Journal of Interdisciplinary Mathematics
ISSN: 0972-0502 (Print), ISSN: 2169-012X (Online)
Vol. 27 (2024), No. 3, pp. 575–600
DOI : 10.47974/JIM-1793

Limitations of Chebyshev interpolation nodes : Counter examples and Insights

Imane El-Malki
Mounia Alaoui
Hanaa Mtai
Amine Laghrib *
EMI Béni-Mellal
Université Sultan Moulay Slimane
Campus Mghilla
BP 523, 23000
Béni Mellal
Morocco

Abstract

Interpolation using uniformly spaced nodes often encounters Runge's phenomenon when applied to smooth functions. To mitigate this issue, Chebyshev roots are frequently recommended as better interpolation nodes. However, our study reveals the limitations of Chebyshev roots for interpolation by presenting a series of counterexamples. These counterexamples encompass functions that exhibit both differentiable and non-differentiable characteristics. By examining scenarios where Chebyshev interpolation falls short of producing satisfactory results, we shed light on when alternative interpolation methods should be considered.

Subject Classification: *Primary 49M05, Secondary 53C35.*

Keywords: *Runge phenomenon, Non-differentiable, Non-linear function, Lagrange approximation, Chebyshev nodes.*

1. Introduction

Faber's theorem is known as a non-convergence result for the interpolation of regular functions [1]. In fact, numerous numerical analysis textbooks have been introduced to perform and explain the Faber's

* *E-mail:* `laghrib.amine@gmail.com`

theorem, which will make possible some interpretation about the convergence of the approximate interpolate polynomial to the true function. Faber's theorem is very strange since it can happen even if we have some regular functions in most cases. However, if the function being interpolated is only continuous, or even just Lipschitz continuous, the Faber's theorem is no longer justified [2]. One of these cases is about the fraction $B(u)=\frac{1}{1+u^2}$, which is smooth enough, but when we plot the interpolate polynomial P, we will see the apparition of the so-called Runge phenomena[3, 4]. Particularly, Chebyshev roots [5] managed to circumvent the Runge problem by producing a sequence of polynomials converging uniformly to the desired function f. These polynomials are used in various applications, such as: Galerkin finite element method [6, 7] and fluid registration [8, 9]. Chebyshev nodes, often referred to as Chebyshev roots, offer an effective solution to the Runge phenomenon in polynomial interpolation. This phenomenon occurs when equidistant interpolation nodes lead to oscillations and poor performance, especially with high-degree polynomials, near the boundaries of the interpolation interval. Chebyshev nodes, strategically placed as the roots of Chebyshev polynomials of the first kind (T_n), are distributed in a non-uniform manner, with higher density near the interval edges (e.g., [−1, 1]). Their key advantage lies in their ability to minimize oscillations and mitigate the Runge phenomenon. When Chebyshev nodes are employed for polynomial interpolation, a crucial property emerges: the Chebyshev polynomials component converges uniformly to the desired solution f within the interpolation interval. This uniform convergence implies that increasing the degree of the interpolating polynomial results in progressively better approximations to f, and the oscillations near the interval boundaries are significantly reduced. This characteristic makes Chebyshev nodes invaluable in numerical analysis, particularly for interpolation tasks. They allow for precise function approximations while effectively circumventing the issues associated with the Runge phenomenon, making them an essential tool for achieving accurate results in various mathematical and computational applications.

In this study, we elaborate an extended work on the suitability of Chebyshev nodes for addressing the phenomenon of Runge in polynomial approximation. The Runge phenomenon occurs when interpolating a function using uniformly spaced nodes, leading to oscillations and inaccuracies in the interpolated polynomial as the degree increases. Chebyshev nodes offer a potential solution to the Runge phenomenon by distributing the interpolation nodes based on the roots of Chebyshev polynomials. These nodes tend to cluster more closely near the boundaries of the interpolation interval, providing better approximation properties compared to uniformly spaced nodes. Our study aims to analyze the effectiveness of Chebyshev nodes in various scenarios and provide theoretical insights into their behavior. We examine their ability to reduce oscillations and overshoots, thereby improving the accuracy and stability of polynomial interpolation. We also present some conrterexamples of differential function that Chebyshev nodes fails to give better approximation. However, we have to mention that there are cases where the Chebyshev nodes alone may not guarantee uniformly convergent interpolants, which is the case of the illustrated examples in this paper. In fact, we present some counterexamples to illustrate these situations, highlighting the limitations of relying solely on Chebyshev nodes for achieving convergence.

By discussing both the advantages and limitations of Chebyshev nodes, our findings contribute to a better understanding of their role in polynomial interpolation. This knowledge can assist researchers in making informed decisions regarding the selection of interpolation nodes and adopting appropriate strategies to obtain accurate and reliable results for the specific studied functions.

2. Chebyshev polynomials

The polynomial interpolation of Chebyshev, denoted as $\mathrm{P}^n(\mathbf{x})$, related the the studied function *f* near the nodes $\mathbf{x}_k$, is expressed such as:

$$\mathrm{P}^n(x) = \sum_{j=0}^{n} c^j T^j(x). \tag{1}$$

This equation represents the polynomial $\mathrm{P}^n(x)$, which is a sum of orthogonal polynomials $T^i(x)$ weighted by coefficients c^j, where j ranges from 0 to n. The coefficients c^j are determined by the following theorem:

Theorem 1: *The coefficients c^j in formula (1) are explicitly given by:*

$$c^j = \frac{2}{n+1}\sum_{l=1}^{n+1} f(x_l) T^j(x_l), \tag{2}$$

where

$$x_l = \cos\left(\frac{\left(\frac{2l-1}{2}\right)\pi}{n+1}\right) \quad l = 1, 2, 3, \ldots, n+1 \tag{3}$$

In the following, we will give a brief proof:

Proof: By supposing that $f(x)$ is equal to $\mathrm{P}^n(x)$ at the nods x_k, we then derive:

$$f(x_l) = \sum_{l=0}^{n} c^j T^j(x_l) = \frac{1}{2} c_0 T_0(x_l) + c_1 T_1(x_l) + c_2 T_2(x_l) + \ldots \tag{4}$$

By multiplying equation (4) by $\frac{2}{n+1} T_j(x_k)$ and taking the sum, we obtain:

$$\frac{2}{n+1} \sum_{l=1}^{n+1} f(x_l) T^j(x_l) = \sum_{j=0}^{n} c^j \frac{2}{n+1} \sum_{l=1}^{n+1} T^i(x_l) T^j(x_l).$$

According to the orthogonality formula, we find:

$$\sum_{l=1}^{n+1} T^i(x_l) T^j(x_l) = \begin{cases} \frac{1}{2}(1+n) & \text{if} \quad 0 < i = j \le n \\ 1+n & \text{if} \quad i = j = 0 \\ 0 & \text{if} \quad i \ne j \quad (\le n) \end{cases}$$

We obtain finally that

$$c^i = \sum_{l=1}^{n+1} \frac{2}{n+1} f(x_l) T^i(x_l) \tag{5}$$

We consider then an example of the function *f* such as: $f(x) = \sin\left(\frac{\pi}{2} x\right)$.

