FUDMA Journal of Sciences (FJS)
ISSN online: 2616-1370
ISSN print: 2645 - 2944
Vol. 6 No. 1, March, 2022, pp 135 - 140
DOI: https://doi.org/10.33003/fjs-2022-0601-849

ON THE APPLICATION OF MATRIX SCALING AND POWERING METHODS OF SMALL STATE SPACES FOR SOLVING TRANSIENT DISTRIBUTION IN MARKOV CHAIN

***[1]Agboola, S. O. and [2]Nehad, A. S**
[1]Department of Mathematics, Faculty of Natural and Applied Sciences, Nigerian Army University Biu, No. 1 Biu – Gombe Road, PMB 1500 Biu, Borno State, Nigeria.
[2]Department of Mechanical Engineering, Sejong University Seoul 05006, Korea

*Corresponding authors' email: agboolasunday70@gmail.com Phone: +2348058222949

ABSTRACT

The iterative solution methods for transient distribution in Markov chain is the computation of state probability distributions at an arbitrary point of time, which in the case of a discrete-time Markov chain means, finding the distribution at some arbitrary time step n denoted $\pi^{(n)}$, a row vector whose i^{th} component is the probability that the Markov chain is in state i at time step n. In this study, the solutions of transient distribution in Markov chain using matrix scaling and powering methods for small state spaces which produce a significantly more accurate response in less time for some types of situations and also, tries to get to the end result as quickly as possible has been investigated, in order to provide some insight into the solutions of transient distribution of Markov chain. Our goal is to compute solutions and algorithms for tiny state spaces utilizing matrix scaling and powering approaches, which begin with an initial estimate of the solution vector and then comes closer and closer to the true solution with each step or iteration. With the help of several existing Markov chain laws, theorems, and formulas, matrices operations such as multiplication with one or more vectors, Padé variant of the matrix-powering and scaling technique are used. While the algorithms are explained, the transient distribution vector's $\pi^{(n)}$, $n = 1, 2, \ldots$, Padé approximants $R_{pq}(X)$, and a backward error analysis of the Padé approximation are obtained for certain illustrative examples..

Keywords: infinitesimal generator, linear combination, matrix-powering, matrix scaling, Padé approximants, uniformization method

INTRODUCTION

In the discipline of numerical analysis, there are two types of solution methods: iterative solution methods and direct solution methods. Iterative approaches start with an initial estimate of the solution vector and then alter it in such a way that it gets closer and closer to the genuine solution with each step or iteration. It eventually converges on the true solution. If there is no known initial approximation, a guess is performed or an arbitrary initial vector is used instead. The solution must be computed when a specified number of well-defined stages have been completed. The most widely utilized methods for deriving the stationary probability vector from either the stochastic transition probability matrix or the infinitesimal generator are iterative methods of one form or another. This decision was made for a variety of reasons. First, a look at the conventional iterative approaches reveals that the matrices are only involved in one operation: multiplication with one or more vectors, which leaves the transition matrices unchanged. When the transition matrix is large and not banded. Romanovsky (1970) established the application and simulation of discrete Markov Chains, while the stable recursion formulae for the steady state vector in Markov chains of M/G/1 type is presented by Ramaswami,and Neuts (1980) as well as Ramaswami (1988) respectively, this was followed by Stewart (1994, 2009) with the development of Numerical Solutions of Markov Chains and, Dayar (1998) helped in Permuting Markov chains to nearly completely decomposable form. while Pesch *et al.* (2015) demonstrated the appropriateness of the Markov chain technique in the wind feed in Germany. Uzun and Kiral (2017) used the Markov chain model of fuzzy state to anticipate the direction of gold price movement and to estimate the probabilistic transition matrix of gold price closing returns, whereas Azizah *et al.* (2019) used the Markov chain model of fuzzy state to predict monthly rainfall data. Clement (2019) demonstrated the application of Markov chain to the spread of disease infection, demonstrating that Hepatitis B became more infectious over time than tuberculosis and HIV, while Vermeer and Trilling (2020) demonstrated the application of Markov chain to journalism. Agboola (2021) introduced direct equation solving algorithms compositions of lower -upper triangular matrix and Grassmann–Taksar–Heyman for the stationary distribution of Markov chains while Agboola, and Ayoade (2021) analysed the matrix geometric and analytical block numerical iterative methods for stationary distribution in the structured Markov chains. Agboola and Ayinde (2021) demonstrated the performance measure analysis on the states classification in Markov chain while Agboola and Badmus (2021) established the application of renewal reward processes in homogeneous discrete Markov chain and, Agboola (2022) discussed the decomposition and aggregation algorithmic numerical iterative solution methods for the stationary distribution of Markov chain. However, in this study, the analysis of transient distribution in Markov chain using matrix scaling and powering methods for small state spaces is considered.

