Modifying Gibbs Sampling to Avoid Self Transitions

Radford M. Neal

University of Toronto, Dept. of Statistical Sciences

https://glizen.com/radfordneal

radford@utstat.utoronto.ca

26 March 2024

Abstract. Gibbs sampling is a popular Markov chain Monte Carlo method that samples from a distribution on n state variables by repeatedly sampling from the conditional distribution of one variable, x_i, given the other variables, x_{-i}, either choosing i randomly, or updating sequentially using some systematic or random order for i. When x_i is discrete, a Gibbs sampling update may choose a new value that is the same as the old value. A theorem of Peskun indicates that, when i is chosen randomly, a reversible method that reduces the probability of such self transitions, while increasing the probabilities of transitioning to each of the other values, will decrease the asymptotic variance of estimates of expectations of functions of the state. This has inspired two modified Gibbs sampling methods, originally due to Frigessi, Hwang, and Younes and to Liu, though these do not always reduce self transitions to the minimum possible. Methods that do reduce the probability of self transitions to the minimum, but do not satisfy the conditions of Peskun's theorem, have also been devised, by Suwa and Todo, some of which are reversible and some not. I review and relate these past methods, and introduce a broader class of reversible methods, including that of Frigessi, *et al.*, based on what I call "antithetic modification", which also reduce asymptotic variance compared to Gibbs sampling, even when not satisfying the conditions of Peskun's theorem. A modification of one method in this class, which I denote as ZDNAM, reduces self transitions to the minimum possible, while still always reducing asymptotic variance compared to Gibbs sampling. I introduce another new class of non-reversible methods based on slice sampling that can also minimize self transition probabilities. I provide explicit, efficient implementations of all these methods, and compare the performance of Gibbs sampling and these modified Gibbs sampling methods in simulations of a 2D Potts model, a Bayesian mixture model, and a belief network with unobserved variables. The assessments look at both random selection of i, and several sequential update schemes. Sequential updates using methods that minimize self transition probabilities are found to usually be superior, with ZDNAM often performing best. There is evidence that the non-reversibility produced by sequential updating can be beneficial, but no consistent benefit is seen from the individual updates being done by a non-reversible method.

1 Introduction

Gibbs sampling has for some time been widely used to sample from complex probability distributions in statistics and machine learning (Geman and Geman 1984; Ackley, Hinton, and Sejnowski 1985; Gelfand and Smith 1990; Thomas, Spiegelhalter, and Gilks 1992), and has been used in statistical physics (where it is often called "Glauber dynamics" or the "heatbath" method) since long before that (see Landau and Binder (2009) for a review). Gibbs sampling is easy to implement in many contexts, and has no adjustable parameters that need tuning. In some applications, Gibbs sampling is used alone, but it is also often combined with other Markov chain Monte Carlo (MCMC) methods, either by alternating between Gibbs sampling and other updates, or by using Gibbs sampling for a subset of variables for which it is well-suited,

and other update methods for the remaining variables. Gibbs sampling or its modifications can also be a component of more elaborate sampling schemes, such as those aimed at exploitation of parallel computation (Tjelmeland 2004), or avoidance of backtracking (Neal 2004). Improvements to Gibbs sampling that require no additional computational capabilities are therefore of considerable interest.

I will review several ways of modifying Gibbs sampling to reduce the probability that a transition leaves the state the same as before, and introduce two new classes of such methods. Some of these methods can be shown to always produce better estimates that Gibbs sampling when the variable to be updated is chosen randomly, using results discussed in detail in a companion theoretical paper (Neal and Rosenthal 2023). I show empirically that these methods can also improve MCMC estimates in other contexts, such as sequential updating of variables, and that the methods that reduce self transitions to the minimum possible generally perform best.

2 Review of Gibbs Sampling (GS) and its implementation

Gibbs sampling and other Markov chain Monte Carlo methods aim to sample from some probability distribution, $\pi(x)$, on a state space, $\mathcal{X}$, by repeatedly applying updates to the state, each of which leaves π invariant, and eventually converge to π regardless of the initial state. Gibbs sampling is applicable when the state is naturally seen as consisting of n variables, with $\mathcal{X}$ written as $\mathcal{X}_1 \times \cdots \times \mathcal{X}_n$. A single Gibbs sampling update of the state x consists of choosing an index $i \in \{1, \ldots, n\}$ and then replacing x_i with a value drawn from the conditional distribution for x_i given the remaining variables (denoted as x_{-i}).

I will write $P(x \to x')$ for the probability that the Markov chain transitions to state x' when in state x. When only variable i is updated, I will write $P(x_i \to x'_i | x_{-i})$ for the probability of transitioning from the state with x_i to that with x'_i, given that the other variables have values x_{-i} (which remain unchanged). For Gibbs sampling, $P(x_i \to x'_i | x_{-i})$ is simply $\pi(x'_i | x_{-i})$. Note that when $\mathcal{X}$ or $\mathcal{X}_i$ is finite, it will sometimes be convenient to put the transition probabilities in a matrix, P, with $P_{uv} = P(u \to v)$.

A Gibbs sampling update for a single variable, i, is *reversible*, meaning that

$$\pi(x_i|x_{-i})P(x_i \to x'_i|x_{-i}) \;=\; \pi(x'_i|x_{-i})P(x'_i \to x_i|x_{-i}) \tag{1}$$

as is easily seen by substituting $P(x_i \to x'_i|x_{-i}) = \pi(x'_i|x_{-i})$. This reversibility condition is sufficient (but not necessary) for the update to leave the conditional distribution *invariant*:

$$\pi(x'_i|x_{-i}) \;=\; \sum_{x_i \in \mathcal{X}_i} \pi(x_i|x_{-i})P(x_i \to x'_i|x_{-i}) \tag{2}$$

Since a Gibbs sampling update for variable i does not change x_{-i}, invariance of this conditional distribution implies invariance of π as a whole — that is,

$$\pi(x') \;=\; \sum_{x \in \mathcal{X}} \pi(x)P(x \to x') \tag{3}$$

One way to use single-variable Gibbs sampling updates to sample for the full state is to randomly select a variable to update each iteration, from some distribution for i on $\{1, \ldots, n\}$, here assumed uniform. This produces a chain with the following transition probabilities:

$$P(x \to x') \;=\; \frac{1}{n} \sum_{i=1}^{n} I(x'_{-i} = x_{-i})\; \pi(x'_i|x_{-i}) \tag{4}$$

(where $I(\cdot)$ is 1 if the enclosed condition is true, and 0 otherwise). These transitions are easily seen to also be reversible, and hence leave π invariant. However, a disadvantage of this strategy is that some variables might, by chance, not be updated for a considerable time.

A more common strategy is to perform Gibbs sampling updates for variables in sequence, with i going from 1 to n. Since each such update leaves π invariant, the sequence of updates will also leave π invariant, and hence be a suitable Markov chain Monte Carlo method. However, such a sequence of updates is not, in general, reversible. It can be made reversible by, in each such series of updates, going through the variables in an order that is randomly chosen for that series, from a distribution in which any update order and its reverse are equally likely. Note, however, that this reversible version is not necessarily better than a non-reversible sequence of updates.

For Gibbs sampling to be feasible, it must be possible to sample from the conditional distribution $\pi(x_i|x_{-i})$ with a reasonable amount of computation. This is sometimes possible when x_i is continuous, or discrete with an infinite number of possible values, and has a form amenable to sampling. When x_i takes values from a finite set, that is not enormous, Gibbs sampling will be feasible as long as $\pi(x_i|x_{-i})$ can be computed for all $x_i \in \mathcal{X}_i$, after which sampling a particular x_i according to these probabilities can be done in a straightforward way (Devroye 1986, page 85).

In some applications, such as the Potts model, $\pi(x_i|x_{-i})$ depends on x_{-i} only through a function, $e(x_{-i})$, that has a small number of possible values. In this case, tables of conditional probabilities for x_i for all possible values of $e(x_{-i})$ can be pre-computed once, before simulating the Markov chain. Using the "alias method" (Devroye 1986, page 107), tables can then be pre-computed that allow for sampling from each of these conditional distributions in time that is independent of the number of possible values for x_i.

The probabilities used for Gibbs sampling will usually have a floating-point representation, and may have been computed with some round-off error. I will assume that these probabilities are guaranteed to be in the interval $[0, 1]$, and that their sum is very close to one, but not necessarily exactly one. This will be the case when, as often, these probabilities are first computed in unnormalized form (guaranteed to be non-negative, and not all zero), and then are all divided by their (possibly inexact) sum. I assume the methods for sampling discussed above can handle probabilities with these characteristics. The detailed algorithms for modified Gibbs sampling that I present here will in turn produce such transition probabilities.

Intuitively, it seems that Gibbs sampling can be inefficient, since it is quite possible that when updating variable i by sampling from $\pi(x_i|x_{-i})$, the new value, x_i', will be the same as the current value, x_i. Since we need the value of the state to move around in order to explore the distribution, this seems sub-optimal.

Such self transitions are sometimes necessary. If some value for x_i has conditional probability greater than $1/2$, this value must sometimes remain unchanged if its frequency of occurrence is to match its probability. In particular, if some value has probability $p > 1/2$, the probability that a transition leaves this value unchanged must be at least $(2p{-}1) \,/\, p > 0$ for the transitions to leave the distribution invariant.[1]

This paper looks at several methods for modifying Gibbs sampling to reduce the probability of self transitions. Some of these methods are reversible, and never decrease non-self transition probabilities. These methods can be justified as being superior to Gibbs sampling by a theorem of Peskun (1973) showing that for such chains the intuition that self transitions are inefficient is correct. Some other reversible methods reduce self transition probabilities and also reduce some non-self transition probabilities, so Peskun's theorem does not apply, but they can be justified as improvements to Gibbs sampling using

[1]Let u be the value with $\pi(u) = p > 1/2$. Then invariance requires that $p = \sum_v \pi(v) P(v \to u) = p\, P(u \to u) + \sum_{v \neq u} \pi(v) P(v \to u) \leq p\, P(u \to u) + \sum_{v \neq u} \pi(v) = p\, P(u \to u) + 1{-}p$, from which it follows that $P(u \to u) \geq (2p{-}1) \,/\, p$.

other theoretical tools (Neal and Rosenthal 2023). Theoretical analysis of non-reversible methods is more difficult, but as will be seen empirically, some non-reversible methods often perform as well or better than reversible methods. However, the reversible ZDNAM method that I introduce here is a good overall choice, when employed with a non-reversible sequential updating schedule.

3 Asymptotic variance, Peskun-dominance, and efficiency-dominance

One fundamental measure of efficiency of an MCMC method designed to sample from a distribution π is the *asymptotic variance* of an estimate of the expectation with respect to π of some function f, found by averaging over states from the chain:

$$v(f,P) \quad = \quad \lim_{K\to\infty} K \operatorname{Var}\Big(\frac{1}{K}\sum_{t=1}^{K} f(x^{(t)})\Big) \tag{5}$$

Here, $x^{(t)}$ is the state after t transitions of the chain with transitions P, initialized at some state $x^{(0)}$. When some large number, K, of transitions are simulated, we expect the variance of the average, $(1/K)\sum f(x^{(t)})$, which is an estimate for the expectation of f, to be approximately $v(f,P)/K$. (In practice, the early part of a chain is often discarded when estimating expectations, but this refinement does not affect the asymptotic variance, and will be ignored here.)

Lowering asymptotic variance is an important goal in designing a Markov chain sampling method, but other criteria such as speed of convergence to π are also important, and can conflict with minimizing asymptotic variance.[2] One should note in particular that independent sampling from π — that is, using transition probabilities $P(x \to x') = \pi(x')$ — results in immediate convergence, but does not minimize asymptotic variance, since lower variance can be obtained by "antithetic" sampling, in which the transitions induce negative correlations between $f(x^{(t)})$ and $f(x^{(t+\delta)})$ for some lags δ.

It is not feasible to actually minimize asymptotic variance for the problems with an enormous state space for which MCMC is used, just as it is not practical to directly sample from a distribution on such a state space. But we can try to improve the asymptotic variance of less direct methods, such as Gibbs sampling. For this, we need to know that an improvement to a component of the method — such as sampling a new value for a single variable, with other variables left unchanged — will result in an improvement to the overall method. This is a main topic of a companion paper (Neal and Rosenthal 2023).

Previous work in this area has utilized a theorem of Peskun (1973), who showed that if two chains with transitions P and P^* are both reversible with respect to π, and $P^*(x \to x') \;\geq\; P(x \to x')$ for all $x \neq x'$, then $v(f,P^*) \;\leq\; v(f,P)$, for all functions f.

In other words, if a reversible chain is modified to reduce the probability of some self transitions, and hence necessarily increase the probability of some non-self transitions, while not reducing the probability of any other non-self transitions, this will not increase the asymptotic variance of the estimate for the expectation of any function. Typically, such a modification will reduce asymptotic variance (with some exceptions, such as when the function is constant, so the variance of the estimate is always zero).

I will say that P^* *Peksun-dominates* P if $P^*(x \to x') \;\geq\; P(x \to x')$ for all $x \neq x'$, and that P^* *efficiency-dominates* P if the asymptotic variance of the estimate of the expectation of every function is

[2] As an extreme example, it is always easy to define and implement a chain that has zero asymptotic variance for estimating the expectation of any function with respect to the uniform distribution on some easily enumerable set, by simply having the chain cycle deterministically through all elements of this set. But such a chain is of no practical use, since it gives accurate estimates only after having visited every value, which is impractical for any problem where one would consider using MCMC. However, examples of this sort are not possible with reversible chains, which cannot be periodic with period greater than two.

at least as small when using P^* as when using P. Peksun's theorem then says that, for reversible chains, Peksun dominance implies efficiency dominance.

Note that Peskun dominance is only a partial ordering — it is possible for two Markov chains to both be reversible with respect to π but for neither to Peskun-dominate the other, since for each chain there is some non-self transition probability that is larger than that for the other chain. Similarly, efficiency-dominance is a partial order over reversible chains (Neal and Rosenthal 2023, Theorem 10), but not a complete order. Furthermore, when neither of two chains Peskun-dominates (or efficiency-dominates) the other, it may be that no other reversible chain Peskun-dominates (or efficiency-dominates) both of these chains.[3]

Suppose we modify the Gibbs sampling update for variable x_i, for some particular values of the other variables, say when $x_{-i} = \bar{x}_{-i}$, in a way that Peskun-dominates the Gibbs sampling update — that is, the probability of changing x_i to any value other than its current value is greater after the modification than it would be for Gibbs sampling — while also being reversible with respect to the conditional distribution for x_i. It is easy to see that a method that randomly selects a variable to update, and uses this modified method when variable i is selected and $x_{-i} = \bar{x}_{-i}$, will Peskun dominate Gibbs sampling with random selection of the variable to update. The probability of moving from x to x' will be at least as large when variable x_i is updated and $x_{-i} = \bar{x}_{-i}$, and the same otherwise. Accordingly, by Peskun's theorem, the overall method using the modified update will efficiency-dominate Gibbs sampling.

We can go on to modify the updates for variable i with other values for x_{-i}, and to modify updates for variables other than i. If each of these local modifications Peskun-dominates Gibbs sampling, then again the overall method (with random selection of variable to update) will Peskun-dominate, and hence also efficiency-dominate, Gibbs sampling. Peskun dominance thus provides a way of showing that local efficiency improvements, to updates of single variables, lead to improvement in the efficiency of the overall method — at least, when the variable to be updated is chosen randomly.

However, the converse of Peskun's theorem is not true. As I will discuss later, it is possible, with reversible P^* and P, for P^* to efficiency-dominate P even though some non-self transition probabilities are greater for P than for P^*. This raises the question of whether a modification to an update for a single variable that efficiency-dominates Gibbs sampling, but does not Peskun-dominate it, will always result in an efficiency improvement when used in an overall method that randomly selects a variable to update. A companion paper (Neal and Rosenthal 2023) shows that this is true, as will be discussed further later.

Peskun's theorem does not apply if the variables are updated by a sequential scan for $i = 1, \ldots n$. One reason is that this (typically) results in a non-reversible chain (taking a full scan to be one iteration of the chain). If the order for updating variables is randomly selected for each iteration, with any order and its reverse equally probable, the resulting chain will be reversible, but Peskun's theorem will still not apply, since it is possible that a sequence of modified Gibbs sampling updates that individually Peskun-dominate

[3]Consider two chains with transition probabilities shown below, both reversible with respect to the uniform distribution on $\{1, 2, 3, 4\}$:

$$\begin{bmatrix} 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 \end{bmatrix} \qquad \begin{bmatrix} 0 & \frac{1}{2} & 0 & \frac{1}{2} \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 \\ 0 & \frac{1}{2} & 0 & \frac{1}{2} \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 \end{bmatrix}$$

The first chain has zero asymptotic variance when estimating the expectation for the function $I(x \in \{1, 2\})$, but not for the function $I(x \in \{1, 3\})$, while the reverse is true for the second chain. No reversible chain has zero asymptotic variance for both functions (a consequence of the fact that periodic reversible chains must have period two), and hence no reversible chain can Peskun-dominate, or efficiency-dominate, both.

the corresponding unmodified Gibbs sampling update will have lower probability of some transition to a different state, even when this does not happen for any single modified Gibbs sampling update.[4]

It is therefore only when the variable to be updated is selected randomly that Peskun's theorem provides a guarantee that modifying Gibbs sampling to increase the probabilities for all non-self transitions will improve asymptotic variance. When variables are updated in sequence, as is the more common practice, Peskun's theorem provides no guarantee that the modified method will be better. One cannot say in general whether updating variables sequentially is better or worse than updating them at random, as is illustrated by He, *et al.* (2016). However, sequential updating, producing a non-reversible chain, seems to usually work better in practical applications, which reduces the practical relevance of Peskun's theorem.

Nevertheless, Peskun's theorem provides a theoretical motivation for looking at ways of reducing self transitions in Gibbs sampling. I will next describe one method of reducing self transitions for which Peskun's theorem applies, based on a Metropolis-Hastings modification of Gibbs sampling. I will then introduce a more general framework for improving the efficiency of Gibbs sampling, and discuss several reversible methods derived in this way. For some of these, Peskun's theorem does not apply, but they can still be shown to efficiency-dominate Gibbs sampling.

4 The Metropolis-Hastings Gibbs Sampling (MHGS) method

Gibbs sampling can be seen as an instance of the Metropolis-Hastings algorithm (Hastings 1970), in which transition probabilities from a state u are defined in terms of probabilities, $Q(u \to v)$, for proposing to move from u to a state v. After sampling a v according to these probabilities, v is accepted as the new state with probability

$$\min\left(1,\ \frac{\pi(v)\,Q(v \to u)}{\pi(u)\,Q(u \to v)}\right) \tag{6}$$

If v is not accepted, the new state is the same as the old state, u. This transition is reversible with respect to π, and hence leaves π invariant.

A Gibbs sampling update for component i of state x is obtained by proposing a new value for the state according to the conditional distribution for x_i given the other components of the state. That is,

$$Q(x \to x') \quad = \quad I(x'_{-i} = x_{-i})\,\pi(x'_i|x_{-i}) \tag{7}$$

This proposal is always accepted, since for any proposed x',

$$\min\left(1,\ \frac{\pi(x')\,Q(x' \to x)}{\pi(x)\,Q(x \to x')}\right) \ = \ \min\left(1,\ \frac{\pi(x')\,\pi(x_i|x'_{-i})}{\pi(x)\,\pi(x'_i|x_{-i})}\right) \ = \ \min\left(1,\ \frac{\pi(x'_{-i})\,\pi(x'_i|x'_{-i})\,\pi(x_i|x'_{-i})}{\pi_(x_{-i})\,\pi(x_i|x_{-i})\,\pi(x'_i|x_{-i})}\right) \ = \ 1 \tag{8}$$

Liu (1996) introduced a method of modifying such a Gibbs sampling update by always proposing a value for x_i that is different from the current value, with probabilities proportional to the conditional probabilities given x_{-i}. That is, the proposal probabilities are

$$Q(x \to x') \quad = \quad I(x'_{-i} = x_{-i})\,I(x'_i \neq x_i)\ \frac{\pi(x'_i|x_{-i})}{1 - \pi(x_i|x_{-i})} \tag{9}$$

[4]For example, consider when π is uniform over $\mathcal{X} = \{0,1\} \times \{0,1\}$. Gibbs sampling updates for both variables give equal probability to the values 0 and 1, and when applied in either order, the probability of transitioning to any of the four possible values is 1/4. Both Gibbs sampling updates can be modified so that the value is changed with probability 1, and viewed individually, these modifications satisfy the condition for Peskun's theorem. But when applied sequentially, in any order, even one chosen at random, these modified updates have probability 0 of moving from state $(0,0)$ to state $(0,1)$ or to state $(1,0)$, compared to probability 1/4 for the unmodified updates. So Peskun's theorem does not apply.

$$\begin{bmatrix} \frac{1}{10} & \frac{2}{10} & \frac{3}{10} & \frac{4}{10} \\ \frac{1}{10} & \frac{2}{10} & \frac{3}{10} & \frac{4}{10} \\ \frac{1}{10} & \frac{2}{10} & \frac{3}{10} & \frac{4}{10} \\ \frac{1}{10} & \frac{2}{10} & \frac{3}{10} & \frac{4}{10} \end{bmatrix} \rightarrow \begin{bmatrix} 0 & \frac{2}{9} & \frac{3}{9} & \frac{4}{9} \\ \frac{1}{9} & \frac{1}{72} & \frac{3}{8} & \frac{4}{8} \\ \frac{1}{9} & \frac{2}{8} & \frac{34}{504} & \frac{4}{7} \\ \frac{1}{9} & \frac{2}{8} & \frac{3}{7} & \frac{106}{504} \end{bmatrix}$$

Figure 1: An illustration of how MHGS modifies Gibbs sampling transition probabilities. The variable updated in this example has four possible values, whose conditional probabilities given the current values of other variables are 1/10, 2/10, 3/10, and 4/10.

The acceptance probability for such a proposal is

$$\min\left(1,\ \frac{\pi(x')\,\pi(x_i|x'_{-i})\,/\,(1-\pi(x'_i|x'_{-i}))}{\pi(x)\,\pi(x'_i|x_{-i})\,/\,(1-\pi(x_i|x_{-i}))}\right) = \min\left(1,\ \frac{\pi(x'_{-i})\,\pi(x'_i|x'_{-i})\,\pi(x_i|x'_{-i})\,(1-\pi(x_i|x_{-i}))}{\pi_(x_{-i})\,\pi(x_i|x_{-i})\,\pi(x'_i|x_{-i})\,(1-\pi(x'_i|x'_{-i}))}\right) \quad (10)$$

$$= \min\left(1,\ \frac{1-\pi(x_i|x_{-i})}{1-\pi(x'_i|x_{-i})}\right) \quad (11)$$

Note that this is 1 whenever $\pi(x'_i|x_{-i}) \geq \pi(x_i|x_{-i})$.

These proposal and acceptance probabilities give the following modified non-self transition probabilities:

$$\text{when } x' \neq x,\quad P^*(x \rightarrow x') = I(x'_{-i} = x_{-i})\,\frac{\pi(x'_i|x_{-i})}{1-\pi(x_i|x_{-i})}\,\min\left(1,\ \frac{1-\pi(x_i|x_{-i})}{1-\pi(x'_i|x_{-i})}\right) \quad (12)$$

$$= I(x'_{-i} = x_{-i})\,\min\left(\frac{\pi(x'_i|x_{-i})}{1-\pi(x_i|x_{-i})},\ \frac{\pi(x'_i|x_{-i})}{1-\pi(x'_i|x_{-i})}\right) \quad (13)$$

The modified self transition probability, $P^*(x \rightarrow x)$, can be found as one minus the sum of non-self transition probabilities from x. The self transition probability also equals the total probability of proposing a value that is not accepted:

$$P^*(x \rightarrow x) = \sum_{x'_i} I\Big(\pi(x'_i|x_{-i}) < \pi(x_i|x_{-i}\Big)\,\frac{\pi(x'_i|x_{-i})}{1-\pi(x_i|x_{-i})}\,\left(1-\frac{1-\pi(x_i|x_{-i})}{1-\pi(x'_i|x_{-i})}\right) \quad (14)$$

It follows that the self transition probability is zero when x_i has minimal conditional probability, since all proposals from such a value are accepted. So at least one modified self transition probability is zero.

I will refer to this method as Metropolis-Hastings Gibbs Sampling (MHGS). Figure 1 shows an example of how MHGS modifies Gibbs sampling transition probabilities.

As Liu notes, the non-self transition probabilities of equation (13) are clearly greater than those for Gibbs sampling, which are $\pi(x'_i|x_{-i})$, so Peskun's theorem guarantees that the asymptotic variance of estimates found using MHGS will be lower than when using GS, when i is selected randomly.

Liu's short paper does not discuss how to implement this method, but there are two obvious ways.

First, transitions can be simulated by computing all non-self transition probabilities from the current value of the state using equation (13), then finding the self transition probability as one minus the sum of these.[5] A new value can then be sampled according to these probabilities, as discussed earlier for Gibbs sampling. This takes expected time asymptotically proportional to m, the number of possible values for x_i,

[5]Using equation (14) is not recommended, as it may have high relative error when $1-\pi(x_i|x_{-i})$ is close to zero.

Input: Gibbs sampling probabilities, $\pi(i)$, for $i = 1, \ldots, m$
The current state value, k, in $\{1, \ldots, m\}$

Output: MHGS transition probabilities, $p(i)$, for $i = 1, \ldots, m$

If $1 - \pi(i) \leq 0$ for any i:

Avoid division by zero by reverting to Gibbs sampling when a probability is 1 (or perhaps very close to 1)

For $i = 1, \ldots, m$: Set $p(i)$ to $\pi(i)$

Else:

Find non-self transition probabilities, and their sum

Set s to 0

For $i = 1, \ldots, m$:

If $i \neq k$:

Set $p(i)$ to $\min\left(1,\ \pi(i) \,/\, (1 - \pi(k)),\ \pi(i) \,/\, (1 - \pi(i))\right)$

The min with 1 above guards against error from rounding

Add $p(i)$ to s

Set the self transition probability, guarding against round-off error producing a negative probability (or change $1-s < 0$ to $1-s < \epsilon$ for some small ϵ to avoid producing tiny probabilities that should be exactly zero)

If $1-s < 0$:

Set $p(k)$ to 0

Else:

Set $p(k)$ to $1-s$

Algorithm 1: Computing MHGS transition probabilities, based on equation (13). Here, the algorithm is phrased in terms of a distribution, π, for a single variable, but in practice, it will be applied to the conditional distribution for one variable given values of the others, as would be sampled for unmodified Gibbs sampling.

if probabilities need to be computed for each update, or takes constant time if probabilities for all possible conditional distributions can be pre-computed, and the alias method used. Algorithm 1 shows in detail how the needed transitions probabilities can be computed, including some precautions for avoiding numerical issues.

Alternatively, a transition can be simulated by first sampling a proposal from the distribution defined by equation (9), and then accepting this proposal with the probability given by equation (11), or instead rejecting it and retaining the current value.[6] When Gibbs sampling probabilities have no useful structure, this procedure also takes time asymptotically proportional to m, since that much time is needed for the computation of $m-1$ proposal probabilities and their use in sampling of a proposal.

Sometimes, however, the conditional probabilities given x_{-i} have a form that allows for fast sampling, which can be modified to sample a proposal according to equation (9). One possibility is when sampling can be done by inverting the cumulative distribution function. For example, suppose the Gibbs sampling

[6]Note that rather than use equation (9) as written, it may be better to replace the expression $\pi(x_i'|x_{-i}) \,/\, (1 - \pi(x_i|x_{-i}))$ with $\pi(x_i'|x_{-i}) \,/\, \sum_{x_i' \neq x_i} \pi(x_i'|x_{-i})$, in order to mitigate effects of round-off error when $\pi(x_i|x_{-i})$ is close to 1.

probabilities are geometric(θ) on $\{1, \ldots, m\}$, with cumulative distribution function

$$F(a) \;=\; P(x_i' \le a) \;=\; \frac{1-(1-\theta)^a}{1-(1-\theta)^m} \tag{15}$$

Inverting the continuous form of this cumulative distribution function, in which a can be any non-negative real, allows sampling from this distribution in constant time, independent of m (Devroye 1986, page 87). For this example,

$$F^{-1}(u) \;=\; \frac{\log(1-u(1-(1-\theta)^m))}{\log(1-\theta)} \tag{16}$$

and we can generate a value as $\lceil F^{-1}(U) \rceil$, where U is drawn from the uniform distribution on $(0, 1)$.

This efficient simulation method can be adapted to MHGS. Given a current value of x_i, the cumulative distribution function of a proposal x_i' from equation (9) will be

$$F_{prop}(a) \;=\; \begin{cases} \dfrac{F(a)}{1-F(x_i)+F(x_i-1)} & \text{if } a < x_i-1 \\[2ex] \dfrac{F(x_i-1)}{1-F(x_i)+F(x_i-1)} & \text{if } x_i-1 \le a < x_i \\[2ex] \dfrac{F(a)-F(x_i)+F(x_i-1)}{1-F(x_i)+F(x_i-1)} & \text{if } x_i \le a \end{cases} \tag{17}$$

The corresponding inverse cumulative distribution function is

$$F^{-1}_{prop}(u) \;=\; \begin{cases} F^{-1}(u\,(1-F(x_i)+F(x_i-1)) & \text{if this is less than } x_i-1 \\ F^{-1}(F(x_i)-F(x_i-1)+u\,(1-F(x_i)+F(x_i-1))) & \text{otherwise} \end{cases} \tag{18}$$

We can use this to generate a proposal as $\lceil F^{-1}_{prop}(U) \rceil$, where U is uniform on $(0, 1)$, and then accept or reject it according to equation (11).

Note that with this technique there is no problem with letting m go to infinity, to obtain a method that works for a variable with a distribution on the positive integers.

More generally, if any method for efficient Gibbs sampling is available (including for a variable with a countably infinite number of possible values), it can be adapted to sample from the proposal distribution of equation (9) by sampling repeatedly until a value for x_i' different from x_i is obtained, which can then be accepted or rejected according to equation (11). However, if the current value, x_i, is such that $\pi(x_i|x_{-i})$ is close to one, a great many repetitions might be required before a different value is obtained. This inefficiency can be avoided by reverting to doing a standard Gibbs sampling update if the maximum value of $\pi(x_i|x_{-i})$ for all possible x_i is close to one, which preserves reversibility since this criterion does not depend on the current value. (Of course, this will slightly increase the probability of a self transition.)

If the value with maximum probability is not easily identifiable, one can use the following approach: First sample a value as for Gibbs sampling, then test whether the probability of this value is in $[\epsilon,\, 1-\epsilon]$, for some ϵ close to zero. If so, repeatedly sample (discarding the value just tested) until a value different from the current value is found, knowing that there is no problematic value with probability greater than $1-\epsilon$. If the probability of the value tested is outside $[\epsilon,\, 1-\epsilon]$, instead revert to Gibbs sampling (again, discarding the value used for the test). The probability of reverting to Gibbs sampling if no value has probability greater than $1-\epsilon$ will be less than $m\epsilon$.

5 Efficiency improvement by Antithetic Modification (AM)

A wide class of methods for modifying Gibbs sampling can be formulated as applying a sequence of *antithetic modifications* to the original Gibbs sampling transition matrix. All these modifications can be shown to produce a chain that efficiency-dominates Gibbs sampling, even though many do not Peskun-dominate it.

The concept of an antithetic modification can be applied to any transition probabilities, P, on a finite state space, $\mathcal{X}$, that are reversible with respect to some distribution, π, on $\mathcal{X}$. Two disjoint subsets of $\mathcal{X}$, $\mathcal{A}$ and $\mathcal{B}$, with $\pi(\mathcal{A})$ and $\pi(\mathcal{B})$ non-zero, are specified, along with a value $\delta > 0$ that controls the magnitude of the modification, which must satisfy $P(a \to a') \geq \delta\pi(a')\pi(\mathcal{B})/\pi(\mathcal{A})$ for all $a, a' \in \mathcal{A}$, and $P(b \to b') \geq \delta\pi(b')\pi(\mathcal{A})/\pi(\mathcal{B})$ for all $b, b' \in \mathcal{B}$.

The modification will alter only transitions to and from values that are both in $\mathcal{A} \cup \mathcal{B}$, with modified transition probabilities, P^*, as follows:

$$
\begin{aligned}
P^*(a \to a') &= P(a \to a') - \delta\pi(a')\pi(\mathcal{B})/\pi(\mathcal{A}), && \text{if } a \in \mathcal{A} \text{ and } a' \in \mathcal{A} \\
P^*(a \to b') &= P(a \to b') + \delta\pi(b'), && \text{if } a \in \mathcal{A} \text{ and } b' \in \mathcal{B} \\
P^*(b \to b') &= P(b \to b') - \delta\pi(b')\pi(\mathcal{A})/\pi(\mathcal{B}), && \text{if } b \in \mathcal{B} \text{ and } b' \in \mathcal{B} \\
P^*(b \to a') &= P(b \to a') + \delta\pi(a'), && \text{if } b \in \mathcal{B} \text{ and } a' \in \mathcal{A} \\
P^*(u \to v') &= P(u \to v'), && \text{if } u \notin \mathcal{A} \cup \mathcal{B} \text{ or } v' \notin \mathcal{A} \cup \mathcal{B}
\end{aligned}
\tag{19}
$$

One can easily verify that the modified transitions probabilities from each value are non-negative and sum to one, and that these transition probabilities are reversible with respect to π.

If P^* can be derived from P by applying a sequence of zero or more antithetic modifications, then I will say that P^* *is antithetically derivable from* P. It is easy to see that this is a partial order, since the only non-trivial condition for this is antisymmetry, which holds because an antithetic modification always reduces some self transition probabilities, and never increases any self transition probabilities, so it is not possible for P and Q to be antithetically derivable from each other unless they are equal.

Figure 2 shows an example in which three antithetic modifications are applied starting from an initial transition probability matrix with all rows equal to π.

For any P^* and P that are reversible with respect to π, if P^* Peskun-dominates P, then P^* must also be antithetically derivable from P by a sequence of modifications in which $\mathcal{A}$ and $\mathcal{B}$ are singleton sets. If $P^*(a \to b) > P(a \to b)$, an antithetic modification with $\mathcal{A} = \{a\}$, $\mathcal{B} = \{b\}$, and $\delta = \left(P^*(a \to b) - P(a \to b)\right) / \pi(b)$ will change the transition probabilities between a and b from those of P to those of P^*, without altering transition probabilities involving values other than a and b. By a sequence of such modifications, P^* is antithetically derivable from P.

However, an antithetic modification in which $\mathcal{A}$ and/or $\mathcal{B}$ have more than one element can change P to a P^* that efficiency-dominates P, but does not Peskun-dominate it. The first modification in Figure 2 provides an example: There, $P(2 \to 3)$ decreases from 1/6 to 1/15 after the first modification with $\mathcal{A} = \{1\}$ and $\mathcal{B} = \{2, 3\}$, while $P(1 \to 2)$ increases from 1/4 to 3/8, so neither the original nor the modified transition matrix Peskun-dominates the other.

However, whenever P^* is antithetically derivable from P it *does* efficiency-dominate P. This follows from Theorem 9 of the companion paper (Neal and Rosenthal 2023), which states that if P and Q are reversible irreducible Markov chains on a finite state space, then P efficiency-dominates Q if and only if the matrix $Q - P$ has only non-negative eigenvalues. (See also (Mira and Geyer 1999)).

$$P = \begin{bmatrix} \frac{1}{2} & \frac{1}{4} & \frac{1}{6} & \frac{1}{12} \\ \frac{1}{2} & \frac{1}{4} & \frac{1}{6} & \frac{1}{12} \\ \frac{1}{2} & \frac{1}{4} & \frac{1}{6} & \frac{1}{12} \\ \frac{1}{2} & \frac{1}{4} & \frac{1}{6} & \frac{1}{12} \end{bmatrix} \rightarrow \begin{bmatrix} \mathbf{\frac{7}{24}} & \mathbf{\frac{3}{8}} & \mathbf{\frac{3}{12}} & \frac{1}{12} \\ \mathbf{\frac{3}{4}} & \mathbf{\frac{1}{10}} & \mathbf{\frac{1}{15}} & \frac{1}{12} \\ \mathbf{\frac{3}{4}} & \mathbf{\frac{1}{10}} & \mathbf{\frac{1}{15}} & \frac{1}{12} \\ \frac{1}{2} & \frac{1}{4} & \frac{1}{6} & \frac{1}{12} \end{bmatrix} \rightarrow \begin{bmatrix} \frac{7}{24} & \frac{3}{8} & \frac{3}{12} & \frac{1}{12} \\ \frac{3}{4} & \frac{1}{10} & \frac{1}{15} & \frac{1}{12} \\ \frac{3}{4} & \frac{1}{10} & \mathbf{\frac{1}{30}} & \mathbf{\frac{7}{60}} \\ \frac{1}{2} & \frac{1}{4} & \mathbf{\frac{7}{30}} & \mathbf{\frac{1}{60}} \end{bmatrix} \rightarrow \begin{bmatrix} \mathbf{\frac{283}{1176}} & \mathbf{\frac{23}{56}} & \mathbf{\frac{23}{84}} & \mathbf{\frac{11}{147}} \\ \mathbf{\frac{23}{28}} & \mathbf{\frac{1}{20}} & \mathbf{\frac{1}{30}} & \mathbf{\frac{2}{21}} \\ \mathbf{\frac{23}{28}} & \mathbf{\frac{1}{20}} & \mathbf{0} & \mathbf{\frac{27}{210}} \\ \mathbf{\frac{22}{49}} & \mathbf{\frac{2}{7}} & \mathbf{\frac{27}{105}} & \mathbf{\frac{6}{735}} \end{bmatrix} = P^*$$

$$\begin{array}{ccc} \mathcal{A} = \{1\},\ \mathcal{B} = \{2,3\} & \mathcal{A} = \{3\},\ \mathcal{B} = \{4\} & \mathcal{A} = \{1,4\},\ \mathcal{B} = \{2,3\} \\ \delta = \frac{1}{2} & \delta = \frac{2}{5} & \delta = \frac{1}{7} \end{array}$$

Figure 2: Changes to a transition probability matrix through three successive antithetic modifications. On the left, P has all rows equal to π. Three antithetic modifications are then applied, with $\mathcal{A}$, $\mathcal{B}$, and δ as shown, to obtain P^*. For each modification, probabilities that change are shown in bold.

