Monte Carlo Markov Chain Methods

Jan Krohn

Supervisor: Dr Peter M Lee

MONTE CARLO MARKOV CHAIN METHODS

JAN KROHN

Contents

Date: 14 May 2001.

I would like to thank Dr Peter Lee for supervising my project and Anke Feiten and Adam Depledge for proof reading it.

1. Introduction

The Gibbs sampler has its origin in digital image processing and was introduced by Geman and Geman [GG] in 1984 for the Gibbs distribution. It was only in 1990 that Gelfand and Smith [GS] discovered that the Gibbs sampler works for other distributions as well.

The general idea of the Gibbs sampler is to approximate the modes and marginal distributions of an unknown distribution $p(\omega)$, where $\omega = (\omega_1, \dots, \omega_n)$, by successively taking samples from the known conditional distributions $p_i = p(\omega_i \mid \omega_j, j \neq i)$.

The actual algorithm works as follows:

(1) Set $j = 0$ and set initial values $\omega^{(0)} = \left(\omega_1^{(0)}, \dots, \omega_n^{(0)}\right)$.

(2) For $1 \leq i \leq n$ obtain samples

$$\omega_i^{(j)} \sim p\left(\omega_1 \mid \omega_1^{(j)}, \dots, \omega_{i-1}^{(j)}, \omega_{i+1}^{(j-1)}, \dots, \omega_n^{(j-1)}\right)$$

from the conditional distributions and, therefore, a new value $\omega^{(j)}$.

(3) Increase j by 1 and continue with step 2.

Since the new values depend only on the immediately preceding ones, we clearly have a Markov process.

The Gibbs sampler was first used in digital image processing, an application we look at in sections 2 and 5.

In section 3 we will look at Markov random fields and Gibbs distributions and see how they are related to each other. The most important result in this section is that they are in fact equivalent.

We will see in section 4 that the Gibbs sampler converges to the unknown joint distribution $p(\omega)$, and we will introduce the process of annealing, that is, gradually reducing the temperature of the system to speed up convergence.

In section 6 we will generalise the results from section 4 and show that the Gibbs sampler works for other distributions as well. We will introduce two other related algorithms, the data-augmentation algorithm, which approximates the joint distribution given the conditional distributions, and the substitution sampling algorithm, which is very close related to the Gibbs sampler.

2. Digital Image Processing

Geman and Geman [GG] developped the Gibbs sampler to restore degraded digital (grey scale) images. Suppose the image has the size $m \times m$ pixels, then let $Z_m = \{(i,j) \mid 1 \leq i,j \leq m\}$ denote the set of all possible co-ordinates of the image, also called the integer lattice. Let $F = (F_{i,j})_{(i,j)\in Z_m}$ denote the matrix of pixel intensities of the original image and $G = \phi\left(H\left(F\right)\right) \odot N$ the matrix of the degraded image. Here we allow blurring, noise and some nonlinearities. The function ϕ is nonlinear, H is a linear blurring function and N an independent Gaussian white noise. The operator $\odot$ is any invertible operator. On the pixel level we write

$$G_{i,j} = \phi\left(\sum_{(k,l)\in Z_m} H\left(i-k,j-l\right) F_{k,l}\right) \odot \nu_{i,j} \tag{2.1}$$

for $(i,j) \in Z_m$.

The original image is a pair $X = (F, L)$ with F as above and L a matrix of unobservable edge elements. These edge elements link horizontally or vertically neighboured pixels and can only have two states; either an edge is present or it is not.

Definition 2.1. For any finite set $\mathcal{S}$ define a **neighbourhood system** as a map

$$\mathcal{N} : \mathcal{S} \to \mathfrak{P}\left(\mathcal{S}\right), \, i \mapsto \mathcal{S}_i \subseteq \mathcal{S}$$

such that $i \notin \mathcal{S}_i$ for all $i \in \mathcal{S}$, and $i \in \mathcal{S}_j$ iff $j \in \mathcal{S}_i$. The elements of $\mathcal{S}_i$ are called the **neighbours** of i.

The pair $(\mathcal{S}, \mathcal{N})$ can be seen as a graph with nodes $\mathcal{S}$ and arcs $\mathcal{N}$.

Definition 2.2. A **Markov Random Field (MRF)** over the (not necessarily finite) graph $(\mathcal{S}, \mathcal{N})$ is a stochastic process $(X_s)_{s\in\mathcal{S}}$ such that

$$\mathrm{P}\left(X = \omega\right) > 0 \tag{2.2}$$

and

$$\mathrm{P}\left(X_s = \omega_s \mid X_t = \omega_t,\, t \neq s\right) = \mathrm{P}\left(X_s = \omega_s \mid X_t = \omega_t,\, t \in \mathcal{N}_t\right) \tag{2.3}$$

for all realisations ω of X. The expressions on the left hand side of (2.3) are called **local characteristics**.

Note that in the case of digital image processing there is only a finite number of realisations for X, namely $L^{|Z_m|}$, where L is the number of possible grey levels. This is usually rather big; for example, take a black/white image with 64×64 pixels, then $L^{|Z_m|} = 2^{64\times 64} = 2^{4096} \doteq 0.1044388305 \times 10^{1234}$.

The MRF is a multidemensional generalisation of the definition of a Markov chain $(X_n)_{n\in\mathbb{N}_0}$; take $\mathcal{S} = \mathbb{N}_0$ and $\mathcal{N}_n = \{n-1, n+1\}$ and use the following

Remark 2.3. The Markov chain property

$$\mathrm{P}\left(X_n = \omega_n \mid X_0 = \omega_0, \ldots, X_{n-1} = \omega_{n-1}\right) = \mathrm{P}\left(X_n = \omega_n \mid X_{n-1} = \omega_{n-1}\right)$$

is equivalent to

$$\mathrm{P}\left(X_n = \omega_n \mid X_0 = \omega_0, \ldots, X_{n-1} = \omega_{n-1}, X_{n+1} = \omega_{n+1}, \ldots\right) = \\ \mathrm{P}\left(X_n = \omega_n \mid X_{n-1} = \omega_{n-1}, X_{n+1} = \omega_{n+1}\right).$$

So all results we will find for MRFs are as well valid for Markov chains.

To prove this, first assume the one-sided Markov property. Therefore we can transform the joint distribution into

$$\mathrm{P}\left(X = \omega\right) = \mathrm{P}\left(X_0 = \omega_0\right) \prod_{s=1}^{\infty} \mathrm{P}\left(X_s = \omega_s \mid X_{s-1} = \omega_{s-1}\right) \tag{2.4}$$

and use the definition of the conditional distribution to get

$$\begin{aligned}
&\mathrm{P}\left(X_n = \omega_n \mid X_0 = \omega_0, \ldots, X_{n-1} = \omega_{n-1}, X_{n+1} = \omega_{n+1}, \ldots\right) = \\
&\qquad \frac{\mathrm{P}\left(X = \omega\right)}{\mathrm{P}\left(X_0 = \omega_0, \ldots, X_{n-1} = \omega_{n-1}, X_{n+1} = \omega_{n+1}, \ldots\right)} = \\
&\qquad \frac{\mathrm{P}\left(X = \omega\right)}{\sum_{\omega'} \mathrm{P}\left(X_0 = \omega_0, \ldots, X_{n-1} = \omega_{n-1}, X_n = \omega', X_{n+1} = \omega_{n+1}, \ldots\right)} .
\end{aligned}$$

Using (2.4) and cancelling out all possible factors then gives

$$\begin{aligned}
&\frac{\mathrm{P}\left(X_n = \omega_n \mid X_{n-1} = \omega_{n-1}\right) \mathrm{P}\left(X_{n+1} = \omega_{n+1} \mid X_n = \omega_n\right)}{\sum_{\omega'} \mathrm{P}\left(X_n = \omega' \mid X_{n-1} = \omega_{n-1}\right) \mathrm{P}\left(X_{n+1} = \omega_{n+1} \mid X_n = \omega'\right)} = \\
&\qquad \frac{\mathrm{P}\left(X_n = \omega_n, X_{n+1} = \omega_{n+1} \mid X_{n-1} = \omega_{n-1}\right)}{\sum_{\omega'} \mathrm{P}\left(X_n = \omega', X_{n+1} = \omega_{n+1} \mid X_{n-1} = \omega_{n-1}\right)} = \\
&\qquad \frac{\mathrm{P}\left(X_n = \omega_n, X_{n+1} = \omega_{n+1} \mid X_{n-1} = \omega_{n-1}\right)}{\mathrm{P}\left(X_{n+1} = \omega_{n+1} \mid X_{n-1} = \omega_{n-1}\right)} = \\
&\qquad\qquad \mathrm{P}\left(X_n = \omega_n \mid X_{n-1} = \omega_{n-1}, X_{n+1} = \omega_{n+1}\right)
\end{aligned}$$

by applying the definition of the conditional distribution once more.