Notation

$\pi^{(n)}$, is transient distribution; $\pi_i(t)$, is the probability that the Markov chain is in state i at time t; π, is the stationary distribution; $R_{pq}(X)$, is Padé approximants; $P(i)$, is the transition probability matrix at step i and Q, is infinitesimal generator.

MATERIALS AND METHODS

The study area consisted of the analysis of matrix scaling and powering methods for small state spaces transient distribution in Markov chain. Transient Distribution in Markov Chain involves the computation of state probability distributions at an arbitrary point of time. In the case of a discrete-time Markov chain, this means finding the distribution at some arbitrary time step n. This distribution is denoted $\pi^{(n)}$, a row vector whose i^{th} component is the probability that the Markov chain is in state i at time step n. This satisfies the relationship

$$\pi^{(n)} = \pi^{(n-1)}P(n-1) = \pi^{(0)}P(0)P(1)\cdots P(n-1), \qquad (1)$$

where $P(i)$ is the transition probability matrix at step i. For a homogeneous discrete-time Markov chain, this reduces to

$$\pi^{(n)} = \pi^{(n-1)}P = \pi^{(0)}P^{n}, \qquad (2)$$

where $P(0) = P(1) = \cdots = P$. For a continuous-time Markov chain with infinitesimal generator Q, we seek the distribution at any time *t*. Such a distribution is denoted $\pi(t)$, a row vector whose component $\pi_i(t)$ is the probability that the Markov chain is in state i at time t and this vector satisfies the relationship

$$\pi(t) = \pi(0)e^{Qt} \qquad (3)$$

where e^{Qt} is the matrix exponential defined by

$$e^{Qt} = \sum_{k=0}^{\infty}\frac{(Qt)^k}{k!}. \qquad (4)$$

In both cases, what is usually required is seldom the probability distribution $\pi^{(n)}$ or $\pi(t)$ itself, but rather some linear combination of the components of these vectors, such as the probability that the Markov chain is in a single state i, ($\pi^{(n)}e_i$ or $\pi(t)e_i$, where e_i is a column vector whose elements are all zero except the i^{th} which is equal to 1) or the probability that the Markov chain is in a subset of states E_i $\left(\sum_{i\in E_i}\pi^{(n)}e_i \;\; or \; \sum_{i\in E_i}\pi(t)e_i\right)$ or yet again a weighted sum of the probabilities wherein the unit component of e_i is replaced by an arbitrary scaler. In addition, rather than only its (single) value at the final time point, the evolution of this statistic from the initial time to the desired time step *n* or time t may be required. Transient distributions of discrete-time Markov chains are rarely difficult to compute. To acquire the probability distribution at step k, multiply the probability distribution vector obtained at step $(k-1)$ with the stochastic transition probability matrix again for $k = 1,2,\ldots,n$. If n is large and the number of states in the Markov chain is small, the Markov chain will have a large number of states (not exceeding several hundreds). Then, by successively squaring the transition probability matrix j times, where j is the largest integer such that $2^j \leq n$, some computation time savings can be obtained. This produces the matrix P^{2j}, which may now be multiplied by P (and powers of P) to obtain the value P^n. $\pi^{(n)} = \pi^{(0)}P^n$ is now used to find the distribution at time step n. Because the sparsity of the matrix P is lost in the computation of P^n, this approach is not suitable for long sparse Markov chains. Furthermore, if a temporal trajectory of a distribution statistic is required, this approach may fall short, as only distributions at computed values of P^{2j} will be accessible. Finally, for large values of n, it may be helpful to keep a watch on convergence to the stationary distribution, as this may occur, correct to some acceptable computational accuracy, prior to step n. Any subsequent vector–matrix multiplications will have no effect on the distribution.

