



## APPLICATION OF SINGLE STEP EULER AND TRAPEZOID METHODS OF SOLVING TRANSIENT DISTRIBUTION IN MARKOV CHAIN

Agboola Sunday. O.<sup>1</sup> Adebisi. Oyeyemi. O.<sup>2</sup> and Obaromi Davies A.<sup>3</sup>

Page | 4669 <sup>1</sup>Department of Mathematics, Faculty of Natural and Applied Sciences, Nigerian Army University Biu,

<sup>2</sup>Department of Economics, Faculty of Business & Social Sciences, Adeleke University Ede, Nigeria

<sup>3</sup>Department of Biostatistics, Faculty of Sciences, Federal University of Health Sciences Otuipo,

*Date Received:* 08-02-2023

*Date Accepted:* 09-05-2023

**DOI:** <https://doi.org/10.48198/NJPAS/23.A08>

### ABSTRACT

The computation of state probability distributions at an arbitrary point in 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  $n$ , and this is the iterative solution methods for transient distribution in Markov chain. In this study, the solutions of transient distribution in Markov chain using Euler and trapezoid methods have been investigated, in order to provide some insight into the solutions of transient distribution in Markov chain, 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 with the solution conform with a specified number of well-defined stages is computed for large space Markov chain. Matrices operations, such as the product and matrix inversion are performed and Markov chain laws, theorems, formulas with MATLAB software are utilized. For illustrative examples, the transient distribution vector's  $\pi_{(i+1)}$ ,  $i = 0, 1, 2, \dots$ ; is computed for both Euler and Trapezoid methods and their corresponding error when compared with different method. The effect of decreasing the step size using the same infinitesimal generator is examined, also, by taking  $\pi_{(0)} = (1 \ 0 \ 0)$  and the length of the interval of integration to be  $\tau = 1$ , it is observed that the accuracy achieved with the explicit Euler method has a direct relationship with the step size. It is also concluded that the much more accurate results are obtained with the trapezoid method.

**Keywords:** Chapman-Kolmogorov equation, finite different method, infinitesimal generator, single step Euler method, transient distribution, trapezoid rule

### Introduction

The solution of ordinary differential equations (ODEs) has triggered a lot of interest as there are a lot of ways to use ODE procedures to compute transient solutions of Markov chains. Numerical methods for the solution of ODEs, unlike uniformization and matrix scaling and powering,

are applicable to nonhomogeneous Markov chains, i.e., Markov chains whose infinitesimal generators are a function of time  $Q(t)$ . 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

**Corresponding Author: Agboola, S.O**

Department of Mathematics, Faculty of Natural and Applied Sciences, Nigerian Army University  
Biu. Email: [sunday.agboola@naub.edu.ng](mailto:sunday.agboola@naub.edu.ng)

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 Stewart (2009). The solution must be computed when a specified number of well-defined stages have been completed.

For large-scale Markov chains, methods currently used to obtain transient solutions are based either on readily available differential equation solvers such as Runge-Kutta methods or the Adams formulae and backward differentiation formulae (BDF). Most of these methods experience difficulty when both  $\max_j |q_{jj}|$  (the largest exit rate from any state) and  $t$  (the time at which the solution is required) are large, and there appears to be little to recommend a single method for all situations. The Euler and Trapezoid methods are extremely simple to program and often outperforms other methods, and these two methods are useful particularly if the solution is required at many points, or if plots need to be drawn to show the evolution of certain performance measures Stewart (1994).

Romanovsky (1970) established the application and simulation of discrete Markov Chains while Norm and John (1989) showed that the integral of an analytic function over the entire real line may be approximated by the trapezoidal rules, and also provides a more accurate method of approximation. Transient solution of Markov Processes by Krylov Subspaces was derived Philippe and Sidje (1993), while Numerical Solutions of Markov chains was initiated by Stewart (1994, 2009), and the Permutation of Markov chains to nearly completely decomposable is discussed by Dayar (1998), while Kanwar and Tomar (2008) presented a new class of Euler's method that are single step and do not require evaluation of derivatives, the accuracy and stability of the proposed methods are considered and applied to stiff problems.

Roberto (2015) described different approaches to generalize the trapezoidal method to fractional differential equation and implement efficient algorithms with illustrative examples, while Pesch *et al.* (2015) demonstrated the appropriateness of the Markov chain technique in the wind feed in Germany, and Matthew and Paul (2017) derived the trapezoidal rule to compute fractional integrals on evenly spaced point, and derivatives to within a given tolerance error, while 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.

Also, Azizah *et al.* (2019) used the Markov chain model of fuzzy state to predict monthly rainfall data, and Clemence (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 Nurujjaman (2020) illustrated the geometrical interpretation of modified Euler's methods and find a way to reduce the error, and Xing *et al.* (2020) improved the accuracy of Euler method by approximating the local truncation error with deep neural network. Yue (2021) proposed a randomized trapezoidal quadrature for continuous functions in some fractional sobolev space to established error bounds for the randomized rule and classical trapezoidal quadrature, while Abushet (2021) explained a modification to the Modified Euler method by testing the stability and consistency to determine the end result, and some numerical results were presented. Agboola (2021) computed the algorithms for Gaussian elimination method and lower – upper triangular matrix approach with the solution of stationary distribution of Markov chain demonstrated when a specified number of well-defined stages are completed.

