An iterated block particle filter for inference on coupled dynamic systems with shared and unit-specific parameters

Edward. L. Ionides, Ning Ning and Jesse Wheeler

Supplementary material

Contents

S1 Parameters for the measles model

Parameter	$\mathcal{A}$	$\mathcal{B}$	$\mathcal{C}$	Simulation
$\pi_{S,u}$	unit-specific	unit-specific	unit-specific	0.032
$\pi_{E,u}$	unit-specific	unit-specific	unit-specific	0.00005
$\pi_{I,u}$	unit-specific	unit-specific	unit-specific	0.00004
ρ_u	unit-specific	unit-specific	unit-specific	0.5
τ_u	shared	unit-specific	unit-specific	0.15
$\mu_{EI,u}$	shared	unit-specific	unit-specific	$1\,\text{week}^{-1}$
μ_{IR}	shared	unit-specific	unit-specific	$1\,\text{week}^{-1}$
$\overline{\beta}_u$	shared	unit-specific	unit-specific	$30\,\text{week}^{-1}$
$\sigma_{SE,u}$	shared	unit-specific	unit-specific	$0.15\,\text{year}^{1/2}$
h_u	shared	unit-specific	unit-specific	0.5
α_u	shared	unit-specific	unit-specific	1
c	shared	unit-specific	unit-specific	0
G_u	shared	unit-specific	0	400
ι_u	0	0	unit-specific	0

Table S1: Parameters for the measles model. Sub-model $\mathcal{A}$ has 4 unit-specific parameters and 9 shared parameters, with movement between units following a gravity equation. Sub-model $\mathcal{B}$ has 13 unit-specific parameters and no shared parameters, with gravity movement. Sub-model $\mathcal{C}$ has 13 unit-specific parameters and no shared parameters, with independent immigration of infections rather than movement between units. The last column shows the parameter values used for the simulated data in Figure 2 of the main text.

Notes on Table S1:

1. In the model formulation, all parameters are written as unit-specific, i.e., with a u subscript. Shared parameters take equal values across all units.

2. For the simulation, all parameters are shared, so one can assess the inferential consequences of estimating models with some or all parameters unit-specific. Establishing a good choice of parameters to be

shared or unit-specific for the data is a data analysis goal.

3. The initial values of the latent states are parameterized as fractions of the total population, so $S_{0,u} = \pi_{S,u} P_u(t_0)$, $E_{0,u} = \pi_{E,u} P_u(t_0)$, $I_{0,u} = \pi_{I,u} P_u(t_0)$, and $R_{0,u} = \big(1 - \pi_{S,u} - \pi_{E,u} - \pi_{I,u}\big) P_u(t_0)$.

4. The parameterization used in the numerical implementation replaces $\overline{\beta}_u$ with the basic reproductive number, a dimensionless ratio defined as $\mathcal{R}_{0,u} = \overline{\beta}_u \big(\mu_{IR,u} + \mu_D\big)^{-1}$. For the simulation, $\mathcal{R}_{0,u} = 30$.

S2 Algorithmic parameters and transformations

Parameter	Value
J	4000
M	100
$\breve{\sigma}$	0.005 (0.00125 on some searches for the simulated data)
r	0.1
a	0.5

Table S2: Algorithmic parameters used for applying IBPF to the measles model. The same values were used for each of the sub-models $\mathcal{A}$, $\mathcal{B}$ and $\mathcal{C}$.

Notes on Table S2:

1. The full $D \times N$ matrix of perturbation standard deviations with entries $\sigma_{d,n}$ was reduced to a single algorithmic parameter, $\breve{\sigma}$, via

$$\sigma_{d,n} = \begin{cases} \breve{\sigma} & \text{if } \theta_d \text{ is not an IVP and not } \alpha_u \\ 0.1\,\breve{\sigma} & \text{if } \theta_d \text{ is } \alpha_u \\ 2\,\breve{\sigma} & \text{if } \theta_d \text{ is an IVP, and } n = 0 \\ 0 & \text{if } \theta_d \text{ is an IVP, and } n \geq 1 \end{cases}$$

   where the initial value parameters (IVPs) are $\pi_{S,u}$, $\pi_{E,u}$ and $\pi_{I,u}$.

2. Parameters perturbations were carried out on a transformed scale. A logit transform was used for $\pi_{S,u}$, $\pi_{E,u}$, $\pi_{I,u}$, ρ_u. A log transformation was used for τ_u, $\sigma_{SE,u}$, $\mu_{EI,u}$, $\mu_{IR,u}$, $\overline{\beta}_u$, α_u, G_u, ι_u. No transformation was used for h_u.