Matrix Scaling and Powering Methods for Small State Spaces

Moler and Van Loan (1978) present 19 dubious methods for computing the exponential of a small-order matrix. The accuracy of the approximations is largely dependent on the matrix's norm, which is a key flaw in all of these techniques. Attempting to compute e^{Qt} directly when the norm of Q or t is big is likely to produce unsatisfactory results. It becomes required to divide the interval [0,*t*] into subintervals (called panels) $[0,t_0],[t_0,t_1],\ldots,[t_{m-1},t_m = t]$ and compute the transient solution at each time t_j , $j = 0,1,\ldots,m$ using the solution at the start of the panel as the starting point.

This is commonly exactly what a user need, allowing them to track the evolution of specific system performance metrics over time. Matrix-scaling and -powering approaches are based on a feature of the exponential function that is unique to it, namely:

$$e^{Qt} = \left(e^{Qt/2}\right)^2 \qquad (5)$$

The main concept is to compute e^{Qt_0} for a small number t_0 such that $t = 2^m t_0$ and then build e^{Qt} by applying the equation (5) repeatedly. Let Q be the infinitesimal generator of a continuous-time, ergodic Markov chain, and $\pi(0)$ be the probability distribution at time *t* = 0. We're looking for $\pi(t)$, which is the transient solution at time t. Let m be an integer and $t_0 \neq 0$ be a time for which $t = 2^m t_0$. Then

$$\pi(t) = \pi(2^m t_0). \qquad (6)$$

By writing $t_j = 2t_{j-1}$, we will compute the matrices e^{Qt_j} for $j = 0,1,2,\ldots,m$ and, as a result, the transient solution at times $t_0,\; 2t_0,\; 2^2t_0,\ldots\;,2^m t_0 = t$. by multiplying with $\pi(0)$. $P(t_j) = e^{Qt_0}$ is a stochastic matrix, and the Chapman–Kolmogorov equations show that

$$P(t_j) = P(t_{j-1})P(t_{j-1}) \qquad (7)$$

After computing $P(t_0)$, each of the remaining $P(t_j)$ can be calculated using Equation (7) by squaring the previous $P(t_{j-1})$ As a result, matrix-powering methods supply the transient solution at intermediate intervals $t_0,\; 2t_0,\; 2^2t_0,\ldots\;,2^m t_0$ during their calculation. However, in addition to computational costs proportional to n^3 and memory needs of n^2, a downside of matrix-powering approaches is that repetitive squaring may cause rounding error buildup, especially in cases when $m \gg 1$.

RESULTS AND DISCUSSIONS

This section discusses the solutions for performance measures from the illustrative example such as stationary distribution π, transient distribution $\pi^{(n)}$, Padé approximant $R_{pq}(X)$, statistic $\pi^{(n)}a$, backward error analysis of the Padé approximation and algorithms

Illustrative Example 1: Given a discrete-time Markov chain with transition probability matrix P given by

$$P = \begin{pmatrix} 0.4 & 0 & 0.6 & 0 \\ 0.0002 & 0.3 & 0 & 0.6998 \\ 0.1999 & 0.0001 & 0.8 & 0 \\ 0 & 0.5 & 0 & 0.5 \end{pmatrix} \tag{8}$$

