Sankhyā: The Indian Journal of Statistics
1996, Volume 58, Series A, Pt. 2, pp. 225-242

A MARKOV CHAIN MONTE CARLO METHOD FOR GENERATING RANDOM (0, 1)-MATRICES WITH GIVEN MARGINALS

By A. RAMACHANDRA RAO
RABINDRANATH JANA*
and
SURAJ BANDYOPADHYAY
Indian Statistical Institute

SUMMARY. In this paper we give an MCMC method using switches along alternating cycles for generating random (0, 1)-matrices with given marginals and examine its empirical performance in several examples.

1. INTRODUCTION

The problem of generating a random (0, 1)-matrix with given marginals occurs in different contexts. We mention two instances.

Wilson (1987) considered the incidence and non-incidence of 56 species of birds in 28 islands of Vanuatu. The data can be represented by a 56×28 matrix $\mathbf{A}$ where the (i, j)-entry is 1 or 0 according as the i-th species is present on the j-th island or not. He wanted to test the null hypothesis that the occurrence of the species on the islands is random in the sense that all possible incidence matrices are equally likely, using cooccurrences of species. Since the chance of occurrence of a species on an island depends on how numerous the species is and on the capacity of the island which are indicated by the row total of $\mathbf{A}$ corresponding to the species and the column total of $\mathbf{A}$ corresponding to the island, he assumed that only those (0, 1)-matrices which have the same row sums as those of $\mathbf{A}$ and the same column sums as those of $\mathbf{A}$ are possible. To estimate the distribution of any statistic by simulation, one has to generate a random (0, 1)-matrix with the given marginals.

Another instance where one has to generate a random (0, 1)-matrix with given marginals occurs in the analysis of social networks. A social network is

Paper received. June 1995; revised December 1995.

AMS subject classification. 05C20, 05C80, 05C90, 15A52, 60J10, 62E25.

Keywords and phrases.. MCMC, simulation, random (0, 1)-matrices, switching along alternating rectangle.

* Work of R. Jana supported by a fellowship from ODA, UK and HFCL, India.

essentially a digraph, the nodes being the actors, an arc being drawn from i to j if i is related to j according to the particular relation being studied. The digraph or social network can also be represented by its adjacency matrix $\mathbf{A}$ defined as follows: Let the number of nodes be n. Then $\mathbf{A}$ is the $(0,1)$-matrix of order $n \times n$ with $a_{ij} = 1$ or 0 according as (i,j) is an arc or not. Note that $\mathbf{A}$ is now a square matrix and the diagonal elements of $\mathbf{A}$ are 'structurally zero'. Also here, the i-th row sum of $\mathbf{A}$ and the i-th column sum of $\mathbf{A}$ represent the 'expansiveness' and the 'popularity' of the i-th actor. Hence one can consider the null model that the observed (network as represented by its) adjacency matrix is a random $n \times n$ $(0,1)$-matrix with structurally zero diagonal and with given marginals. One may be interested in estimating the distribution of a statistic (like the number $\sum_{i<j} a_{ij}a_{ji}$ of reciprocal pairs) under the null model.

Generating a random $(0,1)$-matrix when both marginals are fixed seems to be a difficult problem. Note that, in contrast, if, only the row sums $r_1, r_2, \ldots, r_m$ of an $m \times n$ $(0,1)$-matrix are fixed, the number of matrices is $\prod_{i=1}^{m} \binom{n}{r_i}$ and it is easy to generate a random matrix since all one has to do is to choose r_i cells at random from the i-th row and put 1's there, the choices from different rows being made independently. When both marginals are given, some recursive methods have been given by Sukhatme (1938) and Katz and Powell (1954) for finding the number of matrices. But they are of no practical use even for moderate n like $n = 20$ since the number of matrices can be astronomical and the amount of computational work so huge that even modern computers cannot handle it. However, Snijders (1991) bypassed the problem of generating a random matrix and gave a way of generating $(0,1)$-matrices with given marginals and with computable probabilities. He could then estimate the distribution of any statistic using a ratio estimator. More recently, Pramanik (1994) gave a method for generating an approximately random $(0,1)$-matrix with given marginals but his method does not seem to work well for small orders.

In this paper we present a Markov Chain Simulation method to generate a random $(0,1)$-matrix with given marginals which is applicable when either (i) there are no structural 0's, or (ii) the matrix is square with structurally zero diagonal. We shall refer to these problems as Problem I and Problem II. Our main interest is in Problem II, particularly in estimating the distribution of the number s of reciprocal pairs though the method can be used for estimating the distribution of any statistic. Besag and Clifford (1989) presented a similar method for Problem I, though the details regarding the way the method is implemented differ considerably. Somewhat similar ideas have also been used by several others including Diaconis and Sturmfels (1995) and Holst (1995). A different type of Markov Chain Simulation method for generating random regular undirected graphs was given by Jerrum and Sinclair (1988).

2. MARKOV CHAIN SIMULATION METHOD

We denote by $\mathfrak{C}$ the class of all $(0,1)$-matrices with given row sums $r_1, r_2, \ldots, r_m$ and given column sums $c_1, c_2, \ldots, c_n$. If we are interested in generating a random network we take $m = n$ and impose the further condition that the diagonal entries are all 0. $\mathbf{A}$ will usually denote an element of $\mathfrak{C}$. Our simulation method uses the concept of an *alternating rectangle* defined as follows:

Definition. An *alternating rectangle in* $\mathbf{A}$ is a set of four distinct cells of the type $\{i_1j_1, i_1j_2, i_2j_2, i_2j_1\}$ in $\mathbf{A}$ such that the entries in the four cells are alternately 0's and 1's as one goes around the rectangle in either direction and the 0's are not structural 0's.