Now assume the two-sided Markov property. First note that

$$\begin{aligned}
&\sum_{\omega_{n+2}, \cdots} \mathrm{P}\left(X_{n+1} = \omega_{n+1}, \ldots\right) = \\
&\qquad \sum_{\omega_{n+2}, \cdots} \mathrm{P}\left(X_{n+1} = \omega_{n+1}\right) \mathrm{P}\left(X_{n+2} = \omega_{n+2}, \ldots \mid X_{n+1} = \omega_{n+1}\right) = \\
&\qquad \mathrm{P}\left(X_{n+1} = \omega_{n+1}\right) \sum_{\omega_{n+2}, \cdots} \mathrm{P}\left(X_{n+2} = \omega_{n+2}, \ldots \mid X_{n+1} = \omega_{n+1}\right) = \\
&\qquad\qquad \mathrm{P}\left(X_{n+1} = \omega_{n+1}\right)
\end{aligned}$$

Using this fact and the Theorem of Total Probability we write

$$\begin{aligned}
&\mathrm{P}\left(X_n=\omega_n \mid X_0=\omega_0,\ldots,X_{n-1}=\omega_{n-1}\right)= \\
&\sum_{\omega_{n+1},\ldots}\mathrm{P}\left(X_n=\omega_n,\ldots,X_{n-1}=\omega_{n-1},X_{n+1}=\omega_{n+1},\ldots\right)\mathrm{P}\left(X_{n+1}=\omega_{n+1},\ldots\right)= \\
&\qquad\sum_{\omega_{n+1}}\mathrm{P}\left(X_n=\omega_n \mid X_{n-1}=\omega_{n-1},X_{n+1}=\omega_{n+1}\right)\mathrm{P}\left(X_{n+1}=\omega_{n+1}\right)= \\
&\qquad\qquad\qquad\qquad\mathrm{P}\left(X_n=\omega_n \mid X_{n-1}=\omega_{n-1}\right),
\end{aligned}$$

which is the one-sided Markov property.

Definition 2.4. We call a subset $C \subseteq \mathcal{S}$ a **clique** if $i \in \mathcal{N}_j$ for all $i, j \in C$ with $i \neq j$, that is, every pair of distinct elements in C are neighbours. Let $\mathcal{C}$ denote the set of all cliques.

We assume that F is an MRF over $(\mathcal{S}, \mathcal{F})$ for some suitable set $\mathcal{S}$ and neighbourhood system $\mathcal{F}$, which are usually one of the following cases.

Case 1: Take $\mathcal{S} = Z_m$ and

$$\mathcal{F} = \mathcal{F}_c = \{\mathcal{F}_{i,j} \mid (i,j) \in Z_m\}$$

with

$$\mathcal{F}_{i,j} = \left\{(k,l) \in Z_m \mid 0 < (k-i)^2 + (l-j)^2 \leq c\right\}.$$

Some examples for $1 \leq c \leq 8$ are shown in Figure 1. The symbol $\bullet$ stands for a neighbour of the symbol $\circ$.

Case 2: Take $\mathcal{S} = D_m$ the dual lattice, that is, the set of all co-ordinates of L, the matrix containing the links between pixels. Define a neighbourhood system $\mathcal{L} = \{\mathcal{L}_d \mid d \in D_m\}$ in which each $\mathcal{L}_d$ contains six elements like those denoted by an × in Figure 2 for the × in the middle. Between two pixels there might or might not be a link; the right side of Figure 2 shows a realisation of by a binary line process randomly allocated links.

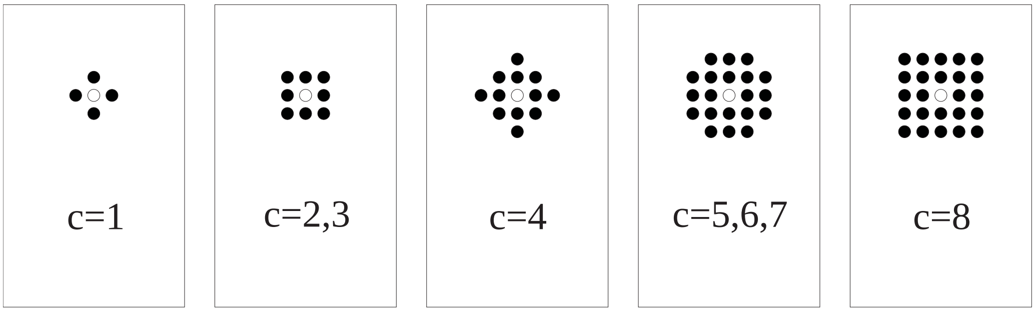

Figure 1. Neighbourhood system for $1 \leq c \leq 8$

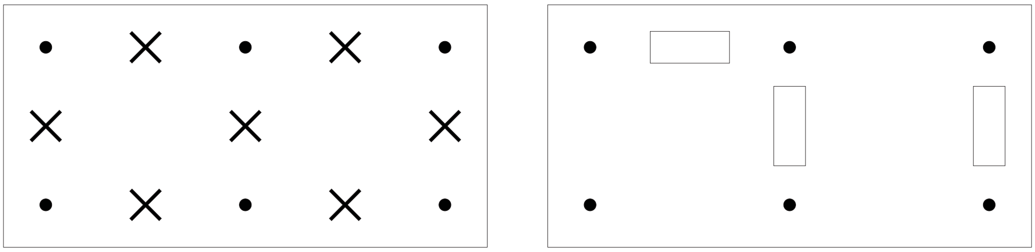

Figure 2. Dual lattice

Case 3: Take $\mathcal{S} = Z_m \cup D_m$ and the neigbourhood systems $\mathcal{F}_1$, also called nearest-neighbour system, for Z_m and the system $\mathcal{L}$ defined in case 2 for D_m. Additionally, an element of Z_m is a neigbour of an element of D_m when they are adjacent.

In the following section we will see that this is equivalent to F having a Gibbs distribution. The example of digital image processing is continued in section 5.

3. Markov Random Fields and Gibbs Distributions

In this section we will have a closer look at MRFs and the Gibbs distribution. Most of the results stated here are from [GG] as well.

Definition 3.1. Given a graph $(\mathcal{S}, \mathcal{N})$ and a realisation space $\Omega = \Lambda^{\mathcal{S}}$, where $\Lambda = \{0, \ldots, L-1\}$ is the state space of the single co-ordinates, a **Gibbs distribution** relative to $(\mathcal{S}, \mathcal{N})$ is given by

$$p(\omega) = \frac{1}{Z} e^{-U(\omega)/T}$$

for $\omega \in \Omega$, where T is a constant, the **temperature** of the system, U the **energy function** given by

$$U(\omega) = \sum_{C \in \mathcal{C}} V_C(\omega),$$

the function $V_C(\omega)$ is independent of co-ordinates ω_s with $s \notin \mathcal{C}$, and Z is a normalising constant as we require $\sum_{\omega \in \Omega} p(\omega) = 1$, so

$$Z = \sum_{\omega \in \Omega} e^{-U(\omega)/T}.$$

The set $\{V_C \mid C \in \mathcal{C}\}$ is called a **potential**.

The modes of the distribution do not depend on the choice of T. Let $p_T(\omega_1) > p_T(\omega_2)$ for some $\omega_1, \omega_2 \in \Omega$. That is, $-U(\omega_1)/T > -U(\omega_2)/T$. Multiplying by T/S gives $-U(\omega_1)/S > -U(\omega_2)/S$ and therefore $p_S(\omega_1) > p_S(\omega_2)$ for all $S > 0$.

In particular, if $S < T$ and ω_0 a mode of p_T and p_S, then for any $\omega \in \Omega$ we have $p_T(\omega_0) \geq p_T(\omega)$ or $U(\omega_0) \leq U(\omega)$. From this we get

$$\frac{1}{p_T(\omega_0)} = \frac{Z_T}{e^{-U(\omega_0)/T}} = \frac{\sum_{\omega \in \Omega} e^{-U(\omega)/T}}{e^{-U(\omega_0)/T}} = \sum_{\omega \in \Omega} \frac{e^{-U(\omega)}}{e^{-U(\omega_0)}} =$$

$$\sum_{\omega \in \Omega} e^{(U(\omega_0) - U(\omega))/T} \geq \sum_{\omega \in \Omega} e^{(U(\omega_0) - U(\omega))/S} = \frac{1}{p_s(\omega_0)},$$

that is, $p_S(\omega_0) \geq p_T(\omega_0)$. So the modes increase with decreasing temperature.

Lemma 3.2. *With $U_{min} = \min\{U(\omega) \mid \omega \in \Omega\}$, the minimal energy, and $\Omega_{min} = \{\omega \in \Omega \mid U(\omega) = U_{min}\}$, the states of minimal energy in Ω, we even* *have*

$$\lim_{T\to 0^+} p(\omega) = \begin{cases} 1/\left|\Omega_{min}\right|, & \omega \in \Omega_{min} \\ 0, & \omega \notin \Omega_{min} \end{cases}$$

and

$$\lim_{T\to\infty} p(\omega) = 1/\left|\Omega\right|$$

for all $\omega \in \Omega$*.*

Proof. To prove the first equality let $\omega_0 \in \Omega_{min}$. Keeping in mind that Ω is finite we then have

$$\lim_{T\to 0^+} \frac{1}{p(\omega_0)} = \lim_{T\to 0^+} \sum_{\omega\in\Omega} e^{(U(\omega_0)-U(\omega))/T} =$$
$$\lim_{T\to 0^+} \left(\sum_{\omega\in\Omega_{min}} e^{(U(\omega_0)-U(\omega))/T} + \sum_{\omega\notin\Omega_{min}} e^{(U(\omega_0)-U(\omega))/T} \right) =$$
$$\lim_{T\to 0^+} \left(\left|\Omega_{min}\right| + \sum_{\omega\notin\Omega_{min}} e^{(U(\omega_0)-U(\omega))/T} \right) =$$
$$\left|\Omega_{min}\right| + \sum_{\omega\notin\Omega_{min}} \lim_{T\to 0^+} e^{(U(\omega_0)-U(\omega))/T} =$$
$$\left|\Omega_{min}\right| + \sum_{\omega\notin\Omega_{min}} e^{-\infty} = \left|\Omega_{min}\right|.$$

Now consider the case $\omega_0 \notin \Omega_{min}$. Then

$$\lim_{T\to 0^+} \frac{1}{p(\omega_0)} = \lim_{T\to 0^+} \sum_{\omega\in\Omega} e^{(U(\omega_0)-U(\omega))/T} \geq$$
$$\lim_{T\to 0^+} e^{(U(\omega_0)-U_{min})/T} = e^{\lim_{T\to 0^+} (U(\omega_0)-U_{min})/T} = e^{\infty} = \infty.$$

Similarly we prove the second equality.

$$\lim_{T\to\infty}\frac{1}{p(\omega_0)}=\lim_{T\to\infty}\sum_{\omega\in\Omega}e^{(U(\omega_0)-U(\omega))/T}=$$
$$\sum_{\omega\in\Omega}e^{\lim_{T\to\infty}(U(\omega_0)-U(\omega))/T}=\sum_{\omega\in\Omega}e^{0}=|\Omega|.$$

□

The idea is to cool down the system by decreasing the temperature T and therefore find the modes more easily through sampling. The principle is similar to simulated annealing, see [Gam] or [Mat], with the difference that the cooling there is used to stabilise the system at the lowest energy level.

