Filomat 40:5 (2026), 1593–1610
https://doi.org/10.2298/FIL2605593Y

Published by Faculty of Sciences and Mathematics,
University of Niš, Serbia
Available at: http://www.pmf.ni.ac.rs/filomat

Numerical stability of barycentric Lagrange interpolation

Alvira Yawar[a,*], **Swarnima Bahadur**[a]

[a] *Department of Mathematics & Astronomy, University of Lucknow, Lucknow, India*

Abstract. This paper presents a refined study on the numerical stability of barycentric Lagrange interpolation constructed using Legendre Gauss Lobatto (LGL) nodes, which consist of the enpoints ±1 and the zeros of $P'_{n-1}(x)$, where $P_{n-1}(x)$ is the Legendre polynomial of degree $(n-1)$. Unlike prior work focused on Chebyshev nodes, our formulation provides a detailed error analysis tailored to this specific node set. We derive the barycentric weights from the structure of the underlying node generating polynomial and establish new interpolation error bounds supported by asymptotic estimates of barycentric weights. Extensive numerical comparisons between classical Lagrange and barycentric interpolation across various challenging test functions demonstrate that the barycentric formulation maintains machine level accuracy and robust stability even at high degrees (up to $n = 550$), while classical Lagrange interpolation suffers from catastrophic numerical error. These results underline the practical superiority of barycentric interpolation at LGL nodes and provide a rigorous theoretical foundation for its use in high-accuracy applications.

1. Introduction

Interpolation, the process of estimating values between known data points, has long played a fundamental role in numerical analysis and applied mathematics. From early numerical tables to modern computational tools, interpolation methods are widely used in science, engineering, and data modeling. One of the earliest and most widely used polynomial interpolation techniques is Lagrange interpolation, introduced by Joseph Louis Lagrange in the 18th century. It constructs a unique polynomial that passes through a given set of points, offering an elegant analytical form useful in both theoretical and practical computations [4, 8].

Given $n+1$ distinct points $(x_0, y_0), (x_1, y_1), ..., (x_{n-1}, y_{n-1}), (x_n, y_n)$, the Lagrange interpolating polynomial is written as:

$$P_n(x) = \sum_{j=0}^{n} y_j l_j(x), \text{ where } l_j(x) = \prod_{\substack{k=0 \\ k\neq j}}^{n} \frac{x - x_k}{x_j - x_k} . \tag{1}$$

While Lagrange interpolation is mathematically simple, it can face two key challenges: Runge's phenomenon and numerical instability. Runge's phenomenon occurs when interpolating at equally spaced

2020 *Mathematics Subject Classification*. Primary 41A05; Secondary 41A20, 65A05.
Keywords. Lagrange Interpolation, Barycentric Interpolation, Legendre Polynomial, Rational Interpolation, Numerical Stability.
Received: 29 July 2025; Revised: 01 October 2025; Accepted: 07 December 2025
Communicated by Dragan S. Djordjević

* Corresponding author: Alvira Yawar
Email addresses: hussainalvira0@gmail.com (Alvira Yawar), swarnimabahadur@ymail.com (Swarnima Bahadur)
ORCID iDs: https://orcid.org/0009-0008-4294-9325 (Alvira Yawar), https://orcid.org/0000-0001-8734-943X (Swarnima Bahadur)

points, causing large oscillations near the endpoints of the interval [10, 28]. This is a mathematical limitation related to the choice of nodes, not numerical error. To reduce this, researchers often use Chebyshev or Legendre based nodes, which are clustered near the boundaries and help control the interpolation error.
The second challenge is numerical instability, which arises from floating-point arithmetic errors introduced during computation due to rounding and subtractive cancellation. These issues become more pronounced as the degree of interpolation increases. Even when good node distributions like Chebyshev or Legendre nodes are used to reduce theoretical error, the standard Lagrange form remains computationally inefficient and sensitive to such numerical issues.
A critical distinction must be made between the theoretical properties of an interpolant, such as its convergence rate, and its numerical stability when implemented in finite-precision arithmetic. Indeed, theoretical studies by Vértesi [19, 37] and others have established that key characteristics of barycentric interpolation, including the behavior of the Lebesgue constant and related convergence theorems, are very similar to those of its classical Lagrange counterpart.However, these theoretical results are derived in the context of exact arithmetic and do not fully predict the behavior in practical computation. The classical Lagrange formula (1) becomes severely ill-conditioned as the degree n increases, suffering from catastrophic rounding errors when evaluated in floating-point arithmetic. In stark contrast, the second (true) barycentric form (Eq. 10) is renowned for its numerical robustness. As proven by Higham [16], this formulation is backward stable, fundamentally avoiding the sources of error that plague the classical approach. This profound difference in computational stability, rather than theoretical convergence, is the primary reason for the major discrepancy in errors observed in our experiments. Consequently, this investigation focuses on empirically demonstrating and quantifying this stability for interpolation at Legendre Gauss Lobatto nodes.

To address this, a more numerically stable formulation known as barycentric interpolation was developed. This approach separates the computation of weights from the evaluation step, reducing both complexity and numerical sensitivity. In the first form of the barycentric interpolation formula, the interpolating polynomial is expressed using weights defined as:

$$w_j = \frac{1}{\prod_{\substack{k=0\\k\neq j}}^{n}(x_j - x_k)} .$$

The interpolating polynomial becomes:

$$P_n(x) = \omega(x) \sum_{j=0}^{n} \frac{w_j y_j}{x - x_j}, \text{ where } \omega(x) = \prod_{j=0}^{n} (x - x_j) .$$

Although more efficient, this form can still suffer from numerical issues when $\omega(x)$ becomes very small or large. To improve this, a second (true) form of the barycentric interpolation was introduced:

$$P_n(x) = \frac{\sum_{j=0}^{n} \frac{w_j y_j}{x - x_j}}{\sum_{j=0}^{n} \frac{w_j}{x - x_j}} .$$

This form cancels the node polynomial $\omega(x)$, making it significantly more stable and scalable for high-degree polynomials [4, 15, 29, 31]. Higham showed that this version is backward stable in floating point arithmetic [16], and it has since been widely adopted in modern numerical software.
Researchers such as Salzer, Henrici, and Rutishauser have contributed to understanding the historical and numerical development of barycentric formulas [14, 31, 34]. The use of special nodes, particularly Chebyshev , has also been shown to improve interpolation accuracy and reduce the Lebesgue constant [6, 27, 36]. Further advancements have led to rational barycentric interpolation, which allows for interpolation using rational functions rather than just polynomials. Floater and Hormann introduced a family of rational barycentric interpolants that avoid real poles and can achieve high approximation orders on arbitrary node distributions [11, 17, 18] These interpolants blend local polynomial approximations to construct highly accurate global interpolants [3, 13, 30].

Further advancements have led to rational barycentric interpolation, which allows for interpolation using rational functions rather than just polynomials. Floater and Hormann introduced a family of rational barycentric interpolants that avoid real poles and can achieve high approximation orders on arbitrary node distributions [11, 17, 18] These interpolants blend local polynomial approximations to construct highly accurate global interpolants [3, 13, 30].

In our previous work [38], authored by Alvira Yawar and Swarnima Bahadur, we explored how barycentric interpolation can be derived from the Lagrange form and expressed as a rational function. We focused on interpolation using the zeros of the Legendre polynomial $P_n(x)$ over the interval $(-1, 1)$, excluding the endpoints. Through numerical experiments, we showed that the second barycentric form is more stable and accurate than traditional polynomial methods, especially for high degrees.

In this current paper, we extend our study by focusing on the zeros of the function $(1 - x^2)P'_{n-1}(x)$, where $P'_{n-1}(x)$ is the derivative of the Legendre polynomial of degree $(n - 1)$. This function plays an important role in Gaussian quadrature and spectral methods, as its zeros are used to determine optimal node distributions for integration and differential equation solvers [12, 21, 35]. Unlike our previous study, we now include the endpoints, working over the closed interval $[-1, 1]$. This allows us to investigate how endpoint inclusion affects node distribution and interpolation behavior.