By *switching along* an alternating rectangle in $\mathbf{A}$, we mean interchanging 0's and 1's in the four cells forming the alternating rectangle. It is easy to see that switching along an alternating rectangle of a matrix $\mathbf{A} \in \mathfrak{C}$ gives another matrix $\mathbf{B}$ in $\mathfrak{C}$. It is also clear that the same four cells form an alternating rectangle in $\mathbf{B}$ also and by switching along it in $\mathbf{B}$ we can get back to $\mathbf{A}$. For example, the following two matrices can be obtained from each other by switching along the alternating rectangle $\{12, 14, 24, 22\}$:

$$\begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix}, \quad \begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}$$

The idea of switching along alternating rectangles is quite old. This has been used by Ryser (1963) to study classes of $(0,1)$-matrices and, with a slight modification, by Diaconis and Sturmfels (1995) and others to study contingency tables. Also, for problem I, it is well known that any matrix in $\mathfrak{C}$ can be obtained from any other by a finite sequence of switches along alternating rectangles, see for example, Ryser (1963). However, we give a proof of the following theorem, including the bound on the number of switches required, for the sake of completeness.

Theorem 1. *Let* $\mathbf{A}$ *and* $\mathbf{B}$ *be matrices belonging to* $\mathfrak{C}$ *and let there be no structural 0's. Then* $\mathbf{B}$ *can be obtained from* $\mathbf{A}$ *by a sequence of* t *or less switchings along alternating rectangles where* t *is the minimum of the number of 0's and the number of 1's in* $\mathbf{A}$.

Proof. If $\mathbf{A} = \mathbf{B}$ there is nothing to prove. So let $\mathbf{A} \neq \mathbf{B}$. Then the *Hamming distance* d between $\mathbf{A}$ and $\mathbf{B}$, i.e., the number of cells ij such that $a_{ij} \neq b_{ij}$ is an even positive integer. We now show that we can switch along an alternating rectangle of either $\mathbf{A}$ or $\mathbf{B}$ so that the Hamming distance between the two matrices decreases by at least two. Let $a_{i_1j_1} \neq b_{i_1j_1}$. Without any loss of generality we may take $a_{i_1j_1} = 1$ and $b_{i_1j_1} = 0$. Since the j_1-th column sums of $\mathbf{A}$ and $\mathbf{B}$ are equal, there exists i_2 such that $a_{i_2j_1} = 0$ and $b_{i_2j_1} = 1$. Since the i_2-th row sums of $\mathbf{A}$ and $\mathbf{B}$ are equal, there exists j_2 such that $a_{i_2j_2} = 1$

and $b_{i_2j_2} = 0$. Proceeding like this it is easy to see that, as the process cannot continue indefinitely, there exists an 'alternating cycle' in **A** with entries 1's and 0's, the corresponding entries in **B** being 0's and 1's. Without loss of generality, let $i_1j_1, i_2j_1, i_2j_2, i_3j_2, \ldots, i_kj_k, i_1j_k$ be such an alternating cycle with the minimum length and let $a_{i_1j_1} = 1$. Note that the i's are distinct and the j's are distinct. If $a_{i_1j_2} = 0$ then switching along $i_1j_1, i_2j_1, i_2j_2, i_1j_2$ in **A** gives a matrix **C** whose Hamming distance from **B** is at least two less than the Hamming distance of **A** from **B**. If $b_{i_1j_2} = 1$ we can perform the switch in **B**. These two cases are exhaustive since if $a_{i_1j_2} = 1$ and $b_{i_1j_2} = 0$ then we have a contradiction to the minimality of the length of the alternating cycle as we can replace i_1j_1, i_2j_1, i_2j_2 by i_1j_2 in the earlier alternating cycle to get a shorter one.

Thus by switching along an alternating rectangle of either **A** or **B**, we can reduce the Hamming distance between them by at least two. Since switching is a reversible process, it follows that we can go from **A** to **B** in at most $d/2$ switchings. Since the number of cells ij such that $a_{ij} = 1$ and $b_{ij} = 0$ equals the number of cells ij such that $a_{ij} = 0$ and $b_{ij} = 1$ and this number cannot exceed t, it follows that $d \leq 2t$ and the theorem is proved. ∎

Perhaps it is worth mentioning that the number of steps required is often much less than the bound given in the theorem.

For Problem II, the situation is not always so nice. There are examples, though rather rare, where some matrix in $\mathfrak{C}$ cannot be reached from some other matrix in $\mathfrak{C}$ by switching along alternating rectangles. For example, both the following matrices have all row sums and column sums equal to 1 but one cannot be obtained from the other by switching along alternating rectangles:

$$\begin{bmatrix} \times & 1 & 0 \\ 0 & \times & 1 \\ 1 & 0 & \times \end{bmatrix}, \quad \begin{bmatrix} \times & 0 & 1 \\ 1 & \times & 0 \\ 0 & 1 & \times \end{bmatrix}$$

Here we have denoted structural 0's by crosses. Note that the above matrices do not have any alternating rectangles. However, the off-diagonal entries in each matrix form what we will call a compact alternating hexagon and one can go from one matrix to the other by switching along it.

Definition. A *compact alternating hexagon* in **A** is a set of 6 cells of the type $\{i_1i_2, i_1i_3, i_2i_3, i_2i_1, i_3i_1, i_3i_2\}$ with entries $1, 0, 1, 0, 1, 0$ respectively, where i_1, i_2, i_3 are distinct.

Switching along such a compact alternating hexagon means interchanging 1's and 0's in the six cells. Note that for the compact alternating hexagon in the second matrix above, we may take $i_1 = 1$, $i_2 = 3$ and $i_3 = 2$.

Theorem 2. *Let* **A** *and* **B** *be two distinct matrices belonging to* $\mathfrak{C}$ *corresponding to Problem II. Then* **B** *can be obtained from* **A** *by a sequence of* t *or less switchings along alternating rectangles and compact alternating hexagons, where* t *is the minimum of the number of 1's and the number*

of non-structural 0's in **A**.

Proof. The Hamming distance $d(\mathbf{A}, \mathbf{B})$ is clearly even. We now show that we can switch along an alternating rectangle or a compact alternating hexagon of either **A** or **B** so that the Hamming distance between the two matrices decreases. We will prove this by contradiction. So assume that the distance cannot be decreased by any such switching.

