Supplement to "Bagged filters for partially observed interacting systems"

E. L. Ionides, K. Asfaw, J. Park and A. A. King

November 24, 2021

Supplementary Content

S1 A generalization to models without latent unit structure

Variations of the algorithms in the main text apply when there is no latent unit structure. In this case, the observation vector $\boldsymbol{Y}_n = (Y_{1,n}, \dots, Y_{U,n})$ consists of a collection of measurements on a general latent vector X_n. We may have the structure that $Y_{1,n}, \dots, Y_{U,n}$ are conditionally independent given X_n, but even this is not essential to the approach. This is most readily seen in the context of the unadapted bagged filter, giving rise to the generalized unadapted bagged filter (G-UBF) algorithm defined as follows.

Algorithm S1: Generalized unadapted bagged filter (G-UBF).

```
input: simulator for $f_{X_n|X_{n-1}}(x_n \,|\, x_{n-1})$ and $f_{X_0}(x_0)$; evaluator for $f_{Y_{u,n}|X_n}(y_{u,n} \,|\, x_n)$;
       data, $\boldsymbol{y}^*_{1:N}$; number of replicates, $\mathcal{I}$; neighborhood structure, $B_{u,n}$
for $i$ in $1\!:\!\mathcal{I}$ do
    simulate $X^{\mathrm{sim}}_{0:N,i}$ from the dynamic model, for $n$ in $1\!:\!N$
    prediction weights, $w^P_{u,n,i} = f_{Y_{B_{u,n}}|X_{1:n}}(y^*_{B_{u,n}} \,|\, X^{\mathrm{sim}}_{1:n,i})$ for $u$ in $1\!:\!U$, $n$ in $1\!:\!N$
    measurement weights, $w^M_{u,n,i} = f_{Y_{u,n}|X_n,Y_{B_{u,n}}}(y^*_{u,n} \,|\, X^{\mathrm{sim}}_{n,i}, y^*_{B_{u,n}})$ for $u$ in $1\!:\!U$, $n$ in $1\!:\!N$
end
$\ell^{\mathrm{MC}}_{u,n} = \log\left(\sum_{i=1}^{\mathcal{I}} w^M_{u,n,i}\, w^P_{u,n,i}\right) - \log\left(\sum_{\tilde{i}=1}^{\mathcal{I}} w^P_{u,n,\tilde{i}}\right)$ for $u$ in $1\!:\!U$, $n$ in $1\!:\!N$
output: log likelihood estimate, $\ell^{\mathrm{MC}} = \sum_{n=1}^{N}\sum_{u=1}^{U} \ell^{\mathrm{MC}}_{u,n}$
```

The algorithm G-UBF operates on an arbitrary POMP model. G-UBF therefore provides a potential approach to extending methodologies from SpatPOMP models to models that have some similarity to a SpatPOMP without formally meeting the definition. For example, there may be collections of interacting processes at different spatial scales in a spatiotemporal system. Alternatively, the potential outcomes of the latent process may vary between spatial units, such as when modeling interactions between terrestrial and aquatic ecosystems. We do not further explore G-UBF here.

S2 Adapted simulation for an Euler approximation

We investigate the adapted simulation process by considering a continuous-time limit where it becomes a diffusion process. We find that adapted simulation can effectively track the latent process when the measurement error is on an appropriate scale. However, when the measurement error is large compared to the latent process noise, adapted simulation can fail in situations where filtering succeeds. We work with a one-dimensional POMP model having a latent process constructed as an Euler approximation,

$$X_{n+1} \quad = \quad X_n + \mu(X_n)\delta + \sigma\sqrt{\delta}\epsilon_{n+1}, \tag{S1}$$

which provides a numerical solution to a one-dimensional stochastic differential equation,

$$dX(t) \quad = \quad \mu\big(X(t)\big)\, dt + \sigma\, dU(t),$$

where $\{U(t)\}$ is a standard Brownian motion. We will consider several different measurement processes.

S2.1 Measurement error on the same scale as the process noise

Here, we consider the measurement model

$$Y_{n+1} = \mu(X_n)\delta + \sigma\sqrt{\delta}\epsilon_{n+1} + \tau\sqrt{\delta}\,\eta_{n+1}. \tag{S2}$$

This is an approximation to the increment $Y(t+\delta)-Y(t)$ of a continuous time measurement model

$$dY(t) = dX(t) + \tau\, dV(t), \tag{S3}$$

where $\{V(t)\}$ is a standard Brownian motion independent of $\{U(t)\}$. The measurement model (S3) makes inference on $X(t)$ given $Y(t)$ a continuous time version of the filtering problem. A feature of this model is that $Y(t)$ does not directly track the level of the state, since the solution with initial conditions $Y(t_0) = X(t_0)$ and $V(t_0) = 0$ is

$$Y(t) = X(t) + \tau V(t).$$

The measurement error, $\tau V(t)$, has variance $\tau^2 t$ that increases with t. However, under appropriate conditions, information on changes in $\{X(t)\}$ obtained via $\{Y(t)\}$ are enough to track $X(t)$ indirectly via the filtering equations. For the POMP given by (S1) and (S2), we can calculate exactly the adapted simulation distribution $f_{X_{n+1}|Y_{n+1},X_n}$. It is convenient to work conditionally on X_n, allowing us to treat X_n and $\mu(X_n)$ as constants, with X_{n+1} and Y_{n+1} therefore being jointly normally distributed. A Gaussian distribution calculation then gives the conditional moments. First, we find

$$\begin{aligned}
\mathbb{E}\big[X_{n+1}|Y_{n+1},X_n\big] \quad &= \quad X_n + \mu(X_n)\delta + \mathbb{E}\big[\sigma\sqrt{\delta}\epsilon_{n+1}\,\big|\,\sigma\sqrt{\delta}\epsilon_{n+1} + \tau\sqrt{\delta}\eta_{n+1}\big] \\
&= \quad X_n + \mu(X_n)\delta + \frac{\sigma^2}{\sigma^2+\tau^2}\big(\sigma\sqrt{\delta}\epsilon_{n+1} + \tau\sqrt{\delta}\eta_{n+1}\big) \\
&= \quad X_n + \mu(X_n)\delta + \frac{\sigma^2}{\sigma^2+\tau^2}\big(Y_{n+1} - \mu(X_n)\delta\big).
\end{aligned}$$

Then,

$$\begin{aligned}
\mathrm{Var}\big[X_{n+1}\,\big|\,Y_{n+1},X_n\big] &= \mathrm{Var}\big[\sigma\sqrt{\delta}\epsilon_{n+1}\,\big|\,\sigma\sqrt{\delta}\epsilon_{n+1}+\tau\sqrt{\delta}\eta_{n+1}\big] \\
&= \sigma^2\delta-\frac{\sigma^4\delta^2}{\sigma^2\delta+\tau^2\delta} \\
&= \delta\frac{\sigma^2\tau^2}{\sigma^2+\tau^2}.
\end{aligned}$$

Call the adapted simulation process $\{A_n, n=1,2,\dots\}$, defined conditionally on $\{Y_n, n=1,2,\dots\}$. We see from the above calculation that A_n can be constructed by the recursion

$$\begin{aligned}
A_{n+1} &= A_n+\mu(A_n)\delta+\frac{\sigma^2}{\sigma^2+\tau^2}\Big(\mu(X_n)\delta+\sigma\sqrt{\delta}\,\epsilon_{n+1}+\tau\sqrt{\delta}\,\eta_{n+1}-\mu(A_n)\,\delta\Big) \\
&\qquad +\frac{\sigma\tau}{\sqrt{\sigma^2+\tau^2}}\sqrt{\delta}\,\zeta_{n+1}
\end{aligned}$$

where $\{\zeta_n\}$ is an iid standard normal sequence independent of $\{\epsilon_n,\eta_n\}$. To study how well the adapted simulation tracks $\{X_n\}$, we subtract X_{n+1} from both sides to get

$$\begin{aligned}
[A_{n+1}-X_{n+1}] &= [A_n-X_n]+[\mu(A_n)-\mu(X_n)]\delta-\sigma\sqrt{\delta}\,\epsilon_{n+1} \\
&\quad +\frac{\sigma^2}{\sigma^2+\tau^2}\Big([\mu(X_n)-\mu(A_n)]\delta+\sigma\sqrt{\delta}\,\epsilon_{n+1}+\tau\sqrt{\delta}\,\eta_{n+1}\Big)+\frac{\sigma\tau}{\sqrt{\sigma^2+\tau^2}}\sqrt{\delta}\,\zeta_{n+1} \\
&= [A_n-X_n]+\frac{2\sigma^2+\tau^2}{\sigma^2+\tau^2}[\mu(X_n)-\mu(A_n)]\delta \\
&\quad +\frac{\sigma^2\tau\sqrt{\delta}\eta_{n+1}-\sigma\tau^2\sqrt{\delta}\epsilon_{n+1}}{\sigma^2+\tau^2}+\frac{\sigma\tau}{\sqrt{\sigma^2+\tau^2}}\sqrt{\delta}\,\zeta_{n+1}.
\end{aligned}$$

A_n tracks X_n when the process $\{A_n-X_n, n=1,2,\dots\}$ is stable. This happens when $\mu(x)-\mu(y)$ is negative when x is sufficiently larger than y. For example, a stable autoregressive process with $\mu(x)=-ax$ gives a stable adapted filter process.

S2.2 Independent measurement error on a scale that gives a finite limiting amount of information about $X(t)$ from measurements on a unit time interval

We now consider the measurement model

$$\begin{aligned}
Y_{n+1} &= X_{n+1}+\frac{\tau}{\sqrt{\delta}}\eta_{n+1} \\
&= X_n+\mu(X_n)\,\delta+\sigma\sqrt{\delta}\,\epsilon_{n+1}+\frac{\tau}{\sqrt{\delta}}\eta_{n+1}, \qquad \text{(S4)}
\end{aligned}$$

where $\{\epsilon_n,\eta_n\}$ is a collection of independent standard normal random variables. The conditional mean is now

$$\begin{aligned}
\mathbb{E}\big[X_{n+1}|Y_{n+1},X_n\big]. &= X_n+\mu(X_n)\delta+\mathbb{E}\Big[\sigma\sqrt{\delta}\epsilon_{n+1}\,\Big|\,\sigma\sqrt{\delta}\epsilon_{n+1}+\frac{\tau}{\sqrt{\delta}}\eta_{n+1}\Big] \\
&= X_n+\mu(X_n)\delta+\frac{\sigma^2\delta}{\sigma^2\delta+\tau^2/\delta}\big(\sigma\sqrt{\delta}\epsilon_{n+1}+\tau\sqrt{\delta}\,\eta_{n+1}\big) \qquad \text{(S5)}
\end{aligned}$$

Using (S4) and (S5) gives

$$\mathbb{E}\big[X_{n+1}|Y_{n+1},X_n\big] \;=\; X_n+\mu(X_n)\delta+\frac{\sigma^2\delta^2}{\sigma^2\delta^2+\tau^2}\Big(Y_{n+1}-X_n-\mu(X_n)\,\delta\Big).$$

In the limit as $\delta\to 0$, the contribution from the measurement is order δ^2 and is therefore negligible. Although the observation process is meaningfully informative about the latent process, the adapted simulation fails to track the latent process in this limit. Intuitively, this is because the adapted simulation is trying to track differences in the latent process, but for this model the signal to noise ratio for the difference in each interval of length δ tends to zero.

S2.3 Independent measurements of the latent process with measurement error on a scale that gives a useful adapted process as $\delta \to 0$

We now consider the measurement model

$$\begin{aligned} Y_{n+1} &= X_{n+1}+\tau\eta_{n+1} \\ &= X_n+\mu(X_n)\delta+\sigma\sqrt{\delta}\epsilon_{n+1}+\tau\eta_{n+1}. \end{aligned} \tag{S6}$$

The conditional mean is now

$$\begin{aligned} \mathbb{E}\big[X_{n+1}|Y_{n+1},X_n\big]. &= X_n+\mu(X_n)\,\delta+\mathbb{E}\big[\sigma\sqrt{\delta}\epsilon_{n+1}\,\big|\,\sigma\sqrt{\delta}\epsilon_{n+1}+\tau\eta_{n+1}\big] \\ &= X_n+\mu(X_n)\delta+\frac{\sigma^2\delta}{\sigma^2\delta+\tau^2}\big(\sigma\sqrt{\delta}\epsilon_{n+1}+\tau\eta_{n+1}\big) \end{aligned} \tag{S7}$$

Using (S6) and (S7) gives

$$\begin{aligned} \mathbb{E}\big[X_{n+1}|Y_{n+1},X_n\big] &= X_n+\mu(X_n)\delta+\frac{\sigma^2\delta}{\sigma^2\delta+\tau^2}\Big(Y_{n+1}-X_n-\mu(X_n)\delta\Big) \\ &= X_n+\mu(X_n)\delta+\frac{\sigma^2}{\tau^2}\delta\Big(Y_{n+1}-X_n-\mu(X_n)\,\delta\Big)+o(\delta) \end{aligned}$$

In the limit as $\delta\to 0$, the adapted simulation has a diffusive drift toward the value of the latent process.

For disease models, incidence data can arguably be considered as noisy measurements of the change of a state variable (number of susceptibles) that is not directly measured. This could correspond to a situation where the measurement error is on the same scale as the process noise (Subsection S2.1). Alternatively, we could think of weekly aggregated incidence as a noisy measurement of the infected class, in which case the measurement error could match the scaling in Subsection S2.3.

The model in Subsection S2.2 is a cautionary tale, warning us against carrying out adapted simulation on short time intervals. An interpretation is that one should not carry out adapted simulation unless a reasonable amount of information has accrued. When each observation has low information, a particle filter may enable solution to the filtering problem without particle depletion. It is when the data are highly informative that the curse of dimensionality makes basic particle filters ineffective, opening up demand for alternative methods.

We are now in a better position to understand why it may be appropriate to keep many particle representations at intermediate timesteps while resampling down to a single representative at each observation time, as ABF and ABF-IR do. We have seen that adaptive simulation can fail when observations occur frequently. Resampling down to a single particle too often can lose the ability for the adapted process to track the latent process. This implies that adapted simulation should not be relied upon more than necessary to ameliorate the curse of dimensionality: once proper importance sampling for filtering problem becomes tractable in a sufficiently small spatiotemporal neighborhood, one should maintain weighted particles on this spatiotemporal scale rather than resorting to adapted simulation.

S3 UBF convergence: Proof of Theorem 1

We consider a collection of models $f_{\boldsymbol{X}_{0:N},\boldsymbol{Y}_{1:N}}$ and data $\boldsymbol{y}^*_{1:N}$ defined for each (U,N). These models and datasets are not required to have any nesting relationship, so we do not insist that $X_{1,1}$ or $y^*_{1,1}$ should be the same for $(U,N)=(10,10)$ as for $(U,N)=(100,100)$. Formally, we define probability and expectation on a product space of the stochastic model and Monte Carlo outcomes. Monte Carlo quantities such as the output of the UBF, ABF and ABF-IR algorithms depend on the data but not on any random variables constructed under the model. As a consequence, we can use $\mathbb{E}$ to correspond both to expectations over Monte Carlo stochasticity (for Monte Carlo quantities) and model stochasticity (for random variables constructed under the model). We suppress discussion of measurability by assuming that all functions considered have appropriate measurability properties. We restate the assumptions and statement for Theorem 1.

Assumption A1. *There is an $\epsilon_{\mathrm{A1}}>0$, independent of U and N, and a collection of neighborhoods $\{B_{u,n}\subset A_{u,n}, u\in 1\!:\!U, n\in 1\!:\!N\}$ such that, for all u and n, any bounded real-valued function $|h(x)|\le 1$, and any value of $x_{B^c_{u,n}}$,*

$$\left|\int h(x_{u,n}) f_{X_{u,n}|Y_{B_{u,n}},X_{B^c_{u,n}}}(x_{u,n}\,|\,y^*_{B_{u,n}},x_{B^c_{u,n}})\,dx_{u,n} - \int h(x_{u,n}) f_{X_{u,n}|Y_{B_{u,n}}}(x_{u,n}\,|\,y^*_{B_{u,n}})\,dx_{u,n}\right| < \epsilon_{\mathrm{A1}}. \qquad \text{(S8)}$$

We use the total variation bound in Assumption A1 via the following Proposition S1, which replaces conditioning on $X_{B^c_{u,n}}$ with conditioning on $Y_{B^c_{u,n}}$. The bound in (S9) could be used in place of Assumption A1.

