APPLICATIONS AND COMPUTER SIMULATIONS OF MARKOV CHAINS

Sanda Micula [1]
Rodica Sobolu [2*]

ABSTRACT

In this paper we discuss Markov chains, theoretical results, applications and algorithms for computer simulations in MATLAB. We describe the use of Monte Carlo methods for estimating probabilities and other characteristics relating to Markov chains. The paper concludes with some interesting applications.

KEYWORDS: *Markov chains, stochastic processes, computer simulations, Monte Carlo methods, MATLAB.*

AMS Subject Classification: 60E05, 60G99, 60J10, 65C05, 65C60.

1. INTRODUCTION

In probability theory and related fields, a Markov process (named after the Russian mathematician *Andrey Markov*), is a stochastic process that satisfies the "memorylessness" property, meaning that one can make predictions for the future of the process based solely on its present state, independently from its history. A Markov chain is a Markov process that has a discrete state space. Markov chains have many applications as statistical models of real-world problems, such as counting processes, queuing systems, exchange rates of currencies, storage systems, population growths and other applications in Bayesian Statistics.

Monte Carlo methods are used to perform many simulations using random numbers and probability to get an approximation of the answer to a problem which is otherwise too complicated to solve analytically. Such methods use approximations which rely on "long run" simulations, based on computer random number generators. Monte Carlo methods can be used for (but are not restricted to) computation of probabilities, expected values and other distribution characteristics.

[1] Department of Mathematics and Computer Science, Babes-Bolyai University, Cluj-Napoca, Romania, smicula@math.ubbcluj.ro

[2*] corresponding author, Department of Land Measurements and Exact Sciences, University of Agricultural Sciences and Veterinary Medicine, Cluj-Napoca, Romania, rodica.sobolu@usamvcluj.ro

1.1. Preliminaries

We recall a few notions from Probability Theory that will be needed later.

Let S be the sample space of some experiment, i.e. the set of all possible outcomes of that experiment (called elementary events and denoted by e_i). Let P be a probability mapping (see [4]).

Definition 1.1. *A random variable is a function $X : \mathcal{S} \longrightarrow \mathbb{R}$ for which $P\ (X\ \leq x)$ exists, for all $x \in \mathbb{R}$.*

*If $X(\mathcal{S}) \subseteq \mathbb{R}$ is at most countable in $\mathbb{R}$, then X is a **discrete random variable**, otherwise, it is a **continuous random variable**.*

If X is a discrete random variable, then a better way of describing it is to give its **probability distribution function** (**pdf**) or **probability mass function** (**pmf**), an array that contains all its values x_i , and the corresponding probabilities with which each value is taken, $p_i = P(X = x_i)$,

$$X\begin{pmatrix} x_i \\ p_i \end{pmatrix}_{i\in I} \tag{1.1}$$

Of the discrete probability laws, we recall two of the most widely used.

Bernoulli distribution *Bern*(p), with parameter $p \in (0,1)$.This is the simplest of distributions, with pdf

$$X\begin{pmatrix} 0 & 1 \\ 1-p & p \end{pmatrix}. \tag{1.2}$$

It is used to model “success/failure” (i.e. a Bernoulli trial), since many distributions are described in such terms.

Binomial distribution $B(n,p)$, with parameters $n \in \mathbb{N},\ p \in (0,1)$. Consider a series of n Bernoulli trials with probability of success p in every trial ($q = 1 - p$). Let X be the number of successes that occur in the n trials. Then X has a Binomial distribution, with pdf

$$X\begin{pmatrix} k \\ C_n^k p^k q^{n-k} \end{pmatrix}_{k=\overline{0,n}}. \tag{1.3}$$

Note that a Binomial $B(n,p)$ variable is the sum of n independent $Bern(p)$ variables and $Bern(p) = B(1,p)$.

Let us also recall the notion of conditional probability and related properties.

Definition 1.2. *Let A and B be two events with $P(B) \neq 0$. The **conditional probability** of A, given B, is defined as*

$$P(A\,|B) = \frac{P(A \cap B)}{P(B)}.$$

The next result is known as the *total probability rule*.

