Colloque International de Statistique Appliquée pour le Développement en Afrique
International Conference on Applied Statistics for Development in Africa
Sada'07 nn, 1–6 (2007)

BAYESIAN NUMERICAL INFERENCE FOR HIDDEN MARKOV MODELS

Fabien Campillo, Rivo Rakotozafy and Vivien Rossi

Abstract. In many situations it is important to be able to propose N independent realizations of a given distribution law. We propose a strategy for making N parallel Monte Carlo Markov Chains (MCMC) interact in order to get an approximation of an independent N-sample of a given target law. In this method each individual chain proposes candidates for all other chains. We prove that the set of interacting chains is itself a MCMC method for the product of N target measures. Compared to independent parallel chains this method is more time consuming, but we show through examples that it possesses many advantages. This approach is applied to a biomass evolution model.

Keywords: Markov chain Monte Carlo method, interacting chains, hidden Markov model

§1. Introduction

Hidden Markov models are powerful modeling tools. They have been extensively developed since the 1970's in the context of discrete state spaces. In the case of general state space, also called *state-space modeling*, we need to utilize approximation procedures. The success of the Bayesian inference is mainly due to the development of efficient Monte Carlo approximation techniques [5]. Among them, MCMC methods allow us to sample from almost any prescribed distribution law [3]. Still high dimensional or "tricky" distribution laws are barely tackled by these techniques and should be approached with realistic expectations. Together with numerical Bayesian inference, hidden Markov models for general state-space (or state-space modeling) have been recently used in environment sciences and ecology, see e.g. [4]

In many models arising in environment (ecology, renewable resource management etc.), measurements $\mathsf{y}_1, \dots, \mathsf{y}_T$ are collected yearly or monthly so that the real-time constraint is not relevant even if the underlying law features a temporal structure. State-space modeling of these data consists in proposing a Markov process $(\mathsf{x}_t, \mathsf{y}_t)_{t=1\cdots T}$, where the state process x_t is not observed and y_t are the associated observation process. This process usually depends on some unknown parameter θ with given a priori law. The goal of the Bayesian inference is to determine the a posteriori law of $(\mathsf{x}_{1:T}, \theta)$ given the measurements $\mathsf{y}_{1:T}$.

MCMC algorithms [5] allow us to draw samples from a probability distribution $\pi(x)\,dx$ known up to a multiplicative constant. This consists in sequentially simulating a single Markov chain whose limit distribution is $\pi(x)\,dx$. There exist many techniques to speed up the convergence toward the target distribution by improving the mixing properties of the chain.

In practice one however can make use of several chains in parallel. It is then tempting to exchange information between these chains to improve mixing properties of the MCMC samplers. A general framework of "Population Monte Carlo" has been proposed in this context

[2]. In this paper we propose an interacting method between parallel chains which provides an independent sample from the target distribution. Contrary to papers previously cited, the proposal law in our work is given and does not adapt itself to the previous simulations. Hence, the problem of the choice of the proposal law still remains.

§2. Parallel/interacting Metropolis within Gibbs (MwG) algorithm

Let $\pi(x)$ be the probability density function of a target distribution defined on $(\mathbb{R}^n, \mathcal{B}(\mathbb{R}^n))$. For $\ell = 1, \dots, n$, we define the conditional laws:

$$\pi_\ell(x_\ell | x_{\neg\ell}) \stackrel{\text{def}}{=} \pi(x_{1:n}) / \textstyle\int \pi(x_{1:n})\, dx_{\neg\ell}\,. \tag{1}$$

where $\neg\ell \stackrel{\text{def}}{=} \{m = 1 : n; m \neq \ell\}$. When we know to sample from (1), we are able to use the Gibbs sampler. It is possible to adapt our interacting method to parallel Gibbs sampler. But very often we do not know how to sample from (1) and therefore we consider proposal conditional densities $\pi^{\text{prop}}_\ell(x_\ell)$ defined for all ℓ. In this case, we use MwG algorithm.

