Explaining the Perfect Sampler

George Casella
Cornell University
Dept. of Biometry

Michael Lavine
Duke University
ISDS

Christian Robert
CREST, Insee, Paris
and
Université Paris 9 – Dauphine

November 15, 2000

Abstract

In 1996, Propp and Wilson introduced Coupling from the Past (`CFTP`), an algorithm for generating a sample from the exact stationary distribution of a Markov chain. In 1998, Fill proposed another so–called *perfect sampling* algorithm. These algorithms have enormous potential in Markov Chain Monte Carlo (`MCMC`) problems because they eliminate the need to monitor convergence and mixing of the chain. This article provides a brief introduction to the algorithms, with an emphasis on understanding rather than technical detail.

1 Setting

A Markov chain is a sequence of random variables $\{X_t\}$ that can be thought of as evolving over time, and where the distribution of X_{t+1} depends on X_t, but not on $X_{t-1}, X_{t-2}, \ldots$. When used in Markov chain Monte Carlo (`MCMC`) algorithms, Markov chains are usually constructed from a *Markov transition kernel* K, a conditional probability density on a state space $\mathcal{X}$ such that $X_{t+1}|X_t \sim K(X_t, \cdot)$. Interest is usually in the *stationary distribution* of the chain, the distribution π that satisfies

$$\int_{\mathcal{X}} K(x, B)\, d\pi(x) = \pi(B) \text{ for any measurable subset } B \text{ of } \mathcal{X}.$$

Thus, if $X_t \sim \pi$ then $X_{t+1} \sim \pi$. In a common application π is the posterior distribution from a Bayesian analysis and K is constructed to have stationary distribution π.

Here is an example that we follow throughout the article.

Beta-Binomial. Following Casella and George (1992), and for some suitable parameters n, α and β, let $\theta \sim \text{Beta}(\alpha, \beta)$ and $X|\theta \sim \text{Bin}(n, \theta)$, leading to the joint density

$$\pi(x, \theta) \propto \binom{n}{x} \theta^{x+\alpha-1}(1-\theta)^{n-x+\beta-1}$$

and the conditional density $\theta|x \sim \text{Beta}(\alpha + x, \beta + n - x)$.

We can construct a Markov chain, in fact a Gibbs sampler, having π as its stationary distribution by using the following transition rule for $(X_t, \theta_t) \mapsto (X_{t+1}, \theta_{t+1})$:

1. choose $\theta_{t+1} \sim \text{Beta}(\alpha + x_t, \beta + n - x_t)$, and
2. choose $X_{t+1} \sim \text{Bin}(n, \theta_{t+1})$.

This transition rule has transition kernel

$$\begin{aligned} &K((x_t, \theta_t), (x_{t+1}, \theta_{t+1})) = f((x_{t+1}, \theta_{t+1})|(x_t, \theta_t)) \\ &\propto \binom{n}{x_{t+1}} \theta_{t+1}^{x_{t+1}+\alpha+x_t-1}(1-\theta_{t+1})^{\beta+2n-x_t-x_{t+1}-1}. \end{aligned}$$

For future reference we note that the subchain $\ldots, X_t, X_{t+1}, \ldots$ is a Markov chain with $X_{t+1}|x_t \sim \text{BetaBin}(n, \alpha + x_t, \beta + n - x_t)$ and transition kernel

$$\begin{aligned} &K(x_t, x_{t+1}) = f(x_{t+1}|x_t) \propto \\ &\binom{n}{x_{t+1}} \frac{\Gamma(\alpha+\beta+n)\Gamma(\alpha+x_t+x_{t+1})\Gamma(\beta+2n-x_t-x_{t+1})}{\Gamma(\alpha+x_t)\Gamma(\beta+n-x_t)\Gamma(\alpha+\beta+2n)}. \end{aligned}$$

■

Theorems about stationary distributions and ergodicity apply when the Markov chain satisfies the three properties of *irreducibility*, *reversibility* and *aperiodicity*, defined in Appendix 6.1. See Robert and Casella (1999, Chap. 4) for a brief description or Meyn and Tweedie (1993) and Resnick (1992) among others for book-length treatments. These properties are assumed true for the rest of this article.

The *stationary distribution* of the Markov chain is also a *limiting distribution*: X_t converges in distribution to $X \sim \pi$. For `MCMC` purposes two useful consequences of our assumptions are that $\frac{1}{M}\sum_{j=1}^{M} h(X_j) \to \mathrm{E}_\pi[h(X)]$ (sometimes called the ergodic theorem) and that a central limit theorem holds.

