International Journal of Social Science and Economic Research

ISSN: 2455-8834

Volume:06, Issue:12 "December 2021"

EXPLICATING MARKOV CHAINS AND TRANSITION PROBABILITY MATRICES VIA SIMPLE BOARD GAMES

Kkhushi Verma

Student , Department Of Statistics, Ram Lal Anand College, Delhi UniversityBenito Juarez Marg, South Campus, South Moti Bagh, New Delhi, Delhi 110021

DOI: 10.46609/IJSSER.2021.v06i12.027 URL: https://doi.org/10.46609/IJSSER.2021.v06i12.027

Received: 15 Dec. 2021 / Accepted: 29 Dec. 2021 / Published: 31 Dec. 2021

ABSTRACT

A fairly common, and simple, way to statistically model random and stochastic processes is Markov chains. They have been applied to a wide variety of fields, like text generation, financial modelling, production, linguistics, marketing, computer science, and Signal Processing[1]. Simple board games can be used to illustrate its fundamental concepts, as in many popular board games, probabilistic reasoning plays a crucial role.

For example, in Monopoly, which is a looping board game that consists of all recurrent states, for imperative purposes, the primary quantity of interest is the probability of being on any of the 40 positions in a specific turn. Ash and Bishop [3], determined the steady state probability of a player landing on any Monopoly square under the assumption that if a player goes to jail, he or she stays there until the Monopoly player rolls doubles or has spent three turns in jail. This model leads to an extremely practical observation for people who play the game often. [1].

In another chance based board game like Snakes & Ladders, games where players win by reaching a final square which consists of all transient states excluding the recurring ending state, players are interested in the probability of finishing the game in certain moves. Thus, understanding of mathematical model behind these games becomes quite crucial and significant in terms of game design, since any modification in rules and parameters could change the whole occurrence.

This paper explores a method to analyse the prototype of chance based simple board games, depicting the application of Markov chain and using these as a basis to further indicate how complex games can be tackled.

KEYWORDS: Board games, Markov Chains, Probability, Stochastic process , Transition probability matrix..

1. Introduction

Markov chains are an important class of probability models. If there are a finite number of states, then the theory is completely known, although solutions are unfeasible to do manually (unless there are only a few states, or unless the transition matrix has a basic form). However, with the advancement of computer algebra systems, Markov chains with tens or hundreds of states are easily analysable. One class of problems amenable to this type of model are board games [2].

Games have been subjugated to myriad studies in applied mathematics, operations research, and economics. The major difference between the mathematical modelling of games and mathematical modelling of real systems is the approximation phase. A model builder approximates the real world, according to the intent of use of such a model, under some restrictive assumptions; then, the model is analysed with mathematical techniques. Accordingly, the outcomes obtained from the mathematical model are approximations to the real system. Games can be visualised as simplified versions of real life situations, therefore mathematical models can be developed under meagre constraining assumptions [1].

We start with a few definitions from Markov chain theory. First, the squares of a board game need to be matched to states of a Markov chain. Sometimes this can be done with a one-to-one mapping, though having multiple states per square can happen too, like in looped board games.

There exists two types of states: transient and recurrent. Transient states are visited a finite number of times almost surely, while recurrent state, once reached, will be revisited infinitely often almost surely. For example, the game of Snakes & Ladders , has 100 squares and each square can be modelled with one state. The first 99 states are transient because a player will reach the end almost surely. The final state is recurrent since once there, the game ends, and the Markov chain remains there forever [2]. These states are further defined in section 1.4 of this paper.

For authentic board games, episodic or erratic states are typical, even though changing a player's moves in a board game can produce periodic states. For example, a Monopoly board has 40 squares, and if moves were determined by flipping a coin where heads means moving one square forward, and tails means moving one square backward , then all squares would be periodic with period 2. Real board games are also finite in size, so all recurrent states are positively recurrent. Therefore, looping board games like Monopoly are ergodic*, as all states are positively recurrent

ISSN: 2455-8834

Volume:06, Issue:12 "December 2021"

and aperiodic (when using two dice to move pieces)[2].

1.2 Theory

When we study a system that can change over time, we need a way to keep track of those changes. So basically a Markov chain is a particular model for keeping track of systems that can change according to given probabilities. They are used to compute the probabilities of event occurring by viewing them as states, transitioning into other states or into the same states as before [4].