Proposition S1. *Under the conditions of Assumption A1, (S8) implies*

$$\left|\int h(x_{u,n}) f_{X_{u,n}|Y_{A_{u,n}}}(x_{u,n}\,|\,y^*_{A_{u,n}})\,dx_{u,n} - \int h(x_{u,n}) f_{X_{u,n}|Y_{B_{u,n}}}(x_{u,n}\,|\,y^*_{B_{u,n}})\,dx_{u,n}\right| < \epsilon_{\mathrm{A1}} \qquad \text{(S9)}$$

Proof. For notational compactness, we suppress the arguments $x_{u,n}$, $x_{B^c_{u,n}}$, $y^*_{A_{u,n}}$, $y^*_{B_{u,n}}$ matching the subscripts of conditional densities. Using the conditional independence of the measurements given the latent process, we calculate

$$\begin{aligned}
&\left| \int h(x_{u,n}) f_{X_{u,n}|Y_{A_{u,n}}}\, dx_{u,n} - \int h(x_{u,n}) f_{X_{u,n}|Y_{B_{u,n}}}\, dx_{u,n} \right| \\
&\quad = \quad \left| \int \left\{ \int h(x_{u,n}) f_{X_{u,n}|Y_{B_{u,n}},X_{B^c_{u,n}}}\, dx_{u,n} - \int h(x_{u,n}) f_{X_{u,n}|Y_{B_{u,n}}}\, dx_{u,n} \right\} f_{X_{B^c_{u,n}}\,|\,Y_{A_{u,n}}}\, dx_{B^c_{u,n}} \right| \\
&\quad \leq \quad \int \left| \int h(x_{u,n}) f_{X_{u,n}|Y_{B_{u,n}},X_{B^c_{u,n}}}\, dx_{u,n} - \int h(x_{u,n}) f_{X_{u,n}|Y_{B_{u,n}}}\, dx_{u,n} \right| f_{X_{B^c_{u,n}}\,|\,Y_{A_{u,n}}}\, dx_{B^c_{u,n}} \\
&\quad < \quad \int \epsilon_{\mathrm{A1}}\, f_{X_{B^c_{u,n}}\,|\,Y_{A_{u,n}}}\, dx_{B^c_{u,n}} \quad = \quad \epsilon_{\mathrm{A1}}.
\end{aligned}$$

□

Assumption A2. *For the collection of neighborhoods in Assumption A1, with* $B^+_{u,n} = B_{u,n} \cup (u,n)$*, there is a constant* b*, depending on* ϵ_{A1} *but not on* U *and* N*, such that*

$$\sup_{u\in 1:U,\, n\in 1:N} \left|B^+_{u,n}\right| \leq b.$$

Assumption A3. *There is a constant* Q*, independent of* U *and* N*, such that, for all* u *and* n*,*

$$Q^{-1} < f_{Y_{u,n}|X_{u,n}}(y^*_{u,n}\,|\,x_{u,n}) < Q$$

Assumption A4. *There exists* $\epsilon_{\mathrm{A4}} > 0$*, independent of* U *and* N*, such that the following holds. For each* u, n*, a set* $C_{u,n} \subset (1\,{:}\,U) \times (0\,{:}\,N)$ *exists such that* $(\tilde{u},\tilde{n}) \notin C_{u,n}$ *implies* $B^+_{u,n} \cap B^+_{\tilde{u},\tilde{n}} = \emptyset$ *and*

$$\left| f_{X_{B^+_{\tilde{u},\tilde{n}}}|X_{B^+_{u,n}}} - f_{X_{B^+_{\tilde{u},\tilde{n}}}} \right| < \epsilon_{\mathrm{A4}}\, f_{X_{B^+_{\tilde{u},\tilde{n}}}} \tag{S10}$$

Further, there is a uniform bound $|C_{u,n}| \leq c$*.*

Assumption A4 is needed only to ensure that the variance bound in Theorem 1 is essentially $O(UN)$ rather than $O(U^2N^2)$. Both these rates avoid the exponentially increasing variance characterizing the curse of dimensionality. Lower variance than $O(UN)$ cannot be anticipated for any sequential Monte Carlo method since the log likelihood estimate can be written as a sum of UN terms each of which involves its own sequential Monte Carlo calculation.

Theorem 1. *Let* ℓ^{MC} *denote the Monte Carlo likelihood approximation constructed by UBF. Consider a limit with a growing number of bootstrap replicates,* $\mathcal{I} \to \infty$*, and suppose assumptions A1, A2 and A3. There are quantities* $\epsilon(U,N)$ *and* $V(U,N)$*, with bounds* $|\epsilon| < \epsilon_{\mathrm{A1}} Q^2$ *and* $V < Q^{4b} U^2 N^2$*, such that*

$$\mathcal{I}^{1/2}\big[\ell^{MC} - \ell - \epsilon U N\big] \xrightarrow[\mathcal{I}\to\infty]{d} \mathcal{N}\big[0, V\big], \tag{S11}$$

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 additionally Assumption A4 holds, we obtain an improved variance bound*

$$V < Q^{4b}\, UN\big(c + \epsilon_{\mathrm{A4}}\,(UN - c)\big). \tag{S12}$$

Proof. Suppose the quantities $w^M_{u,n,i}$ and $w^P_{u,n,i}$ constructed in Algorithm UBF are considered i.i.d. replicates of jointly defined random variables $w^M_{u,n}$ and $w^P_{u,n}$, for each $(u,n) \in 1\!:\!U \times 1\!:\!N$. Also, write

$$\Delta^{MP}_{u,n} = \frac{1}{\sqrt{\mathcal{I}}} \sum_{i=1}^{\mathcal{I}} \big(w^M_{u,n,i} w^P_{u,n,i} - \mathbb{E}[w^M_{u,n} w^P_{u,n}] \big), \qquad \Delta^{P}_{u,n} = \frac{1}{\sqrt{\mathcal{I}}} \sum_{i=1}^{\mathcal{I}} \big(w^P_{u,n,i} - \mathbb{E}[w^P_{u,n}] \big),$$

Then, using the delta method (e.g., Section 2.5.3 in Liu (2001)) we find

$$\begin{aligned} \ell^{\mathrm{MC}}_{u,n} &= \log \left(\frac{\sum_{i=1}^{\mathcal{I}} w^M_{u,n,i} w^P_{u,n,i}}{\sum_{i=1}^{\mathcal{I}} w^P_{u,n,i}} \right) \\ &= \log \Big(\mathbb{E}[w^M_{u,n} w^P_{u,n}] + \mathcal{I}^{-1/2} \Delta^{MP}_{u,n} \Big) - \log \Big(\mathbb{E}[w^P_{u,n}] + \mathcal{I}^{-1/2} \Delta^{P}_{u,n} \Big) \\ &= \log \left(\frac{\mathbb{E}[w^M_{u,n} w^P_{u,n}]}{\mathbb{E}[w^P_{u,n}]} \right) + \mathcal{I}^{-1/2} \left(\frac{\Delta^{MP}_{u,n}}{\mathbb{E}[w^M_{u,n} w^P_{u,n}]} - \frac{\Delta^{P}_{u,n}}{\mathbb{E}[w^P_{u,n}]} \right) + o_P\big(\mathcal{I}^{-1/2}\big) \end{aligned} \tag{S13}$$

The joint distribution of $\big\{ (\Delta^{MP}_{u,n}, \Delta^{P}_{u,n}), (u,n) \in 1\!:\!U \times 1\!:\!N \big\}$ follows a standard central limit theorem as $\mathcal{I} \to \infty$. Each term has mean zero, with covariances uniformly bounded over $(u,n,\tilde{u},\tilde{n})$ due to Assumption A3. Specifically,

$$\mathrm{Var}\begin{pmatrix} \Delta^{MP}_{u,n} \\ \Delta^{P}_{u,n} \\ \Delta^{MP}_{\tilde{u},\tilde{n}} \\ \Delta^{P}_{\tilde{u},\tilde{n}} \end{pmatrix} = \begin{pmatrix} \mathrm{Var}(w^M_{u,n} w^P_{u,n}) & \mathrm{Cov}(w^M_{u,n} w^P_{u,n}, w^P_{u,n}) & \mathrm{Cov}(w^M_{u,n} w^P_{u,n}, w^M_{\tilde{u},\tilde{n}} w^P_{\tilde{u},\tilde{n}}) & \mathrm{Cov}(w^M_{u,n} w^P_{u,n}, w^P_{\tilde{u},\tilde{n}}) \\ \mathrm{Cov}(w^M_{u,n} w^P_{u,n}, w^P_{u,n}) & \mathrm{Var}(w^P_{u,n}) & \mathrm{Cov}(w^P_{u,n}, w^M_{\tilde{u},\tilde{n}} w^P_{\tilde{u},\tilde{n}}) & \mathrm{Cov}(w^P_{u,n}, w^P_{\tilde{u},\tilde{n}}) \\ \mathrm{Cov}(w^M_{u,n} w^P_{u,n}, w^M_{\tilde{u},\tilde{n}} w^P_{\tilde{u},\tilde{n}}) & \mathrm{Cov}(w^P_{u,n}, w^M_{\tilde{u},\tilde{n}} w^P_{\tilde{u},\tilde{n}}) & \mathrm{Var}(w^M_{\tilde{u},\tilde{n}} w^P_{\tilde{u},\tilde{n}}) & \mathrm{Cov}(w^M_{\tilde{u},\tilde{n}} w^P_{\tilde{u},\tilde{n}}, w^P_{\tilde{u},\tilde{n}}) \\ \mathrm{Cov}(w^M_{u,n} w^P_{u,n}, w^P_{\tilde{u},\tilde{n}}) & \mathrm{Cov}(w^P_{u,n}, w^P_{\tilde{u},\tilde{n}}) & \mathrm{Cov}(w^M_{\tilde{u},\tilde{n}} w^P_{\tilde{u},\tilde{n}}, w^P_{\tilde{u},\tilde{n}}) & \mathrm{Var}(w^P_{\tilde{u},\tilde{n}}) \end{pmatrix}$$

Note that

$$\begin{aligned} \log \left[\frac{\mathbb{E}[w^M_{u,n} w^P_{u,n}]}{\mathbb{E}[w^P_{u,n}]} \right] &= \log \left[\frac{\int f_{Y_{u,n}|X_{u,n}}(y^*_{u,n} \,|\, x_{u,n,i})\, f_{Y_{B_{u,n}}|X_{B_{u,n}}}(y^*_{B_{u,n}} \,|\, x_{B_{u,n}})\, f_{X_{B^+_{u,n}}}(x_{B^+_{u,n}})\, dx_{B^+_{u,n}}}{\int f_{Y_{B_{u,n}}|X_{B_{u,n}}}(y^*_{B_{u,n}} \,|\, x_{B_{u,n}})\, f_{X_{B_{u,n}}}(x_{B_{u,n}})\, dx_{B_{u,n}}} \right] \\ &= \log \big[f_{Y_{u,n}|Y_{B_{u,n}}}(y^*_{u,n} \,|\, y^*_{B_{u,n}}) \big], \end{aligned}$$

where $B^+_{u,n} = B_{u,n} \cup (u,n)$. Now, define

$$\Delta^{\ell}_{u,n} = \left(\frac{\Delta^{MP}_{u,n}}{\mathbb{E}[w^M_{u,n} w^P_{u,n}]} - \frac{\Delta^{P}_{u,n}}{\mathbb{E}[w^P_{u,n}]} \right)$$

Summing over all $(u,n) \in 1\!:\!U \times 1\!:\!N$, we get

$$\sqrt{\mathcal{I}} \left(\ell^{\mathrm{MC}} - \sum_{(u,n) \in 1:U \times 1:N} \log f_{Y_{u,n}|Y_{B_{u,n}}}(y^*_{u,n} \,|\, y^*_{B_{u,n}}) \right) = \sum_{(u,n) \in 1:U \times 1:N} \Delta^{\ell}_{u,n} + o(1). \tag{S14}$$

Now,

$$\mathrm{Cov}\big(\Delta^{\ell}_{u,n}, \Delta^{\ell}_{\tilde{u},\tilde{n}}\big) = \mathrm{Cov}\left(\frac{w^M_{u,n} w^P_{u,n}}{\mathbb{E}[w^M_{u,n} w^P_{u,n}]} - \frac{w^P_{u,n}}{\mathbb{E}[w^P_{u,n}]}, \frac{w^M_{\tilde{u},\tilde{n}} w^P_{\tilde{u},\tilde{n}}}{\mathbb{E}[w^M_{\tilde{u},\tilde{n}} w^P_{\tilde{u},\tilde{n}}]} - \frac{w^P_{\tilde{u},\tilde{n}}}{\mathbb{E}[w^P_{\tilde{u},\tilde{n}}]} \right).$$

Since

$$\left| \frac{w^M_{u,n} w^P_{u,n}}{\mathbb{E}[w^M_{u,n} w^P_{u,n}]} - \frac{w^P_{u,n}}{\mathbb{E}[w^P_{u,n}]} \right| < Q^{2b},$$

we have

$$\big|\mathrm{Cov}\big(\Delta^\ell_{u,n}, \Delta^\ell_{\tilde{u},\tilde{n}}\big)\big| < Q^{4b},$$

implying that

$$\mathrm{Var}\left(\sum_{(u,n)\in 1:U\times 1:N} \Delta^\ell_{u,n} \right) < Q^{4b} U^2 N^2. \tag{S15}$$

If, in addition, (u,n) and $(\tilde{u},\tilde{n}) \in A_{u,n}$ are sufficiently separated in the sense of Assumption A4, then Lemma S1 shows that Assumption A4 implies

$$\big|\mathrm{Cov}\big(\Delta^\ell_{u,n}, \Delta^\ell_{\tilde{u},\tilde{n}}\big)\big| < \epsilon_{\mathrm{A4}} Q^{4b}.$$

The number of insufficiently separated neighbors to (u,n) is bounded by c, and so we obtain

$$\mathrm{Var}\left(\sum_{(u,n)\in S} \Delta^\ell_{u,n} \right) < Q^{4b}\, UN\big(c + \epsilon_{\mathrm{A4}}\,(UN - c)\big). \tag{S16}$$

Now we proceed to bound the bias in the Monte Carlo central limit estimator of ℓ. Putting $h(x_{u,n}) = f_{Y_{u,n}|X_{u,n}}(y^*_{u,n}\,|\,x_{u,n})$ into Assumption A1, using Assumption A3, gives

$$\big|f_{Y_{u,n}|Y_{B_{u,n}}}(y^*_{u,n}\,|\,y^*_{B_{u,n}}) - f_{Y_{u,n}|Y_{A_{u,n}}}(y^*_{u,n}\,|\,y^*_{A_{u,n}})\big| < \epsilon_{\mathrm{A1}} Q.$$

Noting that

$$|a-b| < \delta, \quad a > Q^{-1} \text{ and } b > Q^{-1} \text{ implies } |\log(a) - \log(b)| < \delta Q, \tag{S17}$$

we find

$$\big|\log f_{Y_{u,n}|Y_{B_{u,n}}}(y^*_{u,n}\,|\,y^*_{B_{u,n}}) - \log f_{Y_{u,n}|Y_{A_{u,n}}}(y^*_{u,n}\,|\,y^*_{A_{u,n}})\big| < \epsilon_{\mathrm{A1}} Q^2. \tag{S18}$$

Summing over (u,n), we get

$$\left| \ell - \sum_{(u,n)\in S} \log f_{Y_{u,n}|Y_{B_{u,n}}}(y^*_{u,n}\,|\,y^*_{B_{u,n}}) \right| < \epsilon_{\mathrm{A1}} Q^2 UN. \tag{S19}$$

Together, the results in (S14), (S15), (S16) and (S19) confirm the assertions of the theorem. □

Lemma S1. *Suppose U and V are random variables with joint density satisfying*

$$\big|f_{V|U}(v\,|\,u) - f_V(v)\big| < \epsilon f_V(v). \tag{S20}$$

Suppose $|g(U)| < a$ and $|h(V)| < b$ for some real-valued function g and h. Then, $\mathrm{Cov}\big(g(U), h(V)\big) < ab\epsilon$.

Proof. The result is obtained by direct calculation, as follows.

$$\begin{aligned}
\operatorname{Cov}\big(g(U),h(V)\big) &= \mathbb{E}\Big[g(U)\,\mathbb{E}\Big[h(V)-\mathbb{E}[h(V)]\,\Big|\,U\Big]\Big] \\
&= \int\bigg\{\int g(u)h(v)\big(f_{V|U}(v\,|\,u)-f_V(v)\big)\,dv\bigg\}\,f_U(u)\,du \\
&< \int\bigg\{\int ab\epsilon f_V(v)\,dv\bigg\}\,f_U(u)\,du \\
&= ab\epsilon.
\end{aligned}$$

□

S4 ABF and ABF-IR convergence: Proof of Theorem 2

Let $g_{\boldsymbol{X}_{0:N},\boldsymbol{X}^P_{1:N}}(\boldsymbol{x}_{0:N},\boldsymbol{x}^P_{1:N})$ be the joint density of the adapted process and the proposal process,

$$g_{\boldsymbol{X}_{0:N},\boldsymbol{X}^P_{1:N}}(\boldsymbol{x}_{0:N},\boldsymbol{x}^P_{1:N}) = f_{\boldsymbol{X}_0}(\boldsymbol{x}_0)\prod_{n=1}^{N} f_{\boldsymbol{X}_n|\boldsymbol{X}_{n-1},\boldsymbol{Y}_n}\big(\boldsymbol{x}_n\,|\,\boldsymbol{x}_{n-1},\boldsymbol{y}^*_n\big)\ f_{\boldsymbol{X}_n|\boldsymbol{X}_{n-1}}\big(\boldsymbol{x}^P_n\,|\,\boldsymbol{x}_{n-1}\big). \tag{S21}$$

For $B\subset 1\!:\!U\times 1:N$, define $B^{[m]}=B\cap\big(1\!:\!U\times\{m\}\big)$ and set

$$\gamma_B=\prod_{m=1}^{N} f_{Y_{B^{[m]}}|\boldsymbol{X}_{m-1}}\big(y^*_{B^{[m]}}\,|\,\boldsymbol{X}_{m-1}\big), \tag{S22}$$

using the convention that an empty density $f_{Y_\emptyset}$ evaluates to 1. If we denoting $\mathbb{E}_g$ for expectation for $(\boldsymbol{X}_{0:N},\boldsymbol{X}^P_{1:N})$ having density $g_{\boldsymbol{X}_{0:N},\boldsymbol{X}^P_{1:N}}$, (S22) can be written as

$$\gamma_B=\mathbb{E}_g\left[f_{Y_B|X_B}\big(y^*_B\,|\,X^P_B\big)\,\Big|\,\boldsymbol{X}_{0:N}\right],$$

so we have

$$\mathbb{E}_g\big[\gamma_B\big]=\mathbb{E}_g\left[f_{Y_B|X_B}\big(y^*_B|X^P_B\big)\right].$$

Two useful identities are

$$\begin{aligned}
f_{X_{u,n}|Y_{A_{u,n}}}\big(x_{u,n}|y^*_{A_{u,n}}\big) &= \frac{\mathbb{E}_g\Big[f_{Y_{A_{u,n}}|X_{A_{u,n}}}\big(y^*_{A_{u,n}}\,|\,X^P_{A_{u,n}}\big)\,f_{X_{u,n}|X_{A^{[n]}_{u,n}},\boldsymbol{X}_{n-1}}\big(x_{u,n}|X^P_{A^{[n]}_{u,n}},\boldsymbol{X}_{n-1}\big)\Big]}{\mathbb{E}_g\Big[f_{Y_{A_{u,n}}|X_{A_{u,n}}}\big(y^*_{A_{u,n}}|X^P_{A_{u,n}}\big)\Big]}, \\
f_{Y_{u,n}|Y_{A_{u,n}}}\big(y^*_{u,n}|y^*_{A_{u,n}}\big) &= \frac{\mathbb{E}_g\big[\gamma_{A^+_{u,n}}\big]}{\mathbb{E}_g\big[\gamma_{A_{u,n}}\big]}.
\end{aligned}$$

