HAMILTONIAN MONTE CARLO IN PRACTICE

BY

Emily Walker
Samuel Soubeyrand

Research Report No. 49
February 2016

Unité Biostatistique et Processus Spatiaux
Institut National de la Recherche Agronomique
Avignon, France
http://www.avignon.inra.fr/biosp

Technical Report

Hamiltonian Monte Carlo in practice

Emily Walker[1] **and Samuel Soubeyrand**
BioSP, INRA, 84914 Avignon, France

Abstract. This technical report presents the MCMC algorithm with Hamiltonian sampler and provides several applications of this algorithm. It especially shows how to apply MCMC with Hamiltonian sampler to estimate the parameters and the latent variables of a spatial GLMM.

Key words. Hamiltonian dynamics; MCMC; Space-state model; Spatial hierarchical model.

1 Introduction

Markov chain Monte Carlo (MCMC) algorithms are used to sample from target probability distributions. They provide realizations of Markov chains that have the target distributions as their equilibrium distributions. After a transitory set of iterations, one (sub-)samples the states of the chain to obtain a sample from the target distribution.

In Bayesian statistics, MCMC methodology is often used to draw samples in the joint posterior distribution of parameters (and latent variables). This methodology is particularly useful when one handles hierarchical models whose likelihoods are written as integrals which cannot be analytically calculated.

An MCMC algorithm is based on repeated stochastic jumps in the space of parameters and latent variables. Several approaches have been proposed to perform the jumps. The two main approaches are the so-called Gibbs sampler (Casella and George, 1992) and the Metropolis-Hastings sampler (Chib and Greenberg, 1995). The Gibbs sampler consists of drawing new states for the parameters and latent variables by generating values from the conditional distribution of the parameters/variables to be updated given the

[1]Corresponding author: Emily.Walker@avignon.inra.fr

other parameters/variables and given data. The Metropolis-Hastings sampler consists of drawing new states for the parameters and latent variables by generating values from an arbitrary proposal distribution and accepting/rejecting these new values with probabilities depending on "how much the new values increase/decrease the value of the posterior distribution" (the acceptance probabilities depend also on a correction compensating the choice of the proposal distribution).

An alternative sampler is the Hamiltonian sampler first introduced in the statistical physics literature (Duane et al., 1987), and applied afterwards to statistical inference issues; see Neal (2011), Girolami and Calderhead (2011) and references therein. The Hamiltonian sampler can be viewed as a specific Metroplis-Hastings sampler in which the proposal is based on two key components: (i) some auxiliary random variables and (ii) an Hamiltonian dynamics applied to the parameters/variables to be updated and to the auxiliary variables. The auxiliary random variables allow the updating process to be stochastic. The (deterministic) Hamiltonian dynamics allows large jumps that are accepted with high probability.

Theoretical justification of the Hamiltonian sampler and its use in MCMC can be found in Neal (2011) and Girolami and Calderhead (2011). In this technical report, we show how MCMC with Hamiltonian sampler, called Hamiltonian Monte Carlo (HMC), is defined and applied. The first three applications presented below are toy examples. The last application is the estimation of the parameters and the latent variables of a spatial GLMM (Diggle et al., 1998). Our objective, beyond this technical report, is to infer parameters and latent variables of dispersal models which can be viewed as extensions of spatial GLMM. Such a model was built (and estimated via an MCMC algorithm with Metropolis-Hastings algorithm) in Bousset et al. (2015) to infer the spread of phoma canker. The use of HMC in such a case should allow us to significantly reduce computation times required for the estimation and, therefore, a finer exploration of model specifications (being able to rapidly fit the dispersal model to data should allow us to test numerous model specifications).

2 MCMC with Hamiltonian sampler

Let $Y \in \mathbb{R}^n$ denote a vector of response variables with distribution $Y \mapsto p(Y \mid \theta)$, where $\theta \in \mathbb{R}^m$ is a set of parameters with prior distribution $\theta \mapsto \pi(\theta)$. The posterior distribution of θ is:

$$p(\theta \mid Y) = \frac{p(Y \mid \theta)\pi(\theta)}{p(Y)} = \frac{p(Y \mid \theta)\pi(\theta)}{\int_{\mathbb{R}^m} p(Y \mid \eta) d\pi(\eta)}.$$

The MCMC algorithm with Hamiltonian sampler is given by Algorithm 1. It has to be tuned with the probability distribution q of an auxiliary vector A of random variables, and with a time τ at which the Hamiltonian's equation is solved at each iteration. This algorithm samples in the joint distribution $p(\theta \mid Y)q(A)$ whose factorized form implies that the subsample corresponding to θ is drawn in $p(\theta \mid Y)$.

Typically, q is a normal distribution with zero mean vector and identity covariance matrix. Time τ partially governs the amplitude of the jumps and has to be tuned to obtain adequate acceptance probabilities (neither to high nor to low). The solution of Equation (1) at time $t = \tau$ is typically obtained numerically with the leapfrog method (Neal, 2011).

Algorithm 2 details the implementation of the leapfrog algorithm which alternates movements for the parameter vector and movements for the auxiliary vector in the directions $\nabla_{z_1} H(z)$ and $-\nabla_{z_2} H(z)$, respectively, where $\nabla_{z_1} H(z)$ (resp. $\nabla_{z_2} H(z)$) denotes the gradient of H with respect to the components of z_1 (resp. z_2).

The sections below illustrate the implementation of the MCMC algorithm with Hamiltonian sampler in various settings. In the last application, which deals with a spatial hierarchical model, we will use for q a more sophisticated distribution than the standard normal distribution, and we will use for solving Equation (1) a more sophisticated algorithm than the leapfrog algorithm.