* *A random process X(t) is ergodic if all of its statistics can be resolved from a sample function of the process.That is, the ensemble averages equal the corresponding time averages with unit probability. This is known as the property of ergodicity.*

A stochastic process that satisfies the Markov property (sometimes distinguished as "memorylessness"), is known as Markov process. In simpler terms, it is a process for which predictions can be made regarding the future results based solely on its present state and—most importantly— such predictions are just as good as the ones that could be made knowing the process's full history [5].

With context to the Markov process, Memorylessness refers to the Markov property, an assumption which implies that the properties of random variables related to the Future depend only on relevant information about current time and not on information from further in the past[6].

Now, one should not confuse this with the fact that Markov chain determines probability for next event or state given the result of the previous event. It is a stochastic model describing a sequence of possible states in which the probability of each state depends only on the previous state. Hence, the system's state transits from one state to another "memorylessly" i.e. the next state only depends on the current state and is irrelevant to the previous state but the probabilities are based on prior steps.

1.3 Definitions

Definition 1: A stochastic process is a family of variables X_t , where t is a parameter running over a suitable index T= {0, 1, 2…..}. The index t corresponds to discrete units of time and the index set is T. Stochastic processes are distinguished by their state space, or by the range of possible value for the random variable X. By their index set T, and by the dependence relations

ISSN: 2455-8834

Volume:06, Issue:12 "December 2021"

among the random variables X [7][8].

Definition 2: A Markov process (X) is a stochastic process with the property that, given the values of X_t, the values of X_s for s>t are not governed by the values of X_u for u<t. That is, the probability of any specific future behaviour of the process, when its current state is exactly known, is not altered by additional knowledge regarding its past behaviour. A discrete-time Markov chain is a Markov process whose state space is a finite or countable set, and whose (time) index set is T. In formal terms, the Markov property is that

$$P_r\{X_n + 1 = j \mid X_0 = i_0, \dots, X_{n-1} = i_{n-1}, X_n = i\} = P_r\{X_{n+1} = j \mid X_n = i\}$$

The probability of X_{n+1} being in state j given that X_n is in state i is called the one-step transition probability and is denoted by $P_{ij}^{n,n+1}$. That is,

$$P_{ij}^{n,n+1} = P_r\{X_{n+1} = j \mid X_n = i\}.$$

The notation emphasizes that the transition probabilities are functions not only of the initial and final states but also of the time of transition as well. When the one-step transition probabilities are independent of the time variable n , the Markov chain is said to have stationary transition probabilities [7].

Then $P_{ij}^{n,n+1} = P_{ij}$ is independent of n, and P_{ij} is the conditional probability that the state value undergoes a transition from i to j in one trial. It is customary to arrange these numbers P_{ij} in a square matrix,

$$P= \begin{pmatrix} P_{00} & P_{01} & P_{02} & P_{03} & \cdots \\ P_{10} & P_{11} & P_{12} & P_{13} & \cdots \\ P_{20} & P_{21} & P_{13} & P_{23} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \\ P_{i0} & P_{i1} & P_{i2} & P_{i3} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \end{pmatrix}$$

And, refer to $P = P_{ij}$ as the Markov matrix or transition probability matrix of the process. The i^{th} row of P, for i = 0, 1….. is the probability distribution of the values of X_{n+1} under the condition that $X_n = i$. If the number of states is finite, then P is a square matrix whose order is equal to the

ISSN: 2455-8834

Volume:06, Issue:12 "December 2021"

number of states. The quantities P_{ij} satisfy the following

$$P_{ij} \geq 0 \text{ for all } i, j = 0, 1, 2, \dots, \sum_{j=0}^{\infty} P_{ij} = 1$$

A Markov process is completely defined once its transition probability matrix and theprobability distribution of the initial state X_0 are specified [7][8] .

Definition 3: An absorbing state is one in which, when entered it is impossible to leave. An absorbing Markov chain is a Markov chain with absorbing states and the property that it is possible to transition from any state to absorbing state in a finite number of transitions[7][8].

Definition 4: There are two types of states: transient and recurrent. Transient states are visited a finite number of times almost surely, while recurrent state, once reached, will be revisited infinitely often almost surely. For example, the game of Snakes & Ladders, has 100 squares and each square can be modelled with one state. The first 99 states are transient because a player will reach the end almost surely. The final state is recurrent since once there, the game ends, and the Markov chain remains there forever [2].

