Méthodes MCMC en interaction pour l'évaluation de ressources naturelles

Fabien Campillo[1] — Philippe Cantet[2] — Rivo Rakotozafy[3] — Vivien Rossi[4]

[1] Projet MERE, INRIA/INRA
UMR ASB – Bât. 29 – 2 place Viala – 34060 Montpellier cedex 06, France
Fabien.Campillo@inria.fr

[2] Cemagref
Equipe Hydrologie - Division OHAX, CS 40061, 13182 Aix en Provence cedex 5, France
philippe.cantet@cemagref.fr

[3] University of Fianarantsoa
BP 1264, Andrainjato, 301 Fianarantsoa, Madagascar
rrakotozafy@uni-fianar.mg

[4] CIRAD
Campus International de Baillarguet, 34398 Montpellier cedex 5, France
vivien.rossi@cirad.fr

RÉSUMÉ. Les méthodes de Monte Carlo par chaînes de Markov (MCMC) couplées à des modèles de Markov cachés sont utilisées dans de nombreux domaines, notamment en environnement et en écologie. Sur des exemples simples, nous montrons que la vitesse de convergence de ces méthodes peut être très faible. Nous proposons de mettre en interaction plusieurs algorithmes MCMC pour accélérer cette convergence. Nous appliquons ces méthodes à un modèle d'évolution de la biomasse d'une pêcherie.

ABSTRACT. Markov chain Monte Carlo (MCMC) methods together with hidden Markov models are extensively used in the Bayesian inference for many scientific fields like environment and ecology. Through simulated examples we show that the speed of convergence of these methods can be very low. In order to improve the convergence properties, we propose a method to make parallel chains interact. We apply this method to a biomass evolution model for fisheries.

MOTS-CLÉS : Inférence bayésienne, Monte Carlo par chaînes de Markov

KEYWORDS : Bayesian inference, Markov chain Monte Carlo

1. Introduction

Depuis une quinzaine d'années on assiste à une forte progression de l'analyse bayésienne dans de nombreux domaines parmi lesquels la médecine [2], la phylogénétique [1], la génétique [3], la bioinformatique [16], l'économétrie [22], l'écologie [40, 35, 18, 14, 19], l'halieutique [33] etc. On pourra à ce sujet consulter les articles généraux [30, 4]. Cet engouement est essentiellement dû à l'essor de techniques de Monte Carlo adaptées à ce contexte [8].

Ces techniques se répartissent en deux catégories : les *méthodes de Monte Carlo par chaînes de Markov* (MCMC pour "Markov Chain Monte Carlo") et les *méthodes de Monte Carlo séquentielles* (SMC pour "Sequential Monte Carlo"). Concernant les méthodes MCMC, le développement de boîtes à outils telles que BUGS et WinBUGS participe largement à ce succès. Les problèmes statiques (non-temporels) sont généralement traités par des méthodes MCMC et les problèmes dynamiques (temporels) par des méthodes SMC.

Les sciences environnementales et l'écologie font souvent appel à des systèmes dynamiques où la fréquence des observations est faible et où les séries temporelles sont courtes, de l'ordre d'une observation par an sur quelques dizaines d'années. Ces systèmes apparaissent le plus souvent en modélisation d'évolution de ressources naturelles (forêts, pêcheries). Ils peuvent être traités par des méthodes SMC mais aussi par des méthodes MCMC [12, 13, 24]. En effet, dans des modèles temporels où la fréquence des observations est de l'ordre de l'année, il n'existe pas de contrainte "temps réel" et il possible de faire appel à des techniques séquentielles comme à des techniques non–séquentielles ("batch"). Cette approche a par exemple été exploitée en halieutique [32, 36].

Dans les domaines où la contrainte de temps réel est plus forte (robotique, pistage de cibles, traitement d'images, de la parole etc.) les méthodes MCMC présentées dans cet article ne sont pas exploitables. En plus des modèles d'évolution de ressources naturelles, ces techniques sont également utilisables dans certains problèmes de sciences sociales ou économiques.

Considérons un système dynamique sous la forme d'un modèle de Markov caché (HMM pour "Hidden Markov Model") : $(X_t, Y_t)_{1\leq t\leq T}$ dont les noyaux de transition et d'émission dépendent d'un paramètre inconnu Θ. X_t et Y_t sont à valeurs dans $\mathbb{R}^n$ et $\mathbb{R}^d$, Θ est à valeurs dans $\mathbb{R}^p$.

Dans le cadre bayésien, le paramètre Θ est considéré comme une variable aléatoire et il s'agit de déterminer la loi *a posteriori* :

$$\Pi \stackrel{\text{déf}}{=} \text{loi}(X_{1:T}, \Theta | Y_{1:T} = y_{1:T})$$

où on note :

$$X_{1:T} \stackrel{\text{déf}}{=} (X_1, \ldots, X_T)\,.$$

La taille T de l'échantillon est fixée et relativement faible (quelques dizaines). La loi Π est portée par l'espace produit $E = [\mathbb{R}^n]^T \times \mathbb{R}^p$ et on note :

$$Z \stackrel{\text{déf}}{=} (X_{1:T}, \Theta)\,.$$

Dans la suite, afin de simplifier l'exposé, nous supposerons que la loi Π admet une densité π par rapport à la mesure de Lebesgue :

$$\begin{aligned}\pi(z)\,dz &= \pi(x_{1:T}, \theta)\; dx_1 \cdots dx_T\, d\theta \\ &= \mathbb{P}(X_{1:T} \in dx_{1:T}, \Theta \in d\theta | Y_{1:T} = y_{1:T})\,. \end{aligned} \tag{1}$$

En général, il n'est pas possible de calculer des statistiques de la densité π, comme sa moyenne ou sa covariance, même lorsqu'elle admet une expression analytique. Il est alors nécessaire de faire appel aux méthodes MCMC [41, 25] qui, dans ce cadre, sont quasiment les seules à pouvoir fournir des approximations de π. Ces méthodes consistent à construire une chaîne de Markov :

$$(Z^{(k)})_{k\in\mathbb{N}} = (X_{1:T}^{(k)}, \Theta^{(k)})_{k\in\mathbb{N}}$$

à valeurs dans l'espace produit E dont la densité (de la loi) invariante est π. L'approximation empirique de cette densité est obtenue par la simulation d'une réalisation d'une trajectoire $k \mapsto Z^{(k)}(\omega)$ de cette chaîne. Ces algorithmes sont également appelés *échantillonneurs* dans la mesure où ils simulent la densité π, dite cible.