We form a bipartite graph G with vertex set $X \cup X'$ where $X = \{1, 2, \ldots, n\}$ and $X' = \{1', 2', \ldots, n'\}$ as follows: ij' is an edge of G if and only if $i \neq j$ and $a_{ij} \neq b_{ij}$. We also color each edge of G *red* or *blue* thus: edge ij' receives color red if $a_{ij} = 1$ and $b_{ij} = 0$ and color blue if $a_{ij} = 0$ and $b_{ij} = 1$. Since the row sums of **A** and **B** are the same, the number of red edges equals the number of blue edges at any vertex in X. Since the column sums of **A** and **B** are the same, the number of red edges equals the number of blue edges at any vertex in X'.

By an *alternating path in* G we mean a path in which the edges are alternately colored red and blue. By a *diagonal pair* we mean a pair of vertices of G of the type $\{i, i'\}$, $i = 1, 2, \ldots, n$. Note that a diagonal pair of vertices is not adjacent in G. We call an alternating path in G *proper* if all its vertices belong to different diagonal pairs.

We first show that if $i_1 i_2' i_3 i_4'$ is any alternating path in G with $i_1 \neq i_4$ then $i_1 i_4'$ is an edge of G with color the same as that of $i_1 i_2'$. It is clear that i_1, i_2, i_3 and i_4 are distinct. Without loss of generality, let $i_1 i_2'$ be red. If $a_{i_1 i_4} = 0$ then by performing the switch on the matrix **A** along the alternating rectangle $(i_1 i_2, i_1 i_4, i_3 i_4, i_3 i_2)$ we get a matrix **C** with $d(\mathbf{C}, \mathbf{B}) < d(\mathbf{A}, \mathbf{B})$, a contradiction. Thus $a_{i_1 i_4} = 1$. If $b_{i_1 i_4} = 1$ then by performing the switch on the matrix **B** along the alternating rectangle $(i_1 i_2, i_1 i_4, i_3 i_4, i_3 i_2)$ we get a matrix **D** with $d(\mathbf{A}, \mathbf{D}) < d(\mathbf{A}, \mathbf{B})$, a contradiction. Thus $b_{i_1 i_4} = 0$ and so $i_1 i_4'$ is an edge of G with color same as that of $i_1 i_2'$.

We next show that there is no proper alternating path of length more than 2 in G. Suppose there is one such. Let μ be such a path of maximum length. Without loss of generality let $\mu = [i_1, i_2', i_3, i_4', \ldots]$ and $i_1 i_2'$ be red. Then by the statement proved in the preceding paragraph, it easily follows that $i_1 i_2', i_1 i_4', i_1 i_6', \ldots$ are all red edges and $i_2' i_3, i_2' i_5, i_2' i_7, \ldots$ are all blue edges. If the last vertex of μ is i_{2k}' then $k \geq 2$ and there exist at least k blue edges at i_1 and so a blue edge $i_1 i_0'$ with i_0 different from $i_1, i_2, i_3, \ldots, i_{2k}$, a contradiction to the maximality of μ. So let the last vertex of μ be i_{2k+1} where $k \geq 2$. Then there are at least k blue edges at i_1 and, by the maximality of μ, it follows that $i_1 i_3', i_1 i_5', \ldots, i_1 i_{2k+1}'$ are all blue edges. Looking at the path μ in the reverse direction, we similarly get $i_{2k+1} i_{2k}', i_{2k+1} i_{2k-2}', \ldots, i_{2k+1} i_2'$ are all blue edges and $i_{2k+1} i_1', i_{2k+1} i_3', \ldots, i_{2k+1} i_{2k-1}'$ are all red edges. Now the alternating path $i_1 i_3' i_{2k+1} i_2'$ gives a contradiction.

We are now ready to complete the proof of the theorem. Since $\mathbf{A} \neq \mathbf{B}$, there exists a red edge $i_1 i_2'$ and so a blue edge $i_3 i_2'$. Now there exists a red edge at i_3 and by what was proved in the preceding paragraph, it follows that the only red edge at i_3 is $i_3 i_1'$. It follows similarly that the only blue edge at

i_1 is i_1i_3', the only red edge at i_3' is i_2i_3' and the only blue edge at i_2 is i_2i_1'. Thus $\{i_1i_2, i_1i_3, i_2i_3, i_2i_1, i_3i_1, i_3i_2\}$ is a compact alternating hexagon in $\mathbf{A}$ and by switching along it we get a matrix $\mathbf{C}$ such that $d(\mathbf{C}, \mathbf{B}) = d(\mathbf{A}, \mathbf{B}) - 6$, a contradiction.

Thus we can go from any matrix in $\mathfrak{C}$ to any other by a sequence of switches, the distance between the matrices decreasing by at least 2 after every switch. That at most t switches are required follows as in the proof of Theorem 1. This completes the proof of the theorem. ∎

We mention that the idea of alternating cycles (of length 4 or more) is also old and is used in the solution of the *Assignment Problem*. However, we believe our Theorem 2 is new.

From now on we will use the term alternating cycles *to mean (i) alternating rectangles for Problem I and (ii) alternating rectangles and compact alternating hexagons for Problem II.* We mention, however, that for most instances of Problem II, it is enough to consider alternating rectangles.

2.1 *Basic step.* We can now give the basic idea of the Markov Chain Simulation method to generate a random matrix belonging to $\mathfrak{C}$. We start with an initial matrix belonging to $\mathfrak{C}$. At any stage we enumerate the alternating cycles in the current matrix, choose one of them at random and switch along it to get a new matrix in $\mathfrak{C}$. We do this a large number of times. Then one feels that the matrix obtained should be a nearly random matrix (we shall see presently that this is not quite correct). To try to prove this, let us formulate it as a Markov Chain. The states of the Markov Chain are the matrices belonging to $\mathfrak{C}$. Note that $\mathfrak{C}$ is finite. Let $N = |\mathfrak{C}| \geq 2$. We shall say that two states are adjacent if the matrices represented by them can be obtained from each other by switching along one alternating cycle. Let $d(i)$ denote the number of states adjacent to the state i. Then the procedure given above obviously forms a Markov chain with transition probability $p_{ij} = 1/d(i)$ if j is adjacent to i and 0 otherwise. Note that the MC is irreducible since every state can be reached from every other state in a finite number of steps. So, (e.g., Feller, 1960) there exists a unique stationary distribution and, if the MC is aperiodic, the distribution of the state after q steps approaches this stationary distribution as $q \to \infty$, whatever be the initial state. To find the stationary distribution, note that it is a probability vector $\pi' = (\pi_1, \pi_2, \ldots, \pi_N)$ such that $\pi'\mathbf{P} = \pi'$ where $\mathbf{P}$ is the transition probability matrix $((p_{ij}))$. Taking $\theta_i = d(i)/\sum_k d(k)$, it is easy to see that $\sum_i \theta_i p_{ij} = \theta_j$, so $\pi_i = \theta_i$. Thus according to the stationary distribution, the probability of the i-th state is not $1/N$ but is proportional to the number $d(i)$ of alternating rectangles in the matrix corresponding to the i-th state.