Theorem 3.3. *The local characteristics defined by* (2.2) *uniquely determine* $p(\omega)=\mathrm{P}(X=\omega)$ *in case* $p(\omega)>0$ *for all* $\omega\in\Omega$.

Proof. This is proved in [Bes]. Let ω and ψ be two realisations of X. To simplify the notation, let $s=|\mathcal{S}|$, then write $\omega=(\omega_1,\dots,\omega_s)$ and $\psi=(\psi_1,\dots,\psi_s)$. We will prove by induction that

$$\frac{p(\omega)}{p(\omega_1,\dots,\omega_{s-n},\psi_{s-n+1},\dots,\psi_s)}=\prod_{i=0}^{n-1}\frac{p(\omega_{s-i}\mid\omega_1,\dots\omega_{s-i-1},\psi_{s-i+1},\dots,\psi_s)}{p(\psi_{s-i}\mid\omega_1,\dots\omega_{s-i-1},\psi_{s-i+1},\dots,\psi_s)} \tag{3.1}$$

for all $0\le n\le s$. For $n=0$ we have $p(\omega)/p(\omega)=1$, which is obviously true. So assume (3.1) is true for n. We know that

$$p(\omega_1,\dots,\omega_{s-n},\psi_{s-n+1},\dots,\psi_s)=$$
$$p(\omega_{s-n}\mid\omega_1,\dots,\omega_{s-n-1},\psi_{s-n+1},\dots,\psi_s)\,p(\omega_1,\dots,\omega_{s-n-1},\psi_{s-n+1},\dots,\psi_s) \tag{3.2}$$

and that

$$p(\omega_1,\dots,\omega_{s-n-1},\psi_{s-n},\dots,\psi_s)=$$
$$p(\psi_{s-n}\mid\omega_1,\dots,\omega_{s-n-1},\psi_{s-n+1},\dots,\psi_s)\,p(\omega_1,\dots,\omega_{s-n-1},\psi_{s-n+1},\dots,\psi_s),$$

which we rewrite to

$$p\left(\omega_1,\ldots,\omega_{s-n-1},\psi_{s-n+1},\ldots,\psi_s\right)=\frac{p\left(\omega_1,\ldots,\omega_{s-n-1},\psi_{s-n},\ldots,\psi_s\right)}{p\left(\psi_{s-n}\mid\omega_1,\ldots,\omega_{s-n-1},\psi_{s-n+1},\ldots,\psi_s\right)}. \tag{3.3}$$

Inserting (3.3) into (3.2) yields

$$\begin{aligned}&p\left(\omega_1,\ldots,\omega_{s-n},\psi_{s-n+1},\ldots,\psi_s\right)=\\&\qquad\frac{p\left(\omega_{s-n}\mid\omega_1,\ldots,\omega_{s-n-1},\psi_{s-n+1},\ldots,\psi_s\right)}{p\left(\psi_{s-n}\mid\omega_1,\ldots,\omega_{s-n-1},\psi_{s-n+1},\ldots,\psi_s\right)}p\left(\omega_1,\ldots,\omega_{s-n-1},\psi_{s-n},\ldots,\psi_s\right).\end{aligned}$$

Noticing that the fraction is the factor for $i=n$ on the right hand side of (3.1) we insert this and rewrite it to

$$\frac{p\left(\omega\right)}{p\left(\omega_1,\ldots,\omega_{s-n-1},\psi_{s-n},\ldots,\psi_s\right)}=\prod_{i=0}^{n}\frac{p\left(\omega_{s-i}\mid\omega_1,\ldots\omega_{s-i-1},\psi_{s-i+1},\ldots,\psi_s\right)}{p\left(\psi_{s-i}\mid\omega_1,\ldots\omega_{s-i-1},\psi_{s-i+1},\ldots,\psi_s\right)},$$

which proves (3.1) is true for all n. In particular, if we set $n=s$, (3.1) yields, after renumberig the factors,

$$\frac{p\left(\omega\right)}{p\left(\psi\right)}=\prod_{i=1}^{s}\frac{p\left(\omega_i\mid\omega_1,\ldots\omega_{i-1},\psi_{i+1},\ldots,\psi_s\right)}{p\left(\psi_i\mid\omega_1,\ldots\omega_{i-1},\psi_{i+1},\ldots,\psi_s\right)}$$

for all $\omega,\psi\in\Omega$. Using the fact that $\sum_{\omega\in\Omega}p\left(\omega\right)=1$ we can write

$$\frac{1}{p\left(\psi\right)}=\sum_{\omega\in\Omega}\frac{p\left(\omega\right)}{p\left(\psi\right)}=\sum_{\omega\in\Omega}\prod_{i=1}^{s}\frac{p\left(\omega_i\mid\omega_1,\ldots\omega_{i-1},\psi_{i+1},\ldots,\psi_s\right)}{p\left(\psi_i\mid\omega_1,\ldots\omega_{i-1},\psi_{i+1},\ldots,\psi_s\right)}.$$

Therefore we have the representation

$$p\left(\psi\right)=\left(\sum_{\omega\in\Omega}\prod_{i=1}^{s}\frac{p\left(\omega_i\mid\omega_1,\ldots\omega_{i-1},\psi_{i+1},\ldots,\psi_s\right)}{p\left(\psi_i\mid\omega_1,\ldots\omega_{i-1},\psi_{i+1},\ldots,\psi_s\right)}\right)^{-1} \tag{3.4}$$

for all $\psi\in\Omega$, which depends only on the local characteristics. □

In practice, however, this representation of the probability measure as a function of the local characteristics is not very useful as Ω is a very large set in all interesting cases. We also need to determine whether a given set of local characteristics matches some distribution. To avoid these problems we will use the following

Theorem 3.4. *If $\mathcal{N}$ is a neigbourhood system on $\mathcal{S}$ then X is an MRF over $(\mathcal{S},\mathcal{N})$ iff $p(\omega)$ is a Gibbs distribution relative to $(\mathcal{S},\mathcal{N})$.*

Proof. This proof is taken from [KS]. Let p be a Gibbs distribution relative to $(\mathcal{S},\mathcal{N})$. Then

$$p(\omega) = \frac{1}{Z} e^{-U(\omega)/T} = \frac{1}{Z} e^{-\sum_{C \in \mathcal{C}} V_C(\omega)/T}. \tag{3.5}$$

The Theorem of Total Probability yields

$$p(\omega) = \sum_{\omega' : \omega'_t = \omega_t, t \neq s} p(\omega') \, p(\omega_s \mid \omega_t, \, t \neq s)$$

for all $s \in \mathcal{S}$ and therefore

$$p(\omega_s \mid \omega_t, \, t \neq s) = \frac{p(\omega)}{\sum_{\omega'} p(\omega')}.$$

So we have, using (3.5),

$$p(\omega_s \mid \omega_t, \, t \neq s) = \frac{e^{-\sum_{C \in \mathcal{C}} V_C(\omega)/T}}{\sum_{\omega'} e^{-\sum_{C \in \mathcal{C}} V_C(\omega')/T}}. \tag{3.6}$$

For a clique C that does not contain s we have $V_C(\omega) = V_C(\omega')$ since the ω' may only differ from ω in t. So all these cancel in (3.6) such that $p(\omega_s \mid \omega_t, \, t \neq s)$ depends only on t and its neighbours. Therefore p defines an MRF.

The other direction of this equivalence involves much more. Proofs using Möbius inversion can be found in [Pre] and [Gri]. A proof making use of the Hammersley-Clifford expansion can be found in [Bes]. □

Now we are able to specify MRFs by specifying potentials instead of local characteristics, which is a lot easier. In fact, we can convert potentials into local characteristics and vice versa by making the the possible cancellations in (3.6) and obtaining

$$p(\omega_s \mid \omega_t, \, t \neq s) = \frac{1}{Z_s} e^{-\sum_{C:\, s \in C} V_C(\omega)/T} \tag{3.7}$$

with

$$Z_s = \sum_{\omega'} e^{-\sum_{C:\, s \in C} V_C(\omega')/T} = \sum_{x \in \Lambda} e^{-\sum_{C:\, s \in C} V_C(\omega^x)/T}, \tag{3.8}$$

where ω^x denotes the configuration which agrees with ω everywhere except s, where it is x. These formulae are used when describing the Gibbs sampler with given potentials in the following sections.

4. The Gibbs Sampler for Gibbs Distributions

In this section we will define the Gibbs sampler and see why it works. The proofs of the theorems in this section can be found in the appendix of [GG].