Recent studies by Gevorgyan, Boyd, Dutt, and others have shown that the choice of nodes whether Legendre, Chebyshev, or derivative-based can greatly influence both the accuracy and stability of interpolation and quadrature methods [5, 9, 16, 27]. Theoretical advancements on weight construction, error bounds, and convergence rates further support the use of barycentric and rational methods in high accuracy numerical applications [6, 7, 32].

Our results build on this rich foundation, providing new insights into how Legendre polynomial derivatives and their associated zeros can enhance barycentric interpolation, especially when the full interval is considered.

In the broader context of operator approximation theory, several recent works [1, 2, 22–26, 39] have explored stability and convergence of generalized operator systems, highlighting the importance of numerically stable interpolation frameworks such as the one considered in this study.

Main Contributions

The key contributions of this work are as follows:

- A substantial amount of work already exists on Chebyshev nodes in the context of barycentric interpolation. To explore an alternative node distribution, we considered the zeros of $\pi_n(x) = (1 - x^2)P'_{n-1}(x)$.
  Our aim was to study the interpolation behavior and numerical stability of barycentric interpolation specifically on this less investigated set of nodes.
- Construction of a barycentric interpolation formula using Legendre Gauss Lobatto (LGL) nodes.
- Derivation of explicit barycentric weights based on the zeros of $(1 - x^2)P'_{n-1}(x)$.
- Establishment of a quantitative error bound $O(n^{3/2}\,\omega_r(g, n^{-1}))$ for the interpolant.
- Numerical comparison with classical Lagrange interpolation demonstrating catastrophic instability in the latter at high degrees.
- Demonstration that the barycentric scheme remains accurate and stable even for degrees up to $n = 550$, confirming its robustness.

The organization of this paper is as follows. Section 2 introduces the key mathematical background and definitions related to Legendre polynomials and the LGL nodes. In Section 3, we construct the barycentric interpolation formula using these nodes. Section 4 provides theoretical estimates for the interpolation error, including key lemmas and a convergence result. Section 5 presents numerical experiments that compare the classical Lagrange and barycentric interpolation methods for various test functions. Finally, Section 6 summarizes the findings and highlights the improved stability of the barycentric interpolation.

2. Preliminaries

Let $P_n(x)$ denote the Legendre polynomial of degree n, which is orthogonal over $[-1, 1]$ and satisfies the second order differential equation:

$$(1-x^2)P_n''(x) - 2xP_n'(x) + n(n+1)P_n(x) = 0\ . \tag{2}$$

Define the function:

$$\pi_n(x) = (1-x^2)P_{n-1}'(x), \tag{3}$$

which plays a central role in Gaussian quadrature and spectral collocation methods. Its n zeros in $[-1, 1]$, including the endpoints, are known as Legendre Gauss Lobatto (LGL) nodes. These nodes are commonly used in interpolation due to their endpoint inclusion and favorable clustering properties.
The interpolation nodes are given by:

$$x_1 = -1, \quad x_n = 1, \quad \text{and} \quad x_2, x_3, \dots, x_{n-1} \text{ are the zeros of } P_{n-1}'(x).$$

We now define the derivative of $\pi_n(x)$ see [33]

$$\pi_n'(x) = [(1-x^2)P_{n-1}'(x)]' = -n(n-1)P_{n-1}(x). \tag{4}$$

It is also useful to recall a standard upper bound on the derivative of Legendre polynomials [33]. For $x \in [-1, 1]$, the derivative of the Legendre polynomial $P_{n-1}'(x)$ satisfies:

$$|P_{n-1}'(x)| \leq \frac{n(n-1)}{2}. \tag{5}$$

Let $g(x)$ be a smooth function defined on $[-1, 1]$, and define

$$y_k = g(x_k), \text{ for } k = 1, 2, \dots, n\ .$$

Additionally, we introduce the Lagrange interpolation polynomials defined on the zeros of $\pi_n(x)$, which are given by:

$$l_k(x) = \frac{\pi_n(x)}{(x-x_k)\pi_n'(x_k)}\ , \qquad k = 1, 2, 3, ..., n\ . \tag{6}$$

3. Representation of Barycentric Interpolation formula

The equation for $l_k(x)$ satisfies the orthogonality condition:

$$l_k(x_m) = \delta_{km}\ , \qquad k, m = 1, 2, ...n\ . \tag{7}$$

Thus, the Lagrange polynomial can be written as:

$$\mathcal{L}(g, x) = \sum_{k=1}^{n} \frac{y_k \pi_n(x)}{(x-x_k)\pi_n'(x_k)}\ . \tag{8}$$

This expression is known as the first form of the barycentric interpolation formula.
We can also express the Lagrange interpolation formula for lower degree polynomials as follows:

$$\frac{1}{\pi_n(x)} = \sum_{k=1}^{n} \frac{1}{(x-x_k)\pi'_n(x_k)} \, . \tag{9}$$

Therefore, the second (true) form of the barycentric interpolation formula is:

$$\mathscr{L}(g,x) = \sum_{k=1}^{n} \frac{y_k}{(x-x_k)\pi'_n(x_k)} \Bigg/ \sum_{k=1}^{n} \frac{1}{(x-x_k)\pi'_n(x_k)} \, . \tag{10}$$

Lemma 3.1. *Let $\{x_k\}_{k=1}^{n}$ be the Legendre Gauss Lobatto nodes, i.e., the zeros of $\pi_n(x) = (1-x^2)P'_{n-1}(x)$, and let w_k be the barycentric weights. These weights are defined as the reciprocal of the derivative of $\pi_n(x)$ evaluated at the nodes x_k, i.e., $w_k = \frac{1}{\pi'_n(x_k)}$. Using the identity given in Eq.* (4)*, where $\pi_n(x) = -n(n-1)P_{n-1}(x)$, the weights can be computed directly.*
Then, the rational interpolant $\mathcal{L}(g,x)$[9] given by equation (10) can also be expressed as

$$\mathcal{L}(g,x) = \frac{\sum_{k=1}^{n} y_k \xi_k(x)}{\sum_{k=1}^{n} \xi_k(x)} \, , \tag{11}$$

where $\xi_k(x)$ is :

$$\xi_k(x) = \prod_{\substack{j=1 \\ j\neq k}}^{n} (x-x_j) \prod_{\substack{j=1 \\ j\neq k}}^{n} \pi'_n(x_j) \, , \tag{12}$$

the functions $\xi_k(x)$ is constructed by excluding x_k, from the products, ensure the desired interpolation properties without introducing poles within the interpolation interval.

4. Quantative Estimate of Interpolatory Polynomial

Lemma 4.1. *Let $\mu_k = \frac{1}{(x-x_k)\pi'_n(x_k)}$. Then*

$$\frac{1}{|\sum_{k=1}^{n} \mu_k|} = O(n^2) \, . \tag{13}$$

Proof. We have

$$|\sum_{k=1}^{n} \mu_k| = |\sum_{k=1}^{n} \frac{1}{(x-x_k)\pi'_n(x_k)}| = |\frac{\sum_{j=1}^{n} \xi_j(x)(x-x_n)\pi'_n(x_n)}{\prod_{j=1}^{n}(x-x_j)\pi'_n(x_j)(x-x_n)\pi'_n(x_n)}| \, . \tag{14}$$

$$\implies |\sum_{k=1}^{n} \mu_k| \geq |\frac{\xi_n(x)(x-x_n)\pi'_n(x_n)}{\prod_{j=1}^{n}(x-x_j)\pi'_n(x_j)(x-x_n)\pi'_n(x_n)}| = |\frac{1}{(x-x_n)\pi'_n(x_n)}| \, . \tag{15}$$

$$\implies |\sum_{k=1}^{n} \mu_k| \geq \frac{1}{2n(n-1)} \, , \qquad [by \qquad (5)] \, . \tag{16}$$

Hence, the lemma follows.

