BENIN JOURNAL OF PHYSICAL SCIENCES
BJPS Vol. 1(2), December, pg. 22-41 (2024)
ISSN 3043-6931(Print)
3043-694X(Online)
www.bjps.org.ng

Numerical Computation of Cauchy's Integral Theorem for Matrix Function

S. E. UWAMUSI

[1]*Department of Mathematics, Faculty of Physical Sciences, University of Benin, Nigeria*

Received: 30/05/2024 ***Accepted:*** 06/09/2024

Abstract

Cauchy's integral theorem for the matrix function is studied. Using the iteration formula of the Runge–Kutta fourth-order method, we advance the solution to Cauchy's integral theorem, wherein Jacobi elliptic integrals and the Mobius transformation are applicable. It is proven that the method converges uniformly locally on the solid triangle. The hyperbolic matrix tangent in the arc length when the contour winds once counterclockwise around the integral is computed. Numerical results have been reported with great success.

Keywords: *Cauchy integral theorem, Jacobi elliptic integrals, Mobius transformation, Trapezoidal rule, Runge–Kutta method, Complex function*

1. INTRODUCTION

Following Takihira et al (2021), Uwamusi(2022), Hales et al (2008), Higham (2008), Jefferies (2012), we propose computing Cauchy's integral theorem for the matrix function $f(A)$ using the Runge–Kutta fourth-order method, wherein the complete Jacobi elliptic integrals and Mobius transformation are applicable as a useful tool. The well-known method of the Trapezoidal rule has a disadvantage in computing this task most accurately due to its slow rate in convergence speed and interval of absolute stability of [-2,0]. On the other hand, Runge-Kutta method has interval of absolute stability of [-2.78,0]. In terms of speed and computational complexity, the Runge–Kutta method is preferable to the Trapezoidal rule due to its region of interval of absolute stability and efficiency. The complexity of a numerical method is the computational cost and resource requirements of a method when the problem size increases. The time complexity and space complexity of the

*Corresponding author e-mail: stephen.uwamusi@uniben.edu

Trapezoidal rule and Runge-Kutta methods are hereby brought into focus. The Trapezoidal rule is a simple method that involves approximating the integral by dividing the integration interval into equal subintervals and evaluating the function at the end points and midpoints of each interval. Using the big O notation, the time complexity of the Trapezoidal rule is generally considered to be $O(n)$ and that n is the number of subintervals or steps used in the approximation. On the other hand, the space complexities of the Trapezoidal rule do not require significant additional memory space beyond storing the function values and intermediate results. Therefore, the space complexity in Trapezoidal rule is generally considered to be $O(1)$.

On the other hand, the spatial complexity of the Runge–Kutta method (R-K-4), requires additional memory to store the intermediate results at each step. The space complexity is generally considered to be O(4) since the amount of memory required is independent of the problem.

The Runge–Kutta fourth-order method is known for its higher order accuracy, which means that it requires more functional evaluations and therefore has a higher computational cost than does the Trapezoidal rule. Therefore, in the fore going, the improved accuracy can often compensate for the additional cost, especially when higher precision is needed in the computation.

The paper is motivated as follows:

Cauchy's integral theorem in the senses of Hales et al (2008),Uwamusi (2022),Higham (2008) for the matrix function $f(A)$ is defined in the form

$$f(A)=\frac{1}{2\pi i}\oint_{\eta} f(z)\left(zI-A\right)^{-1}dz \tag{1}$$

The η appearing in equation (1) is a contour winding around once the Jordan arc in the counterclockwise direction which carries the eigenvalues of the matrix A. The matrix A is real, and its eigenvalues are located at R^{+}. It is assumed that $A\in R^{n\times n}$. Henceforth, it is assumed that f is analytic in the slit complex plane $\mathbb{C}\setminus\left(-\infty,0\right)$, which can take on any of these functions $z^{\frac{1}{2}}, z^{n}, e^{z}$, and $\log z$ whenever they are needed. We denote $F_k(A)$ to signify the approximate results computed by the iterative methods to the known solution $F(A)$. Then, the relative error is denoted as

$$\text{Relative error}=\frac{\left\|F(A)-F_k(A)\right\|}{\left\|F(A)\right\|}.$$

Denoting the largest and smallest eigenvalues of the matrix A by the symbols S and s, respectively, it is given that the outer radius for the annulus is $R=\frac{1}{2}(S+s)$

, while the inner radius of the annulus is $r = \frac{1}{2}(S - s)$ for the path traced out by the contour integral. It follows also $K(A) = \frac{S}{s}$ denotes condition number of the matrix A. The aim of doing so is to see that $\sigma(A)$ fills $[s, S]$. With this defined, the domain of convergence in Cauchy's integral theorem [6] lies in the region

$$D = \mathbb{C} \setminus ((-\infty, 0) \cup [s, S]) \text{ for } D \subset \mathbb{C}. \tag{2}$$

As a result of definition of $f(A)$ in equation (1), we write in the sense of Uwamusi(2022) that

$$f(A) = \frac{1}{2\pi i} \int_0^{2\pi} f(z_0 + re^{i\theta}) \left((re^{i\theta} I) - A \right)^{-1} ire^{i\theta} d\theta . \tag{3}$$

Uwamusi(2022) therefore had proved that computation of Cauchy's integral theorem for a matrix function using the Trapezoidal rule and its variants was Lebesgue measurable over dissection on the argument θ that runs through the N-partitions. He further confirmed that Cauchy's integral theorem is well defined.

By setting

$$\int_0^{2\pi} \frac{f(z)}{(zI - A)} dz = \int_0^{2\pi} f\left(z_0 + re^{i\theta}\right)(zI - A)^{-1} ire^{i\theta} d\theta = \int_0^{2\pi} f(z_0 + re^{i\theta}) \frac{1}{re^{i\theta} I} ire^{i\theta} d\theta , \tag{4}$$

and adopting the technique that any contour not containing a pole is zero, it follows that:

$$\int_0^{2\pi} f(z_0 + re^{i\theta}) I i d\theta = \lim_{r \to 0} f(z_0 + re^{i\theta}) i \int_0^{2\pi} I d\theta = i\, f(z_1) I\theta \Big|_0^{2\pi} = 2\pi i f(z_1) I = 2\pi i f(A). \tag{5}$$

The winding number (index of the curve approximately 0 is then defined as

$$Ind_0(f) = \frac{1}{2\pi i} \oint_\eta \frac{f(z)}{z} dz . \tag{6}$$

By recalling that the winding number about point $ZI = A$ is the number of times a curve η winds around arbitrary point matrix A carrying along its eigenvalues, then it can be expressed in the form that:

$$Ind_A(f) = \frac{1}{2\pi i} \oint_\eta \frac{f(z)}{zI - A} dz .$$

By further setting that:

$$u = sn(t) = sn\left(t \middle| k^2\right), \tag{7a}$$

$$k = \frac{\sqrt{\frac{S}{s}} - 1}{\sqrt{\frac{S}{s}} + 1} \quad , \tag{7b}$$

and

$$t = \frac{2Ki}{\pi} \log\left(\frac{-is}{r}\right) \ (s \in A), i = \sqrt{-1}, \tag{8}$$

one maps the upper half plane in the arc length in the sense of Hales et al (2008) to the rectangle with end points:

$\left[-k^{-1},-1\right]$, $\left[1,k^{-1}\right]$, according to Mobius transformation. We intend to explore more of Mobius transformation shortly in the next section.