Algorithm 1 MCMC algorithm with Hamiltonian sampler.

initialization: set a value for $\theta^{(0)}$

for $k = 1, 2, \ldots$ **do** the following:

1: draw an auxiliary vector of variables $A \in \mathbb{R}^m$ with probability distribution $A \mapsto q(A)$

2: solve the following Hamiltonian's equation at time $t = \tau$ (i.e. compute the state $z(\tau)$):

$$\frac{dz}{dt} = J\nabla H(z), \tag{1}$$

where the initial condition is $z(0) = (\theta^{(k-1)}, A)$, $z = (z_1, z_2) \in \mathbb{R}^m \times \mathbb{R}^m$,

$$H(z) = -\log\{p(z_1 \mid Y)q(z_2)\},$$
$$J = \begin{pmatrix} \mathbf{0}_{m,m} & \mathbf{I}_{m,m} \\ -\mathbf{I}_{m,m} & \mathbf{0}_{m,m} \end{pmatrix},$$

$\mathbf{0}_{m,m}$ is the zero $m \times m$-matrix, $\mathbf{I}_{m,m}$ is the identity $m \times m$-matrix, and ∇H is the gradient of H (the gradient of H is the vector of size $2m$ whose component j is the partial derivative of H with respect to the j-th component of z, denoted by $z_{[j]}$, i.e. $\partial H/\partial z_{[j]}$)

3: set $\theta^* = z_1(\tau)$ and $A^* = z_2(\tau)$ ▷ θ^* is a candidate value for θ

4: compute the acceptance probability α:

$$\begin{aligned}\alpha &= \min\left\{1, \frac{p(\theta^* \mid Y)q(A^*)}{p(\theta^{(k-1)} \mid Y)q(A^{(k-1)})}\right\} \\ &= \min\left\{1, \exp\{H((\theta^{(k-1)}, A^{(k-1)})) - H((\theta^*, A^*))\}\right\}\end{aligned}$$

5: draw a uniform random variable U over $[0, 1]$ and update θ as follows:

if $U \leq \alpha$ **then**
set $\theta^{(k)} = \theta^*$
else
set $\theta^{(k)} = \theta^{(k-1)}$
end if

end for

For $m = 1$, Equation (1) can be written: $\begin{cases} dz_1/dt = \partial H/\partial z_2 \\ dz_2/dt = -\partial H/\partial z_1. \end{cases}$

Algorithm 2 Leapfrog algorithm for solving the Hamiltonian's equation (1) at time $t = \tau = L\varepsilon$, where the positive integer L and the positive real value ε determine the accuracy of the resolution and the amplitude of the jump. Using the vocabulary of Hamiltonian dynamics, z_1 and z_2 are called position and momentum, respectively.

make a half step for the momentum:

$$z_2(\varepsilon/2) = z_2(0) - (\varepsilon/2)\nabla_{z_1} H(z(0))$$

for $i = 1, \ldots, L-1$ **do** the following if $L > 1$:
set $t = i\varepsilon$
alternate full steps for the position and the momentum:

$$z_1(t) = z_1(t-\varepsilon) + \varepsilon\nabla_{z_2} H(z(t-\varepsilon))$$
$$z_2(t+\varepsilon/2) = z_2(t-\varepsilon/2) - \varepsilon\nabla_{z_1} H(z(t-\varepsilon/2))$$

end for
make a full step for the position:

$$z_1(L\varepsilon) = z_1(L\varepsilon - \varepsilon) + \varepsilon\nabla_{z_2} H(z(L\varepsilon - \varepsilon))$$

make a half step for the momentum:

$$z_2(L\varepsilon) = z_2(L\varepsilon - \varepsilon/2) - (\varepsilon/2)\nabla_{z_1} H(z(L\varepsilon - \varepsilon/2)).$$

3 Sampling in a 2D normal distribution

We want to obtain, with the HMC algorithm, a sample $\{\theta^{(1)}, \ldots, \theta^{(K)}\}$ from the 2D-normal distribution with mean vector $(0, 0)$ and covariance matrix $\Sigma = \left(\begin{smallmatrix} 1 & 0.8 \\ 0.8 & 1 \end{smallmatrix}\right)$. The distribution q is set to the 2D-normal distribution with mean vector $(0, 0)$ and covariance matrix $\left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix}\right)$. Therefore, the function H satisfies:

$$\begin{aligned} H((\theta, A)) &= -\log\{p(\theta)q(A)\} \\ p(\theta) &= \frac{1}{2\pi|\Sigma|^{1/2}} \exp\left(-\frac{\theta'\Sigma^{-1}\theta}{2}\right) \\ q(A) &= \frac{1}{2\pi} \exp\left(-\frac{A'A}{2}\right), \end{aligned}$$

where u' is the transpose of vector u and $|u|$ is the determinant of matrix u.

In this case, by setting $z_1 = \theta = (\theta_1, \theta_2)$ and $z_2 = A = (A_1, A_2)$,

$$J\nabla H = \begin{pmatrix} \nabla_{z_2} H \\ -\nabla_{z_1} H \end{pmatrix} = \begin{pmatrix} \partial H/\partial A_1 \\ \partial H/\partial A_2 \\ -\partial H/\partial \theta_1 \\ -\partial H/\partial \theta_2 \end{pmatrix} = \begin{pmatrix} A_1 \\ A_2 \\ -\Sigma^{-1}\theta \end{pmatrix},$$

and

$$\alpha = \min\left\{1, \frac{\exp\{-(\theta^*)'\Sigma^{-1}\theta^*/2 - (A^*)'A^*/2\}}{\exp\{-\theta'\Sigma^{-1}\theta/2 - A'A/2\}}\right\},$$

where (θ, A) stands for the current value $(\theta^{(k-1)}, A^{(k-1)})$ of (θ, A) in the MCMC.

Figure 1 gives a numerical example of the use of HMC for sampling in the 2D normal distribution. For this example, we solved the Hamiltonian's equation (1) with the leapfrog algorithm tuned by $\varepsilon = 0.3$ and $L = 20$.

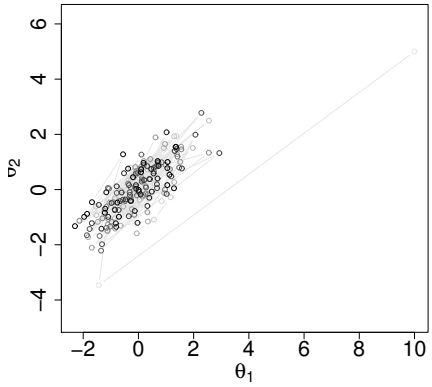

Figure 1: Chain of length 200 obtained with the HMC algorithm designed for sampling in the 2D normal distribution with mean vector $(0, 0)$, variances equal to 1 and correlation equal to 0.8. The chain was initialized at $\theta = (\theta_1, \theta_2) = (10, 5)$. Circles correspond to the states of θ at successive iterations. The grey level of circles evolves with iterations; blackest circles correspond to last iterations.

