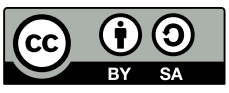
Notes on Approximation Theory

Gentian Zavalani

December 16, 2025



Contents

1 Introduction

1.1 A historical overview of approximation theory

Approximation theory[1] is a well-established branch of mathematics with a history spanning more than 150 years. Its development can be traced through four main eras, each marked by distinctive contributions, ideas, and key figures. Although these eras are not strictly defined, they provide a useful framework for understanding the evolution of the subject. The following overview outlines the Chebyshev, Classical, Neoclassical, and Numerical eras of approximation theory.

1.1.1 Chebyshev era (19th Century)

The origins of modern approximation theory are often associated with the Chebyshev era in the 19th century. Chebyshev, a Russian mathematician, was the central and dominant figure of this period. Other notable contributors include Jacobi and Zolotarev, a brilliant young mathematician in Chebyshev's group.
In parallel, Weierstrass and later Runge also made significant contributions, with Runge playing an important role in the history of numerical mathematics.
During this era, the primary focus was on expansions using orthogonal polynomials and the formulation of the concept of best approximation.

1.1.2 Classical era (1900–1925)

The early 20th century marks the beginning of the classical era of approximation theory, extending roughly from 1900 to 1925. This period placed approximation theory at the center of mathematical inquiry, addressing fundamental questions such as: *What is a function?*
Prominent figures of this era include Lebesgue (whose first paper was devoted to approximation theory), de la Vallée Poussin, Faber, Fejér, Bernstein, and Jackson. Among them, Bernstein, Jackson, and de la Vallée Poussin are frequently mentioned together due to their closely related work.
This era laid the foundation of modern analysis and established approximation theory as an integral component of mathematical research.

[1] This course is closely connected to the book Approximation Theory and Approximation Practice[22].

1.1.3 Neoclassical era (1950–1975)

The neoclassical era, spanning approximately from 1950 to 1975, represents the period just before the widespread use of computers in mathematics. During this time, approximation theory matured significantly, becoming a central topic in textbooks and journals.
Key contributors include Davis, Lorentz, De Boor, and Rice. One of the most influential developments of this era was the theory of splines, which emerged as a powerful tool in constructive approximation. In addition, important advances were made in the study of rational functions and rational approximation.
This period marked the recognition of approximation theory as a field in its own right.

1.1.4 Numerical era (1975–Present)

The numerical era, beginning around 1975 and continuing to the present day, is characterized by the expansion of approximation theory into diverse and modern directions. Notable areas of development include:

- radial basis functions,
- spectral methods for solving differential equations (both PDEs and ODEs, closely related to hp-FEM),
- wavelets,
- and compressed sensing, which involves l_1 and l_0 approximation and remains a challenging problem.

This era highlights the growth of approximation theory from its classical 19th-century origins into a broad, dynamic field with deep connections to numerous areas of mathematics and its applications.

1.2 Chebyshev points and interpolants

Let $n \geq 0$ and denote by P_n the set of polynomials of degree at most n, so that $|P_n| = n+1$.
Suppose we are given $n+1$ distinct points in the interval $[-1, 1]$, denoted by $x_0, \ldots, x_n$, together with data values $f_0, \ldots, f_n \in \mathbb{R}$ (or $\mathbb{C}$).
It is a classical result in interpolation theory that there exists a unique polynomial $p \in P_n$ such that

$$p(x_j) = f_j, \qquad j = 0, 1, \ldots, n.$$

This polynomial is called the *interpolating polynomial* for the data $\{(x_j, f_j)\}_{j=0}^n$.
In this course, we will primarily use the so-called *Chebyshev points*, defined by

$$x_j = \cos\left(\frac{j\pi}{n}\right), \qquad j = 0, 1, \ldots, n.$$

For a visual representation, the distribution of Chebyshev points is illustrated in Fig. (1.1).

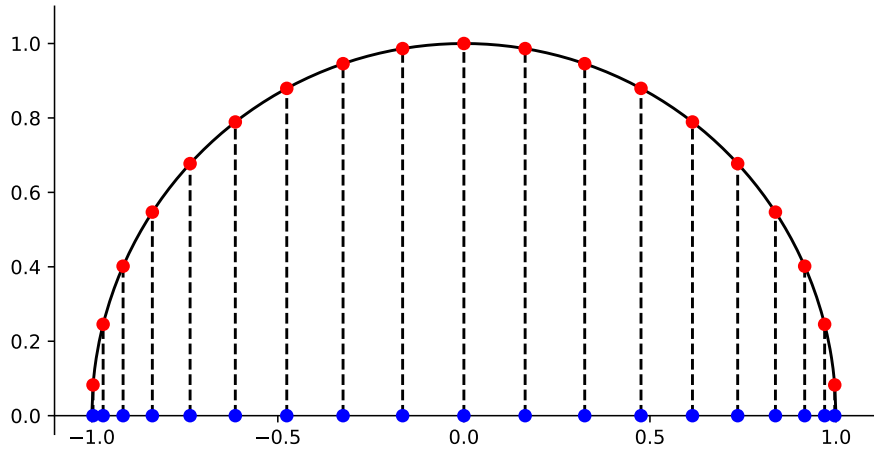


Figure 1.1: Chebyshev points $x_j \in [-1, 1]$ (blue) for degree $n = 20$, along with equidistant auxiliary construction points (red) on the semicircle.

These points are symmetrically distributed in $[-1, 1]$, with clustering near the endpoints of the interval. Such clustering plays a crucial role in ensuring good interpolation properties. By contrast, interpolation at equidistant points is known to behave poorly, often suffering from severe instability.
For convenience, the points may also be ordered from left to right using

$$x_j = -\cos\left(\frac{j\pi}{n}\right).$$

The interpolating polynomial constructed through Chebyshev points is often referred to as the *Chebyshev interpolant*. This distribution of points is distinguished by several features:

- **Endpoint clustering.** Chebyshev points are naturally concentrated near the endpoints of the interval $[-1, 1]$. This clustering mitigates instability and improves approximation quality compared with equidistant nodes.
- **Relation to other optimal sets.** Other point distributions with similar endpoint clustering, such as Legendre points, exhibit analogous interpolation properties (see Section 11.2 for a discussion).

1.3 Chebyshev polynomials and series

Approximation theory may be studied in three equivalent frameworks: the Fourier, Laurent, and Chebyshev[2] settings. In the Fourier world the variable is $\theta \in [-\pi, \pi]$, in the Laurent world the variable is $z \in \mathbb{C}$ constrained to the unit circle $|z| = 1$, and in the Chebyshev world the variable is $x \in [-1, 1]$. These are related through the transformation

$$x = \tfrac{1}{2}\left(z + z^{-1}\right), \qquad z = e^{i\theta}, \qquad x = \cos\theta.$$

In these three settings the corresponding functions satisfy simple symmetries:

$$\text{Fourier: } \mathbb{F}(\theta) = \mathbb{F}(-\theta), \qquad \text{Laurent: } F(z) = F(z^{-1}), \qquad \text{Chebyshev: } f(x).$$

The associated domains of analyticity also differ: in the Fourier setting one considers analyticity in a strip around the real axis, in the Laurent setting analyticity in an annulus, and in the Chebyshev setting analyticity in an ellipse in the complex plane.
Interpolation takes parallel forms in the three frameworks. In the Fourier setting the interpolation nodes are equispaced points on $[-\pi, \pi]$,

$$\theta_j = \tfrac{j\pi}{n}, \qquad -n + 1 \le j \le n. \tag{1.1}$$

in the Laurent setting they correspond to the $2n$ roots of unity,

$$z_j = e^{ij\pi/n}, \qquad -n + 1 \le j \le n, \tag{1.2}$$

and in the Chebyshev setting to the $n + 1$ Chebyshev points in $[-1, 1]$,

$$x_j = \cos\left(\tfrac{j\pi}{n}\right), \qquad 0 \le j \le n. \tag{1.3}$$

The interpolating functions also take analogous forms. In the Fourier framework one works with trigonometric polynomials,

$$\tfrac{1}{2}\sum_{k=-n}^{n} a_k\big(e^{ik\theta} + e^{-ik\theta}\big),$$

in the Laurent framework with Laurent polynomials,

$$\tfrac{1}{2}\sum_{k=0}^{n} a_k\big(z^k + z^{-k}\big),$$

and in the Chebyshev framework with algebraic polynomials in the Chebyshev basis[3],

$$\sum_{k=0}^{n} a_k T_k(x). \tag{1.4}$$

[2]The name Chebyshev is sometimes written with a "T" (Tchebyshev), reflecting the French transliteration used in his publications.
[3]Although the Chebyshev polynomials have not yet been defined.

Series	Assumptions	Setting	Interpolation points
Chebyshev	none	$x \in [-1, 1]$	Chebyshev points
Fourier	$\mathbb{F}(\theta) = \mathbb{F}(-\theta)$	$\theta \in [-\pi, \pi]$	equispaced points
Laurent	$F(z) = F(z^{-1})$	$z \in$ unit circle	roots of unity

Table 1.1: Fourier, Chebyshev, and Laurent series are closely related. Each representation can be converted into the other by a change of variables. Under this transformation, Chebyshev points, equispaced points, and roots of unity are interconnected.

As the number of interpolation points tends to infinity, $n \to \infty$, one arrives at the *Chebyshev series* expansion

$$f(x) = \sum_{k=0}^{\infty} a_k T_k(x).$$

This representation is directly analogous to a Fourier series, but with Chebyshev basis functions. Convergence depends on the smoothness and analyticity of the underlying function. We have summarized these links in Table (1.1)

Chebyshev polynomials

Now let us turn to the definitions, already implicit in Eq.(1.4). The kth *Chebyshev polynomial* can be defined as the real part of the function z^k on the unit circle:

$$x = \Re(z) = \tfrac{1}{2}(z + z^{-1}) = \cos\theta, \qquad \theta = \cos^{-1} x, \tag{3.7}$$

$$T_k(x) = \Re(z^k) = \tfrac{1}{2}(z^k + z^{-k}) = \cos(k\theta). \tag{3.8}$$

(Chebyshev polynomials were introduced by Chebyshev in the 1850s, though without the connection to the variables z and θ. The label T was apparently chosen by Bernstein, following French transliterations such as "Tchebischeff.")

The Chebyshev polynomial T_k oscillates like $\cos(k\theta)$. With small wavelength at the boundaries:

$$T_0(x) = 1, \qquad T_1(x) = x, \qquad T_2(x) = 2x^2 - 1, \qquad T_3(x) = 4x^3 - 3x.$$

The Chebyshev polynomials obey a three-term recurrence relation

$$\boxed{T_{k+1}(x) = 2x\,T_k(x) - T_{k-1}(x)}$$

To prove this, we translate the problem into the z-variable, we have

$$T_{k+1}(x) = \left(\tfrac{1}{2}(z^k + z^{-k})\right)(z + z^{-1}) - \tfrac{1}{2}(z^{k-1} + z^{-(k-1)}),$$

and since $z + z^{-1} = 2x$, this reduces to

$$T_{k+1}(x) = 2x\,T_k(x) - T_{k-1}(x).$$

1.4 Interpolants, projections, and aliasing

Now we state the basic theorem about Chebyshev series and their coefficients.

Theorem 1.1. *Let $f : [-1, 1] \to \mathbb{R}$ be Lipschitz continuous. It admits a* ***Chebyshev expansion***

$$f(x) = \sum_{k=0}^{\infty} a_k T_k(x), \tag{1.5}$$

which is absolutely and uniformly convergent, and the coefficients are given for $k \geq 1$ by the formula

$$a_k = \frac{2}{\pi} \int_{-1}^{1} \frac{f(x)T_k(x)}{\sqrt{1-x^2}}\,dx, \tag{3.12}$$

and for $k = 0$ by the same formula with the factor $2/\pi$ changed to $1/\pi$.

This is true for the same reason functions have Fourier series. For $z = e^{i\theta}$, define

$$F(z) = f\left(\tfrac{1}{2}(z + z^{-1})\right).$$

Then $F(z)$ is a function on the unit circle with a Laurent series:

$$F(z) = \sum_k a_k T_k\left(\tfrac{1}{2}(z + z^{-1})\right) = \sum_k \frac{a_k}{2}\,(z^k + z^{-k}).$$

For $\theta \in [-\pi, \pi]$, define

$$\mathbb{F}(\theta) = g(e^{i\theta}) = f\left(\tfrac{1}{2}(e^{i\theta} + e^{-i\theta})\right).$$

Then $\mathbb{F}(\theta)$ is a 2π-periodic function with a Fourier series:

$$\mathbb{F}(\theta) = \sum_k a_k T_k\left(\tfrac{1}{2}(e^{i\theta} + e^{-i\theta})\right) = \sum_k \frac{a_k}{2}\,(e^{ik\theta} + e^{-ik\theta}).$$

One approximation to f in P_n is by truncation or projection of the series to degree n, whose coefficients through degree n are the same as those of f itself:

(a) Projection (Truncation).

$$\boxed{f_n(x) = \sum_{k=0}^{n} a_k\,T_k(x)}$$

Another is the polynomial obtained by interpolation in Chebyshev points:

(b) Interpolation.

$$\boxed{p_n(x) = \sum_{k=0}^{n} c_k\, T_k(x)}$$

where $\{c_k\}$ denote the Chebyshev coefficients of the interpolant.
The relationship of the Chebyshev coefficients of f_n to those of f is obvious, while the key to understanding $\{c_k\}$is the phenomenon of aliasing,

Theorem 1.2 (Aliasing of Chebyshev polynomials.)**.** *Given $n \geq 1$ and $\{x_k\} = (n+1)$-point Chebyshev grid.*
For any m with $0 \leq m \leq n$, the following are the same on the grid:

$$T_m, \quad T_{2n-m}, \quad T_{2n+m}, \quad T_{4n-m}, \quad T_{4n+m}, \quad T_{6n-m}, \quad T_{6n+m}, \ldots$$

Equivalently, for any $k \geq 0$, T_k takes the same value on the grid as T_m with

$$m = \big|(k+n-1)\,(\mathrm{mod} 2n) - (n-1)\big|, \tag{4.4}$$

a number in the range $0 \leq m \leq n$.

Proof. Transplanting to the z-variable on the unit circle, the Chebyshev polynomials can be expressed (up to a factor of $\frac{1}{2}$) as

$$\big(z^m + z^{-m}\big), \quad \big(z^{2n-m} + z^{-(2n-m)}\big), \quad \big(z^{2n+m} + z^{-(2n+m)}\big), \quad \ldots$$

Since

$$z^{2n} = 1 \quad \text{at the } 2n\text{-th roots of unity,}$$

and these points project onto the Chebyshev points on the x-axis, all the above expressions coincide at the roots of unity.
Transplanting back to the x-variable, we conclude that the corresponding Chebyshev polynomials agree at the Chebyshev points. □

Theorem 1.3 (Aliasing formula for Chebyshev coefficients)**.** *Let f be Lipschitz continuous on $[-1,1]$, and let p_n be its Chebyshev interpolant in P_n, $n \geq 1$. Let $\{a_k\}$ and $\{c_k\}$ be the Chebyshev coefficients of f and p_n, respectively. Then*

$$c_0 = a_0 + a_{2n} + a_{4n} + \cdots, \tag{1.6}$$

$$c_n = a_n + a_{3n} + a_{5n} + \cdots, \tag{1.7}$$

and for $1 \leq k \leq n-1$,

$$c_k = a_k + \big(a_{k+2n} + a_{k+4n} + \cdots\big) + \big(a_{-k+2n} + a_{-k+4n} + \cdots\big). \tag{1.8}$$

Proof. By Theorem 3.1, the function f has a *unique Chebyshev series expansion*

$$f(x) = \sum_{k=0}^{\infty} a_k T_k(x),$$

and this series converges absolutely. Absolute convergence is important because it means we can rearrange the terms in the sum without changing the result.

By absolute convergence at $x = 1$, the series (1.6), (1.7), and (1.8) provide well-defined coefficients $c_0, \ldots, c_n$. These coefficients therefore determine a polynomial, which we denote by

$$q(x) = \sum_{k=0}^{n} c_k T_k(x) \in P_n.$$

This is a polynomial of degree at most n.

Now, we need to verify that this polynomial indeed interpolates the data, and hence, by uniqueness, is the interpolant of the data. To this end, let x_j be a grid point. From the Chebyshev expansion of f, we obtain

$$f(x_j) = \sum_{k=0}^{\infty} a_k T_k(x_j). \tag{1.9}$$

On the other hand, for the polynomial q we obtain expansion of f we have

$$q(x_j) = \sum_{k=0}^{n} c_k T_k(x_j). \tag{1.10}$$

The coefficients a_k that appear in the series (1.9) also occur in the polynomial (1.10), though arranged in a different order. Each a_k is associated with exactly one c_k. Thus, the two expressions represent the same sum, only permuted. Hence $q = p_n$.

□

We can summarize Theorem (1.3) as follows. On the $(n+1)$-point grid, any function f is indistinguishable from a polynomial of degree n.
As a corollary, Theorems (1.2) and (1.3) give absolutely convergent series for the errors $f - f_n$ and $f - p_n$.

Corollary 1.4.

$$f(x) - f_n(x) = \sum_{k=n+1}^{\infty} a_k \, T_k(x), \tag{1.11}$$

$$f(x) - p_n(x) = \sum_{k=n+1}^{\infty} a_k \, (T_k(x) - T_m(x)), \tag{1.12}$$

where $m = \big|(k + n - 1) \bmod (2n) - (n - 1)\big|$.

The meaning is clear: in projection, the error comes solely from discarding the higher-order terms, as in (1.11). Interpolation behaves differently. The higher modes are not simply omitted, but rather *misinterpreted* as lower-order Chebyshev polynomials. Each term that fails to be represented correctly shows up again, but as a different Chebyshev polynomial. In fact, T_m is the alias of T_k, with m the index that k gets mapped to.

Example ($n = 3$): The Chebyshev grid has 4 points:

$$x_j = \cos\Big(\tfrac{j\pi}{3}\Big), \qquad j = 0, 1, 2, 3.$$

So the points are $x = \{1, \frac{1}{2}, -\frac{1}{2}, -1\}$.

Now check:

- $T_1(x) = x$. At the grid points, this is $\{1,\ 0.5,\ -0.5,\ -1\}$.
- $T_5(x) = 16x^5 - 20x^3 + 5x$. At the same grid points, this is also $\{1,\ 0.5,\ -0.5,\ -1\}$.

Observation: On the grid, $T_5(x) = T_1(x)$ look identical. So interpolation cannot tell them apart: it will misinterpret T_5 as T_1.

Interpolation error. So, when building the interpolant, the grid values of $a_5T_5(x)$ look exactly the same as the grid values of $a_5T_1(x)$.
So the error contains

$$a_5(T_5(x) - T_1(x)).$$

We have

$$f(x) - p_n(x) = \sum_{k=n+1}^{\infty} a_k(T_k(x) - T_m(x)), \qquad f(x) - f_n(x) = \sum_{k=n+1}^{\infty} a_k\, T_k(x).$$

Since $|T_k(x)| \le 1$ and $|T_m(x)| \le 1$ for all $x \in [-1, 1]$, it follows that

$$\big|T_k(x) - T_m(x)\big| \ \le\ |T_k(x)| + |T_m(x)| \ \le\ 2.$$

Therefore, pointwise on $[-1, 1]$,

$$|f(x) - p_n(x)| \ \le\ 2 \sum_{k=n+1}^{\infty} |a_k|,$$

and taking the supremum norm gives

$$\|f - p_n\|_\infty \ \le\ 2 \sum_{k=n+1}^{\infty} |a_k|. \tag{1.13}$$

For comparison, the truncation error satisfies

$$|f(x) - f_n(x)| = \Big| \sum_{k=n+1}^{\infty} a_k\, T_k(x)\Big| \le \sum_{k=n+1}^{\infty} |a_k|,$$

so that

$$\|f - f_n\|_\infty \ \le\ \sum_{k=n+1}^{\infty} |a_k|. \tag{1.14}$$

Thus the two approximations are typically within a factor of 2 of each other in accuracy.

2 Convergence results for Chebyshev approximation

A basic principle of approximation theory is: the smoother a function, the faster its polynomial approximations converge as $n \to \infty$. We first focus on a class of functions with a given degree of differentiability and examine how their smoothness influences the convergence rate.

2.1 Total variation

One way to measure smoothness is by the *total variation* V. Introduced by Jordan in the late 19th century [14, 15], it measures how much a function oscillates over an interval.

Definition 2.1 (Bounded variation)**.** *For $f : [a, b] \to \mathbb{R}$, the total variation on $[a, b]$ is*

$$V := \sup \left\{ \sum_{i=1}^{n} |f(x_k) - f(x_{i-1})| : a = x_0 \le x_1 \le \cdots \le x_n = b,\ n \in \mathbb{N} \right\}.$$

If $V < \infty$, we say that f has bounded variation*.*

If f is differentiable, this reduces to

$$V = \int_{-1}^{1} |f'(x)|\, dx = \|f'\|_{L^1}.$$

Classical results such as the Jackson theorems[13] state that if f is k times continuously differentiable on $[-1, 1]$, then the best polynomial approximations converge at rate $\mathcal{O}(n^{-k})$. However, in borderline cases this prediction can be too pessimistic. For example, $f(x) = |x|$ has a cusp at 0 and is not differentiable there, so the theorem would not guarantee convergence. Yet in practice Chebyshev interpolation converges linearly, $\mathcal{O}(n^{-1})$ (See Fig.(2.1)). The concept of variation captures this behavior correctly.

Chebyshev weighted norm. For Chebyshev approximation it is convenient to introduce the weighted 1-norm

$$\|f\|_T = \int_{-1}^{1} \left| \frac{f'(x)}{\sqrt{1 - x^2}} \right| dx. \tag{2.1}$$

This can be defined via a Stieltjes integral for any function of bounded variation, though it may be infinite depending on the behavior near $x = \pm 1$. The relevant condition for our theorems is that

$$\|f^{(k)}\|_T < \infty,$$

where $f^{(k)}$ denotes the kth derivative of f.

2.2 Convergence for differentiable functions

Our first step toward a precise convergence theorem for differentiable functions is to obtain a bound on the Chebyshev coefficients [22, Theorem 7.1].

Theorem 2.2 (Chebyshev coefficients of differentiable functions)**.** *For an integer $\nu \geq 0$, let $f, f', \ldots, f^{(\nu-1)}$ be absolutely continuous on $[-1,1]$ and suppose*

$$\|f^{(\nu)}\|_T = V < \infty.$$

Then for $k \geq \nu + 1$, the Chebyshev coefficients of f satisfy

$$|a_k| \leq \frac{2V}{\pi k(k-1)\cdots(k-\nu)} \leq \frac{2V}{\pi(k-\nu)^{\nu+1}}. \tag{2.2}$$

Proof.

$$a_k = \frac{2}{\pi}\int_{-1}^{1} \frac{f(x)T_k(x)}{\sqrt{1-x^2}}\,dx$$

Changing variables with $x = \cos\theta$, $dx = -\sin\theta\,d\theta$, and $\sqrt{1-x^2} = \sin\theta$, we obtain

$$a_k = \frac{2}{\pi}\int_0^{\pi} f(\cos\theta)\cos(k\theta)\,d\theta.$$

Integration by parts gives

$$a_k = \frac{2}{\pi k}\int_0^{\pi} f'(\cos\theta)\sin(k\theta)\sin\theta\,d\theta.$$

Using the trigonometric identity

$$\sin\theta\,\sin(k\theta) = \tfrac{1}{2}\cos\big((k-1)\theta\big) - \tfrac{1}{2}\cos\big((k+1)\theta\big),$$

this becomes

$$a_k = \frac{2}{\pi k}\int_0^{\pi} f'(\cos\theta)\left[\frac{\cos((k-1)\theta)}{2} - \frac{\cos((k+1)\theta)}{2}\right]d\theta.$$

which implies

$$|a_k| \leq \frac{2}{\pi k}\,\|f'(\cos\theta)\|_1 \left\|\frac{\cos((k-1)\theta)}{2} - \frac{\cos((k+1)\theta)}{2}\right\|_\infty$$

$$= \frac{2}{\pi k}\,\|f(x)\|_T \left\|\frac{\cos((k-1)\theta)}{2} - \frac{\cos((k+1)\theta)}{2}\right\|_\infty$$

since $d\theta = dx/\sqrt{1-x^2}$. The L^∞ norm is bounded by 1, and thus we have established

$$|a_k| \leq \frac{2}{\pi k}V^{(0)} \quad \text{with} \quad V^{(0)} = \|f\|_T.$$

Further integrations by parts bring in higher derivatives of f and corresponding higher variations up to $V^{(\nu)} = V = \|f^{(\nu)}\|_T$. More and more cosine terms appear, but the coefficients are such that their sum always has $\infty-$norm at most 1. Just as the first integration by parts introduced a factor k in the denominator, the second leads to factors $k-1$ and $k+1$, the third to factors $k-2$, k, and $k+2$, and so on. To keep the formulas simple we do not keep track of all these different denominators but weaken the inequality slightly by replacing them all with $k-1$ at the second differentiation, $k-2$ at the third, and so on up to $k-\nu$ at the $(\nu+1)$st differentiation. The result is (2.2). □

From Theorem(2.2) we can derive consequences about the accuracy of Chebyshev projections and interpolants [22, Theorem 7.1].

Theorem 2.3 (Convergence for differentiable functions)**.** *If f satisfies the conditions of Theorem* (2.2)*, with V again denoting the total variation of $f^{(\nu)}$ for some $\nu \geq 1$, then for any $n > \nu$, its Chebyshev projections satisfy*

$$\|f - f_n\| \leq \frac{2V}{\pi\nu(n-\nu)^\nu} \tag{2.3}$$

and its Chebyshev interpolants satisfy

$$\|f - p_n\| \leq \frac{4V}{\pi\nu(n-\nu)^\nu}. \tag{2.4}$$

Proof. For (2.3), applying Theorem (2.2) to (1.14) yields

$$\|f - f_n\| \leq \sum_{k=n+1}^{\infty} |a_k| \leq \frac{2V}{\pi} \sum_{k=n+1}^{\infty} (k-\nu)^{-\nu-1}.$$

Since the summand is monotonically decreasing, we may bound the series by the corresponding integral:

$$\sum_{k=n+1}^{\infty} (k-\nu)^{-\nu-1} \leq \int_n^{\infty} (s-\nu)^{-\nu-1}\, ds = \frac{1}{\nu(n-\nu)^\nu}.$$

Thus,

$$\|f - f_n\| \leq \frac{2V}{\pi\nu(n-\nu)^\nu}.$$

For (2.4), we use the fact that the approximation error of the interpolant can be bounded by twice the bound appearing in the case of truncation (see equation (1.13))

$$\|f - p_n\| \leq 2\|f - f_n\| \leq \frac{4V}{\pi\nu(n-\nu)^\nu}. \tag{2.5}$$

□

Example 2.4. We illustrate the convergence behavior of Chebyshev[1] interpolation for functions of different smoothness. For $f(x) = |x|$ on $[-1, 1]$ with $\nu = 1$, the error decays at the rate $\mathcal{O}(n^{-1})$, as shown in Fig. (2.1). For a smoother function, $f(x) = |x|^3$ with $\nu = 3$, the error decreases at the faster rate $\mathcal{O}(n^{-3})$, as shown in Fig. (2.2). These results confirm the theoretical prediction that the decay rate is governed by the smoothness parameter ν.

Listing 2.1: MATLAB code used in Example 2.4 for Chebyshev interpolation of $f(x) = |x|^3$, showing the decay rate $\mathcal{O}(n^{-3})$; a similar procedure applies to $f(x) = |x|$ with rate $\mathcal{O}(n^{-1})$.

```
% Cubic convergence study with Chebyshev interpolation (Chebfun required)

% Define function
x = chebfun('x');
f = abs(x)^3;

% Degrees
nn = 2*round(2.^(0:0.3:7)) - 1;
ee = zeros(size(nn));

% Compute errors
for j = 1:length(nn)
    n = nn(j);
    fn = chebfun(f, n+1);
    ee(j) = norm(f - fn, inf);
end

% Reference slope ~ n^{-1}
refLine = 1 ./ nn.^3;

% Plot
figure;
loglog(nn, ee, 'bo-', 'MarkerFaceColor', 'b', 'LineWidth', 1.2);hold on;
loglog(nn, refLine, 'r-', 'LineWidth', 1.5);

% Labels & annotations
text(6, 7e-2, '$n^{-3}$', 'FontSize', 22, 'Color', 'k', 'Interpreter', 'latex');

grid on;
set(gca, 'FontSize', 16, 'LineWidth', 1);
xlabel('Polynomial degree $n$', 'Interpreter', 'latex', 'FontSize', 16);
ylabel('$\| f - p_n \|_{\infty}$', 'Interpreter', 'latex', 'FontSize', 16);
title('Cubic convergence', 'Interpreter', 'latex', 'FontSize', 14);
axis tight;
```

[1]The MATLAB code presented below requires the Chebfun package, available at **https://www.chebfun.org/**

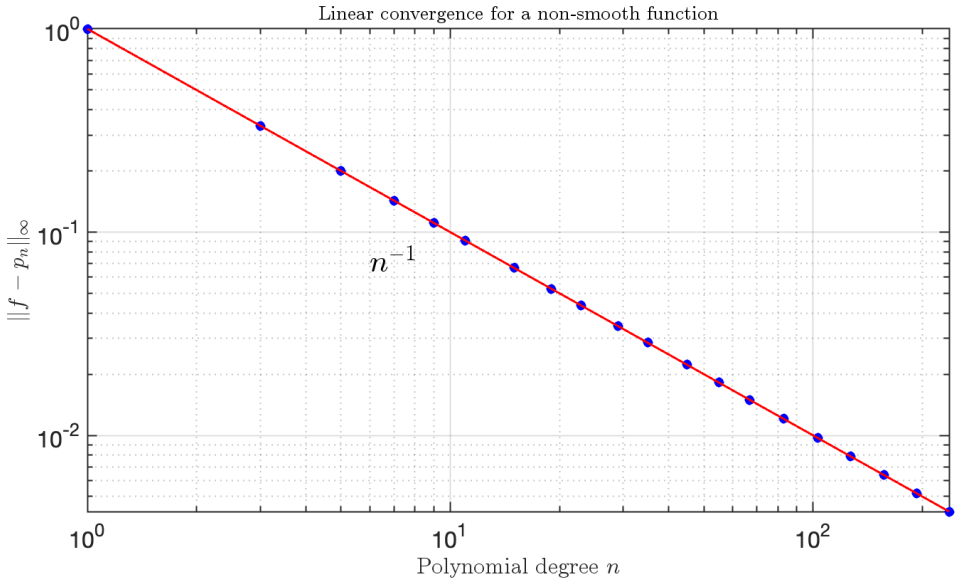


Figure 2.1: Linear convergence of Chebyshev interpolation for the non-smooth function $f(x) = |x|$. The plot shows that the error $\|f - p_n\|_\infty$ decays at the rate n^{-1}.

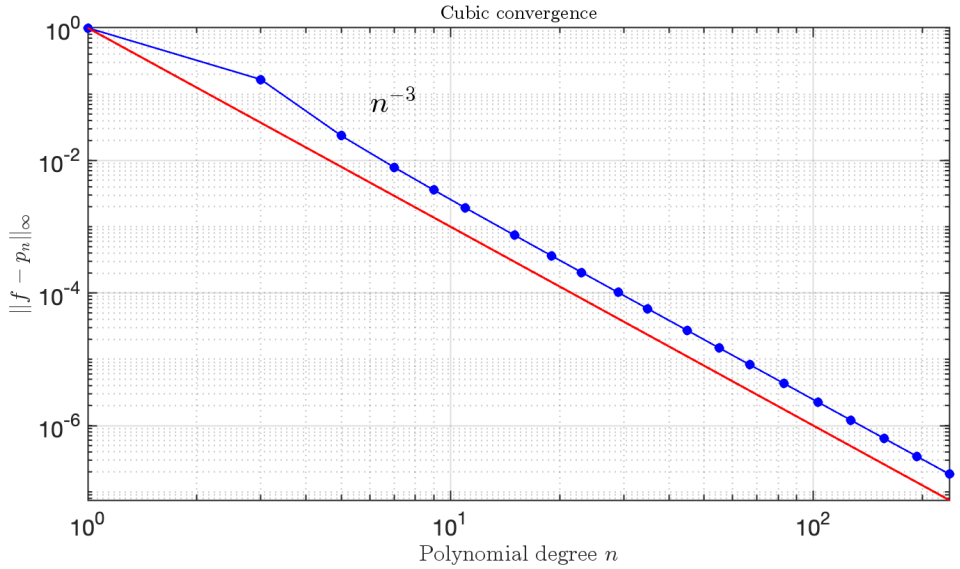


Figure 2.2: Cubic convergence of Chebyshev interpolation for the function $f(x) = |x|^3$. The error $\|f - p_n\|_\infty$ decays at the expected rate $\mathcal{O}(n^{-3})$, as indicated by the red line.

2.2.1 New error estimates for polynomial interpolation in Chebyshev points

Lemma 2.5. *For any positive integers N and m, we have*

$$\sum_{k=N+1}^{\infty} \frac{1}{k(k+1)\cdots(k+m)} = \frac{1}{m(N+1)(N+2)\cdots(N+m)}. \tag{2.6}$$

Proof. Since

$$\frac{1}{k(k+1)\cdots(k+m)} = \frac{1}{m}\left(\frac{1}{k(k+1)\cdots(k+m-1)} - \frac{1}{(k+1)(k+2)\cdots(k+m)}\right),$$

(2.6) follows directly from the sum of the above identity for $k = N+1, N+2, \cdots$. □

If $f, f', \ldots, f^{(\nu-1)}$ are absolutely continuous on $[-1,1]$ and $V < \infty$, it follows from Theorem (2.2) and Lemma (2.5) that for $n \geq \nu + 1$

$$\sum_{k=n+1}^{\infty} |a_k| \leq \frac{2V}{\pi} \sum_{k=n+1}^{\infty} \frac{1}{k(k-1)\cdots(k-\nu)} = \frac{2V}{\nu\pi\, n(n-1)\cdots(n+1-\nu)}.$$

Theorem 2.6. *If $f, f', \ldots, f^{(\nu-1)}$ are absolutely continuous on $[-1,1]$ and $\|f^{(\nu)}\|_T = V < \infty$ for some $\nu \geq 0$, then for each $n \geq \nu+1$, we have that for $\nu > 2$*

$$\|f' - p_n'\|_\infty \leq \frac{4(n+1)V}{n(\nu-2)\pi(n-2)(n-3)\cdots(n+1-\nu)}$$

Proof. Note that $T_k(x) = \cos(k\cos^{-1}(x))$ for $-1 \leq x \leq 1$ and

$$T_k'(x) = \frac{k\sin(k\cos^{-1}(x))}{\sqrt{1-x^2}} = \frac{k\sin(ku)}{\sin(u)}, \qquad \|T_k'\|_\infty = k^2,$$

$$\|f' - p_n'\|_\infty \leq 2\sum_{k=n+1}^{\infty} |a_k|\, k^2$$

$$\leq \sum_{k=n+1}^{\infty} \frac{4(n+1)V}{n\pi(k-2)(k-3)\cdots(k-\nu)}$$

$$= \frac{4(n+1)V_k}{n(\nu-2)\pi(n-2)(n-3)\cdots(n+1-\nu)}.$$

□

2.3 Convergence for analytic functions

In this section, we consider functions that are smoother than merely C^∞. Recall that a function in C^∞ can be differentiated arbitrarily many times. Analyticity, however, is a stronger property: a function is analytic if, in addition to having derivatives of all orders, its Taylor series converges to the function itself.
This property enables analytic continuation into the complex plane. Around each point of the interval $[-1, 1]$, there exists a disk in which the Taylor series converges. Since $[-1, 1]$ is compact, finitely many of these disks cover the entire interval. Consequently, if f is analytic on $[-1, 1]$, then it can be analytically continued to a neighborhood of $[-1, 1]$ in the complex plane. The bigger the neighborhood, the faster the convergence. In particular, for polynomial approximations, the neighborhoods that matter are the regions in the complex plane bounded by ellipses with foci at -1 and 1, known as *Bernstein ellipses* a standard concept in approximation theory [22].