For a general antithetic modification, as defined by (19), the difference matrix, $P - P^*$, will be zero except for the submatrix corresponding to states in $\mathcal{A} \cup \mathcal{B}$. If we order states in $\mathcal{A} = \{a_1, a_2, \ldots\}$ before those in $\mathcal{B} = \{b_1, b_2, \ldots\}$, with any other states following, $P - P^*$ will look like this:

$$P - P^* \;=\; \begin{bmatrix} \delta\pi(a_1)\frac{\pi(\mathcal{B})}{\pi(\mathcal{A})} & \delta\pi(a_2)\frac{\pi(\mathcal{B})}{\pi(\mathcal{A})} & \cdots & -\delta\pi(b_1) & -\delta\pi(b_2) & \cdots & 0 & \cdots \\ \delta\pi(a_1)\frac{\pi(\mathcal{B})}{\pi(\mathcal{A})} & \delta\pi(a_2)\frac{\pi(\mathcal{B})}{\pi(\mathcal{A})} & \cdots & -\delta\pi(b_1) & -\delta\pi(b_2) & \cdots & 0 & \cdots \\ \vdots & \vdots & & \vdots & \vdots & & \vdots & \\ -\delta\pi(a_1) & -\delta\pi(a_2) & \cdots & \delta\pi(b_1)\frac{\pi(\mathcal{A})}{\pi(\mathcal{B})} & \delta\pi(b_2)\frac{\pi(\mathcal{A})}{\pi(\mathcal{B})} & \cdots & 0 & \cdots \\ -\delta\pi(a_1) & -\delta\pi(a_2) & \cdots & \delta\pi(b_1)\frac{\pi(\mathcal{A})}{\pi(\mathcal{B})} & \delta\pi(b_2)\frac{\pi(\mathcal{A})}{\pi(\mathcal{B})} & \cdots & 0 & \cdots \\ \vdots & \vdots & & \vdots & \vdots & & \vdots & \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & \cdots \\ \vdots & \vdots & & \vdots & \vdots & & \vdots & \end{bmatrix} \tag{20}$$

This is a rank-one matrix, since all rows are equal to the first row, or equal the first row times $-\pi(\mathcal{A})/\pi(\mathcal{B})$, or are zero. If we let D be the diagonal matrix with entries $\pi(a_1), \pi(a_2), \ldots, \pi(b_1), \pi(b_2), \ldots, 0, \ldots$ on its diagonal, and let

$$v \;=\; \begin{bmatrix} \sqrt{\frac{\pi(\mathcal{B})}{\pi(\mathcal{A})}} & \sqrt{\frac{\pi(\mathcal{B})}{\pi(\mathcal{A})}} & \cdots & -\sqrt{\frac{\pi(\mathcal{A})}{\pi(\mathcal{B})}} & -\sqrt{\frac{\pi(\mathcal{A})}{\pi(\mathcal{B})}} & \cdots & 0 & \cdots \end{bmatrix}^T \tag{21}$$

then we can write

$$P - P^* \;=\; \delta\, v\, v^T D \tag{22}$$

Since $v^T D v = \sum_{a\in\mathcal{A}} \pi(a)\pi(\mathcal{B})/\pi(\mathcal{A}) + \sum_{b\in\mathcal{B}} \pi(b)\pi(\mathcal{A})/\pi(\mathcal{B}) = \pi(\mathcal{A}) + \pi(\mathcal{B})$, we see that $(P - P^*)\, v = \delta\,(\pi(\mathcal{A}) + \pi(\mathcal{B}))\, v$, so $\delta\,(\pi(\mathcal{A}) + \pi(\mathcal{B}))$ is an eigenvalue of $P - P^*$, with all other eigenvalues being zero (since $P - P^*$ is of rank one). Since this eigenvalue is positive, Theorem 9 from (Neal and Rosenthal 2023) shows that P^* efficiency-dominates P.

Since efficiency-dominance is transitive, it follows that the result of any sequence of antithetic modifications will efficiency-dominate the original transition matrix.

A wide variety of improved methods can be derived using antithetic modifications. In this paper, I will focus on generic methods, in which nothing is known that distinguishes one state from another, except for their probabilities under π. However, antithetic modifications can also be designed in a way that exploits some known structure of the state space as a guide to how to choose the subsets $\mathcal{A}$ and $\mathcal{B}$.

For example, suppose it is beneficial for the value chosen from $\mathcal{X} = \{1, \ldots, m\}$ to be far from the current value. If $m = 2^j$, we can try to flip from the current value to one in the other half of $\mathcal{X}$, which will on average be more distant than a value chosen from all of $\mathcal{X}$. Failing that, we could try to flip from the current value to one in the other quarter of the same half, and so forth. To do this, we can modify the probabilities for independent sampling (all rows equal to π) by applying an antithetic modification with $\mathcal{A} = \{1, \ldots, 2^{j-1}\}$ and $\mathcal{B} = \{2^{j-1}+1, \ldots, 2^j\}$. If $\pi(\mathcal{A}) \geq \pi(\mathcal{B})$, we use $\delta = \pi(\mathcal{B})/\pi(\mathcal{A})$, which results in all transition probabilities amongst values in $\mathcal{B}$ being zero. Otherwise, we use $\delta = \pi(\mathcal{A})/\pi(\mathcal{B})$, and all transition probabilities amongst values in $\mathcal{A}$ will be zero. We then apply another antithetic modification, in the first case using $\mathcal{A} = \{1, \ldots, 2^{j-2}\}$ and $\mathcal{B} = \{2^{j-2}+1, \ldots, 2^{j-1}\}$, which partitions the previous $\mathcal{A}$, and in the second case using $\mathcal{A} = \{2^{j-1}+1, \ldots, 2^{j-1}+2^{j-2}\}$ and $\mathcal{B} = \{2^{j-1}+2^{j-2}+1, 2^j\}$, partitioning the previous $\mathcal{B}$, in both cases with δ chosen to make transition probabilities within either $\mathcal{A}$ or $\mathcal{B}$ zero. This continues until $\mathcal{A}$ and $\mathcal{B}$ are singleton sets.

Here is an example with $m = 4$:

$$\begin{bmatrix} \frac{1}{4} & \frac{3}{10} & \frac{1}{5} & \frac{1}{4} \\ \frac{1}{4} & \frac{3}{10} & \frac{1}{5} & \frac{1}{4} \\ \frac{1}{4} & \frac{3}{10} & \frac{1}{5} & \frac{1}{4} \\ \frac{1}{4} & \frac{3}{10} & \frac{1}{5} & \frac{1}{4} \end{bmatrix} \rightarrow \begin{bmatrix} \frac{10}{121} & \frac{12}{121} & \frac{4}{11} & \frac{5}{11} \\ \frac{10}{121} & \frac{12}{121} & \frac{4}{11} & \frac{5}{11} \\ \frac{5}{11} & \frac{6}{11} & 0 & 0 \\ \frac{5}{11} & \frac{6}{11} & 0 & 0 \end{bmatrix} \rightarrow \begin{bmatrix} 0 & \frac{2}{11} & \frac{4}{11} & \frac{5}{11} \\ \frac{5}{33} & \frac{1}{33} & \frac{4}{11} & \frac{5}{11} \\ \frac{5}{11} & \frac{6}{11} & 0 & 0 \\ \frac{5}{11} & \frac{6}{11} & 0 & 0 \end{bmatrix} \tag{23}$$

Note that only the row of this matrix for transition probabilities from the current value need be computed.

This procedure is equivalent to one used for the No-U-Turn Sampler by Hoffman and Gelman (2014, Section 3.1.2) to select from amongst states found by simulating a trajectory using Hamiltonian dynamics.

When the goal is to improve Gibbs sampling, antithetic modifications can be applied to the Gibbs sampling transition matrix for updating a particular variable, when other variables have particular values. When the variable to be updated is selected randomly, each such modification will improve the efficiency of the overall chain, and hence so will a set of antithetic modifications for updates to every variable, for every combination of values for other variables.

To see this is detail, suppose that there are two state variables, so $\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2$, with $X_1 = \{1, 2\}$ and $X_2 = \{1, 2, 3\}$, and that $\pi((1,1)) = 1/8$, $\pi((1,2)) = 1/4$, $\pi((1,3)) = 1/8$, $\pi((2,1)) = 1/4$, $\pi((2,2)) = 1/8$, and $\pi((2,3)) = 1/8$. With states ordered lexicographically (i.e., with $\mathcal{X}_1$ changing more slowly), the transition matrices for Gibbs sampling updates of the first and second variables will be

$$P_1 = \begin{bmatrix} 1/3 & 0 & 0 & 2/3 & 0 & 0 \\ 0 & 2/3 & 0 & 0 & 1/3 & 0 \\ 0 & 0 & 1/2 & 0 & 0 & 1/2 \\ 1/3 & 0 & 0 & 2/3 & 0 & 0 \\ 0 & 2/3 & 0 & 0 & 1/3 & 0 \\ 0 & 0 & 1/2 & 0 & 0 & 1/2 \end{bmatrix}, \quad P_2 = \begin{bmatrix} 1/4 & 1/2 & 1/4 & 0 & 0 & 0 \\ 1/4 & 1/2 & 1/4 & 0 & 0 & 0 \\ 1/4 & 1/2 & 1/4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/2 & 1/4 & 1/4 \\ 0 & 0 & 0 & 1/2 & 1/4 & 1/4 \\ 0 & 0 & 0 & 1/2 & 1/4 & 1/4 \end{bmatrix} \tag{24}$$

State order: (1,1) (1,2) (1,3) (2,1) (2,2) (2,3) (1,1) (1,2) (1,3) (2,1) (2,2) (2,3)

If the order of states were changed so that $\mathcal{X}_2$ changed more slowly, P_1 would change to

$$\widetilde{P}_1 \;=\; \begin{bmatrix} 1/3 & 2/3 & 0 & 0 & 0 & 0 \\ 1/3 & 2/3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2/3 & 1/3 & 0 & 0 \\ 0 & 0 & 2/3 & 1/3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/2 & 1/2 \\ 0 & 0 & 0 & 0 & 1/2 & 1/2 \end{bmatrix} \tag{25}$$

State order: (1,1) (2,1) (1,2) (2,2) (1,3) (2,3)

So, with a suitable order, both Gibbs sampling updates have block-diagonal transition matrices, with each block being the transition matrix for an update of that one variable, which is reversible with respect to the conditional distribution for that variable given the current value of the other variable (of all other variables, when there are more than two variables). If the variable to update is selected uniformly at random, the transition probability matrix for the entire chain is $P = (1/2)(P_1 + P_2)$. More generally, when there are n variables, $P = (1/n)(P_1 + \cdots + P_n)$.

We can apply an antithetic modification that affects only a single block, in the update for one variable. For example, applying equations (19), the block for updating the second variable in the above example when the first variable has the value 2 can be antithetically modified with $\mathcal{A} = \{(2,1)\}$, $\mathcal{B} = \{(2,2),(2,3)\}$, and $\delta = 2$, giving the following modified version of P_2:

$$P_2^* \;=\; \begin{bmatrix} 1/4 & 1/2 & 1/4 & 0 & 0 & 0 \\ 1/4 & 1/2 & 1/4 & 0 & 0 & 0 \\ 1/4 & 1/2 & 1/4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/2 & 1/2 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \tag{26}$$

State order: (1,1) (1,2) (1,3) (2,1) (2,2) (2,3)

Note that this can also be viewed as an antithetic modification to just the lower-right block, regarding it as a transition matrix that is reversible with respect to a conditional distribution for that variable:

$$\begin{bmatrix} 1/2 & 1/4 & 1/4 \\ 1/2 & 1/4 & 1/4 \\ 1/2 & 1/4 & 1/4 \end{bmatrix} \;\rightarrow\; \begin{bmatrix} 0 & 1/2 & 1/2 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \tag{27}$$

For this block modification, π is the conditional distribution, with probabilities 1/2, 1/4, 1/4, and $\delta = 1$.

As discussed above, the eigenvalues of the difference between the original transition matrix P_2 and the antithetically-modified matrix P_2^* will all be non-negative. But we cannot conclude that P_2^* efficiency-dominates P_2, because neither of these are irreducible — on their own, they cannot move over the full state space. We *can* conclude that the modified overall transition matrix with random selection of variable to update, $(1/2)(P_1 + P_2^*)$, efficiency-dominates the original overall transition matrix, $(1/2)(P_1 + P_2)$, provided these are both irreducible, by using Theorem 12 of (Neal and Rosenthal 2023). More generally, $P^* = (1/n)(P_1^* + \cdots + P_n^*)$ efficiency-dominates $P = (1/n)(P_1 + \cdots + P_n)$ if each of the differences $P_k - P_k^*$ has only non-negative eigenvalues, provided P and P^* are irreducible and the P_k and P_k^* are reversible.

Accordingly, when antithetic modification is used to improve the efficiency of individual Gibbs sampling updates, this improvement extends to an overall method that randomly selects a variable to update. Note,

however, that this guarantee does not apply when variables are updated in some systematic order, even though, as will be seen later in the empirical evaluations, this is often better than random updates.

An antithetic modification may produce transition probabilities that do not converge to π, but instead are periodic, flipping between different distributions at even and odd iterations. Seen in isolation, averages from such an update will nevertheless be correct estimates of expectations. When such transitions are used to update single variables in a Gibbs sampling framework, with the variable to update chosen randomly, such exact periodicity is possible,[7] though rare, but averages will still be correct even with periodicity. When variables are updated in some systematic order, rather than randomly, it is possible for periodicity of individual updates to produce incorrect estimates.[8] Though this problem is rare in practice, if necessary it can be avoided by occasionally doing an unmodified Gibbs sampling update.

6 Nested Antithetic Modification (NAM) methods

I will now look at methods in which a sequence of $m-1$ antithetic modifications are applied to a transition matrix in which all rows are the same, as for a Gibbs sampling update. These antithetic modifications will use subsets of states, $\mathcal{A}_i$ and $\mathcal{B}_i$, for $i = 1, \ldots, m-1$, in which each $\mathcal{A}_i = \{a_i\}$ is a singleton set contained in $\mathcal{B}_{i-1}$ and $\mathcal{B}_i = \mathcal{B}_{i-1} \setminus \mathcal{A}_i$, with $\mathcal{B}_0 = \mathcal{X}$. I call these *nested antithetic modification (NAM)* methods, since $\mathcal{X} = \mathcal{B}_0 \supset \mathcal{B}_1 \supset \mathcal{B}_2 \supset \cdots \supset \mathcal{B}_{m-1}$ are nested sets of states. Different NAM methods result from different ways of choosing which element of $\mathcal{B}_{i-1}$ is chosen as a_i — what I will call the *focal value* for that stage in the sequence. For all these antithetic modifications, δ will be chosen to be as large as possible.

When focal values $a_1, a_2, \ldots, a_{m-1}$ are chosen to have non-decreasing probability under π, the resulting NAM method turns out to be equivalent to a method described by Frigessi, Hwang, and Younes (1992), and later independently by Tjelmeland (2004). In this case, the modified transitions Peskun-dominate Gibbs sampling. This is not true for all NAM methods, but, as discussed in the previous section, they, like all AM methods, do efficiency-dominate Gibbs sampling.

In this section, I look at NAM methods in general, for any selection of $a_1, a_2, \ldots, a_{m-1}$, and present efficient implementations of these methods. For notational simplicity, I will describe how these methods would be applied to modify transitions for the entire state, but in practice they will modify Gibbs sampling probabilities for a single state variable, with π being the conditional distribution for that variable given the current values of other variables.

I will present NAM methods assuming that the state space is $\mathcal{X} = \{1, \ldots, m\}$. The choice of focal values can then be represented using a permutation, σ, on $1, \ldots, m$, with $a_i = \sigma(i)$. The antithetic modifications will produce successive transition probability matrices $P_0, P_1, \ldots, P_{m-1}$, where P_0 has all rows equal to π (i.e., the Gibbs sampling probabilities), and P_i is the result of applying an antithetic modification to P_{i-1} with $\mathcal{A}_i = \{\sigma(i)\}$ and $\mathcal{B}_i = \{\sigma(j) \ : \ j = i+1, \ldots, m\}$. In some cases, all modified transition probabilities, $P_i(b \to b')$, for $b, b' \in \mathcal{B}_i$ will be zero at stage i, in which case the procedure is terminated at that point, with P_i being the final modified transition matrix.

The submatrix of P_i with rows and columns in $\mathcal{B}_i$ will always have all rows the same, with row elements proportional to $\pi(b')$ for $b' \in \mathcal{B}_i$. This is obvious for P_0, and can be seen below to carry over from P_{i-1} to P_i,

[7] Let π be uniform over $\mathcal{X} = \{0,1\} \times \{0,1\} \times \{0,1\}$. There is an antithetic modification of Gibbs sampling for each variable that flips the value with probability one. With random selection of the variable to update, the number of 1s will alternate in periodic fashion between an even number and an odd number when this modified method is used.

[8] With the same example as in footnote 7, flipping values in a systematic scan starting at state $(0,0,0)$ will produce the cycle $(0,0,0)$, $(1,0,0)$, $(1,1,0)$, $(1,1,1)$, $(0,1,1)$, $(0,0,1)$, $(0,0,0)$, ..., in which the values $(0,1,0)$ and $(1,0,1)$ never appear.

since the changes from $P_{i-1}(b \to b')$ to $P_i(b \to b')$ for $b, b' \in \mathcal{B}_i$ do not depend on b and are proportional to $\pi(b')$. Define the ratio of the sum of a row of this submatrix to the sum of probabilities for its values as follows:

$$r_i \;=\; \frac{1}{\pi(\mathcal{B}_i)} \sum_{b' \in \mathcal{B}_i} P_i(b \to b') \tag{28}$$

This will be the same for any $b \in \mathcal{B}_i$. Since $\mathcal{B}_0 = \mathcal{X}$, we will have $r_0 = 1$. We can express the transition probabilities in this sub-matrix as

$$P_i(b \to b') \;=\; r_i \, \pi(b') \tag{29}$$

for any b and b' in $\mathcal{B}_i$.

Stage i of the NAM procedure will operate differently depending on whether or not $\pi(\mathcal{A}_i) < \pi(\mathcal{B}_i)$. If $\pi(\mathcal{A}_i) = \pi(a_i) = \pi(\sigma(i)) < \pi(\mathcal{B}_i)$, we set $\delta_i = r_{i-1} \, \pi(a_i) \,/\, \pi(\mathcal{B}_i)$. Using (19), this gives P_i as follows:

$$\begin{array}{lcll}
P_i(a_i \to a_i) &=& 0 & \\
P_i(a_i \to b') &=& P_{i-1}(a_i \to b') + \delta_i \pi(b'), & \text{if } b' \in \mathcal{B}_i \\
P_i(b \to b') &=& P_{i-1}(b \to b') - \delta_i \pi(b') \pi(a_i) / \pi(\mathcal{B}_i), & \text{if } b \in \mathcal{B}_i \text{ and } b' \in \mathcal{B}_i \\
P_i(b \to a_i) &=& P_{i-1}(b \to a_i) + \delta_i \pi(a_i), & \text{if } b \in \mathcal{B}_i \\
P_i(u \to v') &=& P_{i-1}(u \to v'), & \text{if } u \notin \mathcal{A}_i \cup \mathcal{B}_i \text{ or } v' \notin \mathcal{A}_i \cup \mathcal{B}_i
\end{array} \tag{30}$$

We can see that the value of $P_i(b \to b')$ above will be positive using (29) twice, along with $\pi(a_i) < \pi(\mathcal{B}_i)$:

$$\delta_i \pi(b') \pi(a_i) / \pi(\mathcal{B}_i) \;<\; \delta_i \pi(b') \;=\; r_{i-1} \pi(b') \pi(a_i) / \pi(\mathcal{B}_i) \;=\; P_{i-1}(b \to b') \pi(a_i) / \pi(\mathcal{B}_i) \;<\; P_{i-1}(b \to b') \tag{31}$$

so $P_i(b \to b') = P_{i-1}(b \to b') - \delta_i \pi(b') \pi(a_i) / \pi(\mathcal{B}_i)$ is positive.

When instead $\pi(\mathcal{A}_i) = \pi(a_i) \geq \pi(\mathcal{B}_i)$, we make use of (19) with $\delta_i = r_{i-1} \, \pi(\mathcal{B}_i) \,/\, \pi(a_i)$ and obtain

$$\begin{array}{lcll}
P_i(a_i \to a_i) &=& P_{i-1}(a_i \to a_i) - \delta_i \pi(\mathcal{B}_i) & \\
P_i(a_i \to b') &=& P_{i-1}(a_i \to b') + \delta_i \pi(b'), & \text{if } b' \in \mathcal{B}_i \\
P_i(b \to b') &=& 0, & \text{if } b \in \mathcal{B}_i \text{ and } b' \in \mathcal{B}_i \\
P_i(b \to a_i) &=& P_{i-1}(b \to a_i) + \delta_i \pi(a_i), & \text{if } b \in \mathcal{B}_i \\
P_i(u \to v') &=& P_{i-1}(u \to v'), & \text{if } u \notin \mathcal{A}_i \cup \mathcal{B}_i \text{ or } v' \notin \mathcal{A}_i \cup \mathcal{B}_i
\end{array} \tag{32}$$

$P_i(b \to b') = 0$ because applying equation (29) to its expression in (19), and noting that $\mathcal{A}_i = \{a_i\}$, gives

$$P_{i-1}(b \to b') - \delta_i \pi(b') \pi(\mathcal{A}_i) / \pi(\mathcal{B}_i) \;=\; P_{i-1}(b \to b') - r_{i-1} \pi(b') \;=\; P_{i-1}(b \to b') - P_{i-1}(b \to b') \;=\; 0 \tag{33}$$

$P_i(a_i \to a_i)$ is guaranteed to be non-negative because, using $\pi(a_i) \geq \pi(\mathcal{B}_i)$ and equation (29),

$$\delta_i \pi(\mathcal{B}_i) \;\leq\; \delta_i \pi(a_i) \;=\; r_{i-1} \pi(\mathcal{B}_i) \;\leq\; r_{i-1} \pi(a_i) \;=\; P_{i-1}(a_i \to a_i) \tag{34}$$

so $P_i(a_i \to a_i) \;=\; P_{i-1}(a_i \to a_i) \;-\; \delta_i \pi(\mathcal{B}_i)$ is non-negative. Since a modification in which $\pi(a_i) \geq \pi(\mathcal{B}_i)$ results in $P_i(b \to b')$ being zero for all $b, b' \in \mathcal{B}_i$, the NAM procedure is terminated at this point, with P_i being the final result.

Figures 3 and 4 shows two examples of Nested Antithetic Modification, with different orderings, σ, of focal values. The example in Figure 4 ends after the second stage, when the probability of the focal value ($a_2 = 4$) is as large as the probability of the remaining values (in $\mathcal{B}_2 = \{1, 2\}$). Subsequent stages would operate on an all-zero sub-matrix, and hence do nothing.

$$
\begin{array}{cccc}
r_0 = 1 & & r_1 = \tfrac{80}{81} & \qquad\qquad r_2 = \tfrac{400}{441} \qquad\qquad\qquad r_3 = \tfrac{25}{63} \\
\left[\begin{array}{cccc} \frac{1}{10} & \frac{2}{10} & \frac{3}{10} & \frac{4}{10} \\ \frac{1}{10} & \frac{2}{10} & \frac{3}{10} & \frac{4}{10} \\ \frac{1}{10} & \frac{2}{10} & \frac{3}{10} & \frac{4}{10} \\ \frac{1}{10} & \frac{2}{10} & \frac{3}{10} & \frac{4}{10} \end{array}\right] & \rightarrow & \left[\begin{array}{cccc} 0 & \frac{2}{9} & \frac{3}{9} & \frac{4}{9} \\ \frac{1}{9} & \frac{16}{81} & \frac{24}{81} & \frac{32}{81} \\ \frac{1}{9} & \frac{16}{81} & \frac{24}{81} & \frac{32}{81} \\ \frac{1}{9} & \frac{16}{81} & \frac{24}{81} & \frac{32}{81} \end{array}\right] \rightarrow \left[\begin{array}{cccc} 0 & \frac{2}{9} & \frac{3}{9} & \frac{4}{9} \\ \frac{1}{9} & 0 & \frac{24}{63} & \frac{32}{63} \\ \frac{1}{9} & \frac{16}{63} & \frac{120}{441} & \frac{160}{441} \\ \frac{1}{9} & \frac{16}{63} & \frac{120}{441} & \frac{160}{441} \end{array}\right] \rightarrow \left[\begin{array}{cccc} 0 & \frac{2}{9} & \frac{3}{9} & \frac{4}{9} \\ \frac{1}{9} & 0 & \frac{24}{63} & \frac{32}{63} \\ \frac{1}{9} & \frac{16}{63} & 0 & \frac{40}{63} \\ \frac{1}{9} & \frac{16}{63} & \frac{30}{63} & \frac{10}{63} \end{array}\right] \\
\mathcal{A}_1 = \{1\},\ \mathcal{B}_1 = \{2,3,4\} & & \mathcal{A}_2 = \{2\},\ \mathcal{B}_2 = \{3,4\} & \qquad\qquad \mathcal{A}_3 = \{3\},\ \mathcal{B}_3 = \{4\} \\
\delta_1 = \tfrac{1}{9} & & \delta_2 = \tfrac{160}{567} & \qquad\qquad \delta_3 = \tfrac{300}{441}
\end{array}
$$

Figure 3: An example of the Nested Antithetic Modification (NAM) method, with $m = 4$ values having probabilities 1/10, 2/10, 3/10, 4/10, ordered by $\sigma(i) = i$. The arrows show transition probabilities being modified at each stage, using (30), since here $\pi(a_i) < \pi(\mathcal{B}_i)$ at every stage. Compare with the MHGS modification in Figure 1. Note that the final result has all off-diagonal transition probabilities smaller than in the original, and hence Peskun-dominates it.

$$
\begin{array}{cccc}
r_0 = 1 & & r_1 = \tfrac{40}{49} & \\
\left[\begin{array}{cccc} \frac{1}{10} & \frac{2}{10} & \frac{3}{10} & \frac{4}{10} \\ \frac{1}{10} & \frac{2}{10} & \frac{3}{10} & \frac{4}{10} \\ \frac{1}{10} & \frac{2}{10} & \frac{3}{10} & \frac{4}{10} \\ \frac{1}{10} & \frac{2}{10} & \frac{3}{10} & \frac{4}{10} \end{array}\right] & \rightarrow & \left[\begin{array}{cccc} \frac{4}{49} & \frac{8}{49} & \frac{3}{7} & \frac{16}{49} \\ \frac{4}{49} & \frac{8}{49} & \frac{3}{7} & \frac{16}{49} \\ \frac{1}{7} & \frac{2}{7} & 0 & \frac{4}{7} \\ \frac{4}{49} & \frac{8}{49} & \frac{3}{7} & \frac{16}{49} \end{array}\right] \rightarrow \left[\begin{array}{cccc} 0 & 0 & \frac{3}{7} & \frac{28}{49} \\ 0 & 0 & \frac{3}{7} & \frac{28}{49} \\ \frac{1}{7} & \frac{2}{7} & 0 & \frac{4}{7} \\ \frac{7}{49} & \frac{14}{49} & \frac{3}{7} & \frac{7}{49} \end{array}\right] \\
\mathcal{A}_1 = \{3\},\ \mathcal{B}_1 = \{1,2,4\} & & \mathcal{A}_2 = \{4\},\ \mathcal{B}_2 = \{1,2\} & \\
\delta_1 = \tfrac{3}{7} & & \delta_2 = \tfrac{30}{49} &
\end{array}
$$

Figure 4: The same example as in Figure 3, except with $\sigma(1) = 3$ and $\sigma(2) = 4$, so for the second stage, $\pi(a_i) > \pi(\mathcal{B}_i)$, and hence the modification is done using (32). Since this sets the $\{1, 2\}$ sub-matrix to all zeros, the procedure ends after this stage. Note that the final result does not Peskun-dominate the original, since $P(1 \rightarrow 2)$ and $P(2 \rightarrow 1)$ decrease to zero, but the new matrix does efficiency-dominate the original, as discussed in Section 5.

When simulating a Markov chain, we need only the row of the transition matrix giving the transition probabilities from the current state value, k. Algorithm 2 computes just these probabilities, given a particular order, σ, of focal values, taking time proportional to the number of possible values, m.

The algorithm considers successive focal values, $a_i = \sigma(i)$ for $i = 1, 2, \ldots$, but rather than compute the whole transition matrix, for each focal value it computes only the single transition probability from the current value, k, to that focal value, until k itself is the focal value. Once k is the focal value, the transition probabilities from k to all remaining values are computed. Note that once transition probabilities to and from a focal value are computed at some stage, they are not modified by later stages, so there is no need to consider further focal values past k.

Input: Gibbs sampling probabilities, $\pi(i)$, for $i = 1, \ldots, m$
A permutation, σ, on $\{1, \ldots, m\}$ giving the order of focal values
The current state value, k, in $\{1, \ldots, m\}$

Output: NAM transition probabilities from k, as $p(i)$ for $i = 1, \ldots, m$

Set s to 1 *The sum of probabilities for values that have not yet been focal*

Set f to 1 *The sum of transition probabilities from k to values that have not yet been focal*

Find modified transition probabilities from the current value to successive focal values, until the focal value is the current value

Set i to 1

While $\sigma(i) \neq k$:

After seeing a focal value with probability at least as large as remaining values, just store zeros (can change $f \leq 0$ to $f \leq \epsilon$ for small ϵ to avoid tiny probabilities from rounding)

If $f \leq 0$:

Set $p(\sigma(i))$ to 0

Else:

Let q be the probability of the focal value; update s to be the sum of probabilities for remaining non-focal values

Set q to $\pi(\sigma(i))$
Subtract q from s *Sets variable s to s_i*

Compute the transition probability from the current value, k, to the focal value, and find the new total probability for transitions to remaining values

If $q \geq s$:
Set $p(\sigma(i))$ to f
Set f to 0

Else:
Set $p(\sigma(i))$ to $(q/s)f$ *Guarantees $p(\sigma(i)) \leq f \leq 1$, even with rounding*
Subtract $p(\sigma(i))$ from f *Sets variable f to f_i, was previously f_{i-1}*

Add 1 to i

Compute modified transition probabilities from the current value, k, which is now focal, to values that have not previously been focal, as well as the self transition probability for k

If $f \leq 0$:

Set $p(k)$ to 0
For $j = i{+}1, \ldots, m$: Set $p(\sigma(j))$ to 0

Else:

Set q to $\pi(k)$
Subtract q from s

If $q > s$:
Set $p(k)$ to $((q{-}s)/q)f$ *Guarantees $p(k) \leq f \leq 1$, even with rounding*
For $j = i{+}1, \ldots, m$: Set $p(\sigma(j))$ to $\min(f, (\pi(\sigma(j))/q)f)$ *Min in case of rounding*
Else:
Set $p(k)$ to 0
For $j = i{+}1, \ldots, m$: Set $p(\sigma(j))$ to $\min(f, (\pi(\sigma(j))/s)f)$ *Min in case of rounding*

Algorithm 2: Computation of modified transition probabilities from the current value by the NAM method.

Algorithm 2 incrementally maintains two sums:

$$s_i \;=\; \sum_{j=i+1}^{m} \pi(\sigma(j)) \;=\; \pi(\mathcal{B}_i) \tag{35}$$

$$f_i \;=\; \sum_{j=i+1}^{m} P_i(k \to \sigma(j)) \tag{36}$$

Starting from $s_0 = 1$ and $f_0 = 1$, these are updated by

$$s_i \;=\; s_{i-1} \;-\; \pi(\sigma(i)) \tag{37}$$

$$f_i \;=\; f_{i-1} \;-\; P_i(k \to \sigma(i)) \tag{38}$$

for $j = 1, 2, \ldots$ until $\sigma(j) = k$. Note that $r_i = f_i/s_i$, and that the values of f_i, s_i, and r_i do not actually depend on the value of k.

The stage i computation for the transition probability from the current value, k, to a focal value, $a_i = \sigma(i)$, when $\pi(a_i) < \pi(\mathcal{B}_i) = s_i$, can be re-written from its form in (30) as follows, using equation (29):

$$P_i(k \to a_i) \;=\; P_{i-1}(k \to a_i) \;+\; \delta_i \pi(a_i) \tag{39}$$

$$\;=\; r_{i-1}\pi(a_i) \;+\; r_{i-1}\left(\pi(a_i)/\pi(\mathcal{B}_i)\right)\pi(a_i) \tag{40}$$

$$\;=\; \frac{f_{i-1}}{s_{i-1}}\pi(a_i)\,\left[1 + \pi(a_i)/\pi(\mathcal{B}_i)\right] \tag{41}$$

$$\;=\; \frac{f_{i-1}}{s_{i-1}}\pi(a_i)\,\frac{\pi(a_i) + \pi(\mathcal{B}_i)}{\pi(\mathcal{B}_i)} \tag{42}$$

$$\;=\; \frac{f_{i-1}}{s_{i-1}}\pi(a_i)\,\frac{s_{i-1}}{s_i} \;=\; \frac{f_{i-1}}{s_i}\pi(a_i) \tag{43}$$

This is the method that Algorithm 2 uses to compute $P_i(k \to a_i)$, which is $p(\sigma(i))$ in the program, after which it updates f_{i-1} to f_i by subtracting the result, as in equation (38).

If $\pi(a_i) \geq \pi(\mathcal{B}_i) = s_i$ at some stage before k becomes the focal value, (32) gives

$$P_i(k \to a_i) \;=\; P_{i-1}(k \to a_i) \;+\; \delta_i \pi(a_i) \tag{44}$$

$$\;=\; r_{i-1}\pi(a_i) \;+\; r_{i-1}(\pi(\mathcal{B}_i)/\pi(a_i))\pi(a_i) \tag{45}$$

$$\;=\; r_{i-1}\left[\pi(a_i) \;+\; \pi(\mathcal{B}_i)\right. \tag{46}$$

$$\;=\; \frac{f_{i-1}}{s_{i-1}}\,s_{i-1} \;=\; f_{i-1} \tag{47}$$

and transition probabilities from k to all remaining values are zero.

When instead $\pi(a_i) = \pi(\sigma(i))$ is less than $\pi(\mathcal{B}_i)$ for all stages prior to when k becomes the focal value, the transition probabilities from k to the remaining values are found once k is the focal value using either (30) or (32). When k becomes the focal value at stage i, so $k = a_i = \sigma(i)$, then if $\pi(k) = \pi(a_i) < \pi(\mathcal{B}_i) = s_i$,

using (30) gives $P_i(k \to k) = 0$, and for $b' \in \mathcal{B}_i$,

$$\begin{aligned} P_i(k \to b') &= P_{i-1}(k \to b') \ + \ \delta_i \pi(b') && (48)\\ &= r_{i-1}\pi(b') \ + \ r_{i-1}\left(\pi(k)/\pi(\mathcal{B}_i)\right)\pi(b') && (49)\\ &= \frac{f_{i-1}}{s_{i-1}}\pi(b')\ [1 + \pi(k)/\pi(\mathcal{B}_i))] && (50)\\ &= \frac{f_{i-1}}{s_{i-1}}\pi(b')\,\frac{\pi(k) + \pi(\mathcal{B}_i)}{\pi(\mathcal{B}_i)} && (51)\\ &= \frac{f_{i-1}}{s_{i-1}}\pi(b')\,\frac{s_{i-1}}{s_i} \quad = \quad \frac{f_{i-1}}{s_i}\pi(b') && (52) \end{aligned}$$

Whereas, if $\pi(k) = \pi(a_i) \geq \pi(\mathcal{B}_i) = s_i$, using (32) gives

$$\begin{aligned} P_i(k \to k) &= P_{i-1}(k \to k) \ - \ \delta_i \pi(\mathcal{B}_i) && (53)\\ &= r_{i-1}\pi(k) \ - \ r_{i-1}(\pi(\mathcal{B}_i)/\pi(k))\pi(\mathcal{B}_i) && (54)\\ &= \frac{f_{i-1}}{s_{i-1}}\left[\pi(k) - \pi(\mathcal{B}_i)^2/\pi(k)\right] && (55)\\ &= \frac{f_{i-1}}{\pi(k) + s_i}\,\frac{\pi(k)^2 - s_i^2}{\pi(k)} \quad = \quad f_{i-1}\,\frac{\pi(k) - s_i}{\pi(k)} && (56) \end{aligned}$$

and for $b' \in \mathcal{B}_i$,

$$\begin{aligned} P_i(k \to b') &= P_{i-1}(k \to b') \ + \ \delta_i \pi(b') && (57)\\ &= r_{i-1}\pi(b') \ + \ r_{i-1}(\pi(\mathcal{B}_i)/\pi(k))\pi(b') && (58)\\ &= \frac{f_{i-1}}{s_{i-1}}\pi(b')\left[1 + \pi(\mathcal{B}_i)/\pi(k)\right] && (59)\\ &= \frac{f_{i-1}}{s_{i-1}}\pi(b')\,\frac{\pi(k) + \pi(\mathcal{B}_i)}{\pi(k)} && (60)\\ &= \frac{f_{i-1}}{s_{i-1}}\pi(b')\,\frac{s_{i-1}}{\pi(k)} \quad = \quad \frac{f_{i-1}}{\pi(k)}\pi(b') && (61) \end{aligned}$$

These formulas are used for the computations at the end of Algorithm 2.

As presented, Algorithm 2 computes all transition probabilities from the current value of the state, which will subsequently be used to sample the value for the next state. This is inefficient when many of these transition probabilities are zero, as occurs when at some point $\pi(a_i) \geq s_i$. The algorithm could be modified to return only the non-zero transition probabilities, which also saves time when sampling. Note also that when $\pi(\sigma(1))$ is $1/2$ or more, any value other than $\sigma(1)$ has probability one of transitioning to $\sigma(1)$, so in this case there is no need to generate a random variate except when the current value is $\sigma(1)$.

It would also be possible to modify Algorithm 2 so that, rather than returning transition probabilities from state k, it instead returns a value randomly sampled according to these probabilities. Since Algorithm 2 computes transition probabilities in the order σ, and does not change them once they are first computed, this could be done by sampling a number, U, uniformly distributed on $[0, 1]$, maintaining the cumulative sum of transition probabilities computed so far, and returning the value just considered once this cumulative sum exceeds U. This would save some computation time, though the savings would not

be dramatic in the typical case where computing the normalized probabilities, π, requires looking at all m values in any case.

When $\pi(\sigma(i)) < s_i$ at every step, we can visualize the full matrix of transition probabilities computed by this algorithm (a row at a time) as illustrated below, for $m = 5$ and $\sigma(i) = i$:

$$P^* = \begin{bmatrix} 0 & \pi(2)\frac{f_0}{s_1} & \pi(3)\frac{f_0}{s_1} & \pi(4)\frac{f_0}{s_1} & \pi(5)\frac{f_0}{s_1} \\ \pi(1)\frac{f_0}{s_1} & 0 & \pi(3)\frac{f_1}{s_2} & \pi(4)\frac{f_1}{s_2} & \pi(5)\frac{f_1}{s_2} \\ \pi(1)\frac{f_0}{s_1} & \pi(2)\frac{f_1}{s_2} & 0 & \pi(4)\frac{f_2}{s_3} & \pi(5)\frac{f_2}{s_3} \\ \pi(1)\frac{f_0}{s_1} & \pi(2)\frac{f_1}{s_2} & \pi(3)\frac{f_2}{s_3} & 0 & \pi(5)\frac{f_3}{s_4} \\ \pi(1)\frac{f_0}{s_1} & \pi(2)\frac{f_1}{s_2} & \pi(3)\frac{f_2}{s_3} & \pi(4)\frac{f_3}{s_4} & f_4 \end{bmatrix} \tag{62}$$

Notice that under the diagonal the values in a column are all the same, and that above the diagonal the values in a row equal the probabilities from π times a common factor.