La convergence en variation totale de la loi de cette chaîne vers π est garantie lorsque la chaîne est ϕ–irréductible et apériodique [39] (il y a alors convergence pour preque toute condition initiale). Ces méthodes sont applicables dans de nombreuses situations et sont très simples à mettre en œuvre. En pratique, il est connu qu'elles présentent des vitesses de convergence parfois trop faibles. Cette question est centrale dans le domaine et, malgré les efforts portés sur les développements théoriques, elle n'a encore pas trouvé de réponse pratique satisfaisante.

Dans cet article, nous n'abordons pas frontalement cette question. Nous adoptons plutôt un point de vue pragmatique : nous montrons sur des exemples simples que la convergence de ces méthodes peut être très lente. Nous proposons alors une stratégie consistant à faire interagir N chaînes $Z_1^{(k)}, \ldots, Z_N^{(k)}$ simulées en parallèle afin d'améliorer les propriétés de convergence de l'ensemble.

L'article est organisé comme suit : nous décrivons les HMM et leurs liens avec les modèles bayésiens hiérarchiques. Nous rappelons ensuite le principe de l'échantillonneur de Metropolis–Hastings. Nous précisons alors pourquoi la variante dite hybride de cet échantillonneur est adaptée aux HMM. Puis nous présentons l'algorithme Metropolis–Hastings hybride en interaction. Enfin, pour quantifier l'apport de cette approche, nous effectuons des comparaisons sur deux exemples.

2. Modèles de Markov cachés

Considérons un modèle de Markov caché $(X_t, Y_t)_{1\leq t\leq T}$ à valeurs dans $\mathbb{R}^n \times \mathbb{R}^d$ et dépendant d'un paramètre Θ inconnu. Le processus d'état non observé X_t est une chaîne de Markov homogène de densité initiale :

$$\mu_\theta(x)\, dx \stackrel{\text{déf}}{=} \mathbb{P}(X_1 \in dx|\Theta = \theta)\,, \tag{2}$$

et de noyau de transition :

$$q_\theta(x, x')\, dx' \stackrel{\text{déf}}{=} \mathbb{P}(X_t \in dx'|\Theta = \theta, X_{t-1} = x) \tag{3}$$

(afin de simplifier l'exposé, nous supposons que le noyau de transition admet une densité par rapport à la mesure de Lebesgue).

Le processus d'observation Y_t est une suite de variables, indépendantes conditionnellement à l'état X_t et au paramètre Θ, et la loi conditionnelle de Y_t ne dépend que de X_t et Θ, plus précisément :

$$\mathbb{P}(Y_{1:T} \in dy_{1:T}|\Theta = \theta, X_{1:T} = x_{1:T}) = \prod_{t=1}^{T} \mathbb{P}(Y_t \in dy_t|\Theta = \theta, X_t = x_t)\,.$$

Supposons de plus que la loi conditionnelle de Y_t sachant X_t est absolument continue par rapport à la mesure de Lebesgue et posons :

$$\psi_\theta(x, y)\, dy \stackrel{\text{déf}}{=} \mathbb{P}(Y_t \in dy|\Theta = \theta, X_t = x)\,. \tag{4}$$

Les éléments du modèle dépendent d'un paramètre inconnu Θ à valeurs dans $\mathbb{R}^p$. Ce paramètre est supposé indépendant de tous les autres termes et de densité *a priori* :

$$\nu(\theta)\, d\theta \stackrel{\text{déf}}{=} \mathbb{P}(\Theta \in d\theta)\,. \tag{5}$$

L'inférence de ce type de modèles consiste à estimer les états cachés $X_{1:T}$ du système ainsi que le paramètre inconnu Θ à l'aide des observations $Y_{1:T}$.

Ces modèles sont en fait une extension au cas dynamique des modèles dits "bayésiens hiérarchiques" [11], également appelés "réseaux bayésiens" dans d'autres contextes. Ces modèles admettent une représentation graphique (cf. Fig. 1) qui met en évidence les dépendances locales et hiérarchisées des différentes variables. Comme nous le verrons, ils sont adaptées à l'inférence bayésienne numérique [7].

En utilisant le fait que, conditionnellement à Θ, X_t est une chaîne de Markov et les hypothèses précédentes, on vérifie aisément que la loi du système s'écrit simplement en

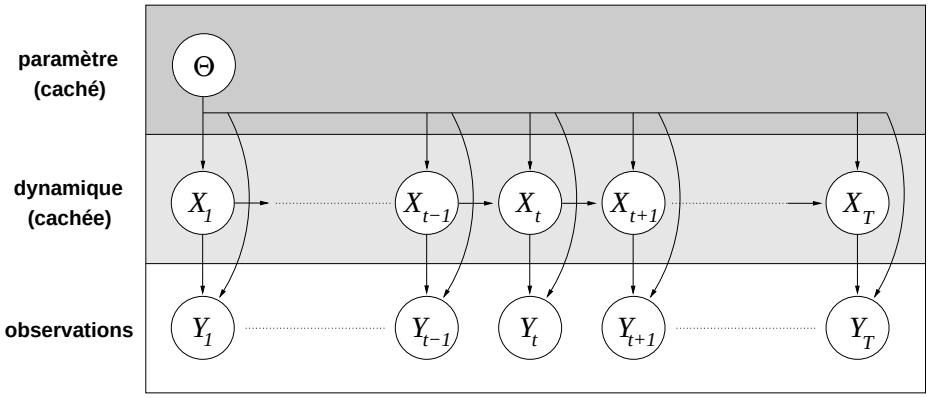

Figure 1. ***Modèle de Markov caché :*** *les flèches décrivent les liens de dépendance directe entre les variables. On peut par exemple remarquer que, conditionnellement aux états $X_{1:T}$ et au paramètre Θ, les observations $Y_1, \ldots, Y_T$ sont mutuellement indépendantes. Ce schéma fait apparaître la structure hiérarchisée : observations/processus/paramètre (cf. Équations (6) et (7)). Cette structure bayésienne hiérarchique se double d'une structure temporelle qui est exploitée dans la version hybride de l'algorithme de Metropolis–Hastings (cf. Section 3.3).*