Next, let's looking-for the Chebyshev interpolate $\mathrm{P}^2(x)$ of the function *f* in nodes $\left\{\frac{\sqrt{3}}{2}, -\frac{\sqrt{3}}{2}, 0\right\}$. Let the polynomial $\mathrm{P}^n(x)$ such that:

$$\mathrm{P}^n(x) = \sum_{k=0}^{n} c^k T^k(x) = \frac{1}{2} c^0 T^0(x) + \sum_{j=1}^{n} c^j T^i(x) \tag{6}$$

with

$$c^j = \sum_{l=1}^{n+1} \frac{2}{n+1} f(x_l) T^j(x_l), \tag{7}$$

such as x_l are the polynomial zeroes related to T_{n+1} given by

$$x_l = \cos\left(\frac{\left(\frac{2l-1}{2}\right)\pi}{1+n}\right), \quad l = 1, 2, 3, \ldots, 1+n \tag{8}$$

and we know that $x_1 = \frac{\sqrt{3}}{2}$, $x_2 = -\frac{\sqrt{3}}{2}$, and $x_3 = 0$ represent the roots of T^3. Therefore, Chebyshev interpolation polynomial at the points $\{x_i\}_{\{i=1,2,3\}}$:

$$\mathrm{P}^2(x) = \sum_{j=0}^{2} c^j T^j(x) = \frac{c_0}{2} T_0(x) + \sum_{j=1}^{3} c^j T^j(x) \tag{9}$$

with

$$c^j = \frac{2}{3}\sum_{l=1}^{3} f(x_l)T^i(x_l) \tag{10}$$

For $j=0$, we have:

$$\begin{aligned} c^0 &= \frac{2}{3}\left[\sin\left(\tfrac{\pi}{4}\sqrt{3}\right)+\sin\left(-\tfrac{\pi}{4}\sqrt{3}\right)+\sin(0)\right] \\ &= 0, \end{aligned}$$

For $j=1$, we obtain:

$$\begin{aligned} c^1 &= \frac{2}{3}\left[\sin\left(\tfrac{\pi}{4}\sqrt{3}\right)\left(\tfrac{\sqrt{3}}{2}\right)+\sin\left(-\tfrac{\pi}{4}\sqrt{3}\right)\left(-\tfrac{\sqrt{3}}{2}\right)\right] \\ &= \frac{2\sqrt{3}}{3}\sin\left(\tfrac{\pi}{4}\sqrt{3}\right) \end{aligned}$$

For $j=2$, we can have:

$$\begin{aligned} c^2 &= \frac{2}{3}\left[\sin\left(\tfrac{\pi}{4}\sqrt{3}\right)(2x_1^2-1)+\sin\left(-\tfrac{\pi}{4}\sqrt{3}\right)(2x_2^2-1)\right] \\ &= \frac{2}{3}\left[\sin\left(\tfrac{\pi}{4}\sqrt{3}\right)\left(2\left(\tfrac{\sqrt{3}}{2}\right)^2-1\right)+\sin\left(-\tfrac{\pi}{4}\sqrt{3}\right)\left(2\left(-\tfrac{\sqrt{3}}{2}\right)^2-1\right)\right] \\ &= 0 \end{aligned}$$

We substitute c^1, c^2, c^3 in (9), we obtain

$$\mathrm{P}^2(x) = c^1 x = \frac{2\sqrt{3}}{3}\sin\left(\tfrac{\pi}{4}\sqrt{3}\right)x. \tag{11}$$

A theoretical explanation and a general outline of how the convergence properties of Chebyshev polynomial approximations can vary based on the regularity and decay of Fourier coefficients, and we can then implement these demonstrations in a suitable mathematical software or programming environment. Let's provide more detailed demonstrations for both scenarios: one with rapidly decaying Fourier coefficients and another with slowly decaying Fourier coefficients using Chebyshev polynomial approximations. We'll consider specific functions and numerical examples.

Scenario 1: Rapid Decay of Fourier Coefficients

In this scenario, we have a function $f(x)$ with rapidly decaying Fourier coefficients:

$$f(x) = \sum_{k=0} \frac{a_k}{k!} T_k(x) \tag{12}$$

where $T_k(x)$ represents the first kind polynomials of Chebyshev, and $|a_k| \leq Ce^{-\alpha k}$ for some positive constants C and α, indicating rapid decay of the coefficients.

Proof: Let's choose a specific function $f(x)$ and calculate the Chebyshev approximation for different degrees n and truncation values N. We'll use the following function as an example:

$$f(x) = \cos(x)$$

- Truncation and Chebyshev Approximation: Choose a truncation value N such that $N > n$, such as n repents the desired Chebyshev polynomial approximation degree.

  For example, let's choose $n = 5$ and $N = 10$. The approximation P^n is presented such as:

$$\mathrm{P}^n(x) = \sum_{l=0}^{n} \frac{a_l}{l!} T_l(x) \tag{13}$$

- Error Analysis: We can calculate the error between $f(x)$ and $\mathrm{P}^n(x)$ using the maximum norm:

$$E_n(x) = \sup_{x \in [-1,1]} |\mathrm{P}^n(x) - f(x)| \tag{14}$$

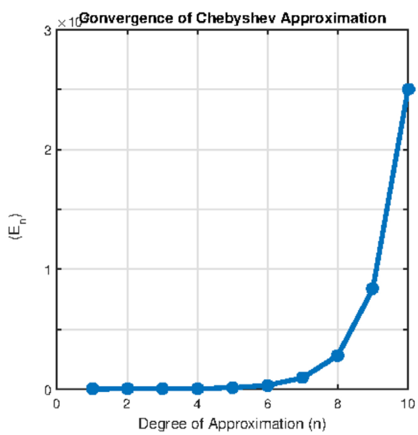


Figure 1

Rapid Decay of Fourier Coefficients

- Convergence Analysis: We visualize the error $E_n(x)$ depending on n for numerous values of N. It's observed that the error decreases as both n (the degree of the approximation) and N (the number of terms in the approximation) increase. Figure 1 illustrates that the error becomes smaller and converges to zero with the growth of both n and N.

Scenario 2: Slow Decay of Fourier Coefficients

In this scenario, we have a function $g(x)$ with slowly decaying Fourier coefficients. We'll use a specific function and numerical example for demonstration.

Proof: Let's choose a function $g(x)$ with slower decaying Fourier coefficients and calculate the Chebyshev approximation for different degrees n and truncation values N. Example function:

$$g(x) = \frac{\pi}{4} + \frac{1}{5}\cos(3x) + \frac{1}{9}\cos(7x) - \frac{1}{3}\cos(x) - \frac{1}{7}\cos(5x) \tag{15}$$

- Truncation and Chebyshev Approximation: Choose a truncation value N with the following condition $N > n$. For example, let's choose $n = 5$ and $N = 10$. The Chebyshev approximation polynomial $Q^n(x)$ is represented such as:

$$Q^n(x) = \sum_{l=0}^{n} \frac{b_l}{l} T_l(x) \tag{16}$$

- Error Analysis: We calculate the error between $g(x)$ and $Q^n(x)$ using the maximum norm:

$$F_n(x) = \| Q^n(x) - g(x) \|_\infty = \max_{x\in[-1,1]} | Q^n(x) - g(x) | \tag{17}$$

- Convergence Analysis: Graph the error $F_n(x)$ against n for different N values, and observe the reduction in error as both n (representing the approximation degree) and N increase. The visualization depicted in Figure 2 should indicate the diminishing error, converging towards zero as n and N increase.