Assumption B1. *There is an $\epsilon_{\mathrm{B1}}>0$, independent of U and N, and a collection of neighborhoods $\{B_{u,n}\subset A_{u,n}, u\in 1\!:\!U, n\in 1\!:\!N\}$ such that the following holds for all u and n, and any bounded real-valued function $|h(x)|\le 1$. Setting $A=A_{u,n}$, $B=B_{u,n}$, $f_A(x_A)=f_{Y_A|X_A}(y^*_A|x_A)$, and $f_B(x_B)=f_{Y_B|X_B}(y^*_B|x_B)$, so that we have the identity*

$$f_{X_{u,n}|Y_A}(x|y^*_A)=\frac{\mathbb{E}_g\big[f_A\big(X^P_A\big)f_{X_{u,n}|X_{A^{[n]}},\boldsymbol{X}_{n-1}}\big(x|X^P_{A^{[n]}},\boldsymbol{X}_{n-1}\big)\big]}{\mathbb{E}_g\big[f_A\big(X^P_A\big)\big]},$$

we require that

$$\left|\int h(x)\left\{\frac{\mathbb{E}_g\big[f_A\big(X_A^P\big)f_{X_{u,n}|X_{A[n]},\boldsymbol{X}_{n-1}}\big(x|X^P_{A[n]},\boldsymbol{X}_{n-1}\big)\big]}{\mathbb{E}_g\big[f_A\big(X_A^P\big)\big]}-\right.\right.$$
$$\left.\left.\frac{\mathbb{E}_g\big[f_B\big(X_B^P\big)f_{X_{u,n}|X_{B[n]},\boldsymbol{X}_{n-1}}\big(x|X^P_{B[n]},\boldsymbol{X}_{n-1}\big)\big]}{\mathbb{E}_g\big[f_B\big(X_B^P\big)\big]}\right\}dx\right| \;<\; \epsilon_{\mathrm{B1}}.$$

Assumption B2. *The bound* $\sup_{u\in 1:U,n\in 1:N}\left|B^+_{u,n}\right|\leq b$ *in Assumption A2 applies for the neighborhoods defined in Assumption B1. This also implies there is a finite maximum temporal depth for the collection of neighborhoods, defined as*

$$d_{\max}=\sup_{(u,n)}\;\sup_{(\tilde{u},\tilde{n})\in B_{u,n}}|n-\tilde{n}|.$$

Assumption B3. *Identically to Assumption A3,* $Q^{-1}<f_{Y_{u,n}|X_{u,n}}(y^*_{u,n}\mid x_{u,n})<Q$.

Proposition S2. *Setting* $h(x)=f_{Y_{u,n}|X_{u,n}}(y^*_{u,n}|x)$*, assumptions B1 and B3 imply*

$$\left|\frac{\mathbb{E}_g\left[\gamma_{A^+_{u,n}}\right]}{\mathbb{E}_g\left[\gamma_{A_{u,n}}\right]}-\frac{\mathbb{E}_g\left[\gamma_{B^+_{u,n}}\right]}{\mathbb{E}_g\left[\gamma_{B_{u,n}}\right]}\right|<Q\epsilon. \tag{S23}$$

Proof. Using the non-negativity of all terms to justify interchange of integral and expectation,

$$\begin{aligned}&\int h(x)\mathbb{E}_g\Big[f_{Y_{B_{u,n}}|X_{B_{u,n}}}(y^*_{B_{u,n}}|X^P_{B_{u,n}})f_{X_{u,n}|X_{B^{[n]}_{u,n}},\boldsymbol{X}_{n-1}}(x|X^P_{B^{[n]}_{u,n}},\boldsymbol{X}_{n-1})\Big]dx\\&=\mathbb{E}_g\Big[\int f_{Y_{u,n}|X_{u,n}}(y^*_{u,n}|x)f_{X_{u,n}|X_{B^{[n]}_{u,n}},\boldsymbol{X}_{n-1}}(x|X^P_{B^{[n]}_{u,n}},\boldsymbol{X}_{n-1})dx\cdot f_{Y_{B_{u,n}}|X_{B_{u,n}}}(y^*_{B_{u,n}}|X^P_{B_{u,n}})\Big]\end{aligned}\tag{S24}$$

But by the construction of g,

$$\begin{aligned}f_{X_{u,n}|X_{B^{[n]}_{u,n}},\boldsymbol{X}_{n-1}}(x|X^P_{B^{[n]}_{u,n}},\boldsymbol{X}_{n-1})&=g_{X^P_{u,n}|X^P_{B^{[n]}_{u,n}},\boldsymbol{X}_{n-1}}(x|X^P_{B^{[n]}_{u,n}},\boldsymbol{X}_{n-1})\\&=g_{X^P_{u,n}|X^P_{B_{u,n}},\boldsymbol{X}_{n-1}}(x|X^P_{B_{u,n}},\boldsymbol{X}_{n-1}).\end{aligned}$$

Thus (S24) becomes

$$\begin{aligned}&\mathbb{E}_g\Big[\int f_{Y_{u,n}|X_{u,n}}(y^*_{u,n}|x)g_{X^P_{u,n}|X_{B^P_{u,n}},\boldsymbol{X}_{n-1}}(x|X^P_{B_{u,n}},\boldsymbol{X}_{n-1})dx\cdot f_{Y_{B_{u,n}}|X_{B_{u,n}}}(y^*_{B_{u,n}}|X^P_{B_{u,n}})\Big]\\&=\mathbb{E}_g\left[\mathbb{E}_g\left[f_{Y_{u,n}|X_{u,n}}(y^*_{u,n}|X^P_{u,n})|X^P_{B_{u,n}},\boldsymbol{X}_{n-1}\right]\cdot f_{Y_{B_{u,n}}|X_{B_{u,n}}}(y^*_{B_{u,n}}|X^P_{B_{u,n}})\right]\\&=\mathbb{E}_g\left[f_{Y_{B^+_{u,n}}|X_{B^+_{u,n}}}(y^*_{B^+_{u,n}}|X^P_{B^+_{u,n}})\right]\\&=\mathbb{E}_g\gamma_{B^+_{u,n}}.\end{aligned}$$

Applying the same argument for the special case of $B_{u,n}=A_{u,n}$, we substitute into Assumption B1 to complete the proof with the fact that $h<Q$. □

Assumption B4. *We use subscripts of g to denote marginal and conditional densities derived from (S21). Suppose there is an ϵ_{B4}, independent of U and N, such that the following holds. For each u and n, a set $C_{u,n} \subset (1\,{:}\,U) \times (0\,{:}\,N)$ exists such that $(\tilde{u},\tilde{n}) \notin C_{u,n}$ implies $B^{+}_{u,n} \cap B^{+}_{\tilde{u},\tilde{n}} = \emptyset$ and*

$$\big| g_{X^P_{B_{\tilde{u},\tilde{n}} \cup B_{u,n}}} - g_{X^P_{B_{\tilde{u},\tilde{n}}}}\, g_{X^P_{B_{u,n}}} \big| < (1/2)\, \epsilon_{\mathrm{B4}}\; g_{X^P_{B_{\tilde{u},\tilde{n}} \cup B_{u,n}}}$$

$$\begin{aligned}\big| g_{X^P_{B_{\tilde{u},\tilde{n}}} | \boldsymbol{X}_{0:N}}\, g_{X^P_{B_{u,n}} | \boldsymbol{X}_{0:N}} - g_{X^P_{B_{\tilde{u},\tilde{n}} \cup B_{u,n}} | \boldsymbol{X}_{0:N}} \big| & \\ < (1/2)\, \epsilon_{\mathrm{B4}}\; g_{X^P_{B_{\tilde{u},\tilde{n}} \cup B_{u,n}} | \boldsymbol{X}_{0:N}} &\end{aligned}$$

Further, there is a uniform bound $|C_{u,n}| \le c$.

The mixing of the adapted process in Assumption B4 replaces the mixing of the unconditional process in Assumption A4. Though mixing of the adapted process may be hard to check, one may suspect that the adapted process typically mixes more rapidly than the unconditional process. Assumption B4 is needed only to ensure that the variance bound in Theorem 2 is essentially $O(UN)$ rather than $O(U^2N^2)$. Either of these rates avoids the exponentially increasing variance characterizing the curse of dimensionality. Lower variance than $O(UN)$ cannot be anticipated for any sequential Monte Carlo method since the log likelihood estimate can be written as a sum of UN terms each of which involves its own sequential Monte Carlo calculation. The following Proposition S3 gives an implication of Assumption B4.

Proposition S3. *Assumption B4 implies that, if $(\tilde{u},\tilde{n}) \notin C_{u,n}$,*

$$\mathrm{Cov}_g\big(\gamma_{B_{u,n}}, \gamma_{B_{\tilde{u},\tilde{n}}}\big) < \epsilon_{\mathrm{B4}} Q^{|B_{u,n}|+|B_{\tilde{u},\tilde{n}}|}. \tag{S25}$$

Proof. Write $\gamma = \gamma_{B_{u,n}}$ and $\tilde{\gamma} = \gamma_{B_{\tilde{u},\tilde{n}}}$. Also, write $B = B_{u,n}$, $\tilde{B} = B_{\tilde{u},\tilde{n}}$ and $f_B(x^P_B) = f_{Y_B|X_B}(y^*_B \,|\, x^P_B)$. Then,

$$\begin{aligned}\mathbb{E}[\gamma\tilde{\gamma}] &= \int \bigg[\int f_B(x^P_B) g_{X^P_B|\boldsymbol{X}_{0:N}}(x^P_B \,|\, \boldsymbol{x}_{0:N})\, dx^P_B \bigg] \\ &\qquad \times \bigg[\int f_{\tilde{B}}(x^P_{\tilde{B}}) g_{X^P_{\tilde{B}}|\boldsymbol{X}_{0:N}}(x^P_{\tilde{B}} \,|\, \boldsymbol{x}_{0:N})\, dx^P_{\tilde{B}} \bigg] g_{\boldsymbol{X}_{0:N}}(\boldsymbol{x}_{0:N})\, d\boldsymbol{x}_{0:N} \\ &= \int\!\!\int f_B(x^P_B)\, f_{\tilde{B}}(x^P_{\tilde{B}}) \bigg\{ \int g_{X^P_B|\boldsymbol{X}_{0:N}}(x^P_B \,|\, \boldsymbol{x}_{0:N}) \\ &\qquad \times g_{X^P_{\tilde{B}}|\boldsymbol{X}_{0:N}}(x^P_{\tilde{B}} \,|\, \boldsymbol{x}_{0:N})\, g_{\boldsymbol{X}_{0:N}}(\boldsymbol{x}_{0:N})\, d\boldsymbol{x}_{0:N} \bigg\} dx^P_B\, dx^P_{\tilde{B}} \end{aligned} \tag{S26}$$

Putting the approximate conditional independence requirement of Assumption B4 into (S26), we have

$$\begin{aligned}\bigg| \mathbb{E}[\gamma\tilde{\gamma}] - \int f_B(x^P_B)\, f_{\tilde{B}}(x^P_{\tilde{B}})\, g_{X^P_B X^P_{\tilde{B}}|\boldsymbol{X}_{0:N}}(x^P_B, x^P_{\tilde{B}} \,|\, \boldsymbol{x}_{0:N})\, g_{\boldsymbol{X}_{0:N}}(\boldsymbol{x}_{0:N})\, dx^P_B\, dx^P_{\tilde{B}}\, d\boldsymbol{x}_{0:N} \bigg| & \\ < (1/2)\, \epsilon_{\mathrm{B4}}\, Q^{|B|+|\tilde{B}|}. &\end{aligned}$$

This gives

$$\left|\mathbb{E}[\gamma\tilde{\gamma}] - \int f_B(x_B^P)\, f_{\tilde{B}}(x_{\tilde{B}}^P)\, g_{X_B^P X_{\tilde{B}}^P}(x_B^P, x_{\tilde{B}}^P)\, dx_B^P\, dx_{\tilde{B}}^P\right| < (1/2)\,\epsilon_{\mathrm{B4}}\, Q^{|B|+|\tilde{B}|}. \tag{S27}$$

Then, using the approximate unconditional independence requirement of Assumption B4 combined with the triangle inequality, (S27) implies

$$\left|\mathbb{E}[\gamma\tilde{\gamma}] - \int f_B(x_B^P)\, f_{\tilde{B}}(x_{\tilde{B}}^P)\, g_{X_B^P}(x_B^P)\, g_{X_{\tilde{B}}^P}(x_{\tilde{B}}^P)\, dx_B^P\, dx_{\tilde{B}}^P\right| < \epsilon_{\mathrm{B4}}\, Q^{|B|+|\tilde{B}|}. \tag{S28}$$

We can rewrite (S28) as

$$\big|\, \mathbb{E}[\gamma\tilde{\gamma}] - \mathbb{E}[\gamma]\, \mathbb{E}[\tilde{\gamma}] \,\big| < \epsilon_{\mathrm{B4}}\, Q^{|B|+|\tilde{B}|}, \tag{S29}$$

proving the proposition. □

Assumption B5. *There is a constant K, independent of U and N, such that, for any $0 \le d \le d_{\max}$, any $n \ge K + d$, and any set $D \subset (1\!:\!U) \times (n\!:\!n-d)$,*

$$\begin{aligned} &\big|g_{X_D|\boldsymbol{X}_{n-d-K}}(x_D \,|\, \boldsymbol{x}_{n-d-K}^{(1)}) - g_{X_D|\boldsymbol{X}_{n-d-K}}(x_D \,|\, \boldsymbol{x}_{n-d-K}^{(2)})\big| \\ &\qquad < \epsilon_{\mathrm{B5}}\ g_{X_D|\boldsymbol{X}_{n-d-K}}(x_D \,|\, \boldsymbol{x}_{n-d-K}^{(1)}) \end{aligned}$$

holds for all $\boldsymbol{x}_{n-d-K}^{(1)}$, $\boldsymbol{x}_{n-d-K}^{(2)}$, and x_D.

Assumption B5 is needed to ensure the stability of the Monte Carlo approximation to the adapted process. It ensures that any error due to finite Monte Carlo sample size has limited consequences at sufficiently remote time points. One could instead propose a bound that decreases exponentially with K, but that is not needed for the current purposes. The following Proposition S4 is useful for taking advantage of Assumption B5.

Proposition S4. *Suppose that f is a non-negative function and that for some $\epsilon > 0$,*

$$|f(x) - f(x')| < \epsilon f(x')$$

holds for all x, x'. Then for any two probability distributions where the expectations are denoted by $\mathbb{E}_1$ and $\mathbb{E}_2$ and for any random variable X, we have

$$|\mathbb{E}_1 f(X) - \mathbb{E}_2 f(X)| \le \epsilon\, \mathbb{E}_2 f(X).$$

Proof. Let the two probability laws be denoted by P_1 and P_2. We have

$$\begin{aligned} |\mathbb{E}_1 f(X) - \mathbb{E}_2 f(X)| &= \left|\int f(x) P_1(dx) - \int f(x') P_2(dx')\right| \\ &\le \left|\int\!\!\int f(x) P_1(dx) P_2(dx') - \int\!\!\int f(x') P_1(dx) P_2(dx')\right|. \\ &\le \int\!\!\int |f(x) - f(x')| P_1(dx) P_2(dx') \\ &\le \int \epsilon f(x') P_2(dx') = \epsilon\, \mathbb{E}_2 f(X). \end{aligned}$$

□

Assumption B6. *Let h be a bounded function with $|h(x)| \leq 1$. Let $\boldsymbol{X}^{\mathrm{IR}}_{n,S,j,i}$ be the Monte Carlo quantity constructed in ABF-IR, conditional on $\boldsymbol{X}^{\mathrm{A}}_{n-1,S,i} = \boldsymbol{x}^{\mathrm{A}}_{n-1,S,i}$. There is a constant $\mathcal{C}_0(U,N,S)$ such that, for all $\epsilon_{\mathrm{B6}} > 0$ and $\boldsymbol{x}^{\mathrm{A}}_{n-1,S,i}$, whenever the number of particles satisfies $J > \mathcal{C}_0(U,N,S)/\epsilon^3_{\mathrm{B6}}$,*

$$\mathbb{E}\left|\frac{1}{J}\sum_{j=1}^{J} h(\boldsymbol{X}^{\mathrm{IR}}_{n,S,j,i}) - \mathbb{E}_g\big[h(\boldsymbol{X}_n)\,|\,\boldsymbol{X}_{n-1} = \boldsymbol{x}^{\mathrm{A}}_{n-1,S,i}\big]\right| < \epsilon_{\mathrm{B6}}.$$

Assumption B6 controls the Monte Carlo error for a single time interval on a single bootstrap replicate. In the case $S = 1$, ABF-IR becomes ABF and this assumption is one of many alternatives for bounding error from importance sampling. The purpose behind the selection of Assumption B6 is to draw on the results of Park and Ionides (2020) for intermediate resampling, and our assumption is a restatement of their Theorem 2. When $S = 1$, the curse of dimensionality for importance sampling has the consequence that C_0 grows exponentially with U. However, Park and Ionides (2020) showed that setting $S = U$ can lead to situations where $C_0(U,N,S)$ in Assumption B6 grows polynomially with U. Here, we do not place requirements concerning the dependence of C_0 on U, N and S since our immediate concern is a limit where $\mathcal{I}$ and J increase. Nevertheless, the numerical results are consistent with the theoretical and empirical results obtained for intermediate resampling in the context of particle filtering by Park and Ionides (2020).

