Bagging and blocking: Inference via particle filters for interacting dynamic systems

Edward Ionides
University of Michigan, Department of Statistics

Statistics seminar series at
Chalmers University of Technology / University of Gothenburg
Tuesday November 23, 2021

Collaborators:
Kidus Asfaw, Ning Ning, Joonha Park and Aaron King

Outline

The curse of dimensionality. Particle filter (PF) methods are effective for inference on low-dimensional nonlinear partially observed stochastic dynamic systems. They scale exponentially badly.

Bagged filters. Combining independent Monte Carlo filters.

- Unadapted bagged filter (UBF)
- adapted bagged filter (ABF)
- adapted bagged filter with intermediate resampling (ABF-IR)

Blocked particle filter (BPF). Theory by Rebeschini and van Handel (2015) and practice independently by Ng et al. (2002).

From filtering to inference. Iterated filtering using stochastically perturbed parameters.

Metapopulation dynamics. Bagged and blocked filters work on collections of weakly coupled populations, in theory and practice.

What is a SpatPOMP?

POMP models are partially observed Markov processes, also known as state space models or hidden Markov models

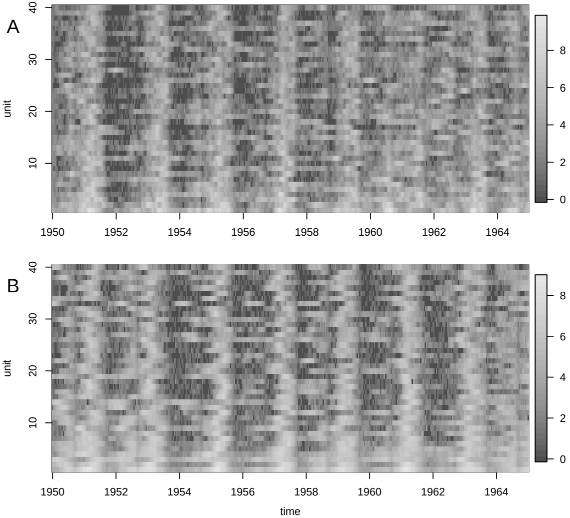
SpatPOMP models are POMP models with a unit structure

Latent Markov process: $X_{u,n} = X_u(t_n), \quad u \in 1\!:\!U, \quad n \in 1\!:\!N$

Observation process: $Y_{u,n}$ depends only on $X_{u,n}$

The units could be a metapopulation, say cities in an epidemic model

$U = 40$ units for a coupled measles SEIR model



A. Measles UK pre-vaccination case reports for the 40 largest cities.
B. Simulated data using Susceptible-Exposed-Latent-Recovered dynamics coupled with a gravity model.

The particle filter (PF)

Evolutionary analogy	**Particle filter algorithm**
Mutation	Predict: stochastic dynamics
$\downarrow$	$\downarrow$
Fitness	Measurement: weight
$\downarrow$	$\downarrow$
Natural selection	Filter: resample

- PF is an evolutionary algorithm with good mathematical properties: an unbiased likelihood estimate and consistent latent state distribution.

The block particle filter (BPF)

Evolutionary analogy

Mutation

$\downarrow$

Fitness
for each chromosome

$\downarrow$

Natural selection
for each chromosome

$\downarrow$

Recombine chromosomes

Block particle filter

Predict: stochastic dynamics

$\downarrow$

Measurement: weight
for each block

$\downarrow$

Filter: resample for each block

$\downarrow$

Recombine blocks

- Blocks in BPF allow recombination (reassortment of chromosomes in sexual reproduction) in the evolutionary analogy.
- Blocks are a partition of the spatial units.

Plug-and-play methods for implicit models

- We address stochastic dynamic models where a simulator is available, but transition densities are not readily accessible.
- These models have been called implicit (Diggle and Gratton, 1984).
- An algorithm that uses a simulator but not transition densities is called plug-and-play (Bretó et al., 2009; He et al., 2010).
- Plug-and-play methods can be applied to implicit models.
- Similar ideas have been called equation-free and likelihood-free.