Agboola and Badmus (2021) demonstrated the importance of multistep backward differentiation formulae (BDF) method in solving stiff systems

(ordinary differential equations), and it was concluded that the solutions of stiff systems of differential equations contain rapidly decaying transient terms. Agboola and Ayinde (2021) investigated the concept of the classification of groups of states, between states that are recurrent and states that are transient, in order to provide some insight into the performance measure analysis for the mean first passage time,  $R_{ij}$ , the mean recurrence time of state  $R_{jj}$  as well as recurrence iterative matrix  $R(k+1)$ .

Agboola and Ayinde (2022) demonstrated the iterative solution methods using the normalization principle, Matrix operations such as Lower, upper and diagonal matrices, some existing laws, theorems and formulas of Markov chain to obtain the stationary distribution vector's  $\pi^{(k+1)}$  of Markov chain. Agboola and Nehad (2022) computed 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.

Agboola (2022) examined the states of the Markov chain with the integers  $0, \pm 1, \pm 2, \dots$  (the drunkard's straight line) in order to determine the player that ends up with all the cards, while Agboola *et al.* (2022) computed solutions and algorithms for large state spaces using uniformization methods with the help of matrix operations, Markov chain laws, theorems, and formulas. Agboola *et al.* (2023) formulated the solutions of stationary distribution in Markov chain using Jacobi iterative method and Gauss-Seidel iterative method with the application of normalization principle and matrix operations. However, in this study, the application of single step Euler and trapezoid methods of solving transient distribution in Markov chain is considered, while the transient distribution vector's  $\pi_{(i+1)}$ ,  $i = 0, 1, 2, \dots$ ; and their corresponding

error when compared with different methods are computed,

**Nomenclature**

$h_i$ , step size at step  $i$ ;  $q_{jj}$ , exit rate;  $Q$ , infinitesimal generator;  $z_{i+1}$ , finite element method;  $z^l = f(t)$ , first-order differential equation;  $z(t)$ , is the row vector  $\pi(t)$  of the solution in Markov chain with function  $f(t, z(t))$ ;  $\pi_{(i+1)}$ , Euler and Trapezoid solution method;  $\pi$ , is the stationary distribution and  $\pi(t)$ , is the transient solution at time  $t$ ;

**Materials and Methods**

The study area consisted of the application of single step Euler and trapezoid methods of solving transient distribution in Markov chain. Given the transient distribution we seek to compute to be  $\pi(t) = \pi(0)e^{Qt}$ , which is the solution of the Chapman–Kolmogorov differential equations

$$\frac{d\pi(t)}{dt} = \pi(t)Q$$

with initial conditions

$$\pi(t = 0) = \pi(0).$$

Given a first-order differential equation  $z^l = f(t)$  and an initial condition  $z(t_0) = z_0$ , a solution is a differentiable function  $z(t)$  such that

$$z(t_0) = z_0, \quad \frac{dz(t)}{dt} = f(t, z(t)). \tag{1}$$

The solution  $z(t)$  is the row vector  $\pi(t)$  in the context of Markov chains, and the function  $f(t, z(t))$  is straightforward  $\pi(t)$ . When introducing general notions, we will utilize Equation (1)'s conventional notation, and the Markov chain notation when the situation calls for it, such as in examples. To compute the solution to Equation (1), numerical algorithms seek to follow a unique solution curve from its value at an initially defined point to its value at another prescribed location  $\tau$ . This usually entails performing a discretization technique on the interval  $[0, \tau]$  as well as computing approximations to the answer at intermediate places.

The exact solution of the differential Equation (1) at time  $t_i$  is denoted by  $z(t_i)$  given a discrete set of points (or mesh)  $\{0 = t_0, t_1, t_2, t_3, \dots, t_n\}$  in  $[0, \tau]$ , The step size or panel width at step  $i$  is defined as  $h_i = t_i - t_{i-1}$ . A numerical approach yields the sequence  $\{z_1, z_2, z_3, \dots, z_n\}$  where  $z_1$  is a close approximation to  $z(t_1)$ . In certain cases, the solution at time  $\tau$  is all that is necessary, and the computed solution is assumed to be  $z_n$  in those cases. In other circumstances, the value of  $z(t)$  for all  $t_0 \leq t \leq \tau$  is required, which is usually in the form of a graph, and this is obtained by fitting a suitable curve to the values  $z_1, z_2, z_3, \dots, z_n$ .  $z_i, i \neq 0$  represents a computed approximation to the exact value  $z(t_i)$ . The initial condition for  $i = 0$  is  $z_0 = z(t_0)$ .