When $\pi(\sigma(i)) \geq s_i$ at some point, the transition matrix is the same as above for rows before i and for columns before i, but then is different for the submatrix of rows and columns from i and later, as is illustrated below, when $m = 5$, $\sigma(i) = i$, and $\pi(3) \geq s_3 = \pi(4) + \pi(5)$:

$$P^* = \begin{bmatrix} 0 & \pi(2)\frac{f_0}{s_1} & \pi(3)\frac{f_0}{s_1} & \pi(4)\frac{f_0}{s_1} & \pi(5)\frac{f_0}{s_1} \\ \pi(1)\frac{f_0}{s_1} & 0 & \pi(3)\frac{f_1}{s_2} & \pi(4)\frac{f_1}{s_2} & \pi(5)\frac{f_1}{s_2} \\ \pi(1)\frac{f_0}{s_1} & \pi(2)\frac{f_1}{s_2} & \frac{\pi(3)-s_3}{\pi(3)} f_2 & \frac{\pi(4)}{\pi(3)} f_2 & \frac{\pi(5)}{\pi(3)} f_2 \\ \pi(1)\frac{f_0}{s_1} & \pi(2)\frac{f_1}{s_2} & f_2 & 0 & 0 \\ \pi(1)\frac{f_0}{s_1} & \pi(2)\frac{f_1}{s_2} & f_2 & 0 & 0 \end{bmatrix} \tag{63}$$

The eigenvalues and eigenvectors of these transition matrices are of some interest. For a transition matrix for the entire state, the eigenvalues determine the rate of convergence of the Markov chain. However, this connection does not hold for partial transitions that update a single variable, rather than the entire state, as for Gibbs sampling and its modifications. Nevertheless, the eigenvalues provide some insight. An eigenvalue of one always exists, with right eigenvector of all ones, since each row of transition probabilities sums to one. When the rows are all equal to π (as for a Gibbs sampling update of a single variable, seen in isolation from others), all the remaining eigenvalues are zero, reflecting immediate convergence to π after one transition. Negative eigenvalues correspond to "antithetic" aspects of the transition, which reduce asymptotic variance, even compared to when the eigenvalues.

For notational simplicity, suppose that $\sigma(i) = i$ for all i. Then at each NAM step, i, prior to any at which $\pi(i) \geq s_i$, we can identify an eigenvalue of $\lambda_i = -\pi(i) f_{i-1}/s_i$. An associated right eigenvector is

$$v_i = [\,0, \ldots, s_{i-1} - \pi(i), -\pi(i), \ldots, -\pi(i)\,]^T \tag{64}$$

$$= [\,0, \ldots, s_{i-1}, 0, \ldots, 0\,]^T - [\,0, \ldots, \pi(i), \pi(i), \ldots, \pi(i)\,]^T \tag{65}$$

where there are $i-1$ leading zero elements in the vector. These eigenvectors (along with $v_0 = [1, \ldots, 1]^T$ with eigenvalue 1) are orthogonal with respect to an inner product based on π, with $v_i^T D v_j^T = 0$ for $i \neq j$, where D is the diagonal matrix with π on the diagonal. If at some step, $i < m$, we find that $\pi(i) \geq s_i$, we can identify an eigenvalue of $\lambda_i = -f_{i-1} s_i / \pi(i)$, with the same eigenvector v_i as above. Since the submatrix of rows and columns after i will be zero, all remaining eigenvalues (to λ_{m-1}) are zero.[9]

These eigenvalues (apart from the single eigenvalue of one) are all negative, except that some are zero when $\pi(\sigma(i)) \geq s_i$ for some i. This provides an alternative proof, via Corollary 15 of (Neal and Rosenthal 2023), that NAM methods efficiency-dominate Gibbs sampling, in addition to the general proof of this for AM methods given in Section 5.

In isolation, the negative eigenvalues of the modified NAM transition matrix introduce an element of "antithetic" sampling, reducing asymptotic variance of estimates, while slowing convergence to π, since the absolute values of the eigenvalues (other than the single one) are greater than for Gibbs sampling. However, when the transitions are used to update single variables, rather than the entire state, the modification will not necessarily lead to slower convergence than Gibbs sampling with random selection of variable to update — that will depend on the eigenvalues of the full transition matrix for an update of a randomly selected variable, which are not zero for Gibbs sampling.

Different orders of focal values for NAM may produce different transition probabilities, so different ways of choosing an order produce different methods for modifying Gibbs sampling. I will use "NAM" without a prefix to refer to a method in which the order of focal values is fixed. I next discuss a method in which focal values are chosen to have non-decreasing probability, as in Figure 3. This order may be different for each Gibbs sampling update, as changes to other variables change the conditional distribution of the variable updated. This will be followed by discussion of the opposite strategy, of focusing on values in non-increasing order of probability, which can have rather different properties.

---

[9] Proof that v_i is an eigenvector of P^*, with eigenvalue as given above: First, $[P^* v_i]_j$ is zero for $j < i$ since it equals

$$s_{i-1} P^*(j \to i) - \sum_{k=i}^{m} \pi(i) P^*(j \to k) = s_{i-1} \pi(i) \frac{f_{j-1}}{s_j} - \pi(i) \sum_{k=i}^{m} \pi(k) \frac{f_{j-1}}{s_j} = s_{i-1} \pi(i) \frac{f_{j-1}}{s_j} - \pi(i) s_{i-1} \frac{f_{j-1}}{s_j} = 0$$

When i is less than any k for which $\pi(k) \geq s_k$, then for any $j > i$, $[P^* v_i]_j$ equals

$$s_{i-1} P^*(j \to i) - \sum_{k=i}^{m} \pi(i) P^*(j \to k) = s_{i-1} \pi(i) \frac{f_{i-1}}{s_i} - \pi(i) f_{i-1} = \pi(i) \frac{f_{i-1}}{s_i} (s_{i-1} - s_i) = \left[-\pi(i) \frac{f_{i-1}}{s_i}\right] \left[-\pi(i)\right]$$

consistent with an eigenvalue of $-\pi(i) f_{i-1}/s_i$. Finally, when $\pi(i) < s_i$, $[P^* v_i]_i$ equals

$$-\sum_{k=i+1}^{m} \pi(i) P^*(i \to k) = -\pi(i) \sum_{k=i+1}^{m} \pi(k) \frac{f_{i-1}}{s_i} = -\pi(i) f_{i-1} = \left[-\pi(i) \frac{f_{i-1}}{s_i}\right] \left[s_{i-1} - \pi(i)\right]$$

again consistent with the eigenvalue $-\pi(i) f_{i-1}/s_i$. When $\pi(i) \geq s_i$, the eigenvalue is $-f_{i-1} s_i / \pi(i)$, since $[P^* v_i]_i$ equals

$$\left(s_{i-1} - \pi(i)\right) P^*(i \to i) - \sum_{k=i+1}^{m} \pi(i) P^*(i \to k) = \left(s_{i-1} - \pi(i)\right) \frac{\pi(i) - s_i}{\pi(i)} f_{i-1} - \sum_{k=i+1}^{m} \pi(i) \frac{\pi(k)}{\pi(i)} f_{i-1}$$

$$= \frac{f_{i-1}}{\pi(i)} \left[s_{i-1}(\pi(i) - s_i) + \pi(i) s_i - \pi(i) \sum_{k=i}^{m} \pi(k)\right] = \frac{f_{i-1}}{\pi(i)} \left[s_{i-1} \pi(i) - s_{i-1} s_i + \pi(i) s_i - \pi(i) s_{i-1}\right] = \left[-\frac{f_{i-1} s_i}{\pi(i)}\right] \left[s_{i-1} - \pi(i)\right]$$

and when $j > i$, $[P^* v_i]_j$ equals $\left(s_{i-1} - \pi(i)\right) P^*(j \to i) = \left(s_{i-1} - \pi(i)\right) f_{i-1} = s_i f_{i-1} = \left[-\frac{f_{i-1} s_i}{\pi(i)}\right] \left[-\pi(i)\right]$.

7 The Upward Nested Antithetic Modification (UNAM) method

In the example of Figure 3, the focal values used (1, 2, and 3) are in increasing order of probability: $\pi(1)=1/10 \;<\; \pi(2)=2/10 \;<\; \pi(3)=3/10$, with the final value having the largest probability, $\pi(4)=4/10$. I will refer to the NAM method in which focal values are chosen in non-decreasing order of probability as the Upwards Nested Antithetic Modification (UNAM) method.

This method is not new. It is equivalent to a method discussed by Frigessi, Hwang, and Younes (1992), and later devised independently by Tjelmeland (2004). As these authors note, and I will discuss below, the UNAM method will always produce a modified transition probability matrix that Peskun-dominates the original matrix, and hence efficiency-dominates it — i.e., produces estimates with lower asymptotic variance. As discussed in Sections 3, this Peskun-dominance and efficiency-dominance for updates of individual variables will carry over to an overall method that randomly selects a variable to update. NAM methods do not in general produce transitions that Peskun-dominate Gibbs sampling, as can be seen for the example of Figure 4, but as discussed for antithetic modifications in general in Section 5, they always efficiency-dominate Gibbs sampling, and this also carries over to an overall method that randomly selects a variable to update.

Unless an ordering by probability is already known, the UNAM method will start by finding a permutation, σ, of the possible values that orders them in non-decreasing probability, so that $\pi(\sigma(i)) \le \pi(\sigma(j))$ when $i \le j$. (In the example of Figure 3, values are already ordered by probability, so $\sigma(i) = i$.) Various sorting algorithms could be used to find this ordering. With m possible values, this can be done in time proportional to $m \log m$ using a comparison sort, or in time linear in m if a radix sort is used.

Once a suitable sorted order, σ, has been found, UNAM can be implemented by just applying Algorithm 2 with that σ. However, when σ puts values in non-decreasing order of probability, this algorithm can be simplified, as shown in Algorithm 3. In particular, within the loop, it is never possible for $\pi(a_i)$, which is q in the program, to be greater than or equal to s_i. As discussed above regarding Algorithm 2, it is possible to modify Algorithm 3 to return a sampled value rather than transition probabilities.

It is useful to see, from looking at the update to f in the loop of Algorithm 3, that for $i < k$,

$$f_i \;=\; f_{i-1} \;-\; f_{i-1}\,\frac{\pi(\sigma(i))}{s_i} \;=\; f_{i-1}\left(1 \;-\; \frac{\pi(\sigma(i))}{s_i}\right) \;=\; f_{i-1}\,\frac{s_i \;-\; \pi(\sigma(i))}{s_i} \tag{66}$$

To show that the UNAM method never decreases non-self transition probabilities, it suffices to show that the transition probability from $\sigma(j)$ to $\sigma(i)$, with $i < j$, never decreases, since reversibility then guarantees the same for the transition probability from $\sigma(i)$ to $\sigma(j)$. When Algorithm 3 is applied with $k = \sigma(j)$, this will be so if $f_{i-1} \ge s_i$ for all $i < j$ (see the entries below the diagonal in (62) above). Using $s_0 = 1$, $f_0 = 1$, and the update of equation (37), we can first see that $f_0 \ge s_1$, and then, using (66), that if $f_{i-2} \ge s_{i-1}$,

$$\begin{aligned}
f_{i-1} \;&=\; f_{i-2}\,\frac{s_{i-1} \;-\; \pi(\sigma(i-1))}{s_{i-1}} && (67)\\
&\ge\; s_{i-1}\,\frac{s_{i-1} \;-\; \pi(\sigma(i-1))}{s_{i-1}} && (68)\\
&=\; s_{i-1} - \pi(\sigma(i-1)) && (69)\\
&\ge\; s_{i-1} - \pi(\sigma(i)) \;=\; s_i && (70)
\end{aligned}$$

where the second inequality is because σ orders values by non-decreasing probability. It follows that

Input: Gibbs sampling probabilities, $\pi(i)$, for $i = 1, \ldots, m$
The current state value, k, in $\{1, \ldots, m\}$

Output: UNAM transition probabilities from k, as $p(i)$, for $i = 1, \ldots, m$

Set σ to some permutation on $\{1, \ldots, m\}$ for which $\pi(\sigma(i)) \leq \pi(\sigma(j))$ when $i \leq j$

Set s to 1 *The sum of probabilities for values that have not yet been focal*

Set f to 1 *The sum of transition probabilities from k to values that have not yet been focal*

Find modified transition probabilities from the current value to successive focal values, until the focal value is the current value

Set i to 1

While $\sigma(i) \neq k$:

Let q be the probability of the focal value; update s to be the sum of probabilities for remaining non-focal values

Set q to $\pi(\sigma(i))$
Subtract q from s *Sets variable s to s_i*

Compute the transition probability from current value, k, to the focal value, and find the new total probability for transitions to remaining values

Set $p(\sigma(i))$ to $\min(f, (q/s)f)$ *Min with f done in case $q > s$ due to rounding*
Subtract $p(\sigma(i))$ from f *Sets variable f to f_i, was previously f_{i-1}*

Add 1 to i

Compute modified transition probabilities from the current value, k, which is now focal, to values that have not previously been focal, as well as the self transition probability for k

If $i = m$:
Set $p(k)$ to f

Else:
Subtract $\pi(k)$ from s
Set $p(k)$ to 0
For $j = i{+}1, \ldots, m$: Set $p(\sigma(j))$ to $\min(f, (\pi(\sigma(j))/s)f)$ *Min guards against rounding*

Algorithm 3: Computation of UNAM transition probabilities, a simplification of Algorithm 2 when focal values have non-decreasing probability.

$f_{i-1} \geq s_i$ for all $i < j$, and hence UNAM never decreases non-self transition probabilities. Peskun's theorem therefore guarantees that estimates using UNAM have lower asymptotic variance than estimates using Gibbs sampling, when the variable to be updated is randomly selected.

UNAM transitions also Peskun-dominate those produced by MHGS. Again, we need only look at transition probabilities from $\sigma(j) = k$ to $\sigma(i)$ with $i < j$, for which $\pi(\sigma(i)) \leq \pi(k)$. For such a transition, the MHGS transition probability, from equation (13), is $\pi(\sigma(i)) \,/\, (1 - \pi(\sigma(i)))$. From equation (43), we see that the UNAM transition probabilities will be at least as large as the MHGS transition probabilities if $\pi(\sigma(i)) f_{i-1}/s_i \geq \pi(\sigma(i)) \,/\, (1 - \pi(\sigma(i)))$, as will be the case if $f_{i-1} \geq s_i \,/\, (1 - \pi(\sigma(i)))$, for all $i < j$. This holds for $i = 1$, since $f_0 = 1$, and from (37), $s_1 = s_0 - \pi(\sigma(1)) = 1 - \pi(\sigma(1))$. Furthermore, if

$f_{i-2} \geq s_{i-1} \,/\, (1-\pi(\sigma(i-1)))$, then again using equation (66, we have

$$
\begin{aligned}
f_{i-1} &= f_{i-2}\, \frac{s_{i-1} - \pi(\sigma(i-1))}{s_{i-1}} && (71)\\
&\geq \frac{s_{i-1}}{1 - \pi(\sigma(i-1))}\, \frac{s_{i-1} - \pi(\sigma(i-1))}{s_{i-1}} && (72)\\
&= \frac{s_{i-1} - \pi(\sigma(i-1))}{1 - \pi(\sigma(i-1))} && (73)\\
&\geq \frac{s_{i-1} - \pi(\sigma(i))}{1 - \pi(\sigma(i))} = \frac{s_i}{1 - \pi(\sigma(i))} && (74)
\end{aligned}
$$

The second inequality follows from $\pi(\sigma(i)) \geq \pi(\sigma(i-1))$ and the fact that if $0 \leq \delta \leq A \leq B$ then $A/B \geq (A-\delta)/(B-\delta)$. So the UNAM non-self transition probabilities are at least as great as those using MHGS, and hence Peskun's theorem implies that with random selection of variable to update, UNAM leads to lower asymptotic variance than MHGS.

If $\pi(\sigma(i)) = \pi(\sigma(i+1))$, then for any $k = \sigma(j)$ with $j > i+1$, Algorithm 3 will produce the same transition probabilities from k to $\sigma(i)$ and from k to $\sigma(i+1)$. To see this, note that in iteration i of the loop, $p(\sigma(i))$ will be set to $\pi(\sigma(i)) f_{i-1}/s_i$, and in the next iteration, $p(\sigma(i+1))$ will be set to

$$
\pi(\sigma(i+1))\, \frac{f_i}{s_{i+1}} = \pi(\sigma(i))\, \frac{f_i}{s_{i+1}} = \pi(\sigma(i))\, f_{i-1}\, \frac{s_i - \pi(\sigma(i))}{s_i}\, \frac{1}{s_i - \pi(\sigma(i+1))} = \pi(\sigma(i))\, \frac{f_{i-1}}{s_i} \quad (75)
$$

which is the same. Furthermore, as for all NAM methods, the transition probabilities to any $\sigma(i)$ from all the $\sigma(j)$ with $j > i$ are the same. Accordingly, when two or more values have equal probability, it makes no difference in what order σ places them.

Algorithm 3 sets all UNAM self transition probabilities to zero, except possibly that for $\sigma(m)$, which will be zero if $\pi(\sigma(m-1)) = \pi(\sigma(m))$ but not otherwise. To see this, note that this self transition probability will be f_{m-1}, which from equation (66) has the factor $(s_{m-1} - \pi(\sigma(m-1))) \,/\, s_{m-1}$, and since $s_{m-1} = \pi(\sigma(m))$, this is zero when $\pi(\sigma(m-1)) = \pi(\sigma(m))$.

As discussed earlier, the MHGS method can be applied when the number of possible values is countably infinite, provided these have a tractable form. Doing this seems much harder for the UNAM method, since Algorithm 3 looks at the possible values starting from the least probable, and so would take an infinite number of steps. Some hope for using UNAM with a countably infinite (or very large) number of possible values comes from reversing the recursions in equations (37) and (66):

$$
\begin{aligned}
f_{m-1} &= P^*(\sigma(m) \to \sigma(m)), \qquad f_{i-1} = f_i\, \frac{s_i}{s_i - \pi(\sigma(i))} && (76)\\
s_{m-1} &= \pi(\sigma(m)), \qquad s_{i-1} = s_i + \pi(\sigma(i)) && (77)
\end{aligned}
$$

After defining these recursions for a finite m, one might find the limiting form as m goes to infinity. One could then sample from the transition distribution computing only finitely many of the f_i, as necessary. Unfortunately, it is not clear how to compute $f_m = P^*(\sigma(m) \to \sigma(m))$ without looking at all m values, but perhaps this is tractable for some distributions.[10] If we can sample from π (i.e., the Gibbs sampling conditional probabilities), we could apply rejection sampling, using our knowledge of the *relative* transition

[10] If it happens that $\pi(\sigma(m-1)) = \pi(\sigma(m))$, we know that $P^*(\sigma(m) \to \sigma(m)) = 0$, but then the recursion from f_{m-1} to f_{m-2} is undefined, and we have a problem computing f_{m-2}.

$$\begin{bmatrix} \frac{1}{10} & \frac{2}{10} & \frac{3}{10} & \frac{4}{10} \\ \frac{1}{10} & \frac{2}{10} & \frac{3}{10} & \frac{4}{10} \\ \frac{1}{10} & \frac{2}{10} & \frac{3}{10} & \frac{4}{10} \\ \frac{1}{10} & \frac{2}{10} & \frac{3}{10} & \frac{4}{10} \end{bmatrix} \xrightarrow[\times\frac{10}{9}]{} \begin{bmatrix} \mathbf{0} & \mathbf{\frac{2}{9}} & \mathbf{\frac{3}{9}} & \mathbf{\frac{4}{9}} \\ \mathbf{\frac{1}{9}} & \mathbf{\frac{17}{90}} & \frac{3}{10} & \frac{4}{10} \\ \mathbf{\frac{1}{9}} & \frac{2}{10} & \mathbf{\frac{26}{90}} & \frac{4}{10} \\ \mathbf{\frac{1}{9}} & \frac{2}{10} & \frac{3}{10} & \mathbf{\frac{35}{90}} \end{bmatrix} \xrightarrow[\times\frac{80}{63}]{} \begin{bmatrix} 0 & \frac{2}{9} & \frac{3}{9} & \frac{4}{9} \\ \frac{1}{9} & \mathbf{0} & \mathbf{\frac{24}{63}} & \mathbf{\frac{32}{63}} \\ \frac{1}{9} & \mathbf{\frac{16}{63}} & \mathbf{\frac{148}{630}} & \frac{4}{10} \\ \frac{1}{9} & \mathbf{\frac{16}{63}} & \frac{3}{10} & \mathbf{\frac{211}{630}} \end{bmatrix} \xrightarrow[\times\frac{100}{63}]{} \begin{bmatrix} 0 & \frac{2}{9} & \frac{3}{9} & \frac{4}{9} \\ \frac{1}{9} & 0 & \frac{24}{63} & \frac{32}{63} \\ \frac{1}{9} & \frac{16}{63} & \mathbf{0} & \mathbf{\frac{40}{63}} \\ \frac{1}{9} & \frac{16}{63} & \mathbf{\frac{30}{63}} & \mathbf{\frac{10}{63}} \end{bmatrix}$$

Figure 5: Modification of Gibbs sampling transition probabilities using the procedure of Frigessi, Hwang, and Younes (1992), equivalent to UNAM. The probabilities altered at each stage are shown in bold. The factors by which non-self transition probabilities in the current row and column are multiplied are below the arrows. At each stage, self transition probabilities are altered so that the probabilities in a row sum to one. Note that the final result is the same as obtained with UNAM, as shown in Figure 3.

probabilities from the current state to the other states, which we can get from these recursions, *except* when the current state is the most probable, for which we would need to know the self transition probability.

The UNAM method gives the same transition probabilities as the method that is implicit in the statement and proof of Theorem 1 of Frigessi, Hwang, and Younes (1992). They note that their method can be applied to the probabilities for a Gibbs sampling update of a randomly-selected state variable, and that the Peskun-dominance of the individual updates extends to this scenario. They also found eigenvalues and eigenvectors of their transition matrices, which I presented above for the more general class of NAM methods.

Like my description of UNAM here, the procedure of Frigessi, *et al.*, illustrated in Figure 5, focuses on values in order of non-decreasing probability, alters transition probabilities to and from each such focal value in turn, and then proceeds to apply the procedure to the sub-matrix of remaining values. However, in their description, the values in the sub-matrix are not rescaled by a common factor in order to keep the row sums equal to one, as happens in the UNAM procedure — instead, self transition probabilities in the sub-matrix are reduced to keep the sum of transition probabilities equal to one, which is always possible when the focal values are in non-decreasing order of probability. The final result is the same as for UNAM.

One feature of this procedure is that modified non-self transition probabilities at every stage (not just the final stage) are at least as large as the Gibbs sampling probabilities, and hence these intermediate transition probabilities Peskun-dominate Gibbs sampling. Indeed, Frigessi, *et al.* consider in detail (on pages 624 and 626–627) only a simplified form of their method, in which only the first stage of modifications is performed, involving the least-probable state (except that when several states have the smallest probability, they modify the transition probabilities for all of them). In this regard, they remark (page 627),

> In the definition of the modified Gibbs sampler, we did not complete all the procedure described in part (b) of Theorem 1, for two reasons: The first is that we are not sure that, from any configuration which is not a local minimum of the energy, this new Markov chain would reach a bottom with positive probability. Our proof cannot be extended to show that this new stochastic matrix has no eigenvalue -1 at temperature 0. The second reason is practical: Each new step of the procedure of Theorem 1(b) would involve more and more computational cost. We therefore restrict ourselves to only one step, which is easy to implement.

Their first reason is particular to applications that aim essentially at optimization rather than sampling. Their second reason has some validity, since if only one stage of the procedure is done, one needn't sort the possible values by probability, but only find the state(s) of lowest probability. But the $m \log m$ sorting cost is not prohibitive in the typical case where all m probabilities must be computed in any case. Perhaps they did not realize that only the m probabilities for transitions from the current state need be computed, as in

Algorithm 3, rather than all m^2 transition probabilities. They give a recursive formula for the eigenvalues (page 617, Remark 4), which might have led them to an efficient simulation procedure, but they do not exploit its computational possibilities.[11]

Frigessi, *et al.* also show their method, when when applied to the entire state (not necessarily when used to sample individual variables as in Gibbs sampling), minimizes the maximum asymptotic variance of the estimated expectation of a function, maximizing over all functions with variance one (under π). This is of limited practical relevance, however, since the worst-case function will be proportional to the indicator function of the least likely state, which is seldom of interest.

A method equivalent to UNAM is also described by Tjelmeland (2004). Tjelmeland was apparently unaware of the work of Frigessi, *et al.*, perhaps since the title and abstract of the paper by Frigessi, *et al.* give little indication that it describes a general method for improving Gibbs sampling, and as noted above, Frigessi, *et al.* are dismissive of the utility of the full method. The title and abstract of Tjelmeland's paper also do not mention that it contains a general-purpose improvement to Gibbs sampling, focusing instead on a particular context involving multiple proposals. The method is described as "Transition alternative 2" on page 5.

The presentation of Tjelmeland's method is somewhat similar to that of Frigessi, *et al.*, but differs in several respects. As described mathematically, it alters the entire sub-matrix at each stage, rather than only the row and column involving the focal value, and the diagonal. The end result is the same, however.[12]

Following the presentation of the method, Tjelmeland remarks that

> The above process defines all elements in $\mathbf{P}(\mathbf{y})$. When simulating the Markov chain one of course only needs the elements in row κ. These can easily be computed without computing the whole matrix $\mathbf{P}(\mathbf{y})$. This is computationally important if m is large.

Tjelmeland gives no details, however. Avoiding such unnecessary computation of the full transition matrix is the point of Algorithm 3 for UNAM, as well as the more general NAM method of Algorithm 2.

Yet another path to a method equivalent to UNAM is mentioned by Pollet, Rombouts, Van Houcke, and Heyde (2004). The Metropolis-Hastings modification of Gibbs sampling probabilities that define the MHGS method can be generalized to modify any set of reversible transition probabilities, $P(u \to v)$. We use a proposal distribution, Q, that gives zero probability to the current state, rescaling $P(u \to v)$ for $v \neq u$ to sum to one:

$$Q(u \to v) \;=\; \frac{P(u \to v)}{1 - P(u \to u)} \tag{78}$$

The acceptance probability for such an update will be

$$\min\left(1,\; \frac{\pi(v)\, Q(v \to u)}{\pi(u)\, Q(u \to v)}\right) \;=\; \min\left(1,\; \frac{\pi(v)\, P(v \to u)\, (1 - P(u \to u))}{\pi(u)\, P(u \to v)\, (1 - P(v \to v))}\right) \;=\; \min\left(1,\; \frac{1 - P(u \to u)}{1 - P(v \to v)}\right) \tag{79}$$

using the fact that $\pi(v)\, P(v \to u) \;=\; \pi(u)\, P(u \to v)$ due to the reversibility of P.

[11] Note that for $i < k$, $P^*(\sigma(k) \to \sigma(i)) = \pi(\sigma(i)) f_{i-1}/s_i = -\lambda_i$, and $P^*(\sigma(i) \to \sigma(k)) = -\lambda_i \pi(\sigma(k))/\pi(\sigma(i))$, so knowing the eigenvalues allows efficient computation of transition probabilities.

[12] The equivalence is easier to see after simplifying Tjelmeland's equation (14), that defines a factor for multiplying transition probabilities:

$$u^t \;=\; \min_{k \in A^t} \left(\frac{1 - \sum_{l \notin A^t} P^t_{k,l}(y)}{\sum_{l \in A^t \setminus \{k\}} P^t_{k,l}(y)} \right)$$

where A_t is the set of states with non-zero self transition probabilities. The numerator in the fraction here is the same for all k, from which it follows that the minimum is for the k with minimum value for $P^t_{k,k}$.

The modified non-self transition probabilities will therefore be

$$\text{when } u \neq v, \quad P^*(u \to v) \;=\; \min\left(\frac{P(u \to v)}{1-P(u \to u)},\; \frac{P(u \to v)}{1-P(v \to v)}\right) \tag{80}$$

with the self transition probabilities determined by probabilities summing to one.[13]

Since these modified transition probabilities are themselves reversible (as for any Metropolis-Hastings method), the procedure can be repeated as many times as desired. $P^*(u \to v)$ will equal $P(u \to v)$ if the self transition probability of either u or v is zero, while otherwise $P^*(u \to v)$ will be greater than $P(u \to v)$. Hence repetition of this procedure asymptotically converges to a transition matrix with at most one non-zero self transition probability.

The same result is obtained with at most m repetitions if only the sub-matrix with non-zero self transition probabilities is updated (scaling it to have rows that sum to one, applying the Metropolis-Hastings procedure, and then scaling it back). The value with the smallest self transition probability will have zero self transition probability after this modification, so all but at most one self transition probability will be zero after $m-1$ applications of the procedure.

The results of these Metropolis-Hasting procedures, and of the methods of Frigessi, *et al.* and of Tjelmeland, are the same as the result obtained by the UNAM method. This is a consequence of three characteristics that they share. First, all these methods produce transition probabilities that are reversible with respect to π. Second, they ultimately set all self transition probabilities to zero, except perhaps for the most probable value. Third, for all methods, the modified transition probabilities $P^*(\sigma(i) \to \sigma(j))$ with $j > i$ are equal to $\pi(\sigma(j))$ times a factor that depends only on i, not on j, which due to reversibility implies also that the modified transition probabilities $P^*(\sigma(i) \to \sigma(j))$ with $j < i$ are equal to $\pi(\sigma(i))$ times a factor that depends only on j, not on i (so elements in a column below the diagonal are all the same, as seen in (62) for example). For UNAM, this can be seen from the last line of Algorithm 3. For the methods that repeatedly apply a Metropolis-Hasting modification, this is a consequence of equation (80), along with the fact that at each stage values ordered by σ have non-decreasing self transition probability, which is true for the initial GS transition probabilities, and is maintained by each MH update.[14] These characteristics determine a unique final result, once all self transition probabilities, apart perhaps for $\sigma(m)$, are zero.[15] All these methods must therefore produce the the same final result as the UNAM method.

---

[13] Pollet, *et al.* (2004) give an incorrect expression for $P^*(u \to v)$ (in their notation, T'_{ij}) on the bottom left of page 2, but this appears to be what they intended.

[14] Let P be transition probabilities before the MH update of (80), and P^* the transition probabilities after this update. Let $\sigma(i)$ and $\sigma(i+1)$ be consecutive focal values, with $\pi(\sigma(i)) \leq \pi(\sigma(i+1))$. Let $R = \pi(\sigma(i))/\pi(\sigma(i+1))$, and define $s_0 = P(\sigma(i) \to \sigma(i))$, $s_1 = P(\sigma(i{+}1) \to \sigma(i{+}1))$, $A = \sum_{k<i} P(\sigma(i) \to \sigma(k)) = \sum_{k<i} P(\sigma(i{+}1) \to \sigma(k))$, $b_0 = P(\sigma(i) \to \sigma(i{+}1))$, $b_1 = P(\sigma(i{+}1) \to \sigma(i)) = Rb_0$, $C_0 = \sum_{k>i+1} P(\sigma(i) \to \sigma(k))$, $C_1 = \sum_{k>i+1} P(\sigma(i{+}1) \to \sigma(k))$, and define s_0^*, s_1^*, A^*, b_0^*, b_1^*, C_0^*, and C_1^* analogously for P^* rather than P. We wish to show that if $s_0 \leq s_1$, then $s_0^* \leq s_1^*$. We have that $C_0 = 1-A-b_0-s_0$, $C_1 = 1-A-b_1-s_1$, $s_0^* = 1-A^*-b_0^*-C_0^*$ and $s_1^* = 1-A^*-b_1^*-C_1^*$. Since $b_0^* = b_0/(1-s_0)$, $b_1^* = Rb_0^* = Rb_0/(1-s_0)$, $C_0^* = C_0/(1-s_0)$, and $C_1^* = C_1/(1-s_1)$, we have

$$s_0^* \;=\; 1 \;-\; A^* \;-\; \frac{b_0}{1-s_0} \;-\; \frac{1-A-b_0-s_0}{1-s_0} \;=\; -A^* \;+\; \frac{A}{1-s_0}$$

$$s_1^* \;=\; 1 \;-\; A^* \;-\; \frac{Rb_0}{1-s_0} - \frac{1-A-Rb_0-s_1}{1-s_1} \;=\; -A^* \;+\; Rb_0\left(\frac{1}{1-s_1} - \frac{1}{1-s_0}\right) \;+\; \frac{A}{1-s_1}$$

Since $s_1 \geq s_0$, we see that the middle term in the expression for s_1^* is non-negative, and the final term is at least as large as the final term in the expression for s_0^*, and hence $s_1^* \geq s_0^*$.

[15] To see this, let h_i for $i = 1, \ldots, m{-}1$ be the factors that are used to multiply $\pi(\sigma(j))$ to get $P^*(\sigma(i) \to \sigma(j))$ for $j > i$, which due to reversibility also determine $P^*(\sigma(j) \to \sigma(i))$, and let g be the self transition probability for $\sigma(m)$. The requirement that transition probabilities sum to one leads to m linear equations in g and the h_i, which uniquely determine them.

8 The Downward Nested Antithetic Modification (DNAM) method

The Nested Antithetic Modification approach can also be applied with values ordered by non-increasing probability, giving the Downward Nested Antithetic Modification (DNAM) method. DNAM sometimes leads to smaller self transition probabilities than UNAM. With order reversed from UNAM, there is no guarantee that all non-self transition probabilities with DNAM are at least as large as with Gibbs sampling, so Peskun's theorem does not apply, but as discussed in Section 5, the transition probabilities produced with DNAM nevertheless efficiency-dominate Gibbs sampling.

DNAM can be implemented by simply applying the NAM procedure of Algorithm 2, passing it a σ that orders values by non-increasing probability. However, finding this order by sorting values according to probability can be avoided when the current value has probability of $1/2$ or more, as shown in Algorithm 4. DNAM sometimes produces transition probabilities that are zero past some point in the downward ordering. The algorithm could be modified to efficiently skip these zero probabilities, as discussed in Section 6.

Some examples of transition matrices obtained using UNAM are shown in Figure 6, with comparison to UNAM and Gibbs sampling.

In example (a), both UNAM and DNAM have a single non-zero self transition probability — the value with largest probability for UNAM, one of those with second-smallest probability for DNAM. Unlike UNAM, DNAM can treat values with the same probability (here, $\pi(2) = \pi(3)$) in substantively different ways, so it matters how the sorting algorithm used to produce σ handles ties. Note that in this example, the non-zero self transition probability is smaller for DNAM than for UNAM, but the reverse is also possible.

In example (b), both UNAM and DNAM produce self transition probabilities that are all zero. This happens with UNAM when the two largest probabilities under π are equal. It happens with DNAM when some value has a probability under π equal to the sum of probabilities of values later in the order σ. In this example, this happens because the second-last value in the order σ has the same probability as the last value. Note that although both UNAM and DNAM produce zero self transition probabilities, the other transition probabilities differ for the two methods.

Example (c) shows that UNAM can produce all zero self transition probabilities while DNAM does not. Example (d) shows the reverse, and also shows that with DNAM a large sub-matrix of transition probabilities may be all zero, a property that can sometimes be exploited to reduce computational cost.

Since neither UNAM nor DNAM is clearly superior to the other in all situations, one might consider randomly choosing between them, with equal probabilities, hoping to obtain the advantages of both. I call this method UDNAM. The transition probabilities for this method are simply the averages of those for UNAM and those for DNAM. Theorem 11 of (Neal and Rosenthal 2023) can be applied (twice) to show that since UNAM and DNAM both efficiency-dominate Gibbs sampling, UDNAM must also efficiency-dominate Gibbs sampling — UDNAM, as the random combination of UNAM and DNAM, must efficiency-dominate the random combination of UNAM and GS, which must efficiency-dominate the random combination of GS and GS, which is simply GS.

As noted earlier, Algorithm 2 used in DNAM can be modified to sample a value from the transition distribution, taking time proportional only to the index of this sampled value in the order σ, rather than computing all probabilities. Since DNAM looks at probabilities in decreasing order, this permits its use when the number of possible values is countably infinite, provided a formula for probabilities of values is available, and a non-increasing ordering can be determined.

For the the geometric(θ) distribution of equation (15), used as an example for MHGS, when m goes to

Input: Gibbs sampling probabilities, $\pi(i)$, for $i = 1, \ldots, m$
The current state value, k, in $\{1, \ldots, m\}$

Output: DNAM transition probabilities, $p(i)$, for $i = 1, \ldots, m$

If $\pi(k) \geq 1/2$:

Quickly handle the case where the current value has probability half or more, without needing to order values by probability

For $i = 1, \ldots, m$:

If $i \neq k$:

Set $p(i)$ to $\min(1, \pi(i)/\pi(k))$ *Min guards against round-off error*

Set $p(k)$ to $(2\pi(k) - 1) \,/\, \pi(k)$

Else:

Set σ to some permutation on $\{1, \ldots, m\}$ for which $\pi(\sigma(i)) \geq \pi(\sigma(j))$ when $i \leq j$

Set p to the output of the NAM procedure of Algorithm 2 with inputs π, σ, and k

Algorithm 4: Computation of DNAM transition probabilities.

