Antithetic Acceleration of the Multiple-Try Metropolis

Radu V. Craiu*

Department of Statistics

University of Toronto

October 5, 2004

Abstract

The Multiple-Try Metropolis is a recent extension of the Metropolis algorithm in which the next state of the chain is selected among a set of proposals. We propose a modification of the Multiple-Try Metropolis algorithm which allows the use of correlated proposals, particularly antithetic proposals. The method is particularly useful for random walk Metropolis in high dimensional spaces and can be used easily when the proposal distribution is Gaussian. In the case of univariate proposals, the stratification induced by the Latin Hypercube sampling can be particularly efficient too. We investigate the performance of the algorithm in combination with a local search sampler, the random-ray Monte Carlo. A series of examples is presented to evaluate the potential of the method.

Key words and phrases: Antithetic variates, Markov chain Monte Carlo, Negative association, Extreme antithesis, Latin Hypercube sampling, Multiple-Try Metropolis, Random-Ray Monte Carlo.

*This work has been supported by an individual grant from the Natural Sciences and Engineering Research Council of Canada

1 Introduction

Is is well recognized that the Markov Chain Monte Carlo (MCMC) methods provide huge support for realistic statistical modelling. In recent years, as the statistical models have increased in complexity and size, there is a greater demand for fast MCMC algorithms and more reliable convergence diagnostics. A promising direction is represented by the so-called local search samplers and adaptive algorithms. For examples, we refer to the papers by Atchade and Perron (2003), Atchade and Rosenthal (2003), Liu, Liang and Wong (2000), Draper and Liu (2003), Gilks, Roberts and George (1994), Chen and Schmeiser (1993). In this paper we discuss possible accelerations of the Multiple-Try Metropolis (MTM) of Liu, Liang and Wong (2000, henceforth denoted LLW) via antithetic variates. LLW introduce the MTM as a generalization of the classical Metropolis algorithm which allows one to select at each update among multiple proposals. The main advantage of the MTM is that it explores a larger portion of the sample space resulting in better mixing and shorter running times. In addition, LLW propose the use of MTM with the Adaptive Direction Sampling of Gilks et al. (1994) as well as the hit-and-run algorithm (Chen and Schmeiser, 1993) and the griddy Gibbs sampler (Ritter and Tanner, 1992). The modifications we propose here for the MTM can be used directly in all of the above.

Recent approaches to the acceleration of MCMC algorithms have included the use of antithetic variates (Frigessi, Gåsemyr and Rue, 2000; Craiu and Meng, 2005). The antithetic coupling has proven to be particularly effective in MCMC algorithms with monotone kernels, such as Gibbs and slice samplers, or in perfect sampling processes in which the updating function is monotone in all arguments. However, one of the most used MCMC algorithms in practice, the Metropolis-Hastings, does not have the monotone properties required for an efficient antithetic coupling. We propose the use of antithetic variates for Metropolis algorithms via the MTM. The new algorithm, *Multiple Correlated-Try Metropolis (MCTM)* selects among correlated proposals instead of independent ones. Section 2 contains the general construction of MCTM. In Section 3 we describe some of the possible implementations with more emphasis on random generation via the multivariate Gaussian and the inverse c.d.f. transform. Section 4 contains two examples for which we compare the performances of MCTM and MTM.

2 Multiple correlated-try Metropolis

Consider $T(x;y)$ a Metropolis proposal transition rule so that one can define

$$w(x,y) = \pi(x)T(x;y)\lambda(x,y),$$

where $\lambda(x,y)$ is a nonnegative symmetric function that can be chosen by the user. The original MTM proposed by LLW has the following steps.

1. Draw k trial proposals $y_1, ..., y_k$ from $T(x;\cdot)$. Compute $w(y_j, x)$ for each j;

2. Select $Y = y$ among the k trials with probability proportional to $w(y_j, x)$.

3. Draw $x_1^*, ..., x_{k-1}^*$ variates from the distribution $T(y,\cdot)$ and let $x_k^* = x$;

4. Accept y with probability
$$r_g = min\left\{1, \frac{w(y_1,x)+...+w(y_k,x)}{w(x_1^*,y)+....+w(x_k^*,y)}\right\}.$$

While it is obvious that the proposals do not need to be independently generated, some care is required in implementing the MTM with correlated proposals, especially if we want to maintain the reversibility of the Markov chain. Consider the MTM algorithm in which the proposals are generated jointly from $\tilde{T}(x;\cdot)$ and are exchangeable. We assume that the marginal transition kernel is equal to the original existing kernel $T(x;y)$ that is used to generate independent trials, i.e.

$$\int \tilde{T}(x;y_1,\ldots,y_k)dy_1...dy_{i-1}dy_{i+1}\ldots dy_k = T(x;y_i),\ \forall\ 1 \le i \le k.$$

1. Draw k trial proposals $y_1, ..., y_k$ from $\tilde{T}(x; \cdot)$. Compute $w(y_j, x) = \pi(y_j)T(y_j; x)$ for each j;