It is typical in practice to have `MCMC` algorithms begin from an arbitrarily chosen state at time $t = 0$, say, and run for a long time T, say, in the hope that X_T is a draw approximately from π. One typically discards $X_0, \ldots, X_{T-1}$ and estimates $\mathrm{E}_\pi[h(X)]$ as $\frac{1}{M}\sum_{j=T}^{T+M-1} h(X_j)$. A serious practical problem is determining the "burn-in" time T. A second practical problem is determining the correlation between X_t and X_{t+1}, which is used to calculate the variance of the estimate. Perfect sampling avoids both problems because it produces independent draws having distribution π precisely.

Indeed, the major drawback with using MCMC methods is that their validity is only asymptotic: if we run the sampler kernel until the end of time, we are bound to explore the entire distribution of interest; but, since computing and storage resources are not infinite, we are bound to stop the MCMC sampler at some point. The influence of this stopping time on the distribution of the chain is not harmless and in some cases may induce serious biases (Roberts and Rosenthal, 1998). Perfect sampling alleviates this difficulty by producing exactly the same chain as one running an infinite number of steps, by simply replacing the starting time with $-\infty$ and ∞ with 0. And, at no additional cost, it also removes the dependence on the starting value! In other words, the burn-in time becomes infinite and the chain is indeed in the stationary distribution at time 0.

2 Coalescence

The first step in obtaining a perfect sample is to find a way to make X_t independent of the starting value. The answer is to work with transition rules. Let $\{U_t\}$ be a collection of mutually independent random variables, one for each value of t. A *transition rule* ϕ determines X_{t+1} as a function of X_t and U_{t+1}. A common and convenient choice is to let $U_{t+1} \sim \mathrm{Uniform}(0,1)$ and take $X_{t+1} = \phi(x_t, u_{t+1}) = F^{-1}_{X_{t+1}|x_t}(u_{t+1})$, the inverse-cdf function of $X_{t+1}|x_t$ determined by the kernel K and a linear ordering on $\mathcal{X}$. For illustration we return to the Beta-Binomial example.

> **Beta-binomial, continued.** Consider the subchain $\{X_t : t \geq 0\}$ from the previous example, and let $n = 2$, $\alpha = 2$ and $\beta = 4$.

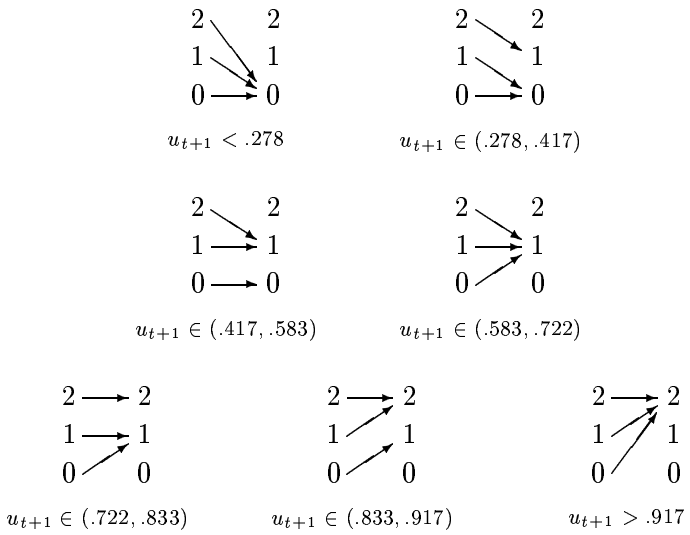


Figure 1: All possible transitions for the Beta-Binomial(2,2,4) example

The state space is $\mathcal{X} = \{0, 1, 2\}$. The transition probabilities are given by the transition matrix

$$P = \begin{pmatrix} .583 & .333 & .083 \\ .417 & .417 & .167 \\ .278 & .444 & .278 \end{pmatrix}$$

and the cdf matrix

$$C = \begin{pmatrix} .583 & .917 & 1.0 \\ .417 & .833 & 1.0 \\ .278 & .722 & 1.0 \end{pmatrix}$$

in which $p_{ij} = \Pr[X_{t+1} = j - 1 | X_t = i - 1]$ and $c_{ij} = \Pr[X_{t+1} \leq j - 1 | X_t = i - 1]$. The entries of C are the break points at which the behavior of the chain changes. Thus we can draw $U_{t+1} \sim \text{Uniform}(0, 1)$ and make the transitions illustrated by Figure 1.