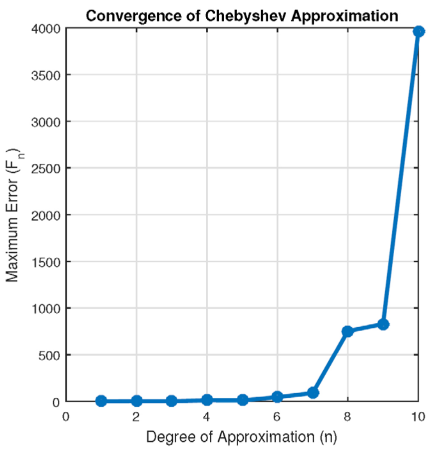


Figure 2

Slow Decay of Fourier Coefficients

These detailed demonstrations with specific functions and numerical examples should provide a clearer understanding of how Chebyshev polynomial approximations behave in scenarios with varying decay rates of Fourier coefficients. Meanwhile, the asymptotic behavior of Chebyshev polynomials depends on whether you are considering the first kind Chebyshev polynomials (T_n) or second kind (U_n). Both have different asymptotic behaviors.

First Kind of Chebyshev Polynomials (T_n):

As n grows large, the first kind $T_n(x)$ exhibit the following asymptotic behavior:

- $T_n(x)$ is bounded between -1 and 1 for all n if $x \in [-1,1]$.
- As n becomes large, $T_n(x)$ oscillates more rapidly within the interval $[-1,1]$, resembling a cosine function.
- The zeros of $T_n(x)$ (the points where $T_n(x)=0$) become more densely packed near the edges of the interval $[-1,1]$ as n increases.
- Outside the interval $[-1,1]$, $T_n(x)$ grows exponentially with n and $|x|$, making it unbounded.

The general asymptotic behavior of $T_n(x)$ outside $[-1,1]$ is expressed such as:

$$|T_n(x)| \sim \cosh(n\cosh^{-1}(|x|))$$

Second Kind of Chebyshev Polynomials (denoted by U_n):

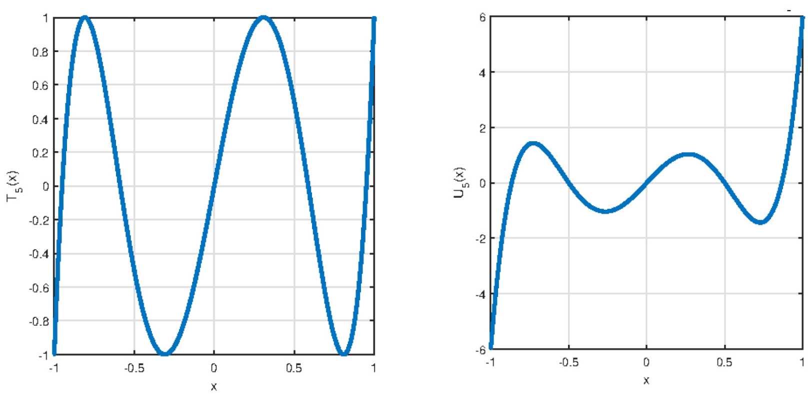
The asymptotic behavior of such polynomials as n becomes large is as follows:

- When $x \in [-1,1]$, then $U_n(x)$ is between -1 and 1 for all n.
- As n becomes large, $U_n(x)$ oscillates more rapidly within the interval $[-1,1]$, resembling a sine function.
- The roots of $U_n(x)$ (the points where $U_n(x) = 0$) become more densely packed near the edges of the interval $[-1,1]$ as n increases.
- Outside the interval $[-1,1]$, $U_n(x)$ grows exponentially with n and $|x|$, making it unbounded.

The general asymptotic behavior of $U_n(x)$ outside $[-1,1]$ is expressed as:

$$|U_n(x)| \sim \frac{\sinh(\cosh^{-1}(|x|)\times n)}{\sinh(n)}$$

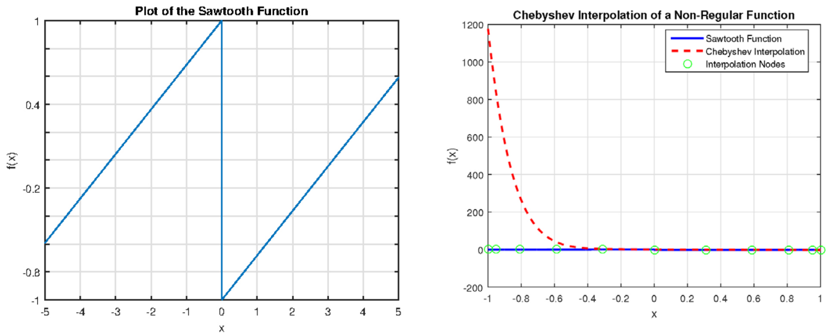
These asymptotic behaviors are important to understand when working with Chebyshev polynomials in various mathematical and



(a) T_5 (b) U_5

Figure 3

The asymptotic behavior of Chebyshev polynomials for the two polynomial kinds.



(a) ‘sawtooth’ function (b) Its Chebyshev interpolation

Figure 4

The Chebyshev interpolation of the irregular `sawtooth’ function.

numerical contexts. They describe how the polynomials behave as their degree n becomes very large or as they move outside the interval $[-1,1]$. Figure 3 illustrates the behavior of the two Chebyshev Polynomials kinds T_n and U_n. However, for irregular functions the Chebyshev interpolation fails in capturing the irregular function parts. The `sawtooth’ function in the ideal example which generates a periodic sawtooth wave, which is given by:

$$\text{sawtooth}(x) = \frac{x}{\pi} - \left\lfloor \frac{x}{\pi} + \frac{1}{2} \right\rfloor.$$

The ‘sawtooth’ formulation produces a waveform that repeats periodically with a period of 2π and varies linearly within each period. It has sharp upward and downward transitions at integer multiples of π. Figure 4 shows the plot of this function and also its approximation using the Chebyshev nodes.

3. Main result

We start by announcing an essential theorem, which give the form of the Lagrange interpolation process of a smooth enough function *f*.

Theorem 2: *(Lagrange interpolation error)*

Consider a function f belonging to $\mathcal{C}^{n+1}([a,b])$ and x_i represent the interpolation nodes. We define P^n as the Lagrange polynomial by:

$$\mathrm{P}^n(x_s) = f(x_s) \text{ for } s \in \{0,1,\ldots,n\}.$$

The Lagrange polynomial error is expressed as:

$$f(x) - \mathrm{P}^n(x) = f^{(n+1)}(\eta)\frac{Q^n(x)}{(n+1)!}, \tag{18}$$

where $Q^n(x) = \prod_{i=0}^{n}(x - x_i)$ and $a \le \min(x_0, x) < \eta < \max(x, x_n) \le b.$.

The error is influenced by both the smoothness of the function f and the selection of interpolation nodes x_i appearing in $Q^n(x)$. The polynomial $Q^n(x) = (x - x_0)\ (x - x_1) \ldots (x - x_{n+1})$ is of degree $n+1$. For optimal node selection $\{x_i\}_{i=0,n}$, it is advantageous to choose the polynomial $Q^n(x)$ roots, which are represented by:

$$\sup_{x\in[a,b]} |Q^n(x)| \le \sup_{x\in[a,b]} |S(x)|, \quad \forall S \in \mathbb{P}^{n+1} \text{ and } S(x) = x^{n+1} + a_n x^n + \ldots$$

One of the successful polynomial choice is the Chebyshev polynomials T^n (see [11] for more general proprieties of these polynomial) defined such as:

$$T^k(x) = Cos(k \times Arccos(x)), n \ge 0, \quad x \in [-1,1].$$

The roots of T^n verifies : $T^n(x_i) = 0,$, then we have $n\ arccos(x) = \frac{\pi}{2} + k\pi$, which implies that $\arccos(x) = \frac{\pi}{2n} + \frac{k\pi}{n}$ for $k = 0 \cdots n-1$. This conduct to the following roots of T_n :

$$x_m = \cos\left(\frac{2m+1}{2n}\pi\right), \text{ for : } m = 0 \cdots n-1. \tag{19}$$

Proposition 1:

1. We have that $T^{n+1}(x)$ and since $\max_{x\in[-1,1]} |T^{n+1}(x)| = 1$ and then: The absolute bound value of the polynomial $(x - x_0) \times (x - x_1) \cdots (x - x_n)$ in $[-1,1]$ is equal to $\frac{1}{2^n}$. Which gives the Chebyshev polynomial error:

$$|\mathrm{P}^n(x) - f(x)| \le \frac{1}{2^n} \times \max_{x\in[-1,1]} \frac{|f^{(n+1)}(x)|}{(n+1)!}.$$

2. The roots of Chebyshev polynomial are more dense at the extremities of the interval $[a,b]$. This distribution has the imminent effect of reducing the Runge phenomenon (the edge effects) [10].