Definition 2.7. *The Bernstein ellipse E_ρ is defined by*

$$E_\rho := \left\{ z \in \mathbb{C} \;\middle|\; z = \tfrac{1}{2}\,(u + u^{-1}),\ |u| = \rho > 1 \right\},$$

which has foci at ± 1, with major and minor semi-axes given by $\frac{1}{2}(\rho+\rho^{-1})$ and $\frac{1}{2}(\rho-\rho^{-1})$, respectively.

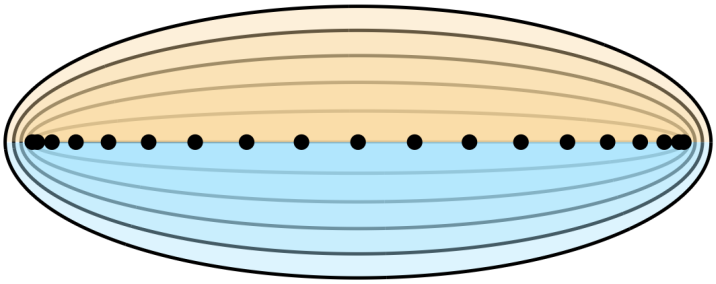


Figure 2.3: Bernstein ellipses for $\rho = 1, 1.2, \ldots, 1.5$, illustrating the growth of the ellipse as ρ increases.

As $\rho \to 1$, the Bernstein ellipse reduces to the interval $[-1, 1]$, while for larger ρ it expands further into the complex plane. The Bernstein ellipse is naturally described via the *Joukowski map*

$$x = \tfrac{1}{2}(z + z^{-1}).$$

This mapping sends each circle of radius $\rho > 1$ in the complex z-plane to an ellipse in the x-plane with foci at ± 1. In particular, the unit circle is mapped to the interval $[-1, 1]$. The relevant region is not only the annulus between the unit circle and the circle of

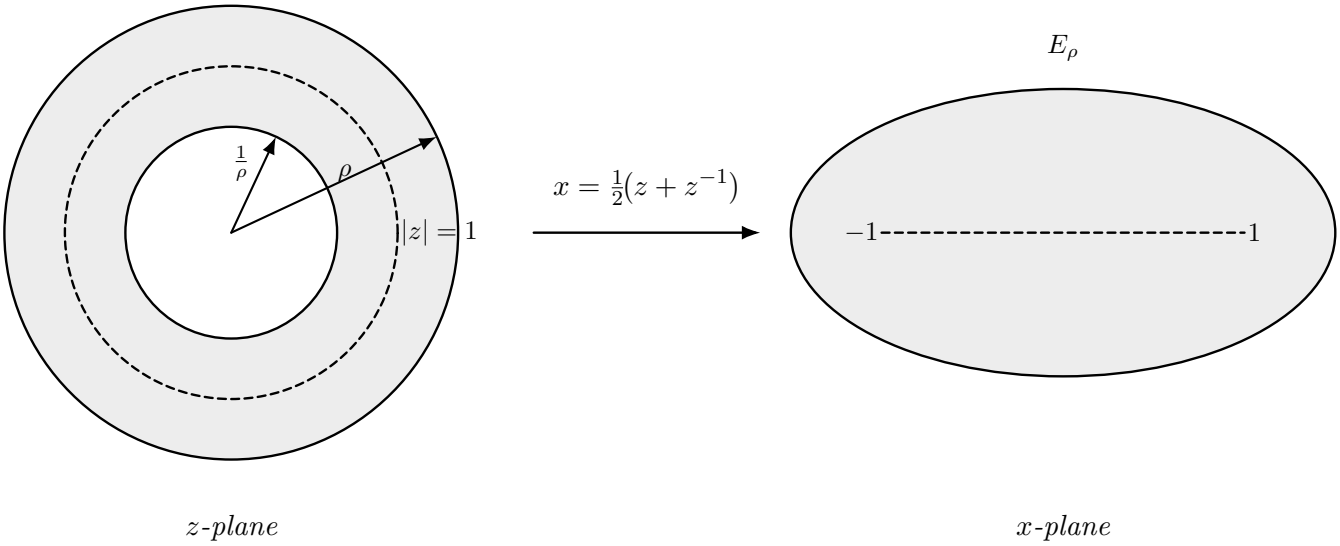


Figure 2.4: The Joukowski map $x = \frac{1}{2}(z + z^{-1})$ sends the annulus $\{\, 1/\rho < |z| < \rho \,\}$ in the z-plane onto the Bernstein ellipse E_ρ in the x-plane; the unit circle $|z| = 1$ collapses to $[-1, 1]$, so the mapping is two-to-one.

radius ρ, but also the annulus between the unit circle and the circle of radius $1/\rho$. Hence the Joukowski map is a two-to-one mapping from

$$\{\, z \in \mathbb{C} : 1/\rho < |z| < \rho \,\}$$

onto the ellipse E_ρ. We begin with the fundamental bound on Chebyshev coefficients of analytic functions, from which many further results follow.

Theorem 2.8 ([22, Theorem 8.1])**.** *Let f be an analytic function in $[-1, 1]$ and analytically continuable to the open Bernstein ellipse, where it satisfies $|f(x)| \leq M$. Then its Chebyshev coefficients satisfy*

$$|a_0| \leq M \quad \textit{and} \quad |a_k| \leq 2M\rho^{-k} \quad \textit{for } k \geq 1. \tag{2.7}$$

Proof. To prove results about Chebyshev coefficients, it is convenient to transform to a Laurent setting. Let $f(x)$ be analytic in a Bernstein ellipse, so that points of the ellipse correspond pointwise to values of f.
First, recall that the unit interval corresponds to the unit circle under the Joukowski map. A point on the unit circle maps to a point in $[-1, 1]$. If instead we take a point on the boundary of the ellipse, this corresponds to a point z on the circle $|z| = \rho > 1$. At the same time, the reciprocal point z^{-1} lies on the smaller circle $|z| = \rho^{-1}$, and both map to the same x. Thus we can write

$$F(z) = f\left(\tfrac{1}{2}(z + z^{-1})\right),$$

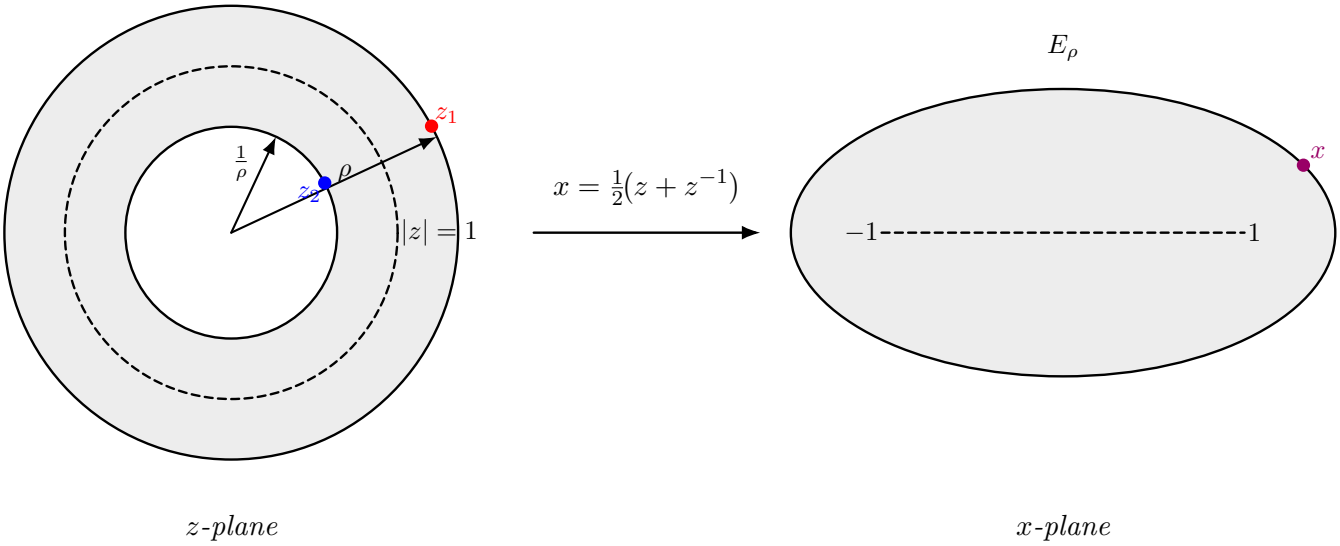


Figure 2.5: Two points $z \in \{|z| = \rho\}$ and $z^{-1} \in \{|z| = 1/\rho\}$ map under the Joukowski transformation to the same point x on the Bernstein ellipse E_ρ, illustrating the 2–1 nature of the mapping.

where $x = \frac{1}{2}(z + z^{-1})$. In particular, $F(z) = F(z^{-1})$. Since the map $z \mapsto \frac{1}{2}(z + z^{-1})$ is analytic in the annulus, and f is analytic in the ellipse, their composition F is analytic in the annulus.

The function F therefore admits a Laurent expansion of the form

$$F(z) = \sum_{k=0}^{\infty} \tfrac{a_k}{2}\,(z^k + z^{-k}),$$

where

$$a_k = \frac{1}{\pi i} \int_{|z|=1} z^{-(k+1)} F(z)\, dz,$$

with the denominator replaced by $2\pi i$ in the case $k = 0$.

Since f is analytic in the ellipse and F is analytic in the annulus, we can deform the contour of integration. By Cauchy's theorem,

$$a_k = \frac{1}{\pi i} \int_{|z|=1} z^{-(k+1)} F(z)\, dz \;=\; \frac{1}{\pi i} \int_{|z|=s} z^{-(k+1)} F(z)\, dz, \qquad 1 \le s < \rho,$$

where the restriction $s < \rho$ reflects that F may not be analytic on the boundary.

Setting $M = \max_{|z|=s} |F(z)|$, we obtain

$$|a_k| \le \frac{1}{\pi}\,(2\pi s)\, M\, s^{-(k+1)} = 2M\, s^{-k}.$$

Suppose now, in addition, that F is analytic not only in the annulus but also on its boundary, i.e. for $\rho^{-1} \le |z| \le \rho$. Then we may choose $s = \rho$ in the above, giving the sharper bound

$$|a_k| \le 2M_\rho \, \rho^{-k}, \qquad M_\rho = \max_{|z|=\rho} |F(z)|.$$

This establishes the exponential decay of Chebyshev coefficients. □

Theorem 2.9 ([22, Theorem 8.2]). *If f has the properties of Theorem 8.1, then for each $n \ge 0$ its Chebyshev projections satisfy*

$$\|f - f_n\| \le \frac{2M\rho^{-n}}{\rho - 1}, \tag{2.8}$$

and its Chebyshev interpolants satisfy

$$\|f - p_n\| \le \frac{4M\rho^{-n}}{\rho - 1}. \tag{2.9}$$

Proof. For (2.8), applying Theorem (2.8) to (1.14) yields

$$\|f - f_n\| \le \sum_{k=n+1}^{\infty} |a_k| \le 2M \sum_{k=n+1}^{\infty} \rho^{-n} = 2M \frac{\rho^{-n-1}}{1 - \frac{1}{\rho}} = \frac{2M\rho^{-n}}{\rho - 1}.$$

For (2.9), we use the fact that the approximation error of the interpolant can be bounded by twice the bound appearing in the case of truncation (see equation (1.13))

$$\|f - p_n\| \le 2\|f - f_n\| \le \frac{4M\rho^{-n}}{\rho - 1}. \tag{2.10}$$

□

To illustrate the theorem, consider Chebyshev interpolants of the Runge function $f(x) = (1 + 25x^2)^{-1}$. The errors decay at a geometric rate $\mathcal{O}(\rho^{-n})$, producing an almost straight line on a semilog plot and continuing steadily down to the level of rounding errors.

Listing 2.2: MATLAB code used in to produce Fig (2.6).

```
% Geometric convergence for the Runge function

% Define function
x = chebfun('x');
f = 1./(1 + 25*x.^2);

% Degrees
nn = 0:10:200;
ee = zeros(size(nn));

% Compute errors
for j = 1:length(nn)
```

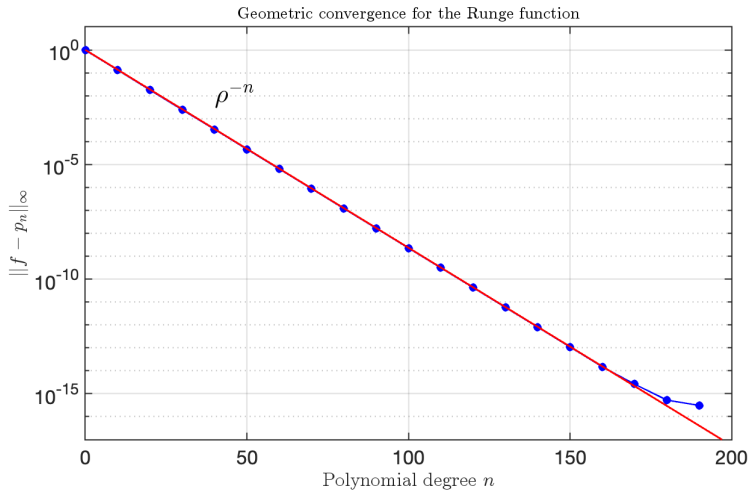


Figure 2.6: Geometric convergence of Chebyshev interpolants for the Runge function $f(x) = (1 + 25x^2)^{-1}$.

```
    n = nn(j);
    fn = chebfun(f, n+1);
    ee(j) = norm(f - fn, inf);
end

% Reference slope ~ rho^{-n}, where rho is the Bernstein ellipse parameter
rho = (1 + sqrt(26))/5;
refLine = rho.^(-nn);

% Plot
figure;
semilogy(nn, ee, 'bo-', 'MarkerFaceColor', 'b', 'LineWidth', 1.2); hold on;
semilogy(nn, refLine, 'r-', 'LineWidth', 1.5);

% Labels & annotations
grid on;
set(gca, 'FontSize', 16, 'LineWidth', 1);
xlabel('Polynomial degree $n$', 'Interpreter', 'latex', 'FontSize', 16);
ylabel('$\| f - p_n \|_{\infty}$', 'Interpreter', 'latex', 'FontSize', 16);
title('Geometric convergence for the Runge function', 'Interpreter', 'latex', 'FontSize', 14);
axis([0 200 1e-17 10]);
% Optional annotation:
text(40, 1e-2, '$\rho^{-n}$', 'FontSize', 22, 'Color', 'k', 'Interpreter', 'latex');
```

Remark 2.10. If f is analytic not just on $[-1, 1]$ but in the whole complex plane— such a function is said to be *entire*—then the convergence is even faster than geometric. Here, for the function $\cos(20x)$, the dots are not approaching a fixed straight line but a curve that gets steeper as n increases, until rounding errors cut off the progress.

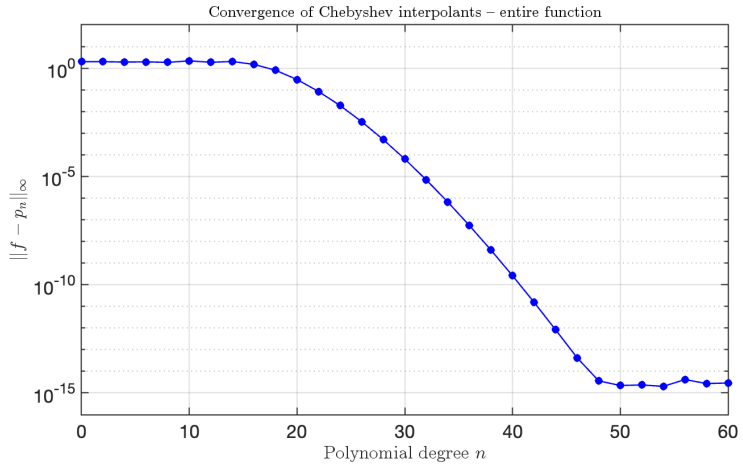


Figure 2.7: Convergence of Chebyshev interpolants for the entire function $f(x) = \cos(20x)$.

Listing 2.3: MATLAB code used in Remark 2.10.

```
% Entire function convergence study with Chebyshev interpolation

% Define function
x = chebfun('x');
f = cos(20*x);

% Degrees
nn = 0:2:60;
ee = zeros(size(nn));

% Compute errors
for j = 1:length(nn)
    n = nn(j);
    fn = chebfun(f, n+1);
    ee(j) = norm(f - fn, inf);
end

% Plot
figure;
semilogy(nn, ee, 'bo-', 'MarkerFaceColor', 'b', 'LineWidth', 1.2);

% Labels & annotations
grid on;
set(gca, 'FontSize', 16, 'LineWidth', 1);
xlabel('Polynomial degree $n$', 'Interpreter', 'latex', 'FontSize', 16);
ylabel('$\| f - p_n \|_{\infty}$', 'Interpreter', 'latex', 'FontSize', 16);
title('Convergence of Chebyshev interpolants -- entire function', ...
    'Interpreter', 'latex', 'FontSize', 14);
axis([0 60 1e-16 100]);
```

We now derive an elegant converse of Theorem 2.9, also due to Bernstein. The converse

is not quite exact: Theorem 2.9, assumes analyticity and boundedness in E_ρ, whereas the conclusion of Theorem 2.11 is analyticity but not necessarily boundedness.

Theorem 2.11 (Converse of Theorem 2.9)**.** *Suppose f is a function on $[-1,1]$ for which there exist polynomial approximations $\{q_n\}$ satisfying*

$$\|f - q_n\|_{[-1,1]} \leq C\rho^{-n}, \qquad n \geq 0,$$

for some constants $\rho > 1$ and $C > 0$. Then f can be analytically continued to an analytic function in the open Bernstein ellipse E_ρ.

Proof. We assume that

$$\|f - q_n\|_{[-1,1]} \leq C\rho^{-n}, \qquad n \geq 0,$$

where $\{q_n\}$ are polynomials and $\rho > 1$.

Step 1. Show successive polynomials get closer on $[-1,1]$. Consider two successive approximations q_n and q_{n-1}. By the triangle inequality,

$$\|q_n - q_{n-1}\|_{[-1,1]} \;\leq\; \|f - q_n\|_{[-1,1]} + \|f - q_{n-1}\|_{[-1,1]}.$$

Using the assumption,

$$\|q_n - q_{n-1}\|_{[-1,1]} \;\leq\; C\rho^{-n} + C\rho^{-(n-1)} \;\leq\; 2C\rho^{\,1-n}.$$

Step 2. Extend the estimate from $[-1,1]$ to ellipses. The key tool is *Bernstein–Walsh inequality*: if p is a polynomial of degree at most n, then for any ellipse E_s with parameter $s > 1$,

$$\|p\|_{E_s} \;\leq\; s^n \|p\|_{[-1,1]}.$$

Apply this to $p := q_n - q_{n-1}$ (which has degree $\leq n$):

$$\|q_n - q_{n-1}\|_{E_s} \;\leq\; s^n \|q_n - q_{n-1}\|_{[-1,1]}.$$

Step 3. Combine the bounds. From Step 1 we know $\|q_n - q_{n-1}\|_{[-1,1]} \leq 2C\rho^{\,1-n}$. Therefore,

$$\|q_n - q_{n-1}\|_{E_s} \;\leq\; 2Cs^n\rho^{\,1-n}.$$

Rewriting,

$$\|q_n - q_{n-1}\|_{E_s} \;\leq\; 2C\rho\left(\frac{s}{\rho}\right)^n.$$

Step 4. Interpret as a series expansion of f. On the interval $[-1,1]$, we can telescope:

$$f = q_0 + (q_1 - q_0) + (q_2 - q_1) + \cdots.$$

This holds pointwise because $q_n \to f$ uniformly on $[-1,1]$.

Now, for each n, the difference $q_n - q_{n-1}$ is a polynomial and hence analytic everywhere, in particular on E_s. Thus, the right-hand side is a series of analytic functions on E_s.

Step 5. Show convergence of the series on E_s. We have

$$\|q_n - q_{n-1}\|_{E_s} \;\leq\; 2C\rho\left(\frac{s}{\rho}\right)^n.$$

Since $s < \rho$, the ratio $\frac{s}{\rho}$ is strictly less than 1. Thus the sequence $M_n = 2C\rho(s/\rho)^n$ decays geometrically, and

$$\sum_{n=1}^{\infty} M_n < \infty.$$

By the *Weierstrass M-test*, the series

$$(q_1 - q_0) + (q_2 - q_1) + \cdots$$

converges uniformly on E_s. This is crucial, because uniform convergence preserves analyticity in the next step.

Step 6. Uniform limit of analytic functions is analytic. Each difference $q_n - q_{n-1}$ is a polynomial, hence analytic on all of $\mathbb{C}$. The uniform limit of analytic functions on a domain is again analytic there (a standard theorem from complex analysis). Therefore, the series sum defines an analytic function F on E_s.
On $[-1, 1]$, the partial sums telescope:

$$q_0 + (q_1 - q_0) + \cdots + (q_n - q_{n-1}) = q_n,$$

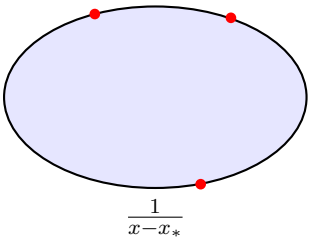
and $q_n \to f$ uniformly on $[-1, 1]$. Thus $F = f$ on the real interval. Hence f has been analytically continued from $[-1, 1]$ to the ellipse E_s.

Step 7. Let s approach ρ. Since this argument works for any $s < \rho$, we can cover the entire open ellipse E_ρ. Therefore, f extends to an analytic function on E_ρ. □

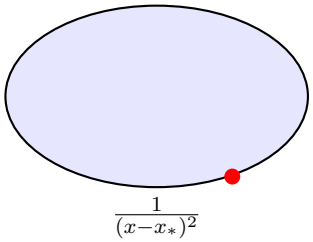
Remark 2.12. The subtlety here is that analyticity in E_ρ does not imply boundedness. For example, suppose f is analytic in E_ρ but has finitely many *simple poles* on the boundary ∂E_ρ. In this case, f is unbounded on ∂E_ρ, but the polynomial approximations still converge geometrically at the rate $O(\rho^{-n})$. This is the example where Theorem 2.9 does not tell you get geometric convergence.
By contrast, if f has *double poles* on ∂E_ρ, the situation worsens. Instead of the exact rate ρ^{-n}, one obtains only a slower geometric convergence $O(\tilde{\rho}^{-n})$ for some $\tilde{\rho} < \rho$. Thus simple poles on the boundary preserve the optimal convergence rate, whereas higher-order poles degrade it.

Putting both last theorems together establishes a simple fact

Theorem 2.13 (Bernstein's Theorem [3])**.** *A function f defined on $[-1, 1]$ can be approximated by polynomials with geometric (exponential) accuracy if and only if it is analytic in a neighborhood of $[-1, 1]$.*



(a) Simple poles on ∂E_ρ



(b) A double pole on ∂E_ρ

Figure 2.8: Analyticity in E_ρ with poles on the boundary. Simple poles preserve convergence rate $O(\rho^{-n})$, whereas double poles reduce the rate to $O(\tilde{\rho}^{-n})$ with $\tilde{\rho} < \rho$.

Where singularities matter most

In practice, not all singularities affect convergence in the same way. Two types of locations on the Bernstein ellipse are especially important:

1. singularities on the real axis near an endpoint of $[-1, 1]$,
2. singularities on the imaginary axis near the middle of the interval.

These two cases behave quite differently.

Case 1: Real axis, near an endpoint. Suppose f has a singularity at $x = \alpha > 1$. Under the inverse Joukowski map this corresponds to

$$\rho = \alpha + \sqrt{\alpha^2 - 1}. \tag{2.11}$$

Take $\alpha = 1 + \varepsilon$ with $\varepsilon \ll 1$. Then

$$\rho = 1 + \varepsilon + \sqrt{2\varepsilon + \varepsilon^2}.$$

Expanding the square root[2]:,

$$\sqrt{2\varepsilon + \varepsilon^2} = \sqrt{2\varepsilon}\,\sqrt{1 + \tfrac{\varepsilon}{2}} \sim \sqrt{2\varepsilon},$$

so

$$\rho \sim 1 + \sqrt{2\varepsilon}.$$

Thus even if the singularity is very close to the endpoint, ρ is not extremely close to 1. In practice, the convergence is only mildly slowed.

[2] $$\sqrt{1+x} \;=\; 1 + \tfrac{1}{2}x - \tfrac{1}{8}x^2 + O(x^3),$$

Case 2: Imaginary axis, near the middle. Now suppose f has a singularity at $x = i\beta$. From the inverse Joukowski map we find

$$z = i\beta + \sqrt{(i\beta)^2 - 1} = i\beta + i\sqrt{\beta^2 + 1},$$

so

$$\rho = |z| = \beta + \sqrt{\beta^2 + 1}. \tag{2.12}$$

For $\beta = \varepsilon \ll 1$ we get

$$\rho = \varepsilon + \sqrt{1 + \varepsilon^2} = \varepsilon + \left(1 + \tfrac{\varepsilon^2}{2} + O(\varepsilon^4)\right) \sim 1 + \varepsilon.$$

Here ρ is extremely close to 1, and the convergence is dramatically slowed. Much higher degree is required to reach the same accuracy.

Conclusion.

- Singularities near the *endpoints* give $\rho - 1 \sim \sqrt{2\varepsilon}$, only a mild slowdown.
- Singularities near the *middle* give $\rho - 1 \sim \varepsilon$, which is far worse.

Hence, in practice, singularities on the real axis near ± 1 are less harmful than those near the middle of the interval. To illustrate this principle, we now turn to two concrete examples. Both show how the distance and location of the nearest singularity control the convergence rate of Chebyshev interpolants.

Example 2.14 (Singularities on the real axis.)**.** We consider

$$f(x) = \sqrt{2 - x},$$

which has a branch point singularity at $x = 2$.

From the formula (2.11) we have

$$\rho = 2 + \sqrt{3}.$$

This gives the convergence rate

$$\mathcal{O}((2 + \sqrt{3})^{-n}) \approx \mathcal{O}(3.732^{-n}),$$

Listing 2.4: MATLAB code used in to produce Fig 2.9.

```
% Geometric convergence for f(x) = sqrt(2-x)

% Define function
x = chebfun('x');
f = sqrt(2 - x);

% Degrees
nn = 0:30;
```

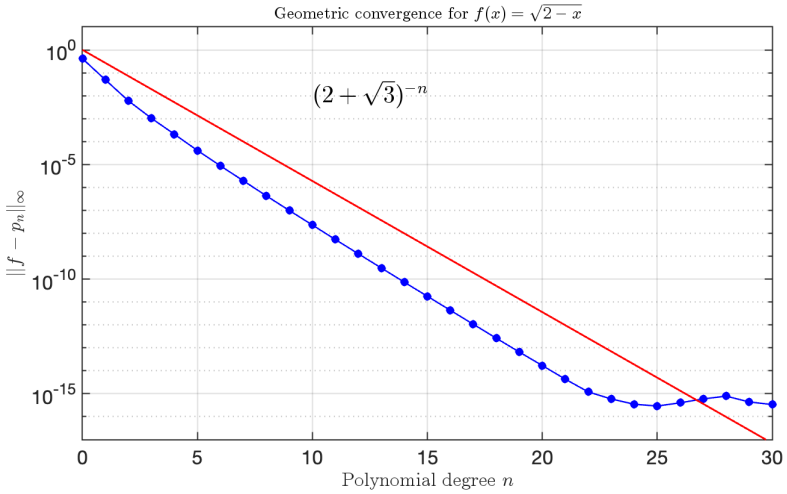


Figure 2.9: Convergence of Chebyshev interpolants for $f(x) = \sqrt{2-x}$.

```
ee = zeros(size(nn));

% Compute errors
for j = 1:length(nn)
    n = nn(j);
    fn = chebfun(f, n+1);
    ee(j) = norm(f - fn, inf);
end

% Reference slope ~ rho^{-n}, with rho = 2 + sqrt(3)
rho = 2 + sqrt(3);
refLine = rho.^(-nn);

% Plot
figure;
semilogy(nn, ee, 'bo-', 'MarkerFaceColor', 'b', 'LineWidth', 1.2); hold on;
semilogy(nn, refLine, 'r-', 'LineWidth', 1.5);

% Labels & annotations
text(10, 1e-2, '$(2+\sqrt{3})^{-n}$', 'FontSize', 20, ...
    'Color', 'k', 'Interpreter', 'latex');

grid on;
set(gca, 'FontSize', 16, 'LineWidth', 1);
xlabel('Polynomial degree $n$', 'Interpreter', 'latex', 'FontSize', 16);
ylabel('$\| f - p_n \|_{\infty}$', 'Interpreter', 'latex', 'FontSize', 16);
title('Geometric convergence for $f(x)=\sqrt{2-x}$', ...
    'Interpreter', 'latex', 'FontSize', 14);
axis([0 30 1e-17 10]);
```

Fig. (2.9) confirms this behavior: the error curve aligns with the predicted slope ρ^{-n} as n increases. This illustrates that, in practice, singularities on the real axis near ± 1 are

less harmful than singularities near the middle of the interval, since they allow a larger Bernstein ellipse and hence faster convergence.

Example 2.15 (Singularities on the imaginary axis.)**.** Consider first

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

which has singularities at $x = \pm i/5$. Using the formula (2.12) with $\beta = \frac{1}{5}$, we obtain

$$\rho = \frac{1+\sqrt{26}}{5} \approx 1.2198,$$

so the convergence rate is

$$\mathcal{O}(1.2198^{-n}).$$

Since ρ is only slightly above 1, this convergence is relatively slow.

Next, consider the case

$$f(x) = \frac{1}{1+64x^2},$$

with singularities at $x = \pm i/8$. Applying formula (2.12), we have

$$\rho = \frac{1+\sqrt{65}}{8} \approx 1.13278.$$

Thus the convergence rate is

$$\mathcal{O}(1.1328^{-n}),$$

This is even slower than in the previous case, since the poles lie closer to the real axis.

Fig. (2.10) illustrates these results: the errors of the Chebyshev interpolants for both Runge-type functions decay geometrically, with the observed rate precisely determined by the distance of the nearest singularities from the interval $[-1, 1]$.

Listing 2.5: MATLAB code used in to produce Fig 2.10.

```
% Effect of singularities on the imaginary axis near the ellipse
% on geometric convergence of Runge-type functions.

% Define functions
x = chebfun('x');
f1 = 1./(1 + 25*x.^2);     % Runge function with 25
f2 = 1./(1 + 64*x.^2);     % Runge function with 64

% Degrees
nn = 0:10:200;
ee1 = zeros(size(nn));
ee2 = zeros(size(nn));

% Compute errors for f1
```

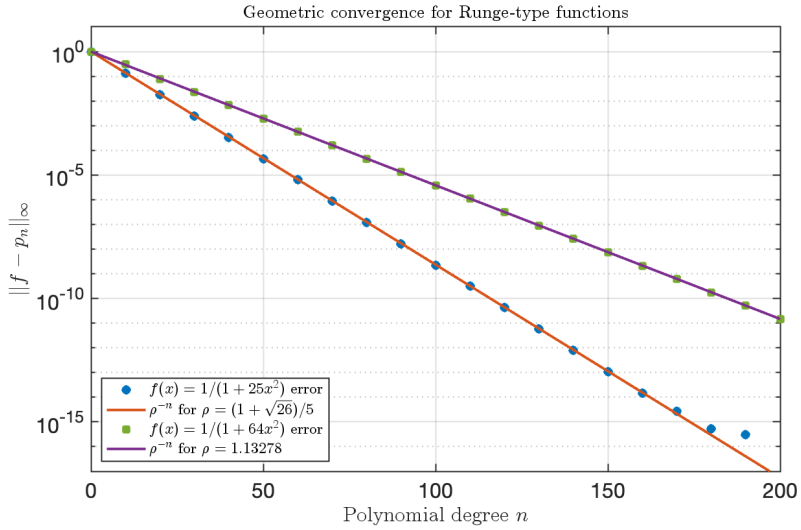


Figure 2.10: Geometric convergence of Chebyshev interpolants for the Runge-type functions $f(x) = \frac{1}{1+25x^2}$ and $f(x) = \frac{1}{1+64x^2}$.

```
for j = 1:length(nn)
    n = nn(j);
    fn = chebfun(f1, n+1);
    ee1(j) = norm(f1 - fn, inf);
end

% Compute errors for f2
for j = 1:length(nn)
    n = nn(j);
    fn = chebfun(f2, n+1);
    ee2(j) = norm(f2 - fn, inf);
end

% Reference slopes
rho1 = (1 + sqrt(26))/5;
rho2 = 1.13278;
refLine1 = rho1.^(-nn);
refLine2 = rho2.^(-nn);

% Plot
figure;
semilogy(nn, ee1, 'o', 'Color', [0 0.45 0.74], 'MarkerFaceColor', [0 0.45 0.74],...
'LineWidth', 1.5); hold on; % nice blue
semilogy(nn, refLine1, '-', 'Color', [0.85 0.33 0.10], 'LineWidth', 2); % orange
semilogy(nn, ee2, 's', 'Color', [0.47 0.67 0.19], 'MarkerFaceColor', [0.47 0.67 0.19], ...
'LineWidth', 1.5); % green
semilogy(nn, refLine2, '-', 'Color', [0.49 0.18 0.56], 'LineWidth', 2); % purple

% Labels & annotations
grid on;
```

```
set(gca, 'FontSize', 16, 'LineWidth', 1);
xlabel('Polynomial degree $n$', 'Interpreter', 'latex', 'FontSize', 16);
ylabel('$\| f - p_n \|_{\infty}$', 'Interpreter', 'latex', 'FontSize', 16);
title('Geometric convergence for Runge-type functions',...
'Interpreter', 'latex', 'FontSize', 14);
axis([0 200 1e-17 10]);

legend({'$f(x) = 1/(1+25x^2)$ error', ...
        '$\rho^{-n}$ for $\rho = (1+\sqrt{26})/5$', ...
        '$f(x) = 1/(1+64x^2)$ error', ...
        '$\rho^{-n}$ for $\rho = 1.13278$'}, ...
        'Interpreter','latex','FontSize',12, 'Location','southwest');
```

3 Barycentric interpolation formula

Lagrange interpolation is one of the simplest and clearest ideas in approximation theory. Its classical form, however, is often considered inefficient for numerical work (see Acton [1]).. To see how the barycentric formula overcomes this limitation, we begin by reviewing the standard Lagrange construction.