Assume that the Markov chain begins in state 1, i.e. $\pi^{(0)} = (1, 0, 0, 0)$, and that the value of being in each state at time step n is given by the vector $a = (0, 4, 0, 10)^T$. In other words, in states 1 and 3, the Markov chain is meaningless, but worth 4 (arbitrary units) in state 2 and 10 in state 4. $\pi^{(n)}a$ is the value at time step n, and we want to compute this statistic for different values of n. For small values of $n = 1, 2, \ldots$, we may compute $\pi^{(1)} = \pi^{(0)}P$, $\pi^{(2)} = \pi^{(1)}P$, $\pi^{(3)} = \pi^{(2)}P, \ldots$, and so on, and we get

$$\pi^{(1)} = (0.4 \quad 0 \quad 0.6 \quad 0)$$
$$\pi^{(2)} = (0.27994 \quad 0.00006 \quad 0.72 \quad 0)$$
$$\pi^{(3)} = (0.255904 \quad 0.00009 \quad 0.74396 \quad 0.000042)$$
$$\pi^{(4)} = (0.25108 \quad 0.000122 \quad 0.748714 \quad 0.000084)$$

which gives the following values of the statistic $\pi^{(n)}a$:

$$0,\ 0.00024,\ 0.00078,\ \ 0.001329, \ldots$$

These numbers can be graphed to show how a statistic has evolved over time. If the transient distribution is required for much larger n values, the small size of the matrix should be employed to compute the matrix's consecutive powers. For example, if the distribution is required at time step 1000, powering the matrix P yields $P^{(1000)}$. Furthermore, until that moment, the behavior of the statistic is

$$\pi^{(100)} = (0.248093 \quad 0.003099 \quad 0.744507 \quad 0.004251)$$
$$\pi^{(200)} = (0.246263 \quad 0.006151 \quad 0.739062 \quad 0.008524)$$
$$\pi^{(300)} = (0.244461 \quad 0.009156 \quad 0.733652 \quad 0.012731)$$
$$\pi^{(400)} = (0.242688 \quad 0.012113 \quad 0.728328 \quad 0.016871)$$
$$\vdots$$
$$\pi^{(1000)} = (0.232618 \quad 0.028906 \quad 0.698094 \quad 0.040382)$$

which gives the following values of the statistic $\pi^{(n)}a$:

$$0.0549,\ 0.109846,\ 0.163928,\ \ 0.217161, \ldots, 0.519446.$$

Finally, we note that the distribution at time step n = 1,000 differs significantly from the stationary distribution provided by $\pi = (.131589, .197384, .394768, .276259)$, with the statistic $\pi a = 3.552123$. For the calculation of transient distributions of continuous-time Markov chains, i.e., the computation of $\pi(t)$) from

$$\pi(t) = \pi(0)e^{Qt}$$

where Q is an irreducible continuous-time Markov chain's infinitesimal generator. $\pi(t)$ can be computed by first forming e^{Qt} and then pre-multiplying this with the initial probability vector, depending on the numerical approach used $\pi(0)$. The matrix-scaling and powering methods are examples of this. When the transition rate matrix is modest, they are the best choice. In some circumstances, $\pi(t)$can be computed without having to create e^{Qt} explicitly. The uniformization method and ordinary differential equation (ODE) solvers both use this methodology. Both can be used to create large-scale Markov chains.

Illustrative Example 2: Assuming we need to find the transition distribution of a Markov chain with infinitesimal generator Q at time *t* = 10.

$$Q = \begin{pmatrix} -0.6 & 0 & 0.6 & 0 \\ 0.1 & -0.9 & 0.1 & 0.7 \\ 0.4 & 0.3 & -0.8 & 0.1 \\ 0 & 0.5 & 0 & -0.5 \end{pmatrix} \tag{9}$$