Proposition 1.3. *Let* $\{A_i\}_{i\in I}$ *be a partition of* $\mathcal{S}$, i.e. $\bigcup_{i\in I} A_i = \mathcal{S}$ ($\{A_i\}_{i\in I}$ *are an exhaustive collection of events*) *and* $A_i \cap A_j = \emptyset,\ i \neq j$ ($\{A_i\}_{i\in I}$ *are mutually exclusive events*). *Let* E *be any event and* B *any event with* $P(B) \neq 0$. *Then*

$$P(E) = \sum_{i\in I} P(A_i)P(E\,|A_i),$$

$$P(E\,|B) = \sum_{i\in I} P(A_i \mid B)P(\,E \mid A_i). \tag{1.4}$$

2. STOCHASTIC PROCESSES AND MARKOV CHAINS

Random variables describe random phenomena at a particular moment of time, but many variables change and evolve in time (think air temperatures, stock prices, currency rates, CPU usage, etc). Basically, stochastic processes are random variables that develop and change in time.

Definition 2.1. *A* ***stochastic process*** *is a random variable that also depends on time. It is denoted by* $X(t,e)$ *or* $X_t(e)$, *where* $t \in \mathcal{T}$ *is time and* $e \in \mathcal{S}$ *is an outcome. The values of* $X(t,e)$ *are called* ***states***.

If $t \in \mathcal{T}$ *is fixed, then* X_t *is a random variable, whereas if we fix e* $\in \mathcal{S}$, X_e *is a function of time, called a* ***realization*** *or* ***sample path*** *of the process* $X(t,e)$.

Definition 2.2. *A* ***stochastic process*** *is called* ***discrete-state*** *if* $X_t(e)$ *is a discrete random variable for all* $t \in \mathcal{T}$ *and* ***continuous-state*** *if* $X_t(e)$ *is a continuous random variable, for all* $t \in \mathcal{T}$.

Similarly, a stochastic process is said to be ***discrete-time*** *if the set* $\mathcal{T}$ *is discrete and* ***continuous-time*** *if the set of times* $\mathcal{T}$ *is a (possibly unbounded) interval in* $\mathbb{R}$.

Throughout the paper, we will omit writing e as an argument of a stochastic process (as it is customary when writing random variables).

2.1. Markov Processes and Markov Chains; Transition Probability Matrix

Definition 2.3. *A stochastic process* X_t *is* ***Markov*** *if for any times* $t_1 < t_2 < \cdots < t_n < t$ *and any sets* $A_1, A_2, \ldots, A_n; A$,

$$P\big(X_t \in A \mid X_{t_1} \in A_1, \ldots, X_{t_n} \in A_n\big) = P\big(X_t \in A \mid X_{t_n} \in A_n\big). \tag{2.1}$$

What this means is that the conditional distribution of X_t given observations of the process at several moments in the past, is the same as the one given only the latest observation.

Definition 2.4. *A discrete-state, discrete-time Markov stochastic process is called a* ***Markov chain***.

To simplify the writing, we use the following notations: Since a Markov chain is a discrete-time process, we can see it as a sequence of random variables $\{X_0, X_1, \ldots\}$ where X_k describes the situation at time $t = k$. It is also a discrete-state process, so we denote the states by 1, 2, …, n (they may start at 0 or some other value and n may possibly be ∞).

Then the random variable X_k has the pdf

$$X_k \begin{pmatrix} 1 & 2 & \dots & n \\ P_k(1) & P_k(2) & \dots & P_k(n) \end{pmatrix}, \tag{2.2}$$

where $P_k(1) = P(X_k = 1), \dots, P_k(n) = P(X_k = n)$. Since the states (the values of the random variable X_k) are the same for each k, one only needs the second row to describe the pdf. So, let

$$P_k = \ [P_k(1) \ \ P_k(2) \ \dots \ P_k(n)] \tag{2.3}$$

denote the vector on the second row of (2.2).

The Markov property (2.1) can be now written as

$$P(X_{t+1} = j \mid X_t = i, X_{t-1} = l, \dots) = P(X_{t+1} = j \mid X_t = i), \text{ for all } t \in \mathcal{T}. \tag{2.4}$$

We summarize this information in a matrix.

Definition 2.5.