**Theory/Calculation**

**Single-Step Euler Method**

If the solution is continuous and differentiable, we can approximate the solution curve by its tangent  $z_0^l$  at  $(t_0, z_0)$  in a small neighborhood of the point  $(t_0, z_0)$  and so proceed from  $(t_0, z_0)$  to the next point  $(t_1, z_1)$ . Because this implies that

$$z_0^l = \frac{z_1 - z_0}{t_1 - t_0}$$

we obtain the formula

$$z_1 = z_0 + h_1 z_0^l = z_0 + h_1 f(t_0, z_0), \tag{2}$$

where  $h_1 = t_1 - t_0$ . This process may be repeated from the computed approximation  $z_1$  to obtain

$$z_2 = z_1 + h_2 f(t_1, z_1), \tag{3}$$

where  $h_2 = t_2 - t_1$ , and so on until the final destination is reached,

$$z_n = z_{n-1} + h_n f(t_{n-1}, z_{n-1}), \tag{4}$$

With  $h_n = t_n - t_{n-1}$ .

This method is called the forward Euler method (finite element method) or explicit Euler method. From Equations (2) – (4), the Finite element method may be written as

$$z_{i+1} = z_i + h_{i+1} f(t_i, z_i), \quad i = 0, 1, 2, \dots, n - 1, \tag{5}$$

where  $h_{i+1} = t_{i+1} - t_i$ . In computing  $z_{i+1}$ . A technique may include the values of previously computed approximations  $z_j$  for  $j = 0, 1, 2, \dots, i$ , or even previous approximations to  $z(t_{i+1})$  in computing  $z_{i+1}$ . The term "explicit single-step approach" refers to a method that computes  $z_{i+1}$  using only  $(t_i, z_i)$ . If it employs approximations from several prior steps to compute its current estimate, it is considered to be a multistep approach. If computing  $z_{i+1}$  requires an approximation to  $z(t_{i+1})$ , the method is said to be implicit; otherwise, it is said to be explicit. The slope at the point  $(t_{i+1}, z_{i+1})$  is used to define an implicit Euler method, also known as a backward Euler method.

This gives

$$z_{i+1} = z_i + h_{i+1} f(t_{i+1}, z_{i+1}), \tag{6}$$

The modified Euler approach uses the average of the slopes at both places in the hopes of getting a better average approximation of the slope throughout the full panel  $[t_i, t_{i+1}]$ . The formula is as follows:

$$z_{i+1} = z_i + h_{i+1} \frac{f(t_i, z_i) + f(t_{i+1}, z_{i+1})}{2}. \tag{7}$$

The trapezoid rule is another name for this. The implicit midpoint rule, which involves the slope at the interval's midway, is a last formula in this category. The formula is as follows:

$$z_{i+1} = z_i + h_{i+1} f\left(t_i + \frac{h_i}{2}, \frac{z_i + z_{i+1}}{2}\right), \tag{8}$$

Because the approximations preceding  $(t_i, z_i)$  are not employed in the production of the next estimate  $z_{i+1}$ , these are all single-step approaches. The explicit Euler technique (Equation 5) is presented in Markov chain issues.

$$\pi_{(i+1)} = \pi_i + h_{i+1} \pi_i Q$$

i.e.

$$\pi_{(i+1)} = (I + h_{i+1} Q) \pi_i \tag{9}$$

The state vector of probabilities at time  $t_i$  is denoted by  $\pi_i$ . To avoid confusing the  $i^{th}$  component of the vector with the full vector at time  $t_i$ , we use this notation instead of  $\pi_i$ . In the explicit Euler technique for Markov chains, a

scalar-matrix product and a vector-matrix product are used to go from one-time step to the next.

**Results**

This section presented the solutions for performance measures such as the transient solution for Markov chain,  $\pi_{(i+1)}$ ; using Euler, Trapezoid methods and examine the effect of decreasing the step size on the two formulae with illustrative examples.

**Illustrative Example 1:** Considering a continuous-time Markov chain with infinitesimal generator

$$Q = \begin{pmatrix} -3 & 2 & 1 \\ 2 & -7 & 5 \\ 1 & 1 & -2 \end{pmatrix}$$

Assume we want to calculate the transient distribution at time  $t = 1$  and the Markov chain starts in state 1. Calculate  $\pi_i, i = 1, 2, \dots, 10$  with a constant step size of 0.1. Equation is derived from the finite element method formula Equation (9)

