A GENERIC APPROACH TO POSTERIOR INTEGRATION AND GIBBS SAMPLING

by

Peter Müller
Purdue University

Technical Report # 91-09

Department of Statistics
Purdue University

February, 1991

△

A GENERIC APPROACH TO POSTERIOR INTEGRATION AND GIBBS SAMPLING

by

Peter Müller*
Purdue University

ABSTRACT

This paper proposes a generic algorithm to generate a posterior Monte Carlo sample, based on the Metropolis algorithm. The algorithm does not depend upon any approximation or envelope function for the posterior density and is therefore ideal as general purpose "black box" algorithm. In particular, the algorithm is robust with respect to it's initialization. However, available information about the posterior can be used to shortcut convergence. Convergence in total variation and an ergodic theorem are shown.

Applying the proposed scheme to generate from the conditional distributions required for the Gibbs sampler extends the applicability of the Gibbs sampling scheme to problems without conjugate structure and makes orthogonalization possible. Orthogonalization improves the convergence of the Gibbs sampler by reducing serial correlation.

Keywords: Metropolis algorithm, Bayesian sampling, stochastic substitution, numerical integration.

---

*This research was supported by NSF grants DMS-8717799 and DMS-8702620 at Purdue University.

1 Introduction

Despite the conceptual appeal of Bayesian analysis, the implementation of Bayesian methods faces a major obstacle in the requirement to evaluate posterior integrals which are often analytically intractable and even difficult to solve numerically. Suppose X having density $p(\boldsymbol{x}|\theta)$ is observed, with θ being an unknown element of the parameter space $\Theta \subset \Re^p$. Bayes' theorem relates the prior density $\pi(\theta)$ to the posterior density by $p(\theta|\boldsymbol{x}) \propto \pi(\theta)p(\boldsymbol{x}|\theta)$. Most Bayesian inference can then be done in terms of integrals of some function $f(\theta)$ with respect to the posterior density $p(\theta|\boldsymbol{x})$:

$$E_p f(\theta) = \int_\Theta f(\theta)p(\theta|\boldsymbol{x})d\theta, \tag{1}$$

where the specification $f(\theta) = \theta_i$ leads to point estimates, $f(\theta) = p(\boldsymbol{x}_0|\theta)$ to the predictive density at $\boldsymbol{x} = \boldsymbol{x}_0$ etc. Clearly these posterior integrals are analytically solvable only in special cases. Extending the notion of conjugacy in a pragmatic way we will refer to such situations as "conjugate" models.

Numerical integration algorithms, suggested specifically for posterior integration, include Monte Carlo integration with importance sampling (Geweke 1989), Laplace's method (Tierney and Kadane 1986), Gaussian quadrature (Naylor and Smith 1982), Tanner and Wong's data augmentation algorithm (Tanner and Wong 1987) and the Gibbs sampler (Gelfand and Smith 1990). Importance sampling, Laplace's method and Gaussian quadrature are essentially based upon availability of an appropriate approximation respectively envelope function for the true posterior. The Gibbs sampler and the similar data augmentation scheme of Tanner and Wong require that conditional versions of the full joint posterior be available, meaning that random samples can

be drawn efficiently from these conditional distributions. Carlin and Gelfand (1990) suggested a procedure combining these two general strategies by using an approximation of the joint posterior as envelope function for accept/reject schemes to generate from the conditional distributions.

In this paper an alternative way of implementing Bayesian methods is suggested. A posterior Monte Carlo sample is generated by an implementation of the Metropolis algorithm (Metropolis, Rosenbluth, Rosenbluth, Teller and Teller 1953 and Hastings 1970). A once generated posterior sample can then be used for virtually any posterior inference. In Section 2 the algorithm is stated. Implementation details and convergence results are discussed in Sections 2.2 and 2.3.

The algorithm is ideally suited to generate from the complete conditional posterior distributions required in the Gibbs sampler. Using the Metropolis scheme in the Gibbs sampler achieves three major goals. In it's basic formulation the Gibbs sampler is restricted to problems where the complete conditional parts of the posterior distribution are "available", meaning that efficient random variate generation is possible from the conditionals. The generality of the Metropolis algorithm removes this restriction. Secondly, using the Metropolis scheme to generate form the conditionals, the Gibbs sampler is not restricted any more to the original parametrization and orthogonalization becomes possible. Orthogonalization reduces serial correlation of subsequent states in the Gibbs sampler and thereby improves convergence. Third, convergence of the Gibbs sampler can be improved by generating from higher dimensional conditional versions of the posterior rather than iterating over only one dimensional complete conditionals, i.e. generating several parameters at a time, rather

than only one at each step. These ideas are discussed in Section 3.

Section 4 contains some application examples. A comparison of the computational effort involved in using the proposed algorithm versus using importance sampling is given in Section 5.

2 The Algorithm

2.1 Statement of the algorithm

A Monte Carlo sample from the posterior $p(\theta) = p(\theta|x)$ is generated by simulating a Markov chain which has $p(\theta)$ as it's limiting distribution. The Monte Carlo sample can then be used for posterior inference. The algorithm is an implementation of the Metropolis algorithm (Metropolis et. al 1953). Implementation details, including the choice of $\tilde{\theta}$, $\tilde{\Sigma}$, the candidate generating density $g(y|\theta)$ and specific estimators, are given in Section 2.2.

ALGORITHM 1: METROPOLIS ALGORITHM.

Start with $\theta^{(0)} \sim N(\tilde{\theta}, \tilde{\Sigma})$.

- Generate a candidate point y from the conditional density $g(y|\theta^{(0)})$.
- With probability $a(\theta^{(0)}, y) = \min(1, p(y)/p(\theta^{(0)}))$, move to y, i.e. $\theta^{(1)} := y$, otherwise $\theta^{(1)} := \theta^{(0)}$.

Repeat the last two steps to generate $\theta^{(2)}, \ldots, \theta^{(M_0)}, \ldots \theta^{(M_1)}, \ldots \theta^{(M_2)}$.

Running this scheme with a sample of n initial points $\{\theta_1^{(0)}, \ldots \theta_n^{(0)}\}$ in parallel generates after sufficiently many iterations an approximate posterior sample.

The jumping probabilities $a(\theta, y)$ only determine the transition probabil-

ities in the Markov chain, and should not be mistaken for some kind of accept/reject weights.

The same Markov chain is used in simulated annealing algorithms to generate from $p(\theta) \propto \exp(-C(\theta)/T)$, where $C(\theta)$ is a function which is to be minimized and T is a parameter which is slowly reduced until, for T close to zero, the p.d.f. $p(\theta)$ is tightly concentrated around the global minimum of $C(\theta)$. See, e.g., van Laarhoven and Aarts (1987) for a discussion of simulating annealing algorithms.