Unadapted bagged filter: inputs, outputs and implicit loops.

input:
simulator for $f_{\boldsymbol{X}_0}(\boldsymbol{x}_0)$ and $f_{\boldsymbol{X}_n|\boldsymbol{X}_{n-1}}(\boldsymbol{x}_n \,|\, \boldsymbol{x}_{n-1})$
evaluator for $f_{Y_{u,n}|X_{u,n}}(y_{u,n} \,|\, x_{u,n})$
number of replicates, $\mathcal{I}$
neighborhood structure, $B_{u,n}$
data, $\boldsymbol{y}^*_{1:N}$

output:
Log likelihood estimate, $\ell^{\mathsf{MC}} = \sum_{n=1}^{N} \sum_{u=1}^{U} \ell^{\mathsf{MC}}_{u,n}$

implicit loops:
u in $1\!:\!U$, n in $1\!:\!N$, i in $1\!:\!\mathcal{I}$

UBF. Unadapted bagged filter.

Simulate $\boldsymbol{X}_{0:N,i} \sim f_{\boldsymbol{X}_{0:N}}(\boldsymbol{x}_{0:N})$

. Measurement weights, $w^M_{u,n,i} = f_{Y_{u,n}|X_{u,n}}(y^*_{u,n} \,|\, X_{u,n,i})$

Prediction weights, $w^P_{u,n,i} = \prod_{(\tilde{u},\tilde{n}) \in B_{u,n}} w^M_{\tilde{u},\tilde{n},i}$

$\ell^{\mathsf{MC}}_{u,n} = \log\left(\sum_{i=1}^{\mathcal{I}} w^M_{u,n,i} w^P_{u,n,i}\right) - \log\left(\sum_{i=1}^{\mathcal{I}} w^P_{u,n,i}\right)$

- We simulate $\mathcal{I}$ times and do local importance sampling.
- A simple algorithm that provides a starting point for ABF and ABF-IR.

The unadapted bagged filter is not entirely naive

- UBF seems naive. Particle filter (PF) method are well known to scale better with N than unconditional simulations.

- With modern computers, large numbers of simulations are feasible even when U and N are not small.

- Initially we studied UBF as a theoretical toy, since it is relatively easy to show theoretically that it can beat the curse of dimensionality as U increases, for weakly coupled systems. Then we found it is competitive in practice on some models of interest.

Adapted simulation: An easier problem than filtering

- We aim to make each replicate track the data in a weak sense, easier and more scalable than solving the full filtering problem.

- The adapted simulation problem is to draw from $f_{\boldsymbol{X}_n|\boldsymbol{Y}_n,\boldsymbol{X}_{n-1}}\big(\boldsymbol{x}_n \,|\, \boldsymbol{y}^*{}_n, \boldsymbol{x}_{n-1}\big)$.

- The adapted bagged filter (ABF) algorithm uses importance sampling to carry out adapted simulation on each replicate, with a sample size J.

- Importance sampling for adapted simulation does NOT bypass the curse of dimensionality. We later combine it with intermediate resampling to give scalability.

- ABF calculates the likelihood using the proper weight restricted to a neighborhood.

ABF. Adapted bagged filter.

Initialize adapted simulation: $\boldsymbol{X}^{\mathrm{A}}_{0,i} \sim f_{\boldsymbol{X}_0}(\boldsymbol{x}_0)$

For n in $1\!:\!N$

Proposals: $\boldsymbol{X}^{\mathrm{P}}_{n,i,j} \sim f_{\boldsymbol{X}_n|\boldsymbol{X}_{n-1}}\big(\boldsymbol{x}_n \,|\, \boldsymbol{X}^{\mathrm{A}}_{n-1,i}\big)$

Measurement weights: $w^{M}_{u,n,i,j} = f_{Y_{u,n}|X_{u,n}}\big(y^*_{u,n} \,|\, X^{\mathrm{P}}_{u,n,i,j}\big)$

Adapted resampling weights: $w^{\mathrm{A}}_{n,i,j} = \prod_{u=1}^{U} w^{M}_{u,n,i,j}$

Resampling: $\mathbb{P}\big[r(i) = a\big] = w^{\mathrm{A}}_{n,i,a}\Big(\sum_{k=1}^{J} w^{\mathrm{A}}_{n,i,k}\Big)^{-1}$

