A Guide to Exact Simulation

Xeni K. Dimakos[1]
[1]Department of Mathematics, University of Oslo, Box 1053 Blindern, 0316 Oslo, Norway

Summary

Markov Chain Monte Carlo (MCMC) methods are used to sample from complicated multivariate distributions with normalizing constants that may not be computable and from which direct sampling is not feasible. A fundamental problem is to determine convergence of the chains. Propp & Wilson (1996) devised a Markov chain algorithm called Coupling From The Past (CFTP) that solves this problem, as it produces exact samples from the target distribution and determines automatically how long it needs to run. Exact sampling by CFTP and other methods is currently a thriving research topic. This paper gives a review of some of these ideas, with emphasis on the CFTP algorithm. The concepts of coupling and monotone CFTP are introduced, and results on the running time of the algorithm presented. The interruptible method of Fill (1998) and the method of Murdoch & Green (1998) for exact sampling for continuous distributions are presented. Novel simulation experiments are reported for exact sampling from the Ising model in the setting of Bayesian image restoration, and the results are compared to standard MCMC. The results show that CFTP works at least as well as standard MCMC, with convergence monitored by the method of Raftery & Lewis (1992, 1996).
Key words: Coupling from the past; Efficient sampling; Exact simulation; Fill's algorithm; Forward coupling; Ising model; MCMC;

1 Introduction

Markov Chain Monte Carlo (MCMC) is an iterative algorithm used when direct sampling from the distribution of interest π is not feasible. This is typically the case in image analysis, spatial statistics and graphical and Bayesian hierarchical modeling where models are often complex, multivariate and difficult to normalize. Instead, an ergodic Markov chain with the target distribution π as stationary distribution is devised. Starting from an arbitrary initial state, the chain is run until it is believed to be close to equilibrium and the final state considered as a sample from π. Assisted by methods for convergence diagnostics (Cowles & Carlin, 1996; Brooks & Roberts, 1999), the MCMC user has to determine how long to run the simulation. However, as the chain is simulated in finite time and can not be started in π, the method always produces an approximate sample.

This paper assumes the reader to be familiar with the idea of MCMC simulation and the most common samplers such as the Metropolis-Hastings algorithm (Metropolis et al.,

1953; Hastings, 1970) and the Gibbs sampler (Geman & Geman, 1984). We will focus on the recently developed methods for exact simulation, which organize MCMC simulations cleverly to produce exact samples from the distribution of interest. The purpose of this paper is to serve as an introduction with pointers to further reading. We also report on our experiments with exact simulation and comparisons with standard MCMC.

Exact simulation is not new. For simple distributions exact samples are obtained by inversion, tabulation, transformation and composition methods (Ripley, 1987, Chapter 3). Rejection sampling (Ripley, 1987, pp. 60–63) is in principle possible for any distribution and does not demand computation of complicated normalizing constants. For practical purposes, rejection sampling is fast enough if we are able to sample easily from a proposal envelope distribution that is reasonably close to the target distribution, something which is difficult to design in multivariate settings.

Asmussen et al. (1992) were the first to show that it is possible to obtain exact samples from a finite chain, provided the chain is irreducible, the state space is finite and the number of states known. Lovász & Winkler (1995) showed that this was computationally feasible, but that the running time was enormous for chains with more than a few states. In contrast, the methods that we present in this paper works for complicated distributions and are fast enough to be often useful in practice.

In Section 2 we introduce the idea of coupling and show how forward coupling produces biased samples. Section 3 is devoted to the original CFTP algorithm of Propp & Wilson (1996) and the simpler monotone CFTP that works under certain monotonicity requirements and Section 4 gives some results on the running time of the CFTP algorithm. An iid sample from the target distribution may be obtained by running independent CFTP runs, but this is computationally expensive. Section 5 includes a presentation of several sampling schemes that are more efficient, in the sense that they utilize more of the generated values. In Section 6 we present a simulation experiment for the Ising model, in which the running time of the CFTP algorithm is compared to standard MCMC with convergence determined by the method of Raftery & Lewis (1992, 1996). We conclude that CFTP is the faster method. Section 7 presents an extension of CFTP to continuous state spaces by Murdoch & Green (1998) and Green & Murdoch (1998), while Section 8 introduces the algorithm of Fill (1998) that is based on rejection sampling and has a different structure than CFTP. The paper ends with a discussion and a guide to sources of information on exact simulation available on the Internet.

2 Forward Coupling

Consider a Markov chain defined on a discrete finite state space $\mathcal{S}$, and suppose that two copies of the chain $\{\mathbf{X}^t\}_{t=0}^{\infty}$ and $\{\mathbf{Y}^t\}_{t=0}^{\infty}$are started from two different initial states. Assume that the chains are *coupled*. Coupling is described as a tool in probability theory in Lindvall (1992). For MCMC it was introduced by Johnson (1996), see also Frigessi & Den Hollander (1993). In the current context, we say that two chains are coupled if they use the same sequence of random numbers for the transitions. If the trajectories of the

chains meet at time T, the chains will merge and proceed together or *coalesce*, that is $\mathbf{x}^t = \mathbf{y}^t \; \forall t \geq T$, due to the coupling. If we similarly couple chains started from all states of $\mathcal{S}$, and wait until all the chains have coalesced, then the initialization bias has worn off. However, when the chains coalesce, they need not have reached the stationary distribution as the following two examples show. The first example, taken from Kendall (1998) and Fill (1998), involves a simple two state Markov chain, while the second example, from Häggström & Nelander (n.d.*b*), fits in the Gibbs sampler framework. For both examples it is convenient to consider the transitions of the chains via an update function. The update function is simply an expression for the transition mechanism that given the previous value of the chain and the necessary random numbers, produces the new state of the chain.

DEFINITION 1 The transitions of a Markov chain $\{\mathbf{X}^t\}_{t=0}^{\infty}$ with transition matrix $\mathbf{P}$ can be described by a deterministic *update function* ϕ by $\mathbf{X}^{t+1} = \phi(\mathbf{x}^t, \mathbf{U}^{t+1})$ if $\mathbf{P}(\mathbf{x}, \mathbf{y}) = P(\phi(\mathbf{x}, \mathbf{U}) = \mathbf{y}) \; \forall \, \mathbf{x}, \mathbf{y} \in \mathcal{S}$ for a vector of random numbers $\mathbf{U}$, or in the simplest case a single uniform random number.

EXAMPLE 1 (A 1-DIMENSIONAL TARGET π)
Assume that we have a Markov Chain on $\mathcal{S} = \{0, 1\}$ that from state 0 moves to 0 or 1 with probability 1/2 each, and from state 1 always moves to state 0. The transitions of the chain may be written in terms of the update function ϕ as $X^{t+1} = \phi(x^t, U^{t+1})$, where

$$\phi(0, U^t) = \begin{cases} 0 & \text{if } U^t \leq 1/2 \\ 1 & \text{if } U^t > 1/2 \end{cases} \qquad \text{and} \qquad \phi(1, U^t) = 0,$$

for uniform random numbers U^t. The stationary distribution of this chain is $\pi(0) = 2/3$ and $\pi(1) = 1/3$. Starting one chain in each of the two states and applying the same random numbers, it is easy to see that the chains *always* coalesce in state 0 (see Figure 1, left panel), and hence the distribution at the time of coalescence is not correct. $\diamond$

[Figure 1 about here.]

EXAMPLE 2 (A 2-DIMENSIONAL TARGET π)
Let $\mathbf{X} = (X_1, X_2)$ and $\mathcal{S} = \{(i, j) : i = 0, 1, 2; j = 0, 1, 2\}$ and take π to be uniformly distributed on the subset $\{(0, 0), (0, 1), (2, 1), (2, 2)\}$ for which the values of the two elements differ by at most 1. For this distribution π a Gibbs sampler that updates the elements in random order is expressed by the update function

$$(X_1^{t+1}, X_2^{t+1}) = \phi(\mathbf{x}^t, V^{t+1}, U^{t+1}) = \begin{cases} (X_1', x_2^t) & \text{if} \quad V^{t+1} \leq 1/2 \\ (x_1^t, X_2') & \text{if} \quad V^{t+1} > 1/2, \end{cases}$$

where

$$X_1' = \begin{cases} \min\{i \in \{0,2\} : \pi(i, x_2^t) > 0\} & \text{if} \quad U^{t+1} \leq 1/2 \\ \max\{i \in \{0,2\} : \pi(i, x_2^t) > 0\} & \text{if} \quad U^{t+1} > 1/2 \end{cases}$$

$$X_2' = \begin{cases} x_1^t & \text{if} \quad U^{t+1} \leq 1/2 \\ 1 & \text{if} \quad U^{t+1} > 1/2 \end{cases}$$

and V^t, U^t are independent uniform random numbers that determine which element that is updated and the new value for this element.

Assume that chains are started from $\{(0,0),(0,1),(2,1),(2,2)\}$. Figure 2 shows how the chains coalesce according to which element is updated. Careful inspection of the figure reveals that the 4 chains only coalesce in the states $(0,1)$ and $(2,1)$ with probability $1/2$ each. The reasoning is as follows.

(i) If X_1 is updated in the first iteration the chains always reduce to three chains, since the chains started in $(0,1)$ and $(2,1)$ coalesce with probability 1.

(ii) As long as X_1 is updated, these three chains never coalesce. The chains in $(0,0)$ and $(2,2)$ remain in these states, while the third chain moves between state $(0,1)$ and $(2,1)$ depending on the value of U.

(iii) The three chains reduce to two chains when for the first time X_2 is updated.

(iv) If X_2 is updated in the first iteration the number of chains always reduces to two chains, either in state $(0,0)$ and $(2,2)$ or in $(0,1)$ and $(2,1)$, depending on the value of U.

(v) Thus, either we start by updating X_1 or X_2, the chains reduce to two chains in $(0,0)$ and $(2,2)$ or in $(0,1)$ and $(2,1)$. After this, coalescence may only happen by updating X_1 when the chains are in $(0,1)$ and $(2,1)$.

◇

These examples show that coalescence of two forward coupled Gibbs sampler chains fails to identify when the chains reach stationarity.

[Figure 2 about here.]

3 Propp and Wilson's Coupling from the Past

3.1 Coupling from the Past

Propp & Wilson (1996) devised a surprisingly simple algorithm which uses coupling from the past, also called backward coupling, to produce exact samples from a stationary distribution. Consider a finite state space $\mathcal{S}$ with M states and an ergodic Markov chain

with limit distribution π that can be described by a deterministic update function ϕ as in Definition 1.