Lemma 4.2. *Let $\mu_k = \frac{1}{(x-x_k)\pi'_n(x_k)}$. Then*

$$\sum_{k=1}^{n} |\mu_k| = O(n^{\frac{-1}{2}}) \, . \tag{17}$$

Proof.

$$|\sum_{k=1}^{n} \mu_k|= |\sum_{k=1}^{n} \frac{1}{(x-x_k)\pi_n'(x_k)}| \,. \tag{18}$$

$$|\sum_{k=1}^{n} \mu_k|= |\frac{1}{(x-x_1)\pi_n'(x_1)} + \sum_{k=2}^{n-1} \frac{1}{(x-x_k)\pi_n'(x_k)} + \frac{1}{(x-x_n)\pi_n'(x_n)}| \,, \tag{19}$$

$$|\sum_{k=1}^{n} \mu_k|= I_1 + I_2 + I_3 \,. \tag{20}$$

After little computation [33], we have

$$I_1 = \frac{1}{(x-x_1)\pi_n'(x_1)} = O(\frac{1}{n}) \,, \tag{21}$$

$$I_2 = \sum_{k=2}^{n-1} \frac{1}{(x-x_k)\pi_n'(x_k)} = O(n^{-\frac{1}{2}}) \,, \tag{22}$$

$$I_3 = \frac{1}{(x-x_n)\pi_n'(x_n)} = O(\frac{1}{n}) \,. \tag{23}$$

Thus, from (21),(22),(23) we have the result.

Theorem 4.3. *Let $g(x)$ be a function that is continous on the interval $[-1, 1]$ and differentiable on $(-1, 1)$. Consider the sequence of barycentric interpolating polynomial $\mathcal{L}(g, x)$, which is defined by*

$$\mathcal{L}(g, x) = \frac{\sum_{k=1}^{n} g(x_k)\mu_k(x)}{\sum_{k=1}^{n} \mu_k(x)} \,. \tag{24}$$

Then, the interpolation error satisfies the estimate

$$|\mathcal{L}(g, x) - g(x)|= O(n^{\frac{3}{2}} \ \omega_r(g, n^{-1})) \,, \qquad (\textit{for } r \geq 2), \tag{25}$$

where $\omega_r(g, n^{-1})$ is the r^{th} modulus of continuity of g .

The following lemmas are used to prove the above mentioned theorem.

Remark 4.4. *Assume that $g(x)$ is continous on [-1,1], differentiable on (-1,1), $g^{(r-1)} \in Lip\ \alpha$, where $\alpha > 0$. Then the sequence $\{\mathcal{L}(g, x)\}$ converges uniformly to $g(x)$ on [-1,1],and the modulus of continuity satisfies*

$$\omega_r(g, n^{-1}) = O(n^{-r+1-\alpha}), \qquad (r \geq 2), \tag{26}$$

where $\omega_r(g, n^{-1})$ is the r^{th} modulus of continuity of g .

Remark 4.5. *Let $g(x)$ be continous on the interval [-1,1] and differentiable on (-1,1).Then there exists a polynomial $\mathcal{S}_n(x)$ of degree atmost $n - 1$ such that **Jackson's inequality** [20] holds*

$$|\mathcal{S}_n(x) - g(x)|= O(\omega_r(g, n^{-1})), \tag{27}$$

where $\omega_r(g, n^{-1})$ is again the r^{th} modulus of continuity of g .

Proof of Theorem 4.3. The interpolating polynomial $\mathcal{L}(g,x)$ is uniquely defined and has degree at most $n-1$. Let the best approximating polynomial be given by $\mathcal{S}_n(x)$ and is of degree at most $n-1$, which interpolates $g(x)$ at n distinct points $x_1, x_2, ..., x_n$, satisfying equation (27) .

$$\mathcal{S}_n(x) = \frac{\sum_{k=1}^{n} \mu_k(x)\mathcal{S}_n(x_k)}{\sum_{k=1}^{n} \mu_k(x)}, \tag{28}$$

where the weights are defined by

$$\mu_k(x) = \frac{1}{(x-x_k)\pi_n'(x_k)} \text{ for } \quad k = 1, \dots, n \, . \tag{29}$$

Now we estimate the error

$$|\mathcal{L}(g,x) - g(x)| \leq |\mathcal{L}(g,x) - \mathcal{S}_n(x)| + |\mathcal{S}_n(x) - g(x)| \, . \tag{30}$$

From (24) and (28), we get

$$|\mathcal{L}(g,x) - g(x)| \leq \frac{\sum_{k=1}^{n} |\mu_k(x)||g(x_k) - \mathcal{S}_n(x_k)|}{|\sum_{k=1}^{n} \mu_k(x)|} + |\mathcal{S}_n(x) - g(x)| \, . \tag{31}$$

by (13),(17) and (27), we get

$$|\mathcal{L}(g,x) - g(x)| \leq O(n^{\frac{3}{2}} \, \omega_r(g, n^{-1})) + O(\omega_r(g, n^{-1})). \tag{32}$$

Hence,

$$|\mathcal{L}(g,x) - g(x)| = O(n^{\frac{3}{2}} \, \omega_r(g, n^{-1})) \, . \tag{33}$$

5. Numerical Comparison of Classical Lagrange and Barycentric Interpolation

To numerically compare the performance of classical Lagrange interpolation and the second form barycentric interpolation formula, we implemented both methods using Legendre Gauss Lobatto (LGL) nodes on the interval $[-1, 1]$. The LGL nodes were constructed by taking the endpoints -1 and 1 together with the roots of the derivative of the Legendre polynomial $P_{n-1}(x)$, where n is the interpolation degree. These nodes are known for clustering near the endpoints, which helps reduce the effects of the Runge phenomenon and improves interpolation stability.

For the barycentric method, we adopted the second (numerically stable) form. Given the LGL nodes x_j, we defined the function $G(x) = \pi_n(x)$ and computed the barycentric weights as $w_j = \frac{1}{G'(x_j)}$.

These weights were then used in the barycentric interpolation formula to evaluate the interpolant. Classical Lagrange interpolation was implemented using polynomial fitting based on the same LGL nodes.

Each interpolant was evaluated on a dense grid of 1000 points in $[-1, 1]$, and the maximum absolute interpolation error was calculated by comparing the interpolated values with the true function values.

Visualisation and Colour Scheme

To visualize the performance of both interpolation schemes, plots were generated for each test case. A consistent color scheme was used across all figures: the exact function is a solid green line, the Lagrange interpolant is a red dashed line, and the barycentric interpolant is a blue dashed line. Absolute interpolation errors are plotted on a semilogarithmic scale, with red for Lagrange and blue for barycentric.To establish the pattern of behavior in detail, our first test case includes a full set of plots, including zoomed-in views near the boundaries at $x = \pm 1$. For the subsequent examples, which reinforce the same conclusion, we present only the direct comparison of the final error plots to maintain focus and avoid redundancy.

Interpolation Degrees and Tabulated Errors

This comparison was performed for interpolation degrees $n = 50, 100, 150, \dots, 550$. For each case, the maximum absolute errors of both the Lagrange and barycentric interpolants were computed and tabulated to assess convergence and numerical stability across increasing degrees. The detailed results and graphical comparisons for each function appear in the subsequent sections.

Example 1: We examined the performance of Lagrange and Barycentric interpolation schemes using the highly oscillatory function

$$g(x) = \sin(100x^2)$$

This function serves as a challenging test case for polynomial interpolation due to its rapid variations over the interval $[-1, 1]$. It helps reveal the numerical stability and accuracy limits of each method as the degree increases.