Let (X_n), $n \geq 0$ be a Markov Chain with transition matrix P. Then, we say that a state i is recurrent if

$P_i(X_n = i \text{ for infinitely many } n) = 1$

We say that i is transient if

$P_i(X_n = i \text{ for infinitely many } n) = 0$

Thus a recurrent state is the one, to which we keep coming back and a transient state is onewhich you will eventually leave forever [9].

1.4 Linear Algebra

Analysing finite Markov chains requires computing the probability transition matrix, P. Name its entries p_{ij} , where this is the probability of transition from state i to j .

The states can be reordered so that all the transient ones are first (if any), then all the recurrent ones. Then let P_T be the transition matrix from transient to transient states; let

ISSN: 2455-8834

Volume:06, Issue:12 "December 2021"

P_{TR} be the probabilities of going from a transient to a recurrent state; and let P_R be the probabilities of going from a recurrent to a recurrent state. Since there is zero probability that a recurrent state can reach a transient one, we can write P as follows:

$$P = \begin{pmatrix} P_T & P_{TR} \\ 0 & P_R \end{pmatrix} \qquad \text{-----}(1)$$

- First, if states i and j are both transient, then let e_{ij} (where e stands for expectation) be the expected number of times of being in state j when starting from state i.
- Second, if state i is transient and state j is recurrent, let us call the probability of starting in i and finally reaching j as f_{ij} (where f stands for final probability).
- Finally, let us define the following matrices:

E= (e_{ij}) and F = (f_{ij}).

In addition, for a game consisting of recurrent states, the goal is to find what the long-term probability of being in state i. Call this π_i, and call the entire vector π.

Fortunately, using linear algebra, it is easy to state the solutions to the above three points . We just need to compute the following equations:

$$E = (I - P_T) - 1 \qquad \text{----}(2)$$

$$F = SP_{TR} \qquad \text{----}(3)$$

$$\pi^T P = \pi^T \qquad \text{----}(4)$$

Using a computer algebra software package, all three equations are easy to solve as matrix inversion, matrix multiplication and finding eigenvalues are all standard tasks. Next, we apply equations (2), (3), & (4) to some simple board games [2].

1.5 Simple Linear Board Game

Let us consider the simplest linear board game, such as shown in Figure 1. Here, a player starts from the left-most square and reaches the right-most square to conclude the game.

ISSN: 2455-8834

Volume:06, Issue:12 "December 2021"

Figure 1: *The simplest linear board game; it has ten squares: the first nine are transient while the last one is recurrent.*

Now, this can be modelled with a Markov chain with ten states, and all these except for the one corresponding to the end are transient. We also need to specify how a player moves, to make calculations of the expected number of visits in each square. Suppose a fair coin is flipped with the following results: move two to the right for heads, and one to the right for tails. Then the transition matrix P is given in Figure 2, where a player in square 8(of Figure 1) is guaranteed to get to "End" in the next turn. Since the probability of getting either heads or tails, on flipping a coin, is 0.5 each, the probability of landing on a given square after flipping a coin is also 0.5. In some board games, it is required that a player must reach the end by moving the correct number of square. If this is desired here, then replace the ninth row by a copy of the eighth row.

	0	1	2	3	4	5	6	7	8	9
0	0	0.5	0.5	0	0	0	0	0	0	0
1	0	0	0.5	0.5	0	0	0	0	0	0
2	0	0	0	0.5	0.5	0	0	0	0	0
3	0	0	0	0	0.5	0.5	0	0	0	0
4	0	0	0	0	0	0.5	0.5	0	0	0
5	0	0	0	0	0	0	0.5	0.5	0	0
6	0	0	0	0	0	0	0	0.5	0.5	0
7	0	0	0	0	0	0	0	0	0.5	0.5
8	0	0	0	0	0	0	0	0	0	1
9	0	0	0	0	0	0	0	0	0	1

Figure 2: *The probability transition matrix, P, for the board given in Figure 1 and movementis determined by flipping a coin. Heads means move two to the right, tails just one to the right.*

Figure 3 gives the result of equation (2). The first row gives the expected number of times a player stops at each square , for a player beginning at start. Note that the first entry is 1, but this must be true since the player begins at start (which counts as one move), and then must move to the right after the first flip. Also note that the average move forward is 1.5 squares, which suggest a limiting value of 1/1.5= 2/3 as the number of squares increases, which is plausible in this case [2].