3. The algorithmic parameters do not have any scientific significance once successful maximization has been demonstrated. They may affect the ease of successful maximization, or even the ability to attain this within an acceptable level of Monte Carlo uncertainty.

4. $M = 100$ was chosen empirically. For a computationally challenging maximization problem, we expect to carry out many searches from a variety of starting values, and we conduct further experiments following up on successful leads. In this case, it is enough to choose M so that each search has a fair chance of finding a higher likelihood when there are local improvements to be made.

5. Numerical experiments were carried out to choose J. We look for the smallest J such that there is not much to be gained by making J larger. This is carried out by looking for evidence about the bias and variance (discussed further in Sec. S3) which both need consideration when carrying out likelihood ratio tests and Monte Carlo adjusted profile confidence intervals (Ionides et al., 2017; Ning et al., 2021).

6. Each iteration of IBPF produces a log-likelihood estimate corresponding to the extended model with dynamically perturbed parameters. At the end of the search, the log-likelihood was re-evaluated using BFP,

with 10 replications at $J = 8000$ particles. A high level of effort on likelihood evaluation assists the task of building understanding about the likelihood surface from repeated Monte Carlo searches.

7. The exact record of all our computations is the source code for our numerical results, which is available at `https://github.com/ionides/ibpf_article`.

S3 Computational efficiency

Numerical error of Monte Carlo methods can be decomposed as bias and variance, and efficiency corresponds to how error scales with the amount of computational resources used. Expended resources can be quantified by objective metrics such as joules or dollars, but in practice we usually assess resources in terms of computational time on the machines that we personally have available.

Here, our main goals are evaluation and maximization of the log-likelihood. Variance in Monte Carlo likelihood estimates results in negative bias on the log-likelihood due to Jensen's inequality. Approximations involved in constructing a filter provide another source of bias for estimating the log-likelihood. This Monte Carlo approximation bias has negative expectation, over when the model is correct, since log-likelihood is a proper scoring rule (Gneiting and Raftery, 2007). Monte Carlo maximization error (meaning the difference between the unknown, exact log-likelihood at the Monte Carlo MLE compared to the unknown, exact MLE) can only be negative. Based on these considerations, we seek methods giving high average Monte Carlo log-likelihood at a Monte Carlo MLE, and our practical goal is to obtain reliably high log-likelihoods using calculations taking no longer than a day or so on one 36-core computing node.

S4 Varying the spatial autoregression parameter, r

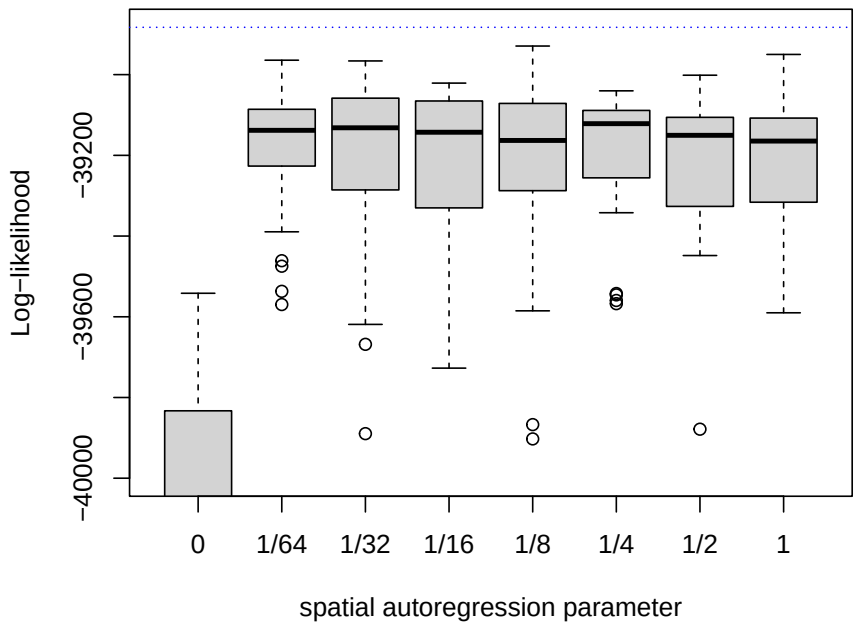

Figure S1: Varying the IBPF spatial autoregression algorithmic parameter for shared parameters with simulated measles data. The log-likelihood was obtained using $M = 100$ iterations of IBPF starting at random parameters (median starting log-likelihood, -275000). The horizontal dashed line denotes the log-likelihood at the true parameters.

References

Gneiting, T. and Raftery, A. E. (2007). Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378.

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.

Ning, N., Ionides, E. L., and Ritov, Y. (2021). Scalable Monte Carlo inference and rescaled local asymptotic normality. *Bernoulli*, 27:2532–2555.