The table below presents the maximum absolute error for both interpolation techniques across varying degrees. For lower values of n, both methods yield comparable results. However, as the degree increases, the Lagrange interpolant becomes severely unstable, producing exponentially growing errors. In contrast, the barycentric formula maintains numerical stability, with its error remaining at machine precision for $n \geq 200$.

Table 1: Table for Error term.

Nodes(n)	Error term for Lagrange Interpolation	Error term for Barycentric formula
50	9.373	2.118
100	3.711×10^{12}	4.283×10^{-1}
150	7.102×10^{17}	1.374×10^{-8}
200	7.433×10^{19}	1.465×10^{-14}
250	1.249×10^{23}	1.688×10^{-14}
300	3.836×10^{25}	1.607×10^{-14}
350	9.832×10^{27}	1.482×10^{-14}
400	3.074×10^{30}	1.588×10^{-14}
450	5.768×10^{32}	1.538×10^{-14}
500	3.171×10^{34}	1.482×10^{-14}
550	5.564×10^{35}	1.699×10^{-14}

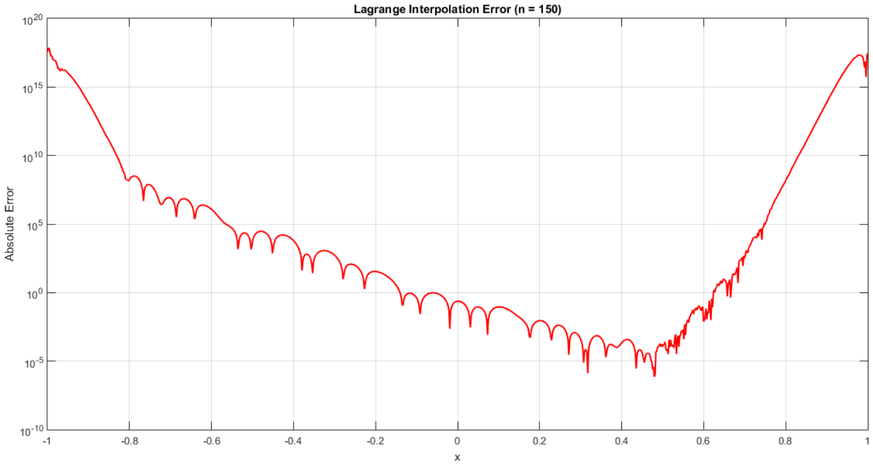


Figure 1: Absolute error of the Lagrange interpolation using $n = 150$ Legendre Gauss Lobatto nodes for the oscillatory function $g(x) = \sin(100x^2)$. The interpolation fails to accurately resolve the high-frequency components, resulting in significant error growth near the boundaries.

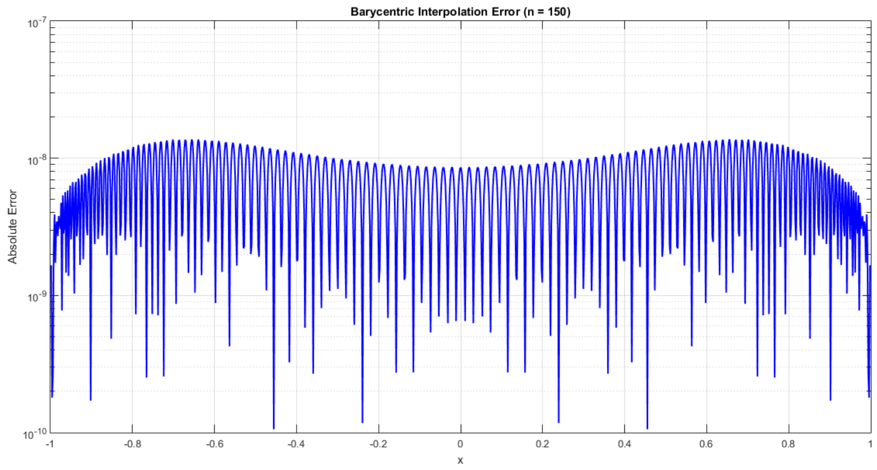


Figure 2: Absolute error of the barycentric interpolation for $g(x) = \sin(100x^2)$ with $n = 150$ Legendre Gauss Lobatto nodes. Although some numerical oscillation is present, the error remains well-contained, demonstrating the methods stability in handling rapidly oscillating functions.

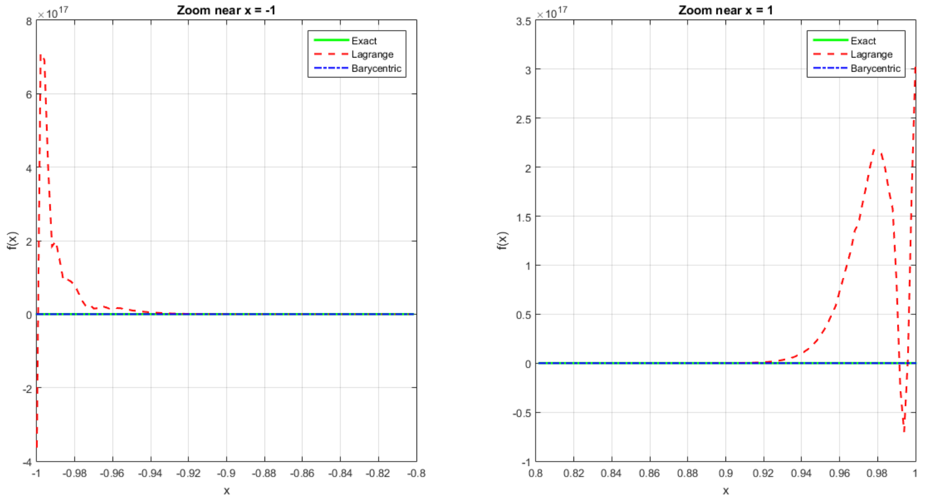
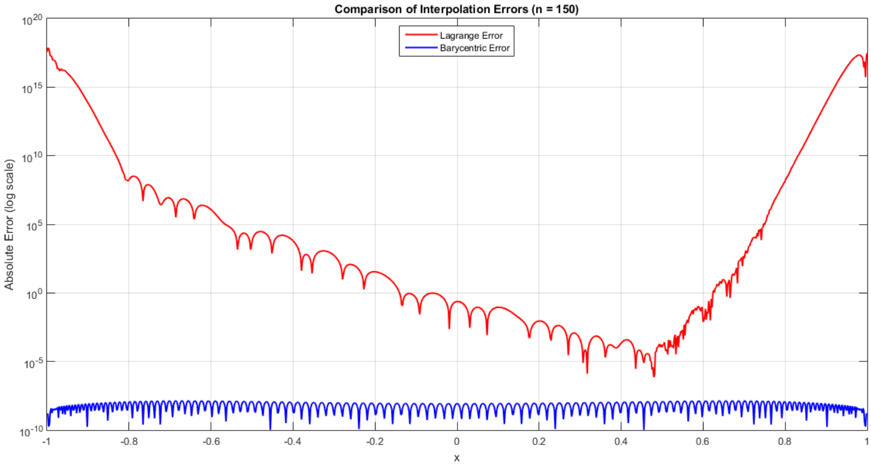


Figure 3: Zoomed-in comparison near $x = -1$ (left) and $x = 1$ (right) for interpolation of $g(x) = \sin(100x^2)$ using $n = 150$ Legendre Gauss Lobatto nodes. The Lagrange interpolant (red dashed) significantly deviates from the exact function (green) at the endpoints, while the barycentric interpolant (blue dash-dot) closely matches the true behavior throughout.

Figure 4: To directly compare the two methods, we also plot both interpolation errors for $g(x) = \sin(100x^2)$ in a single figure for $n = 150$, as shown below.



