Cost and Time Efficient Derivative Based Midpoint Closed Newton-Cotes Quadrature

Sana Khan Abbasi*, Zubair Ahmed Kalhoro†, Sanaullah Jamali‡, Jinrui Guan§

Abstract

A variety of methods, collectively known as "numerical integration," are employed in numerical analysis to approximate the value of a definite integral. Among these, the Newton-Cotes formulas represent a key category of numerical techniques for evaluating such integrals. These methods are especially valuable for integrating functions that involve singularities or nonlinearities. The primary aim of this research is to propose more efficient techniques using centroid mean that offer higher accuracy, greater precision, and reduced errors. The study also emphasizes the theoretical analysis of errors, including theorems related to the order of accuracy and error terms for the developed methods. To assess the effectiveness of the new methods, comparisons are made with other classical approaches through numerical tests on various commonly used integrals, as reported in existing literature. The methods are implemented using MATLAB (R2018b) for high-level computer programming. All results were noted in Intel(R) Core(TM) i3-4010U with RAM 4.00GB Laptop and processing speed of 1.70GHz.

Keywords: Quadrature Rule, Definite Integral, Newton-Cotes Formulae, Precision, Order of Accuracy.

Introduction

Numerical analysis offers a wide range of methods for approximating solutions to complex scientific problems using only arithmetic operations. It focuses on developing, analyzing, and applying algorithms to solve mathematical problems with a specified level of accuracy. As computational power improves or new methods emerge, some algorithms may become outdated or replaced by more effective ones. In both applied and pure mathematics, approximating definite integrals is a crucial task, particularly when their exact evaluation is either impossible with known analytical methods or too time-consuming for practical use. Various numerical techniques are available in the literature, especially for integrating functions that involve singularities and nonlinearities.

*Institute of Mathematics & Computer Science, University of Sindh, Jamshoro 76080, Pakistan, zaabbasi419@gmail.com

†Institute of Mathematics & Computer Science, University of Sindh, Jamshoro 76080, Pakistan, zubair.kalhoro@usindh.edu.pk

‡Corresponding Author: University of Sindh, Laar Campus, Badin 71000, Pakistan, sanaullah.jamali@usindh.edu.pk

§School of Mathematics and Statistics, Taiyuan Normal University, Jinzhong 030619, China, guanjinrui2012@163.com

Examples include integrals $\int_0^1 \frac{e^{-x}}{x^{2/3}} dx$, $\int_{-\infty}^1 sin\, x^2\, dx$ and $\int_0^1 \frac{sin\, x}{x} dx$, among others.

There are numerous applications of integration in the fields of science and engineering, including the determination of the area under a curve, the area between two curves, the length of an arc, the volume of a solid object, the moment of inertia, and the centroid (Saand et al., 2022). Numerical methods are employed to determine an appropriate value for the defined integral.

$$I = \int_\alpha^\beta f(x) \tag{1}$$

Definite integral cannot be solved analytically or when the function to be integrated is unknown and only a few values of the function $y = f(x)$ are provided. In such cases, a suitable interpolation formula is employed to solve the integral. This process is referred to as QUADRATURE when it is applied to a single-variable function.

$$I = \int_\alpha^\beta f(x)\, dx \approx \sum_{i=0}^{i=n} w_i f(x_i) \tag{2}$$

Where w_i is referred to as the weight function and can be obtained through a variety of methods. The most commonly used approach is to interpolate the function $f(x)$ at the $n+1$ points $x_i, x_2, x_3, \dots, x_n$ using the interpolation formula and subsequently integrate it to obtain (2). Alternatively, the precision of QUADRATURE can be employed to obtain w_i, thereby reducing the error.

$$R_n(f) = \int_\alpha^\beta f(x)\, dx \; - \sum_{i=0}^{i=n} w_i f(x_i) \tag{3}$$

For $f(x) = x^j, (j = 0, 1, 2, 3, \dots, n)$, the value is precisely 0. Consequently, the system of $n+1$ linear equations for w_i will be generated by employing (3). The effectiveness of quadrature rules is typically evaluated based on their degree of precision and order of accuracy. The degree of precision refers to the highest degree of polynomial that a given rule can integrate exactly, while the order of accuracy indicates the rate of convergence of the global truncation error term. As a result, achieving higher precision and accuracy in numerical integration formulas presents a key challenge in numerical analysis. Numerous researchers consistently prioritize numerical integration. Atkinson (1978) was the first to employ the concept of end-point derivatives to enhance the accuracy of the original Newton-Cotes formulas by formulating a corrected trapezoidal (Atkinson, 1989). Subsequently, the concept of end-point derivatives was employed to enhance the efficiency of other Newton-Cotes formulas, including Simpson 1/3 rule and Simpson 3/8 rule. Consequently, the corrected Newton-Cotes formulas were further discussed and improved by numerous researchers, including (Ujevic & Roberts, 2004; Acu et al., 2008). Dehghan et al.

(2005a, 2005b, 2005c) conducted research on the Newton Cotes Quadrature rule for closed, open, and semi-open domains. Their approach involved locating the boundaries using two additional parameters and fitting the optimal boundary location by rescaling the original integral, thereby increasing the order of accuracy in comparison to the classical Quadrature rule.

