The Hierarchical Particle Filter

Adam M. Johansen and Arnaud Doucet
http://go.warwick.ac.uk/amjohansen/talks

MCMSki V Lenzerheide
7th January 2016

Context & Outline

Filering in State-Space Models:

- SIR Particle Filters [GSS93]
- Block-Sampling Particle Filters [DBS06]

Exact Approximation of Monte Carlo Algorithms:

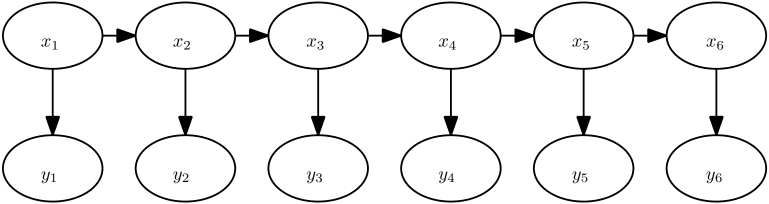
- Particle MCMC [ADH10] and SMC^2 [CJP13]

Exact Approximation and Particle Filters:

- Approximated RBPFs [CSOL11] *Exactly* [JWD12]
- Hierarchical SMC [Here]
- Pseudomarginal State Augmentation: More of the SAME? [Axel Finke; this session]

Hidden Markov Models

Consider:



- Unobserved Markov chain $\{X_n\}$ transition f.
- Observed process $\{Y_n\}$ conditional density g.
- Density:

$$p(x_{1:n}, y_{1:n}) = f_1(x_1)g(y_1|x_1)\prod_{i=2}^{n} f(x_i|x_{i-1})g(y_i|x_i).$$

Sequential Importance Resampling

- Algorithmically, at iteration n:
  - Given $\{W_{n-1}^i, X_{1:n-1}^i\}$ for $i = 1, \ldots, N$:
  - **Resample**, obtaining $\{1/N, \widetilde{X}_{1:n-1}^i\}$.
    - Sample $X_n^i \sim q_n(\cdot|\widetilde{X}_{n-1}^i)$
    - Weight $W_n^i \propto \frac{f(X_n^i|\widetilde{X}_{n-1}^i)g(y_n|X_n^i)}{q_n(X_n^i|\widetilde{X}_{n-1}^i)}$

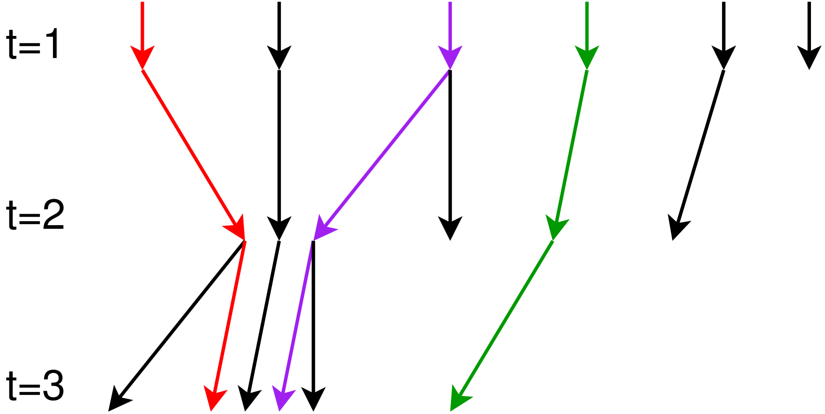
- Actually:
  - Resample efficiently.
  - Only resample when necessary.

Particle MCMC

- MCMC algorithms which employ SMC proposals [ADH10]
- SMC algorithm as a collection of RVs
  - Values
  - Weights
  - Ancestral Lines
- Construct MCMC algorithms:
  - With many auxiliary variables
  - *Exactly* invariant for distribution on extended space
  - Standard MCMC arguments justify strategy
  - SMC^2 employs the same approach within an SMC setting.

- What else does this allow us to do with SMC?

