8-14-2001

Practical Regeneration for Markov Chain Monte Carlo Simulation

Anthony Brockwell  
*Carnegie Mellon University*, abrock@stat.cmu.edu

Joseph B. Kadane  
*Carnegie Mellon University*, kadane@stat.cmu.edu

Follow this and additional works at: [http://repository.cmu.edu/statistics](http://repository.cmu.edu/statistics)

Part of the [Statistics and Probability Commons](http://repository.cmu.edu/statistics)
Practical Regeneration for Markov Chain Monte Carlo Simulation

Anthony E. Brockwell and Joseph B. Kadane

August 14, 2001

Abstract

Regeneration is a useful tool in Markov chain Monte Carlo simulation, since it can be used to side-step the burn-in problem and to construct estimates of the variance of parameter estimates themselves. Unfortunately, it is often difficult to take advantage of, since for most chains, no recurrent atom exists, and it is not always easy to use Nummelin’s splitting method to identify regeneration points. This paper describes a simple and practical method of obtaining regeneration in a Markov chain. The application of this method in simulation is discussed, and examples are given.

Keywords: Markov chain Monte Carlo, regenerative simulation, splitting, atoms, burn-in, convergence, parallel processing

1 Introduction

Markov chain Monte Carlo (MCMC) methods have become popular in the last decade as a tool for exploring properties of distributions which are known only up to a constant of proportionality. The basic idea is to construct an ergodic Markov chain whose limiting distribution is the same as the distribution of interest π (called the target distribution). Further details, as well as information about various aspects of implementation, are given in a variety of sources, including Tierney (1994), Robert and Casella (1999) and Gilks et al. (1996). Development of the underlying probabilistic theory of Markov chains taking values in a general state-space can be found in Revuz (1975), Nummelin (1984), and Meyn and Tweedie (1993).

Current methods of constructing Markov chains for Monte Carlo simulation have two key limitations. The first is the burn-in problem, that is, uncertainty about how long it takes before the chain is (by some measure) sufficiently close to its limiting distribution. The second is the inherent correlation between successive elements of the chain. This correlation makes it difficult to estimate the variance of the Monte Carlo estimates. In this paper, we propose a simple and practical method, based on the concept of regeneration, for dealing with both of these problems.

Early work on regenerative simulation can be found in Crane and Iglehart (1975a), Crane and Iglehart (1975b) and Crane and Lemoine (1977), while more recent discussion can be found in Ripley (1987) and Mykland et al. (1995). Loosely speaking, a regenerative process “starts
again” probabilistically at each of a set of random stopping times, called regeneration points. Furthermore, conditioned on any particular stopping time, the future of the process is independent of the past. (Probably the simplest example of a set of regeneration points is the set of hitting times of some arbitrary fixed state. However, this example is not very interesting for most continuous state-space chains, since the time between such regeneration points can be infinite.)

If regeneration points can be identified in the Markov chain, then tours of the chain between these times are (on an appropriate probability space) independent identically distributed (i.i.d.) entities. Thus if a fixed number of tours is generated, it makes no sense to discard any initial tours, or indeed, any part of the first tour, since properties of the “clipped” tour would be quite different from properties of the other un-clipped tours. In this sense, the problem of burn-in is avoided. In addition, since the tours are i.i.d, when estimating the expectation of an arbitrary function of a parameter, the variance of the estimator itself can be estimated. There are further benefits as well. In constructing a regenerative chain, it is not necessary for all the tours to be generated on a single processor, and it is therefore possible to use parallel processors to generate a single long chain with the desired target distribution.

For continuous state-space chains, Mykland et al. (1995) propose using the splitting method of Nummelin (1978) to identify regeneration points in a Markov chain. The method augments the state-space so that the states include an indicator variable which is equal to one at each regeneration point. While their approach is fundamentally sound, it is not often used since it requires a somewhat detailed analysis of the underlying transition probability kernel of the Markov chain.

In this paper, we adopt a different approach to regeneration. The idea is to modify the initial target distribution π by mixing it with a point mass concentrated on an “artificial atom” α which is outside the state-space \( \mathbb{R}^d \). It is then a straightforward matter (for instance, using the Metropolis-Hastings Algorithm) to construct a Markov chain with the new target distribution. For this chain, the state α is Harris-recurrent (i.e. with probability one, it occurs infinitely many times). By the Markov property, the times at which the new chain hits α are regeneration times. On the face of it, this might not seem particularly useful, since the new chain doesn’t have the desired target distribution. However, to recover an ergodic chain with limiting distribution π, it is sufficient simply to delete every occurrence of the state α from the new chain. The points immediately after the (deleted) occurrences of the state α are then regeneration times in a Markov chain with limiting distribution π. (Moller and Nicholls, 2001, use a generalized form of this mixed target distribution for purposes of perfect-sampling, which allows one to obtain unbiased parameter estimates. They do not, however address applications in regenerative simulation or parallel processing.)

Section 2 introduces our method. Section 3 describes how it can be used for purposes of regenerative simulation. An explicit algorithm is given for estimating the expectation of a function of an unknown parameter as well as the variance of the estimator itself. (A parallel processing version of the algorithm is given in Appendix B.) In Section 4, we present examples of application of the method. The theoretical development of the method is given in Appendix A.
Figure 1: The beginning of a realization of a Markov chain \( \{Y_t\} \) taking values in \( E = \mathbb{R} \cup \{\alpha\} \). Here \( Y_t = \alpha \) for \( t \in \{0, 5, 9, 10, 11, 16\} \).

## 2 The Method

In this section, we give a qualitative description of the method we use to construct a regenerative Markov chain with a specified limiting distribution \( \pi \). For the sake of readability, formal statements and proofs of our results are relegated to Appendix A.

The key idea is to enlarge the state-space from \( \mathbb{R}^d \) to

\[
E = \mathbb{R}^d \cup \alpha,
\]

where \( \alpha \) is a new state called the *artificial atom*. Then it is possible (as described later in the paper) to construct a Markov chain \( \{Y_t, \ t = 0, 1, 2, \ldots\} \) with limiting distribution

\[
\pi^*_p(A) = (1-p)\pi(A \setminus \alpha) + pI_A(\alpha)
\]

(defined on appropriate subsets \( A \) of \( E \)) instead of \( \pi \), where \( p \) is some constant in the interval \((0, 1)\). The new limiting distribution \( \pi^*_p \) is a mixture distribution which assigns mass \( p \) to the new state \( \alpha \) and mass \((1-p)\) to the original distribution \( \pi \). For reasons which will become apparent later, the initial state \( Y_0 \) is set equal to \( \alpha \). An example of the beginning of a realization of such a chain is shown in Figure 1.

Next let \( \tau_Y(j) \) denote the \( j \)th time after time zero that the chain \( \{Y_t\} \) hits the state \( \alpha \), with \( \tau_Y(0) = 0 \), so that

\[
\tau_Y(j) = \min\{k > \tau_Y(j-1) : Y_k = \alpha\}, \quad j = 1, 2, 3, \ldots,
\]

and define the tours \( Y^1, Y^2, \ldots \) to be the segments of the chain between the hitting times, that is

\[
Y^j = \{Y_t, \ \tau_Y(j-1) < t \leq \tau_Y(j)\}, \quad j = 1, 2, \ldots.
\]
Figure 2: The beginning of the chain \( \{ Z_t \} \) obtained from the chain \( \{ Y_t \} \) shown in Figure 1.

For example, the first five tours of the chain \( \{ Y_t \} \) shown in Figure 1 are \( Y^1 = \{ Y_1, Y_2, Y_3, Y_4, Y_5 \} \), \( Y^2 = \{ Y_6, Y_7, Y_8, Y_9 \} \), \( Y^3 = \{ Y_{10} \} \), \( Y^4 = \{ Y_11 \} \), and \( Y^5 = \{ Y_{12}, Y_{13}, Y_{14}, Y_{15}, Y_{16} \} \).