■

Figure 1 shows that if there is ever a time t such that $U_t < .278$, then $X_t = 0$, regardless of the value of X_0. Likewise, if $U_t \in (.583, .722)$ then

$X_t = 1$ or if $U_t > .917$ then $X_t = 2$. The event that the value of X_t does not depend on the value of X_0 is called *coalescence*. More formally, let C_{t_1,t_2}, coalescence between time t_1 and time t_2, be the event that X_{t_2} does not depend on X_{t_1}. For a given transition rule ϕ, C_{t_1,t_2} is a function of $\{U_t : t \in (t_1, t_2]\}$. Conditional on C_{t_1,t_2}, X_{t_2} is a deterministic function of $\{U_t : t \in (t_1, t_2]\}$ and $\{X_s : s \geq t_2\}$ is independent of $\{X_r : r \leq t_1\}$. And because the U_t's are mutually independent coalescence is guaranteed to happen eventually. These facts are collected in the next theorem.

Theorem 1

(a). The minimum time T such that $C_{0,T}$ occurs is a random variable that depends only on $U_1, U_2, \ldots$.

(b). The random variable X_T, the value at coalescence, is independent of X_0.

Proof: Part (a) is immediate by construction, and part (b) follows since X_T is a function only of $U_1, \ldots, U_T$ and not of X_0.

■

Conclusion (b) of Theorem 1 says that T is a time at which the initial state of the chain has "worn off". One might therefore hope that X_T is a draw from the stationary distribution π. This hope is false. It is true that if T^* is a *fixed* time, and X_{T^*} is independent of X_0, then $X_{T^*} \sim \pi$. Unfortunately, T is a random time and in general, $X_T \not\sim \pi$, as the following example illustrates.

> **Two-state.** Consider the Markov chain with state space $\{1, 2\}$ and transition kernel $K(1,1) = K(1,2) = .5$; $K(2,1) = 1$; $K(2,2) = 0$. The stationary distribution is $\pi(1) = 2/3$; $\pi(2) = 1/3$. A little thought shows that coalescence can occur only in $X_T = 1$ and therefore $X_T \not\sim \pi$.

■

3 Propp and Wilson

Propp and Wilson (1996) explained how to take advantage of coalescence while sampling the chain at a fixed time, thereby producing a random variable having distribution π, exactly. Their algorithm is called *Coupling from*

the Past (CFTP), and is based on the two ideas (a) that if a chain were started at time $t = -\infty$ in any state $X_{-\infty}$, it would be in equilibrium by time $t = 0$, so X_0 would be a draw from π and (b) that $C_{-\infty,0}$ would have occurred, so we can calculate X_0 without knowing $X_{-\infty}$. These two things would happen because the chain would have run for an infinite length of time.

To implement CFTP in an algorithm, we use the coalescence strategy. We first find a time $-T$ such that $C_{-T,0}$ occurs, hence $C_{-\infty,0}$ also occurs, and then we calculate X_0.

CFTP goes as follows.

(1). Generate U_0.

(2). Check for $C_{-1,0}$ by applying the transition rule ϕ to $\mathcal{X}$. That is, calculate $I_{-1} \equiv \{\phi(x, U_0) : x \in \mathcal{X}\}$. I_{-1} is the image of $\mathcal{X}$ under one application of ϕ using the random number U_0. If I_{-1} is a singleton, then $C_{-1,0}$ has occurred, $-T = -1$ and X_0 is a draw from π.

(3). Otherwise, move back to time $t = -2$, generate U_{-1}, and calculate $I_{-2} \equiv \{\phi(\phi(x, U_{-1}), U_0) : x \in \mathcal{X}\}$. I_{-2} is the image of $\mathcal{X}$ under two applications of ϕ, using the random numbers U_{-1} and U_0. If I_{-2} is a singleton then $C_{-2,0}$ has occurred, $-T = -2$ and X_0 is a draw from π.

(4). Otherwise, move back to time $t = -3$ and continue.

The algorithm continues backward through time until coalescence occurs. We check for $C_{-t,0}$ by computing I_{-t}, the image of $\mathcal{X}$ obtained by t applications of ϕ, using $U_{-t+1}, \ldots, U_0$ successively. It is important, when taking a backward step, to keep the U_t's that have already been drawn.

Theorem 2 *The CFTP algorithm returns a random variable distributed exactly according to the stationary distribution of the Markov chain.*

Proof: Our proof of CFTP depends on the existence of random variables $\{X_t : t \leq 0\}$ and $\{U_t : t \leq 0\}$ such that

(i). $X_t \sim \pi$ for all t;

(ii). $X_{t+1}|X_t \sim K(X_t, \cdot)$ for all t;

(iii). $X_{t+1} = \phi(X_t, U_{t+1})$ for all t; and

(iv). the U_t's are mutually independent.

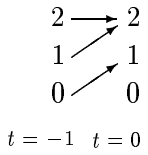
We know such random variables exist both from abstract considerations and because we can simulate them by the following steps (assuming we can simulate from π).

