ENTROPY BASED ADAPTIVE PARTICLE FILTER

Silvia Liverani and Anastasia Papavasiliou

University of Warwick
Department of Statistics
Coventry, CV4 7AL

ABSTRACT

We propose a particle filter for the estimation of a partially observed Markov chain that has a non dynamic component. Such systems arise when we include unknown parameters or when we decompose non ergodic systems to their ergodic classes. Our main assumption is that the value of the non dynamic component determines the limiting distribution of the observation process. In such cases, we do not want to resample the particles that correspond to the non dynamic component of the Markov chain. Instead, we take a weighted average of particle filters corresponding to different values of the non dynamic component. The computation of the weights is based on entropy and the number of particles corresponding to each particle filter is proportional to the weights.

1. SETTING

We are interested in systems of the following form:

$$\begin{cases} X_{n+1} & \sim & K_{\theta_n}(X_n, \cdot) \\ \theta_{n+1} & = & \theta_n \end{cases} \tag{1}$$

that have been partially observed through

$$Y_n = h(X_n) + V_n, \tag{2}$$

where X_n and Y_n take values in Euclidean spaces $\mathbf{R}^p$ and $\mathbf{R}^q$ and θ_n takes values in a compact subset of a Euclidean space $\Theta \subseteq \mathbf{R}^r$. The Markov chain (X_n, θ_n) is clearly not ergodic, but if we fix the value of the non dynamic component to some $\alpha \in \Theta$, X_n becomes a Markov chain with transition kernel K_α which we assume to be mixing.

The motivation for looking at these type of systems comes from adaptive estimation: the non dynamic component θ_n corresponds to unknown parameters (see [1]). However, such systems can also arise by decomposing a non ergodic system to its ergodic classes, in which case each value of the non dynamic component corresponds to a different ergodic class.

Our goal is to estimate the random variable X_n, given all the information up to time n. Since the system could be nonlinear, we would like to apply particle filters. However, we want to avoid resampling the non-dynamic component of the Markov chain because this could lead to divergence of the particle filter.

Note that we are not interested in estimating the parameter, but rather in computing the marginal of the posterior distribution corresponding to the state variable X_n. Thus, we take a different approach to that of [2, 3], where the authors develop an algorithm for estimating the parameters and, more generally, computing the derivatives of the optimal filter. Our approach is also applicable to the case where the system is not ergodic and we do not know the initial conditions which, in this case, play the role of the parameter θ.

This is particularly true in the case when we don't know the exact initial conditions of the Markov chain. Suppose, for example, that the correct initial distribution of θ_0 is δ_α, i.e. θ is really a constant equal to α. Then, if we initialize θ_0 according to some prior u, the corresponding optimal filter will eventually converge to the optimal filter that has been correctly initialized according to δ_α, provided that u is a 'good' prior and certain identifiability conditions hold (see [4]). In this case, the particle filter needs to approximate the optimal filter corresponding to prior u uniformly in time so that it stays close to the true optimal filter.

In order to avoid resampling and construct a particle filter that will converge uniformly in time, we can take a weighted average of particle filters corresponding to different values of θ (treating θ as a constant), picked from Θ according to the prior distribution u: we define the particle filter

$$\tilde{\Phi}_n^{M,N}(\mu \otimes u) = \sum_{j=1}^{M} w_n(\theta_j) \Phi_n^N(\mu \otimes \delta_{\theta_j}), \tag{3}$$

where $\{\theta_j\}_{j=1}^M$ are independent samples from the prior distribution u, $\Phi_n^N(\mu \otimes \delta_{\theta_j})$ is the interacting particle filter where the non dynamic component has been fixed to θ_j and $w_n(\theta_j)$ is computed in such a way that it approximates the likelihood of θ being equal to θ_j:

$$w_n(\theta_j) \approx \mathbf{P}\left(\theta_j | Y_n, \ldots, Y_1\right). \tag{4}$$

In this computation of the weights, the particle filters $\Phi_n^N(\mu \otimes \delta_{\theta_j})$ are used in a crucial way: see [5] and [6], where the uniform convergence of this particle filter is shown. As a result,

1-4244-0581-5/06/$20.00 © 2006 IEEE.

even if the weight corresponding to a particular value of θ is very small, we still need a good approximation of the corresponding particle filter $\Phi_n^N(\mu \otimes \delta_\theta)$, which makes this algorithm computationally very expensive. However, if there was a way to compute the weights that did not involve the interacting particle filters, instead of using the same number of particles N to approximate the optimal filters corresponding to each value θ_j we could use a number proportional to $w_n(\theta_j)$, thus spending most of the computational effort on the particle filters with the higher weight.