The next step is to recover a regenerative chain with limiting distribution \( \pi \). This turns out to be quite simple. Define the chain \( \{ Z_t \} \) to be exactly the chain \( \{ Y_t \} \), with every occurrence of the state \( \alpha \) removed. Noting that the state \( \alpha \) occurs at the end of each tour \( Y^j \), this means that \( \{ Z_t \} \) can be constructed by stringing together the tours \( Y^j \) whose length is larger than one, after removing the last element (which is equal to \( \alpha \)) from each one. Tours \( Z^j \) of \( \{ Z_t \} \) are then defined by their correspondence to the truncated tours of \( \{ Y_t \} \). Let \( T_j \) denote the time at which the \((j + 1)\)th tour of \( \{ Z_t \} \) begins, for \( j = 0, 1, 2, \ldots \). (So \( T_0 \) is always equal to zero.)

To illustrate this construction, Figure 2 shows the first part of the sequence \( \{ Z_t \} \) corresponding to the (partial) chain \( \{ Y_t \} \) given in Figure 1. In this case, the first three tours are \( Z^1 = \{ Y_1, Y_2, Y_3, Y_4 \} \), \( Z^2 = \{ Y_6, Y_7, Y_8 \} \), and \( Z_3 = \{ Y_{12}, Y_{13}, Y_{14}, Y_{15} \} \), the elements of the Markov chain \( \{ Z_t \} \) are \( Z_0 = Y_1, Z_1 = Y_2, Z_2 = Y_3, Z_3 = Y_4, Z_4 = Y_6, \ldots \), and the times at which tours of \( \{ Z_t \} \) begin are \( T_0 = 0, T_1 = 4, T_2 = 7, \ldots \).

A formal description of the construction of \( \{ Z_t \} \) is given in Appendix A, where it is also proved that a chain \( \{ Z_t \} \) constructed in this manner is an ergodic Markov chain with limiting distribution \( \pi \), and that the times \( T_0, T_1, \ldots \) are regeneration points for the chain.

3 Application of the Method

In this section, we review the basic technique of estimation using a regenerative chain, as discussed in Crane and Lemoine (1977). We also give a detailed algorithm for estimating expected values of functions of the target distribution, based on our method of obtaining a regenerative chain.
3.1 Estimating the Variance of an Estimator

Often one is interested primarily in estimating

\[ h_\pi = \int_{\mathbb{R}^d} h(x)d\pi(x) \]

for some function \( h(\cdot) \). (Assume that \( h_\pi \) exists.) Let tours \( Z^j \), tour lengths \( N_j \), and cumulative tour lengths \( T_j \) of the Markov chain \( \{Z_t\} \) with limiting distribution \( \pi \) be as defined in the previous section. Also let

\[ h_j = \sum_{t=T_{j-1}}^{T_j - 1} h(Z_t). \tag{4} \]

Then Crane and Lemoine (1977) suggest using the ratio estimator

\[ \hat{h}_n = \frac{\sum_{j=1}^{n} h_j}{T_n}, \tag{5} \]

which converges almost surely to \( h_\pi \). Furthermore, since the \( h_j \)s and the \( N_j \)s are i.i.d., if they have finite variances, then

\[ \sqrt{n}(\hat{h}_n - h_\pi) \]

converges to a \( N(0, \sigma^2) \) distribution. Ripley (1987), and Mykland et al. (1995) propose using

\[ \sigma^2 = \frac{n \sum_{j=1}^{n} (h_j - \hat{h}_n N_j)^2}{T_n^2} \tag{6} \]

as an estimator of \( \sigma^2 \), but only when the coefficient of variation \( c_n \) of \( T_n/n \) is sufficiently small (Mykland et al., 1995, suggest that the estimator should not be used if the coefficient is larger than 0.01.) The coefficient of variation \( c_n \) depends on the distribution of the tour lengths \( N_j \), which is usually not known. However, it may be estimated by

\[ \hat{c}_n = \frac{1}{n} \sum_{j=1}^{n} (N_j/T_n - 1/n)^2, \tag{7} \]

and since \( c_n \) is proportional to \( n^{-1} \), it is possible, based on an estimate \( \hat{c}_{n1} \), to guess how many additional tours \( n_2 \) would be required in order to ensure that \( \hat{c}_{n1+n_2} \) would be less than (say) 0.01.

Finally, it is worth noting that there are many alternatives to the estimator (5) of \( h_\pi \). We do not discuss them here, as the method given above is sufficient for our purposes. Instead the reader is referred to Section 7 of Crane and Lemoine (1977) for more details.

3.2 Constructing a Chain with Limiting Distribution \( \pi_p^* \)

The method described in Section 3.1 relies on the construction of a regenerative chain \( \{Z_t\} \). The construction of \( \{Z_t\} \), in turn, relies on the construction of a chain \( \{Y_t\} \) with \( Y_0 = \alpha \) and limiting
distribution $\pi_c^*$. In this section we present a simple way, given an existing method for generating a chain with limiting distribution $\pi$, of constructing a chain with limiting distribution $\pi_c^*$. In the following section, we give an explicit algorithm for applying the method of Section 3.1.

Let the probability density of $\pi$ be $\beta f(\cdot)$, where $\beta$ is an unknown constant, and suppose that the transition kernel $P(\cdot, \cdot)$ of a “reasonably good” (i.e., well-mixing) Markov chain taking values in $\mathbb{R}^d$, with limiting distribution $\pi$, is already known. Let $\phi(\cdot)$ denote the probability density of some “re-entry” distribution, satisfying the property that $\phi(x) > 0 \Leftrightarrow f(x) > 0$, and let $p^*$ be some positive constant. Then the following algorithm can be used to construct a Markov chain $\{Y_t\}$ taking values in $E$, with $Y_0 = \alpha$ and limiting distribution $\pi_c^*$, where $p = \beta p^*(1 + \beta p^*)^{-1}$.

**Algorithm 3.1:**

Step 1. Set $Y_0 = \alpha$. Set $t = 0$.

Step 2. Is $Y_t = \alpha$? If so, set $Y_{t+1} = \alpha$. Otherwise choose $Y_{t+1} \in \mathbb{R}^d$ by applying the transition kernel $P(\cdot, \cdot)$ to $Y_t$.

Step 3. Replace $t$ by $t + 1$. Choose a proposal point $W \in E$ as follows. If $Y_t \in \mathbb{R}^d$, then set $W = \alpha$. Otherwise choose $W \in \mathbb{R}^d$ by sampling from the re-entry density $\phi$.

Step 4. Compute
\[
a(Y_t, W) = \begin{cases} 
\min(1, \frac{\phi(W)}{f(Y_t)}), & W = \alpha \\
\min(1, \frac{\phi(Y_t)}{p^* \phi(W)}), & W \in \mathbb{R}^d.
\end{cases}
\]

With probability $a(Y_t, W)$, set $Y_t = W$. Otherwise leave $Y_t$ unchanged.

Step 5. Go back to Step 2.

Algorithm 3.1 can be thought of as a hybrid algorithm, which alternates between applying two different probability kernels. The first, applied in Step 2, is a simple extension of a standard kernel for sampling from $\pi$. Step 2 is trivial if $Y_t = \alpha$. Otherwise, it involves the application of a standard $\pi$-invariant MCMC kernel, such as a Metropolis-Hastings or Gibbs kernel to generate $Y_{t+1}$ from $Y_t$. (For a multi-component Metropolis-Hastings sampler or a Gibbs sampler, the kernel represents a complete cycle of component updates.) This extension has invariant distribution $\pi_c^*$, with $p = \beta p^*(1 + \beta p^*)^{-1}$, but does not allow the chain to move back and forth between $\alpha$ and $\mathbb{R}^d$. The second kernel, applied in Steps 3 and 4, also has invariant distribution $\pi_c^*$, but allows movement between $\alpha$ and $\mathbb{R}^d$. In Appendix A, it is proved that the chain $\{Y_t\}$ generated by Algorithm 3.1 is ergodic with limiting distribution $\pi_c^*$.