- *The conditional probability*

$$p_{ij}(t) = P(X_{t+1} = j \mid X_t = i) \tag{2.5}$$

is called a ***transition probability****; it is the probability that the Markov chain transitions from state i to state j, at time t. The matrix*

$$P(t) = \ \left[p_{ij}(t)\right]_{i,j=\overline{1,n}} \tag{2.6}$$

is called ***the transition probability matrix*** *at time t.*

- *Similarly, the conditional probability*

$$p_{ij}^{(h)}(t) = \ P(X_{t+h} = j \mid X_t = i) \tag{2.7}$$

*is called h-****step transition probability****, i.e. the probability that the Markov chain moves from state i to state j, in h steps and the matrix*

$$P^{(h)}(t) = \left[p_{ij}^{(h)}(t)\right]_{i,j=\overline{1,n}} \tag{2.8}$$

*is the h-****step transition probability matrix*** *at time t.*

Definition 2.6. *A Markov chain is* ***homogeneous*** *if all transition probabilities are independent of time,*

$$p_{ij}(t) = \ p_{ij}, \qquad P(t) = \ P = \left[p_{ij}\right]_{i,j=\overline{1,n}},$$

$$p_{ij}^{(h)}(t) = \ p_{ij}^{(h)}, \qquad P^{(h)}(t) = \ P^{(h)} = \left[p_{ij}^{(h)}\right]_{i,j=\overline{1,n}}.$$

Throughout the rest of the paper, we will only refer to homogeneous Markov chains (even if not specifically stated so).

Proposition 2.7. *Let* $\{X_0, X_1, \dots\}$ *be a Markov chain. Then the following relations hold:*

$$P^{(h)} = P^h, \text{ for all } h = 1, 2, \dots$$

$$P_i = P_0 \cdot P^i, \text{ for all } i = 0, 1, \ldots \tag{2.9}$$

Proof:

The proof of the first relation goes by induction.

Obviously, the first relation in (2.9) is true for h = 1. Assume $P^{(h-1)} = P^{h-1}$. For a matrix M, we use the notation $[M]_{ij} = M(i,j)$ and, similarly, for a vector v, $(v)_i = v\,(i)$. Since the events $\{(X_{h-1} = k)\}_{k=\overline{1,n}}$ form a partition, using the total probability rule (1.4) for $p_{ij}^{(h)} = P(X_h = j \,|X_0 = i)$, we have

$$\begin{aligned} P(X_h = j \,|X_0 = i) &= \sum_{k=1}^{n} P(X_{h-1} = k \mid X_0 = i) \cdot P(X_h = j \,|X_{h-1} = k) \\ &= \sum_{k=1}^{n} p_{ik}^{(h-1)} \cdot p_{kj} \\ &= [P^{h-1} \cdot P]_{ij} \text{ for all } i, j = \overline{1,n}, \end{aligned}$$

so $P^{(h)} = P^h$.

To prove the second relation in (2.9), for each $j = \overline{1,n}$, we have $[P_i]_j = P_i(j) = P(X_i = j)$. Again, using (1.4) for the events $\{(X_0 = k)\}_{k=\overline{1,n}}$, we get

$$\begin{aligned} P(X_i = j) &= \sum_{k=1}^{n} P(X_0 = k) \cdot P(X_i = j \,|X_0 = k) \\ &= \sum_{k=1}^{n} [P_0]_k \cdot p_{kj}^{(i)} \\ &= \left[P_0 \cdot P^{(i)}\right]_j, \end{aligned}$$

so, by the previous relation proved, $P_i = P_0 \cdot P^i$.

Remark 2.8. We used the fact that state destinations are mutually exclusive and exhaustive events, thus forming a partition. That is because from each state, a Markov chain makes a transition to one and only one state. As a consequence, in matrices P and $P^{(h)}(= P^h)$, the sum of all the probabilities on each row is 1. Such matrices are called **stochastic**.

2.2. Steady-State Distribution; Regular Markov Chains

It is sometimes necessary to be able to make long-term forecasts, meaning we want $\lim_{n\to\infty} P_h$, so we need to compute $\lim_{n\to\infty} p_{ij}^{(h)}$.

Definition 2.9. *Let X be a Markov chain. The vector $\pi = [\pi_1, \ldots, \pi_n]$, consisting of the limiting probabilities $\pi_k = \lim_{h\to\infty} P_h\,(k), k = 1, \ldots, n,$ if it exists, is called a* ***steady-state distribution*** *of X.*

In order to find it, let us notice that

$$P_h P = \left(P_0 P^h\right) P = P_0 P^{h+1} = P_{h+1}.$$

Taking the limit as $h \to \infty$ on both sides, we get

$$\pi P = \pi. \tag{2.10}$$

Notice that the system (2.10) is an $n \times n$ singular linear system (multiplication by a constant on each side leads to infinitely many solutions). However, since π must also be a stochastic matrix, the sum of its components must equal 1. We state the following result, without proof.