3.1 Lagrange interpolation and barycentric formula

Let $n+1$ distinct interpolation points (nodes) x_j, $j = 0, \ldots, n$, be given together with corresponding values f_j, which may or may not be samples of a function f.
Let P_n denote the space of all polynomials of degree at most n. The classical interpolation problem consists in finding a polynomial $p \in P_n$ such that

$$p(x_j) = f_j, \quad j = 0, \ldots, n. \tag{3.1}$$

This problem is well-posed: there exists a unique solution that depends continuously on the data. As is standard in numerical analysis, the solution can be expressed in *Lagrange form*:

$$p_n(x) = \sum_{j=0}^{n} f_j \, \ell_j(x), \qquad \ell_j(x) = \frac{\displaystyle\prod_{\substack{k=0 \\ k \neq j}}^{n} (x - x_k)}{\displaystyle\prod_{\substack{k=0 \\ k \neq j}}^{n} (x_j - x_k)}. \tag{3.2}$$

The *Lagrange basis polynomial* ℓ_j associated with node x_j satisfies the interpolation property

$$\ell_j(x_k) = \begin{cases} 1, & j = k, \\ 0, & j \neq k, \end{cases} \qquad j, k = 0, \ldots, n. \tag{3.3}$$

Although Lagrange's formula is theoretically elegant, it is often considered inefficient for practical computation. Common criticisms include:

1. Each evaluation of $p_n(x)$ requires $\mathcal{O}(n^2)$ additions and multiplications.
2. Adding a new data point (x_{n+1}, f_{n+1}) requires recomputing the entire polynomial.
3. The method is sometimes regarded as numerically unstable.

From this observation, it is often concluded that the Lagrange form of p serves mainly as a theoretical construct for proving theorems.
On the contrary, the Lagrange formulation is, in most cases, the method of choice for constructing polynomial interpolants. The essential insight is that the Lagrange polynomial should be expressed and evaluated through the formulas of *barycentric interpolation*. Although barycentric interpolation is not a new concept, it remains unfamiliar to many students, mathematicians, and even numerical analysts. This elegant and powerful approach deserves a central place in introductory courses and textbooks on numerical analysis.
For decades, the significance of barycentric interpolation has been largely underappreciated. Indeed, throughout the second half of the twentieth century, many numerical analysis textbooks continued to repeat statements about polynomial interpolation that are, in fact, incorrect. For example:

- *"Polynomial interpolation does not converge."* — This is false when using Chebyshev points.
- *"Polynomial interpolation is numerically difficult to compute."* — This is also false when employing the barycentric formula or other stable methods.

As a result, a certain pessimism has persisted at the heart of numerical analysis since the dawn of the computational era. Even today, if one consults several numerical analysis textbooks, it is unlikely to find "barycentric interpolation" listed in the index.
Let us now state the theorem for barycentric interpolation using Chebyshev points. Later, we will generalize this result to arbitrary interpolation nodes.

Theorem 3.1 (Barycentric Formula for Chebyshev Interpolation)**.** *Let $x_0, \dots, x_n$ be the Chebyshev nodes on $[-1, 1]$, and let $f_0, \dots, f_n$ be the corresponding data values. The degree-n Chebyshev interpolant p satisfying $p(x_j) = f_j$ for $j = 0, \dots, n$ can be expressed in the barycentric form*

$$p_n(x) = \frac{\sum_{j=0}^{n}{}' \dfrac{(-1)^j f_j}{x - x_j}}{\sum_{j=0}^{n}{}' \dfrac{(-1)^j}{x - x_j}}, \tag{3.4}$$

where the primes indicate that the first and last terms ($j = 0$ and $j = n$) are multiplied by $\frac{1}{2}$.

At first glance, this formula may appear numerically unstable because of the denominators $(x - x_j)$, which become small when x is close to one of the nodes. Surprisingly, however, it turns out to be *numerically stable* [11]. Not only is it stable, but it is also *highly efficient*: the evaluation of $p_n(x)$ requires only $\mathcal{O}(n)$ floating-point operations for a single point x, since both the numerator and denominator involve just $n + 1$ terms.
In other words, interpolating from $n + 1$ data points is achieved by summing $n + 1$ numbers in each expression—an operation whose computational cost is linear in n. One cannot do better, since the data itself already contains $\mathcal{O}(n)$ information.

How not to evaluate the interpolation: A common but misguided approach is to use `polyval` and `polyfit` in MATLAB, which forms the Vandermonde matrix and computes the interpolant in the *monomial basis*. This method is numerically unstable: the error grows exponentially with the degree, roughly like 2^n, so many digits of accuracy are lost even for moderate n.
Although this algorithm is poor, it is the most obvious one—because it mirrors how polynomials are usually written. The problem, however, is not with polynomial interpolation itself but with this *naive algorithm*.

The Vandermonde case. To interpolate data (x_j, f_j), $j = 0, \dots, n$, one assumes

$$p_n(x) = a_0 + a_1 x + \cdots + a_n x^n,$$

and imposes $p(x_j) = f_j$, leading to

$$\underbrace{\begin{pmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^n \\ 1 & x_1 & x_1^2 & \cdots & x_1^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^n \end{pmatrix}}_{V} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} f_0 \\ f_1 \\ \vdots \\ f_n \end{pmatrix},$$

where V is the *Vandermonde matrix*.

This method is numerically disastrous:

- V is highly ill-conditioned for large n or equally spaced nodes;
- its condition number grows exponentially with n.

Remark 3.2. The fundamental cause of this instability is the basis itself. The monomials $1, x, x^2, \dots, x^n$ form a complete system of functions, but they are a **tremendously non orthogonal** one: higher powers overlap so strongly on $[-1, 1]$ that they behave almost like scaled versions of one another. If you have an orthogonal set, Chebyshev polynomials for instance you cannot leave out a single one, because no one can take over the job of the other. But with the powers, it's amazing: if you leave out a power or even an infinite set of them according to the Müntz–Szász theorem,[1] you don't lose anything. A basis that remains complete despite the removal of many elements must be dangerously close to linear dependence, and this is precisely what the Vandermonde matrix reveals through its explosive condition number.

[1]

Theorem 3.3 (Müntz–Szász)**.** *Let* $\{0 = \lambda_0 < \lambda_1 < \cdots\}$ *be a sequence of real numbers satisfying* $\lim_{i\to\infty} \lambda_i = \infty$. *Then* $\mathrm{Span}_{\mathbb{C}}\{x^{\lambda_i} : i \in \mathbb{N} \cup \{0\}\}$ *is dense in* $C([0, 1])$ *if and only if*

$$\sum_{i=1}^{\infty} \frac{1}{\lambda_i} = \infty.$$

Let us now state the theorem for barycentric interpolation using arbitrary interpolation points.

Theorem 3.4 (Barycentric interpolation formula)**.** *The polynomial interpolant through data* $\{f_j\}$ *at* $n+1$ *distinct points* $\{x_j\}$ *is given by*

$$p_n(x) = \frac{\displaystyle\sum_{j=0}^{n} \frac{\lambda_j f_j}{x - x_j}}{\displaystyle\sum_{j=0}^{n} \frac{\lambda_j}{x - x_j}}, \tag{3.5}$$

with the special case $p_n(x) = f_j$ *if* $x = x_j$ *for some* j*, where the weights* $\{\lambda_j\}$ *are defined by*

$$\lambda_j = \frac{1}{\displaystyle\prod_{k \neq j} (x_j - x_k)}. \tag{5.12}$$

Proof. Hidden in formula (3.5) is the Lagrange interpolation representation of the interpolant

$$p_n(x) = \sum_{j=0}^{n} f_j \, \ell_j(x),$$

with

$$\ell_j(x) = \frac{\dfrac{\lambda_j}{x - x_j}}{\displaystyle\sum_{k=0}^{n} \frac{\lambda_k}{x - x_k}}.$$

To prove the theorem, we need to show that the above $\ell_j(x)$ is indeed the Lagrange basis polynomial, i.e.,

$$\ell_j(x_k) = \begin{cases} 1, & k = j, \\ 0, & k \neq j, \end{cases} \tag{5.2}$$

and that it is a polynomial of degree n.

First, as $x \to x_k$ with $k \neq j$, the denominator $\displaystyle\sum_{k=0}^{n} \frac{\lambda_k}{x - x_k}$ diverges to infinity, hence $\ell_j(x_k) \to 0$. As $x \to x_j$, both the numerator and denominator blow up at the same rate, dominated by the term corresponding to $k = j$. Therefore, the limit is one, i.e., $\ell_j(x_j) = 1$.

It is precisely the choice

$$\lambda_j = \frac{1}{\displaystyle\prod_{k \neq j} (x_j - x_k)} \tag{3.6}$$

that ensures $\ell_j(x)$ is a polynomial. Choosing any other set of λ_j would, in general, lead to a rational interpolant. We start from the standard Lagrange form

$$\ell_j(x) = \prod_{\substack{k=0 \\ k\neq j}}^{n} \frac{x - x_k}{x_j - x_k}, \qquad j = 0, 1, \ldots, n.$$

Define the *node polynomial*

$$\ell(x) = \prod_{k=0}^{n} (x - x_k) \in P_{n+1}.$$

Since

$$\ell(x) = (x - x_j) \prod_{\substack{k=0 \\ k\neq j}}^{n} (x - x_k),$$

we can rewrite

$$\ell_j(x) = \frac{\prod_{k\neq j}(x - x_k)}{\prod_{k\neq j}(x_j - x_k)} = \frac{\ell(x)}{(x - x_j)\prod_{k\neq j}(x_j - x_k)}.$$

Hence

$$\ell_j(x) = \frac{\ell(x)}{(x - x_j)}\,\lambda_j.$$

Now, observe that

$$1 = \sum_{k=0}^{n} \ell_k(x),$$

and thus

$$1 = \sum_{k=0}^{n} \ell_k(x) = \sum_{k=0}^{n} \frac{\ell(x)}{x - x_k}\,\lambda_k.$$

Factoring out $\ell(x)$ gives

$$\ell_j(x) = \frac{\ell(x)\dfrac{\lambda_j}{x - x_j}}{\ell(x)\sum_{k=0}^{n} \dfrac{\lambda_k}{x - x_k}}.$$

This completes the proof. □

Remark 3.5 (Geometric interpretation of barycentric weights)**.** The barycentric weights can be interpreted geometrically as

$$\lambda_j = \frac{1}{\left[\text{geometric mean distance of } x_j \text{ to all other nodes}\right]^n}.$$

Hence, the magnitude of λ_j reflects how uniformly the nodes $\{x_j\}$ are distributed.

If the interpolation points are well distributed—so that each x_j has approximately the same geometric mean distance to the others—then all barycentric weights λ_j will have comparable magnitudes. This leads to a well-conditioned interpolation formula.
In contrast, for poorly distributed points, such as equally spaced nodes on an interval, the geometric mean distances vary widely—typically by factors on the order of 2^n. Consequently, the barycentric weights λ_j also vary exponentially with n, producing large numerical cancellations and severe ill-conditioning in the interpolation process.
This geometric perspective thus explains why Chebyshev nodes yield stable and accurate interpolation, while equally spaced nodes lead to numerical instability.

Remark 3.6 (First barycentric formula)**.** Equation (3.5) is known as the *second barycentric formula*. However, there also exists a *first barycentric form* [4], obtained by factoring out the node polynomial

$$\ell(x) = \prod_{k=0}^{n} (x - x_k),$$

which yields

$$p_n(x) = \ell(x) \sum_{j=0}^{n} \frac{\lambda_j f_j}{x - x_j}, \tag{3.7}$$

where the barycentric weights are defined by

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

3.2 Other evaluation schemes for the interpolating polynomial

For completeness, we briefly recall two classical recursive schemes, *Aitken's* and *Neville's* algorithms [23].

Aitken's algorithm. A recursive evaluation scheme that builds the interpolating polynomial incrementally:

$$p_n(x) = P_{n,n}, \qquad P_{j,0} = f_j, \quad j = 0, 1, \ldots, n,$$

$$P_{j,d+1} = \frac{(x_j - x)P_{j-1,d} - (x_{j-d-1} - x)P_{j,d}}{x_j - x_{j-d-1}}, \qquad j = d+1, \ldots, n.$$

This method requires no precomputation and is convenient when only a few evaluations of $p_n(x)$ are needed.

Neville's algorithm. A numerically equivalent recursive formulation obtained by rearranging the interpolation steps:

$$p_n(x) = P_{n,n}, \qquad P_{j,0} = f_j, \quad j = 0, 1, \ldots, n,$$

$$P_{j,d+1} = \frac{(x - x_{j-d-1})P_{j,d} - (x - x_j)P_{j-1,d}}{x_j - x_{j-d-1}}, \qquad j = d+1, \ldots, n.$$

The Neville formulation is often preferred for tabular computation since it evaluates the same recursive structure in a slightly more stable order.

Comparison of computational costs

Scheme	**Setup Cost**	**Evaluation Cost**	**Best For**
Lagrange	$O(n^2)$	$O(n^2)$	$-\,-\,-\,-\,-$
Barycentric	$O(n^2)$ (compute λ_j)	$O(n)$ per evaluation	Many evaluations, large n
Aitken / Neville	$O(1)$	$O(n^2)$	Few evaluations, small n

- The barycentric form is the most efficient when the interpolant is to be evaluated at many points.
- Aitken's and Neville's algorithms are preferable for single or few evaluations due to minimal setup cost.
- The Lagrange representation is conceptually simple and analytically transparent, but computationally inefficient.
- There is a trade-off between setup effort and per-evaluation cost: barycentric interpolation requires more preparation but far less work at each evaluation point.

4 Best and near-best approximation

An idea originating with PONCELET (1820s) and later formalized by CHEBYSHEV (1850s) is to determine, for a given continuous function f, the polynomial p^* of prescribed degree n that provides the *best uniform approximation* on an interval. The notion of "best" is understood in the sense of minimizing the $\|\cdot\|_\infty$ norm of the approximation error.

Uniform norm: For a continuous function f on $[-1,1]$, the *uniform* or *supremum norm* is defined by

$$\|f\|_\infty := \sup_{x\in[-1,1]} |f(x)|.$$

Since f is continuous on the compact interval $[-1,1]$, it follows that $|f|$ attains its maximum, and therefore $\|f\|_\infty < \infty$. In what follows, we may write simply $\|f\|$ when the context is clear.

Best Uniform Approximation: Let $C([-1,1])$ denote the space of continuous functions on $[-1,1]$, and let P_n be the space of real polynomials of degree at most n. For a given $f \in C([-1,1])$, the *best approximation polynomial* $p^* \in P_n$ is defined as the minimizer of the uniform norm of the error:

$$\|f - p^*\|_\infty = \min_{p\in P_n} \|f - p\|_\infty.$$

This defines the *best uniform approximation* of f by polynomials of degree n.

Equioscillation pattern: A key property characterizing the best approximation is the so-called *equioscillation* (or *alternation*) pattern of the error function.

Define the *error function* associated with $p \in P_n$ by

$$E(x) := f(x) - p(x), \qquad x \in [-1,1].$$

The **equioscillation principle** asserts that the best approximation polynomial p^* is characterized by the existence of a sequence of $(n+2)$ distinct points

$$-1 \le x_0 < x_1 < \cdots < x_{n+1} \le 1$$

such that

$$E(x_j) = (-1)^j \|E\|_\infty, \qquad j = 0, 1, \ldots, n+1.$$

That is, the error function $E(x)$ attains its maximal deviation from zero at $(n+2)$ points, with alternating signs and equal magnitude (see Fig. (4.1)).

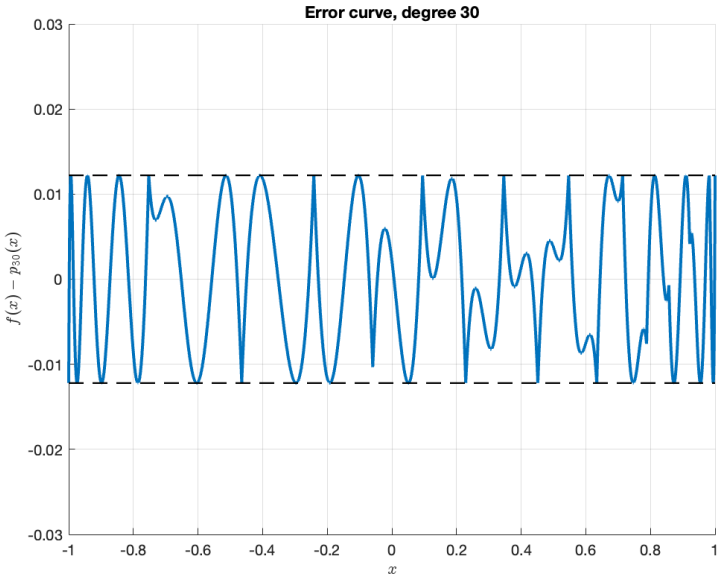


Figure 4.1: Error function $E(x) = f(x) - p_{30}(x)$ on the interval $[-1, 1]$. The dashed horizontal lines show a uniform bound $\|E\|_\infty$. The plot exhibits an *equioscillation pattern*, meaning that $E(x)$ alternates in sign while attaining approximately the same magnitude at a sequence of points: $E(x_0), E(x_1), \ldots, E(x_{n+1}) \approx (\pm)\,\|E\|_\infty, \qquad -1 \le x_0 < \cdots < x_{n+1} \le 1.$ This alternation of extremal values is characteristic of equioscillatory error behavior.

The set of these extremal points,

$$\{x_0, x_1, \ldots, x_{n+1}\},$$

is called an *alternant*. This alternating behavior of the error curve is the defining feature of the best uniform approximation and forms the basis of the **(Equioscillation characterization of best approximants**.

Theorem 4.1 (Equioscillation characterization of best approximants)**.** *Let f be a continuous function on $[-1, 1]$. Then there exists a unique best approximation $p^* \in P_n$ in the uniform norm.*
If f is real-valued, then p^* is real as well, and in this case a polynomial $p \in P_n$ coincides with p^* if and only if the error function

$$E(x) := f(x) - p(x)$$

equioscillates in at least $n + 2$ extreme points; that is, there exist points

$$-1 \le x_0 < x_1 < \cdots < x_{n+1} \le 1$$

such that

$$E(x_j) = (-1)^j \|f - p\|_\infty, \qquad j = 0, 1, \ldots, n+1.$$

Proof. The proof proceeds in four main steps:

1. **Existence.** Show that a best approximation $p_n^* \in P_n$ exists for every continuous $f \in C([-1, 1])$.
2. **Equioscillation ⇒ Optimality.** If the error $E(x) = f(x) - p(x)$ equioscillates at $n + 2$ points, then p is a best uniform approximation.
3. **Optimality ⇒ Equioscillation.** If p^* is a best approximation, then $f - p^*$ equioscillates at least $n + 2$ times.
4. **Uniqueness.** The best approximation is unique, since two distinct minimizers would violate the equioscillation condition.

Existence. Our first step is to confirm the existence of p^*. This will follow from a compactness consideration. Since P_n is a finite-dimensional vector space, every closed and bounded subset of P_n is compact. Define the functional

$$F(p) := \|f - p\|_\infty, \qquad p \in P_n.$$

Continuity of F. For any $p, q \in P_n$,

$$|F(p) - F(q)| = \big|\, \|f - p\|_\infty - \|f - q\|_\infty \,\big| \le \|p - q\|_\infty,$$

so F is continuous with Lipschitz constant 1.

Compactness of the domain. If a best approximation p^* exists, it must lie in the set

$$K := \{\, p \in P_n : \|f - p\|_\infty \le \|f - 0\|_\infty \,\}.$$

Any best approximation cannot perform worse than $p = 0$. The set K is therefore nonempty, bounded, and closed in the finite-dimensional space P_n. By the Bolzano–Weierstrass theorem K is compact. Because F is continuous and K is compact, F attains its minimum on K. Hence, there exists $p^* \in K$ such that

$$\|f - p^*\|_\infty = \min_{p \in P_n} \|f - p\|_\infty.$$

Equioscillation ⇒ Optimality. Suppose $f - p$ equioscillates at at least $n + 2$ points; i.e., there exist

$$-1 \le x_0 < x_1 < \cdots < x_{n+1} \le 1$$

but suppose we have some different polynomial that does better $\|f - q\|_\infty < \|f - p\|_\infty$ for some $q \in P_n$ (see Fig. (6.2)).

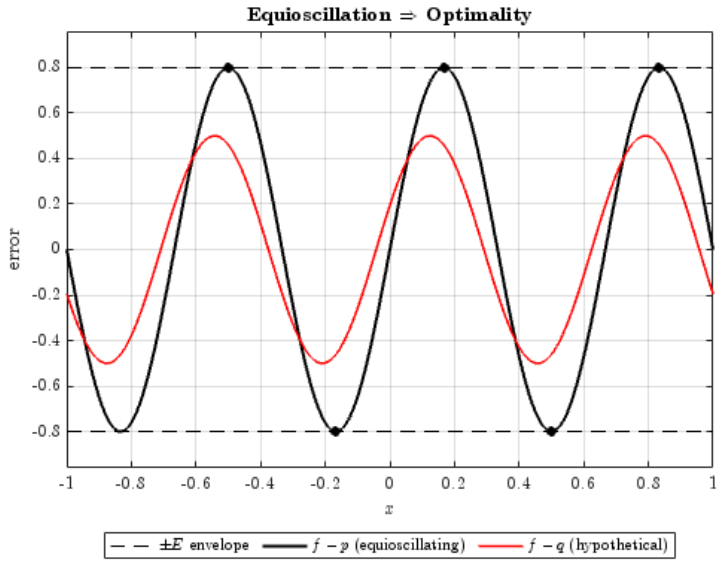


Figure 4.2: Equioscillation implies optimality: the equioscillating error (black) attains $\pm E$ at alternating points, while any other approximant (red) cannot achieve a smaller uniform error.

Define

$$r(x) := p(x) - q(x) \in P_n.$$

The polynomial $r(x)$ alternates in sign at the equioscillation points, and its sign at each point is determined by the sign of the error $(f-p)(x)$. That is,

$$\operatorname{sign}(r(x_j)) = \operatorname{sign}(f-p)(x_j) = (-1)^j \|f-p\|_\infty, \qquad j = 0, 1, \ldots, n+1.$$

Since $r(x_j)$ alternates in sign at consecutive points x_j, x_{j+1}, by the Intermediate Value Theorem there must be a zero of r in each interval (x_j, x_{j+1}). That gives at least $n+1$ distinct zeros of r. But $r \in P_n$, so it can have at most n distinct zeros unless it is identically zero. Therefore, $r \equiv 0$, which means $p \equiv q$.

Optimality ⇒ Equioscillation. Suppose $f-p$ equioscillates at fewer than $n+2$ points, and set $E = \|f-p\|_\infty$. Without loss of generality, assume the leftmost extremum is one where $f-p$ takes the value $-E$. Then there are numbers

$$-1 < x_1 < \cdots < x_k < 1, \qquad k \le n,$$

such that

$$(f-p)(x) < E \quad \text{for } x \in [-1, x_1] \cup [x_2, x_3] \cup [x_4, x_5] \cup \cdots$$

and

$$(f-p)(x) > -E \quad \text{for } x \in [x_1, x_2] \cup [x_3, x_4] \cup \cdots .$$

If we define

$$\delta p(x) := (x_1 - x)(x_2 - x) \cdots (x_k - x),$$

then $(p - \varepsilon\delta p)(x)$ will be a better approximation to f than p for all sufficiently small $\varepsilon > 0$. This contradicts the assumption that p is optimal. Hence $f - p$ must equioscillate at least $n + 2$ times.

Uniqueness. Let both p^*, q^* be best approximation polynomials. Then

$$\left\| f - \tfrac{p^*+q^*}{2} \right\|_\infty = \left\| \tfrac{f-p^*}{2} + \tfrac{f-q^*}{2} \right\|_\infty \leq \tfrac{1}{2}\|f - p^*\|_\infty + \tfrac{1}{2}\|f - q^*\|_\infty = \|f - p^*\|_\infty,$$

since both are best approximations.
That is, $\frac{1}{2}(p^* + q^*)$ is also a best approximation polynomial.
Furthermore, there are $n + 2$ alternating points at which

$$\tfrac{1}{2}(f - p^*) + \tfrac{1}{2}(f - q^*) = \|f - p^*\|_\infty.$$

At each of these points, $f - p^*$ and $f - q^*$ are both $\|f - p^*\|_\infty$ or both $-\|f - p^*\|_\infty$.
So $f - p^*$ and $f - q^*$ agree at $n + 2$ points, hence

$$(f - p^*) - (f - q^*) = q^* - p^* = 0 \quad \text{at these } n + 2 \text{ points.}$$

Since $q^* - p^* \in P_n$, we must have $q^* - p^* = 0$. □

Remark 4.2. The Remez algorithm is an efficient iterative method for computing the *best polynomial approximation in the uniform norm*. It was introduced by the Russian mathematician Evgeny Remez, who published the result in 1934 [17]. The algorithm iteratively refines a polynomial approximation to minimize the maximum deviation from the target function, thereby yielding the minimax (or best uniform) approximation.

4.1 Near-best approximation and Lebesgue constants.

The computation of the best polynomial approximation p^*, obtained for instance by the Remez algorithm, is considerably more difficult than that of the near-best polynomial approximation p. This is because the mapping from f to p^* is *nonlinear*, requiring iteration in numerical implementations, whereas the mapping from f to p is *linear*. In practice, it is perfectly feasible to compute p for polynomial degrees in the millions, whereas for p^* one would rarely attempt degrees higher than a few hundreds.
Nevertheless, in most applications one is not concerned with computing the exact best approximation, but rather with understanding how far a given polynomial interpolation is from the best possible approximation in the uniform norm. In other words, we seek a quantitative measure that allows us to compare the polynomial interpolant of a function with its best uniform approximant. Such a measure provides insight into how close the computed interpolant is to the optimal approximation that minimizes $\|f - p\|_\infty$.

To this end, It is useful to look at Lagrange interpolation in terms of a (linear) operator I_n from (say) the space of continuous functions to the space of polynomials P_n,

$$I_n : C[-1,1] \to P_n, \quad f \mapsto p_n\,.$$

The interval $[-1,1]$ here is any interval containing all points x_k, $k = 0,1,\ldots,n$. The operator I_n has the following properties:
1. $I_n(\alpha f) = \alpha I_n f, \quad \alpha \in \mathbb{R}$ (homogeneity),
2. $I_n(f+g) = I_n f + I_n g$ (additivity).
Combining 1 and 2 shows that I_n is a linear operator,

$$I_n(\alpha f + \beta g) = \alpha I_n f + \beta I_n g, \quad \alpha, \beta \in \mathbb{R}.$$

3. $I_n f = f \quad$ for all $f \in P_n$.
The last property an immediate consequence of uniqueness of the interpolation polynomial says that I_n leaves polynomials of degree $\leq n$ unchanged, and hence is a *projection operator*.
The norm of the interpolation operator I_n with respect to the uniform norm is defined as

$$\|I_n\| = \sup_{\substack{f \in C([-1,1]) \\ \|f\| \leq 1}} \frac{\|I_n f\|}{\|f\|}. \tag{4.1}$$

Let $\{x_k\}_{k=0}^n$ be $n+1$ distinct interpolation nodes in $[-1,1]$. For a function $f \in C([-1,1])$, the Lagrange interpolating polynomial takes the form

$$p_n(x) = \sum_{k=0}^{n} f(x_k)\,\ell_k(x)\,.$$

The *Lebesgue function* associated with the interpolation process is defined as

$$\lambda_n(x) = \sum_{k=0}^{n} |\ell_k(x)|,$$

where $\ell_j(x)$ are the Lagrange basis polynomials.

For this function, we always have $\lambda_n(x) \geq 1$, and in particular,

$$\lambda_n(x_k) = 1.$$

Indeed, since the Lagrange basis functions satisfy

$$\ell_k(x_j) = \delta_{jk},$$

we have, at each interpolation node $x = x_k$,

$$\lambda_n(x_k) = \sum_{j=0}^{n} |\ell_j(x_k)| = \sum_{j=0}^{n} |\delta_{jk}| = 1.$$

To see that $\lambda_n(x) \geq 1$ for all x, note that

$$1 = |\ell_0(x) + \ell_1(x) + \cdots + \ell_n(x)| \leq |\ell_0(x)| + |\ell_1(x)| + \cdots + |\ell_n(x)| = \lambda_n(x).$$

Hence, the Lebesgue function is always at least one, and it equals one exactly at the interpolation nodes.

Theorem 4.3. *The Lebesgue constant associated with the interpolation operator I_n is given by*

$$\Lambda_n = \|I_n\|_\infty = \sup_{x\in[-1,1]} \lambda_n(x). \tag{4.2}$$

Proof.

$$\begin{aligned}
\|I_n f\|_\infty &= \left\| \sum_{k=0}^{n} f(x_k) L_k \right\|_\infty = \sup_{x\in[-1,1]} \left| \sum_{k=0}^{n} f(x_k)\ell_k(x) \right| \\
&\leq \sup_{x\in[-1,1]} \sum_{k=0}^{n} |f(x_k)||\ell_k(x)| \\
&\leq \left(\sup_{k=0,1,\ldots,n} |f(x_k)| \right) \sup_{x\in[-1,1]} \sum_{k=0}^{n} |\ell_k(x)| \\
&\leq \|f\|_\infty \sup_{x\in[-1,1]} \lambda_n(x) \\
&= \|f\|_\infty \Lambda_n
\end{aligned}$$

$$\text{Hence} \quad \frac{\|I_n f\|_\infty}{\|f\|_\infty} \leq \Lambda_n, \quad \text{so} \quad \sup_{f\in C[-1,1]} \frac{\|I_n f\|_\infty}{\|f\|_\infty} \leq \Lambda_n.$$

If $\|f\|_\infty \leq 1$, then

$$\|I_n\|_\infty \leq \Lambda_n. \tag{4.3}$$

To prove the reverse inequality, select $x \in [-1, 1]$ such that $\lambda_n(x) = \Lambda_n$. Then find a function $f \in C([-1, 1])$ such that $\|f\|_\infty = 1$ and $f(x_k) = \operatorname{sgn}(\ell_k(x))$. For this function, we have

$$\|I_n\|_\infty \geq \|I_n f\|_\infty \geq (I_n f)(x) = \sum_{k=0}^{n} f(x_k)\ell_k(x) = \sum_{k=0}^{n} \operatorname{sgn}(\ell_k(x))\ell_k(x) = \sum_{k=0}^{n} |\ell_k(x)| = \lambda_n(x) = \Lambda_n.$$

In order to construct such a function f, one can invoke the *Tietze Extension Theorem*. This theorem states that if Y is a closed subset of a normal topological space X, then every continuous function from Y to $[-1, 1]$ admits a continuous extension from X to $[-1, 1]$. (See Kelley, *General Topology* [16, page 242].) Another equivalent formulation of (4.2) is

$$\Lambda_n = \sup \{\|I_n f\|_\infty : \|f\|_\infty = 1\}$$

□

The Lebesgue constant depends solely on the choice of interpolation points. In fact, the Lebesgue constant provides a bound on the interpolation error in terms of the minimal achievable error among all polynomials in P_n.

Theorem 4.4 (Near-best approximation and Lebesgue constants)**.** *Let Λ_n be the Lebesgue constant for a linear projection $I_n : C([-1,1]) \to P_n$. Let $f \in C([-1,1])$, and let $p_n = I_n f$ be the corresponding polynomial approximant to f, while p_n^* denotes the best uniform approximant. Then*

$$\|f - p_n\|_\infty \leq (\Lambda_n + 1)\|f - p_n^*\|_\infty. \tag{4.4}$$

Proof. Let $p_n^* \in P_n$ denote the best approximation to f in the uniform norm, It is easy to show how to obtain this inequality. From the uniqueness of the interpolating polynomial we have

$$p_n(x) = \sum_{k=0}^{n} f(x_k)\ell_k(x) \quad \text{and} \quad p_n^*(x) = \sum_{k=0}^{n} p_n^*(x_k)\ell_k(x).$$

By subtracting $p_n(x)$ from $p_n^*(x)$ we get

$$\begin{aligned} |p_n^*(x) - p_n(x)| &= \left| \sum_{k=0}^{n} p_n^*(x_k)\ell_k(x) - \sum_{k=0}^{n} f(x_k)\ell_k(x) \right| \\ &= \left| \sum_{k=0}^{n} (p_n^*(x_k) - f(x_k))\,\ell_k(x) \right| \\ &\leq \sum_{k=0}^{n} |\ell_k(x)| \sup_{k=0,\ldots,n} |p_n^*(x_k) - f(x_k)|. \end{aligned}$$

From this it follows that (due to $\lambda_n(x) = \sum_{k=0}^{n} |\ell_k(x)|$)

$$\|p_n^* - p_n\|_\infty \leq \Lambda_n \|f - p_n^*\|_\infty.$$

Finally, we have

$$\begin{aligned} \|f - p_n\|_\infty &= \|f - p_n^* + p_n^* - p_n\|_\infty \\ &\leq \|f - p_n^*\|_\infty + \|p_n^* - p_n\|_\infty \\ &\leq \|f - p_n^*\|_\infty + \Lambda_n \|f - p_n^*\|_\infty \\ &= (1 + \Lambda_n)\|f - p_n^*\|_\infty. \end{aligned}$$

□

A small Lebesgue constant indicates that the interpolation error cannot be much larger than the best possible polynomial approximation error. Another reason for studying the Lebesgue constant is that it quantifies the conditioning of the polynomial interpolation problem in the Lagrange basis. Let $\tilde{p}_n(x)$ denote the polynomial interpolant of degree n corresponding to a perturbed function $\tilde{f}$ at the same interpolation nodes. Then

$$\tilde{p}_n(x) = \sum_{k=0}^{n} \tilde{f}(x_k)\,\ell_k(x).$$

Since $\|p_n\|_\infty \geq \sup_{k=0,\dots,n} |f(x_k)|$, it follows that

$$\frac{\|p_n - \tilde{p}_n\|_\infty}{\|p_n\|_\infty} \leq \frac{\sup\limits_{x\in[-1,1]} \sum\limits_{k=0}^{n} |f(x_k) - \tilde{f}(x_k)|\,|\ell_k(x)|}{\sup\limits_{k=0,\dots,n} |f(x_k)|} \leq \Lambda_n \frac{\sup\limits_{k=0,\dots,n} |f(x_k) - \tilde{f}(x_k)|}{\sup\limits_{k=0,\dots,n} |f(x_k)|}. \quad (7)$$

This observation implies that if the interpolation nodes are chosen such that the Lebesgue constant Λ_n is small, then the corresponding Lagrange interpolant will be less sensitive to perturbations in the function values. Consequently, numerical interpolation in floating-point arithmetic becomes unreliable even for smooth functions f—whenever the Lebesgue constant Λ_n exceeds the reciprocal of the machine precision, which is typically on the order of 10^{16}.
The next result gives a summary of several known results for particular sets of interpolation points for which the behavior of the Lebesgue function has been extensively studied.