Choice of the parameter $p^*$ and the re-entry distribution $\phi$ is important. Since Algorithm 3.2 discards tours of length one (which consist only of the state $\alpha$), it is computationally inefficient to generate a chain with a large number of successive occurrences of the state $\alpha$. Therefore it is desirable to have high acceptance probabilities for jumps from $\alpha$ back into $\mathbb{R}^d$. This can be achieved by choosing small values of $p^*$ relative to “typical” values of $\phi(y)/f(y)$. On the other hand, choosing $p^*$ in this manner leads to small probabilities of accepting jumps from $\mathbb{R}^d$ to $\alpha$,
which in turn leads to long tour lengths. If the tour lengths become too long, then Algorithm 3.2 
(which is introduced in the next section) can take an unbearably long amount of time to finish.

The re-entry distribution \( \phi \) should be equivalent to the original target distribution \( \pi \). That is, 
\( \phi \) should assign positive mass to a set if and only if \( \pi \) does as well. When \( \pi \) is a continuous 
distribution (i.e. it has a density with respect to Lebesgue measure), a relatively effective method 
of selecting \( p^* \) and \( \phi \) is as follows. Run a chain \( \{X_t\} \) using the original (non-regenerative) kernel 
\( P(x, \cdot) \) for a short amount of time, \( t = 0, 2, \ldots, n_b + n_x - 1 \), discard the first \( n_b \) iterations, and 
compute the sample mean and variance

\[
\begin{align*}
m_x & = \frac{1}{n_x} \sum_{i=n_b}^{n_b+n_x-1} X_i \\
\Lambda_x & = \left( \frac{1}{n_x} \sum_{i=n_b}^{n_b+n_x-1} X_i X_i^T \right) - m_x m_x^T.
\end{align*}
\]

Then let \( \phi \) be a multivariate normal distribution with mean \( m_x \) and variance matrix \( \Lambda_x \). To get 
a “reasonable” value for \( p^* \), choose

\[
\log(p^*) = \left[ \frac{1}{n_x} \sum_{i=n_b}^{n_b+n_x-1} \log(\pi(X_i)) \right] - \log(\phi(m_x)) - k_x,
\]

where \( k_x \) is some positive constant chosen to ensure that tour lengths are not too long, while 
at the same time, the time spent in the state \( \alpha \) is minimized. In trial problems, we find that 
choosing \( k_x = 7 \) when \( d = 1 \) leads to tour lengths of around 1,500, and that, generally, increasing 
the dimension \( d \) leads to longer tour lengths, unless \( k_x \) is reduced.

Algorithm 3.1 is particularly useful since it provides a simple means of “wrapping up” an existing 
MCMC algorithm for sampling from \( \pi \) to yield a Markov chain with limiting distribution \( \pi_p^\alpha \). 
However, it is important to note that Algorithm 3.1 does not necessarily improve mixing of the 
underlying kernel \( P(\cdot, \cdot) \) used in Step 2. If the underlying kernel mixes badly, then the resulting 
chain \( \{Y_t\} \) is also likely to mix badly. Of course, Algorithm 3.1 is not the only way to construct a 
chain \( \{Y_t\} \) with limiting distribution \( \pi_p^\alpha \). Another approach to constructing such a chain is simply 
to find the densities of \( \pi_p^\alpha \) and the proposal distribution with respect to a measure defined on 
appropriate subsets of the enlarged state-space \( E \), and apply the standard Metropolis-Hastings 
Algorithm. While this approach is more direct, it cannot be implemented by constructing a 
wrapper for a pre-established Markov transition kernel with limiting distribution \( \pi \).

### 3.3 A Regenerative MCMC Algorithm

In Section 3.2, we gave a straightforward method of generating a Markov chain \( \{Y_t\} \) with \( Y_0 = \alpha \) 
and limiting distribution \( \pi_p^\alpha \). Let \( Q(\cdot, \cdot) \) denote the transition probability kernel of this chain.

The next step is to use this chain to estimate the expectation (with respect to the probability 
measure \( \pi \)) of a “function of interest” \( h(\cdot) \). Given a desired number of tours \( M > 0 \), the
following algorithm builds on Algorithm 3.1 and uses our method to construct an estimate \( \hat{h}_{M} \) of \( \int h(x) d\pi(x) \), as well as an estimate of the variance of \( \hat{h}_{M} \) itself.

**Algorithm 3.2:**

Step 1. Initialization: Set \( Y_0 = \alpha \), \( m = 0 \), \( t = 0 \), \( T_0 = 0 \).

Step 2. Generate \( Y_{t+1} \) given \( Y_t \) by sampling from \( Q(Y_t, \cdot) \). Replace \( t \) by \( t + 1 \).

Step 3. Is \( Y_{t+1} = \alpha \)? If so, go back to Step 2.

Step 4. Set \( n = 1 \). Set \( x = h(Y_{t+1}) \).

Step 5. Generate \( Y_{t+1} \) given \( Y_t \) by sampling from \( Q(Y_t, \cdot) \). Replace \( t \) by \( t + 1 \).

Step 6. Is \( Y_{t+1} = \alpha \)? If so, go to Step 8.

Step 7. Replace \( x \) by \( x + h(Y_{t+1}) \). Replace \( n \) by \( n + 1 \). Go back to Step 5.

Step 8. Replace \( m \) by \( m + 1 \). Set \( h_m = x \), \( N_m = n \) and \( T_m = T_{m-1} + N_m \).

Step 9. Is \( m < M \)? If so, go back to Step 2. Otherwise compute the desired estimate and its estimated variance using equations (5) and (6), respectively.

Most of Algorithm 3.2 is trivial. The only slightly complicated part is in Steps 2 and 5, where it is necessary to generate iterations of a Markov chain with limiting distribution \( \pi_\alpha^* \). (Algorithm 3.1 provides one method of doing this.) Intuitively, Algorithm 3.2 works as follows. It repeatedly generates tours of a Markov chain \( \{Y_t\} \) with limiting distribution \( \pi_\alpha^* \). Tours of length one (that is, tours which consist only of the state \( \alpha \)) are thrown out. Remaining tours are truncated just before they hit \( \alpha \) so as to become tours of a chain \( \{Z_t\} \) with limiting distribution \( \pi \). The sum over each tour of the function of interest, evaluated at the chain’s state, is recorded, as is the length of the tour. Once a total of \( M \) tours have been generated, the algorithm terminates, and equations (5) and (6) can be used to obtain the desired estimates.

It might be tempting to use an alternative approach, stopping the procedure at a fixed time and discarding the last incomplete tour. However, this would not be a valid approach. It would introduce bias since the discarded tour is more likely to be a long tour than a short tour, and different length tours typically have different probabilistic properties.

As mentioned previously, regeneration also provides a convenient means of using parallel processing for MCMC simulation. A detailed parallel processing version of Algorithm 3.2 is given in Appendix B.

On a final note, before using the variance estimate computed in Step 9 of Algorithm 3.2, the estimate \( \hat{c}_M \) (c.f. (7)) of the coefficient of variation should be checked. The variance estimate should not be considered reliable if \( \hat{c}_M \) is too large. Furthermore, it is important to keep in mind that even though Algorithm 3.2 side-steps the burn-in problem in the sense that there is no point in removing any initial portion of the chain, convergence of estimates \( \hat{h}_n \) is still an issue. Consider, for instance, the following example, suggested to us by Geoff Nicholls. Suppose
that a (fair) coin toss is being simulated, so that $E$ can be regarded as $\{H,T,\alpha\}$, and that a chain $\{Y_t\}$ is constructed using Algorithm 3.1, with $P(H,H)$ and $P(T,T)$ close to one, and re-entry distribution $\phi(H) = 1 - \phi(T) = \zeta$. If $\zeta$ is small, then using Algorithm 3.2, one is likely to observe many length-one tours which consist of the state $T$, with an occasional long tour consisting mostly of the state $H$. In this case, if not enough tours are generated, it would be easy to observe only the state $T$ and conclude that the coin never comes up heads. (This example also illustrates the dangers of attempting to improve estimates by removing, say, the first few elements of each tour.)

4 Examples

To examine the performance of Algorithm 3.2 (or its parallel processing version, Algorithm 8.1, given in Appendix B), we apply to it several problems.

