Supplement to “A second-order iterated smoothing algorithm”

D. Nguyen\(^1\) and Edward L. Ionides\(^1\)

October 2, 2016

\(^1\) Department of Statistics
The University of Michigan, Ann Arbor, Michigan, USA.
email: nguyenxd@umich.edu, ionides@umich.edu

Supplementary Content

S1 Comparison of methods on the toy example \(\text{S-2}\)
S2 Algorithms IS1 and RIS1 \(\text{S-2}\)
S3 Assessment of the gradient and Hessian estimates \(\text{S-6}\)
S4 Parameters definitions and starting ranges for the malaria model \(\text{S-10}\)
S1  Comparison of methods on the toy example

We continue with the toy example in the main text, a bivariate, linear, Gaussian discrete-time process. Fig. S-1 shows the results of 40 Monte Carlo replications so that we can see the clustering of the MLE estimates around the true MLE, corresponding to Fig. 1 in the main text. The computations in Fig. S-1 match the setup in the main text. For IS2, most of the replications clustered near the true MLE while none of them stays in a lower likelihood region. Fig. 1, in the main text, can be viewed as a statistical summary of Fig. S-1, with 200 Monte Carlo replications. These results indicate that IS2 is clearly the best of the investigated methods for this test.

We also checked how the methods compared when given additional computational resources, setting \( M = 100 \) iterations and \( J = 10000 \) particles, with the random walk standard deviation decreasing geometrically from 0.23 down to 0.0207 for RIS1 and from 0.02 down to 0.0018 for other methods. In this situation, IS2 is better than both IF2 and RIS1, and IF1 performed substantially worse than the other methods (Fig. S-2). All four of these methods have comparable computational demands for given \( M \) and \( J \). IS1 requires substantially more computational resources, and we did not compute it for this comparison.

S2  Algorithms IS1 and RIS1

The pseudo-code in Algorithm S2 corresponds to the iterated smoothing algorithm of Doucet et al. (2013). The computational complexity of approach in Doucet et al. (2013) is \( O(LN) \), the algorithm is expected to be slow, especially when computing covariance of every pair of time points with distance smaller than \( L \). We also propose a variant of IS1 using a computationally convenient approximation to this covariance; we call this method reduced IS1 (RIS1). RIS1 is a modification of IS2 for which, at line 5 in Algorithm 1, we do not update \( \Theta_{t-1,n}^F \). For IS2, these covariance terms cancel in the theoretical analysis. However, there is no theorem to support RIS1 and it is only justified heuristically based on the observation that covariance between different time points may be small in practice. RIS1 is not presented for its theoretical interest, but for empirical interest in providing a computationally efficient benchmark for comparing between white noise and random walk noise.
Figure S-1: Comparison of different estimators. The likelihood surface for the linear, Gaussian model, with likelihood within 2 log units of the maximum shown in red, within 4 log units in orange, within 10 log units in yellow, and lower in light yellow. The location of the MLE is marked with a green cross. The black crosses show final points from 40 Monte Carlo replications of the estimators: (A) IF1 method; (B) IF2 method; (C) IS2 method; (D) RIS1 method. Each method, except RIS1, was started uniformly over the rectangle shown, with $M = 25$ iterations, $N = 1000$ particles, and a random walk standard deviation decreasing from 0.02 geometrically to 0.011 for both $\alpha_2$ and $\alpha_3$. We use bigger random walk standard deviations for RIS1. Specifically random walk standard deviations decrease from 0.23 geometrically to 0.125 for both $\alpha_2$ and $\alpha_3$. 