fonction de la loi initiale (2) de X_t, de son noyau de transition (3), du noyau d'émission (4) de Y_t et de la densité a priori (5) de θ :

$$
\begin{aligned}
&\mathbb{P}(X_{1:T} \in dx_{1:T}, \Theta \in d\theta, Y_{1:T} \in dy_{1:T}) = \\
&\quad = \mathbb{P}(Y_{1:T} \in y_{1:T} | X_{1:T} = x_{1:T}, \Theta = \theta)\ \mathbb{P}(X_{1:T} \in dx_{1:T} | \Theta = \theta)\ \mathbb{P}(\Theta \in d\theta) \\
&\quad = \big[\prod_{t=1}^{T} P(Y_t \in dy_t | X_t = x_t, \Theta = \theta) \big] \times \\
&\qquad\quad \times \big[\prod_{t=2}^{T} \mathbb{P}(X_t \in dx_t | X_{t-1} = x_{t-1}, \Theta = \theta) \big]\ \mathbb{P}(X_1 \in dx_1 | \Theta = \theta)\ \mathbb{P}(\Theta \in d\theta) \\
&\quad = \big[\prod_{t=1}^{T} \psi_\theta(x_t, y_t)\, dy_t \big] \times \big[\prod_{t=2}^{T} q_\theta(x_{t-1}, x_t)\, dx_t \big] \times\ \mu_\theta(x_1)\, dx_1 \times \nu(\theta)\, d\theta\,.
\end{aligned}
$$

Ainsi :

$$
\pi(x_{1:T}, \theta) = \frac{\big[\prod_{t=1}^{T} \psi_\theta(x_t, y_t) \big]\ \big[\prod_{t=2}^{T} q_\theta(x_{t-1}, x_t) \big]\ \mu_\theta(x_1)\ \nu(\theta)}{\int_E \big[\prod_{t=1}^{T} \psi_\theta(x'_t, y_t) \big]\ \big[\prod_{t=2}^{T} q_\theta(x'_{t-1}, x'_t) \big]\ \mu_\theta(x'_1)\ \nu(\theta')\ dx'_{1:T}\, d\theta'}\,.
$$

c'est à dire

$$\pi(x_{1:T},\theta) \propto \big[\prod_{t=1}^{T}\psi_\theta(x_t,y_t)\big]\ \big[\prod_{t=2}^{T}q_\theta(x_{t-1},x_t)\big]\ \mu_\theta(x_1)\ \nu(\theta) \tag{6}$$

(i.e. à $y_{1:T}$ fixés, π est proportionnel au terme de droite). Ainsi, à une constante multiplicative près, π admet une formulation explicite, mais le calcul des statistiques de π, ainsi que le calcul de la constante de normalisation, ne peut en général pas se faire explicitement et il est nécessaire de faire appel à des méthodes d'approximation.

L'équation (6) est une application de la formule de Bayes qui peut se représenter dans ce cas comme :

$$\text{loi}(\textit{processus, paramètre} \mid \textit{observations}) \propto \text{loi}(\textit{observations} \mid \textit{processus, paramètre}) \times \text{loi}(\textit{processus} \mid \textit{paramètre}) \times \text{loi}(\textit{paramètre}) \tag{7}$$

Les modèles de Markov cachés à espaces d'état finis sont apparus à la fin des années 60 pour traiter des problèmes de reconnaissance automatique de la parole [34]. Ils ont depuis suscité un fort regain d'activité en bio-informatique. Les modèles de Markov cachés à espaces d'état généraux trouvent des applications dans de nombreux domaines et font leur apparition en écologie et environnement depuis une dizaine d'années. On peut noter que ces modèles sont équivalents aux *modèles à espace d'état* non linéaires et non gaussiens :

$$\begin{aligned} X_t &= f_{t-1}(\Theta, X_{t-1}, W_{t-1})\,, \\ Y_t &= h_t(\Theta, X_t, V_t)\,, \end{aligned}$$

où W_t et V_t sont de bruits blancs (i.e. des suites de variables aléatoires indépendantes et centrées) et où on suppose que les termes X_1, Θ, $(W_t)_{1\le t<T}$, $(V_t)_{1\le t\le T}$ sont mutuellement indépendants. Les modèles à espace d'état font également leur apparition en écologie et environnement, notamment pour modéliser des dynamiques de ressources naturelles.

3. Algorithmes de Metropolis–Hasting

L'idée originelle et brillante des méthodes MCMC est publiée en 1953 par Metropolis et ses coauteurs [31], elle est étendue en 1970 par Hastings [26]. En 1984, les frères Geman [21] proposent l'échantillonneur de Gibbs pour la restauration bayésienne d'images. Cet échantillonneur est un cas particulier de l'échantillonneur de Metropolis–Hastings, il est proposé en inférence bayésienne par Gelfand et Smith [20]. L'algorithme de Metropolis est également à l'origine des techniques de recuit simulé [29].

C'est au début des années 90, après le développement et la démocratisation de l'outil informatique, que ces outils rencontrent un important succès. On s'intéresse ici au cas des espaces d'état généraux (continus) [41].

Étant donnée une densité π sur E, ici nous considérerons la densité (1), le but des méthodes MCMC est de construire une chaîne de Markov $(Z^{(k)})_{k\in\mathbb{N}}$ sur E dont la densité de sa loi invariante est justement π. Sous des conditions relativement faibles, le théorème ergodique implique que la mesure empirique associée à la chaîne converge vers la densité cible π [39] :

$$\frac{1}{k}\sum_{\ell=0}^{k-1}\varphi(Z^{(\ell)}) \xrightarrow[k\to\infty]{\text{p.s.}} \int_E \varphi(z)\,\pi(z)\,dz$$

pour toute fonction φ continue et bornée, et pour toute condition initiale $Z^{(0)}$ choisie p.p. dans E.