Ancestral Trees



$a_3^1 = 1 \quad a_3^4 = 3 \quad a_2^1 = 1 \quad a_2^4 = 3$

$b_{3,1:3}^2 = (1, 1, 2) \quad b_{3,1:3}^4 = (3, 3, 4) \quad b_{3,1:3}^6 = (4, 5, 6)$

SMC Distributions

Formally gives rise to the **SMC Distribution**:

$$\psi_{n,L}^{M}\left(\overline{\mathbf{a}}_{n-L+2:n}, \overline{\mathbf{x}}_{n-L+1:n}, \overline{k}; x_{n-L}\right)$$
$$= \left[\prod_{i=1}^{M} q(\overline{x}_{n-L+1}^{i} \,|\, \overline{x}_{n-L})\right] \prod_{p=n-L+2}^{n} \left[r(\overline{\mathbf{a}}_p|\overline{\mathbf{w}}_{p-1}) \prod_{i=1}^{M} q\left(\overline{x}_p^i | \overline{x}_{p-1}^{\overline{a}_p^i}\right)\right] r(\overline{k}|\overline{\mathbf{w}}_{\mathbf{n}})$$

and the **conditional SMC Distribution**:

$$\widetilde{\psi}_{n,L}^{M}\left(\widetilde{\mathbf{a}}_{n-L+2:n}^{\ominus k}, \widetilde{\mathbf{x}}_{n-L+1:n}^{\ominus k}; x_{n-L} \,\middle\|\, \widetilde{b}_{n-L+1:n-1}^{k}, k, \widetilde{x}_{n-L+1:n}^{k}\right)$$
$$= \frac{\psi_{n,L}^{M}(\widetilde{\mathbf{a}}_{n-L+2:n}, \widetilde{\mathbf{x}}_{n-L+1:n}, k; x_{n-L})}{q\left(\widetilde{x}_{n-L+1}^{\widetilde{b}_{n,n-L+1}^{k}} | x_{n-L}\right)\left[\prod_{p=n-L+2}^{n} r\left(\widetilde{b}_{n,p}^{k}|\widetilde{\mathbf{w}}_{\mathbf{p-1}}\right) q\left(\widetilde{x}_p^{\widetilde{b}_{n,p}^{k}}|\widetilde{x}_{p-1}^{\widetilde{b}_{n,p-1}^{n}}\right)\right] r(k|\widetilde{\mathbf{w}}_n)}$$

Block Sampling: An Idealised Approach

At time n, given $x_{1:n-1}$; discard $x_{n-L+1:n-1}$:

- Sample from $q(x_{n-L+1:n}|x_{n-L}, y_{n-L+1:n})$.
- Weight with

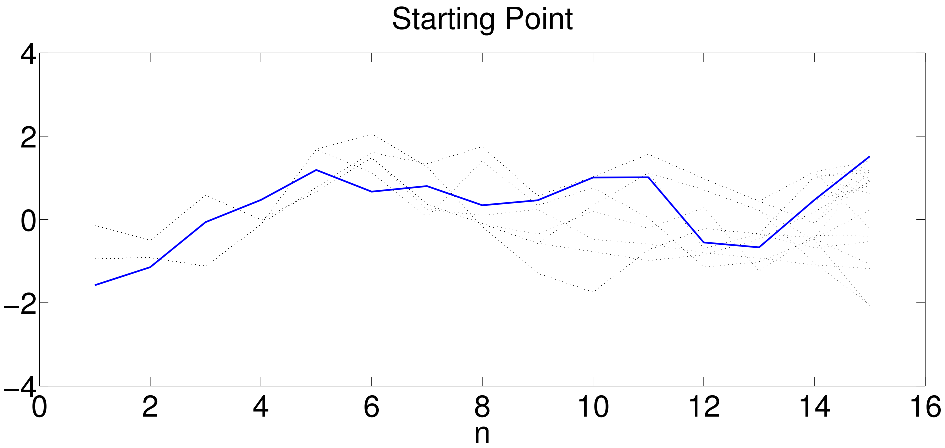
$$W(x_{1:n}) = \frac{p(x_{1:n}|y_{1:n})}{p(x_{1:n-L}|y_{1:n-1})q(x_{n-L+1:n}|x_{n-L}, y_{1:n-L+1:n})}$$