Hashemiparast et al. (2005a) implemented enhancements to the Gauss-Legendre, Gauss-Lobatto, and Gauss-Radan quadrature rules, Subsequently Similarly, Hashemiparast et al. (2006b) implemented the identical methodology on the first and second kinds (open type) of Chebyshev-Newton-Cotes quadrature rules. Burg (2012) introduced the concept of first and higher order derivatives at the evaluation locations within the closed Newton-Cotes quadrature to enhance precision and accuracy. In order to enhance the order of accuracy, he incorporated the first derivative at functional values and a few additional parameters. Later, Burg and Degny (2012) proposed a derivative-based midpoint quadrature rule. Zafar & Mir (2010) introduced several new families of open Newton-Cotes rules that entail the evaluation of derivatives at uniformly spaced points of the interval and the combination of function values. And Zhao & Li (2013) enhanced the precision and accuracy of classical Newton-Cotes quadrature rules by utilizing the derivative values at the midpoint.

Ramachandran et al. (2016a, 2016b, 2016c, 2016d) employed the midpoint derivative with the Geometric mean, Harmonic mean, Heronian mean, Centroidal mean, cotra-harmonic, and root mean square and expanded the order of precision and accuracy. Khatri et al. (2019) implemented an alternative methodology, compared the results with the classical rule and utilized the mean of two distinct measures (Arithmetic and Geometric) and increased the order of accuracy and precision of Closed Newton-Cotes Quadrature. In proposed research midpoint derivatives and the centroidal mean are used to improve both the accuracy and efficiency of the quadrature formula. The midpoint derivative captures local behavior more effectively, while the centroidal mean provides a balanced evaluation point, leading to reduced error and improved convergence without significantly increasing computational cost.

Newton Cotes Quadrature (CNC)

The Newton Cotes Formulae are the most prominent numerical integration formulas to obtain the approximate answer for the definite integral. Newton Cotes formulas have several sub-classes that depend upon the integer value of n. Some of them are as follows:

Classical Trapezoidal (MPT) Rule ($n = 1$)

$$\int_{\alpha}^{\beta} f(x)dx \cong \frac{(\beta-\alpha)}{2}\big(f(\alpha)+f(\beta)\big) - \frac{(\beta-\alpha)^3}{12} f''(\xi) \tag{4}$$

Where $\xi \in (\alpha, \beta)$. Degree of precision is 1 and the Order of accuracy is 3. Classical Simpson $\frac{1}{3}$ (MPS$\frac{1}{3}$) Rule (n = 2).

$$\int_{\alpha}^{\beta} f(x)dx \cong \frac{(\beta-\alpha)}{6}\left[f(\alpha) + 4f\left(\frac{\alpha+\beta}{2}\right) + f(\beta)\right] - \frac{(\beta-\alpha)^5}{2880} f^{iv}(\xi) \quad (5)$$

Where $\xi \in (\alpha, \beta)$. Degree of precision is 3 and Order of accuracy is 5. Classical Simpson $\frac{\mathbf{3}}{\mathbf{8}}$ (MPS$\frac{\mathbf{3}}{\mathbf{8}}$) Rule ($n$ = 3).

$$\int_{\alpha}^{\beta} f(x)dx \cong \frac{(\beta-\alpha)}{8}\begin{bmatrix} f(\alpha) + 3f\left(\frac{2\alpha+\beta}{3}\right) + \\ 3f\left(\frac{\alpha+2\beta}{3}\right) + f(\beta) \end{bmatrix} - \frac{(\beta-\alpha)^5}{6480} f^{iv}(\xi) \quad (6)$$

Where $\xi \in (\alpha, \beta)$. Degree of precision is 3 and order of accuracy is 5.

Midpoint Derivative Based Closed Newton Cotes Quadrature (MPCNC)

Theorem 1:

Efficient Modification of Midpoint Trapezoidal (EM-MPT) Rule is:

$$\int_{\alpha}^{\beta} f(x)dx \cong T = \frac{(\beta-\alpha)}{2}\left(f(\alpha) + f(\beta)\right) - \frac{(\beta-\alpha)^3}{12} f''\left(\frac{\alpha+\beta}{2}\right) - \frac{(\beta-\alpha)^5}{480} f^{iv}\left(\frac{2}{3}\left(\frac{\alpha^2+\alpha\beta+\beta^2}{\alpha+\beta}\right)\right) \quad (7)$$

with degree of precision 4.

*Proof**:*** Since the MPT Rule has degree of precision 3, the above formula has at least 3 degree of precision. Now, verify that the EM-MPT Rule is exact for $f(x) = x^4$. Now when $\int_{\alpha}^{\beta} x^4 dx = \frac{1}{5}(\beta^5 - \alpha^5)$.And put $f(x) = x^4$ in (7) we got

$$T = \frac{1}{2}(\beta-\alpha)(\alpha^4+\beta^4) - \frac{1}{4}(\beta-\alpha)^3(\alpha+\beta)^2 - \frac{1}{20}(\beta-\alpha)^5$$

$$T = \frac{4}{20}(\beta-\alpha)(\beta^4+\beta^3\alpha+\beta^2\alpha^2+\beta\alpha^3+\alpha^4)$$

$$T = \frac{1}{5}(\beta^5-\alpha^5)$$

Because the formula gives us exact result when $f(x) = x^4$. So, the precision of EM-MPT Rule is 4.