4 Sampling in a normal spatial random field incorporated into a spatial GLMM

Independent Poisson random variables $Y_1, \ldots, Y_n$ with means $\exp(\theta_1), \ldots, \exp(\theta_n)$ are observed at locations $x_1, \ldots, x_n$. The vector $\theta = (\theta_1, \ldots, \theta_n)'$ is a normal random vector with mean vector $(0, \ldots, 0)$ and covariance matrix Σ whose term (i, j) is equal to $\Sigma_{ij} = 3\exp(-3||x_i - x_j||)$, where $||\cdot||$ is the Euclidean distance. The locations $x_1, \ldots, x_n$ are independently drawn in the unit square $[0, 1] \times [0, 1]$. Figure 2 shows a realization of this model with $n = 18$.

To simulate $\theta = (\theta_1, \ldots, \theta_n)'$ given $Y = (Y_1, \ldots, Y_n)'$, we apply the HMC algorithm with q set to the standard nD-normal distribution. In this case, the function H satisfies:

$$\begin{aligned} H((\theta, A)) &= -\log\{p(\theta \mid Y)q(A)\} \\ &= -\log p(Y \mid \theta) - \log p(\theta) + \log p(Y) - \log q(A) \\ p(Y \mid \theta) &= \prod_{i=1}^{n} \frac{\exp(\theta_i)^{Y_i}}{Y_i!} \exp(-e^{\theta_i}) \\ p(\theta) &= \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} \exp\left(-\frac{\theta'\Sigma^{-1}\theta}{2}\right) \\ q(A) &= \frac{1}{(2\pi)^{n/2}} \exp\left(-\frac{A'A}{2}\right). \end{aligned}$$

In the expression of H, $p(Y)$ does not depend on (θ, A) and, consequently, will affect neither ∇H nor α. Therefore, we do not need to give the expression of $p(Y)$. It follows, by setting $z_1 = \theta$ and $z_2 = A$:

$$J\nabla H = \begin{pmatrix} \nabla_{z_2} H \\ -\nabla_{z_1} H \end{pmatrix} = \begin{pmatrix} A \\ Y - \exp(\theta) - \Sigma^{-1}\theta \end{pmatrix},$$

and

$$\alpha = \min\left\{1, \frac{\exp\{\sum_{i=1}^{n}(Y_i\theta_i^* - e^{\theta_i^*}) - (\theta^*)'\Sigma^{-1}\theta^*/2 - (A^*)'A^*/2\}}{\exp\{\sum_{i=1}^{n}(Y_i\theta_i - e^{\theta_i}) - \theta'\Sigma^{-1}\theta/2 - A'A/2\}}\right\}.$$

where (θ, A) stands for the current value $(\theta^{(k-1)}, A^{(k-1)})$ of (θ, A) in the MCMC.

Figure 3 gives a numerical example of the use of HMC applied to the data set shown in Figure 2. For this example, we solved the Hamiltonian's equation (1) with the leapfrog algorithm tuned by $\varepsilon = 0.1$ and $L = 20$.

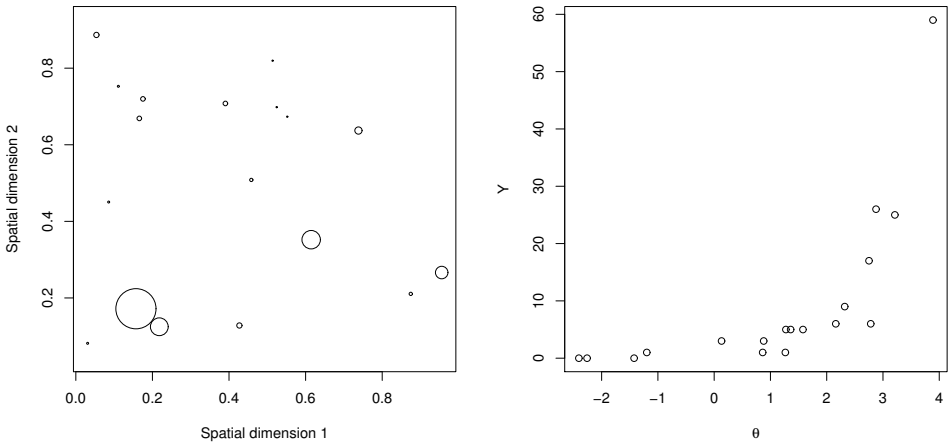

Figure 2: Realization of the spatial GLMM with $n = 18$. Left: Locations of observation sites; the radii of circles are proportional to the value of $Y_1, \ldots, Y_n$. Right: Observations $Y_1, \ldots, Y_n$ versus values of the normal random variables $\theta_1, \ldots, \theta_n$.