- Optimally,

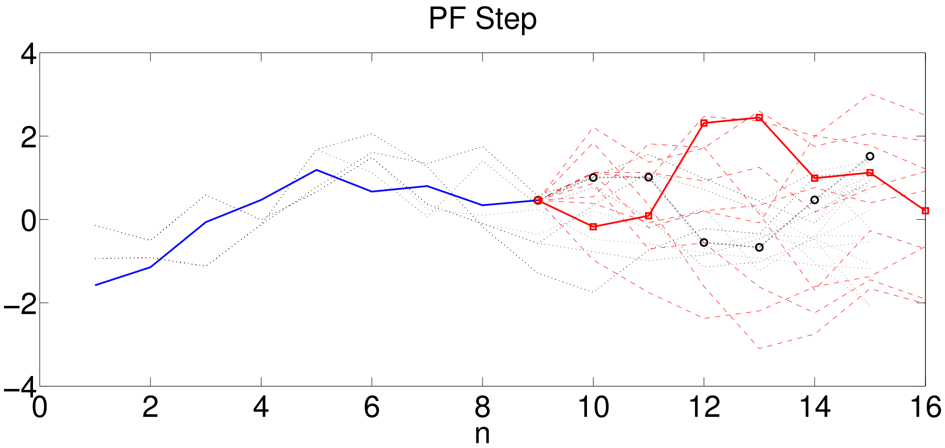
$$\begin{aligned} q(x_{n-L+1:n}|x_{n-L}, y_{n-L+1:n}) =& p(x_{n-L+1:n}|x_{n-L}, y_{n-L+1:n}) \\ W(x_{1:n}) \propto \frac{p(x_{1:n-L}|y_{1:n})}{p(x_{1:n-L}|y_{1:n-1})} =& p(y_n|x_{1:n-L}, y_{n-L+1:n-1}) \end{aligned}$$

- Typically intractable; auxiliary variable approach in [DBS06].

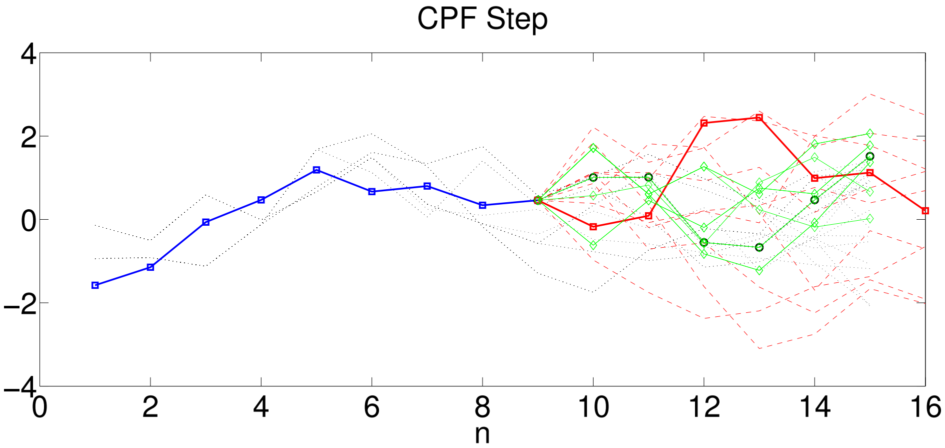
Local Particle Filtering: Current Trajectories



Local Particle Filtering: PF Proposal



Local Particle Filtering: CPF Auxiliary Proposal



Local SMC: Version 1

- Not just a Random Weight Particle Filter.
- Propose from:

$$\mathcal{U}_{1:M}^{\otimes n-1}(b_{1:n-2},k)p(x_{1:n-1}|y_{1:n-1})\psi_{n,L}^{M}(\overline{\mathbf{a}}_{n-L+2:n},\overline{\mathbf{x}}_{n-L+1:n},\overline{k};x_{n-L})$$
$$\widetilde{\psi}_{n-1,L-1}^{M}\left(\widetilde{\mathbf{a}}_{n-L+2:n-1}^{\ominus k},\widetilde{\mathbf{x}}_{n-L+1:n-1}^{\ominus k};x_{n-L}\,||b_{n-L+2:n-1},x_{n-L+1:n-1}\right)$$