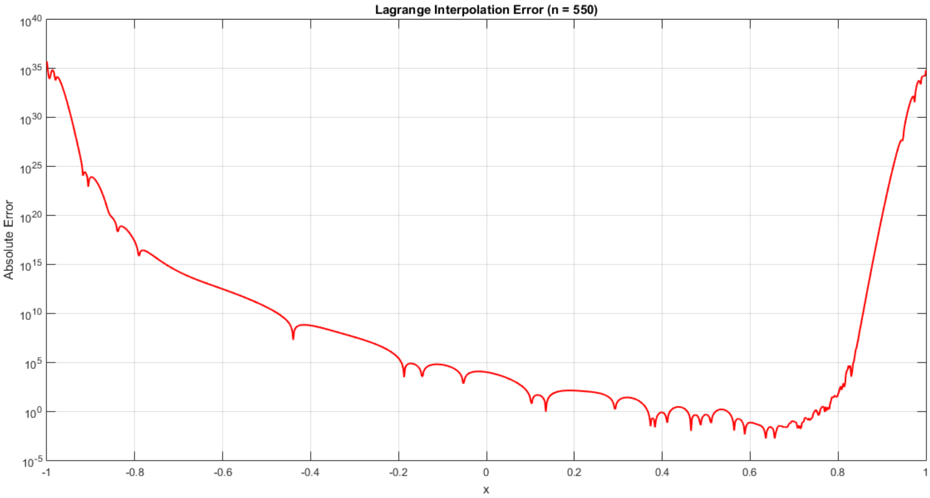


Figure 5: Absolute error of the Lagrange interpolation using $n = 550$ Legendre Gauss Lobatto nodes for the function $g(x) = \sin(100x^2)$. Although the central region is well-approximated, the interpolation error grows rapidly near $x = \pm 1$, exceeding 10^{35} due to high oscillation and numerical instability.

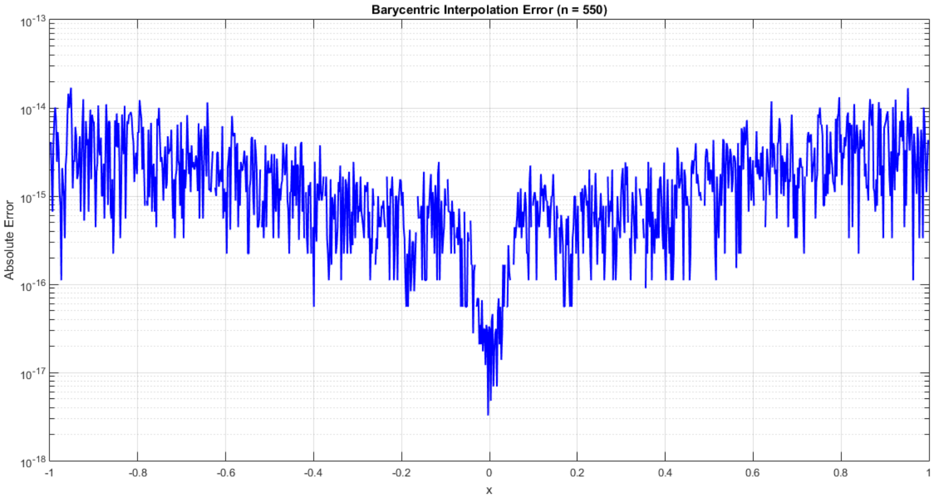


Figure 6: Absolute error of the barycentric interpolation for $g(x) = \sin(100x^2)$ with $n = 550$ Legendre Gauss Lobatto nodes. The method maintains accuracy across the interval, with errors remaining below 10^{-13}, even in the presence of strong oscillations.

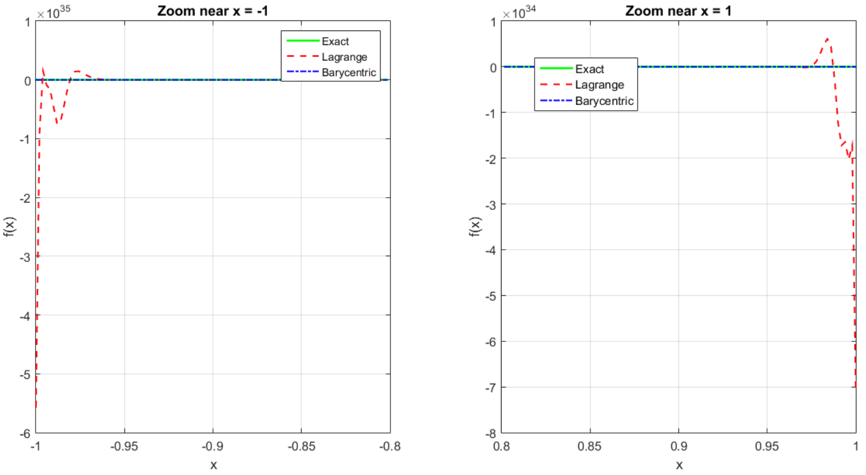
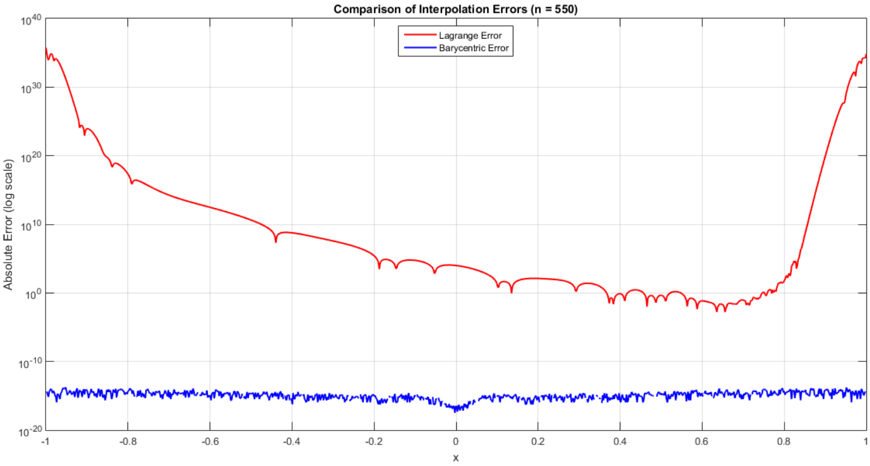


Figure 7: Zoomed comparison of interpolation results near the endpoints $x = -1$ (left) and $x = 1$ (right) for $g(x) = \sin(100x^2)$, using $n = 550$. Lagrange interpolation (red dashed) fails near the boundaries, while barycentric interpolation (blue dash-dot) aligns closely with the exact function (green), preserving accuracy even in high-frequency regimes.

Figure 8: To directly compare the two methods, we also plot both interpolation errors for $g(x) = \sin(100x^2)$ in a single figure for $n = 550$, as shown below.



Example 2: We examined the performance of Lagrange and barycentric interpolation methods for the function

$$g(x) = \frac{1}{1 + 25x^2}$$

This classical Runge-type function is known for its sharp curvature near the endpoints, making it a good candidate for evaluating the stability and accuracy of interpolation schemes, especially for large values of n.

The table below presents the maximum absolute error for both methods as n increases from 50 to 550. The Lagrange method shows a sharp growth in error beyond $n = 100$, highlighting its sensitivity to oscillations near the endpoints (Runge phenomenon). In contrast, the barycentric formula exhibits remarkable stability: the error decreases steadily and reaches machine precision (on the order of 10^{-15}) for $n \geq 200$. This indicates that beyond this point, the barycentric method's accuracy is limited not by the interpolation itself, but by the numerical precision of the system.

Table 2: Table for Error term.