1	0.5	0.75	0.625	0.6875	0.65625	0.671875	0.664063	0.667969
0	1	0.5	0.75	0.625	0.6875	0.65625	0.671875	0.664063
0	0	1	0.5	0.75	0.625	0.6875	0.65625	0.671875
0	0	0	1	0.5	0.75	0.625	0.6875	0.65625
0	0	0	0	1	0.5	0.75	0.625	0.6875
0	0	0	0	0	1	0.5	0.75	0.625
0	0	0	0	0	0	1	0.5	0.75
0	0	0	0	0	0	0	1	0.5
0	0	0	0	0	0	0	0	1

Figure 3: *The E matrix for the board given in Figure 1 and movement is determined by the transition probability matrix in Figure 2.*

Figure 4 shows the Mathematica code used to create Figures 2 and 3. The most complex part of the code defines the function `tranMatrixLinear[]` , which creates the P matrix. This function requires the length of the linear board as well as the probabilities for moving zero, one two, three, and so forth, squares forward. Hence the second argument consisting of `{0, 0, 1}/2`, means a probability of ½ of moving one or two squares forward. The matrix P_T is given by `pt` and E is given by `e`. Finally, the last statement prints out the matrixE in a matrix format.

```
tranMatrixLinear[n_,probs_]:=
Module[{p={},r,i},Do[AppendTo[p,
   {Table[0, {i,1,r-1}], probs[[1;;Min[n-r+1, Length[probs] ]
   ]],
   {Table[0, {i, r + Length[probs], n}]} }//Flatten], {r,1,n}]
   Do[p[[r, n]] = 1 - Fold[Plus, 0, p[[r,1;;n-1]]],{r,1,n}];
   Return[p]
 ]
n = 10;
(* Size of board *)
p = tranMatrixLinear[n, {0,1,1}/2];pt = p[[1;;n-1, 1;;n-1]];
e = Inverse[IdentityMatrix[n-1] - pt];
e//MatrixForm//N
```

Figure 4: *Code that produced the matrices shown in Figures 2 and 3. Note that performing the linear algebra tasks is easy using Mathematica.*

ISSN: 2455-8834

Volume:06, Issue:12 "December 2021"

1.6 Simple Snakes And Ladder

Here we are considering a simple snakes and ladder with 8 blocks starting from 0 and ending at 7 given in Figure 5. We need to specify how a player moves , to make calculations of the expected number of visits in each square. Suppose a fair coin is flipped with the following results: move ahead two square (to the left for bottom row and right for upper row) for heads, and one square ahead (to the left for bottom row and right for upper row) for tails.

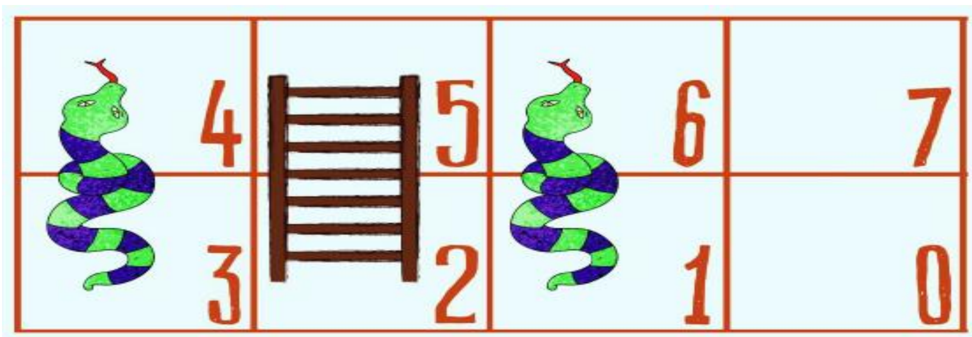

Figure 5: *A simple snakes and ladders with 8 blocks starting from 0 and ending at 7.*

There are two snakes and one ladder. Snake 1 is from 4 to 3, snake 2 is from 6 to 1 and ladder is from 2 to 5. The player starts in the right-most square at the bottom i.e. 0, and moves until the left-most square at the top i.e. 7 is reached as shown in figure 6.