We must find a t_0 such that t_0 is tiny and an integer m exists, resulting in $t = 2^m t_0$, in order to use the matrix-scaling and -powering strategy. When we set *m* = 5 and solve for t_0, we get $t_0 = \frac{10}{32}$, which is small enough for our scenario. Therefore,

$$P(t_0) = \left(e^{10Q/32}\right) = \begin{pmatrix} 0.83864 & 0.007076 & 0.151351 & 0.002933 \\ 0.026528 & 0.769552 & 0.26528 & 0.17393 \\ 0.10208 & 0.074581 & 0.789369 & 0.03397 \\ 0.002075 & 0.126413 & 0.002075 & 0.869437 \end{pmatrix} \tag{10}$$

and that performing the operations $P(t_j) = P(t_{j-1})P(t_{j-1})$ for $j = 0, 1, 2, \ldots, 5$ gives the result

$$P(t_5) = \begin{pmatrix} 0.175703 & 0.265518 & 0.1757 & 0.383082 \\ 0.136693 & 0.289566 & 0.136659 & 0.437156 \\ 0.161384 & 0.274324 & 0.16139 & 0.402903 \\ 0.129865 & 0.293702 & 0.129865 & 0.446568 \end{pmatrix} = \left(e^{10Q}\right). \tag{11}$$

The distribution at time *t* = 10 can be found by pre-multiplying by an initial probability distribution. The computation of e^{Qt_0} is now the focus of our attention. Methods based on approximations about zero are feasible possibilities since t_0 is modest. The rational Padé approximations around the origin are excellent option. The unique (*p*, *q*) rational function $R_{pq}(X)$ is the (*p*, *q*) Padé approximant to the matrix exponential e^X.

$$R_{pq}(X) \equiv \frac{N_{pq}(X)}{D_{pq}(X)}, \tag{12}$$

which matches the Taylor's series expansion of e^X through terms to the power$(p + q)$. Its coefficients are determined by solving the algebraic equations

$$\sum_{j=0}^{\infty} \frac{X^j}{j!} - \frac{N_{pq}(X)}{D_{pq}(X)} = O(X^{p+q+1}), \tag{13}$$

which yields

$$N_{pq}(X) = \sum_{j=0}^{\infty} \frac{(p+q-j)!p!\ X^j}{(p+q)!\ j!\ (p-j)!} \tag{14}$$

and

$$D_{pq}(X) = \sum_{j=0}^{\infty} \frac{(p+q-j)!q!(-X)^j}{(p+q)!\ j!\ (q-j)!}. \tag{15}$$

Baker (1975) has further information about Padé approximants, and one of their primary drawbacks is that they are only accurate at the origin, therefore they should not be employed when $\|X\|_2$ is large. We can choose t_0 so that $\|Qt_0\|_2$ is small enough that the Padé approximant to e^{Qt_0} can be derived with acceptable accuracy, even for relatively low-degree approximants, because we will be employing them in the context of a matrix-scaling and -powering method. The diagonal Padé approximants are obtained when $p = q$, and there are two main reasons why this choice is preferable. They are, first and foremost, more stable. All of the eigenvalues of $X = Qt$ in Markov chain problems can be found in the left half plane. Because either $p > q$ and cancellation difficulties may occur, or $p < q$ and $D_{pq}(X)$ may be inadequately conditioned, the computed approximants $R_{pq}(X)$ for $p \neq q$ have greater rounding errors. Second, using the same amount of computation, we obtain a higher-order approach. $R_{pq}(X)$ with $p < q$ takes around qn^3 flops to compute and provides an approximant of order $(p + q)$. The number of flops required to compute $R_{qq}(X)$ is nearly the same, but the result is an approximant of order $2q > (p+q)$. When $p > q$ similar statements can be made. We find diagonal Padé approximants

$$R_{pp}(X) \equiv \frac{N_{pp}(X)}{N_{pp}(-X)}, \tag{16}$$

Where

$$N_{pp}(X) = \sum_{j=0}^{p} \frac{(2p-j)!p!\ X^j}{(2p)!\ j!\ (p-j)!} = \sum_{j=0}^{p} c_j\, X^j. \tag{17}$$