$\boldsymbol{X}^{\mathrm{A}}_{n,i} = \boldsymbol{X}^{\mathrm{P}}_{n,i,r(i)}$

$$w^{\mathrm{P}}_{u,n,i,j} = \prod_{\tilde{n}=1}^{n-1}\Big[\frac{1}{J}\sum_{k=1}^{J}\prod_{(\tilde{u},\tilde{n})\in B^{[\tilde{n}]}_{u,n}} w^{M}_{\tilde{u},\tilde{n},i,k}\Big]\prod_{(\tilde{u},n)\in B^{[n]}_{u,n}} w^{M}_{\tilde{u},n,i,j}$$

End for

$$\ell^{\mathsf{MC}}_{u,n} = \log\left(\frac{\sum_{i=1}^{\mathcal{I}}\sum_{j=1}^{J} w^{M}_{u,n,i,j} w^{P}_{u,n,i,j}}{\sum_{i=1}^{\mathcal{I}}\sum_{j=1}^{J} w^{P}_{u,n,i,j}}\right)$$

Intermediate resampling

- **Intermediate resampling** splits the time interval between observations into S subintervals.
- Reweighting and/or sampling at each subinterval uses a revised estimate of the anticipated measurement density at the end of the interval called a **guide function**.
- This is applicable to continuous time models.
- Intermediate resampling has useful theoretical and empirical properties (Del Moral and Murray, 2015; Park and Ionides, 2020).
- Intermediate resampling for adapted simulation within ABF gives the ABF-IR algorithm.
- Intermediate resampling within PF gives the guided intermediate resampling filter (GIRF) of Park and Ionides (2020), a generalization of the auxiliary particle filter of Pitt and Shepard (1999).

A guide function for intermediate resampling

- Intermediate resampling with an ideal guide function can beat the curse of dimensionality (Park and Ionides, 2020).

- It is consistent for any guide function, but scalability is limited in practice since the ideal guide is generally intractable.

- In practice, we use moment-matching to approximate the ideal guide for Gaussian models.

- New algorithmic parameters:
  number of intermediate timesteps, S
  measurement variance parameterizations, $\overleftarrow{\mathrm{v}}_{u,n}$ and $\overrightarrow{\mathrm{v}}_{u,n}$
  approximate process and observation mean functions, $\boldsymbol{\mu}$ and $h_{u,n}$

ABF-IR. ABF with intermediate resampling.

Initialize adapted simulation: $\boldsymbol{X}^{\mathrm{A}}_{0,i} \sim f_{\boldsymbol{X}_0}(\boldsymbol{x}_0)$

For n in $1\!:\!N$

Guide simulations: $\boldsymbol{X}^{G}_{n,i,j} \sim f_{\boldsymbol{X}_n|\boldsymbol{X}_{n-1}}\big(\boldsymbol{x}_n \,|\, \boldsymbol{X}^{\mathrm{A}}_{n-1,i}\big)$

Guide variance: $V_{u,n,i} = \mathrm{Var}\big\{h_{u,n}\big(X^{G}_{u,n,i,j}\big), j \text{ in } 1\!:\!J\big\}$

$g^{\mathrm{R}}_{n,0,i,j} = 1$ and $\boldsymbol{X}^{\mathrm{IR}}_{n,0,i,j} = \boldsymbol{X}^{\mathrm{A}}_{n-1,i}$

For s in $1\!:\!S$

Intermediate proposals: $\boldsymbol{X}^{\mathrm{IP}}_{n,s,i,j} \sim f_{\boldsymbol{X}_{n,s}|\boldsymbol{X}_{n,s-1}}\big(\,\cdot\,|\boldsymbol{X}^{\mathrm{IR}}_{n,s-1,i,j}\big)$

$\boldsymbol{\mu}^{\mathrm{IP}}_{n,s,i,j} = \boldsymbol{\mu}\big(\boldsymbol{X}^{\mathrm{IP}}_{n,s,i,j}, t_{n,s}, t_n\big)$

$V^{\mathrm{meas}}_{u,n,s,i,j} = \overrightarrow{\mathrm{v}}_u(\theta, \mu^{\mathrm{IP}}_{u,n,s,i,j})$, $\quad V^{\mathrm{proc}}_{u,n,s,i} = V_{u,n,i}\,\big(t_n - t_{n,s}\big)\Big/\big(t_n - t_{n,0}\big)$