To estimate the original data we use maximum a posteriori estimation, a form of Bayesian estimation, in which we maximise the the posterior distribution $\mathrm{P}\left(X=\omega \mid G=g\right)$. We know that X is Gibbs distributed, so the problem reduces to minimise U with given data g.

First we need to bring the nodes of $\mathcal{S}$ into an order in which we will visit them to apply the new state to them. So let $\left(n_t\right)_{t\in\mathbb{N}}$ denote this sequence of sites. Let $X_s\left(t\right)$ denote the state of node s after t replacement opportunities. We assume that every site is visited infinitely often and hence get the following result about convergence.

Theorem 4.1 (Relaxation). *Assume that for each $s \in \mathcal{S}$ the sequence $\left(n_t\right)_{t\in\mathbb{N}}$ contains s infinitely often. Then for any starting configuration $\psi \in \Omega$ and every $\omega \in \Omega$ we have*

$$\lim_{t\to\infty} \mathrm{P}\left(X\left(t\right)=\omega \mid X\left(0\right)=\psi\right) = p\left(\omega\right).$$

Until now we have kept the temperature of the system constant. But we have already seen that a decreasing temperature exaggerates the modes of the Gibbs distribution and therefore speeds up the convergence process. The process of cooling down the system is called **annealing**. It is easy to modify the Gibbs sampler with an annealing schedule. Let p_T denote the Gibbs distribution dependent on temperature T, and $T\left(t\right)$ the temperature of the system at step t.

Recall that Ω_{min} is the set of lowest energy configurations and define the distribution p_{min} as the uniform distribution on Ω_{min}. Finally, define $U^{\star} = \max_{\omega} U(\omega)$ and $U_{\star} = \min_{\omega} U(\omega)$ as well as the difference $\Delta = U^{\star} - U_{\star}$.

Theorem 4.2 (Annealing). *Assume that there exists an integer $\tau \geq |\mathcal{S}|$ such that $\mathcal{S} \subseteq \{n_t, \ldots, n_{t+\tau-1}\}$ for all $t \in \mathbb{N}$. Let $T(t)$ be a decreasing sequence of temperatures such that*

$$\lim_{t\to\infty} T(t) = 0$$

and

$$T(t) \geq \frac{|\mathcal{S}|\,\Delta}{\log t}$$

for all $t \geq t_0$ for some $t_0 \geq 2$. Then for any starting configuration $\psi \in \Omega$ and every $\omega \in \Omega$ we have

$$\lim_{t\to\infty} \mathrm{P}\left(X(t) = \omega \mid X(0) = \psi\right) = p_{min}(\omega).$$

Altogether we now have all information needed to describe the Gibbs sampling algorithm (for the Gibbs distribution):

1. Initialise $\left(\omega_s^{(0)}\right)_{s\in\mathcal{S}}$ with the given data. Set t=1.
2. Update value $T(t)$.
3. Maximise U.
4. Get samples for each site $s \in \mathcal{S}$ separately using (3.7) and (3.8), and get $\omega^{(i)}$ from them.
5. Increase t by 1 and continue with step 2.

5. Digital Image Processing (Continued)

We will use the powerful results about MRFs and the Gibbs distribution for further examination of the problem stated in section 2 and quote experimental results on the restoration of images.

We are interested in the posterior distribution $\mathrm{P}(F = f, L = l \mid G = g)$ with given degraded image g. We take $\mathcal{S} = Z_m \cup D_m$ from case 3 in section 2, the collection of pixel and line sites. We assume that $X = (F, L)$ is an MRF relative to $(\mathcal{S}, \mathcal{N})$ for some neighbourhood system $\mathcal{N}$. For convenience take $T = 1$, so we have, according to Theorem 3.4,

$$\mathrm{P}\left(F = f, L = l\right) = \frac{1}{Z} e^{-U(f,l)}$$

and

$$U\left(f, l\right) = \sum_{C \in \mathcal{C}} V_C\left(f, l\right)$$

for some potential $\{V_C\}$.

Recall that $G = \phi\left(H\left(F\right)\right) \odot N$. As $\odot$ is invertible, we denote the inverse by $N = \Phi\left(G, \phi\left(H\left(F\right)\right)\right) = \left(\Phi_s\right)_{s \in Z_m}$. For $s \in Z_m$ let $\mathcal{H}_s \subseteq Z_m$ denote the pixels that affect the blurred image $H\left(F\right)$ at s. For example, we choose the co-ordinates

$$H\left(k, l\right) = \begin{cases} \frac{1}{2}, & (k, l) = (0, 0) \\ \frac{1}{16}, & \left|k\right|, \left|l\right| \leq 1, (k, l) \neq (0, 0) \\ 0, & \text{else} \end{cases},$$

that is, the pixels of the blurred image are a mixture of the surrounding 3×3 pixels; $\mathcal{H}_s$ is this 3×3 square centered at s. From (2.1) and the definition of $\mathcal{H}_s$ we see that $\Phi_s = \nu_s$ depends only on g_s and $\{f_t \mid t \in \mathcal{H}_s\}$. Since H is linear and therefore shift-invariant we have $\mathcal{H}_{r+s} = \mathcal{H}_r + s$ as long as none of these crosses the image boundaries, that is, $\mathcal{H}_r \subseteq Z_m$ and $s + r \in Z_m$. To avoid problems at the edges of the image we define $\mathcal{H}_r + s = \{h + s \mid h \in \mathcal{H}_r\} \cap Z_m$. Further, we assume that H is symmetric, that is, $r \in \mathcal{H}_0$ iff $-r \in \mathcal{H}_0$. All these properties are reasonable as H is a blurring matrix and hence should operate equally on different spots of the image and independent of symmetries.

Lemma 5.1. *The families* $\left(\mathcal{H}_s \setminus \{s\}\right)_{s \in Z_m}$ *and* $\left(\mathcal{H}^2_s \setminus \{s\}\right)_{s \in Z_m}$*, where* $\mathcal{H}^2$ *denotes the second-order system, that is,* $\mathcal{H}^2_s = \bigcup_{r \in \mathcal{H}_s} \mathcal{H}_r$*, are neighbourhood systems for* Z_m.

Proof. We have to show that $r \in \mathcal{H}_s$ iff $s \in \mathcal{H}_r$ for all $s \neq r$. Shift-invariance and symmetry straightforwardly yield $r \in \mathcal{H}_s$ iff $r - s \in \mathcal{H}_0$ iff $s - r \in \mathcal{H}_0$ iff $s \in \mathcal{H}_r$.

For the second-order system we have to show that $r \in \mathcal{H}_s^2$ iff $s \in \mathcal{H}_r^2$. Assume that $r \in \mathcal{H}_s^2$. Then there exists some $t_0 \in \mathcal{H}_s$ such that $r \in \mathcal{H}_{t_0}$. By the first part of the proof we also have $s \in \mathcal{H}_{t_0}$, and in particular $s \in \bigcup_{t \in \mathcal{H}_r} \mathcal{H}_t = \mathcal{H}_r^2$. □

Lemma 5.2. *Define* $\mathcal{N}^P = \left(\mathcal{N}_s^P\right)_{s \in \mathcal{S}}$ *where*

$$\mathcal{N}_s^P = \begin{cases} \mathcal{N}_s, & s \in D_m \\ \mathcal{N}_s \cup \mathcal{H}_s^2 \setminus \{s\}, & s \in Z_m \end{cases}.$$

Then $\mathcal{N}^P$ *is a neighbourhood system on* $\mathcal{S}$. *It is called posterior neigbourhood system.*

Proof. We will show that $s \in \mathcal{N}_t^P$ iff $t \in \mathcal{N}_s^P$. All cases where $s \in \mathcal{N}_s$ and $t \in \mathcal{N}_t$ follow directly from the fact that $\mathcal{N}$ is a neighbourhood system, as does the case $s \in \mathcal{H}_s^2$ and $t \in \mathcal{H}_t$. We will show that the two remaining cases never occur.

Case 1: $s \in D_m$ and $t \in Z_m \setminus \mathcal{N}_s$.

Assume $s \in \mathcal{N}_t^P = \mathcal{N}_t \cup \mathcal{H}_t^2 \setminus \{t\}$. Then we know that $s \in \mathcal{N}_t$ and therefore $t \in \mathcal{N}_s$, a contradiction.

On the other hand, assume $t \in \mathcal{N}_s^P = N_s \cup \mathcal{H}_s^2 \setminus \{s\}$. Therefore $t \in \mathcal{H}_s^2 \setminus \{s\}$, and, by Lemma 5.1, $s \in \mathcal{H}_t^2 \setminus \{t\}$, a contradiction as well.

Case 2: $s \in Z_m$ and $t \in Z_m \setminus \mathcal{N}_s$.

Assume $s \in \mathcal{N}_t^P = \mathcal{N}_t \cup \mathcal{H}_t^2 \setminus \{t\}$. The case $s \in \mathcal{H}_t^2 \setminus \{t\}$ is already covered, so assume $s \in \mathcal{N}_t$. Then $t \in \mathcal{N}_s$, a contradiction.

Now assume $t \in \mathcal{N}_s^P = \mathcal{N}_s \cup \mathcal{H}_s^2$. Therefore $t \in \mathcal{H}_s^2$, and, as $\mathcal{H}^2$ is a neighbourhood system, $s \in \mathcal{H}_t^2$. This is one of the covered cases. □

Theorem 5.3. *For given data* g *the posterior distribution* $\mathrm{P}\left(X = \omega \mid G = g\right)$ *is Gibbsian relative to* $\left(\mathcal{S}, \mathcal{N}^P\right)$ *with the energy function*

$$U^P\left(\omega\right) = U\left(\omega\right) + \left\|M - \Phi\left(g, \phi\left(H\left(f\right)\right)\right)\right\|^2 / 2\sigma^2, \tag{5.1}$$

where $M \in \mathbb{R}^{Z_m}$ is the matrix that is μ everywhere. Recall that μ and σ^2 are mean and variance of N.