The motivation for using the Metropolis algorithm to generate a Monte Carlo posterior sample is it's generality. The scheme does not require any assumptions on the posterior density, such as normality, unimodality etc. (see also Section 2.2.1).

2.2 Implementation

In the following description of implementation details the constants n, d, M_0, a and n_2 are free control parameters. In the examples reported in Section 4 they are chosen as $n = 10$, $d = 10$, $M_0 = 200$, $a = 2$ and $n_2 = 100$.

Let $\bar{\theta}^{(m)}$ denote the sample average over all n parallel chains at iteration m, and let $\hat{\theta}^{(m)}$ denote the sample average over all n parallel chains and all iterations up to iteration m, excluding M_0 initial iterations:

$$\bar{\theta}^{(m)} = \sum_{i=1}^{n} \theta_i^{(m)}/n, \tag{2}$$

$$\hat{\theta}^{(m)} = \sum_{j=1}^{k} \bar{\theta}^{(M_0+jd)}/k, \tag{3}$$

where $m = M_0 + k \cdot d$. To reduce serial correlation only every $d-$th iteration is used in the sample average over all iterations.

2.2.1 Initialization. The initial sample is taken from a normal approximation of the posterior: $\theta_i^{(0)} \sim N(\tilde{\theta}, \tilde{\Sigma})$, $i = 1, \ldots n$, where $\tilde{\theta}$ and $\tilde{\Sigma}$ are, e.g., posterior mode and negative inverse Hessian evaluated at $\tilde{\theta}$. Good estimates $\tilde{\theta}$ and $\tilde{\Sigma}$ improve convergence of the algorithm. But the scheme can be used even if the posterior mode is not available or – practically more important – if $\tilde{\Sigma}$ is only a poor estimate of the posterior covariance matrix or asymptotic posterior normality does not apply. The first few iterations of Algorithm 1 can work as a stochastic optimization scheme to move the sample points towards the mass of the posterior $p(\theta)$. This is illustrated in Example 1 in Section 4.

2.2.2 Candidate generating p.d.f. $g(y|\theta)$. The candidate generating conditional p.d.f. $g(y|\theta)$ has to be symmetric in its arguments, i.e. $g(y|x) = g(x|y)$ and will therefore from now on be simply denoted as $g(x, y)$. Theoretically any conditional density g which satisfies the condition of Theorem 1 (Section 2.3) could be chosen. In the examples reported in this paper we chose $g(\theta, y)$ $= g(\theta - y)$ as multivariate normal $N(0, R)$, with the covariance matrix R equal to a scalar multiple of a current estimate $\tilde{\Sigma}$ of the posterior covariance matrix, i.e. $R = c\tilde{\Sigma}$.

The scalar c is initially (and after each reestimation of $\tilde{\Sigma}$) set to $c := 1$. Whenever the average over the 10 most recently observed acceptance probabilities is greater than 0.8 then c is increased by a factor 1.2. If the average is less than 0.2 then c is decreased by a factor 0.7.

2.2.3 Updating $\tilde{\Sigma}$. Over the first M_0 iterations $\tilde{\Sigma}$ is updated a times, for $j = 1, \ldots a$, according to:

$$\tilde{\Sigma} = \frac{1}{kn} \sum_{l=(j-1)k+1}^{jk} \sum_{i=1}^{n} (\theta_i^{(ld)} - \tilde{\theta}_j)^2,$$

where $\tilde{\theta}_j = \sum_{l=(j-1)k+1}^{jk} \bar{\theta}^{(ld)}/k$ and $M_0 = akd$.

2.2.4 Assessing convergence. Available convergence results (Section 2.3) do not offer any help in developing a suitable method of assessing convergence. Therefore in the applications in Section 4 we followed a naive approach. We plotted the trajectories of $\bar{\theta}^{(m)}$ and $\hat{\theta}^{(m)}$ and considered the chains as satisfactory converged if these paths oscillated without any obvious trend within reasonable bounds around the estimated posterior mean ("reasonable" for the number of averaged terms). Clearly a more rigid and automated method would be desirable, but when combined with conservative judgement this simpleminded approach seems to work satisfactory in the examples we worked with.

The same problem, i.e. the lack of a formal, automated method to decide termination of the iteration process, occurs with the implementation of the Gibbs sampling scheme (see Section 3.1). Gelfand, Hills, Racine-Poon and Smith (1990) comment on this and suggest a heuristic approach similar to the one outlined above.

2.2.5 Posterior Inference. Assume the chains are considered to have satisfactory converged after M_1 iterations. Let $J = \int f(\theta) dp(\theta|x)$ be one of the requested posterior integrals. Then J is estimated by

$$\hat{f}(\theta)^{(M_1)} = \sum_{j=1}^{k} \bar{f}(\theta)^{(M_0+jd)}/k, \tag{4}$$

where $\bar{f}(\theta)^{(m)} = \sum_{i=1}^{n} f(\theta_i^{(m)})/n$ is the sample average at iteration m over all n parallel chains and k is the number of d-batches after the initial M_0 iterations, i.e. $M_1 = M_0 + kd$.

After M_1 iterations, increase the number of parallel chains to n_2 and go

through an additional $M_2 - M_1$ iterations to accumulate an approximate posterior sample:

$$X = \{\theta_i^{(M_1+jd)}, i = 1, \ldots n_2, j = 1, \ldots k_2\}, \tag{5}$$

where k_2 is the number of additional d-batches, i.e. $M_2 - M_1 = k_2 d$. The (approximate) posterior sample X is used to estimate densities and quantiles of marginal posterior densities.

2.2.6 Software. The applications reported in Sections 4 were estimated with an implementation of the algorithms as functions in new S (Becker, Chambers and Wilks 1988). We used XLISP-STAT (Tierney 1990) to find the posterior mode $\tilde{\theta}$ and a numerical estimate of the Hessian, evaluated at $\tilde{\theta}$. The negative inverse of the estimated Hessian was used as initial $\tilde{\Sigma}$.

2.3 Convergence

Let $p^{(m)}$ denote the probability density function of the sample after m iterations of Algorithm 1 and let $p(\theta) = p(\theta|x)$ be the true posterior. To show convergence of Algorithm 1 we use a proof paralleling the argument which Diebolt and Robert (1990, Appendix A) use to show convergence of the Gibbs sampler. Let $\Theta = \text{support}(p)$ be the parameter space.

