BULETINUL ACADEMIEI DE ŞTIINŢE
A REPUBLICII MOLDOVA. MATEMATICA
Number 2(63), 2010, Pages 84–99
ISSN 1024–7696

Algorithms for Determining the Transient and Differential Matrices in Finite Markov Processes

Alexandru Lazari

Abstract. The problem of determining the transient and differential matrices in finite Markov processes is considered. New polynomial time algorithms for determining the considered matrices in Markov chains are proposed and grounded. The proposed algorithms find the limit and differential matrices efficiently when the characteristic values of the matrix of probability transition are known; the running time of the algorithms is $O(n^4)$, where n is the number of the states of dynamical system in the Markov process.

Mathematics subject classification: 65C40, 60J22, 90C39, 90C40.
Keywords and phrases: Finite Markov Process; Markov Chain; Transient Matrix; Differential Matrix; Polynomial Time Algorithm; Stationary Recurrent Process.

1 Introduction and Problem Formulation

In this paper we study the problem of determining the differential components of the transient matrix in a finite Markov process. We consider a dynamical system L with finite set of states X ($|X| = n$) and assume that the dynamics of this system is modelled by a Markov process with a matrix of transition probability $P = (p_{ij})_{i,j=\overline{1,n}}$. It is well known that the probability transitions of dynamical system from a state to another during t units of times can be determined by calculating the matrix $P(t) = P^t$, $\forall t \geq 0$; an arbitrary element $p_{i,j}(t)$ of the matrix $P(t)$ gives the probability of system L to pass from the state x_i to the state x_j during t transitions. Asymptotic behavior of the matrix $P(t)$ is studied in [1]. Basing on this asymptotic behavior analysis in [3] an approach for determining the stationary component (the limiting probability matrix) of the transient matrix is proposed. Here we shall use this approach and will show how to determine the differential matrices is proposed. We shall use formula (6) and (7) from [3]. On the bases of these formula we can conclude that an arbitrary element $p_{i,j}(t)$ of the matrix $P(t)$ can be determined as follows

$$p_{i,j}(t) = \sum_{y\in\mathbb{C}\setminus\mathcal{D}} \sum_{k=0}^{m(y)-1} \frac{t^k}{y^t}\beta_{i,j,k}(y),\ \forall t > deg(B_{i,j}(z)),\ i,j = \overline{1,n},$$

where $\mathcal{D} = \{z \in \mathbb{C} \mid |I - zP| \neq 0\}$, $\beta_{ijk}(y) \in \mathbb{C}$, $\forall y \in \mathbb{C}\setminus\mathcal{D}$, $k = \overline{0, m(y)-1}$, $m(y)$ is the order of the root y of the polynomial $\Delta(z) = |I - zP|$ and $B_{ij}(z)$ is a polynomial of degree less or equal to $n-1$, $i,j = \overline{1,n}$.

If we denote $\beta_k(y) = (\beta_{ijk}(y))_{i,j=\overline{1,n}}$, $\forall y \in \mathbb{C}\backslash\mathcal{D}, k = \overline{0, m(y)-1}$, then we obtain formula in the matrix form

$$P(t) = \sum_{y \in \mathbb{C}\backslash\mathcal{D}} \sum_{k=0}^{m(y)-1} \frac{t^k}{y^t} \beta_k(y), \ \forall t \geq n. \tag{1}$$

In [3] it have been proved that $\mathbb{C}\backslash\mathcal{D}$ consists of the set of inverses of the nonzero proper values of the matrix P, where the order of each element is the same as the order of the corresponding proper value. Therefore the relation (1) represents the expression which gives the components of the matrix $P(t)$ with respect to the proper values of the matrix P. Basing on this fact we show how to calculate the matrices $\beta_k(y)$. Note that the stationary component for the transient matrix P^t can be found using algorithms from [3].

2 Algorithm for Determining the Differential Matrices

To describe algorithms for determining the differential matrices we need some auxiliary results concerning with the properties of the linear recurrent equations from [4].

2.1 Some auxiliary results

Consider an arbitrary set K on which the operations of summation and multiplication are defined. On this set we consider the following relation

$$a_n = \sum_{k=0}^{m-1} q_k a_{n-1-k}, \ \forall n \geq m, \tag{2}$$

where a_k are given elements from K. A sequence $a = \{a_n\}_{n=0}^{\infty}$ is called the *linear m-recurrence on K* if there exists the vector $q = (q_k)_{k=0}^{m-1} \in K^m$ such that (2) holds. Here we call q the *generating vector* and we call $I_m^{[a]} = (a_n)_{n=0}^{m-1}$ the *initial value of the sequence a*. The sequence a is called the *linear recurrence on K* if $\exists m \in \mathbb{N}^*$ such that the sequence a is a linear m-recurrence on K. If $q_{m-1} \neq 0$ then the sequence a is called non-degenerate; otherwise it is called degenerate.

Denote:

$Rol[K][m]$ – the set of non-degenerate linear m-recurrences on K;

$Rol[K]$ – the set of non-degenerate recurrences on K;

$G[K][m](a)$ – the set of generating vectors of length m of the sequence $a \in Rol[K][m]$;

$G[K](a)$ – the set of generating vectors of the sequence $a \in Rol[K]$.

In the following we will consider K an subfield of the field of complex numbers $\mathbb{C}$ and $a = \{a_n\}_{n=0}^{\infty} \subseteq \mathbb{C}$.

We call the function $G^{[a]} : \mathbb{C} \to \mathbb{C},\ G^{[a]}(z) = \sum\limits_{n=0}^{\infty} a_n z^n$, the *generating function of the sequence* $a = (a_n)_{n=0}^{\infty} \subseteq \mathbb{C}$ and we call the function $G_t^{[a]} : \mathbb{C} \to \mathbb{C},\ G_t^{[a]}(z) = \sum\limits_{n=0}^{t-1} a_n z^n$ the *partial generating function of order t of the sequence* $a = (a_n)_{n=0}^{\infty} \subseteq \mathbb{C}$.

Let $a \in Rol[K][m],\ q \in G[K][m](a)$. For this sequence we will consider the *characteristic polynomial* $H_m^{[q]}(z) = 1 - zG_m^{[q]}(z)$ and the characteristic equation $H_m^{[q]}(z) = 0$. For an arbitrary $\alpha \in K^*$ we also call the polynomial $H_{m,\alpha}^{[q]}(z) = \alpha H_m^{[q]}(z)$ characteristic polynomial of the sequence a. We introduce the following notation:

$H[K][m](a)$ – the set of characteristic polynomials of degree m of the sequence $a \in Rol[K]$;
$H[K](a)$ – the set of characteristic polynomials of sequence $a \in Rol[K]$.