Assumption B7. *For $1 \leq n \leq N$, the Monte Carlo random variable $X^A_{n,i}$ is independent of $w^M_{u,n,i,j}$ conditional on $X^A_{n-1,i}$.*

The Monte Carlo conditional independence required by Assumption B7 would hold for ABF-IR if the guide variance $V_{u,n,i}$ were calculated using an independent set of guide simulations to those used for evaluating the measurement weights $w^M_{u,n,i,j}$. For numerical efficiency, the ABF-IR algorithm implemented here constructs a shared pool of simulations for both purposes rather than splitting the pool up between them, in the expectation that the resulting minor violation of Assumption B7 has negligible impact.

Theorem 2. *Let $\ell^{\,MC}$ denote the Monte Carlo likelihood approximation constructed by ABF-IR, or by ABF since this is the special case of ABF-IR with $S = 1$. Consider a limit with a growing number of bootstrap replicates, $\mathcal{I} \to \infty$, and suppose assumptions B1, B2, B3, B5, B6 and B7. Suppose the number of particles J exceeds the requirement for B6. There are quantities $\epsilon(U,N)$ and $V(U,N)$ with $|\epsilon| < Q^2\epsilon_{\mathrm{B1}} + 2Q^{2b}\big(\epsilon_{\mathrm{B5}} + (K + d_{\max})\epsilon_{\mathrm{B6}}\big)$ and $V < Q^{4b}U^2N^2$ such that*

$$\mathcal{I}^{1/2}\big[\ell^{\,MC} - \ell - \epsilon U N\big] \xrightarrow[\mathcal{I}\to\infty]{d} \mathcal{N}\big[0, V\big]. \tag{S30}$$

If additionally Assumption B4 holds, we obtain an improved rate of

$$V < Q^{4b} N U\big\{c + \big(\epsilon_{\mathrm{B4}} + 3\epsilon_{\mathrm{B5}} + 4(K + d_{\max})\,\epsilon_{\mathrm{B6}}\big)\big(NU - c\big)\big\} \tag{S31}$$

Proof. First, we set up some notation. For $B_{u,n}$ and $w^M_{u,n,i,j}$ constructed by ABF-IR, define

$$\gamma^{MC,i}_{B_{u,n}} = \prod_{m=1}^{n}\left[\frac{1}{J}\sum_{j=1}^{J}\prod_{(\tilde{u},m)\in B^{[m]}_{u,n}} w^M_{\tilde{u},m,i,j}\right] \quad \text{and} \quad \bar{\gamma}^{MC}_{B_{u,n}} = \frac{1}{\mathcal{I}}\sum_{i=1}^{\mathcal{I}}\gamma^{MC,i}_{B_{u,n}}. \tag{S32}$$

The Monte Carlo conditional likelihoods output by ABF-IR can be written as

$$\ell_{u,n}^{\mathrm{MC}} = \log \bar{\gamma}_{B_{u,n}^{+}}^{\mathrm{MC}} - \log \bar{\gamma}_{B_{u,n}}^{\mathrm{MC}}. \tag{S33}$$

We proceed with a similar argument to the proof of Theorem 1. Since $\gamma_{B_{u,n}}^{MC,i}$ are i.i.d. for $i \in 1\!:\!\mathcal{I}$, we can suppose they are replicates of a Monte Carlo random variable $\gamma_{B_{u,n}}^{MC}$. We define

$$\Delta_{u,n}^{+} = \frac{1}{\sqrt{\mathcal{I}}} \sum_{i=1}^{\mathcal{I}} \big(\gamma_{B_{u,n}^{+}}^{MC,i} - \mathbb{E}\big[\gamma_{B_{u,n}^{+}}^{MC}\big]\big), \qquad \Delta_{u,n} = \frac{1}{\sqrt{\mathcal{I}}} \sum_{i=1}^{\mathcal{I}} \big(\gamma_{B_{u,n}}^{MC,i} - \mathbb{E}\big[\gamma_{B_{u,n}}^{MC}\big]\big).$$

The same calculation as (S13) gives

$$\ell_{u,n}^{\mathrm{MC}} \;=\; \log\left(\frac{\mathbb{E}\big[\gamma_{B_{u,n}^{+}}^{MC}\big]}{\mathbb{E}\big[\gamma_{B_{u,n}}^{MC}\big]}\right) + \mathcal{I}^{-1/2}\left(\frac{\Delta_{u,n}^{+}}{\mathbb{E}\big[\gamma_{B_{u,n}^{+}}^{MC}\big]} - \frac{\Delta_{u,n}}{\mathbb{E}\big[\gamma_{B_{u,n}}^{MC}\big]}\right) + o_P\big(\mathcal{I}^{-1/2}\big) \tag{S34}$$

The joint distribution of $\big\{(\Delta_{u,n}^{+}, \Delta_{u,n}), (u,n) \in 1\!:\!U \times 1\!:\!N\big\}$ follows a standard central limit theorem as $\mathcal{I} \to \infty$. Each term has mean zero, with variances and covariances uniformly bounded over $(u,n,\tilde{u},\tilde{n})$ due to Assumption B3. From Proposition S2, using the same reasoning as (S18),

$$\left|\log\left(\frac{\mathbb{E}_g\big[\gamma_{A_{u,n}^{+}}\big]}{\mathbb{E}_g\big[\gamma_{A_{u,n}}\big]}\right) - \log\left(\frac{\mathbb{E}_g\big[\gamma_{B_{u,n}^{+}}\big]}{\mathbb{E}_g\big[\gamma_{B_{u,n}}\big]}\right)\right| < \epsilon_{\mathrm{B1}} Q^2. \tag{S35}$$

Now we use Lemma S2 and (S17) to obtain

$$\begin{aligned}
&\left|\log\left(\frac{\mathbb{E}_g\big[\gamma_{B_{u,n}^{+}}\big]}{\mathbb{E}_g\big[\gamma_{B_{u,n}}\big]}\right) - \log\left(\frac{\mathbb{E}\big[\gamma_{B_{u,n}^{+}}^{MC}\big]}{\mathbb{E}\big[\gamma_{B_{u,n}}\big]}\right)\right| \\
&\le \Big|\log \mathbb{E}_g\big[\gamma_{B_{u,n}^{+}}\big] - \log \mathbb{E}\big[\gamma_{B_{u,n}^{+}}\big]\Big| + \Big|\log \mathbb{E}_g\big[\gamma_{B_{u,n}}\big] - \log \mathbb{E}\big[\gamma_{B_{u,n}}\big]\Big| \\
&< 2Q^{2b}\big(\epsilon_{\mathrm{B5}} + (K + d_{\max})\epsilon_{\mathrm{B6}}\big).
\end{aligned} \tag{S36}$$

The proof of the central limit result in (S30) is completed by combining (S34), (S35) and (S36). To show (S31) we check that $\Delta_{u,n}$ and $\Delta_{\tilde{u},\tilde{n}}$ are weakly correlated when (u,n) and $(\tilde{u},\tilde{n})$ are sufficiently separated. By the same reasoning as the proof of Theorem 1, it is sufficient to show that $\gamma_{B_{u,n}}^{MC}$ and $\gamma_{B_{\tilde{u},\tilde{n}}}^{MC}$ are weakly correlated. These Monte Carlo quantities approximate $\gamma_{B_{u,n}}(\boldsymbol{X}_{0:n-1})$ and $\gamma_{B_{\tilde{u},\tilde{n}}}(\boldsymbol{X}_{0:\tilde{n}-1})$ with $\boldsymbol{X}$ drawn from g. Let us suppose $n \ge \tilde{n}$, and write $d_{u,n} = n - \inf_{(v,m)\in B_{u,n}} m$. First, we consider the situation $n - \tilde{n} > K + d_{u,n}$, in which case we can use the Markov property to give

$$\mathrm{Cov}\big(\gamma_{B_{u,n}}^{MC}, \gamma_{B_{\tilde{u},\tilde{n}}}^{MC}\big) < \mathbb{E}\big[\gamma_{B_{\tilde{u},\tilde{n}}}^{MC}\big] \sup_{\boldsymbol{x}} \Big\{\mathbb{E}\big[\gamma_{B_{u,n}}^{MC} \big| \boldsymbol{X}_{n-d_{u,n}-K,1}^{A} = \boldsymbol{x}\big] - \mathbb{E}\big[\gamma_{B_{u,n}}^{MC}\big]\Big\} \tag{S37}$$

Then, the triangle inequality followed by applications of Assumption B5 and Lemma S2 gives

$$\begin{aligned}
&\Big|\mathbb{E}\big[\gamma^{MC}_{B_{u,n}}\big|\boldsymbol{X}^{A}_{n-d_{u,n}-K,1}=\boldsymbol{x}\big]-\mathbb{E}\big[\gamma^{MC}_{B_{u,n}}\big]\Big| \\
&\qquad \le \Big|\mathbb{E}_g\big[\gamma_{B_{u,n}}\big|\boldsymbol{X}_{n-d_{u,n}-K}=\boldsymbol{x}\big]-\mathbb{E}_g\big[\gamma_{B_{u,n}}\big]\Big| \\
&\qquad\qquad +\Big|\mathbb{E}\big[\gamma^{MC}_{B_{u,n}}\big|\boldsymbol{X}^{A}_{n-d_{u,n}-K,1}=\boldsymbol{x}\big]-\mathbb{E}_g\big[\gamma_{B_{u,n}}\big|\boldsymbol{X}_{n-d_{u,n}-K}=\boldsymbol{x}\big]\Big| \\
&\qquad\qquad +\Big|\mathbb{E}\big[\gamma^{MC}_{B_{u,n}}\big]-\mathbb{E}_g\big[\gamma_{B_{u,n}}\big]\Big| \\
&\qquad \le Q^{b}\Big(2\epsilon_{\mathrm{B5}}+2\big(K+d_{u,n}\big)\epsilon_{\mathrm{B6}}\Big) \qquad\qquad (\mathrm{S38})
\end{aligned}$$

Putting (S38) into (S37), we get

$$\mathrm{Cov}\big(\gamma^{MC}_{B_{u,n}},\gamma^{MC}_{B_{\tilde{u},\tilde{n}}}\big) < Q^{2b}\Big(2\epsilon_{\mathrm{B5}}+2\big(K+d_{u,n}\big)\epsilon_{\mathrm{B6}}\Big). \qquad (\mathrm{S39})$$

Now we address the situation $n-\tilde{n}\le K+d_{u,n}$. We apply Lemma S2 on the union $B_{u,n}\cup B_{\tilde{u},\tilde{n}}$ for which the temporal depth is bounded by $d\le K+d_{u,n}+d_{\tilde{u},\tilde{n}}$. This gives

$$\Big|\,\mathbb{E}\big[\gamma^{MC}_{B_{u,n}}\gamma^{MC}_{B_{\tilde{u},\tilde{n}}}\big]-\mathbb{E}_g\big[\gamma_{B_{u,n}}\gamma_{B_{\tilde{u},\tilde{n}}}\big]\,\Big| < Q^{2b}\Big((2K+d_{u,n}+d_{\tilde{u},\tilde{n}})\epsilon_{\mathrm{B6}}+\epsilon_{\mathrm{B5}}\Big). \qquad (\mathrm{S40})$$

From Proposition S3, if $(\tilde{u},\tilde{n})\notin C_{u,n}$,

$$\mathrm{Cov}_g\big(\gamma_{B_{u,n}},\gamma_{B_{\tilde{u},\tilde{n}}}\big) < \epsilon_{\mathrm{B4}}Q^{2b}. \qquad (\mathrm{S41})$$

Now, we establish that $\mathrm{Cov}\big(\gamma^{MC}_{B_{u,n}},\gamma^{MC}_{B_{\tilde{u},\tilde{n}}}\big)$ is close to $\mathrm{Cov}_g\big(\gamma_{B_{u,n}},\gamma_{B_{\tilde{u},\tilde{n}}}\big)$.

$$\begin{aligned}
&\Big|\mathrm{Cov}\big(\gamma^{MC}_{B_{u,n}},\gamma^{MC}_{B_{\tilde{u},\tilde{n}}}\big)-\mathrm{Cov}_g\big(\gamma_{B_{u,n}},\gamma_{B_{\tilde{u},\tilde{n}}}\big)\Big| \\
&\qquad \le \Big|\,\mathbb{E}\big[\gamma^{MC}_{B_{u,n}}\gamma^{MC}_{B_{\tilde{u},\tilde{n}}}\big]-\mathbb{E}_g\big[\gamma_{B_{u,n}}\gamma_{B_{\tilde{u},\tilde{n}}}\big]\,\Big| \\
&\qquad\qquad +\Big|\mathbb{E}\big[\gamma^{MC}_{B_{u,n}}\big]\big(\mathbb{E}\big[\gamma^{MC}_{B_{\tilde{u},\tilde{n}}}\big]-\mathbb{E}_g\big[\gamma_{B_{\tilde{u},\tilde{n}}}\big]\big)\Big| \\
&\qquad\qquad +\Big|\mathbb{E}\big[\gamma_{B_{\tilde{u},\tilde{n}}}\big]\big(\mathbb{E}\big[\gamma^{MC}_{B_{u,n}}\big]-\mathbb{E}_g\big[\gamma_{B_{u,n}}\big]\big)\Big| \\
&\qquad < Q^{2b}\Big((2K+d_{u,n}+d_{\tilde{u},\tilde{n}})\epsilon_{\mathrm{B6}}+\epsilon_{\mathrm{B5}}+2(\epsilon_{\mathrm{B5}}+(K+d_{\max})\epsilon_{\mathrm{B6}})\Big). \\
&\qquad < Q^{2b}\Big(3\epsilon_{\mathrm{B5}}+4(K+d_{\max})\,\epsilon_{\mathrm{B6}}\Big) \qquad\qquad (\mathrm{S42})
\end{aligned}$$

Using (S42) together with (S41) to bound the $UN(UN-c)$ off-diagonal covariance terms completes the derivation of (S31).

□

Lemma S2. *Suppose Assumptions B3, B5, B6 and B7. Suppose the number of particles J exceeds the requirement for B6. If we write $d_B=\max_{(u_1,n_1),(u_2,n_2)\in B}|n_1-n_2|$ for $B\subset 1\!:\!U\times 1\!:\!N$, then for any B,*

$$\Big|\,\mathbb{E}\big[\gamma^{MC}_{B}\big|\boldsymbol{X}^{A}_{n-d_B-K,1}=\boldsymbol{x}\big]-\mathbb{E}_g\big[\gamma_{B}\big|\boldsymbol{X}_{n-d_B-K}=\boldsymbol{x}\big]\,\Big| < Q^{|B|}(K+d_B)\epsilon_{\mathrm{B6}},\quad \forall\boldsymbol{x}\in\mathbb{X}^U,$$

and

$$\Big|\,\mathbb{E}\big[\gamma^{MC}_{B}\big]-\mathbb{E}_g\big[\gamma_{B}\big]\,\Big| < Q^{|B|}(\epsilon_{\mathrm{B5}}+(K+d_B)\epsilon_{\mathrm{B6}}). \qquad (\mathrm{S43})$$

Proof. Suppose that $\max_{(u',n')\in B} n' = n$. Define $\eta_n(\boldsymbol{x}_n) = 1$ and, for $0 \le m \le n-1$,

$$\eta_m(\boldsymbol{x}_m) = \mathbb{E}_g\left[\prod_{k=m+1}^{n} \gamma_{B^{[k]}} \,\Big|\boldsymbol{X}_m = \boldsymbol{x}_m\right]. \tag{S44}$$

We have a recursive identity

$$\eta_m(\boldsymbol{X}_m) = \mathbb{E}_g\Big[\gamma_{B^{[m+1]}}\, \eta_{m+1}(\boldsymbol{X}_{m+1})\Big|\boldsymbol{X}_m\Big]. \tag{S45}$$

By taking the expectation of (S44), we have

$$\mathbb{E}_g\big[\eta_0(\boldsymbol{X}_0)\big] = \mathbb{E}_g\big[\gamma_B\big]. \tag{S46}$$

Note that g has marginal density $f_{\boldsymbol{X}_0}$ for $\boldsymbol{X}_0$. We analyze an ABF-IR approximation to (S45). The function $\eta_{m+1}(\boldsymbol{x})$ is not in practice computationally available for evaluation via ABF-IR, but the recursion nevertheless leads to a useful bound. Let $\boldsymbol{X}^{\mathrm{A}}_{m+1}[j](\boldsymbol{x}_m)$ correspond to the variable $\boldsymbol{X}^{IR}_{m+1,S,1,j}$ constructed by ABF-IR conditional on $\boldsymbol{X}^{\mathrm{A}}_{m,1} = \boldsymbol{x}_m$. Equivalently, $\boldsymbol{X}^{\mathrm{A}}_{m+1}[j](\boldsymbol{x}_m)$ matches the variable $\boldsymbol{X}^{\mathrm{A}}_{m+1,1}$ in ABF-IR if the assignment $\boldsymbol{X}^{\mathrm{A}}_{m+1,1} = \boldsymbol{X}^{IR}_{m+1,S,1,1}$ is replaced by $\boldsymbol{X}^{\mathrm{A}}_{m+1,1} = \boldsymbol{X}^{IR}_{m+1,S,1,j}$ conditional on $\boldsymbol{X}^{\mathrm{A}}_{m,1} = \boldsymbol{x}_m$. We define an approximation error $e_m(\boldsymbol{x}_m)$ by

$$\eta_m(\boldsymbol{x}_m) = \frac{1}{J}\sum_{j=1}^{J} f_{Y_{B^{[m+1]}}|\boldsymbol{X}_m}(y^*_{B^{[m+1]}} \,|\, \boldsymbol{x}_m)\, \eta_{m+1}\big(\boldsymbol{X}^{\mathrm{A}}_{m+1}[j](\boldsymbol{x}_m)\big) + e_m(\boldsymbol{x}_m). \tag{S47}$$