Theorem 2:

Efficient Modification of Midpoint Simpson$\frac{1}{3}$ (EM-MPS$\frac{1}{3}$) Rule is

$$\int_{\alpha}^{\beta} f(x)dx \cong S_{1/3} = \frac{(\beta-\alpha)}{6}\left[f(\alpha) + 4f\left(\frac{\alpha+\beta}{2}\right) + f(\beta)\right] - \frac{(\beta-\alpha)^5}{2880} f^{iv}\left(\frac{\alpha+\beta}{2}\right) - \frac{(\beta-\alpha)^7}{241920} f^{vi}\left(\frac{2}{3}\left(\frac{\alpha^2+\alpha\beta+\beta^2}{\alpha+\beta}\right)\right) \quad (8)$$

with degree of precision 6.

Proof: Since the MPS$\frac{1}{3}$ Rule has degree of precision 5, so the above Formula has at least 5 degree of precision. Now, just need to verify that the EM-MPS$\frac{1}{3}$ Rule is exact for $f(x) = x^6$. when $f(x) = x^6$, then $\int_\alpha^\beta x^6 dx = \frac{1}{7}(\beta^7 - \alpha^7)$.And put $f(x) = x^6$ in (8) we got

$$S_{1/3} = \frac{(\beta-\alpha)}{672}\begin{bmatrix} 7\{16\alpha^6 + (\alpha+\beta)^6 + 16\beta^6\} \\ -21(\beta-\alpha)^4(\alpha+\beta)^2 - 2(\beta-\alpha)^6 \end{bmatrix}$$

$$S_{1/3} = \frac{96(\beta-\alpha)}{672}\begin{bmatrix} \beta^6 + \beta^5\alpha + \beta^4\alpha^2 + \beta^3\alpha^3 + \\ \beta^2\alpha^4 + \beta\alpha^5 + \alpha^6 \end{bmatrix}$$

$$S_{1/3} = \frac{(\beta-\alpha)}{7}\begin{bmatrix} \beta^6 + \beta^5\alpha + \beta^4\alpha^2 + \beta^3\alpha^3 + \\ \beta^2\alpha^4 + \beta\alpha^5 + \alpha^6 \end{bmatrix}$$

$$S_{1/3} = \frac{1}{7}(\beta^7 - \alpha^7)$$

Because the formula gives us exact result when $f(x) = x^6$ .So, the precision of EM-MPS$\frac{1}{3}$ Rule is 6.

Theorem 3

Efficient Modification of Midpoint Simpson $\frac{3}{8}$(EM-MPS$\frac{3}{8}$)Rule is

$$\int_\alpha^\beta f(x)dx \cong S_{3/8} = \frac{\beta-\alpha}{8}\begin{bmatrix} f(\alpha) + 3f\left(\frac{2\alpha+\beta}{3}\right) + \\ 3f\left(\frac{\alpha+2\beta}{3}\right) + f(\beta) \end{bmatrix} - \frac{(\beta-\alpha)^5}{6480} f^{iv}\left(\frac{\alpha+\beta}{2}\right) - \frac{23(\beta-\alpha)^7}{9797760} f^{vi}\left(\frac{2}{3}\left(\frac{\alpha^2+\alpha\beta+\beta^2}{\alpha+\beta}\right)\right) \quad (9)$$

with degree of Precision 6.

Proof: Since the MPS$\frac{3}{8}$ Rule has degree of precision 5, so the above formula has at least 5 degree of precision. Now, it needs to verify that the EM-MPS $\frac{3}{8}$ Rule is exact for $f(x) = x^6$.

When $f(x) = x^6$, $\int_\alpha^\beta x^6 dx = \frac{1}{7}(\beta^7 - \alpha^7)$

$$S_{3/8} = \frac{(\beta-\alpha)}{13608}\begin{bmatrix} 7\begin{Bmatrix} 243\alpha^6 + (2\alpha+\beta)^6 + \\ (\alpha+2\beta)^6 + 243\beta^6 \end{Bmatrix} - \\ 189(\beta-\alpha)^4(\alpha+\beta)^2 - 23(\beta-\alpha)^6 \end{bmatrix}$$

$$S_{3/8} = \frac{(\beta-\alpha)}{13608}\begin{pmatrix} 1944\beta^6 + 1944\beta^5\alpha + 1944\beta^4\alpha^2 + \\ 1944\beta^3\alpha^3 + 189\beta^2\alpha^4 + \\ 1944\beta\alpha^5 + 1944\alpha^6 \end{pmatrix}$$

$$S_{3/8} = \frac{1944(\beta-\alpha)}{13608}\begin{pmatrix} \beta^6 + \beta^5\alpha + \beta^4\alpha^2 + \\ \beta^3\alpha^3 + \beta^2\alpha^4 + \beta\alpha^5 + \alpha^6 \end{pmatrix}$$

$$S_{3/8} = \frac{1}{7}(\beta^7 - \alpha^7).$$

Because the formula gives us exact result when $f(x) = x^6$ .So, the precision of EM-MPS$\frac{3}{8}$ Rule is 6.