In the following section, we suggest a way of computing the weights that depends on entropy and does not involve the interacting particle filters.

2. PARTICLE FILTER

Our main assumptions are that the values of θ are in one-to-one correspondence with the limiting distributions of the observation process ν_θ and that for each θ the observation process satisfies the large deviation principle. This means that if we had infinitely many observations, we would know the limiting distribution ν_θ and consequently we would know θ. So, the likelihood of θ_j should be approximately proportional to the distance of ν_{θ_j} from ν_α, where α is the true value θ in the sense that

$$\frac{1}{n}\sum_{k=1}^{n} \delta_{Y_k} \xrightarrow{w} \nu_\alpha, \text{ as } n \to \infty.$$

Based on the above observations, we make the following approximations:

$$\mathbf{P}\left(Y_n, \ldots, Y_1 | \theta\right) \approx \mathbf{P}\left(L_n(Y) \in \mathcal{B}(\nu_\alpha, \epsilon) | \theta\right),$$

where we set $L_n(Y) = \frac{1}{n}\sum_{k=1}^{n} \delta_{Y_k}$ and $\mathcal{B}(\nu_\alpha, \epsilon)$ is the ball of radius ϵ around the distribution ν_α with respect to the Levy-Prohorov metric that metrizes the weak convergence of measures. Then, since the observation process satisfies the large deviation principle, we can say that for large n

$$\mathbf{P}\left(L_n(Y) \in \mathcal{B}(\nu_\alpha, \epsilon) | \theta\right) \approx e^{-nI_\theta},$$

where I_θ is the appropriate rate function. If the observations were an i.i.d. sequence, the rate function would have been the entropy distance between ν_θ and ν_α. This is our next approximation:

$$I_\theta \approx \int_{\mathbf{R}^q} \log\left(\frac{d\nu_\alpha}{d\nu_\theta}(y)\right) \nu_\alpha(dy).$$

Since we do not know α and consequently ν_α, we replace it by $\frac{1}{n}\sum_{k=1}^{n} \delta_{Y_k}$ which converges to ν_α. Then, the right hand side becomes

$$J_n(\theta) := -\frac{1}{n}\sum_{k=1}^{n} \log\left(n\nu_\theta(Y_k)\right).$$

If an analytic expression for ν_θ is not available, we can also approximate is by $\frac{1}{n}\sum_{k=1}^{n} \delta_{Y_k^\theta}$, where $\{Y_k^\theta\}_{k=1}^{n}$ is a simulation of the observation process corresponding to parameter value θ.

Based on the above approximations, we define the weights to be

$$\begin{aligned} W_n(\theta_i) &= \frac{e^{\sum_{k=1}^{n} \log\left(n\nu_{\theta_i}(Y_k)\right)}}{\sum_{j=1}^{M} e^{\sum_{l=1}^{n} \log\left(n\nu_{\theta_j}(Y_l)\right)}} \qquad (5) \\ &= \frac{1}{\sum_{j=1}^{M} e^{\sum_{l=1}^{n}\left(\log\left(\frac{\nu_{\theta_j}(Y_l)}{\nu_{\theta_k}(Y_l)}\right)\right)}}. \end{aligned}$$

The second formula is better in practice because it avoids dealing with large numbers.

So, we have achieved to define the weights in a way that does not involve the particle filters.

Note that as n goes to infinity, we expect that all the mass will be concentrated on one $\theta \in \{\theta_1, \ldots, \theta_M\}$, which is the one that minimizes the entropy distance between ν_θ and ν_α. Then, as M goes to infinity, we expect all the mass to be concentrated on α, which is exactly what we want.