2. Select $Y = y$ among the k trials with probability proportional to $w(y_j, x)$.

3. Draw $(x_1^*, ..., x_{k-1}^*)$ variates from the conditional transition kernel $\tilde{T}(y; \cdot | x_k = x)$ and let $x_k^* = x$;

4. Accept y with probability

$$r_g = min\left\{1, \frac{w(y_1, x) + ... + w(y_k, x)}{w(x_1^*, y) + ... + w(x_k^*, y)}\right\}.$$

Proposition 2.1 *The Markov chain defined with the algorithm above has stationary distribution π and satisfies the detailed balance condition.*

Proof. The proof follows closely from the one given by LLW for the original MTM. If $A(x, y)$ is the actual transition probability and $I(\cdot)$ is the indicator function that shows which y_j has been selected at step 2., then

$$\begin{aligned}
\pi(x)A(x,y) &= \pi(x)P[\cup_{j=1}^k \{Y_j = y\} \cap \{I = j\} | x] = k\pi(x)P[\{Y_1 = y\} \cap \{I = 1\} | x] \\
&= k\pi(x) \int \tilde{T}(x; y, y_2, \ldots, y_k) \frac{w(y,x)}{w(y,x) + \sum_{i=2}^k w(y_i, x)} \times \\
&\times \ \min\left\{1, \frac{w(y,x) + \sum_{i=2}^k w(y_i, x)}{w(x,y) + \sum_{i=2}^k w(x_i^*, y)}\right\} \tilde{T}(y; x_2^*, \ldots, x_k^* | x) dy_2 \ldots dy_k dx_2^* \ldots dx_k^* \\
&= k \frac{w(y,x)w(x,y)}{\lambda(y,x)} \int \min\left\{\frac{1}{w(y,x) + \sum_{i=2}^k w(y_i, x)}, \frac{1}{w(x,y) + \sum_{i=2}^k w(x_i^*, y)}\right\} \times \\
&\times \ \tilde{T}(y; x_2^*, \ldots, x_k^* | x) \tilde{T}(x; y_2, \ldots, y_k | y) dy_2 \ldots dy_k dx_2^* \ldots dx_k^* = \pi(y)A(y,x).
\end{aligned}$$

In the above derivation we have used $\tilde{T}(x; y, y_2, \ldots, y_k) = T(x; y)\tilde{T}(x; y_2, \ldots, y_k | y)$.

3 Correlated proposals

We consider first the situation in which the proposals are exchangeable univariate random variables, say $Y_1, \ldots, Y_k$ with distribution function F. Without loss of generality we can assume that $E[Y_i] = 0$. Intuitively, we would like the proposals to be "well distributed" in the sample space. There is not a single comprehensive mathematical definition of what we mean by "well distributed" but two possible approaches can be outlined. First, one could consider proposals that are, on average, as further away from one another as possible with respect to a particular distance. If we consider the Euclidian distance $d(Y_i, Y_j) = \sqrt{(Y_i - Y_j)^2}$, then we need to consider the pairwise correlation between proposals since $E[d^2(Y_i, Y_j)] = 2\sigma^2(1-\rho)$ where $\sigma^2 = \text{var}(Y_i)$ for all i and $\rho = \text{corr}(Y_i, Y_j)$ for all $i \neq j$. Therefore, the largest distance is achieved on average by proposals that are *extremely antithetic (EA)* (Craiu and Meng, 2005), i.e. they achieve the smallest possible pairwise correlation $\rho = \text{corr}(Y_i, Y_j)$, subject to the constraint that the random variables $Y_1, \ldots, Y_k$ are exchangeable and marginally distributed with distribution function F. However, having a larger distance between proposals is not always the most efficient implementation of the MTM. An alternative is to stratify the sample of proposals. In this case of interest is the *equidistribution* of the proposals in the sample space. In recent years the literature on Quasi Monte Carlo algorithms has explored a wide variety of methods for producing stratified samples that are equidistributed in the unit hypercube (e.g. L'Ecuyer and Lemieux, 2002).

3.1 Univariate proposals

The situation in which the transition kernel $T(x, y)$ relies on an inverse c.d.f. transform to generate univariate proposals is one in which a general MTM can be easily extended to MCTM. Suppose that $y_1, \ldots, y_k$ are such that $y_i = F^{-1}_{|x}(u_i)$ for all $1 \leq i \leq k$.

For situations in which the proposals are sampled from a univariate symmetric and unimodal distribution one could use the construction of Rüschendorf and Uckelmann (2000) to generate EA proposals. Without loss of generality we can assume that the center of symmetry for the proposal density, f, is the origin. If $f_Q(x) = -xf'(x)$ is also a Lebesgue density on $\mathbf{R}$ then, if $Q \sim f_Q$ and $U \sim \text{Uniform}(-1,1)$, $QU \sim f$. As long as we can generate $\{U_1, \ldots, U_k\}$, independent of Q, such that $U_i \sim \text{Uniform}(-1,1)$ and $\sum_{i=1}^k U_i = 0$, then $\{Y_1 = QU_1, \ldots, Y_k = QU_k\}$ are EA with respect to f because $\sum_{i=1}^k X_i = 0$.