Proposition 2.10. *The steady-state distribution of a homogeneous Markov chain X,* $\pi = [\pi_1, \dots, \pi_n]$*, if it exists, is unique and is the solution of the* $(n+1)\times n$ $linear\ system$

$$\begin{cases} \pi\, P = \pi \\ \sum_{k=1}^{n} \pi_k = 1. \end{cases} \tag{2.11}$$

Remark. 2.11.

1. When we need to make predictions after a large number of steps, instead of the lengthy computation of P_h , it may be easier to try to find the steady-state distribution, π, directly.

2. If a steady-state distribution exists, then $P^{(h)} = P^h$ also has a limiting matrix, given by

$$\Pi = \lim_{h\to\infty} P^{(h)} = \begin{bmatrix} \pi \\ \vdots \\ \pi \end{bmatrix} = \begin{bmatrix} \pi_1 & \pi_2 & \dots & \pi_n \\ \vdots & \vdots & \dots & \vdots \\ \pi_1 & \pi_2 & \dots & \pi_n \end{bmatrix}.$$

Notice that π and Π do not depend on the initial state X_0. Actually, in the long run the probabilities of transitioning from any state to a given state are the same, $p_{ik} = p_{jk}, \forall i, j, k = \overline{1,n}$ (all the rows of Π coincide). Then, it is just a matter of "reaching" a certain state (from anywhere), rather than "transitioning" to it (from another state). That should, indeed, depend only on the pattern of changes, i.e. only on the transition probability matrix.

As stated earlier, a steady-state distribution may not always exist. We will mention (without proof) one case, which is really easy to check, when such a distribution does exist.

Definition 2.12. *A Markov chain is called* ***regular*** *if there exists h ≥0, such that*

$$p_{ij}^{(h)} > 0,$$

for all $i, j = 1, \dots, n$.

This is saying that at some step h, $P^{(h)}$ has only non-zero entries, meaning that h-step transitions from any state to any state are possible.

Proposition 2.13. *Any regular Markov chain has a steady-state distribution.*

Remark 2.14.

1. Regularity of Markov chains does not mean that *all* $p_{ij}^{(k)}$ should be positive, for all h. The transition probability matrix P, or some of its powers, may have some 0 entries, but there must exist some power h, for which $P^{(h)}$ has all non-zero entries.

2. If there exists a state i with $p_{ii} = 1$, then that Markov chain cannot be regular. There is no exit (no transition possible) from state *i*. Such a state is called an **absorbing state**.

3. Another example of a non-regular chain is that of a **periodic** Markov chain, i.e. one for which there exists $T > 0$, such that $X_t = X_{t+T}$, for all $t \geq 0$. Obviously, in this case, $\lim\limits_{h\to\infty} P^h$ does not exist, and neither does a steady-state distribution.

3. COMPUTER SIMULATIONS OF MARKOV CHAINS AND MONTE CARLO METHODS

Monte Carlo methods are a class of computational algorithms that can be applied to a vast range of problems, where computation of probabilities and other characteristics of interest is too complicated, resource or time consuming, or simply not feasible. They are based on computer simulations involving random number generators and are used to make predictions about processes involving random variables. A computer code that replicates a certain phenomenon can be put in a loop, be simulated any number of times and, based on the outcomes, conclusions about its real life behaviour can then be drawn. The longer run is simulated, the more accurate the predictions are. Monte Carlo methods can be used for estimation of probabilities, other distribution characteristics, lengths, areas, integrals, etc.

Many important characteristics of stochastic processes require lengthy complex computations. Thus, it is preferable to estimate them by means of Monte Carlo methods. For Markov chains, to predict its future behaviour, all that is required is the distribution of X_0, i.e. P_0 (the initial situation) and the pattern of change at each step, i.e. the transition probability matrix P.

Once X_0 is generated, it takes some value $X_0 = i$ (according to its pdf P_0). Then, at the next step, X_1 is a discrete random variable taking the values $j, j = 1, \ldots, n$ with probabilities p_{ij} from row i of the matrix P. Its pdf will be

$$X_1 \begin{pmatrix} 1 & 2 & \ldots & n \\ p_{i1} & p_{i2} & \cdots & p_{in} \end{pmatrix}.$$

The next steps are simulated similarly.