The coefficients c_j can be conveniently constructed by means of the recursion

$$c_0 = 1; \quad and\ c_j = c_{j-1} \frac{p+1-j}{j(2p+1-j)}. \tag{18}$$

For real implementation, the irreducible form below gives significant computation time savings at the tradeoff of more memory locations:

$$R_{pp}(X) = \begin{cases} 1 + 2\dfrac{X\sum_{K=0}^{\frac{P}{2}-1} c_{2k+1}X^{2k}}{\sum_{K=0}^{\frac{P}{2}} c_{2k}X^{2k} - X\sum_{K=0}^{\frac{P}{2}-1} c_{2k+1}X^{2k}} & if\ p\ is\ even \\ -1 - 2\dfrac{X\sum_{K=0}^{\frac{P}{2}-1} c_{2k}X^{2k}}{X\sum_{K=0}^{\frac{P}{2}} c_{2k+1}X^{2k} - \sum_{K=0}^{\frac{P}{2}-1} c_{2k}X^{2k}} & if\ p\ is\ odd \end{cases} \tag{19}$$

Thus, for even values of *p*,

$$R_{pp}(X) = I + 2\frac{S_e}{T_e - S_e}, \tag{20}$$

Where

$S_e = c_1X + c_3X^3 + \ldots + c_{p-1}X^{p-1}$ and $T_e = c_0 + c_2X^2 + c_4X^4 + \ldots + c_pX^p$ (21)

while for odd values of *p*,

$$R_{pp}(X) = I + 2\frac{S_0}{T_0 - S_0}, \tag{22}$$

Where

$S_0 = c_0 + c_2X^2 + c_4X^4 + \ldots + c_{p-1}X^{p-1}$ and $T_0 = c_1X + c_3X^3 + \ldots + c_pX^p$ (23)

These computations can be merged easily, necessitating the use of a Horner-type evaluation procedure. Horner evaluations of the numerator and denominator in Equation (19), for example, require only half the operations of a conventional implementation of Equation (16). For the computation of e^X, the following four phases, modified from Philippe and Sidje (1993), create a Padé form of the matrix-powering and -scaling approach. The integer *m* is chosen as $m = [log\|X\|_\infty / log2] + 1$. in this implementation.

To compute the transient solution of a Markov chain with generator *Q* and starting state $\pi(t_0)$, at time *t* just use this approach with $X = Qt$ and then create $\pi(t_0)R$ where *R* is the algorithm's approximation to e^X.

1. Find appropriate scaling factor:

    - Compute$m = \max\{0,\ [log\|X\|_\infty/log2] + 1\}$.

2. Compute coefficients and initialize:

```
        • Set $c_0 = 1$.
        • For $j = 1,2,\ldots,p$ do
        ◦ Compute $c_j = c_{j-1}\frac{p+1-j}{j(2p+1-j)}$.
        • Compute $X_1 = 2^{-m};\ X_2 = X_1^2;\ \ T = c_pI;\ \ S = c_{p-1}I$ .
3. Application of Horner scheme:
        • Set odd = 1.
        • For $j = p - 1,\ldots,2,1$ do
        ◦ if odd = 1, then
        ∗ Compute $= T \times X_2 + c_{j-1}I$ ;
else
        ∗ Compute $= S \times X_2 + c_{j-1}I$ .
        ◦ Set odd = 1 − odd.
        • If odd = 0, then
        ◦ Compute $S = S \times X_1;\ R = I + 2 \times (T - S)^{-1} \times S$;
else
        ◦ Compute $T = T \times X_1;\ R = -(I + 2 \times (T - S)^{-1} \times S)$
4. Raise matrix to power 2m by repeated squaring:
        • For j = 1 to m do
        ◦ Compute $R = R \times R$.                Philippe and Sidje (1993)
```