Proof. The proof can be found in Section VIII of [GG]. □

Using the algorithm from section 4 we can now restore degraded images. A computational problem, however, is the minimisation of U^P. Since Ω is too large to compute it exactly, we must at this point make use of another stochastic method, the Metropolis algorithm (see [Gam] or [Mat]), which can approximately optimise U^P. Further, the algorithm can be speeded up on a parallel system when appointing sites to different processors. In the optimal case we would have one processor per site. Geman and Geman have done experiments with degraded images with remarkably good success, which can be found in [GG].

6. The Gibbs Sampler in the General Case and Related Algorithms

Until now we had the case that the conditional distributions $p\left(\omega_i \mid \omega_j, j \neq i\right)$ were Gibbsian. We now want to generalise the Gibbs sampler for more arbitrary distributions. Further, we introduce the data-augmentation algorithm by [TW] and the substitution sampling algorithm by [GS].

6.1. The Data-Augmentation Algorithm.

This algorithm is based on the basic equations

$$p\left(\theta \mid y\right) = \int p\left(\theta \mid z, y\right) p\left(z \mid y\right)\, dz \tag{6.1}$$

and

$$p\left(z \mid y\right) = \int p\left(z \mid \phi, y\right) p\left(\phi \mid y\right)\, d\phi. \tag{6.2}$$

Substituting (6.2) into (6.1) yields

$$p\left(\theta \mid y\right) = \int K\left(\theta, \phi\right) p\left(\theta \mid y\right)\, d\phi, \tag{6.3}$$

with

$$K(\theta,\phi)=\int p(\theta\mid z,y)\,p(z\mid\phi,y)\;dz.$$

Let T be an integral operator defined by

$$Tf=\int K(\cdot,\phi)\,f(\phi)\;d\phi.$$

It was shown by [TW] that under mild conditions which usually hold in practical applications we can solve 6.3 by choosing some initial approximation $p_0(\theta\mid y)$ and successively calculating

$$p_{i+1}(\theta\mid y)=(Tp_i)(\theta\mid y).$$

This integral usually cannot be solved analytically. Hence we use the Monte Carlo method to calculate it in the algorithm.

Therefore the data-augmentation algorithm works as follows:

(1) Initialise $p_0(\theta\mid y)$. Set $i=1$.

(2) Generate a sample $z^{(1)},\cdots,z^{(m)}$ from $p_{i-1}(\theta\mid y)$

(3) Set
$$p_i(\theta\mid y)=\frac{1}{m}\sum_{j=0}^{m}p\left(\theta\mid z^{(j)},y\right).$$

(4) increase i by 1 and continue at step 2.

The proof for the algorithm can be found in [TW].

To illustrate the data-augmentation algorithm we examine an example on genetic linkage, used by both [Lee] and [TW]. Assume that 197 animals are distributed multinomially into four categories $y=(y_1,y_2,y_3,y_4)=(125,18,20,34)$ with cell probabilities

$$\left(\frac{1}{2}+\frac{\theta}{4},\frac{(1-\theta)}{4},\frac{(1-\theta)}{4},\frac{\theta}{4}\right).$$

To illustrate the algorithm, we augment the data y by splitting the first cell into two, having cell probabilities $1/2$ and $\theta/4$. Therefore the augmented data set is given by $x = (x_1, x_2, x_3.x_4, x_5)$, where $x_1 + x_2 = y_1$, $x_3 = y_2$, $x_4 = y_3$ and $x_5 = y_4$. Hence we have the likelihood function

$$p(y \mid \theta) \propto (2+\theta)^{y_1} (1-\theta)^{y_2+y_3} \theta^{y_4},$$

and for the augmented data the much simpler likelihood

$$p(x \mid \theta) \propto \theta^{x_2+x_5} (1-\theta)^{x_3+x_4} .$$

For the prior distribution of θ we choose a $Be(1,1)$ distribution or, what is the same, a $U[0,1]$ distribution. For the posterior distribution we then get

$$p(\theta \mid x) \propto p(\theta) p(x \mid \theta) \propto \theta^{x_2+x_5+1} (1-\theta)^{x_3+x_4+1} ,$$

a Beta distribution as well. Therefore the algorithm has to be implemented as follows:

- Obtain $\theta^{(i)}$ from the current estimate of $p(\theta \mid y)$ for $1 \leq i \leq m$.
- For each draw generate $x_2^{(i)}$ from a $b\left(y_1, \theta^{(i)} / \left(\theta^{(i)}+2\right)\right)$ distribution to obtain the augmented data.
- Set the posterior of θ equal to a mixture of the obtained Beta distributions, that is,
$$p(\theta \mid y) = \frac{1}{m} \sum_{i=1}^{m} Be\left(x_2^{(i)} + x_5 + 1, x_3 + x_4 + 1\right)(\theta) .$$
- Repeat this process with the new estimate.

The R source code for this example can be found in section A.1. Plots of the estimates after 1, 10 and 50 executions of the algorithm can be seen in Figures 3, 4 and 5. These figures suggest that after 50 executions the estimate of the posterior is close enough to the true posterior.

The algorithm can be further improved by initially choosing m small and gradually increasing as the approximated distribution approaches the true one [TW].

If we take three random variables instead of two we can write, analogous to (6.1) and (6.2),

$$p(\theta \mid y) = \int\int p(\theta, x \mid z, y)\, p(z \mid y)\; dz\, dx, \tag{6.4}$$

$$p(z \mid y) = \int\int p(z, \sigma \mid x, y)\, p(x \mid y)\; dx\, d\sigma \tag{6.5}$$

and

$$p(x \mid y) = \int\int p(x, \phi \mid \rho, y)\, p(\rho \mid y)\; d\rho\, d\phi. \tag{6.6}$$

Proceeding similar as before we can substitute (6.6) into (6.5), and this new expression into (6.4). This provides us with an analogous, but much more complicated, fixed point equation to (6.3) and a new integral operator, and the convergence theorems from [TW] hold. Similarly it is possible to develop algorithms for any finite number of random variables.

6.2. **The Substitution Sampling Algorithm.** As in the section before, first look at the case of two random variables. Assume the conditional densities $p(x \mid z, y)$ and $p(z \mid x, y)$ are known. Choose an arbitrary prior density $p_0(x \mid y)$ and draw a sample $x^{(0)}$ from it. Since $p\left(z \mid x^{(0)}, y\right)$ is available, draw a sample $z^{(1)}$ from it, and a sample $x^{(1)}$ from $p_1(x \mid y) = p\left(x \mid z^{(1)}, y\right)$. We repeat this procedure and get a sequence $\left(x^{(i)}, z^{(i)}\right)_{i\in\mathbb{N}}$. Now we use the fact that by (6.2)

$$p_i(z \mid y) = \int p(z \mid x, y)\, p_i(x \mid y)\; dx = \\ \int p(z \mid x, y)\, p\left(x \mid z^{(i-i)}, y\right)\; dx = \int p(z \mid x, y)\, p_{i-1}(x \mid z, y)\; dx$$

together with (6.1) yielding

$$p_i(x \mid y) = \int p(x \mid \phi, y)\, p_i(\phi \mid y)\; d\phi = \\ \int K(x, \phi)\, p_{i-1}(\phi \mid y)\; d\phi = (Tp_{i-1})(x \mid y)\,.$$

Therefore this generation scheme converges according to the theorems from [TW]. For a natural number m generate m iid sequences like this. We call this scheme **substitution sampling algorithm**.

When programming the substitution sampling algorithm we again use the Monte Carlo integration and get the new estimates by

$$p_i(x \mid y) = \frac{1}{m} \sum_{j=1}^{m} p\left(x \mid z_j^{(i)}, y\right)$$

and

$$p_i(z \mid y) = \frac{1}{m} \sum_{j=1}^{m} p\left(z \mid x_j^{(i-1)}, y\right).$$

Similar to the data-augmentation algorithm we can extend the substitution sampling algorithm to more than two random variables, see [TW].

6.3. **The Gibbs Sampler.** We have an unknown distribution $p(\theta_1, \ldots, \theta_d)$ with known local characteristics $p_i(\theta_i) = p(\theta_i \mid \theta_j, j \neq i)$ with $1 \leq i \leq d$. We set an initial value θ_0 as an arbitrary estimation of the mode of p and calculate θ_i for $i \in \mathbb{N}$ according to the following scheme:

(1) Set $j = 0$ and set initial values $\theta^{(0)} = \left(\theta_1^{(0)}, \ldots, \theta_d^{(0)}\right)$.

(2) For $1 \leq i \leq n$ obtain samples

$$\theta_i^{(j)} \sim p\left(\theta_1 \mid \theta_1^{(j)}, \ldots, \theta_{i-1}^{(j)}, \theta_{i+1}^{(j-1)}, \ldots, \theta_d^{(j-1)}\right)$$

from the conditional distributions and, therefore, a new value $\theta^{(j)}$.

(3) Increase j by 1 and continue with step 2.

Clearly every site is visited infinitely often. As a neighbourhood system we take the trivial one $\mathcal{N}_i = \{1, \ldots, i-1, i+1, \ldots, d\}$, so we have an MRF, and therefore $p(\theta)$ is Gibbsian. So we can use the Relaxation Theorem 4.1 to verify that the Gibbs sampling algorithm works.