3.1. Échantillonneur Metropolis–Hasting (MH)

L'algorithme MH répond à la question suivante : on se donne un noyau markovien, dit de proposition, de densité $q^{\text{prop}}(x'|x)$ défini sur E, comment perturber q^{prop} afin d'obtenir une chaîne de Markov dont la densité de la mesure invariante est π ? La solution consiste à perturber le noyau de proposition à l'aide d'une étape d'acceptation/rejet. L'itération $Z^{(k)} \to Z^{(k+1)}$ se fait en deux étapes :

Proposition : On simule une proposition

$$\tilde{Z} \sim q^{\text{prop}}(\,\cdot\,|Z^{(k)})\,.$$

Acception/rejet : On accepte $\tilde{Z}$ avec probabilité α et on la rejette avec probabilité $1-\alpha$, i.e.

$$Z^{(k+1)} = \begin{cases} \tilde{Z}\,, & \text{avec probabilité } \alpha\,, \\ Z^{(k)}\,, & \text{avec probabilité } 1-\alpha\,. \end{cases}$$

La probabilité α est choisie de telle façon que ce noyau perturbé laisse invariante la densité cible π, un simple calcul donne :

$$\alpha = \frac{q^{\text{prop}}(\tilde{Z}|Z^{(k)})}{\pi(\tilde{Z})}\,\frac{\pi(Z^{(k)})}{q^{\text{prop}}(Z^{(k)}|\tilde{Z})} \wedge 1\,.$$

Même si la convergence de cet algorithme est prouvée dans un cadre relativement général, en pratique tous les noyaux de proposition ne conviennent pas. Le noyau doit "guider" l'algorithme dans les "bonnes régions" de l'espace d'état. En particulier, cet algorithme n'est pas adapté à la situation présente où l'espace d'état est un espace de trajectoires. Il est préférable d'utiliser une approche "composante à composante" comme celle de l'échantillonneur de Gibbs.

3.2. Échantillonneur de Gibbs

Supposons que l'on sache échantillonner selon les densités marginales conditionnelles :

$$\pi_t(x_t|x_{\neg t},\theta)\,dx_t \stackrel{\text{déf}}{=} \mathbb{P}(X_t \in dx_t|X_{\neg t}=x_{\neg t},\Theta=\theta,Y_{1:T}=y_{1:T})\,, \tag{8a}$$

$$\pi_\Theta(\theta|x_{1:T})\,d\theta \stackrel{\text{déf}}{=} \mathbb{P}(\Theta \in d\theta|X_{1:T}=x_{1:T},Y_{1:T}=y_{1:T}) \tag{8b}$$

pour $t = 1 \cdots T$, avec

$$\neg t \stackrel{\text{déf}}{=} \{s\,;\,1 \le s \le T,\, s \ne t\}\,.$$

Partant d'une configuration initiale $Z^{(0)}$, l'itération $k \to k+1$ consiste à mettre à jour une seule composante : soit $X_t^{(k)}$ (par exemple pour un indice t tiré au hasard), soit $\Theta^{(k)}$. La mise à jour de la composante $X_t^{(k)}$ se fait simplement en posant :

$$X_t^{(k+1)} \sim \pi_t^{\text{prop}}(x_t|X_{\neg t}^{(k)},\Theta^{(k)})\,dx_t$$

les autres composantes restant inchangées ($X_s^{(k+1)} = X_s^{(k)}$ pour tout $s \ne t$). La mise à jour de la composante $\Theta^{(k)}$ se fait de façon analogue.

En pratique, cet algorithme se montre plutôt rapide, mais il nécessite de savoir échantillonner selon les lois marginales conditionnelles ce qui en dehors des modèles gaussiens, notamment pour les modèles de Markov cachés traités ici, n'est pas possible. Dans le cas général, il est possible d'utiliser une technique hybride.

3.3. Échantillonneur de Metropolis–Hasting hybride (MHh)

L'échantillonneur de Metropolis–Hastings hybride (MHh) est particulièrement adapté aux modèles de Markov cachés [23]. Comme pour l'échantillonneur de Gibbs les composantes sont mises à jour les unes après les autres, mais comme pour l'échantillonneur MH, cette mise à jour se fait sans savoir explicitement échantillonner selon les densités marginales conditionnelles (8). On se donne deux noyaux de proposition :

$$q_t^{\text{prop}}(x'_t|x_{1:T},\theta)\,dx'_t\,,$$
$$q_\Theta^{\text{prop}}(\theta'|x_{1:T},\theta)\,d\theta'\,.$$

Partant d'une configuration initiale $Z^{(0)}$, l'itération $k \to k+1$ consiste à mettre à jour une seule composante : soit $X_t^{(k)}$ (par exemple pour un indice t tiré au hasard), soit $\Theta^{(k)}$.

La mise à jour de la composante $X_t^{(k)}$ se fait en deux étapes :

Proposition : On génère d'abord un candidat selon le noyau de proposition :

$$\tilde{X}_t \sim q_t^{\text{prop}}(x'_t|X_t^{(k)},X_{\neg t}^{(k)},\Theta^{(k)})\,dx'_t\,.$$

Acceptation/rejet : Ce candidat est alors accepté ou rejeté selon la règle :

$$X_t^{(k+1)} \leftarrow \begin{cases} \tilde{X}_t\,, & \text{avec probabilité } \alpha \\ X_t^{(k)}, & \text{avec probabilité } 1-\alpha \end{cases}$$

où

$$\alpha \stackrel{\text{déf}}{=} \frac{\pi_t(\tilde{X}_t|X_{\neg t}^{(k)},\Theta^{(k)})}{q_t^{\text{prop}}(\tilde{X}_t|X_t^{(k)},X_{\neg t}^{(k)},\Theta^{(k)})}\frac{q_t^{\text{prop}}(X_t^{(k)}|\tilde{X}_t,X_{\neg t}^{(k)},\Theta^{(k)})}{\pi_t(X_t^{(k)}|X_{\neg t}^{(k)},\Theta^{(k)})} \wedge 1\,.$$

Les autres composantes restant inchangées : $X_s^{(k+1)} = X_s^{(k)}$ pour $s \neq t$ et $\Theta^{(k+1)} = \Theta^{(k)}$.