$\theta_{u,n,s,i,j} = \overleftarrow{\mathrm{v}}_u\big(V^{\mathrm{meas}}_{u,n,s,i,j} + V^{\mathrm{proc}}_{u,n,s,i},\, \mu^{\mathrm{IP}}_{u,n,s,i,j}\big)$

$g_{n,s,i,j} = \prod_{u=1}^{U} f_{Y_{u,n}|X_{u,n}}\big(y^*_{u,n} \,|\, \mu^{\mathrm{IP}}_{u,n,s,i,j}\,; \theta_{u,n,s,i,j}\big)$

Guide weights: $w^{G}_{n,s,i,j} = g_{n,s,i,j} \big/ g^{\mathrm{R}}_{n,s-1,i,j}$

Resampling: $\mathbb{P}\big[r(i,j) = a\big] = w^{G}_{n,s,i,a}\Big(\sum_{k=1}^{J} w^{G}_{n,s,i,k}\Big)^{-1}$

$\boldsymbol{X}^{\mathrm{IR}}_{n,s,i,j} = \boldsymbol{X}^{\mathrm{IP}}_{n,s,i,r(i,j)}$ and $g^{\mathrm{R}}_{n,s,i,j} = g_{n,s,i,r(i,j)}$

End For

Set $\boldsymbol{X}^{\mathrm{A}}_{n,i} = \boldsymbol{X}^{\mathrm{IR}}_{n,S,i,1}$

Measurement weights: $w^{M}_{u,n,i,j} = f_{Y_{u,n}|X_{u,n}}\big(y^*_{u,n} \,|\, X^{G}_{u,n,i,j}\big)$

$$w^{\mathrm{P}}_{u,n,i,j} = \prod_{\tilde{n}=1}^{n-1}\Big[\frac{1}{J}\sum_{a=1}^{J} \prod_{(\tilde{u},\tilde{n})\in B^{[\tilde{n}]}_{u,n}} w^{M}_{\tilde{u},\tilde{n},i,a}\Big] \prod_{(\tilde{u},n)\in B^{[n]}_{u,n}} w^{M}_{\tilde{u},n,i,j}$$

End for

$$\ell^{\mathsf{MC}}_{u,n} = \log\left(\frac{\sum_{i=1}^{\mathcal{I}}\sum_{j=1}^{J} w^{M}_{u,n,i,j} w^{P}_{u,n,i,j}}{\sum_{i=1}^{\mathcal{I}}\sum_{j=1}^{J} w^{P}_{u,n,i,j}}\right)$$

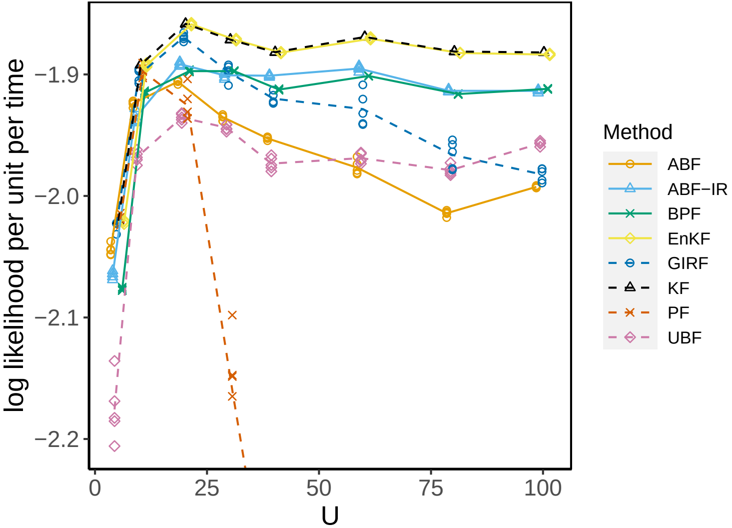
Software for SpatPOMP models

- We use the `asif`, `asifir`, `bpfilter`, `enkf` and `girf` implementations in the R package `spatPomp` (Asfaw et al., 2019).

- `spatPomp` offers a class '`spatPomp`' that extends the '`pomp`' class for POMP models in the R package `pomp` (King et al., 2016).