2.2 *Modification.* As observed in the preceding paragraph, the basic MCS method given above does not choose the matrices in $\mathfrak{C}$ with equal probabilities and needs modification. Suppose we know an upper bound K for the $d(i)$'s. Then we can modify the basic method as follows: At any stage, if we

are currently in state i, we go to any one of the states adjacent to i, each with probability $1/K$ and remain at the state i itself with probability $1 - d(i)/K$. Then the transition probability p_{ij} is $1/K$ if $i \neq j$ and i, j are adjacent, 0 if $i \neq j$ and i, j are not adjacent, and $1 - d(i)/K$ if $i = j$. Clearly now $\mathbf{P}$ is (symmetric and) doubly stochastic and so the stationary distribution gives probability $1/N$ to each state provided the MC is aperiodic. If at least one $d(i)$ is strictly less than K, then as a bonus we have that for such an i, $p_{ii} > 0$ and so the state i and (noting that the MC is irreducible) the entire MC are aperiodic and the distribution of the state after q steps tends to the discrete uniform distribution as $q \to \infty$, whatever be the initial state. We mention in passing that if $d(i) = K$ for all i, then the MC need not be aperiodic (if we consider Problem II with $\mathbf{r} = (2, 2, 1, 1)$ and $\mathbf{c} = (1, 2, 2, 1)$, there are exactly six states and $d(i) = 2$ and the period is 2 for each state.)

The difficulty one has to face in using the modification mentioned in the preceding paragraph is in getting a good upper bound for the $d(i)$'s. If K is too large compared to the $d(i)$'s, then the p_{ii}'s become close to 1 and the convergence to the uniform distribution will be too slow. The best K would, of course, be the maximum of $d(i)$'s over all the states, increased by a small number to take care of periodicity. Since this exact maximum cannot be found out, we may do one of two things:

(i) estimate the maximum using a pilot study and use the estimate as K, and

(ii) use the maximum of the $d(i)$'s of the states visited till any stage as the K for the next stage.

We shall refer to these two alternatives as *Choice* (i) and *Choice* (ii).

2.3 *Method* I. For the MCS method with choice (i), which we call *Method I*, we have the following result.

Theorem 3. *Suppose we use Method I and so adopt the following procedure: If the current state is i and $d(i) \leq K$, then we go to one of the states adjacent to i, each with probability $1/K$ and remain at i with probability $1 - d(i)/K$. If the current state is i and $d(i) > K$, then we go to one of the states adjacent to i, each with probability $1/d(i)$. Then the stationary distribution of the MC gives a probability to the i-th state which is proportional to* $\max(d(i), K)$.

Proof. Let $\theta_i = \max(d(i), K)$. Then $p_{ij} = 1/\theta_i$ if $i \neq j$ and i is adjacent to j, $p_{ij} = 0$ if $i \neq j$ and i is not adjacent to j and $p_{jj} = \max(1 - d(j)/K, 0)$. So $\theta_j p_{jj} = \max(K - d(j), 0)$ and $\sum_i \theta_i p_{ij} = \max(K, d(j)) = \theta_j$. Thus $\theta' \mathbf{P} = \theta'$ and $\pi' \mathbf{P} = \pi'$ where π is obtained from θ by normalizing. This proves the theorem. ∎

Note that the estimate of K obtained by pilot study may not be a true upper bound for the $d(i)$'s but is expected to be close to the maximum of the $d(i)$'s. Hence Theorem 3 shows that after a large number of steps, the MCS method with Choice (i) chooses the matrices in $\mathfrak{C}$ with nearly equal probabilities, the

probabilities being equal if $K \geq \max_i d(i)$. Note that periodicity poses no problem at least theoretically since we can make $K > \max d(i)$ by increasing K a little if all the states visited in the pilot study have the same value of $d(i)$.

We note that if we can assume that K is an upper bound for the $d(i)$'s, then while implementing Method I, we do not have to generate all the alternating cycles in the current matrix at every stage. We may first choose a random number R between 1 and K and then start generating alternating cycles in a systematic way. As soon as we reach the R-th alternating cycle, we switch along that cycle and go to the next stage. If all the alternating cycles are exhausted before reaching the R-th, then we go to the next stage without altering the matrix. This simplification reduces the time taken by about 30%–40%. However, for this simplification to work properly, K has to be an upper bound.

2.4 *Method* II. Here we use the basic MCS method with Choice (ii). Strictly speaking, this is no more a Markov chain since the transition probabilities at any stage depend not only on the current state but also on the states visited earlier through their $d(i)$'s which are used in updating K. However, it is easy to see that with probability 1, a state with the maximum $d(i)$ is reached and then on the transition probability matrix does not change and it behaves like a MC, the limiting distribution being the discrete uniform distribution. A rigorous proof of this is given below.