From Assumptions B3 and B6, $\mathbb{E}\big|e_m(\boldsymbol{x}_m)\big| < \epsilon_{\mathrm{B6}}\, Q^{\left|B^{[m+1:n]}\right|}$ uniformly over $\boldsymbol{x}_m$, Thus, setting $r_m = \mathbb{E}|e_m(\boldsymbol{X}^A_{m,1})|$, we have

$$r_m < \epsilon_{\mathrm{B6}}\, Q^{\left|B^{[m+1:n]}\right|}. \tag{S48}$$

Now, setting $K' = K + d_{u,n}$, we commence to prove inductively that, for $n - K' \le m \le n$,

$$\left|\eta_{n-K'}(\boldsymbol{x}) - \mathbb{E}\Big[\eta_m(\boldsymbol{X}^A_{m,1}) \prod_{k=n-K'+1}^{m} f_{Y_{B^{[k]}}|\boldsymbol{X}_{k-1}}(y^*_{B^{[k]}} \,|\, \boldsymbol{X}^A_{k-1,1})\Big|\boldsymbol{X}^A_{n-K',1} = \boldsymbol{x}\Big]\right| < (m-n+K')\epsilon_{\mathrm{B6}}\, Q^{|B|}. \tag{S49}$$

First, suppose that (S49) holds for m. From (S47) and (S48),

$$\left|\eta_m(\boldsymbol{x}_m) - \mathbb{E}\left[\frac{1}{J}\sum_{j=1}^{J} f_{Y_{B^{[m+1]}}|\boldsymbol{X}_m}(y^*_{B^{[m+1]}} \,|\, \boldsymbol{x}_m)\, \eta_{m+1}\big(\boldsymbol{X}^{\mathrm{A}}_{m+1}[j](\boldsymbol{x}_m)\big)\right]\right| < \epsilon_{\mathrm{B6}}\, Q^{\left|B^{[m+1:n]}\right|}. \tag{S50}$$

Since the particles are exchangeable, the expectation of the mean of J particles can be replaced with the expectation of the first particle. Plugging in $\boldsymbol{x}_m = \boldsymbol{X}^A_{m,1}$ gives us

$$\Big|\eta_m(\boldsymbol{X}^A_{m,1}) - f_{Y_{B^{[m+1]}}|\boldsymbol{X}_m}(y^*_{B^{[m+1]}} \,|\, \boldsymbol{X}^A_{m,1})\mathbb{E}\left[\eta_{m+1}\big(\boldsymbol{X}^A_{m+1,1}\big)\big|\boldsymbol{X}^A_{m,1}\right]\Big| < \epsilon_{\mathrm{B6}}\, Q^{\left|B^{[m+1:n]}\right|} \tag{S51}$$

Putting (S51) into (S49), for $m \le n$, and taking an iterated expectation with respect to $\boldsymbol{X}^A_{m,1}$, we find that (S49) holds also for $m+1$. Since (S49) holds trivially for $m = n - K'$, it holds for $n - K' \le m \le n$ by induction. Then, noting $\eta_n(\boldsymbol{x}) = 1$, we have from (S49) that

$$\left|\eta_{n-K'}(x) - \mathbb{E}\left[\prod_{k=n-K'+1}^{n} f_{Y_{B^{[k]}}|\boldsymbol{X}_{k-1}}\big(y^*_{B^{[k]}} \,|\, \boldsymbol{X}^A_{k-1,1}\big)\middle|\boldsymbol{X}^A_{n-K',1} = x\right]\right| < K'\epsilon_{\mathrm{B6}}\, Q^{|B|}.$$

Integrating the above inequality over $\boldsymbol{x}$ with respect to the law of $\boldsymbol{X}^A_{n-K',1}$, we obtain

$$\left|\mathbb{E}[\eta_{n-K'}(\boldsymbol{X}^A_{n-K',1})] - \mathbb{E}\left[\prod_{k=n-K'+1}^{n} f_{Y_{B[k]}|\boldsymbol{X}_{k-1}}\big(y^*_{B[k]} \mid \boldsymbol{X}^A_{k-1,1}\big)\right]\right| < K' \epsilon_{\text{B6}}\, Q^{|B|}. \tag{S52}$$

But under Assumption B7, we have

$$\mathbb{E}\left[\prod_{k=n-K'+1}^{n} f_{Y_{B[k]}|\boldsymbol{X}_{k-1}}\big(y^*_{B[k]} \mid \boldsymbol{X}^A_{k-1,1}\big)\right] = \mathbb{E}\big[\gamma_B^{MC}\big]. \tag{S53}$$

Assumption B5 says

$$|\eta_{n-K'}(\boldsymbol{x}^{(1)}_{n-K'}) - \eta_{n-K'}(\boldsymbol{x}^{(2)}_{n-K'})| < \epsilon_{\text{B5}}\, \eta_{n-K'}(\boldsymbol{x}^{(2)}_{n-K'}). \tag{S54}$$

Application of Proposition S4 to (S54) gives

$$\Big|\mathbb{E}_g\big[\eta_{n-K'}(\boldsymbol{X}_{n-K'})\big] - \mathbb{E}\big[\eta_{n-K'}(\boldsymbol{X}^A_{n-K',1})\big]\Big| < \epsilon_{\text{B5}}\, \mathbb{E}_g\big[\eta_{n-K'}(\boldsymbol{X}_{n-K'})\big] < \epsilon_{\text{B5}}\, Q^{|B|}. \tag{S55}$$

Combining (S52), (S53), and (S55) completes the proof of Lemma S2. □

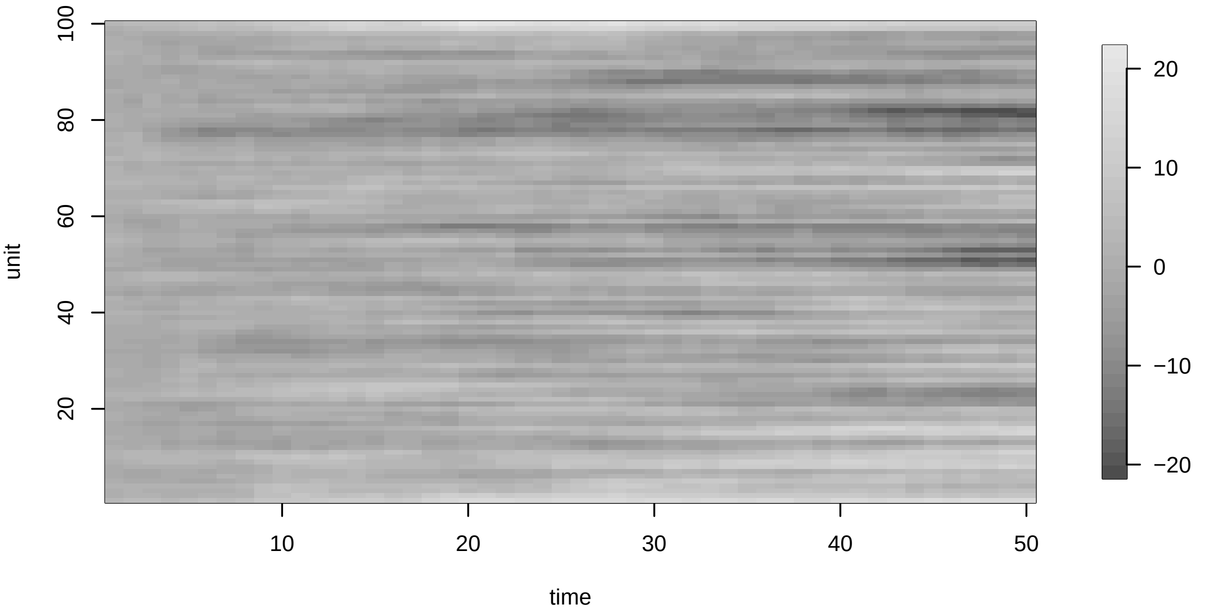

Figure S-1: correlated Brownian motion simulation used in the main text

S5 The correlated Brownian motion example

To help visualize the correlated Brownian motion model, Fig. S-1 shows one of the simulations used for the results in Figure 4 of the main text. Table S-1 gives the algorithmic settings used for the filters and corresponding computational resource requirements. Broadly speaking, $\mathcal{I}J$ for ABF and AIRSIF should be compared with $\mathcal{I}$ for UBF, J for PF, and JG for GIRF.

The computational effort allocated to each algorithm in Table S-1 is given in core minutes. UBF, ABF and ABF-IR parallelize readily, which is less true for PF and GIRF. Therefore, the UBF, ABF and ABF-IR implementations run on all available cores (36 for this experiment) whereas the PF and GIRF implementations run on a single core. If sufficient replications are being carried out to utilize all available cores, comparison of core minute utilization is equivalent to comparison of total computation time. However, a single replication of UBF, ABF or ABF-IR proceeds more quickly due to the parallelization.

ABF-IR and GIRF have computational time scaling quadratically with U in this example, whereas the other methods scale linearly. This is because the number of intermediate steps used, S, grows linearly with U.

The main purpose of this example is not to provide a comparison between the functional capabilities of the methods on interesting scientific problems. It is a toy example without the complexities that the methods are intended to address. This simple example does show clearly the quick decline of PF and the slower declines of GIRF and ABF as dimension increases.

	UBF	ABF	ABF-IR	GIRF	PF	BPF	EnKF
particles, J	—	400	200	1000	100000	20000	10000
bootstrap replications, $\mathcal{I}$	40000	400	200	—	—	—	—
guide simulations, G	—	—	—	50	—	—	—
lookahead lag, L	—	—	—	2	—	—	—
intermediate steps, S	—	—	$U/2$	U	—	—	—
neighborhood, $B_{u,n}$ or block size	$\{$(u-1,n),(u-2,n), (u,n-1),(u,n-2)$\}$			—	—	3	—
forecast mean, $\boldsymbol{\mu}(\boldsymbol{x}, s, t)$	—	—	$\boldsymbol{x}$		—	—	—
measurement mean, $h_{u,n}(x)$	—	—	x		—	—	x
$\tau = \overleftarrow{\mathrm{v}}_{u,n}(V, x)$	—	—	$\sqrt{V}$		—	—	—
$V = \overrightarrow{\mathrm{v}}_{u,n}(\tau, x)$	—	—	τ^2		—	—	τ^2
effort (core mins, $U = 100$)	0.9	2.1	4.2	0.3	0.3	0.2	0.0
effort (core mins, $U = 80$)	1.4	3.2	12.1	0.7	0.6	0.3	0.1
effort (core mins, $U = 60$)	2.2	6.0	32.0	1.9	1.3	0.6	0.2
effort (core mins, $U = 30$)	4.0	11.6	87.1	6.1	3.2	1.3	0.4
effort (core mins, $U = 10$)	7.6	26.1	311.6	47.1	9.1	3.2	1.1

Table S-1: Algorithmic settings for the correlated Brownian motion numerical example. Computational effort is measured in core minutes for running one filter, corresponding to a point on Figure 1 in the main text. The time taken for computing a single point using the parallel UBF, ABF and ABF-IR implementations is the effort divided by the number of cores, here 40. The time taken for computing a single point using the single-core GIRF, PF, BPF and EnKF implementations is equal to the effort in core minutes.

parameter	value	unit	description
$\bar{\beta}$	1560.6	year^{-1}	mean contact rate
$\mu_{\bullet D}^{-1}$	50.0	year	mean duration in the population
μ_{EI}^{-1}	7.0	day	latent period
μ_{IR}^{-1}	7.0	day	infectious period
σ_{SE}	0.150	year$^{1/2}$	process noise
a	0.500	—	amplitude of seasonality
α	1	—	mixing exponent
τ	4	year	delay from birth to entry into susceptibles
ρ	0.5	—	reporting probability
ψ	0.15	—	reporting overdispersion
G	400	—	gravitation constant
$S_u(0), u \in 1:U$	0.032	—	initial susceptible fraction
$E_u(0), u \in 1:U$	0.00005	—	initial exposed fraction
$I_u(0), u \in 1:U$	0.00004	—	initial infectious fraction

Table S-2: Parameters for the spatiotemporal measles transmission model

S6 The measles example

Table S-2 gives the model parameter values. The parameters were selected based on values found in time series analysis of single cities (He et al., 2010; Bjørnstad et al., 2002). The coupling parameter, G, was chosen so that the number of cities with simulated epidemics outside the dominant 2-year cycle is visually similar to the data. Initial values were chosen to initialize the system with the 2-year cycles observed in the UK during this period. The model code was parameterized via the basic reproduction number, $\mathcal{R}_0 = \bar{\beta}\big(\mu_{\bullet D} + \mu_{IR}\big)^{-1}$. We used $\mathcal{R}_0 = 30$, corresponding to $\bar{\beta} = 1560.6 = 30 \times (52 + 0.02)$. $\mathcal{R}_0 = 30$ is higher than most estimates for measles but consistent with estimates previously found via time series analysis of pre-vaccine UK measles in single cities (He et al., 2010; Bjørnstad et al., 2002).

Table S-3 gives the algorithmic settings used for the filters. The times in Table S-3 give the total time required by each algorithm to calculate all its results for Figure 3 in the main text, using 36 cores. The expected forecast function $\mu(x, s, t)$ needed for ABF-IR and GIRF was computed using a numerical solution to the deterministic skeleton of the stochastic model, i.e, a system of ODEs with derivative matching the infinitesimal mean function of the stochastic dynamic model. In the specifications of $h_u\big(x\big)$, $\overleftarrow{\mathrm{v}}_u(V, x, \theta)$ and $\overrightarrow{\mathrm{v}}_u(\theta, x)$, the latent process value x contains a variable C giving the cumulative removed infections in the current observation interval.

In Table S-3, we see that the effort allocated to UBF, ABF and PF scales linearly with U, since the number of bootstrap replications and particles is fixed in this experiment. GIRF computational effort scales fairly linearly in U, since its effort is dominated by the guide simulations (which are linear in U) rather than by the intermediate timestep calculations (which are quadratic in U since we carry out U intermediate calculations each of size linear in U). The effort allocated to ABF-IR scales with U^2, since ABF-IR is more parsimonious with guide simulations (all particles in one bootstrap replication share the same guide simulations) and so the intermediate timestep calculations dominate the effort. To obtain stable variance in the log likelihood estimate, the number of particles and bootstrap replications would have to grow with U. However, given a constraint on total computational resources, the number of particles and bootstrap replications would have to shrink as U increases. The limit studied in this experiment is a balance between the two: the assumption is that one is prepared to invest a growing amount of computational effort as the data grow, but this should not grow too fast. ABF-IR was permitted the greatest computational effort, but the following two considerations balance this:

1. Parallelization. UBF, ABF and ABF-IR are trivially parallelizable. The value of parallelization depends, among other things, on how many replications are being computed simultaneously and on how many cores are available. Nevertheless, it is helpful that the core minute effort requirement for ABF and ABF-IR can be divided by the number of available cores to give the computational time. Parallelizations of GIRF and PF can be constructed (Park and Ionides, 2020) but these involve non-trivial interaction between processors leading to additional algorithmic complexity and computational overhead.

2. Memory. The intermediate timestep calculations in ABF-IR and GIRF do not add to the memory requirement, and the memory demands of UBF, ABF and ABF-IR are distributed across the parallel computations. A basic PF implementation for a large model can become constrained by its memory requirement (linear in the number of particles) before it can match the processor effort employed by the other algorithms.

	UBF	ABF	ABF-IR	GIRF	EnKF	PF	BPF
particles, J	1	500	200	2000	10000	100000	20000
replicates, $\mathcal{I}$	20000	500	200	—	—	—	—
guide simulations, G	—	—	—	40	—	—	—
lookahead lag, L	—	—	—	1	—	—	—
intermediate steps, S	—	—	$U/2$	U	—	—	—
neighborhood, $B_{u,n}$ or block size	$\big\{(u,n-1),(u,n-2)\big\}$			—	—	—	2
measurement mean, $h_{u,n}(x)$	—	—	ρC			—	—
$V = \overset{\rightarrow}{\mathrm{v}}_{u,n}(\psi,\rho,x)$	—	—	$\rho(1-\rho)C + \rho^2 C^2 \psi^2$			—	—
forecast mean, $\boldsymbol{\mu}(\boldsymbol{x},s,t)$	—	—	ODE model		—	—	—
$\psi = \overset{\leftarrow}{\mathrm{v}}_{u,n}(V,x)$	—	—	$\frac{\sqrt{V-\rho(1-\rho)C}}{\rho C}$		—	—	—
effort (core mins, $U = 2$)	28.4	11.5	6.4	2.6	0.3	3.2	0.8
effort (core mins, $U = 4$)	35.7	19.0	19.0	5.5	0.6	5.9	1.5
effort (core mins, $U = 8$)	52.5	35.0	60.1	12.7	1.2	11.5	2.9
effort (core mins, $U = 16$)	87.9	66.7	217.8	36.3	2.4	22.5	5.8
effort (core mins, $U = 32$)	155.2	133.3	1032.1	133.7	4.7	45.3	11.6

Table S-3: Algorithmic settings for the measles example calculations in Figures 3 and 4. Computational effort is measured in core minutes for running one filter, corresponding to a point on Figure 3. The time taken for computing a single point using the parallel UBF, ABF and ABF-IR implementations is the effort divided by the number of cores, here 36. The time taken for computing a single point using the single core GIRF, EnKF, PF and BPF implementations is equal to the effort in core minutes.

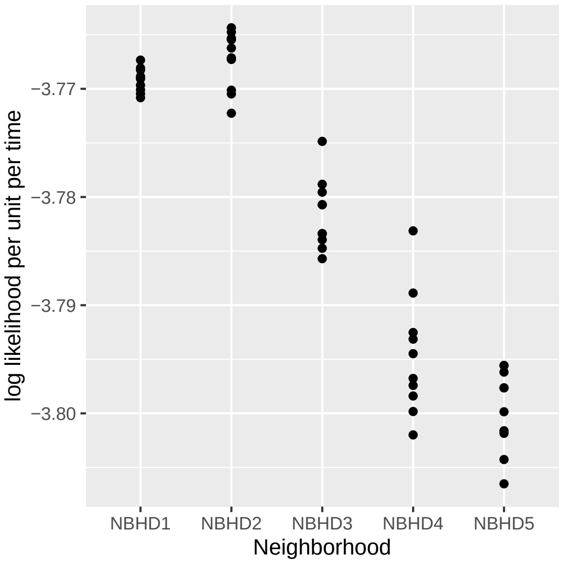

Figure S-2: log likelihood estimates for simulated data from the measles model using ABF, with varying neighborhoods.