An alternative approach can be devised using Gaussian EA random variables instead of uniforms. Denote $\mathcal{L}_f(t) = \int_0^\infty e^{-tu} f(u)du$ the Laplace transform of the density f in t. If we define a density g on $(0,\infty)$ using

$$g(w) = \mathcal{L}^{-1}[\sqrt{2\pi} f(\sqrt{t})](w^2/2), \tag{3.1}$$

where $\mathcal{L}^{-1}$ is the inverse Laplace transform, then the following holds.

Proposition 3.1 *Sample W from g defined in (3.1) and define, for each $1 \le i \le k$, $Y_i = X_i/W$ where marginally $X_i \sim N(0,1)$ and $\sum_{i=1}^k X_i = 0$. Then $\sum_{i=1}^k Y_i = 0$ and marginally each Y_i is distributed with density f.*

Proof The result follows from

$$P(Y_i \le t) = P(X_i \le Wt) = \int_0^\infty \Phi(wt)g(w)dw,$$

because then the density of Y_i is

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}P(Y_i \le t) &= \int_0^\infty \phi(wt)wg(w)dw = \qquad (3.2) \\ &= \int_0^\infty \frac{1}{\sqrt{2\pi}} \int_0^\infty e^{-\xi t^2} g(\sqrt{2\xi})d\xi. \qquad (3.3) \end{aligned}$$

In turn, this implies that

$$\frac{\mathrm{d}}{\mathrm{d}t}P(Y_i \le t) = \frac{1}{\sqrt{2\pi}}\mathcal{L}_{\tilde{g}}(t^2), \tag{3.4}$$

where $\tilde{g}(t) = g(\sqrt{t})$. Replacing (3.1) in (3.4) we obtain that the marginal density of Y_i is f for all $1 \le i \le k$.

Alternatively, as discussed at the beginning of this section, one may choose to produce a stratified sample of proposals. While there are many ways to do so, here we discuss the use of Latin hypercube sampling (LHS) introduced by McKay, Beckman and Conover (1979) (see also Stein, 1987; Owen, 1992; Loh, 1996; Craiu and Meng, 2005). The LHS method involves the following three steps.

Step I Generate independently $v_1, \ldots v_k \sim \text{Uniform}(0,1)$.

Step II Select a random permutation τ of $\{0, \ldots, k-1\}$

Step III Construct $u_i = (v_i + \tau(i))/k$, for all $1 \le i \le k$.

It can be noticed that the LHS adds little computational overhead when compared to the independent generation of samples. In addition, there is no requirement for a symmetric distribution of the proposals.

The MCTM can be implemented using the following steps.

1. Draw k proposals using the uniform deviates $u_1 \ldots, u_k$ constructed via the LHS algorithm using permutation τ.

2. Assuming that $y = y_{j_0}$ is selected, generate $x_i^* = F_y^{-1}(u_i^*)$ where the u_i^*'s are sampled using the LHS construction by ensuring that the balance condition is satisfied. More precisely, take $j_0 = [k * F_y(x)]$ (where $[u]$ is the integer part of u) and for all $0 \le j \le k-1$, $j \ne j_0$ construct $u_j^* = (\tau(j) + w_j)/k$ where the $w_j \sim \text{Uniform}(0,1)$ are independent.

3. For each $j \ne j_0$, $x_j^* = F_y^{-1}(u_j^*)$ and $x_{j_0}^* = x$.

Incidentally, one can see that LHS produces uniform Uniform$(-1,1)$ deviates $U_1, \ldots, U_k$ that satisfy $\mathrm{E}(\sum_{i=1}^k U_i) = 0$ instead of $\sum_{i=1}^k U_i = 0$.

3.2 Multivariate Gaussian proposals

In many instances in which the Metropolis algorithm is used to sample from a multivariate distribution, the Gaussian distribution is used to generate the proposals. Proposition 1 is particularly attractive in the normal case as the conditional kernel is easy to compute and to sample from. In local-search algorithms the goal is to explore larger portions of the sample space while, in the case of a Metropolis algorithm, not diminishing significantly the acceptance rate.

In the following we consider the case in which we want to maximize the average inter-proposal Euclidian distance. Consider a r-dimensional sample space for the Markov chain X_t constructed via MTM with multivariate Gaussian proposals. More specifically, the original MTM algorithm generates k proposals from $N_r(\tilde{x}, \Sigma)$ whenever the current state is $\tilde{x}$. A general version of the original MTM uses at each step k proposals which are jointly normal from $N_{kr}(\tilde{x_k}, \Sigma_{kr})$. To simplify the notation we assume that $r = 2$ but the discussion is true in general.