Theorem 4. *Suppose we use Method II and so adopt the following procedure: Let the initial state be i_0. Take the initial value of K to be $d(i_0)$. Whenever we move from a state i to a state i' we update K by replacing it by $d(i')$ if $d(i') > K$. At any stage, if the current state is i, then we go to one of the states adjacent to i, each with probability $1/K$ and remain at i with probability $1 - d(i)/K$. Then, in the limit, all states are equally probable.*

Proof. As already observed, our procedure is not exactly Markovian. However, we can reformulate it so that it becomes a Markov chain and then prove that the stationary distribution is uniform.[1]

We define the new Markov chain M' to have state space $\{(i, j) : i$ is an original state and $d(i) \leq j \leq L\}$ where L is the maximum of $d(i)$ over all original states. (Here, j denotes the 'current value of K'.) It is easy to see that our procedure is equivalent to: we can go from state (i, j) to state (i', j') in a single step iff either (i) the original states i and i' are adjacent and $j' = \max(j, d(i'))$ or (ii) $i = i'$, $j = j'$ and $d(i) < j$. Also the probability of transition is $1/j$ in case (i) and $1 - d(i)/j$ in case (ii). The MC starts with a state of the type $(i, d(i))$. Since it is possible to reach every original state from every other and their number is finite, it easily follows that all states of the type (i, j) with $j < L$ are transient and the complementary set $\{(i, L) : i$ arbitrary$\}$ is a closed set of persistent states. Moreover, the transition matrix restricted to the latter set is

[1] We are indebted to Prof. B. V. Rao for this proof.

doubly stochastic, so the stationary distribution of M' gives probability 0 to the transient states and equal probabilities to the persistent states, see Feller (1960). This is equivalent to saying that, in the limit, all original states are equally probable. ∎

Remark. The conclusion of Theorem 4 remains valid if the initial value K_0 of K is any integer $\geq d(i_0)$.

This remark follows easily from the proof of Theorem 4. The only modification needed is that L has to be replaced by K_0 throughout in case $L < K_0$. This remark will be used in the next subsection.

We note that in Method II, we have to generate all the alternating cycles in the matrix at every stage and cannot use the simplification mentioned for Method I. Moreover, the information on K obtained in earlier runs of the MC is not used in later runs and the convergence to uniform distribution may be slower.

2.5 *Method* III. Method I makes the assumption that the estimated K is a true upper bound for the $d(i)$'s while in Method II, there is some wastage of information. So we adopt a combination of the two, viz.: we estimate K based on a pilot study and use it as the starting value of K for the actual generation of random matrices. However, at every stage of generating the random matrices, we list all alternating cycles of the current matrix, update K if necessary, and use the updated value as the K for the next stage. If we use this method, which we call Method III, it follows from the Remark following Theorem 4 that, in the limit, all states are equally probable.

We prefer Method III even though it takes slightly longer than the other two methods.

We end this section by noting that after completing most of the work, we came to know that a MCMC method close to ours was presented by Besag and Clifford (1989) for Problem I, though the details differ considerably. Holst (1995) also did similar work in a slightly different setting. Hendrickson, Jr. (1995) states that Stone and Roberts (1990) adopted Besag and Clifford's approach but their Monte Carlo sample tends to favour a restricted subset of $(0, 1)$-matrices and that future work should evaluate the long-run probability distribution of $(0, 1)$-matrices from the transition probabilities to compare it to the exact distribution of $(0, 1)$-matrices. We have attempted this last type of evaluation of the MCMC method in Examples 2 and 3 in Section 4.

3. Some practical considerations

Whether we use Method I, II or III, we have to decide on the number of steps to be used before the matrix obtained can be considered to be (approximately) random. By Theorems 1 and 2, any matrix in $\mathcal{C}$ can be reached from any other in at most t steps, where t is the minimum of the number of 1's and the number

of non-structural 0's. In many situations, we have noticed that the number of steps needed will be much less than t. Hence for adequate mixing we think $2t$ to $3t$ steps will be enough. After working with several examples, we found that using $3t$ steps will be better even though the distribution of the number s of reciprocal pairs in a network can be estimated quite closely by using $2t$ steps.

We thus take the matrix obtained in the $3t$-th step to be a random matrix. To get another random matrix, we again start with the given matrix and run the MC for $3t$ steps.This way we can obtain a sample of any required size M. We call running the MC for $3t$ steps starting with the given matrix a *run*. We denote the number of alternating cycles (rectangles in Problem I and rectangles and compact hexagons in Problem II) by ACCT and the number of random matrices generated at any stage by MATCT. If d is the ACCT of the initial matrix, we use the matrices obtained in $2d$ runs as a pilot to estimate K. After working with several examples, we found that increasing the maximum ACCT obtained in the pilot study by about 10% (and adding 1 to take care of small values) gives a reasonable value for K for the actual generation of random matrices. This is large enough to counter the negative effects of periodicity, if present, and is likely to be an upper bound for $d(i)$'s in most examples but is not too large so that running the MC for $3t$ steps will be enough.

The probability that any statistic X takes a particular value x is estimated by the proportion p of matrices in the sample for which $X = x$. The standard error is $\sqrt{p(1-p)/M}$. We note that the standard error of the estimate of $P(X = x)$ obtained by Snijders' method is also close to this.

We now give the main steps of the procedure we have adopted in the form of an algorithm.

Algorithm.

Given: A $(0,1)$-matrix **A** and the number SS of random matrices to be generated. (We will denote the current matrix at every stage by **A**.)

1. Find $t = \min$(number of 1's, number of non-structural 0's).
   MAXUSED = 0.
   MATCT = 0.
2. RUN = 1.
3. **A** = initial matrix.
   STEP = 0.
4. List the alternating cycles (alternating rectangles and, in case of problem II, the compact alternating hexagons) in the current matrix **A** and find their number ACCT.
   MAXUSED = max(MAXUSED, ACCT).
   IF RUN = 1 and STEP = 0, RPILOT = 2 ACCT.
   IF STEP = $3t$ GO TO 5.
   Choose a random integer IRD between 1 and MAXUSED.
   IF IRD > ACCT, GO TO 5.
   Interchange 1's and 0's along the IRD-th alternating cycle of **A**.

5. IF STEP $< 3t$, GO TO 6.
   IF RUN > RPILOT THEN
   Choose **A**.
   MATCT = MATCT + 1.
   IF MATCT = SS, GO TO 8.
   ENDIF
   IF RUN = RPILOT, MAXUSED = 10(MAXUSED/ 9) + 1.
   GO TO 7.
6. Increase STEP by 1 and GO TO 4.
7. Increase RUN by 1 and GO TO 3.
8. STOP.

We have made an interesting empirical observation while working with several examples. In all the examples we found that the maximum number of alternating cycles never exceeds twice the minimum number. (In one example, the minimum was 15 and the maximum 28.) We do not know whether this holds always and whether there is any theoretical basis for this.