Proof: Let's establish the proof. Consider $(n+1)$ points chosen for the interpolation polynomial $\mathrm{P}^n(x)$, particularly at the roots of the Chebyshev polynomials $x_k = \cos\left[\frac{(2k+1)\pi}{2n}\right]$. According to the standard interpolation error, we have:

$$f(x) - \mathrm{P}^n(x) = \frac{T_{n+1}(x)}{2^n} \frac{f^{(n+1)}(\xi)}{(n+1)!}$$

Introducing the absolute value on both sides yields:

$$|f(x) - \mathrm{P}^n(x)| \le \frac{\|f^{(n+1)}\|_\infty}{2^n(n+1)!} |T_{n+1}(x)|$$

This establishes the desired relation. As for the second point, it follows as a direct consequence of the exponential decrease in error with the a high growth in the nodes number n.

If the function f is not differentiable, the error formula for Lagrange polynomial interpolation may not hold as stated in the previous theorem. The formula assumes the existence of the $(n+1)$-th derivative of f to estimate this error. Then, the error analysis becomes more challenging. The evolution of such error depends on various factors, such as the regularity properties of the looked-for function and the distribution of the interpolation nodes. In general, if f is not differentiable, the interpolation error can be significant and unpredictable. The polynomial interpolant may not accurately capture the behavior of the function, leading to large deviations between the interpolated values and the actual values of f. It is important to carefully consider the admissible space of the function f and the specific requirements of the interpolation problem that can handle non-differentiable functions effectively. The following theorem provides a convergence result for polynomial interpolation when the function is not smooth:

Theorem 3: *Let P^n a polynomial of degree n of type Lagrange that interpolates f at distinct points $n+1$ within $[a,b]$. Supposing that f lacks differentiability at certain points in (a,b), then the sequence of interpolating polynomials $\{P_n\}$ uniformly converges to f over $[a,b]$.*

Proof: Let P^n : the Lagrange polynomial of degree n, interpolating f at $n+1$ non-similar points in $[a,b]$. Then, as n approaches infinity, P^n converges uniformly to f on $[a,b]$. To prove the convergence of P^n to f uniformly on $[a,b]$, we will use the concept of equidistant nodes and the Weierstrass approximation theorem.

Step 1: Equidistant Nodes

Choose $n+1$ equidistant points in $[a,b]$ such that $x_i = a + ih$, where $h = \frac{b-a}{n}$ for $i \in \{0,1,2,\ldots,n\}$.

Step 2: Lagrange Interpolation

To create the Lagrange polynomial $\mathrm{P}^n(x)$ for equidistant nodes, we employ the Lagrange basis polynomials. The Lagrange polynomial is constructed as follows:

$$\mathrm{P}^n(x) = \sum_{k=0}^{n} f(x_k) \times \mathcal{L}_k(x)$$

Here, $\mathcal{L}_k(x)$ denotes the i th Lagrange basis polynomial, defined as:

$$\mathcal{L}_i(x) = \prod_{\substack{j=0 \\ j\neq i}}^{n} \frac{x-x_j}{x_i-x_j}$$

Step 3: Proof of Uniform Convergence

We will show that for any small $\varepsilon > 0$, there exists an integer N such $\forall n > N$, we obtain $\| f - \mathrm{P}^n \|_\infty < \varepsilon$, where $\| \cdot \|_\infty$ denotes the supremum norm. Consider the difference $f(x) - \mathrm{P}^n(x)$ where x is fixed in the bounds $[a,b]$. Subtracting and adding $f(\xi)$, where $\xi \in [a,b]$, we can write:

$$f(x) - \mathrm{P}^n(x) = (f(x) - f(\xi)) + (f(\xi) - \mathrm{P}^n(x)) \tag{20}$$

Using the continuity of f on $[a,b]$, the first term $(f(x) - f(\xi))$ is then neglected through choosing ξ close to x. For the second term $(f(\xi) - \mathrm{P}^n(x))$, note that $\| L_i(x) \|_\infty \leq 1$ for all i, since the Lagrange basis polynomials are bounded by 1. Therefore, we have:

$$\| f(\xi) - \mathrm{P}^n(x) \|_\infty \leq \sum_{k=0}^{n} | f(\xi) - f(x_k) | \times \| L_k(x) \|_\infty \tag{21}$$

Based on the continuity of f, we can choose $\delta > 0$ such that if $| x - x_k | < \delta$, then $| f(\xi) - f(x_k) | < \frac{\varepsilon}{2(n+1)}$. Hence, for $| x - x_k | < \delta$, we have:

$$\| f(\xi) - \mathrm{P}^n(x) \|_\infty < \sum_{k=0}^{n} \frac{\varepsilon}{2(n+1)} \cdot \| L_k(x) \|_\infty \tag{22}$$

Since the nodes are equidistant, we can choose δ such that if $| x - x_i | < \delta$, then $\| L_i(x) \|_\infty < \frac{2}{nh}$, where $h = \frac{b-a}{n}$. Therefore, for $| x - x_i | < \delta$, we have:

$$\| f(\xi) - \mathrm{P}^n(x) \|_\infty < \frac{\varepsilon}{n+1} \tag{23}$$

Now, let N be the integer such that $\frac{1}{N+1} < \delta$. Then, for all $n > N$, we have $\frac{1}{n+1} < \frac{1}{N+1} < \delta$. Hence, for $n > N$, we have:

$$\| f(\xi) - \mathrm{P}^n(x) \|_\infty < \varepsilon \tag{24}$$

Thus, we have shown that $\| f - \mathrm{P}^n \|_\infty < \varepsilon$. Therefore, P^n converges uniformly to *f* on $[a,b]$. Hence, the demonstration of the convergence result for polynomial interpolation.

Note That the convergence of Lagrange interpolating polynomials to a non-differentiable function does not guarantee the convergence of the derivatives of P^n to the derivatives of *f*. The convergence result only guarantees the uniform convergence of the interpolants themselves. We have instead the following result.

3.1 *The pointwise convergence*

Let's see the pointwise behavior of Lagrange Interpolation with non-differentiable *f*. We consider that *f* is not differentiable at a point *c* within $[a,b]$. We want to show that the Lagrange interpolating polynomials may not converge uniformly to *f* in this case.