The Error Terms of Efficient Modification of Derivative Based Midpoint Closed Newton Cotes Quadrature

The error term can be given in mainly three different ways, here the concept of precision to calculate the error terms. The error of the integration is given by

$$R_n = \frac{C}{(m+1)!} f^{m+1}(\xi) \qquad \alpha < \xi < \beta \tag{10}$$

Here m is the degree of precision and

$$C = \int_\alpha^\beta x^{m+1} dx - \sum w_i f(x_i) \tag{11}$$

Theorem 4

EM-MPT Rule (7) has error term $R_1 = \frac{(\beta-\alpha)^7}{2880(\beta+\alpha)} f^5(\xi)$ Where $\alpha < \xi < \beta$ and this scheme is seventh-order accurate.

Proof: The degree of precision of EM-MPT Rule is 4, so by putting $m = 4$ in (10).

Got $R_1 = \frac{C_1}{5!} f^5(\xi)$.Now For C_1 when put $f(x) = x^5$ in (11), after simplification.

$$C_1 = \frac{\beta^6-\alpha^6}{6} - \frac{(\beta-\alpha)}{24(\alpha+\beta)} \begin{pmatrix} 3\beta^6 + 14\beta^5\alpha - 7\beta^4\alpha^2 + \\ 28\beta^3\alpha^3 - 7\beta^2\alpha^4 + 14\beta^5\alpha + 3\alpha^6 \end{pmatrix}$$

$$C_1 = \frac{(\beta-\alpha)}{24(\alpha+\beta)} \begin{pmatrix} \beta^6 - 6\beta^5\alpha + 15\beta^4\alpha^2 - \\ 20\beta^3\alpha^3 + 15\beta^2\alpha^4 - 6\beta\alpha^5 + \alpha^6 \end{pmatrix}$$

$$C_1 = \frac{(\beta-\alpha)^7}{24(\alpha+\beta)}$$

This implies that the method is seventh-order accurate, and the error term is $R_1[f] = \frac{(\beta-\alpha)^7}{24(\alpha+\beta)\times 5!} f^5(\xi) = \frac{(\beta-\alpha)^7}{2880(\alpha+\beta)} f^5(\xi)$

Theorem 5

EM-MPS$\frac{1}{3}$ Rule (8) has error term $R_2 = \frac{(\beta-\alpha)^9}{1451520(\beta+\alpha)} f^7(\xi)$ Where $\alpha < \xi < \beta$ and the scheme is ninth-order accurate.

Proof: The degree of precision of EM-MPS$\frac{1}{3}$ Rule is 6, so by putting m = 6 in (10) got $R_2 = \frac{C_2}{7!} f^7(\xi)$. Now for C_2 put $f(x) = x^7$ in (11) after simplification.

$$C_2 = \frac{\beta^8-\alpha^8}{8} - \frac{(\beta-\alpha)}{576(\alpha+\beta)}(70\beta^8 + 160\beta^7\alpha + 88\beta^6\alpha^2 + 256\beta^5\alpha^3 + 4\beta^4\alpha^4 + 256\beta^3\alpha^5 + 88\beta^2\alpha^6 + 160\beta\alpha^7 + 70\alpha^8)$$

$$C_2 = \frac{(\beta-\alpha)}{576(\alpha+\beta)}\left[72(\alpha+\beta)\begin{pmatrix}\beta^7 + \beta^6\alpha + \beta^5\alpha^2 + \beta^4\alpha^3 + \\ \beta^3\alpha^4 + \beta^2\alpha^5 + \beta\alpha^6 + \alpha^7\end{pmatrix} - \begin{pmatrix}70\beta^8 + 160\beta^7\alpha + 88\beta^6\alpha^2 + 256\beta^5\alpha^3 + \\ 4\beta^4\alpha^4 + 256\beta^3\alpha^5 + 88\beta^2\alpha^6 + 160\beta\alpha^7 + 70\alpha^8\end{pmatrix}\right]$$

$$C_2 = \frac{(\beta-\alpha)^9}{288(\alpha+\beta)}$$

This implies that the method is ninth order accurate, and the error term is $R_2[f] = \frac{(\beta-\alpha)^9}{288\times 7!(\alpha+\beta)} f^7(\xi) = \frac{(\beta-\alpha)^9}{1451520(\alpha+\beta)} f^7(\xi)$

Theorem 6

EM-MPS$\frac{3}{8}$ Rule (9) has error term $R_3 = \frac{(\beta-\alpha)^9}{7!\times 11664(\beta+\alpha)} f^7(\xi)$ Where $\alpha < \xi < \beta$ and the scheme is ninth-order accurate.