4. Results of simulation

We give detailed results for three examples: the first to illustrate the estimation of the distribution of the number s of reciprocal pairs in a network (Problem II), the second to compare the frequencies of all possible networks (Problem II) and the last to compare the frequencies of all possible matrices when there are no structural zeros (Problem I). We used the MCMC algorithm given towards the end of the preceding section in all the examples.

Example 1. We consider estimating the distribution of s when the initial network is

$$\begin{bmatrix} 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 \end{bmatrix}$$

This is one matrix with row sum and column sum vectors $(3,3,3,3,3,3,2)$ and $(5,4,4,2,1,1,3)$ and with structurally zero diagonal, considered in Snijders (1991). In the following table we give the estimates of the distribution of s obtained by MCMC and Snijders' methods with sample size 10,000, the true distribution obtained by complete enumeration as given in Snijders (1991) and the standard error for each probability according to Snijders' method.

	Value of s					
	2	3	4	5	6	7
MCMC	.0051	.0906	.2787	.3706	.2095	.0455
True	.0043	.0900	.2829	.3722	.2037	.0469
Snijders	.0045	.0900	.2961	.3586	.2046	.0461
Std. Error	.0008	.0032	.0052	.0054	.0046	.0023

Here the range of s is from 2 to 7. The total number of matrices is 33,351. The mean and variance of s obtained by the MCMC method are 4.83 and 1.05, while those obtained by Snijders' method are 4.81 (with standard error 0.01) and 1.05 and the true values are 4.82 and 1.04. The number of alternating cycles (rectangles and compact hexagons) varied between 27 and 42 in the pilot study of 68 runs and we took the initial K for the generation of random matrices to be 47. The minimum and maximum number of alternating cycles as well as the minimum and maximum number of alternating rectangles in all the matrices generated were 26 and 44; the minimum and maximum number of compact alternating hexagons in all the matrices generated were 0 and 8.

We estimated the distribution of s in a few other examples. Consider the sequences O: (5, 5, 4, 4, 3, 3, 2, 2, 1, 1), A: (4, 4, 4, 4, 3, 3, 2, 2, 2, 2), B: (3, 3, 3, 3, 3, 3, 3, 3, 3, 3) and C: (2, 2, 2, 2, 3, 3, 4, 4, 4, 4), considered in Snijders (1991). For the row sum vector O and each of the column sum vectors O, A, B and C, we estimated the mean and standard deviation of s by the MCMC method and by Snijders' method. The estimates by the two methods were very close to each other. Note that the out-degrees and in-degrees are skewed in the same way for O and A while they are oppositely skewed for O and C. The number of networks in each of the four cases here will be in billions if not in trillions and the exact distribution of s is not known. We finally looked at the larger data set of Sampson with $n = 18$, see Snijders (1991). Here the row sum and column sum vectors are 4, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 4, 3, 3 and 0, 6, 4, 2, 4, 2, 2, 6, 4, 6, 2, 2, 5, 1, 2, 3, 2, 3. Here also the estimates (of the mean as well as standard deviation) obtained by the two methods were very close.

Example 2. To find out whether all the matrices in $\mathfrak{C}$ are generated with approximately equal probabilities by the MCMC method, we consider generating random $(0, 1)$-matrices with row sum and column sum vectors $(2, 2, 2, 1, 1)$ and $(2, 2, 1, 2, 1)$ and with structurally zero diagonal.

By enumeration it was found that there are 73 possible matrices. We generated 146,000 matrices by the MCMC method and found the frequencies of all the 73 matrices. The results obtained are summarized in the following table.

No. of mat's	Minfr	Maxfr	Ratio	Mean	Std.Dev	χ^2_{72}	$N(0,1)$
29200	347	453	1.305	400	19.02	66.02	−0.47
58400	737	859	1.166	800	27.29	67.96	−0.30
87600	1122	1300	1.159	1200	37.09	83.71	0.98
116800	1510	1690	1.119	1600	42.23	81.37	0.80
146000	1898	2103	1.108	2000	45.99	77.22	0.47

The range of the frequencies was 1898 to 2103 with a mean of 2000 and standard deviation of 45.99. If the matrices generated are indeed random, the frequency of each possible matrix should have distribution $B(146000, 1/73)$, the Binomial distribution with mean 2000 and standard deviation 44.41. Though the frequencies of the 73 matrices are not independent, they are nearly so and may be considered to be a random sample of size 73 from $B(146000, 1/73)$ under the null model. The observed standard deviation 45.99 of the frequencies is marginally larger than the binomial standard deviation 44.41 and thus we may say that the method generates approximately random matrices. To test this more accurately, we also computed the chi-square statistic for goodness of fit to the multinomial distribution with each of the 73 cells having theoretical probability equal to 1/73. The value of χ^2 was 77.22 with 72 df. We incidentally note that since the theoretical probability of each of the 73 cells is 1/73, the value of the χ^2 statistic is simply 73 times the variance of the frequencies divided by the average frequency. Since the df is large, we computed the corresponding standard normal variate and its value is 0.47. Thus the observed distribution of the frequencies is very close to the distribution under equal probabilities for all the 73 matrices. The ratio of the maximum frequency to the minimum was 1.305 when 29,200 matrices were generated and decreased to 1.108 when 146,000 matrices were generated. This also shows that the probabilities of the different matrices are very close to one another. (The ratio possibly decreases further if still larger samples are taken; conversely, the ratio is very large like 20 if a small sample of, say, 730 is drawn.) Thus by all these tests, we conclude that the 73 possible matrices are chosen with nearly equal probabilities. We actually obtained the minimum and maximum frequencies, ratio of the maximum to the minimum, χ^2 and the corresponding standard normal variate etc. after generating 29,200, 58,400, 87,600, 116,800 and finally 146,000 matrices. These are given in the above table. We incidentally note that the number of alternating cycles varied between 7 and 10 in the pilot study as well as in the population and the value 12 was used for K. The number of alternating rectangles varied between 6 and 10 in the population and the number of compact alternating hexagons varied between 0 and 3 in the population. We did not attempt any comparison here with Snijders' method since the latter does not choose matrices with equal probabilities.