1. Simulate $X_0 \sim \pi$.
2. For $t \in -1, -2, \ldots,$ simulate $X_t \sim K(X_{t+1}, \cdot)$ and $U_{t+1}|X_{t+1}, X_t$.

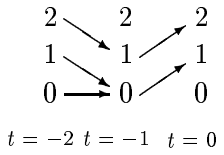
`CFTP` works by simulating these random variables in a different order. It begins by simulating U_t's, simulates a sufficient number of them so that $C_{-T,0}$ has occurred, and then calculates X_0. Appendix 6.2 verifies the detail that T is finite with probability 1.

We use the Beta-Binomial example for illustration.

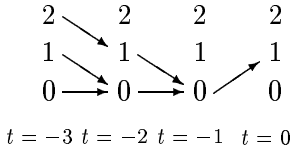
Beta-Binomial, continued. Begin at time $t = -1$ and draw U_0. Suppose $U_0 \in (.833, .917)$. The next picture shows the check for $C_{-1,0}$.



$C_{-1,0}$ did not occur, so we go to time $t = -2$ and draw U_{-1}. Suppose $U_{-1} \in (.278, 417)$. The next picture shows the check for $C_{-2,0}$.



$C_{-2,0}$ did not occur, so we go to time $t = -3$. Suppose $U_{-2} \in (.278, .417)$. The next picture shows the check for $C_{-3,0}$.



$C_{-3,0}$ did occur. We accept $X_0 = 1$ as a draw from π . Note: even though $C_{-3,-1}$ also occurred, we do not accept $X_{-1} = 0$ as a draw from π.

■

In `CFTP`, T and X_0 are dependent random variables. Therefore, a user who gets impatient or whose computer crashes and who therefore restarts runs when T gets too large will generate biased samples. Another algorithm, due to Fill (1998), generates samples from π in a way that is independent of the number of steps.

4 Fill's algorithm

A simple version of Fill's algorithm (`Fill`) is:

1. Arbitrarily choose a time T and state $X_T = z$.
2. Generate $X_{T-1}|X_T$, $X_{T-2}|X_{T-1}$, ... , $X_0|X_1$.
3. Generate $[U_1|X_0, X_1]$, $[U_2|X_1, X_2]$, ... , $[U_T|X_{T-1}, X_T]$
4. Check for $C_{0,T}$.
5. If $C_{0,T}$ has occurred, then accept X_0 as a draw from π
6. Otherwise begin again, possibly with a new T and z.

We note that all random variables are generated according to the kernel K and transition rule ϕ, conditional on $X_T = z$.

There are two ways to prove that `Fill` is correct. We present one here and one in the appendix.

First proof: `Fill` delivers a value only if $C_{0,T}$ occurs, so we need to prove $\Pr[X_0 = x | C_{0,T}, X_T = z] = \pi(x)$. This probability is

$$\begin{aligned}
&\Pr[X_0 = x | C_T(z), X_T = z] \\
&\qquad = \frac{\Pr[X_0 = x | X_T = z] \Pr[C_{0,T} | X_0 = x, X_T = z]}{\sum_{x'} \Pr[X_0 = x' | X_T = z] \Pr[C_{0,T} | X_0 = x', X_T = z]} \\
&= \frac{\Pr[X_0 = x | X_T = z] \Pr[C_{0,T}, X_T = z | X_0 = x] / \Pr[X_T = z | X_0 = x]}{\sum_{x'} \Pr[X_0 = x' | X_T = z] \Pr[C_{0,T}, X_T = z | X_0 = x'] / \Pr[X_T = z | X_0 = x']} \\
&\qquad = \frac{K^T(z,x) \Pr[C_{0,T}, X_T = z] / K^T(x,z)}{\sum_{x'} K^T(z,x') \Pr[C_{0,T}, X_T = z] / K^T(x',z)} \\
&\qquad\qquad = \frac{K^T(z,x)/K^T(x,z)}{\sum_{x'} K^T(z,x')/K^T(x',z)} = \frac{\pi(x)/\pi(z)}{\sum_{x'} \pi(x')/\pi(z)} = \pi(x).
\end{aligned}$$

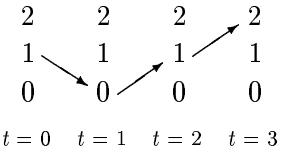
The first two equalities follow from the definition of conditional probability. In the third equality we write $K^T(\cdot,\cdot)$ for the T-step transition probabilities and use reversibility to get $\Pr[X_0 = x | X_T = z] = K^T(z,x)$. We also use the fact that the event $[C_{0,T}, X_T = z]$ depends only on $\{U_1, \ldots, U_T\}$ and is therefore independent of X_0. The last line uses another implication of reversibility, namely: $K^T(z,x)/K^T(x,z) = \pi(x)/\pi(z)$.