Theorem 1 *If the candidate generating function g is such that $g(\theta, y) > 0$ for all $\theta, y \in \Theta$, then $||p^{(m)} - p|| \to 0$, where $||\cdot||$ denotes L_1 norm.*

PROOF. see Appendix.

The following ergodic theorem can be shown:

Corollary 2 *If f is integrable with respect to p, i.e. $\int |f(\theta)|dp(\theta) < \infty$, then*

$$\frac{1}{M}\sum_{m=1}^{M} f(\theta^{(m)}) \to E_p(f),\ p\text{ - }a.s.$$

PROOF. see Appendix.

3 Application to the Gibbs Sampler

In the following let $p(\theta|x)$ and $p(x|\theta)$ denote the posterior p.d.f. and the likelihood function. Also for $\theta = (\theta_1, \ldots \theta_s)' \in \Re^s$, let $p(\theta_i|\theta_1, \ldots, \theta_{i-1}, \theta_{i+1}, \ldots, \theta_s, x)$ and $p(\theta_i|x)$ denote the conditional respectively marginal posterior p.d.f. for θ_i. Assume all involved densities exist with respect to Lebesgue or counting measure.

3.1 The Gibbs Sampling Scheme

The Gibbs sampling scheme is useful for problems where the conditional distributions $p(\theta_i|\theta_1, \ldots, \theta_{i-1},\ \theta_{i+1}, \ldots, \theta_s, x)$, $i = 1, \ldots n$, are "available" for sampling, meaning that efficient random variate generation from these distributions is possible. A sample from the multivariate posterior $p(\theta|x)$ is generated by iterative sampling from the univariate conditional distributions $p(\theta_i|\theta_1, \ldots, \theta_{i-1},\ \theta_{i+1}, \ldots, \theta_s, x)$, $i = 1, \ldots n$. Starting with an arbitrary initial sample $\theta^{(0)} = (\theta_1^{(0)}, \ldots \theta_s^{(0)})'$, the m-th iteration step of the Gibbs sampler can be written as follows:

ALGORITHM 2: GIBBS SAMPLER.

Assume $\theta^{(m-1)}$ is given. Generate $\theta^{(m)} = (\theta_1^{(m)}, \ldots \theta_s^{(m)})'$ by subsequently, for

$i = 1, \ldots s$, generating from the one-dimensional conditionals:

$$\theta_i^{(m)} \sim p(\theta_i | \theta_1^{(m)}, \ldots \theta_{i-1}^{(m)}, \theta_{i+1}^{(m-1)}, \ldots, \theta_s^{(m-1)}, x).$$

Geman and Geman (1984) originally introduced the Gibbs sampling scheme in the context of image reconstruction. See, e.g., Tanner and Wong (1987) and Gelfand and Smith (1990) for further discussion of the Gibbs sampler. Diebold and Robert (1990, appendix A) give convergence results for the Gibbs sampler.

3.2 An Accelerated Gibbs Sampler

The applicability of the Gibbs sampler is restricted by the need to sample from the conditional distributions. Removing this restrictions would achieve two goals: First, it would make the Gibbs sampler available also for problems without this conjugate structure. And secondly, it would allow to orthogonalize the parametrization. The motivation for the orthogonalization is that in the ideal case of independence the Gibbs sampler would produce a true posterior sample after only one additional iteration.

This generalization of the Gibbs sampler becomes possible when using Algorithm 1 to generate from the one dimensional conditionals required in the Gibbs sampling scheme. Together with asymptotic normality of posterior distributions this motivates the following algorithm, which reparametrizes using a current estimate of the posterior covariance matrix. Algorithm 3 describes the generic step of the modified algorithm. In the description of the algorithm, let $p_\eta(\eta) = p(K\eta) \cdot |K|$ denote the posterior p.d.f. in terms of the reparametrization $\eta = K^{-1}\theta$. The updating of $\tilde{\Sigma}$, the choice of g_i and other implementation details are discussed in Section 3.3.

ALGORITHM 3: ACCELERATED GIBBS SAMPLER.

Let $\tilde{\Sigma} = KK'$ be a current estimate of the posterior covariance matrix. Let $\eta^{(m)} = K^{-1}\theta^{(m)}$. Generate $\eta^{(m+1)} = (\eta_1^{(m+1)}, \ldots \eta_s^{(m+1)})'$ by subsequently, for $i = 1, \ldots s$, generating form the one-dimensional conditionals:

$$\eta_i^{(m+1)} \sim p_\eta(\eta_i | \eta_1^{(m+1)}, \ldots \eta_{i-1}^{(m+1)}, \eta_{i+1}^{(m)}, \ldots \eta_s^{(m)}, x).$$

Generate $\eta_i^{(m+1)}$ by running Algorithm 1 over T iterations, using $x_0 := \eta_i^{(m)}$ as starting point and setting $p(\cdot) = p_\eta(\eta_i | \eta_1^{(m+1)}, \ldots \eta_{i-1}^{(m+1)}, \eta_{i+1}^{(m)}, \ldots \eta_s^{(m)}, x)$:

- Generate $y \sim g_i(y|x^{(0)})$.
- With probability $a(x^{(0)}, y) = \min(1, p(y)/p(x^{(0)}))$, accept y as the new state, i.e. $x^{(1)} := y$, otherwise $x^{(1)} := x^{(0)}$.

Repeat the last two steps T times to generate $x^{(2)}, \ldots x^{(T)}$ and take $\eta_i^{(m+1)} = x^{(T)}$ as approximate generation from $p(\cdot)$.

Running Algorithm 3 with an initial sample $\{\theta_1^{(0)}, \ldots \theta_n^{(0)}\}$ in parallel generates after sufficiently many iterations an approximate posterior sample. See Section 3.3 for implementation details.

Example 2 in Section 4 illustrates how the orthogonalization improves convergence in the Gibbs sampling scheme, by comparing sample mean trajectories with and without orthogonalization. Since the Gibbs sampler needs only one random variable from each of the distinct conditional distributions, the orthogonalization is only possible when using some generic scheme without setup time to generate from the one dimensional conditionals. The use of conventional schemes, such as ratio of uniforms or accept/reject, is impractical because of the required time to build envelope functions etc. The application

of Algorithm 1 in the above algorithm requires zero setup time. When generating $\eta_i^{(m+1)} \sim p_\eta(\eta_i|\eta_1^{(m+1)},\ldots,\eta_{i-1}^{(m+1)}, \eta_{i+1}^{(m)}, \ldots \eta_s^{(m)}, x)$ the initial point $x^{(0)}$ is taken to be the – already available – previous sample point $\eta_i^{(m)}$. Because of the orthogonalization the density $p_\eta(\eta_i|\eta_1^{(m+1)},\ldots,\eta_{i-1}^{(m+1)}, \eta_{i+1}^{(m)}, \ldots \eta_s^{(m)}, x)$ is similar to $p_\eta(\eta_i|\eta_1^{(m)},\ldots,\eta_{i-1}^{(m)}, \eta_{i+1}^{(m-1)}, \ldots \eta_s^{(m-1)}, x)$, making $\eta_i^{(m)}$ a good starting point.