	(a)	(b)	(c)	(d)
GS	$\begin{bmatrix} \frac{1}{12} & \frac{3}{12} & \frac{3}{12} & \frac{5}{12} \\ \frac{1}{12} & \frac{3}{12} & \frac{3}{12} & \frac{5}{12} \\ \frac{1}{12} & \frac{3}{12} & \frac{3}{12} & \frac{5}{12} \\ \frac{1}{12} & \frac{3}{12} & \frac{3}{12} & \frac{5}{12} \end{bmatrix}$	$\begin{bmatrix} \frac{2}{10} & \frac{2}{10} & \frac{3}{10} & \frac{3}{10} \\ \frac{2}{10} & \frac{2}{10} & \frac{3}{10} & \frac{3}{10} \\ \frac{2}{10} & \frac{2}{10} & \frac{3}{10} & \frac{3}{10} \\ \frac{2}{10} & \frac{2}{10} & \frac{3}{10} & \frac{3}{10} \end{bmatrix}$	$\begin{bmatrix} \frac{1}{10} & \frac{3}{10} & \frac{3}{10} & \frac{3}{10} \\ \frac{1}{10} & \frac{3}{10} & \frac{3}{10} & \frac{3}{10} \\ \frac{1}{10} & \frac{3}{10} & \frac{3}{10} & \frac{3}{10} \\ \frac{1}{10} & \frac{3}{10} & \frac{3}{10} & \frac{3}{10} \end{bmatrix}$	$\begin{bmatrix} \frac{1}{10} & \frac{1}{10} & \frac{3}{10} & \frac{5}{10} \\ \frac{1}{10} & \frac{1}{10} & \frac{3}{10} & \frac{5}{10} \\ \frac{1}{10} & \frac{1}{10} & \frac{3}{10} & \frac{5}{10} \\ \frac{1}{10} & \frac{1}{10} & \frac{3}{10} & \frac{5}{10} \end{bmatrix}$
UNAM	$\begin{bmatrix} 0 & \frac{3}{11} & \frac{3}{11} & \frac{5}{11} \\ \frac{1}{11} & 0 & \frac{15}{44} & \frac{25}{44} \\ \frac{1}{11} & \frac{15}{44} & 0 & \frac{25}{44} \\ \frac{1}{11} & \frac{15}{44} & \frac{15}{44} & \frac{10}{44} \end{bmatrix}$	$\begin{bmatrix} 0 & \frac{2}{8} & \frac{3}{8} & \frac{3}{8} \\ \frac{2}{8} & 0 & \frac{3}{8} & \frac{3}{8} \\ \frac{2}{8} & \frac{2}{8} & 0 & \frac{1}{2} \\ \frac{2}{8} & \frac{2}{8} & \frac{1}{2} & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & \frac{3}{9} & \frac{3}{9} & \frac{3}{9} \\ \frac{1}{9} & 0 & \frac{4}{9} & \frac{4}{9} \\ \frac{1}{9} & \frac{4}{9} & 0 & \frac{4}{9} \\ \frac{1}{9} & \frac{4}{9} & \frac{4}{9} & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & \frac{1}{9} & \frac{3}{9} & \frac{5}{9} \\ \frac{1}{9} & 0 & \frac{3}{9} & \frac{5}{9} \\ \frac{1}{9} & \frac{1}{9} & 0 & \frac{7}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{21}{45} & \frac{14}{45} \end{bmatrix}$
DNAM	$\begin{bmatrix} 0 & \frac{3}{42} & \frac{3}{14} & \frac{5}{7} \\ \frac{1}{42} & \frac{2}{42} & \frac{3}{14} & \frac{5}{7} \\ \frac{1}{14} & \frac{3}{14} & 0 & \frac{5}{7} \\ \frac{1}{7} & \frac{3}{7} & \frac{3}{7} & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & \frac{1}{7} & \frac{3}{7} & \frac{3}{7} \\ \frac{1}{7} & 0 & \frac{3}{7} & \frac{3}{7} \\ \frac{2}{7} & \frac{2}{7} & 0 & \frac{3}{7} \\ \frac{2}{7} & \frac{2}{7} & \frac{3}{7} & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & \frac{3}{21} & \frac{3}{7} & \frac{3}{7} \\ \frac{1}{21} & \frac{2}{21} & \frac{3}{7} & \frac{3}{7} \\ \frac{1}{7} & \frac{3}{7} & 0 & \frac{3}{7} \\ \frac{1}{7} & \frac{3}{7} & \frac{3}{7} & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ \frac{1}{5} & \frac{1}{5} & \frac{3}{5} & 0 \end{bmatrix}$
π	$\begin{matrix} \frac{1}{12} & \frac{3}{12} & \frac{3}{12} & \frac{5}{12} \end{matrix}$	$\begin{matrix} \frac{2}{10} & \frac{2}{10} & \frac{3}{10} & \frac{3}{10} \end{matrix}$	$\begin{matrix} \frac{1}{10} & \frac{3}{10} & \frac{3}{10} & \frac{3}{10} \end{matrix}$	$\begin{matrix} \frac{1}{10} & \frac{1}{10} & \frac{3}{10} & \frac{5}{10} \end{matrix}$

Figure 6: Some comparisons of transitions probabilities for Gibbs Sampling (GS), UNAM, and DNAM. For all examples, values are ordered by non-decreasing probability, so UNAM focuses on values as ordered, and DNAM focuses on values in the reverse order.

infinity, $\pi(i) = \theta(1-\theta)^{i-1}$ and $s_i = \sum_{j>i} \pi(j) = (1-\theta)^i$. The decreasing ordering is $\sigma(i) = i$). When $\theta \geq 1/2$, we will have $\pi(1) \geq 1/2$, so the DNAM transition probabilities computed by Algorithm 4 will be

$$P^*(1 \to 1) = (2\pi(1) - 1) \,/\, \pi(1) = (2\theta - 1) \,/\, \theta \tag{81}$$

$$P^*(1 \to j) = (1-\theta)^{j-1} \text{ for } j > 1 \tag{82}$$

$$P^*(i \to 1) = 1, \quad \text{for } i > 1 \tag{83}$$

$$P^*(i \to j) = 0, \quad \text{for } i, j > 1 \tag{84}$$

When $\theta < 1/2$, we will never have $\pi(i) \geq s_i$, so the DNAM transition probabilities will follow the pattern of (62), giving:[16]

$$P^*(i \to j) = \frac{\theta}{1-\theta} \left(\frac{1-2\theta}{1-\theta} \right)^{j-1}, \quad \text{for } j < i \tag{87}$$

$$P^*(i \to i) = 0 \tag{88}$$

$$P^*(i \to j) = \frac{\theta(1-2\theta)^{i-1}}{(1-\theta)^{2i-1}} (1-\theta)^{j-1} \quad \text{for } j > i \tag{89}$$

So for this geometric distribution, the DNAM method produces the minimum possible self transition probability. The transition distributions are piecewise geometric, and so are easily sampled from.

Finally, note that self transition probabilities that are all zero can sometimes be obtained with NAM using an ordering that is neither upward (as in UNAM) nor downward (as in DNAM). For example (a) of Figure 6, we can obtain the transition probability matrices below by using the ordering 1,4,2,3 (or 1,4,3,2), shown on the left, and by using the ordering 4,1,2,3 (or 4,1,3,2), shown on the right:

$$\begin{bmatrix} 0 & \frac{3}{11} & \frac{3}{11} & \frac{5}{11} \\ \frac{1}{11} & 0 & \frac{5}{33} & \frac{25}{33} \\ \frac{1}{11} & \frac{5}{33} & 0 & \frac{25}{33} \\ \frac{1}{11} & \frac{15}{33} & \frac{15}{33} & 0 \end{bmatrix} \qquad \begin{bmatrix} 0 & \frac{3}{21} & \frac{3}{21} & \frac{5}{7} \\ \frac{1}{21} & 0 & \frac{5}{21} & \frac{5}{7} \\ \frac{1}{21} & \frac{5}{21} & 0 & \frac{5}{7} \\ \frac{1}{7} & \frac{3}{7} & \frac{3}{7} & 0 \end{bmatrix} \tag{90}$$

However, although a transition matrix having self transition probabilities that are all zero always exists when no value has probability greater than 1/2, such a transition matrix cannot generally be obtained using NAM with some ordering — this is possible only when there is exact equality between the probability of some value and a sum of probabilities of some other values. In the next three sections, I will discuss methods that do always minimize self transition probabilities.

---

[16] For $j < i$, we will have

$$P^*(i \to j) = \pi(j) \frac{f_{j-1}}{s_j} = \frac{\theta(1-\theta)^{j-1}}{(1-\theta)^j} f_{j-1} = \frac{\theta}{1-\theta} f_{j-1} \tag{85}$$

So then, $f_j = f_{j-1} - P^*(i \to j) = f_{j-1} - \frac{\theta}{1-\theta} f_{j-1} = \frac{1-2\theta}{1-\theta} f_{j-1}$, from which it follows that $f_j = \left(\frac{1-2\theta}{1-\theta} \right)^j$, and hence that $P^*(i \to j) = \frac{\theta}{1-\theta} \left(\frac{1-2\theta}{1-\theta} \right)^{j-1}$.

For $j > i$,

$$P^*(i \to j) = \pi(j) \frac{f_{i-1}}{s_i} = \frac{\theta(1-\theta)^{j-1}}{(1-\theta)^i} \left(\frac{1-2\theta}{1-\theta} \right)^{i-1} = \frac{\theta(1-2\theta)^{i-1}}{(1-\theta)^{2i-1}} (1-\theta)^{j-1} \tag{86}$$

9 The Zero-self DNAM (ZDNAM) method

A non-zero self transition probability is necessary only for a value whose probability under π is more than one half. But DNAM will produce a non-zero self transition probability for a value with probability less than one half if this probability is greater than the sum of the probabilities of values with lower probability, as is the case in examples (a) and (c) of Figure 6. Note that when this happens the transition probabilities among the remaining values are all zero, so the DNAM procedure ends at this point.

Here, I describe a modified procedure, the Zero-self DNAM method (ZDNAM), which modifies the DNAM procedure to operate differently at the step just before the one where DNAM would produce a non-zero self transition probability, substituting transition probabilities that avoid this. As for DNAM, the remaining transition probabilities are all zero, so no further steps are necessary.

The idea can be illustrated by an example with $m = 5$ and $\sigma(i) = i$, with $\pi(1) = 6/18$, $\pi(2) = 5/18$, $\pi(3) = 4/18$, $\pi(4) = 2/18$, and $\pi(5) = 1/18$. DNAM modifies the original transitions as follows:

$$\begin{bmatrix} \frac{6}{18} & \frac{5}{18} & \frac{4}{18} & \frac{2}{18} & \frac{1}{18} \\ \frac{6}{18} & \frac{5}{18} & \frac{4}{18} & \frac{2}{18} & \frac{1}{18} \\ \frac{6}{18} & \frac{5}{18} & \frac{4}{18} & \frac{2}{18} & \frac{1}{18} \\ \frac{6}{18} & \frac{5}{18} & \frac{4}{18} & \frac{2}{18} & \frac{1}{18} \\ \frac{6}{18} & \frac{5}{18} & \frac{4}{18} & \frac{2}{18} & \frac{1}{18} \end{bmatrix} \rightarrow \begin{bmatrix} 0 & \frac{5}{12} & \frac{4}{12} & \frac{2}{12} & \frac{1}{12} \\ \frac{6}{12} & \frac{5}{36} & \frac{4}{36} & \frac{2}{36} & \frac{1}{36} \\ \frac{6}{12} & \frac{5}{36} & \frac{4}{36} & \frac{2}{36} & \frac{1}{36} \\ \frac{6}{12} & \frac{5}{36} & \frac{4}{36} & \frac{2}{36} & \frac{1}{36} \\ \frac{6}{12} & \frac{5}{36} & \frac{4}{36} & \frac{2}{36} & \frac{1}{36} \end{bmatrix} \rightarrow \begin{bmatrix} 0 & \frac{5}{12} & \frac{4}{12} & \frac{2}{12} & \frac{1}{12} \\ \frac{6}{12} & 0 & \frac{4}{14} & \frac{2}{14} & \frac{1}{14} \\ \frac{6}{12} & \frac{5}{14} & \frac{4}{49} & \frac{2}{49} & \frac{1}{49} \\ \frac{6}{12} & \frac{5}{14} & \frac{4}{49} & \frac{2}{49} & \frac{1}{49} \\ \frac{6}{12} & \frac{5}{14} & \frac{4}{49} & \frac{2}{49} & \frac{1}{49} \end{bmatrix} \rightarrow \begin{bmatrix} 0 & \frac{5}{12} & \frac{4}{12} & \frac{2}{12} & \frac{1}{12} \\ \frac{6}{12} & 0 & \frac{4}{14} & \frac{2}{14} & \frac{1}{14} \\ \frac{6}{12} & \frac{5}{14} & \frac{1}{28} & \frac{2}{28} & \frac{1}{28} \\ \frac{6}{12} & \frac{5}{14} & \frac{4}{28} & 0 & 0 \\ \frac{6}{12} & \frac{5}{14} & \frac{4}{28} & 0 & 0 \end{bmatrix}$$

The non-zero self transition probability of $P^*(3 \rightarrow 3) = 1/28$ results from $\pi(3) = 4/18$ being greater than the sum of probabilities for later values, which in this example is $s_3 = \pi(4) + \pi(5) = 3/18$.

For this example, the ZDNAM method operates the same as DNAM for the first step, but at step $i = 2$, the ZDNAM algorithm recognizes that $\pi(\sigma(i+1)) > s_{i+1} = \sum_{j>i+1} \pi(\sigma(j))$ — in this example, that $\pi(3) = 4/18 \ > \ \pi(4) + \pi(5) = 3/18$ — and employs a special construction to avoid a non-zero self transition probability for $\sigma(i+1)$ — in this example, for the value 3. The result is as follows:

$$\begin{bmatrix} \frac{6}{18} & \frac{5}{18} & \frac{4}{18} & \frac{2}{18} & \frac{1}{18} \\ \frac{6}{18} & \frac{5}{18} & \frac{4}{18} & \frac{2}{18} & \frac{1}{18} \\ \frac{6}{18} & \frac{5}{18} & \frac{4}{18} & \frac{2}{18} & \frac{1}{18} \\ \frac{6}{18} & \frac{5}{18} & \frac{4}{18} & \frac{2}{18} & \frac{1}{18} \\ \frac{6}{18} & \frac{5}{18} & \frac{4}{18} & \frac{2}{18} & \frac{1}{18} \end{bmatrix} \rightarrow \begin{bmatrix} 0 & \frac{5}{12} & \frac{4}{12} & \frac{2}{12} & \frac{1}{12} \\ \frac{6}{12} & \frac{5}{36} & \frac{4}{36} & \frac{2}{36} & \frac{1}{36} \\ \frac{6}{12} & \frac{5}{36} & \frac{4}{36} & \frac{2}{36} & \frac{1}{36} \\ \frac{6}{12} & \frac{5}{36} & \frac{4}{36} & \frac{2}{36} & \frac{1}{36} \\ \frac{6}{12} & \frac{5}{36} & \frac{4}{36} & \frac{2}{36} & \frac{1}{36} \end{bmatrix} \rightarrow \begin{bmatrix} 0 & \frac{5}{12} & \frac{4}{12} & \frac{2}{12} & \frac{1}{12} \\ \frac{6}{12} & 0 & \frac{12}{40} & \frac{4}{30} & \frac{2}{30} \\ \frac{6}{12} & \frac{15}{40} & 0 & \frac{2}{24} & \frac{1}{24} \\ \frac{6}{12} & \frac{10}{30} & \frac{4}{24} & 0 & 0 \\ \frac{6}{12} & \frac{10}{30} & \frac{4}{24} & 0 & 0 \end{bmatrix} \tag{91}$$

This special operation is uniquely determined by the requirements that the result be reversible with respect to π, that it not alter transition probabilities to or from $\sigma(j)$ for $j < i$ that were found in previous steps, that transition probabilities among the $\sigma(j)$ with $j > i+1$ be zero, and that transition probabilities to $\sigma(j)$ for $j > i+1$ from both $\sigma(i)$ and $\sigma(i+1)$ be proportional to $\pi(\sigma(j))$.

The derivation of the general scheme can be illustrated with reference to the NAM transition matrix shown in (63), which represents the result of DNAM when $m = 5$, $\sigma(i) = i$, and a non-zero self transition probability is produced at step 3. At step 2, the ZDNAM method will alter the matrix produced so that

it instead has the following form:

$$P^* \;=\; \left[\begin{array}{ccccc} 0 & \pi(2)\,\dfrac{f_0}{s_1} & \pi(3)\,\dfrac{f_0}{s_1} & \pi(4)\,\dfrac{f_0}{s_1} & \pi(5)\,\dfrac{f_0}{s_1} \\ \pi(1)\,\dfrac{f_0}{s_1} & 0 & \dfrac{1}{\pi(2)}Af_1 & \dfrac{\pi(4)}{\pi(2)}Bf_1 & \dfrac{\pi(5)}{\pi(2)}Bf_1 \\ \pi(1)\,\dfrac{f_0}{s_1} & \dfrac{1}{\pi(3)}Af_1 & 0 & \dfrac{\pi(4)}{\pi(3)}Cf_1 & \dfrac{\pi(5)}{\pi(3)}Cf_1 \\ \pi(1)\,\dfrac{f_0}{s_1} & Bf_1 & Cf_1 & 0 & 0 \\ \pi(1)\,\dfrac{f_0}{s_1} & Bf_1 & Cf_1 & 0 & 0 \end{array}\right] \tag{92}$$

This transition matrix is reversible with respect to π by construction. A, B, and C can be found from the requirement that the rows sum to one.

I will now switch to using a general notation, with i being the step at which ZDNAM recognizes that $\pi(\sigma(i{+}1)) > s_{i+1}$, and hence the special construction is needed. The example above has $i = 2$ and $\sigma(i) = i$. Recall that s_i is the sum of $\pi(\sigma(j))$ for all $j > i$, and that for any $k > i$, f_i is the sum of transition probabilities from $\sigma(k)$ to $\sigma(j)$ for all $j > i$.

When finding A, B, and C, the requirement that rows of the matrix sum to one is equivalent to requiring that for $k \geq i$, the sum of $P^*(\sigma(k) \to \sigma(j))$ for $j \geq i$ must be f_{i-1}. This gives the following equations:

$$A \;+\; Bs_{i+1} \;=\; \pi(\sigma(i)), \quad A \;+\; Cs_{i+1} \;=\; \pi(\sigma(i{+}1)), \quad B \;+\; C \;=\; 1 \tag{93}$$

Solving this system of equations, we get

$$A = \frac{\pi(\sigma(i)) + \pi(\sigma(i{+}1)) - s_{i+1}}{2},\; B = \frac{\pi(\sigma(i)) - \pi(\sigma(i{+}1)) + s_{i+1}}{2\,s_{i+1}},\; C = \frac{s_{i+1} + \pi(\sigma(i{+}1)) - \pi(\sigma(i))}{2\,s_{i+1}} \tag{94}$$

Algorithm 5 implements this procedure. As in Algorithm 4 for DNAM, it starts by handling the case where the current value has probability 1/2 or more specially, which avoids the need to sort by probability. The case where the most-probable value has probability 1/2 or more is also handled specially. Otherwise, the DNAM procedure is applied for i from 1 on up, until the current value is reached in the ordering found, while also checking whether the next step, $i{+}1$, will be one in which $\pi(\sigma(i{+}1)) \;\geq\; s_{i+1}$, and hence the special construction will be used. Because of this forward check, no check for whether $\pi(\sigma(i)) \;\geq\; s_i$ is needed within the loop.

If the special construction is needed, the values A, B, and C of (94) are computed and used, taking care to avoid division by zero.

As is the case for other NAM methods, the ZDNAM algorithm computes transition probabilities sequentially, and hence can easily be modified to sample a value from the transition distribution based on a uniform random variate, terminating once the cumulative probability exceeds the uniform variate. The possibilities for handling distributions with a countably infinite number of values are similar to DNAM.

The reduction in self transition probability for ZDNAM compared to DNAM is not uniformly beneficial — it is not always the case that the ZDNAM transition matrix efficiency-dominates the DNAM transition

Input: Gibbs sampling probabilities, $\pi(i)$, for $i = 1, \ldots, m$
The current state value, k, in $\{1, \ldots, m\}$

Output: ZDNAM transition probabilities, $p(i)$, for $i = 1, \ldots, m$

If $\pi(k) \geq 1/2$:

Quickly handle the case where the current value has probability half or more, without needing to order values by probability

For $i = 1, \ldots, m$:
If $i \neq k$:
Set $p(i)$ to $\min(1, \pi(i)/\pi(k))$ *Min guards against round-off error*

Set $p(k)$ to $(2\pi(k) - 1) \,/\, \pi(k)$

Else:

Set σ to some permutation on $\{1, \ldots, m\}$ for which $\pi(\sigma(i)) \geq \pi(\sigma(j))$ when $i \leq j$

If $\pi(\sigma(1)) \geq 1/2$:

Handle the case where a value has probability of 1/2 or more. Won't be the current value, since that's handled above.

Set $p(\sigma(1))$ to 1

For $i = 2, \ldots, m$:
Set $p(\sigma(i))$ to 0

Else:

Set s to 1 *The sum of probabilities for values that have not yet been focal*

Set f to 1 *The sum of transition probabilities from k to values not yet focal*

Find modified transition probabilities from the current value to successive focal values, until the focal value is the current value, or special handling to avoid a non-zero self transition probability is needed.

Set i to 1

While $f > 0$ and $\sigma(i) \neq k$ and $\pi(\sigma(i{+}1)) < s - \pi(\sigma(i)) - \pi(\sigma(i{+}1))$:

Let q be the probability of the focal value; update s to be the sum of probabilities for remaining non-focal values

Set q to $\pi(\sigma(i))$
Subtract q from s *Sets variable s to s_i, guaranteed positive*

Compute the transition probability from the current value, k, to the focal value, and find the new total probability for transitions to remaining values

Set $p(\sigma(i))$ to $(q/s)f$ *Guaranteed $p(\sigma(i)) \leq f \leq 1$, even with rounding*
Subtract $p(\sigma(i))$ from f *Sets variable f to f_i, was previously f_{i-1}*

Add 1 to i

Set q to $\pi(\sigma(i))$
Subtract q from s

Continue with the procedure of Algorithm 5: Part 2.

Algorithm 5: Part 1. Procedure for computing ZDNAM transition probabilities.

Continuation of Algorithm 5: Part 1.

If $f > 0$ and $s > 0$ and $i < m$:

Set q_2 to $\pi(\sigma(i+1))$
Set s_2 to $\max(0,\, s - q_2)$ *max guards against round-off error*

If $q_2 \geq s_2$:

Use the special construction to avoid a non-zero self transition probability.

Set A to $(q + q_2 - s_2) \,/\, 2$
If $k = \sigma(i)$:
Set $p(\sigma(i))$ to 0
Set $p(\sigma(i+1)$ to fA/q
Else If $k = \sigma(i+1)$:
Set $p(\sigma(i))$ to fA/q_2
Set $p(\sigma(i+1)$ to 0

If $s_2 \leq 0$:
Add 2 to i
Else:
Set B to $(q - q_2 + s_2) \,/\, (2s_2)$
Set C to $(s_2 + q_2 - q) \,/\, (2s_2)$
If $k = \sigma(i)$:
Add 2 to i
While $i \leq m$:
Set $p(\sigma(i))$ to $fB\pi(\sigma(i))/q$
Add 1 to i
Else If $k = \sigma(i+1)$:
Add 2 to i
While $i \leq m$:
Set $p(\sigma(i))$ to $fC\pi(\sigma(i))/q_2$
Add 1 to i
Else:
Set $p(\sigma(i))$ to fB
Set $p(\sigma(i+1))$ to fC
Add 2 to i

Else:

Compute modified transition probabilities from the current value, k, which is now focal, to values that have not previously been focal.

Set $p(\sigma(i))$ to 0
Add 1 to i
While $i \leq m$:
Set $p(\sigma(i))$ to $(\pi(\sigma(i)) \,/\, s)\, f$
Add 1 to i

Set any remaining transition probabilities to zero.

While $i \leq m$:
Set $p(\sigma(i))$ to 0
Add 1 to i

Algorithm 5: Part 2. Continuation of procedure for computing ZDNAM transition probabilities.

matrix. This can be seen, for example, when $m = 3$ and $\pi(1) = 4/9$, $\pi(2) = 3/9$, and $\pi(3) = 2/9$, for which

$$P^*_{\text{DNAM}} \;=\; \begin{bmatrix} 0 & \frac{9}{15} & \frac{6}{15} \\ \frac{12}{15} & \frac{1}{15} & \frac{2}{15} \\ \frac{12}{15} & \frac{3}{15} & 0 \end{bmatrix}, \qquad P^*_{\text{ZDNAM}} \;=\; \begin{bmatrix} 0 & \frac{15}{24} & \frac{9}{24} \\ \frac{20}{24} & 0 & \frac{4}{24} \\ \frac{18}{24} & \frac{6}{24} & 0 \end{bmatrix} \tag{95}$$

Numerical calculation finds that the eigenvalues of $P^*_{\text{DNAM}} - P^*_{\text{ZDNAM}}$ are 0.10306, −0.03639, and zero. Since their signs are mixed, Theorem 9 of (Neal and Rosenthal 2023) shows that neither P^*_{DNAM} nor P^*_{ZDNAM} efficiency-dominates the other.

A ZDNAM transition probability matrix has eigenvalues and eigenvectors that can be associated with each step followed when constructing it. Until the special construction is used, when $\pi(\sigma(i+1)) \geq s_{i+1}$, these are the same as for any NAM procedure, as described in Section 6 (e.g., equation (65)). Two eigenvalues and eigenvectors are associated with steps i and $i+1$ when the special construction is applied at step i. Assuming for notational simplicity that the non-increasing ordering is $\sigma(i) = i$, these two eigenvalues are given by

$$\lambda \;=\; -\frac{f_{i-1}}{2}\left[1 \;\pm\; \sqrt{1 \;-\; \big(\pi(i) - \pi(i{+}1) + s_{i+1}\big)\,\big(\pi(i{+}1)^2 - (\pi(i) - s_{i+1})^2\big) \,/\, \big(\pi(i)\pi(i{+}1)s_{i+1}\big)}\;\right] \tag{96}$$

An associated right eigenvector for such a λ is

$$v \;=\; [\,0, \ldots,\;\; Cf_{i-1}s_{i+1} + \lambda\pi(i{+}1),\;\; -Bf_{i-1}s_{i+1} - \lambda\pi(i),\;\; Bf_{i-1}\pi(i{+}1) - Cf_{i-1}\pi(i),\;\; \ldots]^T \tag{97}$$

where there are $i-1$ leading zero elements in the vector, and the elements after position $i+1$ are all the same.[17] The eigenvalues after those associated with steps i and $i+1$ are all zero.

The eigenvalues of a ZDNAM transition matrix are all zero or negative, apart from the one eigenvalue of 1 with eigenvector $[1, \ldots, 1]^T$. This is so for the eigenvalues associated with the NAM steps before step i,

[17] Here is the proof that either one of the λ of (96), which can be written as $\lambda = -(f_{i-1}/2)\big[1 \pm \sqrt{D}\,\big]$, in which $D = 1 - \big(\pi(i) - \pi(i{+}1) + s_{i+1}\big)\,\big(\pi(i{+}1)^2 - (\pi(i) - s_{i+1})^2\big) \,/\, \big(\pi(i)\pi(i{+}1)s_{i+1}\big)$, is an eigenvalue of the ZDNAM transitions P^* visualized in (92) with the corresponding v from (97) as an associated eigenvector.

We first show that $[P^*v]_j = 0$ for $j < i$:

$$\begin{aligned}
[P^*v]_j \;&=\; (Cf_{i-1}s_{i+1} + \lambda\pi(i{+}1))P^*(j \to i) - (Bf_{i-1}s_{i+1} + \lambda\pi(i))P^*(j \to i{+}1) + \sum_{k=i+2}^{m} (Bf_{i-1}\pi(i{+}1) - Cf_{i-1}\pi(i))P^*(j \to k) \\
&=\; \frac{f_{j-1}}{s_j}\Big[(Cf_{i-1}s_{i+1} + \lambda\pi(i{+}1))\,\pi(i) \;-\; (Bf_{i-1}s_{i+1} + \lambda\pi(i))\,\pi(i{+}1) \;+\; (Bf_{i-1}\pi(i{+}1) - Cf_{i-1}\pi(i))\,s_{i+1}\Big] \;=\; 0
\end{aligned}$$

Next, we see that

$$\begin{aligned}
[P^*v]_i \;&=\; -(Bf_{i-1}s_{i+1} + \lambda\pi(i))P^*(i \to i{+}1) \;+\; \sum_{k=i+2}^{m} (Bf_{i-1}\pi(i{+}1) - Cf_{i-1}\pi(i))P^*(i \to k) \\
&=\; \big(f_{i-1}/\pi(i)\big)\,\big(-(Bf_{i-1}s_{i+1} + \lambda\pi(i))\,A \;+\; (Bf_{i-1}\pi(i{+}1) - Cf_{i-1}\pi(i))\,Bs_{i+1}\big) \\
&=\; -\big(f_{i-1}^2/(2\pi(i))\big)\,\Big(\big((\pi(i){-}\pi(i{+}1){+}s_{i+1}) \;-\; \big[1 \pm \sqrt{D}\big]\pi(i)\big)A \;-\; (B\pi(i{+}1){-}C\pi(i))\,(\pi(i){-}\pi(i{+}1){+}s_{i+1})\Big) \\
&=\; -\big(f_{i-1}^2/4\big)\,\big((-\pi(i{+}1){+}s_{i+1})\,(\pi(i){+}\pi(i{+}1){-}s_{i+1})\,/\,\pi(i) \;\mp\; (\pi(i) + \pi(i{+}1) - s_{i+1})\sqrt{D} \\
&\qquad\qquad -\,((\pi(i) - \pi(i{+}1) + s_{i+1})\pi(i{+}1) - (s_{i+1} + \pi(i{+}1) - \pi(i))\pi(i))\,(\pi(i){-}\pi(i{+}1){+}s_{i+1})\,/\,(\pi(i)s_{i+1})\big) \\
&=\; -\big(f_{i-1}^2/4\big)\,\big(\pm\,(s_{i+1}{-}\pi(i){-}\pi(i{+}1))\sqrt{D} \;+\; (\pi(i)\pi(i{+}1)^2{+}\pi(i)^2\pi(i{+}1){+}\pi(i{+}1)s_{i+1}^2{+}\pi(i{+}1)^2s_{i+1} \\
&\qquad\qquad +2\pi(i)s_{i+1}^2{-}\pi(i)^3{-}\pi(i{+}1)^3{-}s_{i+1}^3{-}3\pi(i)\pi(i{+}1)s_{i+1})\,/\,(\pi(i)s_{i+1})\big) \\
&=\; -\big(f_{i-1}^2/4\big)\,\big[1 \pm \sqrt{D}\,\big]\,\big(s_{i+1} + \pi(i{+}1) - \pi(i) - \big[1 \pm \sqrt{D}\,\big]\pi(i{+}1)\big) \\
&=\; \lambda\,(Cf_{i-1}s_{i+1} + \lambda\pi(i{+}1))
\end{aligned}$$

In similar fashion, we have:

where the special construction is needed, as demonstrated in Section 6. The two eigenvalues given by (96) are also negative — the value of the square root is less than one, hence the quantity in square brackets is positive, and the eigenvalue is negative. To see that the square root is less than one, note first that $\pi(i) \geq \pi(i{+}1)$, since the ordering is non-increasing, and hence the factor $(\pi(i) - \pi(i{+}1) + s_{i+1})$ is positive. Also, $\pi(i) < \pi(i{+}1) + s_{i+1}$, since otherwise the special construction would have been used before step i, and $\pi(i{+}1) \geq s_{i+1}$, since the special construction was used at step i, and hence $\pi(i) \geq s_{i+1}$. It follows that $0 \leq \pi(i) - s_{i+1} < \pi(i{+}1)$, and hence the factor $(\pi(i{+}1)^2 - (\pi(i) - s_{i+1})^2)$ is positive.

Since the eigenvalues (apart from the single 1) are all zero or negative, Corollary 15 of (Neal and Rosenthal 2023) can then be applied to show that ZDNAM transitions efficiency-dominate Gibbs sampling. As discussed for antithetic modifications in Section 5, Theorem 12 of (Neal and Rosenthal 2023) allows us to then conclude that using ZDNAM to update a randomly selected variable efficiency-dominates using Gibbs sampling with such random updates.

10 The Shifted Tower (ST) and Half Shifted Tower (HST) methods

Suwa and Todo (2010) and Suwa (2022) describe a class of methods for defining transition probabilities that can be viewed in terms of building a "tower" of probabilities for values, applying a circular shift operation to produce a second tower, and then defining transition probabilities by the alignment of the first and second towers.

The first method, of Suwa and Todo (2010), shifts by the probability of the most probable value. I will refer to this as the Shifted Tower (ST) method. It always reduces self transitions to the minimum possible. Unlike all the methods considered previously in this paper, it may produce non-reversible transitions (though note that when there are only two possible values, transitions leaving π invariant are always reversible with respect to π). Suwa (2022) generalized this method to an arbitrary shift, and in particular noted that shifting by $1/2$ minimizes self transitions while also producing transitions that are reversible. I call this the Half Shifted Tower (HST) method.

The ST and HST methods are illustrated in Figure 7. Algorithm 6 implements these methods, for any

$$
\begin{aligned}
[P^* v]_{i+1} = \;& (C f_{i-1} s_{i+1} + \lambda \pi(i{+}1)) P^*(i{+}1 \to i) \;+\; \sum_{k=i+2}^{m} (B f_{i-1} \pi(i{+}1) - C f_{i-1} \pi(i)) P^*(i \to k) \\
=\;& \big(f_{i-1}/\pi(i{+}1)\big) \big((C f_{i-1} s_{i+1} + \lambda \pi(i{+}1))A \;+\; (B f_{i-1} \pi(i{+}1) - C f_{i-1} \pi(i))\, C s_{i+1}\big) \\
=\;& -\big(f_{i-1}^2/(2\pi(i{+}1))\big) \big((-(s_{i+1}{+}\pi(i{+}1){-}\pi(i)) + \big[1 \pm \sqrt{D}\,\big]\pi(i{+}1))A \;-\; (B\pi(i{+}1){-}C\pi(i))\,(s_{i+1}{+}\pi(i{+}1){-}\pi(i))\big) \\
=\;& -\big(f_{i-1}^2/4\big) \big((-s_{i+1}{+}\pi(i))\,(\pi(i){+}\pi(i{+}1) - s_{i+1})\,/\,\pi(i{+}1) \;\pm\; (\pi(i){+}\pi(i{+}1) - s_{i+1})\sqrt{D} \\
&\qquad - ((\pi(i) - \pi(i{+}1) + s_{i+1})\pi(i{+}1) - (s_{i+1} + \pi(i{+}1) - \pi(i))\pi(i))\,(s_{i+1}{+}\pi(i{+}1){-}\pi(i))\,/\,(\pi(i{+}1)s_{i+1})\big) \\
=\;& -\big(f_{i-1}^2/4\big) \big(\pm\; (\pi(i){+}\pi(i{+}1) - s_{i+1})\sqrt{D} \;+\; (\pi(i)\pi(i{+}1)^2 - \pi(i)^2\pi(i{+}1) - \pi(i)s_{i+1}^2 - \pi(i)^2 s_{i+1} \\
&\qquad\qquad - 2\pi(i{+}1)s_{i+1}^2 + \pi(i)^3 + \pi(i{+}1)^3 + s_{i+1}^3 + 3\pi(i)\pi(i{+}1)s_{i+1})\,/\,(\pi(i{+}1)s_{i+1})\big) \\
=\;& -\big(f_{i-1}^2/4\big)\,\big[1 \pm \sqrt{D}\,\big]\,\big(-\pi(i) + \pi(i{+}1) - s_{i+1} + \big[1 \pm \sqrt{D}\,\big]\pi(i)\big) \\
=\;& \lambda\,(-B f_{i-1} s_{i+1} - \lambda \pi(i))
\end{aligned}
$$

Finally, for $j > i + 1$, we have that

$$
\begin{aligned}
[P^* v]_j \;&=\; (C f_{i-1} s_{i+1} + \lambda \pi(i{+}1)) P^*(j \to i) \;-\; (B f_{i-1} s_{i+1} + \lambda \pi(i)) P^*(j \to i{+}1) \\
&=\; (C f_{i-1} s_{i+1} + \lambda \pi(i{+}1)) B f_{i-1} \;-\; (B f_{i-1} s_{i+1} + \lambda \pi(i)) C f_{i-1} \\
&=\; \lambda\,(B f_{i-1} \pi(i{+}1) - C f_{i-1} \pi(i))
\end{aligned}
$$

specified shift, and any ordering of values, using a formula adapted from one given by Suwa (2022).[18]

For a given shift amount, $s \in [0, 1]$, and ordering of values, σ, the formula computes the "flow" from value k to value i, defined by $v_{ki} = \pi(k)\, P^*(k \to i)$, as

$$\begin{aligned} v_{ki} \;\; &= \;\; \max\left(0,\, \min\left(\Delta_1,\, \pi(k) + \pi(i) - \Delta_1,\, \pi(k),\, \pi(i)\right)\right) \\ &\quad + \;\; \max\left(0,\, \min\left(\Delta_2,\, \pi(k) + \pi(i) - \Delta_2,\, \pi(k),\, \pi(i)\right)\right) \\ &\quad \text{where } \Delta_1 \;=\; \pi(k) - s + C_k - C_i \text{ and } \Delta_2 \;=\; \Delta_1 + 1 \end{aligned} \tag{98}$$

Here, C_k is the sum of probabilities for values before k in the ordering σ. We can compute these as follows:

$$C_{\sigma(i)} \;\; = \;\; \sum_{j=1}^{i-1} \pi(\sigma(j)) \tag{99}$$

Once v_{ki} has been computed, we can find the transition probability from k to i as

$$P^*(k \to i) \;\; = \;\; v_{ki} \,/\, \pi(k) \tag{100}$$

Figure 8 illustrates how the formula for v_{ki} of equation (98) is derived. The left of the figure shows a situation in which $v_{ki} = \Delta_1$, while the right shows a situation in which $v_{ki} = \pi(k) + \pi(i) - \Delta_1$. When the shifted region for value i completely encloses the original region for value k, the flow will be $\pi(k)$, and in the opposite situation, the flow will be $\pi(i)$. Taking the minimum of all these possibilities, and then replacing a negative value by zero, gives the flow in all situations where wrap-around is not an issue. To this, we need to add the value for the flow that is found accounting for the possibility that after shifting by s, the start of the region for value i wraps around from 1 to 0, which we do by replacing Δ_1 by $\Delta_2 = \pi(k) - s + C_k - (C_i - 1) = \Delta_1 + 1$. The final result is given by equation (98).

In Algorithm 6, this procedure is modified to avoid issues with round-off error. Rather than compute Δ_2 as $\Delta_1 + 1$, the program sets Δ_2 to $\Delta_1 + S$, where S is the sum of probabilities for all values. If the probabilities are normalized, one would expect this to be 1, but it may not be due to round-off error. Similarly, the transition probability $P^*(k \to i)$ is not found as $v_{ki}/\pi(k)$, but rather as $v_{ki} \,/\, \sum_j v_{kj}$, which guarantees that these transition probabilities are not greater than one even if $\sum_j v_{kj}$ is not exactly $\pi(k)$.

For both ST and HST, the ordering of values can matter. I will use ST and HST to refer to these methods with the original order retained. I use Ordered HST (OHST) to refer to HST with values ordered by probability — whether by non-increasing or non-decreasing probability makes no difference. I use Upward ST (UST) or Downward ST (DST) to refer to the ST method in which the most probable value is followed by the other values in non-decreasing or non-increasing order. For all these methods, how values with equal probability are ordered may matter.

Both UST and DST produce transition probabilities are (in general) non-reversible, but which are, however, reverses of each other — that is,

$$\pi(u)\, P_{UST}(u \to v) \;\; = \;\; \pi(v)\, P_{DST}(v \to u) \tag{101}$$

[18] Suwa's formula appears to erroneously treat the values as having the reverse of their specified order, comparing to Fig. 1 of Suwa (2022), though this has no practical effect if the ordering was arbitrary anyway. The formula used in Algorithm 6 corrects for this. Note that F_i in Suwa's formulas (12) and (13) corresponds to $C(i) + \pi(i)$ in the notation used here.