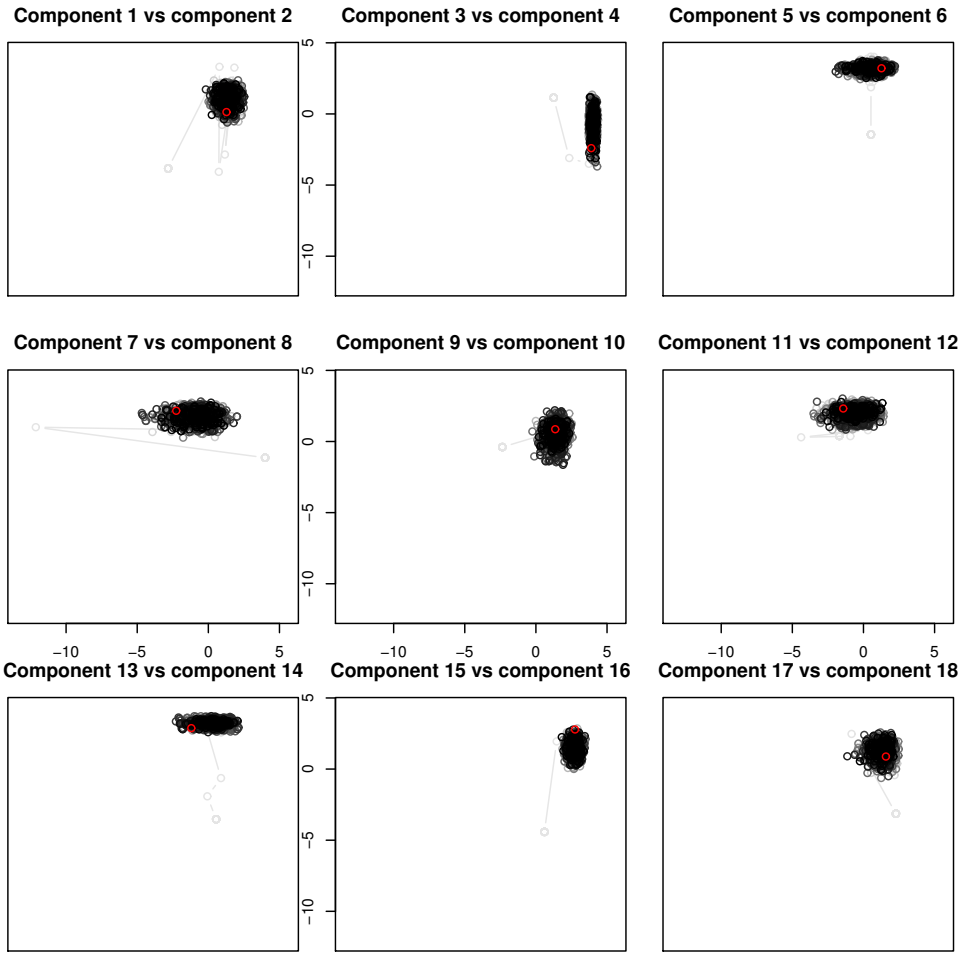

Figure 3: Chains of length 2000 obtained with the HMC algorithm designed for sampling in the normal random field incorporated in a spatial GLMM. Each panel of this plot shows a projection of the chain over two dimensions of θ. The chain was initialized at a value drawn in a normal vector with mean vector $(0, \ldots, 0)$ and covariance matrix equal to 2 times the identity matrix. Circles correspond to the states of θ at successive iterations. The grey level of circles evolves with iterations; blackest circles correspond to last iterations. In each panel, the red dot is the true value of θ.

5 Sampling in the posterior distribution of parameters and latent variables of a spatial GLMM

Here, we consider the model described above but the number of observations n is larger and the model parameters have also to be estimated.

5.1 Model

Given $\theta_1, \ldots, \theta_n$, the response variables $Y_1, \ldots, Y_n$, which are observed at locations $x_1, \ldots, x_n$, are independent Poisson random variables with means $\exp(\theta_1), \ldots, \exp(\theta_n)$. The vector $\theta = (\theta_1, \ldots, \theta_n)'$ is a normal random vector with mean vector $(\beta_1, \ldots, \beta_1)$ and covariance matrix $\Sigma(\beta_2, \beta_3)$ whose term (i, j) is equal to $\Sigma_{ij} = \beta_2 \exp(-\beta_3 ||x_i - x_j||)$. The locations $x_1, \ldots, x_n$ are regularly drawn in the unit square $[0, 1] \times [0, 1]$. Moreover, β_1, β_2 and β_3 have independent prior distributions; the prior of β_1 is normal whereas the priors of β_2 and β_3 are lognormal. Figure 4 shows a realization of this model with $n = 100$ and $\beta = (\beta_1, \beta_2, \beta_3) = (3, 1, 5)$.

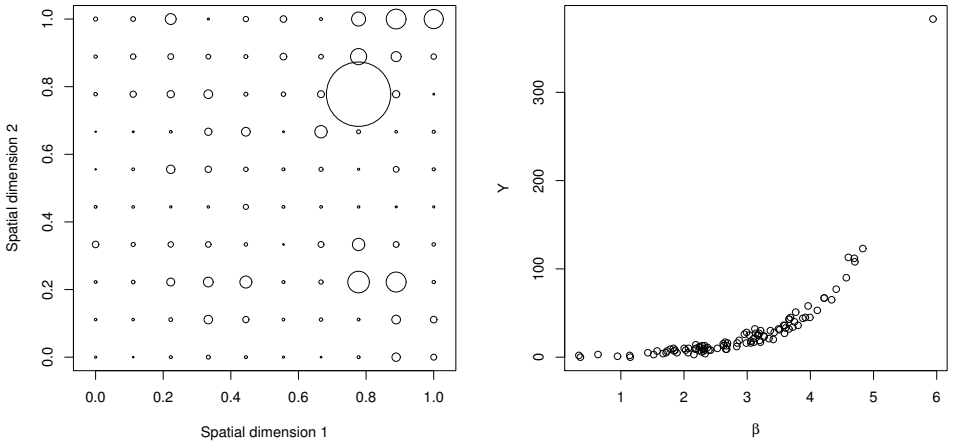

Figure 4: Realization of the spatial GLMM with $n = 100$. Left: Locations of observation sites; the radii of circles are proportional to the value of $Y_1, \ldots, Y_n$. Right: Observations $Y_1, \ldots, Y_n$ versus values of the normal random variables $\theta_1, \ldots, \theta_n$.

5.2 Semi-separable HMC

To estimate latent variables θ and parameters β of the model described above, we will use the procedure introduced by Zhang and Sutton (2014) who sequentially update these unknowns for solving the hamiltonian dynamics, and who propose (and justify) a normal distribution for q whose covariance matrix depends on (θ, β). This procedure is called semi-separable HMC.

The motivation for using a covariance-varying normal distribution for q is given by Girolami and Calderhead (2011): *"The potential of [...] HMC methodology may be more fully realized by employing transitions that take into account the local structure of the target density when proposing moves to different probability regions, as this may improve the overall mixing of the chain. Therefore, rather than employing a fixed global covariance matrix [...], a position-specific covariance could be adopted."*

Let $A \in \mathbb{R}^n$ denote the auxiliary vector for θ and $B \in \mathbb{R}^3$ the auxiliary vector for β. Let the auxiliary vector $\binom{A}{B}$ follow a normal distribution with zero mean vector and with covariance matrix $\Omega(\theta, \beta; x)$ (where $x = (x_1, \ldots, x_n)$), which is assumed to take the following block-diagonal form:

$$\Omega(\theta, \beta; x) = \begin{pmatrix} \Omega_A(\beta; x) & \mathbf{0}_{n,3} \\ \mathbf{0}_{3,n} & \Omega_B(\theta) \end{pmatrix}, \tag{2}$$

where $\Omega_A(\beta; x)$ and $\Omega_B(\theta)$ are the covariance matrices of A and B, respectively. This block-diagonal form (where Ω_A does not depend on θ and Ω_B does not depend on β) will allow simplifications in the expression of ∇H which will provide a computational advantage.