Equation (8) carries the upper half plane in the arc length to the rectangle with vertices $\pm K$ and $\pm K + iK'$, where K and K' are the complete elliptic integrals. The Jacobi elliptic integrals have several uses in the calculations of the polar decomposition of a matrix, computing arc length in mathematical physics, e.g., as in astrodynamics, the earth's meridian surface, and a body undergoing cubic potential wells, Uwamusi (2024). For these reasons, we provide here the basis of Jacobi elliptic integrals for easy accessibility to the readers.

Lawden(1989) had showed that the values K and K' are expressed as:

$$K(k) = \int_0^1 \left\{\left(1-t^2\right)\left(1-k^2t^2\right)\right\}^{\frac{1}{2}} dt \tag{9}$$

$$K'(k) = \int_0^1 \left\{\left(1-t^2\right)\left(1-k'^2t^2\right)\right\}^{\frac{-1}{2}} dt \ . \tag{10}$$

In views of Benrard et al (2009), Lawden (1989), and Dobrushkin (2022) , one obtains the series representation for Jacobi complete elliptic integrals:

$$K(k) = \int_0^1 \frac{dt}{\sqrt{1-t^2}.\sqrt{1-k^2t^2}} = \frac{\pi}{2}\left\{1 + \sum_{j\geq 1}\left(\frac{(2j-1)!}{(2j)!}\right)^2 k^j\right\} \tag{11}$$

$$K'(k) = \int_0^1 \sqrt{\frac{1-k^2t^2}{1-t^2}}dt = \frac{\pi}{2}\left\{1 + \sum_{j\geq 1}\left(\frac{(2j-1)!}{(2j)!}\right)^2\right\}\frac{k^j}{2j-1} = {}_2F_1\left(-\frac{1}{2};\frac{1}{2};1;k\right)\frac{\pi}{2} \tag{12}$$

For instance, the term ${}_2F_1(a_1,a_2;b_1;x) = \sum_{n=0}^{\infty}\frac{(a_1)_n(a_2)_n}{(b_1)_n\, n!}x^n$ appearing in equation (12) is the hypergeometric function as expressed in Lawden (1989) , Uwamusi (2021), Jordan and Cortis (1998),Jordan and Sastry (1989), and it is a solution to the Jacobi elliptic integral of the first kind:

$$\int_0^{\frac{\pi}{2}} \frac{d\theta}{\sqrt{1-k^2\sin^2\theta}}, \tag{13}$$

where the Pochhammer symbol $(a)_n = a(a+1)(a+2)...(a+n-1)$ for $n \geq 1$ is well utilized in Bernard et al (2009), Lawden (1989), Nagar et al (2012), Nakatsukasa and Higham (2011), Uwamusi 2021), Uwamusi (2022), Batista (2019), Viennor (1980), Yang and Tian (2017), Dobrushkin (2022), Hales et al (2008). The asymptotic sterling series accompanying the hypergeometric function is then equal to

$$\log\Gamma s = \left(s-\frac{1}{2}\right)\log(s) - s + \log\left(\frac{2\pi}{2}\right) + \sum_{n=1}^{N-1}\frac{B_{2n}}{2n(2n-1)s^{2n-1}} + R(N,s),$$

and that $\Gamma(s+v) = (s)_v\Gamma(s), (s)_v = s(s+1)\times...\times(s+v-1)$. The $\Gamma(s)\Gamma(1-s) = \frac{\pi}{\sin(\pi s)}$. The $R(N,s) \approx$ error bound. B_{2n} are the Bernoulli numbers. Thus, B_{2n} are given by the equation

$B_{2n} = (-1)^{n+1}2(2n)!\xi(2n)(2\pi)^{-2n}$, and $\xi(2n) = \sum_{m=1}^{\infty} m^{-2n}$, $m^{-2n} = m^2.m^{-(2n+2)}$.

Introducing modulus of the Grotzsch extremal ring in line with ideas due to Yang and Tian (2017) for the quasiconformal distortion in the Jacobi elliptic integral it can be deduced that:

$$\mu(k) = \frac{K(k')}{K(k)}, \text{ (where } K'(k) = K(k')\text{)}. \tag{14}$$

Thus the asymptotic formula for the Jacobi elliptic function Yang and Tian (2017) can expressed by the equation

$$K(k) = \frac{\pi}{2}F\left(\frac{1}{2},\frac{1}{2};1;k^2\right) \sim \ln\frac{4}{k'};\ (k' \to 1^-, k' = \sqrt{1-k^2},\ (0 \leq k' < 1)). \tag{15}$$

As a result, it follows from the above information that the inequality

$$\frac{K(k)}{\ln\left(\frac{e^{4/3}}{k'}\right)} + \frac{K(k')}{\ln\left(\frac{e^{4/3}}{k}\right)} \leq 2v_0$$

holds good.

Where the value of v_0 is

$v_0 = \frac{3\Gamma\left(\frac{1}{4}\right)^2}{2(3\ln 2+8)\sqrt{\pi}} \approx 1.107$. It is the best constant one can obtain for $k \in (0,1)$. In addition to the fore goings, the inequality $K(k)K(k') \leq v_0^2 \ln\left(\frac{e^{4/3}}{k}\right)\ln\left(\frac{e^{4/3}}{k'}\right)$ holds.

Previous works demonstrated in Uwamusi (2017a), Uwamusi (2017b) have been conducted to compute most accurately the function F(A), but none has fitted into what is being studied here. For this reason, a new concept of innovation is proposed in this paper.

1.1 A critical appraisal of the use of the Mobius transformation in the Cauchy integral theorem

The following theorem is applied in our work.

Theorem 1, Springborn (2018). The Mobius transformation forms a subgroup in the group of homeomorphisms $\mathbb{C} \to \mathbb{C}$, called the Mobius group. The map defined by $\mathrm{SL}(2,\mathbb{C}) \to M, \begin{pmatrix} a & b \\ c & d \end{pmatrix} \to \left(z \to \frac{az+b}{cz+d} \right)$ is a homeomorphism of groups with kernel $\left\{ \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix} \right\}$.