■

We follow the Beta-binomial (2,2,4) example through the steps in `Fill`.

Beta-Binomial, continued.

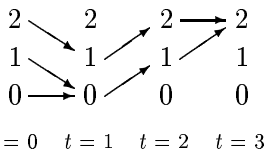
1. We arbitrarily choose $T = 3$ and $X_T = 2$.
2. Our chain is reversible, so $[X_2 | X_3 = 2] = [X_3 | X_2 = 2] = \text{BetaBin}(2,4,4)$. The probabilities are given on page 4. We generate X_2. Suppose it turns out to equal 1. Similarly, $X_1 | X_2 = 1 \sim \text{BetaBin}(2,3,5)$; suppose we get $X_1 = 2$; $X_0 | X_1 = 2 \sim \text{BetaBin}(2,4,4)$; suppose we get $X_0 = 1$. The next picture shows the transitions we've generated.



3. $X_0 = 1$, $X_1 = 0$, $X_2 = 1$ and $X_3 = 2$ imply $U_1 \sim \text{U}(0, .417)$; $U_2 \sim \text{U}(.583, .917)$; and $U_3 \sim \text{U}(.833, 1)$. (See Figure 1.)

Suppose we generate $U_1 \in (.278, .417)$, $U_2 \in (.833, .917)$ and $U_3 > .917$.

4. Check for $C_{0,3}$. The following picture illustrates the check.



5. $C_{0,3}$ occurred; so we accept $X_0 = 1$ as a draw from π.

■

`Fill` depends on an arbitrary choice of T and X_T. To get some feeling for how big T needs to be and whether the choice of X_T is important, we ran `Fill` on a Beta-binomial$(16, 2, 4)$ example. For each of $X_T = 0, 2, \ldots, 16$, we ran `Fill` in a loop with $T = 1, 3, \ldots$ successively until the algorithm returned a value. The whole simulation was repeated 50 times. Figure 2 is a boxplot, sorted by X_T, of the T for which coalescence was achieved. The horizontal axis is the value of X_T which we fixed in advance. The vertical axis is the value of T for which coalescence occurred. The figure shows that coalescence occurred much more quickly when we chose either $X_T = 0$ or $X_T = 16$ than any other value of X_T.

5 Discussion

- A potentially troublesome point is detecting whether coalescence has occurred. In general, starting and keeping track of chains from every state is computationally infeasible. In (partially) ordered state spaces with a *monotone* transition rule it is only necessary to keep track of chains started from the maximal and minimal members. A monotone transition rule is one in which $X_t \geq Y_t \Rightarrow X_{t+1} = \phi(X_t, u_{t+1}) \geq Y_{t+1} = \phi(Y_t, u_{t+1})$. If we use an inverse-cdf function ϕ (with an appropriate linear order) and the kernel K is stochastically monotone, then the transition rule will be monotone.

  This is the case in our example, where a chain started from state 1 is sandwiched between chains started from states 0 and 2. Therefore it is only necessary to keep track of chains started from 0 and 2 to determine whether coalescence has occurred. In fact, if there exist maximal and