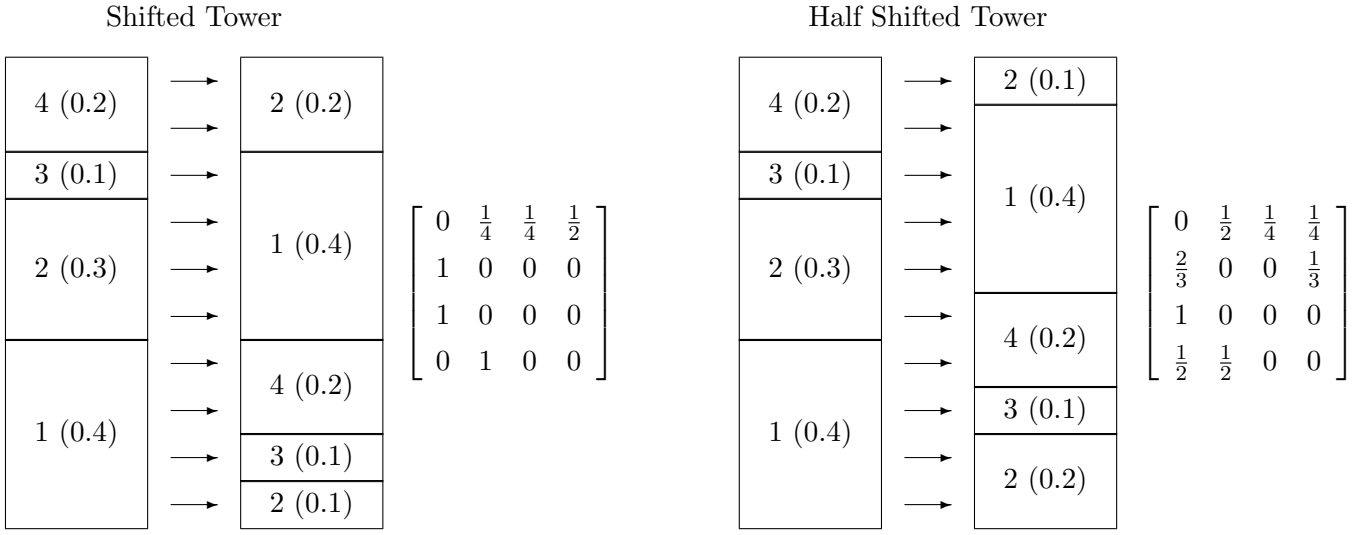


Figure 7: The Shifted Tower (ST) and Half Shifted Tower (HST) methods. In this example, values 1, 2, 3, and 4 have probabilities of 0.4, 0.3, 0.1, and 0.2. On the left for each method is the tower of regions for each value, with heights proportional to their probabilities. On the right of this tower is a shifted tower, with regions that move out of the top moving into the bottom, which may result in the region for a value being split between top and bottom. For the ST method, the shift is by the probability of the most probable symbol. For the HST method, the shift is always by 1/2. Transitions are defined by randomly sampling from the region of the left tower corresponding to the current value, then following the arrows right to a region of the shifted tower. The resulting matrices of transition probabilities are shown to the right.

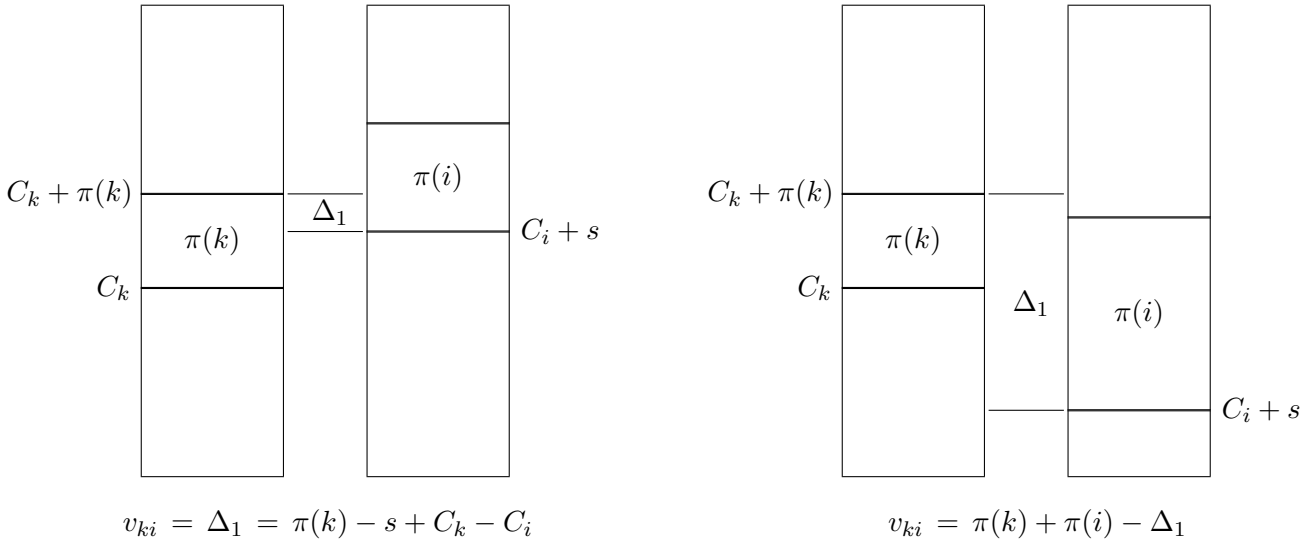


Figure 8: Illustration of how v_{ki} for the shifted tower method can be found in two situations.

Input: Gibbs sampling probabilities, $\pi(i)$, for $i = 1, \ldots, m$
The current state value, k, in $\{1, \ldots, m\}$
Amount of shift, s, in $(0, 1)$
A permutation, σ, on $\{1, \ldots, m\}$, giving an ordering of values

Output: ST transition probabilities, $p(i)$, for $i = 1, \ldots, m$

Temporary storage: Flows of probability, $v(i)$, from k to each value, for $i = 1, \ldots, m$
Cumulative probabilities, $C(i)$, for $i = 1, \ldots, m$, with $C(i) = \sum_{j=1}^{i-1} \pi(i)$

If $\pi(k) \geq 1/2$:

Quickly handle the case where the current value has probability half or more, without needing to compute cumulative probabilities.

For $i = 1, \ldots, m$:
If $i \neq k$:
Set $p(i)$ to $\min(1, \pi(i)/\pi(k))$ *Min guards against round-off error*
Set $p(k)$ to $(2\pi(k) - 1) \,/\, \pi(k)$

Else:

Compute cumulative probabilities, in the order given by σ, but stored in the original order. Set S to the sum of all probabilities, which should be one, but may differ due to rounding.

Set S to 0
For $i = 1, \ldots, m$:
Set $C(\sigma(i))$ to S
Add $\pi(\sigma(i))$ to S

Find the flows from the current value to each value, and the total flow.

Set t to 0
For $i = 1, \ldots, m$:
Set Δ_1 to $\pi(k) - s + C(k) - C(i)$ *Will be exactly zero if $i = k$ and $s = \pi(k)$*
Set Δ_2 to $\Delta_1 + S$
Set $v(i)$ to $\max(0, \min(\Delta_1, \pi(k) + \pi(i) - \Delta_1, \pi(k), \pi(i)))$
$+ \max(0, \min(\Delta_2, \pi(k) + \pi(i) - \Delta_2, \pi(k), \pi(i)))$
Add $v(i)$ to t

If $t = 0$:

If the total flow is zero, return a result giving probability 1 to the most probable value.

Set j to 1
For $i = 2, \ldots, m$:
If $\pi(i) > \pi(j)$:
Set j to i
For $i = 1, \ldots, m$:
Set $p(i)$ to 1 if $i = j$, otherwise to 0

Else:

Find transition probabilities by normalizing flows by their sum, which should be $\pi(k)$, but may differ due to rounding.

For $i = 1, \ldots, m$:
Set $p(i)$ to $v(i)/t$

Algorithm 6: Computation of ST transition probabilities.

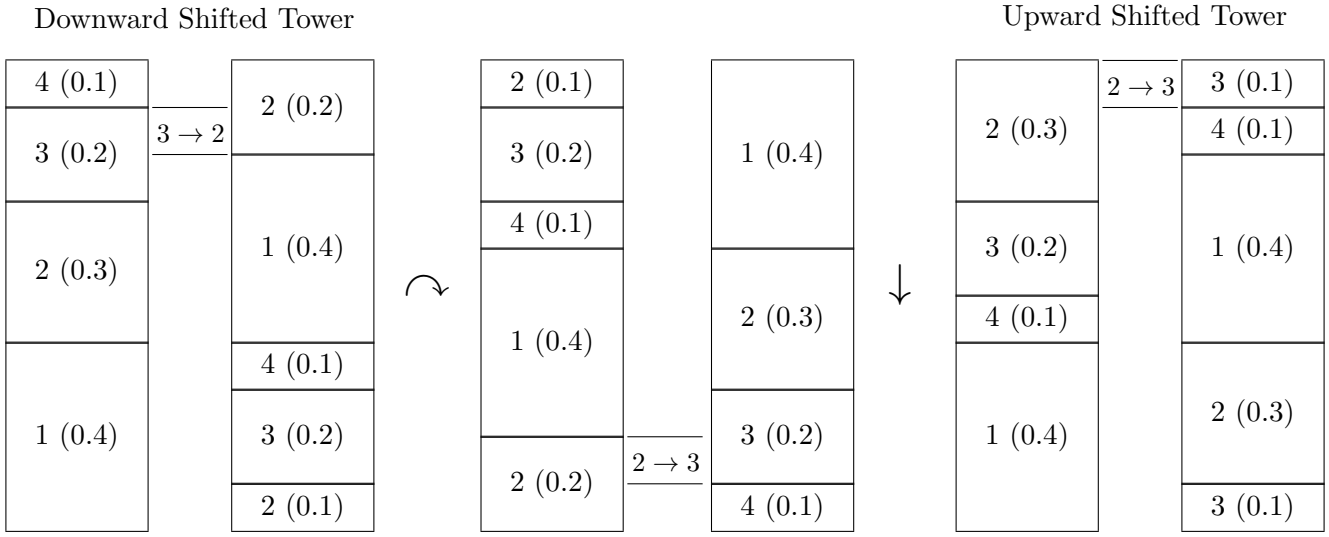


Figure 9: Illustration of why UST and DST are reversals of each other. On the left is an illustration of DST transition probabilities, showing in particular that $\pi(3)\,P_{DST}(3 \to 2) = 0.1$. In the middle is the result of rotating the diagram on the left by 180 degrees, which produces reversed transition probabilities, where in particular $\pi(2)\,P_{DST}(2 \to 3) = 0.1$. On the right is the result of shifting the two towers in the middle down by 0.2 (wrapping bottom to top). This shift of both towers has no effect on the transition probabilities, which are now seen to be those of P_{UST}.

This relationship is illustrated in Figure 9. Averaging the transition probabilities produced by UST and DST therefore gives a method, which I will call UDST, that is reversible:

$$\begin{aligned} \pi(u)\,P_{UDST}(u \to v) &= \pi(u)\,(P_{UST}(u \to v) + P_{DST}(u \to v))\,/\,2 && (102)\\ &= \pi(v)\,(P_{DST}(v \to u) + P_{UST}(v \to u))\,/\,2 && (103)\\ &= \pi(v)\,P_{UDST}(v \to u) && (104) \end{aligned}$$

Like UST and DST, UDST produces the minimum possible self transition probabilities, so it will provide interesting information on the effect of reversibility in the experimental comparisons.

Algorithm 6 also starts by checking whether the current value has probability of $1/2$ or more, and if so, finds the transition probabilities from this value quickly, without needing to compute cumulative probabilities. This check could be omitted, as might be desirable if it is known that probabilities of a half or more are unlikely. Also, when this check is omitted, it is not necessary for the input probabilities, π, to be normalized to sum to one, given the adjustments described in the previous paragraph, provided the shift amount, s, is on the same scale as these unnormalized probabilities. Indeed, the procedure described by Suwa (2022) does not assume that probabilities are normalized.

The ST method can be implemented by applying Algorithm 6 with s set to the maximum value of π. For the HST method, Algorithm 6 is called with s set to $1/2$.

In many contexts, computing transition probabilities is not necessary — all that is needed is a way of sampling from the transition distribution given the current state value. For ST methods, sampling directly may be significantly faster than first computing transition probabilities and then sampling using them. Algorithm 7 implements such a direct sampling method, based on randomly choosing a point within the region of the "tower" corresponding to the current value, then moving this point down by the shift amount,

Input: Gibbs sampling probabilities, $\pi(i)$, for $i = 1, \ldots, m$
The current state value, k, in $\{1, \ldots, m\}$
Amount of shift, s, in $(0, 1)$
A permutation, σ, on $\{1, \ldots, m\}$, giving an ordering of values

Output: A state value, j, sampled from the ST transition probabilities from state k, guaranteed not to be a value with transition probability zero

Find the sum, u, of probabilities of values before k in the ordering used.

Set i to 1
Set u to 0
While $\sigma(i) \neq k$:
    Set u to $u + \pi(\sigma(i))$
    Add 1 to i

Add a random amount to u, while subtracting the shift, with wrap-around.

Set r to a uniform random variate on $[0, 1]$
Add $r\pi(k) - s$ to u — *Guaranteed not to increase u when $s = \max_j \pi(j)$*
If $u \leq 0$:
    Add 1 to u — *Guarantees that u is greater than zero*

Use this value of u to pick a value, j, to transition to, picking an arbitrary value with non-zero probability if no value is chosen due to round-off error.

Set i to 0
Set s to 0
While $i < m$ and $u > s$:
    Add 1 to i
    If $\pi(\sigma(i)) > 0$:
        Add $\pi(\sigma(i))$ to s
        Set j to $\sigma(i)$

The value j is now a sample from the transition probabilities from the current state, k.

Algorithm 7: Sampling from ST transition probabilities.

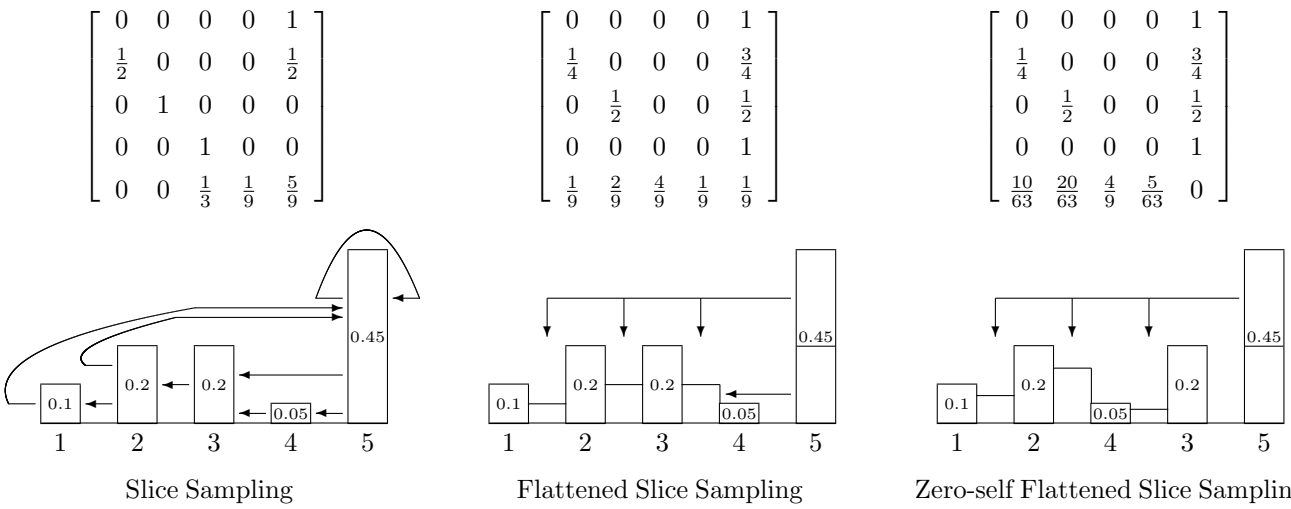
with wrap-around (equivalent to moving the tower up), and choosing a new value using this shifted point as if it were a random $[0, 1]$ variate.

This technique could be used when the state has a countably infinite number of values, as long as the cumulative distribution function and its inverse can be computed efficiently.

11 Flattened slice sampling methods (FSS and ZFSS)

Modified Gibbs sampling methods can also be derived using the "slice sampling" framework (Neal 2003). For discrete distributions, slice sampling can be visualized using bars associated with each possible value, with the height of a bar equal to its value's probability. A vertical level within the bar for the current value is sampled uniformly, and some update is then made that moves amongst the bars that intersect the horizontal line drawn at this level, with the property of leaving the uniform distribution on this horizontal "slice" invariant.

One update that seems promising for avoiding self transitions is to move from the bar for the current



Slice Sampling | Flattened Slice Sampling | Zero-self Flattened Slice Sampling

Figure 10: Illustration of SS, FSS, and ZFSS methods. These diagrams portray transitions that leave invariant the distribution on $\{1, 2, 3, 4, 5\}$ with probabilities 0.1, 0.2, 0.2,, 0.05, and 0.45. The left diagram shows simple slice sampling, in which a vertical position is randomly chosen from the bar for the current value, with height equal to its probability, and a movement to the left (with wrap-around) is then made until the next bar is encountered. In the middle diagram, the self transition probability for the most probable value is reduced by distributing the excess of its probability over that of the next-most probable value to new bars that follow the bars for values other than the most probable value and the one before it. Arrows showing the subsequent transitions are omitted, except for one left arrow showing that there is still a non-zero self transition probability, going from the most probable value to another bar also associated with this value. In the right diagram, value 3 is moved to just before the most-probable value, which blocks such a self transition. The resulting transition probability matrices are shown above the diagrams.

value, at the sampled level, to the next bar to the left that rises to that level, wrapping around to the right side if the left end is reached. This method is shown in the left illustration of Figure 10. Unfortunately, the method will produce a non-zero self transition probability if one value has a probability greater than all other values, as is the case for value 5 in Figure 10. If the vertical level sampled when this is the current value is greater than the probabilities of all other values, the movement to the left will wrap around to the same value. In this example, the resulting self transition probability from value 5 is $(0.45 - 0.2)/0.45 = 5/9$, but the minimum possible self transition probability for this distribution is zero.

This self transition probability can be reduced by distributing the portion of the probability of the most-probable symbol that is greater than all other symbols amongst another set of bars, which follow the bars for values other than the most probable value and the value to its left (with wraparound). This modification, called Flattened Slice Sampling (FSS), is shown in the middle illustration of Figure 10. The 0.25 excess probability for value 5 is moved to bars to the right of values 1, 2, and 3, in proportion to their probabilities. When 5 is the current value, a bar is selected from amongst these three new bars and the original bar with probabilities 0.05/0.45, 0.1/0.45, 0.1/0.45, and 0.2/0.45. Movement to the left then occurs as before. If the bar moved to is any of those associated with the most-probable value, that becomes the new state.

However, the self transition probability for FSS is still not zero in this example. The bar for value 4 is lower than the new bar to the right of value 3. Consequently, a portion of the bar for value 5 encounters this new bar, which is also associated with value 5, when leftward movement occurs, resulting in a self transition probability of 1/9 when value 5 is the current state.

The Zero-self Flattened Slice Sampling (ZFSS) method avoids such unnecessary self transitions by re-ordering values to put a value that blocks such movement immediately to the left of the most-probable value, while leaving the order of values otherwise unchanged. The value moved is the one closest on the left to the most-probable value that will block any resulting movement from the original bar for the most-probable value to one of the new bars also associated with this value. In the example of Figure 10, value 3 is moved to the left of value 5.

This procedure assumes all values have probability less than one half, which also implies that $m > 2$. Situations where a value has probability one half or more are handled specially, in the same manner as for ZDNAM and the ST methods, which, as will be discussed below in Section 13, is the only method that minimizes the probability of a self transition in this situation. The FSS and ZFSS methods are implemented in Algorithm 8, with an input flag specifying whether the possible re-ordering for ZFSS is done.

As for the ST methods, an FSS transition can be simulated directly more efficiently than it can be by first computing transition probabilities from the current value and then sampling a new value according to these probabilities. Since the flow is computed in Algorithm 8 by adding portions (with the flow never decreasing), one can keep track of the cumulative flow computed so far, and make a transition to the value associated with the portion just computed when this cumulative sum exceeds a random variate chosen at the beginning. Algorithm 9 implements this approach.

The FSS and ZFSS methods are non-reversible, whenever the maximum probability is less than one half. For FSS, this non-reversibility takes the form of consistent movement to the left (with wrap-around), except for possible transitions to the most-probable value. One might speculate that such consistent movement improves efficiency. The re-ordering that may be done for ZFSS is designed to disturb this leftward movement as little as possible. self transitions could instead be avoided by ordering the values by non-decreasing probability, but this would often disturb the original ordering more, and could lead to the ordering changing from one update to another, preventing consistent movement.

FSS and ZFSS are feasible for some distributions with a countably infinite number of values. Consider the geometric(θ) distribution on $\{1, 2, \ldots\}$ used previously as an example for MHGS and DNAM, with $\theta < 1/2$. If we use a reverse order, so that value 1 is rightmost, the excess in probability of the most probable value (1) over the next-most probable (2) will be $\theta - \theta(1-\theta) = \theta^2$, which will be distributed over new bars that follow values 3, 4, 5, etc. to the right, in proportion to the probabilities of these values. The height of the new bar to the right of value $i+1$ will be $\theta^2 \cdot \theta(1-\theta)^{i-2} = \theta^3(1-\theta)^{i-2}$. In comparison, the height of the bar for value i will be $\theta(1-\theta)^{i-1}$. The transition probability from value i (for $i > 1$) to value 1 will be the sum of the ratio of these, $\theta^2 \,/\, (1-\theta)$, plus the ratio of the excess of the probability for value i over that for value $i+1$ to the probability for value i, which is θ. This gives the transition probabilities from value i for $i > 1$ as

$$\begin{aligned} P^*(i \to 1) &= \theta^2 \,/\, (1-\theta) \;+\; \theta \;=\; \theta \,/\, (1-\theta) && (105) \\ P^*(i \to i+1) &= 1 \;-\; P^*(i \to 1) \;=\; 1 \;-\; \theta \,/\, (1-\theta) && (106) \\ P^*(i \to j) &= 0, \quad \text{for } j \neq 1 \text{ and } j \neq i+1 && (107) \end{aligned}$$

For value 1, we have

$$\begin{aligned} P^*(1 \to 1) &= 0 && (108) \\ P^*(1 \to 2) &= \theta(1-\theta) \,/\, \theta \;=\; 1-\theta && (109) \\ P^*(1 \to j) &= \theta^3(1-\theta)^{j-3} \,/\, \theta \;=\; \theta^2(1-\theta)^{j-3} \quad \text{for } j > 2 && (110) \end{aligned}$$

Input: Gibbs sampling probabilities, $\pi(i)$, for $i = 1, \ldots, m$
The current state value, k, in $\{1, \ldots, m\}$
A flag, ZERO, for whether ZFSS should be used

Output: FSS or ZFSS transition probabilities, $p(i)$, for $i = 1, \ldots, m$

Temporary storage: Flows of probability, $v(i)$, from k to each value, for $i = 1, \ldots, m$

If $\pi(k) \leq 0$:

Handle transition from zero-probability value specially.

For $i = 1, \ldots, m$:
Set $p(i)$ to $\pi(i)$

Else:

Find the index, x_1, of the most probable value, and its probability, π_1.

Set x_1 to 1
For $i = 2, \ldots, m$:
If $\pi(i) > \pi(x_1)$:
Set x_1 to i
Set π_1 to $\pi(x_1)$
If $\pi(x_1) \geq 1/2$ or $m \leq 2$: *Checking for $m \leq 2$ guards against round-off error*

Handle the case where the current value has probability half or more specially.

If $k = x_1$:
For $i = 1, \ldots, m$:
Set $p(i)$ to $(2\pi_1 - 1) \,/\, \pi_1$ if $i = k$, otherwise to $\pi(i)/\pi_1$
Else:
For $i = 1, \ldots, m$:
Set $p(i)$ to 1 if $i = x_1$, otherwise to 0

Else:

Find the probability, π_2, of the second most probable value.

Set π_2 to 0
for $i = 1, \ldots, m$:
If $i \neq x_1$ and $\pi(i) > \pi_2$:
Set π_2 to $\pi(i)$

Find the index, x_0, of the value before the most probable value, or for ZFSS, the index of the first value before the most probable value which will block movement beyond it from encountering a piece of the most probable value.

Set x_0 to x_1
Loop:
Set x_0 to m if $x_0 = 1$, otherwise to $x_0 - 1$
Set π^* to $(0.5 - \pi_1) + (0.5 - \pi(x_0))$ *Computing this way reduces round-off error*
Set f to $(\pi_1 - \pi_2) \,/\, \pi^*$ *Guaranteed to be in $[0, 1]$ even with rounding*
Repeat loop as long as ZERO and $\pi(x_0) < f\pi_2$

Continue with the procedure of Algorithm 8: Part 2.

Algorithm 8: Part 1. Procedure for computing FSS or ZFSS transition probabilities.

Continuation of Algorithm 8: Part 1.

Find the part of the flow due to distributing the difference in probability between most probable and second-most probable values among other values. Here, f is the factor to multiply probabilities of values besides x_1 and x_0 by to get the part of x_1 flowing there.

For $i = 1, \ldots, m$:
    If $k = x_1$ and $i \neq x_1$ and $i \neq x_0$:
        Set $v(i)$ to $f\pi(i)$
    Else:
        Set $v(i)$ to 0

Find the flow due to slice movement.

Set ℓ to 0 — *Lower end of probability region to move*
Set u to π_2 if $k = x_1$, otherwise to $\pi(k)$ — *Upper end of probability region to move*
Set i to k

While $\ell < u$:

Move i backwards, going from x_1 to x_0, from x_0 to before x_1, and skipping x_0 when otherwise going back.

If $i = x_1$:
    Set i to x_0
Else:
    If $i = x_0$:
        Set i to m if $x_1 = 1$, otherwise to $x_1 - 1$
    Else:
        Set i to m if $i = 1$, otherwise to $i - 1$
    If $i = x_0$:
        Set i to m if $x_0 = 1$, otherwise to $x_0 - 1$

Add to flow from slice movement of $[\ell, u]$ region, and update ℓ and u.

If $\ell < \pi(i)$:
    If $i \neq x_1$ and $i \neq x_0$:
        Set t to $\min(u, f\pi(i))$
        If $\ell < t$:
            Add $t - \ell$ to $v(x_1)$
            Set ℓ to t
    Set t to $\min(u, \pi(i))$
    Add $t - \ell$ to $v(i)$
    Set ℓ to t

Return transition probabilities derived from flow.

For $i = 1, \ldots, m$:
    Set $p(i)$ to $v(i)/\pi(k)$

Algorithm 8: Part 2. Continuation of procedure for computing FSS or ZFSS transition probabilities.

Input: Gibbs sampling probabilities, $\pi(i)$, for $i = 1, \ldots, m$
The current state value, k, in $\{1, \ldots, m\}$
A flag, ZERO, for whether ZFSS should be used

Output: A state value, j, sampled from the FSS/ZFSS transition probabilities from state k, guaranteed not to be a value with transition probability zero

If $\pi(k) \leq 0$:

Handle transition from zero-probability value specially.

Set r to a uniform random variate on $[0, 1]$
For $i = 1, \ldots, m$:
Set s to 0; Set i to 0
While $i < m$ and $r \geq s$:
Add 1 to i
If $\pi(i) > 0$:
Add $\pi(i)$ to s; Set j to i

Else:

Find the index, x_1, of the most probable value, and its probability, π_1.

Set x_1 to 1
For $i = 2, \ldots, m$:
If $\pi(i) > \pi(x_1)$:
Set x_1 to i
Set π_1 to $\pi(x_1)$
If $\pi(x_1) \geq 1/2$ or $m \leq 2$: *Checking for $m \leq 2$ guards against round-off error*

Handle the case where the current value has probability half or more specially.

If $k \neq x_1$:
Set j to x_1
Else:
Set r to a uniform random variate on $[0, 1]$
Set s to 0; Set i to 0
While $i < m$ and $r \geq s$:
Add 1 to i
If $\pi(i) > 0$:
If $i = k$:
Add $(2\pi_1 - 1) / \pi_1$ to s
Else:
Add $\pi(i) / \pi_1$ to s
Set j to i

Else:

Find the probability, π_2, of the second most probable value, and set j to its index.

Set π_2 to 0
for $i = 1, \ldots, m$:
If $i \neq x_1$ and $\pi(i) > \pi_2$:
Set π_2 to $\pi(i)$; Set j to i

Generate a uniform random variate from zero to height of bar for the current value.

Set r to a uniform random variate on $[0, \pi(k)]$

Continue with the procedure of Algorithm 9: Part 2.

Algorithm 9: Part 1. Procedure for sampling from FSS or ZFSS transition probabilities.

Find the index, x_0, of the value before the most probable value, or for ZFSS, the index of the first value before the most probable value which will block movement beyond it from encountering a piece of the most probable value.

Set x_0 to x_1
Loop:
    Set x_0 to m if $x_0 = 1$, otherwise to $x_0 - 1$
    Set π^* to $(0.5 - \pi_1) + (0.5 - \pi(x_0))$ *Computing this way reduces round-off error*
    Set f to $(\pi_1 - \pi_2) \,/\, \pi^*$ *Guaranteed in* $[0, 1]$ *even with rounding*
Repeat loop as long as ZERO and $\pi(x_0) < f\pi_2$

If $k = x_1$ and $r \geq \pi_2$:

    If the transition is from the most probable value, x_1, and r is in the region to be distributed among values other than x_1 and x_0, then select such a value, j.

    Subtract π_2 from r
    Set s to 0; Set i to 0
    While $i < m$ and $r \geq s$:
        Add 1 to i
        If $i \neq x_1$ and $i \neq x_0$ and $\pi(i) > 0$:
            Add $f\pi(i)$ to s, Set j to i

Else:

    Return a value that is transitioned to due to slice movement.

    Set ℓ to 0 *Lower end of region to move*
    Set u to π_2 if $k = x_1$, otherwise to $\pi(k)$ *Upper end of region to move*

    Set i to k, Set s to 0

    While $\ell < u$ and $r \geq s$:

        Move i backwards, going from x_1 to x_0, from x_0 to before x_1, and skipping x_0 when otherwise going back.

        If $i = x_1$:
            Set i to x_0
        Else:
            If $i = x_0$:
                Set i to m if $x_1 = 1$, otherwise to $x_1 - 1$
            Else:
                Set i to m if $i = 1$, otherwise to $i - 1$
            If $i = x_0$:
                Set i to m if $x_0 = 1$, otherwise to $x_0 - 1$

        Look at slice movement from $[\ell, u]$ region, and update ℓ and u.

        If $\ell < \pi(i)$:
            If $i \neq x_1$ and $i \neq x_0$:
                Set t to $\min(u, f\pi(i))$
                if $\ell < t$:
                    Add $t - l$ to s; Set j to x_1; Set ℓ to t
            If $r \geq s$:
                Set t to $\min(u, \pi(i))$
                    Add $t - l$ to s; Set j to i; Set ℓ to t

Algorithm 9: Part 2. Continuation of procedure for sampling from FSS or ZFSS transition probabilities.

12 Non-domination of reversible methods minimizing self transitions

Proposition 13 of (Neal and Rosenthal 2023) provides a way of showing that a method cannot be efficiency-dominated by another (see also (Mira and Geyer 1999)). It states that for reversible, irreducible transitions P and Q, if P efficiency-dominates Q, then P eigen-dominates Q, where eigen-dominance of P over Q means that if the eigenvalues of P and Q are ordered (retaining multiplicity), all eigenvalues of P are less than or equal to the corresponding eigenvalue of Q. Put in contrapositive form, this proposition says that if P does not eigen-dominate Q, it does not efficiency-dominate Q. Corollary 17 of (Neal and Rosenthal 2023) shows that if P and Q are different, but have the same set of eigenvalues, then neither efficiency-dominates the other.

It was shown in Section 9 that ZDNAM always efficiency-dominates Gibbs sampling, but this is not true for the other reversible methods that minimize self transitions. For example, with $m = 4$ and $\pi(1) = 0.4$, $\pi(2) = 0.3$, $\pi(3) = 0.2$, and $\pi(4) = 0.1$, the UDST method produces a transition matrix with eigenvalues of -0.69246, -0.35046, 0.04292, and one. Gibbs sampling transition matrices have all zero eigenvalues (apart from the single eigenvalue of one). So neither UDST nor GS eigen-dominates the other (two of the eigenvalues of UDST are less than those of GS, but one eigenvalue is greater), and hence neither can efficiency-dominate the other. There are functions that are more efficiently estimated by Gibbs sampling, and other functions that are more efficiently estimated by UDST. HST and OHST also do not efficiency-dominate Gibbs sampling for this example.

Several methods for modifying Gibbs sampling probabilities always produce transitions with the minimum possible self transition probability — zero when $\pi_{\max} = \max_i \pi(i) \leq 1/2$, and $(2\pi_{\max} - 1) \,/\, \pi_{\max}$ when $\pi_{\max} \geq 1/2$ — specifically, ZDNAM, all the ST methods, and ZFSS. The transitions produced by ZDNAM, UDST, HST, and OHST are also reversible.

Theorem 19 of (Neal and Rosenthal 2023) shows that an irreducible, reversible transition matrix with minimum possible self transition probabilities cannot be efficiency-dominated by any other reversible transition matrix. So, considered in isolation, transition matrices produced by ZDNAM, UDST, HST, and OHST cannot be dominated by a different reversible method.

This can be extended to when any reversible method minimizing self transitions is used to update a randomly-chosen variable — the resulting overall method cannot be efficiency-dominated by any other reversible method that updates a single variable chosen randomly in the same way.

The key fact to note is that the trace of a reversible transition matrix is both the sum of its self transition probabilities (which are on the diagonal) and the sum of its eigenvalues (Horn and Johnson 2013, p. 51). Theorem 16 of (Neal and Rosenthal 2023) states (in contrapositive form) that if $\text{trace}(P) \geq \text{trace}(Q)$, and $P \neq Q$, then P cannot efficiency-dominate Q.

The full transition matrix for a Gibbs sampling update of a particular variable will (with a suitable ordering of values) be block diagonal, with one block for each possible combination of values for other variables, as was previously discussed in Section 5. If the Gibbs sampling updates are modified to minimize self transitions, the trace of the full transition matrix will be the sum of the traces for each block, which will each have the minimum possible value. If a variable to update is chosen randomly with probabilities $a_1, \ldots, a_n$ (for example, with each $a_k = 1/n$), the combined transition matrix can be written as $P = \sum_k a_k P_k$, where P_k is the transition matrix for an update of variable k. The trace of P will $\sum_k a_k \text{trace}(P_k)$.

If each block of each P_k minimizes self transition probabilities, then P will have the minimum possible trace of any such method. That is, if Q is any other method (not equal to P) that operates by updating

a variable chosen at random with probabilities $a_1, \ldots, a_n$, then trace(Q) $\geq$ trace(P). It follows that Q cannot efficiency-dominate P

A stronger result applies when, for some particular problem, a method produces self transition probabilities that are always zero (something that is not always possible) — random updating of variables using this method cannot in this case be efficiency-dominated by any reversible method at all, including methods that simultaneously change the values of several (or all) variables, since no transition matrix can have a trace (sum of self transition probabilities) less than zero.

When variables are updated sequentially in some order that is randomly chosen from a distribution in which an order and its reversal are equally likely, an even stronger result is possible — as long as it is guaranteed that at least one of the variable updates has zero self transition probability, the random order scan will have zero probability of leaving the state unchanged, and hence the scan as a whole cannot be efficiency-dominated by any other reversible method.[19]

One should keep in mind, though, that such non-domination results are rather weak justifications for using a method in practice. They say only that for estimating the mean of *some* function the method is better than whatever alternative is being considered. But that does not rule out the possibility that the method is much worse for the functions of actual interest.

13 Comparisons on simple distributions

We can gain some insight into the differences between the various methods by seeing how they behave on some simple distributions. Note, though, that in real applications, the distributions will generally be more complex, and will change from one update to the next, as other variables change (unless the variables are independent, which would be an uninteresting case). So behaviour in these simple situations should not be taken as a definite indication of how well the methods will work in practice.

To begin, consider distributions in which all m values have equal probability — that is, $\pi(i) = 1/m$ for $i = 1, \ldots, m$. (Similar behaviour will occur for distributions that are approximately uniform over some subset of values, with the total probability of other values being small.) The probability of a self transition from a state chosen from π when using these probabilities directly as in Gibbs Sampling will be

$$p_{GS}^{\text{self}} \;=\; \sum_i \pi(i)\, P_{GS}(i \to i) \;=\; \sum_i \pi(i)\, \pi(i) \;=\; m\,(1/m)^2 \;=\; 1/m \tag{111}$$

The minimum possible self transition probability for such a distribution is zero, which will of course be achieved by the methods that always produce minimum self transition probabilities — namely, ZDNAM, ST, UST, DST, UDST, HST, OHST, and ZFSS. It is easy to see that, for this distribution, zero self transition probabilities will also be produced by all the other methods besides Gibbs sampling — that is, by MHGS, UNAM, DNAM, UDNAM, and FSS.

However, these methods do not all produce the same transition probabilities. MHGS, UNAM, DNAM, UDNAM, and ZDNAM all produce transitions in which $P^*(i \to j) = 1/(m-1)$ for $i \neq j$. ST, UST, DST, FSS, and ZFSS produce transitions that are periodic with period m — cycling through the m states — while UDST produces transitions that have probability 1/2 of moving to the value before or after the current value, performing a random walk around the cycle of values. For even values of m, HST and OHST produce transitions with period two that are not irreducible, while for odd values of m, their transitions

[19] Note that this applies only to estimates based on the states after each full scan (that is, on "thinned" estimates, as described below in Section 14), not necessarily to estimates that use the state after every variable update within a scan.

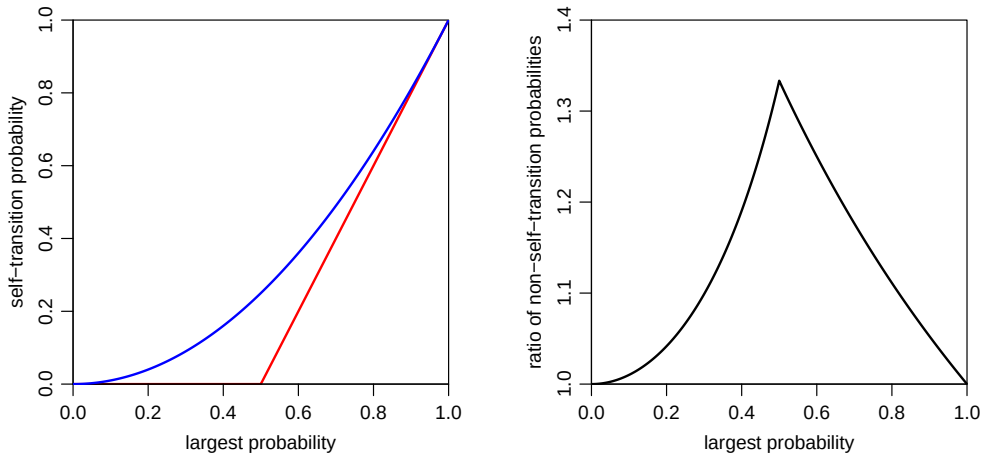


Figure 11: Behaviour of self transition probabilities for distributions with largest probability p and other probabilities of $(1-p)\,/\,(m-1)$, in the limit as m goes to infinity. The plot on the left shows the self transition probability as a function of p for GS, MHGS, and UNAM in blue, and all other methods (except UDNAM) in red. The plot on the right shows the ratio of the non-self transition probability for methods other than GS, MHGS, UNAM, and UDNAM to the non-self transition probability for GS, MHGS, and UNAM, as a function of p.

are irreducible and aperiodic. The effects of these differences when these transitions are applied to multiple variables with changing distributions, using various scan orders, are not obvious.