If the independent proposals are sampled from $N_2\left((x,y)^T, \Sigma\right)$, then a pair of correlated proposals is

$$(x_1, y_1, x_2, y_2)^T \sim N_4\left((x,y,x,y)^T, \begin{pmatrix} \Sigma & \Psi \\ \Psi & \Sigma \end{pmatrix}\right),$$

where $\Sigma = \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix}$ and $\Psi = \begin{pmatrix} \rho_1\sigma_1^2 & \rho_2\sigma_1\sigma_2 \\ \rho_2\sigma_1\sigma_2 & \rho_1\sigma_2^2 \end{pmatrix}$. We seek a correlation structure, as determined by (ρ_1, ρ_2), so that the average Euclidian distance between proposals is maximized. It can be assumed without loss of generality that Σ is diagonal, say $\Sigma = diag(\sigma_1^2, \sigma_2^2)$. Otherwise one can apply an orthogonal transformation $(x_1', y_1')^T = C(x_1, y_1)^T$ and $(x_2', y_2')^T = C(x_2, y_2)^T$ so that, if $d((x,y),(x',y')) = \sqrt{(x-x')^2 + (y-y')^2}$, then $d((x_1', y_1'), (x_2', y_2')) = d((x_1, y_1), (x_2, y_2))$ and $x_i' \perp y_i'$. The marginal distribution of

$$\begin{pmatrix} x_1 - x_2 \\ y_1 - y_2 \end{pmatrix} \sim N\left((0,0)^T, \begin{pmatrix} 2\sigma_1^2 - 2\rho_1\sigma_1^2 & -2\rho_2\sigma_1\sigma_2 \\ -2\rho_2\sigma_1\sigma_2 & 2\sigma_2^2 - 2\rho_1\sigma_2^2 \end{pmatrix}\right).$$

Therefore

$$E[d((x_1, y_1), (x_2, y_2))] = 2(\sigma_1^2 + \sigma_2^2)(1 - \rho_1)$$

is maximized when ρ_1 is equal to its smallest possible value. In general it is more efficient to take $\rho_2 = 0$. Therefore, for the MCTM with Gaussian proposals, $y_i \sim N_r(\tilde{x}, \Sigma)$, one can use $(y_1^T, \ldots, y_k^T)^T \sim N((\tilde{x}^T, \tilde{x}^T, \ldots, \tilde{x}^T)^T, \Sigma_{kr})$ with

$$\Sigma_{kr} = \begin{pmatrix} \Sigma & \Psi & \ldots & \Psi \\ \Psi & \Sigma & \Psi & \Psi \\ \ldots & \ldots & \ldots & \ldots \\ \Psi & \Psi & \Psi & \Sigma \end{pmatrix}$$

where $\Psi = diag(\rho\sigma_1^2, \ldots, \rho\sigma_k^2) \in R^{r\times r}$ and $\rho = -1/(r-1)$. The lower bound $\rho = -\frac{1}{k-1}$ is obtained from the constraint that the joint correlation matrix of *all* the proposals, Σ_{kr}, is a positive definite matrix.

4 Examples

4.1 Logit regression

We apply our method to a multi-dimensional real-data application. The data are taken from van Dyk and Meng (2001) and consist of measurements on 55 patients of which 18 have been diagnosed with latent membranous lupus. Table 1 shows the data with two clinical covariates, IgA and IgG, that measure the levels of immunoglobulin of type A and of type

Table 1: *The number of latent membranous lupus nephritis cases, the numerator, and the total number of cases, the denominator, for each combination of the values of the two covariates.*

	IgA				
IgG3-IgG4	0	0.5	1	1.5	2
-3.0	0/ 1	-	-	-	-
-2.5	0/ 3	-	-	-	-
-2.0	0/ 7	-	-	-	0/ 1
-1.5	0/ 6	0/ 1	-	-	-
-1.0	0/ 6	0/ 1	0/ 1	-	0/ 1
-0.5	0/ 4	-	-	1/ 1	-
0	0/ 3	-	0/ 1	1/ 1	-
0.5	3/ 4	-	1/ 1	1/ 1	1/ 1
1.0	1/ 1	-	1/ 1	1/ 1	4/ 4
1.5	1/ 1	-	-	2/ 2	-

G, respectively. Of interest is the prediction of disease occurrence using the two covariates $IgG3 - IgG4$ and IgA. We consider a logit regression model in which

$$logit\ P(Y_i = 1) = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i}$$

where $X_i^T = (1, X_{i1}, X_{2i})$ is the vector of covariates for the i-th individual. We follow Tan (2003) and consider that the prior for $\beta = (\beta_0, \beta_1, \beta_2)^T$ is trivariate normal with zero mean and variance diag$(100^2, 100^2, 100^2)$. The posterior density is then proportional to

$$\pi(\beta|x, y) \propto \prod_{j=0}^{2} \frac{e^{\beta_j/100^2}}{\sqrt{1002\pi}} \prod_{i=1}^{55} \left[\frac{exp(X_i^T\beta)}{1 + exp(X_i^T\beta)}\right]^{y_i} \left[\frac{1}{1 + exp(X_i^T\beta)}\right]^{1-y_i}$$

The random walk Metropolis is used with multiple proposals, antithetic and independent, to generate 1000 replicates of samples of size 1000. The proposal $T(\cdot|\beta)$ is trivariate normal with mean β and variance $\Sigma = diag(\sigma^2, \sigma^2, \sigma^2)$.