- Target:

$$\mathcal{U}_{1:M}^{\otimes n}(b_{1:n-L},\overline{b}_{n,n-L+1:n-1}^{\bar{k}},\bar{k})p(x_{1:n-L},\overline{x}_{n-L+1:n}^{\overline{b}_{n,n-L+1:n}^{\bar{k}}}|y_{1:n})$$
$$\widetilde{\psi}_{n,L}^{M}\left(\overline{\mathbf{a}}_{n-L+2:n}^{\ominus\overline{k}},\overline{\mathbf{x}}_{n-L+1:n}^{\ominus\overline{k}};x_{n-L}\,\middle|\middle|\overline{b}_{n,n-L+1:n}^{\bar{k}},\overline{x}_{n-L+1:n}^{\overline{b}_{n,n-L+1:n}^{\bar{k}}}\right)$$
$$\psi_{n-1,L-1}^{M}\left(\widetilde{\mathbf{a}}_{n-L+2:n-1},\widetilde{\mathbf{x}}_{n-L+1:n-1},k;x_{n-L}\right).$$

- Weight: $\bar{Z}_{n-L+1:n}/\widetilde{Z}_{n-L+1:n-1}$.

Stochastic Volatility Bootstrap Local SMC

- Model:

$$f(x_i|x_{i-1}) = \mathcal{N}\left(\phi x_{i-1}, \sigma^2\right)$$
$$g(y_i|x_i) = \mathcal{N}\left(0, \beta^2 \exp(x_i)\right)$$

- Top Level:
  - Local SMC proposal.
  - Stratified resampling when ESS$< N/2$.
- Local SMC Proposal:
  - Proposal:

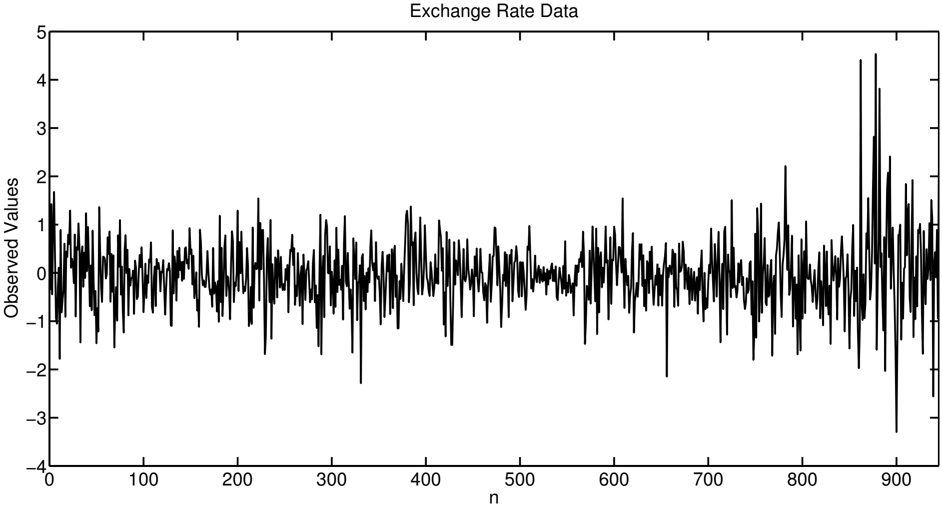
$$q(x_t|x_{t-1}, y_t) = f(x_t|x_{t-1})$$

  - Weighting:

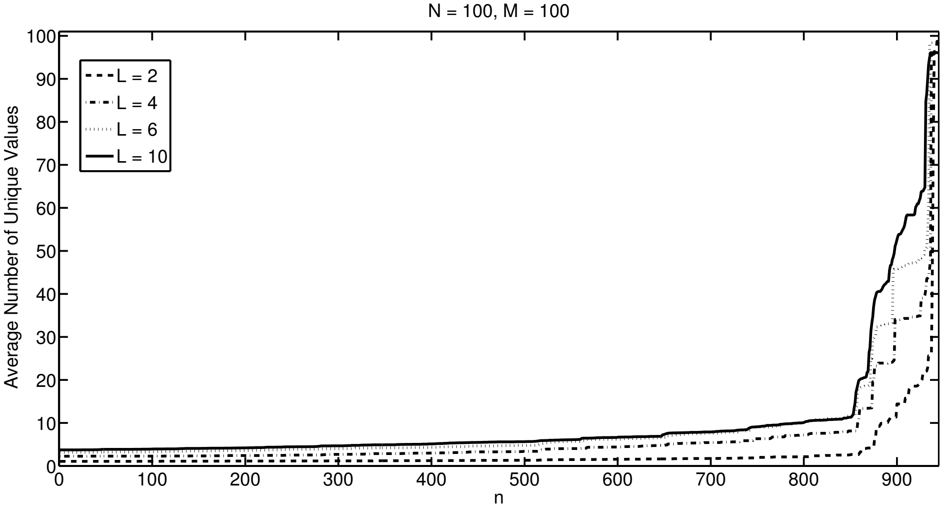
$$W(x_{t-1}, x_t) \propto \frac{f(x_t|x_{t-1})g(y_t|x_t)}{f(x_t|x_{t-1})} = g(y_t|x_t)$$