$$\begin{aligned} \pi(0.1) &\approx \pi_{(1)} = \pi_{(0)}(I + 0.1Q) \\ &= (1 \ 0 \ 0) \left[ \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} + 0.1 \begin{pmatrix} -3 & 2 & 1 \\ 2 & -7 & 5 \\ 1 & 1 & -2 \end{pmatrix} \right] \\ &= (1 \ 0 \ 0) \begin{pmatrix} 0.7 & 0.2 & 0.1 \\ 0.2 & 0.3 & 0.5 \\ 0.1 & 0.1 & 0.8 \end{pmatrix} = (0.7 \ 0.2 \ 0.1) \\ \pi(0.2) &\approx \pi_{(2)} = \pi_{(1)}(I + 0.1Q) \\ &= (0.7 \ 0.2 \ 0.1) \begin{pmatrix} 0.7 & 0.2 & 0.1 \\ 0.2 & 0.3 & 0.5 \\ 0.1 & 0.1 & 0.8 \end{pmatrix} \\ &= (0.54 \ 0.21 \ 0.25) \\ \pi(0.3) &\approx \pi_{(3)} = \pi_{(2)}(I + 0.1Q) \\ &= (0.54 \ 0.21 \ 0.25) \begin{pmatrix} 0.7 & 0.2 & 0.1 \\ 0.2 & 0.3 & 0.5 \\ 0.1 & 0.1 & 0.8 \end{pmatrix} \\ &= (0.445 \ 0.196 \ 0.359) \\ \pi(0.4) &\approx \pi_{(4)} = \pi_{(3)}(I + 0.1Q) \\ &= (0.445 \ 0.196 \ 0.359) \begin{pmatrix} 0.7 & 0.2 & 0.1 \\ 0.2 & 0.3 & 0.5 \\ 0.1 & 0.1 & 0.8 \end{pmatrix} \\ &= (0.3866 \ 0.1837 \ 0.4297) \\ \pi(0.5) &\approx \pi_{(5)} = \pi_{(4)}(I + 0.1Q) \\ &= (0.3866 \ 0.1837 \ 0.4297) \begin{pmatrix} 0.7 & 0.2 & 0.1 \\ 0.2 & 0.3 & 0.5 \\ 0.1 & 0.1 & 0.8 \end{pmatrix} \\ &= (0.35033 \ 0.1754 \ 0.47427) \end{aligned}$$

$$\begin{aligned} \pi(0.6) &\approx \pi_{(6)} = \pi_{(5)}(I + 0.1Q) \\ &= (0.35033 \ 0.1754 \ 0.47427) \begin{pmatrix} 0.7 & 0.2 & 0.1 \\ 0.2 & 0.3 & 0.5 \\ 0.1 & 0.1 & 0.8 \end{pmatrix} \\ &= (0.327738 \ 0.170113 \ 0.502149) \\ \pi(0.7) &\approx \pi_{(7)} = \pi_{(6)}(I + 0.1Q) \\ &= (0.327738 \ 0.170113 \ 0.502149) \begin{pmatrix} 0.7 & 0.2 & 0.1 \\ 0.2 & 0.3 & 0.5 \\ 0.1 & 0.1 & 0.8 \end{pmatrix} \\ &= (0.3136541 \ 0.1667964 \ 0.5195495) \\ \pi(0.8) &\approx \pi_{(8)} = \pi_{(7)}(I + 0.1Q) \\ &= (0.3136541 \ 0.1667964 \ 0.5195495) \begin{pmatrix} 0.7 & 0.2 & 0.1 \\ 0.2 & 0.3 & 0.5 \\ 0.1 & 0.1 & 0.8 \end{pmatrix} \\ &= (0.3048721 \ 0.16472469 \ 0.53040321) \\ \pi(0.9) &\approx \pi_{(9)} = \pi_{(8)}(I + 0.1Q) \\ &= (0.3048721 \ 0.16472469 \ 0.53040321) \begin{pmatrix} 0.7 & 0.2 & 0.1 \\ 0.2 & 0.3 & 0.5 \\ 0.1 & 0.1 & 0.8 \end{pmatrix} \\ &= (0.299395729 \ 0.163432148 \ 0.537172123) \\ \pi(1.0) &\approx \pi_{(10)} = \pi_{(9)}(I + 0.1Q) \\ &= (0.299395729 \ 0.163432148 \ 0.537172123) \begin{pmatrix} 0.7 & 0.2 & 0.1 \\ 0.2 & 0.3 & 0.5 \\ 0.1 & 0.1 & 0.8 \end{pmatrix} \\ &= (0.295980652 \ 0.162626002 \ 0.541393346) \end{aligned}$$

The Trapezoid rule, Equation (7), when applied to Markov chains, becomes

$$\begin{aligned} \pi_{(i+1)} &= \pi_i + \frac{h_{i+1}}{2} (\pi_i Q + \pi_{(i+1)} Q) \\ \pi_{(i+1)} \left( I - \frac{h_{i+1}}{2} Q \right) &= \pi_i \left( I + \frac{h_{i+1}}{2} Q \right) \end{aligned} \tag{10}$$

It necessitates the solution of a system of equations at each step in addition to the operations required by the explicit Euler method (plus a scalar-matrix product). The greater accuracy attained with the trapezoid rule offsets these additional computations every step to some extent. The matrix is small if the Markov chain is small and the step size is kept constant.

$$\left( I + \frac{h_{i+1}}{2} Q \right) \left( I - \frac{h_{i+1}}{2} Q \right)^{-1}$$

may be computed before beginning the stepping process, resulting in the modified Euler method's computation per step being similar to that of the explicit Euler method. The inverse should not be created explicitly if the Markov chain is long. Instead, compute a lower-upper matrix decomposition and replace the inverse with a backward and forward substitution process using upper and lower triangular matrices, respectively. When the step size is not kept constant, solving a

system of linear equations for each variable value of  $h$  is required. The work required by the trapezoid rule to compute the solution to a certain precision may or may not be less than that required by the explicit Euler depending on the size and sparsity pattern of  $Q$ . An implicit technique that requires more calculation per step but fewer steps vs an explicit method that requires more steps but less effort per step is the trade-off.