One iteration $X \to Z$ of the parallel/interacting MwG method consists in updating the components X_ℓ successively for $\ell = 1, \dots, n$, i.e. $[X_{1:n}] \to [Z_1 X_{2:n}] \to [Z_{1:2} X_{3:n}] \cdots [Z_{1:n-1} X_n] \to [Z_{1:n}]$. For each ℓ fixed, the subcomponents X^i_ℓ are updated sequentially for $i = 1 : N$ in two steps:

1. *Proposal step:* We sample independently N candidates $Y^j_\ell \in \mathbb{R}$ according to:

$$Y^j_\ell \sim \pi^{\ell,\text{prop}}_{i,j}(\xi | [\![Z, X^i_\ell, X]\!]^i_\ell)\, d\xi\,, \qquad 1 \le j \le n$$

where $[\![Z, \xi, X]\!]^i_\ell \stackrel{\text{def}}{=} \left[\begin{array}{c|c|c} & Z^1_\ell & \\ & \vdots & \\ & Z^{i-1}_\ell & \\ Z_{1:\ell-1} & \xi & X_{\ell+1:n} \\ & X^{i+1}_\ell & \\ & \vdots & \\ & X^N_\ell & \end{array}\right]$.

We also use the following lighter notation: $\pi^{\ell,\text{prop}}_{i,j}(\xi|\xi') = \pi^{\ell,\text{prop}}_{i,j}(\xi|[\![Z, \xi', X]\!]^i_\ell)$.

2. *Selection step:* The subcomponent X^i_ℓ could be replaced by one of the N candidates $Y^{1:N}_\ell$ or stay unchanged according to a multinomial sampling, the resulting value is called Z^i_ℓ, i.e.:

$$Z^i_\ell \leftarrow \begin{cases} Y^1_\ell & \text{with probability } \frac{1}{N}\, \alpha^{i,1}_\ell(X^i_\ell, Y^1_\ell)\,, \\ \vdots & \\ Y^N_\ell & \text{with probability } \frac{1}{N}\, \alpha^{i,N}_\ell(X^i_\ell, Y^N_\ell)\,, \\ X^i_\ell & \text{with probability } \tilde\rho^i_\ell(X^i_\ell, Y^{1:N}_\ell) \end{cases}$$

where:

$$\alpha_\ell^{i,j}(\xi,\xi') \overset{\text{def}}{=} \tfrac{\pi_\ell(\xi'|X^i_{\neg\ell})}{\pi_\ell(\xi|X^i_{\neg\ell})} \tfrac{\pi_{i,j}^{\ell,\text{prop}}(\xi|\xi')}{\pi_{i,j}^{\ell,\text{prop}}(\xi'|\xi)} \wedge 1\,, \quad \tilde\rho^i_\ell(X^i_\ell, Y^{1:N}_\ell) \overset{\text{def}}{=} 1 - \tfrac{1}{N}\textstyle\sum_{j=1}^N \alpha_\ell^{i,j}(X^i_\ell, Y^j_\ell)\,.$$

The proofs of the following results are technical, so they are not presented here. They are detailed in [1].

Lemma 1. *The Markov kernel on $\mathbb{R}^{n\times N}$ associated with the MwG algorithm is*

$$P(X,dZ) \overset{\text{def}}{=} P_1(X_{1:n};dZ_1)\; P_2(Z_1,X_{2:n};dZ_2)\cdots P_n(Z_{1:n-1},X_n;dZ_n)\,. \tag{2}$$

At iteration ℓ, the kernel $P_\ell(Z_{1:\ell-1},X_{\ell:n};dZ_\ell)$ generates $Z^{1:N}_\ell$ from the already updated components $Z^{1:N}_{1:\ell-1}$ and the remaining components $X^{1:N}_{\ell:n}$. Each component $Z^i_{1:\ell}$, for $i = 1\cdots N$, is updated independently one from each other:

$$P_\ell(Z_{1:\ell-1},X_{\ell:n};dZ_\ell) \overset{\text{def}}{=} \textstyle\prod_{i=1}^N P^i_\ell([\![Z,X^i_\ell,X]\!]^i_\ell;dZ^i_\ell)\,. \tag{3}$$