Instead of running the chain from the present and into the future, we will run it from the past to the present, that is, we assume that at time $-\infty$ one chain is started in each of the states of $\mathcal{S}$ and run to time 0. Also, we use the same sequence of random numbers for all these chains. Since the state space is finite and the chains are ergodic with the same unique stationary distribution π, all the chains will coalesce a.s. and be stationary by time 0. Of course, starting a chain in the infinite past is not possible, and the following strategy is applied instead. One chain is started in each state of $\mathcal{S}$ at time -1. For each such chain the same random vector $\mathbf{U}^0$ is applied. If the chains are not all coalesced at time 0, we repeat, but now the chains are started from -2. New random numbers $\mathbf{U}^{-1}$ determine the transitions from time -2 to -1. In the transitions from -1 to 0, the same random numbers are used as in the preceding trial. We continue going back in time in this manner, reusing the random numbers generated in the previous steps of the procedure, until the chains have all coalesced by time 0. Note that even if we only check for coalescence at time 0, the chains may have merged before this time point.

Let the mth state of $\mathcal{S}$ be denoted $\mathbf{x}_m$, $m = 1, \ldots, M$ and denote by $\mathbf{X}^{t_2}(t_1, \mathbf{x}_m)$ the state at time t_2 of a chain started in state $\mathbf{x}_m$ at time $t_1 < t_2$. The algorithm can then be expressed as follows.

ALGORITHM 1 (CFTP)

1. Set the starting value for the time to go back, $T_0 \leftarrow -1$.

2. Generate a random vector $\mathbf{U}^{T_0+1}$.

3. Start a chain in each state $\mathbf{x}_m$, $m = 1, \ldots, M$, of $\mathcal{S}$ at time T_0, and run the chains $\mathbf{X}^{t+1}(T_0, \mathbf{x}_m) = \phi(\mathbf{x}^t(T_0, \mathbf{x}_m), \mathbf{U}^{t+1})$ to time 0, $t = T_0, T_0 + 1, \ldots, -1$.

4. Check for coalescence at time 0, that is check if $\mathbf{X}^0(T_0, \mathbf{x}_m)$ occupy the same state $\forall m$. If so, this common value $\mathbf{X}^0$ is returned. Otherwise let $T_0 \leftarrow (T_0 - 1)$ and continue from step 2.

The following theorem from Propp & Wilson (1996) guarantees that the value $\mathbf{X}^0$ returned by Algorithm 1 is distributed according to the stationary distribution. We give our own version of the proof, other versions are found in Propp & Wilson (1996) and Häggström & Nelander (n.d.*b*).

THEOREM 1

(i) With probability 1 the CFTP algorithm (Algorithm 1) returns a value, and (ii) this value is a realization of a random variable distributed according to the stationary distribution of the Markov chain.

Proof:

(i) Since the chain is irreducible, there exists a time L such that there is a positive proba-

bility of going from state $\mathbf{x}_n$ to $\mathbf{x}_m$ in L steps, that is

$$P(\mathbf{X}^t(t-L,\mathbf{x}_n)=\mathbf{x}_m)>0 \tag{1}$$

$\forall$ states $\mathbf{x}_n,\mathbf{x}_m$. Let A^t_{t-L} be the event that the chains started from all states in $\mathcal{S}$ at time $t-L$ have all coalesced by time t and let the common state at time t be $\mathbf{x}'$. Then

$$A^t_{t-L}=\bigcap_{m=1}^{M}(\mathbf{X}^t(t-L,\mathbf{x}_m)=\mathbf{x}'), \tag{2}$$

and because the events in (2) are not disjoint and by (1) then $P(A^t_{t-L})>0\ \forall\, t$. Since the events $A^0_{-L}, A^{-L}_{-2L},\ldots$ are independent and have the same positive probability of occurrence, the algorithm terminates with probability 1. (ii) We then have that there exists a time $-T_*$ so that when starting from this time all trajectories have coalesced by time 0. If the event $A^0_{-T_*}$ is true, then also A^0_{-T} is true for $-T<-T_*$, and moreover $\mathbf{X}^0(-T_*,\mathbf{x}_m)=\mathbf{X}^0(-T,\mathbf{x}_n)\ \forall$ states $\mathbf{x}_n,\mathbf{x}_m$ because the same random numbers are applied. Since this holds for any $-T<-T_*$, also $A^0_{-\infty}$ is true, and thus $\mathbf{X}^0(-T_*,\mathbf{x}_m)\sim\pi$ since it is the state visited by an infinitely long trajectory of an ergodic Markov chain with stationary distribution π. The CFTP algorithm finds T_*, and the returned value has the correct distribution. □

Kendall & Thönnes (1998) described the CFTP algorithm as a *virtual simulation* from time $-\infty$, since it allows us to sample an infinitely long simulation by reconstructing it over a finite time interval. We may regard the method as a search algorithm that reconstructs the part of the infinite sequence it needs to obtain a correctly distributed sample.

Care must be taken when implementing the CFTP algorithm, so that when the chains are started from $-t_2<-t_1$ the same realizations of the random numbers $\mathbf{U}^{-t_1+1},\ldots,\mathbf{U}^0$ are used for the steps from $-t_1$ to time 0. In a practice this is done by either saving the random numbers as they are generated, or in order to save storage space, resetting the seed for the random number generator so that the same random numbers are produced.

In Algorithm 1 the times $-1,-2,-3,\ldots$ are successively tried as starting points for the chains. From the proof it is evident that Theorem 1 holds for any decreasing sequence of starting points and also for any value of T_0 in step 1. In many situations it would be highly inefficient to try all integer time points $t<0$. Propp & Wilson (1996) recommended to take simulation start times $-T_i=-2^i$, so that in each iteration the starting time is doubled. Propp & Wilson (1996) showed that this choice minimizes the worst-case number of steps and is close to minimizing the expected number of steps in the search. The argument is as follows. A natural choice for the sequence of starting points (all start times are of course negative, but we omit the sign in this discussion about the number of steps) is $T_1=rT_0, T_2=rT_1,\ldots$ for an initial value T_0 and ratio r. If M chains are started each time, the number of required steps is

$$\begin{aligned}S_{\text{tot}}&=MT_0+MrT_0+\cdots+Mr^kT_0\\&=MT_0\frac{r^{k+1}-1}{r-1}<\frac{r^2}{r-1}r^{k-1}MT_0\le\frac{r^2}{r-1}MT_{\text{cftp}}=S_{\text{up}},\end{aligned}$$

where T_{cftp} is the smallest time so that all the chains have coalesced by time 0, and where we assume that $T_0 \leq T_{\text{cftp}}$ and k is such that $r^k T_0 \geq T_{\text{cftp}}$. The smallest number of steps possible, $S_{\text{low}} = MT_{\text{cftp}}$, would emerge if we were able to guess exactly right and let $T_0 = T_{\text{cftp}}$. The ratio of the upper bound S_{up} and S_{low} is $r^2/(r-1)$ which is minimized by $r = 2$, and hence the doubling. Similar arguments show that to minimize the expected number of steps, we should let $r = e$. With this ratio, the expected number of steps is bounded above by MeT_{cftp}, while the similar bound with $r = 2$ is $M2.89T_{\text{cftp}}$. Thus, in this respect very little is lost by applying the simpler doubling of time.

An important feature of the CFTP algorithm is that the running time is a random variable that is typically difficult to predict prior to the run. Also, the coalescence time T_{cftp} and the returned value $\mathbf{X}^0$ are dependent. In fact, a CFTP run is determined by the sequence of random numbers only and the algorithm stops when it has reached T_{cftp} steps into the past, and then outputs the value at time 0. Thus, both the coalescence time and the sampled value are functions of the same random numbers. This means that an impatient user who aborts long runs will produce a biased sample, sampled from something like $\pi(\cdot \mid \text{short runs only})$. The problem is similar in classic MCMC sampling where there is a tendency to restart long runs that do not meet a chosen convergence criteria after a certain number of "unlucky" iterations. The method of Fill, reviewed in Section 8, like classical rejection sampling, is interruptible, so that runs may be aborted without biasing the samples.

Figure 1 illustrates the difference of forward coupling and CFTP in Example 1. Another nice illustration is provided by Kendall & Thönnes (1998) who compared forward simulation and CFTP for the Dead Leaves process. The two approaches are illustrated in an animated simulation on the website `http://www.warwick.ac.uk/statsdept/Staff/WSK/dead.html`.

3.2 Monotone CFTP

An unappealing feature of Algorithm 1 is the need to start a chain from every state for a possibly huge state space $\mathcal{S}$. However, if the state space and the sampler possess certain monotonicity properties, this can be avoided and the CFTP algorithm may be simplified considerably.

We first assume a partial order of the statespace $\mathcal{S}$. We say that $\mathcal{S}$ admits the natural componentwise partial order $\mathbf{x} \leq \mathbf{y}$ if $x_j \leq y_j$ for $j = 1, \ldots, N$ for $\mathbf{x}, \mathbf{y} \in \mathcal{S}$.

DEFINITION 2 A transition rule expressed by a deterministic update function ϕ as in Definition 1 is monotone with respect to the componentwise partial order if $\phi(\mathbf{x}, \mathbf{u}) \leq \phi(\mathbf{y}, \mathbf{u})\ \forall\, \mathbf{u}$ when $\mathbf{x} \leq \mathbf{y}$.

PROPOSITION 1
Denote by $\mathbf{x}_{min}$ *and* $\mathbf{x}_{max}$ *the minimum and maximum elements of* $\mathcal{S}$ *with the componentwise partial order, so that* $\mathbf{x}_{min} \leq \mathbf{x} \leq \mathbf{x}_{max} \forall \mathbf{x}$. *Assume that the update function* ϕ *of the Markov chain is monotone in the sense of Definition 2. Then we only need to consider the two*

chains started in $\mathbf{x}_{min}$ *and* $\mathbf{x}_{max}$ *in the CFTP algorithm, since all the other trajectories will be kept within these.*

When the CFTP algorithm can be modified so that only two chains need to be run, we shall call it monotone CFTP. Several models π and algorithms ϕ have been shown to fulfill these monotonicity requirements. Propp & Wilson (1996) considered the class of attractive spin systems from statistical physics. This is a subclass of MRF models that includes the random cluster model introduced by Fortuin & Kasteleyn (1972) which generalizes the Ising model and the Potts model. These models are difficult to sample from and MCMC algorithms must be used for this.

DEFINITION 3 Consider a finite set $\mathcal{V}$ of vertices or sites. *A spin system* is a configuration $\mathbf{X} = \{X_v; v \in \mathcal{V}\}$ where each element assigns a spin X_v to each site v. The spin X_v takes as value either $+1$ (up) or -1 (down).