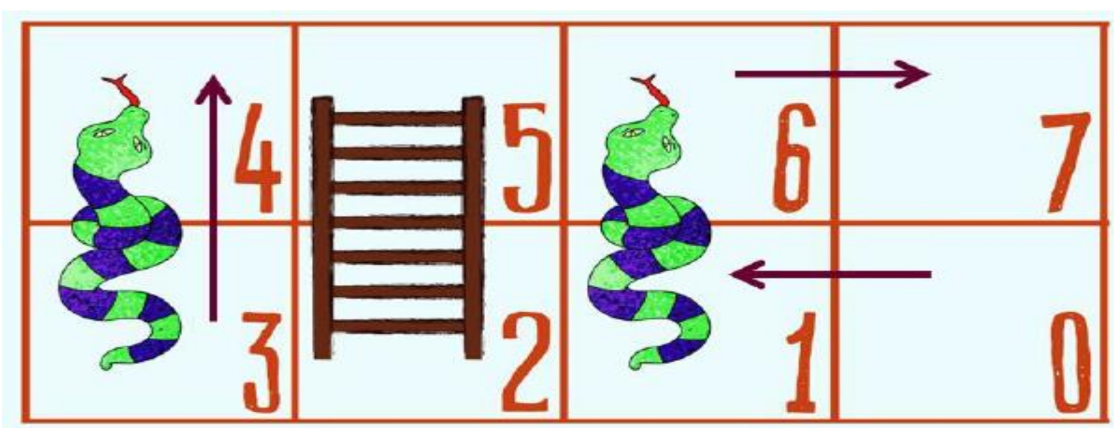

Figure 6: *The player starts in the right-most square at the bottom i.e. 0, and moves until the left-most square at the top i.e. 7 is reached.*

Now, let us consider a few questions for this particular game prototype. First, what would be the

ISSN: 2455-8834

Volume:06, Issue:12 "December 2021"

states for the given snakes and ladder model?

The answer to the above question is, the given snakes and ladders can be modelled with a Markov chain with five states S= {0, 1, 3, 5, 7}, and all these except for the one corresponding to end are transient. To understand this, consider the state diagram given in Figure 7. Here, we should note that 2,4 & 6 are not explicitly stated in sample space as these are temporary states because once you land on them you are promoted (ladder at 2) or demoted (snakes at 4 & 6) depending on the square . And hence, either way the player will land on 1, 3 or 5.

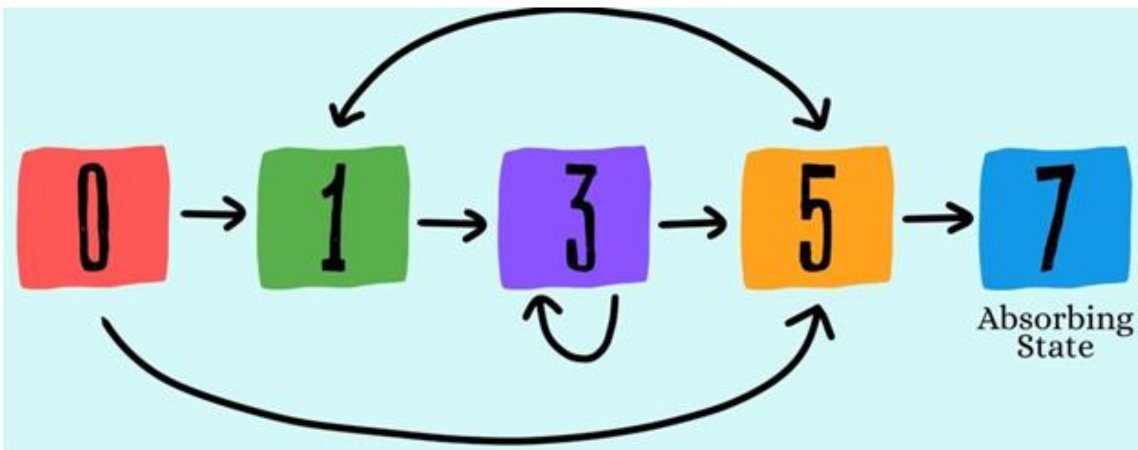

Figure 7: *State diagram specifies/outlines the possible routes that you can take from 0 to 7.*

We will now classify our class structure as : C1={0} ; C2={1,3,5} ; C3={7}.

