0:T) \leq \frac{c}{N + c}
\]

Is there a version which will work with a fix number of particles \(N\) irrespective of \(T\)?

The idea is rely to coupling progressively

\[
\kappa_n = \max \left\{ 0 \leq t \leq T : X_0:t[n] = \tilde{X}_0:t[n] \right\}
\]

With CCPF implemented with *backward sampling* the coupling boundary \(\kappa_n\) drifts to the right!
Let $\tau = \text{first time } n \text{ s.t. } X_{0:T}[n] = \tilde{X}_{0:T}[n]$ then for any positive constants $\alpha > 1$ and $\beta < 1/\alpha$

$$\mathbb{P}(\tau \geq n) \leq \alpha^T \beta^n, \quad \text{for all } n, T$$

if particle number $N$ is large enough
Let $\tau$ = first time $n$ s.t. $X_{0:T}[n] = \tilde{X}_{0:T}[n]$ then for any positive constants $\alpha > 1$ and $\beta < 1/\alpha$

$$P(\tau \geq n) \leq \alpha^T \beta^n, \quad \text{for all } n, T$$

if particle number $N$ is large enough

- Among the corollaries, an important one is coupling for unbiased simulation is assured in time proportional to time series length $T$.

$$P(\text{Coupling time exceeds } T) \xrightarrow{T \to \infty} 0$$
Let $\tau$ = first time $n$ s.t. $X_{0:T}[n] = \tilde{X}_{0:T}[n]$ then for any positive constants $\alpha > 1$ and $\beta < 1/\alpha$

$$\mathbb{P}(\tau \geq n) \leq \alpha^T \beta^n, \quad \text{for all } n, T$$

if particle number $N$ is large enough

- Among the corollaries, an important one is coupling for unbiased simulation is assured in time proportional to time series length $T$.

$$\mathbb{P} (\text{Coupling time exceeds } T) \xrightarrow{T \to \infty} 0$$

- The conjecture that Whiteley’s backward sampling version of Andrieu et al’s CPH is stable for a fix $N$ and any $T$ is true

$$\|P_N^n - p(x_{0:T}|y_{0:T})\|_{tv} \leq \alpha^T \beta^n, \quad (\forall N > N_0, T, n)$$
Boundary against iteration for obsVariance = 1000.0

Statistic: Cost of coupling (first time \( n \) s.t.) \( X_{0:999}[n] = \tilde{X}_{0:999}[n] \)
Statistic: Best $N$ for coupling $X_0 : T[n] = \tilde{X}_0 : T[n]$


5 A. Andrieu, A. Lee and M. Vihola, *Bernoulli*, 2018