Using assumptions on the model and the auxiliary vectors,

$$
\begin{aligned}
q(A, B \mid \theta, \beta, x) &= q_A(A; \beta, x) q_B(B; \theta) \\
H(\theta, \beta, A, B) &= -\log\{p(\theta, \beta \mid Y) q(A, B \mid \theta, \beta, x)\} \\
&= -\log p(Y \mid \theta) - \log p(\theta \mid \beta) - \log p(\beta) + \log p(Y) \\
&\quad - \log q_A(A; \beta, x) - \log q_B(B; \theta) \\
p(Y \mid \theta) &= \prod_{i=1}^{n} \frac{\exp(\theta_i)^{Y_i}}{Y_i!} \exp(-e^{\theta_i}) \\
p(\theta \mid \beta) &= \frac{1}{(2\pi)^{n/2} |\Sigma(\beta_2, \beta_3)|^{1/2}} \exp\left(-\frac{(\theta - \beta_1 \mathbf{1}_n)' \Sigma(\beta_2, \beta_3)^{-1} (\theta - \beta_1 \mathbf{1}_n)}{2} \right) \\
p(\beta) &= \phi(\beta_1; b_{11}, b_{12}) \times \frac{1}{\beta_2} \phi(\log \beta_2; b_{21}, b_{22}) \times \frac{1}{\beta_3} \phi(\log \beta_3; b_{31}, b_{32}) \\
q_A(A; \beta, x) &= \frac{1}{(2\pi)^{n/2} |\Omega_A(\beta; x)|^{1/2}} \exp\left(-\frac{A' \Omega_A(\beta; x)^{-1} A}{2} \right) \\
q_B(B; \theta) &= \frac{1}{(2\pi)^{3/2} |\Omega_B(\theta)|^{1/2}} \exp\left(-\frac{B' \Omega_B(\theta)^{-1} B}{2} \right),
\end{aligned}
$$

where $\mathbf{1}_n$ is the unit vector of size n; $u \mapsto \phi(u; \mu, \sigma) = (2\pi\sigma)^{-1/2} \exp(-(u - \mu)^2/(2\sigma^2))$ is the density probability function of the normal distribution with mean μ and standard deviation σ; and $\{b_{ij} : i = 1, 2, 3, j = 1, 2\}$ are parameters of the prior distribution of β; β_1 has a normal prior distributions whereas β_2 and β_3 have log-normal prior distributions; *a priori*, the components of β are independent.

In the expression of H, $p(Y)$ does not depend on (θ, β, A, B) and, consequently, will affect neither ∇H nor α. Therefore, we do not need to give the expression of $p(Y)$.

It follows, by setting $z_1 = \binom{\theta}{\beta}$ and $z_2 = \binom{A}{B}$:

$$
\begin{aligned}
J \nabla H &= \begin{pmatrix} \nabla_{z_2} H \\ -\nabla_{z_1} H \end{pmatrix} \\
&= \begin{pmatrix} \Omega_A(\beta; x)^{-1} A \\ \Omega_B(\theta)^{-1} B \\ Y - \exp(\theta) - \Sigma(\beta_2, \beta_3)^{-1} (\theta - \beta_1 \mathbf{1}_n) + \nabla_\theta \log q_B(B; \theta) \\ \nabla_\beta \log p(\theta \mid \beta) + \nabla_\beta \log p(\beta) + \nabla_\beta \log q_A(A; \beta, x) \end{pmatrix},
\end{aligned}
$$

where the gradients $\nabla_\theta \log q_B(B; \theta)$, $\nabla_\beta \log p(\theta \mid \beta)$, $\nabla_\beta \log p(\beta)$ and $\nabla_\beta \log q_A(A; \beta, x)$ are specified in Appendix A.

Since $\nabla_{z_2} H$ depends on (θ, β) and not only on (A, B) and $\nabla_{z_1} H$ depends on (A, B) and not only on (θ, β), the Hamiltonian is non-separable[2] and, consequently, the leapfrog algorithm does not solve adequately Equation (1); see Girolami and Calderhead (2011). Girolami and Calderhead (2011) proposed a generalized leapfrog algorithm which has adequate properties but which is time consuming according to Zhang and Sutton (2014). Nevertheless, when the matrix $\Omega(\theta, \beta; x)$ satisfies Equation (2), the Hamiltonian is semi-separable, and Zhang and Sutton (2014) proposed to solve Equation (1) with the alternating block-wise leapfrog algorithm (ABLA) exploiting the semi-separability property. This procedure is described below.

Let H_1 and H_2 denote the two following Hamiltonian energies:

$$H_1(\theta, A, \beta, B) = -\log p(Y \mid \theta) - \log p(\theta \mid \beta) - \log q_A(A; \beta, x) - \log q_B(B; \theta)$$
$$H_2(\theta, A, \beta, B) = -\log p(\theta \mid \beta) - \log p(\beta) - \log q_A(A; \beta, x) - \log q_B(B; \theta).$$

Then, Equation (1) can be written:

$$\begin{cases} \frac{d\theta}{dt} &= \nabla_A H_1(\theta, A, \beta, B) \\ \frac{d\beta}{dt} &= \nabla_B H_2(\theta, A, \beta, B) \\ \frac{dA}{dt} &= -\nabla_\theta H_1(\theta, A, \beta, B) \\ \frac{dB}{dt} &= -\nabla_\beta H_2(\theta, A, \beta, B), \end{cases} \tag{3}$$

where $\nabla_A H_1(\theta, A, \beta, B)$ does not depend on θ, $\nabla_B H_2(\theta, A, \beta, B)$ does not depend on β, $\nabla_\theta H_1(\theta, A, \beta, B)$ does not depend on A, and $\nabla_\beta H_2(\theta, A, \beta, B)$ does not depend on B. Therefore, the Hamiltonian system corresponding to H_1 (resp. H_2) is separable with respect to (θ, A) (resp. (β, B)), and the non-separable Hamiltonian system (3) is said to be semi-separable[3]. This system can be numerically solved by sequentially applying the leapfrog algorithm (i) to solve (with respect to (θ, A)) the separable sub-system corresponding to H_1 and (ii) to solve (with respect to (β, B)) the separable sub-system corresponding to H_2.