In Figure 1 we compare the autocorrelation plots obtained for $\sigma = 2, 3, 4$ and $k = 3, 4, 5, 6$ in the case of independent and antithetic proposals. It can be seen that the autocorrelation between samples is smaller when antithetic proposals are used. The graphical evidence is confirmed by empirical calculations of the efficiency of the Monte Carlo estimators in the independent and antithetic cases.

In Table 2 we report, for β_1 and $p_{25} = 1_{\{\beta_1 > 25\}}$ the ratio $R = \frac{\text{MSE}_{\text{anti}}}{\text{MSE}_{\text{ind}}}$ where MSE represents the Monte Carlo mean squared error and the index refers to the method of generating the proposals, i.e. independently or antithetically. To be more specific, we replicated $M = 1000$ samples, each of $N = 1000$ draws. If we denote by b_{ij} the j-th sample point drawn in the i-th replicate from the posterior distribution of β_1 then, using $\bar{b}_{\cdot\cdot} = \frac{\sum_{ij} b_{ij}}{MN}$ and $\bar{b}_{i\cdot} = \frac{\sum_j b_{ij}}{N}$ for all $i = 1, \ldots, M$ the MSE is defined as

$$\text{MSE} = (\bar{b}_{\cdot\cdot} - E[\beta_1|\text{data}])^2 + \frac{\sum_i (\bar{b}_{i\cdot} - \bar{b}_{\cdot\cdot})^2}{(M-1)}.$$

Similar calculations can be done for p_{25}. Numerical integration yields $E[\beta_1|\text{data}] \approx 13.57$ and $E[p_{25}|\text{data}] \approx 0.073$ (see Tan, 2004). It is seen that the use of antithetic proposals is more effective when the number of streams is larger ($k = 6$) and when the variance of the proposal distribution is such that probability of acceptance is close to the optimal (see Roberts and Rosenthal, 2001) 0.24. In such cases increases in efficiency are as high as 30% (R=.70) for β_1 and 35% ($R = 0.65$) for p_{25}. On the root scale, this corresponds to savings between 20-25%. In none of the situations explored has the use of antithetic proposals been inflating the MSE.

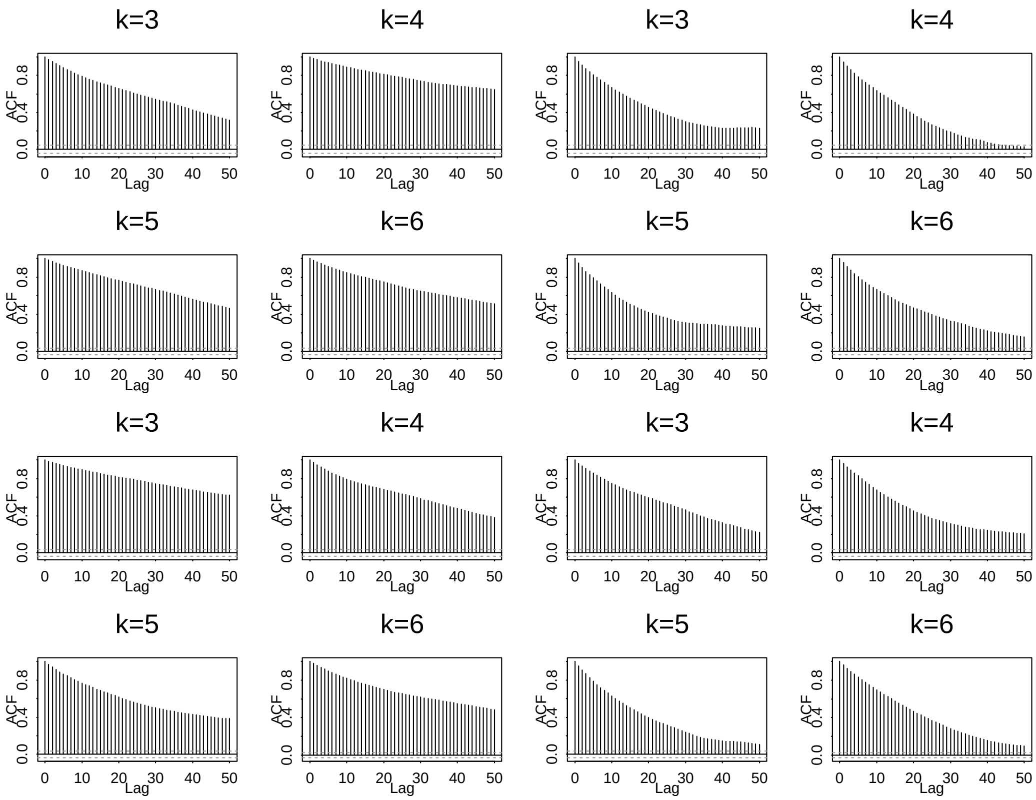

Figure 1: Autocorrelation plots for the logit regression model. First two rows correspond to $\sigma = 3$ and last two rows to $\sigma = 4$. First two columns on the left represent samples obtained with independent proposals and the last two columns on the right represent samples obtained with antithetic proposals.