If we want to obtain not just the modes, but the marginal densities, we must make use of the same method as before to run the process m times separately and calculate the estimates by

$$p\left(\theta_{i_0}\right)=\frac{1}{m}\sum_{k=1}^{m}p\left(\theta_{i_0}\mid\theta_i^{(j)},i\neq i_0\right).$$

Definition 6.1. If $X\sim\Gamma_{a,b}$ is gamma distributed with parameters a and b, then we call $X^{-1}\sim\bar{\Gamma}_{a,b}$ inverse gamma distributed with parameters a and b.

As an example for how the Gibbs sampler works we will look at a Poisson process with a change point, an example first given in [CGS] and quoted in [Lee] and [Gam]. Given is a sample $y=(y_1,\dots,y_n)$ from a Poisson process with a change point, that is $y_i\sim Poi\,(\lambda_1)$ for $1\leq i\leq k$ and $y_i\sim Poi\,(\lambda_2)$ for $1+k\leq i\leq n$ for unknown k, λ_1 and λ_2. As independent prior distributions we choose k Laplace distributed on $\{1,\dots n\}$, $\lambda_1\sim\Gamma_{a_1,b_1}$ and $\lambda_2\sim\Gamma_{a_2,b_2}$ gamma distributed. As a third stage in this hierarchical model we say that $b_1\ \sim\bar{\Gamma}_{c_1,d_1}$ and $b_2\ \sim\bar{\Gamma}_{c_2,d_2}$ are inverse gamma distributed, and a_1, a_2, c_1, c_2, d_1 and d_2 are known. From [CGS] we get the conditional distributions

$$\lambda_1\mid y,\lambda_2,b_1,b_2,k\sim\Gamma_{a_1+\sum_{i=1}^{k}y_i,b_1/(kb_1+1)},\tag{6.7}$$

$$\lambda_2\mid y,\lambda_1,b_1,b_2,k\sim\Gamma_{a_2+\sum_{i=k+1}^{n}y_i,b_2/((n-k)b_2+1)},\tag{6.8}$$

$$b_1\mid y,\lambda_1,\lambda_2,b_2,k\sim\bar{\Gamma}_{a_1+c_1,d_1((\lambda_1d_1+1)},\tag{6.9}$$

$$b_2\mid y,\lambda_1,\lambda_2,b_1,k\sim\bar{\Gamma}_{a_2+c_2,d_2((\lambda_2d_2+1)}\tag{6.10}$$

and

$$p\left(k\mid y,\lambda_1,\lambda_2,b_1,b_2\right)=\frac{e^{(\lambda_2-\lambda_1)k}\left(\lambda_1/\lambda_2\right)^{\sum_{i=1}^{k}y_i}}{\sum_{k'=1}^{n}e^{(\lambda_2-\lambda_1)k'}\left(\lambda_1/\lambda_2\right)^{\sum_{i=1}^{k'}y_i}}.\tag{6.11}$$

Note that the first factor in 6.11 can be very large, and the second one very small. As expected, testing formula 6.11 gave some enormous numerical errors, so that we rewrite it to the more expensive, but also numerically more accurate form

$$p\left(k \mid y, \lambda_1, \lambda_2, b_1, b_2\right) =$$

$$\left(\sum_{k'=1}^{n} \exp\left(\left(\lambda_2 - \lambda_1\right)\left(k' - k\right) + \left(\sum_{i=1}^{k'} y_i - \sum_{i=1}^{k} y_i\right) \log\left(\lambda_1/\lambda_2\right)\right)\right)^{-1}. \quad (6.12)$$

An example for a data set are the coal-mining disasters in Britain during the years 1851-1962 from [MPW] and corrected by [Jar]. The complete data can be found in Table 1.

So we have $n = 112$, and we choose $a_1 = a_2 = 1/2$, $c_1 = c_2 = 0$ and $d_1 = d_2 = 1$. Let $m = 100$ and iterate the algorithm 15 times. From this parameters we develop the R programme seen in Appendix A.2. This programme gives us an approximation of the expected value $Ek \doteq 47.85836$, which is the year 1897.86 or 9 November 1897. This is a slightly different result than the one from [Lee] and [CGS], who both get values between late 1889 and early 1892. [Lee] suggests that the change could be a result of the Coal Mines Regulation Act which came into force on 1 May 1888. Looking at the approximation of the marginal density in Figure 6 could give a hint in this matter. Apparently there are two peaks in the density, one of which is around 1889, the other one around 1948. That could mean that we have two change points in the data. The second one might possibly result from the radical social reforms introduced by the Labour government 1945-1951 and push the estimate for the first point slightly into the future. The question arises why [Lee] and [CGS] did not detect this second peak. This might be due to [Lee]'s simplification of the model into a two-stage hierarchical one and [CGS]'s numerical instability in their implementation, see equations 6.11 and 6.12.

7. CONCLUSION

We have now constructed the Gibbs sampler and seen some examples. However, we have said very few about convergence and how fast the algorithm converges. Most of the recent research on the Gibbs sampler has been concentrating on the

Year	Count	Year	Count	Year	Count	Year	Count
1851	4	1879	3	1907	0	1935	2
1852	5	1880	4	1908	3	1936	1
1853	4	1881	2	1909	2	1937	1
1854	1	1882	5	1910	2	1938	1
1855	0	1883	2	1911	0	1939	1
1856	4	1884	2	1912	1	1940	2
1857	3	1885	3	1913	1	1941	4
1858	4	1886	4	1914	1	1942	2
1859	0	1887	2	1915	0	1943	0
1860	6	1888	1	1916	1	1944	0
1861	3	1889	3	1917	0	1945	0
1862	3	1890	2	1918	1	1946	1
1863	4	1891	2	1919	0	1947	4
1864	0	1892	1	1920	0	1948	0
1865	2	1893	1	1921	0	1949	0
1866	6	1894	1	1922	2	1950	0
1867	3	1895	1	1923	1	1951	1
1868	3	1896	3	1924	0	1952	0
1869	5	1897	0	1925	0	1953	0
1870	4	1898	0	1926	0	1954	0
1871	5	1899	1	1927	1	1955	0
1872	3	1900	0	1928	1	1956	0
1873	1	1901	1	1929	0	1957	1
1874	4	1902	1	1930	2	1958	0
1875	4	1903	0	1931	3	1959	0
1876	1	1904	0	1932	3	1960	1
1877	5	1905	3	1933	1	1961	0
1878	5	1906	1	1934	1	1962	1

Table 1. British coal-mining disasters 1851-1962

improvement of convergence. Improvement can be gained by different strategies for forming the sample, visiting and updating the different sites, arranging the components of θ into blocks and several other strategies, see the references in [Gam]. In [BBDS] A. Gelman and D. Rubin proved that it is not possible to get precise results from a single sample series, but that for fast convergence we always need to run several processes simultaneously. The title of this article gives a good summary about its main result: "A Single Series from the Gibbs Sampler Provides a False Sense of Security". [RS] provide us with several good results about convergence optimisation. They compared different random and non-random updating strategies and found out optimal strategies for different types of applications.

The Gibbs sampler can be used in quite a few different applications. We have already seen its usefulness in digital image processing and change point analysis. It is still possible to apply the Gibbs sampler if we have some missing data in the data that we want to analyse for change points, see for example [CGS], who used the algorithm on the coal mining data with 20% removed. The Gibbs sampler can often be applied in hierarchical and dynamic models, see [Gam].

Appendix A. Source Code and Output

This R source code can also be found on my webpage `http://maths.vic-fontaine.de/` on the Internet.

A.1. The Genetic Linkage Example.

```
#
# linkage.r (Genetic Linkage Example)
#

# Start algorithm
y <- c(125, 18, 20, 34)
# input data y
integral <- .2357695165e29
# The integral was calculated beforehand using MAPLE
# integral:= int((2+x)^125*(1-x)^(18+20)*x^34, x=0..1);
true <- function(x)((2+x)^y[1]*(1-x)^(y[2]+y[3])*x^y[4]/integral)
# calculate the true posterior
m <- 1600
# take 1600 samples
theta <- runif(m, min=0, max=1)
# get samples from prior distribution
x2 <- rbinom(m, y[1], theta/(theta+2))
# augmentation
posterior <- function(x)(1/m*sum(dbeta(x, x2+y[4], y[2]+y[3])))
# estimate posterior distribution
datax <- (0:1000)
for (i in 1:1001) datax[i] <- (i-1)/1000
datay <- (0:1000)
for (i in 1:1001) datay[i] <- posterior((i-1)/1000)
```

```
x11(record=T)
plot(datax, datay, ylim=c(0,8), type="l",
     xlab="theta", ylab="density", main=1)
# plot the estimate
curve(true, 0, 1, n=1001, add=TRUE)
# plot the true posterior to compare it to the estimate

for (j in 1:49)
# execute the algorithm 49 more times
{
    decomp <- sample(m, m, replace=TRUE)
    # decomposite the estimate according to [KG] section 6.4.2
    theta <- (0:1000)
    for (i in 1:1001) theta[i] <-
         rbeta(1, x2[decomp[i]]+y[4], y[2]+y[3])
    x2 <- rbinom(m, y[1], theta/(theta+2))
    if ((j==9)|(j==24)|(j==49))
    {
        posterior <-
             function(x)(1/m*sum(dbeta(x, x2+y[4], y[2]+y[3])))
        for (i in 1:1001) datay[i] <- posterior((i-1)/1000)
        x11(record=T)
        plot(datax, datay, ylim=c(0,8), type="l",
             xlab="theta", ylab="density", main=j+1)
        curve(true, 0, 1, n=1001, add=TRUE)
    }
}
```