The algorithm takes the following form:

At time n=0 (Initialization):

We sample M particles $\{\theta_j\}_{j=1,\ldots,M}$ from distribution u and $N \cdot M$ particles $\{\xi_i^0(\theta_j)\}_{i=1,\ldots,N; j=1,\ldots,M}$ from distribution μ. The weights of the parameters and the number of particles corresponding to each interacting particle filter are set to be equal, i.e. $W_0(\theta_j) = \frac{1}{M}$ and $N_0^j = N$. Then,

$$\begin{aligned} \tilde{\Phi}_0^{M,N}(\mu \otimes u) &:= \sum_{j=1}^{M} W_0(\theta_j)\Phi_0^{N_0^j}(\mu \otimes \delta_{\theta_j}) \\ &= \frac{1}{NM}\sum_{i,j=1}^{N,M} \delta_{\xi_i^0(\theta_j)}, \end{aligned}$$

where $\Phi_0^{N_0^j}(\mu \otimes \delta_{\theta_j})$ is the interacting particle filter at time $n = 0$, corresponding to parameter value θ_j.

For $n \geq 0$ (Evolution):

1. We compute the new weights $W_n(\theta_j)$ according to (5) and we sample the N_n^j according to these weights, so that they sum up to NM.

   Alternatively, we can set $N_n^j = \lceil W_n(\theta_j) \cdot NM \rceil$ to simplify the algorithm, i.e. set the number of particles equal to the rounded-up corresponding proportion of the total number of particles. In this case, the total number to particles may vary but will never be higher than $(N+1)M$.

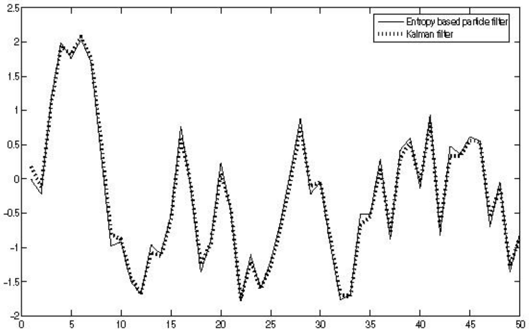

Fig. 1. Estimation of the state process X_n using the Kalman filter and the entropy based particle filter.

2. For each θ_j, we evolve the interacting particle filter as usual except that at the resampling stage, we start with N^j_{n-1} particles and we end with N^j_n particles.

Then, the particle filter is defined by

$$\tilde{\Phi}^{M,N}_n(\mu \otimes u) := \sum_{j=1}^{M} W_n(\theta_j)\Phi^{N^j_n}_n(\mu \otimes \delta_{\theta_j}). \qquad (6)$$

In the following section, we apply this algorithm to a toy example.

3. EXPERIMENT

Let us consider the model

$$\begin{aligned} X_n &= \theta X_{n-1} + \varepsilon_n \quad \theta \in (0,1) & (7)\\ Y_n &= X_n + \eta_n \qquad n = 1, 2, \dots & (8)\end{aligned}$$

where ε_n and η_n are standard Gaussian and uncorrelated random variables and the parameter θ is unknown. The values of Y_n are observed and we aim to estimate the values X_n. Note that for this model, the limiting distribution of the observation process is a Gaussian with mean zero and variance $\frac{1}{1-\theta^2}$, so it is in one-to-one correspondence with the parameter space $(0, 1)$ as required.

According to the algorithm introduced in the previous sections, we initially sampled $M = 100$ parameters from an uniform distribution over $(0, 1)$ and $N = 100$ particles for each parameter. The evolution of the algorithm follows as explained in the previous sections. The real value of the parameter is $\theta = 0.7$.

The graph in Fig.1 shows the estimation of the state process using the method presented here (solid line) compared to the Kalman filter estimation (dotted line) for 50 time points.