4.1 A Mixture of Gaussian Distributions

The first problem is relatively simple. The target distribution is a mixture of two Gaussian distributions, one with mean 4 and variance 1, and one with mean 10 and variance 1. The mixture assigns equal probabilities to each component. Thus $\pi(x) = \frac{1}{2} \int_{\mathbb{R}} \varphi(x; 4,1) + \varphi(x; 10,1) \, dx$, where $\varphi(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma}} \exp(-\frac{(x - \mu)^2}{2\sigma^2})$.

The object is to estimate the components of the vector $h_\pi = (\int_{\mathbb{R}} x d\pi(x), \int_{\mathbb{R}} x^2 d\pi(x))^T$. In this case, it is easily verified that $h_\pi = (7, 59)^T$.

Algorithm 3.1 can be used to construct a Markov chain $\{Y_t, t = 0, 1, 2, \ldots\}$ with $Y_0 = \alpha$ and limiting distribution $\pi^*_p$. For the underlying kernel $P(x, \cdot)$, we use the Metropolis-Hastings algorithm with target density

$$\exp(x - 4)^2 + \exp(x - 10)^2,$$

and a random walk proposal which, from a given current point, takes normally distributed steps with mean zero and variance one. Choosing the parameter $p^*$ and the re-entry distribution $\phi$ using the method described in Section 3.2, with $n_0 = n_x = 200$, we obtain $p^* \simeq \exp(-5.68)$, $m_x \simeq 9.37$, and $\Lambda_x \simeq 4.74$.

Having determined the transition kernel for a chain $\{Y_t\}$ with limiting distribution $\pi^*_p$, we can apply Algorithm 8.1 (the parallel processing version of Algorithm 3.2) to estimate $h_\pi$. We use a set of 44 dual-CPU Pentium-III-based machines with the Linux operating system to implement Algorithm 8.1 with the parameters given above. Communication between processors is handled using the MPICH implementation of the Message Passing Interface (MPI) standard (see, e.g., Gropp et al., 1999), and random number seeds are set differently for each process.

After $n = 1,000,000$ tours (which takes approximately one minute using the machines described above), the total length of the resulting regenerative chain is $T_n = 1,479,080,020$, the estimated
<table>
<thead>
<tr>
<th>Function</th>
<th>Estimate</th>
<th>Std. Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>$E_\pi[X]$</td>
<td>6.999784</td>
<td>0.001412</td>
</tr>
<tr>
<td>$E_\pi[X^2]$</td>
<td>58.997235</td>
<td>0.019765</td>
</tr>
</tbody>
</table>

Table 1: Estimates for parameters of the mixed Gaussian distribution of Section 4.1.

Figure 3: A histogram of the 1,000,000 tour lengths for the Markov chain of Section 4.1 with mixed Gaussian target distribution.

The coefficient of variation of $T_n/n$ is 0.000001129, and we obtain estimates, along with estimated standard errors of the estimates, as shown in Table 1.

A histogram of the resulting set of 1,000,000 tour lengths is shown in Figure 3.

4.2 A Time Series Example

Next consider a more interesting example. Figure 4 shows the yearly sunspot counts $X = \{X_t, \ t = 1, 2, \ldots, 252\}$ (of numbers of “spots” on the surface of the sun) from the year 1749 to the year 2000, inclusive. A simple model for the data, after the sample mean is subtracted, is the standard AR(2) model

$$X_t = \phi_1 X_{t-1} + \phi_2 X_{t-2} + Z_t,$$

where $\{Z_t\}$ is a Gaussian white noise process with variance $\sigma^2 = 1/\tau$. The parameters $\phi_1, \phi_2$ and $\tau$ are to be determined.

Adopting a Bayesian approach to parameter estimation, let the parameters $\phi_1$ and $\phi_2$ be assigned
a uniform joint prior over the region where the model is causal. To make this more explicit, let \( \lambda_1(\phi_1, \phi_2) \) and \( \lambda_2(\phi_1, \phi_2) \) be the two (not necessarily distinct) complex zeros of the polynomial
\[
\phi(z) = 1 - \phi_1 z - \phi_2 z^2.
\]

Then the prior is proportional to
\[
p_\phi(\phi_1, \phi_2) = \begin{cases} 
1, & \min(|\lambda_1(\phi_1, \phi_2)|, |\lambda_2(\phi_1, \phi_2)|) > 1, \\
0, & \text{otherwise}.
\end{cases}
\]

(Knowledge of the normalizing constant for \( p_\phi(\cdot, \cdot) \) is not necessary.) Let \( \tau \) be assigned the lognormal prior
\[
\log(\tau) \sim N(0, 1000) \quad (9)
\]
and define the parameter vector \( \theta = (\phi_1, \phi_2, \log(\tau)) \). Then, given \( \theta \), it is possible using standard techniques (see, e.g., Brockwell and Davis, 1991, Sections 8.6 and 8.7) to evaluate the likelihood \( p(X|\theta) \) of the data and the model (8). The posterior joint density \( p(\theta|X) \) of the parameters (with respect to Lebesgue measure on \( \mathcal{B}(\mathbb{R}^3) \)) is proportional to
\[
f(\theta) = p_\phi(\phi_1, \phi_2) \varphi(\log(\tau); 0, 1000) p(X|\theta).
\]

We use Algorithm 8.1, along with Algorithm 3.1, to estimate \( \mathbf{E}[\theta|X] \), as well as \( \mathbf{E}[\phi_1^2|X] \) and \( \mathbf{E}[\phi_2^2|X] \), using 50,000 tours. The underlying kernel in the implementation of Algorithm 3.1 is a Metropolis-Hastings kernel with random walk proposals. To be specific, at time \( t \) the proposal \( W \), given the state \( X_t \in \mathbb{R}^3 \), is chosen as
\[
W = X_t + Z,
\]
<table>
<thead>
<tr>
<th>Function</th>
<th>Estimate of Posterior Mean</th>
<th>Std. Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\phi_1$</td>
<td>1.38649</td>
<td>0.000014</td>
</tr>
<tr>
<td>$\phi_2$</td>
<td>1.92445</td>
<td>0.000040</td>
</tr>
<tr>
<td>$\phi_1^2$</td>
<td>-0.69500</td>
<td>0.000014</td>
</tr>
<tr>
<td>$\phi_2^2$</td>
<td>0.48513</td>
<td>0.000020</td>
</tr>
<tr>
<td>$\log(\tau)$</td>
<td>-10.6864</td>
<td>0.000015</td>
</tr>
</tbody>
</table>

Table 2: AR(2) parameter estimates for the sunspots data.

where \(Z\) was multivariate normal with mean \((0, 0, 0)^T\) and diagonal covariance matrix

\[
\begin{bmatrix}
0.05^2 & 0 & 0 \\
0 & 0.05^2 & 0 \\
0 & 0 & 0.2^2
\end{bmatrix}.
\]

Using the method described in Section 3.2, again with \(n_x = n_b = 200\), \(p^*\) is chosen to be \(\exp(-1499.85)\) and parameters of the (Gaussian) re-entry distribution are

\[
m_x = \begin{bmatrix} 1.3626 \\ -0.6731 \\ -10.677 \end{bmatrix} \quad \text{and} \quad \Lambda_x = \begin{bmatrix} 0.00284 & -0.00293 & 0.00176 \\ -0.00293 & 0.00395 & -0.00234 \\ 0.00176 & -0.00234 & 0.00741 \end{bmatrix}.
\]

Algorithm 8.1 can again be implemented using the collection of Pentium-based Linux machines described in Section 4.1. We use it to generate a chain consisting of a total of 50,000 tours, with total length \(T_{50,000} = 342, 101, 406\). The estimated coefficient of variation \(c_{50,000}\) is approximately 0.00002, and the execution time of the program is approximately six minutes. Parameter estimates and their estimated standard errors are given in Table 2.

### 4.3 Dugongs

Next we consider a data set which is used in Ratkowsky (1983), and has also been considered in Carlin and Gelfand (1991). Length (\(Y\)) and age (\(X\)) measurements were made of 27 specimens of a particular species of sea cows (dugongs), captured near Townsville, Queensland. The data are shown in Table 3.