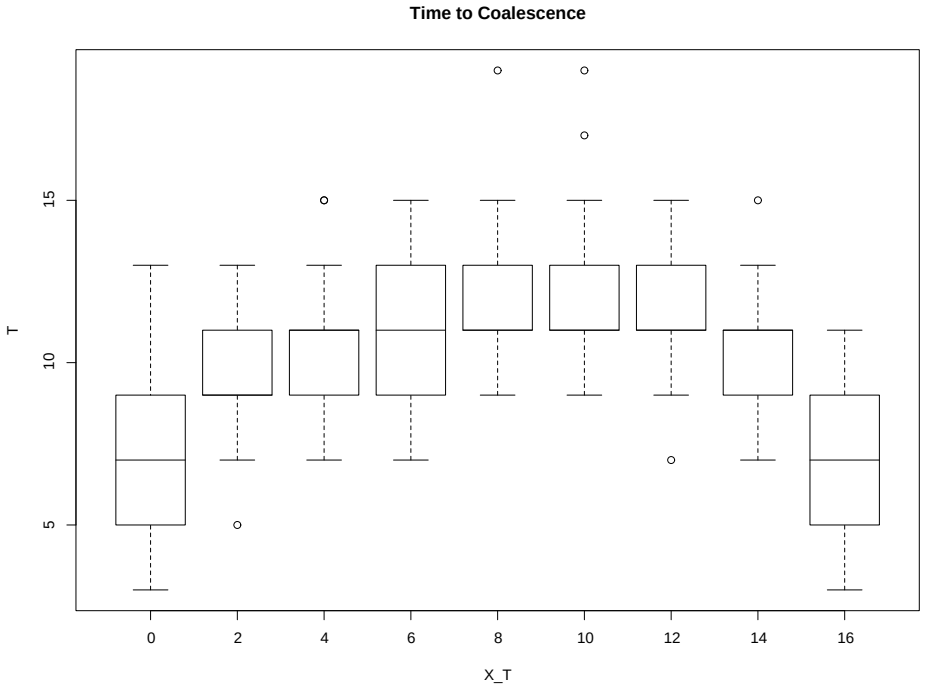


Figure 2: Time to coalescence for 50 runs of Fill's algorithm, for each value of X_T.

minimal elements, coalescence is detectable even with a continuous state space. Non-monotone transition rules or state spaces without minimal and maximal elements require more sophisticated methods. See Fill et al. (1999) or Green and Murdoch (1999) for details and extensions.

- In describing `CFTP` we set T successively equal to -1, -2, ... In fact, any decreasing sequence would do as well. Propp and Wilson (1996) argue that $T = -1, -2, -4, -8, \ldots$ is near optimal. In `Fill`, if X_0 is rejected, or if one is generating many realizations, one may wish to choose new values of T and z for the next proposal. Figure 2 shows that some combinations of (T, z) are more likely to lead to coalescence than others. There is no general theory at present to guide the choice of (T, z). In practice the results of early iterations may guide the choice of (T, z) in later iterations.

- In his original algorithm described here, when checking for coalescence, Fill used constrained uniform variables $U_1, \ldots, U_T$ conditional on $X_0, \ldots, X_T$, generating $[U_1|X_0, X_1]$, $[U_2|X_1, X_2]$, ... , $[U_T|X_{T-1}, X_T]$. This insures that the chain starting in x will end up in z. This is practical as long as sampling from the conditional distribution of the U_i's given the X_i's is not too difficult.

  An alternative to the algorithm described in `Fill` is to generate the U_i's unconditionally. (Typically $U_i \sim U(0, 1)$.) Use these U_i's to check whether X_0 goes to z under T applications of ϕ. If yes, then also check for $C_{0,T}$ and either accept or reject X_0 accordingly. Otherwise, discard both the X_i's and the U_i's and generate another set until finding one that is suitable. Ultimately we will accept X_0 with the desired probability. However, the reader may quickly realize that the implementation of such an alternative is too impractical in real applications.

- Some practical applications of Markov chains iterate between a discrete X and a parameter θ which might be either discrete or continuous. In such cases we can obtain perfect samples from the joint distribution of both X and θ. For example, consider modeling the data Y as a mixture of Normal distributions. The model is usually extended to include indicator variables X, which are not observed but which indicate which Y's come from the same mixture components. Conditional on X, the model is a straightforward collection of Normals. Let θ denote all unknown parameters other than X. The posterior is typically analyzed through a Gibbs sampler that iterates between $[X|\theta]$ and $[\theta|X]$. The

iterates of X form a subchain on a finite state space and are amenable to perfect sampling. Given a perfect sample of X, one can simulate from $[\theta|X]$ to obtain a perfect sample of θ.

This remark extends to other latent variable models, but one must keep in mind that the size of the finite parameter space of X in the mixture example is k^n, which rapidly gets unmanageable unless monotonicity features can be exhibited, as in Hobert et al. (1999).

- To remove the difficulty with continuous state space chains, another promising direction relies on *slice sampling*. This technique is a special case of Gibbs sampling (See Robert and Casella 1999, Sect. 7.1.2) and takes advantage of the fact that the marginal (in x) of the uniform distribution on $\{(x,u);\ u \le \pi(x)\}$ is $\pi(x)$. The idea, detailed in Mira et al. (1999), is that, if x_0' is a variable generated from the uniform distribution on $\{x;\ \pi(x) \ge \epsilon\pi(x_0)\}$, it can also be taken as a variable generated from the uniform distribution on $\{x;\ \pi(x) \ge \epsilon\pi(x_1)\}$ for all x_1's such that $\epsilon\pi(x_0) \le \epsilon\pi(x_1) \le \pi(x_0')$ by a simple accept–reject argument. Therefore, assuming a bounded state space $\mathcal{X}$, if one starts with x_0' generated uniformly on $\mathcal{X}$, a finite sequence $x_0', \ldots, x_T'$ can be used instead of the continuum of possible starting values, with x_i' being generated from a uniform distribution on $\{x;\ \pi(x) \ge \pi(x_{i-1}')\}$, and T being such that $\pi(x_T') \ge \epsilon \sup \pi(x)$. Moreover, slice sampling exhibits natural monotonicity structures which can be exploited to further reduce the number of chains. The practical difficulty of this approach is that uniform distributions on $\{x;\ \pi(x) \ge \epsilon\pi(x_0)\}$ may be hard to simulate, as shown in Casella et al. (1999) in the setup of mixtures.