Proof: The degree of precision of EM-MPS$\frac{3}{8}$ Rule is 6, so by putting m = 6 in (10) We got $R_3 = \frac{C_3}{7!} f^7(\xi)$.Now for C_3 put $f(x) = x^7$ in (11) and after simplification

$$C_3 = \frac{\beta^8-\alpha^8}{8} = \frac{\beta-\alpha}{11664(\beta-\alpha)}\begin{pmatrix}1435\beta^8 + 3100\beta^7\alpha + 2272\beta^6\alpha^2 + \\ 4204\beta^5\alpha^3 + 1306\beta^4\alpha^4 + 4204\beta^3\alpha^5 + \\ 2272\beta^2\alpha^6 + 3100\beta\alpha^7 + 1435\alpha^8\end{pmatrix}$$

$$C_3 = \frac{(\beta-\alpha)}{11664(\alpha+\beta)}\begin{bmatrix}1458(\alpha+\beta)\begin{pmatrix}\beta^7 + \beta^6\alpha + \beta^5\alpha^2 + \beta^4\alpha^3 + \\ \beta^3\alpha^4 + \beta^2\alpha^5 + \beta\alpha^6 + \alpha^7\end{pmatrix} - \\ \begin{pmatrix}1435\beta^8 + 3100\beta^7\alpha + 2272\beta^6\alpha^2 + \\ 4204\beta^5\alpha^3 + 1306\beta^4\alpha^4 + 4204\beta^3\alpha^5 + \\ 2272\beta^2\alpha^6 + 3100\beta\alpha^7 + 1435\alpha^8\end{pmatrix}\end{bmatrix}$$

$$C_3 = \frac{(\beta-\alpha)^9}{11664(\alpha+\beta)}$$

This implies that the method is ninth-order accurate, and the error term is $R_3[f] = \frac{(\beta-\alpha)^9}{7!\times 11664(\alpha+\beta)} f^7(\xi)$.

Table 1 summarizes the precision, order and the error terms for CNC, MPCNC and EM-MPCNC respectively.

Table 1: Precision, order, and the error terms for CNC, MPCNC and EM-MPCNC respectively.

	Classical CNC			MPCNC			EM-MPCNC		
Methods	Precision	Order	Error	Precision	Order	Error	Precision	Order	Error
Trapezoidal	1	3	$-\frac{(\beta-\alpha)^3}{12} f''(\xi)$	3	5	$-\frac{(\beta-\alpha)^5}{480} f^{iv}(\xi)$	4	7	$\frac{(\beta-\alpha)^7 f^5(\xi)}{2880(\alpha+\beta)}$
Simpson $\frac{1}{3}$	3	5	$-\frac{(\beta-\alpha)^5}{2880} f^{iv}(\xi)$	5	7	$-\frac{(\beta-\alpha)^7}{241920} f^{vi}(\xi)$	6	9	$\frac{(\beta-\alpha)^9 f^7(\xi)}{1451520(\alpha+\beta)}$
Simpson $\frac{3}{8}$	3	5	$-\frac{(\beta-\alpha)^5}{6480} f^{iv}(\xi)$	5	7	$-\frac{23(\beta-\alpha)^7}{9797760} f^{vi}($	6	9	$\frac{(\beta-\alpha)^9 f^7(\xi)}{7!\times 11664(\alpha +}$

Computational Efficiency

Numerical experiments are conducted to compare the CPU time of MPCNC, and EM-MPCNC. Figure 1 and Tables 2 to 4 show the comparison of the CPU time for the same level of accuracy of 10^{-17} using two integrals: $\int_0^1 \left(\frac{1}{1+x}\right) dx$ and $\int_0^1 e^{-x^2}\, dx$ and for level of accuracy of 10^{-20} using integral $\int_0^1 3^x dx$.

The required number of function and derivative evaluations is computed in Tables 5, 6, 7 and 8, to achieve a certain level of accuracy of 10^{-10}, 10^{-8}, 10^{-5}, and 10^{-8} for the following integrals $\int_0^1 \left(\frac{dx}{1+x}\right)$, $\int_0^1 3^x dx$, $\int_0^2 e^x dx$, and $\int_0^{\frac{\pi}{2}} \cos x \;\, dx$ respectively, are tested in order to calculate the efficiency of (EM-MPCNC) with classical CNC and MPCNC.

To achieve a level of accuracy of 10^{-10} the classical Trapezoidal rule necessitates 25002 function evaluations, while MPT necessitates 106 Func. Eval and 105 Mid. derivatives (total = 211). EM-MPT necessitates 21 Func. Eval and 41 Mid. derivatives (total = 61) in order to solve the integral $\int_0^1 (dx/(1+x))$. This implies that EM-MPT requires a lower number of calculations than the other two methods.