**Illustrative Example 2**

To solve Example 1, using the trapezoid rule with a step size of 0.1 and compute  $\pi_{(i)}$ ,  $i = 1, 2, \dots, 10$  one by one. Depending on the size and sparsity pattern of  $Q$ , the work required by the trapezoid rule to compute the solution to a specified precision may or may not be less than that required by the explicit Euler. The trade-off is between an implicit method requiring more computation per step but fewer steps and an explicit method requiring more steps but less work per step.

From the Trapezoid formula of Equation (10), we obtain

$$\pi_{(i+1)} = \pi_i \left( I + \frac{h_{i+1}}{2} Q \right) \left( I - \frac{h_{i+1}}{2} Q \right)^{-1}$$

Such that  $\pi_{(0)} = (1 \ 0 \ 0)$ ,  $\frac{h_{i+1}}{2} = \frac{0.1}{2} = 0.05$ ,  $Q = \begin{pmatrix} -3 & 2 & 1 \\ 2 & -7 & 5 \\ 1 & 1 & -2 \end{pmatrix}$ .

$$\frac{h_{i+1}}{2} Q = \begin{pmatrix} -0.15 & 0.1 & 0.05 \\ 0.1 & -0.35 & 0.25 \\ 0.05 & 0.05 & -0.1 \end{pmatrix}$$

$$I + \frac{h_{i+1}}{2} Q = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} + \begin{pmatrix} -0.15 & 0.1 & 0.05 \\ 0.1 & -0.35 & 0.25 \\ 0.05 & 0.05 & -0.1 \end{pmatrix} = \begin{pmatrix} 0.85 & 0.1 & 0.05 \\ 0.1 & 0.65 & 0.25 \\ 0.05 & 0.05 & 0.9 \end{pmatrix}$$

$$I - \frac{h_{i+1}}{2} Q = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} - \begin{pmatrix} -0.15 & 0.1 & 0.05 \\ 0.1 & -0.35 & 0.25 \\ 0.05 & 0.05 & -0.1 \end{pmatrix} = \begin{pmatrix} 1.15 & -0.1 & -0.05 \\ -0.1 & 1.35 & -0.25 \\ -0.05 & -0.05 & 1.1 \end{pmatrix}$$

Therefore,

$$\pi_{(i+1)} = \pi_i \left( I + \frac{h_{i+1}}{2} Q \right) \left( I - \frac{h_{i+1}}{2} Q \right)^{-1} = \pi_i \begin{pmatrix} 0.85 & 0.1 & 0.05 \\ 0.1 & 0.65 & 0.25 \\ 0.05 & 0.05 & 0.9 \end{pmatrix} \begin{pmatrix} 1.15 & -0.1 & -0.05 \\ -0.1 & 1.35 & -0.25 \\ -0.05 & -0.05 & 1.1 \end{pmatrix}^{-1}$$

For  $\begin{pmatrix} 1.15 & -0.1 & -0.05 \\ -0.1 & 1.35 & -0.25 \\ -0.05 & -0.05 & 1.1 \end{pmatrix}^{-1}$ , Let  $A = \begin{pmatrix} 1.15 & -0.1 & -0.05 \\ -0.1 & 1.35 & -0.25 \\ -0.05 & -0.05 & 1.1 \end{pmatrix}$ .

$$|A| = 1.15 \begin{vmatrix} 1.35 & -0.25 \\ -0.05 & 1.1 \end{vmatrix} + 0.1 \begin{vmatrix} -0.1 & -0.25 \\ -0.05 & 1.1 \end{vmatrix} - 0.05 \begin{vmatrix} -0.1 & 1.35 \\ -0.05 & -0.05 \end{vmatrix}$$

$$|A| = 1.15(1.485 - 0.0125) + 0.1(-0.11 - 0.0125) - 0.05(0.005 + 0.0675),$$

$$|A| = 1.15(1.4725) + 0.1(-0.1225) - 0.05(0.0725) = 1.6775.$$

$$\text{Cof } A = \begin{pmatrix} 1.4725 & 0.1225 & 0.0725 \\ 0.1125 & 1.2625 & 0.0625 \\ 0.0925 & -0.2825 & -1.5515 \end{pmatrix}$$

$$\text{Adj } A = \begin{pmatrix} 1.4725 & 0.1125 & 0.0925 \\ 0.1225 & 1.2625 & -0.2825 \\ -0.0725 & 0.0625 & -1.5515 \end{pmatrix}$$

$$A^{-1} = \frac{1}{|A|} \text{Adj } A = \frac{1}{1.6775} \begin{pmatrix} 1.4725 & 0.1125 & 0.0925 \\ 0.1225 & 1.2625 & -0.2825 \\ -0.0725 & 0.0625 & -1.5515 \end{pmatrix} = \begin{pmatrix} 0.8778 & 0.0671 & 0.0551 \\ 0.0730 & 0.7526 & -0.1684 \\ -0.0432 & 0.0373 & -0.9249 \end{pmatrix}$$