La mise à jour de la composante $\Theta^{(k)}$ se fait également en deux étapes :

Proposition : On génère d'abord un candidat selon le noyau de proposition :

$$\tilde{\Theta} \sim q_\Theta^{\text{prop}}(\theta'|X_{1:T}^{(k)},\Theta^{(k)})\,d\theta'\,.$$

Acceptation/rejet : Ce candidat est alors accepté ou rejeté selon la règle :

$$\Theta^{(k+1)} \leftarrow \begin{cases} \tilde{\Theta}\,, & \text{avec probabilité } \alpha \\ \Theta^{(k)}, & \text{avec probabilité } 1-\alpha \end{cases}$$

où

$$\alpha \stackrel{\text{déf}}{=} \frac{\pi_\Theta(\tilde{\Theta}|X_{1:T}^{(k)})}{q_t^{\text{prop}}(\tilde{\Theta}|X_{1:T}^{(k)},\Theta^{(k)})}\frac{q_\Theta^{\text{prop}}(\Theta^{(k)}|X_{1:T}^{(k)},\tilde{\Theta})}{\pi_\Theta(\Theta^{(k)}|X_{1:T}^{(k)})} \wedge 1\,.$$

Les autres composantes restant inchangées : $X_{1:T}^{(k+1)} = X_{1:T}^{(k)}$.

Il est nécessaire de définir un noyau de proposition q_t^{prop}. Celui-ci dirige l'exploration de l'espace par la chaîne de Markov, son choix est donc critique notamment lorsque l'espace à explorer est de grande dimension.

Il existe un choix relativement simple de noyaux de proposition dans le cas présent. La décomposition suivante (cf. [9]) :

$$\pi_t(x_t|x_{\neg t},\theta) \propto \underbrace{q_\theta(x_{t-1},x_t)}_{=:q_t^{\text{prop}}(x_t|x_{t-1},\theta)}\ \underbrace{q_\theta(x_t,x_{t+1})\,\psi_\theta(x_t,y_t)}_{=:\rho_t(x_t,x_{t+1},\theta)} \tag{9}$$

fournit un noyau de proposition simple puisqu'il s'agit du noyau de transition de la chaîne de Markov X_t. Dans ce cas la probabilité α s'écrit :

$$\alpha \stackrel{\text{déf}}{=} \frac{\rho_t(\tilde{X}_t,X_{t+1}^{(k)},\Theta^{(k)})}{\rho_t(X_t^{(k)},X_{t+1}^{(k)},\Theta^{(k)})}\,.$$

Le choix naturel (9) fait que l'échantillonneur de Metropolis–Hastings hybride est souvent utilisé dans cette situation. Toutefois, comme nous le verrons, bien que ce choix soit naturel, il n'est pas forcément judicieux.

3.4. Remarques

Les algorithmes MH sont itératifs et convergent sous des hypothèses relativement faibles. Ils souffrent néanmoins de problèmes de vitesse de convergence et de mise en œuvre [38, 17, 6, 5, 28, 39, 27].

En pratique, on ne peut simuler qu'un nombre fini d'itérations de MCMC, le choix de la condition initiale peut donc influencer la qualité de convergence. Il est alors nécessaire de ne pas tenir compte des premières itérations de l'algorithme, mais il est difficile de déterminer la durée de cette période de "chauffe" ("burn-in period"). En dimension plus grande que 1, il n'existe pas de critère rigoureux de convergence permettant de savoir si la chaîne est proche de son régime asymptotique. Il existe en revanche des techniques empiriques [17]. De plus, la vitesse de convergence de ces algorithmes peut être trop faible dans certaines situations.

En pratique, l'algorithme de Gibbs se montre le plus rapide, mais il ne peut pas s'utiliser dans toutes les situations. Il nécessite en effet de savoir échantillonner selon les marginales conditionnelles (8). Dans l'algorithme MH, le choix du noyau de proposition est important. Ce choix devient crucial dans le cas de l'algorithme MHh. En pratique, comme nous le verrons dans la Section 5, le noyau de proposition "canonique" (9) est le plus souvent médiocre. L'algorithme MHh s'applique dans de nombreux cas de modèles de Markov cachés, mais il peut induire de très faibles vitesses de convergence. Nous proposons comme alternative, de faire interagir plusieurs chaînes de Metropolis–Hastings hybrides.

4. Metropolis–Hastings hybride en interaction (MHhi)

Nous allons décrire une méthode faisant interagir N chaînes. Pour alléger la présentation, nous ne précisons pas la mise à jour du paramètre (elle se fait de façon analogue) et nous l'omettons dans les notations de cette section. Nous considérons donc un processus noté :

$$k \mapsto (X^{(k)}_{1:T,1}, \ldots, X^{(k)}_{1:T,N})$$

à valeurs dans $((\mathbb{R}^n)^T)^N = \mathbb{R}^{n\times T\times N}$.

Soient N configurations initiales notées

$$(X^{(0)}_{1:T,1}, \ldots, X^{(0)}_{1:T,N}),$$

l'itération $X^{(k)}_{0:T} \rightarrow X^{(k+1)}_{0:T}$ consiste à choisir une chaîne i et une composante t (par exemple choisies au hasard). Chaque chaîne j propose un candidat $X_{t,j}$ ($j = 1, \ldots, N$)

et le terme $X_{t,i}^{(k)}$ est mis à jour en choisissant parmi l'ensemble de ces candidats ou en les rejetant.

Proposition : Les N candidats sont simulés de la façon suivante :

$$\tilde{X}_{t,j} \sim q_t^{\text{prop}}(x'_t | X_{t,j}^{(k)}, X_{\neg t,j}^{(k)})\, dx'_t\,, \quad j = 1 \cdots N$$

où q_t^{prop} peut être choisi comme en (9).