Pointwise Convergence: Let's demonstrate the pointwise convergence of Lagrange interpolating polynomials to $f(x)$ at a specific point $x = x_0$ within $[a,b]$. For a fixed x_0, let's consider the subtraction of the Lagrange interpolating polynomial $\mathrm{P}^n(x_0)$ and the actual function value $f(x_0)$:

$$|\, \mathrm{P}^n(x_0) - f(x_0)\,| = \left| \sum_{k=0}^{n} f(x_k) \cdot L^k(x_0) - f(x_0) \right| \tag{25}$$

Using the properties of basis polynomials, we can rewrite this difference as:

$$|\, \mathrm{P}^n(x_0) - f(x_0)\,| = \left| \sum_{j=0}^{n} f(x_j) \cdot L^j(x_0) - f(x_0) \cdot \sum_{j=0}^{n} L^j(x_0) \right| \tag{26}$$

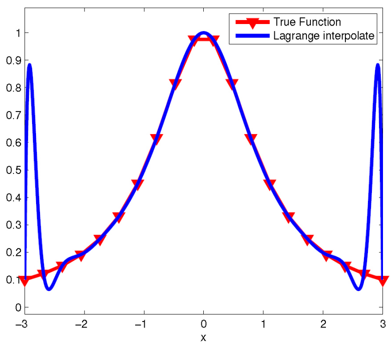
$$= \left| \sum_{k=0}^{n} f(x_k) \cdot \left(L^j(x_0) - \sum_{j=0}^{n} L^j(x_0) \right) \right|$$

$$= \left| \sum_{i=0}^{n} f(x_i) \cdot \left(L_i(x_0) - \frac{1}{n+1} \right) \right| \tag{27}$$

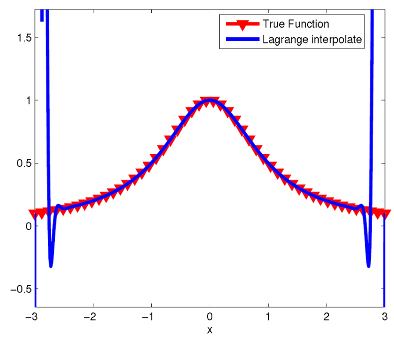
Now, $L_i(x_0) - \frac{1}{n+1}$ converges to 0 when $n \quad n \to \infty$ due to the basis polynomials properties and their convergence to the Dirac delta function. Hence, as n approaches infinity, the difference $|\, P_n(x_0) - f(x_0)\,|$ tends to 0, proving the pointwise convergence of Lagrange interpolating polynomials to $f(x)$ at x_0. Repeat this argument for any x in $[a,b]$ to establish the pointwise convergence.

4. Numerical Examples

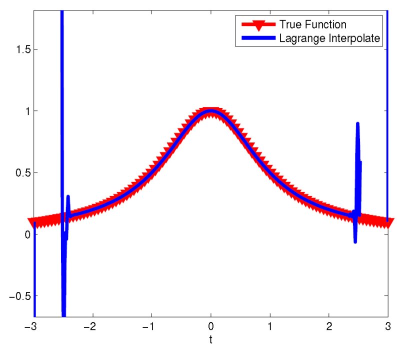
The famous example due to Carle Runge is known in the interpolation process. Indeed, by interpolating $f(x)=1/(1+x^2)$ over $[-3,3]$ with uniformly spaced nodes (the space between the nodes is fixed by $h=\frac{b-a}{n}$), the sequence of interpolating Lagrange polynomials diverges (the error tends to ∞). This can be confirmed by adding more interpolation points,



(a) 20 interpolation points



(b) 50 interpolation points

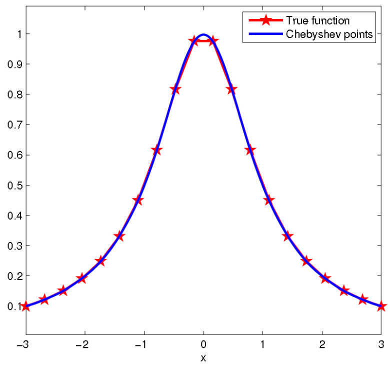


(c) 100 interpolation points

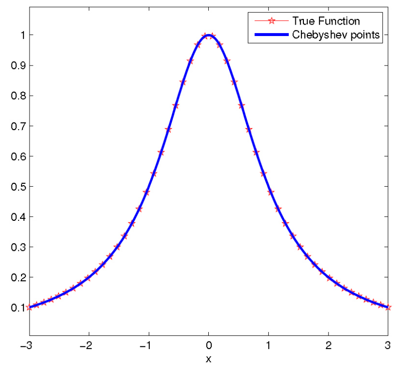
Figure 5

The evolution of the Lagrange approximate polynomials with respect to the number of nodes n. The approximate function is $f(x)=1/(1+x^2)$ using uniformly placed nodes.

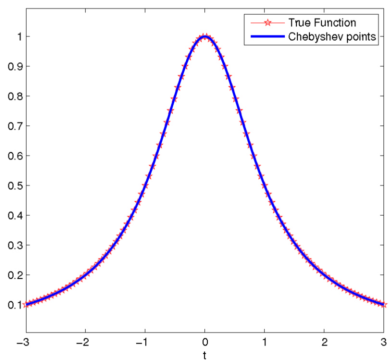
which makes the approximation not appropriate. See Figure 5 to look the behavior of the Lagrange interpolate polynomial with respect to evenly spaced nodes number. The error near the bounds of the interval is outstanding, while the comparison looks much better in the interior of [0,1]. The Runge phenomenon can be avoided using the Chebyshev nodes. Moreover, the use of large nodes can enhance the approximation. In fact, we can obtain the convergence to the function f earlier using only 20 points as shown in Figure 6. These plots show how interpolating at the roots of T_{20} can efficiently eliminates the bad behavior at the extremities of the interval $[-3,3]$.



(a) 20 interpolation points



(b) 50 interpolation points



(c) 100 interpolation points

Figure 6

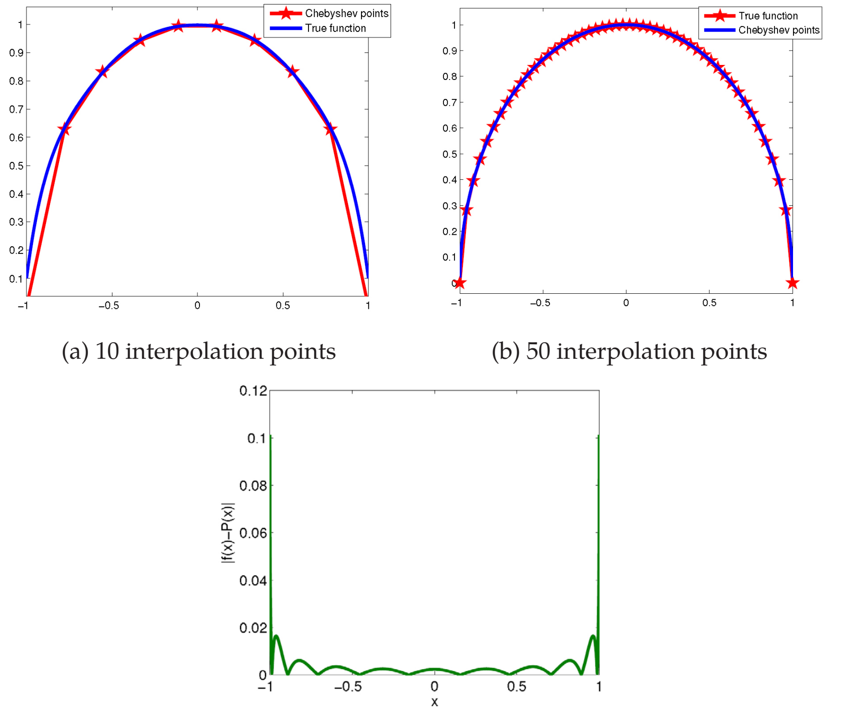
The evolution of the Lagrange approximate polynomials with respect to the number of nodes n. The approximate function is $f(x)=1/(1+x^2)$ using roots of Chebyshev polynomials as nodes.