One intuitive measure of how much benefit we might expect from using a method for avoiding self transitions is the ratio of the probabilities of a non-self transition for such a method to that for Gibbs sampling. For the uniform distribution, this ratio is $1\,/\,(1-1/m) \;=\; m\,/\,(m-1)$ for all the non-GS methods. If we see self transitions as wasted effort, and non-self transitions as useful, this ratio represents the factor by which we might (rather naively) expect efficiency to be improved over Gibbs sampling.

Another simple case to look at is when one value has much larger probability than any other value. Specifically, let $\pi(1) = p$, let $\pi(j) \;=\; (1-p)\,/\,(m-1)$ for $j = 2, \ldots, m$, and look at the limit as m increases.

In this scenario, the probability of a self transition using Gibbs sampling from a value $j \neq 1$ is zero in the limit as m increases, while the probability of a self transition from value 1 is p, giving an overall self transition probability for GS of p^2.

One can easily compute that this is also the self transition probability for MHGS and UNAM, in the limit as m increases. MHGS and UNAM in fact produce exactly the same transition probabilities in this situation (for any m).

For methods that produce minimum self transition probabilities, the overall self transition probability is zero if $p \leq 1/2$, and $2p-1$ if $p > 1/2$. Also, in this situation DNAM and FSS produce the same transition probabilities as ZDNAM and ZFSS. (UDNAM of course has a self transition probability halfway between UNAM and DNAM.)

Figure 11 shows the self transition probabilities for this scenario, as well as the ratio of the non-self transition probability for all the methods minimizing self transitions to the non-self transition probability for GS, MHGS, and UNAM. This ratio peaks at $4/3$ when $p = 1/2$.

When $p < 1/2$, the transition probabilities in this scenario produced by ZDNAM, ST, HST, and ZFSS are all different (both for finite m and in the limit), even though they all have zero self transition probability.

However, when $p \geq 1/2$, all these methods produce the same transition probabilities.

Indeed, for *any* distribution with maximum probability one half or more, *all* methods that produce the minimum overall self transition probability of $2p-1$ must produce the same transition probabilities. Specifically, if $\pi(1) = p \geq 1/2$, then these transition probabilities must be as follows:

$$P^* \;=\; \begin{bmatrix} \frac{2p-1}{p} & \frac{\pi(2)}{p} & \ldots & \frac{\pi(m)}{p} \\ 1 & 0 & \ldots & 0 \\ \vdots & \vdots & & \vdots \\ 1 & 0 & \ldots & 0 \end{bmatrix} \tag{112}$$

To see this, note that for P^* to leave π invariant, we must have

$$\pi(1) \;=\; p \;=\; \sum_{i=1}^{m} \pi(i)\, P^*(i \to 1) \;=\; p\,\frac{2p-1}{p} \;+\; \sum_{i=2}^{m} \pi(i)\, P^*(i \to 1) \tag{113}$$

and hence

$$\sum_{i=2}^{m} \pi(i)\, P^*(i \to 1) \;=\; p \;-\; p\,\frac{2p-1}{p} \;=\; 1-p \tag{114}$$

Since $\sum_{i=2}^{m} \pi(i) \;=\; 1-p$, this is possible only if $P^*(i \to 1) = 1$ for $i = 2, \ldots, m$. Note that this P^* is reversible with respect to π, so any method that produces minimal self transition probabilities produces reversible transitions in this context, even if the method is not generally reversible.

14 Framework for empirical comparisons

I will empirically compare the performance of the modified Gibbs sampling procedures with each other and with standard Gibbs sampling for three problems: the well-known Potts model used in statistical physics and image processing, sampling of mixture indicators for a Bayesian mixture model, and sampling of unobserved variables in a belief network. Of course, the results of these experiments are only suggestive of performance in other applications, in which the distributions sampled may have different characteristics.

I will evaluate all the fourteen methods discussed earlier, which can be grouped as follows:

1) Gibbs sampling and methods that can be viewed as deriving from it: GS, MHGS, UNAM, DNAM, UDNAM, and ZDNAM.

2) Shifted tower methods: ST, DST, UST, UDST, HST, and OHST.

3) Slice sampling methods: FSS and ZFSS.

Of these, ZDNAM, all the shifted tower methods, and ZFSS always minimize self transition probability, and all the methods in group (1) plus UDST, HST, and OHST always produce reversible transitions.

Each method will be used in combination with several schemes for choosing which of the n variables are updated in each iteration. For all schemes, n variable updates are considered to constitute a *scan*, which is sometimes viewed as a single iteration. The schemes used may include the following:

1) **Random.** For each iteration, one of the n variables is randomly selected to be updated, independently of previous iterations.

2) **Sequential.** The variables are updated in a predefined order from 1 to n, which constitutes one scan. Not done for mixture models, for which there is no meaningful predefined order.

3) **Shuffled sequential.** The variables are randomly shuffled, once, at the beginning of a run, the same way for all runs. They are then repeatedly updated in this shuffled order, with each set of n updates considered one scan.

4) **Checkerboard.** Only done for the Potts models, for which the n variables are arranged in a square array, on which one can imagine a checkerboard pattern being placed. A scan consists of updates for all the variables on black squares, followed by updates for all the variables on white squares.

5) **Random order.** For each scan, an order of the n variables is chosen at random, and the variables are then updated in this order. A new random order is chosen for the next scan.

6) **Random order times four.** Like the random order method, except that the same random order is used for four scans in a row, before a new random order is chosen.

For each combination of method and scan order, the Markov chain is simulated for a large number, K, of scans, starting with a random state, producing a total of nK states. These states are then used to form estimates for the expectation of several functions of the state variables. No iterations are discarded as "burn-in", since the length of the runs and the speed of convergence make this unnecessary. Both *thinned* and *unthinned* estimates are found. The unthinned estimate for the expectation of a function is the average value of the function at all iterations. The thinned estimate is the average over only the values after the last update of a scan. The unthinned estimates are therefore averages over nK function values, whereas the thinned estimates are averages over K function values.

For each distribution tested, three groups of methods are tested using sets of four runs for each method, with all runs being independent (using different random number seeds). The three groups compare the following selections of methods:

1) GS, MHGS, UNAM, DNAM, UDNAM, ZDNAM.

2) ST, DST, UST, UDST, HST, OHST.

3) UNAM, ZDNAM, ST, UDST, FSS, ZFSS.

The first group compares methods related to Gibbs sampling, the second compares the shifted tower methods, and the third compares what appear to be the best from the first two groups along with the slice sampling methods.[20] Summary graphs are produced for each group, for each of the distributions tested.

The efficiency of a method and scan order for a particular function is measured by an estimate of the asymptotic variance (equation (5)) for that function, found using the following formula:

$$v(f,P) \;=\; \gamma_0 \;+\; 2\sum_{k=1}^{\infty}\gamma_k \tag{115}$$

where γ_k is the autocovariance of f at lag k, defined by

$$\gamma_k \;=\; = E\Big[\Big(f(X^{(t)})-\mu\Big)\Big(f(X^{(t+k)})-\mu\Big)\Big] \tag{116}$$

[20]Note that runs in the third group are independent of those in the first two groups for the same method.

where the expectation is over realizations of the Markov chain with transitions P, which leave π invariant, started from a state drawn from π, and μ is the expectation of f with respect to π. Since the realization will be stationary, the choice of t in the above formula makes no difference. Note that γ_0 is the variance of f with respect to π.

This formula is proved for homogeneous reversible chains with a finite state space in (Neal and Rosenthal 2023, Proposition 3)), but holds more generally, including for chains with non-reversible transitions, and those in which the transitions depend on the time index in a periodic way (as for Gibbs sampling with a sequential scan), if we interpret γ_k as the average covariance between $f(X^{(t)})$ and $f(X^{(t+k)})$ as transitions at time t vary periodically, provided that the distribution at time t converges to π as t goes to infinity, and the variance is finite (as is always the case for a finite state space).[21]

From a realization of the chain of length N, the standard estimate of γ_k is

$$\hat{\gamma}_k \;=\; \frac{1}{N} \sum_{t=1}^{N-k} \Big(f(x^{(t)}) - \mu\Big)\Big(f(x^{(t+k)}) - \mu\Big) \tag{118}$$

If μ is not known, it may be replaced by $\hat{\mu} = (1/N) \sum_{t=1}^{N} f(x^t)$.

The asymptotic variance for f is then estimated as

$$\hat{v}(f,P) \;=\; \hat{\gamma}_0 \;+\; 2 \sum_{k=1}^{M} \hat{\gamma}_k \tag{119}$$

where M is selected such that $\hat{\gamma}_k$ is nearly zero for $k > M$. Note that this estimate will be good only if the length of the run, N, is much larger than a suitably chosen value of M.[22]

For unthinned estimates, the estimate for the asymptotic variance based on a run of K scans will use $N = nK$ in the above formulas. For thinned estimates, there are only $N = K$ function values used, but in the presentations of results, the asymptotic variance estimates with thinning are multiplied by n to account for each value used in estimation requiring a factor of n more computation time.

The practical motivation for thinning is usually to reduce memory requirements and time for computing function values by a factor of n, with the expectation that the efficiency of estimation will be worse than if all values were used for estimates, though only slightly worse for typical problems. The belief that thinning gives worse estimates is generally correct (provably so for reversible updates on randomly chosen variables (Geyer 1992, Section 3.6)), but as will be seen below, there is one context in which thinning actually improves estimation efficiency.

---

[21] Supposing that $\mu = 0$ for simplicity, this is a simple consequence of expanding N times the variance of the mean estimate from a run of length N as

$$NE\Big[\Big(\frac{1}{N}\sum_{t=1}^{N} f(X^{(t)})\Big)^2\Big] \;=\; E\Big[\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{N} f(X^{(i)})f(X^{(j)})\Big] \;=\; E\Big[\frac{1}{N}\sum_{t=1}^{N} f(X^{(t)})^2 \;+\; 2\sum_{k=1}^{N-1}\frac{1}{N}\sum_{t=1}^{N-k} f(X^{(i)})f(X^{(i+k)})\Big] \tag{117}$$

which equals $E\big[\hat{\gamma}_0 + 2\sum_{k=1}^{N-1}\hat{\gamma}_k\big]$, where $\hat{\gamma}_k$ is the estimate for γ_k given in equation (118) below. Since the expectations of these estimates converge to the true γ_k as N goes to infinity, equation (115) will hold generally.

[22] Note that setting M to the largest possible value of $N-1$ is not good, since the estimate will then have a large variance dominated by estimates $\hat{\gamma}_k$ whose means are close to zero.

15 Comparisons for Potts models

The Potts model originates in statistical physics (e.g., Landau and Binder 2009, Section 4.3.2), but similar models are also used for image analysis (e.g., Geman and Geman 1984) and other applications in which some discrete aspect of a system exhibits local spatial coherence.

I will consider two-dimensional Potts models, which define a distribution on the space of arrays of variables, $x_{r,c} \in \{1, \ldots, m\}$, for $r = 0, \ldots, R-1$ and $c = 0, \ldots, C-1$, with the total number of variables being $n = RC$. Variables are "neighbors" if one is immediately above, below, left, or right of the other, with row or column positions wrapping around from $R-1$ or $C-1$ to 0. The distribution is defined by

$$\pi(x) \;=\; \frac{1}{Z} \exp\Big(b \sum_{r=0}^{R-1} \sum_{c=0}^{C-1} \big(I(x_{r,c} = x_{r^+,c}) \,+\, I(x_{r,c} = x_{r,c^+}) \big) \Big) \tag{120}$$

where $r^+ = r+1 \mod R$ and $c^+ = c+1 \mod C$. Here, Z is the normalizing constant needed to make these probabilities sum to one. The parameter b controls how strongly variables at neighboring positions tend to be the same (if $b > 0$) or different (if $b < 0$). In physical terms, minus the sum inside the exponential above is proportional to the "energy" of the system, and $1/b$ is proportional to the "temperature".

A Gibbs sampling update for this model will choose a new value for $x_{r,c}$ from $\{1, \ldots, m\}$, with r and c chosen either randomly or sequentially in some order. The conditional distribution for $x_{r,c}$ given the other variables depends only on the four variables above, below, left, or right of $x_{r,c}$. There are m^4 possible values of these four neighbors, so when m is fairly small, it is possible to precompute the Gibbs sampling probabilities for all combinations of neighboring values. Similarly, modified Gibbs sampling probabilities, found with any of the methods discussed, could be precomputed for all combinations of neighboring values and all possibilities for the current value of the variable being updated. All methods would then take close to the same computation time (though for ordinary Gibbs sampling, the table of possible distributions would be m times smaller).

For my experiments, I used models with $m = 4$, and either $R = C = 8$ ($n = 64$) with $b = 0.85$ or $R = C = 5$ ($n = 25$) with $b = -0.4$. In actual applications, R and C are typically larger, but with these smaller values, very long runs can be done to obtain accurate comparisons of asymptotic variance. For simplicity of implementation, I did not precompute probabilities for these experiments.

All the scan orders described in Section 14 were tested. The pre-defined sequential order was a raster scan over rows and columns. When R and C are even, note that the checkerboard scan updates of black positions can be done in parallel, since there are no interactions between the sites being updated, after which the updates of the white positions can be done in parallel. This will often be a significant computational advantage of this scan. However, when R or C are odd, the checkerboard scan will have adjacent sites of the same colour at the point of wrapping around from $R-1$ or $C-1$ to 0, which inhibits parallel updates at these positions. In these experiments, I did not do updates in parallel for the checkerboard scans, nor do the presentations of results account for any efficiency advantage of using a checkerboard scan.

The expectations of three functions of state were estimated from the runs done:

1) **Count of 1s.** The number of the $n = RC$ variables whose value is 1. Since the distribution is symmetrical with respect to the m possible values, this function has the same expectation as that of the number of variables with any other value. From symmetry, the expectation of this function must be RC/m, but its variance will vary with b.

2) **Sum of squared counts.** The sum of the squares of the counts of how many variables have each

```
1 3 3 4 4 4 3 4 | 1 2 1 3 2 2 4 3 | 3 2 4 4 4 1 3 3
4 1 2 2 3 3 3 4 | 4 4 1 4 4 4 4 4 | 3 1 4 4 2 1 1 3
2 4 4 1 4 1 4 4 | 4 3 3 1 2 2 3 4 | 3 3 3 3 2 3 3 3
2 3 2 2 2 2 4 2 | 3 3 1 1 1 1 3 3 | 3 3 3 3 2 1 3 3
4 2 2 2 2 1 4 4 | 2 4 1 4 4 4 3 3 | 3 3 3 3 3 3 3 3
4 2 4 4 3 2 2 4 | 4 3 4 4 4 3 1 1 | 2 2 3 4 4 3 3 3
4 2 4 3 3 2 2 4 | 4 3 4 4 2 3 4 4 | 2 4 4 4 4 4 3 2
4 1 2 4 1 1 1 1 | 2 2 2 2 2 2 4 4 | 3 2 4 4 1 1 3 3
```

Figure 12: Three 8×8 arrays of values sampled from the Potts distribution with $m = 4$ and $b = 0.85$.

of the m possible values. This has its largest possible value, of $(RC)^2$, when all variables have the same value, so one value has a count of RC and the others have counts of zero.

3) **Number of neighbor pairs with equal values.** The number of the $2RC$ pairs of neighboring variables that have the same value. If the variables took on the m values uniformly and independently, the expected value would be $2RC/m$ which is $RC/2$ when $m = 4$ as in these experiments. Note that this function is proportional to minus the "energy" in the physical interpretation; it is also proportional to the log of the joint probability of all variables.

The 8×8 Potts models used a positive value for b of 0.85, so there will be a tendency for neighboring sites to have the same values. Three arrays of values sampled from this distribution are shown in Figure 12.

For this distribution, the average count of 1 values is exactly 16, from symmetry, with a variance of approximately 66. The sum of squared counts of the four possible values has an average of approximately 1.29×10^3 and variance of approximately 5.6×10^4. The average number of neighbor pairs with equal values is approximately 61.9, more than 32, which it would be if values for sites were drawn uniformly and independently, as expected with a positive b. The variance is approximately 67.

Each run for the 8×8 Potts model consisted of $K = 200000$ scans, each with $n = RC = 64$ variable updates. For each of the three groups of methods, four independent runs of this length were done for each method in the group.

Estimates of autocovariance functions for the number of equal neighbors (proportional to the energy) based on one of the four sets of runs done for the third group of methods are shown in Figure 13. These plots show that for the 8×8 Potts model autocovariances for all methods with all scan orders are positive. This is expected, since a fairly large positive value for b of 0.85 leads to variables often having most neighbors the same (as seen in Figure 12), and hence the conditional distribution for that variable is concentrated on this dominant neighboring value, leading to slow movement through the state space, and high autocovariances for most functions.

This can also be seen from the frequencies of self transitions of the various methods for the 8×8 Potts model (which are the same for all scan orders):

GS: 0.46, MHGS: 0.33, UNAM: 0.31, DNAM: 0.24, UDNAM: 0.28, FSS: 0.24

ZDNAM, ST, DST, UST, UDST, HST, OHST, ZFSS: 0.23 (the minimum possible)

The maximum conditional probability for an update was half or more 40% of the time.

Although eight methods achieve the minimum self transition probability, these methods have substantially different transition probabilities. Figure 14 shows, for each method, how the transition probabilities

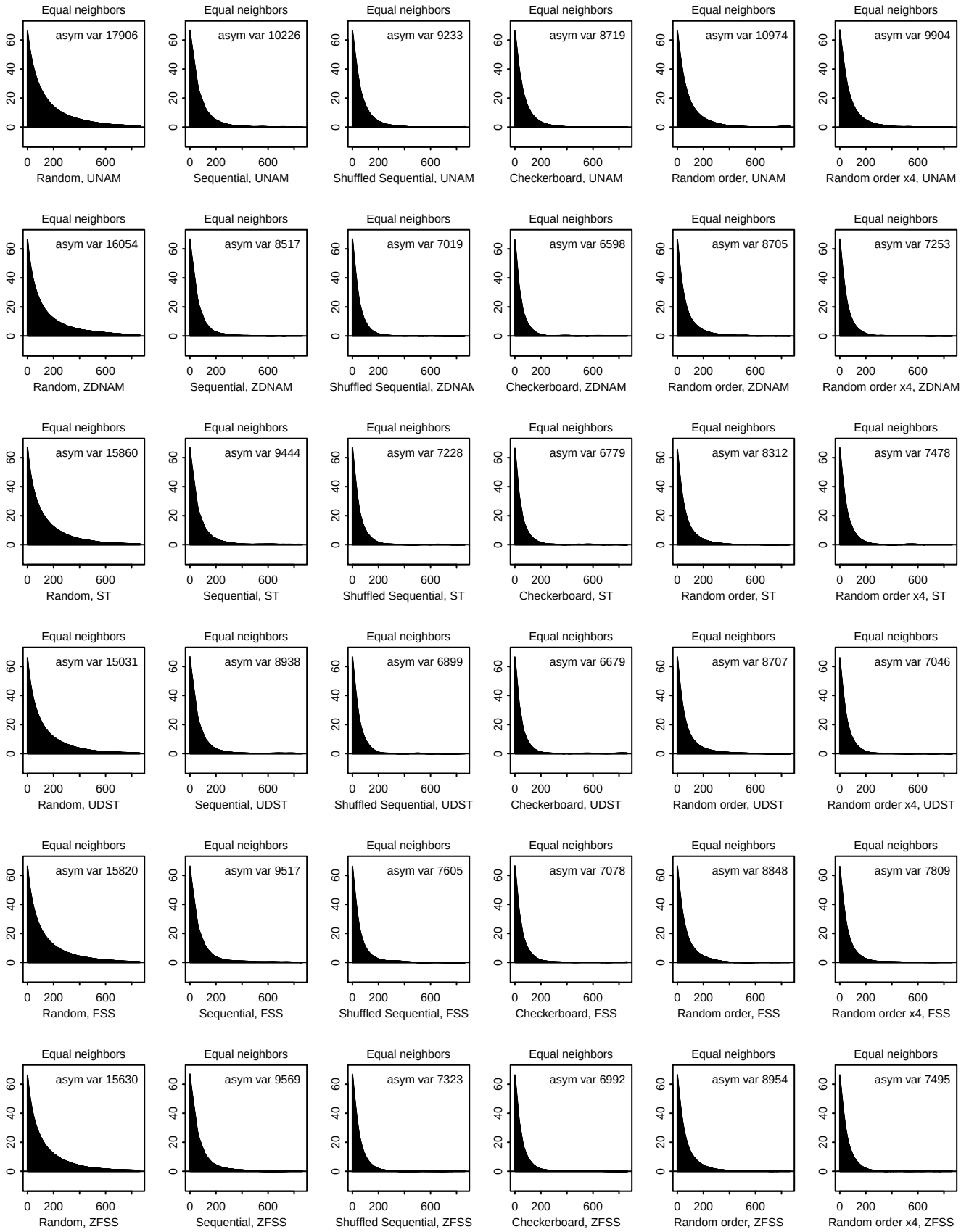


Figure 13: Autocovariance function estimates for the number of equal neighbors, from one set of runs for the 8×8 Potts model, for methods in the third group.

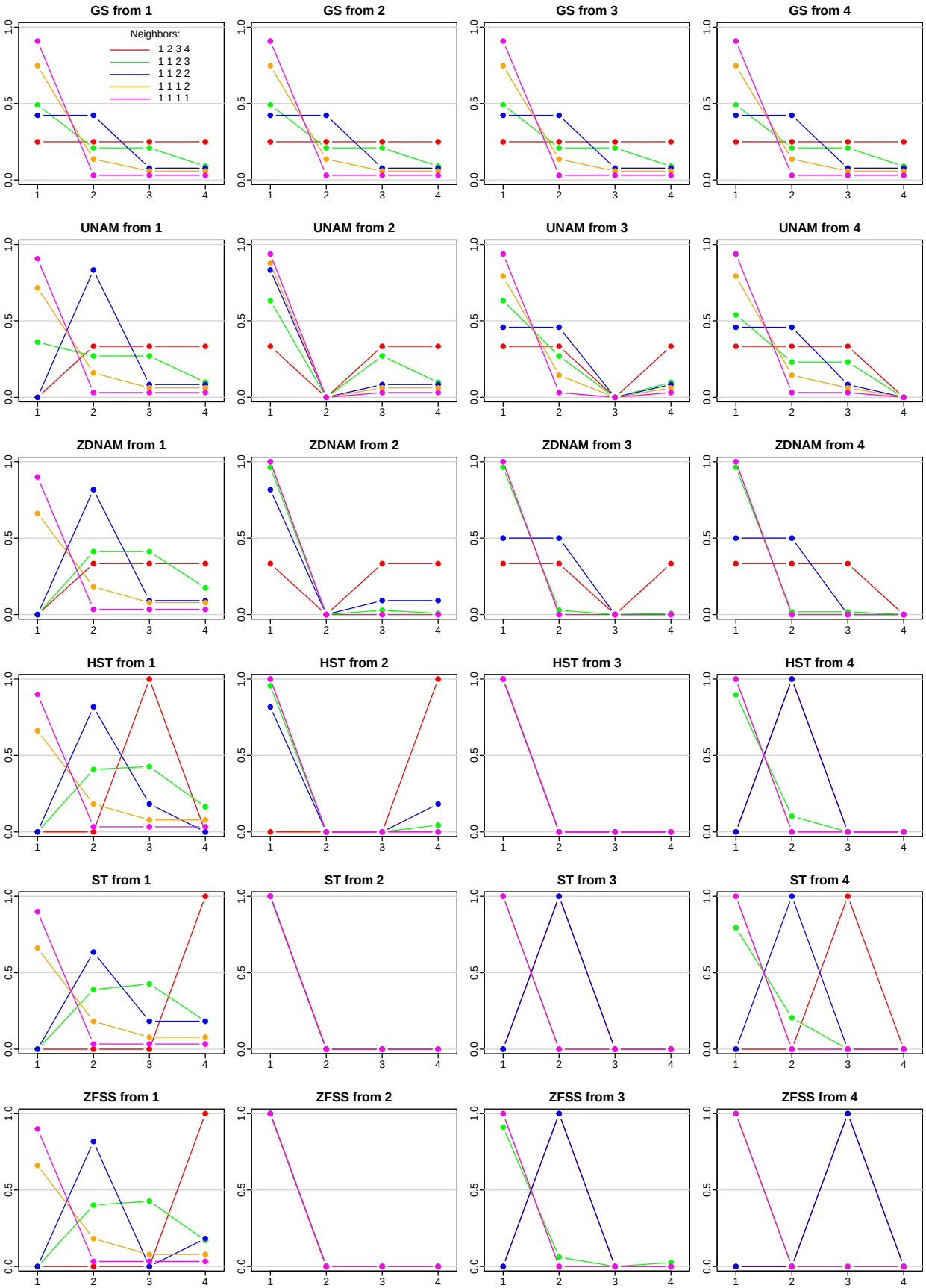


Figure 14: Transition probabilities in different contexts for the 8×8 Potts model.

vary depending on current value of the variable being updated and the context of values for its neighbors. The five possible contexts are (1) all neighbors different, (2) two neighbors the same, the others different, (3) two neighbors the same, other two also the same, but different from the first two, (4) three neighbors the same, the remaining one different, and (5) all neighbors the same. For each context, the transition probabilities are shown for each current value of the variable (all the same for Gibbs sampling, since it ignores the current value).

Summaries of asymptotic variance estimates for all three functions looked at, for all groups of methods, and all scan orders, are shown in Figures 15 through 17. The summary plots show both the asymptotic variance estimates from the four individual runs, as dots, and the average of these estimates, as lines connecting average estimates for the various methods (for each scan order, as indicated by colour).

It is evident from these figures that random selection of the variable to update (black dots and lines) is greatly inferior to the other scan orders. With a few exceptions, this will prove to also be true for the problems looked at later. One disadvantage of random selection is that by chance some variables will not be updated for many iterations. This may suffice to explain why it usually performs poorly. It is, however, the only scan order for which the theoretical analysis presented earlier applies (apart from some of the non-dominance results).

The results when the variable to be updated is selected at random (black dots and lines) are consistent with these theoretical results. Theory says that MHGS, UNAM, DNAM, UDNAM, and ZDNAM efficiency-dominate GS, and for the three functions looked at, we do see in Figure 15 that these methods have substantially lower asymptotic variance than GS. Theory also says that UNAM should efficiency-dominate MHGS, but in this case the differences in asymptotic variance are quite small, and for the sum of squared counts, the average estimate for asymptotic variance for UNAM is actually slightly greater than for MHGS — though from the spread in results of the four individual runs, this can be attributed to chance.

With random selection of variable to update, DNAM and ZDNAM perform somewhat better than UNAM or UDNAM, though there is no theoretical guarantee of this. DNAM, ZDNAM, the shifted tower methods (see Figure 16) and the slice sampling methods (see Figure 17) all perform very similarly.

Theory also says that for reversible methods, with random selection of variable to update, thinning (looking only at the state after every n updates) should produce worse estimates (Geyer 1992, Section 3.6). The results on the 8×8 Potts model for the methods in Figure 15 (all reversible) and for the reversible UDST, HST, and OHST methods in Figure 16 are consistent with this, but a higher asymptotic variance with thinning (after multiplying by n to account for computation time) is only noticeable for the "equal neighbors" function, and even there the difference is small. This is expected when, as here, autocovariances are high.

For this problem, thinning also has a very small effect on efficiency for non-reversible methods and scan orders other than random selection, with one surprising exception — when each scan updates all variables in a random order (different for each scan), thinning often gives a noticeable *reduction* in asympotic variance. See the blue dots and lines in Figures 15 through 17. This is true for all methods, and all three functions, though it is less noticeable for the 'equal neighbors" function than for the other two.

The same phenomenon will be seen later for other problems. A possible explanation can be seen by considering an extreme circumstance in which we are estimating the expectation of a function that depends on only a single variable, which is independent of the other variables. When each scan updates variables in a random order, this variable will sometimes be updated early in the order, and sometimes late in the order. If a scan in which it is updated late is followed by a scan in which it is updated early, the newly

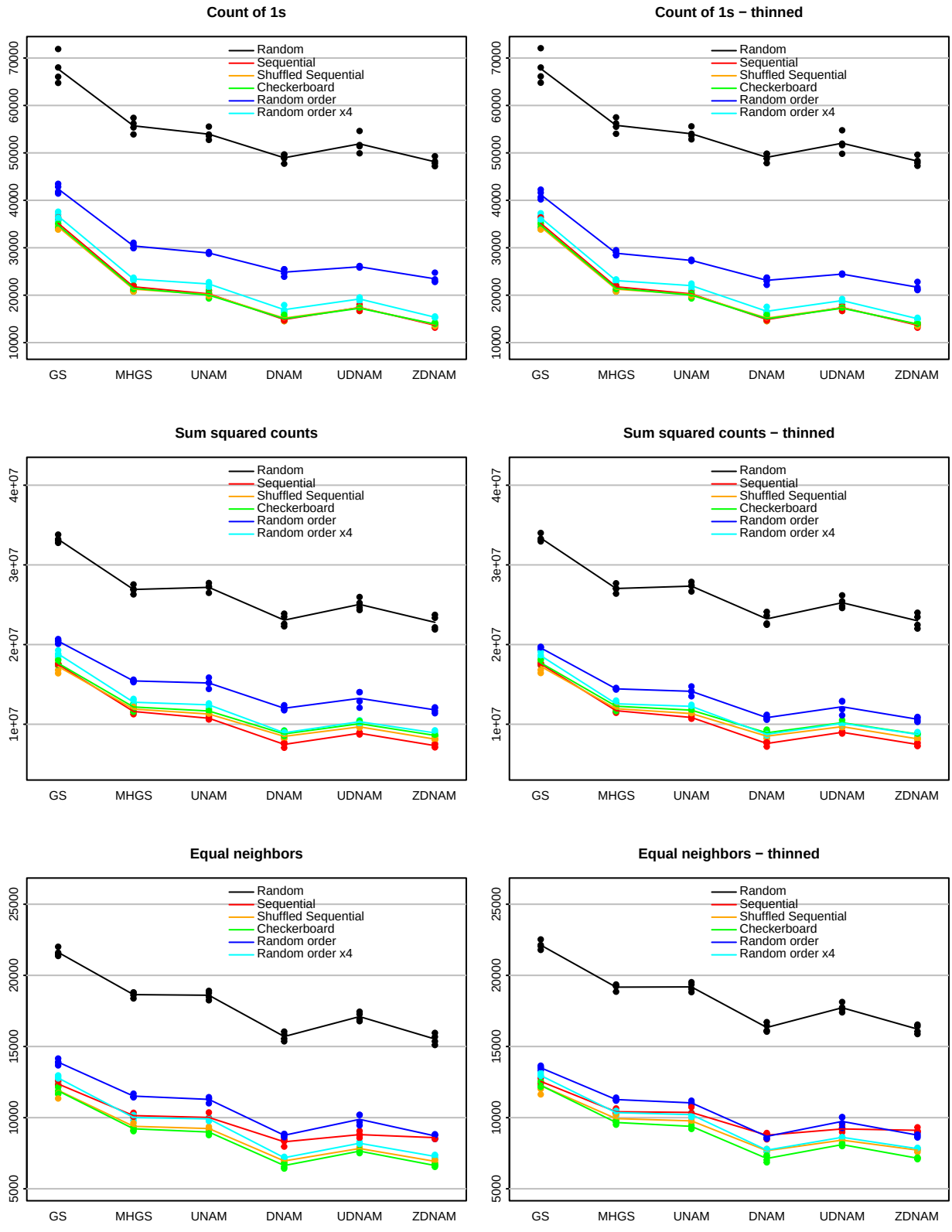


Figure 15: Summaries of autocovariance function estimates for the 8×8 Potts model, for the first group of methods.

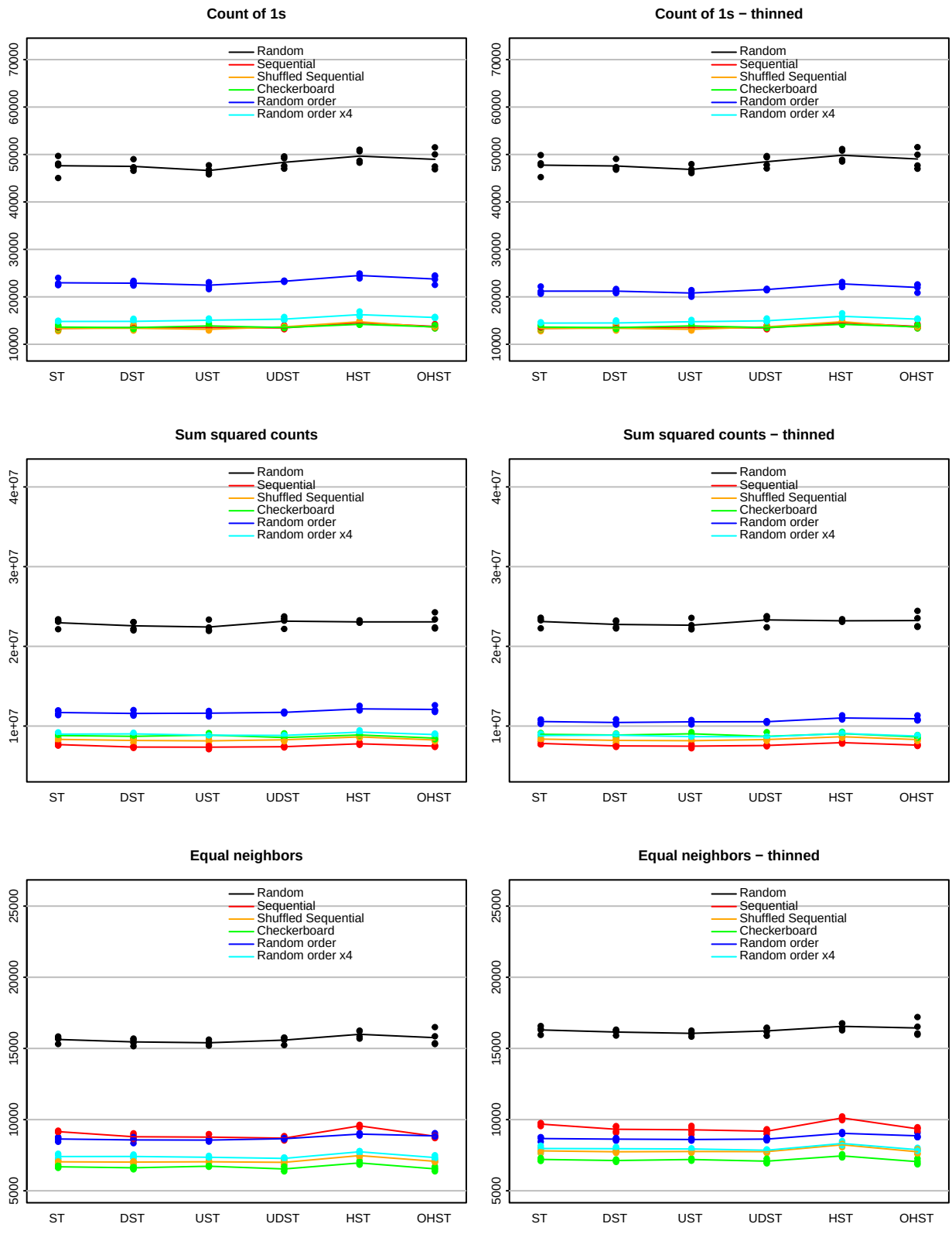


Figure 16: Summaries of autocovariance function estimates for the 8×8 Potts model, for the second group of methods.

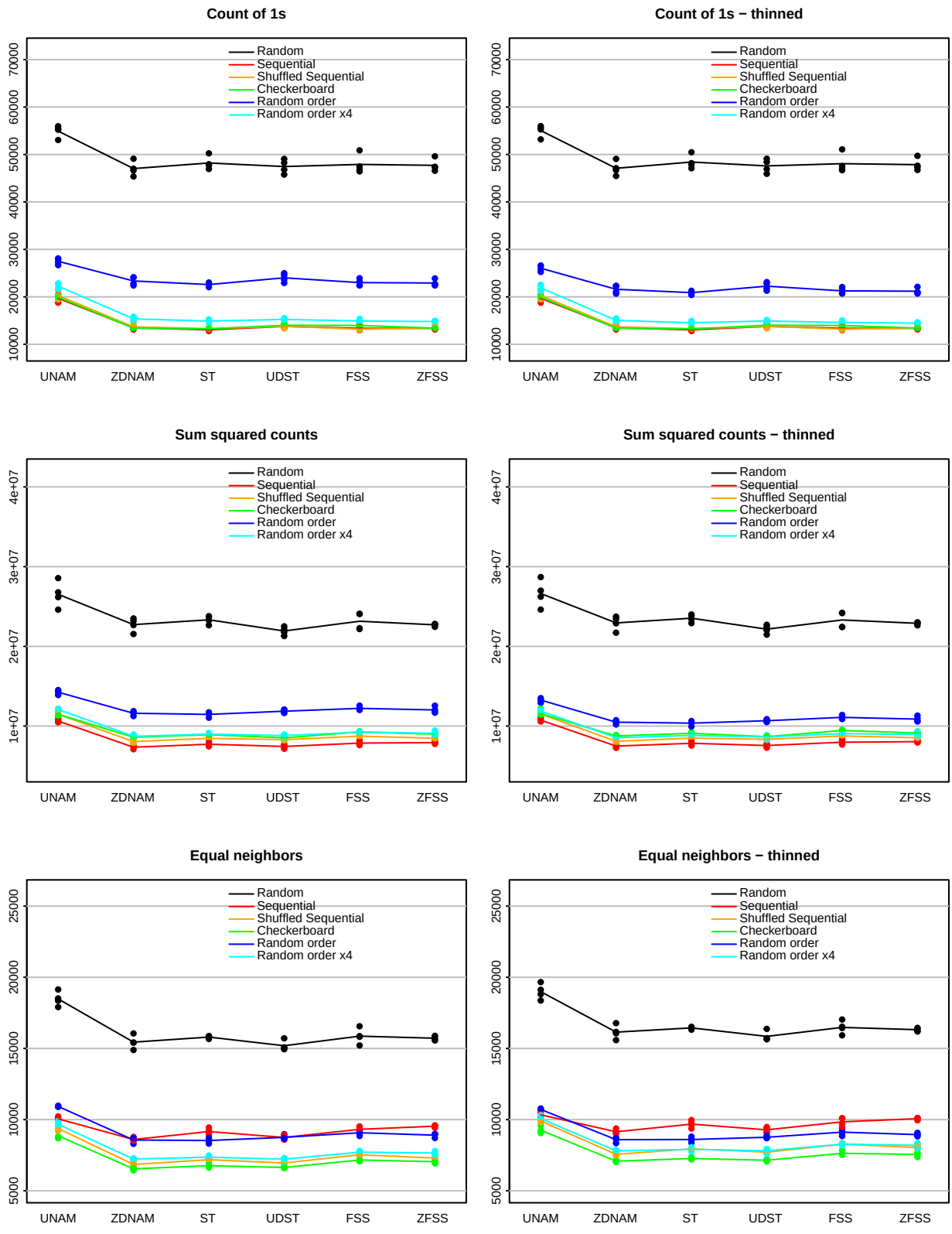


Figure 17: Summaries of autocovariance function estimates for the 8×8 Potts model, for the third group of methods.

```
1 2 1 3 2 | 3 2 2 4 4 | 2 1 2 1 3 | 1 1 1 1 2 | 3 2 1 4 3
3 3 1 3 1 | 1 4 3 3 4 | 4 3 2 2 1 | 3 2 2 2 1 | 2 3 1 2 1
4 1 2 3 2 | 2 3 2 4 3 | 3 2 3 3 4 | 2 1 4 1 3 | 2 2 2 3 3
2 3 2 3 1 | 3 1 3 1 2 | 4 4 1 4 2 | 4 2 3 4 1 | 3 2 1 4 2
2 1 1 4 3 | 2 3 3 3 4 | 2 4 3 1 3 | 3 4 1 4 2 | 2 3 1 4 1
```

Figure 18: Five 5×5 arrays of values sampled from the Potts distribution with $m = 4$ and $b = -0.4$.

sampled value will be present for only a few iterations (much less than n), whereas in the opposite case, the newly sampled value could be present for almost $2n$ iterations. When all iterations are used for estimation, this introduces random variation into how much each sampled value affects the estimate, which reduces estimation efficiency. However, a thinned estimate will look only at the last iteration of each scan, and use each sampled value equally, giving an estimate with lower variance. This effect should also be present to some extent in less extreme circumstances.