Algorithm 3 can be seen as just introducing a slightly different candidate generating density in the basic Algorithm 1. Instead of generating at each step from the same multivariate conditional distribution $g(y|\theta^{(m)})$, Algorithm 3 changes only one co-ordinate at each step, using $g_i(y|\theta^{(m)})$, iterating over all coordinates in subsequent steps. The convergence results from Section 2.3 therefore apply with only minor modifications also for Algorithm 3.

In many cases the joint posterior can be adequately approximated by a multivariate envelope density such as a multivariate split Student t density (see Geweke 1989). For such problems an alternative implementation of a generalized Gibbs sampler, which does not require a conjugate structure, was given by Carlin and Gelfand (1990). They suggested the use of a multidimensional envelope density to derive good envelope densities for the complete conditional densities.

3.3 Implementation

For Example 2 we implemented Algorithm 3 with initialization, updating of $\tilde{\Sigma}$, convergence assessment and posterior inference as specified in Section 2.2. In particular, the remarks on the lack of a rigid automated method for the assessment of convergence apply equally for the Gibbs sampling scheme. The

control parameters n, d, M_0, a, n_2 and T were chosen as $n = 10$, $d = 1$, $M_0 = 40$, $a = 2$, $n_2 = 100$ and $T = 10$. Notice that compared with Algorithm 1 the Gibbs sampling scheme includes an additional level of iterations over all coordinates. Therefore the parameters d and M_0 are chosen smaller than in Section 2.2.

Choice of the candidate generating distributions. As candidate generating densities $g_i(y|x)$ we chose $g_i(y|x) = g_i(y - x) \sim N(0, c_i)$. Initially (and after each update of $\tilde{\Sigma}$) all c_i are set to $c_i = 1$. Whenever the average over the 10 most recently observed acceptance probabilities for parameter η_i is greater than 0.8 then c_i is increased by a factor 1.2. If the average is less than 0.2 then c_i is decreased by a factor 0.7.

4 Examples

Example 1: Hierarchical Event Rate Model.

Carlin and Gelfand (1990) considered a hierarchical event rate model with a Poisson likelihood $Y_i \sim \text{Poisson}(\lambda_i t_i)$, where Y_i is the number of occurrences over an exposure time of length $t_i, i = 1, \ldots n$. A conjugate prior choice would be to assume $\lambda_i \sim \text{Gamma}(\alpha, \beta)$, with $\beta \sim \text{Inverse Gamma}(c, d)$ and known α. As alternative to this prior Carlin and Gelfand considered a logstudent-t prior. Let $\epsilon_i := \log(\lambda_i)$ and assume $\epsilon_i \sim t_w(\eta, \sigma)$, with second stage prior $\eta \sim N(\mu, \tau^2)$ and $\sigma^2 \sim \text{Inverse Gamma}(a, b)$, where the hyperparameters μ, τ^2, a and b and the degrees of freedom of the student-t distribution w are assumed known. The fatter tails of the logstudent-t prior make the model more robust than the exponential tails of a gamma distribution.

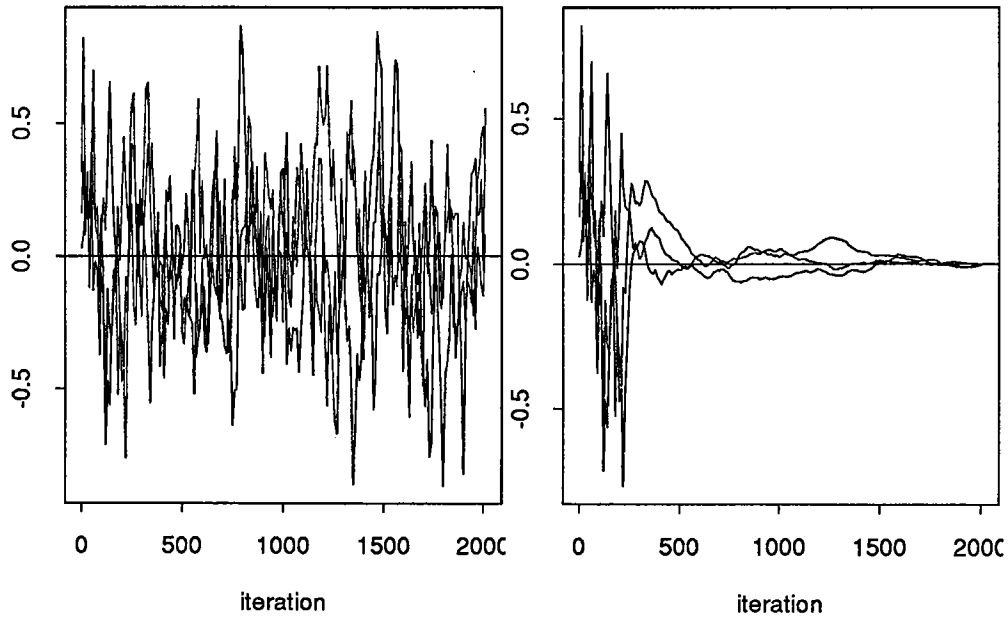
Table 1: Pump Failures data set.

Y_i	5	1	5	14	5
t_i	94.320	15.72	62.880	125.760	5.240
Y_i	19	1	1	4	22
t_i	31.440	1.048	1.048	2.096	10.480

Source: Gaver and O'Muircheartaigh (1987).

The data set in Table 1 concerns the number Y_i of pump failures over given exposure times t_i. Carlin and Gelfand (1990) analyzed this data set using the hyperparameters $\mu = -1$, $\tau^2 = 1$ for the prior on η and $a = 2.01$ and $b = .99$ for σ^2, corresponding to a prior mean of 1 and variance of 100 on σ^2. They use three different specifications for the d.f. w. We will use here only $w = 5$.