Table 2: Values of R for β_1/p_{25} in the logit example

$k\backslash\sigma$	2	3	4
3	0.90/0.79	0.79/0.72	0.74/0.69
4	0.88/0.71	0.77/0.72	0.81/0.72
5	0.81/0.67	0.79/0.69	0.72/0.66
6	0.71/0.64	0.67/0.66	0.71/0.64

4.2 MCTM for local search MCMC

LLW have showed that the MTM algorithm can be embedded in local search algorithms such as the random-ray Monte Carlo (Liu, Liang and Wong, 2000), the hit-and-run algorithm (Chen and Schmeiser, 1993) or the Adaptive Direction Sampling algorithm (Gilks, Roberts and George, 1994). The random-ray Monte Carlo is a modified form of the hit-and-run algorithm and is especially effective when the distribution of interest is multimodal and the modes are aligned on a direction which is not parallel to any of the coordinate axes. We consider here one target density from a bimodal family of bivariate distributions constructed by Gelman and Meng (1991). More precisely, the density

$$f(x,y) \propto \exp\{-(9x^2y^2 + x^2 + y^2 - 8x - 8y)/2\} \tag{4.1}$$

has the property that the two conditional densities $f(x|y)$ and $f(y|x)$ are normal but the joint density is not normal. A three-dimensional plot of the density $f(x,y)$ is shown in Figure 2.

The construction of the random-ray Monte Carlo via MTM has been detailed by LLW and is followed here. More precisely, at each iteration t of the algorithm, a random direction, say $\mathbf{e}$, is generated and then, along direction $\mathbf{e}$, the proposals $y_1, \dots, y_k$ are generated from

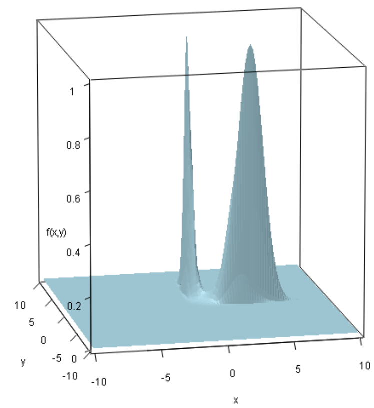

Figure 2: Random-ray Monte Carlo. The bivariate density $f(x, y)$

the distribution $T_{\mathbf{e}}(x, \cdot)$ where x is the state of the chain at time t. The proposals are generated using $y_i = x + r_i\mathbf{e}$ where $r_1, \ldots, r_k$ are sampled from Uniform$[-\sigma, \sigma]$. In our implementation, we use MCTM to select among k antithetic variates r_i. The parameters chosen here are $\sigma \in \{3, 4, 5\}$ and $k \in \{3, 4, 5, 6\}$. Due to the stratification induced by the hypercube sampling the MCTM has a higher acceptance rate and thus mixes better than the original MTM.

Table 3 offers support to the previous observations. As in the previous example we look at the Monte Carlo MSE reduction factor, R, for different choices of σ and k. In each case we perform 1000 updates with each algorithm and we replicate the analysis 500 times. The starting points are the same for the two algorithms. The numbers reported in each cell represent the estimates of R. The true marginal mean of X is approximately equal to 1.83. Unlike the previous example, the acceptance rates are different for the MTM and the MCTM so are also reported in Table 4. One can notice that the largest increase in efficiency, 70% (45% on the root scale) occurs for the combination of k, σ that results in an acceptance probability of 0.24. As before, the MCTM produces samples with smaller autocorrelation, as seen in Figure 3. Figure 4 offers additional evidence that the use of antithetic proposals increases the mobility of the chain. The left two columns represent the samples obtained after 2000 iterations of the original MTM and the right two columns of plots show the same number of samples generated using the MCTM. The contour lines represent the true density $f(x, y)$. It is seen that the MCTM samples offer a more representative image of the support of f.

Table 3: Values of the MSE reduction factor $R = \frac{\text{MSE}_{\text{anti}}}{\text{MSE}_{\text{ind}}}$

$\sigma \backslash k$	3	4	5	6
3	0.35	0.53	0.64	0.81
4	0.31	0.42	0.58	0.76
5	0.29	0.40	0.49	0.62

Table 4: Probability of acceptance for MTM/MCTM

$\sigma \backslash k$	3	4	5	6
3	26.5/46.1	31.2/47.8	35.2/50.3	38.7/49.7
4	24.5/40.9	26.6/41.8	29.8/44.5	32.3/46.2
5	18.8/35.4	22.7/37.6	26.2/40.3	29.4/42.4