- Perfect sampling is currently an active area of research. David Wilson maintains a web site of papers on perfect sampling at `http://dimacs.rutgers.edu:80/~dbwilson/exact.html`. The interested reader can find links to articles ranging from introductory to the latest research.

References

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

G. Casella, K.L. Mengersen, C.P. Robert, and D.M. Titterington. Perfect sampling for mixtures. Technical report, CREST, Insee, 1999.

J. A. Fill. An interruptible algorithm for perfect sampling via Markov chains. *Annals of Applied Probability*, 8:131–162, 1998.

James Allen Fill, Motoya Machida, Duncan J. Murdoch, and Jeffrey S. Rosenthal. Extension of Fill's perfect rejection sampling algorithm to general chains. Technical report, The Johns Hopkins University, Dept. of Mathematical Sciences, 1999.

P. J. Green and D. J. Murdoch. Exact sampling for Bayesian inference: towards general purpose algorithms. In J. M. Bernardo, J. O. Berger, A. P. Dawid, and A. F. M. Smith, editors, *Bayesian Statistics 6*, Oxford, 1999. Clarendon Press.

J.P. Hobert, C.P. Robert, and D.M. Titterington. On perfect simulation for some mixtures of distributions. *Statistics and Computing*, 9:287–298, 1999.

S. P. Meyn and R. L. Tweedie. *Markov Chains and Stochastic Stability*. Springer-Verlag, New York, 1993.

A. Mira, J. Møller, and G. O. Roberts. Perfect slice sampler. Technical Report R-99-2020, Dept. of Mathematical Science, Aalborg University, 1999.

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

S. I. Resnick. *Adventures in Stochastic Processes*. Birkhäuser, Boston, 1992.

C. P. Robert and G. Casella. *Monte Carlo Statistical Methods*. Springer-Verlag, New York, 1999.

G.O. Roberts and J.S. Rosenthal. Markov chain monte carlo: Some practical implications of theoretical results (with discussion). *Canadian Journal of Statistics*, 26:5–32, 1998.

E. Thönnes. A primer on perfect sampling. Technical report, Department of Mathematical Statistics, Chalmers University of Technology, 1999.

6 Appendix

6.1 A Markov Chain Glossary

We will work with discrete state space Markov chains. The following definitions can be extended to continuous state spaces as long as the usual measurability complications are carefully dealt with.

A Markov chain $X_1, X_2, \ldots$, is *irreducible* if the chain can move freely throughout the state space; that is, for any two states x and x', there exists an n such that $\Pr[X_n = x' | X_0 = x] > 0$. Moreover, as the chains we are considering are all *positive*, that is, the stationary distribution is a probability distribution, irreducibility also implies that the chain is *recurrent*. A recurrent chain is one in which the average number of visits to an arbitrary state is infinite.

A state x has *period* d if $P(X_{n+t} = x | X_t = x) = 0$ if n is not divisible by d, d being the largest integer with this property. For example, if a chain starts ($t = 0$) in a state with period 3, the chain can only return to that state at times $t = 3, 6, 9, \ldots$. If a state has period $d = 1$, it is *aperiodic*. In an irreducible Markov chain, all states have the same period. If that period is $d = 1$, the Markov chain is aperiodic.

We then have the following theorems.

Theorem 3 Convergence to the stationary distribution *If the countable state space Markov chain $X_1, X_2, \ldots$, is positive, recurrent and aperiodic with stationary distribution π, then from every initial state*

$$X_n \to X \sim \pi.$$

A positive, recurrent and aperiodic Markov chain is often called *ergodic*, a name also given to the following theorem, a cousin of the Law of Large Numbers.

Theorem 4 Convergence of Sums *If the countable state space Markov chain $X_1, X_2, \ldots$, is ergodic with stationary distribution π, then from every initial state*

$$\frac{1}{n} \sum_{i=1}^{n} h(X_i) \to E_\pi h(X)$$

provided $E_\pi |h(X)| < \infty$

Adding the property of *reversibility* will get us a Central Limit Theorem. A Markov chain is *reversible* if the distribution of X_{t+1} conditional on $X_{t+2} = x$ is the same as the distribution of X_{t+1} conditional on $X_t = x$. For any set B we have

$$\sum_{y \in \mathcal{X}} \sum_{x \in B} K(y, x) = \sum_{y \in \mathcal{X}} \sum_{x \in B} K(x, y)$$

so the transition probabilities are the same whether we go forward or backward along the chain.