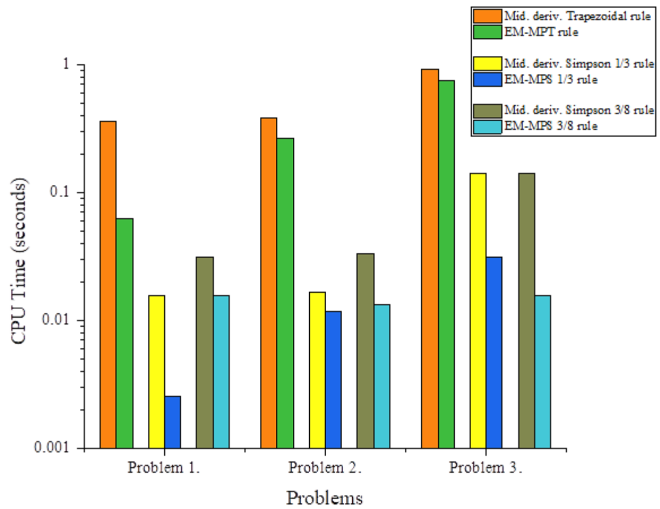

Figure 1: Graphical representation of CPU time (seconds) from Tables 2–4 for Problems 1–3.

Table 2: CPU time for problem 1 for accuracy 10^{-17}.

Formula	CPU Time (seconds)
MP Trapezoidal rule	0.3593552
MPS 1/3 rule	0.0156235
MPS 3/8 rule	0.0312448
EM-MPT rule	0.0624974
EM-MPS 1/3 rule	0.002538
EM-MPS 3/8 rule	0.0156241

Table 3: CPU time for problem 2 for accuracy 10^{-17}.

Formula	CPU Time (seconds)
MP Trapezoidal rule	0.3837622
MPS 1/3 rule	0.0166243
MPS 3/8 rule	0.0332496
EM-MPT rule	0.2656133
EM-MPS 1/3 rule	0.0116221
EM-MPS 3/8 rule	0.0132881

Table 4: CPU time for problem 3 for accuracy 10^{-20}.

Formula	CPU Time (seconds)
MP Trapezoidal rule	0.9196397
MPS 1/3 rule	0.1406182
MPS 3/8 rule	0.1406197
EM-MPT rule	0.7499646
EM-MPS 1/3 rule	0.0312467
EM-MPS 3/8 rule	0.0156219

Table 5: Computational cost comparison for Problem 1 using various quadrature rules for accuracy 10^{-10}.

Formula	Order	Sub intervals	Func. eval	Mid. deriv.	Total deriv.	Total
Trapezoidal rule	2	25001	25002	0	0	25002
Simpson 1/3 rule	4	67	135	0	0	135
Simpson 3/8 rule	4	55	166	0	0	166
MP Trapezoidal rule	4	105	106	105	105	211
MPS 1/3 rule	6	14	29	14	14	43
MPS 3/8 rule	6	12	37	12	12	49
EM-MPT rule	7	20	21	20	40	61
EM-MPS 1/3 rule	9	7	15	7	14	29
EM-MPS 3/8 rule	9	6	19	6	12	31

Table 6: Computational cost comparison for Problem 3 using various quadrature rules for accuracy 10^{-8}.

Formula	Order	Sub intervals	Func. eval	Mid. deriv.	Total deriv.	Total
Trapezoidal rule	2	4350	4351	0	0	4351
Simpson 1/3	4	18	37	0	0	37
Simpson 3/8	4	15	46	0	0	46
MP Trapezoidal rule	4	28	29	28	56	85
MPS 1/3 rule	6	4	9	4	8	17
MPS 3/8 rule	6	4	13	4	8	21
EM-MPT rule	7	8	9	8	16	25
EM-MPS 1/3 rule	9	3	7	3	6	13
EM-MPS 3/8 rule	9	2	7	2	4	11

Table 7: Computational cost comparison for Problem 4 using various quadrature rules for accuracy 10^{-5}.

Formula	Order	Sub intervals	Func. eval	Mid. deriv.	Total deriv.	Total
Trapezoidal rule	2	462	463	0	0	463
Simpson 1/3 rule	4	8	17	0	0	17
Simpson 3/8 rule	4	7	22	0	0	22
MP Trapezoidal rule	4	13	14	13	13	27
MPS 1/3 rule	6	3	7	3	3	10
MPS 3/8 rule	6	3	10	3	3	13
EM-MPT rule	7	3	4	4	8	12
EM-MPS 1/3 rule	9	2	5	2	4	9
EM-MPS 3/8 rule	9	2	7	2	4	11

Table 8: Computational cost comparison for Problem 5 using various quadrature rules for accuracy 10^{-8}.

Formula	Order	Sub intervals	Func. eval	Mid. deriv.	Total deriv.	Total
Trapezoidal rule	2	4663	4664	0	0	4664
Simpson 1/3 rule	4	22	45	0	0	45
Simpson 3/8 rule	4	18	55	0	0	55
MP Trapezoidal rule	4	34	35	34	34	69
MPS 1/3 rule	6	5	11	5	5	16
MPS 3/8 rule	6	4	13	4	4	17
EM-MPT rule	7	9	10	9	18	28
EM-MPS 1/3 rule	9	3	7	3	6	13
EM-MPS 3/8 rule	9	3	10	3	6	16

Numerical Experiments

This section analyzes various numerical examples from the literature and compares their approximated results and errors to demonstrate that the EM-MPCNC displays improved precision, reduced error and takes less CPU time (seconds), and midpoint derivatives are calculated numerically. The results acquired are further illustrated by a series of bar graphs that illustrate the comparison between the proposed and the other methods. Tables 9, 10, 11, 12, 13, and 14 display the comparison results of the integrals $\int_0^1 3^x\,dx$ and $\int_0^2 e^x\,dx$ respectively. The corresponding graph is located in Figure 2.