S-3
Algorithm S1: Iterating smoothing using white noise perturbations (IS1)

Input:
- starting parameter, $\theta_0$
- simulator for $f_n(x_n|x_{n-1}; \theta)$
- simulator for $g_n(y_n|x_n; \theta)$
- initial value parameters, $I \subset \{1, \ldots, d\}$
- number of iteration, $M$
- perturbation scales, $\sigma_{1:d}$; $\Psi = \text{diag}(\sigma_{1,d}^2)$
- data, $y_{1:N}$
- number of particles, $J$
- cooling rate, $0 < c < 1$, lag, $L$

Output:
- Monte Carlo maximum likelihood estimate, $\theta_M$

for $m$ in $1 : M$ do
  initialize parameters: $[\Theta_{0,j}^F]_i \sim N([\theta_{m-1}]_i, (c^{m-1}\sigma_i)^2)$ for $j$ in $1 : J$, $i$ in $1 : d$
  initialize states: simulate $X_{0,j}^F \sim \mu(x_0; \Theta_{0,j}^F)$ for $j$ in $1 : J$
  for $n$ in $1 : N$ do
    perturb at time $n$: $[\Theta_{n,j}^F]_i \sim N([\Theta_{n-1,j}^F]_i, (c^{n-1}\sigma_i)^2)$ for $i \notin I$, $j$ in $1 : J$
    simulate prediction particles: $X_{n,j}^P \sim f_n(x_n|X_{n-1,j}^F; \Theta_{n,j}^P)$ for $j$ in $1 : J$
    evaluate weights: $w(n,j) = g_n(y_n^n|X_{n,j}^F; \Theta_{n,j}^P)$ for $j$ in $1 : J$
    normalize weights: $\tilde{w}(n,j) = w(n,j)/\sum_{l=1}^{J} w(n,l)$
    apply re-sampling to select indices $k_{1:j}$ with $P\{k_u = j\} = \tilde{w}(n,j)$
    re-sample particles: $X_{n,j}^F = X_{n,k_j}$ and $\Theta_{n,j}^F = \Theta_{n,k_j}^F$ for $j$ in $1 : J$
    let $a_1(n,k_j) = j$, $a_{l+1}(n,j) = a_1(n-l, a_1(n,j))$ for $j$ in $1 : J$, $l$ in $0 : L - 1$
  end for
  for $n$ in $1 : N$ do
    smoothed mean: $\hat{\theta}_{n-L}^L = \sum_{j=1}^{J} \tilde{w}(n,j)\Theta_{n-L,a_L(n,j)}^P$ if $n > L$
    for $l$ in $n : \min(n + L, N)$ do
      Covariance: $C_{n-L,a_L(n,j)}^L = \sum_{j=1}^{J} \tilde{w}(n,j)(\Theta_{n-L,a_L(n,j)}^P - \hat{\theta}_{n-L}^L)(\Theta_{n-L,a_L(n,j)}^P - \hat{\theta}_{n-L}^L)^\dagger$ if $n > L$
    end for
  end for
  for $j$ in $0 : L$ do
    smoothed mean: $\hat{\theta}_{N+j-L}^L = \sum_{j=1}^{J} \tilde{w}(N,j)\Theta_{N+j-L,a_L(n,j)}^P$
    for $l$ in $N + j - L : N$ do
      Covariance: $C_{N+j-L,a_L(n,j)}^L = \sum_{j=1}^{J} \tilde{w}(N,j)(\Theta_{N+j-L,a_L(n,j)}^P - \hat{\theta}_{N+j-L}^L)(\Theta_{N+j-L,a_L(n,j)}^P - \hat{\theta}_{N+j-L}^L)^\dagger$
    end for
  end for
  update: $S_m = c^{-2(m-1)}\Psi^{-1}\sum_{n=1}^{N} \left[ (\hat{\theta}_{n}^L - \theta_{m-1}) \right]$
  $I_m = -c^{-4(m-1)}\Psi^{-1}\sum_{n=1}^{N} \left( C_{n,n}^m - c^{2(m-1)}\Psi + 2 \sum_{s=n+1}^{(s+L)\wedge N} C_{s,n}^m \right)\Psi^{-1}$
  update parameters: $\theta_m = \theta_{m-1} + I_m^{-1}S_m$
end for
Figure S-2: The distributions of likelihoods corresponding to Monte Carlo MLE approximations estimated by IF1, IF2, RIS1 and IS2 methods for toy model. The MLE is shown as a dashed vertical line (dark blue in electronic version). The optimizations were started from 200 randomly uniform initial values over a rectangle.
S3  Assessment of the gradient and Hessian estimates

Although the primary goal of our methodology is to maximize the likelihood, one can investigate the numerical accuracy of the gradient and Hessian estimates used to construct the maximization scheme. One can also investigate the relationship between accurate derivative estimates and successful optimization.