Therefore, Mobius group is then isomorphic to $\mathrm{SL}(2,\mathbb{C})$ which holds for $z \to \frac{(az+b)}{cz+d}$ as the Mobius transformation. The terms a,b,c,d are complex numbers and $ad-bc \neq 0$. It must be noted that the term $ad-bc=1$ holds in general. Since $\frac{az+b}{cz+d} \in \mathrm{SL}(2,\mathbb{C})$, which has the capacity to preserving circles to circles and lines to lines in the complex plane and due to discussion earlier given in section 1 above, the Mobius transformation for the complex number z with a parameter k and matrix eigenvalues in line with ideas in Hales et al (2008) could be written as follows:

$$z = \sqrt{sS}\left(\frac{k^{-1}+u}{k^{-1}-u}\right) \quad . \tag{16}$$

By substituting u in equation (7) into equation (16) would yield the expression following :

$$z = \sqrt{sS}\left(\frac{k^{-1}+sn(t)}{k^{-1}-sn(t)}\right), \quad where \quad k = \frac{\sqrt{\frac{S}{s}}-1}{\sqrt{\frac{S}{s}}+1}, \ (u = sn(t)). \tag{17}$$

Further applying equation (17) in equation (1), one obtains a modified form version of [Hales et al (2008):

$$f(A) = \frac{-A}{2\pi i}\int_{-K+i\frac{K'}{2}}^{K+i\frac{K'}{2}} z^{-1} f(z(t))\big((z(t)*I)-A\big)^{-1}\frac{dz}{du}\frac{du}{dt}dt \tag{18}$$

where:

$$\frac{dz}{du}=\frac{2k^{-1}\sqrt{sS}}{\left(k^{-1}-u\right)^2},\quad \frac{du}{dt}=sn'(t)=\sqrt{1-k^2u^2}\sqrt{1-u^2}\ . \tag{19}$$

The term z^{-1} introduced in the equation is a stabilization factor aimed at counteracting the effects of $\frac{dz}{du}$ as Cauchy's integral swings around the contour in the counterclockwise direction.

The equidistant space interval is $\left[-K+i\frac{K'}{2},K+i\frac{K'}{2}\right]$, where

$$t_j=-K+i\frac{K'}{2}+\frac{2(j-\frac{1}{2})K}{N},\ 1\le j\le N\ . \tag{20}$$

N is equal to the number of subdivisions in the interval. Equation (20) induces equidistant steps in the trapezoidal rule for computing Cauchy's integral matrix function due to Hales et al (2008) expressed in equation (18) in terms of the Mobius transformation as given below:

$$f_N(A)=\frac{-4KA\sqrt{sS}}{\pi Nk}\,\mathrm{Im}\sum_{j=1}^{N}\frac{f(z(t_j))\left((z(t_j)I)-A\right)^{-1}cn(t_j)dn(t_j)}{z(t_j)\left(k^{-1}-u(t_j)\right)^2}\quad . \tag{21}$$

The terms *cn* and *dn* in equation (21) are Jacobi elliptic functions, as discussed in Lawden (1989) and Hales et al (2008).

The upper half plane in the arc length in equation (18) is :

$$f(A)=\frac{-A\sqrt{sS}}{\pi ik}\int_{-K+i\frac{K'}{2}}^{K+i\frac{K'}{2}}\frac{z^{-1}f(z(t))\left((z(t)*I)-A\right)^{-1}cn(t)dn(t)}{\left(k^{-1}-u\right)^2}dt\quad . \tag{22}$$

The term $cntdnt=\frac{d}{dt}(snt)=\sqrt{\left(1-sn^2t\right)\left(1-k^2sn^2t\right)}$ is a good substitute in equation (22).

The remainder of the paper is arranged as follows. Section 2 provides a description of the materials and methods. In section 3, we demonstrated with numerical examples based on what was discussed in section 1. Numerically it is computed for the function of a matrix as expressed in equation (18) described in Hales et al. (2008) by using the Trapezoidal rule and the Runge–Kutta fourth-order method incorporating Jacobi elliptic integrals and the Mobius transformation into calculations. The matrix hyperbolic tangent for the arc length is then computed. We discuss these results in section as well. In section 4, we draw conclusions based on the strengths of our findings.

2. MATERIALS AND METHOD

The materials and methods adopted in the paper for computing Cauchy's integral theorem for the matrix function are the set of complete Jacobi elliptic integrals and Mobius transformation, which carry a half plane in the arc length into the rectangle with vertices $\pm K . \pm K + iK'$ The integrators used are the Trapezoidal rule and Runge–Kutta fourth-order method. All calculations of the results are implemented using **Python code** commands in MATLAB. It is further proven that conver

gence of Cauchy's integral theorem for the matrix function converged uniformly locally on the solid triangle. In addition, the hyperbolic tangent matrix accompanying Cauchy's integral theorem for the matrix function is calculated.

The work is motivated as follows.

The standard trapezoidal rule in the explicit form is given by

$$\int_0^b f(z)dx = \frac{h}{2}\left[f(z_0) + 2\sum_{j=1}^{N-1} f(z_j) + f(z_S) \right], \tag{23}$$

$h = \frac{b-a}{N}, z_j = a + jh, j = 1,2,...,N-1$.The implementation of equation (23) was considered an initial value problem

$$u_{j+1}(A) = u_j(A) + \frac{h}{2}\left[f(A_0) + 2\sum_{j=1}^{N-1} f(A_j) + f(A_s) \right], j = 0,1,2,..$$

Similarly, the standard Runge–Kutta method stage 4 (fourth-order method) in the explicit form as in Suli(2013) is written as:

$$w_{n+1} = w_n + h\phi(z_n, w_n; h). \tag{24}$$

$$\phi(z, w; h) = \sum_{j=1}^{R} c_j p_j ,$$

$$p_1 = f(z_0, w_0)$$

$$p_j = f\left(z + ha_j, w + h\sum_{q=1}^{j-1} b_{jq} p_q \right), \; j = 2,3,...,R , \tag{25}$$

$$a_j = \sum_{q=1}^{j-1} b_{jq}, \; j = 2,3,...,R . \tag{26}$$

Hence equation (24) expressing the four-stage Runge–Kutta method (R-K-4) in the explicit form is

$$w_{n+1} = w_n + \frac{h}{6}(p_1 + 2p_2 + 2p_3 + p_4) \tag{27}$$

$$p_1 = f(z_n, w_n),$$
$$p_2 = f\left(z_n + \frac{h}{2}, w_n + \frac{hp_1}{2}\right)$$
$$p_3 = f\left(z_n + \frac{h}{2}, w_n + \frac{hp_2}{2}\right)$$
$$p_4 = f(z_n + h, w_n + hp_3)$$
$$n = 0,1,2,...,.$$