For e^X the Padé approximation needs about $(p + m + 4/3)n^3$ multiplications. In addition to the storage necessary for the matrix, it can be implemented with three double-precision arrays, each of size n^2. As a result, we are left with the option of p. Moler and Van Loan (1978) (1978) gave a backward error analysis of the Padé approximation, demonstrating that if $\frac{\|X\|_2}{2^m} \le \frac{1}{2}$, then

$$\left[R_{pp}(2^{-m}X)\right]^{2m} = e^{X+E},$$

Where

$$\frac{\|E\|_2}{\|X\|_2} \le \left(\frac{1}{2}\right)^{2p-3} \quad \frac{(p\,!)^2}{(2p\,!)(2p+1)!} \approx \begin{cases} 0.77\times 10^{-12} & (p=5)\\ 0.34\times 10^{-15} & (p=6)\\ 0.11\times 10^{-18} & (p=7)\\ 0.27\times 10^{-22} & (p=8)\end{cases} \tag{24}$$

This suggests that Padé approximants of low degree for value of $p = 5$ to $p = 8$ are adequate. The above analysis, however, does not account for rounding error. Ward (1977) looked into this and proposed some criteria for selecting appropriate values for certain computers. Saff (1973) also provides a discussion on the degree of greatest rational approximation to the exponential function. Finally, Philippe and Sidje (1993) found that even values of p are better than odd values, and that p = 6 is generally satisfactory in numerical experiments on Markov chains.

CONCLUSION

The solutions of transient distribution in Markov chain using matrix scaling and powering methods for small state spaces, which produce a significantly more accurate response in less time for some types of situations and also, tries to get to the end result as quickly as possible while the solution must be computed when a specified number of well-defined stages have been completed has been investigated, in order to provide some insight into the solutions of transient distribution in Markov chain. Our quest is to compute the solutions and algorithms of matrix scaling and powering methods. Matrices operation such as multiplication with one or more vectors, Padé variant of the matrix-powering and scaling approach are used with the help of some existing laws, theorems and formulas of Markov chain. The transient distribution vector's $\pi^{(n)}$, $n = 1, 2, ...,$ Padé approximants $R_{pq}(X)$ and a backward error analysis of the Padé approximation are obtained for some illustrative examples while the algorithms is presented. The first illustrative example was considered on Markov chain which begins in state 1, i.e. $\pi^{(0)} = (1, 0, 0, 0)$, and that the value of being in each state at time step n is given by the vector $a = (0, 4, 0, 10)^T$. $\pi^{(n)}a$ statistic which is the value at time step n were computed for different values of n and, it was observed that for $n = 1000$, $\pi^{(1000)} = (0.232618 \quad 0.028906 \quad 0.698094 \quad 0.040382)$, the statistic $\pi^{(n)}a$ is $0.0549,\ 0.109846,\ 0.163928,\ 0.217161, \ldots, 0.519446$. While the second illustrative example considered the transition distribution of a Markov chain with infinitesimal generator Q at time *t* = 10 for continuous distribution Markov chain using Padé approximants technique and, it was found that even values of p are better than odd values, and that p = 6 is generally satisfactory in numerical experiments on Markov chains.

REFERENCES

Agboola, S. O. (2022). The Decomposition and Aggregation Algorithmic Numerical Iterative Solution Methods for the Stationary Distribution of Markov Chain, *Journal of Scientific and Engineering 2022, 9(1): 116 - 123. CODEN (USA): JSERBR. www.jsaer.com.*

Agboola, S. O. (2021). Direct Equation Solving methods Algorithms Compositions of Lower -Upper Triangular Matrix and Grassmann–Taksar–Heyman for the stationary Distribution of Markov chains, *International Journal of Applied Science and Mathematics (IJASM),* Volume 8, Issue 6, pp. 87 – 96. *www.ijasm.org*.