To sum up, Algorithm 3 describes the MCMC algorithm with Hamiltonian sampler adapted to a hierarchical model and a non-separable Hamiltonian system. Algorithm 3 reduces to Algorithm 1 when the Hamiltonian system is separable. As stated above, to obtain a valid MCMC, the

[2]This can also be viewed by noting that the Hamiltonian energy $(\theta, \beta, A, B) \mapsto H(\theta, \beta, A, B)$ cannot be written as the sum of a function of (θ, β) and a function of (A, B).

[3]The separability of both Hamiltonian systems corresponding to H_1 and H_2 can be seen by noting that the function $(\theta, A) \mapsto H_1(\theta, A, \beta, B)$ can be written as the sum of a function of θ and a function of A, and the function $(\beta, B) \mapsto H_2(\theta, A, \beta, B)$ can be written as the sum of a function of β and a function of B.

system (4) in Algorithm 3 should not be solved with the leapfrog algorithm. However, when the Hamiltonian system is semi-separable (i.e. when $q(A, B \mid \theta, \beta) = q(A \mid \beta)q(A \mid \theta)$) one can use successive calls to the leapfrog algorithm, that is to say the alternating block-wise leapfrog algorithm (ABLA), described in Algorithm 4.

5.3 Application

We applied Algorithm 3 including Algorithm 4 to the data set shown on Figure 4 simulated with $\beta = (3, 1, 5)$. We followed Zhang and Sutton (2014) to choose the covariance matrices of the auxiliary vectors: $\Omega_A(\beta; x) = \Sigma(\beta_2, \beta_3)$ and $\Omega_B(\theta) = \mathbf{I}_{3,3}$. We tuned the algorithm with $\tilde{L} = 20$ and $\tilde{\varepsilon} = 0.1$. Prior parameters for β_i were fixed at $(b_{i1}, b_{i2}) = (0, 3)$, $i = 1, 2, 3$. We ran a chain of length 2000. The initial value for θ_i was $\log(\max(0.1, Y_i))$, $i = 1, \ldots, n$. The initial values for β_1 and β_2 were, respectively, the average and the variance of $\{\log(\max(0.1, Y_i)) : i = 1, \ldots, n\}$. The initial value for β_3 was 1.

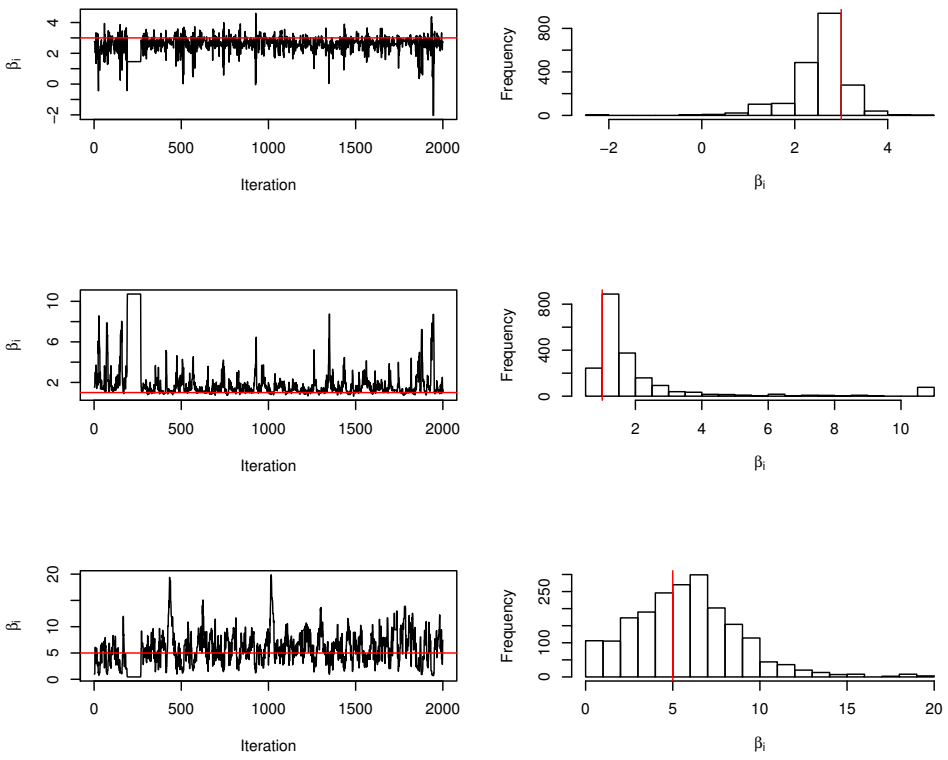

Figure 5: Chains of length 2000 (left) and corresponding histograms (right) obtained for β_1, β_2 and β_3 with the semi-separable HMC algorithm designed for sampling in the posterior distribution of latent variables and parameters incorporated in a spatial GLMM. True values of parameters are indicated in every panels by red lines.