4.1 *Examples of non-differentiable functions*

Taking into account the Proposition 1, the interpolate function *f* using the Chebyshev nodes must be smooth enough $(f \in \mathcal{C}^{n+1}([a,b]))$. Effectively, for non-differentiable function, the Chebyshev nodes fails to give a better approximation to the function *f*. To illustrate this limitation, we treat $f(x)=\sqrt{1-x^2}$ where $x \in [-1,1]$. This function represents a semicircle centered at the origin with a radius of 1. Since the function has a sharp corner at $x=\pm 1$, it is not differentiable at those points. Let's attempt to interpolate this function using Chebyshev polynomial interpolation. We'll choose a relatively low degree for simplicity, such as $n=4$ and we will use $n=10$ and $n=20$ in the numerical approximation. The Chebyshev nodes for $n=4$ are: $x_0 = \cos\left(\frac{\pi}{8}\right) \approx 0.9239$ $x_1 = \cos\left(\frac{3\pi}{8}\right) \approx 0.3827$ $x_2 = \approx -0.3827$ $x_3 = \cos\left(\frac{7\pi}{8}\right) \approx -0.9239$. Now, let's compute the Chebyshev interpolating $P^4(x)$ using these nodes. We have:

$$\mathrm{P}^4(x)=c_0T^0(x)+c_1T^1(x)+c_2T^2(x)+c_3T^3(x)+c_4T^4(x), \tag{28}$$

where $T^i(x)$ represents the Chebyshev polynomial of degree *i* and c_i are the coefficients determined by the function values at the nodes. The function values at the Chebyshev nodes are: $f(x_0) \approx 0.3827$ $f(x_1) \approx 0.9239$ $f(x_2) \approx 0.9239$ $f(x_3) \approx 0.3827$ By computing the interpolation polynomial using these function values, we obtain: $\mathrm{P}^4(x) = 0.6545x^4 - 0.8314x^2 + 0.3660$. If we compare this approximation to the right one $f(x)=\sqrt{1-x^2}$, we can see that it fails to accurately represent the sharp corner at $x=\pm 1$. See Figure 1 for such comparison. It seems that for $n=50$, the approximation is adequate, however, when we plot the absolute error $|f(x)-\mathrm{P}^{50}(x)|$ between the function *f* and P^{50} the error increases near the nodes –1 and 1, where the function in non-differentiable. The interpolation polynomial smooths out the corner, resulting in a polynomial that does not converge to $f(x)$ as *x* approaches ± 1. This failure to converge is a consequence of the non-differentiability of the function at the interpolation points. Chebyshev polynomial interpolation assumes certain smoothness properties of the function, and when those properties are violated, the convergence of the interpolation process is compromised. Therefore, in this demonstration, Chebyshev polynomial interpolation fails to accurately approximate the non-differentiable function $f(x)=\sqrt{1-x^2}$. The interpolation polynomial cannot capture the sharp corner of the function, highlighting the limitations of Chebyshev interpolation in such cases, see Figure 7.



(a) 10 interpolation points

(b) 50 interpolation points

(c) Absolute error of interpolation

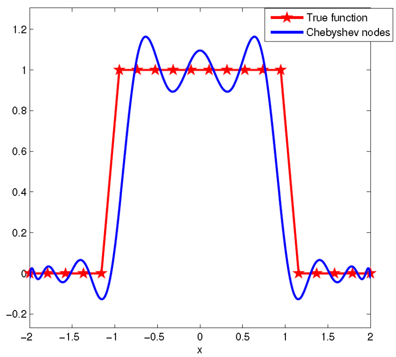
Figure 7

The evolution of the Lagrange approximate polynomials with respect to the number of nodes n. The approximate function is $f(x)=\sqrt{1-x^2}$ using roots of Chebyshev polynomials as nodes.

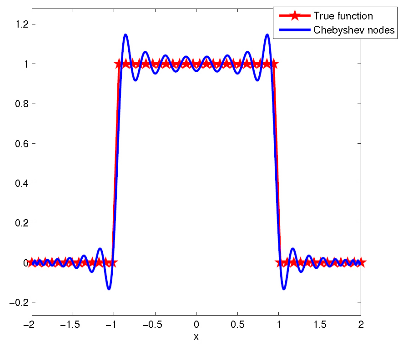
If the function has a rough allure, we can detect the so-called Gibbs phenomena, which is very similar to the Runge phenomena detected previously. We consider another example of the interpolating problem of the indicator function δ over the interval $[-1,1]$ at 20, 50 and 100 Chebyshev nodes. The function δ is defined by:

$$\delta(x):=\begin{cases}0 & \text{if } x\notin[-1,1],\\ 1 & \text{if } x\in[-1,1].\end{cases}$$

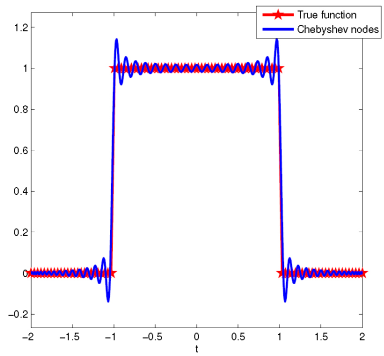
Figure 8 shows the polynomial Lagrange interpolate using the three number of points. As we can see, even with a large number of points, we cannot avoid the Gibs problem and the error between the true function and the approximate one remains higher.



(a) 20 interpolation points



(b) 50 interpolation points



(c) 100 interpolation points

Figure 8

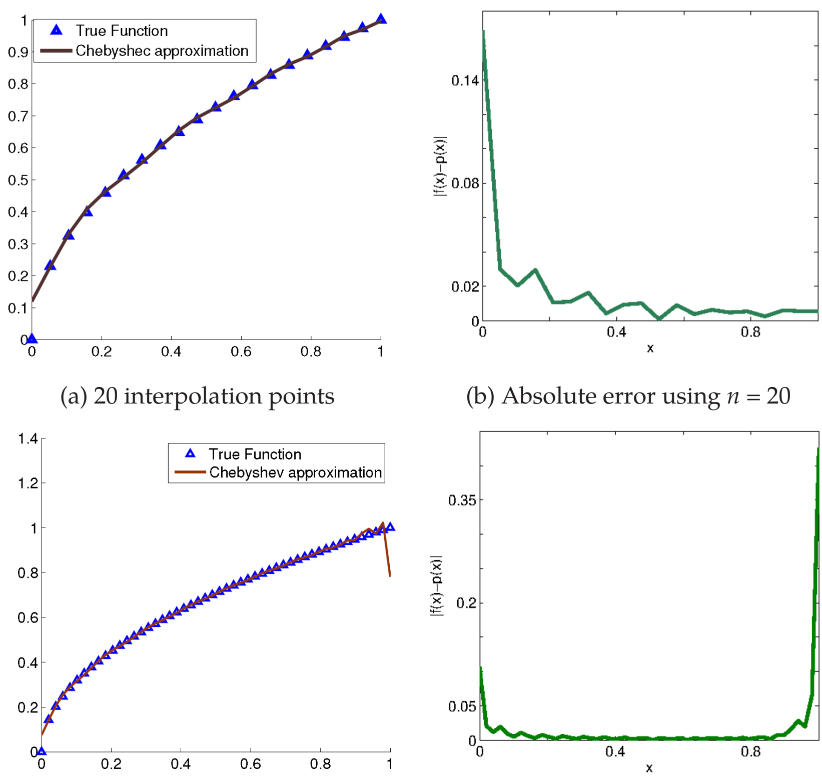
The evolution of the Lagrange approximate polynomials with respect to the number of nodes n. The approximate function is the indicator function δ using roots of Chebyshev polynomials as nodes.