Nodes(n)	Error term for Lagrange Interpolation	Error term for Barycentric formula
50	1.592×10^{-3}	1.061×10^{-4}
100	2.147×10^{7}	5.118×10^{-9}
150	4.270×10^{12}	2.470×10^{-13}
200	1.915×10^{13}	1.665×10^{-15}
250	2.187×10^{16}	2.109×10^{-15}
300	1.821×10^{19}	2.220×10^{-15}
350	2.114×10^{21}	3.997×10^{-15}
400	2.646×10^{23}	2.776×10^{-15}
450	3.006×10^{25}	2.887×10^{-15}
500	6.558×10^{26}	3.886×10^{-15}
550	1.060×10^{28}	2.998×10^{-15}

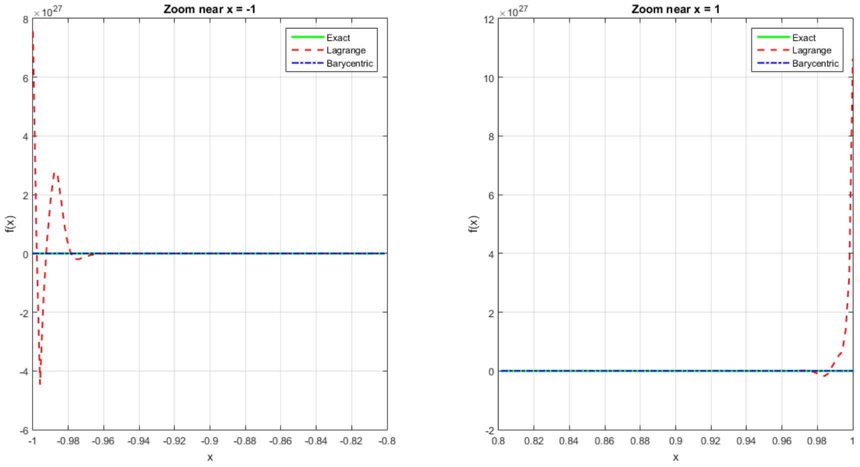
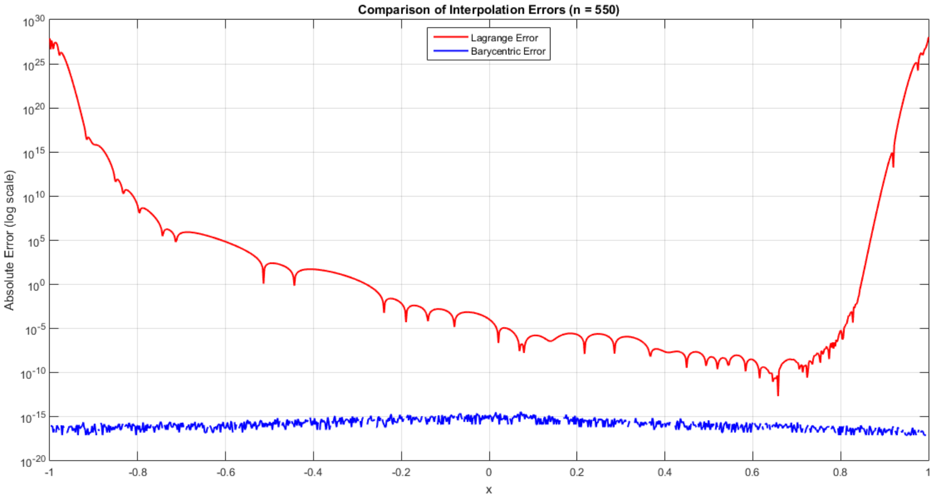


Figure 9: Zoomed comparison near $x = -1$ (left) and $x = 1$ (right) for interpolation of $g(x) = \frac{1}{1+25x^2}$, with $n = 550$ Legendre Gauss Lobatto nodes. Lagrange interpolation (red dashed) diverges significantly from the exact function (green), while the barycentric interpolant (blue dash-dot) maintains exceptional agreement, even at the endpoints.

Figure 10: To directly compare the two methods, we also plot both interpolation errors for $g(x) = \frac{1}{1+25x^2}$ in a single figure for $n = 550$, as shown below.



Example 3: We considered the rational algebraic function

$$g(x) = \frac{x^2 - 4}{x^2 + 1}$$

This smooth function is well behaved on $(-1, 1)$, making it useful for testing the stability and precision of interpolation methods without the influence of endpoint singularities or rapid oscillations.

The table below reports the maximum absolute errors of Lagrange and barycentric interpolants for various values of n. For small n, both methods achieve high accuracy, with barycentric interpolation consistently yielding smaller errors. As n increases, the error in the Lagrange method grows rapidly indicating numerical instability while the barycentric formula maintains low error, remaining near machine precision (around 10^{-14}) even for large n. This again highlights the superior stability and reliability of the barycentric scheme for large degree interpolation.

Table 3: Table for Error term.

Nodes(n)	Error term for Lagrange Interpolation	Error term for Barycentric formula
50	1.168×10^{-13}	4.885×10^{-15}
100	3.486×10^{-2}	5.773×10^{-15}
150	4.870×10^{7}	7.994×10^{-15}
200	3.264×10^{7}	7.550×10^{-15}
250	7.767×10^{10}	9.770×10^{-15}
300	4.154×10^{13}	1.288×10^{-14}
350	6.799×10^{15}	1.199×10^{-14}
400	2.622×10^{18}	1.377×10^{-14}
450	8.231×10^{20}	1.199×10^{-14}
500	9.168×10^{21}	1.377×10^{-14}
550	3.843×10^{23}	1.332×10^{-14}

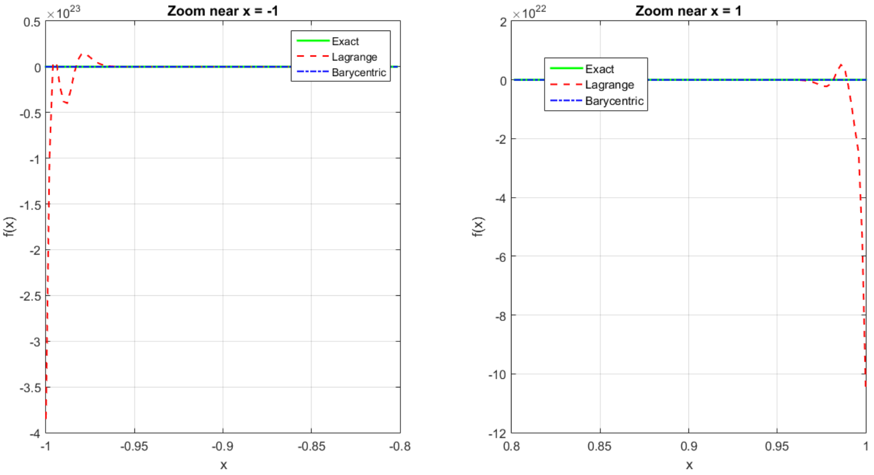
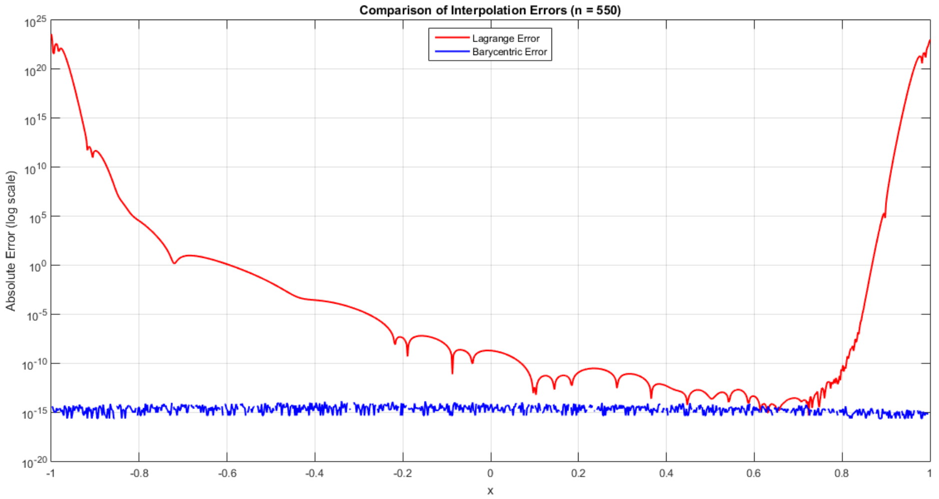


Figure 11: Close-up view of interpolation accuracy near $x = -1$ (left) and $x = 1$(right) for the function $g(x) = \frac{x^2-4}{x^2+1}$, interpolated using $n = 550$ Legendre Gauss Lobatto nodes. Lagrange interpolation (red dashed) deviates dramatically from the true function (green) near the endpoints, while the barycentric interpolant (blue dash-dot) closely approximates the true function.