Acceptation/rejet : La mise à jour s'effectue alors selon la règle :

$$X_{t,i}^{(k+1)} \leftarrow \begin{cases} \tilde{X}_{t,1}\,, & \text{avec probabilité } \frac{1}{N}\alpha_1\,, \\ \vdots & \\ \tilde{X}_{t,N}\,, & \text{avec probabilité } \frac{1}{N}\alpha_N\,, \\ X_{t,i}^{(k)}\,, & \text{avec probabilité } 1 - \frac{1}{N}\sum_{j=1}^{N}\alpha_j\,, \end{cases}$$

où

$$\alpha_j \stackrel{\text{déf}}{=} \frac{\pi_t(\tilde{X}_{t,j} | X_{\neg t,i}^{(k)})}{q_t^{\text{prop}}(\tilde{X}_{t,j} | X_{t,i}^{(k)}, X_{\neg t,j}^{(k)})} \frac{q_t^{\text{prop}}(X_{t,i}^{(k)} | \tilde{X}_{t,j}, X_{\neg t,j}^{(k)})}{\pi_t(X_{t,i}^{(k)} | X_{\neg t,i}^{(k)})} \wedge 1\,.$$

Les processus $X_{1:T,i}^{(k)}$ pris séparément *ne sont pas markoviens*, en revanche l'ensemble :

$$k \mapsto \mathbb{X}^{(k)} = (X_{1:T,1}^{(k)}, \ldots, X_{1:T,N}^{(k)})$$

est markovien et sa densité invariante est la densité produit $\pi^{\otimes N}$. Ce résultat technique est démontré dans [10]. En d'autres termes, asymptotiquement l'algorithme produit un N échantillon de la densité cible π.

Nous n'abordons pas les problèmes d'implémentation dans cet article. Il faut néanmoins noter que *la nature parallèle demande un travail particulier afin d'être exploité en pratique*. Il serait en particulier nécessaire de faire appel à des générateurs parallèles de nombres pseudo–aléatoires. De plus, pour pleinement profiter de l'aspect parallèle de l'algorithme, il faudrait également ne pas faire interagir les chaînes à toutes les itérations. Une stratégie serait de faire beaucoup d'interactions au début de la procédure MCMC et d'en diminuer la fréquence au fur et à mesure de l'évolution de la procédure.

5. Simulations numériques

Afin d'évaluer le gain apporté par la procédure d'interaction, nous allons comparer deux algorithmes :

Metropolis–Hastings hybride en interaction (MHhi) : c'est l'algorithme décrit à la Section 4 proposant N chaînes en interaction ;

Metropolis–Hastings hybride en parallèle MHhp : dans cet algorithme nous simulons indépendamment N chaînes MHh en parallèle.

Asymptotiquement, ces deux algorithmes produisent un échantillon de taille N de la densité cible π. Nous nous proposons de comparer empiriquement les vitesses de convergence vers π des deux méthodes.

On considère en premier lieu un exemple simple qui montre déjà les limites de MHhp. Le deuxième exemple est un modèle de pêcherie non linéaire.

5.1. Modèle conditionnellement linéaire–gaussien

Considérons le modèle :

$$X_{t+1} = \Theta\, X_t + W_t\,, \qquad\qquad X_1 \sim \mathcal{N}(4, 3^2)\,,$$
$$Y_t = 2\, X_t + V_t$$

pour $t = 1, \ldots, T = 10$. Les processus $W_{1:T}$ et $V_{1:T}$ sont des bruits blancs gaussiens de variances $\sigma_W^2 = 9$ et $\sigma_V^2 = 25$. $W_{1:T}$, $V_{1:T}$, X_1 et Θ sont mutuellement indépendants. Nous supposons que Θ est inconnu. Nous avons donc $q_\theta(x, x')\,\mathrm{d}x' = \mathcal{N}(\theta x, \sigma_W^2)$ et $\psi^\theta(x, y)\,\mathrm{d}y = \mathcal{N}(2x, \sigma_V^2)$, i.e.

$$q_\theta(x, x') \propto \exp(-\frac{1}{2\sigma_W^2}|x' - \theta x|^2),$$
$$\psi^\theta(x, y) \propto \exp(-\frac{1}{2\sigma_V^2}|y - 2\,x|^2)\,.$$

Comme le système est conditionnellement linéaire–gaussien, l'échantillonneur de Gibbs fournit une très bonne estimation $\hat{\pi}$ de π [37].

Pour chacune des méthodes, MHhi et MHhp, nous pouvons calculer des indicateurs $\varepsilon^{\text{MHhi}}$ et $\varepsilon^{\text{MHhp}}$ de l'erreur L^1 entre $\hat{\pi}$ et l'estimation par noyau de convolution de la densité cible construite à partir des valeurs finales des N chaînes. Ces indicateurs doivent décroître et rester petits lorsque l'algorithme converge vers la densité stationnaire π (cf. [10] pour les détails). Ils ne tendent pas vers 0 dans la mesure où l'on travaille à un nombre N de chaînes fixé. Il y a convergence vers 0 lorsque $N \uparrow \infty$.

Afin de comparer équitablement les algorithmes MHhi et MHhp, nous représentons sur la Figure 2 les indicateurs $\varepsilon^{\text{MHhi}}$ et $\varepsilon^{\text{MHhp}}$, *non pas en fonction du nombre d'itérations des algorithmes, mais en fonction du temps CPU.*

Sur la Figure 2 nous constatons que bien que les itérations de l'algorithme MHhi soient plus coûteuses en temps CPU que celles de MHhp, l'algorithme MHhi converge beaucoup plus rapidement.

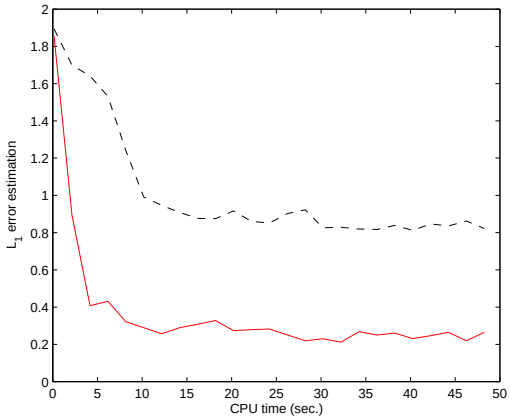

Figure 2. *Évolution de l'indicateur ε^{MHhp} pour MHhp (- -) et ε^{MHhi} pour MHhi (–) en fonction du temps CPU. L'indicateur pour MHhi atteint rapidement une valeur proche de 0.2, cette valeur n'est pas proche de 0 dans la mesure on l'on travaille à un nombre fixé $N = 50$ de chaînes. L'indicateur pour MHhp ne converge que très lentement.*