Since, at each step, the generation of a discrete random variable is needed, we can use any algorithm that simulates an arbitrary discrete distribution. Let

$$X \begin{pmatrix} x_i \\ p_i \end{pmatrix}_{i \in I}$$

be any discrete random variable. We use the following simple algorithm (see [4]).

Algorithm 3.1.

1. Divide the interval [0,1] into the subintervals $\{A_i\}_{i\in I}$ as follows:

$$A_0 = [0, p_0)$$
$$A_1 = [p_0, p_0 + p_1)$$
$$\vdots$$
$$A_i = [p_0 + \cdots + p_{i-1}, p_0 + \cdots + p_{i-1} + p_i)$$
$$\vdots$$

2. Let $U \in \mathcal{U}(0,1)$ be a Standard Uniform random variable.

3. If $U \in A_i$, let $X = x_i$.

Indeed, then X takes the values x_i and $P(X = x_i) = \text{length}(A_i) = p_i$. We put Algorithm 3.1 in a loop to generate a Markov chain.

Algorithm 3.2.

1. Given:

$N_M =$ sample path size (length of Markov chain),

$P_0 = [P_0(1) \ldots P_0(n)]$,

$P = [p_{ij}]_{i,j=\overline{1,n}}$.

2. Generate X_0 from its pdf P_0.

3. Transition: if $X_t = i$, generate X_{t+1}, with $p_{ij}, j = \overline{1,n}$, using Algorithm 3.1.

4. Return to step 3 until a Markov chain of length N_M is generated.

4. APPLICATIONS

Let us consider the following example:

An encrypting program generates sequences of letters, such that a vowel is followed by a consonant with probability 0.3, *while a consonant is followed by a vowel with probability* 0.4.

(1) *If the first character is a consonant, make predictions for the second and third character.*

This stochastic process, say X, has two states, 1 =“vowel” and 2 =“consonant”, so it is *discrete-state*. The time set consists of the position of each character in the sequence, so X is also *discrete-time*. Since the prognosis of each character depends only on the previous one, it is a *Markov* process and, hence, a *Markov chain*. Finally, the probability of transitioning from a vowel or from a consonant at *any position* in the sequence, is the same, hence, X is a *homogeneous Markov chain*.

The initial situation (first character) is

$$X_0 \begin{pmatrix} 1 & 2 \\ 0 & 1 \end{pmatrix},\ P_0 = [0,1].$$

The transition probability matrix is

$$P = \begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix} = \begin{bmatrix} 0.7 & 0.3 \\ 0.4 & 0.6 \end{bmatrix}.$$

For the second character, at $t = 1$, the pdf will be

$$P_1 = P_0 \cdot P = [0\ \ 1] \begin{bmatrix} 0.7 & 0.3 \\ 0.4 & 0.6 \end{bmatrix} = [0.4\ \ 0.6].$$

So, the second character has 40% chance of being a vowel and 60% chance of being a consonant. For the third character, the pdf is

$$P_2 = P_0 \cdot P^2 = [0\ \ 1] \begin{bmatrix} 0.61 & 0.39 \\ 0.52 & 0.48 \end{bmatrix} = [0.52\ \ 0.48].$$

The third character is a vowel with probability 0.52 and a consonant with probability 0.48.

(2) *Suppose now that the first character is a consonant with probability 0.8. What is the prognosis for the third and the 100th character?*

In this case, *P* is the same, but X_0 (i.e. P_0) changes.

$$P_0 = [0.2\ \ 0.8]$$

and

$$P_2 = P_0 \cdot P^2 = [0.538\ \ 0.462].$$

The 100th character is many steps away, so instead of computing P_{100}, we find the steady-state distribution. Notice that *P* has all nonzero entries, so the Markov chain is regular, which means a steady-state distribution does exist. We find it by solving the system (2.11), i.e,

$$\begin{cases} 0.7\,\pi_1 + 0.4\,\pi_2 = \pi_1 \\ 0.3\,\pi_1 + 0.6\,\pi_2 = \pi_2 \\ \quad \pi_1 + \quad \pi_2 = 1 \end{cases},$$

with solution $\pi_1 = 0.5714$ and $\pi_2 = 0.4286$. So, in the “long run ”, the pdf of the situation is

$$\lim_{h\to\infty} P_h = \pi = [\pi_1\ \ \pi_2] = [0.5714\ \ 0.4286],$$

i.e., about 57% of the characters are vowels and around 43% are consonants.