Figure 12: To directly compare the two methods, we also plot both interpolation errors for $g(x) = \frac{x^2-4}{x^2+1}$ in a single figure for $n = 550$, as shown below.



Example 4: We examined the interpolation performance for the logarithmic-trigonometric function

$$g(x) = \log(\cos(x)),$$

which is smooth and well defined on the interval $(-1, 1)$, avoiding any singularities. This function serves as a useful benchmark for analyzing interpolation error when dealing with non-polynomial yet analytic functions.

The table below shows the maximum absolute errors for Lagrange and barycentric interpolation methods across increasing values of n. The barycentric method consistently maintains high accuracy, with errors staying near machine precision ($\sim 10^{-15}$) throughout. In contrast, the error in the Lagrange interpolation begins small but rapidly grows with increasing n, highlighting numerical instability. This trend again emphasizes the robustness of the barycentric formula, especially in high-degree interpolation.

Table 4: Table for Error term.

Nodes(n)	Error term for Lagrange Interpolation	Error term for Barycentric formula
50	3.109×10^{-15}	6.661×10^{-16}
100	8.548×10^{-3}	8.882×10^{-16}
150	1.012×10^{6}	1.332×10^{-15}
200	8.141×10^{5}	1.221×10^{-15}
250	3.444×10^{9}	1.554×10^{-15}
300	2.320×10^{12}	1.998×10^{-15}
350	1.837×10^{14}	1.332×10^{-15}
400	2.871×10^{16}	1.554×10^{-15}
450	2.781×10^{18}	1.832×10^{-15}
500	5.933×10^{20}	2.054×10^{-15}
550	2.157×10^{22}	1.332×10^{-15}

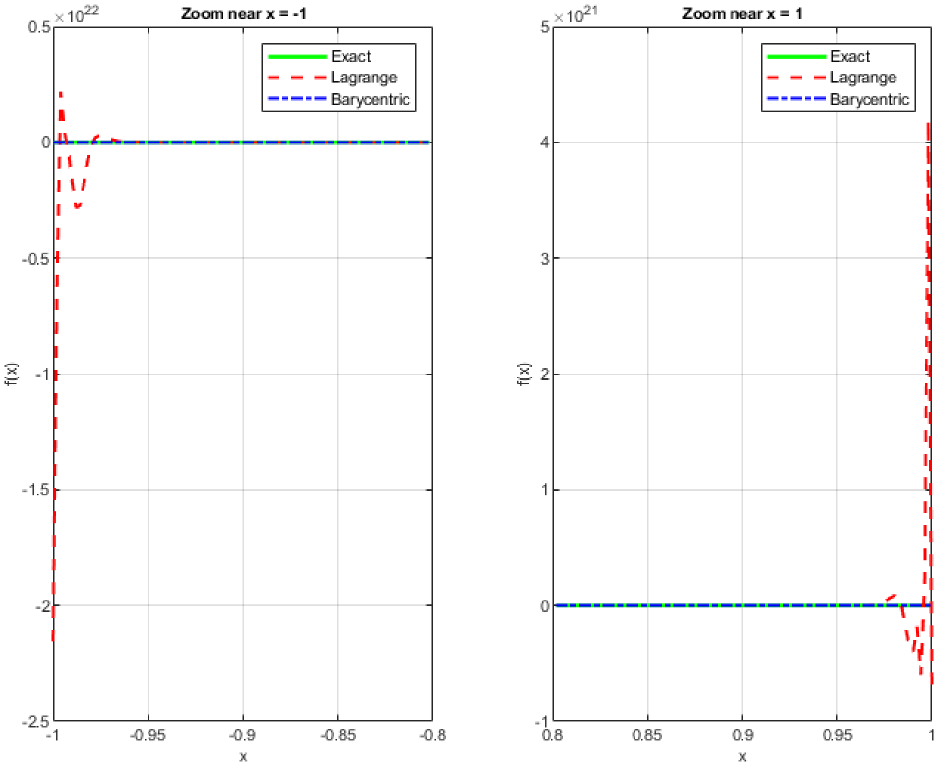
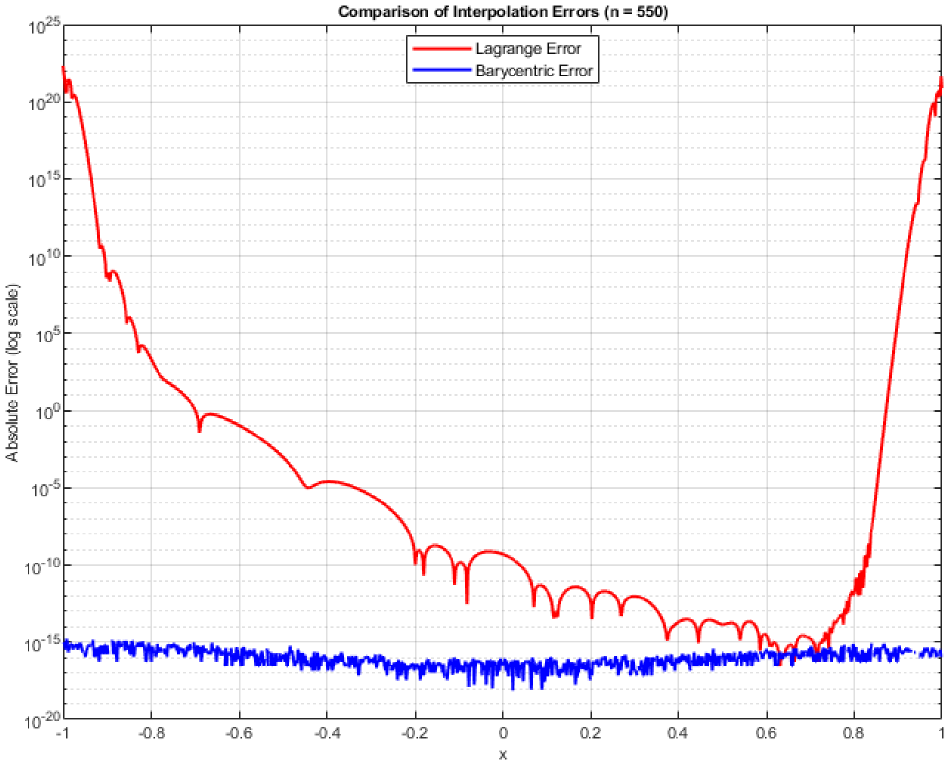


Figure 13: Zoomed view of interpolation behavior near the endpoints $x = -1$ (left) and $x = 1$ (right) for $n = 550$. The Lagrange interpolant (red dashed) diverges significantly from the true function (green), while the barycentric interpolant (blue dash-dot) tracks the exact function with high fidelity, demonstrating robustness near critical regions.

Figure 14: To directly compare the two methods, we also plot both interpolation errors for $g(x) = log(cos(x))$ in a single figure for $n = 550$, as shown below.