The set of all possible spin configurations $\mathcal{S}$ then admits the componentwise partial order. The conditional distribution of assigning spin up to X_v in site v given the spins $\mathbf{x}_{-v}$ at all other sites is

$$\pi(X_v = 1|\mathbf{x}_{-v}) = \{1 + \frac{\pi(X_v = -1, \mathbf{x}_{-v})}{\pi(X_v = 1, \mathbf{x}_{-v})}\}^{-1}.$$

DEFINITION 4 When $\pi(X_v = 1|\mathbf{x}_{-v})$ is an increasing function in $\mathbf{x}_{-v}$, or equivalently in $\mathbf{x}$, in the componentwise partial order, the spin system π is called *attractive.*

Attractiveness implies that configurations where neighbouring sites tend to have the same spin are preferred.

The Gibbs sampler is particularly useful for sampling from spin models. As Theorem 2 states, the Gibbs sampler is monotone for attractive models, and hence monotone CFTP is possible.

THEOREM 2
The Gibbs sampler is monotone according to the componentwise partial order for attractive spin systems π*.*

Proof:
For spin systems, the Gibbs sampler may be defined through the update function ϕ given by

$$\begin{aligned}\mathbf{X}^{t+1} &= \phi(\mathbf{x}^t, v, U^{t+1}) \\ &= \begin{cases}(X_v = 1, \mathbf{x}^t_{-v}) & \text{if} \quad U^{t+1} \leq \pi(X_v = 1|\mathbf{x}^t_{-v}) \\ (X_v = -1, \mathbf{x}^t_{-v}) & \text{otherwise.}\end{cases}\end{aligned} \tag{3}$$

The site to be updated, v, is either chosen at random or found from a deterministic scanning rule. If the system is attractive then $\pi(X_v = 1|\mathbf{x}^t_{-v}) \leq \pi(Y_v = 1|\mathbf{y}^t_{-v})$ for $\mathbf{x}^t \leq \mathbf{y}^t$. Thus if $u \leq \pi(X_v = 1|\mathbf{x}^t_{-v})$ for a realization u of the random variable U^{t+1} then also

$u \leq \pi(Y_v = 1|\mathbf{y}^t_{-v})$ and hence $X_v^{t+1} = Y_v^{t+1} = 1$. If $\pi(X_v = 1|\mathbf{x}^t_{-v}) < u \leq \pi(Y_v = 1|\mathbf{y}^t_{-v})$ then $X_v^{t+1} = -1$ and $Y_v^{t+1} = 1$. Finally if $\pi(Y_v = 1|\mathbf{y}^t_{-v}) < u$ then $X_v^{t+1} = Y_v^{t+1} = -1$. In conclusion, $\phi(\mathbf{x}^t, u) \leq \phi(\mathbf{y}^t, u) \forall u$ when $\mathbf{x}^t \leq \mathbf{y}^t$ which is the definition of monotonicity. □

EXAMPLE 3 (MONOTONICITY OF THE GIBBS SAMPLER FOR THE ISING MODEL)
The Ising model was introduced in statistical physics as a model for spontaneous magnetization of ferromagnetic materials. The model describes a set of interacting magnets, sometimes in the presence of an external magnetic field.

We consider an Ising model on a finite two dimensional rectangular grid $\mathcal{V}$, where $\mathbf{X} = \{X_v; v \in \mathcal{V}\}$, $X_v \in \{-1, 1\}$ is a configuration of the spins. The interaction between sites (called pixels in image analysis applications that we will come back to in Section 6) is described by a first order neighbour system, so that the neighbours of a site are the sites above and below, and those to the right and to the left. (We assume free boundary conditions. For instance, in Figure 3 the central site has 4 neigbhour sites, the sites in the corners have 2, and the remaining sites have 3.) The distribution of the Ising model without external field and with a constant interaction β is

$$\pi(\mathbf{x}) = \exp\{\beta \sum_{i \sim j} x_i x_j\}/Z_\beta,$$

where $i \sim j$ indicates that i and j is a neighbouring pair and Z_β is the normalizing constant. For this model we have that

$$\pi(X_v = 1|\mathbf{x}_{-v}) = \{1 + \exp(-2\beta \sum_{j \sim v} x_j)\}^{-1}.$$

This quantity is increasing in $\mathbf{x}_{-v}$, i.e. attractive, only for $\beta > 0$ which induces clustering of sites with the same spin value. This model is called ferromagnetic, while the opposite case, $\beta < 0$, is referred to as the anti-ferromagnetic, anti-monotone or repulsive Ising model. Under the influence of an sitevarying external field $\boldsymbol{\alpha}$, the model admits the form

$$\pi(\mathbf{x}) = \exp\{\beta \sum_{i \sim j} x_i x_j + \sum_j \alpha_j x_j\}/Z_{\beta,\alpha}. \tag{4}$$

Since

$$\pi(X_v = 1|\mathbf{x}_{-v}) = \{1 + \exp(-2\beta \sum_{j \sim v} x_j - 2\alpha_v)\}^{-1},$$

is increasing in $\mathbf{x}_{-v}$, the Gibbs sampler is also monotone for the Ising model with an external field. ◇

When consulting the literature on exact simulation, the reader will find that most attention is given to the Gibbs sampler, which is a result of the important monotonicity property of this sampler. There is however nothing in the CFTP algorithm that says that

the chain has to be a Gibbs sampler. A popular alternative is the Metropolis-Hastings algorithm (Metropolis et al., 1953; Hastings, 1970) and the MCMC literature contains a lot of discussion on what chain is most suitable in various settings, see Tierney (1994) for a discussion. The Metropolis-Hastings algorithm is however not generally monotone for the componentwise partial order. As Theorem 3 shows, the Ising model is an exception when one site only is updated in each iteration.

THEOREM 3
The Metropolis-Hastings algorithm with uniform proposals that updates one site in each iteration is monotone with respect to the componentwise partial order for the ferromagnetic Ising model on a two-dimensional grid with or without a sitevarying external field.

Proof:
Let Z_v be a candidate value for site v in the Metropolis-Hastings algorithm, drawn uniformly from $\{-1, 1\}$. The update function is

$$\begin{aligned}\mathbf{X}^{t+1} &= \phi(\mathbf{x}^t, Z_v, U^{t+1}) \\ &= \begin{cases} (Z_v, \mathbf{x}^t_{-v}) & \text{if} \quad U^{t+1} \leq r(\mathbf{x}^t, z_v) \\ \mathbf{x}^t & \text{otherwise,} \end{cases}\end{aligned}$$

where $r(\mathbf{x}^t, z_v) = \pi(z_v, \mathbf{x}^t_{-v})/\pi(\mathbf{x}^t) = \exp\{(z_v - x^t_v)(\beta \sum_{j\sim v} x^t_j + \alpha_v)\}$. For realizations z_v and u of Z_v and U^{t+1} the algorithm is monotone if $\phi(\mathbf{x}^t, z_v, u) \leq \phi(\mathbf{y}^t, z_v, u)$ for $\mathbf{x}^t \leq \mathbf{y}^t$. When $z_v = 1$ ($z_v = -1$) this is equivalent to $r(\mathbf{x}^t, z_v) \leq r(\mathbf{y}^t, z_v)$ ($r(\mathbf{x}^t, z_v) \geq r(\mathbf{y}^t, z_v)$).

We first consider the case when $x^t_v = y^t_v$. If $z_v = 1$ then $z_v - x^t_v = z_v - y^t_v \geq 0$. Since $\sum_{j\sim v} x^t_j \leq \sum_{j\sim v} y^t_j$, this implies that $r(\mathbf{x}^t, z_v) \leq r(\mathbf{y}^t, z_v)$. Similarly if $z_v = -1$ we have that $z_v - x^t_v = z_v - y^t_v \leq 0$, and $r(\mathbf{x}^t, z_v) \geq r(\mathbf{y}^t, z_v)$. We then consider the case when $x^t_v < y^t_v$. If $z_v = 1$, then $z_v - y^t_v = 0$, and $r(\mathbf{y}^t, z_v) = 1$, and the proposal is always accepted. If $z_v = -1$, then $z_v - x^t_v = 0$, and $r(\mathbf{x}^t, z_v) = 1$. In both cases the order is preserved. This proof holds for all values of α_v and thus monotonicity for the Ising model with and without a sitevarying external field is proved. □

A componentwise partial ordering does not represent a full ordering of a state space $\mathcal{S}$. Given two arbitrary elements of $\mathcal{S}$, it will often not be possible to say which is the largest. For multivariate models like the Ising model, this means that samplers that update blocks (more than one element per iteration) can not be monotone with respect to the componentwise partial order. The following example gives two illustrations.

EXAMPLE 4 (NON-MONOTONICITY OF THE METROPOLIS-HASTINGS SAMPLER)
(a) Consider the three configurations of the Ising model in Figure 3 (a) for which $\mathbf{z} < \mathbf{x}^t < \mathbf{y}^t$ in the partial order. Assume that each site is numbered from the top left to bottom right, and that an external field is present. We find that $\pi(\mathbf{z})/\pi(\mathbf{x}^t) = \exp\{4\beta - 2\alpha_9\}$ and $\pi(\mathbf{z})/\pi(\mathbf{y}^t) = \exp\{12\beta - 2(\alpha_5 + \alpha_9)\}$. Thus for several realizations u and values of β, α_5 and α_9 we have $\pi(\mathbf{z})/\pi(\mathbf{x}^t) \leq u \leq \pi(\mathbf{z})/\pi(\mathbf{y}^t)$ (try $\beta = 1/3$, $\alpha_5 = \alpha_9 = 1$, $u = 0.8$), and thus $\phi(\mathbf{x}^t, \mathbf{z}, u) \geq \phi(\mathbf{y}^t, \mathbf{z}, u)$, and the order is reversed.

(b) For the configurations in Figure 3 (b) we have that $\pi(\mathbf{z})/\pi(\mathbf{x}^t) = \exp\{12\beta\} > \pi(\mathbf{z})/\pi(\mathbf{y}^t) = \exp\{-2\beta\}$ for an Ising model without external field. Thus for $\beta > 0$ we may find realizations $u \in (\exp\{-2\beta\}, \exp\{12\beta\}]$ so that $\phi(\mathbf{x}^t, \mathbf{z}, u) = \mathbf{z}$ and $\phi(\mathbf{y}^t, \mathbf{z}, u) = \mathbf{y}^t$, which can not be ordered with respect to the componentwise partial order. ◇

[Figure 3 about here.]