Here Z^i_ℓ is generated from $[\![Z,X^i_\ell,X]\!]^i_\ell$ according to:

$$P^i_\ell([\![Z,\xi,X]\!]^i_\ell;d\xi') \overset{\text{def}}{=} \tfrac{1}{N}\textstyle\sum_{j=1}^N \alpha_\ell^{i,j}(\xi,\xi')\;\pi_{i,j}^{\ell,\text{prop}}(\xi'|\xi)\;d\xi' + \rho^i_\ell(\xi)\;\delta_\xi(d\xi') \tag{4}$$

with $\alpha_\ell^{i,j}(\xi,\xi') \overset{\text{def}}{=} r_\ell^{i,j}(\xi,\xi')\wedge 1$ if $(\xi,\xi')\in R_\ell^{i,j}$, 0 otherwise, and

$$r_\ell^{i,j}(\xi,\xi') \overset{\text{def}}{=} \tfrac{\pi_\ell(\xi'|Z^i_{1:\ell-1},X^i_{\ell+1:n})}{\pi_\ell(\xi|Z^i_{1:\ell-1},X^i_{\ell+1:n})}\,\tfrac{\pi_{i,j}^{\ell,\text{prop}}(\xi|\xi')}{\pi_{i,j}^{\ell,\text{prop}}(\xi'|\xi)}\,,$$

$$\rho^i_\ell(\xi) \overset{\text{def}}{=} 1 - \tfrac{1}{N}\textstyle\sum_{j=1}^N \int_{\mathbb{R}} \alpha_\ell^{i,j}(\xi,\xi')\;\pi_{i,j}^{\ell,\text{prop}}(\xi'|\xi)\;d\xi'\,.$$

Finally, $R_\ell^{i,j}$ is the set of ordered pairs $(\xi,\xi')\in\mathbb{R}^2$ such that

$$\pi_\ell(\xi'|Z^i_{1:\ell-1},X^i_{\ell+1:n})\;\pi_{i,j}^{\ell,\text{prop}}(\xi|\xi') > 0 \quad \textit{and} \quad \pi_\ell(\xi|Z^i_{1:\ell-1},X^i_{\ell+1:n})\;\pi_{i,j}^{\ell,\text{prop}}(\xi'|\xi) > 0\,.$$

Proposition 2. *The measure $\Pi(dX) = \pi(X^1)\,dX^1\cdots\pi(X^N)\,dX^N$ is invariant for the kernel P, that is $\Pi P = \Pi$ i.e.:*

$$\textstyle\int_X P(X,dZ)\,\Big\{\prod_{i=1}^N \pi(X^i)\,dX^i\Big\} = \prod_{i=1}^N \pi(Z^i)\,dZ^i\,.$$

§3. Numerical tests

An hidden Markov model. We apply the parallel/interacting MwG sampler to a toy problem where a good estimate $\hat\pi$ of the target distribution π is available. Consider

$$\mathsf{s}_{\ell+1} = \mathbf{a}\,\mathsf{s}_\ell + \mathsf{w}_\ell\,, \qquad\qquad \mathsf{y}_\ell = \mathbf{b}\,\mathsf{s}_\ell + \mathsf{v}_\ell$$

for $\ell = 1\cdots n$, where $\mathsf{s}_1 \sim \mathcal{N}(\bar{\mathsf{s}}_1, Q_1)$, $\mathsf{w}_{1:n}$ and $\mathsf{v}_{1:n}$ are centered white Gaussian noises with variances σ^2_{w} and σ^2_{v}. Suppose that $\mathbf{b}$ is known and $\mathbf{a} = \theta$ is unknown with a priori law $\mathcal{N}(\mu_\theta,\sigma^2_\theta)$. We also suppose that $\mathsf{w}_{1:n}$, $\mathsf{v}_{1:n}$, s_1 and θ are mutually independent.

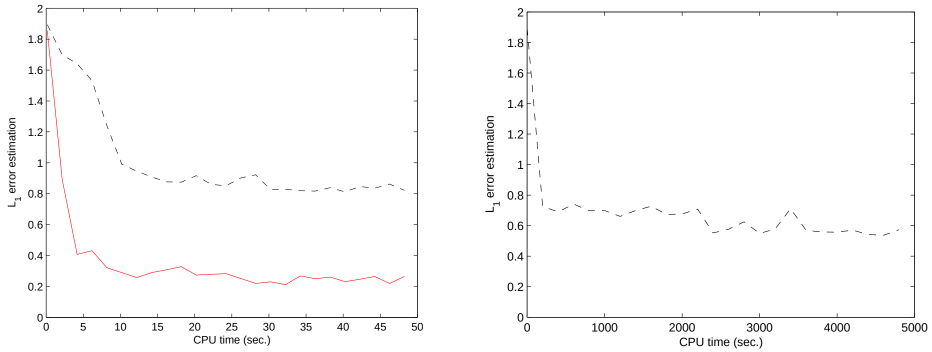

Figure 1: **Left**: Evolution of the indicator ϵ^k, see (5), for the parallel/independent MwG sampler (- -), and for the parallel/interacting MH sampler (–). This evolution is depicted as a function of the CPU time and not as a function of the iteration number k. The residual error of about 0.22 for the second method is due to the limited size of the sample. **Right**: Evolution of the indicator ϵ^k, see (5), for the parallel/independent MwG sampler (- -). After 5000 sec. CPU time, the convergence of this method is still unsatisfactory.