Class C1 is an open class because once you reach this class can't return back to it. While, class C3 is closed class because once we reach this class there is no other exit and the game is essentially over.

The second question is, what would be the transition probability matrix for the given snakes and ladder model?

Keeping note of the movement of the player depicted in Figure 6 and the state diagram in Figure 7, The TPM for the given model would be given by matrix P in Figure 8:

ISSN: 2455-8834

Volume:06, Issue:12 "December 2021"

P:	0	1	3	5	7
0	0	0.5	0	0.5	0
1	0	0	0.5	0.5	0
3	0	0	0.5	0.5	0
5	0	0.5	0	0	0.5
7	0	0	0	0	1

Figure 8: *Transition probability matrix for given game model*

Here we should note that since the probabilities are determined by fair coin toss, the probabilities are equal i.e. 0.5.

Now we can use this to derive the n-step TPM for the given simple snakes & ladders. With the n-step TPM we would be able to calculate the probabilities of landing on each square after n-coin tosses. The ij^{th} entry of the matrix P^n gives the probability that the Markov chain starting in state i will be in state j after n steps.

Third, what would be the expected value of probabilities after 3 coin tosses for the given snakes and ladder model?

To answer this, firstly, we will calculate P^3 which is given below in Figure 9;

P^3	0	1	3	5	7
0	0	0.125	0.25	0.25	0.375
1	0	0.125	0.25	0.25	0.375
3	0	0.125	0.25	0.25	0.375
5	0	0.125	0.125	0.125	0.625
7	0	0	0	0	1

Figure 9: *Matrix P^3 The ij^{th} entry of the matrix P^3 gives the probability that the Markov chain starting in state i will be in state j after 3 steps.*

Secondly, since we started at state 0, we are only interested in the first row of matrix P^3, which is

ISSN: 2455-8834

Volume:06, Issue:12 "December 2021"

: [0 0.125 0.25 0.25 0.375]

This gives us all the probability of ending up in any particular square after three coin tosses starting from square zero , and now since we have all these values we can go ahead and calculate the expected value of X = 3 , which is done by summing up all the probability values obtained (P_{0j}) multiplied by their respective numbers (j), i.e.

$$E(X = 3|\text{ for } i = 0) = \sum P_{0j} * j = 4.75$$

This means that, on average, after 3 coin tosses, we are expected to move 4.75 squares away from 0.

1.7 Summary

In this paper, the importance and application of Markov chain has been highlighted. Various terms like stochastic processes, Markov chains, transition probability matrix, transient and recurrent states, and absorbing states, were mathematically defined. This was then followed by defining Markov chains to model a simple linear board game, and simple Snakes & Ladders, with the help of linear algebra and advance software packages like Mathematica. Since board games are easily comprehensible, and with Mathematica, easily analysable, these are a great source of examples for teaching .Reviewing the rules and the prototypes of the considered board games, aids in providing conceptual clarity of Markov processes and transition probability matrices, how they work, and what are their applications in finding probabilities for strategic purposes.

References

[1] Baris Tan, Markov Chains and the RISK Board Game, this Magazine 70 (1997), 349–357.

[2] Bilisoly, Roger. (2014). Using Board Games and Mathematica to Teach the Fundamentals of Finite Stationary Markov Chains.

[3] Robert Ash and Richard Bishop, Monopoly as a Markov Process, this Magazine 45 (1972), 26-29.

[4] https://study.com/academy/lesson/markov-chain-definition-applications- examples.html

ISSN: 2455-8834

Volume:06, Issue:12 "December 2021"

[5] https://en.wikipedia.org/wiki/Markov_chain#Definition

[6] https://en.wikipedia.org/wiki/Memorylessness

[7] Zin Mar Tun(2021). Markov Process of Snake And Ladders Board *International Journal of Mathematics Trends and Technology*, 44-52.

[8] L.A. Cheteyan, S. Hengeveld, M,A,Jones, and "Chutes and Ladders for the impatient", vol.42, No1, the mathematical association of America, The College Mathematics Journal, January (2011).

[9] Norris, J. R. (James R.) Markov chains / J. R. Norris. p. cm. – (Cambridge series on statistical and probabilistic mathematics ; no. 2)