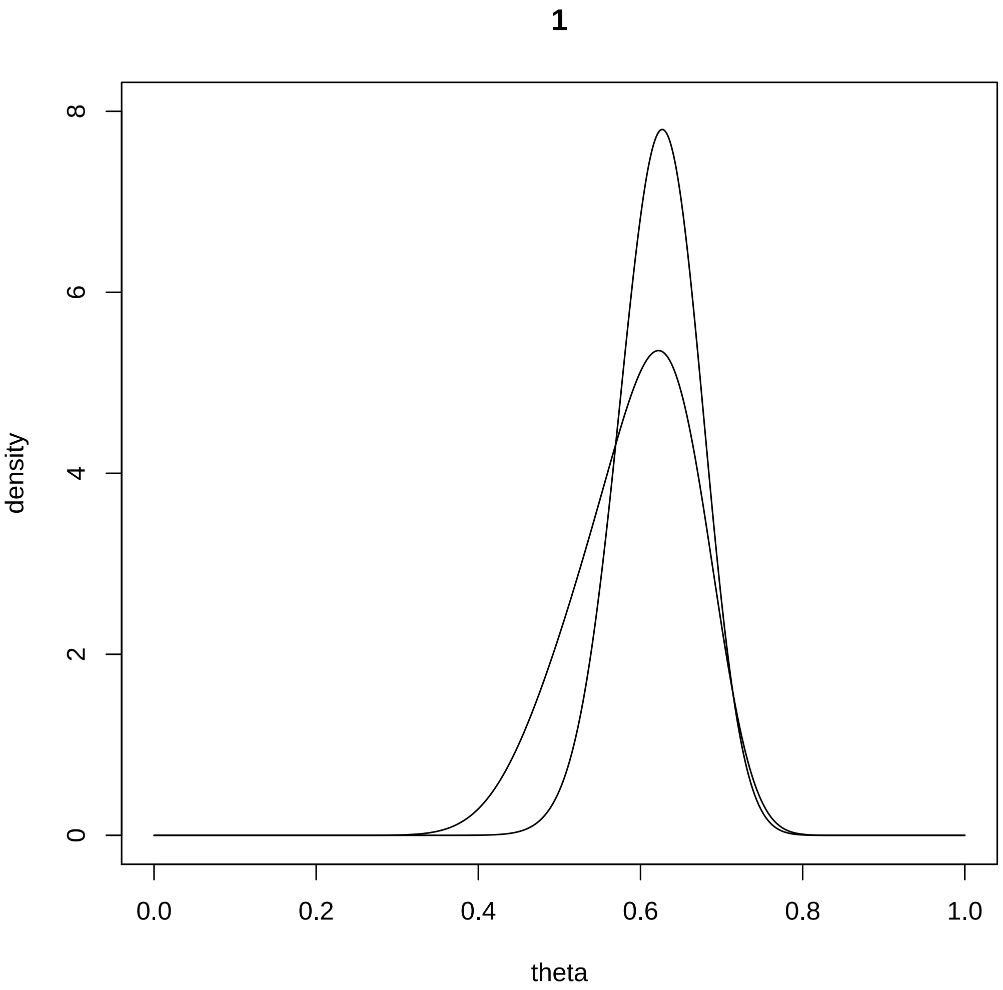

Figure 3. Estimate of the posterior density after 1 execution of the data-augmentation algorithm for the data $(125, 18, 20, 34)$

.

A.2. **The Coal-Mining Disasters Example.**

```
#
# mining.r (Coal-Mining Disasters Example)
#

# set constants, parameters and data
```

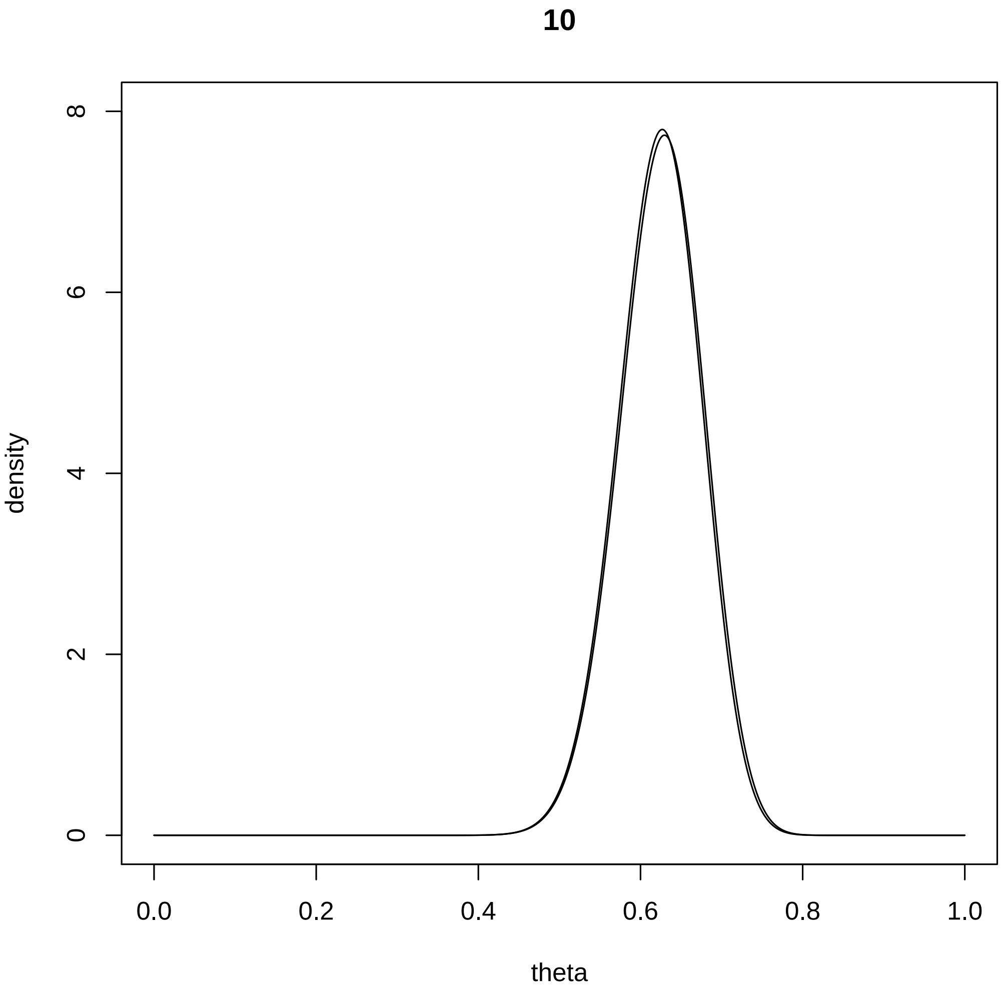

Figure 4. Estimate of the posterior density after 10 executions of the data-augmentation algorithm for the data $(125, 18, 20, 34)$

.

```
m <- 100
t <- 15
n <- 112
y <- c(4, 5, 4, 1, 0, 4, 3, 4, 0, 6, 3, 3, 4, 0, 2, 6, 3, 3,
    5, 4, 5, 3, 1, 4, 4, 1, 5, 5, 3, 4, 2, 5, 2, 2, 3, 4, 2,
    1, 3, 2, 2, 1, 1, 1, 1, 3, 0, 0, 1, 0, 1, 1, 0, 0, 3, 1,
```

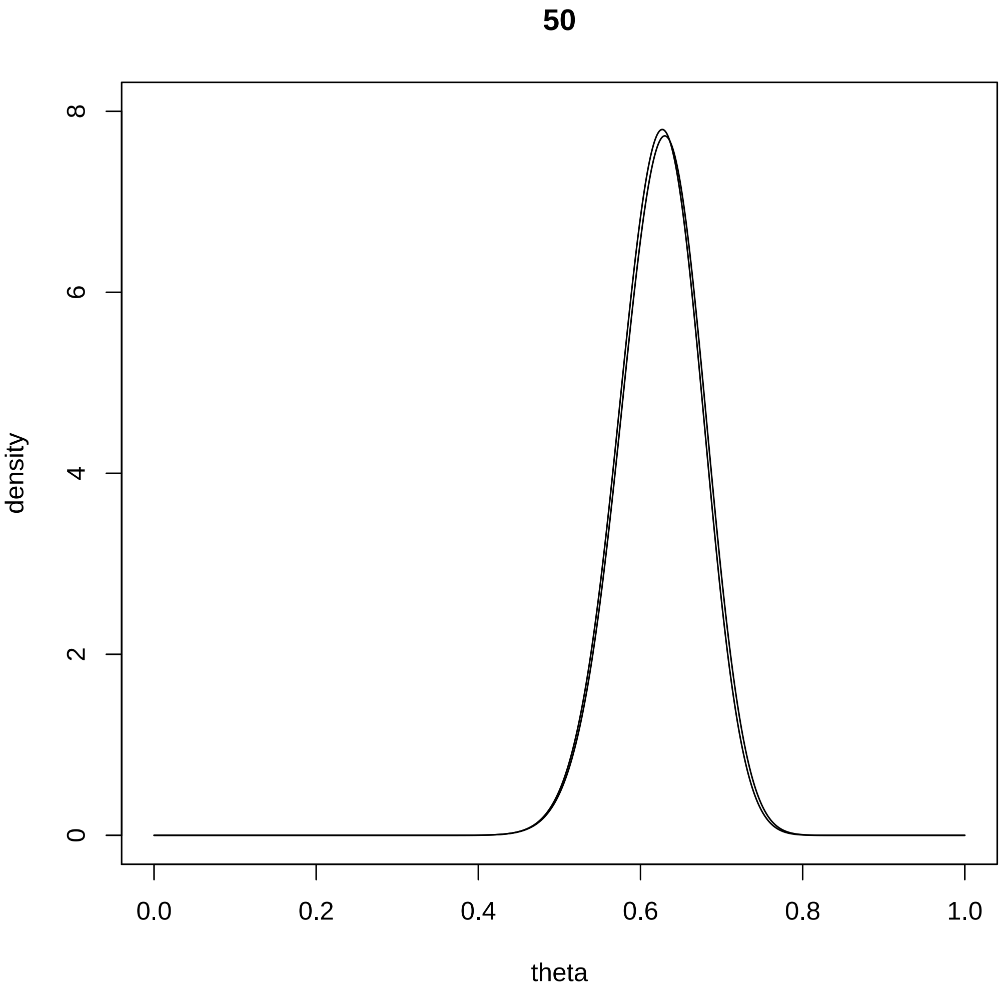

Figure 5. Estimate of the posterior density after 50 executions of the data-augmentation algorithm for the data $(125, 18, 20, 34)$