This demonstrates once again that Chebyshev polynomial interpolation fails to accurately approximate the non-differentiable function δ. The Chebyshev interpolation polynomial cannot capture the sharp corner of the function, as guaranteed convergence requires sufficient differentiability of the function being interpolated.

4.2 *Numerical Counterexamples*

In this subsection, we present firstly an example where the Chebyshev polynomials nodes choice is not sufficient to ensure the convergence of the polynomial interpolation result even if f is smooth enough. Let's consider an example where the function being interpolated is differentiable, but

Chebyshev polynomial interpolation fails to converge. We consider $f(x) = \sqrt{x}$ such as: $x \in [0,1]$. This function is differentiable for all $xin]0,1]$. Chebyshev polynomial interpolation aims to approximate $f(x)$ using a polynomial $P_n(x)$ of degree n that passes through the Chebyshev nodes. Let's choose $n = 4$ for simplicity. The Chebyshev nodes for $n = 4$ are: x_0 $= \cos\left(\frac{\pi}{10}\right) \approx 0.9511$ x_1 $\cos\left(\frac{3\pi}{10}\right) \approx 0.5878$ $x_2 = \cos\left(\frac{5\pi}{10}\right) = 0$ $x_3 = \cos\left(\frac{7\pi}{10}\right) \approx$ -0.5878 $x_4 = \cos\left(\frac{9\pi}{10}\right) \approx -0.9511$ Now, let's compute the Chebyshev interpolating polynomial $P_n(x)$ of degree $n = 4$ using these nodes. The polynomial will pass through the function values at these nodes. The



(a) 20 interpolation points

(b) Absolute error using $n = 20$

(c) 50 interpolation points

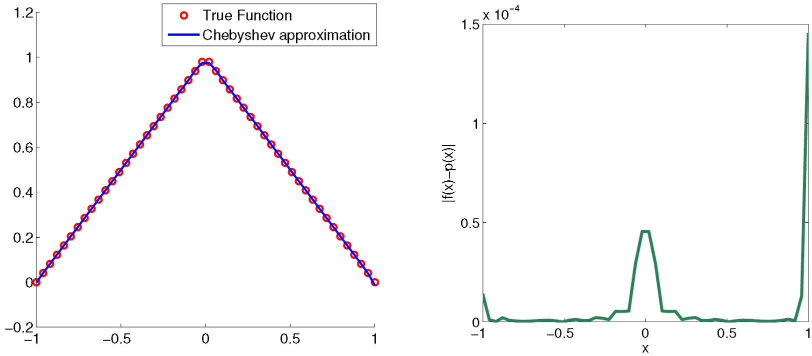
(d) Absolute error using $n = 50$

Figure 9

The evolution of the Chebyshev approximate polynomials in function of nodes n and the associated errors. The approximate function is the smooth function $\sqrt{x}$ using roots of Chebyshev polynomials as nodes.

function values at the Chebyshev nodes are: $f(x_0) = \sqrt{0.9511} \approx 0.9752$ $f(x_1) = \sqrt{0.5878} \approx 0.7661$ $f(x_2) = \sqrt{0} = 0$ $f(x_3) = \sqrt{0.5878} \approx 0.7661$ $f(x_4) = \sqrt{0.9511} \approx 0.9752$ By computing the interpolation polynomial using these function values and by using $n = 20$ and $n = 50$ nodes, we plot the function f with the Chebyshev interpolate function in Figure 9. If we compare this polynomial to the function $f(x) = \sqrt{x}$, , we can observe that Chebyshev polynomial interpolation fails to accurately approximate the function when the number of nodes increases, which not endure the convergence. The interpolation polynomial deviates from the true function, resulting in a poor approximation. This demonstrates a case where Chebyshev polynomial interpolation fails to converge even when the function being interpolated is differentiable. The failure occurs due to the specific arrangement of the interpolation nodes and the inherent limitations of polynomial approximation methods.

The last example illustrated a non-differentiable function where the Chebyshev interpolation works. Let's take $f(x) = 1 - |x|$ $x \in [-1, 1]$. By considering a specific degree for the interpolating polynomial, such as $n = 4$. We compute the Chebyshev nodes x_i for $n = 4$ using the formula $x_i = \cos\left(\frac{(2i+1)\pi}{2(n+1)}\right)$. The Chebyshev nodes for $n = 4$ are: $x_0 = \cos\left(\frac{\pi}{10}\right) \approx 0.9511$, $x_1 = \cos\left(\frac{3\pi}{10}\right) \approx 0.5878$, $x_2 = \cos\left(\frac{5\pi}{10}\right) = 0$, $x_3 = \cos\left(\frac{7\pi}{10}\right) \approx -0.5878$, $x_4 = \cos\left(\frac{9\pi}{10}\right) \approx -0.9511$. We have then the function values $f(x_i)$ at the Chebyshev nodes. For example: $f(x_0) = 1 - |0.9511| = 1 - 0.9511 = 0.0489$, $f(x_1) = 1 - |0.5878| = 1 - 0.5878 = 0.4122$, $f(x_2) = 1 - |0| = 1$, $f(x_3) = 1 - |-0.5878| = 1 - 0.5878 = 0.4122$, $f(x_4) = 1 - |-0.9511| = 1 - 0.9511 = 0.0489$. We compute the Chebyshev interpolating polynomial $P_n(x)$ of degree $n = 50$ using the function values at the nodes and we plot this approximation in Figure 10. By comparing the interpolation results of Chebyshev polynomial interpolation and the original function $f(x)$, we will likely observe that the Chebyshev polynomial interpolation provides an accurate approximation of the function $f(x) = 1 - |x|$ near the corner at $x = 0$ even if the function is not-differentiable. Chebyshev nodes are specifically designed to minimize oscillations and provide better convergence for non-differentiable functions with sharp features too. This example showcases how Chebyshev polynomial interpolation can provide



(a) 50 interpolation points (b) Absolute error using $n = 50$

Figure 10

The approximate function $|x|$ using roots of Chebyshev polynomials as nodes.

more accurate results for non-differentiable functions with sharp corners, such as $f(x) = 1 - |x|$.

4.3 *Application to PDE example*

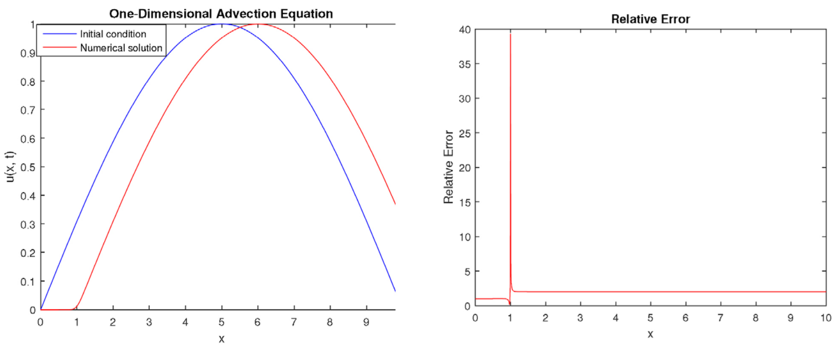
Let's consider the one-dimensional advection equation, a simple PDE often used to illustrate oscillatory behavior and instability, which can be analogous to the Runge phenomenon in interpolation.