Theorem 4.5 (Lebesgue constants for polynomial interpolation)**.** *The Lebesgue constants Λ_n for degree $n \geq 0$ polynomial interpolation in any set of $n+1$ distinct points in $[-1,1]$ satisfy*

$$\Lambda_n \geq \frac{2}{\pi}\log(n+1) + 0.52125\dots$$

the number $0.52125\dots$ is $\frac{2}{\pi}(\gamma + \log(4/\pi))$, where

$$\gamma = \lim_{n\to\infty}\left(\sum_{i=1}^{n}\frac{1}{i} - \log n\right) = 0.577\dots$$

is Euler's constant.
For Chebyshev points, they satisfy

$$\Lambda_n \leq \frac{2}{\pi}\log(n+1) + 1, \qquad \Lambda_n \sim \frac{2}{\pi}\log n, \quad n\to\infty.$$

For the set of equidistant points

$$E = \left\{x_k = -1 + \frac{2k}{n},\ k = 0, 1, \dots, n\right\},$$

they satisfy

$$\Lambda_n(E) > \frac{2^{n-2}}{n^2}, \qquad \Lambda_n(E) \sim \frac{2^{n+1}}{en(\log n + \gamma)}, \qquad n\to\infty,$$

with the first inequality applying for $n \geq 1$.

Theorem 4.6 (Lebesgue constants for Chebyshev projection)**.** *The Lebesgue constants Λ_n for degree $n \geq 1$ Chebyshev projection in $[-1,1]$ are given by*

$$\Lambda_n = \frac{1}{2\pi}\int_{-\pi}^{\pi}\left|\frac{\sin((n+\frac{1}{2})t)}{\sin(t/2)}\right| dt.$$

They satisfy

$$\Lambda_n \leq \frac{4}{\pi^2}\log(n+1) + 3, \qquad \Lambda_n \sim \frac{4}{\pi^2}\log n, \quad n\to\infty.$$

Having introduced the Lebesgue constants, we can now interpret what theoretical results reveal about the relationship between the interpolating polynomial p and the best approximation p^*. Theorems 2.3 and 2.9, discussed earlier, established convergence rates of p_n to f depending on the smoothness of f. Analogous results hold for p_n^*, differing only by constant factors. In particular, the same asymptotic convergence rates apply to both, demonstrating that the interpolating polynomial p achieves, up to a constant multiple governed by the Lebesgue constant, the same order of accuracy as the best uniform approximation p^*.

Theorem 4.7 (Chebyshev projections and interpolants are near-best)**.** *Let $f \in C([-1,1])$ have degree n Chebyshev projection f_n, Chebyshev interpolant p_n, and best approximation p_n^*, with $n \geq 1$. Then*

$$\|f - f_n\| \leq \left(4 + \frac{4}{\pi^2}\log(n+1)\right)\|f - p_n^*\|$$

and

$$\|f - p_n\| \leq \left(2 + \frac{2}{\pi}\log(n+1)\right)\|f - p_n^*\|.$$

Proof. Follows from Theorems 4.4, 4.5, and 4.6. □

Example 4.8. For instance, when $n = 100$, the interpolation error using Chebyshev points satisfies

$$\|f - p_n\|_\infty \leq (4.9 \times 10^0)\,\|f - p_n^*\|_\infty,$$

meaning that, even in the worst case, the error $\|f - p_n\|_\infty$ is at most 4.9 times larger than the smallest possible error $\|f - p_n^*\|_\infty$.
By contrast, if one uses equidistant interpolation points of the same degree, the corresponding upper bound becomes

$$\|f - p_n\|_\infty \leq (1.8 \times 10^{27})\,\|f - p_n^*\|_\infty.$$

5 Quadrature

The topic we are entering now is one of the core subjects in numerical analysis. It is also one of the oldest: it goes back to Newton, and even today it remains fundamental.
Let us assume that $f \in C([-1,1])$. Our goal is to compute the integral

$$I = \int_{-1}^{1} f(x)\,dx. \tag{5.1}$$

To approximate this integral numerically, we will sample the function at a set of points. We choose a set of distinct nodes

$$x_0, x_1, \ldots, x_n \in [-1,1],$$

and we also assume that we have weights

$$w_0, w_1, \ldots, w_n,$$

whose construction we will discuss shortly.
The classical quadrature idea is to approximate the integral by

$$I \approx I_n = \sum_{k=0}^{n} f(x_k)\, w_k. \tag{5.2}$$

In most cases (about 95% of the time), the weights come from polynomial interpolation. The idea is simple: we interpolate the function values by a polynomial and integrate that polynomial. We do not build the polynomial explicitly; instead, we compute the numbers w_k that represent the integral of each Lagrange basis polynomial.
We have

$$w_k = \int_{-1}^{1} \ell_k(x)\,dx, \tag{5.3}$$

where $\ell_k(x)$ is the k-th Lagrange basis polynomial: the degree-n polynomial that is 1 at x_k and 0 at all other nodes.
This gives the so-called interpolatory quadrature:

$$I_n = \int_{-1}^{1} p(x)\,dx, \tag{5.4}$$

where $p \in P_n$ satisfies $p(x_k) = f(x_k)$.
There are three important special cases of interpolatory quadrature rules:

- **Newton–Cotes.** This is the rule obtained when we choose equally spaced points $\{x_k\}$. It is the most natural thing to try first, and for low-order rules it works fine. For example, when $n = 2$ we obtain Simpson's rule. However, as $n \to \infty$ the method becomes very poor: it diverges exponentially. From interpolation theory we already know that using equispaced nodes leads to the Runge phenomenon: the interpolating polynomial oscillates wildly, and so the quadrature can produce completely unreliable answers. In fact, Newton–Cotes rules are nonconvergent even for analytic functions f. Moreover, the weights satisfy $|w_k| \sim 2^n$ and their signs alternate, which is already a hint that the rule is likely to misbehave.
- **Clenshaw–Curtis.** This rule is obtained by choosing the nodes $\{x_k\}$ to be the Chebyshev points (i.e., the extrema of the Chebyshev polynomials).
- **Gauss quadrature.** This consists of choosing the nodes $\{x_k\}$ to be the Legendre points, meaning the roots of the Legendre polynomial of degree $n + 1$.

The first rule is terrible for large n, while the last two are both good and essentially equally good in practice. For Clenshaw–Curtis and Gauss, the weights are all positive, of order $\mathcal{O}(n^{-1})$, and the rules converge for every continuous function. If the function is smooth, the convergence is very fast.
The first theorem to record is the classical result that appears in all books: it describes the order of exactness of these quadrature formulas.

Theorem 5.1 (Polynomial degree of quadrature formulas)**.** *For any $n \geq 0$, an $(n + 1)$-point interpolatory quadrature formula—such as Clenshaw–Curtis, Gauss, or Newton–Cotes—is exact for all polynomials $f \in \mathcal{P}_n$. Moreover, the $(n+1)$-point Gauss quadrature formula is exact for all $f \in \mathcal{P}_{2n+1}$.*

In simple words: For any interpolatory quadrature rule, we always have

$$I(f) = I_n(f) \quad \text{whenever } f \in \mathcal{P}_n.$$

But Gauss quadrature gives something extra:

$$I(f) = I_n(f) \quad \text{for all } f \in \mathcal{P}_{2n+1}.$$

That is, Gauss quadrature achieves a *double* degree of exactness.

Proof. The first part is immediate from the construction of interpolatory quadrature rules.
For the second part, we follow Jacobi's argument using orthogonal polynomials, rather than Gauss's original approach with continued fractions.
Let f be a polynomial of degree at most $2n + 1$. Then we can write

$$f(x) = P_{n+1}(x)\, q_n(x) + r_{nn}(x),$$

where $q_n, r_{nn} \in \mathcal{P}_n$. The idea is simple: we extract from f the component that contains the powers from degree $n+1$ up to $2n+1$, and collect these terms in the product $P_{n+1}\, q_n$. The remainder r_{nn} is then a polynomial of degree at most n.

We now compare the exact integral I and the Gauss quadrature approximation I_n.

$$I = \int_{-1}^{1} f(x)\,dx = \int_{-1}^{1} P_{n+1}(x)\,q_n(x)\,dx + \int_{-1}^{1} r_{nn}(x)\,dx. \tag{5.5}$$

The first integral is zero, because P_{n+1} is orthogonal to every polynomial of degree at most n, and $q_n \in \mathcal{P}_n$. Thus

$$I = \int_{-1}^{1} r_{nn}(x)\,dx.$$

Next we compute the Gauss quadrature approximation:

$$I_n = \sum_{k=0}^{n} w_k\,f(x_k) = \sum_{k=0}^{n} w_k\,P_{n+1}(x_k)\,q_n(x_k) + \sum_{k=0}^{n} w_k\,r_{nn}(x_k).$$

The first sum is zero because the nodes x_k are precisely the roots of the Legendre polynomial P_{n+1}; hence $P_{n+1}(x_k) = 0$ for all k.
Therefore

$$I_n = \sum_{k=0}^{n} w_k\,r_{nn}(x_k).$$

Finally, since Gauss quadrature is an interpolatory rule exact for every polynomial of degree at most n, and since $r_{nn} \in \mathcal{P}_n$, we have

$$\int_{-1}^{1} r_{nn}(x)\,dx = \sum_{k=0}^{n} w_k\,r_{nn}(x_k).$$

Combining the above equalities gives

$$I = I_n,$$

which completes the proof. □

A few comments on how these quadrature rules are implemented. We begin with the Clenshaw–Curtis (CC) quadrature. The key identity used in the method is

$$\int_{-1}^{1} T_k(x)\,dx = \begin{cases} 0, & k \text{ odd}, \\ \dfrac{2}{1-k^2}, & k \text{ even}. \end{cases}$$

This allows us to integrate a Chebyshev interpolant term by term. Given the function samples $\{f(x_k)\}$ taken at Chebyshev points, we convert them into the Chebyshev coefficients $\{c_k\}$ of the interpolating polynomial. This conversion is performed using an FFT, which requires $\mathcal{O}(n \log n)$ operations.
With the coefficients available, the Clenshaw–Curtis quadrature formula becomes

$$I_n = \sum_{\substack{k=0 \\ k \text{ even}}}^{n} \frac{2\,c_k}{1-k^2}. \tag{5.6}$$

The overall procedure is summarized as

$$\boxed{\{f(x_k)\} \xrightarrow{\text{FFT}} \{c_k\} \quad \Longrightarrow \quad I_n = \sum_{\substack{k=0 \\ k \text{ even}}}^{n} \frac{2\,c_k}{1-k^2}}$$

Theorem 5.2 (Integral of a Chebyshev series)**.** *The integral of a degree n polynomial expressed as a Chebyshev series is*

$$\int_{-1}^{1} \sum_{k=0}^{n} c_k\, T_k(x)\, dx = \sum_{\substack{k=0 \\ k \text{ even}}}^{n} \frac{2\,c_k}{1-k^2}.$$

Proof. We make use of the known integral identity for Chebyshev polynomials:

$$\int_{-1}^{1} T_k(x)\, dx = \begin{cases} 0, & k \text{ odd}, \\ \dfrac{2}{1-k^2}, & k \text{ even}. \end{cases}$$

Now consider the Chebyshev series

$$f(x) = \sum_{k=0}^{n} c_k\, T_k(x).$$

Integrating term-by-term on $[-1, 1]$ (justified since the sum is finite), we obtain

$$\int_{-1}^{1} f(x)\, dx = \sum_{k=0}^{n} c_k \int_{-1}^{1} T_k(x)\, dx.$$

Applying the identity above: - Every term with k odd integrates to 0. - Every term with k even contributes $\dfrac{2}{1-k^2}$.
Therefore,

$$\int_{-1}^{1} f(x)\, dx = \sum_{\substack{k=0 \\ k \text{ even}}}^{n} c_k \cdot \frac{2}{1-k^2} = \sum_{\substack{k=0 \\ k \text{ even}}}^{n} \frac{2\,c_k}{1-k^2},$$

which completes the proof. □

Another approach is to compute the Clenshaw–Curtis nodes and weights explicitly in $\mathcal{O}(n \log n)$ operations. The implementation is straightforward, and the entire method is essentially $n \log n$ mathematics.

Let us now compare this with the more complicated story of Gauss quadrature. When digital computers first appeared, nobody worked with Clenshaw–Curtis quadrature, because the method had not yet been proposed. Gauss quadrature, on the other hand, was already one of the most established ideas in numerical analysis, so early numerical analysts focused on computing Gauss nodes and weights.

The problem reduces to finding the roots of the Legendre polynomial $P_{n+1}(x)$. Accurately computing these roots is delicate, and inaccurate computation leads to very unstable quadrature rules.

There is a long history of efforts to compute these roots, leading to a major breakthrough in 1969. In that year, Gene H. Golub and John H. Welsch [7] showed that Gauss quadrature nodes and weights can be obtained by solving an eigenvalue problem. Specifically, the Gauss nodes (the roots $\{x_k\}$ of P_{n+1}) are the eigenvalues of a certain symmetric tridiagonal (Jacobi) matrix. The recurrence coefficients for the Legendre polynomials lead to the matrix

$$\begin{pmatrix} 0 & 1 & & & & \\ \frac{1}{3} & 0 & \frac{2}{3} & & & \\ & \frac{2}{5} & 0 & \frac{3}{5} & & \\ & & \frac{3}{7} & 0 & \frac{4}{7} & \\ & & & \ddots & \ddots & \ddots \end{pmatrix}.$$

This matrix is not symmetric, so we symmetrize it by a diagonal similarity transformation. This corresponds to multiplying the first column by a constant and dividing the first row by the same constant. Using $\sqrt{3}$ for the first entry, we obtain the symmetric version

$$\begin{pmatrix} 0 & \frac{1}{\sqrt{1\cdot 3}} & 0 & 0 & \cdots \\ \frac{1}{\sqrt{1\cdot 3}} & 0 & \frac{2}{\sqrt{3\cdot 5}} & 0 & \cdots \\ 0 & \frac{2}{\sqrt{3\cdot 5}} & 0 & \frac{3}{\sqrt{5\cdot 7}} & \cdots \\ 0 & 0 & \frac{3}{\sqrt{5\cdot 7}} & 0 & \ddots \\ \vdots & \vdots & \vdots & \ddots & \ddots \end{pmatrix}.$$

The nodes of the Gauss quadrature rule are precisely the eigenvalues of this matrix. The weights also follow from the eigenvalue problem: if v_k is the kth normalized eigenvector of the Jacobi matrix (with $\|v_k\| = 1$), then

$$w_k = 2\,(v_k)_1,$$

i.e., the weight w_k is twice the first component of v_k. This is a remarkably elegant result.

From the point of view of computational complexity, however, this leads to an expensive algorithm: solving a tridiagonal eigenvalue problem typically costs $\mathcal{O}(n^2)$ operations. If one calls `eig` in MATLAB, the cost becomes $\mathcal{O}(n^3)$, since `eig` does not exploit tridiagonal structure. Quadrature should not be this expensive. Clenshaw–Curtis quadrature, in contrast, costs only $\mathcal{O}(n \log n)$ operations via the FFT.

The situation changed dramatically about eighteen years ago. A 2007 paper of Glaser–Liu–Rokhlin [6] introduced an algorithm that computes Gauss nodes and weights in $\mathcal{O}(n)$ operations, using clever ideas from differential equations (in contrast to the linear-algebra approach of Golub–Welsch).

Later, in 2013, Nick Hale and Alex Townsend [10] observed that for $n > 100$, one can simply use classical asymptotic formulas—known for more than a century—to compute the Legendre roots to machine precision.

5.1 Accuracy of Clenshaw–Curtis and Gauss quadrature

We now discuss the accuracy of the quadrature rules. The quantity of interest is the error

$$E_n(f) = I_n(f) - I. \tag{5.7}$$

If f is Lipschitz on $[-1, 1]$, then it admits a Chebyshev expansion

$$f(x) = \sum_{k=0}^{\infty} a_k \, T_k(x). \tag{5.8}$$

Using this expansion, the quadrature error becomes

$$E_n(f) = \sum_{k=0}^{\infty} a_k \, E_n(T_k). \tag{5.9}$$

Since $E_n(T_k) = 0$ whenever k is odd, this reduces to

$$E_n(f) = \sum_{\substack{k=0 \\ k \text{ even}}}^{\infty} a_k \, E_n(T_k). \tag{5.10}$$

Recall that every interpolatory quadrature rule is exact for all polynomials of degree at most n. This leads to the following descriptions of the error for the two main rules.
For the Clenshaw–Curtis rule,

$$E_n(f) = \sum_{\substack{k=n+1 \\ k \text{ even}}}^{\infty} a_k \, E_n(T_k), \tag{5.11}$$

since all Chebyshev modes up to n are integrated exactly.
For Gauss quadrature, the exactness doubles:

$$E_n(f) = \sum_{\substack{k=2n+2 \\ k \text{ even}}}^{\infty} a_k \, E_n(T_k), \tag{5.12}$$

reflecting the exactness for all polynomials of degree up to $2n + 1$.
To bound the individual mode errors $E_n(T_k)$, observe that

$$|T_k(x)| \le 1 \quad \text{for } x \in [-1, 1],$$

which implies

$$\left| \int_{-1}^{1} T_k(x) \, dx \right| \le \int_{-1}^{1} |T_k(x)| \, dx \le 2. \tag{5.13}$$

Similarly,

$$\left|\sum_{j=0}^{n} w_j\, T_k(x_j)\right| \le \sum_{j=0}^{n} w_j\, |T_k(x_j)| \le \sum_{j=0}^{n} w_j = 2.$$

Therefore,

$$|E_n(T_k)| = \left|\int_{-1}^{1} T_k(x)\, dx - \sum_{j=0}^{n} w_j\, T_k(x_j)\right| \le 4. \tag{5.14}$$

We now have estimates for how large the quadrature errors can be. Let us begin with the case of analytic functions. Recall the classical fact:
If f is analytic in the Bernstein ellipse E_ρ and satisfies $|f(x)| \le M$ on E_ρ, then the Chebyshev coefficients obey (Theorem (2.8))

$$|a_k| \le 2M\, \rho^{-k}. \tag{5.15}$$

For the Clenshaw–Curtis formula, the error is

$$|E_n(f)| \;\le\; \sum_{\substack{k\ge n+1\\ k \text{ even}}} |a_k|\, |E_n(T_k)| \;\le\; 4 \sum_{\substack{k\ge n+1\\ k \text{ even}}} 2M\, \rho^{-k} = 8M \sum_{m\ge \lceil (n+1)/2\rceil} \rho^{-2m}.$$

This is a geometric series with ratio ρ^{-2}. Therefore,

$$\boxed{\text{CC:} \qquad |E_n(f)| \;\le\; \frac{8M\, \rho^{-\,n-1}}{1-\rho^{-2}}.} \tag{5.16}$$

The factor ρ^{-2} (instead of ρ^{-1}) appears because every *other* Chebyshev coefficient enters the error: only even k contribute.

For Gauss quadrature,

$$|E_n(f)| \;\le\; 4 \sum_{\substack{k\ge 2n+2\\ k \text{ even}}} |a_k| \;\le\; 4 \sum_{\substack{k\ge 2n+2\\ k \text{ even}}} 2M\, \rho^{-k} = 8M \sum_{m=n+1}^{\infty} \rho^{-2m}.$$

Hence

$$|E_n(f)| = 8M \frac{\rho^{-2(n+1)}}{1-\rho^{-2}} = \frac{8M\, \rho^{-\,2n-2}}{1-\rho^{-2}},$$

and we may write the final bound as

$$\boxed{\text{G:} \qquad |E_n(f)| \;\le\; \frac{8M\, \rho^{-\,2n-2}}{1-\rho^{-2}}.} \tag{5.17}$$

Thus Gauss quadrature converges at *roughly twice the exponential rate* of Clenshaw–Curtis when f is analytic. (The constant can in fact be improved from 64/15, but the main point is the difference in decay rates.)

Let us also record an estimate applicable to functions of finite smoothness. If $V = \mathrm{Var}(f^{(\nu)}) < \infty$ for some integer $\nu \geq 1$, then the Chebyshev coefficients satisfy the algebraic decay estimate

$$|a_k| \;\leq\; \frac{2V}{\pi\,(k-\nu)^{\nu+1}}. \tag{5.18}$$

This will later allow us to compare the algebraic convergence rates of Clenshaw–Curtis and Gauss quadrature for non-analytic functions.
From the estimate derived above for the Chebyshev coefficients (Theorem (2.2)),

$$|a_k| \leq \frac{2V}{\pi(k-\nu)^{\nu+1}}, \qquad V = \mathrm{Var}(f^{(\nu)}) < \infty,$$

we now derive algebraic error bounds for both quadrature rules.

For Clenshaw–Curtis (CC) quadrature we found

$$|E_n(f)| \;\leq\; \frac{8V}{\pi} \sum_{\substack{k \geq n+1 \\ k \text{ even}}} \frac{1}{(k-\nu)^{\nu+1}}.$$

Set $k = 2m$. Then $m \geq \left\lceil \frac{n+1}{2} \right\rceil$, and therefore

$$|E_n(f)| \;\leq\; \frac{8V}{\pi} \sum_{m \geq \lceil (n+1)/2 \rceil} \frac{1}{(2m-\nu)^{\nu+1}} \;\leq\; \frac{8V}{\pi} \int_{\frac{n+1}{2}}^{\infty} (2s-\nu)^{-(\nu+1)}\, ds.$$

Evaluating the integral yields

$$\int_{\frac{n+1}{2}}^{\infty} (2s-\nu)^{-(\nu+1)}\, ds = \frac{1}{2\nu(n-\nu)^{\nu}}.$$

Thus,

$$|E_n(f)| \;\leq\; \frac{8V}{\pi} \cdot \frac{1}{2\nu(n-\nu)^{\nu}} = \boxed{\frac{4V}{\pi\nu(n-\nu)^{\nu}}}.$$

For Gauss quadrature we have

$$E_n(f) = \sum_{\substack{k=2n+2 \\ k \text{ even}}}^{\infty} a_k\, E_n(T_k), \qquad |E_n(T_k)| \leq 2.$$

Using the coefficient bound again gives

$$|E_n(f)| \;\leq\; 2 \sum_{\substack{k=2n+2 \\ k \text{ even}}} |a_k| \;\leq\; \frac{4V}{\pi} \sum_{\substack{k=2n+2 \\ k \text{ even}}}^{\infty} \frac{1}{(k-\nu)^{\nu+1}}.$$

Setting $k = 2m$ now gives $m \geq n+1$, so

$$|E_n(f)| \;\leq\; \frac{4V}{\pi} \sum_{m=n+1}^{\infty} \frac{1}{(2m-\nu)^{\nu+1}} \;\leq\; \frac{4V}{\pi} \int_{n+1}^{\infty} (2s-\nu)^{-(\nu+1)}\, ds.$$

The integral evaluates to

$$\int_{n+1}^{\infty} (2s-\nu)^{-(\nu+1)}\, ds = \frac{1}{2\nu(\,2n+2-\nu\,)^{\nu}}.$$

Therefore,

$$|E_n(f)| \;\le\; \frac{4V}{\pi}\cdot\frac{1}{2\nu(2n+2-\nu)^{\nu}} = \boxed{\frac{4V}{\pi\nu\,(2n+2-\nu)^{\nu}}}.$$

This shows that Gauss quadrature converges algebraically at a rate faster than Clenshaw–Curtis: the effective degree scale is $2n$ rather than n, just as in the analytic case.

Experimental Comparison. Fig. (5.1) displays the convergence as $n \to \infty$ of the Gauss and Clenshaw–Curtis quadrature rules for six representative test functions $f(x)$:

Six Representative Test Functions

$$f_1(x) = x^{20}, \qquad f_4(x) = \begin{cases} 0, & x = 0, \\ e^{-1/x^2}, & x \neq 0, \end{cases}$$

$$f_2(x) = e^{x}, \qquad f_5(x) = \frac{1}{1+16x^2},$$

$$f_3(x) = e^{-x^2}, \quad f_6(x) = |x|^3.$$

These numerical experiments are adapted from [20]. From Fig 5.1 we see that Gauss quadrature is usually slightly more efficient than Clenshaw–Curtis, though typically by less than a factor of 2. The key observation is that Clenshaw–Curtis converges much faster than the classical theoretical bound E_n suggests; in practice its performance is remarkably close to that of Gauss quadrature.

Listing 5.1: MATLAB code used in to produce Fig 5.1.

```
% Compares GaussLegendre vs. ClenshawCurtis (Chebfun sums) for several test integrands.
%
% Requires Chebfun (uses chebfun, legpts).

close all; clear; clc

x  = chebfun('x',[ -1, 1 ]);

tests = { ...
    {@(x) x.^20,                                    'x^{20}'} ...
  , {@(x) exp(x),                                   'e^{x}'} ...
  , {@(x) exp(-x.^2),                               'e^{-x^{2}}'} ...
  , {@(x) ((x==0).*0 + (x~=0).*exp(-1./(x.^2))),'e^{-1/x^{2}}'} ...
  , {@(x) 1./(1+16*x.^2),                           '\frac{1}{1+16x^{2}}'} ...
  , {@(x) abs(x).^3,                                '|x|^{3}'} ...
```

```
};

Nmax = 30;
ns   = 2:1:Nmax;

for k = 1:numel(tests)
    f       = tests{k}{1};
    fname   = tests{k}{2};

    %    Exact    integral via adaptive Chebfun (ClenshawCurtis under the hood)
    Iexact  = sum( chebfun(f,[-1,1]) );

    err_cc     = zeros(size(ns));
    err_gauss  = zeros(size(ns));

    for j = 1:numel(ns)
        n = ns(j);

        % ClenshawCurtis with n+1 points via fixed-degree chebfun
        Icc         = sum( chebfun(f, n+1) );
        err_cc(j)   = abs(Iexact - Icc);

        % GaussLegendre with n+1 points
        [s,w]       = legpts(n+1);       % nodes & weights on [-1,1]
        Igl         = w * f(s);
        err_gauss(j)= abs(Iexact - Igl);
    end

    figure(k)

    % --- Styled plotting
    % ClenshawCurtis: blue circles
    semilogy(ns, err_cc, 'o', ...
        'Color', [0 0.45 0.74], ...
        'MarkerFaceColor', [0 0.45 0.74], ...
        'LineWidth', 1.5);
    hold on;

    % GaussLegendre: orange squares
    semilogy(ns, err_gauss, 's', ...
        'Color', [0.85 0.33 0.10], ...
        'MarkerFaceColor', [0.85 0.33 0.10], ...
        'LineWidth', 1.5);

    grid on;
    set(gca, 'FontSize', 16, 'LineWidth', 1);

    xlabel('$n$', 'Interpreter', 'latex', 'FontSize', 16);
    ylabel('$|I - I_n|$', 'Interpreter', 'latex', 'FontSize', 16);

    title(['Gauss vs.\ Clenshaw--Curtis for $f(x) = ', fname, '$'], ...
        'Interpreter', 'latex', 'FontSize', 14);

    legend({'Clenshaw--Curtis', 'Gauss--Legendre'}, ...
```

```
        'Interpreter','latex','FontSize',12, 'Location','southwest');
end
```

Why is Clenshaw–Curtis often better than the bounds suggest?
What have we overlooked in our theoretical estimates?
From the Chebyshev expansion, the quadrature error takes the form

$$E_n(f) = \sum_{\substack{k=n+1 \\ k \text{ even}}}^{\infty} a_k \, E_n(T_k).$$

For Gauss quadrature we have complete exactness up through degree $2n+1$:

$$E_n(T_n) = 0, \ \ldots, \ E_n(T_{2n+1}) = 0.$$

For Clenshaw–Curtis these terms are not zero. Our crude bound was

$$|E_n(T_{n+1})|, \ldots, |E_n(T_{2n+1})| \le 4,$$

but this estimate is very far from sharp. In fact,

$$\text{for Clenshaw–Curtis:} \qquad E_n(T_{n+k}) = O(n^{-3}), \qquad k \ll n.$$

The reason: aliasing. Clenshaw–Curtis quadrature does not evaluate T_{n+k} directly. Instead it *aliases* it to a lower-degree Chebyshev polynomial:

$$T_{n+k} \longmapsto T_{n-k}, \qquad (k < n).$$

Since CC integrates T_{n-k} exactly, we find

$$E_n(T_{n+k}) = I_n(T_{n+k}) - I(T_{n+k}) = I(T_{n-k}) - I(T_{n+k}).$$

Using the explicit integrals,

$$I(T_{n-k}) = \frac{2}{1-(n-k)^2}, \qquad I(T_{n+k}) = \frac{2}{1-(n+k)^2},$$

we obtain

$$E_n(T_{n+k}) = \frac{2}{1-(n-k)^2} - \frac{2}{1-(n+k)^2} \sim -\frac{8k}{n^3}.$$

The terms that appear to be "non-zero" for Clenshaw–Curtis quadrature are actually extremely small: they decay like $O(n^{-3})$ because high-frequency modes are aliased to low-frequency ones that the rule integrates exactly. Thus, although Clenshaw–Curtis does not integrate $T_{n+1}, T_{n+2}, \ldots$ correctly (while Gauss does), the effective error is controlled by the tiny difference

$$I(T_{n-k}) - I(T_{n+k}) = O(n^{-3}).$$

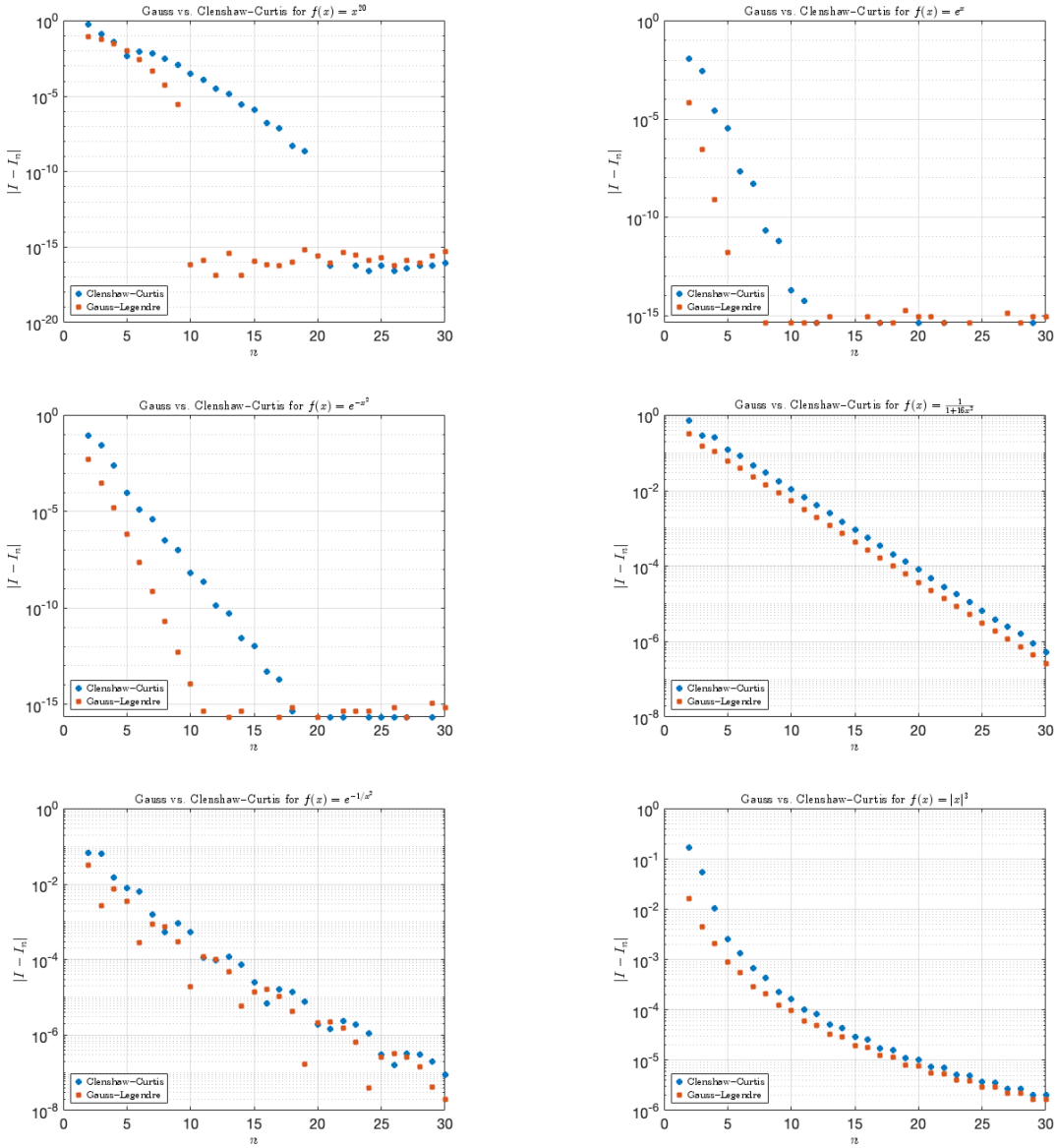


Figure 5.1: Convergence in floating-point arithmetic of Gauss and Clenshaw–Curtis quadrature for six functions f on $[-1, 1]$ for n ranging from 1 to 30. The first function is polynomial, the second and third entire, the fourth analytic, the fifth C^∞, and the sixth C^2.