We estimated the model using Algorithm 1. Figure 1 shows the trajectories of the sample means $\bar{\theta}^{(m)}$ and $\hat{\theta}^{(m)}$, where θ is the parameter vector $\theta = (\epsilon_1, \ldots \epsilon_{10}, \eta, \sigma)'$. The averages are defined as in (2) and (3) of Section 2.2. As discussed in Section 2.2 no formal method is available to assess convergence. We decided to stop after $M_1 = 2000$ iterations, since the trajectories of $\hat{\theta}^{(m)}$ are leveling off after around 1500 iterations and the paths of $\bar{\theta}^{(m)}$ oscillate within reasonable bounds of the estimated posterior mean ("reasonable" for a sample mean of $n = 10$ chains). Over an additional $M_2 = 30$ iterations an approximate posterior sample $X = \{\theta_i = (\epsilon_{1,i}, \ldots \epsilon_{10,i}, \eta_i, \sigma_i)', i = 1, \ldots N\}$ was accumulated to estimate the marginal posteriors $p(\epsilon_1|data)$, $p(\epsilon_5|data)$ and $p(\epsilon_{10}|data)$. Rather than using $\{\epsilon_{1,i}, i = 1, \ldots N\}$ directly we made use of the available structural information and estimated $p(\epsilon_1|data)$ by the following



a) Trajectory of $\bar{\theta}^{(m)}$.

b) Trajectory of $\hat{\theta}^{(m)}$.

Figure 1: Trajectory of the sample means. Only ϵ_1, ϵ_5 and ϵ_{10} are plotted. The vertical scale is in posterior deviations from the posterior mean.

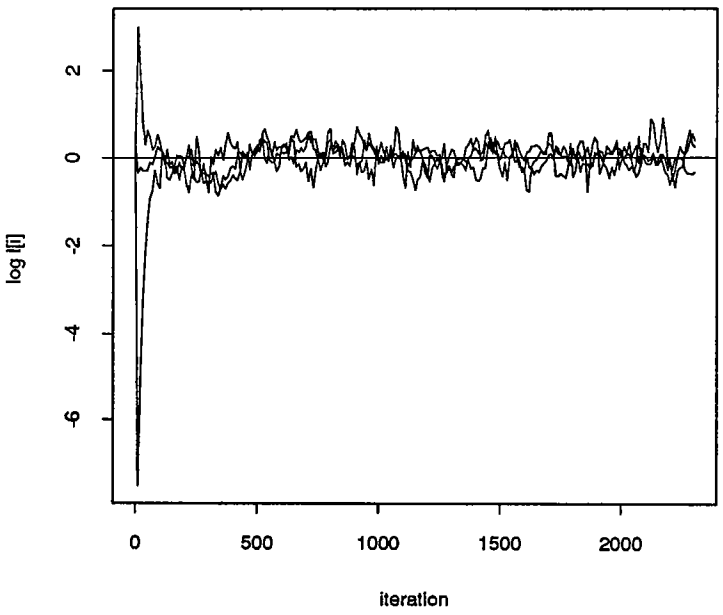


Figure 2: Trajectory of $\bar{\theta}^{(m)}$, started at the prior mean $\tilde{\theta}^*$. The vertical scale is in posterior deviations from the posterior mean. The first 100 iterations move the Monte Carlo sample towards the mass of the posterior.

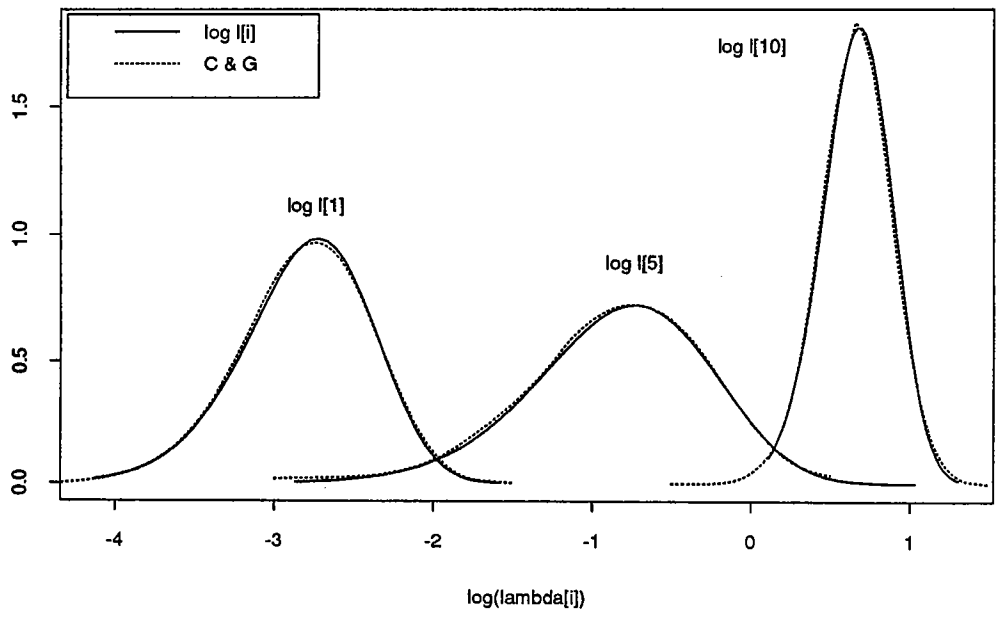


Figure 3: Marginal posterior densities $p(\epsilon_1|x)$, $p(\epsilon_5|x)$ and $p(\epsilon_{10}|x)$. The dotted lines plot for reference the density estimate given in Carlin and Gelfand (1990).

Monte Carlo estimate:

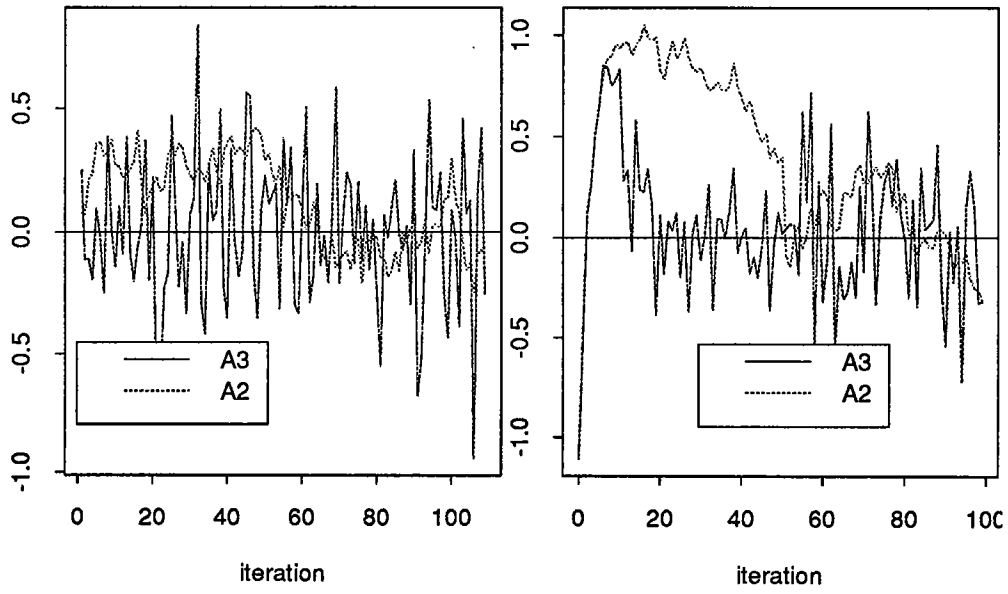
$$\hat{p}(\epsilon_1|data) = \frac{1}{N}\sum_{i=1}^{N} p(\epsilon_1|\epsilon_{2,i},\ldots\epsilon_{10,i},\eta_i,\sigma_i,data). \tag{6}$$

