Approximation of Integrals with 2D and 3D Surfaces of Absolute Errors using Various Quadrature Methods

Suraj Powar* and Maaz Mahmood Siddiqui**

under supervision of Prof. Dr. Henri Schurz*

*Department of Mathematics, SIU, Carbondale, IL, 62901, **Department of Biomedical Engineering, SIU, Carbondale, IL, 62901.

Final project submission for credits towards the course Math 572: Advanced Topics in Numerical Analysis: Numerical Methods & Their Analysis

Southern Illinois University

July 5, 2024

Abstract

Error estimates are immensely crucial in qualitative evaluation of Numerical Integration. The estimation of integrals using numerical quadrature is a very popular technique. In this study, we estimated an integrable function, using methods like Left Riemann Method, Midpoint Riemann Method, Right Hand Riemann Method, Trapezoidal Method, Simpsons Method, Boole's Method and lastly Weddle's Method. The smallest error is achieved by Simpson Method if we only compare the Riemann, Trapezoidal and Simpson Quadratures. The largest errors occurs with Left hand and Right Hand Riemann Sums, whereas the error of Trapezoidal was better than LHRS and RHRS. After implementation of Boole's and Weddle's method, It was evident from our study that Boole's and Weddle's Method were most efficient for error estimation.

Contents

※ Introduction

1.1 Literature Review

Let us suppose we have a function $f \in L^1([a,b])$ in the space of absolutely integrable function with respect to Lebesgue measure. Now consider approximation methods of integrals,

$$I_a^b(f) := \int_a^b f(x)dx$$

This integral can be evaluated using the classical calculus methods. Our study revolves around various methods approximate this integral such that the error is minimized in an ideal setting. The way to address this problem is by making using of Quadrature Methods to eventually figure out the quadrature errors.

Definition 1.1. Let $f : [a,b] \to \mathbb{R}^d$ be an integrable, real-valued function and

$$a \leq x_{1,n} < x_{2,n} ... << x_{k,n} < ... < x_{n,n} = b$$

be a partition of given [a, b]. Then

$$Q_a^b(f,n) := \sum_{k=1}^{n} w_{k,n} f(x_{k,n})$$

is called **quadrature methods** (also **numerical integral of f**) with **weights** $w_{k,n}$, **nodes** $x_{k,n}$ (also **instants, nodal and sample points**), where $n \in \mathbb{N}$ is the number of instances. The related **(quadrature)** error on [a, b] is defined by,

$$\epsilon_n(f,a,b) := \left\| I_a^b(f) - Q_a^b(f,n) \right\|_d$$

The quadrature method $Q_a^b(f,n)$ is said to be **convergent to** $I_a^b(f)$ iff

$$\lim_{n \to +\infty} \epsilon_n(f,a,b) = 0$$

The quadrature method $Q_a^b(f,n)$ is called **convergent to** $I_a^b(f)$ **with order (rate) r $>$ 0 with respect to function class** F iff

$$\forall f \in F \exists\ c = c(f,a,b) \geq 0, \forall n > 0$$

$$\epsilon_n(f,a,b) \leq \frac{c(f,a,b)}{n^r}$$

In our study we will investigate the different quadrature methods as suggested in section of Methodology and check the convergence and error rate based on the theorems discussed in the Math 572 class [1]. The detailed methods will be discussed in the section of Methodology. The Results section will comprise of 3 dimensional error plots the function with different methods. In the Conclusion section we will articulate the theorems and output that we generated in form of error plots. The simulations were executed using python3 with the help of libraries numpy and matplotlib [4]. The architecture of machine used was Macbook Air M1, with 8 cores and 16 GB RAM, LP-DDR4X memory running at 3733MHz integrated to the SoC of M1 ARM based chip.

※ Methodology

2.1 Quadrature Methods

In this section we will discuss the methods we used to estimate error in detail. Let us begin by considering a **Left Hand Riemann Quadrature Methods (LRMs)**

$$Q_a^b(f,n) := \sum_{k=1}^{n} x_{k-1}^p \cdot \frac{b-a}{n}$$

along partitions

$$a = x_0 < x_1 < ... < x_k < ... < x_n = b$$

of [a, b] with $\Delta x_k = x_k - x_{k-1}$.
Similarly, we can also have **Right Hand Riemann Quadrature Methods (RRMs)** and **Midpoint Riemann Quadrature Methods (MPRMs)**.
The quadrature of right hand riemann quadrature and midpoint riemann quadrature can be give as follows,