The basic extra properties of Runge–Kutta method are discussed herewith in the strict sense. Here p_i are the nodes in the Runge–Kutta method. For instance, p_2 and p_3 are approximations of derivative $w'(.)$ at points on the solution curve, an intermediate between $(z_n, w(z_n))$ and $(z_{n+1}, w(z_{n+1}))$. Therefore, $\Phi(z_n, w_n, h)$ is a weighted average of $p_i, i = 1,2,3,4$. The weights correspond to those of Simpson's rule method, for which the Runge–Kutta method decreases when $\frac{\partial f}{\partial w} \equiv 0$. Thus, the R-stage explicit Runge–Kutta method of order of accuracy is $1 \le R \le 4$. It should be noted that Euler's method is the least common Runge–Kutta method over an interval $(z_0, z_N), z_N >> z_0$. The absolute stability $\bar{h}$ of R-K order 4 in equation is $\bar{h} \in (-2.78, 0)$.

The two methods of equations (23) and (27) in matrix form for Cauchy's integral theorem are hereby implemented. These steps are described in section 3 in this paper.

2.1 Analysis of convergence of the Cauchy's integral theorem for the matrix function.

Theorem

The Runge-Kutta method when applied to the Cauchy's integral theorem given in equation 1.1 converges locally uniformly on the solid triangle $\partial\nabla$.

Proof

The analysis of convergence is presented as follows. Following carefully standard discussions in the complex analysis, it suffices to show that the sequence of iterates $F_1, F_2, ..., F_k$ of $A \to F(A)$ as $k \to \infty$ converges. Take $A_0 \in D \subseteq \mathbb{C}$, in which the largest open disk $B(A_0, r)$ holds. Since F is holomorphic on D and continuous at each point of the ball $B(A_0, r)$ for every solid triangle $\partial\nabla \subseteq B(A_0, r)$, $\int_{\partial\nabla} F_k = 0$ (by Cauchy's integral theorem). Via Borel's covering theorem for each family $\{D_i\}_{k\in I}^{\infty}$ of open sets in $\mathbb{C}$ covering $B \subseteq \bigcup_{k\in I} D_k$ there are infinitely many indices $i_1, i_2, ..., i_k \in I$ such that $B \subseteq D_{i_1} \bigcup D_k \bigcup_{i_0} ... \bigcup D_{i_k}$.

If it can be proven that $F_k : D \to \mathbb{C}$, $A_0 \in D$ for $r > 0$ when $B(A_0, r) \subseteq D$, then $F_k(A) \to F(A)$ uniformly locally will hold verbatim. Take for each $A_0 \in B(A, r)$ such that

$F_k'(A_0) = F'(A_0) = \frac{1}{2\pi i} \int_{\partial\nabla} \frac{F_k(A) - F(A)}{(z - A_0)^2} dz$. If we take $A_0 = B\left(A, \frac{r}{2}\right)$ and $z \in \partial B(A, r)$, then

$\left|(z - A_0)^2\right| \geq \left(\frac{r}{2}\right)^2$, where $\left|F_k'(A_0) - F'(A_0)\right| \leq \frac{2\pi r}{2\pi}\left(\frac{2}{r}\right)^2 \sup_{z \in \partial B(A_0, r)} \left|F_k'(A) - F(A)\right| =$

$\sup_{A_0 \in B(A, \frac{r}{2})} \left|F_k'(A) - F'(A)\right| \leq \frac{4}{r} \sup_{A \in \partial B(A_0, \frac{r}{2})} \left|F_k(A) - F(A)\right|$. Hence, $F_k' \to F'$ is uniformly distributed locally on $B(A, \frac{r}{2})$ and $A_0 \in D$.

Therefore, $F_k \to F$ uniformly on the solid triangle $\partial\nabla$, implying that $\int_{\partial\nabla} F_i \to \int_{\partial\nabla} F = 0$ according to Morera's theorem. That $|F(A)|$ is bounded follows from Liouville's theorem which says that a bounded entire function is a constant. This proves the theorem as expected.

2.2 The matrix hyperbolic tangent

In this section, the matrix hyperbolic tangent calculation for the Cauchy integral theorem is presented. By writing out the hyperbolic cosh and sinh for the argument A, we have

$$\cosh(A) = \frac{1}{2}\left(e^A + e^{-A}\right), \sinh(A) = \frac{1}{2}\left(e^A - e^{-A}\right),$$

Defining the matrix hyperbolic $\tanh(A)$ in the form:

$$\tanh(A) = \frac{\sinh(A)}{\cosh(A)} = \left(e^{2A} - I\right)\left(e^{2A} + I\right)^{-1},$$

(28)

we are able to demonstrate using the ideas in Higham (2008) that equation (28) is the rate at which the matrix tangent is approximated in the arc length in the Jacobi elliptic integral. Forming the matrix version of equation (28) by the following equation:

$$\tanh(2A) = 2\left(I + \tanh^2(A)\right)^{-1} \tanh(A), \tag{29}$$

the iterative function for the hyperbolic tangent is written as:

$$\begin{aligned} T_0 &= \tanh(2^{-s} A) \\ T_i &= 2\left(I + T_{i-1}^2(A)\right)^{-1} T_{i-1}(A),\ i = 1,2,3,\ldots,s. \end{aligned} \tag{30}$$

A bound for the cosine matrix in terms of the hyperbolic cosine matrix function can be found in Higham (2008). As a result, we omit here.

3. RESULT AND DISCUSSION

Problem 1

We first demonstrate by computing the Jacobi elliptic integral of equation (31) with a parameter modulus k=0.5, range of integration from 0 to $\frac{\pi}{2}$ and an error tolerance =10^{-16}:

$$\int_0^{\frac{\pi}{2}} \frac{1}{\sqrt{1-0.5^2\sin^2 x}}\,dx. \tag{31}$$

Using Gauss–Kronrod quadrature formula in transforming the range of the integral from $[0,\frac{\pi}{2}]$ to the interval $[-1,1]$ by substitution of $t=\sin(x), dt=\cos(x)dx, x=\arcsin(t)$, then equation (31) is solved numerically. The described procedure is

$$\int_0^{\frac{\pi}{2}} \frac{1}{\sqrt{1-0.5^2\sin^2(x)}}\,dx = \int_0^{1} \frac{1}{\sqrt{\left(1-0.5^2t^2\right)}}\,dt. \tag{32}$$

By integrating the integral in equation (32) at the equatorial plane with the adaptive Gauss–Kronrod quadrature and an error of tolerance (tol)=10^{-16}, the result obtained from the sciPython codes in MATLAB is 1.4674622093394272 with an error = $2.270350431570743e^{-17}$.