In the case when we will operate with arbitrary recurrence (not obligatory non-degenerate) for the corresponding set we shall use the similar notation and will specify with the mark "*", i.e. we will denote respectively: $Rol^*[K][m]$, $Rol^*[K]$, $G^*[K][m](a)$, $G^*[K](a)$, $H^*[K][m](a)$, $H^*[K](a)$). We shall use the following well known properties:

1) Let $a \in Rol[K][m],\ q \in G[K][m](a),\ H_{m,\alpha}^{[q]}(z) = \prod\limits_{k=0}^{p-1} (z - z_k)^{s_k},\ z_i \neq z_j,\ \forall i \neq j$. Then $a_n = I_m^{[a]} \cdot ((B^{[a]})^T)^{-1} \cdot (\beta_n^{[a]})^T,\ \forall n \in \mathbb{N}$, where $\beta_i^{[a]} = \left(\dfrac{\tau_{ij}}{z_k^i} \right)_{k=\overline{0,p-1},\ j=\overline{0,s_k-1}}$, $\tau_{ij} = \begin{cases} i^j, & if\ i^2 + j^2 \neq 0 \\ 1, & if\ i = j = 0 \end{cases}$, $i \in \mathbb{N},\ B^{[a]} = (\beta_i^{[a]})_{i=0}^{m-1}$;

2) If a is a matrix sequence, $a \in Rol[M_n(K)][m]$ and $q \in G[M_n(K)][m](a)$, then $a \in Rol^*[K][mn]$ and $|I - zG_m^{[q]}(z)| \in H^*[K][mn](a)$.

2.2 The Main Results and Algorithm

Consider the matrix sequence $a = (P(t))_{t=0}^{\infty}$. Then it is easy to observe that the recurrent relation $a_t = Pa_{t-1},\ \forall t \geq 1$ holds. So, $a \in Rol[M_n(\mathbb{R})][1]$ with generating vector $q = (P) \in G[M_n(\mathbb{R})][1](a)$. Therefore according to the mentioned above property 2 we have $a \in Rol^*[\mathbb{R}][n]$ and $\Delta(z) \in H^*[\mathbb{R}][n](a)$.

Let $r = deg\Delta(z)$ and consider the subsequence $\overline{a} = (P(t))_{t=n-r}^{\infty}$ of the sequence a. We have $\overline{a} \in Rol[\mathbb{R}][r]$ and $\Delta(z) \in H[\mathbb{R}][r](\overline{a})$. For the corresponding elements this relation can be expressed as follows: $\overline{a}_{ij} \in Rol[\mathbb{R}][r]$, $\Delta(z) \in H[\mathbb{R}][r](\overline{a}_{ij}),\ i,j = \overline{1,n}$.

According to property 1) mentioned above we obtain:

$$p_{ij}(t) = a_{ij}(t) = \overline{a}_{ij}(t-n+r) = I_r^{[\overline{a}_{ij}]}(B^T)^{-1}(\beta_{t-n+r})^T,\ i,j = \overline{1,n},\ \forall t \geq n-r, \quad (3)$$

where

$$\beta_t = \left(\frac{t^k}{y^t}\right)_{\substack{y\in\mathbb{C}\setminus\mathcal{D},\\ k=\overline{0,m(y)-1}}}, \ \forall t \geq 0, \ B = (\beta_j)_{j=\overline{0,r-1}}, \ 0^0 \equiv 1. \tag{4}$$

Now it is evident how to determine the initial values of subsequences $\overline{a}_{ij}, \ i,j=\overline{1,n}$:

$$I_r^{[\overline{a}_{ij}]} = (\overline{a}_{ij}(t))_{t=0}^{r-1} = (a_{ij}(t))_{t=n-r}^{n-1} = (p_{ij}(t))_{t=n-r}^{n-1}, \ i,j=\overline{1,n}. \tag{5}$$

If we denote

$$I_r^{[\overline{a}_{ij}]}(B^T)^{-1} = (\gamma_{ijs}(y))_{y\in\mathbb{C}\setminus\mathcal{D}, \ s=\overline{0,m(y)-1}}, \ i,j=\overline{1,n}, \tag{6}$$

then formula (3) takes the following form:

$$p_{i,j}(t) = \sum_{y\in\mathbb{C}\setminus\mathcal{D}} \sum_{s=0}^{m(y)-1} \frac{(t-n+r)^s}{y^{t-n+r}} \gamma_{ijs}(y) =$$

$$= \sum_{y\in\mathbb{C}\setminus\mathcal{D}} \sum_{s=0}^{m(y)-1} \sum_{k=0}^{s} C_s^k (r-n)^{s-k} y^{n-r} \gamma_{ijs}(y) \frac{t^k}{y^t} =$$

$$= \sum_{y\in\mathbb{C}\setminus\mathcal{D}} \sum_{k=0}^{m(y)-1} \frac{t^k}{y^t} \sum_{s=k}^{m(y)-1} C_s^k (r-n)^{s-k} y^{n-r} \gamma_{ijs}(y) =$$

$$= \sum_{y\in\mathbb{C}\setminus\mathcal{D}} \sum_{k=0}^{m(y)-1} \frac{t^k}{y^t} \beta_{ijk}(y), \ i,j=\overline{1,n}, \ \forall t \geq n-r, \tag{7}$$

where

$$\beta_{ijk}(y) = y^{n-r} \sum_{s=k}^{m(y)-1} C_s^k (r-n)^{s-k} \gamma_{ijs}(y), \ \forall y \in \mathbb{C}\setminus\mathcal{D}, \ k=\overline{0,m(y)-1}, \ i,j=\overline{1,n}. \tag{8}$$

Rewriting relations (7) in the matrix form we obtain the representation (1) of the matrices $\beta_k(y)$ ($y \in \mathbb{C}\setminus\mathcal{D}, \ k=\overline{0,m(y)-1}$) which can be determined according formula (8). This means that we have grounded the following algorithm for the decomposition of the transient matrix:

Algorithm 1. ***Decomposition of the transient matrix***

Input Data: The matrix of transition probability P.
Output Data: The matrices $\beta_k(y)$ ($y \in \mathbb{C}\setminus\mathcal{D}, \ k=\overline{0,m(y)-1}$).

1. Calculate the coefficients of the characteristic polynomial $\Delta(z)$ of the matrix P using algorithm from [3] (the algorithm from [3] is based on Leverrier's method ([5]); the computational complexity of this algorithm is $O(n^4)$;

2. Solve the equation $\Delta(z) = 0$ and find all roots of these equations in $\mathbb{C}$ and determine $\mathbb{C}\backslash\mathcal{D}$;

3. Determine the order of each root $m(y)$ of the characteristic polynomial. The order of each root can be found using Horner's scheme, i.e. the order of the root by the number of successive factorization of the polynomial $\Delta(z)$ by $(z-y)$, $\forall y \in \mathbb{C}\backslash\mathcal{D}$;

4. Calculate the matrix B using formula (4);

5. Determine the matrix $(B^T)^{-1}$. This matrix can be found using $(O(n^3))$ elementary operations;

6. Calculate the values $C_s^k,\ s = \overline{0, \max\limits_{y\in\mathbb{C}\backslash\mathcal{D}} m(y) - 1},\ k = \overline{0,s}$, according to Pascal triangle rule : $C_s^0 = C_s^s = 1,\ C_s^k = C_{s-1}^{k-1} + C_{s-1}^k\ (k = \overline{1, s-1})$;

7. Find recursively $(r-n)^s,\ s = \overline{0, \max\limits_{y\in\mathbb{C}\backslash\mathcal{D}} m(y) - 1}$;

8. For every $i, j = \overline{1,n}$ do the following steps:

a. Find the initial value $I_r^{[\overline{a}_{ij}]}$ according to formula (5);

b. Calculate the values $\gamma_{ijs}(y),\ y \in \mathbb{C}\backslash\mathcal{D},\ s = \overline{0, m(y)-1}$, according to (6);

c. For arbitrary $y \in \mathbb{C}\backslash\mathcal{D},\ k = \overline{0, m(y)-1}$, determine the coefficients $\beta_{ijk}(y)$ of the matrix $\beta_k(y)$ using formula (8) and the values calculated at the steps 6. – 7.