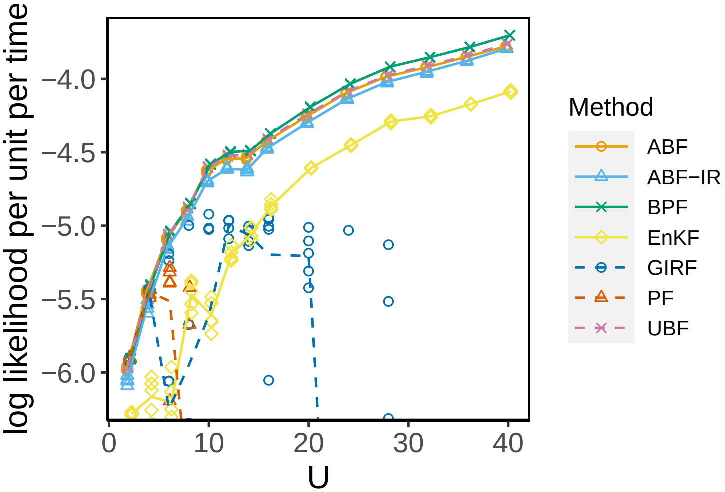
- All methods available in `pomp` can formally be applied to '`spatPomp`' objects, though they may not be practically effective for spatiotemporal POMPs.

Filtering U-dimensional correlated Brownian motion



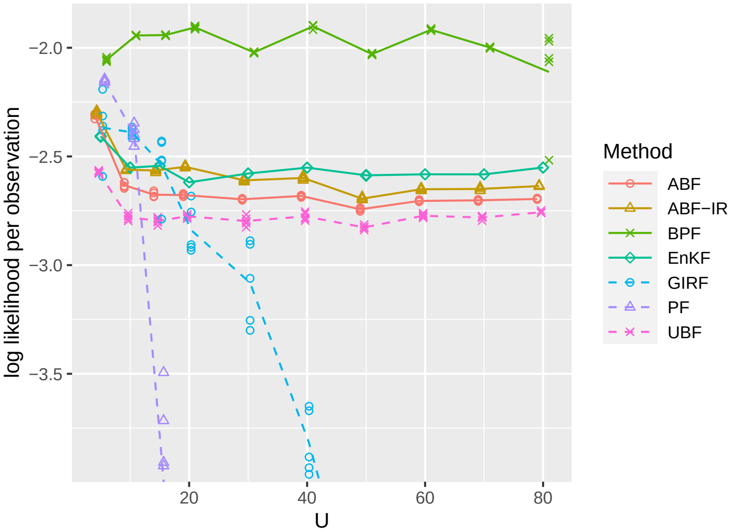
$$\mathrm{Cov}\left(X_{u,n} - X_{u,n-1}, X_{\tilde{u},n} - X_{\tilde{u},n-1}\right) \sim 0.4^{|u-\tilde{u}|}$$

Filtering U units of a coupled measles SEIR model



Simulated data using a gravity model with geography, demography and transmssion parameters corresponding to UK pre-vaccination measles.

Filtering U units of Lorenz 96 toy atmospheric model



$$dX_u(t) = \Big\{X_{u-1}(t)\big(X_{u+1}(t) - X_{u-2}(t)\big) - X_u(t) + F\Big\}dt + \sigma\, dB_u(t)$$

From filtering to parameter inference

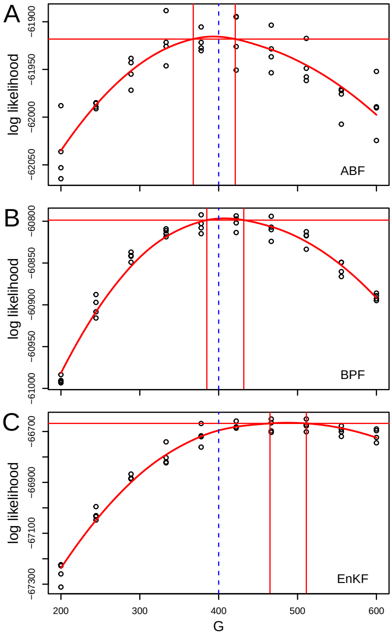
- Log likelihood evaluation in principle enables likelihood-based or Bayesian inference.