S7 Varying the neighborhood for measles

We compared five different neighborhoods for the measles model:

NBHD1	One co-located lag	$\{(u, n-1)\}$
NBHD2	Two co-located lags	$\{(u, n-1), (u, n-2)\}$
NBHD3	Three co-located lags	$\{(u, n-1), (u, n-2), (u, n-3)\}$
NBHD4	Two co-located lags and the previous city	$\{(u, n-1), (u, n-2), (u-1, n)\}$
NBHD5	Two co-located lags and London	$\{(u, n-1), (u, n-2), (1, n)\}$

We filtered simulated data for $U = 40$ and $N = 130$, with 10 replications. We used ABF with 200 particles on each of 1000 bootstrap replications. The results are shown in Fig. S-2. Larger neighborhoods should increase the expected likelihood, but their increased Monte Carlo variability can decrease the expected log likelihood due to Jensen's inequality. In this case, we see that a neighborhood of two co-located lags provides a reasonable bias-variance tradeoff. The time taken for the above calculation was insensitive to the size of the neighborhood. The total run time for each neighborhood in Fig. S-2 was 223.0 mins for NBHD1, 225.5 mins for NBHD2, 225.4 mins for NBHD3, 218.6 mins for NBHD4, 234.8 mins for NBHD5.

S8 A Lorenz-96 example

Our primary motivation for ABF and ABF-IR is application to population dynamics arising in ecological and epidemiological models. Geophysical models provide an alternative situation involving spatiotemporal data analysis. We compare methods on the Lorenz-96 model, a nonlinear chaotic system providing a toy model for global atmospheric circulation (Lorenz, 1996; van Kekem and Sterk, 2018). We consider a stochastic Lorenz-96 model with added Gaussian process noise (Park and Ionides, 2020) defined as the solution to the following system of stochastic differential equations,

$$dX_u(t) = \big\{\big(X_{u+1}(t) - X_{u-2}(t)\big)\cdot X_{u-1}(t) - X_u(t) + F\big\}dt + \sigma_p dB_u(t), \qquad u \in 1\!:\!U. \tag{S56}$$

We define $X_0 = X_U$, $X_{-1} = X_{U-1}$, and $X_{U+1} = X_1$ so that the U spatial locations are placed on a circle. The terms $\{B_u(t), u \in 1:U\}$ denote U independent standard Brownian motions. F is a forcing constant, and we use the value $F=8$ which was demonstrated by Lorenz (1996) to induce chaotic behavior. The process noise parameter is set to $\sigma_p = 1$. The system is started with initial state $X_u(0)$ drawn as an independent normal random variable with mean 5 and standard deviation 2 for $u \in 1\!:\!U$. This initialization leads to short transient behavior. Observations are independently made for each dimension at $t_n = n$ for $n \in 1:N$ with Gaussian measurement noise of mean zero and standard deviation $\tau = 1$,

$$Y_{u,n} = X_u(t_n) + \eta_{u,n} \qquad\qquad \eta_{u,n} \sim N(0, \tau^2). \tag{S57}$$

We used an Euler-Maruyama method for numerical approximation of the sample paths of $\{\boldsymbol{X}(t)\}$, with timestep of 0.005. A simulation from this model is shown in Fig. S-4.

The ensemble Kalman filter (EnKF) is a widely used filtering method in weather forecasting for high dimensional systems (Evensen and van Leeuwen, 1996). EnKF involves a local Gaussian approximation which is problematic in highly nonlinear systems (Ades and Van Leeuwen, 2015). Methods that make local Gaussian assumptions like EnKF are necessary to scale up to the dimensions of the problems in weather forecasting. Figure S-3 shows that for a small number of units, the basic particle filter (PF) and GIRF out-perform EnKF. Then, as the number of spatial units increases, the performance of PF rapidly deteriorates whereas GIRF continues to perform well up to a moderate number of units. UBF, ABF, and particularly ABF-IR, scales well despite underperforming EnKF on this example. The additive Gaussian observation and process noise in the Lorenz-96 model is well suited to the approximations involved in EnKF. By contrast, it is less clear how to apply EnKF to discrete population non-Gaussian models such as the measles example, and how effective the resulting approximations might be.

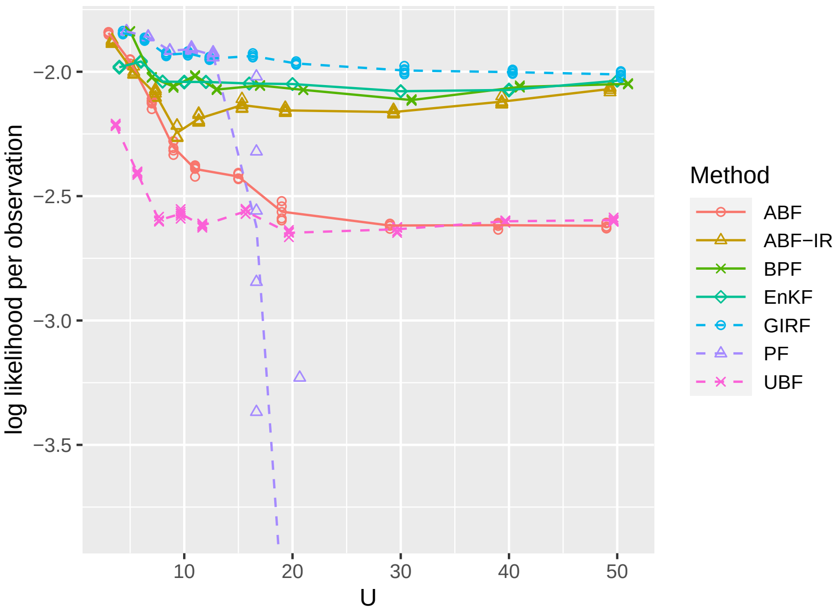

Figure S-3: Log likelihood estimates for a Lorenz-96 model of various dimensions. UBF, ABF and ABF-IR are compared with a guided intermediate resampling filter (GIRF), a standard particle filter (PF), a block particle filter (BPF) and an ensemble Kalman filter (EnKF).

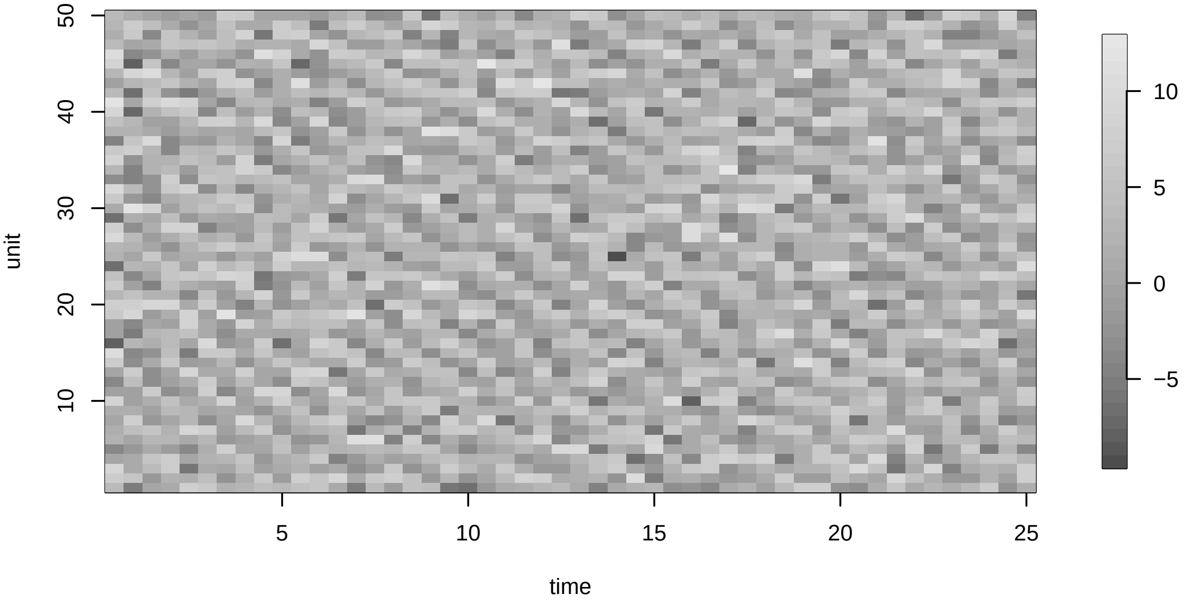

Figure S-4: Lorenz ’96 simulation used for Figure S-3

	UBF	ABF	ABF-IR	GIRF	PF	EnKF	BPF
particles, J	1	400	200	1000	100000	10000	10000
bootstrap replicates, $\mathcal{I}$	40000	400	200	—	—	—	—
guide simulations, G	—	—	—	50	—	—	—
lookahead lag, L	—	—	—	2	—	—	—
intermediate steps, S	—	—	$U/2$	U	—	—	—
neighborhood, $B_{u,n}$ or block size	$\big\{$(u,n-1),(u,n-2), (u-1,n),(u-2,n)$\big\}$			—	—	—	4
forecast mean, $\boldsymbol{\mu}(\boldsymbol{x}, s, t)$	—	—	ODE model		—	—	—
measurement mean, $h_{u,n}(x)$	—	—	x		—	x	—
$\tau = \overleftarrow{\mathrm{v}}_{u,n}(V, x)$	—	—	$\sqrt{V}$		—	—	—
$V = \overrightarrow{\mathrm{v}}_{u,n}(\tau, x)$	—	—	τ^2		—	τ^2	—
effort (core mins, $U = 4$)	34.5	5.2	5.1	2.0	2.0	0.2	0.2
effort (core mins, $U = 6$)	40.9	7.3	8.2	3.0	2.9	0.3	0.3
effort (core mins, $U = 10$)	54.5	11.2	16.3	5.0	4.8	0.5	0.5
effort (core mins, $U = 16$)	75.5	16.8	33.7	8.3	7.6	0.8	0.9
effort (core mins, $U = 50$)	202.3	48.6	174.8	34.4	23.7	2.5	2.7

Table S-4: Algorithmic settings for the Lorenz-96 numerical example. Computational effort is measured in core minutes for running one filter, corresponding to a point on Figure Figure S-3. The time taken for computing a single point using the parallel UBF, ABF and ABF-IR implementations is the effort divided by the number of cores, here 36. The time taken for computing a single point using the single-core GIRF, PF, EnKF and BPF implementations is equal to the effort in core minutes.

S9 A memory-efficient representation of ABF

The ABF pseudocode in the main text emphasizes the logical structure of the mathematical quantities computed, rather than describing a specific implementation. Arguably, an algorithm should specify not only what is to be computed, but also details of what variables should be created and saved to carry out these computations efficiently, and how computations and storage are shared across multiple locations when the algorithm is parallelized. We use the term algorithm to denote a higher level description of the quantities to be calculated and we will say that alternative pseudocodes arriving at the same quantities are representations of the algorithm. Here, we present an alternative representations of ABF which we call ABF$_2$, and we refer to the representation in the main text as ABF$_1$. The most concrete form of an algorithm is the actual computer code implementing the algorithm in a programming language. The implementation of ABF in `spatPomp`, used for numerical results in this paper, uses an embarrassingly parallel approach based on the representation ABF$_1$. This strategy facilitates robust and simple parallelization, and is appropriate when memory constraints are not limiting. To develop ABF$_2$, we set up notation similar to that used for the mathematical theory. Let

$$\gamma_{u,n,i,k} = \frac{1}{J} \sum_{j=1}^{J} \prod_{\tilde{u}:(\tilde{u},n-k)\in B_{u,n}} w^{M}_{\tilde{u},n-k,i,j} \tag{S58}$$

Also, let

$$\gamma^{+}_{u,n,i,0} = \frac{1}{J} \sum_{j=1}^{J} \prod_{(\tilde{u},n)\in B^{+}_{u,n}} w^{M}_{\tilde{u},n,i,j} \tag{S59}$$

Now set K to be the largest value of k for which $B^{[n-k]}_{u,n}$ is nontrivial for some (u,n), i.e., K is the largest temporal lag in any neighborhood. With this notation, we can write (S32) as

$$\begin{aligned} \gamma^{MC,i}_{B_{u,n}} &= \prod_{k=0}^{K} \gamma_{u,n,i,k} \\ \gamma^{MC,i}_{B^{+}_{u,n}} &= \gamma^{+}_{u,n,i,0} \prod_{k=1}^{K} \gamma_{u,n,i,k} \end{aligned}$$

This motivates the following representation of ABF.

Algorithm S2: Adapted bagged filter, representation 2 (ABF$_2$).

```
for $i$ in $1\!:\!\mathcal{I}$ do
    initialize adapted simulation: $\boldsymbol{X}^{\mathrm{A}}_{0,i} \sim f_{\boldsymbol{X}_0}(\boldsymbol{x}_0)$
    for $n$ in $1\!:\!N$ do
        proposals: $\boldsymbol{X}^{\mathrm{P}}_{n,i,j} \sim f_{\boldsymbol{X}_n|X_{1:U,n-1}}\big(\boldsymbol{x}_n \,|\, \boldsymbol{X}^{\mathrm{A}}_{n-1,i}\big)$ for $j$ in $1\!:\!J$
        measurement weights: $w^{M}_{u,n,i,j} = f_{Y_{u,n}|X_{u,n}}\big(y^{*}_{u,n} \,|\, X^{\mathrm{P}}_{u,n,i,j}\big)$ for $u$ in $1\!:\!U$, $j$ in $1\!:\!J$
        adapted resampling weights: $w^{\mathrm{A}}_{n,i,j} = \prod_{u=1}^{U} w^{M}_{u,n,i,j}$ for $j$ in $1\!:\!J$
        resampling: $\mathbb{P}\big[r(i)=a\big] = w^{\mathrm{A}}_{n,i,a}\Big(\sum_{k=1}^{J} w^{\mathrm{A}}_{n,i,k}\Big)^{-1}$
        $\boldsymbol{X}^{\mathrm{A}}_{n,i} = \boldsymbol{X}^{\mathrm{P}}_{n,i,r(i)}$
        compute $\gamma_{u,n+k,i,k}$ for $u$ in $1\!:\!U$, $k$ in $0\!:\!K$,
        compute $\gamma^{+}_{u,n,i,0}$ for $u$ in $1\!:\!U$
    end
end
$\ell^{\mathrm{MC}}_{u,n} = \log\left(\dfrac{\sum_{i=1}^{\mathcal{I}} \gamma^{+}_{u,n,i,0} \prod_{k=1}^{K} \gamma_{u,n,i,k}}{\sum_{i=1}^{\mathcal{I}} \prod_{k=0}^{K} \gamma_{u,n,i,k}}\right)$ for $u$ in $1\!:\!U$, $n$ in $1\!:\!N$
```

It is clearer from the ABF$_2$ representation than from the ABF$_1$ representation that we do not have to save every individual particle and its weight $w^{M}_{u,n,i,j}$ after the quantities $\boldsymbol{X}^{A}_{n,i}$, $\gamma_{u,n+k,i,k}$ and $\gamma^{+}_{u,n,i,0}$ have been computed. ABF$_2$ has an embarrassingly parallel implementation, since the replicates do not need to interact until they are combined to compute $\ell^{\mathrm{MC}}_{u,n}$. An embarrassingly parallel implementation therefore requires $O\big(U(KN+J)\big)$ memory for each replicate, and $O\big(UN\mathcal{I}K\big)$ memory when the results from each replicate are collected together.

By using additional communication, for example when the implementation is designed for a single core or multiple cores with shared memory, it is possible to further reduce the memory requirement. At time n, we need to save only $\boldsymbol{X}^{A}_{n,i}$ and $\gamma_{u,n-K:n+K,i,k}$ to compute $\ell^{\mathrm{MC}}_{u,n}$ and all subsequent quantities. Computing $\boldsymbol{X}^{A}_{n,i}$ requires $O(UJ)$ storage. Therefore, ABF can be implemented with memory requirement $O\big(U(K\mathcal{I}+J)\big)$, independent of N.

S10 Bagged filters for functions of the latent states

The theory and methodology in the main article focused on filtering for likelihood estimation. Here, we describe extensions to other filtering problems. Let $\{\phi_k : \boldsymbol{X}^{U} \to \mathbb{R},\ k \text{ in } 1\!:\!K\}$ be a collection of functions where ϕ_k depends only on a subset of units $\Phi_k \subset 1\!:\!U$. We suppose that there exist neighborhoods

$$B'_{k,n} \subset A'_n = (1\!:\!U) \times (0\!:\!n) \qquad \text{(S60)}$$

such that $\phi_k(\boldsymbol{X}_n)$ is approximately independent of $\{Y_{u,n} : (u,n) \in B'^{c}_{k,n}\}$ given $\{Y_{u,n} : (u,n) \in B'_{k,n}\}$, where $B'^{c}_{k,n}$ is a complement in A'_n. Unlike the sets $A_{u,n}$ and $B_{u,n}$ defined for likelihood estimation, the sets A'_n and $B'_{k,n}$ can include any locations at time n. We consider Monte Carlo estimation of

$$\overline{\phi}_{k,n} = \mathbb{E}\big[\phi_k(\boldsymbol{X}_n)\big|\boldsymbol{Y}_{1:n}\big], \quad k \text{ in } 1\!:\!K. \qquad \text{(S61)}$$

For example, if we set $\phi_u(\boldsymbol{x}_n) = x_{u,n}$ and $K = U$, the collection of quantities $\{\overline{\phi}_{u,n}\}$ estimated in (S61) corresponds to a vector of filter means. Setting $\phi_k(\boldsymbol{x}_n) = x_{u_k,n}\, x_{\tilde{u}_k,n}$ enables calculation of a collection of filter covariances between units u_k and $\tilde{u}_k$ for k in $1\!:\!K$.