2.3 Computational Aspects of the Algorithm

The proposed algorithm can be used efficiently for determining the differential matrices in the case when the characteristic values of the matrix P are known. Therefore the computational complexity of the algorithm depends on computational complexity of determining the characteristic values of the matrix P. If the set of characteristic values of the matrix P are known then it is easy to observe that the algorithm determines the differential matrices in time $O(n^4)$. We obtain this estimation of the running time of the algorithm if we estimate in the worst case the number of elementary operations of the steps 3)-8) of the algorithm.

Note that the matrix $\beta_0(1)$ corresponds to limit probability matrix Q of the Markov chains and therefore this matrix can be calculated using $O(n^4)$ elementary operations.

So, basing on results described above we may conclude that the matrix $P(t)$ can be represented as follows

$$P(t) = \sum_{y\in\mathbb{C}\backslash\mathcal{D}} \sum_{k=0}^{m(y)-1} \beta_k(y)\frac{t^k}{y^t},\ \forall t \geq n-r.$$

For $t = \overline{0, n-r-1}$ this formula can be expressed in the form

$$P(t) = L(t) + \sum_{y\in\mathbb{C}\backslash\mathcal{D}} \sum_{k=0}^{m(y)-1} \beta_k(y)\frac{t^k}{y^t}, \tag{9}$$

where $L(t)$ is a matrix that depends only on t. If the matrices $\beta_k(y)$, $\forall y \in \mathbb{C}\backslash\mathcal{D}$, $k = \overline{0, m(y)-1}$, are known then we can determine the matrices $L(t)$ from (9), taking into account that $P(t) = P^t$, $\forall t \geq 0$.

In [1, 2] it is noted that the matrices $L(t)$, $t = \overline{0, n-r-1}$, and $\beta_k(y)$, for each $y \in (\mathbb{C}\backslash\mathcal{D})\backslash\{1\}$, $k = \overline{0, m(y)-1}$, are differential matrices, i.e. the sum of elements across to each row is equal to zero. The unique non-differential component matrix in representation (9) is the matrix $\beta_0(1)$; the remainder matrices $\beta_k(1)$, $k = \overline{1, m(1)-1}$, are null (see ([3]).

3 Algorithm for Determining the Limit and Differential Matrices in Markov Chains

We shall use the ideas from previous section for a simultaneous calculation of the limit and differential matrices in Markov chains. We propose a modification of algorithms from [3] and previous section that allows to determine the limit and differential matrices. In a similar way as in Section 2 we assume that all characteristic values of the matrix P are known and show how to determine all components of the transient matrix represented in the form (1). The main details concerned with the specification and argumentation of the modified algorithm are described in the next two subsections.

3.1 Some Auxiliary Results Concerning with Representation of z-transform

We shall use the same method from Subsection 2.3 of [3] for determining the matrix $F(z) = (I - zP)^{-1}$ described in Subsection 2.3 of [3], but here we will not divide $F(z)$ by $(z-1)^{m(1)-1}$. In a such way we obtain

$$F(z) = \frac{1}{\Delta(z)} \sum_{k=0}^{n-1} R^{(k)} z^k, \tag{10}$$

where the matrix-coefficients $R^{(k)}$, $k = \overline{0, n-1}$, are determined recursively according to formula

$$R^{(0)} = \beta_0 I; \quad R^{(k)} = \beta_k I + PR^{(k-1)}, \ k = \overline{1, n-1}; \tag{11}$$

and the values β_k, $k = \overline{0, n}$, represent the coefficients of the polynomial $\Delta(z)$ calculated according to Algorithm 1.1 from [3].

Basing on formula (5) from [3] we can observe that the elements of the matrix $F(z)$ can be expressed in the following form

$$F_{ij}(z) = B_{ij}(z) + \sum_{y \in \mathbb{C}\backslash\mathcal{D}} \sum_{k=1}^{m(y)} \frac{\alpha_{i,j,k}(y)}{(z-y)^k}, \ i, j = \overline{1, n}. \tag{12}$$

So, in general form the relation (12) can be written in the following way

$$F_{ij}(z) = B_{ij}(z) + \frac{Q_{ij}(z)}{\Delta(z)},\ i,j=\overline{1,n}, \tag{13}$$

where $Q_{ij}(z) \in \mathbb{C}[z]$ and $deg(Q_{ij}(z)) < deg(\Delta(z)) = r,\ i,j=\overline{1,n}$.

If we express the equality (10) for each element and after that substitute in (13) then we obtain formula

$$\sum_{k=0}^{n-1} R_{ij}^{(k)} z^k = B_{ij}(z)\Delta(z) + Q_{ij}(z),\ i,j=\overline{1,n}.$$

So, $B_{ij}(z) = \sum\limits_{k=0}^{n-1-r} b_{ijk} z^k$ and $Q_{ij}(z) = \sum\limits_{k=0}^{r-1} q_{ijk} z^k$ represent the quotient and the rest, respectively, after the division of the polynomial $\sum\limits_{k=0}^{n-1} R_{ij}^{(k)} z^k$ by $\Delta(z)$. Therefore the polynomials $B_{ij}(z)$ and $Q_{ij}(z)$ can be found using the procedure described bellow.

Calculation procedure for Determining the polynomials $B_{ij}(z)$ and $Q_{ij}(z)$, $i,j=\overline{1,n}$:

- For $i,j=\overline{1,n}$ do:
  - $q_{ijk} = R_{ij}^{(k)},\ k=\overline{0,n-1}$;
- For $k = n-1, n-2, \ldots, r$ do:
  - $b_{i,j,k-r} = \dfrac{q_{ijk}}{\beta_r}$;
  - $q_{i,j,k-t} = q_{i,j,k-t} - b_{i,j,k-r}\beta_{r-t},\ t=\overline{0,r}$.