We repeated the type of tests done in Example 2 for a few other data sets. We first took the row sum and column sum vectors to be $(3, 2, 2, 1, 1, 1)$ and $(1, 2, 3, 1, 1, 2)$. By enumeration we found that there are 440 matrices. We

then generated 880,000 random matrices by the MCMC method and noted the frequencies of the 440 matrices after 176000, 352000, ..., 880000 matrices. The values of the standard normal variate for goodness of fit were between -1.32 and 1.21. We next took the out-degree and in-degree sequences to be $(3,3,2,2,1,1)$ and $(1,1,2,2,3,3)$ respectively. Then the number of matrices is 1153. We generated 576,500 matrices by the MCMC method. Here the values of the standard normal variate for goodness of fit were between -0.55 and 1.12. We incidentally note that, here, the number of alternating cycles varied between 15 and 28 (the number of alternating rectangles varying between 14 and 28) in the pilot study and as well as in the population.

We finally took both out-degree and in-degree sequences to be $(2,2,2,2,2,2)$. Then the number of matrices is 7570. We generated 378,500 matrices by the MCMC method. The values of the standard normal variate for goodness of fit were between -2.30 and 1.35. We mention that we used MS FORTRAN (on an AT486) and with this could not handle larger examples for the frequency analysis. Even for this example with 7570 matrices, we encoded the matrices by converting each row into a single integer by treating the row (in the reverse order) as the binary representation of the integer.

Example 3. We now consider comparing the frequencies of all possible matrices in a sample obtained by the MCMC method when there are no structural zeros and the matrix may not be square.

We first consider generating 3×4 matrices with row sums $3,2,2$ and column sums $3,1,2,1$ and no structural zeros. There are 5 possible matrices. The number of alternating rectangles is either 3 or 4 for each of them. Based on the pilot study, we chose $K = 5$. We generated 10,000 matrices and the data are given in the following table. The χ^2 values are somewhat on the low side, but we think this is due to sampling errors and the error is, in any case, on the side of the frequencies being too close to one another. This shows that the 5 matrices are chosen with very nearly equal probabilities.

No. of mat's	Minfr	Maxfr	Ratio	Mean	Std.Dev	χ^2_4
2000	387	429	1.109	400	16.505	3.405
4000	768	813	1.059	800	16.420	1.685
6000	1181	1222	1.035	1200	16.529	1.138
8000	1565	1636	1.045	1600	26.405	2.179
10000	1965	2049	1.043	2000	29.353	2.154

We next considered a few other examples of Problem I. There are 156 matrices of order 4×5 with row sums $3,3,2,2$ and column sums $2,3,1,2,2$ and no structural zeros. When we generated 312,000 matrices by the MCMC method, the values of the standard normal variate for goodness of fit were between -0.82 and 0.53. We finally considered generating 5×6 matrices with row sums $3,3,2,2,2$ and column sums $2,2,3,1,2,2$ and no structural zeros. There are 6114 possible matrices. We generated 611,400 matrices and the values of the standard normal variate for goodness of fit were between 0.83 and 1.50.

The main advantage of the MCMC method seems to be that it can produce (at least in the limit) a random matrix while the other methods (Snijders' and Pramanik's) cannot. However, its main disadvantage is that it takes a huge amount of time.

We wrote the MCMC program in MS FORTRAN whereas Snijders' program was written in TURBO PASCAL. When both these were run on the same AT486, the former seems to take 1.5 times to 8 times as much time as the latter. Our MCMC program (for Problem II) was written specifically for finding the distribution of s but the time taken for other statistics would be roughly the same as most of the time is taken for generating the random matrices. The same probably holds for Snijders' method though his program finds, by default, the distribution of s and the distribution of the proportion of the triads which are transitive. Thus the above time comparison seems meaningful.

However, the MCMC method can be made faster if one is satisfied with slightly less accuracy. For example, if $2t$ steps are used instead of $3t$ in each run, the time taken reduces by about 30%. One could also reduce the number of pilot runs substantially. One can also use the following method: choose a certain number of matrices, say the matrices from the $(2t+1)$-th step to the $3t$-th step in each run. This will obviously cut the time required by a factor of nearly t (and can make the method much faster than Snijders'). However, then, the successive matrices generated are not statistically independent since from any given matrix we can go only to a few adjacent ones. In spite of this, the procedure works reasonably well for the estimation of the distribution of any statistic provided the number of 'runs' used is reasonably large like 100. However, the standard deviation of the frequencies of the possible matrices in the sample obtained thus is somewhat higher than the binomial one.

5. Comparison with other methods

In this section, we briefly explain Pramanik's and Snijders' methods and compare our method with theirs.

Pramanik's method for generating a nearly random matrix starts by fixing as many entries in the matrix as possible thus: if there are structural 0's, they are filled in. At any stage, the *free cells* are those whose entries are not yet determined. If the number of free cells in a row equals the row sum minus the number of 1's filled in that row, then fill all the free cells in that row by 1's. If a row sum equals the number of 1's filled in that row, then fill all free cells in that row by 0's. Do a similar thing for columns. Repeat these until no more cells can be fixed.

Then choose one cell from among the free cells, the cell (i, j) with probability proportional to $r_i c_j$ and put a 1 there. Then fix as many cells as possible. Then choose a cell from among the currently free cells as above and repeat the procedure until a full matrix is obtained. If, at any stage, the partial matrix

cannot be completed to a full matrix with the given row sums and column sums, then abandon the partial matrix and start from the beginning.

Though Pramanik's method seems to work reasonably well when the number of possible matrices is large, it is not exact and does not work well for small order matrices. For example, for problem I with $\mathbf{r} = \mathbf{c} = (2, 1, 1)$, there are 5 matrices and Pramanik's method chooses one of them (the one with 0 in the $(1, 1)$-place) with probability $17/108 = 0.157$ and each of the others with probability $91/432 = 0.211$. Updating the r_i's and c_j's at every stage can improve the performance of Pramanik's method in some cases. In the example just referred to, this increases the smaller probability to $1/6 = 0.167$ and decreases the larger probability to $5/24 = 0.208$. When 880,000 matrices were generated with row sum and column sum vectors $(3, 2, 2, 1, 1, 1)$ and $(1, 2, 3, 1, 1, 2)$ and with zero-diagonal, the ratio of the maximum and minimum frequencies was about 2 for Pramanik's method and 1.14 for MCMC method. Also, the accuracy of Pramanik's method cannot be improved by increasing the sample size unlike for MCMC method. The main advantage of his method seems to be its simplicity. We have not made any time comparison between his method and ours.