As discussed in Ratkowsky (1983), a frequently-used model for such a data set is

\[
Y_i \sim N(\mu_i, \tau^{-1}) \quad \text{(10)}
\]

\[
\mu_i = \alpha - \beta X_i \quad \text{(11)}
\]

for the data, where \(X_i\) and \(Y_i\) are the age and length of the \(i\)th dugong, respectively, and \(\alpha > 0, \beta > 0, \gamma \in (0, 1), \) and \(\tau > 0\) are unknown model parameters. We assign the (relatively uninformative) priors

\[
\alpha \sim N(0, 10000)
\]
\[ \begin{array}{cccccccc}
X & 1.0 & 1.5 & 1.5 & 1.5 & 2.5 & 4.0 & 5.0 & 5.0 & 7.0 \\
Y & 1.80 & 1.85 & 1.87 & 1.77 & 2.02 & 2.27 & 2.15 & 2.26 & 2.35 \\
\hline
X & 8.0 & 8.5 & 9.0 & 9.5 & 10.0 & 12.0 & 12.0 & 13.0 \\
Y & 2.47 & 2.19 & 2.26 & 2.40 & 2.39 & 2.41 & 2.50 & 2.32 & 2.43 \\
\hline
X & 13.0 & 14.5 & 15.5 & 15.5 & 16.5 & 17.0 & 22.5 & 29.0 & 31.5 \\
Y & 2.47 & 2.56 & 2.65 & 2.47 & 2.64 & 2.56 & 2.70 & 2.72 & 2.57 \\
\end{array} \]

Table 3: The dugong data set used in Section 4.3.

\[ \begin{align*}
\beta & \sim \text{N}(0, 10000) \\
\gamma & \sim \text{U}(0, 1) \\
\tau & \sim \text{Ga}(0.001, 0.001).
\end{align*} \]

Our object is to determine the posterior mean of \( \log(\alpha), \log(\beta), \log(\gamma/(1 - \gamma)) \), and \( \tau^{-1} \).

A standard non-regenerative chain can be constructed by computing full-conditional distributions and updating each of the four parameters in turn. The parameters \( \alpha, \beta \) and \( \tau \) have conjugate priors for the likelihood (11). Hence they can be updated by sampling directly from their respective full-conditional distributions. The parameter \( \gamma \) does not have a conjugate prior. However, it can be updated by using a single Metropolis-Hastings step whose target density is proportional to the full-conditional density of \( \gamma \). As our proposal distribution for the \( \gamma \)-update, we choose a uniform distribution on the interval \([0, 1]\). Incorporating this cycle of updates in Step 2 of Algorithm 3.1, we are able to construct a chain \( \{Y_t\} \) with limiting distribution \( \pi_p^* \). Using the method described in Section 3.2 again, we obtain \( p^* = \exp(11.266) \), and re-entry distribution parameters

\[ m_x = \begin{bmatrix} 2.5919 \\ 0.96505 \\ 0.83374 \\ 119.54 \end{bmatrix} \quad \text{and} \quad \Lambda_x = \begin{bmatrix} 0.00113 & 0.00156 & 0.00025 & 0.10406 \\ 0.00156 & 0.00512 & 0.00014 & 0.34123 \\ 0.00025 & 0.00014 & 0.00015 & 0.02854 \\ 0.10406 & 0.34123 & 0.02854 & 1193.3 \end{bmatrix} \].

It is now a straightforward matter to implement Algorithm 8.1. After \( n = 50,000 \) tours (which takes about 14 minutes of execution time), we obtain a chain of length \( T_n = 518, 166, 488 \), coefficient of variation \( \hat{c}_n = 0.00022 \), and the results shown in Table 4. For the sake of comparison, Table 4 also gives posterior modes estimated by Carlin and Gelfand (1991) (using a prior proportional to \( \alpha^{-1}\sqrt{\tau} \)) as well as functions of least-squares estimates computed by Ratkowsky (1983).

5 Concluding Remarks

Regeneration in Markov chains is useful since it can be used to avoid the burn-in problem, to obtain estimates of the variance of MCMC estimators, and to effectively use parallel processors.
<table>
<thead>
<tr>
<th>Function</th>
<th>Estimate of Posterior Mean</th>
<th>Std. Error</th>
<th>Carlin &amp; Gelfand Mode Estimate</th>
<th>Ratkowsky L.S. Estimate</th>
</tr>
</thead>
<tbody>
<tr>
<td>log(α)</td>
<td>0.979085</td>
<td>0.000028</td>
<td>0.975</td>
<td>0.981</td>
</tr>
<tr>
<td>log(β)</td>
<td>-0.022052</td>
<td>0.000021</td>
<td>-0.014</td>
<td>-0.028</td>
</tr>
<tr>
<td>logit(γ)</td>
<td>1.891709</td>
<td>0.000279</td>
<td>1.902</td>
<td>1.932</td>
</tr>
<tr>
<td>1/τ</td>
<td>0.009033</td>
<td>0.000001</td>
<td></td>
<td>0.0077</td>
</tr>
</tbody>
</table>

Table 4: Posterior means and estimated standard errors using Algorithm 8.1, as well as estimated posterior modes from Carlin and Gelfand (1991) and functions of least squares estimates of α, β, γ, 1/τ from Ratkowsky (1983), for the dugong data set.

to construct a single long chain with a specified target distribution. This paper introduces a simple and practical method for constructing a regenerative Markov chain with a specified target distribution and identifying its regeneration points. Furthermore, in light of recent work by Møller and Nicholls (2001), the construction we use is well-suited for perfect-sampling. This suggests that it may be possible to adapt our method to obtain not only variance estimates and a convenient framework for parallel processing MCMC simulation, but also to obtain unbiased parameter estimates.

6 Acknowledgements

The authors are grateful to Chris Genovese, Mark Schervish, Geoff Nicholls, Antonietta Mira, Peter Müller, and Mike Steele for their careful reading of the paper and their comments and suggestions. They are also grateful to Howard Seltman for his useful suggestions with regard to various aspects of implementation of the algorithms in this paper. This research was supported in part by the National Science Foundation under grants DMS 9819950 and DMS 9801401.

7 Appendix A

In this appendix, we present and prove a formal version of the result stated in Section 2. We also prove that Algorithm 3.1 constructs a Markov chain with limiting distribution πp.

We begin by introducing some notation and definitions.

IA(·) will denote the indicator function

\[ I_A(x) = \begin{cases} 
  1 & x \in A, \\
  0 & \text{otherwise}.
\end{cases} \]

We consider discrete-time Markov chains \( \{X_t\} \) with the time index \( t \) assumed to be the set
The chains take values either in $\mathbb{R}^d$ or in an augmented state-space
\[ E = \mathbb{R}^d \cup \{\alpha\}, \]
where $d$ is some positive integer and $\alpha$ is the “artificial atom”. We sometimes abuse notation and refer to the set $\{\alpha\}$ simply as $\alpha$. Let $\mathcal{B}(\mathbb{R}^d)$ denote the family of Borel sets in $\mathbb{R}^d$, and let $\mathcal{B}(E)$ denote the $\sigma$-field generated by all rectangles in $\mathbb{R}^d$ as well as the point $\alpha$ (thus $\mathcal{B}(E)$ is an extension of $\mathcal{B}(\mathbb{R}^d)$ to the space $E$). When the particular choice of the space ($\mathbb{R}^d$ or $E$) and corresponding $\sigma$-field ($\mathcal{B}(\mathbb{R}^d)$ or $\mathcal{B}(E)$) is context-dependent, we will refer to them simply as $X$ and $\mathcal{B}$, respectively. (In principle, the results in this paper also apply when $\mathbb{R}^d$ is replaced by a more general space. However, for the sake of simplicity, and since most chains of interest take values in a space which can be mapped to $\mathbb{R}^d$, we do not present explicit versions of our results which apply for more general spaces.)

The transition probability kernel of a Markov chain $\{X_t, t = 0, 1, 2, \ldots\}$ taking values in $X$ is a function
\[ P(x, A) = \Pr[X_{t+1} \in A | X_t = x], \quad x \in X, A \in \mathcal{B}. \]
The $k$-step transition probability kernels are defined by $P^k(x, A) = \Pr[X_{t+k} \in A | X_t = x]$. The notation $P_x(A)$ is used to denote the conditional probability of the event $A$, given the initial condition $X_0 = x$ for the chain $\{X_t\}$ with transition probability kernel $P(\cdot, \cdot)$.