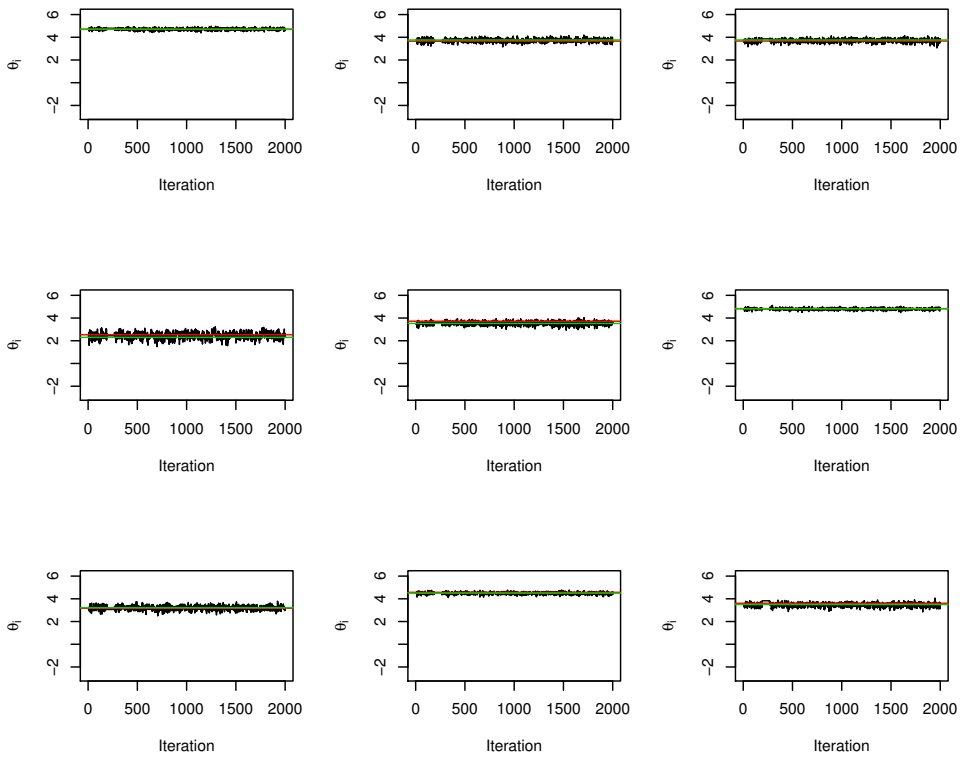

Figure 6: Chains of length 2000 obtained for 9 variables θ_is (among the 100 θ_is) with the semi-separable HMC algorithm designed for sampling in the posterior distribution of latent variables and parameters incorporated in a spatial GLMM. In each panel, the red line indicates the true value of θ_i and the green line indicates the observed value of $\log Y_i$ if $Y_i > 0$.

Algorithm 3 MCMC algorithm with Hamiltonian sampler for hierarchical models and non-separable Hamiltonian systems.

initialization: set a value for $\theta^{(0)}$ and $\beta^{(0)}$

for $k = 1, 2, \ldots$ **do** the following:

1: draw an auxiliary vector of variables $\binom{A}{B} \in \mathbb{R}^m$ with probability distribution $(A, B) \mapsto q(A, B \mid \theta^{(k-1)}, \beta^{(k-1)})$

2: solve the following Hamiltonian's equation at time $t = \tau$ (i.e. compute the state $z(\tau)$):

$$\frac{dz}{dt} = J \nabla H(z), \tag{4}$$

where the initial condition is $z(0) = (\theta^{(k-1)}, \beta^{(k-1)}, A, B)$, $z = (z_1, z_2) \in \mathbb{R}^m \times \mathbb{R}^m$, and

$$H(\theta, \beta, A, B) = -\log\{p(Y \mid \theta) p(\theta \mid \beta) p(\beta) q(A, B \mid \theta, \beta)\}$$
$$J = \begin{pmatrix} \mathbf{0}_{m,m} & \mathbf{I}_{m,m} \\ -\mathbf{I}_{m,m} & \mathbf{0}_{m,m} \end{pmatrix},$$

3: set $(\theta^*, \beta^*) = z_1(\tau)$ and $(A^*, B^*) = z_2(\tau)$

4: compute the acceptance probability α:

$$\alpha = \min\left\{1, \exp\{H((\theta^{(k-1)}, A^{(k-1)}, \beta^{(k-1)}, B^{(k-1)})) - H((\theta^*, A^*, \beta^*, B^*))\}\right\}$$

5: draw a uniform random variable U over $[0, 1]$ and update θ as follows:

if $U \leq \alpha$ **then**
set $\theta^{(k)} = \theta^*$ and $\beta^{(k)} = \beta^*$
else
set $\theta^{(k)} = \theta^{(k-1)}$ and $\beta^{(k)} = \beta^{(k-1)}$
end if

end for

Algorithm 4 Alternating block-wise leapfrog algorithm (ABLA) for solving, in the case of a hierarchical model, the Hamiltonian's equation (3) at time $t = \tau = \tilde{L}\tilde{\varepsilon}$ with initial condition $(\theta, \beta, A, B)(0)$. The positive integer $\tilde{L}$ and the positive real value $\tilde{\varepsilon}$ determine the accuracy of the resolution and the amplitude of the jump. Here, $\mathbf{leapfrog}(u, h, L, \varepsilon)$ designates Algorithm 2 solving with the leapfrog method the Hamiltonian dynamics corresponding to energy h, initialized at u, and tuned by L and ε.

for $i = 1, \ldots, \tilde{L}$ **do** the following:

set $t = i\tilde{\varepsilon}$

make a half step for (θ, A):

$$(\theta, A)(t - \tilde{\varepsilon}/2) = \mathbf{leapfrog}\{(\theta, A)(t - \tilde{\varepsilon}), (\theta, A) \mapsto H_1(\theta, \beta(t - \tilde{\varepsilon}), A, B(t - \tilde{\varepsilon})), L = 1, \varepsilon = \tilde{\varepsilon}/2\}$$

make a full step for (β, B):

$$(\beta, B)(t) = \mathbf{leapfrog}\{(\beta, B)(t - \tilde{\varepsilon}), (\beta, B) \mapsto H_2(\theta(t - \tilde{\varepsilon}/2), \beta(t - \tilde{\varepsilon}), A(t - \tilde{\varepsilon}/2), B(t - \tilde{\varepsilon})), L = 1, \varepsilon = \tilde{\varepsilon}\}$$

make a half step for (θ, A):

$$(\theta, A)(t) = \mathbf{leapfrog}\{(\theta, A)(t - \tilde{\varepsilon}/2), (\theta, A) \mapsto H_1(\theta, \beta(t), A, B(t)), L = 1, \varepsilon = \tilde{\varepsilon}/2\}.$$

end for

A Derivatives required in Section 5

The i-th component of $\nabla_\theta \log q_B(B;\theta)$ $(i = 1, \ldots, n)$ is:

$$\begin{aligned}
\frac{\partial}{\partial \theta_i} \log q_B(B;\theta) = &-\frac{1}{2}\mathrm{tr}\left(\Omega_B(\theta)^{-1}\frac{\partial \Omega_B(\theta)}{\partial \theta_i}\right) \\
&+\frac{1}{2}B'\Omega_B(\theta)^{-1}\frac{\partial \Omega_B(\theta)}{\partial \theta_i}\Omega_B(\theta)^{-1}B \\
\frac{\partial \Omega_B(\theta)}{\partial \theta_i} = &\ \mathbf{0}_{3,3}.
\end{aligned}$$