Though usually better than random selection, a random scan order is usually worse than all the other scan orders, for both this problem and for ones considered later. Repeating the same random order for four scans before generating a new order (see the cyan dots and lines) is almost always an improvement on using a random order for just one scan. This is understandable, since using the same random order four times reduces random variation in intervals between updates of the same variable, which plausibly is beneficial in most circumstances, though there is no theoretical guarantee of this. The "shuffled sequential" order takes this further, generating one random order that is used for all scans (the same order for all runs). This is almost always better that repeating the same scan only four times.

For the Potts model, two other scans are also tried — a sequential raster scan across each row, then across the next row, etc., and the "checkerboard" scan, of first all "black" variables and then all "white" variables, as described earlier. For the 8×8 Potts model, one or the other of these is always the best scan, for the functions tested, but which is best depends on the function. The sequential scan is best for the sum of squared counts, the checkerboard scan is best for the number of equal neighbors, and these two are almost the same (and better than the others) for the count of 1s. Somewhat surprisingly, the sequential raster scan is worse than the shuffled sequential scan when estimating the expected number of equal neighbors (though better for the other two functions).

For the most part, the choice of scan order does not affect which of the modified Gibbs sampling methods is best. GS, MHGS, UNAM, and UDNAM are uniformly worse than the other methods. Very little difference is seen amongst the shifted tower methods (Figure 16), except that HST is perhaps slightly worse than the others. DNAM and FSS do not minimize self transition probabilities, but for this problem their self transition probabilities are only slightly greater than the minimum, and they perform only slightly worse than ZDNAM and ZFSS. The performances of the ZDNAM, ST, DST, UST UDST, OHST, and ZFSS methods are difficult to distinguish, but they equal or exceed the performance of the other methods for all scan orders.

The 5×5 Potts models used a negative value for b of -0.4, so neighboring sites will tend to have different values. Five arrays of values sampled from this distribution are shown in Figure 18.

For this distribution, the average count of 1 values is exactly 6.25, from symmetry, with a variance of approximately 3.37. The sum of squared counts of the four possible values has an average of approximately 170 and variance of approximately 116. The average number of neighbor pairs with equal values is approximately 9.09, less than 12.5, which it would be if values for sites were drawn uniformly and independently,

as expected with a negative value for b. The variance is approximately 7.7.

Each run for the 5×5 Potts model consisted of $K = 1000000$ scans, each with $n = RC = 25$ variable updates. For each of the three groups of methods, four independent runs of this length were done for each method in the group.

Estimates of autocovariance functions for the number of equal neighbors (proportional to the energy) based on one of the four sets of runs done for the third group of methods are shown in Figure 19. In contrast to the 8×8 model with positive b, this 5×5 model with negative b has negative autocovariances for some combinations of update method and scan order. Of particular note are the negative autocovariances for ZDNAM and UDST when the checkerboard scan order is used, which result in the smallest asymptotic variances for this function.

The antithetic effects of modified Gibbs sampling updates have more scope to produce negative autocovariances when b is negative, since *avoiding* the value of a neighboring variable can (with $m = 4$) be done in more than one way, so an antithetic method can switch between them, whereas *matching* a neighboring value can be done in only one way.

This effect shows up in the frequencies of self transitions for the various methods, which are:

GS: 0.274, MHGS: 0.064, UNAM: 0.031, DNAM: 0.011, UDNAM: 0.021, FSS: 0

ZDNAM, ST, DST, UST, UDST, HST, OHST, ZFSS: 0

The maximum conditional probability for an update was never half or more, and hence the minimum self transition probability is zero, ensuring that there is an antithetic aspect to the sampling.

For the 5×5 model with negative b, Figure 20 shows, for each method, how the transition probabilities vary, depending on the current value of the variable being updated and on the context of values for its neighbors. This may be compared to Figure 14 for the 8×8 model with positive b. One thing to note for the 5×5 model is that ZDNAM, HST, ST, and ZFSS all have zero self transition probability in all contexts, but ZDNAM differs from the others in almost always having non-zero transition probabilities to values other than the current value. The HST, ST, and ZFSS methods have many zero transition probabilities, both for the 5×5 and 8×8 models.

These zero transition probabilities may be responsible for the somewhat erratic performance of these methods on the 5×5 model, as can be seen in the summaries of asymptotic variance estimates in Figures 21 through 23. (Note that, in these figures, the dots for the four runs with each method and scan order are close enough to mostly appear as one dot.)

For the methods deriving from Gibbs sampling, results without thinning, shown on the left in Figure 21, are similar to those for the 8×8 Potts model. GS has the highest asymptotic variances, followed by MHGS, with asymptotic variances for UNAM slightly lower than MHGS. This is as expected by theory for a random scan. The DNAM, UDNAM, and ZDNAM methods are somewhat better than UNAM, with ZDNAM performing best. One difference from the 8×8 model is that the sequential scan is never the best — the shuffled sequential scan (which uses a fixed random order rather than a systematic raster scan) is always better. For the "equal neighbors" function, the checkerboard scan is best of all.

The results with thinning are shown on the right in Figure 21. For the random order scan, thinning reduces asymptotic variance for the "count of 1s" function, a phenomenon discussed earlier for the 8×8 Potts model. For all other functions, scans, and methods, thinning increases asymptotic variance. This is as expected, but for the "sum squared counts" function, the amount of increase varies substantially with scan order, so much so that the random scan is better than all other scan orders for all methods except

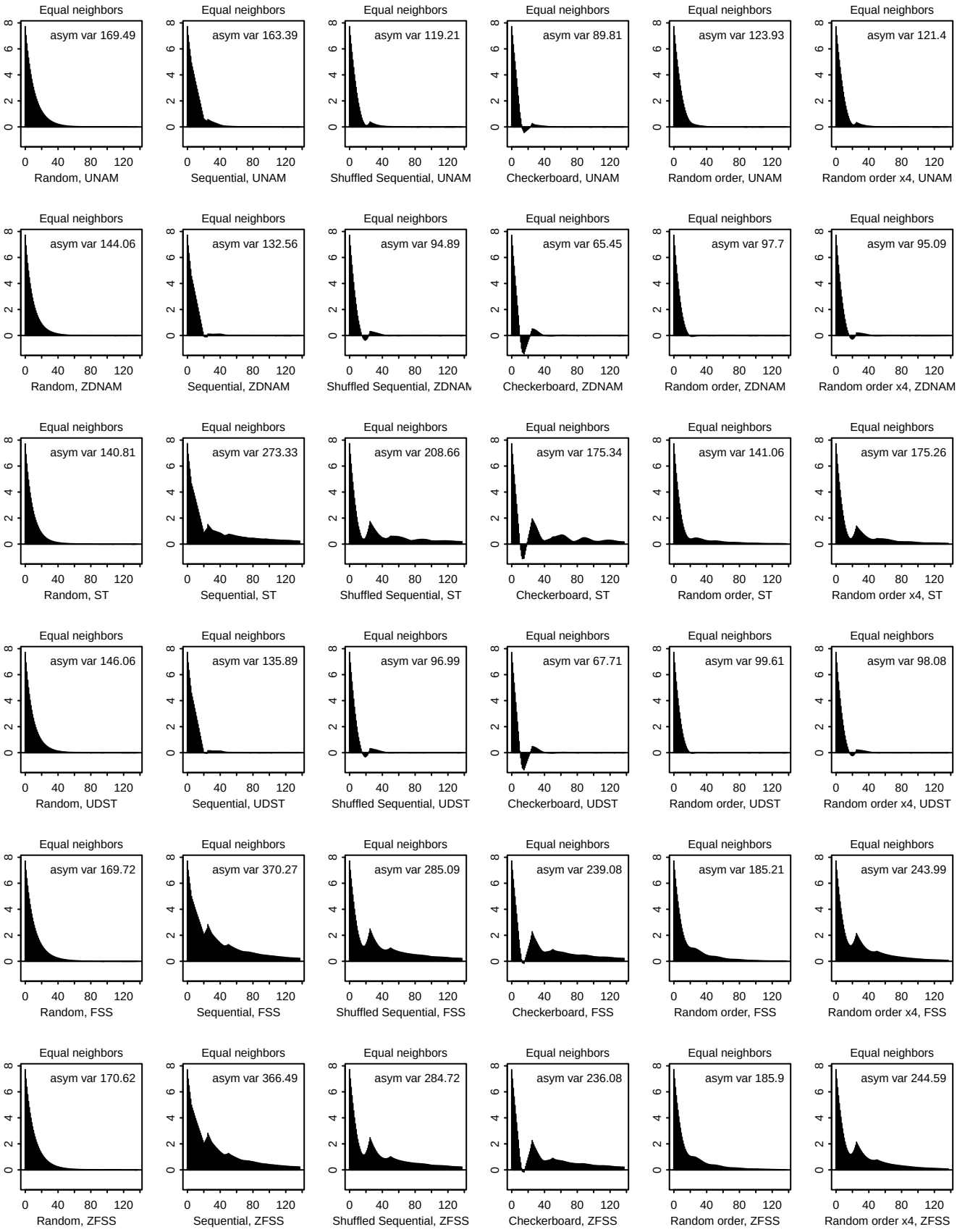


Figure 19: Autocovariance function estimates for the number of equal neighbors, from one set of runs for the 5×5 Potts model, for methods in the third group.

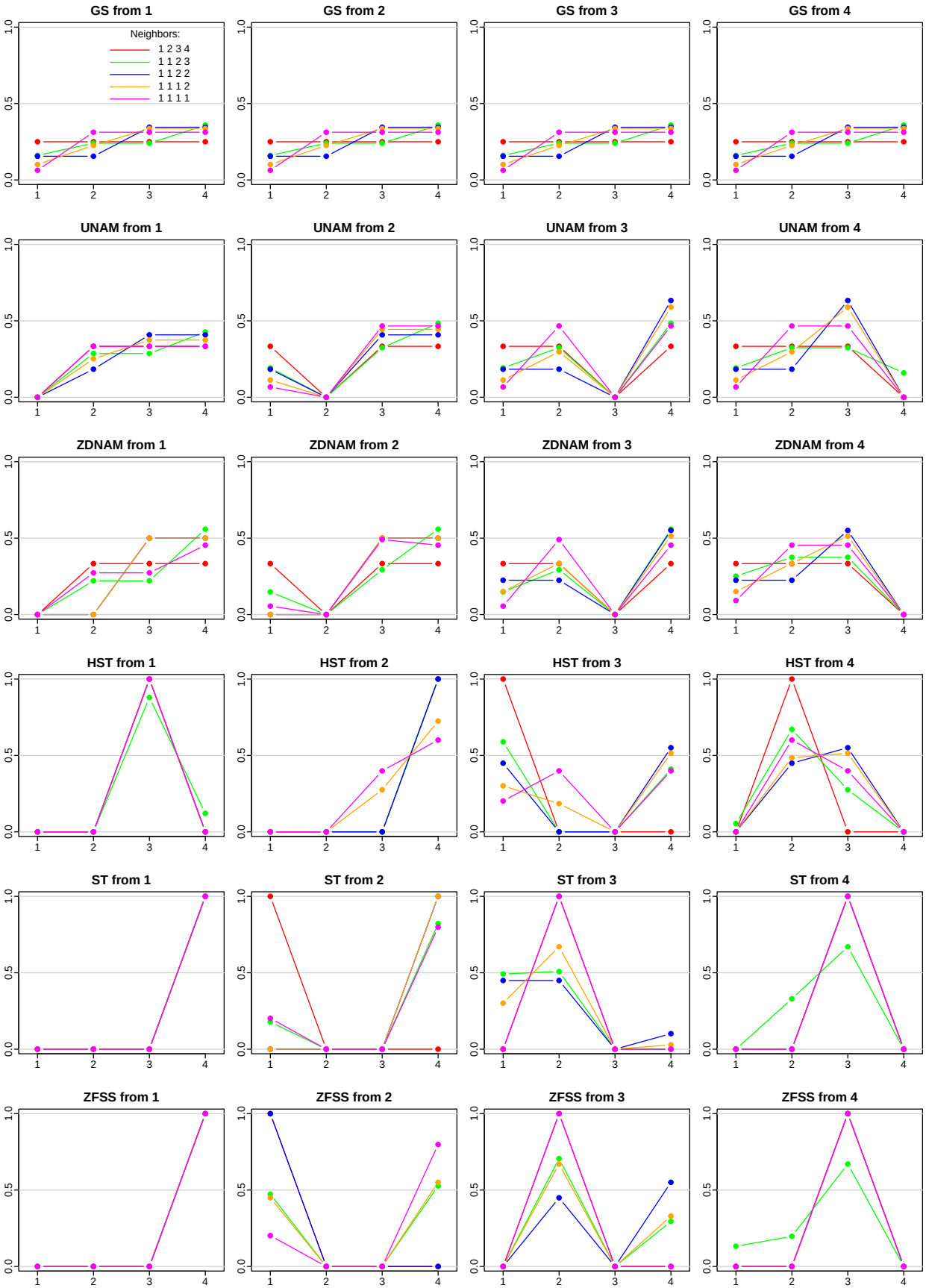


Figure 20: Transition probabilities in different contexts for the 5×5 Potts model.

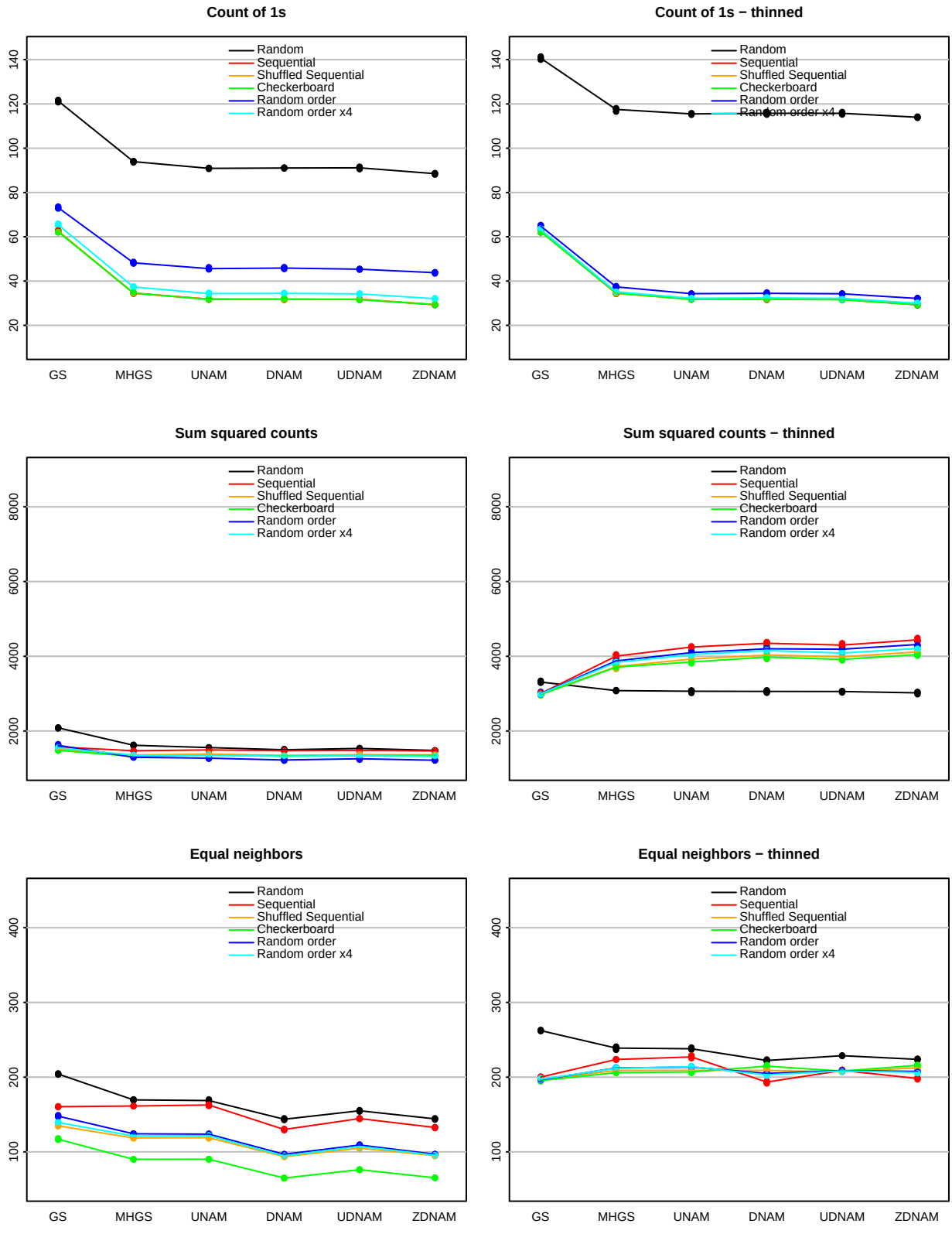


Figure 21: Summaries of autocovariance function estimates for the 5×5 Potts model, for the first group of methods.

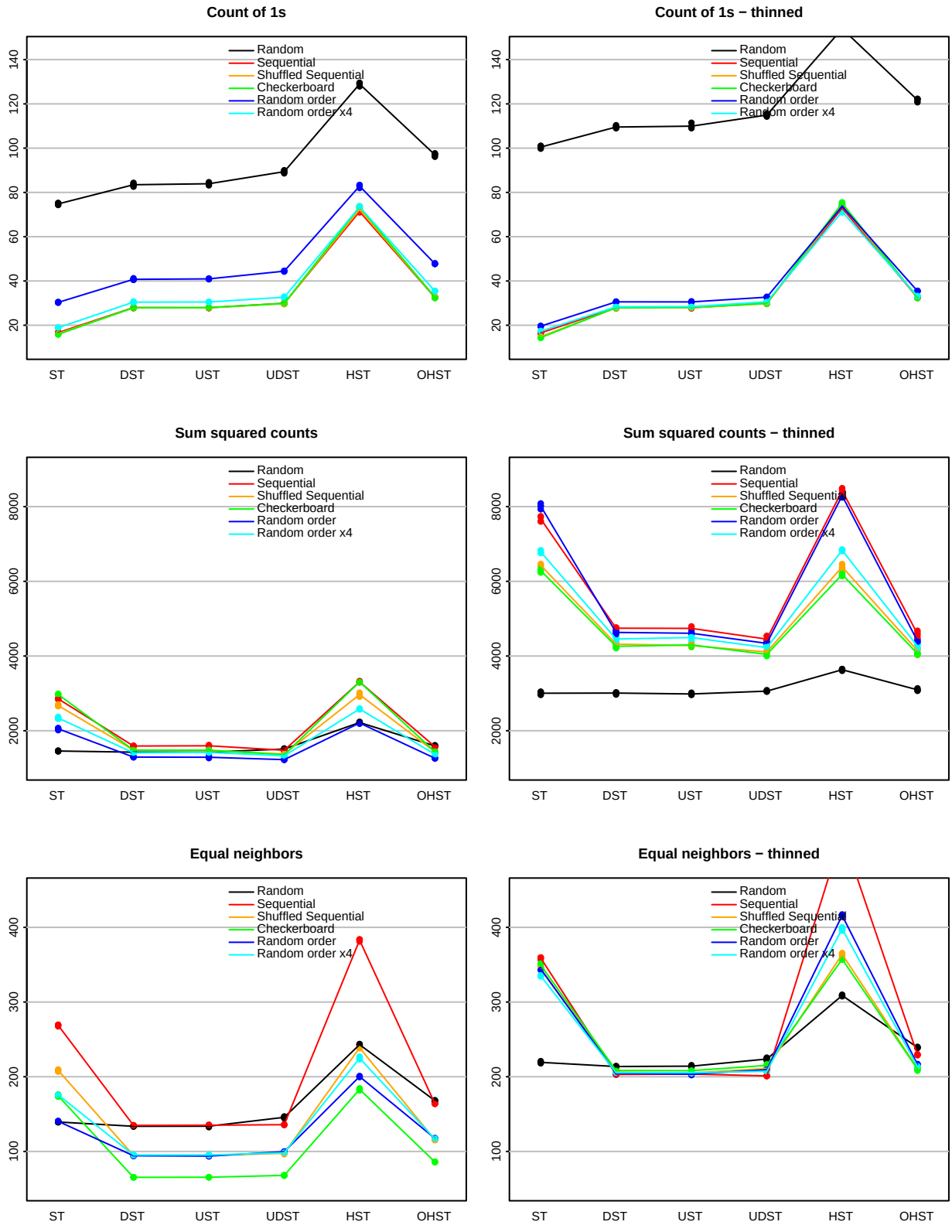


Figure 22: Summaries of autocovariance function estimates for the 5×5 Potts model, for the second group of methods.

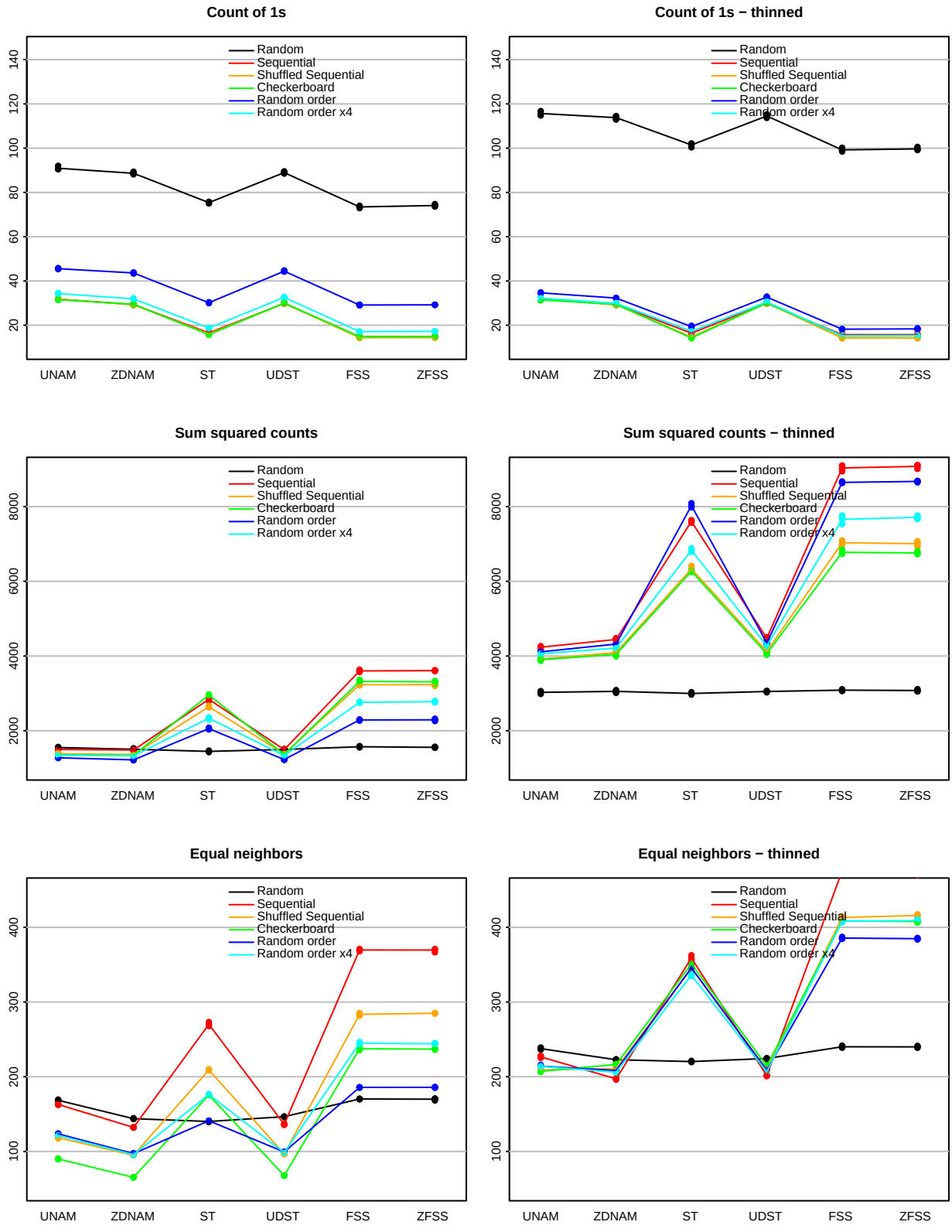


Figure 23: Summaries of autocovariance function estimates for the 5×5 Potts model, for the third group of methods.

Gibbs sampling, an unusual occurrence for practical problems. A similar but less pronounced effect is visible for the "equal neighbors" function.

I speculate that combining a scan other than random selection of a variable with a method other than Gibbs sampling (one having an antithetic aspect) can induce somewhat periodic movement, which when sampled only every n iterations can produce inefficient estimates. One would usually expect this to occur only for fairly easy problems, such as this one. For difficult problems, one expects that many scans will be needed to move to an almost independent state, and autocovariances for most functions of interest will be strongly positive. A modification to Gibbs sampling that introduces antithetic aspects would then only be expected to somewhat reduce the magnitude of these autocovariances, not make them negative. Thinning would then behave more in the expected way.

The shifted tower methods (Figure 22) and slice sampling methods (FSS and ZFSS in Figure 23) show the same surprising behaviours. In addition, the ST, HST, FSS, and ZFSS methods show large variation in performance. For the "count of 1s" function, the ST, FSS, and ZFSS methods have nearly the same asymptotic variance for all scan orders, which is lower than that of all other methods. However, for the other two functions, these methods perform very poorly. The ZDNAM, DST, UST, and UDST methods show the best overall performance, with OHST behaving similarly, but with slightly higher asymptotic variance.

Note that the erratic ST, HST, FSS, and ZFSS methods are the ones that often have some zero non-self transition probabilities, and that also use a fixed ordering of values, so these zero transition probabilities may apply consistently. In some circumstances, this could be beneficial, but from these results, it seems it can also have bad effects. As discussed in Section 11, ZFFS was deliberately designed to preserve this order as much as possible, but in light of these results, it might be interesting to design a slice sampling method in which the values do not keep the same order.

Pollet, *et al.* (2004) have also compared Gibbs sampling with MHGS and UNAM,[23] for a 4×4 Potts model, with random selection of the variable to be updated, and also report that UNAM performs significantly better than Gibbs sampling at estimating the expectation of the energy, and that MHGS is only slightly worse than UNAM. They did not consider sequential updates of variables, or functions of state other than the energy.

Another comparison of methods on the Potts model was done by Suwa (2022), who compared GS, MHGS, UNAM, ST, HST, and other shifted tower methods in which the amount shift varies from 0 to 1/2 (with corresponding variation in self transition probability).[24] Suwa considers Potts models with m (their q) equal to 2, 3, 4, 5, and 6, with the temperature set to the value corresponding to a phase transition in an infinite lattice. For $m = 4$, this corresponds to choosing $b = 1.098$ in equation (120). They used $R = C = 32$, so $n = 1024$, and updated variables in a fixed sequential order, which was not specified, but presumably corresponded to a simple scan across and down the lattice (corresponding to what is labeled as "Sequential" in Figures 15 through 17). Suwa evaluated methods by their "integrated autocorrelation time", which is proportional to asympotic variance, of an "order parameter".

Suwa's results show that MHGS is substantially better than GS, and that UNAM is only slightly better than MHGS, in agreement with the results of Pollet (2004), and the results reported here for the 8×8

[23] They refer to GS as "heatbath", to MHGS as "MG", and to UNAM as "Opt".

[24] Suwa refers to GS as "heatbath", MHGS as "Metropolized Gibbs", UNAM as "iterative Metropolized Gibbs", and ST as the "Suwa-Todo algorithm"; other shifted tower methods were characterized by the shift amount (with $s = 1/2$ corresponding to HST).

Potts model. Suwa also shows a substantial advantage of ST over UNAM, again in agreement with results here.

A larger claim by Suwa is that the autocorrelation time is an exponential function of the frequency of non-self transitions — equivalently, that the log of the autocorrelation time (or asymptotic variance) is a linear function of the frequency of non-self transitions, as pictured in Fig. 2 of (Suwa 2022). This figure shows results for shifted tower methods in which the amount of shift is varied from just above 0 to 1/2 (with the latter value corresponding to HST), with a consequent variation in self transition probability from just below 1 to the minimum possible. The results obtained are fit reasonably well by a linear relationship of log autocorrelation time to non-self transition probability. Furthermore, the results with GS, MHGS, UNAM, and ST (with shift not constant, but equal to the maximum probability) are also close to this line.

This claim seems misleading, however. First, note that the non-self transition probability is upper bounded by a value no greater than one, so an exponential improvement as it increases does not permit arbitrarily large improvements in autocorrelation time. Second, the autocorrelation time must go to infinity as the non-self transition probability goes to zero, so the exponential relationship cannot hold in this limit. One may question whether the experimental results with the smallest non-self transition probabilities are accurate, considering the difficulty of estimating autocorrelation times when they are very large. The alternative of autocorrelation time being proportional to some power of the non-self transition probability is almost indistinguishable from an exponential relationship over the range of non-self transition probabilities for which the fit of the latter is good in Suwa's Fig. 2, which is from 0.23 to 0.29 for $q = 4$.

Suwa also compares a sequential scan with a random scan, with results shown in Fig. 4 of (Suwa 2022), which appears to be for $q = 4$ (though this is not stated). For the ST method, the sequential scan is a factor of about 3.5 more efficient than a random scan, similar to, though a bit greater, than the advantage seen here in Figure 16. These results are seen by Suwa as following a relationship in which the autocorrelation time with a random scan is proportional to a power of the non-self transition probability. While this is more plausible than an exponential relationship for small non-self transition probabilities, I think that further research is needed to elucidate these relationships. The results for the 5×5 Potts model here show that methods with the same self transition probability (including those that minimize it) can have substantially different efficiencies (for example, ST, UDST, and HST in Figure 22).

16 Comparisons for a Bayesian mixture model

Mixture models are commonly used for data that comes from several distinct sources, for example, data on symptoms of patients suffering from different diseases. In a Bayesian modeling approach (Neal 1992a), the parameters of the mixture model are integrated over, with respect to a prior distribution, leaving as the only unknowns which component of the mixture is associated with each data point (e.g., which disease each patient has). Sampling for these component indicators can be done by Gibbs sampling, which can be modified to avoid self transitions by the methods discussed in this paper.

A mixture model for independent observations $y_1, \ldots, y_n$ represents their distribution as a mixture of m component distributions, as follows:

$$P(y_i|\alpha,\theta) \;=\; \sum_{x_i=1}^{m} \alpha_{x_i} P(y_i|x_i,\theta_{x_i}) \tag{121}$$

Here, x_i indicates which mixture component is associated with observation y_i, $\alpha = [\alpha_1, \ldots, \alpha_m]$ is a vector of mixture weights, with $\sum_x \alpha_x = 1$, and θ_x gives the parameters of mixture component x. For the model

used in the experiments here, each observation consists of H binary variables, $y_i = [y_{i1}, \ldots, y_{iH}]$ with $y_{ih} \in \{0, 1\}$, and θ_x contains the probabilities for each of these binary variables to have the value 1, so $\theta_x = [\theta_{x1}, \ldots, \theta_{xH}]$ with $\theta_{xh} \in [0, 1]$. Conditional on observation i coming from mixture component x_i, the H binary variables are assumed to be independent, so

$$P(y_i|x_i, \theta) = \prod_{h=1}^{H} \theta_{x_i h}^{y_{ih}} (1-\theta_{x_i h})^{1-y_{ih}} \tag{122}$$

The joint probability of all observations, y_i, along with all component indicators, x_i, for given values of the model parameters α and θ, is therefore

$$P(y_1, \ldots, y_n, x_1, \ldots, x_n|\alpha, \theta) = \prod_{i=1}^{n} \alpha_{x_i} \prod_{h=1}^{H} \theta_{x_i h}^{y_{ih}} (1-\theta_{x_i h})^{1-y_{ih}} \tag{123}$$

$$= \left[\prod_{x=1}^{m} \alpha_x^{C_x}\right] \left[\prod_{x=1}^{m} \prod_{h=1}^{H} \theta_{xh}^{S_{xh}} (1-\theta_{xh})^{C_x - S_{xh}}\right] \tag{124}$$

where C_x is the number of x_i for $i = 1, \ldots, n$ that are equal to x, and S_{xh} is the sum of y_{ih} for those i for which x_i equals x.

In a Bayesian treatment of this problem, a prior distribution for the unknown parameters α and θ is specified. If in this prior the θ_{xh} and α parameters are independent, with each θ_{xh} uniform over $(0, 1)$ and α uniform over the simplex with $\alpha_x > 0$ and $\sum_x \alpha_x = 1$, it is possible to analytically integrate over the prior for these parameters (Neal 1992a), giving a joint distribution for the observations and component indicators alone:

$$P(y_1, \ldots, y_n, x_1, \ldots, x_n) = \left[\frac{(m-1)!}{(n+m-1)!} \prod_{x=1}^{m} C_x!\right] \left[\prod_{x=1}^{m} \prod_{h=1}^{H} \frac{S_{xh}!\,(C_x - S_{xh})!}{(C_x+1)!}\right] \tag{125}$$

When we have observed $y_1, \ldots, y_n$, we may wish to sample from the conditional distribution of the component indicators $x_1, \ldots, x_n$ given these observations, both because this distribution is of interest in itself (giving possible "clusterings" of the observations), and because it assists inference for the parameters and predictions for future observations. This can be done using Gibbs sampling. The conditional distribution for x_i given x_{-i} can be obtained from equation (16), as

$$P(x_i = x|y_1, \ldots, y_n, x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n) \propto (C_x^- + 1) \prod_{h=1}^{H} \frac{(S_{xh}^- + 1)^{y_{ih}} (C_x^- - S_{xh}^- + 1)^{1-y_{ih}}}{C_x^- + 2} \tag{126}$$

where $C_x^- = C_x - I(x_i = x)$ is the number of x_j for $j \neq i$ that are equal to x, and $S_{xh}^- = S_{xh} - y_{ih} I(x_i = x)$ is the sum of y_{jh} for those $j \neq i$ for which x_j equals x.

The experiments in this section compare use of Gibbs sampling with the modified Gibbs sampling methods, on a problem in which there are $n = 30$ observations, each consisting of $H = 10$ binary variables. The model used has $m = 9$ mixture components. The data, shown in Figure 24, was manually constructed to have five clusters of observations, which would be expected to correspond to mixture components, so we anticipate that several of the mixture components will be associated with few or no observations in typical samples from the posterior distribution

```
 1:  1 1 1 1 0 0 0 0 1 0      8:  0 0 0 0 1 1 1 1 1 0     14:  1 0 1 1 0 0 1 1 0 1
 2:  1 1 1 1 0 0 0 0 0 0      9:  0 0 0 0 1 1 1 1 1 0     15:  0 0 1 1 0 0 1 1 1 1
 3:  1 1 1 1 0 0 0 0 1 0     10:  0 0 0 0 1 1 1 1 1 1     16:  0 0 1 1 0 0 1 1 1 0
 4:  1 0 1 1 0 0 0 0 1 0     11:  0 0 0 1 1 1 1 1 0 0     17:  0 0 1 1 0 1 1 1 1 0
 5:  1 1 1 1 0 0 0 0 0 1     12:  0 0 0 0 0 1 1 1 1 1     18:  0 0 1 1 0 0 1 1 0 0
 6:  1 1 1 1 0 0 1 0 1 1     13:  0 0 1 0 1 1 1 0 1 0     19:  0 0 1 1 0 0 1 1 0 1
 7:  0 1 1 1 0 0 0 0 0 0

20:  1 1 0 0 1 1 0 0 0 0     26:  1 0 0 0 1 0 0 0 0 0
21:  1 1 0 0 1 1 0 0 1 1     27:  0 0 0 0 0 1 0 0 0 1
22:  1 1 0 0 1 1 0 0 1 0     28:  0 0 0 1 0 0 0 0 0 0
23:  1 1 0 0 1 1 0 0 0 1     29:  0 1 0 0 0 0 0 0 1 0
24:  1 1 1 0 1 1 0 0 1 1     30:  0 0 0 0 0 0 1 0 0 0
25:  1 1 0 0 1 1 0 0 1 0
```

Figure 24: The $n = 30$ observations used for the mixture model example. The observations are here grouped by the five manually-created clusters. The order is randomized in the runs done, so this "true" clustering does not affect the results.

The expectations of the following functions of state were estimated:

1) **Obs 1 in cluster 1.** The indicator function for whether $x_1 = 1$. Since the mixture components (clusters) are treated symmetrically in the model, the true expectation of this function must be $1/m = 1/9 = 0.111111$, and its variance must be $(1/m)(1-1/m) = 8/81 = 0.098765$.

2) **Obs 10 cluster size.** The number of observations in the cluster associated with observation 10 — that is, $\sum_{i=1}^{30} I(x_i = x_{10})$. The expectation of this function is approximately 5.56 and its variance is approximately 3.26.