3.2 Expansion of z-transform with Respect to Nonzero Characteristic Values

Let $\mu \in \mathbb{C}\backslash\mathcal{D},\ m(\mu) = m$ (μ^{-1} be a nonzero characteristic value of the matrix P and assume that the order of this characteristic value is m). According to formula (12) – (13), for the separated root μ we have

$$\frac{Q_{ij}(z)}{\Delta(z)} = \sum_{k=1}^{m} \frac{\alpha_{i,j,k}(\mu)}{(z-\mu)^k} + \sum_{y\in(\mathbb{C}\backslash\mathcal{D})\backslash\{\mu\}} \sum_{k=1}^{m(y)} \frac{\alpha_{i,j,k}(y)}{(z-y)^k},\ i,j=\overline{1,n}. \tag{14}$$

Let $\Delta(z) = (z-\mu)^m D(z),\ D(z) = \sum\limits_{k=0}^{r-m} d_k z^k$ and denote $deg(D(z)) = M$. The relation (14) can be written as follows

$$\frac{Q_{ij}(z)}{\Delta(z)} = \frac{G_{ij}(z)}{(z-\mu)^m} + \frac{E_{ij}(z)}{D(z)},\ i,j=\overline{1,n},$$

where $E_{ij}(z) = \sum\limits_{k=0}^{M-1} e_{ijk} z^k,\ G_{ij}(z) = \sum\limits_{k=0}^{m-1} g_{ijk} z^k \in \mathbb{C}[z],\ i,j = \overline{1,n}$. Making elementary transformation we obtain

$$Q_{ij}(z) = G_{ij}(z)D(z) + E_{ij}(z)(z-\mu)^m,\ i,j = \overline{1,n}.$$

By expansion the function $(z-\mu)^m = \sum\limits_{k=0}^{m} C_m^k (-\mu)^{m-k} z^k$ and then introducing the notation $\xi(k) = C_m^k(-\mu)^{-k},\ k = \overline{0,m}$ we have

$$(z-\mu)^m = \sum_{k=0}^{m} C_m^k (-\mu)^{m-k} z^k = (\xi(m))^{-1} \sum_{k=0}^{m} \xi(k) z^k.$$

Now for our relation we make the following transformations:

$$\sum_{t=0}^{r-1} q_{ijt} z^t = \sum_{k=0}^{m-1} g_{ijk} z^k \sum_{s=0}^{M} d_s z^s + (\xi(m))^{-1} \sum_{s=0}^{M-1} e_{ijs} z^s \sum_{k=0}^{m} \xi(k) z^k =$$

$$= \sum_{k=0}^{m-1} \sum_{s=0}^{M} g_{ijk} d_s z^{k+s} + (\xi(m))^{-1} \sum_{k=0}^{m} \sum_{s=0}^{M-1} \xi(k) e_{ijs} z^{k+s} =$$

$$= \sum_{t=0}^{r-1} z^t \left[\sum_{\substack{k+s=t \\ 0 \le k \le m-1 \\ 0 \le s \le M}} g_{ijk} d_s + (\xi(m))^{-1} \sum_{\substack{k+s=t \\ 0 \le k \le m \\ 0 \le s \le M-1}} \xi(k) e_{ijs} \right].$$

Equated the corresponding coefficients we obtain

$$q_{ijt} = \sum_{\substack{0 \le k \le m-1 \\ 0 \le t-k \le M}} d_{t-k} g_{ijk} + (\xi(m))^{-1} \sum_{\substack{0 \le s \le t \\ t-m \le s \le M-1}} \xi(t-s) e_{ijs} =$$

$$= \sum_{k=0}^{m-1} d_{t-k} I_{\{0 \le x \le M\}}(t-k) g_{ijk} + (\xi(m))^{-1} \sum_{s=0}^{t} \xi(t-s) I_{\{t-m \le x \le M-1\}}(s) e_{ijs},$$

where $I_A(x)$ is index of the set A: $I_A(x) = 1,\ \forall x \in A$ and $I_A(x) = 0,\ \forall x \notin A$.

For $t \le M-1$ formula above can be written in the following form

$$q_{ijt} = \sum_{k=0}^{m-1} d_{t-k} I_{\{x \le t\}}(k) g_{ijk} + (\xi(m))^{-1} \sum_{s=0}^{t} \xi(t-s) I_{\{x \ge t-m\}}(s) e_{ijs} =$$

$$= \sum_{k=0}^{m-1} d_{t-k} I_{\{x \le t\}}(k) g_{ijk} + (\xi(m))^{-1} e_{ijt} + (\xi(m))^{-1} \sum_{s=0}^{t-1} \xi(t-s) I_{\{x \ge t-m\}}(s) e_{ijs} \ \Leftrightarrow$$

$$e_{ijt} = \xi(m) \left[q_{ijt} - \sum_{k=0}^{m-1} d_{t-k} I_{\{x \le t\}}(k) g_{ijk} - (\xi(m))^{-1} \sum_{s=0}^{t-1} \xi(t-s) I_{\{x \ge t-m\}}(s) e_{ijs} \right].$$

So, finally we will obtain the following expression

$$e_{ijt} = w_{ijt} + \sum_{k=0}^{m-1} x_{tk} g_{ijk},\ t = \overline{0, M-1},\ i, j = \overline{1, n}.$$

In the following we will determine the coefficients w_{ijt} and x_{tk} from the expression above. We have

$$w_{ijt} + \sum_{k=0}^{m-1} x_{tk} g_{ijk} = e_{ijt} = \xi(m) q_{ijt} - \sum_{k=0}^{m-1} \xi(m) d_{t-k} I_{\{x \le t\}}(k) g_{ijk} -$$
$$- \sum_{s=0}^{t-1} \xi(t-s) I_{\{x \ge t-m\}}(s) \left[w_{ijs} + \sum_{k=0}^{m-1} x_{sk} g_{ijk} \right] =$$
$$= \left[\xi(m) q_{ijt} - \sum_{s=0}^{t-1} \xi(t-s) I_{\{x \ge t-m\}}(s) w_{ijs} \right] +$$
$$+ \sum_{k=0}^{m-1} g_{ijk} \left[-\xi(m) d_{t-k} I_{\{x \le t\}}(k) - \sum_{s=0}^{t-1} \xi(t-s) I_{\{x \ge t-m\}}(s) x_{sk} \right].$$

So we have obtained

$$\begin{aligned} x_{tk} &= -\xi(m) d_{t-k} I_{\{x \le t\}}(k) - \sum_{s=max\{0,\ t-m\}}^{t-1} \xi(t-s) x_{sk},\ k = \overline{0, m-1}, \\ w_{ijt} &= \xi(m) q_{ijt} - \sum_{s=max\{0,\ t-m\}}^{t-1} \xi(t-s) w_{ijs},\ t = \overline{0, M-1},\ i, j = \overline{1, n}. \end{aligned} \tag{15}$$