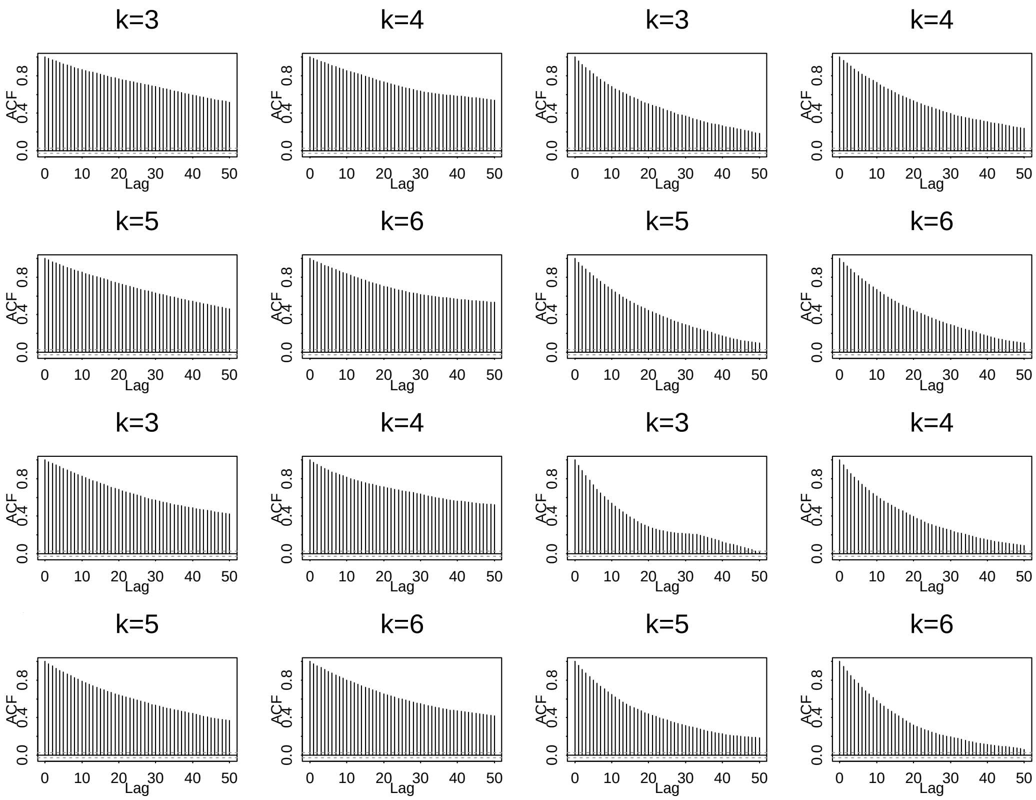

Figure 3: Autocorrelation plots for the bimodal example. First two rows correspond to $\sigma = 4$ and last two rows to $\sigma = 5$. First two columns on the left represent samples obtained with independent proposals and the last two columns on the right represent samples obtained with antithetic proposals.

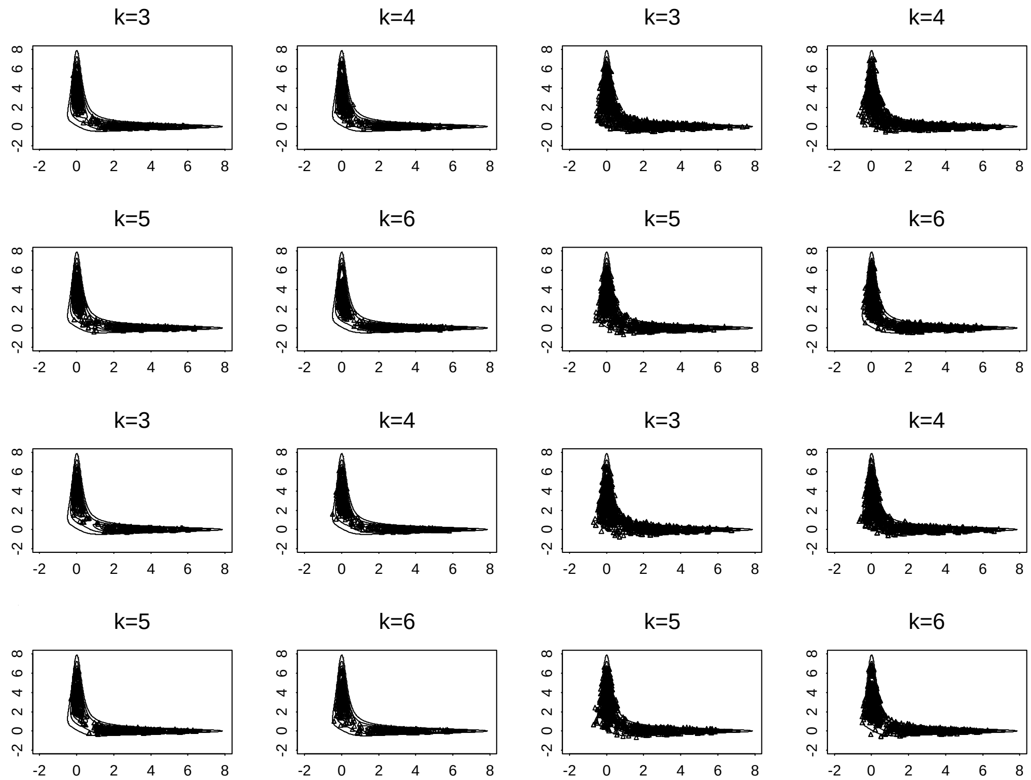

Figure 4: Coverage of the MTM and the MCTM. All plots are based on 2000 samples. First two rows correspond to $\sigma = 4$ and last two rows to $\sigma = 5$. First two columns on the left represent samples obtained with independent proposals and the last two columns on the right represent samples obtained with antithetic proposals.