- Iterated filtering for PF (Ionides et al., 2015) and GIRF (Park and Ionides, 2020) maximizes the likelihood by randomly perturbing the parameters.

- Particle Markov chain Monte Carlo can be applied with any likelihood estimate (Andrieu et al., 2010). It is numerically intractable when Monte Carlo estimates are costly and noisy.

- Iterated filtering is harder for bagged filters; it is possible but expensive (Ionides et al., 2021).
- Iterated filtering works well for BPF when parameters are unit-specific, i.e., each city has its own parameters (Ning and Ionides, 2021). It also can work with shared parameters (current unpublished work).

Measles likelihood slices for G



Simulating 15 year of data from $U = 40$ cities for the measles model. Slice likelihood, varying G with other paramters fixed at the truth.

A. Evaluation using ABF.

B. Evaluation using BPF.

C. Evaluation using EnKF.

Theorem

Let ℓ^{MC} denote the Monte Carlo likelihood approximation constructed by UBF, ABF or ABF-IR. Consider a limit with a growing number of replicates, $\mathcal{I} \to \infty$. Suppose regularity assumptions listed in the paper. There are quantities $\epsilon(U,N) = O(1)$ and $V(U,N) = O(U^2N^2)$ such that

$$\mathcal{I}^{1/2}\big[\ell^{MC} - \ell - \epsilon U N\big] \xrightarrow[\mathcal{I}\to\infty]{d} \mathcal{N}\big[0, V\big],$$

where $\xrightarrow[\mathcal{I}\to\infty]{d}$ denotes convergence in distribution and $\mathcal{N}[\mu,\Sigma]$ is the normal distribution with mean μ and variance Σ. If an additional spatiotemporal mixing assumption holds, we obtain an improved variance bound

$$V(U,N) = O(UN)$$

References I

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

Asfaw, K., Ionides, E. L., and King, A. A. (2019). `spatPomp`: R package for statistical inference for spatiotemporal partially observed Markov processes. *https://github.com/kidusasfaw/spatPomp*.

Bretó, C., He, D., Ionides, E. L., and King, A. A. (2009). Time series analysis via mechanistic models. *Annals of Applied Statistics*, 3:319–348.

Del Moral, P. and Murray, L. M. (2015). Sequential Monte Carlo with highly informative observations. *Journal on Uncertainty Quantification*, 3:969–997.

Diggle, P. J. and Gratton, R. J. (1984). Monte Carlo methods of inference for implicit statistical models. *Journal of the Royal Statistical Society, Series B (Statistical Methodology)*, 46:193–227.

References II

He, D., Ionides, E. L., and King, A. A. (2010). Plug-and-play inference for disease dynamics: Measles in large and small towns as a case study. *Journal of the Royal Society Interface*, 7:271–283.

Ionides, E. L., Asfaw, K., Park, J., and King, A. A. (2021). Bagged filters for partially observed interacting systems. *Journal of the American Statistical Association*, 0(ja):1–33.

Ionides, E. L., Nguyen, D., Atchadé, Y., Stoev, S., and King, A. A. (2015). Inference for dynamic and latent variable models via iterated, perturbed Bayes maps. *Proceedings of the National Academy of Sciences of the USA*, 112:719–724.

King, A. A., Nguyen, D., and Ionides, E. L. (2016). Statistical inference for partially observed Markov processes via the R package pomp. *Journal of Statistical Software*, 69:1–43.

References III

Ng, B., Peshkin, L., and Pfeffer, A. (2002). Factored particles for scalable monitoring. *Proceedings of the 18th Conference on Uncertainty and Artificial Intelligence*, pages 370–377.

Ning, N. and Ionides, E. L. (2021). Iterated block particle filter for high-dimensional parameter learning: Beating the curse of dimensionality. *arXiv:2110.10745*.

Park, J. and Ionides, E. L. (2020). Inference on high-dimensional implicit dynamic models using a guided intermediate resampling filter. *Statistics and Computing*, 30(5):1497–1522.

Pitt, M. K. and Shepard, N. (1999). Filtering via simulation: Auxillary particle filters. *Journal of the American Statistical Association*, 94:590–599.

Rebeschini, P. and van Handel, R. (2015). Can local particle filters beat the curse of dimensionality? *The Annals of Applied Probability*, 25:2809–2866.