The conditional distributions $p(\epsilon_1|\epsilon_{2,i},\ldots\sigma_i,data)$ can easily be evaluated, since – up to an integration constant – they are the joint posterior as a function of ρ when the other parameters are held fixed at $\epsilon_{2,i},\ldots\sigma_i$. Computing the integration constants involves N one dimensional integrations, which, however, do not require any additional computational effort, since the conditional distributions have to anyway be evaluated at the grid points for the density plot. Figure 3 shows the estimated marginal posteriors together with the density estimates given in Carlin and Gelfand (1990). The sample size N used for the density estimation was determined by the pragmatic aim of producing a "smooth" density plot. In this application $N = 300$ was used.

To explore the robustness of Algorithm 1 with respect to the initial sample we repeated the analysis pretending that the posterior mode and covariance matrix were unknown. We initialized the algorithm with a normal sample around the prior expectation $\tilde{\mu} = -1$, $\tilde{\sigma^2} = 1$ and $\tilde{\epsilon}_i = -1$. In the absence of any better guess we used the identity matrix as covariance matrix $\tilde{\Sigma}$ for the initial sample. The first few iterations of Algorithm 1 worked as stochastic optimization scheme and shifted the sample points towards the mass of the posterior distribution. This is shown in Figure 2, which plots $\bar{\theta}^{(m)}$ for $m = 1,\ldots 1000$, where $\bar{\theta}^{(m)} = \sum_{i=1}^{n}\theta_i^{(m)}/n$ is the sample average over all n parallel chains at iteration m. The vertical scale is in posterior standard deviations from the posterior mean. After around only 100 iterations the sample averages $\bar{\theta}^{(m)}$ have moved towards the posterior mean. □

Example 2: A Yield-Density Model.

Define X to be the number of plants per unit area and Y to be the yield per plant. To model the relationship between yield and planting density Holliday (1960) proposed the model $Y_i = (\alpha + \beta X + \gamma X^2)^{-1} + \epsilon_i$, where $\epsilon_i \sim N(0, \sigma^2)$. A noninformative prior $\pi(\alpha, \beta, \gamma, \sigma) \propto 1/\sigma$ completes the model. If $\gamma = 0$, then $\alpha = lim_{X \to 0} Y$ has a biological interpretation as "genetic potential" and $1/\beta = \lim_{X \to \infty} XY$ can be considered a measure of "environmental potential". As an illustration of Algorithm 3 we analyzed this model with a data set taken from Ratkowsky (1983, p. 58, MG data set). The data set consists of 42 observations on yield and planting density from an onion spacing trial in southern Australia. Posterior mode $\tilde{\theta}$ and negative inverse Hessian $\tilde{\Sigma}$, evaluated at $\tilde{\theta}$, provide a reasonable initialization for Algorithm 3. Figure 4a shows the trajectory of $\bar{\rho}^{(m)} = \sum_{i=1}^{n} \rho_i^{(m)}/n$, i.e. the sample means over the n parallel chains at iteration m. The vertical scale is in terms of posterior standard deviations from the posterior mean. The solid line shows the trajectory using reparametrization (after the first 20 and 40 iterations, using the estimates updated as described in Section 4.3.2). The dotted line shows the trajectory when running the Gibbs sampler without reparametrization (with common random numbers). It can be clearly seen how the reparametrization reduced serial correlation. The effect is seen even stronger in Figure 4b which plots the same trajectories with the chains started with a sample around $\tilde{\theta}^* = (0.01, 0, 0, 0.12)'$, where $\tilde{\alpha}^* = 0.01$ and $\tilde{\sigma}^* = 0.12$ are sample mean and sample variance of the $\frac{1}{Y_i}$'s. The process $\bar{\rho}^{(m)}$ from the reparametrized Gibbs sampler is practically converged after only around 25 iterations, whereas for the simple Gibbs sampler they are still drifting even after 100 iterations.



a) Started at $\tilde{\theta}$.

b) Started at $\tilde{\theta}^*$.

Figure 4: Trajectory for $\bar{\gamma}^{(m)}$, (A3) – with reparametrization, (A2) – without reparametrization. Vertical scale is in posterior S.D. from the posterior mean.

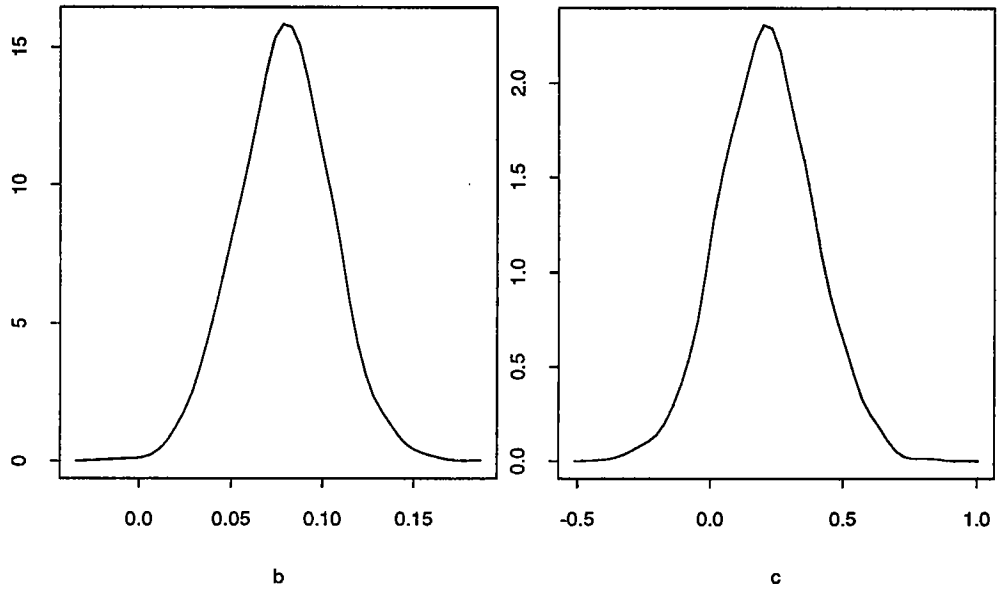


Figure 5: Marginal posteriors $p(\beta(10^3)|x)$ and $p(\gamma(10^6)|x)$.