Snijders' method to estimate the distribution of any statistic based on a random matrix is essentially the following: the cells in the matrix to be generated are ordered lexicographically (top to bottom and right to left within each row). First the structural 0's are filled in and these cells are treated as determined. At any stage, if we arrive at a cell (i, j) not yet determined, we see if its entry can be determined, i.e., if the number of 1's already filled in in the i-th row is r_i or the number of 0's already filled in in the i-th row is $n - r_i$ or the number of 1's already filled in in the j-th column is c_j or the number of 0's already filled in in the j-th column is $m - c_j$. (Here m denotes the number of rows and n the number of columns.) If the entry in (i, j) can be determined, we do so and go to the next cell. Otherwise we put 1 there with probability π and 0 with probability $1 - \pi$ where π, which depends on the earlier choices made, is computed in a rather complicated way as explained in Snijders' paper. Proceeding like this, a matrix is generated. If at any stage we get a partial matrix which cannot be completed to a full matrix with the given row sums and column sums, we abandon it and start from the beginning. When a valid matrix is obtained, the probability with which it is generated is computed by multiplying the probabilities with which the free cells have been filled with 1's and 0's. The π's are chosen in a complicated way to ensure that the matrices are generated with nearly equal probabilities. However, the probabilities do differ considerably and the mean and so the distribution of any statistic is estimated by using a ratio estimator as in importance sampling.

The main differences between our MCMC method and Snijders' method are the following: Snijders' method can be used with any set of structural 0's and needs only the row sums and the column sums as data (these are true also for Pramanik's method) unlike our MCMC method. The MCMC method needs an initial matrix but this does not pose a problem in applications where the basic

data consists of such a matrix. Snijders' method produces matrices which are somewhat, but not quite, equally probable. The (ratio) estimates obtained by his method are consistent but are not unbiased. The probability of any possible matrix obtained by his method depends on the *order* in which the row sums and column sums are given.

We did not program Snijders' method and so did not make any analysis of the frequencies of different possible matrices in a large sample (like that done in Examples 2 and 3 above). However, in some small examples, we have noticed that the ratio of the maximum to the minimum *probability* of a matrix generated by his method could be quite large like 2 or 3. Comparison of the distribution of s as estimated by Snijders' method and MCMC method was presented in Example 1 and the discussion following it.

6. Conclusions

We have presented a Markov Chain Monte Carlo method in the form of an algorithm in Section 3. This algorithm generates (nearly) random $(0,1)$-matrices with given row sums and column sums, starting from one such matrix, when either there are no structural 0's or we are looking for a square matrix with structurally zero diagonal. We have verified empirically with several examples that the algorithm generates matrices with very nearly equal probabilities.

The accuracy of the algorithm can possibly be improved by increasing the number of steps in each run and by increasing the number of runs. The time taken by the MCMC method depends more on the numbers of alternating cycles in the possible matrices (which is somewhat related to t, the minimum of the number of 1's and the number of non-structural 0's) than on the order of the matrix or the number of 1's in it. The method cannot possibly handle matrices of larger orders like 100×100 unless t is small or one has access to very large computing systems or one adopts some time-saving methods as explained in the last paragraph of Section 4, thereby accepting somewhat lower accuracy.

Acknowledgements. The authors wish to thank Prof. J. K. Ghosh for his encouragement and suggestions and the anonymous referee for his comments leading to several improvements in the paper. The authors are also grateful to Prof. B. V. Rao for the proof of Theorem 4 without which the proof of the procedure would be incomplete.

REFERENCES

BESAG, JULIAN and CLIFFORD, PETER. (1989). Generalized Monte Carlo significance tests, *Biometrika*, **76**, 633–642.

DIACONIS, P. and B. STURMFELS. Algebraic algorithms for sampling from conditional distributions, Submitted to *Ann. Stat.*

FELLER, W. (1960). *An Introduction to Probability Theory and Its Applications*, Vol. 1, Asia Publ. House.

HENDRICKSON, JR. JOHN A. (1995). On the enumeration of rectangular (0, 1)-matrices, *J. Statist. Comput. Simul.*, **51**, 291–313.

HOLST, CLAUS (1995). Item response theory, *Ph.D. Thesis*, The Danish National Institute for Educational Research, Copenhagen.

JERRUM, M. and A. J. SINCLAIR. (1988). Fast uniform generation of regular graphs, *Theoret. Comp. Sci.*, **73** , 91–100.

KATZ, L. and J. H. POWELL. (1954). The number of locally restricted directed graphs, *Proc. Amer. Math. Soc.*, **5**, 621–626.

PRAMANIK, P. (1994). Generating random (0, 1)-matrices with given marginals, Presented at *The Third International Conference on Lattice Path Combinatorics*, New Delhi.

RAO, A. R. and BANDYOPADHYAY, SURAJ (1987). Measures of reciprocity in a social network, *Sankhyā*, Ser. A, **49**, 141–188.

RYSER, H. J. (1963). Combinatorial mathematics, *Carus Mathematical Monographs*, The Mathematical Association of America.

SNIJDERS, TOM A. B. (1991). Enumeration and simulation methods for 0-1 matrices with given marginals, *Psychometrika*, **56**, 397–417.

SUKHATME, P. V. (1938). On bipartitional functions, *Philosophical Transactions of the Royal Society of London*, Ser. A, **237**, 375–409.

WILSON, J. B. (1987). Methods for detecting non-randomness in species co-occurrences: A contribution, *Oecologia*, **73**, 579–582.

STATISTICS AND MATHEMATICS UNIT
INDIAN STATISTICAL INSTITUTE
203, B.T. ROAD
CALCUTTA - 700035
INDIA.