6. Conclusion

In this study, we compared the numerical behavior of classical Lagrange interpolation with the barycentric formulation across various test functions. Our results show that while both methods aim to approximate the same function, the stability and accuracy of their outputs differ significantly especially as the number of interpolation points increases.

The Lagrange method, though mathematically sound, suffers from severe numerical instability for high degrees. This is especially visible near the endpoints of the interpolation interval, where the absolute error grows exponentially(as shown in tables 1,2,3,4 and figures 1,5). On the other hand, barycentric interpolation consistently produces errors that remain very small and stable, even as n increases (as shown in tables 1,2,3,4 and figures 2,6). This improved stability is further evidenced by the zoomed-in views near $x = \pm 1$ (See figures 3,7,9, 11,13), which clearly reveal that the barycentric interpolant accurately follows the true function even in critical boundary regions, while the Lagrange interpolant shows significant deviation. To further emphasize the distinction between the two methods, we plotted both Lagrange and barycentric errors on the same graph for selected values of n (See figures 4,8, 10, 12, 14). These side by side comparisons vividly illustrate how barycentric interpolation maintains fidelity to the original function, whereas Lagrange interpolation diverges drastically, particularly at $x = \pm 1$. Overall, this investigation confirms that barycentric interpolation is not only more efficient but also significantly more robust for practical applications, especially when working with large data sets or requiring high accuracy. Given its simplicity, speed, and numerical reliability, barycentric interpolation should be preferred in most real world scenarios where polynomial interpolation is required.

References

[1] Ayman Mursaleen, M. (2025).*Quadratic function preserving wavelet type Baskakov operators for enhanced function approximation.* Computational and Applied Mathematics 44 (8): 395.

[2] Ayman-Mursaleen, M., M. Nasiruzzaman, and N. Rao. (2025). *On the Approximation of Szász Jakimovski-Leviatan Beta Type Integral Operators Enhanced by Appell Polynomials.* Iranian Journal of Science 49 (4): 1013-1022.

[3] Berrut, J.-P., Baltensperger, R. and Mittelmann, H. D. (2005). *Recent developments in barycentric rational interpolation, Trends and Applications in Constructive Approximation*, International Series of Numerical Mathematics, vol.151, Birkhäuser, Basel, pp. 27-51.

[4] Berrut, J.-P., and Trefethen, L. N. (2004). *Barycentric Lagrange Interpolation.* SIAM Review, 46, 501-517.

[5] Boyd, J. P. (1992).*Multipole Expansions and Cardinal Functions. Journal of Computational Physics.*

[6] Boyd, J. P. (2001).*Chebyshev and Fourier Spectral Methods. Dover Publications.*

[7] Brutman, L. (1997).*Lebesgue Functions for Polynomial Interpolation: A Survey. Annals of Numerical Mathematics*, 4, 111-127.

[8] Davis, P. J. (1975).*Interpolation and Approximation.* Dover Publications.

[9] Dutt, A., Gu, M., and Rokhlin, V. (1996).*Fast algorithms for polynomial interpolation, integration, and differentiation*, SIAM J. Numer. Anal., 33 (1996), pp. 1689-1711.

[10] Epperson, J. F. (1987).*On the Runge Example. The American Mathematical Monthly*, 94(4), 329-341.

[11] Floater, M. S., and Hormann, K. (2007).*Barycentric Rational Interpolation with No Poles and High Rates of Approximation*. TU Clausthal Technical Report.

[12] Gautschi, W. (2004). Orthogonal Polynomials:*Computation and Approximation*. Oxford University Press.

[13] Gevorgyan, L. Z. (2020).*On the General Rational Interpolation*. Proceedings of the NPUA Annual Conference.

[14] Henrici, P. (1979).*Barycentric Formulas for Interpolating Trigonometric Polynomials*. Numerische Mathematik, 33(3), 225-234.

[15] Henrici, P. (1982). *Essentials of Numerical Analysis*. Wiley.

[16] Higham, N. J. (2004).*The Numerical Stability of Barycentric Lagrange Interpolation*. IMA Journal of Numerical Analysis, 24(4), 547-556.

[17] Hormann, K. (2011). *Barycentric Rational Interpolation*. Numerische Mathematik, 117(2), 333-342.

[18] Hormann, K. (2014).*Barycentric Interpolation*. Springer Proceedings in Mathematics and Statistics, Vol. 83.

[19] Horváth, A. P. and Vertesi P. (2016). *On Barycentric Interpolation. II (Grünwald-Marcinkiewicz type Theorems)* Acta Math Hung., 148, 147-156.

[20] Jackson, D. (1911).*Ueber die Genauigkeit der Annäherung stetiger Funktionen durch ganze rationale Funktionen gegebenen Grades und trigonometrische Summen gegebener Ordnung, Göttingen*.

[21] McMullen, C. (2017). *Advanced Complex Analysis*. Harvard Lecture Notes.

[22] Rani, M., Rao N. and Malik P. (2022). *Generalized Bivariate Baskakov Durrmeyer Operators and Associated GBS Operators.* FILOMAT 36 (5): 1539-1555.

[23] Rao, N., Farid, M., Raiz, M. (2025). *On the Approximations and Symmetric Properties of Frobenius-Euler-Simsek Polynomials Connecting Szász Operators.* Symmetry 2025, 17, 648.

[24] Rao N., and Gulia K. (2023). *Ontwodimensional Chlodowsky-Szász operators via Sheffer Polynomials.* Palestine J. of Mathematics 12 (3), 521-531.

[25] Rao, N. and Malik P. (2023). *α-Baskakov-Durrmeyer type operators and their approximation properties.* Filomat 37 (3), 935-948.

[26] Rao, N., M. Farid, and N. K. Jha. (2025). *Szász-integral operators linking general-Appell polynomials and approximation.* AIMS Mathematics 10 (6): 13836-13854.

[27] Rivlin, T. J. (1974).*The Chebyshev Polynomials*. Wiley, New York.

[28] Runge, C. (1901).*Über empirische Funktionen und die Interpolation zwischen äquidistanten Ordinaten*. Zeitschrift für Mathematik und Physik.

[29] Rutishauser, H. (1990).*Lectures on Numerical Mathematics*, M. H. Gutknecht etal., eds., Birkhâuser, Basel.

[30] Salazar Celis, O. (2008). *Practical Rational Interpolation of Exact and In exact Data: Theory and Algorithms*,Ph.D.thesis, University of Antwerp, Antwerp, Belgium.

[31] Salzer, H. E. (1982).*Lagrangian Interpolation at Chebyshev Points*. Essentials of Numerical Analysis, Wiley, New York, 1982

[32] Schneider, C., and Werner, W. (1986).*Some New Aspects of Rational Interpolation*. Mathematics of Computation, 47(175), 285-299.

[33] Szegö, G. (1939) *Orthogonal Polynomials*, American Mathematical Society Colloquium Publications, Vol. 23.

[34] Taylor, W. (1945).*Method of Lagrangian curvilinear interpolation*, J.Res.Nat.Bur.Stand., 35 (1945),pp.151-155.

[35] Trefethen, L. N. (2008). *Is Gauss Quadrature Better than Clenshaw-Curtis?* SIAM Review, Vol 50, Iss.1.

[36] Trefethen, L. N. (2012).*Approximation Theory and Approximation Practice*, SIAM, Philadelphia.

[37] Vertesi, P. (2015)*On barycentric interpolation. I. (On the T-Lebesgue function and T-Lebesgue constant)* Acta Math Hung., 147, 396-407.

[38] Yawar, A., and Bahadur, S.*Error Estimates of Barycentric Lagrange Interpolation*. Applications and Applied Mathematics: An International Journal (AAM), Volume 20, Issue 2, December 2025, Article 7.

[39] Yawar, A., and Bahadur, S. *Lagrange Interpolation with Hermite condition at an Interior Point*. GANITA, Volume 75, Issue 2, 2025.