The chain $\{X_t\}$ is said to be irreducible (or $\phi$-irreducible) if there exists a measure $\phi$ on $\mathcal{B}$ with $\phi(X) > 0$ such that
\[ \phi(A) > 0 \Rightarrow \sum_{k=1}^{\infty} P(x, A) > 0 \text{ for all } x \in X. \tag{12} \]
A $\phi$-irreducible chain $\{X_t\}$ is said to have a $v$-cycle if there exist sets $D_1, \ldots, D_v \in \mathcal{B}$ such that
\[ P(x, D_{i+1}) = 1, \quad \forall x \in D_i, \quad i = 1, 2, \ldots, v-1, \]
\[ P(x, D_1) = 1, \quad x \in D_v, \]
and $\phi(\bigcup_{i=1}^{v} D_i^C) = 0$. If the largest $v$ for which a $v$-cycle occurs is equal to one, then the chain is said to be aperiodic. The first return time for a set $A \in \mathcal{B}$ is
\[ \tau_A = \min\{t \geq 1 : X_t \in A\} \tag{13} \]
and the occupation time (after time zero) is defined to be
\[ \eta_A = \sum_{t=1}^{\infty} I_A(X_t). \]
A set $A \in \mathcal{B}$ is said to be Harris recurrent if
\[ P_x(\eta_A = \infty) = 1 \text{ for all } x \in A, \]
and the Markov chain $\{X_t\}$ is said to be Harris recurrent if it is $\phi$-irreducible and every set $A$ with $\phi(A) > 0$ is Harris recurrent. A set $B \in \mathcal{B}$ is said to be uniformly accessible from $A \in \mathcal{B}$ if there exists $\delta > 0$ such that
\[ \inf_{x \in A} P_x(\tau_B < \infty) \geq \delta. \]
The chain \( \{X_t\} \) is said to be \textit{ergodic} if there exists a probability measure \( \pi \) on \( \mathcal{B} \) such that
\[
\lim_{n \to \infty} \|P^n(x, \cdot) - \pi(\cdot)\| = 0 \quad \text{for all } x \in X,
\] (14)
where \( \| \cdot \| \) is the total variation norm defined for signed measures on \( \mathcal{B} \) by
\[
\|\mu\| = \sup_{A \in \mathcal{B}} \mu(A) - \inf_{A \in \mathcal{B}} \mu(A).
\]
If this is the case, then the probability measure \( \pi \) is called the \textit{limiting distribution} of the chain.

### 7.1 Incorporating the Artificial Atom

Recall that the space \( E \) includes the artificial atom \( \alpha \) as well as every element of \( \mathbb{R}^d \). As introduced in (1), \( \pi_p^\alpha \) denotes the probability measure
\[
\pi_p^\alpha(A) = (1 - p)\pi(A \setminus \alpha) + pI_A(\alpha), \quad A \in \mathcal{B}(E),
\] (15)
where \( p \) is some constant in the interval \((0,1)\). Let \( \{Y_t, \ t = 0, 1, 2, \ldots\} \) be an ergodic Markov chain taking values in \( E \) with initial value \( Y_0 = \alpha \), whose limiting distribution is \( \pi_p^\alpha \). Let the hitting times \( \tau_y(j) \) and the tours \( Y^j \) be as defined by (2) and (3), respectively. The tour lengths are given by
\[
M_j = \tau_y(j) - \tau_y(j - 1), \quad j = 1, 2, \ldots,
\] (16)
and the elements of a tour \( Y^j \) are referred to as
\[
Y^j_k = Y_{\tau_y[j-1]+k}, \quad k = 1, 2, \ldots, M_j.
\]

The following result shows that \( \alpha \) is a proper Harris recurrent atom for the chain. Hence the chain \( \{Y_t\} \) is regenerative with regeneration points \( \{\tau_y(j), \ j = 0, 1, 2, \ldots\} \).

**Theorem 7.1** Let \( \{Y_t, \ t = 0, 1, 2, \ldots\} \) be an ergodic Markov chain taking values in \( E \) with initial condition \( Y_0 = \alpha \) and limiting distribution \( \pi_p^\alpha \) given by (15). Let the tours and their lengths be as defined in (3) and (16). Then

1. \( \alpha \) is a Harris recurrent state for the chain, and with probability one, the tour lengths \( \{M_j, j = 1, 2, \ldots\} \) are all finite, and
2. \( \mathbf{E}[M_j] = p^{-1} \).

**Proof of Theorem 7.1:** To prove the first part, note (c.f. (13)) that \( P_x(\tau_\alpha < \infty) \geq P^n(x, \alpha) \) for every \( n \). Also, \( P^n(x, \alpha) \to \pi_p^\alpha(\alpha) = p \) as \( n \to \infty \), regardless of \( x \). Thus \( \{\alpha\} \) is uniformly accessible from the state-space \( E \). It then follows from Theorem 9.1.3(ii) of Meyn and Tweedie (1993) that
\[
P_x(\tau_\alpha = \infty) = 1
\]
for all \( x \in E \), and hence that \( \alpha \) is Harris recurrent and with probability one, the tour lengths are all finite. The second part of the result is a direct consequence of Kac’s Theorem (see Kac, 1947; Meyn and Tweedie, 1993, Theorem 10.2.2). \( \square \)
7.2 Removing the Artificial Atom

Next let \{Z_t\} be the sequence derived from \{Y_t\} as described in Section 2. Formally, its construction can be described as follows. Let \(s_1 = \min\{i \geq 1 : M_i > 1\}\), and define

\[s_j = \min\{j > s_{j-1} : M_j > 1\},\]

(Thus \(\{s_j\}\) is the set of indices of tours of \{Y_t\} whose length is strictly greater than one.) Then the tours \(Z^j\) can be written as

\[Z^j = \{Y^s_k,\ k = 1, \ldots, M_{s_j} - 1\},\]

and their corresponding lengths are

\[N_j = M_{s_j} - 1,\]

for \(j = 1, 2, \ldots\). The cumulative tour lengths are then

\[T_j = \sum_{i=1}^{j} N_j,\ j = 1, 2, \ldots,\]

with \(T_0 = 0\). Finally, \(\{Z_t\}\) is the sequence obtained by concatenating the tours \(Z^1, Z^2, \ldots\) together in sequence. (It is not difficult to verify that the sequence \(\{Z_t\}\) is exactly the sequence \(\{Y_t\}\) with every occurrence of the state \(\alpha\) removed.)

Our main result is as follows.

**Theorem 7.2** Suppose that \(\{Y_t,\ t = 0, 1, 2, \ldots\}\) is an ergodic Markov chain taking values in \(E\) with limiting distribution \(\pi^*_p\) given by (15). Let the process \(\{Z_t\}\) be constructed from \(\{Y_t\}\) in the manner described above. Then \(\{Z_t\}\) is an ergodic Markov chain taking values in \(\mathbb{R}^d\) with limiting distribution \(\pi\). Furthermore, the times \(T_j,\ j = 0, 1, 2, \ldots\) are regeneration times for the chain. In other words, the Markov chains \(\{Z_{t-T_i},\ t = T_i, T_i + 1, \ldots\}\), \(i \geq 0\) are identically distributed, and given any \(T_i\) and non-negative integers \(s\) and \(t\), \(Z_s\) and \(Z_t\) are independent if \(s < T_i < t\).

**Proof of Theorem 7.2:** The proof consists of three main parts. First it is shown that \(\{Z_t\}\) is a Markov chain, and its transition probability kernel is derived. Next it is shown that it has invariant distribution \(\pi\), and finally it is shown that the chain is ergodic, and that the times \(T_j\) are regeneration points.