Therefore,

$$\pi_{(i+1)} = \pi_i \begin{pmatrix} 0.85 & 0.1 & 0.05 \\ 0.1 & 0.65 & 0.25 \\ 0.05 & 0.05 & 0.9 \end{pmatrix} \begin{pmatrix} 0.8778 & 0.0671 & 0.0551 \\ 0.0730 & 0.7526 & -0.1684 \\ -0.0432 & 0.0373 & -0.9249 \end{pmatrix}$$

$$\pi_{(i+1)} = \pi_i \begin{pmatrix} 0.7513 & 0.1342 & 0.1145 \\ 0.1244 & 0.5052 & 0.3704 \\ 0.0087 & 0.1532 & 0.8381 \end{pmatrix}$$

Beginning with  $\pi_{(0)} = (1 \ 0 \ 0)$ , this allows us to compute

$$\pi(0.1) \approx \pi_{(1)} = (1 \ 0 \ 0) \begin{pmatrix} 0.7513 & 0.1342 & 0.1145 \\ 0.1244 & 0.5052 & 0.3704 \\ 0.0087 & 0.1532 & 0.8381 \end{pmatrix} = (0.7513 \ 0.1342 \ 0.1145)$$

$$\pi(0.2) \approx \pi_{(2)} = (0.7513 \ 0.1342 \ 0.1145) \begin{pmatrix} 0.7513 & 0.1342 & 0.1145 \\ 0.1244 & 0.5052 & 0.3704 \\ 0.0087 & 0.1532 & 0.8381 \end{pmatrix} = (0.58214232 \ 0.1861637 \ 0.23169398)$$

$$\pi(1.0) \approx \pi_{(10)} = (0.385980652 \ 0.261626002 \ 0.35239335)$$

**Illustrative Example 3**

Considering a continuous-time Markov chain with infinitesimal generator

$$Q = \begin{pmatrix} -2 & 1 & 1 \\ 3 & -8 & 5 \\ 1 & 2 & -3 \end{pmatrix}$$

Assume we want to calculate the transient distribution at time  $t = 1$  and the Markov chain starts in state 1. Calculate  $\pi_i, i = 1, 2, \dots, 10$  with a constant step size of 0.1. Examine the effect of lowering the step size using the Euler and Trapezoid formulas.

For the Euler technique, we obtain

$$\begin{aligned} \pi(0.1) &\approx \pi_{(1)} = \pi_{(0)}(I + 0.1Q) \\ &= (1 \ 0 \ 0) \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ &\quad + 0.1 \begin{bmatrix} -2 & 1 & 1 \\ 3 & -8 & 5 \\ 1 & 2 & -3 \end{bmatrix} \\ &= (1 \ 0 \ 0) \begin{pmatrix} 0.8 & 0.1 & 0.1 \\ 0.3 & 0.2 & 0.5 \\ 0.1 & 0.2 & 0.7 \end{pmatrix} = (0.8 \ 0.1 \ 0.1) \end{aligned}$$

$$\begin{aligned} \pi(0.2) &\approx \pi_{(2)} = \pi_{(1)}(I + 0.1Q) \\ &= (0.8 \ 0.1 \ 0.1) \begin{pmatrix} 0.8 & 0.1 & 0.1 \\ 0.3 & 0.2 & 0.5 \\ 0.1 & 0.2 & 0.7 \end{pmatrix} \\ &= (0.68 \ 0.12 \ 0.20) \end{aligned}$$

Continuing in this fashion we finally compute

$$\begin{aligned} \pi(1.0) &\approx \pi_{(10)} = \pi_{(9)}(I + 0.1Q) \\ &= (0.45222 \ 0.154053 \ 0.393726) \begin{pmatrix} 0.8 & 0.1 & 0.1 \\ 0.3 & 0.2 & 0.5 \\ 0.1 & 0.2 & 0.7 \end{pmatrix} \\ &= (0.447365 \ 0.154778 \ 0.397857) \end{aligned}$$

The correct answer, obtained by a finite different method, is

$$\pi(1.0) = (0.457446 \ 0.152269 \ 0.389285),$$

and a measure of the error is given as

$$\|\pi(1.0) - \pi_{(10)}\|_2 = 0.013319.$$

For Trapezoid rule, when applied to Markov chains, becomes

$$\pi_{(i+1)} = \pi_i \left( I + \frac{h_{i+1}}{2} Q \right) \left( I - \frac{h_{i+1}}{2} Q \right)^{-1}$$