6 Nonlinear approximation: Why rational functions?

We now switch gears to a non-polynomial topic that is extremely important in practice: *rational approximation*. We begin by fixing notation and stating precisely what we mean by rational functions.
Let $m, n \geq 0$ be integers, and define

$$\mathcal{R}_{mn} = \{\text{rational functions of type } (m, n)\}.$$

This means a function whose numerator has degree at most m and denominator at most n, not necessarily written in lowest terms. Thus a general element of $\mathcal{R}_{mn}$ has the form

$$r(x) = \frac{\sum_{k=0}^{m} a_k x^k}{\sum_{k=0}^{n} b_k x^k}, \qquad \text{(not necessarily reduced)}.$$

Sometimes it is convenient to work in *lowest terms*. In that case we prefer to write the degrees as (μ, ν) rather than (m, n), meaning that r has numerator of exact degree μ and denominator of exact degree ν:

$$\text{Lowest terms:} \qquad r(x) = \frac{\sum_{k=0}^{\mu} a_k x^k}{\sum_{k=0}^{\nu} b_k x^k}, \qquad a_\mu \neq 0, \quad b_\nu = 1, \quad \text{no common factors.}$$

In this case we say that r is of *exact type* (μ, ν).
If r is not the zero function, then:

- the numerator has exactly μ finite zeros (counted with multiplicity),
- the denominator has exactly ν finite poles (counted with multiplicity).

Case 1: $\mu > \nu$. The numerator has higher degree than the denominator. Thus $r(x) \to \infty$ as $x \to \infty$, and r has a *pole at infinity* of order $\mu - \nu$.

Case 2: $\nu > \mu$. The denominator dominates, so $r(x) \to 0$ as $x \to \infty$, implying a *zero at infinity* of order $\nu - \mu$.

There is one function that cannot be written in lowest terms with $a_\mu \neq 0$: the zero function. We treat it separately:

$$\text{Special case: } r \equiv 0, \qquad \mu = -\infty, \quad \nu = 0, \qquad \text{exact type } (-\infty, 0).$$

These conventions allow us to state theorems cleanly without repeatedly introducing special clauses for the zero function.

Compared to polynomials, rational functions behave very differently. Polynomials form a linear space; rational functions do not:

$$\textbf{Nonlinearity:} \qquad \mathcal{R}_{mn} \text{ is not a vector space.}$$

In particular:
- it is not closed under addition; - adding two rational functions may increase the number of poles; - two functions in $\mathcal{R}_{mn}$ remain in the same space only if they share the same poles.
This may suggest that rational approximation is vastly more complicated than polynomial approximation. Surprisingly, that is not the case, thanks to the tool of *partial fractions*. Consider the simplest example:

$$r(x) = \sum_{k=1}^{n} \frac{c_k}{x - \xi_k}, \qquad \xi_k \text{ distinct}, \quad c_k \neq 0.$$

This represents a function with n simple poles. In general, poles may also be multiple (double, triple, etc.), which leads to the following basic theorem:

Theorem 6.1. *[22, Theorem 23.1]Tre. Partial fraction representation. Given $m, n \geq 0$, let $r \in \mathcal{R}_{mn}$ be arbitrary. Then r has a unique representation in the form*

$$r(x) = p_0(x) + \sum_{k=1}^{\mu} p_k\Big((x - \xi_k)^{-1}\Big), \tag{6.1}$$

where p_0 is a polynomial of exact degree ν_0 for some $\nu_0 \leq m$ (unless $p_0 = 0$), and $\{p_k\}$, $1 \leq k \leq \mu$, are polynomials of exact degrees $\nu_k \geq 1$ with $p_k(0) = 0$ and $\sum_{k=1}^{\mu} \nu_k \leq n$.

This concludes the preliminary material on the structure of rational functions. The remainder of this chapter addresses the central question: *why rational functions are effective tools in approximation theory and in numerical practice.* A natural point of comparison is polynomial approximation. In many situations polynomial and rational approximants exhibit similar performance, while in others rational approximants provide substantially superior accuracy.

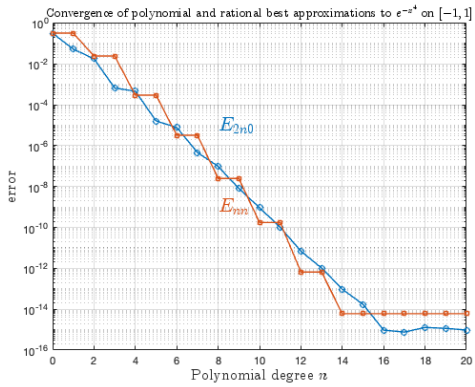
Experiment: polynomial vs. rational best approximation This experiment compares the performance of polynomial and rational best approximants on the interval $[-1, 1]$ for two representative functions:

$$g(x) = e^{-x^4}, \qquad f(x) = |x|.$$

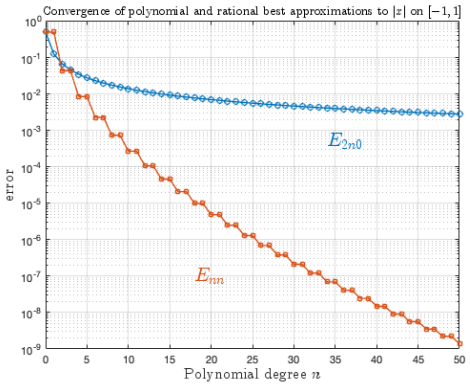
The first is analytic and smooth on $[-1, 1]$; the second is continuous but non-differentiable at the origin.

For each function, the error of the best polynomial approximant of degree $2n$ is denoted by $E_{2n,0}$, and the error of the best rational approximant of type (n, n) by $E_{n,n}$. Errors are measured in the uniform norm on $[-1, 1]$.

Fig. (6.1) presents the results. For the smooth function $g(x) = e^{-x^4}$, the convergence curves of polynomial and rational approximants are comparable, differing only by a modest constant factor. For the non-smooth function $f(x) = |x|$, the situation is markedly different: polynomial approximation converges only algebraically, whereas rational approximation exhibits a much faster decay of the error. This demonstrates the substantial advantage of rational functions when approximating functions with singularities .



(a) Convergence of polynomial and rational best approximations to e^{-x^4} on $[-1, 1]$. The curves show the errors $E_{2n,0}$ (polynomial) and $E_{n,n}$ (rational).



(b) Convergence of polynomial and rational best approximations to $|x|$ on $[-1, 1]$. Again the curves show $E_{2n,0}$ and $E_{n,n}$. The superiority of rational approximation for the non-smooth function is evident.

Figure 6.1: Comparison of polynomial and rational best approximations on $[-1, 1]$ for a smooth function e^{-x^4} and a non-smooth function $|x|$.

6.1 Best rational approximation and equioscillation

This section introduces the structure of best rational approximants and the equioscillation principle that characterizes them. As in the polynomial case, best rational approximants exist, are unique, and satisfy an equioscillation property; however, the situation is

complicated by the possibility of *defects*, which reduce the effective number of free parameters in a rational function of type (m, n).

Theorem 6.2 (Equioscillation characterization of best approximants)**.** *Let $f \in C([-1,1])$ be real-valued. There exists a unique best real rational approximant $r^* \in \mathcal{R}_{mn}^{\text{real}}$. Moreover, a function $r \in \mathcal{R}_{mn}^{\text{real}}$ equals r^* if and only if the error $f - r$ equioscillates between at least $m + n + 2 - d$ alternating extreme points on $[-1, 1]$, where d denotes the defect of r in $\mathcal{R}_{mn}$.*

Before proving the equioscillation theorem, it is necessary to review certain structural properties of rational functions of type (m, n). The maximal number of equioscillating points is reduced by the defect parameter d. In the special case $n = 0$ and $d = 0$, one recovers the classical polynomial equioscillation theorem.

Defect: Let $r \in \mathcal{R}_{mn}$ be represented in lowest terms as $r = p/q$, where p and q have exact degrees $\mu \le m$ and $\nu \le n$, respectively. The *defect* d of r in $\mathcal{R}_{mn}$ is defined by

$$d = \min\{\, m - \mu,\ n - \nu \,\} \ge 0. \tag{6.2}$$

The defect measures the number of degrees "missing" from the numerator or denominator.

Example. Consider

$$r(x) = \frac{x^3}{1 + x^2}.$$

Then

$$d = \begin{cases} 0, & r \in \mathcal{R}_{3,2}, \\ 0, & r \in \mathcal{R}_{3,3}, \\ 0, & r \in \mathcal{R}_{3,7}, \\ 1, & r \in \mathcal{R}_{4,3}. \end{cases}$$

The defect plays a central role in rational approximation theory. A special case arises when $r \equiv 0$. By definition, the zero function has exact type $(-\infty, 0)$, and the formula (6.2) gives

$$d = n \qquad \text{for } r \equiv 0 \in \mathcal{R}_{mn}.$$

Why defects matter? Let $r = p/q$ have exact type (μ, ν) and let $\tilde{r} = \tilde{p}/\tilde{q}$ be another element of $\mathcal{R}_{mn}$. Then

$$r - \tilde{r} = \frac{p}{q} - \frac{\tilde{p}}{\tilde{q}} = \frac{p\tilde{q} - \tilde{p}q}{q\tilde{q}}.$$

Denote d as the defect of r in $\mathcal{R}_{mn}$. One finds that

$$p\tilde{q} - \tilde{p}q \quad \text{is of degree at most} \quad m + n - d,$$

and

$$q\tilde{q} \quad \text{is of degree} \quad 2n - d.$$

Thus

$$r - \tilde{r} \in \mathcal{R}_{m+n-d,\,2n-d},$$

and consequently

$$r - \tilde{r} \quad \text{has at most} \quad m + n - d \quad \text{zeros.}$$

This bound on the number of zeros is fundamental in establishing both the equioscillation property and the uniqueness of the best rational approximant.
The equioscillation theorem for rational approximants is established in four steps:

1. **Existence** of a best approximant (via compactness).
2. **Equioscillation** ⇒ **Optimality**.
3. **Optimality** ⇒ **Equioscillation**.
4. **Uniqueness** (via the zero-count bound above).

Proof. 1. Existence: The objective is to show that for every continuous function f on $[-1, 1]$ and for every pair of integers (m, n), there exists a rational function

$$r(x) = \frac{p(x)}{q(x)}, \qquad p \in \mathcal{P}_m,\ q \in \mathcal{P}_n,$$

that attains the minimum of $\|f - r\|_\infty$ over $\mathcal{R}_{mn}$.
A direct compactness argument, as in the polynomial case, is not available: bounded subsets of $\mathcal{R}_{mn}$ need not be compact. For instance,

$$r_\varepsilon(x) = \frac{x^3 + \varepsilon}{x^2 + \varepsilon}, \qquad \varepsilon > 0, \tag{6.3}$$

satisfies $\|r_\varepsilon\|_\infty = 1$ for all ε, but

$$\lim_{\varepsilon \to 0} r_\varepsilon(x) = \begin{cases} 1, & x = 0, \\ x, & x \neq 0, \end{cases}$$

which is discontinuous. Thus the limit leaves $\mathcal{R}_{32}$, showing that bounded families of rational functions of fixed type may fail to be compact (see Fig. (6.2)).
To overcome this lack of compactness, the argument proceeds at the level of numerators and denominators. The key observation is that bounded subsets of $\mathcal{P}_m$ and $\mathcal{P}_n$ are compact, even though the associated rational functions need not be.
Let $\{r_k\}$ be a sequence in $\mathcal{R}_{mn}$ satisfying

$$\|f - r_k\|_\infty \xrightarrow[k\to\infty]{} E = \inf_{r \in \mathcal{R}_{mn}} \|f - r\|_\infty, \qquad \|r_k\|_\infty \leq 2\|f\|_\infty.$$

Write $r_k = p_k/q_k$ with $p_k \in \mathcal{P}_m$ and $q_k \in \mathcal{P}_n$. Normalize each denominator by

$$\|q_k\|_\infty = 1.$$

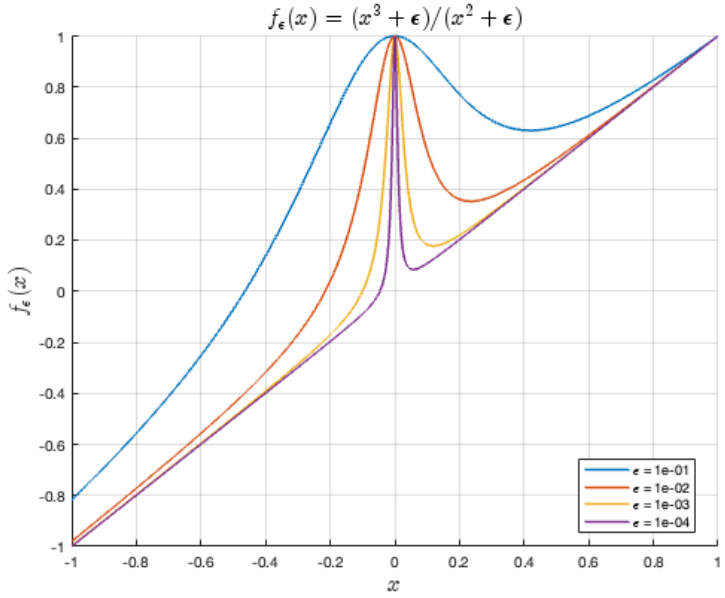


Figure 6.2: Behavior of the family (6.3) near the origin. As $\varepsilon \to 0$, a spike forms and the limit function is discontinuous.

Then

$$\|p_k\|_\infty = \|q_k r_k\|_\infty \le \|q_k\|_\infty \|r_k\|_\infty \le 2\|f\|_\infty.$$

The sets

$$\{p \in \mathcal{P}_m : \|p\| \le 2\|f\|\}, \qquad \{q \in \mathcal{P}_n : \|q\| \le 1\},$$

are compact. Hence, after passing to subsequences,

$$p_k \to p^*, \qquad q_k \to q^*,$$

for some $p^* \in \mathcal{P}_m$ and $q^* \in \mathcal{P}_n$. Since $\|q_k\|_\infty = 1$ for all k, the limit q^* is not the zero polynomial.
Define

$$r^*(x) = \frac{p^*(x)}{q^*(x)}.$$

For any x such that $q^*(x) \neq 0$,

$$r_k(x) = \frac{p_k(x)}{q_k(x)} \longrightarrow \frac{p^*(x)}{q^*(x)} = r^*(x).$$

By continuity of f,

$$|f(x) - r^*(x)| = \lim_{k\to\infty} |f(x) - r_k(x)| \le E.$$

By continuity, the same must hold for all $x \in [-1, 1]$, with p^* having zeros in $[-1, 1]$ wherever q^* does.

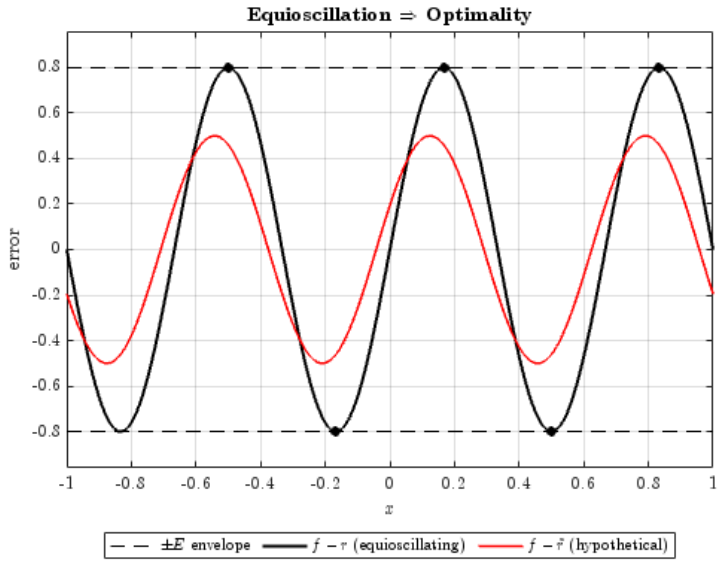


Figure 6.3: Equioscillation implies optimality: the equioscillating error (black) attains $\pm E$ at alternating points, while any other approximant (red) cannot achieve a smaller uniform error.

2. Equioscillation ⇒ Optimality: Suppose $f - r$ equioscillates between $m + n + 2 - d$ points

$$x_0 < x_1 < \cdots < x_{m+n+1-d} \in [-1, 1],$$

and suppose

$$\|f - \tilde{r}\| < \|f - r\|.$$

Then $r - \tilde{r}$ takes nonzero values of alternating sign at $x_0, \ldots, x_{m+n+1-d}$. So it has at least $m + n + 1 - d$ zeros.
But

$$r - \tilde{r} \;=\; \frac{p}{q} - \frac{\tilde{p}}{\tilde{q}} \;=\; \frac{p\tilde{q} - \tilde{p}q}{q\,\tilde{q}}$$

is of type $(m + n - d,\, 2n - d)$ while the numerator $p\tilde{q} - \tilde{p}q$ is a polynomial of degree at most $m + n - d$, and therefore cannot have $m + n + 1 - d$ zeros unless it is identically 0. Thus $r = \tilde{r}$, a contradiction. Therefore r is optimal.
3. Optimality ⇒ Equioscillation: Suppose $f - r$ equioscillates at fewer than $m + n + 2 - d$ points, and set $E = \|f - r\|$. Without loss of generality, suppose the leftmost extremum is one where $f - r$ takes the value $-E$. Then, by a compactness argument, for all sufficiently small $\varepsilon > 0$, there exist numbers

$$-1 < x_1 < x_2 < \cdots < x_k < 1, \qquad k \le m + n - d,$$

such that

$$(f-r)(x) < E-\varepsilon \quad \text{for } x \in [-1, x_1+\varepsilon] \cup [x_2-\varepsilon, x_3+\varepsilon] \cup [x_4-\varepsilon, x_5+\varepsilon] \cup \cdots,$$

and

$$(f-r)(x) > -E+\varepsilon \quad \text{for } x \in [x_1-\varepsilon, x_2+\varepsilon] \cup [x_3-\varepsilon, x_4+\varepsilon] \cup \cdots.$$

Let r be written in the form p/q, where p has degree $\mu \leq m-d$ and q has degree $\nu \leq n-d$, with p and q having no roots in common. The proof now consists in showing that r can be perturbed to a function

$$\widetilde{r} = \frac{p+\delta p}{q+\delta q} \in \mathcal{R}_{mn}$$

with the properties that $\|\widetilde{r}-r\| < \varepsilon$ and $\widetilde{r}-r$ is strictly negative for

$$x \in [-1, x_1-\varepsilon] \cup [x_2+\varepsilon, x_3-\varepsilon] \cup [x_4+\varepsilon, x_5-\varepsilon] \cup \cdots,$$

and strictly positive for

$$x \in [x_1+\varepsilon, x_2-\varepsilon] \cup [x_3+\varepsilon, x_4-\varepsilon] \cup \cdots.$$

Such a function $\widetilde{r}$ will have error less than E throughout the whole interval $[-1,1]$.
We calculate(see Appendix 11.3)

$$\widetilde{r} = \frac{p+\delta p}{q+\delta q} = \frac{(p+\delta p)(\,q-\delta q\,)}{q^2} + O(\|\delta q\|^2), \tag{6.4}$$

and therefore

$$\widetilde{r} - r = \frac{q\,\delta p - p\,\delta q}{q^2} + O(\|\delta p\|\,\|\delta q\| + \|\delta q\|^2). \tag{6.5}$$

We are done if we can show that δp and δq can be chosen so that $q\,\delta p - p\,\delta q$ is a nonzero polynomial of degree exactly k with roots $x_1, \ldots, x_k$; by scaling δp and δq sufficiently small, the quadratic terms above can be made arbitrarily small relative to the others, so that the required ε conditions are satisfied. This can be shown by the Fredholm alternative of linear algebra.
Consider the linear map

$$T : (\delta p, \delta q) \longmapsto q\,\delta p - p\,\delta q,$$

from the $(m+n+2)$–dimensional space of admissible perturbations $(\delta p, \delta q)$ to the $(m+n+1-d)$–dimensional space of polynomials of degree at most $m+n-d$. To prove that T is surjective, it suffices to verify that

$$\dim(\ker T) = d+1.$$

Indeed,

$$\dim(\text{image}\,T) = \dim(\text{domain}\,T) - \dim(\ker T) = (m+n+2) - (d+1) = m+n+1-d,$$

which equals the full dimension of the codomain. Thus T is surjective.
Assume that $(\delta p, \delta q)$ belongs to the kernel of T, i.e.

$$q\,\delta p - p\,\delta q = 0, \qquad \text{so that} \qquad q\,\delta p = p\,\delta q.$$

Since p and q have no common zeros, every root of p must also be a root of δp, and every root of q must be a root of δq. Hence there exists a polynomial g such that

$$\delta p = g\,p, \qquad \delta q = g\,q.$$

Because $\deg(\delta p) \le m$ and $\deg(\delta q) \le n$, the degree of g cannot exceed $d = \min(m - \mu,\, n - \nu)$. The space of polynomials of degree at most d has dimension $d + 1$, so the kernel of T also has dimension $d + 1$.
4. Uniqueness via equioscillation: Finally, to prove uniqueness, suppose r is a best approximation whose error curve equioscillates between extreme points at $x_0 < x_1 < \cdots < x_{m+n+1-d}$, and suppose $\|f - \tilde{r}\| \le \|f - r\|$ for some $\tilde{r} \in \mathcal{R}_{mn}^{\text{real}}$. Then (without loss of generality) $(r - \tilde{r})(x)$ must be ≤ 0 at $x_0, x_2, x_4, \ldots$ and ≥ 0 at $x_1, x_3, x_5, \ldots$. This implies that $r - \tilde{r}$ has roots in each of the $m+n+1-d$ closed intervals $[x_0, x_1], \ldots, [x_{m+n-d}, x_{m+n+1-d}]$, and since $r - \tilde{r}$ is a rational function of type $(m + n - d, 2n - d)$, the same must hold for its numerator polynomial. We wish to conclude that its numerator polynomial has at least $m + n + 1 - d$ roots in total, counted with multiplicity, implying that $r = \tilde{r}$. The numerator degree is at most $m + n - d$. The function $r - \tilde{r}$ has $m + n + 1 - d$, which means are the zeros of the numerator, but the numerator could only have $m + n - d$, unless it is the zero polynomial. □

7 Two classical problems in rational approximation.

In this chapter we take a closer look at two classical and surprisingly rich problems in rational approximation. The first is the approximation of $|x|$ on $[-1, 1]$, where the cusp at $x = 0$ makes polynomial approximation inherently slow. The second is the approximation of e^x on the half-line $(-\infty, 0]$, a setting in which the function decays rapidly but still presents challenging geometric behavior.
Our goal is to understand why these two simple functions lead to such interesting approximation phenomena, and to present the key theorems that describe their best rational approximations.

7.1 The approximation of $|x|$ on $[-1, 1]$

Bernstein proved that in the best polynomial approximation of $|x|$ as $n \to \infty$, the errors decrease linearly but no faster: they converge at the rate $O(n^{-1})$. The function $f(x) = |x|$ has a derivative of bounded variation $V = 2$ on $[-1, 1]$, so by Theorem 2.3, its Chebyshev interpolations $\{p_n\}$ satisfy

$$\|f - p_n\| \leq \frac{8}{\pi(n-1)}$$

for $n \geq 2$. Thus approximations to $|x|$ converge at least at the rate $O(n^{-1})$.
Bernstein's remarkable contribution was to show that this cannot be improved: no polynomial approximations to $|x|$ can beat Chebyshev projection or interpolation by more than a constant factor.

To study rational approximation, we define the error in the best type-(m, n) rational approximation by

$$E_{mn} := \| \, |x| - r^*_{mn} \, \|_{[-1,1]}.$$

We now state the main theorem describing this optimal error.

Theorem 7.1 ([22, Theorem 25.1])**.** *The errors in best polynomial and rational approximation of $|x|$ on $[-1, 1]$ satisfy as $n \to \infty$:*

$$E_{n0}(|x|) \sim \frac{\beta}{n}, \ \beta = 0.2801\ldots \tag{7.1}$$

and

$$E_{nn}(|x|) \sim 8e^{-\pi\sqrt{n}}. \tag{25.5}$$

The theorem tells us that polynomial approximation of $|x|$ is inherently slow: the best possible rate is only $O(n^{-1})$, and nothing faster is achievable. Rational approximations behave very differently. The error E_{nn} decreases dramatically faster—not just exponentially, but in fact at a *root-exponential rate*.

Remark 7.2. Let us briefly recall the history.

Polynomial case. The estimate $E_{n0}(|x|) \sim \beta/n$ goes back to Bernstein (1914), who proved a bound of the form const$/n$. The sharp constant was later determined by Varga & Carpenter (1975).

Rational case. For type–(n, n) approximants,

$$E_{nn}(|x|) \sim 8e^{-\pi\sqrt{n}}.$$

The first $\sqrt{n}$-type exponent goes back to Newman (1964), who proved the two-sided bounds

$$\frac{1}{2}e^{-9\sqrt{n}} \;\le\; E_{nn}(|x|) \;\le\; 3e^{-\sqrt{n}}.$$

As usual, mathematicians sought sharper constants.

– The optimal Zolotarev constant arises from his work in the 1870s.
– The value π was identified by Vyacheslavov (1975).
– The final constant 8 was obtained by Varga, Ruttan, Carpenter, and Stahl (1993) using potential-theory.

We now explain the relationship among the three functions

$$\boxed{|x|, \qquad \sqrt{x}, \qquad \sqrt{1-x^2}}.$$

Although they may appear unrelated, from the point of view of rational approximation they represent essentially the *same problem*. Suppose we consider the approximation of $|x|$ on $[-1, 1]$ by a rational function r_{nn}, which may be the best approximant (Theorem 7.1) or any other even type–(n, n) approximant:

$$\boxed{|x| \;\approx\; r_{nn}(x) \quad \text{on } [-1,1], \qquad r_{nn} \text{ even, } n \text{ even.}}$$

The equivalence begins with the identity

$$|x| = \sqrt{x^2} \;\approx\; r_{nn}(x^2) \quad \text{on } [-1,1].$$

$$\Updownarrow$$

Replacing x^2 by x and adjusting the degrees yields

$$\sqrt{x} \;\approx\; r_{\frac{n}{2},\frac{n}{2}}(x) \quad \text{on } [0,1].$$

$$\Updownarrow$$

Likewise,

$$\sqrt{1-x^2} \;\approx\; r_{nn}(x^2) \quad \text{on } [-1,1],$$

which leads to

$$\boxed{\sqrt{1-x^2} \;\approx\; r_{nn}(x) \quad \text{on } [-1,1], \qquad r_{nn} \text{ even, for } n \text{ even.}}$$

Next, instead of proving (25.5) directly, let us present a more hands-on (and rather pleasant) derivation of the estimate $E_{nn} = O(c^{-n})$ using the trapezoidal rule. Before doing so, we briefly recall some facts about contour integrals and their relation to rational functions.

Cauchy integrals and rational approximation

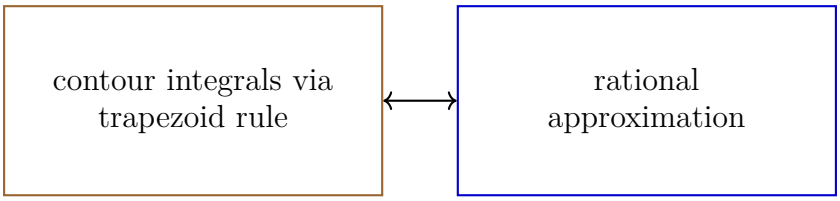
In complex analysis, one of the most powerful tools for representing analytic functions is the Cauchy integral formula. For a function f analytic inside and on a contour Γ enclosing a point a,

$$f(a) = \frac{1}{2\pi i} \int_\Gamma \frac{f(z)}{z-a}\, dz.$$

A key observation is that when a quadrature rule—such as the trapezoid rule—is applied to these contour integrals, the resulting expressions are *rational functions*. For instance, using quadrature points $\{z_k\}$ and weights $\{c_k\}$,

$$f(a) \;\approx\; \frac{1}{2\pi i} \sum_k c_k \frac{f(z_k)}{z_k - a}.$$

The formula is rational approximation $r(a)$. Thus contour integrals on the one hand and rational approximation on the other are intimately linked. The quadrature rule acts as a bridge: discretizing an analytic integral representation automatically produces a rational approximant.



Proof. We begin from the classical identity

$$\frac{1}{|x|} = \frac{2}{\pi} \int_0^\infty \frac{dt}{t^2 + x^2},$$

familiar from elementary calculus. Multiplying by x^2 gives

$$|x| = \frac{2x^2}{\pi} \int_0^\infty \frac{dt}{t^2 + x^2}.$$

Now make the substitution $t = e^s$, $dt = e^s ds$. This transforms the integral into

$$|x| = \frac{2x^2}{\pi} \int_{-\infty}^{\infty} \frac{e^s \, ds}{e^{2s} + x^2}.$$

This representation is very convenient: the integrand decays exponentially as $|s| \to \infty$, and it is analytic in the strip

$$|\operatorname{Im} s| < a = \frac{\pi}{2}.$$

For such integrals, the trapezoidal rule with step size $h > 0$ has the well-known error behavior $O(e^{-2\pi a/h})$ [21].
Applying the trapezoidal rule, we obtain the rational approximation

$$r(x) = \frac{2h\, x^2}{\pi} \sum_{k=-(n-2)/4}^{(n-2)/4} \frac{e^{kh}}{e^{2kh} + x^2}. \tag{7.2}$$

This is a rational function of type (n, n): the sum itself has type $(n-2, n)$, and the prefactor x^2 shifts it to (n, n).

There are now two sources of error:

(1) Truncation error. We have replaced an infinite sum by a finite one. Because the terms decay exponentially, the dominant contribution comes from the first omitted index, which is around $k \approx n/4$. Thus

$$\boxed{\text{truncation error } = O\left(e^{-nh/4}\right).}$$

(2) Finite step size. The trapezoidal rule itself contributes an error of size

$$O\left(e^{-\pi^2/h}\right).$$

If h is too large, the discretization error dominates; if h is too small, the truncation error dominates. So the optimal choice is obtained by balancing the two:

$$e^{-nh/4} \approx e^{-\pi^2/h}.$$

Taking logarithms:

$$-\frac{nh}{4} = -\frac{\pi^2}{h}.$$

Solving for h:

$$\frac{nh}{4} = \frac{\pi^2}{h}, \qquad h^2 = \frac{4\pi^2}{n},$$

$$\boxed{h = \frac{2\pi}{\sqrt{n}}}.$$

With this choice we obtain

$$\boxed{\text{error} \;=\; O\Big(e^{-\pi\sqrt{n}/2}\Big)\,.}$$

This is slightly weaker than the sharp constant in (25.5): we lose a factor of 4 inside the $\sqrt{n}$, meaning that we would need n to be about four times larger to achieve the same accuracy. Nevertheless, the derivation is simple, insightful, and still yields the correct root-exponential behavior. □

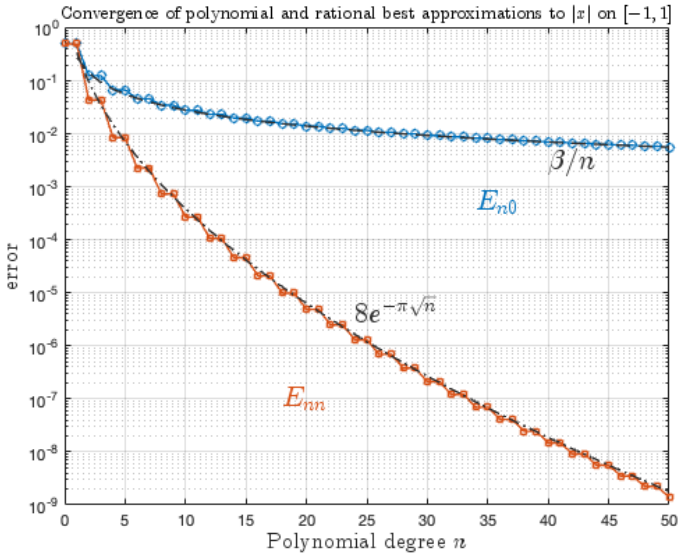


Figure 7.1: Convergence of best uniform polynomial and rational approximations of $|x|$ on $[-1,1]$. The dotted and dash-dotted curves show the asymptotic rates (7.1)–(25.5).

Using Chebfun, we compute the degree-n best polynomial approximation (via `remez`), and compare with values of the best rational approximation error. The results are shown in Fig. (7.1). The numerical results confirm the theory: polynomial approximation converges algebraically at rate $O(n^{-1})$, while rational approximants converge much faster, with root-exponential decay $O(e^{-\pi\sqrt{n}})$.

7.2 The approximation of e^x on $(-\infty, 0]$

We now turn to the second of the classical problems in this chapter: approximating e^x on the half-line $(-\infty, 0]$. As before, we measure the quality of approximation by

$$E_{mn} := \| e^x - r^*_{mn} \|_{(-\infty,0]},$$

where r^*_{mn} is the best type–(m,n) rational approximant. The next result describes the two fundamental asymptotic behaviors.

Theorem 7.3. *For the best type* $(0,n)$ *and* (n,n) *rational approximations of* $\exp(x)$ *on* $(-\infty,0]$*, the errors satisfy*

$$\lim_{n\to\infty} E_{0n}^{1/n} = \frac{1}{3},$$

and

$$E_{nn} \sim 2H^{\,n+1/2}, \qquad H = \frac{1}{9.2890254919208\ldots},$$

where H *is Halphen's constant.*

A first simple observation is that polynomial approximation does not work at all on this interval. Since any non-constant polynomial $p(x)$ diverges to $\pm\infty$ as $x \to -\infty$, the only polynomials with finite error on $(-\infty,0]$ are constants. This means that the minimax polynomial error is at least $1/2$, so polynomials simply cannot approximate e^x effectively on this domain.
In contrast, rational approximations behave much better. Inverse polynomials of the form $1/p_n(x)$ can be chosen to converge geometrically on $(-\infty,0]$. Cody, Meinardus, and Varga showed that if p_n is the degree-n Taylor polynomial of e^x, then $1/p_n(x)$ achieves an error of order 2^{-n}; this was later improved to 2.298^{-n} by shifting the expansion point. Schönhage proved that the optimal rate for inverse-polynomial approximation is 3^{-n}. Since $1/p_n(x)$ is a rational function of type (n,n), this shows that type–(n,n) rational approximants converge at least geometrically. Newman proved that the rate cannot be better than geometric. For many years the conjectured optimal rate was 9^{-n} (the classical "1/9 conjecture"), but the true constant is $H \approx 1/9.28903$, Halphen's constant.