Agboola, S. O. and Ayoade, A. A. (2021). On the Analysis of Matrix Geometric and Analytical Block Numerical Iterative Methods for Stationary Distribution in the Structured Markov

Chains, *International Journal of Contemporary Applied Researches (IJCAR),* **8**(11): 51 – 65, Turkey. http://www.ijcar.net

Agboola, S. O. and Ayinde, S. A. (2021). The Performance Measure Analysis on the States Classification in Markov Chain, *Dutse Journal of Pure and Applied Sciences (DUJOPAS), Faculty of Science Journal, Federal University Dutse, Jigawa State.* **7**(04b): 19 – 29. *https://fud.edu.ng/dujopas.*

Agboola, S. O. and Badmus, N. I. (2021). Application of Renewal Reward Processes in Homogeneous Discrete Markov Chain, *FUDMA Journal of Sciences (FJS), Faculties of Earth Science and Physical Science Journal, Federal University DutsinMa.* ***5****(4): 210 – 215. https://fjs.fudutsinma.edu.ng*

Azizah, A., Welastica, R., Nur, F., Ruchjana, B. and Abdullah, A. (2019). An application of Markov chain for predicting rainfall data at West Java using data mining approach, *Earth and Environmental Science*, **303**(1): 203 – 216.

Baker, A. (1975). *Essentials of Padé Approximants*. Academic Press, New York,

Clement, T. (2019). Markov chain modelling of HIV, Tuberculosis, and Hepatitis B transmission in Ghana, *Hindawi*, *Interdisciplinary Perspective on Infectious Disease,* **27**(1)**:** 204 – 214.

Dayar., T. (1998). Permuting Markov chains to nearly completely decomposable form. *Technical Report BU-CEIS-9808*, Department of Computer Engineering and Information Science, Bilkent University, Ankara, Turkey. pp. 18 – 31.

Moler, C and Van Loan, C (1978). Nineteen dubious ways to compute the exponential of a Matrix. *SIAM Review,* **20(**4): 801 – 836.

Pesch, T., Schroder, S., Allelein, H. and Hake, J. (2015). A new Markov chain related statistical approach for modelling synthetic wind power time series, *New Journal of Physics, Dentsche Physikalishe,* **35**(2)**:** 64 – 85.

Philippe, B and Sidje, B. (1993) *Transient Solution of Markov Processes by Krylov Subspaces*, Technical Report, *IRISA*—Campus de Beaulieu, Rennes, France. pp. 11 – 24
.

Ramaswami, V. (1988). A Stable Recursion for the Steady State Vector in Markov chains of M/G/1 type. *Communication in Statist. Stochastic Models*, **4**(1)**:** 183–188.

Ramaswami, V and Neuts, M. F. (1980). Some explicit formulas and computational methods for infinite server queues with phase type arrivals. *Journal of Applied Probability,* **17**(1)**:** 498–514.

Romanovsky, V.I. (1970). *Discrete Markov Chains*, Wolters-Noord off, Groningen, Netherlands. pp. 23 – 44.

Saff, E. (1973). On the Degree of the Best Rational Approximation to the Exponential Function. *Journal of Approximation Theory*, **9**(1): 97 – 101.

Stewart, W. J. (1994). *Introduction to the Numerical Solution of Markov Chains*. Princeton University Press, Princeton, N.J. pp. 14 – 38.

Stewart, W. J. (2009). *Probability, Markov Chain, Queues and Simulation*, Princeton University Press, United Kingdom. pp. 1- 42.

Uzun, B. and Kiral, E. (2017). Application of Markov chain-fuzzy states to gold price, *Science Direct. ELSEVIER,* **120**(1): 365 – 371.

Vermeer, S. And Trilling, D. (2020): Toward a better understanding of a new user journeys: A Markov chain approach. *Journalism Journal*, **21**(1): 879 – 894.

Ward, R (1977). Numerical Computation of the Matrix Exponential with Accuracy Estimate, *SIAM Journal on Numerical Analysis*, **14**(4): 600 – 610.

Zakaria, N. N., Mahmod, O., Rajalmgan, S., Hamita, D., Lazim, A. and Evizal, A. (2019). Markov chain model development for forecasting air pollution index of Miri, Sarawak, *Sustainability* **11**(1)**:** 5190 -5202.