The componentwise partial order is not the only possible ordering of the state space. Proposition 1 is stated for this ordering, but it is easy to realize that monotone CFTP is valid for any other ordering of the state space with a minimum and maximum element and an update function that is monotone with respect to that ordering. Here is an example: Assume that candidate values $\mathbf{z}$ in the Metropolis-Hastings algorithm are drawn from a distribution q independent on the current state of the chain and accepted with probability $\alpha(\mathbf{x}^t, \mathbf{z}) = \min\{1, (\pi(\mathbf{z})q(\mathbf{x}^t))/(\pi(\mathbf{x}^t)q(\mathbf{z}))\}$. According to the ordering $\preceq$,where $\mathbf{x} \preceq \mathbf{y} \Leftrightarrow \pi(\mathbf{x})q(\mathbf{y}) \geq \pi(\mathbf{y})q(\mathbf{x})$ the Metropolis-Hastings algorithm is monotone for all target distributions. This ordering was suggested and its monotonicity proved by Corcoran & Tweedie (1998). In most situations this ordering is not directly applicable, since one would need to know the minimum and maximum of the target distribution to implement CFTP.

4 Results on Coalescence Time

As in Section 3.1 we let $\mathbf{X}^{t_2}(t_1, \mathbf{x}_m)$ be the state at time t_2 of a Markov chain started from state $\mathbf{x}_m$ at time t_1, $t_1 < t_2$ and we let $\mathbf{X}_j^{t_2}(t_1, \mathbf{x}_m)$ denote the jth element of this vector, $j = 1, \ldots, N$. We will need some definitions in order to make precise statements.

DEFINITION 5 Let the forward coalescence time T_{fc} be defined as the time to coalescence in a forward simulation where chains are started at time 0. If the sampler is monotone with minimum and maximum elements, $\mathbf{x}_{\min}$ and $\mathbf{x}_{\max}$, then $T_{\text{fc}} = \min\{t \geq 0 \ : \mathbf{X}_j^t(0, \mathbf{x}_{\min}) = \mathbf{X}_j^t(0, \mathbf{x}_{\max}) \ \forall j\}$.

DEFINITION 6 Let the backward coalescence time T_{cftp} be the smallest value of t so that chains started at time $-t$ have coalesced by time 0 in CFTP. For monotone CFTP it is $T_{\text{cftp}} = \min\{t \geq 0 \ : \mathbf{X}_j^0(-t, \mathbf{x}_{\min}) = \mathbf{X}_j^0(-t, \mathbf{x}_{\max}) \ \forall j\}$.

In Example 1 it is not difficult to see that the distribution of both T_{fc} and T_{cftp} is geometric with parameter 1/2. This is not a coincidence since by the nature of the two algorithms T_{fc} and T_{cftp} will always be governed by the same probability distribution. Many of the results on the coalescence time of the CFTP algorithm are derived by focusing on the conceptually simpler forward coalescence time.

THEOREM 4
The distributions of the forward and backward coalescence times T_{fc} and T_{cftp} are identical.

Proof:
The proof is straightforward with the observation that chains started in $-t$ and run to time 0 yield the same result as if the chains were shifted in time to start at 0 and end at time t.

$$\begin{aligned} P(T_{\text{cftp}} > t) &= P(\mathbf{X}_j^0(-t, \mathbf{x}_{\min}) \neq \mathbf{X}_j^0(-t, \mathbf{x}_{\max}) \text{ for at least one } j) \\ &= P(\mathbf{X}_j^t(0, \mathbf{x}_{\min}) \neq \mathbf{X}_j^t(0, \mathbf{x}_{\max}) \text{ for at least one } j) \\ &= P(T_{\text{fc}} > t). \end{aligned}$$

□

In practical situations it would be of great interest to have knowledge about the distribution of the coalescence time prior to initiating a CFTP run. Apart from providing us with information about probable running times, this could allow us to construct a more efficient search for T_{cftp}. General results on the distribution of the running time are not easily obtained except for very simple Markov chains. However, Propp & Wilson (1996) provided an upper bound for the expected coalescence time. Let $\tau = \min\{t > 0 : \bar{d}(t) \leq 1/e\}$ be the variation threshold of a Markov chain with transition matrix $\mathbf{P}$, where $\bar{d}(t) = \max_{\mathbf{x},\mathbf{y}\in\mathcal{S}} ||\mathbf{P}^t(\mathbf{x}, \cdot) - \mathbf{P}^t(\mathbf{y}, \cdot)||$ is the maximum variation distance. For a sampler that preserves the order of a partially ordered state space and that updates one of the N components of $\mathbf{X}$ at a time (such as the Gibbs sampler or Metropolis-Hastings algorithm for an attractive spin model), Propp & Wilson (1996) show that

$$\mathrm{E}(T_{\text{cftp}}) \leq 2\tau(1 + \log N). \tag{5}$$

Roughly speaking the expected time to coalescence will be small if the underlying sampler converges quickly in total variation distance. In a certain sense this inequality says that the time to coalescence in CFTP will not be much larger than the time to convergence for standard forward MCMC. Note that for more complicated models, N is replaced by the length of the largest totally ordered subset of the partially ordered state space $\mathcal{S}$. In Propp & Wilson (1998) other results on the running times for different CFTP algorithms are given.

5 Exact Sampling for Inference

In the previous sections we have been concerned with the problem of generating a single sample from a target distribution π. We now investigate how CFTP may be used for statistical inference which often requires repeated samples from π or the calculation of expectations $\xi = \mathrm{E}_\pi\{f(\mathbf{X})\}$ for a computable function f and $\mathbf{X} \sim \pi$. When iid samples $\mathbf{X}^1, \ldots, \mathbf{X}^K$ from π are available, the standard estimate is $\widehat{\xi} = \sum_{k=1}^K f(\mathbf{X}^k)/K$. When MCMC is used to obtain the K samples, these are both dependent and only asymptotically π distributed. A common procedure is to run a long MCMC chain, discard the first part of the trajectories (the burn-in) and take the average of the remaining samples.

It is possible to obtain an iid sample from π by repeating independent CFTP runs. We call this approach Independent CFTP. The approach requires a large computational effort, and uses only the last value in each CFTP run. Other alternatives make more use of the generated values. All the methods produce exact samples, but the samples are dependent to various degrees. We will not argue as to which method is to be preferred, as this will depend on the sampler and on f.

LONG CFTP: Run CFTP to obtain a sample $\mathbf{X}^0 \sim \pi$. Run the chain forward in time from $\mathbf{X}^0$ for T steps. The sampled values $\mathbf{X}^1, \ldots, \mathbf{X}^T$ are then all distributed according to π, but are dependent.

RANDOM CFTP: Run CFTP to obtain a sample $\mathbf{X}^0 \sim \pi$. Run the chain forward in time from $\mathbf{X}^0$ for $\widetilde{T}$ steps to produce $\mathbf{X}^1, \ldots, \mathbf{X}^{\widetilde{T}}$. Draw K values at random from these $\widetilde{T}$ samples and use these correctly π distributed variables as starting values for K forward simulations of length T_0. The resulting values $\mathbf{X}^{1,k}, \ldots, \mathbf{X}^{T_0,k}$, $k = 1, \ldots, K$ are then π distributed, and possibly less dependent than the values obtained by Long CFTP.

REPEATED CFTP: Run CFTP to obtain a sample $\mathbf{X}^{0,1} \sim \pi$. Run the chain forwards in time for T_0 steps, to produce $\mathbf{X}^{1,1}, \ldots, \mathbf{X}^{T_0,1}$. Repeat independently to produce $\mathbf{X}^{1,k}, \ldots, \mathbf{X}^{T_0,k}$, $k = 1, \ldots, K$. This gives KT_0 samples from π, where the T_0 values from one CFTP run are dependent, but different runs are independent.

CONCATENATED CFTP: Run CFTP to obtain a sample $\mathbf{X}^{0,0} \sim \pi$. Run another independent CFTP to obtain $\mathbf{X}^{0,1}$, and let T_1 be the time to coalescence for this run. Look at the trajectory of this last run started in $\mathbf{X}^{0,0}$ at time $-T_1$ and take it as the first T_1 sampled values, i.e. the path from $\mathbf{X}^{0,0}$ to $\mathbf{X}^{0,1}$ that we denote $\mathbf{X}^{1,1}, \ldots, \mathbf{X}^{T_1,1}$, counting $\mathbf{X}^{0,1} = \mathbf{X}^{T_1,1}$ but not $\mathbf{X}_0^0$. Continue with an independent CFTP run, taking the next T_2 values to be the path from $\mathbf{X}^{0,1}$ to $\mathbf{X}^{0,2}$ etc. This produces samples $\mathbf{X}^{1,k}, \ldots, \mathbf{X}^{T_k,k}$ of random size T_k, $k = 1, \ldots, K$

GUARANTEE TIME CFTP: Fix a time T_g to use as initial guess for coalescence time in CFTP. Follow the scheme of Concatenated CFTP, but for each CFTP run, collect only the T_g last values in each path. Thus, the procedure is the same as for Concatenated CFTP, but each run generates a fixed number $T_k = T_g$, $k = 1, \ldots, K$ of samples.

The methods Repeated CFTP, Concatenated CFTP and Guarantee Time CFTP were proposed by Murdoch & Rosenthal (1998*b*). Murdoch & Rosenthal (1998*b*) also consider forward coupling as well as a modification of Concatenated CFTP where the resulting value of a CFTP run is used as starting point to pick a path in the same run. This approach may look appealing, but due to the dependency between the output value and the coalescence time it will yield a biased sample and the distribution π may be lost.

For all methods except for Concatenated CFTP, the estimator $\widetilde{\xi} = \sum_{k=1}^{K} \sum_{t=1}^{T_k} f(\mathbf{X}^{t,k})/T_k$ (with $T_k = T_0$ or T_g) is unbiased and consistent. Because of the randomness of the sample

sizes in Concatenated CFTP, the estimator $\xi^* = \sum_{k=1}^{K}\sum_{t=1}^{T_k} f(\mathbf{X}^{t,k})/\sum_{k=1}^{K} T_k$ must be used instead to assure consistency. Murdoch & Rosenthal (1998*b*) approximate the computational cost for Repeated CFTP, Concatenated CFTP and Guarantee Time CFTP (as well as forward coupling), and compared these strategies for a random walk on the integers from 0 to 20 with equal probability of stepping up and down a unit, and steps outside the range being rejected. It would be interesting to compare the various approaches in other practical problems.