For $t \ge M$ we have the transformations

$$q_{ijt} = \sum_{k=0}^{m-1} d_{t-k} I_{\{0 \le x \le M\}}(t-k) g_{ijk} + (\xi(m))^{-1} \cdot \sum_{s=0}^{M-1} \xi(t-s) I_{\{x \ge t-m\}}(s) \cdot$$
$$\cdot \left[w_{ijs} + \sum_{k=0}^{m-1} x_{sk} g_{ijk} \right] = (\xi(m))^{-1} \sum_{s=0}^{M-1} \xi(t-s) I_{\{x \ge t-m\}}(s) w_{ijs} +$$
$$+ \sum_{k=0}^{m-1} g_{ijk} \left[d_{t-k} I_{\{0 \le x \le M\}}(t-k) + (\xi(m))^{-1} \sum_{s=0}^{M-1} \xi(t-s) I_{\{x \ge t-m\}}(s) x_{sk} \right] \Leftrightarrow$$
$$\Leftrightarrow \sum_{k=0}^{m-1} r_{tk} g_{ijk} = s_{ijt},\ t = \overline{M, r-1},\ i, j = \overline{1, n}, \tag{16}$$

where

$$\begin{aligned} r_{tk} &= d_{t-k} I_{\{0 \le x \le M\}}(t-k) + (\xi(m))^{-1} \sum_{s=max\{0,\ t-m\}}^{M-1} \xi(t-s) x_{sk}, \\ s_{ijt} &= q_{ijt} - (\xi(m))^{-1} \sum_{s=max\{0,\ t-m\}}^{M-1} \xi(t-s) w_{ijs},\ k = \overline{0, m-1}. \end{aligned} \tag{17}$$

Now let us determine the values $\alpha_{ijk}(\mu),\ k=\overline{1,m},\ i,j=\overline{1,n}$. According to formula (14) we have

$$\frac{G_{ij}(z)}{(z-\mu)^m}=\sum_{k=1}^{m}\frac{\alpha_{i,j,k}(\mu)}{(z-\mu)^k}=\frac{1}{(z-\mu)^m}\sum_{k=1}^{m}\alpha_{i,j,k}(\mu)(z-\mu)^{m-k}\Leftrightarrow$$

$$\Leftrightarrow\sum_{s=0}^{m-1}g_{ijs}z^s=\sum_{k=1}^{m}\alpha_{i,j,k}(\mu)(z-\mu)^{m-k}=\sum_{k=0}^{m-1}\alpha_{i,j,m-k}(\mu)(z-\mu)^k=$$

$$=\sum_{k=0}^{m-1}\alpha_{i,j,m-k}(\mu)\sum_{s=0}^{k}C_k^s(-\mu)^{k-s}z^s=\sum_{s=0}^{m-1}z^s\sum_{k=s}^{m-1}\alpha_{i,j,m-k}(\mu)C_k^s(-\mu)^{k-s}\Leftrightarrow$$

$$\Leftrightarrow g_{ijs}=\sum_{k=s}^{m-1}C_k^s(-\mu)^{k-s}\alpha_{i,j,m-k}(\mu),\ s=\overline{0,m-1},\ i,j=\overline{1,n}.$$

If we substitute the expression of g_{ijs} in (16) then we obtain

$$s_{ijt}=\sum_{k=0}^{m-1}r_{tk}\sum_{s=k}^{m-1}C_s^k(-\mu)^{s-k}\alpha_{i,j,m-s}(\mu)=\sum_{s=0}^{m-1}\alpha_{i,j,m-s}(\mu)\sum_{k=0}^{s}C_s^k(-\mu)^{s-k}r_{tk}=$$

$$=\sum_{s=1}^{m}\alpha_{ijs}(\mu)\sum_{k=0}^{m-s}C_{m-s}^k(-\mu)^{m-s-k}r_{tk}=\sum_{s=1}^{m}r^*_{ts}\alpha_{ijs}(\mu),\ t=\overline{M,r-1},\ i,j=\overline{1,n},$$

where

$$r^*_{ts}=\sum_{k=0}^{m-s}C_{m-s}^k(-\mu)^{m-s-k}r_{tk},\ t=\overline{M,r-1},\ s=\overline{1,m}. \tag{18}$$

The solution of the system is

$$\alpha_{ij}(\mu)=(R^*)^{-1}S_{ij},\ i,j=\overline{1,n}, \tag{19}$$

where $\alpha_{ij}(\mu)=((\alpha_{ijs}(\mu))_{s=\overline{1,m}})^T$, $S_{ij}=((s_{ijt})_{t=\overline{M,r-1}})^T$ and $R^*=(r^*_{ts})_{t=\overline{M,r-1},\ s=\overline{1,m}}$.

3.3 The Main Conclusion and Description of the Algorithm

In Section 2.1 from [3] the numerical complex functions $\nu_k(z)=(1-z)^{-k}$, $\forall k\geq 1$ have been introduced. In [3] it was shown that these functions satisfy the recurrent relation $\nu_{k+1}(z)=\dfrac{d\nu_k(z)}{kdz}$, $\forall k\geq 1$. In addition it was shown that $\nu_k(z)=\sum\limits_{t=0}^{\infty}T_{k-1}(t)z^t$, $\forall k\geq 1$, where the coefficient $T_{k-1}(t)$ is a polynomial of degree less or equal to $k-1$. Moreover, the calculation formula for the elements $\beta_{ijk}(y)$ and for the corresponding matrices

$$W_{ij}(y,t)=\sum_{k=0}^{m(y)-1}(-y)^{-k-1}\alpha_{i,j,k+1}(y)T_k(t),\ \forall y\in\mathbb{C}\backslash\mathcal{D},\ i,j=\overline{1,n}$$

have been obtained.

Let $T_k(t)=\sum\limits_{s=0}^{k} u_s^{(k)} t^s,\ \forall k\ge 0$. Then

$$\nu_{k+1}(z)=\frac{d}{kdz}\sum_{t=0}^{\infty}T_{k-1}(t)z^t=\frac{1}{k}\sum_{t=1}^{\infty}tT_{k-1}(t)z^{t-1}=\frac{1}{k}\sum_{t=0}^{\infty}(t+1)T_{k-1}(t+1)z^t \Leftrightarrow$$

$$\Leftrightarrow T_k(t)=\frac{1}{k}(t+1)T_{k-1}(t+1)=\frac{1}{k}(t+1)\sum_{s=0}^{k-1}u_s^{(k-1)}(t+1)^s=$$

$$=\frac{1}{k}\sum_{s=0}^{k-1}u_s^{(k-1)}(t+1)^{s+1}=\frac{1}{k}\sum_{s=0}^{k-1}u_s^{(k-1)}\sum_{l=0}^{s+1}C_{s+1}^l t^l=$$

$$=\frac{1}{k}\sum_{s=0}^{k-1}u_s^{(k-1)}\left(1+\sum_{l=1}^{s+1}C_{s+1}^l t^l\right)=\frac{1}{k}\sum_{s=0}^{k-1}u_s^{(k-1)}+\frac{1}{k}\sum_{l=1}^{k}t^l\sum_{s=l-1}^{k-1}u_s^{(k-1)}C_{s+1}^l \Leftrightarrow$$

$$\Leftrightarrow u_0^{(0)}=1,\ u_0^{(k)}=\frac{1}{k}\sum_{s=0}^{k-1}u_s^{(k-1)},\ u_l^{(k)}=\frac{1}{k}\sum_{s=l-1}^{k-1}C_{s+1}^l u_s^{(k-1)},\ \forall k\ge 1,\ l=\overline{1,k}. \quad (20)$$

In such a way we obtain formula for calculating the elements of the matrices in the representation

$$\beta_{ijk}(y)=\sum_{s=k+1}^{m(y)}(-y)^{-s}\alpha_{ijs}(y)u_k^{(s-1)},\ y\in\mathbb{C}\backslash\mathcal{D}, k=\overline{0,m(y)-1}, i,j=\overline{1,n}. \quad (21)$$

Basing on result described above we can use the following algorithm for determining the limit and differential matrices in Markov chain

Algorithm 2. ***Determining the Limit and Differential Matrices***

Input Data: The matrix of probability transition P.
Output Data: The matrices $\beta_k(y)$ ($y\in\mathbb{C}\backslash\mathcal{D},\ k=\overline{0,m(y)-1}$).