We consider bagged filtering approaches to estimation of $\{\overline{\phi}_{k,n}, k \text{ in } 1:K\}$. Although each $\overline{\phi}_{k,n}$ is required to have only local dependence, some global quantities such as filter means and their variances across all units, can be expressed in terms of collections of such quantities. For bagged filtering to operate successfully, the neighborhoods $\{B'_{k,n}\}$ should not be large. Since $B'_{k,n}$ will typically be larger than $\{\Phi_k\}$, this rules out estimation of filtered quantities that cannot be adequately represented by a collection of localized filtering calculations. We now present variants of the pseudocode in the main text, targeted at estimation of $\{\overline{\phi}_{k,n}, k \text{ in } 1\!:\!K\}$. We do not prove theorems about these algorithms, but we conjecture from their similarity to the algorithms in the main text that comparable theoretical results should exist. Table S-5 lists the inputs, outputs and ranges of the implicit loops for the following algorithms.

Latent state estimation via bagged filters.

input:
collection of functions, ϕ_k
neighborhoods, $B'_{k,n}$
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, $\mathcal{I}$
data, $\boldsymbol{y}^*_{1:N}$
ABF and ABF-IR: particles per replicate, J
ABF-IR: number of intermediate timesteps, S
ABF-IR: measurement variance parameterizations, $\overleftarrow{\mathrm{v}}_{u,n}$ and $\overrightarrow{\mathrm{v}}_{u,n}$
ABF-IR: approximate process and observation mean functions, $\boldsymbol{\mu}$ and $h_{u,n}$

output:
filter estimate, $\overline{\phi}^{\,\mathrm{MC}}_{k,n}$ for k in $1\!:\!K$

Table S-5: Notation for bagged filter for latent state estimation: inputs, outputs and implicit loops.

Algorithm S3: Unadapted bagged filter for latent state estimation.

```
for $i$ in $1\!:\!\mathcal{I}$ do
    initialize simulation, $\boldsymbol{X}_{0,i} \sim f_{\boldsymbol{X}_0}(\cdot)$
    for $n$ in $1\!:\!N$ do
        simulate $\boldsymbol{X}_{n,i} \sim f_{\boldsymbol{X}_n|\boldsymbol{X}_{n-1}}\big(\cdot \mid \boldsymbol{X}_{n-1,i}\big)$
        measurement weights, $w^M_{u,n,i} = f_{Y_{u,n}|X_{u,n}}(y^*_{u,n} \mid X_{u,n,i})$ for $u$ in $1\!:\!U$
        filtering weights, $w^F_{k,n,i} = \prod_{(\tilde{u},\tilde{n})\in B'_{k,n}} w^M_{\tilde{u},\tilde{n},i}$ for $k$ in $1\!:\!K$
    end
end
$\overline{\phi}^{\,\mathrm{MC}}_{k,n} = \dfrac{\sum_{i=1}^{\mathcal{I}} \phi_k(\boldsymbol{X}_{n,i})\, w^F_{k,n,i}}{\sum_{i=1}^{\mathcal{I}} w^F_{k,n,i}}$ for $k$ in $1\!:\!K$, $n$ in $1\!:\!N$
```

Algorithm S4: Adapted bagged filter for latent state estimation.

```
for $i$ in $1\!:\!\mathcal{I}$ do
    initialize adapted simulation, $\boldsymbol{X}^{\mathrm{A}}_{0,i} \sim f_{\boldsymbol{X}_0}(\cdot)$
    for $n$ in $1\!:\!N$ do
        proposals: $\boldsymbol{X}^{\mathrm{P}}_{n,i,j} \sim f_{\boldsymbol{X}_n|X_{1:U,n-1}}\big(\boldsymbol{x}_n \mid \boldsymbol{X}^{\mathrm{A}}_{n-1,i}\big)$ for $j$ in $1\!:\!J$
        measurement weights: $w^M_{u,n,i,j} = f_{Y_{u,n}|X_{u,n}}\big(y^*_{u,n} \mid X^{\mathrm{P}}_{u,n,i,j}\big)$ for $u$ in $1\!:\!U$, $j$ in $1\!:\!J$
        adapted resampling weights: $w^{\mathrm{A}}_{n,i,j} = \prod_{u=1}^{U} w^M_{u,n,i,j}$ for $j$ in $1\!:\!J$
        resampling: $\mathbb{P}\big[r(i)=a\big] = w^{\mathrm{A}}_{n,i,a}\Big(\sum_{q=1}^{J} w^{\mathrm{A}}_{n,i,q}\Big)^{-1}$
        $\boldsymbol{X}^{\mathrm{A}}_{n,i} = \boldsymbol{X}^{\mathrm{P}}_{n,i,r(i)}$
        $w^F_{k,n,i,j} = \prod_{\tilde{n}=1}^{n-1}\Big[\dfrac{1}{J}\sum_{q=1}^{J}\prod_{\tilde{u}:(\tilde{u},\tilde{n})\in B'_{k,n}} w^M_{\tilde{u},\tilde{n},i,q}\Big]\prod_{\tilde{u}:(\tilde{u},n)\in B'_{k,n}} w^M_{\tilde{u},n,i,j}$ for $j$ in $1\!:\!J$, $k$ in $1\!:\!K$
    end
end
$\overline{\phi}^{\,\mathrm{MC}}_{k,n} = \dfrac{\sum_{i=1}^{\mathcal{I}}\sum_{j=1}^{J} \phi_k(\boldsymbol{X}^{\mathrm{P}}_{n,i,j})\, w^F_{k,n,i,j}}{\sum_{i=1}^{\mathcal{I}}\sum_{j=1}^{J} w^F_{k,n,i,j}}$ for $k$ in $1\!:\!K$, $n$ in $1\!:\!N$
```

Algorithm S5: ABF-IR for latent state estimation.

```
for $i$ in $1\!:\!\mathcal{I}$ do
    initialize adapted simulation, $\boldsymbol{X}^{\mathrm{A}}_{0,i} \sim f_{\boldsymbol{X}_0}(\cdot)$
    for $n$ in $1\!:\!N$ do
        guide simulations: $\boldsymbol{X}^{G}_{n,i,j} \sim f_{\boldsymbol{X}_n|\boldsymbol{X}_{n-1}}\big(\boldsymbol{x}_n \,|\, \boldsymbol{X}^{\mathrm{A}}_{n-1,i}\big)$ for $j$ in $1\!:\!J$
        $V_{u,n,i} = \mathrm{Var}\big\{h_{u,n}\big(X^{G}_{u,n,i,j}\big),\ j \text{ in } 1\!:\!J\big\}$
        $g^{\mathrm{R}}_{n,0,i,j} = 1$ and $\boldsymbol{X}^{\mathrm{IR}}_{n,0,i,j} = \boldsymbol{X}^{\mathrm{A}}_{n-1,i}$ for $j$ in $1\!:\!J$
        for $s$ in $1\!:\!S$ do
            $\boldsymbol{X}^{\mathrm{IP}}_{n,s,i,j} \sim f_{\boldsymbol{X}_{n,s}|\boldsymbol{X}_{n,s-1}}\big(\cdot\,|\boldsymbol{X}^{\mathrm{IR}}_{n,s-1,i,j}\big)$ for $j$ in $1\!:\!J$
            $\boldsymbol{\mu}^{\mathrm{IP}}_{n,s,i,j} = \boldsymbol{\mu}\big(\boldsymbol{X}^{\mathrm{IP}}_{n,s,i,j}, t_{n,s}, t_n\big)$ for $j$ in $1\!:\!J$
            $V^{\mathrm{meas}}_{u,n,s,i,j} = \overrightarrow{\mathrm{v}}_u(\theta, \mu^{\mathrm{IP}}_{u,n,s,i,j})$ for $u$ in $1\!:\!U$, $j$ in $1\!:\!J$
            $V^{\mathrm{proc}}_{u,n,s,i} = V_{u,n,i}\,\big(t_n - t_{n,s}\big)\Big/\big(t_n - t_{n,0}\big)$ for $u$ in $1\!:\!U$
            $\theta_{u,n,s,i,j} = \overleftarrow{\mathrm{v}}_u\big(V^{\mathrm{meas}}_{u,n,s,i,j} + V^{\mathrm{proc}}_{u,n,s,i},\ \mu^{\mathrm{IP}}_{u,n,s,i,j}\big)$ for $u$ in $1\!:\!U$, $j$ in $1\!:\!J$
            $g_{n,s,i,j} = \prod_{u=1}^{U} f_{Y_{u,n}|X_{u,n}}\big(y^*_{u,n} \,|\, \mu^{\mathrm{IP}}_{u,n,s,i,j}\,; \theta_{u,n,s,i,j}\big)$ for $j$ in $1\!:\!J$
            guide weights: $w^{G}_{n,s,i,j} = g_{n,s,i,j}\big/g^{\mathrm{R}}_{n,s-1,i,j}$ for $j$ in $1\!:\!J$
            resampling: $\mathbb{P}\big[r(i,j)=a\big] = w^{G}_{n,s,i,a}\Big(\sum_{k=1}^{J} w^{G}_{n,s,i,k}\Big)^{-1}$ for $j$ in $1\!:\!J$
            $\boldsymbol{X}^{\mathrm{IR}}_{n,s,i,j} = \boldsymbol{X}^{\mathrm{IP}}_{n,s,i,r(i,j)}$ and $g^{\mathrm{R}}_{n,s,i,j} = g_{n,s,i,r(i,j)}$ for $j$ in $1\!:\!J$
        end
        $\boldsymbol{X}^{\mathrm{A}}_{n,i} = \boldsymbol{X}^{\mathrm{IR}}_{n,S,i,1}$
        $w^{M}_{u,n,i,j} = f_{Y_{u,n}|X_{u,n}}\big(y^*_{u,n} \,|\, X^{G}_{u,n,i,j}\big)$ for $u$ in $1\!:\!U$, $j$ in $1\!:\!J$
        $w^{F}_{k,n,i,j} = \prod_{\tilde{n}=1}^{n-1}\Big[\frac{1}{J}\sum_{q=1}^{J}\prod_{\tilde{u}:(\tilde{u},\tilde{n})\in B'_{k,n}} w^{M}_{\tilde{u},\tilde{n},i,q}\Big]\prod_{\tilde{u}:(\tilde{u},n)\in B'_{k,n}} w^{M}_{\tilde{u},n,i,j}$ for $k$ in $1\!:\!K$, $u$ in $1\!:\!U$
    end
end
$\overline{\phi}^{\,\mathrm{MC}}_{k,n} = \dfrac{\sum_{i=1}^{\mathcal{I}}\sum_{j=1}^{J}\phi_k(\boldsymbol{X}^{\mathrm{P}}_{n,i,j})\,w^{F}_{k,n,i,j}}{\sum_{i=1}^{\mathcal{I}}\sum_{j=1}^{J} w^{F}_{k,n,i,j}}$ for $k$ in $1\!:\!K$, $n$ in $1\!:\!N$
```

S11 An iterated bagged filter for parameter estimation

This paper focuses on evaluating the likelihood function for SpatPOMP models via filtering. Although the likelihood function is fundamental for inference, evaluation alone is not sufficient. Likelihood maximization enables calculation of the maximum likelihood estimate, profile likelihood confidence intervals, likelihood ratio tests and likelihood-based model selection critera. Iterated filtering methodology (Ionides et al., 2006, 2011, 2015) provides an approach to extending filtering algorithms to likelihood maximization algorithms. Here, we demonstrate one such extension in the context of bagged filters. This demonstration provides a proof of concept to motivate future work.

Iterated filtering approaches apply filtering to a modified version of the model where parameters are perturbed at each time point. The filtering procedure directs the perturbed parameters toward values consistent with the data. At the end of each filtering operation, a parameter updating rule is applied and a new filtering iteration is started with reduced perturbation variance. Under suitable conditions, iterative procedures of this type converge to a neighborhood of a maximum likelihood parameter despite the presence of Monte Carlo filtering error. We implemented an iterated bagged filter procedure, described by the pseudocode below. The pseudocode is presented for an iterated adapted bagged filter (IABF) but the iterated unadapted filter (IUBF) corresponds to the case $J = 1$ and the iterated bagged filter with intermediate resampling (IABF-IR) follows by adding the intermediate resampling procedure used by ABF-IR. The filtering step of IABF uses ABF to estimate the likelihood at the K perturbed parameter sets. The selection step does not resample parameters based on these estimated likelihood. Rather, it selects the parameters with the top p quantile of likelihoods and copies them appropriately to get K new parameters for the next filtering step. This quantile-based resampling allows us to maintain the diversity of the K parameter sets and avoid *parameter degeneracy*, whereby very few parameters are resampled, leading to an inefficient search of parameter space.

For simplicity, in this description, we assume that parameters are transformed so that their values are unconstrained. Our software implementation, provided the R package `spatPomp` (Asfaw et al., 2021), provides facilities for carrying out such transformations. The Gaussian distribution used for perturbations, and the geometric perturbation variance reduction factor, α, are convenient specifications but are not required in theory (Ionides et al., 2015). As another simplification, the pseudocode for IABF represents the logical structure of the algorithm without attending to issues of memory management and parallelization. For implementation issues, we refer to `spatPomp` (Asfaw et al., 2021).

In Figure 5 of the main text, we use this IABF implementation to construct a profile likelihood for the measles model. We use $J = 1$, $\mathcal{I} = 30000$, $K = 250$, $p = 0.8$, $\alpha = 0.5$, $M = 15$ and Σ set to be a diagonal matrix with perturbation variance for each non-initial value parameter set to 0.02. For this exercise, we fix the initial value parameters at their true values.

The resulting Monte Carlo parameter estimates have Monte Carlo uncertainty in both likelihood evaluation and maximization. Therefore, we used a Monte Carlo adjusted profile likelihood (Ionides et al., 2017; Ning et al., 2021) that accounts for this uncertainty. Fig. 5gives empirical demonstration of this procedure on the measles model.

Algorithm S6: Iterated adapted bagged filter (IABF).

```
input: same as ABF table in main text plus: maximization iterations, $M$; number of
       parameter vectors, $K$; perturbation variance, $\Sigma$; variance reduction factor after 50
       iterations, $\alpha$; starting parameters, $\theta^{(0)}_{1:K}$; resampling proportion, $p$
for $m$ in $1\!:\!M$ do
    $\theta^F_{0,k} = \theta^{(m-1)}_k$ for $k$ in $1\!:\!K$
    initialize adapted simulation: $\boldsymbol{X}^F_{0,i,k} \sim f_{\boldsymbol{X}_0}\big(\boldsymbol{x}_0\,;\theta^F_{0,k}\big)$ for $k$ in $1\!:\!K$
    for $n$ in $1\!:\!N$ do
        $\theta^P_{n,k} \sim \mathcal{N}\big[\theta^F_{n-1,k}, \alpha^{2m/50}\Sigma\big]$ for $k$ in $1\!:\!K$
        for $i$ in $1\!:\!\mathcal{I}$ do
            $\boldsymbol{X}^{\mathrm{P}}_{n,i,j,k} \sim f_{\boldsymbol{X}_n|\boldsymbol{X}_{n-1}}\big(\boldsymbol{x}_n \,|\, \boldsymbol{X}^F_{n-1,i,k}\,;\theta^P_{n,k}\big)$ for $j$ in $1\!:\!J$, $k$ in $1\!:\!K$
            $w^M_{u,n,i,j,k} = f_{Y_{u,n}|X_{u,n}}\big(y^*_{u,n} \,|\, X^{\mathrm{P}}_{u,n,i,j,k}\,;\theta^P_{n,k}\big)$ for $u$ in $1\!:\!U$, $j$ in $1\!:\!J$, $k$ in $1\!:\!K$
            $w^{\mathrm{A}}_{n,i,j,k} = \prod_{u=1}^U w^M_{u,n,i,j,k}$ for $j$ in $1\!:\!J$, $k$ in $1\!:\!K$
            resampling: $\mathbb{P}\big[r(i,k)=a\big] = w^{\mathrm{A}}_{n,i,a,k}\Big(\sum_{\xi=1}^J w^{\mathrm{A}}_{n,i,\xi,k}\Big)^{-1}$ for $k$ in $1\!:\!K$
            $\boldsymbol{X}^{\mathrm{A}}_{n,i,k} = \boldsymbol{X}^{\mathrm{P}}_{n,i,r(i,k),k}$ for $k$ in $1\!:\!K$
            $w^{\mathrm{P}}_{u,n,i,j,k} = \prod_{\tilde{n}=1}^{n-1}\Big[\frac{1}{J}\sum_{\xi=1}^{J}\prod_{\tilde{u}:(\tilde{u},\tilde{n})\in B_{u,n}} w^M_{\tilde{u},\tilde{n},i,\xi,k}\Big]\prod_{\tilde{u}:(\tilde{u},n)\in B_{u,n}} w^M_{\tilde{u},n,i,j,k}$
              for $u$ in $1\!:\!U$, $j$ in $1\!:\!J$, $k$ in $1\!:\!K$
        end
        $\ell^{\mathrm{MC}}_{n,k} = \sum_{u=1}^{U}\log\left(\frac{\sum_{i=1}^{\mathcal{I}}\sum_{j=1}^{J} w^M_{u,n,i,j,k}w^P_{u,n,i,j,k}}{\sum_{i=1}^{\mathcal{I}}\sum_{j=1}^{J} w^P_{u,n,i,j,k}}\right)$ for $k$ in $1\!:\!K$
        select the highest $pK$ likelihoods: find $s$ with
          $\{s(1),\dots,s(\lceil pK\rceil)\} = \big\{k : \sum_{\tilde{k}=1}^K \mathbf{1}\{\ell^{\mathrm{MC}}_{n,\tilde{k}} > \ell^{\mathrm{MC}}_{n,k}\} < (1-p)K\big\}$
        make $1/p$ copies of successful parameters, $\theta^F_{n,k} = \theta^P_{n,s(\lceil pk\rceil)}$ for $k$ in $1\!:\!K$
        $\boldsymbol{X}^F_{n,i,k} = \boldsymbol{X}^A_{n,i,s(\lceil pk\rceil)}$ for $i$ in $1\!:\!\mathcal{I}$, $k$ in $1\!:\!K$
    end
    $\theta^{(m)}_k = \theta^F_{N,k}$ for $k$ in $1\!:\!K$
end
output: Parameter estimates approaching the maximum likelihood estimate, $\theta^{(M)}_{1:K}$
```

S12 Replicates versus particles for the measles model

It is necessary in practice to decide whether computational resources are best directed toward a large number of replicates, $\mathcal{I}$, or a large number of particles per replicate, J. Computational effort for ABF and ABF-IR is approximately proportional to $J\mathcal{I}$, and UBF corresponds to ABF with $J = 1$. Suitable algorithmic parameters may depend on the model under consideration, and here we consider resource allocation for the measles model studied in the main text, using simulated data with $U = 40$ and $N = 104$. From Figure 4, we know that this implementation of the model is well suited to ABF and UBF. ABF-IR performs less well, and the weak performance of GIRF

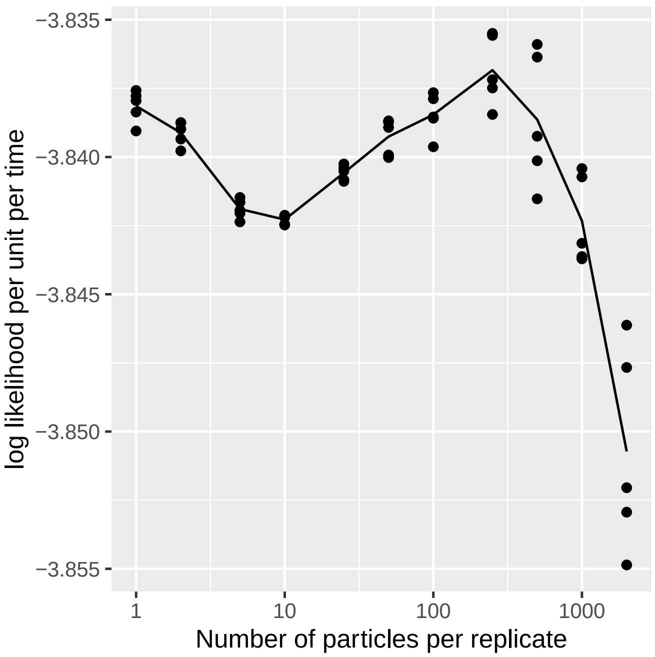

Figure S-5: log likelihood estimates for simulated data from the measles model using ABF, with varying algorithmic parameters.

suggests that the weakness may be due to an inadequate guide function. Figure S-5 investigates the tradeoff between UBF and ABF by plotting evaluated log likelihood against $\mathcal{I}$ with J chosen to give approximately a constant computational effort. Algorithmic settings and run times are reported in Table S-6. For our implementation, choosing J very low and $\mathcal{I}$ correspondingly high led to greater computation time, perhaps because the code was written to parallelize nicely when J is relatively large.