As remarked in Section 3.3 no formal method of judging satisfactory convergence is available. Following the informal procedure outlined in Section 2.2 we stopped iteration after $M_1 = 100$ iterations. A sample $X = \{\theta_i, i = 1, \ldots N\}$ of size $N = 3000$ was accumulated to be used for density estimation. The procedure outlined in Example 1 (6) is impractical here since the conditional distributions are very sharp due to high correlations between the parameters. The density estimates shown in Figure 5 are obtained by conventional kernel density estimates from the posterior sample X (β and γ are rescaled by 10^3 and 10^6 respectively).

The estimated posterior means $E(\alpha|x) = .0045$, $E(\beta|x) = .08 \cdot 10^{-3}$, $E(\gamma|x) = .20 \cdot 10^{-6}$ and $E(\sigma^2|x) = .012$ confirm the LS estimates reported by Ratkowsky: $\hat{\alpha} = .004524$, $\hat{\beta} = .08113 \cdot 10^{-3}$, $\hat{\gamma} = .1976 \cdot 10^{-6}$ and $\hat{\sigma^2} = .01231$.

□

5 Comparison with importance sampling

As an indication of how the method compares with conventional schemes we estimated the examples from Section 4 and three more examples which we worked with also by Monte Carlo integration with importance sampling. See Geweke (1989) for a complete discussion of importance sampling. Following a suggestion of Geweke (1989) we chose as importance sampling densities multivariate split Student densities $t^*(\tilde{\theta}, \tilde{\Sigma}, \boldsymbol{q}, \boldsymbol{r}, \nu)$, where $\tilde{\theta}$ is the posterior mode and $\tilde{\Sigma}$ is the negative inverse Hessian evaluated at $\tilde{\theta}$. The split scaling parameters $\boldsymbol{r}$ and $\boldsymbol{q}$ were derived by the algorithm given in Geweke (1989). The d.f. ν are problem specific.

Let $p(\theta)$ denote the posterior density $p(\theta|x)$. Assume θ is an s-dimensional parameter vector $\theta = (\theta_1, \ldots \theta_s)$ with posterior expectations $E_p\theta_j$. Let $\hat{\theta}^{(n)}_{M,j}$ denote the estimator for $E_p\theta_j$ derived by Algorithm 1 and let $\hat{\theta}^{(n)}_{IS,j}$ be the importance sampling estimator for $E_p\theta_j$, where the superscript n refers to the number of posterior evaluations. Denote the variances of these two estimators by: $\tau^{2(n)}_{IS,\,j} = \mathrm{var}(\hat{\theta}^{(n)}_{IS,j}) = E(\hat{\theta}^{(n)}_{IS,j} - E_p\theta_j)^2$ and $\tau^{2(n)}_{M,\,j} = \mathrm{var}(\hat{\theta}^{(n)}_{M,j}) \approx E(\hat{\theta}^{(n)}_{M,j} - E_p\theta_j)^2$ (For sufficiently large n, when the chain is "practically" converged, $E\hat{\theta}^{(n)}_{M,j} \approx E_p\theta_j$). The variances $\tau^{2(n)}_{M,\,j}$ and $\tau^{2(n)}_{IS,\,j}$ are expectations with respect to the random variables generated in the estimation schemes and will be referred to as numerical variances to distinguish them from the posterior variances. It is relatively easy to estimate the numerical variances $\tau^{2(n)}_{IS,\,j}$ by statistics

$\hat{\tau}^{2(n)}_{IS,\,j}$, given in Geweke (1989), which can be computed while running the importance sampling algorithm. Estimation of the $\tau^{2(n)}_{M,\,j}$ is unfortunately not as straightforward. We estimated them by five independent repetitions of Algorithm 1 for each example.

Since in both algorithms, Algorithm 1 as well as importance sampling, most computer time is spent on evaluating the posterior density at the sampled points, the number of posterior evaluations is useful to compare the algorithms. Table 2 lists for each example N_M and N_{IS}, where N_M is the number of posterior evaluations required for Algorithm 1, and N_{IS} is the minimum number n of posterior evaluations required to obtain $\hat{\tau}^{(n)}_{IS,\,j} \leq \hat{\tau}^{(N_M)}_{M,\,j}$, for $j = 1, \ldots s$. The hierarchical event rate model and the yield-density model were analyzed in Examples 1 and 2 in Section 4. The exponential regression model was found in Berger and Ye (1991). The ARCH linear model example is discussed in Geweke (1989). The multiplicative row and column (RC) effects model was taken from Agresti and Chuang (1986). The last three models are only used here to give some indication of the computational effort involved in using Algorithm 1 respectively Algorithm 3. See the cited references for detailed descriptions of the models. The yield-density model and the row and column effects model were estimated using the Metropolis algorithm embedded in the Gibbs sampling scheme (Algorithm 3).

In the exponential regression model the standard importance sampling algorithm as outlined above failed to provide reasonable estimates within the first 30,000 drawings from the importance sampling density. The estimated posterior standard deviations after 30,000 drawings were still too small by a factor five (by comparison with an "exact" solution by conventional numerical

Table 2: Number of posterior evaluations for Algorithm 1 (N_M) and importance sampling (N_{IS}).

	N_M	N_{IS}
Algorithm 1		
Hierachical Event Rate Model	20,000	> 20,000
Exponential Regression	30,000	NA
ARCH Linear Model	10,000	3,000
Algorithm 3		
Yield-Density Model	40,000	3,000
Multiplicative RC Model	75,000	71,000

integration). The main reason probably was that the negative inverse Hessian, which was used as scale matrix $\tilde{\Sigma}$ for the importance sampling density, underestimates the posterior standard deviations by up to a factor five. (Further sampling from the importance sampling density should eventually hit at sample points with extreme importance sampling weights, which should then cause the algorithm to revise the estimates).

Appendix: Proof of Theorem 1

The argument parallels the proof which Diebold and Robert (1990) give for the convergence of the Gibbs sampling scheme. Only the transition probabilities here are determined by Algorithm 1, rather than the Gibbs sampler.