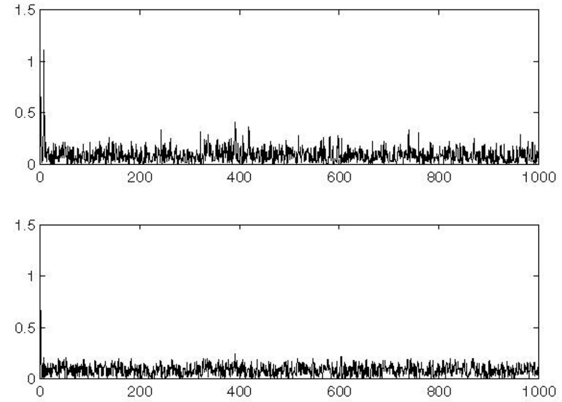

Fig. 2. L_1-distance from the Kalman filter when the parameter θ is unknown and known. When the parameter is unknown, the average error is 0.0912, whilst when the parameter is known, the average error is 0.0871.

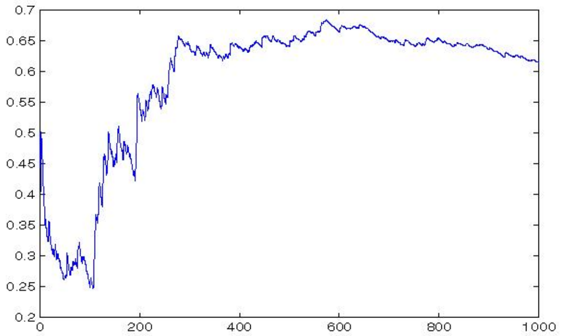

Fig. 3. Weighted mean of the weights $W_n(\theta_j)$ at each time point.

We now look at the L_1-error of the particle filter compared to the Kalman filter. Let us assume that we have 1000 observations of Y_n. The graphs in Fig.2 show the error for the entropy based particle filter and the interacting particle filter with known parameter θ. In the first case, we see that the size of the error decreases quickly to become comparable to that of the interacting particle filter for the correct parameter. The interacting particle filter has been computed using $N \cdot M = 10000$ particles.

Finally, the graph in Fig.3 shows the weighted mean of the weights $W_n(\theta_j)$ of the particle filter at each time point. As n increases they converge to the true value of the parameter, as expected. Note that the particle filter converges to the state process much earlier than the parameters converge to the true value of the parameter, due to the robustness of the optimal filter with respect to θ.

4. CONCLUSIONS

From the above example, we see that this method can give comparable errors to those of the particle filter with $N \cdot M$ parameters that has been initialized correctly, i.e. it corresponds to the correct value of the parameter.

If the goal is to compute the complete posterior distribution of (1) rather than the marginal, one can combine the method presented here with that in [6] as follows: the weights can be computed so as to approximate the posterior distribution of the parameter, as in [6], while the number of particles of the corresponding particle filters can be computed as above, so as not to depend on the previous estimation of the particle filters.

5. REFERENCES

[1] J. Liu and M. West, "Combined parameter and state estimation in simulation-based filtering," in *Sequential Monte Carlo in Practice*, N. de Freitas A. Doucet and N. Gordon, Eds. 2001, pp. 197–223, Springer.

[2] G. Poyadjis, A. Doucet, and S.S. Singh, "Particle methods for optimal filter derivative: Application to parameter estimation," in *Proceedings Intl. Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2005.

[3] M. Klaas, J.F.G. De Freitas, and A. Doucet, "Towards practical n^2 monte carlo: The marginal particle filter," in *Proceedings UAI*, 2005.

[4] A. Papavasiliou, "Parameter estimation and asymptotic stability in stochastic filtering," *Stochastic Processes and their Applications*, vol. 116, pp. 1048–1065, 2006.

[5] N. Gordon, S. Maskell, and T. Kirubarajan, "Efficient particle filters for joint tracking and classification," in *Proceedings SPIE Signal Data Process. Small Targets*. SPIE, 2002, vol. 4728, pp. 439–449.

[6] A. Papavasiliou, "A uniformly convergent adaptive particle filter," *Journal of Applied Probability*, vol. 42, pp. 1053–1068, 2005.