1–3. Do steps $1-3$ of Algorithm 1;

4. Calculate the matrices $R^{(k)},\ k=\overline{0,n-1}$, according to formula (11);

5. Find the values $q_{ijk},\ k=\overline{0,r-1},\ i,j=\overline{1,n}$, using the calculation procedure described in Subsection 3.1;

6. Calculate $C_s^k,\ s=\overline{1,\max\limits_{y\in\mathbb{C}\backslash\mathcal{D}}m(y)},\ k=\overline{0,s}$, using Pascal's triangle rule;

7. Determine $u_l^{(k)},\ k=\overline{0,\max\limits_{y\in\mathbb{C}\backslash\mathcal{D}}m(y)-1},\ l=\overline{0,k}$, using formula (20);

8. For every $\mu\in\mathbb{C}\backslash\mathcal{D}$ do items a)–g):

a. Determine the values $\xi(k)=C_m^k(-\mu)^{-k},\ k=\overline{0,m}$ ($m=m(\mu)$);

b. Determine the coefficients $d_k,\ k = \overline{0, r-m}$, using Horner's scheme;

c. Calculate the values $x_{tk},\ t = \overline{0, M-1},\ k = \overline{0, m-1}$, according to (15);

d. Calculate the values $r_{tk},\ t = \overline{M, r-1},\ k = \overline{0, m-1}$, using formula (17);

e. Determine the elements of the matrix R^* according to relation (18);

f. Find the matrix $(R^*)^{-1}$ using known numerical algorithms;

g. For $i, j = \overline{1, n}$ do items $g_1) - g_4)$:

g_1. Calculate the values $w_{ijt},\ t = \overline{0, M-1}$, according to formula (15);

g_2. Calculate the values $s_{ijt},\ t = \overline{M, r-1}$, using formula (17);

g_3. Determine the vector $\alpha_{ij}(\mu)$ according to relation (19);

g_4. Calculate the elements $\beta_{ijk}(\mu)$ of the matrix $\beta_k(\mu),\ k = \overline{0, m(\mu)-1}$, according to formula (21).

For this algorithm we may give the same comments as for the previous algorithm. If the characteristic values of the matrix P are known then the algorithm finds the limit and differential matrices using $O(n^4)$ elementary operations. However this algorithm can be used also if only a subset of characteristic values of the matrix P is known; in this case the set $\mathbb{C} \setminus \mathbb{D}$ will consists of the inverses of known nonzero characteristic values and the algorithm will find the corresponding matrices which correspond to known characteristic values. The computational complexity of the algorithm in the case when the characteristic values are unknown depends on the complexity of determining the characteristic values.

4 Numerical Examples

We will illustrate the details of proposed algorithms for periodic and aperiodic Markov chains.

Example 1. Let be given the Markov process with the matrix of probability transition

$$P = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0.5 & 0.5 \\ 0.5 & 0 & 0.5 \end{pmatrix}$$

and consider the problem of determining the differential components of the matrix $P(t)$. We apply Algorithm 1 :

1) $P^2 = \begin{pmatrix} 1 & 0 & 0 \\ 0.25 & 0.25 & 0.5 \\ 0.75 & 0 & 0.25 \end{pmatrix},\ P^3 = \begin{pmatrix} 1 & 0 & 0 \\ 0.5 & 0.125 & 0.375 \\ 0.875 & 0 & 0.125 \end{pmatrix},$

$$s_1 = trP = 2,\ s_2 = trP^2 = 1.5,\ s_3 = trP^3 = 1.25,\ \overline{\beta}_0 = 1,\ \overline{\beta}_1 = -s_1 = -2,$$

$$\overline{\beta}_2 = -(s_2 + \overline{\beta}_1 s_1)/2 = 1.25,\ \overline{\beta}_3 = -(s_3 + \overline{\beta}_1 s_2 + \overline{\beta}_2 s_1)/3 = -0.25;$$

2–3) $\Delta(z)=\sum_{k=0}^{3}\overline{\beta}_k z^k = 1-2z+1.25z^2-0.25z^3=(1-z)(1-0.5z)^2\Rightarrow$

$$\Rightarrow \mathbb{C}\backslash\mathcal{D}=\{z\in\mathbb{C}\mid\Delta(z)=0\}=\{1,2\},\ m(1)=1,\ m(2)=2,\ r=n=3;$$

4) $\beta_0=(1,1,0),\ \beta_1=(1,0.5,0.5),\ \beta_2=(1,0.25,0.5)\Rightarrow B=\begin{pmatrix}1&1&0\\1&0.5&0.5\\1&0.25&0.5\end{pmatrix}$;

5) $(B^T)^{-1}=\begin{pmatrix}1&0&-2\\-4&4&6\\4&-4&-4\end{pmatrix}$;

6–7) $C_0^0=C_1^0=C_1^1=1,\ (r-n)^0=0^0=1,\ (r-n)^1=0^1=0;$

8a–8b) $\Gamma_{11}=I_3^{[\overline{a}_{11}]}(B^T)^{-1}=(1,1,1)\begin{pmatrix}1&0&-2\\-4&4&6\\4&-4&-4\end{pmatrix}=(1,0,0);$

$$\Gamma_{12}=(0,0,0)(B^T)^{-1}=(0,0,0);\ \Gamma_{13}=(0,0,0)(B^T)^{-1}=(0,0,0);$$

$$\Gamma_{21}=(0,0,0.25)(B^T)^{-1}=(1,-1,-1);\ \Gamma_{22}=(1,0.5,0.25)(B^T)^{-1}=(0,1,0);$$

$$\Gamma_{23}=(0,0.5,0.5)(B^T)^{-1}=(0,0,1);\ \Gamma_{31}=(0,0.5,0.75)(B^T)^{-1}=(1,-1,0);$$

$$\Gamma_{32}=(0,0,0)(B^T)^{-1}=(0,0,0);\ \Gamma_{33}=(1,0.5,0.25)(B^T)^{-1}=(0,1,0);$$