The state variable is $\mathsf{x}_{1:n+1} \overset{\text{def}}{=} (\mathsf{s}_{1:n}, \theta)$ and the target law is $\pi(s_{1:n}, \vartheta)\, ds_{1:n}\, d\vartheta \overset{\text{def}}{=} \text{law}(\mathsf{s}_{1:n}, \theta | \mathsf{y}_{1:n} = y_{1:n})$. This law is not Gaussian, but we can perform a Gibbs sampler:

$$\pi_{\mathsf{s}_\ell}(s_\ell | s_{\neg\ell}, \vartheta)\, ds_\ell \overset{\text{def}}{=} \text{law}(\mathsf{s}_\ell | \mathsf{s}_{\neg\ell} = s_{\neg\ell}, \theta = \vartheta, \mathsf{y}_{1:n} = y_{1:n}) = \mathcal{N}(\mathrm{m}_\ell, \mathrm{r}^2)\,,$$

$$\pi_\theta(\vartheta | s_{1:n})\, d\vartheta \overset{\text{def}}{=} \text{law}(\theta | \mathsf{s}_{1:n} = s_{1:n}, \mathsf{y}_{1:n} = y_{1:n}) = \mathcal{N}(\tilde{\mathrm{m}}, \tilde{\mathrm{r}}^2)$$

where r^2, m_ℓ, $\tilde{\mathrm{r}}^2$ and $\tilde{\mathrm{m}}$ are known, see [1]. We will perform three algorithms: (*i*) N parallel/interacting MwG samplers, (*ii*) N parallel/independent MwG samplers, (*iii*) N_{Gibbs} parallel/independent Gibbs samplers. Our aim is to show that making parallel samplers interact could speed up the convergence toward the stationary distribution. Because of its good convergence property, method (*iii*) is considered as a reference method. Here we perform $k = 10000$ iterations of $N_{\text{Gibbs}} = 5000$ independent Gibbs samplers. We obtain a kernel density estimate $\hat{\pi}$ of the target density based on the $N_{\text{Gibbs}} = 5000$ final values. Let $\hat{\pi}_{\mathsf{x}_\ell}$ be the corresponding ℓ-th marginal density. For methods (*i*) and (*ii*) we perform $N = 50$ parallel samplers. Let $\pi^{\text{int},k}$ and $\pi^{\text{ind},k}$ be the kernel density estimates of the target density based on the final values of methods (*i*) and (*ii*) respectively. Let $\pi^{\text{int},k}_{\mathsf{x}_\ell}$ and $\pi^{\text{ind},k}_{\mathsf{x}_\ell}$ be the corresponding ℓ-th marginal densities.

For each algorithm (*i*) and (*ii*), that is for $\pi^k_{\mathsf{x}_\ell} = \pi^{\text{ind},k}_{\mathsf{x}_\ell}$ and $\pi^{\text{int},k}_{\mathsf{x}_\ell}$, we compute

$$\epsilon^k = \tfrac{1}{n+1} \textstyle\sum_{\ell=1}^{n+1} \epsilon^k_\ell \quad \text{with} \quad \epsilon^k_\ell \overset{\text{def}}{=} \int |\pi^k_{\mathsf{x}_\ell}(\xi) - \hat{\pi}_{\mathsf{x}_\ell}(\xi)|\, d\xi\,, \quad \ell = 1 \cdots n+1\,. \qquad (5)$$

Hence ϵ^k is a criteria of the error between the target probability distribution and its estimation provided by the algorithm used.