Proof. We now sketch how the geometric behavior of E_{nn} can be recovered from a trapezoidal-rule approximation.
Begin with the Cauchy integral formula [1]

$$e^x = \frac{1}{2\pi i}\int_\Gamma \frac{e^t\,dt}{t-x},$$

where Γ is a contour enclosing the point x. Next, choose Γ to be a parabola and make the change of variables

$$t = (is+a)^2, \qquad dt = 2i(is+a)\,ds,$$

with $a > 0$. This transforms the integral into

$$e^x = \frac{1}{\pi}\int_{-\infty}^{\infty} \frac{e^{(is+a)^2}(is+a)\,ds}{(is+a)^2 - x}.$$

[1] This identity follows directly from the residue theorem: the integrand $f(t) = \frac{e^t}{t-x}$ has a simple pole at $t = x$ with residue e^x,

$$\mathrm{Res}_{t=a}\, f(t) = \lim_{t\to a}(t-a)\,f(t), \text{ so } \frac{1}{2\pi i}\int_\Gamma \frac{e^t}{t-x}\,dt = e^x$$

.

Applying the trapezoidal rule with step size $h > 0$ gives the rational approximation

$$r(x) = \frac{h}{\pi} \sum_{k=-(n-1)/2}^{(n-1)/2} \frac{e^{(ikh+a)^2}(ikh+a)}{(ikh+a)^2 - x},$$

which is of type (n, n).
To obtain a geometric rate of convergence, one must choose a and h so that the discretization and truncation errors balance. Taking

$$a = \sqrt{\frac{\pi n}{24}}, \qquad h = \sqrt{\frac{3\pi}{2n}},$$

leads to the estimate

$$\|e^x - r_{nn}(x)\| = O\Big(e^{-\pi n/3}\Big) \approx O\big((2.849\ldots)^{-n}\big)\,.$$

This is already good evidence of geometric convergence, though still far from the optimal constant $\frac{1}{H} = 9.2890254919208\ldots$, proved much later in the deep work of Aptekarev, Tulyakov, and collaborators (see [2]). □

We finish this chapter by showing that the numerical computation of these best approximants is surprisingly easy. The crucial matter is to note that the change of variables

$$x = a\,\frac{s-1}{s+1}, \qquad s = \frac{a+x}{a-x} \tag{25.19}$$

where a is a positive parameter, maps the negative real axis $(-\infty, 0]$ in x to the interval $(-1, 1]$ in s. Since the mapping is a rational function of type $(1, 1)$, it transplants a rational function of type (n, n) in s or x to a rational function of type (n, n) in the other variable. In particular, for the approximation of $f(x) = e^x$ on $(-\infty, 0]$, let us define

$$F(s) = e^{a(s-1)/(s+1)}, \qquad s \in (-1, 1]. \tag{25.20}$$

A good choice of the parameter is $a = 9$, which has a big effect for numerical computation in improving the conditioning of the approximation problem. We now find we have a function that can be approximated to machine precision by a Chebyshev interpolating polynomial $p(s)$ of degree less than 50:

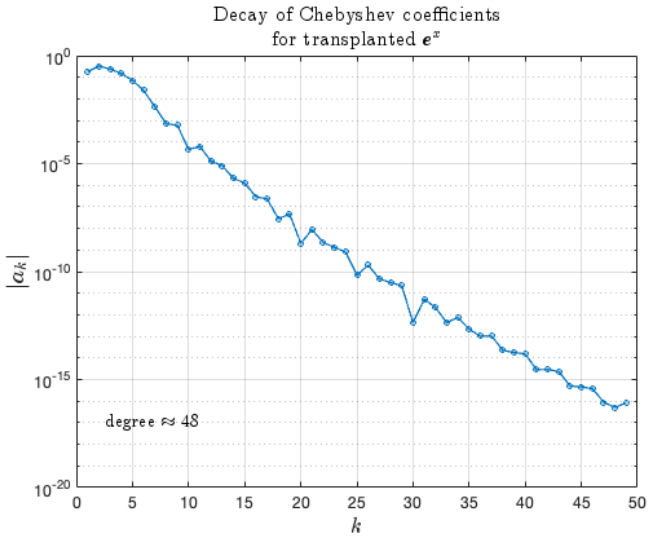


Figure 7.2: Chebyshev coefficients of the transplanted exponential $F(s) = \exp(9\,\frac{s-1}{s+1})$ on $[-1, 1]$. The rapid decay of the coefficients shows that F is extremely well approximated by a polynomial of moderate degree (here `length(F)` is about 47 in CHEBFUN), reflecting the analyticity of F in a Bernstein ellipse containing $[-1, 1]$.

8 Rational interpolation and linearized least-squares

When working with polynomials, we discovered a reassuring fact: although best (minimax) approximants have elegant theoretical properties, in practice Chebyshev interpolation or projection usually delivers accuracy that is just as good, while remaining computationally simpler because the problem stays fully linear.
Rational approximation follows a somewhat similar pattern. Best rational approximants are mathematically very sophisticated, but for computational purposes it is often more practical to use rational interpolants, where we can again exploit linearity once we multiply through by the denominator.
At the same time, rational interpolation displays certain features that do not appear in the purely polynomial case. The formulation is not entirely linear, and in some settings a solution may fail to exist, or it may respond very sensitively to the data. These phenomena are not flaws; they arise from the greater expressive power of rational functions.
To treat these aspects in a stable and systematic way, we will reformulate the problem using the singular value decomposition (SVD) together with a least-squares perspective. Assume $m, n \geq 0$, and suppose we are given exactly the needed data $\{f(x_k)\}_{k=0}^{m+n}$. It is natural to ask whether we can construct a rational function that interpolates this data. We look for a numerator $p \in P_m$ and a denominator $q \in P_n$ satisfying

$$f(x_k) = \frac{p(x_k)}{q(x_k)}, \qquad k = 0, 1, \cdots m + n. \tag{8.1}$$

Questions of existence, uniqueness, and well-posedness inevitably arise. They are closely related to the notion of defect, which we will discuss shortly. To motivate these ideas, let us consider some examples illustrating nonexistence, nonuniqueness, and sensitivity to perturbations. (All examples will be of type $(1, 1)$.)

8.1 Nonexistence, Nonuniqueness, and Ill-posedness

- **Nonexistence.** Seek $r \in \mathcal{R}_{11}$ with

  $$r(-1) = r(0) = 1, \qquad r(1) = 2.$$

  Any $r \in \mathcal{R}_{11}$ has the form

  $$r(x) = \frac{ax + b}{cx + d},$$

which is either constant or a one-to-one Möbius transformation. Neither can take two equal values and a third distinct one. Hence no such r exists.

- **Nonuniqueness.** Seek $r \in \mathcal{R}_{11}$ with

$$r(-1) = r(0) = 0.$$

The rational function is uniquely $r = 0$, but its representation is not: $p = 0$ and *any* denominator q give the same function. Algorithms that solve for (p, q) separately must therefore treat this freedom with care.

- **Ill-posedness.** Now impose

$$r(-1) = 1 + \varepsilon, \qquad r(0) = 1, \qquad r(1) = 1 + 2\varepsilon.$$

A solution exists for all ε:

$$r(x) = \begin{cases} 1 + \dfrac{\frac{4}{3}\varepsilon\, x}{x - \frac{1}{3}}, & \varepsilon \neq 0, \\ 1, & \varepsilon = 0. \end{cases} \tag{8.2}$$

As $\varepsilon \to 0$, these functions converge to 1 everywhere except at $x = \frac{1}{3}$, where the pole persists. The interpolation problem is therefore unstable near $\varepsilon = 0$.

In order to determine when the interpolation problem (8.1) has a solution and how to find the solution, we need to examine the problem more carefully. We begin with a definition.

Definition 8.1. *Let $r = \frac{p}{q}$ and $\tilde{r} = \frac{\tilde{p}}{\tilde{q}}$ be elements of $\mathcal{R}_{mn}$. We say that r and $\tilde{r}$ are equal if*

$$p\,\tilde{q} = q\,\tilde{p}.$$

Theorem 8.2. *The interpolation problem in $\mathcal{R}_{mn}$, that is, the problem of satisfying* (8.1)*, has at most one solution.*

Proof. If $r, \tilde{r} \in \mathcal{R}_{mn}$ satisfy (8.1), then

$$r(x_k) - \tilde{r}(x_k) = 0, \qquad k = 0, \ldots, m + n,$$

and hence

$$p(x_k)\tilde{q}(x_k) - \tilde{p}(x_k)q(x_k) = 0.$$

But $p\tilde{q} - \tilde{p}q \in P_{m+n}$ and is therefore identically zero. □

If $r = p/q$ satisfies (8.1), the coefficients of p and q satisfy the following system of homogeneous linear equations:

$$q(x_k)f_k - p(x_k) = (b_0 + b_1x_k + \cdots + b_nx_k^n)f_k - (a_0 + \cdots + a_mx_k^m) = 0, \qquad k = 0, \ldots, m+n. \tag{8.3}$$

Equation (8.3) consists of $m+n+1$ equations in $m+n+2$ unknowns and therefore always has a nontrivial solution. Moreover, each nontrivial solution of (8.3) defines a rational function; that is, no nontrivial solution has $b_0 = b_1 = \cdots = b_n = 0$. For if this were the case, then we would have $p(x_k) = 0$, $k = 0, \ldots, m+n$, hence $p = 0$, that is, $a_0 = \cdots = a_m = 0$.

Theorem 8.3. *All nontrivial solutions of* (8.3) *define the same rational function.*

Proof. Suppose that $q, \tilde{q} \in P_n$, $p, \tilde{p} \in P_m$, and

$$q(x_k)f_k - p(x_k) = \tilde{q}(x_k)f_k - \tilde{p}(x_k) = 0, \qquad k = 0, \ldots, m+n.$$

Then

$$p(x_k)\tilde{q}(x_k) - q(x_k)\tilde{p}(x_k) = 0, \qquad k = 0, \ldots, m+n,$$

and

$$p\tilde{q} - q\tilde{p} \in P_{m+n}.$$

□

Thus the linear equations (8.3) lead to a unique rational function, call it $\bar{r} \in \mathcal{R}_{mn}$. If (8.1) is satisfied, then $\bar{r}$ is the rational function that interpolates.

8.2 Froissart doublet

Getting back to equation (8.2), the pole is

$$x_p = \frac{1}{3}.$$

The zero satisfies

$$1 + \frac{\frac{4}{3}\varepsilon\, x}{x - \frac{1}{3}} = 0, \qquad x\Big(1 + \tfrac{4}{3}\varepsilon\Big) = \tfrac{1}{3},$$

and hence

$$x_z = \frac{\frac{1}{3}}{1 + \frac{4}{3}\varepsilon} = \frac{1/3}{1 - \frac{4}{3}\varepsilon} + O(\varepsilon^2).$$

The pole and zero are therefore separated by only $O(\varepsilon)$, forming a spurious pole–zero pair (known as a Froissart doublet). Such doublets are characteristic of rational interpolation and explain why convergence is often obtained in capacity rather than uniformly.
In floating-point computation, Froissart doublets arise even more frequently because rounding errors tend to generate near-cancelling pole–zero pairs. These spurious pole–zero pairs neither reflect genuine information about the function f nor contribute to the quality of the approximation. Numerically, such doublets can be identified when their residues are close to machine precision. These difficulties are ultimately connected with

the ill-posedness of analytic continuation[1], for which rational approximation is the most powerful general tool.
Let us look at a computed example. We interpolate the function

$$f(x) = e^x + 0.3\sin(3x)$$

on $[-1, 1]$ at $2n+1$ Chebyshev points and compute type (n, n) rational interpolants using the `ratinterp`[2] command in Chebfun.
For $n = 1$ and $n = 2$ (interpolating at three and five points), the convergence looks excellent, and the same holds for $n = 3$ as shown in Fig.(8.1). At $n = 4$, however, a pole suddenly appears, even though f is analytic and has no singularities in $[-1, 1]$. Such a pole is not related to the function f itself and is accompanied by a nearby zero; together they form a spurious pole–zero pair.
This behavior can be explained as follows. For small values of m and n, all degrees of freedom in the rational approximant are needed, and the resulting poles tend to reflect the analytic structure of the function being approximated. As m and n increase, or even if m increases while n is fixed, the approximant acquires more parameters than are required to approximate f to machine precision. In this regime the method begins to fit rounding errors rather than the function, and spurious poles appear. In finite precision such effects are even more common, and perturbations can introduce poles that would not exist in exact arithmetic.
In the present example, the spurious pole has an extremely small residue, so it is almost cancelled by a nearby zero. At any moderate distance their influence is negligible, yet the pair is present and must be accounted for. This illustrates a general lesson: rational approximants can exhibit unexpected behaviour even when their approximation error is small. In our case we see beautiful convergence, with only one exception, and no complete theoretical explanation fully accounts for such phenomena. They are not caused solely by rounding errors—though rounding makes them more delicate—but are inherent to the ill-posed nature of analytic continuation.
Next we sketch an approach for dealing with these issues. What we need is a form of *regularization*. Regularization refers broadly to procedures that stabilize ill-posed problems by suppressing the influence of small perturbations or by imposing additional structure. Rational interpolation is inherently sensitive to perturbations, since small changes in the data may cause large changes in the positions of poles and zeros. A regularized approach is therefore essential.
Our strategy consists of three main steps:

- linearize the interpolation equations,
- oversample so that the system becomes rectangular rather than square,

[1]The goal is typically to gain information about a function in a region of the complex plane using information at a single point.
[2]From this chapter through the end of the manuscript, the MATLAB codes used to generate all figures are available at the following link:`https://github.com/zavala92/Approximation-theory-course-at-TU`

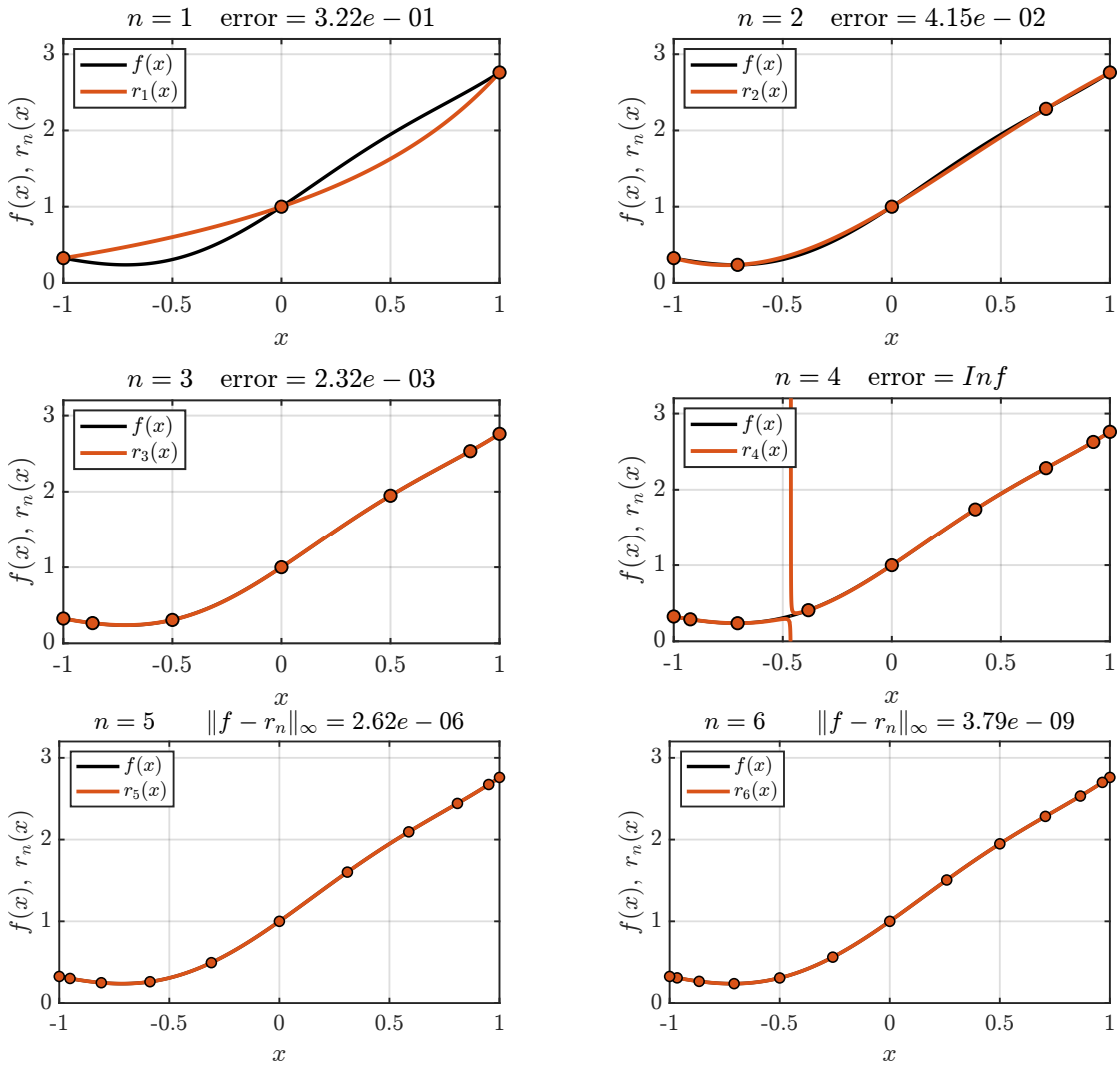


Figure 8.1: Type (n, n) rational interpolants of $f(x) = e^x + 0.3\sin(3x)$ on $[-1, 1]$ constructed from $2n+1$ Chebyshev points. For $n = 1, 2, 3, 5, 6$ the interpolants converge well, while for $n = 4$ a spurious pole–zero pair appears, leading to loss of convergence despite the analyticity of f.

- regularize via the SVD by discarding the smallest singular values.

These steps are conceptually simple but remarkably effective. They convert an unstable nonlinear problem into a robust linear-algebra problem.

Linearization. Begin by replacing the nonlinear conditions

$$f(x_k) = \frac{p(x_k)}{q(x_k)}, \qquad k = 0, \dots, m+n,$$

with the linear relations

$$f(x_k)\, q(x_k) = p(x_k).$$

This reformulation avoids division and produces a linear system for the unknown coefficients of p and q. We write the polynomials in the Chebyshev basis

$$p(x) = \sum_{j=0}^{m} a_j\, T_j(x), \qquad q(x) = \sum_{j=0}^{n} b_j\, T_j(x),$$

so that the interpolation conditions become

$$f(x_k)\, q(x_k) = p(x_k), \qquad k = 0, \dots, m+n.$$

In matrix form this becomes

$$\begin{pmatrix} f(x_0) & & \\ & \ddots & \\ & & f(x_{m+n}) \end{pmatrix} \begin{pmatrix} T_0(x_0) & \cdots & T_n(x_0) \\ \vdots & & \vdots \\ T_0(x_{m+n}) & \cdots & T_n(x_{m+n}) \end{pmatrix} \begin{pmatrix} b_0 \\ \vdots \\ b_n \end{pmatrix} = \begin{pmatrix} p(x_0) \\ \vdots \\ p(x_{m+n}) \end{pmatrix}.$$

The first matrix is the diagonal matrix

$$\operatorname{diag}(f(x_0), \dots, f(x_{m+n})),$$

and the second is the Chebyshev–Vandermonde matrix of size $(m+n+1) \times (n+1)$, consisting of the sampled values of the Chebyshev polynomials $T_0, \dots, T_n$.

Transforming to coefficient space. To improve conditioning, we multiply the system on the left by the discrete Chebyshev transform matrix

$$C \in \mathbb{R}^{(m+n+1)\times(m+n+1)},$$

which maps sampled values to Chebyshev coefficients. For a sample vector $v = (v(x_0), \dots, v(x_{m+n}))^T$,

$$C\, v = (\hat{v}_0, \dots, \hat{v}_{m+n})^T,$$

where the $\hat{v}_j$ are the Chebyshev coefficients of v. Applying C to the linearized system yields

$$\widehat{C}\begin{pmatrix} b_0 \\ b_1 \\ \vdots \\ b_n \end{pmatrix} = \left(\begin{array}{c} a_0 \\ a_1 \\ \vdots \\ a_m \\ \hline 0 \\ \vdots \\ 0 \end{array}\right) \qquad \widehat{C} \in \mathbb{R}^{(m+n+1)\times(n+1)}. \tag{8.4}$$

The last n entries of the right-hand side are zero because p has degree at most m. Thus the bottom part of the system gives

$$\widetilde{C}\begin{pmatrix} b_0 \\ b_1 \\ \vdots \\ b_n \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix},$$

where $\widetilde{C}$ is of size $(n+1)\times n$. The problem reduces to finding a null vector of a rectangular matrix. This null vector defines the coefficients of q, then using the upper part of the system (8.4) we find the coefficients of p.

SVD regularization. We now use the singular value decomposition to extract a stable approximate null vector. The smallest singular values correspond to directions in which the matrix is nearly rank-deficient; such directions represent candidate denominators q. Discarding the smallest singular values suppresses the influence of rounding errors and noise, yielding a regularized denominator. Once q is known, the earlier linear system determines the Chebyshev coefficients of the numerator p.
Oversampling plays an important role here. By taking more sample points than unknown coefficients, the problem becomes a least-squares problem; the SVD is extremely effective in identifying vectors that map as close to zero as possible, even when no exact null vector exists. This is a standard and powerful form of regularization.

The result is a rational interpolant that mitigates the instabilities inherent in the unregularized formulation. It should be emphasized, however, that rational interpolants—even when regularized—are not known to converge uniformly in general, and no universally convergent regularization method is currently available. In practice, SVD-based regularization has proven to be the most reliable approach. Accordingly, all computations and examples presented below rely on the SVD-regularized rational interpolation function `ratdisk` [9].

We illustrate these effects with a simple example. Consider the function

$$f(z) = \frac{e^z}{1.1 - z^2},$$

which is analytic in the complex plane except for two simple poles at

$$z = \pm\sqrt{1.1}.$$

Each plot in Fig. (8.2) shows the approximation of f on the $(N+1)$st roots of unity by a rational function of type (m, n); the unit circle is also shown. The triplet (m, n, N) is listed in the upper-right corner of each plot. A label in the upper-left reads *Interpolation* when $N = m + n$ and *Least-squares* when $N > m + n$; in the latter case we take $N = 4(m + n) + 1$. The lower-left corner lists the exact type (μ, ν) of the computed rational interpolant.
In each plot, poles of the rational approximant are shown as dots, whose color indicates the magnitude of the corresponding residue, computed by a finite-difference approximation, following the color scheme of [9]:

$$|\text{residue}| \in \begin{cases} [10^{-3}, \infty) & \text{blue,} \\ [10^{-6}, 10^{-3}) & \text{light blue,} \\ [10^{-9}, 10^{-6}) & \text{green,} \\ [10^{-12}, 10^{-9}) & \text{light green,} \\ [10^{-14}, 10^{-12}) & \text{pink,} \\ (0, 10^{-14}) & \text{red.} \end{cases}$$

Blue and green correspond to relatively large residues, while pink and red indicate very small residues.

For a rational interpolant of type $(5, 5)$ (top row), exactly two poles are observed, located near $z = \pm\sqrt{1.1}$, and no spurious poles are present. When the approximation order is increased to $(10, 10)$ without regularization (middle row), additional poles appear that do not correspond to singularities of f. These spurious poles arise from over-parameterization and sensitivity to numerical perturbations.
In the third row, a regularized $(10, 10)$ approximation is shown. The spurious pole–zero pairs are eliminated, while the two genuine poles are preserved. Notably, the exact type of the resulting rational interpolant is reduced to $(10, 6)$, indicating that regularization has automatically removed numerically insignificant degrees of freedom.

We next consider the function

$$f(z) = \frac{\log(2 + z^4)}{1 - 16z^4},$$

which is even and exhibits a fourfold symmetry.

We first examine a low-denominator approximation. Fig. (8.3) shows rational approximations of type $(100, 4)$. Since the denominator degree is small, no spurious poles appear, and regularization makes no significant difference. Next, we increase the degree of the denominator.
Fig. (8.4) shows a type $(100, 100)$ approximation of the even function $\log(2+z^4)/(1-16z^4)$. In addition to the useful poles tracking the fourfold symmetric branch cuts, many spurious

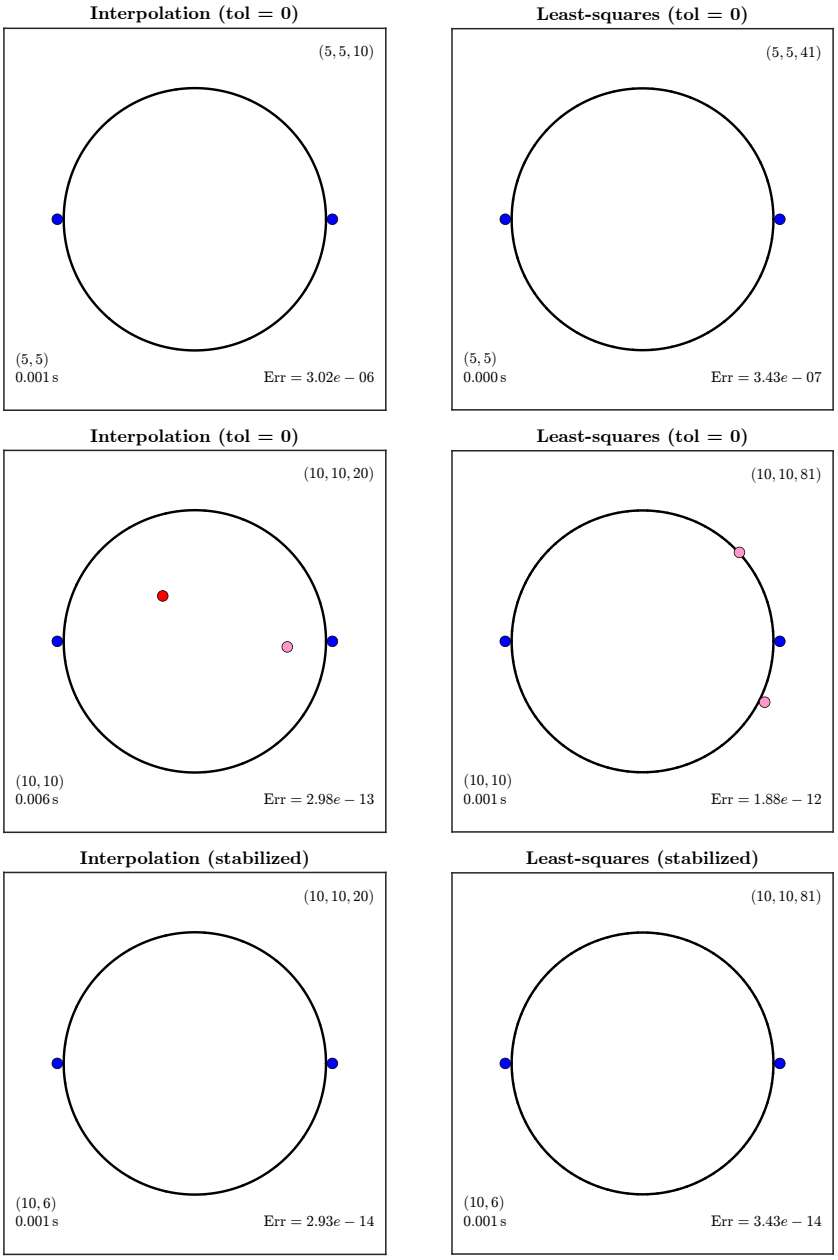


Figure 8.2: Rational approximations of $f(z) = \frac{e^z}{1.1-z^2}$.

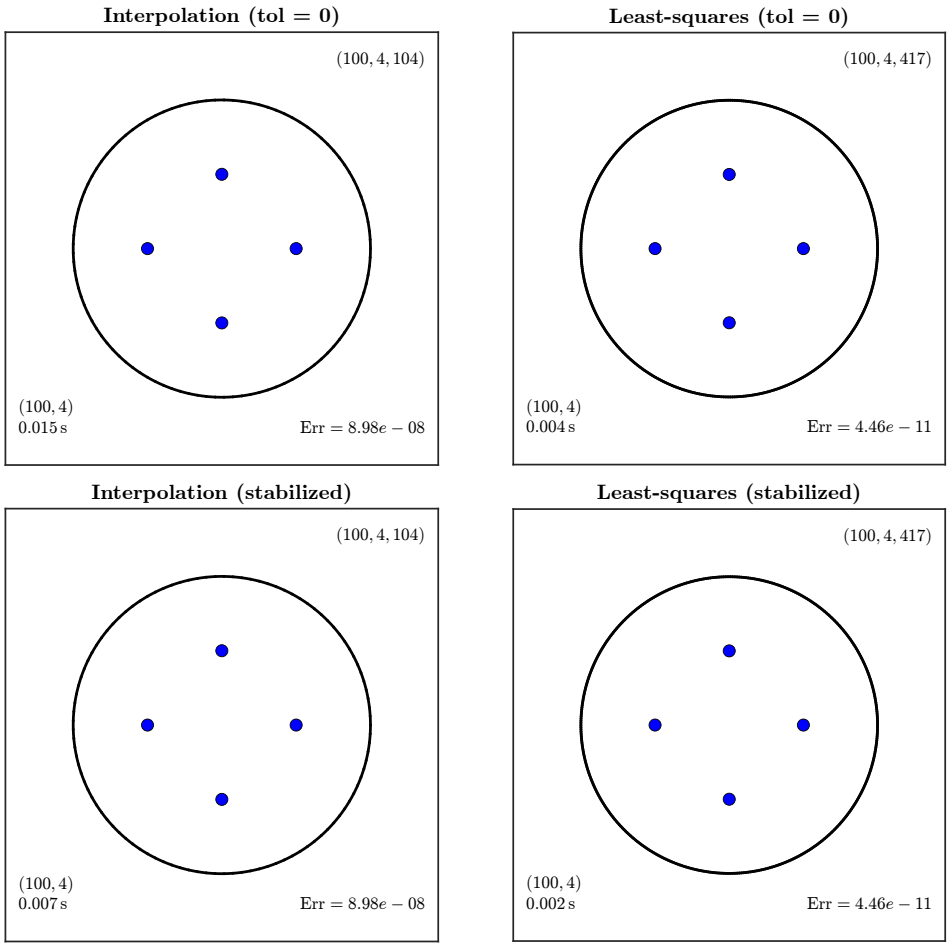


Figure 8.3: Type $(100, 4)$ rational approximations of $f(z) = \log(2 + z^4)/(1 - 16z^4)$. Top: unregularized. Bottom: regularized.

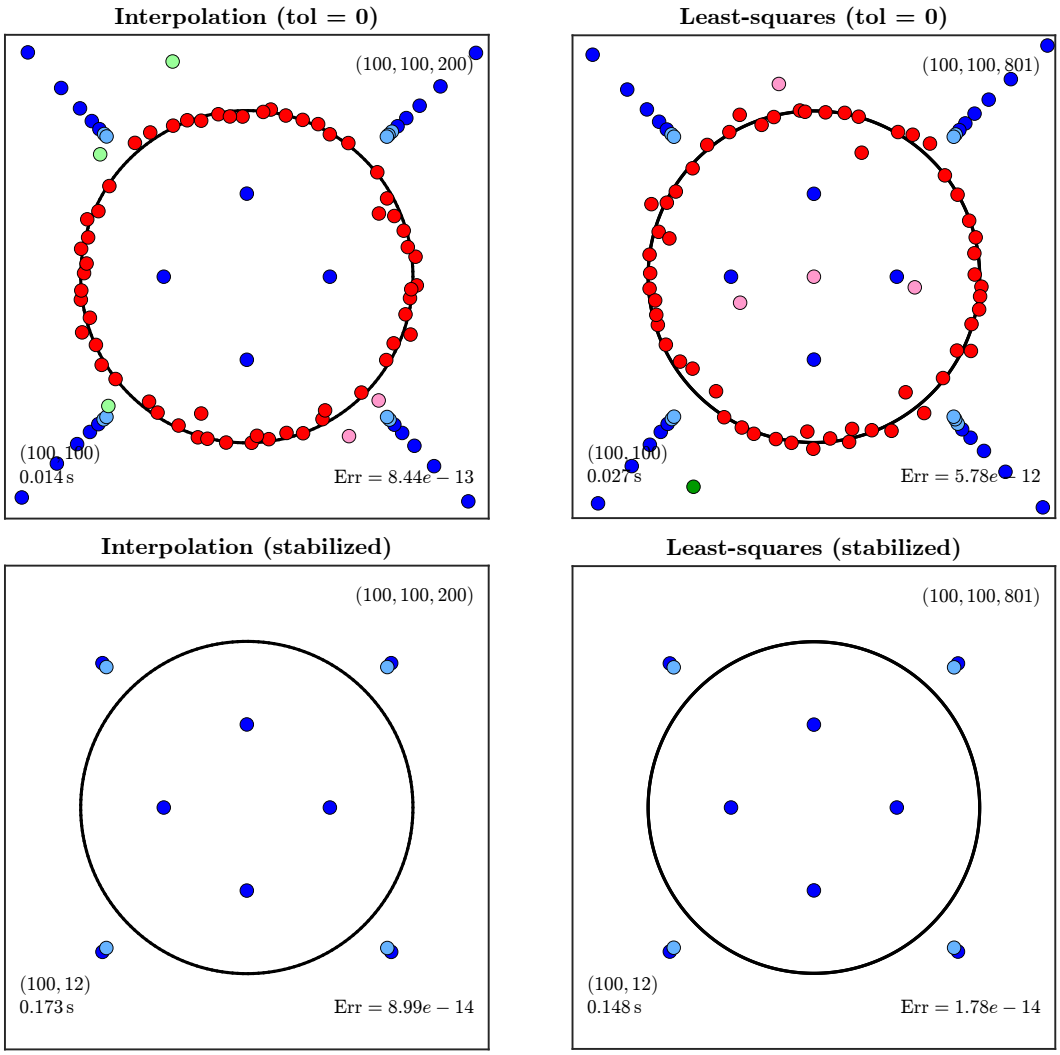


Figure 8.4: Type $(100, 100)$ rational approximations of $\log(2 + z^4)/(1 - 16z^4)$. Top: unregularized. Bottom: regularized.

poles are present. As in previous examples, these spurious poles are not symmetric, despite the symmetry of the function. In the lower plot the symmetries are enforced and the type is reduced, with both μ and ν divisible by four as a consequence of the fourfold symmetry.