8c) $\beta_{ijk}(y)=\sum_{s=k}^{m(y)-1}0^{s-k}\gamma_{ijs}(y)=\gamma_{ijk}(y),\ \forall y\in\mathbb{C}\backslash\mathcal{D},\ k=\overline{0,m(y)-1},\ i,j=\overline{1,3}\Rightarrow$

$$\Rightarrow \beta_0(1)=\begin{pmatrix}1&0&0\\1&0&0\\1&0&0\end{pmatrix},\ \beta_0(0.5)=\begin{pmatrix}0&0&0\\-1&1&0\\-1&0&1\end{pmatrix},\ \beta_1(0.5)=\begin{pmatrix}0&0&0\\-1&0&1\\0&0&0\end{pmatrix}.$$

So, the transient matrix can be represented as follows:

$$P(t)=\begin{pmatrix}1&0&0\\1&0&0\\1&0&0\end{pmatrix}+\begin{pmatrix}0&0&0\\-1&1&0\\-1&0&1\end{pmatrix}\left(\frac{1}{2}\right)^t+\begin{pmatrix}0&0&0\\-1&0&1\\0&0&0\end{pmatrix}t\left(\frac{1}{2}\right)^t,\ \forall t\geq 0. \quad (22)$$

If we apply Algorithm 2 for the same example then we obtain:

1–3) $P^2=\begin{pmatrix}1&0&0\\0.25&0.25&0.5\\0.75&0&0.25\end{pmatrix}, P^3=\begin{pmatrix}1&0&0\\0.5&0.125&0.375\\0.875&0&0.125\end{pmatrix};\ \beta_0=1,$

$$\beta_1=-2,\ \beta_2=1.25,\ \beta_3=-0.25;\ \mathbb{C}\backslash\mathcal{D}=\{1,2\},\ m(1)=1,\ m(2)=2,\ r=n=3;$$

4–5) $R^{(0)}=\beta_0 I=\begin{pmatrix}1&0&0\\0&1&0\\0&0&1\end{pmatrix},\ R^{(1)}=\beta_1 I+PR^{(0)}=\begin{pmatrix}-1&0&0\\0&-1.5&0.5\\0.5&0&-1.5\end{pmatrix},$

$$R^{(2)}=\beta_2 I+PR^{(1)}=\begin{pmatrix}0.25&0&0\\0.25&0.5&-0.5\\-0.25&0&0.5\end{pmatrix};\ q_{ijk}=R_{ij}^{(k)},\ i,j=\overline{1,3},\ k=\overline{0,2};$$

6–7) $C_1^0 = C_1^1 = C_2^0 = C_2^2 = 1,\ C_2^1 = C_1^0 + C_1^1 = 2;\ u_0^{(0)} = u_0^{(1)} = u_1^{(1)} = 1;$
8$'$) $\mu = 1,\ m = m(\mu) = 1,\ M = r - m = 2;\ \xi(0) = 1,\ \xi(1) = -1;$

	-0.25	1.25	-2	1
1	-0.25	1	-1	0

$\Rightarrow\ d_0 = -1,\ d_1 = 1,\ d_2 = -0.25;$

$$x_{00} = -\xi(1)d_0 = -1,\ x_{10} = -\xi(1)d_1 - \xi(1)x_{00} = 0;$$

$$r_{20} = d_2 - \xi(1)x_{10} = -0.25,\ r_{21}^* = r_{20} = -0.25; R^* = (-0.25); (R^*)^{-1} = (-4);$$

$$w_{ij0} = -q_{ij0} = -R_{ij}^{(0)},\ w_{ij1} = -q_{ij1} + w_{ij0} = -R_{ij}^{(0)} - R_{ij}^{(1)};$$

$$s_{ij2} = q_{ij2} - w_{ij1} = R_{ij}^{(0)} + R_{ij}^{(1)} + R_{ij}^{(2)} = \begin{pmatrix} 0.25 & 0 & 0 \\ 0.25 & 0 & 0 \\ 0.25 & 0 & 0 \end{pmatrix}_{ij},$$

$$\alpha_{ij}(1) = (-4)(s_{ij2}) = \begin{pmatrix} -1 & 0 & 0 \\ -1 & 0 & 0 \\ -1 & 0 & 0 \end{pmatrix}_{ij},\ i,j = \overline{1,3};$$

$$\beta_{ij0}(1) = -\alpha_{ij}(1) = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}_{ij},\ i,j = \overline{1,3} \Rightarrow \beta_0(1) = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix};$$

8$''$) $\mu = 2,\ m = m(\mu) = 2,\ M = r - m = 1;\ \xi(0) = 1,\ \xi(1) = -1,\ \xi(2) = 0.25;$

	-0.25	1.25	-2	1
2	-0.25	0.75	-0.5	0
2	-0.25	0.25	0	

$\Rightarrow\ d_0 = 0.25,\ d_1 = -0.25;$

$$x_{00} = -\xi(2)d_0 = -0.0625,\ x_{01} = 0,\ r_{10} = 0,\ r_{11} = 0.25,\ r_{20} = -0.0625,$$

$$r_{21} = -0.25 \Rightarrow r_{11}^* = 0.25,\ r_{12}^* = 0,\ r_{21}^* = -0.125,\ r_{22}^* = -0.0625 \Rightarrow$$

$$\Rightarrow R^* = \begin{pmatrix} 0.25 & 0 \\ -0.125 & -0.0625 \end{pmatrix} \Rightarrow (R^*)^{-1} = \begin{pmatrix} 4 & 0 \\ -8 & -16 \end{pmatrix};$$

$$w_{ij0} = 0.25q_{ij0} = 0.25R_{ij}^{(0)},\ s_{ij1} = q_{ij1} + 4w_{ij0} = R_{ij}^{(0)} + R_{ij}^{(1)},$$

$$s_{ij2} = q_{ij2} - w_{ij0} = R_{ij}^{(2)} - 0.25R_{ij}^{(0)} \Rightarrow S_{ij} = \begin{pmatrix} R_{ij}^{(0)} + R_{ij}^{(1)} \\ R_{ij}^{(2)} - 0.25R_{ij}^{(0)} \end{pmatrix} \Rightarrow$$

$$\Rightarrow \alpha_{ij}(2) = (R^*)^{-1}S_{ij} = \begin{pmatrix} 4R_{ij}^{(0)} + 4R_{ij}^{(1)} \\ -4R_{ij}^{(0)} - 8R_{ij}^{(1)} - 16R_{ij}^{(2)} \end{pmatrix},\ i,j = \overline{1,3};$$

$$\beta_{ij0}(2) = -0.5\alpha_{ij1}(2) + 0.25\alpha_{ij2}(2) = -3R_{ij}^{(0)} - 4R_{ij}^{(1)} - 4R_{ij}^{(2)},$$

$$\beta_{ij1}(2) = 0.25\alpha_{ij2}(2) = -R_{ij}^{(0)} - 2R_{ij}^{(1)} - 4R_{ij}^{(2)},\ i,j = \overline{1,3} \Rightarrow$$

$$\Rightarrow \beta_0(2) = \begin{pmatrix} 0 & 0 & 0 \\ -1 & 1 & 0 \\ -1 & 0 & 1 \end{pmatrix}, \ \beta_1(2) = \begin{pmatrix} 0 & 0 & 0 \\ -1 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}.$$

So,we obtain the same representation (22) of the transient matrix $P(t)$.