The gradient of $\log p(\theta \mid \beta)$ with respect to β satisfies:

$$\begin{aligned}
\nabla_\beta \log p(\theta \mid \beta) &= \begin{pmatrix} \partial \log p(\theta \mid \beta)/\partial \beta_1 \\ \partial \log p(\theta \mid \beta)/\partial \beta_2 \\ \partial \log p(\theta \mid \beta)/\partial \beta_3 \end{pmatrix} \\
\frac{\partial \log p(\theta \mid \beta)}{\partial \beta_1} &= (\theta - \beta_1 \mathbf{1}_n)'\Sigma(\beta_2,\beta_3)^{-1}\mathbf{1}_n \\
\frac{\partial \log p(\theta \mid \beta)}{\partial \beta_2} &= -\frac{1}{2}\mathrm{tr}\left(\Sigma(\beta_2,\beta_3)^{-1}\frac{\partial \Sigma(\beta_2,\beta_3)}{\partial \beta_2}\right) \\
&\quad +\frac{1}{2}(\theta - \beta_1\mathbf{1}_n)'\Sigma(\beta_2,\beta_3)^{-1}\frac{\partial \Sigma(\beta_2,\beta_3)}{\partial \beta_2}\Sigma(\beta_2,\beta_3)^{-1}(\theta - \beta_1\mathbf{1}_n) \\
\frac{\partial \log p(\theta \mid \beta)}{\partial \beta_3} &= -\frac{1}{2}\mathrm{tr}\left(\Sigma(\beta_2,\beta_3)^{-1}\frac{\partial \Sigma(\beta_2,\beta_3)}{\partial \beta_3}\right) \\
&\quad +\frac{1}{2}(\theta - \beta_1\mathbf{1}_n)'\Sigma(\beta_2,\beta_3)^{-1}\frac{\partial \Sigma(\beta_2,\beta_3)}{\partial \beta_3}\Sigma(\beta_2,\beta_3)^{-1}(\theta - \beta_1\mathbf{1}_n) \\
\frac{\partial \Sigma(\beta_2,\beta_3)}{\partial \beta_2} &= \frac{1}{\beta_2}\Sigma(\beta_2,\beta_3) = \exp(-\beta_3 D) \\
\frac{\partial \Sigma(\beta_2,\beta_3)}{\partial \beta_3} &= -D \circ \Sigma(\beta_2,\beta_3)
\end{aligned}$$

where D is the matrix of distances whose term $(i,j) \in \{1,\ldots,n\}^2$ is $||x_i - x_j||$, and $\circ$ denotes the Hadamard product (element-by-element multiplication).
The gradient of $\log p(\beta)$ with respect to β satisfies:

$$\nabla_\beta \log p(\beta) = \begin{pmatrix} -(\beta_1 - b_{11})/b_{12}^2 \\ -(\log \beta_2 - b_{21} + b_{22}^2)/(\beta_2 b_{22}^2) \\ -(\log \beta_3 - b_{31} + b_{32}^2)/(\beta_3 b_{32}^2) \end{pmatrix}$$

The i-th component of $\nabla_\beta \log q_A(A;\beta,x)$ $(i = 1,\ldots,3)$ is:

$$\begin{aligned}
\nabla_\beta \log q_A(A;\beta,x) = &-\frac{1}{2}\mathrm{tr}\left(\Omega_A(\beta;x)^{-1}\frac{\partial\Omega_A(\beta;x)}{\partial\beta_i}\right) \\
&+\frac{1}{2}A'\Omega_A(\beta;x)^{-1}\frac{\partial\Omega_A(\beta;x)}{\partial\beta_i}A \\
\frac{\partial\Omega_A(\beta;x)}{\partial\beta_1} &= \frac{\partial\Sigma(\beta_1,\beta_2)}{\partial\beta_1} = 0 \\
\frac{\partial\Omega_A(\beta;x)}{\partial\beta_2} &= \frac{\partial\Sigma(\beta_1,\beta_2)}{\partial\beta_2} \\
\frac{\partial\Omega_A(\beta;x)}{\partial\beta_3} &= \frac{\partial\Sigma(\beta_1,\beta_2)}{\partial\beta_3},
\end{aligned}$$

with $\partial\Sigma(\beta_1,\beta_2)/\partial\beta_3$ and $\partial\Sigma(\beta_1,\beta_2)/\partial\beta_3$ given above.

References

Bousset, L., S. Jumel, V. Garreta, H. Picault, and S. Soubeyrand (2015). Transmission of *Leptosphaeria maculans* from a cropping season to the following one. *Annals of Applied Biology 166*, 530–543.

Casella, G. and E. I. George (1992). Explaining the Gibbs sampler. *The American Statistician 46*, 167–174.

Chib, S. and E. Greenberg (1995). Understanding the Metropolis-Hastings algorithm. *The American Statistician 49*, 327–335.

Diggle, P. J., J. A. Tawn, and R. A. Moyeed (1998). Model-based geostatistics. *Journal of the Royal Statistical Society C 47*, 299–350.

Duane, S., A. D. Kennedy, B. J. Pendleton, and D. Roweth (1987). Hybrid Monte Carlo. *Physics letters B 195*, 216–222.

Girolami, M. and B. Calderhead (2011). Riemann manifold Langevin and Hamiltonian Monte Carlo methods. *Journal of the Royal Statistical Society B 73*, 123–214.

Neal, R. M. (2011). MCMC using Hamiltonian dynamics. In S. Brooks, A. Gelman, G. Jones, and X.-L. Meng (Eds.), *Handbook of Markov Chain Monte Carlo*, Chapter 5, pp. 113–162. Boca Raton: Chapman and Hall – CRC Press.

Zhang, Y. and C. Sutton (2014). Semi-separable Hamiltonian Monte Carlo for inference in Bayesian hierarchical models. In *Advances in Neural Information Processing Systems*, pp. 10–18.