9 Quadrature formulas from rational approximations

Many of the best-known quadrature formulas, including Newton–Cotes, Gauss, and Clenshaw–Curtis rules, are based on polynomial interpolation. Given $n+1$ nodes $\{z_k\}$, the corresponding weights $\{c_k\}$ are chosen so that the quadrature rule integrates all polynomials of degree at most n exactly. Equivalently, applying the quadrature rule to a function f returns the exact integral of the unique polynomial of degree n that interpolates the values $\{f(z_k)\}$ at the nodes. For this reason, such rules are commonly called *interpolatory* quadrature formulas, more precisely *polynomial interpolatory* quadrature.
In this chapter we introduce a different, but closely related, point of view. Instead of polynomial interpolation, we use *rational approximation* to construct quadrature formulas. The key idea is that the quadrature nodes can be identified with the poles of a rational approximant, while the quadrature weights are given by the corresponding residues. From this perspective, quadrature formulas arise naturally from complex approximation theory, and their structure becomes easier to interpret geometrically.
The material in this chapter is based on recent work by Horning and Trefethen [12], who showed that quadrature rules can be generated in a simple and systematic way from rational approximation. Their approach provides new insight into both classical quadrature formulas and modern generalizations.

9.1 Quadrature as integration of a rational interpolant

We begin with the geometric setup. Let γ be a Jordan arc in the complex plane $\mathbb{C}$, i.e. a non-self-intersecting smooth curve that is the image of $[-1,1]$ under a real or complex homeomorphism. Assume $\gamma \subset \Omega$, where Ω is a Jordan region, and let $\Gamma = \partial\Omega$ be its boundary.
Assume f is analytic in the closure $\overline{\Omega} = \Omega \cup \Gamma$, allowing us to apply Cauchy's integral formula for all $z \in \gamma$. Let $w(z)$ be an integrable weight defined on γ. Our object of study is the weighted contour integral

$$I = \int_\gamma f(z)\, w(z)\, dz. \tag{9.1}$$

For a function f analytic inside and on a contour Γ enclosing a point z, Cauchy's integral formula asserts that

$$f(z) = \frac{1}{2\pi i} \int_\Gamma \frac{f(s)}{s-z}\, ds, \qquad z \in \gamma. \tag{9.2}$$

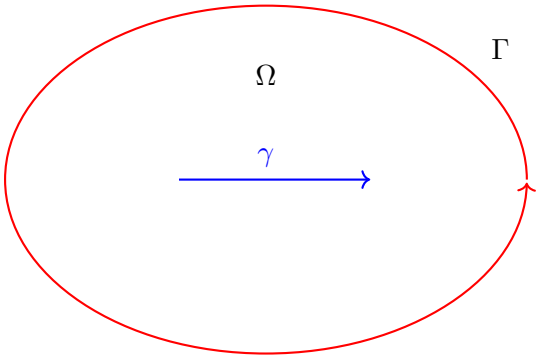


Figure 9.1: The curve γ lies within a Jordan region Ω whose boundary is Γ.

Substituting (9.2) into (9.1) gives

$$I = \int_\gamma \left[\frac{1}{2\pi i} \int_\Gamma \frac{f(s)}{s-z}\, ds \right] w(z)\, dz. \tag{9.3}$$

Because $s \in \Gamma$ and $z \in \gamma$ never coincide and the integrand is continuous and analytic in both variables, we may exchange the order of integration and obtain

$$I = \int_\Gamma f(s) \left[\frac{1}{2\pi i} \int_\gamma \frac{w(z)}{s-z}\, dz \right] ds. \tag{9.4}$$

It is natural at this point to introduce the Cauchy transform of the weight w. Let $C(s)$ denote the function in square brackets, i.e.

$$C(s) = \frac{1}{2\pi i} \int_\gamma \frac{w(z)}{s-z}\, dz. \tag{9.5}$$

Then (9.4) can be rewritten in the compact form

$$I = \int_\Gamma f(s)\, C(s)\, ds. \tag{9.6}$$

The function $C(s)$ is analytic in $\mathbb{C} \bigcup \{\infty\} \setminus \gamma$. This representation is the starting point for constructing quadrature formulas from rational approximations to C.
Let us recall the rational interpolation problem. The task is to determine a rational function $r_{mn} \in \mathcal{R}_{mn}$ that satisfies

$$r_{mn}(z_i) = f(z_i), \qquad i = 1, \ldots, k, \tag{9.7}$$

where $z_1, \ldots, z_k$ are distinct real numbers and $f_1, \ldots, f_k$ are arbitrary prescribed values. The number $k = m+n+1$ is the largest for which one can hope to satisfy (9.7) in general, since a rational function of type (m, n) contains exactly $m + n + 1$ free parameters once normalization is taken into account.
In the context of quadrature derived from Cauchy transforms, we now consider a rational function $r_{nn}(s)$ of degree at most n that approximates $2\pi i\, C(s)$ on Γ. For what follows we

assume that $r_{nn}(s)$ interpolates $2\pi i\, C(s)$ at $n+1$ points $s_0, \ldots, s_n \in (\mathbb{C} \cup \{\infty\}) \setminus \gamma$, which need not be distinct and need not be finite. Typically there will in fact be more than $n+1$ interpolation points, generically $2n+1$ or more for best or near-best approximations, in which case any $n+1$ of them may be chosen as $s_0, \ldots, s_n$. The same weights $\{c_k\}$ are determined by these $n+1$ interpolation conditions, regardless of which particular points $s_0, \ldots, s_n$ are selected.
We define the approximate value of the integral

$$I_n = \frac{1}{2\pi i} \int_\Gamma f(s)\, r_{nn}(s)\, ds. \tag{9.8}$$

Assume that the n poles of r_{nn} are distinct finite numbers $z_1, \ldots, z_n$. Then r_{nn} has the partial fraction decomposition

$$r_{nn}(s) = c_\infty + \sum_{k=1}^{n} \frac{c_k}{s - z_k}, \tag{9.9}$$

where $c_\infty = r_{nn}(\infty)$ is typically very small because $c(\infty) = 0$, hence a good approximation must nearly vanish at infinity.
Assuming all poles z_k lie inside Γ, the residue theorem leads directly to the desired quadrature formula. We begin with the rational–approximation expression

$$I_n = \frac{1}{2\pi i} \int_\Gamma f(s)\, r_{nn}(s)\, ds,$$

where r_{nn} has the partial fraction representation

$$r_{nn}(s) = c_\infty + \sum_{k=1}^{n} \frac{c_k}{s - z_k},$$

and all the poles z_k lie inside the contour Γ. Substituting this expansion into the definition of I_n gives

$$I_n = \frac{1}{2\pi i} \int_\Gamma c_\infty f(s)\, ds + \frac{1}{2\pi i} \sum_{k=1}^{n} c_k \int_\Gamma \frac{f(s)}{s - z_k}\, ds.$$

For each k, the integrand $(f(s)/(s - z_k))$ has a simple pole at $s = z_k$, and by the residue theorem,

$$\int_\Gamma \frac{f(s)}{s - z_k}\, ds = 2\pi i\, f(z_k).$$

Hence

$$I_n = \sum_{k=1}^{n} c_k f(z_k),$$

which is the desired quadrature rule: the poles z_k serve as the quadrature nodes and the residues c_k serve as the quadrature weights.
To estimate the error of this quadrature rule, suppose the rational approximation satisfies

$$\| 2\pi i C - r_{nn} \|_\Gamma \le \varepsilon, \tag{9.10}$$

where $\|\cdot\|_\Gamma$ denotes the maximum norm on Γ. We find

$$I - I_n = \int_\Gamma f(s) \left(C(s) - \frac{1}{2\pi i} r_{nn}(s) \right) ds.$$

Taking absolute values and applying the triangle inequality yields

$$|I - I_n| \le \int_\Gamma \left| f(s) \left(C(s) - \tfrac{1}{2\pi i} r_{nn}(s) \right) \right| |ds|.$$

Using the bounds $|f(s)| \le \|f\|_\Gamma$ and

$$|C(s) - \tfrac{1}{2\pi i} r_{nn}(s)| \le \tfrac{1}{2\pi} \|2\pi i C - r_{nn}\|_\Gamma,$$

we obtain

$$|I - I_n| \le \frac{1}{2\pi} \|f\|_\Gamma \|2\pi i C - r_{nn}\|_\Gamma \int_\Gamma |ds| = \frac{1}{2\pi} |\Gamma| \, \|f\|_\Gamma \|2\pi i C - r_{nn}\|_\Gamma.$$

Thus, if $\|2\pi i C - r_{nn}\|_\Gamma \le \varepsilon$, the quadrature error satisfies

$$|I - I_n| \le \frac{\varepsilon}{2\pi} \, |\Gamma| \, \|f\|_\Gamma.$$

This shows that the accuracy of the quadrature formula is determined entirely by the quality of the rational approximation to the Cauchy transform on Γ. Rational approximation therefore provides a natural and powerful framework for constructing quadrature rules, with the geometry and analytic structure of the integrand encoded in the poles and residues of the approximating rational function.
We now introduce a second rational function that plays a role in the quadrature construction. Whereas r_{nn} interpolates the Cauchy transform $C(s)$ at the points $s_0, \ldots, s_n$, the new rational function will interpolate the integrand f at the quadrature nodes. These two functions form a dual pair that together govern the exactness of the quadrature rule.

The quadrature interpolant $q_{n+1}(z)$. This is a rational function of degree at most $n+1$ whose poles occur at the points $s_0, \ldots, s_n$ that are finite. (A pole may be absent if the associated residue vanishes.) The function is required to satisfy the interpolation conditions

$$q_{n+1}(z_k) = f(z_k), \qquad k = 1, \ldots, n,$$

where $z_1, \ldots, z_n$ are the quadrature nodes, i.e., the poles of r_{nn}.
The behavior of q_{n+1} at infinity depends on how many of the interpolation points s_k equal ∞:

- If all s_k are finite, then q_{n+1} is uniquely determined by also requiring a *double zero at infinity*; that is,
$$q_{n+1}(z) = O(z^{-2}) \qquad (z \to \infty).$$
- If exactly one of the s_k is ∞, then q_{n+1} has degree at most n and must vanish to order 1 at ∞.

- If $\nu \geq 2$ of the s_k are ∞, then q_{n+1} has degree at most $n+1-\nu$ and has a pole of order at most $\nu - 2$ at ∞.

The two rational functions r_{nn} and q_{n+1} exhibit a striking duality. Apart from subtleties at infinity, the poles of r_{nn} are the interpolation points of q_{n+1}, since $q_{n+1}(z_k) = f(z_k)$, and the poles of q_{n+1} are the interpolation points of r_{nn}, since $r_{nn}(s_k) = C(s_k)$. Each rational function encodes the analytic information supplied by the other.

Theorem 9.1. *Let the quadrature formula*

$$I_n = \sum_{k=1}^{n} c_k \, f(z_k) \tag{9.11}$$

be defined by nodes $z_1, \ldots, z_n$, weights $c_1, \ldots, c_n$, and interpolation of the Cauchy transform $C(s)$ at any $n+1$ points $s_0, \ldots, s_n$ as described above. Then this quadrature rule integrates exactly every rational function q_{n+1} of the class defined above; that is,

$$I_n = \int_{\gamma} q_{n+1}(z) \, w(z) \, dz. \tag{9.12}$$

Proof. Let q_{n+1} be any rational function in the stated class, and let Γ be a contour enclosing γ but excluding the points $s_0, \ldots, s_n$. From the representations of I and I_n in Section 2, we obtain

$$I_n - I = \frac{1}{2\pi i} \int_{\Gamma} q_{n+1}(s) \, [r_{nn}(s) - 2\pi i \, C(s)] \, ds. \tag{9.13}$$

The only possible singularities of the integrand occur at the poles of $q_{n+1}(s)$, namely $s_0, \ldots, s_n$, but these are precisely the points at which $r_{nn}(s) - 2\pi i \, C(s)$ vanishes. Thus each singularity of q_{n+1} is canceled by a zero of the difference, and the integrand is analytic on and exterior to Γ.
We may therefore deform the contour outward to a large circle. When all s_k are finite, $q_{n+1}(s) = O(|s|^{-2})$ as $s \to \infty$ by the imposed double zero at infinity, and the integral over the large circle tends to (see Section 11.5). If one of the s_k equals ∞, then $q_{n+1}(s) = O(|s|^{-1})$, while the difference $r_{nn}(s) - 2\pi i \, C(s)$ contributes another factor $O(|s|^{-1})$, since it also vanishes at infinity. Thus their product again decays like $O(|s|^{-2})$. If $\nu \geq 2$ of the points s_k are infinite, then $r_{nn}(s) - 2\pi i \, C(s) = O(|s|^{-\nu})$ and $q_{n+1}(s) = O(|s|^{\nu-2})$, so the product still decays as $O(|s|^{-2})$. In all cases the integral in (9.13) vanishes, proving that $I_n = I$. □

This exactness result reduces to the classical theory of polynomial interpolatory quadrature when all poles of q_{n+1} lie at ∞, in which case q_{n+1} is a polynomial of degree $n-1$. The Cauchy interpolant r_{nn} then reduces to the unique degree-n Padé or near-Padé approximant of $C(s)$ at $s = \infty$.
We now illustrate how the rational–approximation framework produces quadrature formulas in practice.

9.2 Application 1: Gauss–Legendre quadrature.

For the integral

$$I = \int_{-1}^{1} f(z)\, dz,$$

with weight $w \equiv 1$, whose Cauchy transform is

$$C(s) = \frac{1}{2\pi i} \int_{-1}^{1} \frac{1}{s - z}\, dz = \frac{1}{2\pi i} \log\left(\frac{s+1}{s-1}\right).$$

As we have seen in Chapter 5, the standard way to talk about Gauss quadrature is via orthogonal polynomials, but Gauss's original way of doing it is through a Padé approximation of a logarithmic function. In particular, he considered the function $\log((z+1)/(z-1))$ and its approximation at the point $z = \infty$. Of course, by introducing the change of variables $z = 1/\xi$, one obtains an equivalent problem at $\xi = 0$.
Gauss quadrature may thus be viewed as arising from a rational approximation of $2\pi i\, C(s)$ by a type (n, n) Padé approximant at $s = \infty$ (see Chapter 10); the poles of the approximant become the quadrature nodes, and the residues give the corresponding weights.
As an example, consider

$$f(z) = \frac{1}{1 + 25z^2},$$

analytic except at $z = \pm i/\sqrt{25}$. We approximate $2\pi i\, C(s)$ on a Bernstein ellipse Γ with parameter $\rho = \frac{1}{\sqrt{25}} + \sqrt{\frac{26}{25}} \approx 1.2198$, using the AAA algorithm [5], then the poles z_k and residues c_k of the approximant then give the quadrature formula

$$I_n = \sum_{k=1}^{n} c_k\, f(z_k).$$

The Fig. (9.2) illustrates AAA quadrature based on rational approximation of the Cauchy transform associated with integration over $[-1, 1]$. The upper-left panel shows the Bernstein ellipse together with the poles of the degree-20 AAA approximant, which cluster near the real axis and closely resemble Gauss–Legendre quadrature nodes. The lower-left panel displays the phase portrait of the rational approximant $r_{20}(z)$, highlighting rapid phase variation near the interval. The right panel shows the convergence of the quadrature error $|I_n - I|$ as a function of the degree n, demonstrating exponential convergence comparable to Gauss–Legendre quadrature for this analytic integrand.
However, the assumption that f is analytic in $\mathcal{E}_\rho$ is unbalanced from a practical point of view. It allows f to be 'less analytic" near the ends of the interval, where the ellipse is narrow, than in the middle, where it is wide. This nonuniform analyticity condition lacks intrinsic justification. Rational approximation mitigates this nonuniformity by permitting contours better adapted to the geometry of the problem. In particular, replacing the Bernstein ellipse by a stadium-shaped contour that maintains nearly constant distance

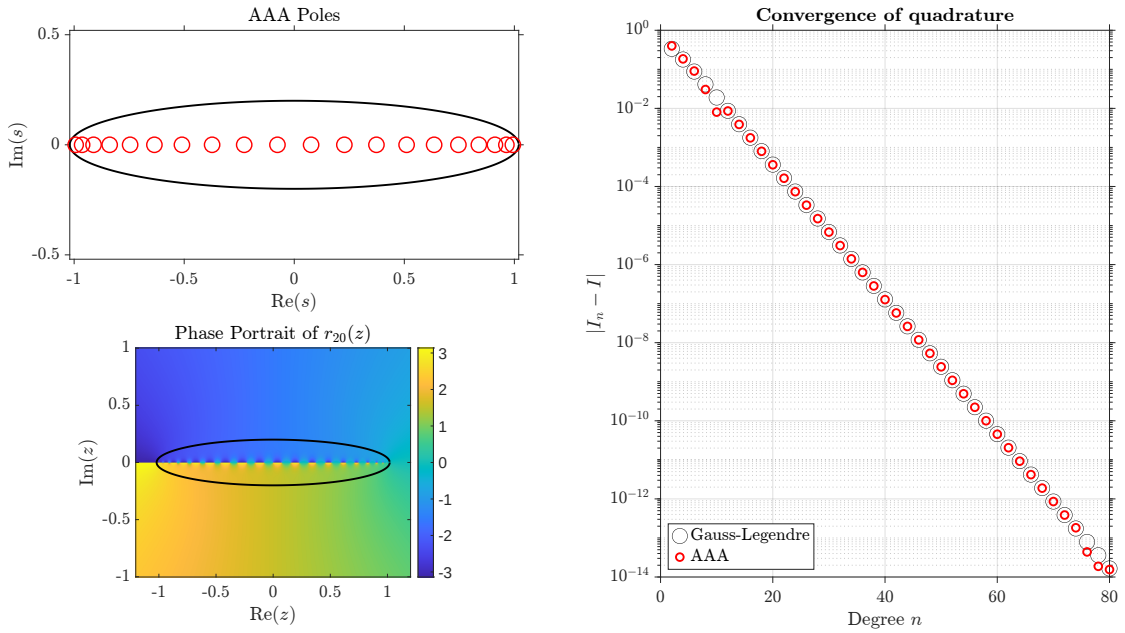


Figure 9.2: AAA quadrature based on rational approximation of $2\pi i\, C(s) = \log\big((s+1)/(s-1)\big)$ on a Bernstein ellipse. Left: AAA poles for degree $n = 20$ and the phase portrait of the approximant. Right: convergence of the quadrature error as a function of n.

from $[-1, 1]$ leads to a more uniform analyticity requirement. The stadium-shaped contour is defined as follows.

$$\Gamma = \{x - i\varepsilon \mid x \in [-1, 1]\} \cup \{1 + \varepsilon e^{i\theta} \mid \theta \in [-49\pi/100,\, 49\pi/100]\}$$

$$\cup\, \{-x + i\varepsilon \mid x \in [-1, 1]\} \cup \{-1 - \varepsilon e^{i\theta} \mid \theta \in [-49\pi/100,\, 49\pi/100]\},$$

with $\varepsilon = \frac{1}{\sqrt{25}}$.
Repeating the above construction with such a contour, we find that the resulting AAA-based quadrature converges faster than Gauss–Legendre quadrature by a factor of approximately $\pi/2$ for the same test integrand; see Fig. (9.3).

9.3 Application 2: Nearby singularities.

Suppose f is analytic on $[-1, 1]$ but has singularities close to the real axis. Classical Gaussian quadrature may then converge slowly: if a singularity lies at distance ε, one typically needs $O(\varepsilon^{-1})$ nodes for high accuracy.
Rational approximation offers a natural remedy. Consider

$$f(z) = \frac{1}{1 + 100z^2},$$

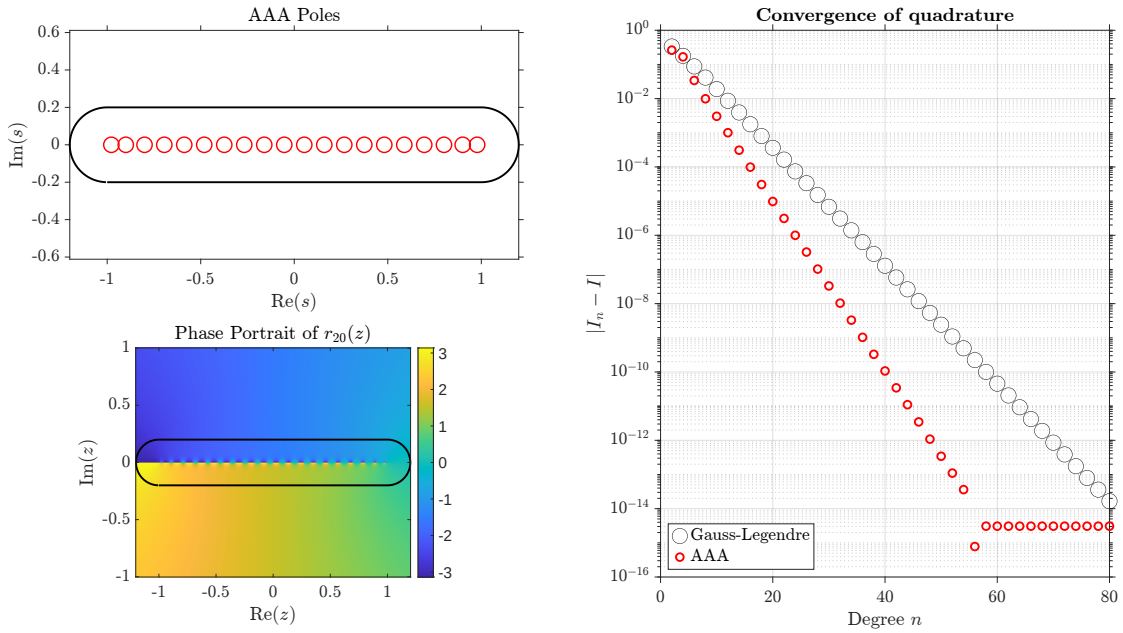


Figure 9.3: AAA quadrature using a stadium-shaped contour enclosing $[-1, 1]$. Compared with the Bernstein ellipse, the poles are more uniformly distributed and the convergence is faster for the same test integrand.

which has poles at $\pm 0.1i$. To reflect these nearby singularities, we modify the approximation contour by augmenting the Bernstein ellipse with short vertical slits near the poles and approximate $2\pi i\, C(s)$ using the AAA algorithm on this contour. The resulting poles z_k and residues c_k define a quadrature rule

$$I_n = \sum_{k=1}^{n} c_k\, f(z_k).$$

Fig. (9.4) illustrates this construction. The left panels show the modified contour together with the poles and phase portrait of the degree-20 AAA approximant, while the right panel compares the convergence of the resulting quadrature rule with Gauss–Legendre quadrature. For this example, the AAA-based rule achieves comparable accuracy with roughly a factor of five fewer degrees of freedom, with the advantage increasing as the singularities approach the real axis.

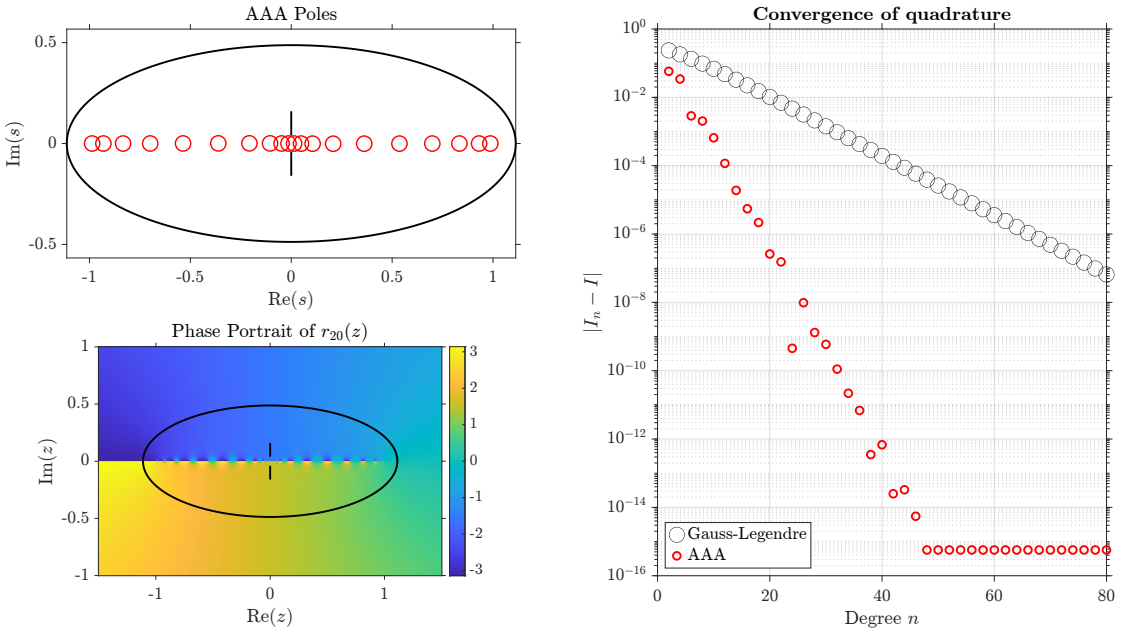


Figure 9.4: AAA quadrature for an integrand with nearby singularities. Left: poles and phase portrait of the AAA rational approximant on a Bernstein ellipse augmented with short slits near the singularities at $\pm 0.1i$. Right: convergence of the quadrature error compared with Gauss–Legendre quadrature.

10 Padé Approximation

In this chapter, we move from rational interpolation at multiple points to rational interpolation at a single point—this is exactly what Padé approximation is. It is a limiting case in which the interpolation points degenerate to a single point. Instead of matching the function at several points, we match the function and its derivatives at one point, usually $x = 0$ or $x = x_0$.
We will work in the complex plane. Padé approximation can be viewed as the rational generalization of Taylor approximation.
Suppose f is a function that has a Taylor series

$$f(z) = c_0 + c_1 z + c_2 z^2 + \cdots \tag{10.1}$$

expanded about $z = 0$. Whether or not this series converges has no effect on the *definition* of Padé approximation (of course, convergence becomes important once we begin using the approximant). For the purposes of developing Padé approximation, it is enough to regard f as a *formal power series*, meaning that we make no claim about convergence of the coefficients.
For any integer $m \geq 0$, the *degree-m Taylor approximant* of f is the unique polynomial $p_m \in \mathcal{P}_m$—of course just a truncation of the Taylor series—whose Taylor expansion agrees with that of f as far as possible, at least through degree m. In other words,

$$(f - p_m)(z) = O(z^{m+1}). \tag{10.2}$$

The statement above means that the first nonzero term in the Taylor series of $f - p_m$ must be z^k for some $k \geq m + 1$, although k does not have to be exactly $m + 1$.
Padé approximation is the rational generalization of this idea. For integers $m, n \geq 0$, a function $r_{mn} \in \mathcal{R}_{mn}$ is called the *type* (m, n) *Padé approximant* of f. Its Taylor expansion at the origin is again a power series, and it is chosen so that this series matches the Taylor series of f to the greatest possible order. Symbolically,

$$(f - r_{mn})(z) = O\Big(z^{\text{maximum}}\Big). \tag{10.3}$$

Here we cannot say *a priori* what power of z will be involved. However, generically we expect the error to be at least of order $m+n+1$, since we now have more free parameters than in the Taylor case.
As with rational interpolation, it is often convenient to work with the linearized form of the problem:

$$(fq - p)(z) = O\Big(z^{m+n+1}\Big). \tag{10.4}$$

By itself this condition is vacuous, since matching to all orders could be achieved by taking p and q identically zero. The condition becomes meaningful once we impose the requirement $q \neq 0$. With this assumption, it is known that the matching condition can always be satisfied through degree $m+n$ or higher:

$$p(z) = f(z)q(z) + O\Big(z^{m+n+1}\Big), \tag{10.5}$$

as we shall confirm.
Next, we show an example of what people call the Padé table for e^z.

	$m=0$	$m=1$	$m=2$
$n=0$	1	$1+z$	$1+z+\frac{1}{2}z^2$
$n=1$	$\dfrac{1}{1-z}$	$\dfrac{1+\frac{1}{2}z}{1-\frac{1}{2}z}$	$\dfrac{1+\frac{2}{3}z+\frac{1}{6}z^2}{1-\frac{1}{3}z}$
$n=2$	$\dfrac{1}{1-z+\frac{1}{2}z^2}$	$\dfrac{1+\frac{1}{3}z}{1-\frac{2}{3}z+\frac{1}{6}z^2}$	$\dfrac{1+\frac{1}{2}z+\frac{1}{12}z^2}{1-\frac{1}{2}z+\frac{1}{12}z^2}$

Table 10.1: Padé table for e^z

In the first row, $n=0$, that is, the polynomial case. The off-diagonal elements are obtained by flipping across the diagonal and adjusting the signs appropriately. In principle, the table goes on forever.
The first theorem is an approximation theorem that we have already seen in the context of polynomial and rational approximation. There exists a unique Padé approximant r_{mn} to f, characterized not by equioscillation, but by the property that the error satisfies

$$(f - r_{mn})(z) = O\Big(z^{\,m+n+1-d}\Big). \tag{10.6}$$

Theorem 10.1 (Characterization of Padé approximants)**.** *For each $m, n \geq 0$, a function f has a unique Padé approximant $r_{mn} \in \mathcal{R}_{mn}$ as defined by the Padé condition* (10.3)*. Moreover, a function $r \in \mathcal{R}_{mn}$ is equal to r_{mn} if and only if*

$$(f - r)(z) = O\Big(z^{\,m+n+1-d}\Big),$$

where d is the defect of r in $\mathcal{R}_{mn}$.

Proof. We first show that

$$(f - r)(z) = O\Big(z^{\,m+n+1-d}\Big) \implies (f - \tilde{r})(z) = O\Big(z^{\text{maximum}}\Big).$$

Thus the Padé condition plays the role of an "equioscillation" principle: it implies optimality, and moreover such a function is unique.
Suppose

$$(f - \tilde{r})(z) = O\Big(z^{\,m+n+1-d}\Big), \qquad \tilde{r} \in \mathcal{R}_{mn}.$$

Then

$$r - \tilde{r} = \text{type } (m-d,\, n-d) \;-\; \text{type } (m,n),$$

and hence

$$r - \tilde{r} = \text{type } (\, m+n-d,\, 2n-d\,).$$

Writing this difference explicitly,

$$r - \tilde{r} = \frac{p}{q} - \frac{\tilde{p}}{\tilde{q}} = \frac{p\tilde{q} - \tilde{p}q}{q\tilde{q}}.$$

Denote by d the defect of r in $\mathcal{R}_{mn}$. One finds that

$$p\tilde{q} - \tilde{p}q \quad \text{has degree at most} \quad m+n-d,$$

while

$$q\tilde{q} \quad \text{has degree} \quad 2n-d.$$

Thus,

$$r - \tilde{r} \in \mathcal{R}_{\,m+n-d,\,2n-d},$$

and consequently

$$r - \tilde{r} \quad \text{has at most} \;\; m+n-d \text{ zeros.}$$

If, in addition,

$$(r-\tilde{r})(z) = O\Big(z^{\,m+n+1-d}\Big)\,,$$

then $r - \tilde{r}$ must vanish identically. This proves uniqueness.
The remaining part of the proof is to show the existence of a function $r \in \mathcal{R}_{mn}$ such that

$$(f-r)(z) = O\Big(z^{\,m+n+1-d}\Big)\,.$$

Suppose we seek a representation of the Padé approximant r_{mn} as a quotient $r = p/q$, where $p \in \mathcal{P}_m$ and $q \in \mathcal{P}_n$. The Padé condition (10.3) is nonlinear, but multiplying through by the denominator leads to the linearized condition

$$p(z) = f(z)\,q(z) + O\Big(z^{m+n+1}\Big)\,. \tag{10.7}$$

Here

$$p(z) = \sum_{k=0}^{m} a_k z^k, \qquad q(z) = \sum_{k=0}^{n} b_k z^k,$$

and the function f itself is expanded as

$$f(z) = \sum_{j=0}^{\infty} c_j z^j.$$

We now write the linearized Padé problem in matrix form. The resulting matrix has a Toeplitz structure, meaning that its entries are constant along diagonals. There are two cases.

First, for $m \geq n$, the system takes the form

$$\left(\begin{array}{c} a_0 \\ a_1 \\ \vdots \\ a_n \\ \vdots \\ a_m \\ \hline 0 \\ \vdots \\ 0 \end{array}\right) = \left(\begin{array}{cccc} c_0 & & & \\ c_1 & c_0 & & \\ \vdots & \vdots & \ddots & \\ c_n & c_{n-1} & \cdots & c_0 \\ \vdots & \vdots & & \vdots \\ c_m & c_{m-1} & \cdots & c_{m-n} \\ \hline c_{m+1} & c_m & \cdots & c_{m+1-n} \\ \vdots & \vdots & & \vdots \\ c_{m+n} & c_{m+n-1} & \cdots & c_m \end{array}\right) \begin{pmatrix} b_0 \\ b_1 \\ \vdots \\ b_n \end{pmatrix}. \tag{10.8}$$

Above the horizontal line we determine the coefficients $a_0, \ldots, a_m$. Below the line we obtain a homogeneous system involving only the coefficients $\mathbf{b}$. The lower block is an $n \times (n+1)$ matrix and therefore must have a nontrivial null vector.
For $m \leq n$, an analogous Toeplitz system arises:

$$\left(\begin{array}{c} a_0 \\ a_1 \\ \vdots \\ a_m \\ \hline 0 \\ \vdots \\ 0 \end{array}\right) = \left(\begin{array}{cccc} c_0 & & & \\ c_1 & c_0 & & \ddots \\ \vdots & \vdots & \ddots & \\ c_m & c_{m-1} & \cdots & c_0 \\ \hline c_{m+1} & c_m & \cdots & c_1 \\ \vdots & \vdots & & \vdots \\ c_{m+n} & c_{m+n-1} & \cdots & c_m \end{array}\right) \begin{pmatrix} b_0 \\ b_1 \\ \vdots \\ b_n \end{pmatrix}. \tag{10.9}$$

In other words for both systems of equations (10.8) and (10.9) , $\mathbf{b}$ must be a (right) null vector of the $n \times (n+1)$ matrix displayed below the horizontal line. The coefficients $a_0, \cdots, a_m$ of p are then obtained by multiplying out the matrix-vector product above the line. For simplicity, we treat both cases uniformly by introducing the $n \times (n+1)$ matrix

$$C = \begin{pmatrix} c_{m+1} & c_m & \cdots & c_{m+1-n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{m+n} & c_{m+n-1} & \cdots & c_m \end{pmatrix}, \tag{10.10}$$

with the convention that $c_k = 0$ for $k < 0$.
One solution would be $a = 0$ and $b = 0$, corresponding to the useless candidate $r = 0/0$. However, an $n \times (n+1)$ matrix always has a nonzero null vector,

$$Cb = 0, \qquad b \neq 0,$$

and once b is chosen, the coefficients $a_0, \ldots, a_m$ of p can be obtained by multiplying out the matrix–vector product above the line. Thus there is always a solution to (10.7) with

$q \neq 0$. Suppose b is a nonzero null vector satisfying $Cb = 0$. If $b_0 \neq 0$, then dividing equation (10.7) by q shows immediately that p/q is a solution of (10.6). However, some nonzero null vectors may begin with one or more zero components. Assume that

$$b_0 = b_1 = \cdots = b_{\sigma-1} = 0, \qquad b_\sigma \neq 0,$$

for some $\sigma \geq 1$. Then by the Toeplitz structure of (10.8), the corresponding coefficients of a satisfy

$$a_0 = a_1 = \cdots = a_{\sigma-1} = 0,$$

while a_σ may or may not be zero.
Because both a and b begin with σ zeros, we may shift each vector upward by σ positions to obtain

$$\tilde{a} = (a_\sigma, a_{\sigma+1}, \ldots, a_m, 0, \ldots, 0)^T, \qquad \tilde{b} = (b_\sigma, b_{\sigma+1}, \ldots, b_n, 0, \ldots, 0)^T.$$

The Toeplitz nature of (10.8) ensures that $(\tilde{a}, \tilde{b})$ still satisfies equations (10.8), and the quotient remains unchanged:

$$r = \frac{p}{q} = \frac{\tilde{p}}{\tilde{q}}.$$

This shift shows that r has defect $d \geq \sigma$. Equation (10.8) remains valid except possibly in the highest σ rows, where

$$\tilde{a}_{m+n-\sigma+1}, \ldots, \tilde{a}_{m+n}$$

may no longer vanish. Thus we obtain

$$(f - r)(z) = O\Big(z^{\,m+n+1-\sigma}\Big).$$

Since $d \geq \sigma$, it follows that

$$(f - r)(z) = O\Big(z^{\,m+n+1-d}\Big).$$

□

As an example, consider the type $(1, 1)$ Padé approximant of e^z,

$$r_{11}(z) = \frac{1 + \frac{1}{2}z}{1 - \frac{1}{2}z}.$$

Its defect is $d = 0$, and

$$r_{11}(z) - e^z = \frac{1}{12}z^3 + \frac{1}{12}z^4 + \cdots = O(z^3).$$

Since $m + n + 1 - d = 3$, this confirms that r_{11} is indeed the Padé approximant.