To see that \(\{Z_t\}\) is a Markov chain, note that \(Z_t = Y_\tau\), where \(\tau\) is the stopping time defined to be the \(t\)-th time that \(\{Y_k\}\) is not equal to \(\alpha\). Also define the function \(\gamma_A(\cdot, \cdot, \cdot, \cdot)\) to be equal to one if its second non-\(\alpha\) argument is in the set \(A\), and zero otherwise. Then

\[\gamma_A(Y_\tau, Y_{\tau+1}, Y_{\tau+2}, \ldots) = I_A(Z_{t+1}),\ A \in \mathcal{B}.\]

By the strong Markov property (which every discrete-time Markov chain satisfies),

\[\mathbb{E}[\gamma_A(Y_\tau, Y_{\tau+1}, \ldots)|Y_\tau, Y_{\tau-1}, \ldots, Y_0] = \mathbb{E}_{Y_\tau}[\gamma_A(Y_0, Y_1, \ldots)].\]
The term on the right of (19) only depends on $Y_r = Z_t$. The term on the left is equal to
$\Pr[Z_{t+1} \in A|Z_t, Y_{r-1}, \ldots, Y_0]$. Thus $Z_{t+1}|Z_t$ is independent of $Y_{r-1}, Y_{r-2}, \ldots, Y_0$, and since
$$
\{Z_{t-1}, Z_{t-2}, \ldots, Z_0\} \subseteq \{Y_{r-1}, Y_{r-2}, \ldots, Y_0\},
$$
it follows that
$$
\Pr[Z_{t+1} \in A|Z_t, Z_{t-1}, \ldots, Z_0] = \Pr[Z_{t+1} \in A|Z_t],
$$
which establishes that $\{Z_t\}$ is a Markov chain.

Now let $P(x, A)$ denote the transition probability kernel of $\{Y_t\}$ and define
$$
\pi'(A) = \pi(A \setminus \alpha), \quad A \in \mathcal{B}(E). \quad (20)
$$
Let $A$ be some set in $\mathcal{B}(E)$, and let $A' = A \setminus \{\alpha\}$. Consider the probability $Q(x, A)$ that
$Z_{t+1} \in A$ given that $Z_t = x$. By construction, $Z_{t+1}$ can never be equal to $\alpha$, so $Q(x, \alpha) = 0$ and
$Q(x, A') = Q(x, A')$. Also, $Z_t$ is equal to $Y_t$ for some $s \geq t$. So $Z_{t+1}$ will be in $A'$ if and only if
$Y_{s+1} \in A'$, or $Y_{s+1} = \alpha$ and $Y_{s+2} \in A'$, or $Y_{s+1} = Y_{s+2} = \alpha$ and $Y_{s+3} \in A'$, ... Since these
events are mutually exclusive, the transition probability kernel for $\{Z_t\}$ is
$$
Q(x, A) = Q(x, A') = \Pr[Z_{t+1} \in A'|Z_t = x] = P(x, A') + P(x, \alpha) \sum_{j=0}^{\infty} \frac{P(\alpha, \alpha)^j}{1 - P(\alpha, \alpha)} P(\alpha, A')
$$
$$
= P(x, A') + \frac{P(x, \alpha)P(\alpha, A')}{1 - P(\alpha, \alpha)}. \quad (21)
$$
Now (c.f. (15) and (20))
$$
d\pi'(x) = (1 - p)^{-1}[d\pi_p^*(x) - pI_\alpha(x)].
$$
Hence, using (21),
$$
\int_E Q(x, A)d\pi'(x) = \int Q(x, A')d\pi'(x) = \int P(x, A')d\pi'(x) + \int \frac{P(x, \alpha)P(\alpha, A')}{1 - P(\alpha, \alpha)} d\pi'(x)
$$
$$
= (1 - p)^{-1} \int P(x, A')d\pi_p^*(x) - p(1 - p)^{-1}P(\alpha, A')
$$
$$
+ \frac{P(\alpha, A')}{1 - P(\alpha, \alpha)} [(1 - p)^{-1} \int P(x, \alpha)d\pi_p^*(x) - p(1 - p)^{-1}P(\alpha, \alpha)]. \quad (22)
$$
Since $\pi_p^*$ is the invariant distribution of the chain with kernel $P(\cdot, \cdot)$, we know that
$$
\int P(x, A)d\pi_p^*(x) = \pi_p^*(A)
$$
for $A \in \mathcal{B}(E)$. Making use of this result, along with the fact that $\pi_p^*(\alpha) = p$, equation (22) becomes
$$
\int_E Q(x, A)d\pi'(x) = (1 - p)^{-1} \pi_p^*(A') - p(1 - p)^{-1}P(\alpha, A')
$$
$$
+ \frac{P(\alpha, A')}{1 - P(\alpha, \alpha)} [(1 - p)^{-1} \pi_p^*(\alpha) - p(1 - p)^{-1}P(\alpha, \alpha)]
$$
$$
= (1 - p)^{-1} \pi_p^*(A') = (1 - p)^{-1}(1 - p)\pi'(A) = \pi'(A),
$$
which applies for all \( A \in \mathcal{B}(E) \). This means that \( \pi' \) is the invariant distribution of the chain \( \{Z_t\} \). Since \( \{Z_t\} \) can never hit the point \( \alpha \), it can also be regarded as a chain taking values in \( \mathbb{R}^d \) with invariant distribution \( \pi \).

Next it is necessary to show that \( \{Z_t\} \) is ergodic. First, note that from (21),

\[
Q(x, A) \geq P(x, A) \text{ for any } A \in \mathcal{B}(\mathbb{R}^d) \tag{23}
\]

(because \( A \in \mathcal{B}(\mathbb{R}^d) \) implies that \( A = A' \)). Since \( \pi(A) > 0 \Rightarrow \pi_p^*(A) > 0 \), and \( \pi_p^*(A) > 0 \Rightarrow \sum_{k=1}^\infty P(x, A) > 0 \), it follows that \( \pi(A) > 0 \Rightarrow \sum_{k=1}^\infty Q(x, A) > 0 \) for all \( A \in \mathcal{B}(\mathbb{R}^d) \) and all \( x \in \mathbb{R}^d \). Thus \( \{Z_t\} \) is \( \pi \)-irreducible. Aperiodicity of \( \{Z_t\} \) also follows from (23), as does the property that every Harris recurrent set \( A \) for \( \{Y_t\} \) with \( \alpha \notin A \) must be a Harris recurrent set for \( \{Z_t\} \). Hence the chain \( \{Z_t\} \) is Harris recurrent. Then since \( \{Z_t\} \) is irreducible, aperiodic and positive Harris, it follows from the Aperiodic Ergodic Theorem (see, e.g., Meyn and Tweedie, 1993, Theorem 13.0.1) that it is ergodic, and hence that the invariant distribution \( \pi \) is also the limiting distribution.

Finally, it follows directly from the strong Markov property that the times \( \{T_i, i = 0, 1, 2, \ldots\} \) are regeneration times for the chain \( \{Z_t\} \). \( \square \)

### 7.3 Analysis of Algorithm 3.1

Next we establish that Algorithm 3.1 does indeed generate a Markov chain with limiting distribution \( \pi_p^* \).

**Theorem 7.3** Suppose that \( \pi \) has density \( \beta f(\cdot) \) with respect to some measure \( \nu \), where \( \beta \) is an unknown constant, and let \( \phi \) denote a re-entry density with respect to \( \nu \), satisfying the property that

\[
\phi(x) > 0 \iff f(x) > 0, \quad x \in \mathbb{R}^d.
\]

Let \( P(\cdot, \cdot) \) be the kernel of an ergodic Markov chain with limiting distribution \( \pi \), and let \( p^* \) be some arbitrary positive constant. Then Algorithm 3.1 generates an ergodic Markov chain with limiting distribution \( \pi_p^* \), where \( p = \beta p^*(1 + \beta p^*)^{-1} \).