Problem 2

We generated a random matrix of order 5, labeled Table 1, for the experiment.

Table 1: *Generated random matrix of order 5 used for the experiment*

$$A=\begin{bmatrix} 0.25801885 & 0.08437114 & 0.71655646 & 0.10041998 & 0.44629278 \\ 0.77205093 & 0.01997296 & 0.21189686 & 0.18978914 & 0.75854856 \\ 0.88895881 & 0.73204706 & 0.71331618 & 0.60247364 & 0.1601281 \\ 0.13290724 & 0.67665497 & 0.4167684 & 0.65027641 & 0.06418972 \\ 0.28532094 & 0.71014126 & 0.06093577 & 0.54659253 & 0.3023927 \end{bmatrix}$$

Adopting the following parameters for implementation, with step size (h) =0.1 and n=10 at the equatorial plane and a range of integration from 0 to $\pi/2$, it is displayed below the following results computed by the described methods using equation (1):

Table 2: *Results for the accelerated trapezoidal rule from Equation (1)*

Iteration (i)	$F_i(A)$
1	$\begin{bmatrix} 0.66122355 & 0.26202712 & 0.52997965 & 0.26204834 & 0.52993904 \\ 0.58369595 & 0.57492095 & -0.06000577 & 0.52486723 & 0.18409195 \\ -0.10415391 & 0.19986788 & 0.4674309 & -0.38085506 & -0.4500728 \\ 0.56807995 & -0.2875909 & 0.50925365 & 0.28910214 & 0.13780262 \\ 0.58485072 & 0.47704239 & 0.53273793 & -0.30375126 & -0.18237996 \end{bmatrix}$
2	$\begin{bmatrix} 0.66855409 & 0.2595369 & 0.52891868 & 0.25878217 & 0.5335394 \\ 0.57048726 & 0.59008025 & -0.04324825 & 0.52420147 & 0.21171219 \\ -0.12668553 & 0.16106762 & 0.47274845 & -0.36595635 & -0.47170857 \\ 0.54857079 & -0.2922356 & 0.51570533 & 0.31808538 & 0.15205018 \\ 0.60842813 & 0.45680555 & 0.53600438 & -0.3291462 & -0.1740645 \end{bmatrix}$
3	$\begin{bmatrix} 0.67150253 & 0.25809938 & 0.52919607 & 0.25763915 & 0.5352494 \\ 0.5611075 & 0.59660832 & -0.03229445 & 0.52380309 & 0.22703619 \\ -0.13761839 & 0.145684 & 0.47496315 & -0.35593102 & -0.48665474 \\ 0.53279124 & -0.29388497 & 0.5139907 & 0.34329814 & 0.16281644 \\ 0.62212906 & 0.44259916 & 0.53814107 & -0.35235828 & -0.16537235 \end{bmatrix}$
4	$\begin{bmatrix} 0.67252492 & 0.25776133 & 0.52937848 & 0.25744506 & 0.53633315 \\ 0.55673505 & 0.59985462 & -0.02536512 & 0.52358506 & 0.23768249 \\ -0.14465248 & 0.13657618 & 0.47604356 & -0.34919988 & -0.49870489 \\ 0.52267549 & -0.29473825 & 0.51266629 & 0.35479856 & 0.17040961 \\ 0.62770381 & 0.43291652 & 0.53895159 & -0.35665647 & -0.16954192 \end{bmatrix}$
5	$\begin{bmatrix} 0.67304637 & 0.25760855 & 0.52948581 & 0.25734122 & 0.53709347 \\ 0.55418155 & 0.60154977 & -0.02072446 & 0.52346017 & 0.24439899 \\ -0.14853273 & 0.13041324 & 0.47675523 & -0.34589002 & -0.50644966 \\ 0.51675562 & -0.29522115 & 0.5116795 & 0.36116733 & 0.17453261 \\ 0.63085409 & 0.42644143 & 0.53946744 & -0.3589423 & -0.17238914 \end{bmatrix}$

Table 3: *Results for the accelerated Runge–Kutta method from Equation (1.1)*

Iteration (i)	$F_i(A)$
1	$\begin{bmatrix} 0.68028624 & 0.25140807 & 0.52597836 & 0.25064018 & 0.54363047 \\ 0.53569447 & 0.61640679 & 0.00790439 & 0.5231249 & 0.26720814 \\ -0.17424053 & 0.10079003 & 0.48100695 & -0.33073379 & -0.53714707 \\ 0.4823942 & -0.2975003 & 0.50673045 & 0.40193421 & 0.19331875 \\ 0.65433654 & 0.39728332 & 0.54568855 & -0.38179333 & -0.21266457 \end{bmatrix}$

2	$\begin{bmatrix} 0.68229572 & 0.2505735 & 0.5262397 & 0.250073 & 0.54508753 \\ 0.52562545 & 0.6215093 & 0.01851547 & 0.52283671 & 0.27482613 \\ -0.18006684 & 0.09300326 & 0.48230001 & -0.32803818 & -0.54357813 \\ 0.47253362 & -0.29838432 & 0.50550496 & 0.41123074 & 0.20043846 \\ 0.63428713 & 0.39628777 & 0.54622596 & -0.3917793 & -0.20664057 \end{bmatrix}$
3	$\begin{bmatrix} 0.6846653 & 0.25014754 & 0.52660676 & 0.2498409 & 0.54634967 \\ 0.51887278 & 0.62606069 & 0.02840634 & 0.52268235 & 0.28330772 \\ -0.185616659 & 0.08643407 & 0.4836778 & -0.325554311 & -0.54927157 \\ 0.46423896 & -0.29904057 & 0.50433754 & 0.41738403 & 0.20624229 \\ 0.63671633 & 0.38728723 & 0.54671909 & -0.39974654 & -0.20196377 \end{bmatrix}$
4	$\begin{bmatrix} 0.68679836 & 0.24996739 & 0.52682462 & 0.24973844 & 0.54707808 \\ 0.5140653 & 0.62867415 & 0.03441661 & 0.52260317 & 0.28892945 \\ -0.18952096 & 0.08233679 & 0.48479306 & -0.3238389 & -0.55231332 \\ 0.45939874 & -0.29941442 & 0.50352091 & 0.42117221 & 0.20938303 \\ 0.63825215 & 0.38345502 & 0.54705771 & -0.40540613 & -0.20010367 \end{bmatrix}$
5	0.68869211 0.24988417 0.52695461 0.24967749 0.54746835 0.51080418 0.63067383 0.03853275 0.52257226 0.2929114 −0.19221012 0.07994274 0.48572737 −0.32259946 −0.55459047 0.45594415 −0.29965007 0.5029662 0.42367594 0.21125775 0.63914899 0.38093069 0.5472893 −0.40920944 −0.19881107

3.1 Numerical results calculated from equation (18)

In another development, we also used the randomly generated matrix of order 5 from Table 1. We computed a version of Cauchy's integral theorem as represented in equation (18) using Trapezoidalrule and Runge-Kutta method by typing instruction command F(A) = (-A/2ipi)*inv(z)*f(z(t))*inv(z((t)*I)-A)*(dz/du)*(du/dt)*dt, range of integration is from (-K+(i*0.5*Kprime)), K+i*0.5*Kprime)), t(j) = -K+i*0.5*Kprime+inv(N)*2*(j-0.5)*K, where 1 is less than *j* and *j* is less than or equal to *N*. Use Mobius transformation to speed up the Trapezoidal rule and Runge Kutta fourth-order method, respectively. Display results for F(A) are in tabular forms as given below .