Problem 1. $\int_0^1 (dx/(1+x))$. (Zhao & Li, 2013)

Problem 2. $\int_0^1 e^{-x^2} dx$. (Ramachandran et al., 2017)

Problem 3. $\int_0^1 3^x dx$. (Ramachandran et al., 2016c)

Problem 4. $\int_0^2 e^x dx$ (Zhao & Li, 2013; Burg & Degny, 2012)

Problem 5. $\int_0^{\frac{\pi}{2}} \cos x\ dx$ (Ramachandran et al., 2016b)

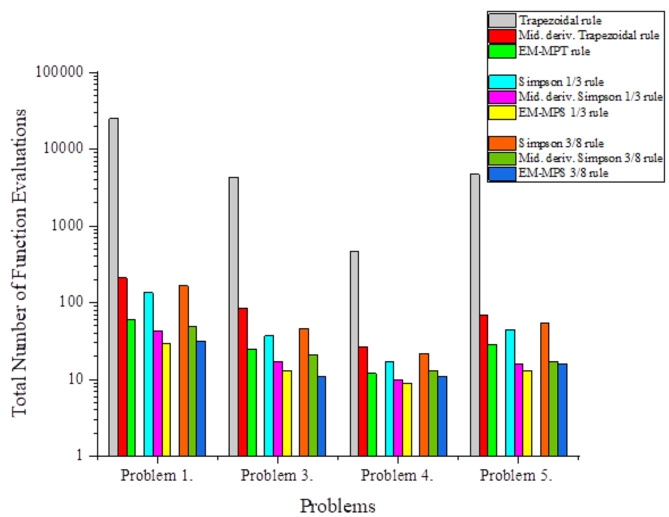

Figure 2: Graphical Representation of the total number of function evaluations used for problems 1 & 3-4.

Table 9: Comparison table of Approx: value and error term of problem 3 CST, MPT, and EM-MPT.

	CST		MPT		EM-MPT	
Sub-interval	Approx. value	Error	Approx. value	Error	Approx. value	Error
N=1	2	0.17952	1.82579	5.313e-03	1.81947	9.9926e-04
N=2	1.86602	0.04554	1.82082	3.4192e-04	1.82046	1.775e-05
N=3	1.84077	0.02029	1.82054	6.791e-05	1.82047	1.690e-06

Table 10: Comparison table of Approx: value and error term of problem 3 CS1/3, MPS1/3, and EM-MPS1/3.

	CS1/3		MPS1/3		EM-MPS1/3	
Sub-interval	Approx. value	Error	Approx. value	Error	Approx. value	Error
N=1	1.82136	0.00088	1.82049	1.267e-05	1.82048	7.628e-06
N=2	1.82053	5.704e-05	1.82047	2.045e-07	1.82047	7.015e-08
N=3	1.82048	1.132e-05	1.82047	1.806e-08	1.82047	4.366e-09

Table 11: Comparison table of Approx: value and error term of problem 3 CS3/8, MPS3/8, and EM-MPS3/8.

	CS3/8		MPS3/8		EM-MPS3/8	
Sub-interval	Approx. value	Error	Approx. value	Error	Approx. value	Error
N=1	1.82087	0.00039	1.82048	7.198e-06	1.82048	3.965e-06
N=2	1.82050	2.538e-05	1.82047	1.161e-07	1.82047	3.633e-08
N=3	1.82048	5.035e-06	1.82047	1.026e-08	1.82047	2.247e-09

Table 12: Comparison table of Approx: value and error term of problem 4 CST, MPT, and EM-MPT.

	CT		MPT		EM-MPT	
Sub-interval	Approx. value	Error	Approx. value	Error	Approx. value	Error
N=1	8.38905	2	6.57686	1.878e-01	6.32395	6.510e-2
N=2	6.91280	0.52375	6.40194	1.288e-02	6.38801	1.041e-03
N=3	6.62395	0.23489	6.39164	2.591e-03	6.38896	9.551e-05

Table 13: Comparison table of Approx: value and error term of problem 4 CS1/3, MPS1/3, and EM-MPS1/3.

	CS1/3		MPS1/3		EM-MPS1/3	
Sub-interval	Approx. value	Error	Approx. value	Error	Approx. value	Error
N=1	6.42072	3.167e-2	6.39052	1.469e-3	6.39025	1.197e-3
N=2	6.39121	2.154e-3	6.38908	2.547e-05	6.38906	1.351e-05
N=3	6.38948	4.325e-4	6.38905	2.282e-06	6.38905	8.983e-07

Table 14: Comparison table of Approx: value and error term of problem 4. CS3/8, MPS3/8, and EM-MPS3/8.