We computed the derivatives on a toy model for which exact derivatives are numerically tractable. We computed the bias squared, the variance and the mean square error of the gradient, Hessian and Newton-Raphson step (the gradient multiplied by the inverse of the Hessian) for the IS2 algorithm with different values of the lag, $L$. We used a simple linear Gaussian model defined by a scalar process $X_n = \theta X_{n-1} + \epsilon_n$ with $X_0 = 0$ and a scalar observation process $Y_n = X_n + \eta_n$ where both noises $\epsilon_n$ and $\eta_n$ are random variables with standard normal distribution. The true value for $\theta$ was 0.8. We generated a time series with length $N = 100$. For this model, we started from $\theta = 0.6$ and ran one iteration of IS2 with 5000 particles at varying values of the random walk perturbation intensity. We computed the gradient, Hessian and Newton-Raphson steps estimated by this iteration of IS2. The true values of these quantities were computed by numerical differentiation (using the numDeriv package in R) of the true log likelihood function computed by the Kalman filter. At $\theta = 0.6$, the true gradient was 1.071 and the Hessian was 236.137.

Figure S-3 shows that the gradient estimated by IS2 has increasing bias and decreasing variance for increasingly large values of the perturbation standard deviation, as expected. The bias decreases with lag and the variance increases, also as expected. Figure S-4 shows a similar pattern for the Hessian, also as expected. Figure S-5 may be surprising: larger perturbations seem to decrease the bias of the estimated step while also decreasing its variance. We infer that errors for larger perturbations were canceling out, and the maximization was therefore more successful than is suggested by our marginal analysis of the gradient and Hessian approximations. This demonstrates a limitation for our theoretical understanding, but apparently a happy one.
Figure S-3: Comparison of bias squared, variance and MSE of the gradient estimated by IS2 method for lag 1, 2, 3, 4, 5 at a point $\theta = 0.6$. For each value of perturbation, we perturb the starting value $\theta = 0.6$ of IS2 method and compute the gradient for IS2 with different lags. The true values of log likelihood are computed from Kalman filter while true value for gradient are computed from numDeriv package. The true gradient at $\theta = 0.6$ is 1.071 and the true Hessian is 236.137. The true parameter $\theta$ for the model is 0.8.
Figure S-4: Comparison of bias squared, variance and MSE of the Hessian estimated by IS2 method for lag 1, 2, 3, 4, 5 at a point $\theta = 0.6$. For each value of perturbation, we perturb the starting value $\theta = 0.6$ of IS2 method and compute the gradient for IS2 with different lags. The true values of log likelihood are computed from Kalman filter while true value for gradient are computed from numDeriv package. The true gradient at $\theta = 0.6$ is 1.071 and the true Hessian is 236.137. The true parameter $\theta$ for the model is 0.8.
Figure S-5: Comparison of bias squared, variance and MSE of the Newton-Raphson step size estimated by IS2 method for lag 1, 2, 3, 4, 5 at a point $\theta = 0.6$. For each value of perturbation, we perturb the starting value $\theta = 0.6$ of IS2 method and compute the gradient for IS2 with different lags. The true values of log likelihood are computed from Kalman filter while true value for gradient are computed from numDeriv package. The true gradient at $\theta = 0.6$ is 1.071 and the true Hessian is 236.137. The true parameter $\theta$ for the model is 0.8.
S4 Parameters definitions and starting ranges for the malaria model