**Proof:** Let \( P_1(\cdot, \cdot) \) represent the extension of \( P(\cdot, \cdot) \) given by

\[
P_1(x, A) = \begin{cases} P(x, A \setminus \alpha), & x \in \mathbb{R}^d \\ I_A(\alpha), & x = \alpha, \end{cases}
\]

for \( x \in E \) and \( A \in \mathcal{B}(E) \). Let \( P_2(\cdot, \cdot) \) be the kernel

\[
P_2(x, A) = \begin{cases} I_A(\alpha) \int_{\mathbb{R}^d} \left[ 1 - \alpha(x, y) \right] \phi(dy) + \int_{A} \alpha(x, y) \phi(dy), & x = \alpha, \\ I_A(x) \left[ 1 - \alpha(x, \alpha) \right] + I_A(\alpha) \alpha(x, \alpha), & x \in \mathbb{R}^d, \end{cases}
\]

where \( \alpha(\cdot, \cdot) \) is the function defined in Step 4 of the algorithm. Thus \( P_1 \) represents the operation carried out in Step 2 of the algorithm, \( P_2 \) represents the operation carried out in Steps 3 and 4 of
the algorithm, and the transition probability kernel for the chain \( \{ Y_t \} \) generated by Algorithm 3.1 can then be written as

\[
Q(x, A) = (P_2 \circ P_1)(x, A) = \int P_1(x, dy)P_2(y, A).
\]

It is not too difficult to show that for every \( q \in [0, 1] \), \( \pi_q^* \) is an invariant distribution for \( P_1(\cdot, \cdot) \). Also, \( \pi_p^* \) is an invariant distribution for \( P_2(\cdot, \cdot) \). It follows directly that \( \pi_p^* \) is also an invariant distribution for \( Q(\cdot, \cdot) \), and hence that the Markov chain \( \{ Y_t \} \) is positive recurrent.

Furthermore, since \( Q(x, \alpha) = \int_E P_1(x, dy)P_2(y, \alpha) \), \( P_2(\alpha, \alpha) > 0 \) for all \( y \in E \), and \( P_1(x, E) = 1 \), by basic properties of integrals,

\[
Q(x, \alpha) > 0 \quad \forall x \in E. \tag{24}
\]

Also, by the Chapman-Kolmogorov equations,

\[
Q^j(x, A) = \int_E Q^{j-1}(x, dy)Q(y, A) \geq Q^{j-1}(x, \alpha)Q(\alpha, A) \tag{25}
\]

for \( j = 2, 3, \ldots \), and since \( P_1(\alpha, \alpha) = I_A(\alpha) \),

\[
Q(\alpha, A) = P_2(\alpha, A).
\]

Since for any \( B \in \mathcal{B}(\mathbb{R}^d) \), \( \phi(B) > 0 \Rightarrow P_2(\alpha, B) > 0 \), and \( \phi \) is equivalent to \( \pi \), it follows that \( \pi(B) > 0 \Rightarrow Q(\alpha, B) > 0 \). In conjunction with the property (24) and the definition (1), it follows that

\[
\pi_p^*(A) > 0 \Rightarrow Q(\alpha, A) > 0 \quad \forall A \in \mathcal{B}(E). \tag{26}
\]

Substituting this result in turn into the inequality (25), with \( j = 2 \), and using (24), we have

\[
\pi_p^*(A) > 0 \Rightarrow Q^2(x, A) > 0 \quad \forall x \in E, A \in \mathcal{B}(E).
\]

An inductive argument using inequality (25) with \( j = 3, 4, \ldots \) shows that, more generally,

\[
\pi_p^*(A) > 0 \Rightarrow Q^j(x, A) > 0 \quad \forall x \in E, A \in \mathcal{B}(E), \quad j = 2, 3, \ldots. \tag{27}
\]

This establishes that the Markov chain \( \{ Y_t \} \) is \( \pi_p^* \)-irreducible and aperiodic. Since it also has invariant distribution \( \pi_p^* \), it is positive, and thus (see, e.g., Meyn and Tweedie, 1993, Theorem 9.0.1) the space \( E \) can be decomposed as \( E = H \cup N \), where \( \pi_p^*(N) = 0 \), \( N \) is transient, and the chain restricted to \( H \) is Harris recurrent. To show that \( \{ Y_t \} \) is Harris recurrent, it suffices to show that \( Q_x(\tau_H < \infty) = 1 \) for all \( x \in N \). Let \( q_n = Q_x(\tau_H > n) \) for \( n = 1, 2, \ldots \), so that

\[
q_1 = \int_H Q(x, dy),
\]

\[
q_2 = \int_H \int_H Q(x, dy)Q(y, dz),
\]

and so on. Similarly, let \( p_n = P_2(\tau_H > n) \) for \( n = 1, 2, \ldots \). Then since \( Q(x, A) \leq P(x, A) \) when \( A \subseteq H \), it follows that \( q_n \leq p_n \) for \( n = 1, 2, \ldots \). Next, since \( P(\cdot, \cdot) \) is assumed to be ergodic with limiting distribution \( \pi \), it is (by definition) Harris recurrent. Furthermore, since \( \pi_p^*(H) = 1 \), \( \pi(H) \) must be positive, so \( \lim_{n \to \infty} p_n = P_2(\tau_H = \infty) = 0 \). Hence \( Q_x(\tau_H = \infty) = \lim_{n \to \infty} q_n \leq \lim_{n \to \infty} p_n = 0 \), so \( \{ Y_t \} \) is Harris recurrent.

Finally, since \( \{ Y_t \} \) is irreducible, aperiodic, and Harris recurrent with invariant distribution \( \pi_p^* \), it is ergodic with \( \pi_p^* \) as the limiting distribution. \( \square \)
Appendix B: A Parallel Processing Version of Algorithm 3.2

Let \( \{Z_t\} \) be the Markov chain constructed as described in Section 2. The tours \( \{Z^j, j = 1, 2, \ldots\} \) of \( \{Z_t\} \) are independent of each other. Thus there is no need for them to be generated on a single processor. Suppose there are \( P \) processors available, and that they can construct realizations of tours independently. Now imagine that the first processor constructs some number \( n_1 \) of tours, the second processor constructs \( n_2 \) tours, and so on. If the tours generated on the processors are collected together and concatenated (in some order which is chosen independently of the tours themselves), then the resulting sequence is a realization of a single Markov chain with limiting distribution \( \pi \), with a total of \( \sum_{j=1}^{P} n_j \) tours.

In light of these observations, we can state the following parallel processing version of Algorithm 3.2.

**Algorithm 8.1: Master Process**

Step 1. Send requests for pairs \((h_1, N_1), \ldots, (h_{P-1}, N_{P-1})\) to slave processes 2 to \( P \), respectively. Set \( k = P \). Set \( n = 0 \). Mark all slave processes as “busy”.

Step 2. Wait for any slave to return its pair. Store the pair. Replace \( n \) by \( n + 1 \) and mark the slave as “free”.

Step 3. If \( n = M \), then go to Step 6.

Step 4. If \( k \leq M \), send a new request to the freed slave to return the pair \((h_k, N_k)\), mark the slave as busy again, and replace \( k \) by \( k + 1 \).

Step 5. Go back to Step 2.

Step 6. Terminate all the slave processes. Compute \( \hat{h}_M \) and its estimated variance \( \hat{\sigma}^2 / M \) using equations (5) and (6), respectively.

**Algorithm 8.1: Slave Process**

Step 1. Wait for a request from the master process for a pair \((h_j, N_j)\).

Step 2. Generate a portion \( \{Y_t\} \) of a Markov chain with transition kernel \( P(\cdot, \cdot) \), starting with \( Y_0 = \alpha \). Run the chain until it hits \( \alpha \) again and let this (second) hitting time be denoted \( \tau \).

Step 3. If \( \tau = 1 \), then discard the values \( \{Y_0, Y_1\} \) and go back to Step 2.

Step 4. Compute \( h_j = \sum_{k=1}^{\tau-1} h(Y_k) \) and \( N_j = \tau - 1 \). Return the pair \((h_j, N_j)\) to the master process and go back to Step 1.
Note that it is essential that random number generators on separate processors generate independent sequences of random variables. This is an easy rule to violate, since it is often the case that a process replicated on several machines will use the same pseudo-random number generator with the same seed. A simple (although not ideal) solution is to use a different seed for the generator on each separate processor. A review of more sophisticated techniques for dealing with this problem can be found in Coddington (1996).

References