6 CFTP vs. Standard MCMC: Simulation Experiments

Our aim is to do Bayesian image restoration by applying the CFTP algorithm with Gibbs sampling to sample the exact posterior distribution. We study a two dimensional binary image with sites or *pixels* that are either black (+1) or white (-1). We assume that our true image is degraded by noise, and that we only observe the noisy image. We will compare the results on overall running time and misclassification rates of CFTP and classical forward Gibbs sampling. Convergence of the Gibbs sampler is here determined by the method of Raftery & Lewis, see Raftery & Lewis (1992, 1996). Such a comparison will give us an impression of what prize we pay (if any), in terms of running time, to obtain exact samples.

Let $\mathbf{X}$ be the true image, and assume apriori that $\mathbf{X}$ is distributed according to the Ising model (Example 3). Let $\mathbf{y}$ be the observed image. We assume that the noise is Bernoulli, i.e. with probability ϵ the color of a pixel is changed, and with probability $1-\epsilon$ we observe the true color, independently of the other pixels. The probability of observing $\mathbf{y}$ when $\mathbf{x}$ is the true image is

$$\begin{aligned} f(\mathbf{y}|\mathbf{x}) &= \prod_j \epsilon^{I(x_j \neq y_j)}(1-\epsilon)^{I(x_j = y_j)} \\ &= \epsilon^N \{(1-\epsilon)/\epsilon\}^{\sum_j I(x_j = y_j)}, \end{aligned}$$

where I is the indicator function and the pixels are numbered $j = 1, \ldots, N$. Thus, the posterior is

$$\begin{aligned} \pi(\mathbf{x}|\mathbf{y}) &= f(\mathbf{y}|\mathbf{x})\pi(\mathbf{x}) \\ &\propto \exp\{\beta \sum_{i\sim j} x_i x_j + \log((1-\epsilon)/\epsilon)/2 \sum_j x_j y_j\}, \end{aligned}$$

which may also be regarded as an Ising model with an sitevarying external field (see (4)). The Gibbs sampler for this model can be expressed by an update function as in (3), with

$$\pi(X_v = 1|\mathbf{x}^t_{-v}, \mathbf{y}) = \{1 + \exp(-2\beta \sum_{j\sim v} x_j^t - \log((1-\epsilon)/\epsilon) y_v)\}^{-1}.$$

In our simulation study the true image is a 40×40 realization of the Ising model with parameter $\beta = 0.45$ (that we obtained by CFTP, see Figure 5). Noise is added for $\epsilon =$

$\{0.1, 0.2, 0.3, 0.4\}$, and for each noisy image restorations are done for $\beta = 0.45$ only. Note that for $\beta = 0.45$ restoration is quite difficult, due to the occasional single pixels that are surrounded by pixels of the other color. In a situation with unknown β one would have to simulate over grid of ϵ and β values or estimate ϵ and β in some way.

Since the Gibbs sampler is monotone with respect to the componentwise partial order for an Ising model with external field (see Definition 2 and Example 3), we can apply monotone CFTP with the upper chain started from the image with only black pixels and the lower chain from the image with only white pixels. In the Gibbs sampler one pixel is updated at a time, but we define one time step to be a full sweep where all pixels are updated in random permutation order, and we only check for coalescence after full sweeps. As an estimate of the true image we choose the Marginal Posterior Mode (MPM), see Winkler (1995, p. 219). To approximate it we use 500 Independent CFTP restorations of the same noisy image. The misclassification rate is the fraction of pixels of the MPM image that are different from the true image.

In order to compare standard forward Gibbs sampling with CFTP, we need to estimate the speed of convergence of the Gibbs sampler chain. The method of Raftery & Lewis is widely applied, with computer code being available on the Internet. We used `gibbsit` from Statlib (`http://lib.stat.cmu.edu/`), but an implementation in `CODA` (Best et al., 1995) is also available. As for most practical convergence diagnostics, several arguments have been raised against this method, see for instance Mengersen et al. (1998). For the purpose of obtaining estimates of the time to convergence in a simple comparison study, we find the method suitable because this is typically what would be done in practice. We choose to monitor the magnetization, $\mu = \sum_j X_j/N$, the interaction statistics $\rho = \sum_{i\sim j} X_i X_j/2N$ and the fraction of black pixels $\omega = \sum_j I(X_j = 1)/N$, estimated by

$$\widehat{\mu}(T) = \frac{1}{T}\sum_{t=1}^{T}\{\frac{1}{N}\sum_{j=1}^{N} X_j^t\}$$

$$\widehat{\rho}(T) = \frac{1}{T}\sum_{t=1}^{T}\{\frac{1}{2N}\sum_{i\sim j} X_i^t X_j^t\}$$

$$\widehat{\omega}(T) = \frac{1}{T}\sum_{t=1}^{T}\{\frac{1}{N}\sum_{j=1}^{N} I(X_j^t = 1)\},$$

where $\mathbf{X}^1, \mathbf{X}^2, \ldots, \mathbf{X}^T$ is the Gibbs sampler chain. In addition we monitor some single pixels. The method of Raftery & Lewis provides several diagnostics. However, we will use only the estimated burn-in (the number of sweeps necessary to eliminate the effect of the starting value) and the estimated thinning parameter (if k is the thinning, only every kth iteration is kept after the burn-in). The method is designed for estimation of posterior quantiles of the statistics of interest. To do this, the method needs as input a pilot sample, the quantile q of interest, and a precision parameter δ (for the allowed distance between estimated transition probabilities and estimated stationary distribution in a two-state chain constructed from the input, for details see Raftery & Lewis (1992,

1996)). Raftery & Lewis (1996) recommended that `gibbsit` is run for several values of q, and we have chosen to run with $q = \{0.025, 0.5, 0.975\}$ for all the monitored statistics. We choose as precision $\delta = 0.01$, which is a weaker requirement than the default which is 0.001, and pilot samples of length 3000. To our experience the method can be quite sensitive to the input parameters, and occasionally the estimated burn-ins are very large. To remove a dominating effect of such instabilities and to exhibit some of the characteristics of the method, we estimate the burn-in of one run by the 0.75 and 0.90 quantiles of the estimated burn-ins over all the monitored statistics and over all the values of q. Our final estimates are the mean and standard deviation of 500 independent posterior simulations with different initial images. Since the estimated thinning turn out to be on average 1, an estimate of the MPM approximated using 500 simulations following the most conservative (0.90 quantile) burn-in estimate. The misclassification rates where found for this MPM estimate.

Table 1 (a) shows the mean and the standard deviation of the time to coalescence based on the 500 independent CFTP runs. The numbers are based on the exact coalescence times, in the sense that the simulations are done by going back in time in steps of size one. Figure 4 shows density estimates of the distribution of T_{cftp} for the different noise levels. The distributions are all heavily skewed with a tail that increases exponentially in the noise. In Table 1 (b) the mean and standard deviation of the number of full sweeps required for CFTP and MCMC convergence are compared. In this comparison we have used the "doubling of time" approach for CFTP. According to our convergence criteria, CFTP does not require more computation time than MCMC. In light of (5) this is not surprizing and we emphasize that we have not manipulated the results in order to favour CFTP. We have tried to make the comparison fair, and the presented MCMC results are indeed similar to results we got from inspecting plots of the monitored statistics. The true and noisy images and the MPM restorations and their misclassification rates are shown in Figure 5. We see that the misclassification rates are very similar, the largest difference being for $\epsilon = 0.4$ for which both estimates are quite far from the true image. This is an indication that standard Gibbs sampling with Raftery & Lewis diagnostics works well. We conclude that CFTP is an appealing algorithm for simple problems in Bayesian image restoration, despite requiring more work in implementation.

[Table 1 about here.]

[Figure 4 about here.]

[Figure 5 about here.]

7 Exact Sampling for Continuous State Spaces

For a continuous state space the idea of coupling from the past is not directly applicable since trajectories from different initial states will not coalesce. Methods that deal with this problem have been constructed, and exact sampling for some continuous state space

distributions is possible. The range of models that can be handled is still limited, but ongoing research might lead to extensions.

Murdoch & Green (1998) and Green & Murdoch (1998) have developed several exact sampling algorithms which introduce a discreteness into the state space by updating sets of states into one single state. We present the simplest of their algorithms, the *multigamma coupler*, to illustrate the general idea, and refer to their papers for other algorithms and detailed pseudo-codes.

For notational simplicity we will in the following description assume $\mathcal{S} = \mathbb{R}$, and we let $\{X^t\}$ be a Markov chain on $\mathcal{S}$ with a stationary density π. The transition kernel of the chain, $f(\cdot|x) \geq 0$, is defined by

$$\int \pi(x) f(y|x) dx = \pi(y).$$

As in the discrete case the transitions of the chain are described by the update function ϕ defined so that

$$P(\phi(x, \mathbf{U}) \leq y) = \int_{-\infty}^{y} f(v|x) dv.$$

Two chains $\{X_1^t\}$ and $\{X_2^t\}$ started in two different initial states will coalesce at time t if $\phi(x_1^{t-1}, \mathbf{U}) = \phi(x_2^{t-1}, \mathbf{U})$. The idea is to construct an algorithm that with probability 1 makes a transition, at some random time point, such that $\phi(x, \mathbf{U}) = \psi(\mathbf{U})$ and all chains coalesce into a single state independently of where they originate.

In the multigamma coupler the transition kernel is assumed to be bounded below so that $f(y|x) > r(y)\ \forall x, y \in \mathcal{S}$, for some nonnegative function r. Define $\rho = \int r(y) dy$ and

$$R(y) = \rho^{-1} \int_{-\infty}^{y} r(v) dv$$
$$Q(y|x) = (1-\rho)^{-1} \int_{-\infty}^{y} \{f(v|x) - r(v)\} dv.$$

Define the update function

$$\phi(x, \mathbf{U}) = \begin{cases} R^{-1}(U_2) & \text{if} \quad U_1 < \rho \\ Q^{-1}(U_2|x) & \text{if} \quad U_1 \geq \rho, \end{cases}$$

where $\mathbf{U} = (U_1, U_2)$ are independent uniform random numbers. To see that this update function provides draws from the right transition kernel, note that

$$\begin{aligned} P(\phi(x, \mathbf{U}) \leq y) &= \rho R(y) + (1-\rho) Q(y|x) \\ &= \int_{-\infty}^{y} f(v|x) dv. \end{aligned}$$

If $\mathcal{R}_t$ denotes the set of states occupied by the Markov chains at time t, the multigamma coupler may be outlined as follows. As for CFTP we start at a time $-T$ with $\mathcal{R}_{-T} = \mathcal{S}$ and

run to time 0. At each time point random numbers $\mathbf{U}^t$ are drawn. If $U_1^t < \rho$ the updating is done via $R^{-1}(U_2^t)$ and the first time this happens $\mathcal{R}_t$ will shrink to a single state. Then from this state a single chain is run forward to time 0 using R^{-1} or Q^{-1} for the update according to whether U_1^t is smaller or greater than ρ. Until $U_1^t < \rho$ (before $\mathcal{R}_t$ is reduced to a single state), $\mathcal{R}_t$ is set to be $\mathcal{S}$. If the number of states at time 0 is 1, this state is the returned sample and the algorithm terminates. Otherwise, the algorithm is restarted from time $-2T$ (or any other timepoint $-T' < -T$) reusing the previously generated random numbers from time $-T$ to 0.