We interpret the bimodal curve as follows. When ABF is carried out with an inadequate number of particles for each bootstrap replicate, the algorithm cannot make a good representation of a draw from the adapted distribution. In that case, there is an advantage to using UBF, which does not attempt to carry out adapted simulation. For very small numbers of particles per replicate, the ABF algorithm behaves like a not-quite-properly-weighted version of UBF. Large numbers of particles per replicate presumably lead to improved Monte Carlo representation of draws from the adapted distribution, but computational cost constraints prevent combining this with a large number of replicates. We see a mode around 500 particles per replicate where ABF out-performs UBF. On this problem, UBF is relatively successful, presumably because the measles dynamics in each city are strongly attracted toward relatively few stable cycles (annual epidemics, or peaks in odd years, or peaks in even years) and a tractable number of simulations can represent all these scenarios. The Lorenz model of Sec. S8 provides an alternative situation, where adaptation has more advantages. Also, the plug-and-play guide function appears to operate successfully in this case, as evidenced by relatively strong performance from ABF-IR and GIRF.

J	$\mathcal{I}$	time
2000	500	70.7
1000	1000	71.3
500	2000	72.4
250	4000	75.0
100	10000	80.9
50	20000	93.8
25	40000	114.9
10	40000	71.4
5	40000	56.1
2	40000	47.7
1	40000	43.0

Table S-6: Bootstrap replications, $\mathcal{I}$, particles per replicate, J, and computational time (total minutes for the five points) for the results presented in Figure S-5

	UBF	ABF	ABF-IR	EnKF	BPF
Monte Carlo	100	10 × 10	10 × 10	100	100
sample size,	200	20 × 10	20 × 10	200	200
$\mathcal{I} \times J$	400	40 × 10	40 × 10	400	400
	800	80 × 10	80 × 10	800	800
	1600	160 × 10	160 × 10	1600	1600
	3200	160 × 20	160 × 20	3200	3200
	6400	320 × 20	320 × 20	6400	6400
	12800	320 × 40	320 × 40	12800	12800
	25600	640 × 40	640 × 40	25600	25600
	51200	1280 × 40	1280 × 40	51200	51200
intermediate steps, S	—	—	$U/2$	—	—
effort, in core mins.	2.8	0.7	5.5	1.0	1.3
	3.4	0.7	6.1	2.0	2.5
	4.5	1.0	10.9	4.0	4.8
	6.5	1.3	15.6	8.1	9.6
	11.2	2.0	24.4	16.1	19.1
	20.3	2.7	46.7	32.1	38.6
	37.6	4.4	82.6	64.6	77.4
	70.6	6.9	174.3	129.5	156.1
	137.9	13.1	344.4	260.1	314.7
	276.2	25.3	686.3	526.2	636.6

Table S-7: Algorithmic parameters and run times (in core-minutes) that correspond to the effort levels shown in Figure S-6.

S13 Varying Monte Carlo effort for measles with fixed U

The theoretical results in our manuscript concern sub-exponential computational requirements as U increases. However, the behavior as effort varies for fixed U has practical consequences for the problems on which different methods are applicable. We therefore investigated empirically how the variance of our estimates scales with effort across multiple methods, on the measles model. We fixed our data to be the 20-city measles data from Figure 3 in the main text. Table S-7 below shows the schedule of Monte Carlo effort used to compute the likelihood estimates in Figure S-6. For ABF and ABF-IR we vary $\mathcal{I}J$; for UBF we vary $\mathcal{I}$; for BPF and EnKF we vary J.

If computational considerations prevented an ensemble size larger than 100, EnKF or UBF would be a good choice on this problem. EnKF cannot benefit much from a larger ensemble size, since its limitation is the Gaussian approximation in its update rule, rather than Monte Carlo variability. For an ensemble size of 1000 or more, the bagged filters and block particle filter provide a substantial improvement over EnKF. ABF-IR is relatively computationally demanding on this problem, perhaps because further research is needed into the choice of guide function.

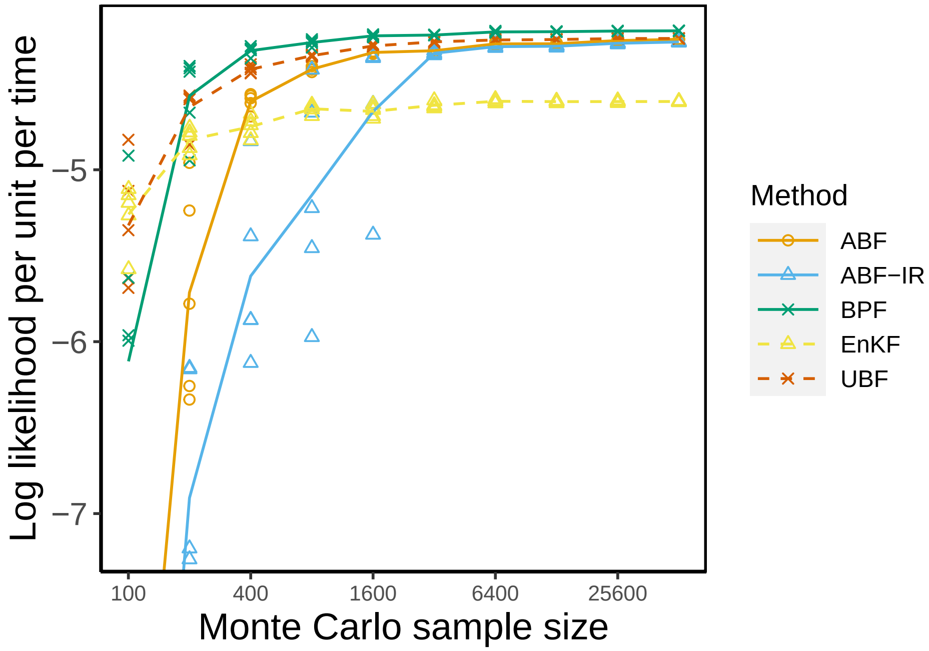

Figure S-6: Exploring the efficiency of the bagged filter methods with varying Monte Carlo effort. For each of 10 levels of computational effort, we run our methods on a simulated measles data set from 20 cities. The variability of our estimates decreases as effort increases and we observe that the unadapted bagged filter has a competitive advantage over BPF in a low computational effort regime. Methods that perform less well are clipped off to focus on the best methods for this example. The algorithmic parameters and run times are given in Table S-7.

	PF	EnKF	UBF	ABF	BPF
Log likelihood per observation	-1.82	-1.83	-1.89	-1.87	-3.75
Monte Carlo effort, $\mathcal{I} \times J$	10000	10000	10000	100×100	10000

Table S-8: Numerical results for likelihood evaluation on the model with a linear constraint given in equations (S62) and (S63).

S14 A constraint that breaks the block particle filter

A potential advantage of the bagged filters over the block particle filter is that the former have filter trajectories preserving constraints on the system, whereas the latter does not. Potentially, conservation constraints could be an important part of a coupled model: physical matter, or money, or members of a biological population can move from location to location but cannot so easily be spontaneously created. We present a toy example to demonstrate the issue.

Suppose $\boldsymbol{W}(t) = W_{1:U}(t)$ is a collection of U independent standard Brownian motions, and let $\boldsymbol{X}(t)$ be the Ito solution to a system of stochastic differential equations,

$$dX_u(t) = \sigma \left(dW_u(t) - \frac{1}{U} \sum_{\tilde{u}=1}^{U} dW_{\tilde{u}}(t) \right) + \sum_{\tilde{u}=1}^{U} X_{\tilde{u}}(t)\, dt, \tag{S62}$$

for $u \in 1\!:\!U$, with initial condition $X_u(0) = 0$. We consider additive Gaussian measurement error,

$$Y_{u,n} = X_u(n) + \tau \epsilon_{u,n}, \tag{S63}$$

where $\{\epsilon_{u,n}\}$ is a collection of iid standard normal random variables. This toy model has stable trajectories constrained to the linear space $\sum_{\tilde{u}=1}^{U} X_{\tilde{u}}(t) = 0$, but trajectories diverge if that constraint is broken. A small demonstration is enough to make the point that the block particle filter can behave very poorly on this problem. Results are reported in Table S-8 with with $U = 5$, $N = 10$, $\sigma = 1$, $\tau = 1$, a block size of 1 for BPF, neighborhoods for UBF and ABF consisting of two co-located lags, and an Euler solution to (S62) with time step 0.2. Any block size less than a single block of size U gives the same qualitative conclusion of poor performance of BPF.

The Monte Carlo effort in Table S-8 is large for all methods compared to the size of the problem, so the results primarily reflect inherent bias in the approaches rather than Monte Carlo variability. On this small problem, PF can provide an accurate likelihood estimate. EnKF preserves linear constraints, but does not preserve nonlinear constraints. Also, the system is Gaussian, which matches the approximation inherent in EnKF. Therefore, EnKF closely matches PF here, but an example with a nonlinear constraint could also break EnKF. PF and the bagged filters preserve linear and nonlinear constraints.

S15 Deriving Assumption B6 from a previous intermediate resampling result

Intermediate resampling has been studied in the context of filtering highly informative observations (Del Moral and Murray, 2015) and high-dimensional observations (Park and Ionides, 2020). Adapted simulation is a special case of filtering with a single observation ($N = 1$) and a known initial condition. We show that a theorem from Park and Ionides (2020) implies Assumption B6 holds with the constant $\mathcal{C}_0$ being polynomial in the number of units, U, when carrying out intermediate resampling with an ideal guide function. Recall the statement of Assumption B6:

Assumption B6. *Let h be a bounded function with $|h(x)| \leq 1$. Let $\boldsymbol{X}^{\mathrm{IR}}_{n,S,j,i}$ be the Monte Carlo quantity constructed in ABF-IR, conditional on $\boldsymbol{X}^{\mathrm{A}}_{n-1,S,i} = \boldsymbol{x}^{\mathrm{A}}_{n-1,S,i}$. There is a constant $\mathcal{C}_0(U, N, S)$ such that, for all $\epsilon_{\mathrm{B6}} > 0$ and $\boldsymbol{x}^{\mathrm{A}}_{n-1,S,i}$, whenever the number of particles satisfies $J > \mathcal{C}_0(U, N, S)/\epsilon^3_{\mathrm{B6}}$,*

$$\mathbb{E}\left|\frac{1}{J}\sum_{j=1}^{J} h(\boldsymbol{X}^{\mathrm{IR}}_{n,S,j,i}) - \mathbb{E}_g\big[h(\boldsymbol{X}_n)\,|\,\boldsymbol{X}_{n-1} = \boldsymbol{x}^{\mathrm{A}}_{n-1,S,i}\big]\right| < \epsilon_{\mathrm{B6}}.$$

We check this follows as a consequence of the following theorem, in which $\boldsymbol{X}^{F,j}_{t_N}$ is the j*th* filter particle at the Nth observation time, t_N, constructed using the guided intermediate resampling filter (GIRF) algorithm of Park and Ionides (2020) with an ideal guide function. We do not introduce the regularity conditions required for this theorem.

Theorem 2 (Park and Ionides, 2020). *Suppose multinomial resampling is used in GIRF, with an ideal guide function. Under regularity conditions, there is a constant $C_1^* > 0$ such that for any real-valued function $h(\boldsymbol{x})$ with $\|h\|_\infty \leq 1$ and any constant $a > 1$,*

$$\left|\frac{1}{J}\sum_{j=1}^{J} h(\boldsymbol{X}^{F,j}_{t_N}) - \mathbb{E}\big[h(\boldsymbol{X}_{t_N})|\boldsymbol{Y}_{1:N} = \boldsymbol{y}_{1:N}\big]\right| \leq \frac{4a(C_1^*+1)}{\sqrt{J}}(NS+1) \tag{S64}$$

with probability at least

$$1 - \frac{(2NS+1)(NS+1)}{a^2}, \tag{S65}$$

given that $\sqrt{J} \geq 8a(C_1^*+1)NS$.

In our context, $N = 1$ since we are carrying out guided sampling over one timestep. Now, set

$$\frac{(2NS+1)(NS+1)}{a^2} < \epsilon/2, \tag{S66}$$

say,

$$a = \sqrt{\frac{2(2S+1)(S+1)}{\epsilon}}. \tag{S67}$$

We want J large enough that

$$\frac{4a(C_1^*+1)}{\sqrt{J}}(NS+1) < \epsilon/2, \tag{S68}$$

since then the bounds in (S64) and (S65), together with the bound on h, give the bound in expectation in Assumption B6. Putting (S67) into (S68) gives

$$\frac{4(C_1^*+1)\sqrt{2(2S+1)(S+1)}}{\sqrt{\epsilon J}}(S+1) < \epsilon/2. \tag{S69}$$

and so

$$J > \frac{128(2S+1)(S+1)^3(C_1^*+1)^2}{\epsilon^3} \tag{S70}$$

which implies that we can pick

$$\mathcal{C}_0 = 128(2S+1)(S+1)^3(C_1^*+1)^2 \tag{S71}$$

in Assumption B6.

The constant C_1^* depends on the number of intermediate resapling intervals, S. Proposition 1 of Park and Ionides (2020) shows that C_1^* in Theorem 2 can be bounded as a function of U if intermediate resampling is carried out with $S = U$.

In practical situations, one can only approximate an ideal guide function, leading to an exponential bound on $\mathcal{C}_0$ which is small if the guide function is effective. ABF-IR is less sensitive to the choice of guide function than GIRF, since the likelihood is calculated using local weights and the intermediate resampling is used only for the adapted simulation.

Supplementary References

Ades, M. and Van Leeuwen, P. J. (2015). The equivalent-weights particle filter in a high-dimensional system. *Quarterly Journal of the Royal Meteorological Society*, 141(687):484–503.

Asfaw, K., Park, J., Ho, A., King, A. A., and Ionides, E. L. (2021). Statistical inference for spatiotemporal partially observed Markov processes via the R package spatpomp. *arXiv:2101.01157*.

Bjørnstad, O. N., Finkenstädt, B. F., and Grenfell, B. T. (2002). Dynamics of measles epidemics: Estimating scaling of transmission rates using a time series SIR model. *Ecological Monographs*, 72(2):169–184.

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

Evensen, G. and van Leeuwen, P. J. (1996). Assimilation of geostat altimeter data for the Agulhas Current using the ensemble Kalman filter with a quasigeostrophic model. *Monthly Weather Review*, 124:58–96.

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., Bhadra, A., Atchadé, Y., and King, A. A. (2011). Iterated filtering. *Annals of Statistics*, 39:1776–1802.

Ionides, E. L., Bretó, C., and King, A. A. (2006). Inference for nonlinear dynamical systems. *Proceedings of the National Academy of Sciences of the USA*, 103:18438–18443.

Ionides, E. L., Breto, C., Park, J., Smith, R. A., and King, A. A. (2017). Monte Carlo profile confidence intervals for dynamic systems. *Journal of the Royal Society Interface*, 14:1–10.

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(3):719—-724.

Liu, J. S. (2001). *Monte Carlo Strategies in Scientific Computing.* Springer, New York.

Lorenz, E. N. (1996). Predictability: A problem partly solved. *Proceedings of the Seminar on Predictability*, 1:1–18.

Ning, N., Ionides, E. L., and Ritov, Y. (2021). Scalable Monte Carlo inference and rescaled local asymptotic normality. *Bernoulli*, pre-published online.

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

van Kekem, D. L. and Sterk, A. E. (2018). Travelling waves and their bifurcations in the Lorenz-96 model. *Physica D: Nonlinear Phenomena*, 367:38–60.