3) **Obs 30 cluster size.** The number of observations in the cluster associated with observation 30. The expectation of this function is approximately 4.35 and its variance is approximately 6.38.

Each run consisted of $K = 200000$ scans, each with $n = 30$ updates component indicators. For each group of methods, four independent runs of this length were done, for each method in the group, and each scan order.

The frequencies of self transitions for the various methods are as follows:

GS: 0.69, MHGS: 0.65, UNAM: 0.64, DNAM: 0.61, UDNAM: 0.62, FSS: 0.61

ZDNAM, ST, DST, UST, UDST, HST, OHST, ZFSS: 0.61

The maximum conditional probability for an update was half or more 86% of the time.

Summaries of asymptotic variance estimates for the three function above, for all groups of methods, are shown in Figures 25 through 27. Note that there is no meaningful original order for the variables, so there is no "sequential" scan order.

The results for the mixture model problem are qualitatively similar to those for the 8×8 Potts model. The shuffled sequential scan order gives the best results. Thinning increases asymptotic variance, except for the random scan, for which thinning is beneficial. Amongst the methods deriving from Gibbs sampling (Figure 25), ZDNAM performs best. The shifted tower methods (Figure 26) all perform about equally well, except that HST may be slightly worse than the others. FSS and ZFSS (see Figure 27) also perform well.

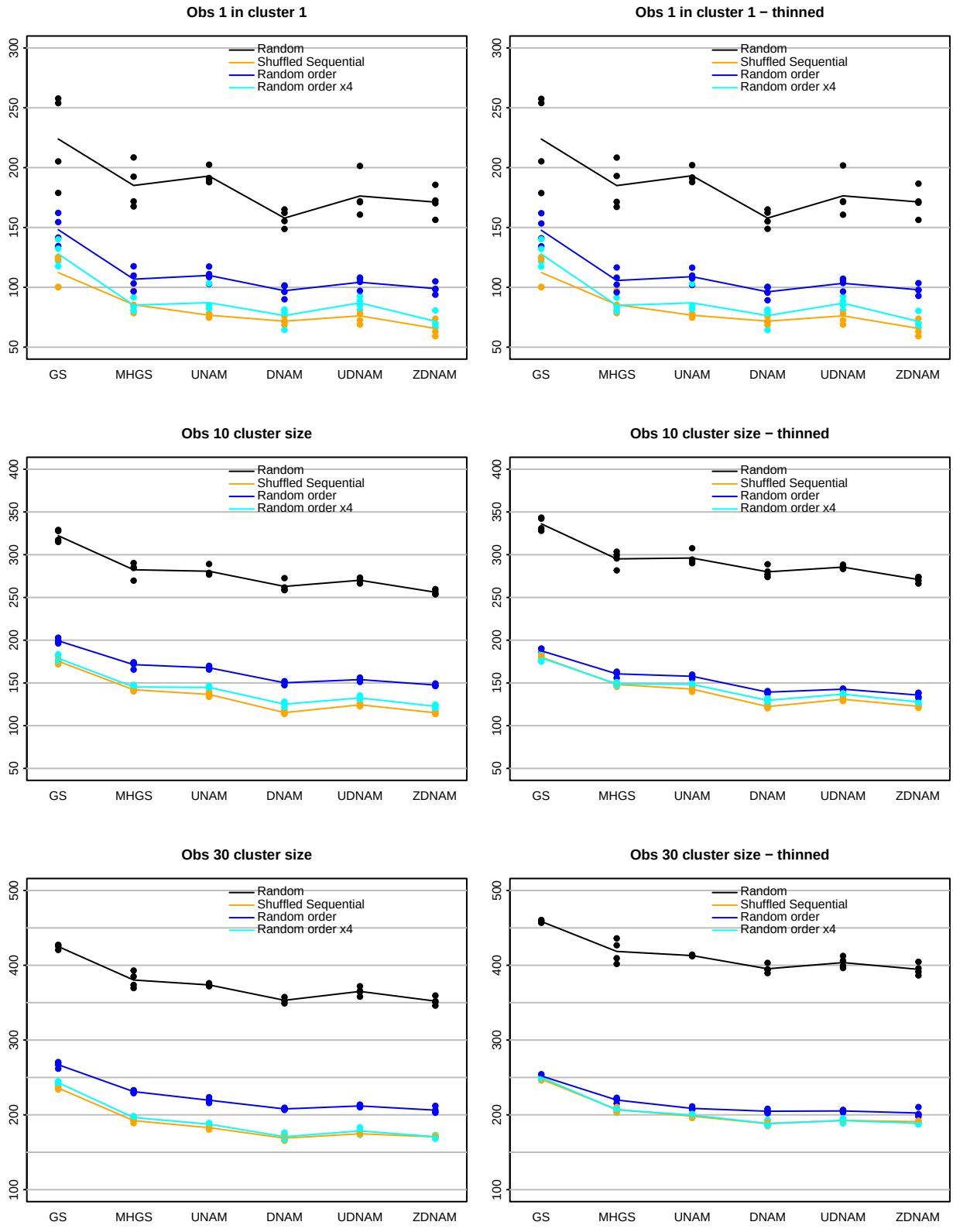


Figure 25: Summaries of autocovariance function estimates for the Bayesian mixture model, for the first group of methods.

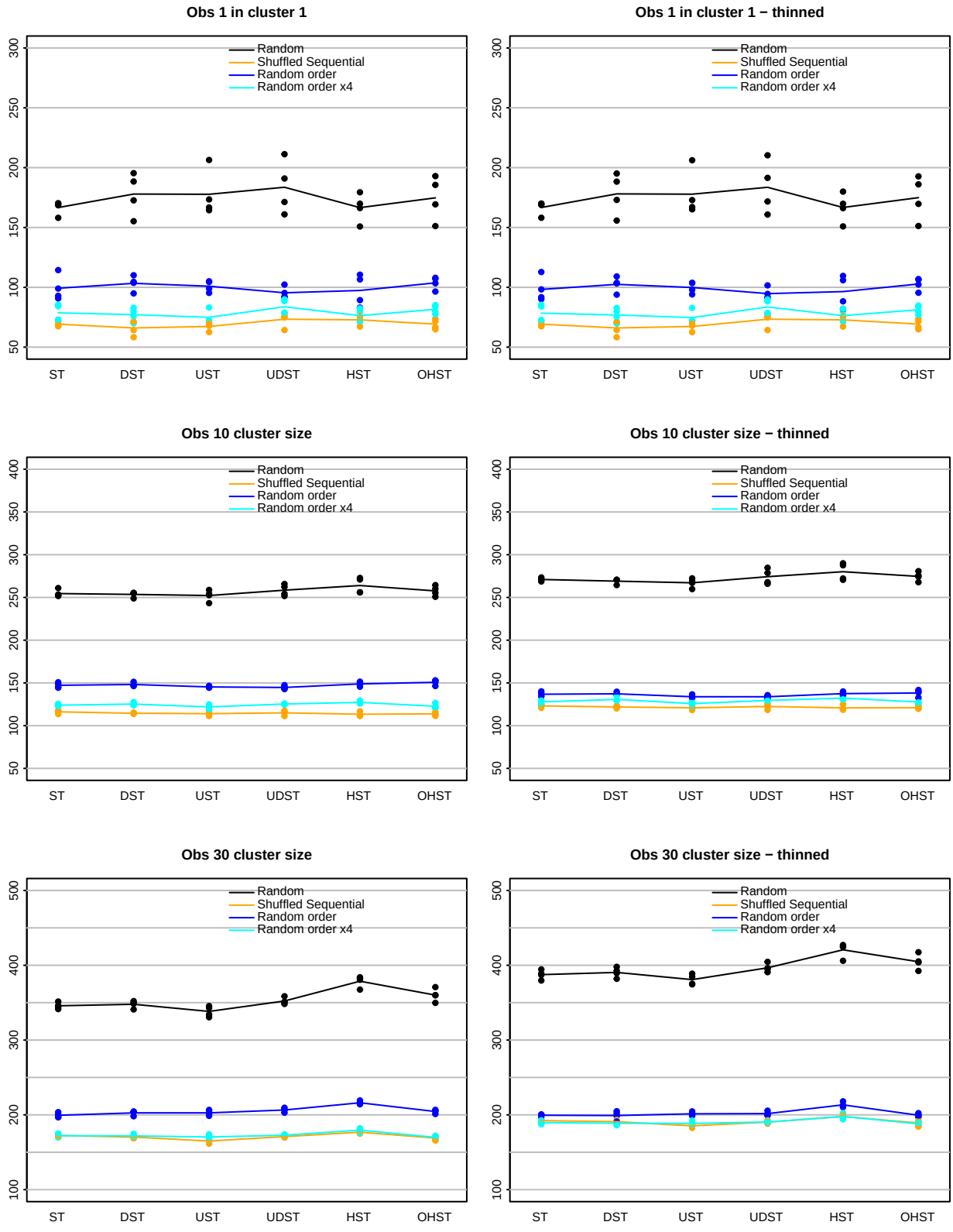


Figure 26: Summaries of autocovariance function estimates for the Bayesian mixture model, for the second group of methods.

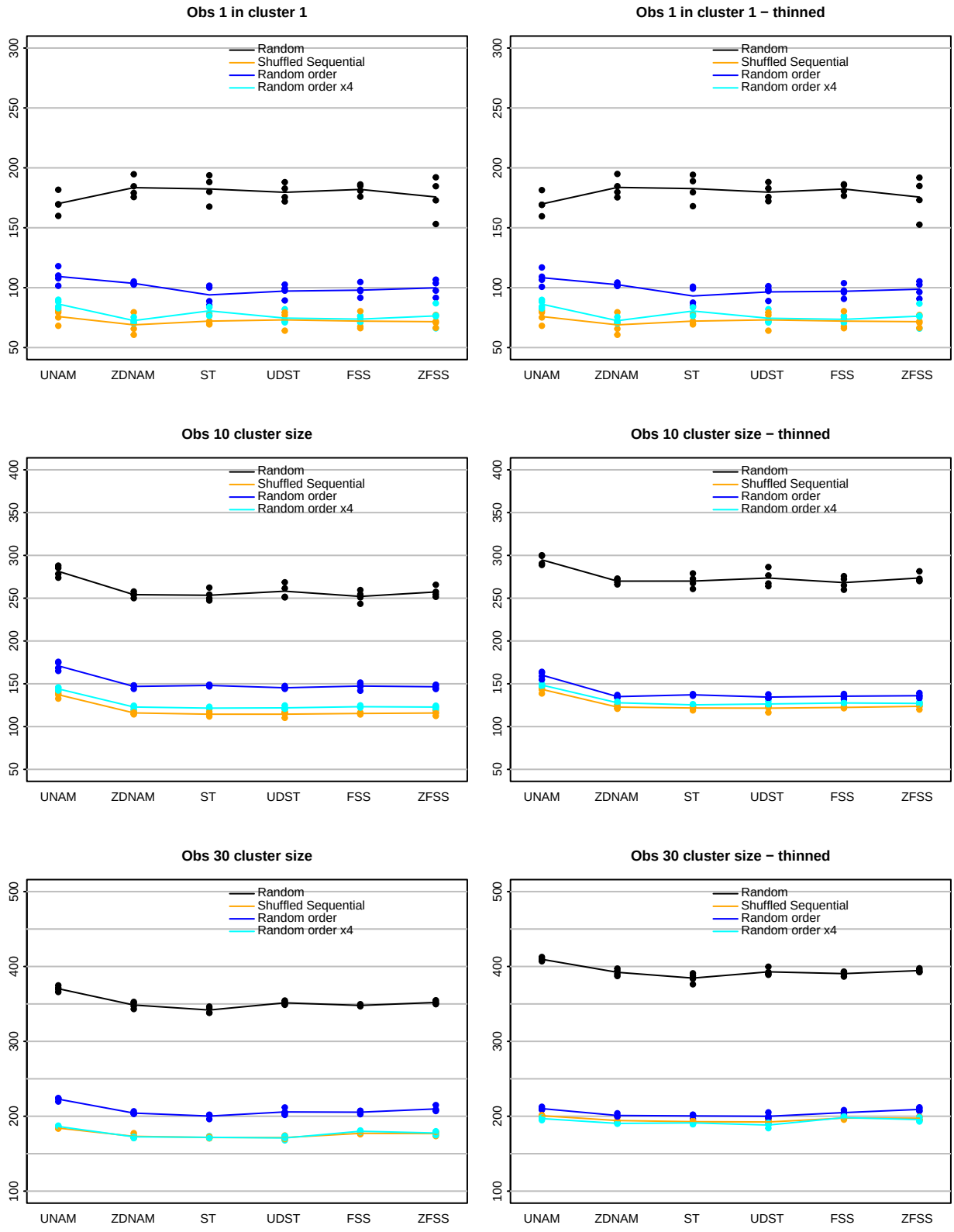


Figure 27: Summaries of autocovariance function estimates for the Bayesian mixture model, for the third group of methods.

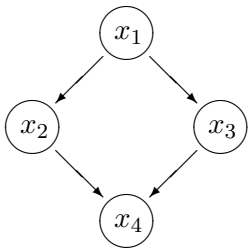
17 Comparisons for a belief network

A joint distribution for random variables $x_1, \ldots, x_n$ can be written as a product of successive conditional distributions:

$$\pi(x) \quad = \quad \pi(x_1)\,\pi(x_2|x_1)\,\pi(x_3|x_1,x_2)\cdots\pi(x_n|x_1,\ldots,x_{n-1}) \tag{127}$$

A belief network (sometimes called a "Bayesian network" or "directed graphical model") is a directed graph with arrows that go from some x_i to some x_j with $j > i$, which summarizes how the representation above can be simplified by omitting some conditioning variables — in the factor $\pi(x_j|\ldots)$, the only x_i that need be conditioned on are those (the "parents" of x_j) for which there is an arrow from x_i to x_j in the network.

Consider, for example, the network below:



The absence of arrows from x_1 to x_4 and from x_2 to x_3 means that the joint distribution can be written as

$$\pi(x) \quad = \quad \pi(x_1)\,\pi(x_2|x_1)\,\pi(x_3|x_1)\,\pi(x_4|x_2,x_3) \tag{128}$$

Many common statistical models (e.g., state space time series models) can be seen as belief networks. They have also been used to represent knowledge elicited from experts (Pearl 1988; Lauritzen and Spiegelhalter 1988), and as models in the style of neural networks that can be learned from data (Neal 1992b).

If some variables in a belief network are known, the conditional distribution of the other variables given these known variables can be sampled from using Gibbs sampling. The unknown variables are updated in some systematic or random order. An update to variable x_i is done by sampling a new value from its conditional distribution given current values of other variables, including both ones with known values (which are fixed) and the current values of other unknown variables.

The conditional distribution for x_i needed for Gibbs sampling depends only on the parents of x_i, the children of x_i, and the parents of the children of x_i, with the conditional probabilities being proportional to the product of factors of the joint distribution that involve x_i. For the example network above,

$$\pi(x_1|x_{-1}) \quad \propto \quad \pi(x_1)\,\pi(x_2|x_1)\,\pi(x_3|x_1) \tag{129}$$

$$\pi(x_2|x_{-2}) \quad \propto \quad \pi(x_2|x_1)\,\pi(x_4|x_2,x_3) \tag{130}$$

$$\pi(x_3|x_{-3}) \quad \propto \quad \pi(x_3|x_1)\,\pi(x_4|x_2,x_3) \tag{131}$$

$$\pi(x_4|x_{-4}) \quad \propto \quad \pi(x_4|x_2,x_3) \tag{132}$$

The belief network used for the experiments reported here is shown in Figure 28. The 10 variables in the model, represented as circles in the network, are arranged in three layers — a layer at the top of two variables (each with five possible values), a layer of five variables in the middle (each with four possible values), and a layer of three variables at the bottom (each with three possible values). Arrows go from every variable in the top layer to every variable in the middle layer, and from every variable in the middle layer to every variable in the bottom layer.

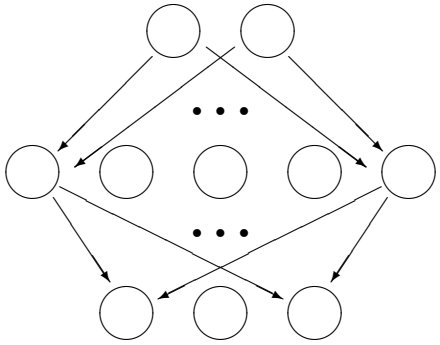


Figure 28: The belief network used for the experiments. The variables represented by the circles at the top have possible values in $\{1, 2, 3, 4, 5\}$, those in the middle have possible values $\{1, 2, 3, 4\}$, and those at the bottom have possible values $\{1, 2, 3\}$.

The marginal distributions for the two variables at the top (which have no parents) are determined by parameters $\alpha_{i,u}$, where $i \in \{1, 2\}$ identifies the variable and $u \in \{1, 2, 3, 4, 5\}$ is a possible value for the variable. The probability for variable i in this top group having value u is $\exp(\alpha_{i,u}) \,/\, \sum_{u'} \exp(\alpha_{i,u'})$. (Note that this and other parameterizations for this model are redundant, with multiple parameter values producing the same distribution.)

The conditional distribution for a middle variable given values for its parent variables is defined using a multinomial logit model (also known as a "softmax" model). For each possible value, v, of variable j in the middle layer, a summed input, $s_{j,v} = \sum_i \exp(\beta_{ij,x_i v})$ is computed, where $\beta_{ij,uv}$ are parameters giving the influence of variable i having value u on variable j having value v. The probability that variable j has value v is then $\exp(s_{j,v}) \,/\, \sum_{v'} \exp(s_{j,v'})$. In similar fashion, parameters $\gamma_{jk,vw}$ define multinomial logit models for the values of variables in the bottom layer, given values for variables in the middle layer.

For the experiments, a single set of parameters, $\alpha_{i,u}$, $\beta_{ij,uv}$, and $\gamma_{jk,vw}$, were randomly sampled, independently, from the t distribution with four degrees of freedom. Gibbs sampling and the other methods were then used to sample from the distribution for all $n = 2 + 5 + 3 = 10$ variables. This is not the typical usage — one would usually condition on known values for some of the variables — and if one did want to sample from this distribution, it can be done more easily by sampling top down from the conditional distribution of each node given its parents. However, it is a useful test of sampling methods, since moving around the whole unconstrained distribution should be more challenging.

Estimates of the expectations of the following functions of state were found:

1) **Unit 1 of layer 1 is 1.** The indicator that the first variable in the middle layer (1) of variables has the value 1. The expectation of this function is 0.2109 (and consequently its variance is 0.1664).

2) **Unit 1 of layer 2 is 1.** The indicator that the first variable in the bottom layer (2) of variables has the value 1. The expectation of this function is 0.07353 (and its variance is 0.06812).

1) **Unit 1 layer 0 and unit 1 layer 2 is 1.** The indicator that the logical "and" of the first variable of the top layer (0) and the first variable of the bottom layer (2) is 1 (i.e., that both variables are 1). The expectation of this function is 0.04950 (and its variance is 0.04705).

The expectations above were computed by brute-force marginalization over all possible combinations of values for the ten variables.

Four runs with $K = 1000000$ scans, each with $n = 10$ variable updates, were done for each scan order and each method within each group of methods.

The frequencies of self transitions for the various methods are:

GS: 0.68, MHGS: 0.59, UNAM: 0.58, DNAM: 0.56, UDNAM: 0.57, FSS: 0.56

ZDNAM, ST, DST, UST, UDST, HST, OHST, ZFSS: 0.56

The maximum conditional probability for an update was half or more 89% of the time.

Summaries of asymptotic variance estimates for the three function above, for all groups of methods, are shown in Figures 29 through 31. Note that the results for the sequential scan (red) and shuffled sequential scan (orange) are almost identical (with the latter overlaying the former).

The results on the belief network problem are qualitatively quite similar to those for both the 8×8 Potts model and the mixture model. The sequential and shuffled sequential scan orders gives the best results. Thinning increases asymptotic variance, except for the random scan, for which thinning is beneficial. For all scan orders, with and without thinning, there is almost no difference in asymptotic variance between DNAM, ZDNAM, FSS, ZFSS, and the shifted tower methods, all of which are noticeably better than GS, MHGS, UNAM, and UDNAM.

It is not surprising that all the methods minimizing self transition probability have nearly the same performance on this problem. As noted above, the maximum conditional probability for this problem is one half or more 89% of the time, and as shown at the end of Section 13, in such situations any method that minimizes self transition probability must have the same transition probabilities. There is therefore little scope for differences amongst these methods, or with DNAM and FSS, both of which almost minimize self transition probability for this problem.

18 Conclusions

Liu's (1996) MHGS modification of Gibbs sampling and the UNAM method due to Frigessi, Hwang, and Younes (1992) and Tjelmeland (2004) can both be justified as improvements to Gibbs sampling by applying Peskun's (1973) theorem. In this paper, I have introduced a more general class of methods based on nested antithetic modification (NAM), which can also be shown to efficiency-dominate Gibbs sampling, using a more general theory, presented in a companion paper (Neal and Rosenthal 2023). The DNAM method in this class appears in the experimental evaluations to usually be superior to UNAM, though this is not theoretically guaranteed. The ZDNAM modification to DNAM reduces self transitions to the minimum possible, and can also be shown to efficiency-dominate Gibbs sampling, when the variable to update is chosen randomly.

The minimum possible self transition probability can also be achieved with the ST method (Suwa and Todo 2010) and the HST method (Suwa 2022). In this paper, I also consider UST, DST, UDST, and OHST variations on these methods. One can show, using theory developed in (Neal and Rosenthal 2023), that the reversible methods in this class (UDST, HST, and OHST) cannot be efficiency-dominated by any reversible method (within the framework of randomly-selected variable updates). However, unlike ZDNAM, these methods do not always efficiency-dominate Gibbs sampling.

In this paper, I have also introduced two new non-reversible methods based on slice sampling, FSS and ZFSS, with the latter minimizing self transitions.

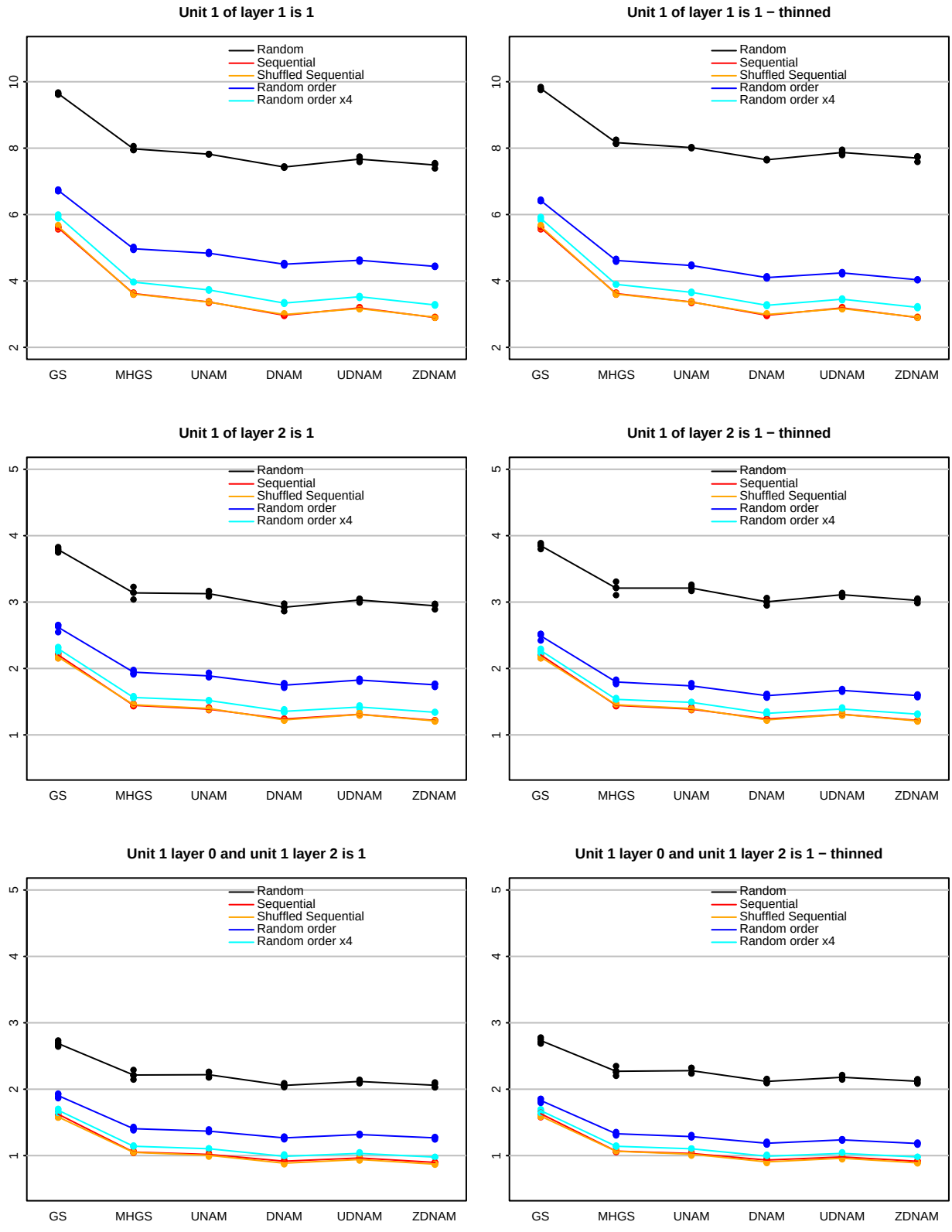


Figure 29: Summaries of autocovariance function estimates for the belief network, for the first group of methods.

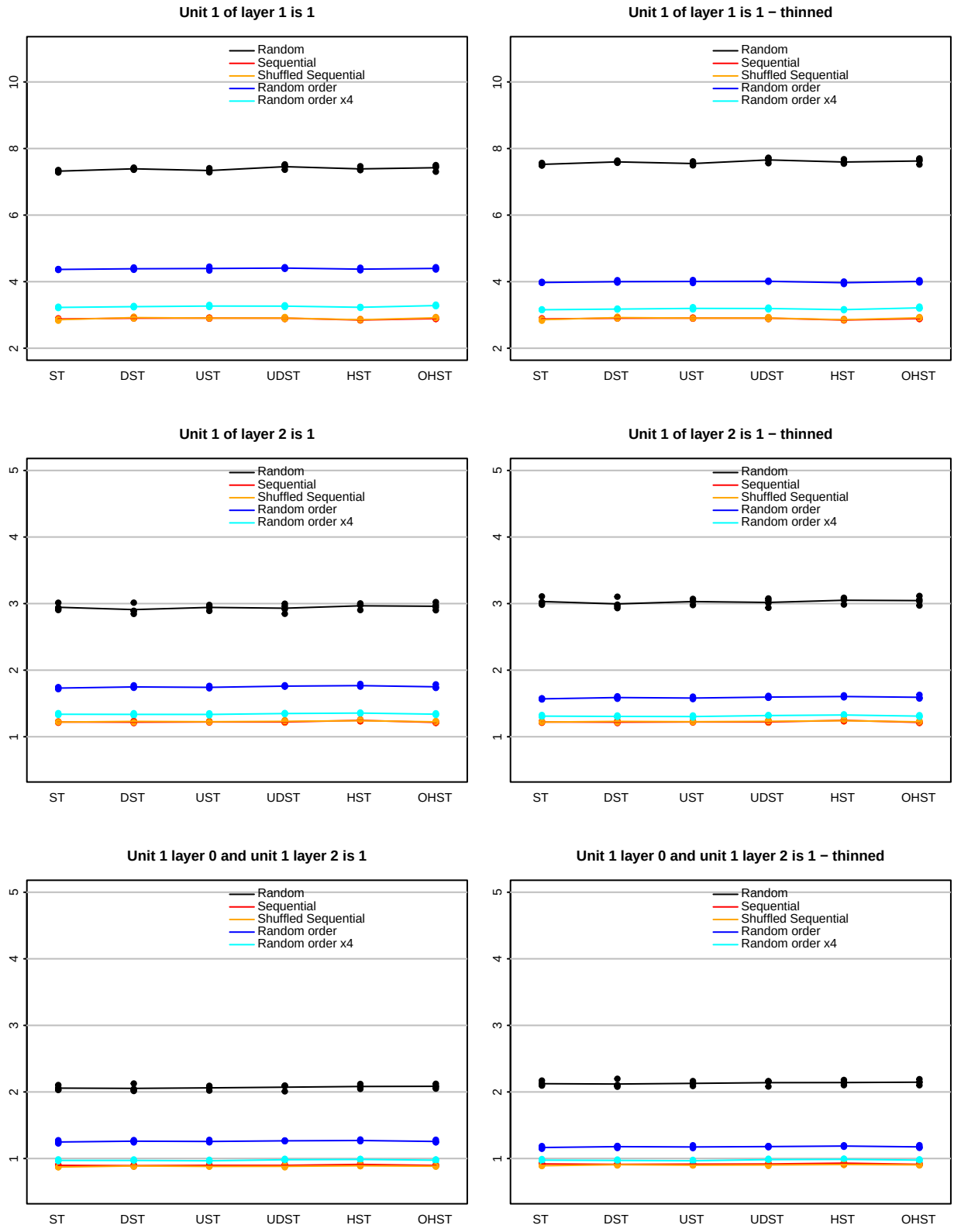


Figure 30: Summaries of autocovariance function estimates for the belief network, for the second group of methods.

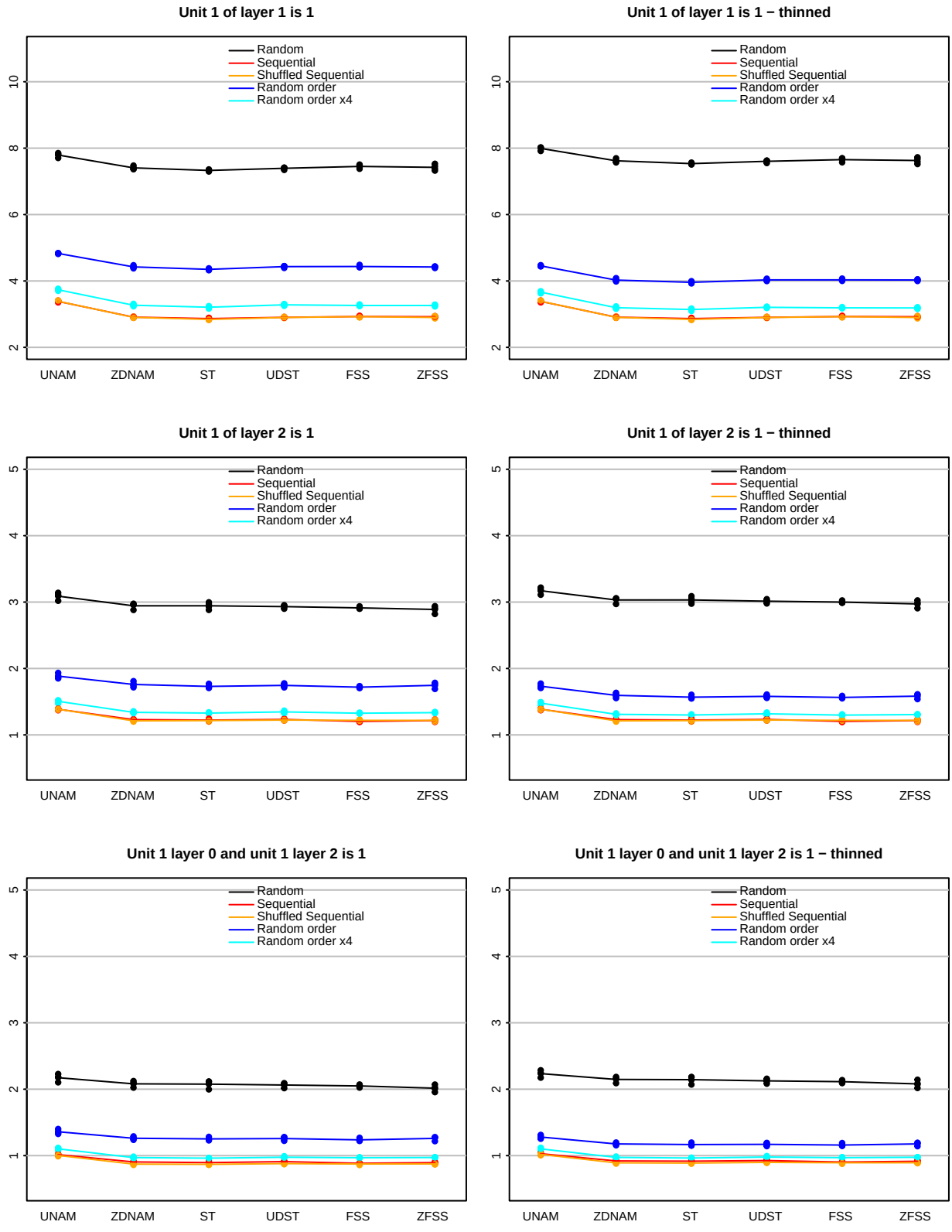


Figure 31: Summaries of autocovariance function estimates for the belief network, for the third group of methods.

The experimental evaluations here show that, with Gibbs sampling and its modifications, random selection of a variable to update is usually (but not quite always) worse than using other scan orders, such as sequential updates. Random updating is necessary for the overall updates to be reversible, when the modification of Gibbs sampling used is reversible. Unfortunately, the theoretical justifications in this paper apply only to reversible methods, so the practical choice of method to use when a non-random scan is used must be largely based on experiment. However, the experiments do show that the relative performance of different methods is usually (but not always) similar for random and systematic scans, so theoretical results for random scans are still of some interest.

On the four problems looked at, the best overall performance was achieved using the DNAM, ZDNAM, DST, UST, UDST, and OHST methods (with DNAM and OHST perhaps being slightly worse than the others). The ST, FSS and ZFSS methods also performed well in most circumstances, but had erratic performance for the 5×5 Potts model with negative b. The problems with these methods, as well as HST, may be due to the zero non-self transition probabilities that they can produce. DST, UST, UDST, and OHST can also have zero non-self transition probabilities, but any bad effect of them may be mitigated by the changing order of values with these methods. The ZDNAM method produces zero non-self transition probabilities only in the context of moving to or from a higher-probability value (as in equation (112)), which seems less problematic. For this reason, I at present recommend ZDNAM as most suitable for general use.

Amongst the methods minimizing self transitions, these experiments give no evidence that a non-reversible update method, such as DST or UST, provides an advantage over reversible methods, such as ZDNAM or UDST. In contrast, using a scan order that leads to the overall method being non-reversible usually has a large advantage. This highlights the need for better theoretical tools for analysing non-reversible methods.

Efficient algorithms for all the methods evaluated are given in this paper, which I hope will facilitate their use in applications and in general-purpose MCMC software. The programs used for the experimental evaluations, written in R, along with the output files, are available at `github.com/radfordneal/gibbsmod`.

References

Ackley, D. H., Hinton, G. E., and Sejnowski, T. J. (1985) "A learning algorithm for Boltzmann machines", *Cognitive Science*, vol. 9, pp. 147–169.

Devroye, L. (1986) *Non-Uniform Random Variate Generation*, Springer-Verlag.

Frigessi, A., Hwang, C.-R., and Younes, L. (1992) "Optimal spectral structure of reversible stochastic matrices, Monte carlo methods and the simulation of Markov random fields", *The Annals of Applied Probability*, vol. 2, pp. 610–628.

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

Geman, S. and Geman, D. (1984) "Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, pp. 721–741.

Geyer, C. J. (1992) "Practical Markov chain Monte Carlo", *Statistical Science*, vol. 7, pp. 473-511.

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

He, B., De Sa, C., Mitliagkas, I., and Ré, C. (2016) "Scan order in Gibbs sampling: Models in which it matters and bounds on how much", `https://arxiv.org/abs/1606.03432`

Hoffman, M. D. and Gelman, A. (2014) "The No-U-Turn Sampler: Adaptively setting path lengths in Hamiltonian Monte Carlo", *Journal of Machine Learning Research*, vol. 15, pp. 1593-1623.

Horn, R. A. and Johnson, C. R. (2013) *Matrix Analysis*, 2nd edition, Cambridge University Press.

Landau, D. P. and Binder, K. (2009) *A Guide to Monte Carlo Simulations in Statistical Physics*, Third Edition, Cambridge University Press.

Lauritzen, S. L. and Spiegelhalter, D. J. (1988) "Local computations with probabilities on graphical structures and their application to expert systems" (with discussion), *Journal of the Royal Statistical Society B*, vol. 50, pp. 157-224.

Liu, J. S. (1996) "Peskun's theorem and a modified discrete-state Gibbs sampler", *Biometrika*, vol. 83, pp. 681–682.

Mira, A. and Geyer, C. J. (1999), "Ordering Monte Carlo Markov chains". Technical Report No. 632, School of Statistics, University of Minnesota.

Neal, R. M. (1992a) "Bayesian mixture modelling", in C. R. Smith, G. J. Erickson, and P. O. Neudorfer (editors) *Maximum Entropy and Bayesian Methods: Proceedings of the 11th International Workshop on Maximum Entropy and Bayesian Methods of Statistical Analysis, Seattle 1991*, pp. 197-211, Dordrecht: Kluwer Academic Publishers.

Neal, R. M. (1992b) "Connectionist learning of belief networks", *Artificial Intelligence*, vol. 56, pp. 71-113.

Neal, R. M. (2003) "Slice sampling" (with discussion), *Annals of Statistics*, vol. 31, pp. 705-767.

Neal, R. M. (2004) "Improving asymptotic variance of MCMC estimators: Non-reversible chains are better", `https://arxiv.org/abs/math/0407281`

Neal, R. M. and Rosenthal, J. S. (2023) "Efficiency of reversible MCMC methods: Elementary derivations and applications to composite methods", `https://arxiv.org/abs/2305.18268` (revised version of March 2024).

Pearl, J. (1988) *Probabilistic Reasoning in Intelligent System: Networks of Plausible Inference*, San Mateo, California: Morgan Kaufmann.

Peskun, P. H. (1973) "Optimum Monte-Carlo sampling using Markov chains", *Biometrika*, vol. 60, pp. 607–612.

Pollet, L., Rombouts, S. M.A., Van Houcke, K., and Heyde, K. (2004) "Optimal Monte Carlo updating", `https://arxiv.org/abs/cond-mat/0405150`

Suwa, H. (2022) "Reducing rejection exponentially improves Markov chain Monte Carlo sampling", `https://arxiv.org/abs/2208.03935`

Suwa, H. and Todo S. (2010) "Markov chain Monte Carlo method without detailed balance", https://arxiv.org/abs/1007.2262

Thomas, A., Spiegelhalter, D. J., and Gilks, W. R. (1992) "BUGS: A program to perform Bayesian inference using Gibbs sampling", in J. M. Bernardo, J. O. Berger, A. P. Dawid, and A. F. M. Smith (editors), *Bayesian Statistics 4*, pp. 837–842, Oxford University Press.

Tjelmeland, H. (2004) "Using all Metropolis-Hastings proposals to estimate mean values", Statistics Preprint No. 4/2004, Norwegian University of Science and Technology.