(3) *It was found that if more than 15 vowels or more than 12 consonants are generated in a row, in a sequence of 100 characters, then the code becomes vulnerable to cracking. Assuming that the first character is a consonant with probability 0.8, conduct a Monte Carlo study for estimating the probability of the code becoming vulnerable.*

We use Algorithm 3.2 to generate a sample path of length 100, for a large number of simulations. The MATLAB code is given below.

```
% Simulate Markov chain.
clear all
Nm = input(‘ length of sample path = ‘);
N = input(‘ nr. of simulations = ‘)
For j = 1 : N
p = [0.2 0.8]; % initial distr. of vowels/consonants
P = [0.7 0.3; 0.4 0.6]; %trans. prob. matrix
prob(1, :) = p;
for t = 1 : Nm
    U = rand;
    X(t)=1*(U<p(1))+2*(U>=p(1));
% simulate X(1), … , X(Nm)as Bernoulli variables
   prob(t+1, :) = prob(t, :)*P;
   p = P(X(t), :); % prepare the distribution for X(t+1);
                   % its pdf is the (X(t))th row of matrix P
end

i_change = [find(X(1:end-1)~=X(2:end)), Nm];
% find all indices where X changes states
longstr(1) = 1; % find a vector containing the long streak
                % of consecutive vowels/consonants
if (i_change(1) ~=1)
% if X does not change state at step 1, the first long streak
begins at the first change of states
 longstr(1) = i_change(1);
end

for i = 2 : length(i_change)
 longstr(i) = i_change(i) – i_change(i-1);
% find all streaks
end

if(X(1)==1)
vowel = longstr(1:2:end); % find the long streaks of vowels
conson = longstr(2:2:end); % find the long streaks of the consonants
else
 vowel = longstr(2:2:end);
 conson = longstr(1:2:end);
end

maxv(j) = max(vowel); % longest streak of vowels
maxc(j) = max(conson); % longest streak of consonants
end

fprintf(‘probability of more than 15 vowels in a row is
        %1.4f\n’, mean(maxv>15))
fprintf(‘probability of more than 12 consonants in a row is
        %1.4f\n’, mean(maxc>12))
```

After running this code several times, for a sample path of length 100 and for a number of 10^4 and 10^5 simulations, it was found that the probability of having more than 12 vowels in a row is approximately 0.07, whereas the chance of getting more than 12 consonants in a row is around 0.04. Based on these results, the encrypting technique can be properly adjusted.

REFERENCES

[1] C. Andrieu, A. Doucet, R. Holenstein, Particle Markov chain Monte Carlo methods, J. Royal. Statist. Soc. B. Vol. 72(3), 2010, 269–342.

[2] M. Baron, Probability and Statistics for Computer Scientists, 2nd Edition, CRC Press, Taylor & Francis, Boca Raton, FL, USA, 2014.

[3] L. Gurvits, J. Ledoux, Markov property for a function of a Markov chain: A linear algebra approach, Linear Algebra and its Applications, Vol. 404, 2005, 85–117.

[4] D. V. Khmelev, F. J. Tweedie, Using Markov Chains for Identification of Writers, Literary and Linguistic Computing, Vol. 16(4), 2001, 299–307.

[5] T. Liu, Application of Markov Chains to Analyse and Predict the Time Series, Modern Applied Science, Vol. 4(5), 2010, 161–166.

[6] S. Micula, Probability and Statistics for Computational Sciences, Cluj University Press, 2009.

[7] S. Micula, Statistical Computer Simulations and Monte Carlo Methods, J. of Information Systems and Operations Management, Vol. 9(2), 2015, 384–394.

[8] J. S. Milton, J. C. Arnold, Introduction to Probability and Statistics: Principles and Applications for Engineering and the Computing Sciences, 3rd Edition. McGraw-Hill, New York, 1995.

[9] J. Pan, A. Nagurney, Using Markov chains to model human migration in a network equilibrium framework, Mathematical and Computer Modelling, Vol. 19(11), 1994, 31–39.

[10] http://www.mathworks.com/help/matlab/, 2017.