5.2. Un modèle de pêcherie : le modèle Ricker

Le modèle de Ricker permet de décrire l'évolution de la biomasse N_t d'une pêcherie sur une période de T années. Nous le considérons sous forme logarithmique, $X_t = \log N_t$, associé à un processus d'observation bruité :

$$X_{t+1} = X_t + \Theta - 0.02\, e^{X_t} + W_t\,, \qquad X_1 \sim \mathcal{N}(\mu_1, \sigma_1^2)\,,$$
$$Y_t = X_t + V_t$$

pour $t = 1, \ldots, T = 20$, où $W_{1:T}$ et $V_{1:T}$ sont des bruits blancs gaussiens de variances $\sigma_W^2 = 1$ et $\sigma_V^2 = 0.5$. $W_{1:T}$, $V_{1:T}$, X_1 et Θ sont mutuellement indépendants. Nous supposons que Θ est un paramètre inconnu de loi *a priori* $\mathcal{N}(4, 2^2)$ et que la vraie valeur du paramètre est 1.5.

Les noyaux de transition et d'émission sont :

$$q_\theta(x, x') \propto \exp\Big(-\frac{1}{2\sigma_W^2} |x' - (x + \theta - 0.02\, e^x)|^2 \Big)\,,$$
$$\psi^\theta(x, y) \propto \exp\Big(-\frac{1}{2\sigma_V^2} |y - x|^2 \Big)\,.$$

Contrairement au cas linéaire–gaussien, nous ne sommes pas capables d'évaluer la distance avec la densité cible π. Pour comparer les deux méthodes, nous considérons l'évolution de la composante des chaînes correspondant à Θ, i.e. $k \mapsto \Theta_i^{(k)}$ pour $i = 1, \ldots, N$.

Nous considérons les deux méthodes MHhp et MHhi avec $N = 50$ chaînes, pour les comparer équitablement les représentations sont en temps CPU.

Les résultats (cf. Figures 3 et 4) sont encore en faveur du MHhi. En effet, pour MHhi toutes les chaînes convergent rapidement vers un état stationnaire correspondant à la vraie valeur du paramètre alors que pour MHhp la convergence des chaînes est lente.

6. Conclusion

Les modèles de Markov cachés permettent de modéliser beaucoup de problèmes d'évolution notamment en écologie et en environnement. Leur nature probabiliste et hiérarchique convient parfaitement aux méthodes d'inférence bayésienne comme les méthodes MCMC. Parmi celles-ci, l'échantillonneur de Metropolis–Hastings hybride est souvent utilisé car il peut être appliqué dans la plupart des cas et notamment aux modèles de Markov cachés (i.e. modèles à espace d'état non linéaires et non gaussiens).

Nous avons constaté sur des simulations que cet échantillonneur peut converger beaucoup trop lentement pour être utilisé en pratique, y compris dans des cas simples. Nous avons proposé de faire interagir N de ces échantillonneurs afin d'améliorer la convergence de l'algorithme. Nous avons illustré le gain obtenu sur des simulations. Cette simple constatation empirique n'est naturellement pas suffisante, il est en effet nécessaire d'analyser le gain obtenu en termes de *vitesse* de convergence. Pour cela il est nécessaire d'étudier les théorèmes de limite centrale pour les procédures MCMC ce qui demande de profonds développements et sera abordé dans le futur.

7. Bibliographie

[1] M. E. ALFARO et M. T. HOLDER : The posterior and the prior in Bayesian phylogenetics. *Annual Review of Ecology, Evolution, and Systematics*, 37:19–42, 2006.

[2] D. ASHBY : Bayesian statistics in medicine : a 25 year review. *Statistics in Medecine*, 25:3589–3631, 2006.

[3] M. A. BEAUMONT et B. RANNALA : The Bayesian revolution in genetics. *Nature Review Genetics*, 5(4):251–261, April 2004.

[4] S. P. BROOKS : Bayesian computation : A statistical revolution. *Transactions of the Royal Society, Series A*, 361:2681–2697, 2003.

[5] S. P. BROOKS et A. GELMAN : General methods for monitoring convergence of iterative simulations. *Journal of Computational and Graphical Statistics*, 7(4):434–455, 1998.

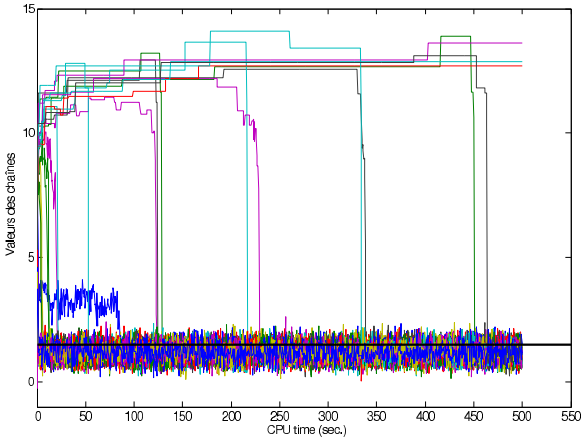

Figure 3. *Tracés de $k \mapsto \Theta_i^{(k)}$ pour $i = 1, \ldots, 50$ pour l'approche MHhp (sans interaction) : beaucoup de chaînes n'ont pas encore convergé. D'autres simulations ont montré que certaines ne convergent que très lentement.*

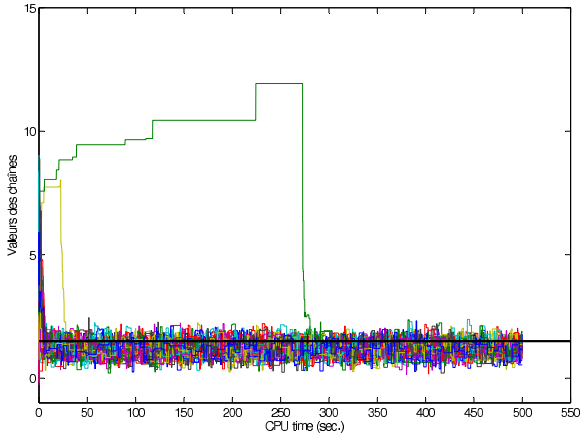

Figure 4. *Tracés de $k \mapsto \Theta_i^{(k)}$ pour $i = 1, \ldots, 50$ avec l'approche MHhi (avec interactions) : les chaînes convergent rapidement vers un état stationnaire correspondant à la vraie valeur.*

[6] S. P. BROOKS et G. O. ROBERTS : Convergence assessment techniques for Markov chain Monte Carlo. *Statistics and Computing*, 8(4):319–335, 1998.

