

14. Discussion of high-order interpolation

As mentioned at various points in this book, high-order polynomial interpolation has a bad reputation. For equispaced points this is entirely appropriate, as shown in the last chapter, but for Chebyshev points it is entirely inappropriate. Here are some illustrative quotes from fifty years of numerical analysis textbooks, which we present anonymously.

We cannot rely on a polynomial to be a good approximation if exact matching at the sample points is the criterion used to select the polynomial. The explanation of this phenomenon is, of course, that the derivatives grow too rapidly. (1962)

However, for certain functions the approximate representation of $f(x)$ by a single polynomial throughout the interval is not satisfactory. (1972)

But there are many functions which are not at all suited for approximation by a single polynomial in the entire interval which is of interest. (1974)

Polynomial interpolation has drawbacks in addition to those of global convergence. The determination and evaluation of interpolating polynomials of high degree can be too time-consuming for certain applications. Polynomials of high degree can also lead to difficult problems associated with roundoff error. (1977)

We end this section with two brief warnings, one against trusting the interpolating polynomial outside [the interval] and one against expecting too much of polynomial interpolation inside [the interval]. (1980)

Although Lagrangian interpolation is sometimes useful in theoretical investigations, it is rarely used in practical computations. (1985)

Polynomial interpolants rarely converge to a general continuous function. Polynomial interpolation is a bad idea. (1989)

While theoretically important, Lagrange's formula is, in general, not as suitable for actual calculations as some other methods to be described below, particularly for large numbers n of support points. (1993)

Unfortunately, there are functions for which interpolation at the Chebyshev points fails to converge. Moreover, better approximations of functions like $1/(1+x^2)$ can be obtained by other interpolants—e.g., cubic splines. (1996)

In this section we consider examples which warn us of the limitations of using interpolation polynomials as approximations to functions. (1996)

Increasing the number of interpolation points, i.e., increasing the degree of the polynomials, does not always lead to an improvement in the approximation. The spline interpolation that we will study in this section remedies this deficiency.

(1998)

The surprising state of affairs is that for most continuous functions, the quantity $\|f - p_n\|_\infty$ will not converge to 0. (2002)

Because its derivative has $n-1$ zeros, a polynomial of degree n has $n-1$ extreme or inflection points. Thus, simply put, a high-degree polynomial necessarily has many “wiggles,” which may bear no relation to the data to be fit. (2002)

By their very nature, polynomials of a very high degree do not constitute reasonable models for real-life phenomena, from the approximation and from the handling point-of-view. (2004)

The oscillatory nature of high degree polynomials, and the property that a fluctuation over a small portion of the interval can induce large fluctuations over the entire range, restricts their use. (2005)

In addition to the inherent instability of Lagrange interpolation for large n , there are also classes of functions that are not suitable for approximation by certain types of interpolation. There is a celebrated example of Runge... (2011)

A great deal of confusion underlies remarks like these. Some of them are literally correct, but they are all misleading. In fact, polynomial interpolants in Chebyshev points are problem-free when evaluated by the barycentric interpolation formula. They have the same behavior as discrete Fourier series for periodic functions, whose reliability nobody worries about. The introduction of splines is a red herring: the true advantage of splines, as mentioned in Chapter 9, is not that they converge where polynomials fail to do so, but that they are more easily adapted to irregular point distributions and more localized, giving errors that decay exponentially away from singularities rather than just algebraically.

It is interesting to speculate as to how the distrust of high-degree polynomials became so firmly established. I think the crucial circumstance is that not one but several combined problems affect certain computations with polynomials, a situation complex enough to have obscured the truth from easy elucidation. If working with polynomials had been the central task of scientific computing, the truth would have been worked out nonetheless, but over the years there were always bigger problems pressing upon the attention of numerical analysts, like matrix computations and differential equations. Polynomial computations were always a sideline.

At the most fundamental level there are the two issues of conditioning and stability: both crucial, and not the same. See [Trefethen & Bau 1997] for a general discussion of conditioning and stability.

(1) *Conditioning of the problem.* The interpolation points must be properly

spaced (e.g., Chebyshev or Legendre) for the interpolation problem to be well-conditioned. This means that the interpolant should depend not too sensitively on the data. The Runge phenomenon for equally spaced points is the well-known consequence of extreme ill-conditioning, with sensitivities of order 2^n . The next chapter makes such statements precise through the use of Lebesgue constants.

(2) *Stability of the algorithm.* The interpolation algorithm must be stable (e.g., the barycentric interpolation formula) for a computation to be accurate, even when the problem is well-conditioned. This means that in the presence of rounding errors, the computed solution should be close to an exact solution for some interpolation data close to the exact data. In particular, the best-known algorithm of all, namely solving a Vandermonde linear system of equations to find the coefficients of the interpolant expressed as a linear combination of monomials, is explosively unstable, relying on a matrix whose condition number grows exponentially with the dimension (Exercise 5.2).

These facts would be enough to explain a good deal of confusion, but another consideration has muddied the water further, namely crosstalk with the notoriously troublesome problem of finding roots of a polynomial from its coefficients (to be discussed in Chapter 18). The difficulties of polynomial rootfinding were widely publicized by Wilkinson beginning in the 1950s, who later wrote an article called the “The perfidious polynomial” that won the Chauvenet Prize of the Mathematical Association of America [Wilkinson 1984]. Undoubtedly this negative publicity further discouraged people from the use of polynomials, even though interpolation and rootfinding are different problems. They are related, with related widespread misconceptions about accuracy: just as interpolation on an interval is trouble-free for a stable algorithm based on Chebyshev points, rootfinding on an interval is trouble-free for a stable algorithm based on expansions in Chebyshev polynomials (Chapter 18). But very few textbooks tell readers these facts.

SUMMARY OF CHAPTER 14. *Generations of numerical analysis textbooks have warned readers that polynomial interpolation is dangerous. In fact, if the interpolation points are clustered and a stable algorithm is used, it is bulletproof.*

Exercise 14.1. Convergence as $n \rightarrow \infty$. The 1998 quote asserts that increasing n “does not always lead to an improvement”. Based on the theorems of this book, for interpolation in Chebyshev points, for which functions f do we know that increasing n must lead to an improvement?

Exercise 14.2. Too many wiggles. Using Chebfun’s `roots(f, 'all')` option, plot all the roots in the complex plane of the derivative of the chebfun corresponding to $f(x) = \exp(x) \tanh(2x - 1)$ on $[-1, 1]$. What is the error in the argument in the second 2002 quote used to show that “a high-degree polynomial necessarily has many wiggles”?

Exercise 14.3. Your own textbook. Find a textbook of numerical analysis and examine its treatment of polynomial interpolation. (a) What does it say about behavior for large n ? If it asserts that this behavior is problematic, is this conclusion based on the assumption of equally spaced points, and does the text make this clear? (b) Does it mention interpolation in Chebyshev points? Does it state that such interpolants converge exponentially for analytic functions? (c) Does it mention the barycentric formula? (d) Does it claim that one should use a Newton rather than a Lagrange interpolation formula for numerical work? (This is a myth.)

Exercise 14.4. Spline interpolants. (a) Use Chebfun's `spline` command to interpolate $f(x) = 1/(1 + 25x^2)$ by a cubic spline in $n + 1$ equally spaced points on $[-1, 1]$. Compare the ∞ -norm error as $n \rightarrow \infty$ with that of polynomial interpolation in Chebyshev points. (b) Same problem for $f(x) = |x + 1/\pi|$. (c) Same problem for $f(x) = |x + 1/\pi|$, but measuring the error by the ∞ -norm over the interval $[0, 1]$.