5 Discussion and Further work

The MCTM algorithm requires small modifications once a MTM is designed. Both algorithms aim at speeding-up the traditional Metropolis-Hastings algorithms. Provided the acceptance rates of the two are comparable, the MCTM is more efficient with improvements of the root mean squared error as high as 45%. Further research is necessary to understand possible relations between the acceptance rate of MTM and the increase in efficiency brought by MCTM.

While it is well known that in order to use efficiently the Metropolis-Hastings algorithm one has to strike a balance between the a reasonable acceptance rate and a good mixing of the algorithm, the problem of optimal scaling (Roberts, Gelman and Gilks, 1997; Roberts and Rosenthal, 2001) is more complex in the case in which the proposals are not independent and requires further investigation.

References

Atchade, Y. F. and Perron, F. (2002). Improving on the Independent Metropolis-Hastings Algorithm. *Technical Report*, Department of Mathematics and Statistics, University of Montreal.

Atchade, Y. F. and Rosenthal, J. (2003). On Adaptive Markov Chain Monte Carlo Algorithms. *Technical Report*, Department of Statistics, University of Toronto.

Chen, M.-H. and Schmeiser, B.W. (1993). Performances of the Gibbs, hit-and-run, and Metropolis samplers. *J. Computat. Graph. Statist.*, **2**, 251–272.

Craiu, R. V. and Meng, X. L. (2005). Multi-Process Parallel Antithetic coupling for Backward and Forward MCMC. *Ann. Statist.*, to appear.

Draper, D. and Liu, S. (2003). Strategies for MCMC acceleration. *Technical Report*, Department of Applied Mathematics and Statistics, University of California, Santa Cruz.

Frigessi, A., Gåsemyr, J. and Rue, H. (2000). Antithetic coupling of two Gibbs sampler chains. *Ann. Statist.*, **28**, 1128–1149.

Gelman, A. and Meng, X. L. (1991). A note on bivariate distributions that are conditionally normal. *Am. Statistician*, 125–6.

Gilks, W. R., Roberts, R. O. and George, E. I. (1994). Adaptive Direction Sampling. *Statistician*, **43**, 179–189.

Hammersley, D. C. and Morton, K. V. (1956). A new Monte Carlo technique: Antithetic Variates. *Math. Proc. Cambridge Philos. Soc.*, **52**, 449–475.

Joag-Dev, K. and Proschan, F. (1983). Negative Association of Random Variables with Applications. *Ann. Statist.*, **11**, 286–295.

L'Ecuyer, P. and Lemieux, C. (2002). Recent advances in randomized quasi-Monte Carlo methods. In M. Dror, P. L'Ecuyer, and F. Szidarovszki, editors *Modelling Uncertainty: An Examination of Stochastic Theory, Methods, and Applications*, pages 419-474. Kluwer Academic Publishers, Boston, 2002.

Liu, J. S., Liang, F. and Wong, W. H. (2000). The use of Multiple-Try Method and local optimization in Metropolis sampling. *J. Amer. Statist. Assoc.*, **95**, 121–134.

Loh, W. L. (1996). On Latin Hypercube Sampling. *Ann. Statist.*, **24**, 2058–2080.

McKay, M. D., Beckman, R. J. and Conover, W. J. (1979). A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output From a Computer Code. *Technometrics*, **21**, 239–245.

Owen, A. B. (1992). A Central Limit Theorem for Latin Hypercube Sampling. *J. Roy. Statist. Soc. Ser. B*, **54**, 541–551.

Owen, A. B. (2003). Quasi Monte Carlo. *Presented at the First Cape Cod Workshop on Monte Carlo Methods.*

Ritter, C. and Tanner, M. A. (1992). Facilitating the Gibbs Sampler: the Gibbs Stopper and the Griddy-Gibbs Sampler. *J. Amer. Statist. Assoc.*, **87**, 861-868.

Robert, C. P. (1995). *Méthodes de Monte Carlo par Chaînes de Markov.* Economica, Paris.

Roberts, G. O., Gelman, A. and Gilks, W. (1997). Weak convergence and optimal scaling of random walk Metropolis algorithm. *Ann. Appl. Probab.*, **7**, 110–120.

Roberts, G. O. and Rosenthal, J. S. (2001). Optimal scaling of various Metropolis-Hastings algorithms. *Statist. Sci.*, **16**, 351–367.

Rüschendorf, L. and Uckelmann, L. (2000). Variance minimization and random variables with constant sum. *Technical Report*, Institüt für Mathematische Stochastik, University of Freiburg.

Stein, M. (1987). Large Sample Properties of Simulations Using Latin Hypercube Sampling. *Technometrics*, **29**, 143–151, (Correction: **32**, 367).

Tan, Z. (2003). Monte Carlo Integration with Acceptance-Rejection. *Dept. of Biostatistics Working Papers*, Johns Hopkins University,

van Dyk, D. and Meng, X. L. (2001). The art of data augmentation (with discussion). *J. Comput. Graphical Statist.*, **10**, 1–111.