Padé approximation, viewed as a rational approximation, is an ill-posed problem. The underlying reason is closely related to analytic continuation[1]: the aim is typically to gain information about a function in a region of the complex plane using information at a single point. Starting from (10.8), (10.9), and (10.10), we show how this ill-posedness can be addressed through SVD-based regularization.
Let $\tilde{C}$ denote the square $n \times n$ matrix obtained by deleting the first column of C:

$$\tilde{C} = \begin{pmatrix} c_m & \cdots & c_{m+1-n} \\ \vdots & \ddots & \vdots \\ c_{m+n-1} & \cdots & c_m \end{pmatrix}. \tag{10.11}$$

We shall make use of the SVD of C, a factorization

$$C = U\Sigma V^*, \tag{10.12}$$

where U is $n \times n$ and unitary, V is $(n+1) \times (n+1)$ and unitary, and Σ is an $n \times (n+1)$ real diagonal matrix with diagonal entries $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$.
Suppose $\sigma_n > 0$. Then C has rank n and the final column of V provides a unique nonzero null vector $\mathbf{b}$ of C up to a scale factor. This null vector defines the coefficients of q, with two subcases of special interest. If the square supmatrix $\tilde{C}$ is singular, then necessarily $b_0 = 0$. Indeed, write $\tilde{C} = [\, c^{(0)} \mid C \,]$, where $c^{(0)}$ denotes the first column of $\tilde{C}$. The equation $\tilde{C}b = 0$ can then be written as

$$b_0 c^{(0)} + C(b_1, \ldots, b_n)^T = 0.$$

If $b_0 \neq 0$, this implies that

$$c^{(0)} = -\frac{1}{b_0} C(b_1, \ldots, b_n)^T,$$

so that the first column of $\tilde{C}$ lies in the column space of C. Consequently, removing this column does not reduce the rank, and hence $\operatorname{rank}(\tilde{C}) = \operatorname{rank}(C)$. If $\tilde{C}$ is singular, then $\operatorname{rank}(\tilde{C}) \leq n-1$, which would imply $\operatorname{rank}(C) \leq n-1$, contradicting the assumption $\operatorname{rank}(\tilde{C}) = n$. From Eq. (10.8) or Eq. (10.9) we see that this also implies $a_0 = 0$. Thus p and q share a common factor z, or possibly z^λ for some $\lambda > 1$, and this factor can be divided out at the end. If $\tilde{C}$ is nonsingular, then b_0 must be nonzero, and the defect is $\sigma = 0$. Indeed, if $b_0 = 0$, then $Cb = 0$ reduces to $\tilde{C}(b_1, \ldots, b_n)^T = 0$, which by nonsingularity of $\tilde{C}$ implies $b = 0$, a contradiction. Hence $b_0 \neq 0$, so the denominator has no vanishing constant term and no common factor occurs. In this case the defect is $\sigma = 0$.
On the other hand, suppose $\sigma_n = 0$. Then the Toeplitz matrix C has rank $\rho < n$, with $\sigma_{\rho+1} = \cdots = \sigma_n = 0$. Since $\tilde{C}$ is obtained by deleting a single column from C, thus C must have rank ρ or $\rho - 1$; in particular, $\tilde{C}$ is singular.

[1] If f and g are analytic functions in Ω which are equal in an arbitrarily small disc in Ω, then

$$f = g \quad \text{everywhere in } \Omega.$$

Because C has rank ρ, any set of $\rho+1$ columns must be linearly dependent. In particular, the supmatrix of C consisting of its last $\rho+1$ columns is rank-deficient. This linear dependence yields coefficients, not all zero, whose linear combination of these columns vanishes, and therefore produces a nonzero null vector b of C satisfying

$$b_0 = b_1 = \cdots = b_{n-\rho-1} = 0.$$

It follows that the denominator polynomial satisfies $q(z) = z^{n-\rho}\tilde{q}(z)$, and, by the Toeplitz structure of the equations above the line, the same factor appears in the numerator. Thus the corresponding rational function has defect at least $n-\rho$. Dividing out this common factor, we may reduce n to ρ and m to $m-(n-\rho)$, and restart the construction.
These considerations naturally lead to an SVD-based algorithm [8] for computing the unique Padé approximant of type (m,n) to a function f given by its Taylor series, as described below.

Algorithm 1. Pure Padé approximation in exact arithmetic.

Input: $m \geq 0$, $n \geq 0$, and Taylor coefficients $c_0, \ldots, c_{m+n}$ of a function f.
Output: Polynomials $p(z) = a_0 + \cdots + a_\mu z^\mu$ and $q(z) = b_0 + \cdots + b_\nu z^\nu$, $b_0 = 1$, of the minimal degree type (m,n) Padé approximation of f.

1. If $c_0 = \cdots = c_m = 0$, set $p = 0$ and $q = 1$ and stop.
2. If $n = 0$, set $p(z) = c_0 + \cdots + c_m z^m$ and $q = 1$ and go to step 8.
3. Compute the SVD (10.12) of the $n \times (n+1)$ matrix C. Let $\rho \leq n$ be the number of nonzero singular values.
4. If $\rho < n$, reduce n to ρ and m to $m-(n-\rho)$ and return to step 2.
5. Get q from the null right singular vector $\mathbf{b}$ of C and then p from the upper part of (10.8) or . (10.9).
6. If $b_0 = \cdots = b_{\lambda-1} = 0$ for some $\lambda \geq 1$, which implies also $a_0 = \cdots = a_{\lambda-1} = 0$, cancel the common factor of z^λ in p and q.
7. Divide p and q by b_0 to obtain a representation with $b_0 = 1$.
8. Remove trailing zero coefficients, if any, from $p(z)$ or $q(z)$.

This algorithm produces the unique Padé approximant r_{mn} in a minimal-degree representation of type (μ,ν) with $b_0 = 1$.

10.1 Examples of the elimination of Froissart doublets.

As discussed in the treatment of rational interpolation, rounding errors or other perturbations commonly introduce *Froissart doublets*. The same phenomenon occurs in Padé approximations. These spurious pole–zero pairs neither reflect genuine information about the function f nor contribute to the quality of the approximation.
We now present an example illustrating the appearance of Froissart doublets and show how Padé approximation based on the singular value decomposition (SVD) approach of [8] removes these effects by adaptively reducing the degrees m and n. In the examples

that follow, we compute Padé approximants using the routine `padeapprox`[2] [8], with tolerance $tol = 10^{-14}$, available as part of *Chebfun*, and visualize their poles in the complex plane.
Consider the function

$$f(z) = \log(1.2 - z),$$

This function has a branch cut $[1.2, \infty)$ that attracts poles in keeping with the theory of Stahl [18].

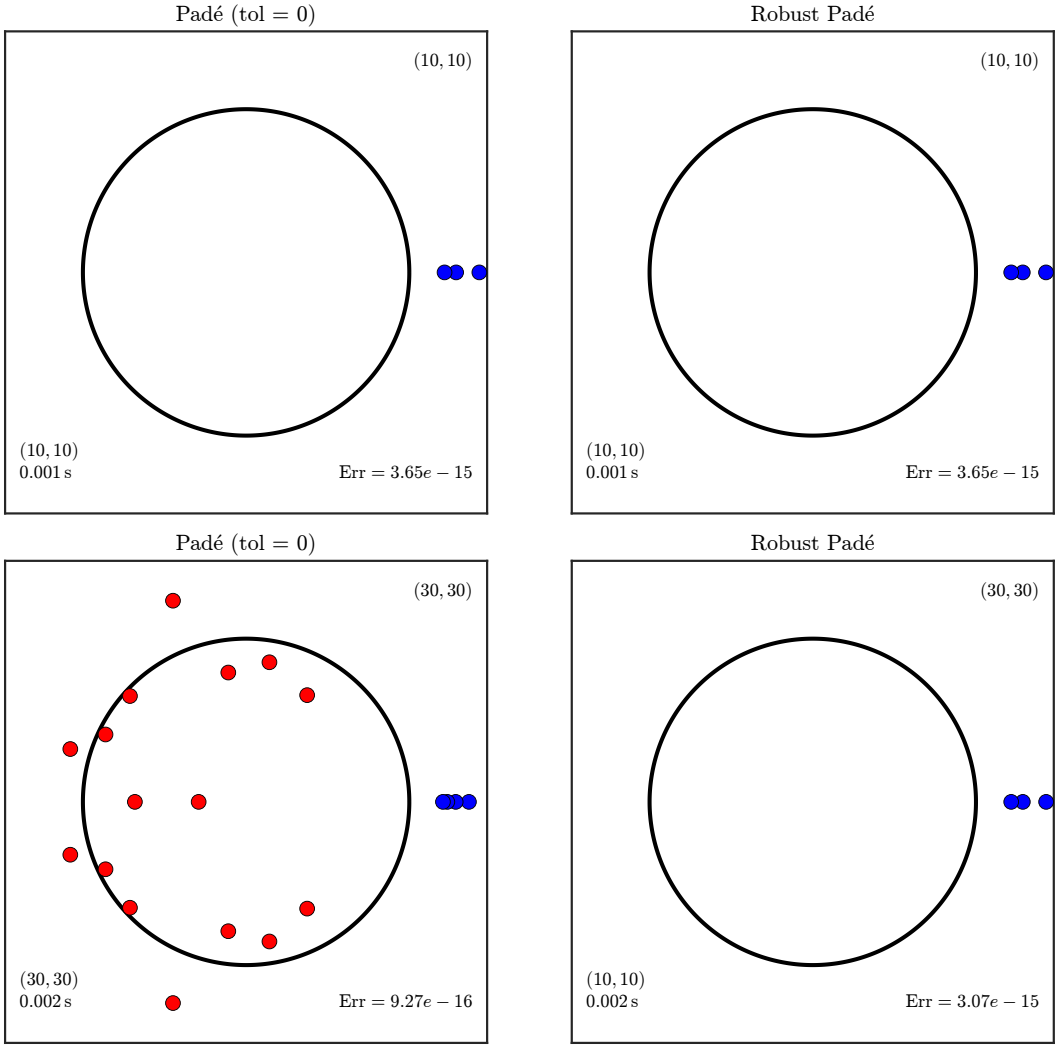
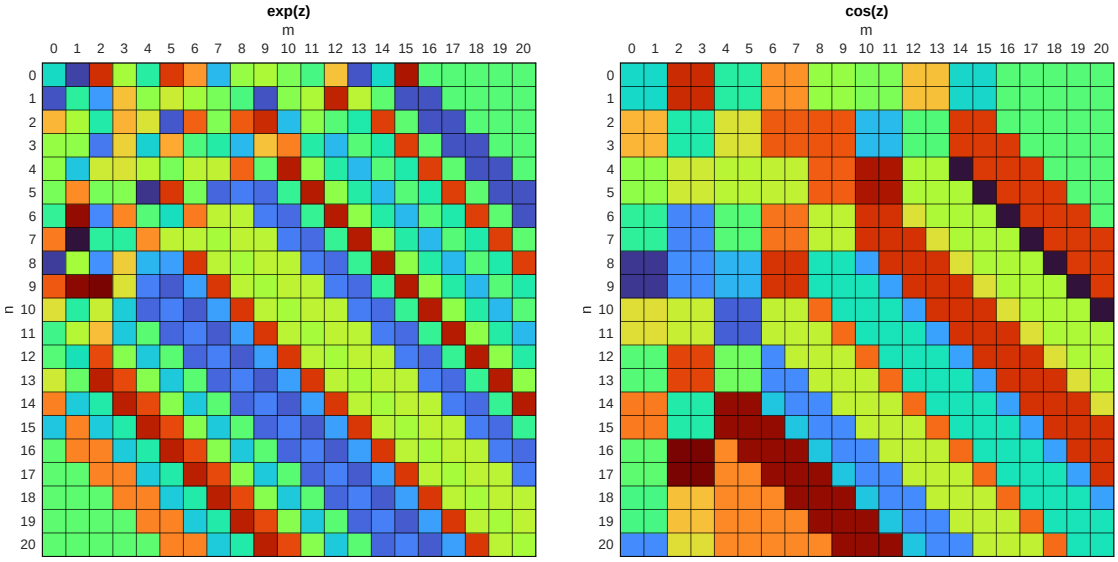


Figure 10.1: Padé approximations of $\log(1.2-z)$. Left: unregularized. Right: regularized.

Each panel in Fig. (10.1) corresponds to the approximation of f on the $(N+1)$st roots of unity by a rational function of type (m, n). The unit circle is shown, a label in the

[2]Matlab implementation of ALGORITHM 1



(a) $\exp(z)$ (b) $\cos(z)$

Figure 10.2: Padé tables computed numerically by `padeapprox` [8] with m and n on the horizontal and vertical axes, respectively. Each square (m, n) is marked by a color determined by the exact type (μ, ν) of the corresponding approximation, so that each square block appears in a single color. For $\exp(z)$, all the entries lie in 1×1 blocks until the function is resolved to machine precision, after which the numerator and denominator degrees are systematically reduced as far as possible, causing diagonal stripes. For the even function $\cos(z)$, 2×2 square blocks appear.

upper-left corner indicates the approximation strategy: *Interpolation* when $N = m + n$, and *Least-squares* when $N > m + n$; in the latter case we take $N = 8(m + n) + 1$. The lower-left corner reports the exact type (μ, ν) of the computed Padé approximant.

10.2 Examples of computed Padé tables

One appealing feature of `padeapprox` is the numerical computation of block structure in the Padé table for a given function f. For example, in Fig. (10.2) is a table of the computed pair (μ, ν) for each (m, n) in the upper-left portion of the Padé table of $\exp(z)$, $\cos(z)$ with $0 \leq m, n \leq 20$. Fig. (10.2) shows Padtables computed by `padeapprox` with $tol = 10^{-14}$ for the functions $\exp(z)$, $\cos(z)$. As described in the caption, the images clearly show the block structures for the various functions. One sees the 2×2 block structure resulting from the evenness of $\cos(z)$. Each block in Fig. (10.3) corresponds to a fixed minimal Padé type (μ, ν), reflecting the vanishing of odd Taylor coefficients.

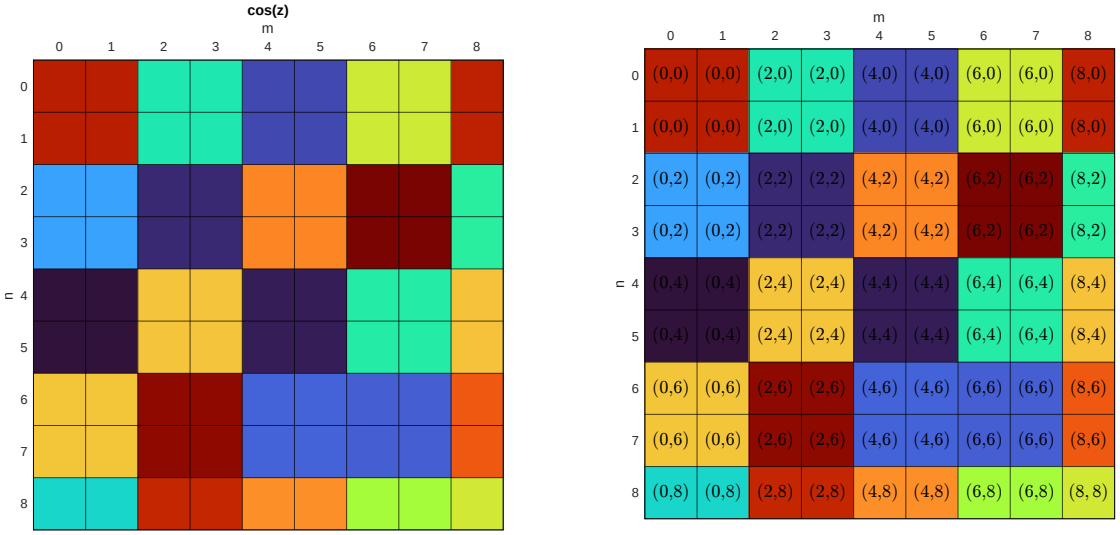


Figure 10.3: Padé table diagnostics for $f(z) = \cos z$: minimal-degree types (μ, ν) (right) and the corresponding color plot (left).

11 Appendix

11.1 Orthogonal polynomials

Now let $w \in C(-1,1)$ be a weight function such that $w(x) > 0$ for $x \in (-1,1)$ and

$$\int_{-1}^{1} w(x)\,dx < \infty. \tag{11.1}$$

We allow $w(x)$ to approach 0 or $+\infty$ as $x \to \pm 1$. Such a function defines an inner product

$$(f,g) = \int_{-1}^{1} w(x)\,\overline{f(x)}\,g(x)\,dx. \tag{11.2}$$

A family of orthogonal polynomials associated with w is a sequence

$$p_0, p_1, p_2, \ldots$$

where p_n has degree exactly n and satisfies

$$(p_j, p_k) = 0 \quad \text{for } j \neq k. \tag{11.3}$$

The Chebyshev polynomials $\{T_k\}$ are orthogonal with respect to the weight

$$w(x) = \frac{2}{\pi\sqrt{1-x^2}}. \tag{11.4}$$

A more general family is obtained by allowing power singularities at the endpoints, giving the Jacobi weight function

$$w(x) = (1-x)^{\alpha}(1+x)^{\beta}, \qquad \alpha, \beta > -1. \tag{11.5}$$

The associated orthogonal polynomials are the Jacobi polynomials $P_n^{(\alpha,\beta)}$.
The most special case is $\alpha = \beta = 0$, which gives the Legendre polynomials with the constant weight function $w(x) = 1$.
A normalized version of the first three Legendre polynomials is

$$p_0(x) = \sqrt{\tfrac{1}{2}}, \qquad p_1(x) = \sqrt{\tfrac{3}{2}}\,x, \qquad p_2(x) = \sqrt{\tfrac{45}{8}}\,x^2 - \sqrt{\tfrac{5}{8}}.$$

For the classical normalization $P_j(1) = 1$, the orthogonality relation becomes

$$\int_{-1}^{1} P_j(x)\,P_k(x)\,dx = \begin{cases} 0, & j \neq k, \\ \dfrac{2}{2k+1}, & j = k. \end{cases} \tag{11.6}$$

11.2 Chebyshev vs. Legendre polynomials

This note revisits a classical question: *Which family—Legendre or Chebyshev—is preferable for function approximation?* As we shall see, the answer depends on what notion of "best" approximation one has in mind. Both families form fundamental orthogonal bases on $[-1, 1]$, but they differ in orthogonality, oscillatory structure, and in the type of error they naturally control.

A key distinction lies in their oscillations. Chebyshev polynomials oscillate between ± 1 with *constant amplitude*, with extremal points

$$x_j = \cos\left(\frac{j\pi}{n}\right), \qquad j = 0, 1, \ldots, n.$$

which cluster near the endpoints. This leads to highly balanced oscillations and a nearly uniform distribution of approximation error. Legendre polynomials, by contrast, have more evenly spaced extrema but their amplitudes increase near $x = \pm 1$, often producing larger endpoint errors even when the global L^2 error remains small.

The Chebyshev weight $(1 - x^2)^{-1/2}$ precisely compensates for these endpoint effects by assigning greater importance to accuracy near the boundaries.

These structural differences yield complementary notions of optimality:

- Legendre polynomials optimize the *average error*, i.e. the unweighted L^2 error.
- Chebyshev polynomials excel at controlling the *pointwise maximum error*, due to their uniform oscillatory behavior.

Which family is "best" thus depends on the context and on the error metric one wishes to optimize. The richness of their complementary properties is part of what makes classical orthogonal polynomials such enduring tools of modern mathematics.

11.3 Expansion of $\widetilde{r}$ in the proof of optimality

In this appendix we derive the expansion (6.4), (6.5), namely:

$$\widetilde{r} = \frac{(p + \delta p)(q - \delta q)}{q^2} + O(\|\delta q\|^2), \qquad \widetilde{r} - r = \frac{q\,\delta p - p\,\delta q}{q^2} + O(\|\delta p\|\,\|\delta q\| + \|\delta q\|^2),$$

used in the proof of optimality $\Rightarrow$ equisoscillation.

Derivation of the first expansion

We start from

$$\widetilde{r} = \frac{p + \delta p}{q + \delta q}.$$

Factor out q from the denominator:

$$\widetilde{r} = \frac{p + \delta p}{q(1 + \delta q/q)} = \frac{p + \delta p}{q} \cdot \frac{1}{1 + \delta q/q}.$$

Using the Taylor expansion

$$\frac{1}{1+z} = 1 - z + O(z^2) \qquad (z \to 0),$$

with $z = \delta q/q$, we obtain

$$\frac{1}{1+\delta q/q} = 1 - \frac{\delta q}{q} + O(\|\delta q\|^2),$$

where we have used that q is fixed and bounded away from zero on $[-1, 1]$. Thus

$$\widetilde{r} = \frac{p+\delta p}{q}\left(1 - \frac{\delta q}{q} + O(\|\delta q\|^2)\right),$$

and distributing gives

$$\widetilde{r} = \frac{p+\delta p}{q} - \frac{(p+\delta p)\,\delta q}{q^2} + O(\|\delta q\|^2).$$

Combining the first two terms over the common denominator q^2 yields

$$\frac{(p+\delta p)q}{q^2} - \frac{(p+\delta p)\,\delta q}{q^2} = \frac{(p+\delta p)(q-\delta q)}{q^2}.$$

Hence the first expansion:

$$\boxed{\widetilde{r} = \frac{(p+\delta p)(q-\delta q)}{q^2} + O(\|\delta q\|^2)}.$$

Derivation of the expansion for $\widetilde{r} - r$

Since $r = p/q = pq/q^2$, subtracting yields

$$\widetilde{r} - r = \frac{(p+\delta p)(q-\delta q) - pq}{q^2} + O(\|\delta q\|^2).$$

Expanding the numerator:

$$(p+\delta p)(q-\delta q) = pq - p\,\delta q + q\,\delta p - \delta p\,\delta q.$$

Thus

$$(p+\delta p)(q-\delta q) - pq = q\,\delta p - p\,\delta q - \delta p\,\delta q.$$

Therefore

$$\widetilde{r} - r = \frac{q\,\delta p - p\,\delta q}{q^2} - \frac{\delta p\,\delta q}{q^2} + O(\|\delta q\|^2).$$

Since

$$\|\delta p\,\delta q\| \le \|\delta p\|\,\|\delta q\|,$$

the term $\delta p\,\delta q/q^2$ is of order $O(\|\delta p\|\,\|\delta q\|)$, and we conclude that

$$\boxed{\widetilde{r} - r = \frac{q\,\delta p - p\,\delta q}{q^2} + O(\|\delta p\|\,\|\delta q\| + \|\delta q\|^2)}.$$

□

11.4 Branch points and Branch cuts.

Consider the complex–valued function

$$\log(z) = \ln(r) + i\theta, \tag{11.7}$$

where $z = re^{i\theta}$ with $r > 0$ and θ real. As one goes around the closed path in Figure 1.1, starting counterclockwise from point A and returning to A, it is clear that θ_0 increases to $\theta_0 + 2\pi$. Therefore, upon tracing the path, we have

$$\log(A) \longrightarrow \log(A) + 2\pi i. \tag{11.8}$$

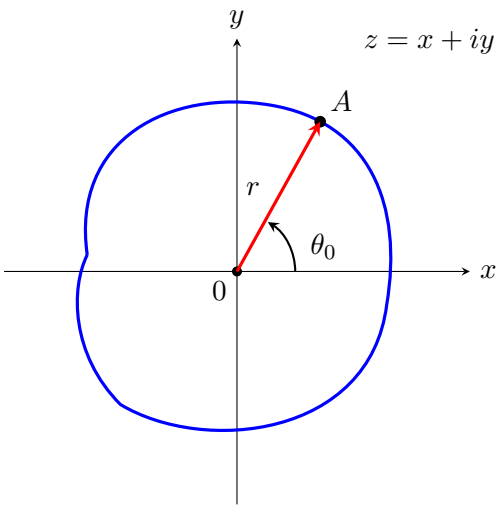


Figure 11.1: A point A on a closed curve and its polar coordinates (r, θ_0) with respect to the origin.

This means that $\log(z)$ **does not return** to its original value when one tries to define it continuously along the closed path. From Fig. (11.1), we see that the path encloses the origin $z = 0$. This is why θ increases by 2π as one goes around the path. Thus, the origin is a **branch point**of $\log(z)$.

Definition 11.1. *The point z_0 is called a **branch point** for the complex (multiple) valued function $f(z)$ if the value of $f(z)$ does not return to its initial value when a closed curve around z_0 is traced (starting from some arbitrary point on the curve), in such a way that f varies continuously along the path.*

A branch cut is any curve that joins the branch points, with the objective of preventing paths from looping around any of them. For $\log(z)$, a branch cut is therefore simply any curve connecting its two branch points. The purpose of the cut is to exclude curves that encircle a branch point, since $\log(z)$ can be defined uniquely only if one avoids going around either $z = 0$ or $z = \infty$ (using $\frac{1}{\zeta} = z$), both of which are branch points.

Example 11.2. In this example we determine the branch points of

$$\log\left(\frac{z-1}{z+1}\right)$$

and draw a possible branch cut for this function. We begin by observing that

$$\log\left(\frac{z-1}{z+1}\right) = \log(z-1) \; - \; \log(z+1). \tag{2.4}$$

The function $\log(z-1)$ has a branch point at $z=1$, and the function $\log(z+1)$ has a branch point at $z=-1$. Therefore $z=\pm 1$ are the branch points of

$$\log\left(\frac{z-1}{z+1}\right)$$

for all finite z.
To determine whether $z=\infty$ is also a branch point, we substitute $z=\frac{1}{\zeta}$ and rewrite

$$\log\left(\frac{z-1}{z+1}\right) = \log\left(\frac{1-\zeta}{1+\zeta}\right). \tag{2.5}$$

The expression on the right has branch points only at $\zeta=\pm 1$, and not at $\zeta=0$. Since $\zeta=0$ corresponds to $z=\infty$, we conclude that $z=\infty$ is not a branch point.
Thus the only branch points of

$$\log\left(\frac{z-1}{z+1}\right)$$

are $z=\pm 1$. The point $z=\infty$ is not a branch point because the function remains finite and single-valued there.

The branch cuts must be drawn to prevent curves from going around the two branch points $z=\pm 1$. For example, we may take the branch cut to be the straight line segment joining the two branch points, as in Fig. (11.2).

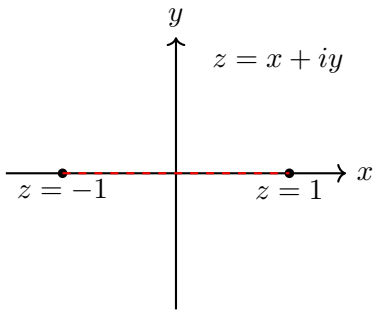


Figure 11.2: A branch cut for $\log\left(\frac{z-1}{z+1}\right)$.

11.5 Cauchy Integral Formula

Let f be analytic on and inside a positively oriented simple closed contour Γ, and let z be any point inside Γ. Then the Cauchy integral formula [19] states that

$$f(z) = \frac{1}{2\pi i} \int_\Gamma \frac{f(\zeta)}{\zeta - z}\, d\zeta.$$

If f is analytic in an open set $\Omega \subset \mathbb{C}$, then f possesses derivatives of all orders in Ω. Moreover, if $\Gamma \subset \Omega$ is a circle whose interior is also contained in Ω, then for any $n \geq 0$,

$$f^{(n)}(z) = \frac{n!}{2\pi i} \int_\Gamma \frac{f(\zeta)}{(\zeta - z)^{n+1}}\, d\zeta.$$

Corollary 11.3 (Cauchy inequalities)**.** *Let f be analytic in an open set containing the closure of a disc Ω centered at z_0 and of radius R. Then, for all $n \geq 0$,*

$$|f^{(n)}(z_0)| \leq \frac{n!}{R^n}\, \|f\|_\Gamma,$$

where

$$\|f\|_\Gamma = \sup_{z \in \Gamma} |f(z)|$$

denotes the supremum of $|f|$ on the boundary circle $\Gamma = \partial\Omega$.

Proof. Applying the Cauchy integral formula for $f^{(n)}(z_0)$, we obtain

$$\begin{aligned}
\left|f^{(n)}(z_0)\right| &= \left| \frac{n!}{2\pi i} \int_\Gamma \frac{f(\zeta)}{(\zeta - z_0)^{n+1}}\, d\zeta \right| \\
&= \left| \frac{n!}{2\pi} \int_0^{2\pi} \frac{f(z_0 + Re^{i\theta})}{(Re^{i\theta})^{n+1}}\, Rie^{i\theta}\, d\theta \right| \\
&\leq \frac{n!}{2\pi} \frac{\|f\|_\Gamma}{R^n}\, 2\pi.
\end{aligned}$$

□

Bibliography

[1] F. S. Acton. *Numerical methods that work*. Maa, 1990.

[2] A. I. Aptekarev. "Sharp constants for rational approximations of analytic functions". **in***Sbornik: Mathematics*: 193.1 (2002), **page** 1.

[3] T. Bagby **and** N. Levenberg. "Bernstein theorems". **in***New Zealand Journal of Mathematics*: 22 (1993). Presentation of four proofs of Bernstein's result that best polynomial approximants to a function $f \in C([-1,1])$ converge geometrically if and only if f is analytic, with discussion of extension to higher dimension., **pages** 1–20.

[4] J.-P. Berrut **and** L. N. Trefethen. "Barycentric lagrange interpolation". **in***SIAM review*: 46.3 (2004), **pages** 501–517.

[5] T. A. Driscoll, Y. Nakatsukasa **and** L. N. Trefethen. "AAA rational approximation on a continuum". **in***SIAM Journal on Scientific Computing*: 46.2 (2024), A929–A952.

[6] A. Glaser, X. Liu **and** V. Rokhlin. "A fast algorithm for the calculation of the roots of special functions". **in***SIAM Journal on Scientific Computing*: 29.4 (2007), **pages** 1420–1438.

[7] G. H. Golub **and** J. H. Welsch. "Calculation of Gauss quadrature rules". **in***Mathematics of computation*: 23.106 (1969), **pages** 221–230.

[8] P. Gonnet, S. Guttel **and** L. N. Trefethen. "Robust Padé approximation via SVD". **in***SIAM review*: 55.1 (2013), **pages** 101–117.

[9] P. Gonnet, R. Pachón **and** L. Trefethen. "Robust rational interpolation and least-squares". **in**(2011).

[10] N. Hale **and** A. Townsend. "Fast and accurate computation of Gauss–Legendre and Gauss–Jacobi quadrature nodes and weights". **in***SIAM Journal on Scientific Computing*: 35.2 (2013), A652–A674.

[11] N. J. Higham. "The numerical stability of barycentric Lagrange interpolation". **in***IMA Journal of Numerical Analysis*: 24.4 (2004), **pages** 547–556.

[12] A. Horning **and** L. N. Trefethen. "Quadrature formulas from rational approximations". **in***arXiv preprint arXiv:2507.14971*: (2025).

[13] D. Jackson. "Über die Genauigkeit der Annäherung stetiger Funktionen durch ganze rationale Funktionen gegebenen Grades und trigonometrische Summen gegebener Ordnung". Jackson's PhD thesis under Landau in Göttingen, which together with Bernstein's work (1912) established many of the fundamental results of approximation theory. Despite being written in German, Jackson was American, from Massachusetts (Harvard Class of 1908). phdthesis. Universität Göttingen, 1911.

[14] C. Jordan. *Cours d'analyse de l'École polytechnique*. **volume** 1. Gauthier-Villars et fils, 1893.

[15] C. Jordan. "Sur la series de Fourier". **in***CR Acad. Sci., Paris*: 92 (1881), **pages** 228–230.

[16] J. L. Kelley. *General topology*. Courier Dover Publications, 2017.

[17] E. Y. Remez. "Sur la détermination des polynômes d'approximation de degré donnée". **in***Comm. Soc. Math. Kharkov*: 10.196 (1934), **pages** 41–63.

[18] H. Stahl. "The convergence of Padé approximants to functions with branch points". **in***Journal of Approximation Theory*: 91.2 (1997), **pages** 139–204.

[19] E. M. Stein **and** R. Shakarchi. *Complex analysis*. **volume** 2. Princeton University Press, 2010.

[20] L. N. Trefethen. "Is gauss quadrature better than Clenshaw–Curtis?" **in***SIAM review*: 50.1 (2008), **pages** 67–87.

[21] L. N. Trefethen **and** J. Weideman. "The exponentially convergent trapezoidal rule". **in***SIAM review*: 56.3 (2014), **pages** 385–458.

[22] L. N. Trefethen. *Approximation theory and approximation practice*. **volume** 164. SIAM, 2019.

[23] L. B. Winrich. "Note on a comparison of evaluation schemes for the interpolating polynomial". **in***The Computer Journal*: 12.2 (1969), **pages** 154–155.