These estimations are based on a sample of size $N = 50$ only, so they suffer from variability. This is not problematical, indeed we do not want to estimate L^1 errors but to diagnose

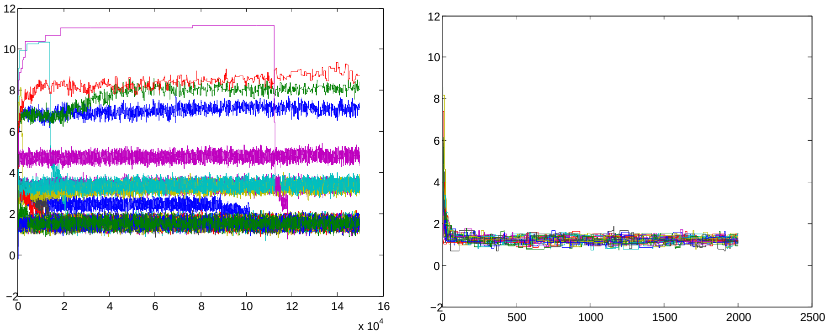

Figure 2: Evolution of the estimation of the parameter r versus the MCMC iterations: $N = 30$ parallel samplers without interaction (left) and 30 parallel samplers with interaction (right). Interactions clearly improve the convergence behavior.

the convergence toward the stationary distribution. So we use ϵ^k_ℓ as an indicator which must decrease and remain close to a small value when convergence occurs.

To compare fairly the parallel/independent MwG algorithm and the parallel/interacted MwG algorithm, we represent on Figure 1 the indicator ϵ^k for each algorithm not as a function of k but as a function of the CPU time. In Figure 1 (left) we see that even if one iteration of algorithm (i) needs more CPU than one of (ii), still the first algorithm converges more rapidly than the second one. This shows the inefficiency of parallel/independent MwG on this simple model.

Ricker model. We consider the Ricker discrete-time stock-recruitment model perturbated by a noise:

$$\mathsf{x}_{t+1} = \mathsf{x}_t \, e^{r - b\,\mathsf{x}_t} \, e^{w_t} \,.$$

where r is the growth parameter and w_t is a white Gaussian noise $\mathcal{N}(0, \sigma_w^2)$. We suppose that measurements satisfy:

$$\mathsf{y}_t = h\,\mathsf{x}_t + v_t$$

where v_t is a white Gaussian noise $\mathcal{N}(0, \sigma_v^2)$. For notational convenience we assume that $h = 1$. Suppose that only r is unknow so that the target law is $\mathrm{law}(\mathsf{x}_{1:T}, \theta | \mathsf{y}_{1:T})$.

We ran two parallel MwG samplers with and without interaction. Figure 2 shows that interaction deeply improve the behavior of the algorithm.

§4. Conclusion

This work showed that making parallel MCMC chains interact could improve their convergence properties. We presented the basic properties of the MCMC method, we did not prove that the proposed strategy speeds up the convergence. This difficult point is related to the problem of the rate of the convergence of the MCMC algorithms. Through simple examples we saw that the MwG strategy could be a poor strategy. In this situation our strategy improved the convergence properties.

Acknowledgements

This work was partially done during a stay of the second and third authors in INRIA/IRISA, Rennes. This work is partially supported by the SARIMA project of the French ministry of Foreign Affairs.

References

[1] F. Campillo and V. Rossi. Parallel interacting MCMC's for a class of hidden Markov models. Technical report, INRIA, 2006. `http://hal.inria.fr/inria-00103871`.

[2] O. Cappé, A. Guillin, J.-M. Marin, and C.P. Robert. Population Monte Carlo. *Journal of Computational and Graphical Statistics*, 13(4):907–929, 2004.

[3] W.R. Gilks, S. Richardson, and D.J. Spiegelhalter, editors. *Markov Chain Monte Carlo in practice*. Chapman & Hall, London, 1995.

[4] E. Rivot, E. Prevost, E. Parent, and J.-L. Blaginière. A Bayesian state-space modelling framework for fitting a salmon stage-structured population dynamic model to multiple time series of field data. *Ecological Modelling*, 179:463–485, 2004.

[5] C.P. Robert and G. Casella. *Monte Carlo Statistical Methods*. Springer–Verlag, Berlin, 2nd edition, 2004.

Fabien Campillo
INRIA/IRISA, Campus de Beaulieu
35042 Rennes Cedex
France
`Fabien.Campillo@inria.fr`

Rivo Rakotozafy
University of Fianarantsoa
BP 1264
Andrainjato
301 Fianarantsoa
Madagascar
`rrakotozafy@uni-fianar.mg`

Vivien Rossi
IURC, University of Montpellier I
641 avenue du Doyen Gaston Giraud
34093 Montpellier cedex 5
France
`Viven.Rossi@iurc.montp.inserm.fr`