$$Q_a^b(f,n) := \sum_{k=1}^{n} x_k^p \cdot \frac{b-a}{n}$$

$$Q_a^b(f,n) := \sum_{k=1}^{n} \frac{(x_k + x_{k-1})^p}{2} \cdot \frac{b-a}{n}$$

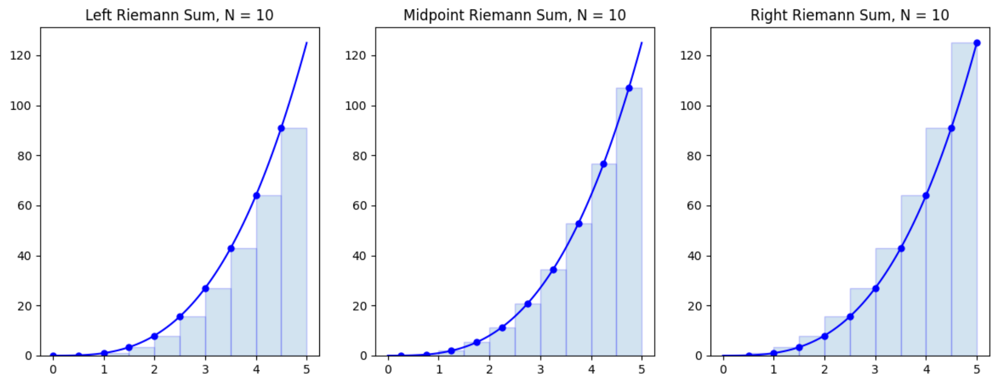

Figure 1: 2D plot of an approximation of a function

Apart from the Riemann Quadrature we also explored the possibility of inclusion of **Trapezoidal Method** and the **Simpsons Method** for our investigation.

The **Trapezoidal Method** is essentially also called as **Outer Midpoint Methods** because it obeys the scheme,

$$TM_a^b(f,n) = \sum_{k=1}^{n} \frac{x_k^p + x_{k-1}^p}{2} \cdot \frac{b-a}{n}$$

whereas **Inner Theta Methods (ITM)** with implicitness constants $\theta \in [0,\ 1]$ are governed by,

$$Q_a^b(f,n) = \sum_{k=1}^{n} f(\theta x_k + (1-\theta)x_{k-1}) \cdot \Delta x_k$$

The scheme we will be using for our quadrature method study is as follows:

$$T_a^b(f,n) := \sum_{k=1}^{n} \frac{f(x_k) + f(x_{k-1})}{2} \cdot \Delta x_k \textbf{ as Trapezoidal Method,}$$

$$S_a^b(f,n) := \sum_{k=1}^{n} \frac{f(x_{k-1}) + 4f(\bar{x_k}) + f(x_k)}{6} \cdot \Delta x_k \textbf{ as Simpsons Method,}$$

In addition to these schemes, we implemented two more schemes: A quintic-polynomially exact method (inner form), **as Boole's Method,**

$$B_a^b(f,n) := \sum_{k=1}^{n} \left[\frac{7}{90} f(x_{k-1}) + \frac{32}{90} f(x_{k-1} + \frac{\Delta x_k}{4}) + \frac{12}{90} f(\bar{x_k}) + \frac{32}{90} f(x_{k-1} + 3\frac{\Delta x_k}{4}) + \frac{7}{90} f(x_k) \right] \cdot \Delta x_k$$

And a Newton-Cotes Method with m = 6 which obeys that globally implemented scheme as,

$$W_a^b(f,n) := \sum_{k=1}^{n} \left[\frac{41}{840} f(x_{k-1}) + \frac{216}{840} f(x_{k-1} + \frac{\Delta x_k}{6}) + \frac{27}{840} f(x_{k-1} + \frac{\Delta x_k}{3}) + \right.$$

$$\frac{272}{840}f(x_{k-1}+\frac{\Delta x_k}{2})+\frac{27}{840}f(x_{k-1}+2\frac{\Delta x_k}{3})+\frac{216}{840}f(x_{k-1}+5\frac{\Delta x_k}{6})+\frac{41}{840}f(x_k)$$

as Weddle's Method.

2.2 Experiment

In our study we are interested to investigate a continuous function with different quadrature methods. Let us take any real positive constant b. Consider $x \in [0,b] \longmapsto$ f(x) = x^p. The related absolute error is given as

$$\epsilon_n(b,p) := \left| \int_0^b x^p dx - Q_0^b(f,n) \right|$$

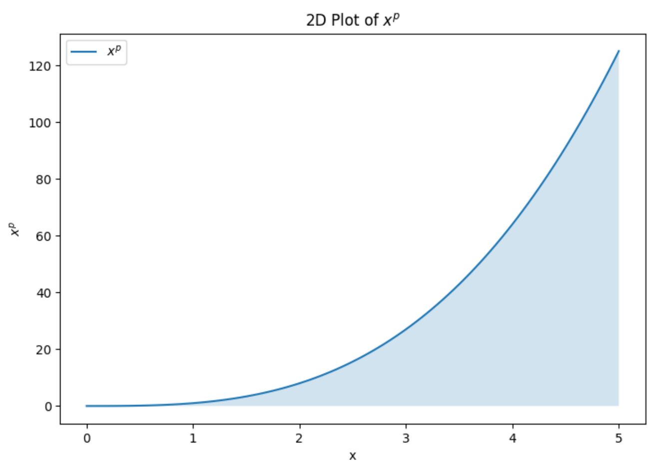

Figure 2: 2D plot of the function x^p

Here $Q_0^b(f,n)$ is the left hand Riemann quadrature (LRM) of $f(x)=x^p$ on [0, b]. If we were to use calculus to compute the integral of the function, we know that,

$$\int_0^b x^p dx = \frac{b^{p+1}}{p+1}$$

With this we can compute the exact value of the integral. For example if b = 5, p = 0.5 and n=10, the integral value is 7.4535599249993005.