Table S-4. Parameters for the malaria $SEIH^3Q$ model.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
<th>Units</th>
<th>$\theta_{low}$</th>
<th>$\theta_{high}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mu_{EI}$ (*)</td>
<td>$E \rightarrow I$ transition rate</td>
<td>yr$^{-1}$</td>
<td>24</td>
<td>24</td>
</tr>
<tr>
<td>$\mu_{IH}$</td>
<td>$I \rightarrow H$ transition rate</td>
<td>yr$^{-1}$</td>
<td>1.00</td>
<td>5.00</td>
</tr>
<tr>
<td>$\mu_{HI}$</td>
<td>$H \rightarrow I$ transition rate</td>
<td>yr$^{-1}$</td>
<td>1.00</td>
<td>5.00</td>
</tr>
<tr>
<td>$\mu_{IS}$</td>
<td>$I \rightarrow S$ transition rate</td>
<td>yr$^{-1}$</td>
<td>0.5</td>
<td>2.00</td>
</tr>
<tr>
<td>$\mu_{IQ}$</td>
<td>$I \rightarrow Q$ transition rate</td>
<td>yr$^{-1}$</td>
<td>1.00</td>
<td>2.00</td>
</tr>
<tr>
<td>$\mu_{QS}$</td>
<td>$Q \rightarrow S$ transition rate</td>
<td>yr$^{-1}$</td>
<td>10.00</td>
<td>20.00</td>
</tr>
<tr>
<td>$q$ (*)</td>
<td>relative infectivity of $Q$ class</td>
<td>—</td>
<td>0.001</td>
<td>0.001</td>
</tr>
<tr>
<td>$\tau$</td>
<td>mean lag for mosquitoes</td>
<td>month</td>
<td>0.10</td>
<td>0.50</td>
</tr>
<tr>
<td>$\rho$</td>
<td>case reporting fraction</td>
<td>—</td>
<td>0.001</td>
<td>0.01</td>
</tr>
<tr>
<td>$\sigma_{pro}$</td>
<td>s.d. of dynamic noise</td>
<td>yr$^{0.5}$</td>
<td>0.1</td>
<td>0.5</td>
</tr>
<tr>
<td>$\sigma_{obs}$</td>
<td>s.d. of measurement noise</td>
<td>—</td>
<td>0.1</td>
<td>0.5</td>
</tr>
<tr>
<td>$b_r$</td>
<td>coefficient of rainfall covariate</td>
<td>—</td>
<td>0.5</td>
<td>0.9</td>
</tr>
<tr>
<td>$S_0$</td>
<td>initial fraction in $S$ class</td>
<td>—</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$E_0$</td>
<td>initial fraction in $E$ class</td>
<td>—</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$I_0$</td>
<td>initial fraction in $I$ class</td>
<td>—</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$H_{i_0}$</td>
<td>initial fraction in $H_i$ class</td>
<td>—</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$Q_0$</td>
<td>initial fraction in $Q$ class</td>
<td>—</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$\kappa_0$</td>
<td>initial value, $\kappa(t_0)$</td>
<td>—</td>
<td>0.1</td>
<td>0.5</td>
</tr>
<tr>
<td>$\mu_{SE,0}$</td>
<td>initial value, $\mu_{SE}(t_0)$</td>
<td>—</td>
<td>0.1</td>
<td>0.5</td>
</tr>
<tr>
<td>$b_1$</td>
<td>1$^{st}$ spline coefficient</td>
<td>—</td>
<td>-5</td>
<td>5</td>
</tr>
<tr>
<td>$b_2$</td>
<td>2$^{nd}$ spline coefficient</td>
<td>—</td>
<td>-5</td>
<td>5</td>
</tr>
<tr>
<td>$b_3$</td>
<td>3$^{rd}$ spline coefficient</td>
<td>—</td>
<td>-5</td>
<td>5</td>
</tr>
<tr>
<td>$b_4$</td>
<td>4$^{th}$ spline coefficient</td>
<td>—</td>
<td>-5</td>
<td>5</td>
</tr>
<tr>
<td>$b_5$</td>
<td>5$^{th}$ spline coefficient</td>
<td>—</td>
<td>-5</td>
<td>5</td>
</tr>
<tr>
<td>$b_6$</td>
<td>6$^{th}$ spline coefficient</td>
<td>—</td>
<td>-5</td>
<td>5</td>
</tr>
<tr>
<td>$1/\delta$ (*)</td>
<td>mean human life span</td>
<td>yr</td>
<td>0.02</td>
<td>0.02</td>
</tr>
</tbody>
</table>

We follow definitions as in Roy et al. (2013). $\theta_{low}$ and $\theta_{high}$ are the lower and upper bounds for a hyper-rectangle used to generate starting points for the search. Parameters labeled with (*) were set at fixed values. Non-negative parameters were logarithmically transformed for optimization.
Supplementary References