The results are presented in Tables 4 and 5.

Table 4: *Trapezoidal rule computed from equation (18)*

Iteration (i)	$F_i(A)$
1	$\begin{bmatrix} 0.61465978 & 0.32399915 & 0.44445184 & 0.32401969 & 0.44441297 \\ 0.65136457 & 0.49124121 & -0.01158776 & 0.42796052 & 0.24620115 \\ 0.09131779 & 0.20228201 & 0.38125882 & -0.29612714 & -0.2868623 \\ 0.44836479 & -0.15646665 & 0.40015325 & 0.18073812 & -0.07514439 \\ 0.46078923 & 0.23046281 & 0.44818892 & -0.23387258 & -0.07453366 \end{bmatrix}$
2	$\begin{bmatrix} 0.65317095 & 0.31599829 & 0.44687218 & 0.31401975 & 0.44681821 \\ 0.62539479 & 0.53522886 & 0.02043178 & 0.42940639 & 0.28633807 \\ 0.08457808 & 0.1614653 & 0.38626679 & -0.28832977 & -0.32445791 \\ 0.43632614 & -0.17407206 & 0.40226721 & 0.21846889 & -0.16625455 \\ 0.45535983 & 0.24386613 & 0.44737045 & -0.19114177 & -0.16107213 \end{bmatrix}$
3	$\begin{bmatrix} 0.66468002 & 0.30898676 & 0.4491375 & 0.30401981 & 0.4490726 \\ 0.60352901 & 0.57207475 & 0.04920886 & 0.43231926 & 0.32649298 \\ 0.08085806 & 0.12648711 & 0.39137786 & -0.27956673 & -0.36593503 \\ 0.4226447 & -0.18941321 & 0.40589919 & 0.25143777 & -0.25507435 \\ 0.4483083 & 0.25619278 & 0.44660535 & -0.14890639 & -0.23612714 \end{bmatrix}$
4	$\begin{bmatrix} 0.67921769 & 0.302165 & 0.45140282 & 0.29401986 & 0.45131833 \\ 0.57543125 & 0.60425444 & 0.08525241 & 0.43665692 & 0.36713667 \\ 0.08008088 & 0.09556726 & 0.39652937 & -0.26960746 & -0.4020445 \\ 0.403554 & -0.20473723 & 0.40908018 & 0.28213668 & -0.34394226 \\ 0.43543784 & 0.26724776 & 0.44605352 & -0.10311986 & -0.30868219 \end{bmatrix}$
5	$\begin{bmatrix} 0.69709872 & 0.29563332 & 0.45366814 & 0.2840199 & 0.45356403 \\ 0.54851534 & 0.63050948 & 0.12458032 & 0.44196904 & 0.40957456 \\ 0.08133448 & 0.07098832 & 0.40164086 & -0.26063142 & -0.43443955 \\ 0.38612381 & -0.21687453 & 0.41233466 & 0.31248741 & -0.42977506 \\ 0.42660522 & 0.27625735 & 0.44551324 & -0.05767039 & -0.37830273 \end{bmatrix}$

Table 5: *The results were computed with the Runge–Kutta method from equation (18)*

Iteration (i)	$F_i(A)$
1	$\begin{bmatrix} 0.68639699 & 0.25000115 & 0.52687567 & 0.24962379 & 0.54731377 \\ 0.51445561 & 0.62895703 & 0.03452149 & 0.52214654 & 0.28933822 \\ -0.19246126 & 0.08275806 & 0.48486508 & -0.32311043 & -0.55368788 \\ 0.45711006 & -0.29940295 & 0.50355096 & 0.4211544 & 0.20988447 \\ 0.63848085 & 0.38357862 & 0.54726911 & -0.40442361 & -0.20059477 \end{bmatrix}$

2	$\begin{bmatrix} 0.68639855 & 0.24999836 & 0.52687466 & 0.24962541 & 0.54731233 \\ 0.51445485 & 0.6289574 & 0.03452071 & 0.52214681 & 0.28933753 \\ -0.19246067 & 0.08275815 & 0.48486505 & -0.32310984 & -0.55368777 \\ 0.45710957 & -0.29940303 & 0.50355076 & 0.42115445 & 0.20988433 \\ 0.63847999 & 0.38357868 & 0.54726899 & -0.40442346 & -0.20059471 \end{bmatrix}$
3	$\begin{bmatrix} 0.68639859 & 0.24999834 & 0.52687466 & 0.24962544 & 0.54731231 \\ 0.51445484 & 0.6289574 & 0.0345207 & 0.52214682 & 0.28933752 \\ -0.19246067 & 0.08275816 & 0.48486505 & -0.32310984 & -0.55368777 \\ 0.45710956 & -0.29940303 & 0.50355075 & 0.42115445 & 0.20988433 \\ 0.63847998 & 0.38357868 & 0.54726899 & -0.40442346 & -0.20059471 \end{bmatrix}$
4	$\begin{bmatrix} 0.68639859 & 0.24999834 & 0.52687466 & 0.24962544 & 0.54731231 \\ 0.51445484 & 0.6289574 & 0.0345207 & 0.52214682 & 0.28933752 \\ -0.19246067 & 0.08275816 & 0.48486505 & -0.32310984 & -0.55368777 \\ 0.45710956 & -0.29940303 & 0.50355075 & 0.42115445 & 0.20988433 \\ 0.63847998 & 0.38357868 & 0.54726899 & -0.40442346 & -0.20059471 \end{bmatrix}$
5	$\begin{bmatrix} 0.68639859 & 0.24999834 & 0.52687466 & 0.24962544 & 0.54731231 \\ 0.51445484 & 0.6289574 & 0.0345207 & 0.52214682 & 0.28933752 \\ -0.19246067 & 0.08275816 & 0.48486505 & -0.32310984 & -0.55368777 \\ 0.45710956 & -0.29940303 & 0.50355075 & 0.42115445 & 0.20988433 \\ 0.63847998 & 0.38357868 & 0.54726899 & -0.40442346 & -0.20059471 \end{bmatrix}$