The image of the 3D plot for above values, can be given as follows,

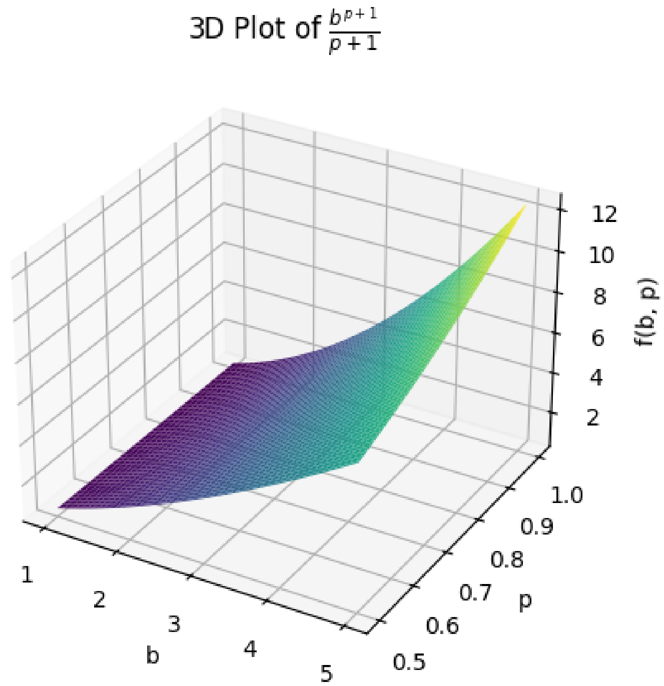

Figure 3: 3D plot of the function $\frac{b^{p+1}}{p+1}$

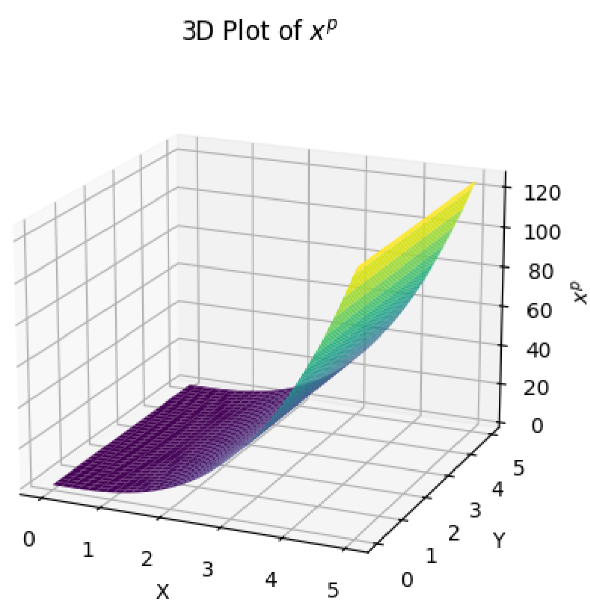

Figure 4: 3D plot of the function x^p

We will try to construct different test cases for plotting the three dimensional surface of absolute error $\epsilon_n(b, p)$ for

- fixed p = 0.5, 1, 2 versus n $\in \mathbb{N}$ and b $\in \mathbb{R}_+$,
- fixed b = 1, 10, 100, versus n $\in \mathbb{N}$ and $0 < p \leq 3$,
- fixed n = 100, versus b > 0 and p > 0.

※ Results

In this section, we will explore and produce some surface error plots based on the different values of p. b and n.

3.1 Experiment 1

For our first experiment, we will fix p = 0.5, n = 10 and b = 5 and below we have the error surface of the following,