  - Resample residually every iteration.

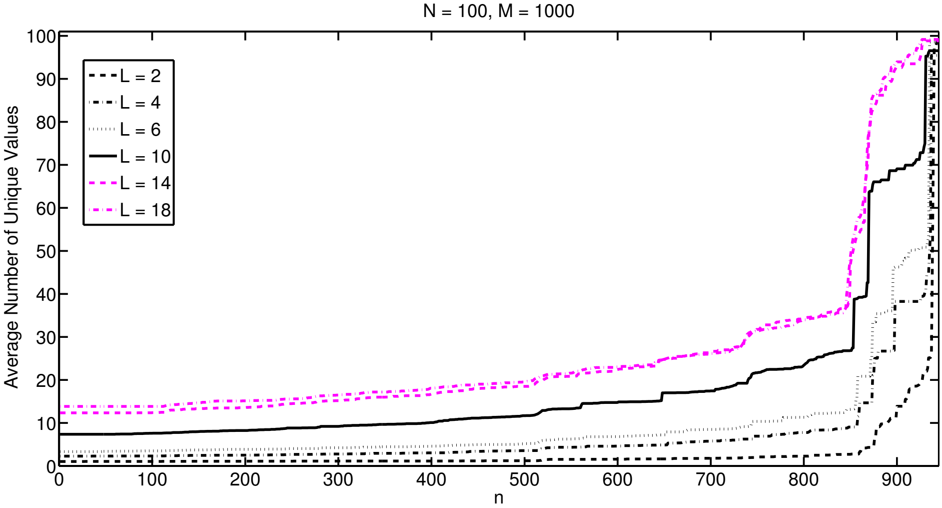
SV Exchange Rata Data



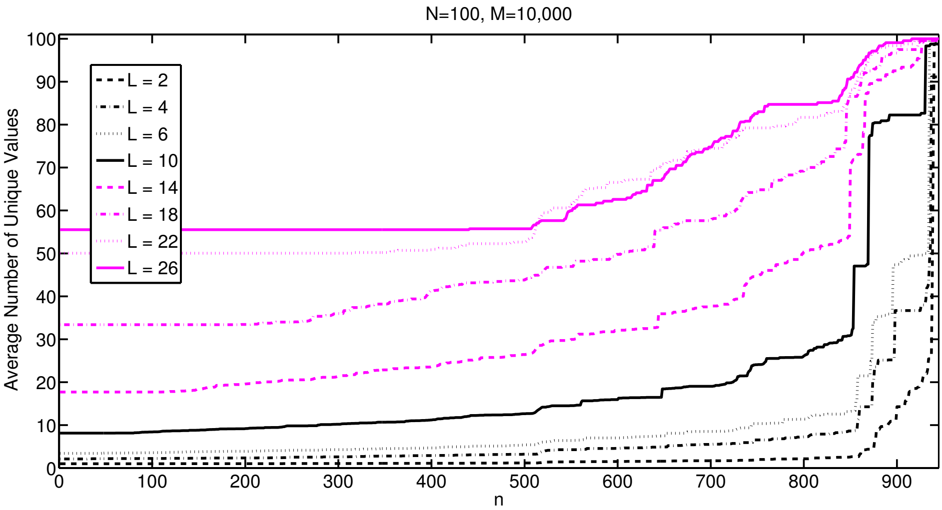
SV Bootstrap Local SMC: M=100



SV Bootstrap Local SMC: M=1000



SV Bootstrap Local SMC: M=10000



Local SMC: Version 2

Problems with this PF+CPF scheme:

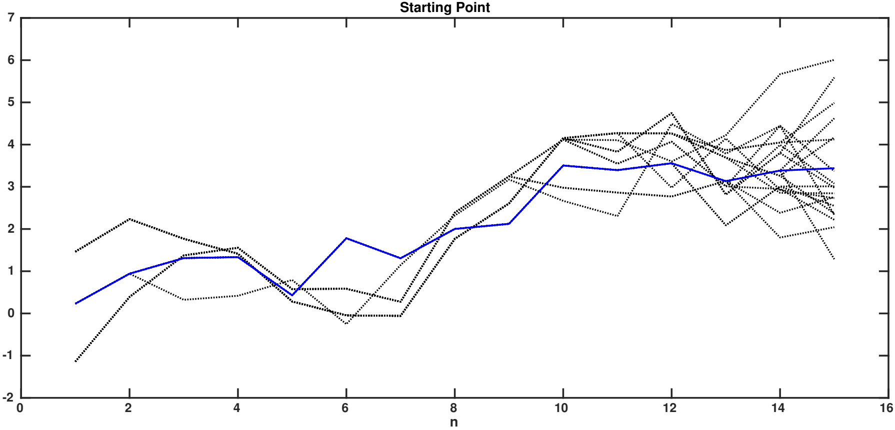
- Expensive to run *2 filters* per proposal...
- and large M is required...
- can't we do better?

Using a non-standard CPF/PF proposal is preferable:

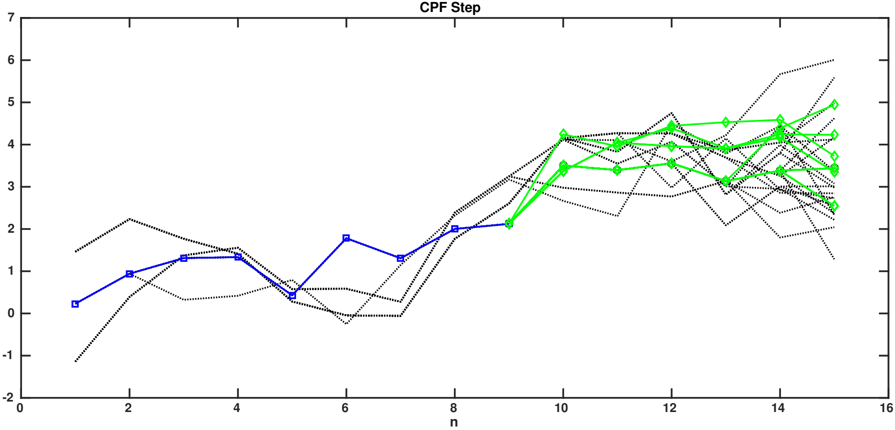
- Running a CPF from time $n-L+1$ to n,
- and extending it's paths with a PF step to $n+1$,
- invoking a careful auxiliary variable construction,
- we reduce computational cost *and* variance.
- Leads to importance weights:

$$\propto \frac{\hat{p}(y_{n-L+1:n}|x_{n-L}^{b^k_{n-1,n-L}})}{\hat{p}(y_{n-L+1:n-1}|x_{n-L}^{b^k_{n-1,n-L}})}$$

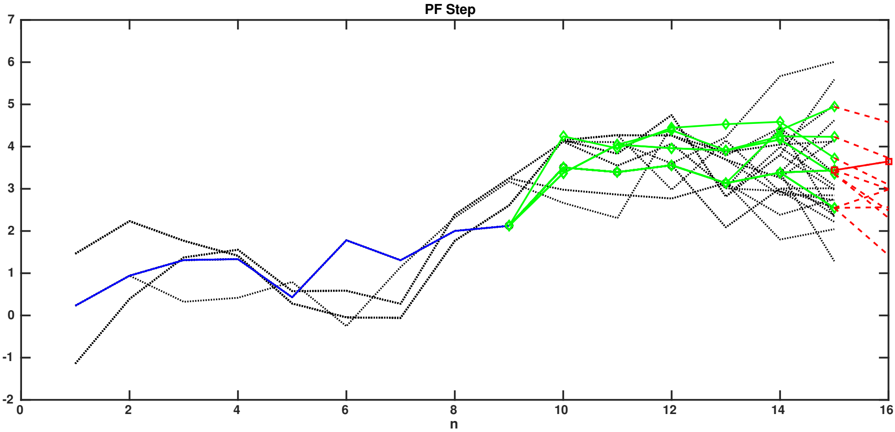
Hierarchical Particle Filtering: Current Trajectories