The other algorithms of Murdoch & Green (1998) are called the *partitioned multigamma coupler*, the *rejection coupler* and the *Metropolis-Hastings coupler*. As with the multigamma coupler these algorithms are based on the idea of updating sets to a single value, but the assumptions on which the algorithms are based differ from the multigamma coupler. The couplers are applied to a dataset on pump reliability.

Møller (n.d.) presented another approach to exact sampling for continuous distributions, where the user specifies an ϵ, and the algorithm returns a simulation with accuracy within ϵ of π. If ϵ is the machine precision, we can consider this algorithm as returning an exact sample. The method is applied to the auto-Gamma distribution for the same pump reliability data as in Murdoch & Green (1998).

8 Fill's Interruptible Algorithm

When applying CFTP it may sometimes be tempting to abort long runs that have not coalesced in favour of a new run with a new sequence of random numbers. Because of the dependency between the length of the run and the returned value in CFTP, the impatient user who aborts long runs may introduce a bias in the sense that the finally obtained sample will not be distributed according to π. Examples are given by Fill (1998) and Thönnes (1997). There are some chains that protect against this bias, and Thönnes (1997) showed that the chain of Example 1 is such a chain.

Fill (1998) introduced an alternative exact simulation for finite state spaces with a maximal and minimal state based on rejection sampling, which in contrast to CFTP is unbiased for user impatience, because the output is independent of the running time. Fill's algorithm is harder to implement than CFTP and has received somewhat less attention. In Fill's algorithm a Markov chain is started in the minimal state of a partially ordered state space, and run forward for T transitions, where T is fixed in advance. The output is then proposed as a sample, and this sample is accepted if and only if a second chain started in the maximal state and coupled to the reversed trajectory of the first chain arrives at the minimal state after T transitions. If it does not, the sample is rejected, and the procedure is repeated independently of the previous run, with $2T$ transitions, until acceptance. We shall now describe the how the forward and reversed chains are coupled.

As with CFTP, Fill's algorithm assumes that we can construct an ergodic Markov chain, $\mathbf{X}^0, \mathbf{X}^1, \ldots$ with stationary distribution π expressed by an update function ϕ and

iid random numbers $\mathbf{U}$ as

$$\mathbf{X}^{t+1} = \phi(\mathbf{x}^t, \mathbf{U}^{t+1}).$$

The state space is assumed to be finite and partially ordered with minimal and maximal states, $\mathbf{x}_{\min}$ and $\mathbf{x}_{\max}$. Let $\mathbf{P}$ denote the transition matrix of $\mathbf{X}$. The time-reversal $\widetilde{\mathbf{X}}$ of $\mathbf{X}$ is defined through the transition matrix $\widetilde{\mathbf{P}}$ by

$$\widetilde{\mathbf{P}}(\mathbf{x}, \mathbf{y}) = \frac{\pi(\mathbf{y})}{\pi(\mathbf{x})}\mathbf{P}(\mathbf{x}, \mathbf{y}).$$

In our presentation of Fill's algorithm we will assume that there exits an monotone update function for the time reversed chain such that

$$\widetilde{\mathbf{X}}^{t+1} = \widetilde{\phi}(\widetilde{\mathbf{x}}^t, \mathbf{U}^{t+1}).$$

(Note that this is not required, and a sufficient condition for the algorithm is that $\widetilde{\mathbf{P}}$ is stochastically monotone, see Fill (1998) for details.) Fill's algorithm for exact simulation is then as follows.

Algorithm 2 (Fill's algorithm)

1. Choose the number of transitions $T > 0$.
2. Generate iid random numbers for the update function, $\mathbf{U}^1, \mathbf{U}^2, \ldots, \mathbf{U}^T$.
3. Use the random number vectors to generate $\mathbf{X}^{t+1} = \phi(\mathbf{x}^t, \mathbf{U}^{t+1})$ with $\mathbf{X}^0 = \mathbf{x}_{\min}$ for $t = 0, \ldots, T-1$.
4. Draw random T numbers $\widetilde{\mathbf{U}}^t$ for $t = 1, \ldots, T$, where $\widetilde{\mathbf{U}}^t$ is a sample from the conditional distribution of $\mathbf{U}$ given $\widetilde{\phi}(\mathbf{x}^{T-t+1}, \mathbf{U}) = \mathbf{x}^{T-t}$. This is the conditional distribution of $\mathbf{U}$ given that it caused the transition from $\mathbf{x}^{T-t+1}$ to $\mathbf{x}^{T-t}$ in the reversed Markov chain.
5. Start in $\widetilde{\mathbf{Y}}^0 = \mathbf{x}_{\max}$ and generate a second chain $\widetilde{\mathbf{Y}}^1, \ldots, \widetilde{\mathbf{Y}}^T$ according to $\widetilde{\mathbf{Y}}^{t+1} = \widetilde{\phi}(\widetilde{\mathbf{y}}^t, \widetilde{\mathbf{U}}^{t+1})$.
6. If $\widetilde{\mathbf{Y}}^T = \mathbf{x}_{\min}$ accept $\mathbf{X}^T$ as a sample from π, else double T and start from (2), with new random numbers in step 2 and 4 independently from the previous runs.

The proof of convergence of this algorithm is found in Møller & Schladitz (1998). A proof of Fill's algorithm in the general setting is found in Fill (1998). Fill's algorithm in the general case, with the only restriction begin that $\widetilde{\mathbf{P}}$ is stochastically monotone, differs from Algorithm 2 only in step 4, see Fill (1998) for details. The doubling of the number of transitions in Algorithm 2 is suggested by Fill (1998). However, the correctness of the algorithm holds for any non-decreasing sequence of values for $T > 0$.

Fill (1998) compared his algorithm and CFTP for an attractive spin system with a total of N sites and a random site updating Gibbs sampler. The two algorithms run in

time of the same order in N. Fill's algorithm uses logarithmically more space, as the space requirement is $O(N \log N)$ vs. $O(N)$ for CFTP. However, for this "price" we get an algorithm that protects against user imposed bias.

Thönnes (1997) extended Fill's algorithm to situations where the state space is not finite and does not have a minimal and maximal state. This was done by defining a particular partial order of the state space as well as states that serve as unique minimal and maximal states.

Møller & Schladitz (1998) presented Fill's algorithm for general repulsive (anti-monotone) systems which can not be transformed to the monotone case simply by defining a suitable partial order. To solve this problem sandwiching by having an upper and lower bounding chain was introduced.

In Murdoch & Rosenthal (1998*a*)'s extension of Fill's algorithm, no assumption on order, monotonicity or discreteness of the state space is required. In their algorithm a chain is started from any chosen state of $\mathcal{S}$ and run forward for T steps, and the output is the proposed sample. Then the direction is reversed and chains are run from all states of $\mathcal{S}$ conditioned on the first chain as in Fill's algorithm. If the resulting values (at time 0) are *all* equal, the proposed sample is accepted. For most sample spaces, running chains from all states of $\mathcal{S}$ and checking them for equality will require methods as those of Murdoch & Green (1998).

9 Discussion

In this paper we have outlined some of the recently developed methods for exact simulation. We have focussed on the CFTP algorithm for samplers that are monotone with respect to discrete and partially ordered state spaces. We have outlined how CFTP can be adapted to a continuous state space, and we have presented the interruptible method of Fill.

The research and development of exact simulation following the papers of Propp & Wilson (1996) and Fill (1998) have been extensive. Only a small sample of the quickly growing literature on exact sampling is mentioned in this paper. We emphasize that many successful extensions and applications of the basic algorithms have been done for spatial point processes. Kendall (1998) extended monotone CFTP to the area-interaction point process where the state space is infinite and does note have a maximal state by introducing a dominating process. The same paper shows how to apply CFTP to repulsive point processes, which are anti-monotone. In a follow up article, Kendall (1997), CFTP is applied to attractive birth and death processes and exclusion processes. The anti-monotone case is also treated in Häggström & Nelander (n.d.*a*), where simulation results are given for the hard-core gas model and the random cluster model with $q < 1$. Häggström et al. (1996) showed how to apply CFTP to the Widom-Rowlinson point process, where there is no minimal or maximal state, by using two other states in the state space for testing coalescence. The Widom-Rowlinson point process is also treated by Thönnes (1997) who extended Fill's algorithm to this model. Møller (n.d.) extended CFTP to a class of Markov random fields (locally specified exponential family distributions) by using Gibbs sampling

in combination with sandwiching properties. Markov random fields are also treated in Häggström & Nelander (n.d.*a*). Also, it should be mentioned that the CFTP algorithm has been extended to the Swendsen-Wang algorithm, see Huber (1998*a*) and Huber (n.d.*b*). Propp & Wilson (1998) included several versions of CFTP particularly suited for simulation of spanning trees on directed graphs.

In conclusion, exact simulation is feasible for sampling from a collection of models. However, as exact simulation will be difficult to extend and implement for more general models, it is important that statisticians do not compromise unreasonably with respect to model choice in order to be able to do exact simulation. The revolution of MCMC that freed Bayesian users from conjugate priors would otherwise be betrayed.

10 Literature and Software on the Internet

D.B. Wilson's web-page on exact simulation, `http://dimacs.rutgers.edu/~dbwilson/exact.html/` is continuously updated and is an excellent source of information. The site includes links to most papers and preprints in the field and has a full annotated bibliography available in BibTeX format. Most preprints are also available on the MCMC preprint service at `http://www.stats.bris.ac.uk/MCMC/`. At the present there is no professional software available for exact simulation. Some authors have made their implementations available on the web. See for instance Geoff Nicholls page `http://matu1.math.auckland.ac.nz/~nicholls/` for a C code for CFTP with the Metropolis Hasting algorithm. Java-applets that illustrate CFTP are available at `http://markov.utstat.toronto.edu/jeff/java/` and `http://www.warwick.ac.uk/statsdept/Staff/WSK/dead.html`.