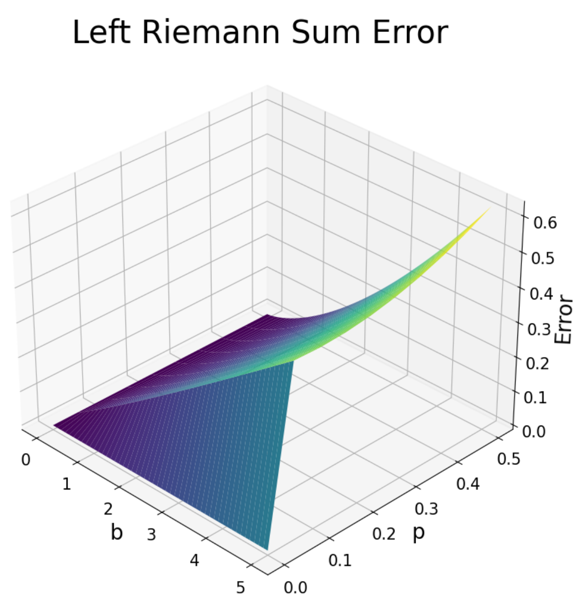

Figure 5

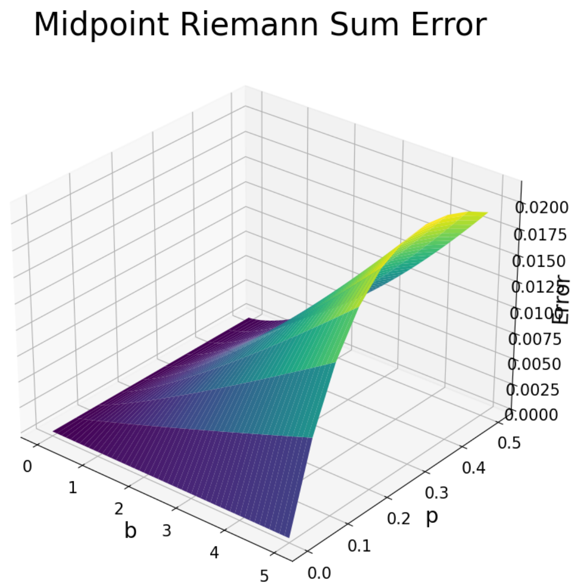

Figure 6

3.2 Experiment 2

We will now fix p = 1, n = 10 and b = 5 and view the error surface of the following:

3.3 Experiment 3

We will fix p = 2, n = 10 and b = 5 and see the error surface of the following:

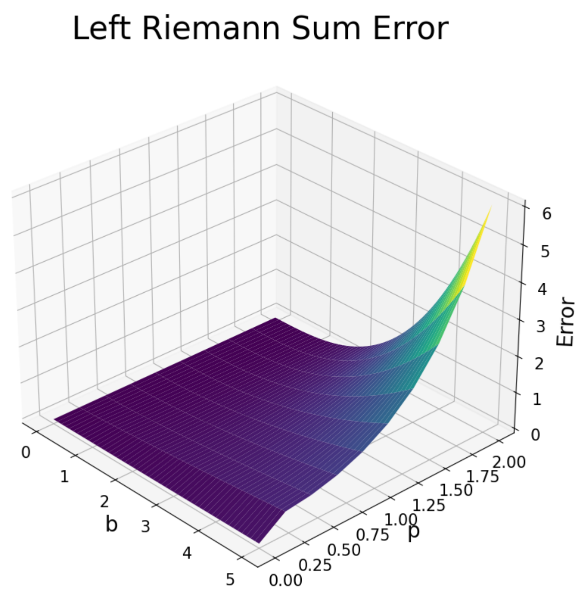

Figure 7

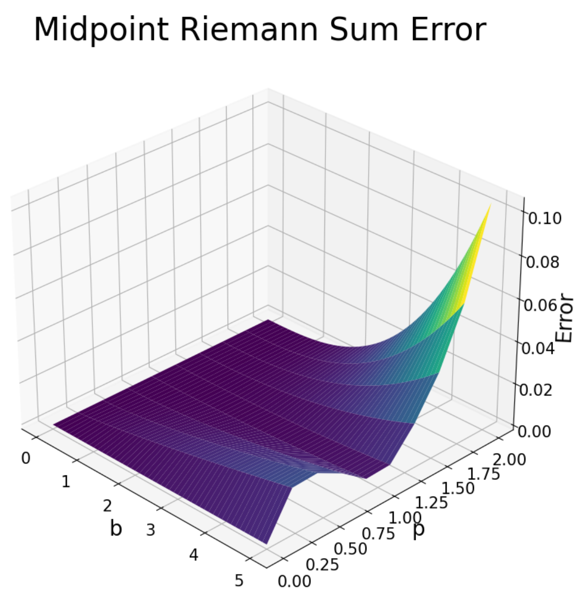

Figure 8

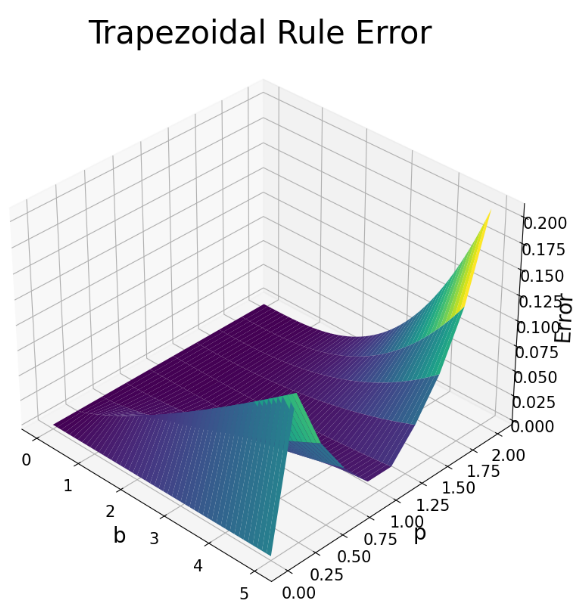

Figure 9

3.4 Experiment 4

We will fix p = 3, n = 10 and b = 1 and see the error surface of the following:

3.5 Experiment 5

We will fix p = 3, n = 10 and b = 10 and see the error surface of the following:

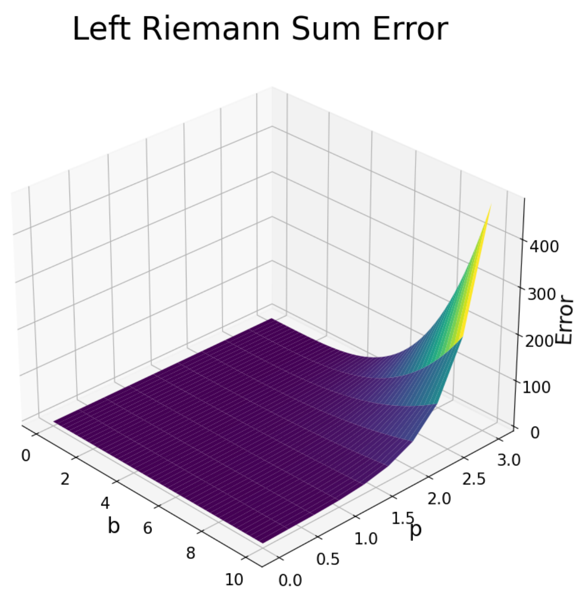

Figure 10

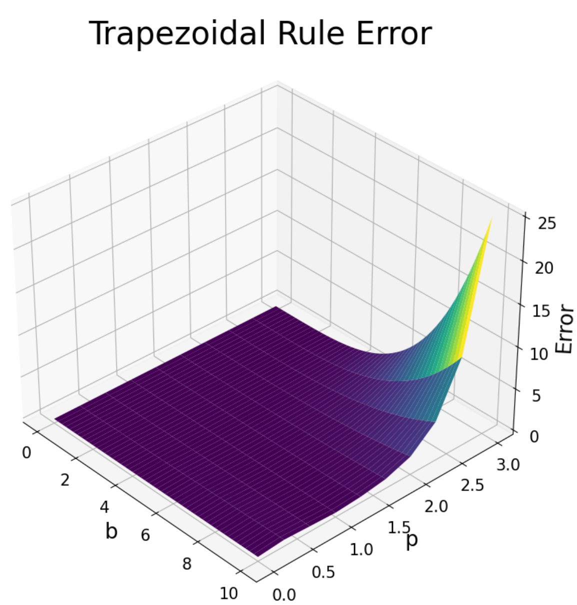

Figure 11

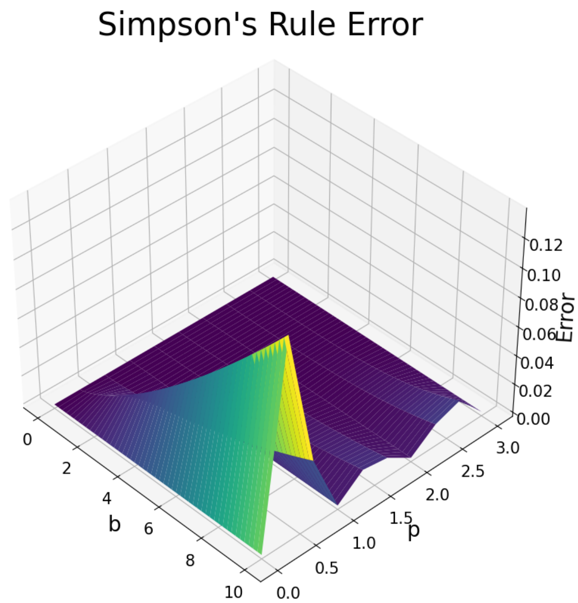

Figure 12

3.6 Experiment 6

We will fix p = 3, n = 10 and b = 100 and see the error surface of the following:

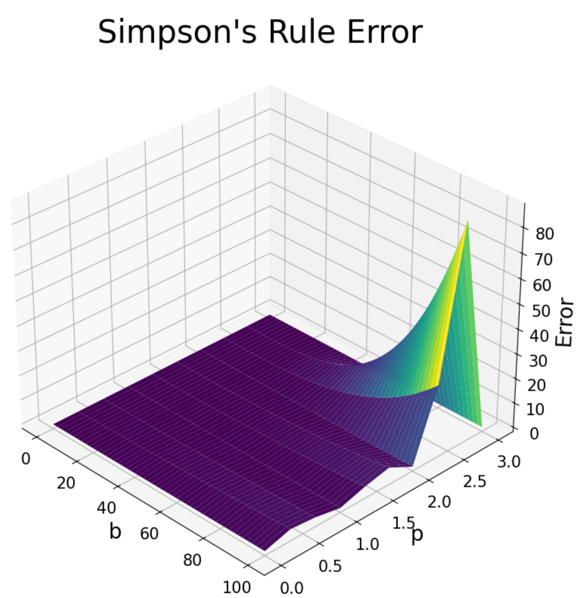

Figure 13

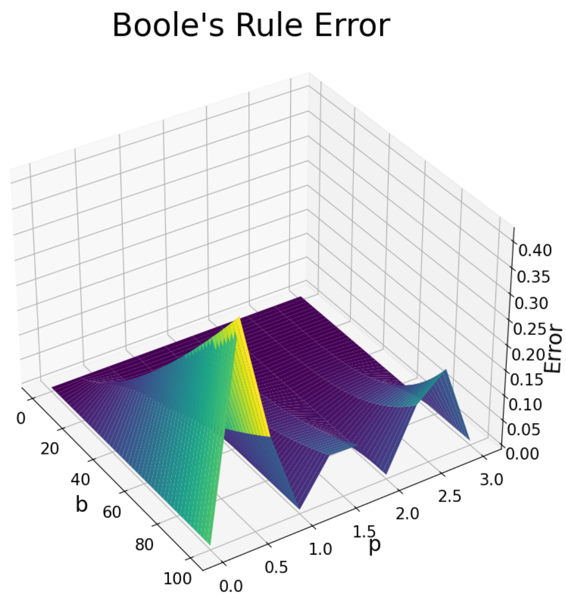

Figure 14

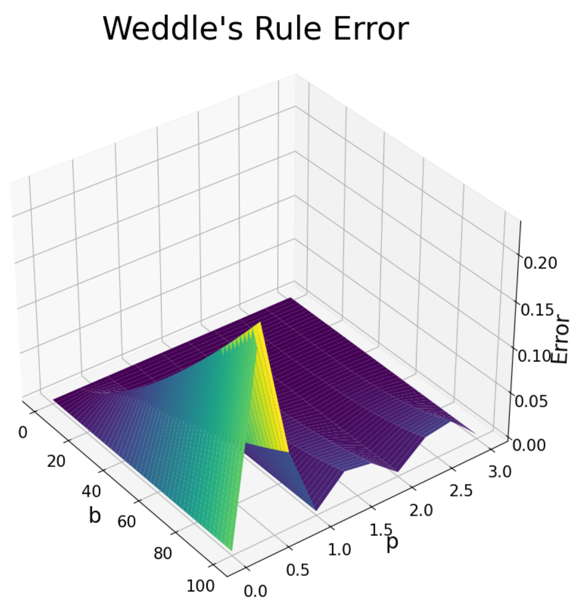

Figure 15

3.7 Experiment 7

We will fix p = 3, n = 100 and b = 5 and see the error surface of the following:

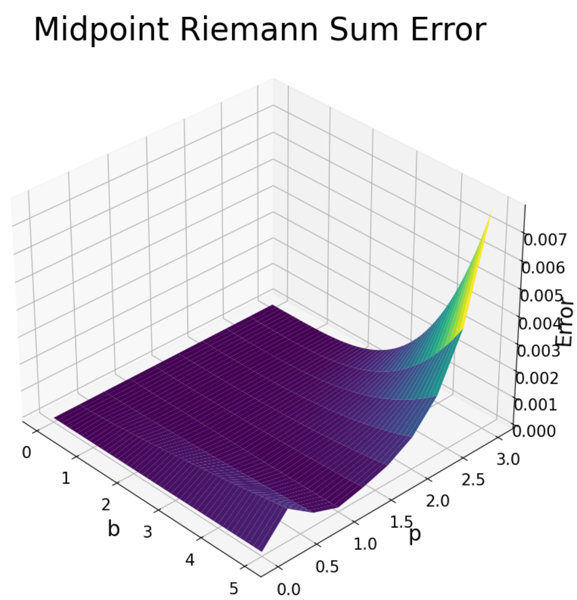

Figure 16

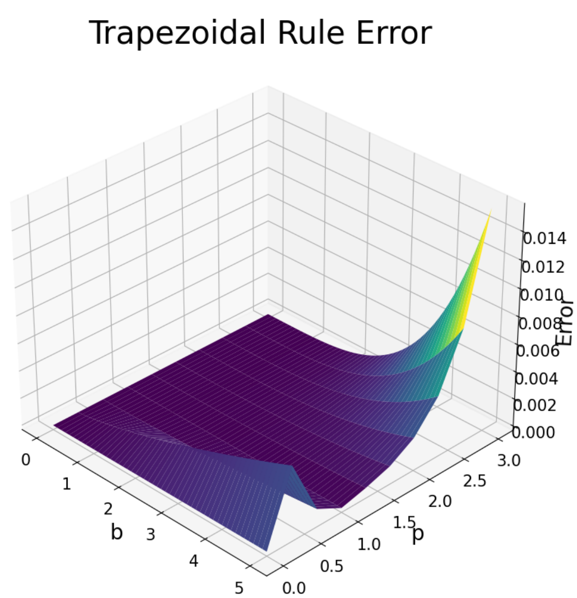

Figure 17

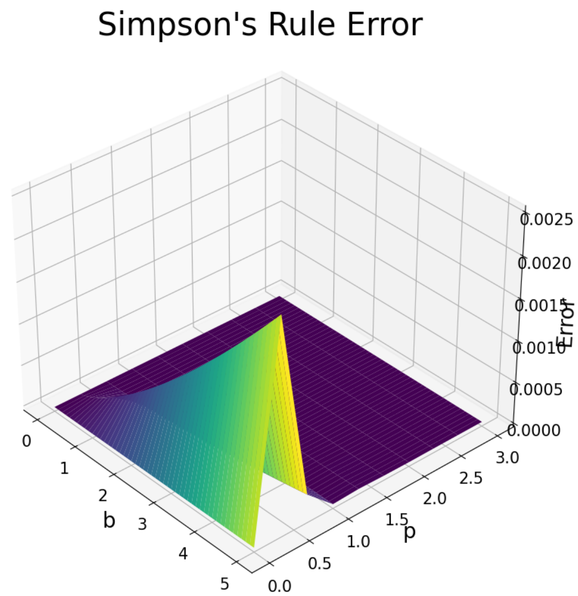

Figure 18

We also investigated the effect of n on the error surface during our analysis. We computed the error vs p vs n. It was astonishing to find out that with varying values of n, we have drastic change in the error surface. We will try to show some significant results we obtained below:

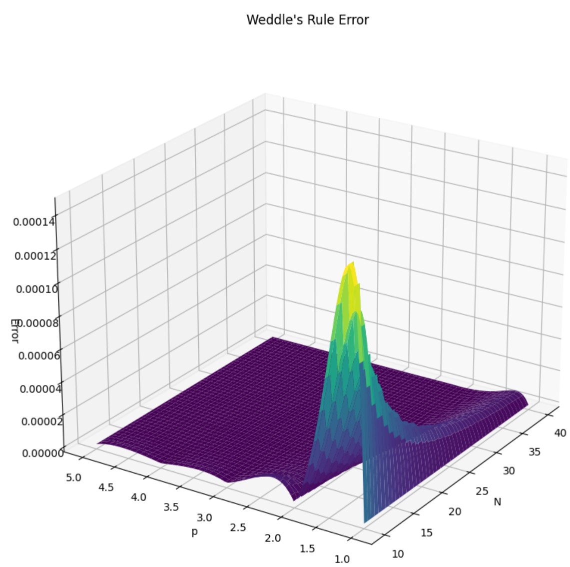

Figure 19

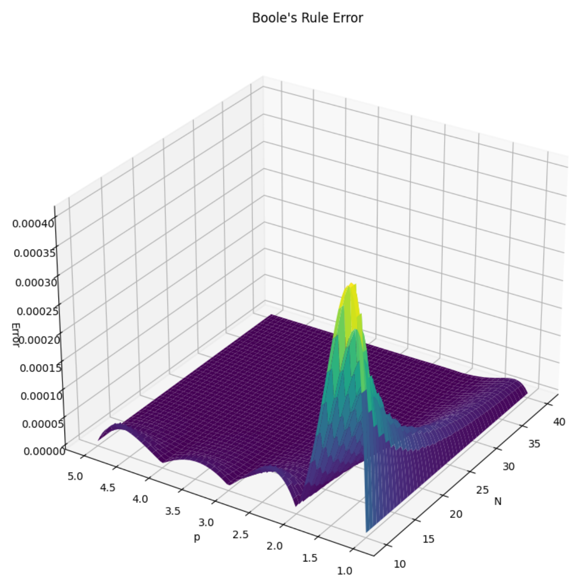

Figure 20

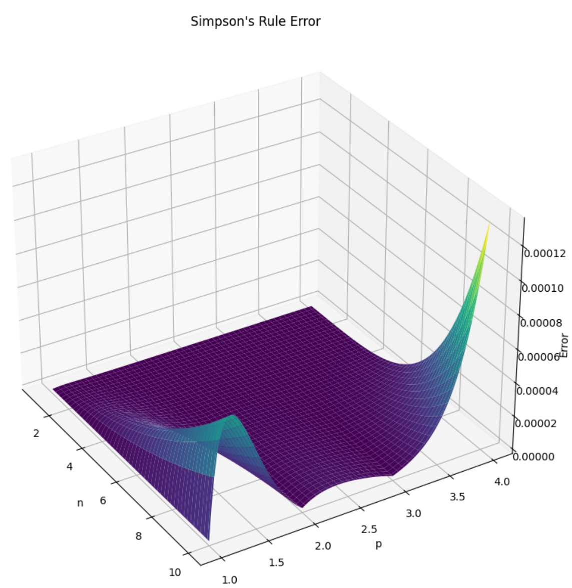

Figure 21

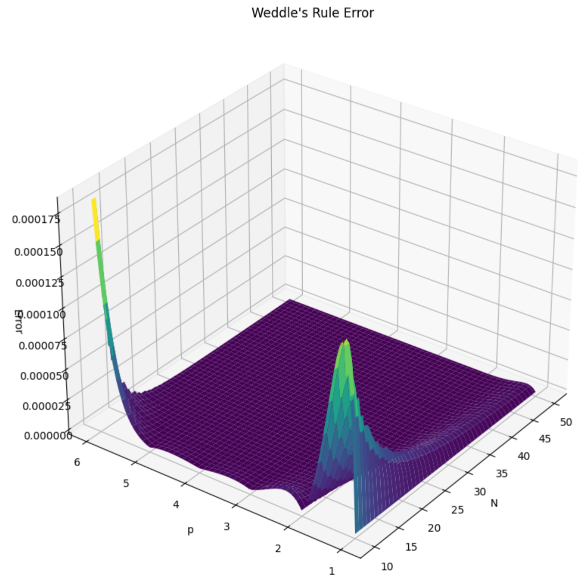

Figure 22

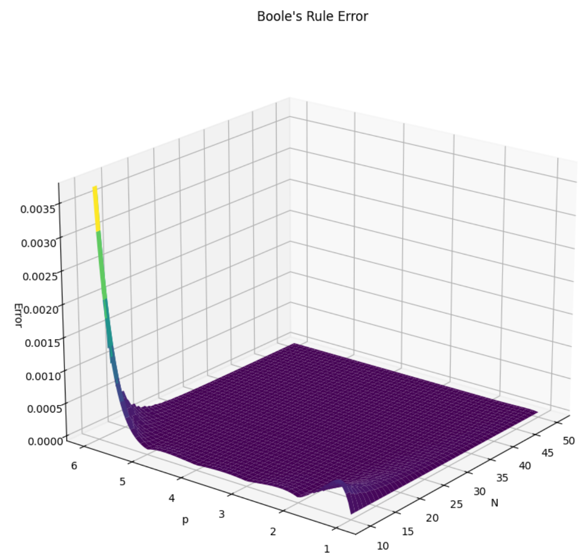

Figure 23

Table 1: Errors

Approximation Scheme	Left Hand Riemann	Midpoint Riemann	Right Hand Riemann	Trapezoidal	Simpson	Boole	Weddle
p=0.5, n=10, b=5	0.6278579802238449	0.019198594330299024	0.49017600852604915	0.0688409858488983	0.028700792362719163	0.00315054119990954	0.0016131062124529194
p=1, n=10, b=5	1.25	0.0	1.25	0.06	0.03	0.0	0.0
p=2, n=10, b=5	6.041666666666657	0.10416666666665719	6.458333333333343	0.2083333333333428	7.105427357601002e-15	1.4210854715202004e-14	7.105427357601002e-15
p=3, n=10, b=1	0.047499999999999993	0.0012500000000000011	0.052499999999999999	0.0025000000000000577	0.0	5.551115123125783e-17	5.551115123125783e-17
p=3, n=10, b=10	475.00000000000045	12.500000000000455	524.9999999999995	24.999999999999545	4.547473508864641e-13	4.547473508864641e-13	4.547473508864641e-13
p=3, n=10, b=100	4750000.0	125000.0	5250000.0	250000.0	0.0	0.0	0.0
p=3, n=100, b=5	3.1093749999999716	0.007812500000056843	3.1406249999999716	0.015624999999971578	0.0	0.0	2.842170943040401e-14

※ Conclusion

In our study of error estimation, we were able to comparatively analyze Left hand Riemann Method, Midpoint Riemann Method, Right hand Riemann Method, Trapezoidal Method, Simpson Method, Boole's Method and Weddle's Method. We found that midpoint method produces a smaller error as opposed to LHR and RHR method. This is directly associated to the fact that midpoint method with $\theta = 0.5$ is optimal, producing the smallest possible error under fairly low smoothness assumption as proved in Theorem 1.8 (2nd order error bounds for the Riemann sums with f $\in C^2[a, b]$) [1].
In addition, we can also see controlled error with the methods like Trapezoidal and Simpsons method. We were able to show that the error of Simpsons method tends to be proportional to the Trapezoidal method for less smooth integrals, $f \in C^2$, which are linear polynomial p-exact on [a, b]. and guarantee at least second order convergence as h $\downarrow$ 0.
We also need to pay attention to significance of n in the error surface. As suggested in the results section, we can clearly see that Boole's and Weddle's rule shined during the error estimates. It is worth noting that we got better curves of error estimates for n between 10 and 50. We could also try to compute n more than 100 but the 'TimeWarning Error' occurred due computational limitations. It is also evident that we have to have a finite bounds to compute n and we can not compute it for infinite interval. We noticed that even if Simpson's method is relatively better than Trapezoidal, LRS, RRS and Midpoint Method, The Boole's and Weddle's method performed far better with comparatively lower error rates.
After the implementation of Boole's and Weddle's Method, we proved that the error was much smaller than all the previous schemes. As suggested in [3], we confirmed that the error of Boole's and Weddle's method proved to be the lowest.

※ Acknowledgement

S.P and H.S: Conceptualization, Methodology, Software, Validation, Formal analysis, Writing - original draft, Visualization, **S.P. and M.S**: Conceptualization, Writing - original draft, review and editing, **H.S**: Project administration.

※ References

References

[1] H. Schurz. *Lecture Notes of Approximation Theory*. SIU, Carbondale, 2016

[2] Kendall Atkinson. *An Introduction to Numerical Analysis,*. Wiley, NY, 1978

[3] J. Stoer and R. Bulirsch. *Introduction to Numerical Analysis*. 2nd Edition, Springer, NY, 1993

[4] J. D. Hunter *Matplotlib: A 2D Graphics Environment*. Computing in Science and Engineering, vol. 9, no. 3, pp. 90-95, 2007.