Hierarchical Particle Filtering: CPF Proposal Component



Hierarchical Particle Filtering: PF Proposal Component



A Toy Model: Linear Gaussian HMM

- Linear, Gaussian state transition:

$$f(x_t|x_{t-1}) = \mathcal{N}(x_t; x_{t-1}, 1)$$

- and likelihood

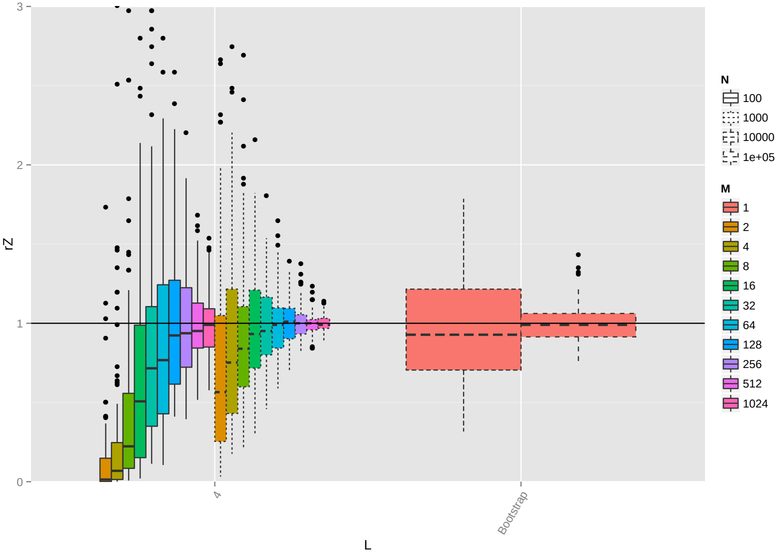
$$g(y_t|x_t) = \mathcal{N}(y_t; x_t, 1)$$

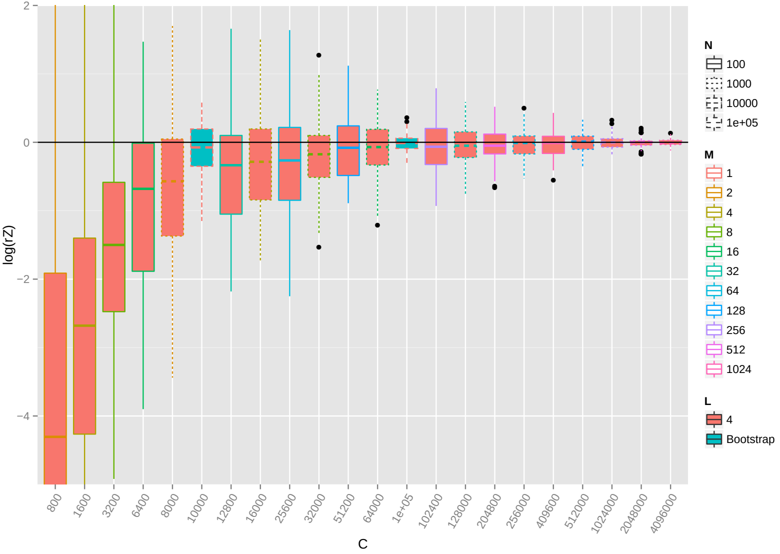
- Analytically: Kalman filter/smoother/etc.
- Simple bootstrap HPF:
  - Local proposal:

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

  - Weighting:

$$W(x_{t-1}, x_t) \propto g(y_t|x_t)$$


3
2
1
0
rZ
4
Bootstrap
L
N
100
1000
10000
1e+05
M
1
2
4
8
16
32
64
128
256
512
1024



log(rZ)
2
0
−2
−4
C
800
1600
3200
6400
8000
10000
12800
16000
25600
32000
51200
64000
1e+05
102400
128000
204800
256000
409600
512000
1024000
2048000
4096000
N
100
1000
10000
1e+05
M
1
2
4
8
16
32
64
128
256
512
1024
L
4
Bootstrap