Such that  $\pi_{(0)} = (1 \ 0 \ 0), \frac{h_{i+1}}{2} = \frac{0.1}{2} = 0.05, Q =$

$$\begin{pmatrix} -2 & 1 & 1 \\ 3 & -8 & 5 \\ 1 & 2 & -3 \end{pmatrix}$$

$$\frac{h_{i+1}}{2} Q = \begin{pmatrix} -0.1 & 0.05 & 0.05 \\ 0.15 & -0.4 & 0.25 \\ 0.05 & 0.1 & -0.15 \end{pmatrix},$$

$$\begin{aligned} I + \frac{h_{i+1}}{2} Q &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} + \begin{pmatrix} -0.1 & 0.05 & 0.05 \\ 0.15 & -0.4 & 0.25 \\ 0.05 & 0.1 & -0.15 \end{pmatrix} \\ &= \begin{pmatrix} 0.9 & 0.05 & 0.05 \\ 0.15 & 0.60 & 0.25 \\ 0.05 & 0.10 & 0.85 \end{pmatrix}, \end{aligned}$$

$$\begin{aligned} I - \frac{h_{i+1}}{2} Q &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} - \begin{pmatrix} -0.1 & 0.05 & 0.05 \\ 0.15 & -0.4 & 0.25 \\ 0.05 & 0.1 & -0.15 \end{pmatrix} \\ &= \begin{pmatrix} 1.10 & -0.05 & -0.05 \\ -0.05 & 1.40 & -0.25 \\ -0.05 & -0.1 & 1.15 \end{pmatrix} \end{aligned}$$

Therefore,

$$\begin{aligned} \pi_{(i+1)} &= \pi_i \left( I + \frac{h_{i+1}}{2} Q \right) \left( I - \frac{h_{i+1}}{2} Q \right)^{-1} \\ &= \pi_i \begin{pmatrix} 0.9 & 0.05 & 0.05 \\ 0.15 & 0.60 & 0.25 \\ 0.05 & 0.10 & 0.85 \end{pmatrix} \begin{pmatrix} 1.10 & -0.05 & -0.05 \\ -0.05 & 1.40 & -0.25 \\ -0.05 & -0.1 & 1.15 \end{pmatrix}^{-1} \\ \pi_{(i+1)} &= \pi_i \begin{pmatrix} 0.832370 & 0.072254 & 0.95376 \\ 0.213873 & 0.459538 & 0.326590 \\ 0.096266 & 0.130058 & 0.771676 \end{pmatrix}. \end{aligned}$$

Beginning with  $\pi_{(0)} = (1 \ 0 \ 0)$ , this allows us to compute

$$\pi(0.1) \approx \pi_{(1)} = (0.832370 \ 0.072254 \ 0.95376),$$

$$\pi(0.2) \approx \pi_{(2)} = (0.717665 \ 0.105750 \ 0.176585),$$

$$\pi(0.3) \approx \pi_{(3)} = (0.637332 \ 0.123417 \ 0.239251),$$

$$\pi(1.0) \approx \pi_{(10)} = (0.456848 \ 0.153361 \ 0.389791),$$

**Table 1:** Euler and Trapezoidal Method Results

	Explicit Euler	Trapezoid
Step size	$\ z(t_1) - z_1\ _2$	$\ z(t_1) - z_1\ _2$
h = 0.1	0.133193e - 01	0.788907e - 03
h = 0.01	0.142577e - 02	0.787774e - 05
h = 0.001	0.143269e - 03	0.787763e - 07
h = 0.0001	0.143336e - 04	0.787519e - 09
h = 0.00001	0.143343e - 05	0.719906e - 11

## Discussion

Considering the results obtained for illustrative examples 3 using finite Euler approach and Trapezoid rule, it was observed that the error for Euler is given as  $\|\pi(1.0) - \pi_{(10)}\|_2 = 0.013319$ , while that of Trapezoid is measured as  $\|\pi(1.0) - \pi_{(10)}\|_2 = 0.0007889$ , and this implied that the result obtained by Trapezoid rule is significantly better than the result obtained using the Euler. The effect of reducing the step size using the same infinitesimal generator with  $\pi_{(0)} = (1 \ 0 \ 0)$  while the length of the integration interval is set to be  $\tau = 1$  is investigated. By considering Equations (9) and (10) using Matlab, which the results are shown in Table 1, with the first column of the table gives the step size used throughout the range, the second column is the 2-norm of the absolute error using the explicit Euler method, and the third column gives the 2-norm of the absolute error when the trapezoid method is used. It was observed that the accuracy achieved with the explicit Euler method has a direct relationship with the step size, and this indicates that the trapezoid method produces far more accurate results than Euler, and both instances need an effectively equivalent amount of processing due to the usage of equal step sizes.

## Conclusion

Solutions for large state spaces of transient distribution in Markov chain is computed using Euler and Trapezoid methods, which begin with an initial estimate of the solution vector and then alter it in such a way that it gets closer and closer to the true solution with each step or iteration, saving time. Matrices operations, such as the product and matrix inversion are performed, and Markov chain laws, theorems, formulas with Matlab software are utilized. For illustrative examples, the transient distribution vector's  $\pi_{(i+1)}$ ,  $i = 0, 1, 2, \dots$ ; is computed for both Euler and Trapezoid methods with their corresponding error when compared to different method. the effect of decreasing the step

size using the same infinitesimal generator is examined, taking  $\pi_{(0)} = (1 \ 0 \ 0)$  and the length of the interval of integration to be  $\tau = 1$ , and we observe that the accuracy achieved with the explicit Euler method has a direct relationship with the step size. It is concluded through the error results from both Euler and Trapezoid rules that the much more accurate results is obtained with the trapezoid method.