[7] F. CAMPILLO, R. RAKOTOZAFY et V. ROSSI : Bayesian numerical inference for hidden Markov models. *In International Conference on Applied Statistics for Development in Africa Sada'07*, 2007.

[8] F. CAMPILLO : Probabilistic modeling an Bayesian inference. *In Conférence en l'honneur de C. Lobry, Saint-Louis du Sénégal*, 2008. To appear.

[9] F. CAMPILLO et R. RAKOTOZAFY : MCMC for nonlinear/non–Gaussian state–space models, Application to fishery stock assessment. *In CARI'04*, Hammamet, Tunisia, 2004.

[10] F. CAMPILLO et V. ROSSI : Parallel and interacting Markov chains Monte Carlo method. Research Report RR-6008, INRIA, 06 2006. `http://hal.inria.fr/inria-00103871`.

[11] B. P. CARLIN, J. S. CLARK et A. E. GELFAND : Elements of hierarchical bayesian influence. *In [15]*, 2006.

[12] S. CHIB et E. GREENBERG : Markov Chain Monte Carlo simulation methods in econometrics. *Econometric Theory*, 12:409–431, 1996.

[13] S. CHIB, F. NARDARI et N. SHEPHARD : Markov chain Monte Carlo methods for stochastic volatility models. *Journal of Econometrics*, 108(2):281–316, 2002.

[14] J. S. CLARK : Why environmental scientists are becoming Bayesians. *Ecology Letters*, 8(1):2–14, 2005.

[15] J. S. CLARK et A. GELFAND, éditeurs. *Hierarchical Modelling for the Environmental Sciences : Statistical Methods and Applications*. Oxford University Press, Inc., New York, NY, USA, 2006.

[16] J. CORANDER : Is there a real Bayesian revolution in pattern recognition for bioinformatics ? *Current Bioinformatics*, 1(2):161–165, May 2006.

[17] M. K. COWLES et B. P. CARLIN : Markov chain Monte Carlo convergence diagnostics : a comparative review. *Journal of the American Statistical Association*, 91(434):883–904, 1996.

[18] A. M. ELLISON : Bayesian inference in ecology. *Ecology Letters*, 7(6):509–520, 2004.

[19] C. GAUCHEREL, F. CAMPILLO, L. MISSON, J. GUIOT et J.-J. BOREUX : Parameterization of a process-based tree-growth model : comparison of optimization, MCMC and particle filering algorithms. *Ecological Modelling, To appear*, 2007.

[20] A. E. GELFAND et A. F. M. SMITH : Sampling-based approaches to calculating marginal densities. *Journal of the American Statistical Association*, 85(410):398–409, juin 1990.

[21] S. GEMAN et D. GEMAN : Stochastic relaxation, Gibbs distribution, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6, 1984.

[22] J. GEWEKE : Using simulation methods for Bayesian econometric models : Inference, development, and communication. *Econometric Reviews*, 18:1–126, 1999.

[23] J. GEWEKE et H. TANIZAKI : Bayesian estimation of state-space model using the Metropolis-Hastings Algorithm within Gibbs sampling. *Computational Statistics and Data Analysis*, 37(2): 151–170, 2001.

[24] S. G. GIAKOUMATOS, P. DELLAPORTAS et D. N. POLITIS : Bayesian analysis of the unobserved arch model. *Statistics and Computing*, 15(2):103–111, 2005.

[25] W. R. GILKS, S. RICHARDSON et D. J. SPIEGELHALTER, éditeurs. *Markov Chain Monte Carlo in practice*. Chapman & Hall, London, 1996.

[26] K. W. HASTINGS : Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, avril 1970.

[27] G. L. JONES : On the Markov chain central limit theorem. *Probability Surveys*, 1:299–320, 2004.

[28] G. L. JONES et J. P. HOBERT : Honest exploration of intractable probability distributions via Markov chain Monte Carlo. *Statistical Science*, 16:312–334, 2001.

[29] S. KIRKPATRICK, C. D. GELATT et M. P. VECCHI : Optimization by simulated annealing. *Science*, 220:671–680, 1983.

[30] D. MALAKOFF : Bayes offers a "new" way to make sense of numbers. *Science*, 286:1460–1464, 1999.

[31] N. METROPOLIS, A. W. ROSENBLUTH, M. N. ROSENBLUTH, A. H. TELLER et E. TELLER : Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1091, 1953.

[32] R. MEYER et R. B. MILLAR : Bayesian stock assessment using a state-space implementation of the delay difference model. *Canadian Journal of Fisheries and Aquatic Sciences*, 56:37–52, 1999.

[33] A. E. PUNT et R. HILBORN : Fisheries stock assessment and decision analysis : the Bayesian approach. *Reviews in Fish Biology and Fisheries*, 7:35–63, 1997.

[34] L. RABINER : A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, février 1989.

[35] K. H. RECKHOW : Bayesian approaches in ecological analysis and modeling. *In* C. D. CANHAM, J. J. COLE et W. K. LAUENROTH, éditeurs : *Models in ecosystem science*, pages 168–183. Princeton University Press, Princeton, New Jersey, USA., 2003.

[36] E. RIVOT, E. PREVOST, E. PARENT et 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.

[37] C. P. ROBERT : *Méthodes de Monte Carlo par Chaînes de Markov*. Economica, Paris, 1996.

[38] J. S. ROSENTHAL : Convergence rates for Markov chains. *SIAM Review*, 37(3):387–405, 1995.

[39] J. S. ROSENTHAL : A review of asymptotic convergence for general state space Markov chains. *Far East J. Theor. Stat.*, 5:37–50, 2001.

[40] C. J. SCHWARZ et G. A. F. SEBER : Estimating animal abundance : Review III. *Statistical Science*, 14(4):427–456, 1999.

[41] L. TIERNEY : Markov chains for exploring posterior distributions (with discussion). *The Annals of Statistics*, 22(4):1701–1728, décembre 1994.