Problem3

The iteration of the hyperbolic tangent matrix (Algorithm):

```
import numpy as np
# Define the recursive function for the matrix hyperbolic tangent
def matrix_tanh(A, s):
I = np.eye(A.shape[0])
T = np.tanh(2 ** (-s) * A)
print("T(0):")
print(T)
for i in range(1, s + 1):
T = (I + T @ T)-1(2 * T)
print(f"T({i}):")
print(T)
```

```
return T
# Generate the randomly generated matrix A of order 5
A = np.random.rand(5, 5)
# Print the original randomly generated matrix A
print("Original Matrix A:")
print(A)
# Compute the hyperbolic matrix tangent iteratively with s = 5
s = 5
T = matrix_tanh(A, s)
```

Table 6: *Results for the hyperbolic tangent matrix computed from the generated random matrix in Table 1:*

Iteration (i)	Hyperbolic tangent matrix $T_i(A)$
0	$\begin{bmatrix} 0.15753113 & 0.05141178 & 0.39582785 & 0.05516332 & 0.24396313 \\ 0.46368214 & 0.01000924 & 0.10620726 & 0.09403456 & 0.37584837 \\ 0.53669286 & 0.41094985 & 0.39571235 & 0.33337243 & 0.0889126 \\ 0.08112564 & 0.39199043 & 0.24500977 & 0.37875992 & 0.03745621 \\ 0.17385077 & 0.40455655 & 0.03551933 & 0.31669717 & 0.19704917 \end{bmatrix}$
1	$\begin{bmatrix} 0.21449192 & 0.06996306 & 0.5391831 & 0.07526016 & 0.33330992 \\ 0.6041413 & 0.01357378 & 0.14390823 & 0.12748464 & 0.50512694 \\ 0.69542571 & 0.53178338 & 0.51342213 & 0.43382329 & 0.11504995 \\ 0.10571284 & 0.51733415 & 0.32323036 & 0.49894448 & 0.04950642 \\ 0.22730964 & 0.53270297 & 0.03926975 & 0.43762252 & 0.27375433 \end{bmatrix}$
2	$\begin{bmatrix} 0.25707015 & 0.083816 & 0.65076286 & 0.09106935 & 0.40140824 \\ 0.72437042 & 0.01637052 & 0.17278868 & 0.15303652 & 0.61028836 \\ 0.8370439 & 0.63874081 & 0.61741553 & 0.52089633 & 0.13819535 \\ 0.12666991 & 0.61728655 & 0.38542533 & 0.59469714 & 0.0590924 \\ 0.2734785 & 0.63380182 & 0.04710944 & 0.5265505 & 0.3106507 \end{bmatrix}$
3	$\begin{bmatrix} 0.28634417 & 0.09320348 & 0.72116584 & 0.10099513 & 0.44361969 \\ 0.8022867 & 0.01907137 & 0.20066106 & 0.1774957 & 0.70802812 \\ 0.92313724 & 0.70328649 & 0.67910974 & 0.5728142 & 0.15212507 \\ 0.14006668 & 0.6790742 & 0.42501347 & 0.65492725 & 0.06802047 \\ 0.30252053 & 0.69613196 & 0.05226591 & 0.57862206 & 0.34006683 \end{bmatrix}$

4	$\begin{bmatrix} 0.30582663 & 0.09914739 & 0.77363366 & 0.10813796 & 0.47619454 \\ 0.85534307 & 0.02034745 & 0.21374696 & 0.18938825 & 0.75712011 \\ 0.98878547 & 0.75355911 & 0.72875861 & 0.61477423 & 0.16317469 \\ 0.14991932 & 0.7273317 & 0.45530306 & 0.70366016 & 0.07457629 \\ 0.32411178 & 0.74569767 & 0.05768224 & 0.62026323 & 0.38936309 \end{bmatrix}$
5	$\begin{bmatrix} 0.32047785 & 0.10366497 & 0.81145715 & 0.11378992 & 0.50079897 \\ 0.89396871 & 0.0213989 & 0.22451019 & 0.1990252 & 0.79681288 \\ 1.03400185 & 0.78749714 & 0.76148679 & 0.641464 & 0.17047864 \\ 0.15764049 & 0.75967075 & 0.47625517 & 0.73574888 & 0.08339052 \\ 0.34033269 & 0.77777116 & 0.06022539 & 0.64734485 & 0.40628483 \end{bmatrix}$

3.2 Discussion of Results

In this section it is analysed the results obtained from the computation of Cauchy's integral theorem for the matrix function. Using the Jacobi elliptic integrals in conjunction with Mobius transformation, it is executed in the computation the Trapezoidal rule and Runge-Kutta fourth order method.

The computed values of Jacobi elliptic integrals at the equatorial plane in the course of calculations with modulus k from equation (7b) are given below**:**

K(k): 1.5707963267948966, K'(k): 1.5707963267948966. $K(k)$ and $K'(k)$ are the Jacobi elliptic integrals of the first and second kinds, respectively.

These computed results are presented in tabular form as shown in Tables 2-5 for the approximate values for the matrix function $F_i(A)$ while Table 6 shows the results computed for the matrix hyperbolic tangent in that order. It is the rate at which the hyperbolic sinh is changing with respect to hyperbolic coshine in the argument matrix A. Looking closely at Tables 2 and 3, these were computed using the equation 1.1 with described operators namely, Trapezoidal rule and Runge-Kutta fourth order method making uses of the Jacobi elliptic integrals and Mobius transformation respectively. it can be observed that Tables 4 and 5 are the numerical values computed using equation (18) with specified acceleration techniques. The computed results are of high quality, as attested by the use of pyton codes in MATLAB. We used Jacobi elliptic integrals for the arc length, which appeared in the Cauchy integral theorem. The Mobius transformation that was incorporated into the computation acted as a preservation map for the improved results and thus the sped up computation processes in the Trapezoidal rule and Runge–Kutta fourth-order method. Especially the interval of absolute stability of Runge-Kutta method was stated which is much higher than the Trapezoidal rule with low absolute interval of stability [5]. The Mobius transformation involves

conformal mapping that maps the region of interest to the unit disk. Then, the Cauchy integral theorem is used in the unit disk, relating the values of the function on the boundary of the region to its values in the interior. It uses Jacobi elliptic integrals to compute the values of the function in the interior and then uses the inverse Mobius transformation to map these values back to the original region.