Example 2. Let be given the 2-periodic Markov process determined by the matrix of probability transition

$$P = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

and consider the problem of determining the limit and differential components of the matrix $P(t)$. If we apply Algorithm 1 then we obtain:

1–3) $P^2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \ s_1 = trP = 0, \ s_2 = trP^2 = 2 \Rightarrow \overline{\beta}_0 = 1, \ \overline{\beta}_1 = -s_1 = 0,$

$$\overline{\beta}_2 = -(s_2 + \overline{\beta}_1 s_1)/2 = -1 \Rightarrow \Delta(z) = \sum_{k=0}^{2} \overline{\beta}_k z^k = 1 - z^2 = (1-z)(1+z) \Rightarrow$$

$$\Rightarrow \mathbb{C} \backslash \mathcal{D} = \{z \in \mathbb{C} \mid \Delta(z) = 0\} = \{1, -1\}, \ m(1) = m(-1) = 1, \ r = n = 2;$$

4–5) $\beta_0 = (1,1), \ \beta_1 = (1,-1) \Rightarrow B = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \Rightarrow (B^T)^{-1} = \begin{pmatrix} 0.5 & 0.5 \\ 0.5 & -0.5 \end{pmatrix};$

6–8) $C_0^0 = 1, \ (r-n)^0 = 0^0 = 1; \ \Gamma_{11} = \Gamma_{22} = (1,0) \begin{pmatrix} 0.5 & 0.5 \\ 0.5 & -0.5 \end{pmatrix} = (0.5, 0.5),$

$$\Gamma_{12} = \Gamma_{21} = (0,1) \begin{pmatrix} 0.5 & 0.5 \\ 0.5 & -0.5 \end{pmatrix} = (0.5, -0.5) \Rightarrow$$

$$\Rightarrow \beta_0(1) = \begin{pmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{pmatrix}, \ \Rightarrow \beta_0(-1) = \begin{pmatrix} 0.5 & -0.5 \\ -0.5 & 0.5 \end{pmatrix}.$$

So we obtain the following representation of the transient matrix:

$$P(t) = \begin{pmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{pmatrix} + \begin{pmatrix} 0.5 & -0.5 \\ -0.5 & 0.5 \end{pmatrix} (-1)^t, \ \forall t \geq 0. \tag{23}$$

If we apply algorithm 2 then we obtain:

1–3) $P^2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \ \beta_0 = 1, \ \beta_1 = 0, \ \beta_2 = -1 \Rightarrow \Delta(z) = 1 - z^2 = (1-z)(1+z) \Rightarrow$

$$\Rightarrow \mathbb{C} \backslash \mathcal{D} = \{1, -1\}, \ m(1) = m(-1) = 1, \ r = n = 2;$$

4) $R^{(0)} = \beta_0 I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \ R^{(1)} = \beta_1 I + PR^{(0)} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix},$

5–7) $q_{ijk} = R_{ij}^{(k)}, \ i,j = \overline{1,2}, \ k = \overline{0,1}; \ C_1^0 = C_1^1 = 1; \ u_0^0 = 1;$

8$'$) $\mu = 1, \ m = m(\mu) = 1, \ M = r - m = 1; \ \xi(0) = 1, \ \xi(1) = -1;$

	-1	0	1
1	-1	-1	0

$\Rightarrow\ d_0 = -1,\ d_1 = -1;$

$$x_{00} = -\xi(1)d_0 = -1,\ r_{10} = d_1 - \xi(1)x_{00} = -2 \Rightarrow r^*_{11} = -2 \Rightarrow$$

$$\Rightarrow R^* = (-2); (R^*)^{-1} = (-0.5);\ w_{ij0} = \xi(1)q_{ij0} = -R^{(0)}_{ij};$$

$$s_{ij1} = q_{ij1} + \xi(1)w_{ij0} = R^{(0)}_{ij} + R^{(1)}_{ij} \Rightarrow \alpha_{ij}(1) = (-0.5)(s_{ij1}) = -0.5R^{(0)}_{ij} - 0.5R^{(1)}_{ij};$$

$$\beta_{ij0}(1) = -\alpha_{ij1}(1) = 0.5R^{(0)}_{ij} + 0.5R^{(1)}_{ij},\ i,j = \overline{1,2} \Rightarrow \beta_0(1) = \begin{pmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{pmatrix};$$

$8'')$ $\mu = -1,\ m = m(\mu) = 1,\ M = r - m = 1;\ \xi(0) = 1,\ \xi(1) = 1;$

	-1	0	1
-1	-1	1	0

$\Rightarrow\ d_0 = 1,\ d_1 = -1;$

$$x_{00} = -\xi(1)d_0 = -1,\ r_{10} = d_1 + \xi(1)x_{00} = -2 \Rightarrow r^*_{11} = -2 \Rightarrow$$

$$\Rightarrow R^* = (-2); (R^*)^{-1} = (-0.5);\ w_{ij0} = \xi(1)q_{ij0} = R^{(0)}_{ij};$$

$$s_{ij1} = q_{ij1} - \xi(1)w_{ij0} = R^{(1)}_{ij} - R^{(0)}_{ij} \Rightarrow \alpha_{ij}(-1) = (-0.5)(s_{ij1}) = -0.5R^{(1)}_{ij} + 0.5R^{(0)}_{ij};$$

$$\beta_{ij0}(-1) = \alpha_{ij0}(-1) = -0.5R^{(1)}_{ij} + 0.5R^{(0)}_{ij},\ i,j = \overline{1,2} \Rightarrow \beta_0(-1) = \begin{pmatrix} 0.5 & -0.5 \\ -0.5 & 0.5 \end{pmatrix}.$$

So we obtain the representation (23).

References

[1] Howard R. A. *Dynamic Programming and Markov Processes.* Wiley, 1960.

[2] Puterman M. *Markov Decision Processes.* Wiley, 1993.

[3] Lozovanu D., Lazari A. *An Approach for Determining the Matrix of Limiting State Probabilities in Discrete Markov Processes.* Bulletin of the Academy of Science of RM, Matematica, 2010, No. 1(62), 77–91.

[4] Lazari A. *Caracteristicile probabilistice ale timpului de evoluţie al sistemelor aleatoare discrete.* Studia Universitatis, CEP USM, 2009, No. 2(22), 5–16.

[5] Helmberg G., Voltkamp G. *On Fadeev-Leverrier's Method for the Computation of the Characteristic Polynomial of the Matrix and of Eigenvectors.* Linear Algebra and its Application, 1993, No. 185, 219–233.

Alexandru Lazari
Moldova State University
60 Mateevici str., Chişinău, MD−2009
Moldova.
E-mail: *lazarialexandru@mail.md*

Received March 5, 2010