In Conclusion

- SMC can be used hierarchically.
- Software implementation is not difficult [Joh09, Zho13, Mur13, TCF+14].
- The optimal block-sampling particle filter can be approximated *exactly*
- but at a considerable computational cost.
- Many related things are also possible...
  - EA-RBPFs [JWD12]
  - Tempering in blocks [Joh15]
  - Iterated Particle Filters [GJL15]

References I

[ADH10] C. Andrieu, A. Doucet, and R. Holenstein. Particle Markov chain Monte Carlo. *Journal of the Royal Statistical Society B*, 72(3):269–342, 2010.

[CJP13] N. Chopin, P. Jacob, and O. Papaspiliopoulos. SMC^2: an efficient algorithm for sequential analysis of state space models. *Journal of the Royal Statistical Society B*, 75(3):397–426, 2013.

[CSOL11] T. Chen, T. Schön, H. Ohlsson, and L. Ljung. Decentralized particle filter with arbitrary state decomposition. *IEEE Transactions on Signal Processing*, 59(2):465–478, February 2011.

[DBS06] A. Doucet, M. Briers, and S. Sénécal. Efficient block sampling strategies for sequential Monte Carlo methods. *Journal of Computational and Graphical Statistics*, 15(3):693–711, 2006.

[GJL15] P. Guarniero, A. M. Johansen, and A. Lee. The iterated auxiliary particle filter. ArXiv mathematics e-print 1511.06286, ArXiv Mathematics e-prints, 2015.

[GSS93] N. J. Gordon, S. J. Salmond, and A. F. M. Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proceedings-F*, 140(2):107–113, April 1993.

[Joh09] A. M. Johansen. SMCTC: Sequential Monte Carlo in C++. *Journal of Statistical Software*, 30(6):1–41, April 2009.

[Joh15] A. M. Johansen. On blocks, tempering and particle MCMC for systems identification. In *Proceedings of 17th IFAC Symposium on System Identification*, Beijing, China., 2015. IFAC. To appear. Invited submission.

[JWD12] A. M. Johansen, N. Whiteley, and A. Doucet. Exact approximation of Rao-Blackwellised particle filters. In *Proceedings of 16th IFAC Symposium on Systems Identification*, pages 488–493, Brussels, Belgium, July 2012. IFAC, IFAC.

[Mur13] L. M. Murray. Bayesian state-space modelling on high-performance hardware using LibBi. Mathematics e-print 1306.3277, ArXiv, 2013.

[TCF+14] A. Todeschini, F. Caron, M. Fuentes, P. Legrand, and P. Del Moral. Biips: Software for Bayesian inference with interacting particle systems. Mathematics e-print, ArXiv, 2014.

[Zho13] Y. Zhou. vSMC: Parallel sequential Monte Carlo in C++. Mathematics e-print 1306.5583, ArXiv, 2013.

Key Identity

$$\frac{\psi^M_{n,L}(\mathbf{a}_{n-L+2:n}, \mathbf{x}_{n-L+1:n}, k; x_{n-L})}{p(x_{n-L+1:n}|x_{n-L}, y_{n-L+1:n})\widetilde{\psi}^M_{n,L}(\mathbf{a}^{\ominus k}_{n-L+2:n}, \mathbf{x}^{\ominus k}_{n-L+1:n}, k; x_{n-L} \,||\ldots\,)}$$

$$= \frac{q\left(x^{b^k_{n,n-L+1}}_{n-L+1} | x_{n-L}\right)\left[\prod_{p=n-L+2}^{n} r\left(b^k_{n,p} | \mathbf{w_{p-1}}\right) q\left(x^{b^k_{n,p}}_p | x^{b^n_{n,p-1}}_{p-1}\right)\right] r(k|\mathbf{w}_n)}{p(x_{n-L+1:n}|x_{n-L}, y_{n-L+1:n})}$$

$$= \widehat{Z}_{n-L+1:n} / p(y_{n-L+1:n}|x_{n-L})$$