Theorem 5 Central Limit Theorem *If the countable state space Markov chain $X_1, X_2, \ldots$, is ergodic and reversible with stationary distribution π, then from every initial state*

$$\frac{1}{\sqrt{n}} \sum_{i=1}^{n} [h(X_i) - E_\pi h(X)] \to \mathcal{N}(0, \sigma^2),$$

provided $0 < \sigma^2 = \text{Var } h(X_0) + \sum_{i=1}^{\infty} \text{Cov}_\pi(h(X_0), h(X_i)) < \infty$

6.2 Proof of Theorem 2

We will establish the detail mentioned in the proof of Theorem 2, that coalescence occurs at some finite time with probability 1. We adapt the proof presented in Thönnes (1999).

As $\{X_t\}$ is irreducible, there exists an n such that $K^n(x, y) > 0$ for all states x and y in $\mathcal{X}$. The use of a monotone transition rule ϕ implies (but is not necessary for) $\varepsilon \equiv \Pr[C_{0,n}] > 0$. The events $C_{-n,0}, C_{-2n,-n}, \ldots$ all have probability ε and are independent because $C_{-(i+1)n,-in}$ depends only on $U_{-(i+1)n+1}, U_{-(i+1)n+2}, \ldots, U_{-in}$, which are independent of all of the other U's.

Finally, we observe that

$$\begin{aligned} P(\text{ No coalescence after } I \text{ blocks of size } n) &\leq \prod_{i=1}^{I} (1 - \Pr[C_{-(i)n,-(i-1)n}]) \\ &< (1 - \varepsilon)^I \\ &\to 0 \text{ as } I \to \infty, \end{aligned}$$

showing that the probability of coalescence is 1.

We can, in fact, make the stronger conclusion that the coalescence time is almost surely finite by noting that

$$\sum_{i=1}^{\infty} \Pr[C_{-(i+1)n,-in}] = \infty \Rightarrow \Pr[C_{-(i+1)n,-in} \text{ infinitely often }] = 1,$$

from the Borel-Cantelli Lemma, and therefore, with probability 1, there exists a finite T such that $C_{-T,0}$ occurs.

■

6.3 Alternate Proof of `Fill`

We can view `Fill` as a rejection algorithm: generate and propose $X_0 = x$; then accept x as a draw from π if $C_{0,T}$ has occurred. The proposal distribution is the T-step transition density $K^T(z, \cdot)$. In this section we use the notation $C_T(z)$ to denote the event $[C_{0,T} \cup X_T = z]$ and $x \to z$ to denote the event $[X_0 = x \cup X_T = z]$.

`Fill` is a valid rejection algorithm if we accept $X_0 = x$ with probability

$$\frac{1}{M} \frac{\pi(x)}{K^T(z,x)} \text{ where } M \geq \sup_x \frac{\pi(x)}{K^T(z,x)}.$$

From detailed balance we can write $\pi(x)/K^T(z,x) = \pi(z)/K^T(x,z)$ and, since $\Pr[C_T(z)] \leq K^T(x', z)$ for any x', and hence $\Pr[C_T(z)] \leq \min_{x'} K^T(x', z)$, we have the bound

$$\frac{\pi(x)}{K^T(z,x)} = \frac{\pi(z)}{K^T(x,z)} \leq \frac{\pi(z)}{\min_{x'} K^T(x',z)} \leq \frac{\pi(z)}{\Pr[C_T(z)]} = M.$$

So we accept $X_0 = x$ with probability $\frac{1}{M}\frac{\pi(x)}{K^T(z,x)}$, which is quite difficult to compute. However,

$$\frac{1}{M} \frac{\pi(x)}{K^T(z,x)} = \frac{\Pr[C_T(z)]}{\pi(z)} \frac{\pi(x)}{K^T(z,x)} = \frac{\Pr[C_T(z)]}{\pi(z)} \frac{\pi(z)}{K^T(x,z)} = \frac{\Pr[C_T(z)]}{K^T(x,z)},$$

where we have again used detailed balance. But now, because $C_T(z)$ entails $x \to z$, we have $\frac{\Pr[C_T(z)]}{K^T(x,z)} = \frac{\Pr[C_T(z), x \to z]}{\Pr[x \to z]} = \Pr[C_T(z) | x \to z]$, exactly the event that `Fill` simulates.

■

Finally, note that the algorithm is more efficient if the acceptance probability $1/M$ is as large as possible, so choosing z to be the state that maximizes $\Pr[C_T(z)]/\pi(z)$ is a good choice. This, also, will be a difficult calculation, but in running the algorithm, these probabilities can be estimated.