A careful analysis of the results computed from using equations (1) and (18) showed that the Runge–Kutta fourth-order method always performed substantially faster than previously known techniques involving the trapezoid rule. To illustrate this, the Runge–Kutta method converged at the second iteration, whereas the Trapezoidal rule converged at the fourth iteration in each tested experiment; see, for example, Tables 2-4 above. We halt the iterations when $\|F_{i+1}(A) - F_i(A)\|$ is less than a certain prescribed threshold. For example, the results displayed in Tables 4 and 5 provide further insights into what was studied.

4. CONCLUSION

The Cauchy integral theorem for the matrix function was modified to accommodate the Runge–Kutta fourth-order method, which involved the calculations of Jacobi elliptic integrals and the Möbius transformation. In course of calculating the Jacobi elliptic integrals, it was introduced the use of Gauss-Konrod - quadrature formula that transformed the range of integration from the interval $[0,\frac{\pi}{2}]$ to interval $[-1,1]$ at the equatorial plane. The numerical results obtained in using Runge-Kutta method were compared with those calculated by the trapezoidal rule for the matrix $F(A)$. The results were quite revealing the high performances of Runge-Kutta method over Trapezoidal rule. This further confirmed the space and time complexities for each method discussed earlier in section 1 above in the paper. It is proven that Cauchy's integral theorem converges uniformly locally to the requested approximate solution as the sequence of iterates goes to infinity. The intervals of absolute stability of the Trapezoidal rule and Runge–Kutta method were given. Comparing computational complexities in both space and time, it should be noted that much more work is required when the Runge–Kutta method is used than when the trapezoidal rule is used. Notwithstanding the extra efforts which Runge –Kutta method entail in the execution of the presented Cauchy's integral theorem for the matrix function, it is worthwhile of its reputation over Trapezoidal rule. The numerical results computed from using equations (1) and (18) showed close similarities to the desired solutions. Furthermore, in another development, the calculation of the hyperbolic tangent matrix was executed as demonstrated in problem 3. It is the rate at which hyperbolic sinh matrix changes with respect to the coshine matrix

according to Cauchy's integral theorem when the contour swings around the integral with the matrix eigenvalues in the counter-clockwise direction. The results are iterative and are reported in Table 6.

CONFLICT OF INTEREST

No conflict of interest was declared by the author.

ACKNOWLEDGEMENT

The author is grateful to Kogi State University, Anyigba, Kogi State for the TETFUND grant in 2016.

REFERENCES

[1] Barnard, R. W., Pearce, K. and Richards, K. C. (2009). An inequality involving the generalized hypergeometric function and the arc length of an ellipse. *SIAM J. on Mathematical Analysis*, 31(3), 693-699.

[2] Batista, M. (2019). A collection of MATLAB functions for the computation of elliptic integrals and Jacobian elliptic functions of real arguments. Software X 10 , 100245.

[3] Dobrushkin, V. (2022). Mathematical Tutorial for the second course. Part VII. Elliptic functions.

file:///C:Users/User/Desktop/Mathematical%2TUTORIAL,%20part%202.7%20Elliptic%20Functios.html

[4] Hales, N., Higham, N. J. and Trefethen, L. N. (2008). Computing A^{α}, $\log(A)$ and related matrix functions by Contour integrals. *SIAM J. Numer. Anal.* 48(5), 2505-2523.

[5] Higham, N. J. (2008). Functions of Matrices: Theory and Computation. *Society for Industrial and Applied Mathematics*, *Philadelphia*.

[6] Ibanez, J., Alonso, J. M., Sastre, J., Defez, E., and Alonso-Jorda, P. (2021). Advances in the approximation of the matrix hyperbolic tangent. *Mathematics,* 1219 (9). https://doi.org/10.3390/math9111219.

[7] Jefferies, B. (2012). A generalization of the Cauchy integral formula for normal matrices. Complex Analysis and Operator theory. https://www.researchgate.net/publication/257759396.

[8] Jodar, L. and Cortis, J. C. (1989). Some properties of Gamma and Beta matrix functions, *Appl. Math. Lett.*, 11(1), 89-93.

[9] Jodar, L. and Sastry, J. (1998). On matrix polynomials. *Util. Math.* 53,37-48.

[10] Lawden, D. (1989). Elliptic Functions and Applications, *Springer-Verlag, New York*

[11] Nagar, K. D., Joshi, L., and Gupta, A. (2012). Matrix variate Pareto distribution of the second kind. International Scholarly Research Network, *Probability and Statistics*, 2012, Article ID 789273. Doi:10.5402/2012/789273

[12] Nakatsukasa, Y. and Higham, N. J. (2011). Backwards stability of iterations for the computing the polar decomposition. MIMS E print; 2011.103. *The University of Machester, UK.*

[13] Springborn, B. (2018). Complex analysis 1, Oliver Gross and Tobias Paul, Germany.

[14] Suli, E. (2013). Numerical solution of Ordinary differential equations, Lecture Notes, *Oxford*.

[15] Takihira, S., Ohashi, A., Sogabe, T. and Usuda, T.S. (2021). Quantum algorithm for matrix functions by Cauchy's integral theorem formula. *Quantum Information and Computation*, 0(0) , 00-00.

[16] Uwamusi, S. E. (2017a). Extracting p th root of a matrix with positive eigenvalues via Newton and Halley's methods. *University of Ilorin Journal of Science*, 4(1), 1-16.

[17] Uwamusi, S. E. (2017b). Computing Square Root of Diagonalizable Matrix with Positive Eigenvalues and Iterative Solution to Nonlinear System of Equation: The Role of Lagrange Interpolation Formula, *Transactions of the Nigerian Association of Mathematical Physics,* 5, 65-72.

[18] Uwamusi,S.E. (2021). The Gamma and Beta Matrix Functions and Other Applications. *Unilag Journal of Mathematics and Applications*,1(2), 186-207. URL:https://lagjma.edu.ng

[19] Uwamusi, S. E. (2022). Computing Cauchy Integral Theorem of a Matrix Function via Similarity Transformation and Behavior of Hypergeometric Matrix Density Function. *Transaction of the Nigerian Association of Mathematical Physics*, 18, 87-100.

[20] Uwamusi, S.E. (2024). Bounds for Weierstrass elliptic function and Jacobi elliptic integrals of first and second kinds, *Earthline Journal of Mathematical Sciences*, 14(3), 535-564. https://doi.org/10.34198/ejms.14324.535564.

[21] Viennot, G. (1980). Un interpretation combinatoire des coe cients de developments en s enrie enti ere des fonctions elliptiques de Jacobi, *J. Combin. Theory Ser. A*. 29, 121(133).

[22] Yang, Z. H. and Tian, J. (2017). Convexity and monotonicity for the elliptic integrals of the first kind and applications. arXiv:1705.05703V1 [math.CA]