Acknowledgments

This work was supported by The Research Council of Norway (NFR) under grant no. 110673/420 (Numerical Computations in Applied Mathematics). Arnoldo Frigessi, who is supervising the author's PhD thesis, is supported by the grant no. 121144/420 (Knowledge, data and decisions. Modern Statistics in Action). The C code of Geoff Nicholls (see `http://matu1.math.auckland.ac.nz/~nicholls/`) was of great help as we used some of its main features as a basis for our implementation of CFTP.

References

Asmussen. S., Glynn. P. W. & Thorisson. H. (1992). Stationary detection in the initial transient problem. *ACM Trans. Model. Comput.* **2**(2), 130–157.

Best. N., Cowles. M. & Vines. S. (1995). *CODA Manual version 0.3.* MRC Biostatistics Unit, Cambridge, UK.

Brooks. S. & Roberts. G. (1999). Assesing convergence of Markov chain Monte Carlo algorithms. *Statist. Comput.* To appear.

Corcoran. J. N. & Tweedie. R. L. (1998). Perfect sampling from independent Metropolis-Hastings chains. Preprint.

Cowles. M. & Carlin. B. (1996). Markov Chain Monte-Carlo convergence diagnotics: A comparative study. *J. Amer. Statist. Assoc.* **91**, 883–904.

Fill. J. A. (1998). An interruptible algorithm for perfect sampling via Markov chains. *Ann. Appl. Probab.* **8**, 131–162.

Fortuin. C. M. & Kasteleyn. P. W. (1972). On the Random Cluster Model. I. Introduction and Relation to Other Models. *Physica* **57**(4), 536–564.

Frigessi. A. & Den Hollander. F. (1993). A dynamical phase transition in a caricature of a spin glass. *J. Statist. Physics* **75**, 585–625.

Geman. S. & Geman. D. (1984). Stocastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. *I.E.E.E. Trans. Pattn. Anal. Mach. Intel.* **6**, 721–741.

Green. P. J. & Murdoch. D. J. (1998). Exact sampling for bayesian inference: Towards general purpose algorithms. In *Bayesian Statistics 6*, Eds. J. Bernardo, J. Berger, A. Dawid & A. Smith. Oxford: Oxford University Press. To appear.

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

Häggström. O. & Nelander. K. (n.d.*a*). Exact sampling from anti-monotone systems. *Statist. Neerlandica.* To appear.

Häggström. O. & Nelander. K. (n.d.*b*). On exact simulation of Markov random fields using coupling from the past. *Scand. J. Statist.* To appear.

Häggström. O., van Lieshout. M. N. M. & Møller. J. (1996). Characterisation results and Markov chain Monte Carlo algorithms including exact simulation for some spatial point processes. Technical Report R-96-2040. Department of Mathematics, Aalborg University. To appear in *Bernoulli.*

Huber. M. (1998*a*). Efficient Exact Sampling From the Ising Model Using Swendsen-Wang. Preprint.

Huber. M. (n.d.*b*). Exact sampling and approximate counting techniques. *Proceedings of the 30th ACM Symp. Theory of Computing.* To appear.

Johnson. V. E. (1996). Studying convergence of Markov chain Monte Carlo algorithms using coupled sample paths. *J. Amer. Statist. Assoc.* **91**(433), 154–166.

Kendall. W. S. (1997). On some weighted Boolean models. In *Advances in Theory and Applications of Random Sets*, Ed. D. Jeulin., pp. 105–120. Singapore: World Scientific Publishing Company.

Kendall. W. S. (1998). Perfect simulation for the area-interaction point process. In *Probability Towards 2000*, Eds. C. Heyde & L. Accardi., pp. 218–234. New York: Springer-Verlag.

Kendall. W. S. & Thönnes. E. (1998). Perfect simulation in stochastic geometry. Preprint.

Lindvall. T. (1992). *Lectures on the Coupling Method*. London: Wiley.

Lovász. L. & Winkler. P. (1995). Exact mixing in an unknown Markov chain. *Electron. J. Combin.* Paper #R15.

Mengersen. K., Robert. C. & Guihenneuc-Jouyaux. C. (1998). MCMC Convergence Diagnostics: "A Review`www`". In *Bayesian Statistics 6*, Eds. J. Bernardo, J. Berger, A. Dawid & A. Smith. Oxford: Oxford University Press. To appear.

Metropolis. N., Rosenbluth. A. W., Rosenbluth. M. N., Teller. A. H. & Teller. E. (1953). Equations of state calculations by fast computing machine. *J. Chem. Physics* **21**, 1087–1091.

Møller. J. (n.d). Perfect simulation of conditionally specified models. *J. Roy. Statist. Soc. Ser. B.* To appear.

Møller. J. & Schladitz. K. (1998). Extensions of Fill's algorithm for perfect simulation. Preprint.

Murdoch. D. J. & Green. P. J. (1998). Exact sampling from a continuous state space. *Scand. J. Statist.* **25**(3), 483–502.

Murdoch. D. J. & Rosenthal. J. S. (1998*a*). An extension of Fill's exact sampling algorithm to non-monotone chains. Preprint.

Murdoch. D. J. & Rosenthal. J. S. (1998*b*). Efficient exact sampling. Preprint.

Propp. J. G. & Wilson. D. B. (1996). Exact sampling with coupled Markov chains and applications to statistical mechanics. *Random Structures Algorithms.* **9**(1&2), 223–252.

Propp. J. G. & Wilson. D. B. (1998). How to get a perfectly random sample from a generic Markov chain and generate a random spanning tree of a directed graph. *J. Algorithms* **27**, 170–217.

Raftery. A. & Lewis. S. (1992). How many iterations in the Gibbs sampler? In *Bayesian Statistics 4*, Eds. J. Bernardo, J. Berger, A. Dawid & A. Smith, pp. 765–776. Oxford: Oxford University Press.

Raftery. A. & Lewis. S. (1996). Implementing MCMC. In *Markov chain Monte-Carlo in Practice*, Eds. W. Gilks & D. Richardson. S.T. and Spiegelhalter, pp. 115–130. London: Chapman and Hall.

Ripley. B. (1987). *Stochastic Simulation*. New York: Wiley.

Thönnes. E. (1997). Perfect simulation of some point processes for the impatient user. Preprint.

Tierney. L. (1994). Markov Chains for Exploring Posterior Distriubtions (with discussion). *Ann. Statist.* **22**, 1701–1762.

Winkler. G. (1995). *Image Analysis, Random Fields and Dynamic Monte Carlo Methods*. Berlin: Springer-Verlag.

Résumé

Cet article présente certains des derniers travaux sur la simulation exacte, et en particulier l'algorithme CFTP. Les concepts de couplage et de monotone CFTP sont présentés, et des résultats concernant les temps de calcul de l'algorithme sont décrites. Des nouvelles expériences de simulation pour le modèle d'Ising sont données, et les résultats sont comparés à ceux du MCMC classique. La méthode interruptible de Fill (1998) et de Murdoch & Green (1998) pour l'échantillonage exacte d'une distribution continue sont présentées.

Table 1
The left table (a) displays the mean and standard deviation of the time to coalescence based on 500 independent CFTP runs for the noisy images in Figure 5. The right table (b) shows the number of full sweeps required by the CFTP algorithm and standard MCMC. For MCMC the estimates are based on the 0.75 and 0.90 quantiles over all the monitored statistics and values of q, and for CFTP the numbers are based on a "doubling of the time"approach.

(a)

ϵ	$\widehat{\mathrm{E}}(T_{\mathrm{cftp}})$	$\widehat{\mathrm{sd}}(T_{\mathrm{cftp}})$
0.1	10	2
0.2	26	7
0.3	63	17
0.4	300	110

(b)

	MCMC				CFTP	
	0.75		0.90			
ϵ	mean	sd	mean	sd	mean	sd
0.1	309	684	548	994	26	8
0.2	340	656	744	1853	71	25
0.3	385	402	978	2128	175	65
0.4	716	1072	2397	3494	860	357

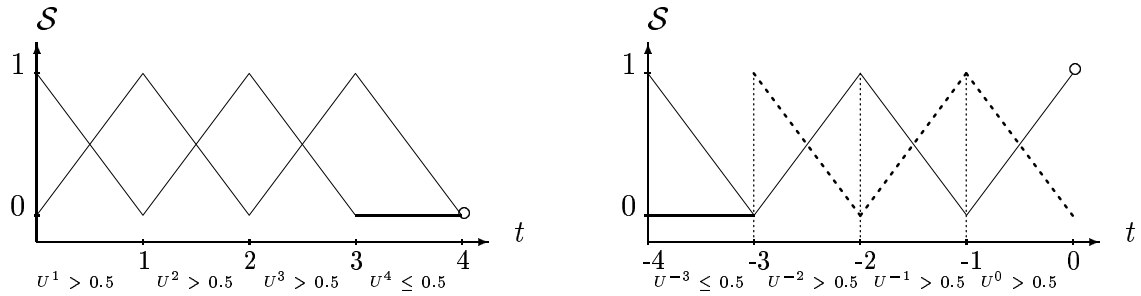

Figure 1. *The left panel illustrates forward coupling for Example 1, and the right panel illustrates CFTP for the same example. In both examples the same sequence of random numbers is used, and we see how forward coupling always will happen in state 0, while with CFTP the chains coalesce in state 0, but the state at time 0 may be either 0 or 1. For instance, for* $U^{-2} \leq 0.5$ *then* X^0 *would be 0. To understand the right panel, the reader should try startingfrom times* $t = -1, -2, \ldots$ *etc. The solid line indicates the final path.*