The one-dimensional advection equation is given by:

$$\frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x} = 0$$

where u is a function of x and t, and c is a constant representing the advection speed.

We'll solve this equation using a finite difference method, specifically the upwind scheme, which can exhibit oscillations and instability similar to the Runge phenomenon in polynomial interpolation. In this example, the upwind scheme is used to solve the advection equation. The upwind scheme is known to produce oscillations and instability, especially if the advection speed c is high compared to the grid resolution. You'll observe oscillations and instability in the solution (see Figure 11, akin to the oscillations seen in the Runge phenomenon during polynomial interpolation.



(a) Aprroximate u (b) Relative error using $n = 1000$

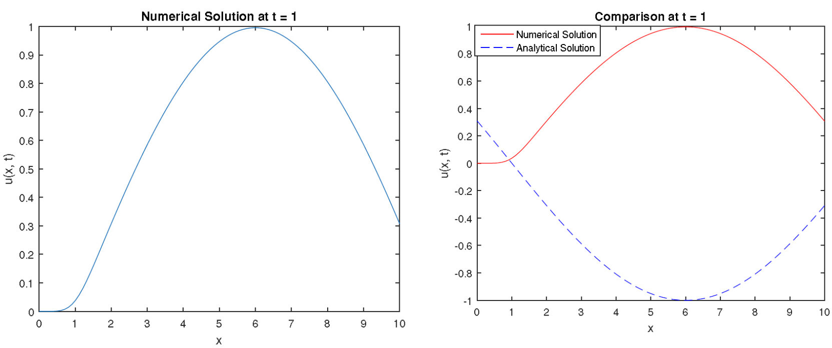
Figure 11

The approximate function c and the relative error.

To provide a comparison between the numerical solution and the true function, we'll modify the previous example to include a known analytical solution for the advection equation. This will allow us to observe how the numerical solution deviates from the true solution. The analytical solution for the advection equation $u(x,t)$ with a sinusoidal initial condition is:

$$u_{\text{analytical}}(x,t) = \sin(\pi(x - ct + L)/L)$$

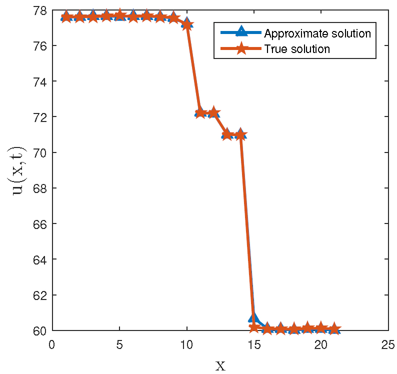
We'll calculate this analytical solution and plot it alongside the numerical solution for comparison in Figure 12.



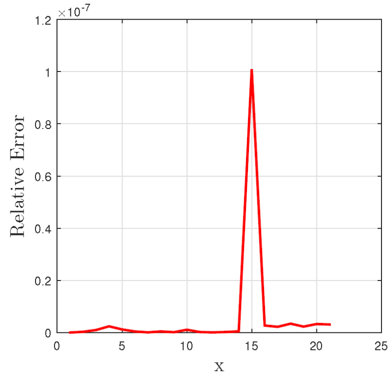
(a) Aprroximate u (b) Comparison with analytical solution

Figure 12

The approximate function u and comparison to the analytical one.



(a) Approximate c



(b) Relative error using $n = 1000$

Figure 13

The approximate function u and the associated relative error.

This modification calculates the analytical solution and plots it alongside the numerical solution at the final time step for comparison.

Keep in mind that for the advection equation, the numerical solution should converge to the analytical solution as the grid becomes finer (i.e., as the number of spatial points increases). However, in certain cases, especially with schemes that are susceptible to oscillations and instability, the numerical solution might deviate from the analytical solution. This deviation is what we'll observe and compare in the plot. Next, if we construct a non-smooth solution with irregular points and we try to recover the true function through a finite element scheme using Chebyshev interpolation, the approximate solution looks very well even id the function is not smooth, which give us another counterexample. In fact the function not verifies the assumptions, nevertheless the relative error is very interesting and very low, which confirm that in this case the Chebyshev interpolation worked very well as seen in Figure 13.

5. Conclusion

In summary, this paper discusses the phenomenon known as Runge's phenomenon, which arises during the interpolation process when uniformly spaced nodes are used for interpolating smooth functions. To address this issue, the abstract suggests utilizing Chebyshev roots as interpolation nodes, which have shown to be an effective solution. We explores the efficiency of Chebyshev nodes by presenting several counterexamples. These examples encompass both differentiable and non-differentiable functions, aiming to demonstrate the limitations of using Chebyshev roots as nodes for Lagrange interpolation. By examining these counterexamples, this work provides valuable insights into the potential challenges and drawbacks associated with employing Chebyshev polynomial interpolation in practice.

Acknowledgements

We appreciate the remarks given by the anonymous reviewers and also the Editor for the helpful suggestions and corrections.

Compliance with Ethical Standards

Note applicable.

Conflict of Interest

Note applicable.

Funding

This work haven't received any funds.

References

[1] L. N. Trefethen. Approximation Theory and Approximation Practice, Extended Edition. SIAM (2019).

[2] S. Roberts, A. Sarshar and A. Sandu. Parallel implicit-explicit general linear methods. *Communications on Applied Mathematics and Computation,* 3(4) : 649-669 (2019).

[3] J. P. Boyd and J. R. Ong. Exponentially-convergent strategies for defeating the Runge Phenomenon for the approximation of non-periodic functions, part two: Multi-interval polynomial schemes and multidomain Chebyshev interpolation. Applied Numerical Mathematics, 61(4) : 460-472 (2011).

[4] J. P. Boyd and J. R. Ong. Whitney edge elements and the Runge phenomenon. *Journal of Computational and Applied Mathematics,* 427(1) : 115-127 (2023).

[5] E. Cheney. Chebyshev methods in numerical approximation (martin avery snyder). *SIAM Review*, 9(2) : 264 (1967).

[6] S. Jawad Jaaywel, K. Hashim Abdul-Khaliq. The error analysis of linearized discontinuous Galerkin finite element method for incompressible miscible displacement in porous media. *Journal of Interdisciplinary Mathematics*, 22(8) : 1471-1484 (2019).

[7] J. Jabbar Hussein, K. Hashim Abdul-Khaliq. A weak Galerkin finite element method for two-dimensional coupled burgersâ€™ equation by using polynomials of order (k, k-1, k-1). *Journal of Interdisciplinary Mathematics*, 23(4) : 777-790 (2020).

[8] A. Laghrib, A. Hadri, M. Hakim. An Enhanced Fluid Registration for Image Multi-Frame Super Resolution. *Scientific Annals of Computer Science*, 32(2):11-34 (2022).

[9] M. Alahyane, A. Hakim, A. Laghrib, S. Raghay. An improved SIMPLEC scheme for fluid registration. *Mathematical Modelling and Analysis*, 28(1) : 71-90 (2023).

[10] B. Fornberg and J. Zuev. The runge phenomenon and spatially variable shape parameters in rbf interpolation. *Computers & Mathematics with Applications*, 54(3) : 379-398 (2007).

[11] J. C. Mason and D. C. Handscomb. Chebyshev polynomials. Chapman and Hall/CRC (2002).

Received June, 2023