.

```
    0, 3, 2, 2, 0, 1, 1, 1, 0, 1, 0, 1, 0, 0, 0, 2, 1, 0, 0,
    0, 1, 1, 0, 2, 3, 3, 1, 1, 2, 1, 1, 1, 1, 2, 4, 2, 0, 0,
    0, 1, 4, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 1)
a1 <- 0.5
a2 <- 0.5
c1 <- 0
```

```
c2 <- 0
d1 <- 1
d2 <- 1
# calculate partial sums that are frequently needed
partsum <- (0:n)
for (i in 2:(n+1))
{
    partsum[i] <- partsum[i-1]+y[i-1]
}
# set initial data (arbitrary choices)
k <- (1:m)
l1 <- (1:m)
l2 <- (1:m)
b1 <- (1:m)
b2 <- (1:m)
for (i in 1:m)
{
    b1[i] <- 0.5
    b2[i] <- 0.5
    k[i] <- 56
}
# execute the algorithm t times
for (j in 1:t)
{
    # create m independent sequences
    for (l in 1:m)
    {
        # step for lambda_1
        l1[l] <- rgamma(1, a1+partsum[k[l]+1],
            b1[l]/(k[l]*b1[l]+1))
        # step for lambda_2
        l2[l] <-
            rgamma(1, a2+partsum[n+1]-partsum[k[l]+1],
                b2[l]/((n-k[l])*b2[l]+1))
        # step for b_1
        b1[l] <- 1/rgamma(1, a1+c1, d1/(l1[l]*d1+1))
        # step for b_2
        b2[l] <- 1/rgamma(1, a2+c2, d2/(l2[l]*d2+1))
        # step for k
        # calculate the inverses due to
        # overflows in the exponentials
        invers <- (1:n)
        for (i in 1:n)
        {
            invers[i] <- 0
            for (p in 1:n)
                invers[i] <- invers[i] +
                    exp((l2[l]-l1[l])*(p-i)+
                    (partsum[p+1]-partsum[i+1])*log(l1[l]/l2[l]))
        }
        # generate vector with probabilities
        prob <- 1/invers
        # sample
        k[l] <- sample(n,1,prob)
    }
}
# Monte Carlo Integration, here only for k
marg <- (1:n)
```

```
for (i in 1:n)
{
    marg[i] <- 0
}
for (l in 1:m)
{
    # use the same scheme as in the algorithm
    invers <- (1:n)
    for (i in 1:n)
    {
        invers[i] <- 0
        for (p in 1:n)
            invers[i] <- invers[i] +
                exp((l2[l]-l1[l])*(p-i)+
                (partsum[p+1]-partsum[i+1])*log(l1[l]/l2[l]))
    }
    prob <- 1/invers
    marg <- marg + prob/m
}
# calculate expected value
print(sum((1:n)*marg))
# plot the marginal distribution
plot((1:n),marg)
```

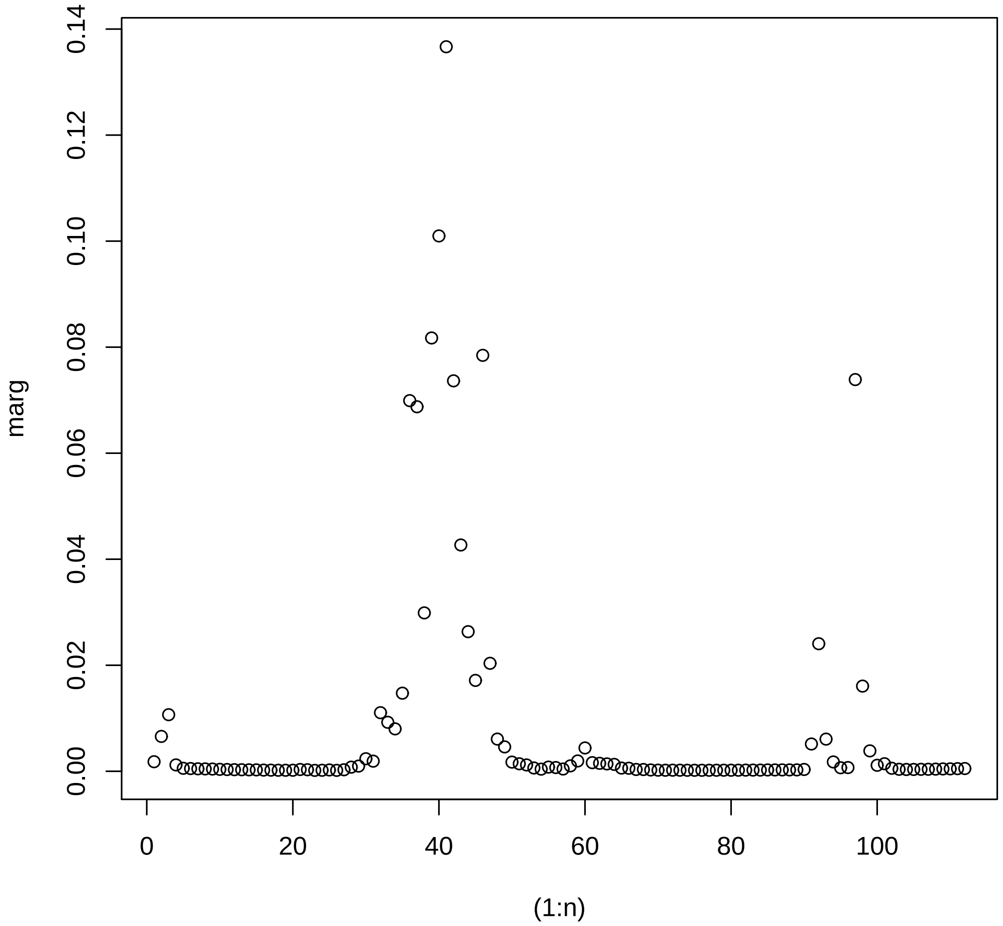

Figure 6. Estimate of the marginal density of k using the Gibbs sampler on the coal mining data.

References

[BBDS] J. Bernardo, J. Berger, A. Dawid and . Smith, *Bayesian Statistics 4* (Oxford University Press 1992)

[Bes] J. Besag, *Spatial Interaction and the Analysis of Lattice Systems* (Journal of the Royal Statistical Society, 36, pp 192-326, 1974)

[CGS] B. Carlin, A. Gelfand and A. Smith, *Hierarchical Bayesian Analysis of Changepoint Problems* (Applied Statistics, 41, pp 389-405, 1992)

[Gam] D. Gamerman, *Markov Chain Monte Carlo* (Chapman & Hall 1997)

[GG] S. Geman and D. Geman, *Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images* (IEEE Transactions on Pattern Analysis and Machine Intelligence, 6, pp 721-741, 1984)

[Gri] G. Grimmett, *A Theorem about Random Fields* (Bulletin of the London Mathematical Society, 5, pp 81-84, 1973)

[GS] A. Gelfand and A. Smith, *Sampling-Based Approaches to Calculating Marginal Densities* (Journal of the American Statistical Association, 85, pp 398-409, 1990)

[Jar] R. Jarrett, *A note on the Intervals between Coal-Mining Disasters* (Biometrika, 66, pp 191-193, 1979)

[KG] W. Kennedy and J. Gentle, *Statistical Computing* (Dekker 1980)

[KS] R. Kindermann and J. Snell, *Markov Random Fields and their Applications* (American Mathematical Society 1980)

[Lee] P. Lee, *Bayesian Statistics* (Arnold 1997)

[Mat] R. Mathar, *Zufallsgesteuerte Optimierungsverfahren* (lecture notes, RWTH Aachen summer semester 2000)

[MPW] B. Maguire, E. Pearson and A. Wynn, *The Time Intervals between Industrial Accidents* (Biometrika, 38, pp 168-180, 1952)

[Pre] C. Preston, *Gibbs States on Countable Sets* (Cambridge University Press 1974)

[RS] G. Roberts and S. Sahu, *Updating Schemes, Correlation Structure, Blocking and Parametrisation for the Gibbs Sampler* (Journal of the Royal Statistical Society, 59, pp 291-317, 1997)

[TW] M. Tanner and W. Wong, *The Calculation of Posterior Distributions by Data Augmentation* (Journal of the American Statistical Association, 82, pp 528-540, 1987).

E-mail address: jk133