Notice that $\{\theta^{(m)}, m \geq 1\}$ is a Markov chain on $(\Theta, \mathcal{B})$, where $\mathcal{B}$ are the

Borel sets in Θ, with transition probabilities given by the kernel $T(\theta, A) := q(\theta)1_A(\theta) + \int_A a(\theta, y)g(\theta, y)dy$, where $q(\theta) = (1 - \int a(\theta, y)g(\theta, y)dy)$.

Define an m-step transition probability by $T^m(\theta, A) = \int T(\theta', A)T^{m-1}(\theta, d\theta')$. Given a σ-finite measure ν, a kernel $T(\theta, A)$ is called ν-irreducible if, for all $A \in \mathcal{B}$ such that $\nu(A) > 0$ and all $\theta \in \Theta$, there exists an integer m such that $T^m(\theta, A) > 0$.

Lemma 3 *Let μ be defined by $\mu(A) = \int_A p(\theta)d\theta$. The kernel $T(\theta, A)$ is μ-irreducible.*

PROOF. $T(\theta, A) \geq \int_A a(\theta, y)g(\theta, y)dy > 0.$ □

Associated with a transition probability $T(\theta, A)$ is a potential kernel $G(\theta, A)$, defined as $G(\theta, A) = \sum_{n=1}^{\infty} T^n(\theta, A)$. The kernel is said to be *proper* if Θ can be written as the union of an increasing sequence of Θ_n sets in $\mathcal{B}$ such that the functions $G(., \Theta_n)$ are uniformly bounded for each n.

A Markov chain is said to be *recurrent in the sense of Harris* if there exists a positive, σ-finite measure λ, such that λ is invariant with respect to T, i.e. $\int T(\theta, A)\lambda(d\theta) = \lambda(A)$, and such that $\lambda(C) > 0$ implies

$$P_{\theta^{(0)}}[\theta^{(m)} \text{ visits } C \text{ i.o.}] = 1.$$

Lemma 4 *The Markov chain defined by the transition probabilities $T(A, \theta)$ is Harris recurrent with the posterior measure μ as invariant measure.*

PROOF. The invariance of the posterior measure μ, defined by $\mu(A) = \int_A p(\xi)d(\xi)$, is easily shown by substituting into the definition of $T(\theta, A)$. By Theorems 2.3. and 2.5. from Revuz (1984), chapter 3, we have then only left to show that $G(\theta, A)$ is not proper. Assume Θ_n is increasing to Θ. Since

$\mu(\Theta) = 1$, there exists some integer N, such that $\mu(\Theta_n) > 0$ for all $n \geq N$ and hence $\int G(\theta, \Theta_n)\mu(d\theta) = \int \sum_{m=1}^{\infty} T^m(\theta, \Theta_n)\mu(d\theta) = \sum_{m=1}^{\infty} \mu(\Theta_n) = \infty$, which implies that $G(., \Theta_n)$ cannot be bounded, i.e. the potential kernel $G(\theta, A)$ is not proper. □

Theorem 1 becomes now a corollary of Proposition 2.5. in Revuz (1984), chapter 6.

Having shown in the previous proof that $\{\theta^{(m)}, m \geq 1\}$ is a Harris chain, Corollary 2 becomes a consequence of an ergodic theorem for Harris chains: Theorem 4.3., Revuz (1984), chapter 4.

References

Agresti, A. and Chuang, C. (1986), "Bayesian and maximum likelihood approaches to order-restricted inference for models for ordinal categorical data," in *Advances in Order Restricted Statistical Inference*, ed. D. Brillinger et al., Lecture Notes in Statistics, vol. 37, pp. 6-27, Berlin: Springer-Verlag.

Becker, R.A, Chambers, J.M. and Wilks, A.R. (1988), *The New S Language*, Pacific Grove: Wadsworth and Brooks/Cole.

Berger, J.O. and Ye, K. (1990), "Noninformative priors for inferences in exponential regression models," Technical Report **90-05**, Purdue University, Statistics Department.

P., Carlin, B. and Gelfand A.E. (1990), "An iterative Monte Carlo method for nonconjugate Bayesian analysis," Technical Report, Carnegie Mellon University, Department of Statistics.

Diebolt, J. and Robert C. (1990), "Bayesian estimation of finite mixture distributions, Part II: Sampling implementation," Technical Report, Universite Paris VI, L.S.T.A.

Gaver, D.P. and O'Muircheartaigh, I.G. (1987), "Robust empirical Bayes analysis of event rates," *Technometrics* **29**, pp. 1-15.

Gelfand, A.E., Hills S.E., Racine-Poon A., and Smith A.F.M. (1990), "Illustration of Bayesian inference in normal data models using Gibbs sampling," *Journal of the American Statistical Association* **85**, pp. 972-985.

Gelfand, A.E. and Smith A.F.M. (1990), "Sampling based approaches to calculating marginal densities," *Journal of the American Statistical Association* **85**, pp. 398-409.

Geman, S. and Geman A. (1984), "Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **6(6)**, pp. 721-740.

Geweke, J. (1989), "Bayesian inference in econometric models using Monte Carlo integration," *Econometrica* **57**, pp. 1317-39.

Hastings, W.K. (1970), "Monte Carlo sampling methods using Markov chains and their applications," *Biometrika* **57**, pp. 97-109.

Holliday, R. (1960), "Plant population and crop yield," *Field Crop Abstracts* **13**, pp. 159-167, 247-254.

Laarhoven, van P.J.M. (1987), *Simulated Annealing: Theory and Applications*, Amsterdam: D. Reidlt.

Metropolis, N, Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H. and Teller, E. (1953), "Equations of state calculations by fast computing machines," *Journal of Chemical Physics* **21**, pp. 1087-1091.

Naylor, J.C. and Smith, A.F.M. (1988), "Economic illustrations of novel numerical integration strategies for Bayesian inference," *Journal of Econometrics* **38**, pp. 103-126.

Ratkowsky, D.A. (1983), *Nonlinear Regression Modeling*, New York: Marcel Dekker.

Revuz, D. (1984), *Markov Chains*, Amsterdam: North-Holland.

Tanner, M.A. and Wong W.H. (1987), "The calculation of posterior distributions by data augmentation," *Journal of the American Statistical Association* **82**, pp. 528-550.

Tierny, L. and Kadane J. (1986), "Accurate approximations for posterior moments and marginal densities," *Journal of the American Statistical Association* **81**, pp. 82-86.

Tierny, L. (1990), *Lisp-Stat: An object-oriented environment for statistical computing and dynamic graphics.* New York: J. Wiley.