	CS3/8		MPS3/8		EM-MPS3/8	
Sub-interval	Approx. value	Error	Approx. value	Error	Approx. value	Error
N=1	6.40331	0.01425	6.38989	8.358e-04	6.38969	6.431e-04
N=2	6.39001	0.00096	6.38907	1.447e-05	6.38906	7.132e-06
N=3	6.38924	0.00019	6.38905	1.296e-06	6.38905	4.682e-07

Conclusion

Introduction of an Efficient Modification of the Closed Newton-Cotes Quadrature Rules by adding one more term in midpoint derivative CNC formulae Zhao and li in which centroidal mean has been used at derivative value, this modification increased the degree of precision by three in comparison to the Classical CNC Rules and by one in comparison to the MPCNC Rules. The error terms were derived using the precision method, and the order of accuracy analysis demonstrated that the proposed modification is computationally superior and robust to both classical and midpoint derivative-based rules in terms of precision, error and CPU time (seconds). In comparison, Tables 2–14 and Figures 1–2 present numerical experiments that demonstrate the efficiency and effectiveness of the proposed scheme, showing that it outperforms others in terms of precision, error, and CPU time (seconds) across all test problems. Future research in this field could focus on creating quadrature rules that employ the same technique as open, semi-open, and semi-CNC quadrature rules.

References

Acu, A. M., Acu, M., & Rafiq, A. (2008). The estimates for the remainder term of some quadrature formulas. *Acta Universitatis Apulensis, 16*, 53–62.

Atkinson, K. E. (1978). An introduction to numerical analysis. New York: John Wiley & Sons.

Atkinson, K. E. (1989). An Introduction to Numerical Analysis. 2nd Edition, John Wiley & Sons, Hoboken.

Burg, C. O. (2012). Derivative-based closed Newton–Cotes numerical quadrature. *Applied Mathematics and Computation, 218*(13), 7052–7065.

Burg, C. O., & Degny, E. (2012). Derivative-based midpoint quadrature rule. *Applied Mathematics, 4*(1), 228.

Dehghan, M., Jamei, M. M., & Eslahchi, M. R. (2005a). On numerical improvement of closed Newton–Cotes quadrature rules. *Applied Mathematics and Computation, 165*(2), 251–260.

Dehghan, M., Jamei, M. M., & Eslahchi, M. R. (2005b). On numerical improvement of open Newton–Cotes quadrature rules. *Applied Mathematics and Computation, 175*(1), 618–627.

Dehghan, M., Masjed-Jamei, M., & Eslahchi, M. R. (2005c). The semi-open Newton–Cotes quadrature rule and its numerical improvement. *Applied Mathematics and Computation, 171*, 1129–1140.

Hashemiparast, S. M., Eslahchi, M. R., Dehghan, M., & Masjed-Jamei, M. (2006a). The first kind Chebyshev–Newton–Cotes quadrature rules (semi-open type) and its numerical improvement. *Applied Mathematics and Computation, 174*(2), 1020–1032.

Hashemiparast, S. M., Masjed-Jamei, M., Eslahchi, M. R., & Dehghan, M. (2006b). On numerical improvement of the second kind Chebyshev-Newton-Cotes quadrature rules (open type). *Applied Mathematics and Computation, 180*(2), 605–613.

Khatri, A., Shaikh, A. A., & Abro, K. A. (2019). Closed Newton–Cotes quadrature rules with derivatives. *Mathematical Theory and Modeling, 9*(5), 65–72.

Ramachandran, T., Udayakumar, D., & Parimala, R. (2016a). Comparison of arithmetic mean, geometric mean, and harmonic mean derivative-based closed Newton–Cotes quadrature. *Progress in Nonlinear Dynamics and Chaos, 4*, 35–43.

Ramachandran, T., Udayakumar, D., & Parimala, R. (2016b). Harmonic mean derivative-based closed Newton–Cotes quadrature. *IOSR Journal of Mathematics*, 12, 36–41.

Ramachandran, T., Udayakumar, D., & Parimala, R. (2016c). Centroidal mean derivative-based closed Newton–Cotes quadrature. *International Journal of Science and Research*, 5, 338–343.

Ramachandran, T., Udayakumar, D., & Parimala, R. (2016d). Heronian mean derivative-based closed Newton–Cotes quadrature. *International Journal of Mathematics Archive*, 7(7).

Ramachandran, T., Udayakumar, D., & Parimala, R. (2017). Contra-harmonic mean derivative-based closed Newton–Cotes quadrature. *Global Journal of Pure and Applied Mathematics, 13*(5), 1319–1330.

Saand, M. S. K, Kalhoro, Z. A., Shaikh, A. W., Rajput, O. R., & Rind, S. (2022). Improved mid-point derivative-based closed Newton-Cotes quadrature rule. *VFAST Transactions on Mathematics*, 10(2): 213–229.

Ujevic, N., & Roberts, A. J. (2004). A corrected quadrature formula and applications. *ANZIAM Journal, 45*(E), E41–E56.

Zafar, F., & Mir, N. A. (2010). A generalized family of quadrature-based iterative methods. *General Mathematics, 18*(4), 43–51.

Zhao, W., & Li, H. (2013). Midpoint derivative-based closed Newton-Cotes quadrature. *Abstract and Applied Analysis, 2013*.