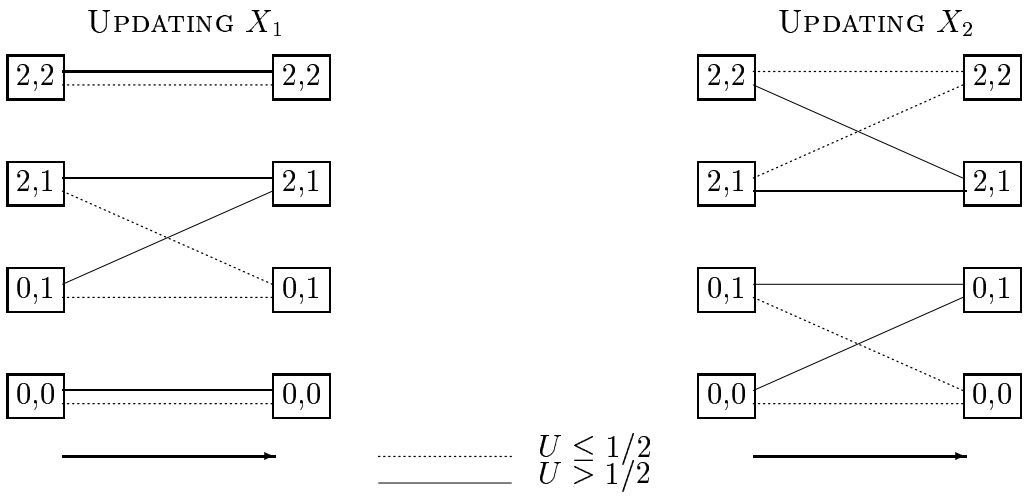

Figure 2. *Illustration of coalescence in Example 2. The left and right panel show the updating of the first and second element of* $\mathbf{X}$ *respectively. The dotted lines show the updating for* $U \leq 1/2$ *and the solid lines for* $U > 1/2$*. The same random numbers are applied for all 4 chains, and thus for each update we only get dotted or solid lines.*

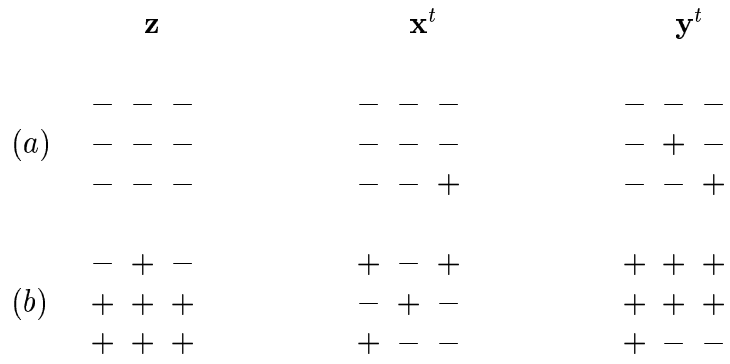

Figure 3. *Configurations from the 2D Ising model on a* 3×3. *In (a)* $\mathbf{z} \leq \mathbf{x}^t \leq \mathbf{y}^t$ *with respect to the componentwise partial order. In (b)* $\mathbf{x}^t \leq \mathbf{y}^t$, *while* $\mathbf{z}$ *can not be ordered.*

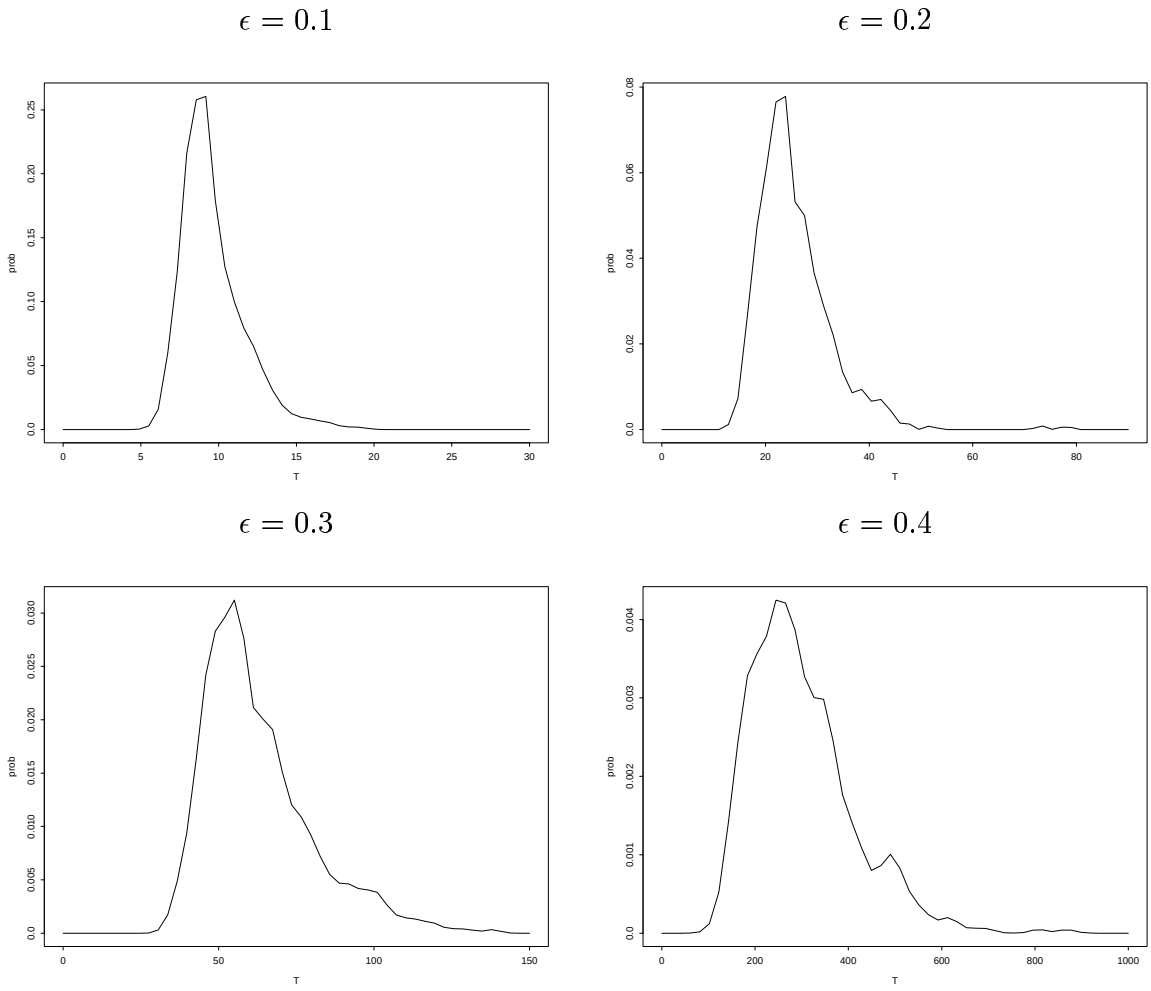

Figure 4. *Density estimates of the time to coalescence for the Ising 0.45 prior degraded by different amounts of noise.*

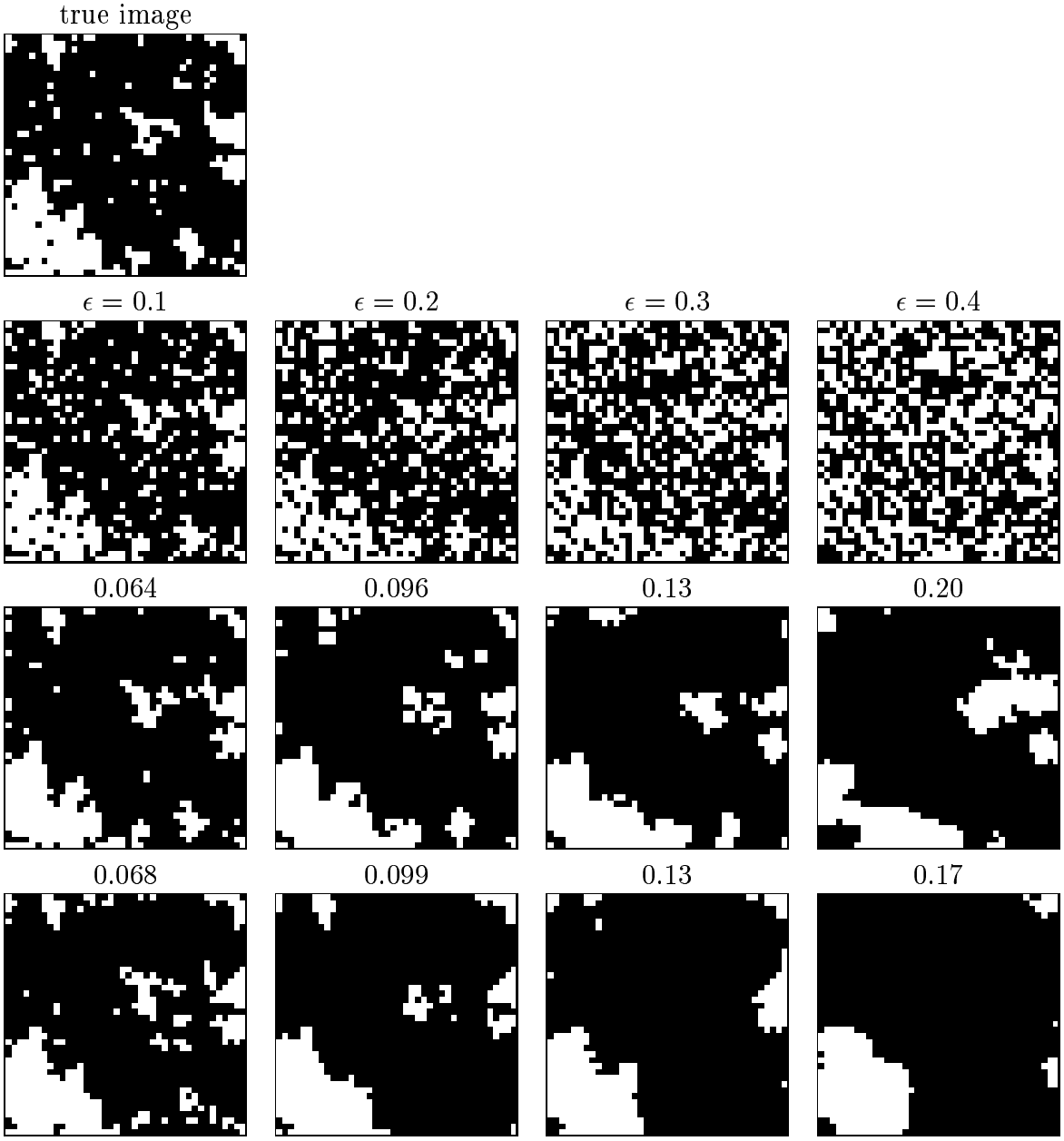

Figure 5. *The top image shows a simulation of the Ising model with $\beta = 0.45$ using CFTP with the Gibbs sampler. The second row of images shows the top image degraded by various levels of noise. The third row shows the modes of 500 simulations from the posterior using CFTP and the Gibbs sampler with $\beta = 0.45$. The misclassification rates are shown in the headings. The bottom row shows the modes and misclassification of 500 MCMC simulations following the conservative burn-in estimates of Table 1 (b).*