## References

- Abushet, A.W. (2021). Small modification on modified Euler method for solving initial value problem, *Abstract and Applied Analysis*, volume 2021, Article ID 9951815, 5 pages, Hindawi.
- 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)*, 8(6): 87 – 96, [www.ijasm.org](http://www.ijasm.org).
- Agboola, S. O. (2022). Analysis and Applications of Random Walk and Gambler’s Ruin on Irreducible Periodic Markov Chain, *Jewel Journal of Scientific Research, Faculty of Science, Federal University of Kashere (JJSR)*. 7(1): 1 – 8. [www.fukashere.edu.ng/jjsr](http://www.fukashere.edu.ng/jjsr).
- 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. [www.ajol.info > index.php > dujopas > article > view](http://www.ajol.info/index.php/dujopas). <https://dx.doi.org/10.4314/dujopas.v7i4b.4>, <https://fud.edu.ng/dujopas>.



- Agboola, S. O. and Ayinde, S. A. (2022). On the Application of Successive Overrelaxation Algorithmic and Block Numerical Iterative Solutions for the Stationary Distribution in Markov Chain, *Nigerian Journal of Pure and Applied Sciences, Faculty of Physical Sciences and Faculty of Life Sciences, University of Ilorin, Nigeria (NJPAS)*, 35(1): 4263 – 4272. <http://njpas.com.ng>. DOI: <https://doi.org/10.48198/NJPAS/22.A02>
- Agboola, S. O. and Badmus, N. I. (2021). The Application of Runge-Kutta and Backward Differentiation Methods for Solving Transient Distribution in Markov Chain, *Nigerian Journal of Mathematics and Applications*. Department of Mathematics, University of Ilorin, Nigeria, 31: 191 – 201. <https://www.njmaman.com>.
- Agboola, S. O. and Nehad, A. S. (2022). On the Application of Matrix Scaling and Powering Methods of Small State Spaces For Solving Transient Distribution In Markov Chain, *FUDMA Journal of Sciences (FJS), Faculties of Earth Science and Physical Science Journal, Federal University DutsinMa*, 6(1): 135 – 140. <https://fjs.fudutsinma.edu.ng>. DOI: <https://doi.org/10.33003/fjs-2022-0601-849>.
- Agboola. S.O., Ibikunle, O. and Balami, H.M. (2022). On the Application of Uniformization Method for Large Spaces in Solving the Transient Distribution in Markov Chain. *Faculty of Science Journal, Federal University Wukari Trends in Science & Technology Journal, Taraba State, Nigeria*, 7(2): 885- 889. [www.ftstjournal.com](http://www.ftstjournal.com).
- Agboola, S.O., Ayinde, S.A., Ibikunle, O. and Obaromi, D.A. (2023). Jacobi and Gauss–Seidel numerical Iterative Solution Methods for the Stationary Distribution of Markov Chain, *Dutse Journal of Pure and Applied Sciences (DUJOPAS), Faculty of Science Journal, Federal University Dutse, Jigawa State*, 9(1a):127 – 138. <https://dx.doi.org/10.4314/dujopas.v9i1a.13>. [www.ajol.info](http://www.ajol.info) > [index.php](#) > [dujopas](#) > [article](#) > [view](https://fud.edu.ng/dujopas). <https://fud.edu.ng/dujopas>
- 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.
- Clemence. 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. P 18 – 31.
- Kanwar, V, and Tomar, S. K. (2008). Exponential fitted variants of Euler’s method for Ordinary Differential Equations, *International Journal of Mathematical Education in Science and Technology*, 39(8): 1112 – 1116.
- Matthew, H and Paul, O. (2017). Trapezoidal rule and its error analysis for the Grunward-Letnikov operator,” *International Journal of Dynamics and Control*, 5:18 – 29.
- Norm, E. and John, L. (1989). The Trapezoidal rule for analytic functions and rapid increase, *Journal of Computational and Applied Mathematics*, 27(3): 389 – 406.
- Nurujjaman, M. (2020). *Enhance Euler’s method to solve first order ordinary differential*

equations with better accuracy, Md Nurujjaman's Lab. Doi:10.5281/zenodo.3731020.

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, Deutsche Physikalische*, 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. P 11 - 24.

Roberto, G. (2015). Trapezoidal methods for fractional differential equations: Theoretical and computational aspects, *Mathematics and Computers in Simulation, Elsevier*, 110: 96 – 112.

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

Stewart, W. J. (1994). *Introduction to the Numerical Solution of Markov Chains*. Princeton University Press, Princeton, N.J. P 310 – 328.

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

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

Xing, S. Xiaoliang, C and Kewei, L (2020). Deep Euler method: Solving ODEs by approximating local truncation error in the Euler method, *Numerical Analysis*, Cornell University, arXiv:2003.09573 [math.NA].

Yue, W. (2021). A randomized trapezoidal quadrature, *International Journal of Computer Mathematics*, 99(4): 12 – 25.