Student Name: Minh Phan
Course: AMATH 352
Email address: minhphan@uw.edu

Homework 3

---

1 Introduction and overview of the problem

This report deals much with different approaches to interpolate Runge's function, which has pretty much a big association with Runge's phenomenon. In Runge's phenomenon, he observed that oscillations can occur, especially at the edges of an interval, when "performing polynomial interpolation with polynomials of high degree over a set of equispaced interpolation point." The function that he used in the phenomenon's discovery was therefore called the Runge function. [6]

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

In this report, I want to reproduce different approaches to mitigate Runge's phenomenon, including the use of Chebyshev nodes, and the use of trigonometric interpolation which have different properties from polynomial interpolation, the initial approach that cause the phenomenon in the first place. Through these approaches I want to investigate the numerical accuracy and stability, which will be discussed and graphed to evaluate their efficiencies.

2 Theoretical background and description of algorithm

Mathematically, interpolation is the estimation of the data points (of a function) from certain known data points of the function itself. For a set of coordinated points on a 2D graph, interpolation is performed by determining a function whose graph passes through all the known points. If the estimated points are within the range of the known points, we call it interpolation, and otherwise called extrapolation. [1] In the scope of this report, we stick to interpolation. Here are some other concepts that will help readers comprehend the process.

- Polynomial interpolation

  Polynomial interpolation is an interpolation method of fitting m data points using a polynomial function with the form $f(x) = a_0 + a_1^x + a_2x^2 + ... + a_nx^n$, with $a_0, a_1, ..., a_n \in \mathbb{R}$ as coefficients. [5]

- Trigonometric interpolation

  Trigonometric interpolation or trigonometric polynomial interpolation is an interpolation method of fitting data points using a trigonometric function with the form $f(x) = \sum_{k=0}^{N/2-1} a_k cos(k\pi x) + \sum_{k=N/2}^{N-1} b_k sin(k\pi x)$ with $a_k, b_k \in \mathbb{R}$ as coefficients. [2] Notice that the number of coefficients associated with the cosine and sine functions are equal.

- Determinant

  The determinant of a square matrix A is a real (scalar) number, denoted $det(A)$, characterizes some properties of the matrix and the linear map represented by the matrix, like whether it is invertible or not. It is computed using the entries of the matrix. [3]

- Chebyshev nodes

  Chebyshev nodes are specific real algebraic numbers used in polynomial interpolation because it minimizes the effect of Runge's phenomenon. [4] In the scope of this report, it is defined as:

$$x_k = cos\left(\frac{2k-1}{2n}\pi\right)$$

Here are the main steps of the algorithm I implemented to perform polynomial and trigonometric interpolation on Runge's function.

1. Choose interpolation nodes

   Regard these nodes as the predetermined data points in the definition of interpolation above, whose y-coordinates can be computed using Runge's function. Call $y_1, y_2, ..., y_n$ as the y-coordinates of the data points whose x-coordinates are $x_1, x_2, ..., x_n$, respectively.

2. Fit the system of equations into the matrix Write out the interpolation set of equations and substitute x_j's into the interpolation equations $p(x_k) = y_j$ with the x_j's and y_j's from the interpolation nodes chosen above. For the polynomial interpolation it is in the form

$$\begin{cases} p(x_1) & = a_1 + a_2 x_1 + a_3 x_1^2 + ... + a_n x_1^{n-1} \\ p(x_2) & = a_1 + a_2 x_2 + a_3 x_2^2 + ... + a_n x_2^{n-1} \\ ... & \\ p(x_n) & = a_1 + a_2 x_n + a_3 x_n^2 + ... + a_n x_n^{n-1} \end{cases} \rightarrow \begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & \cdots & x_2^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^{n-1} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

   The matrix on the left has a special name called the Vandermonde matrix. It is guaranteed to be invertible and therefore the equation is solvable, which also guarantees the vector of coefficients to be found. [7]

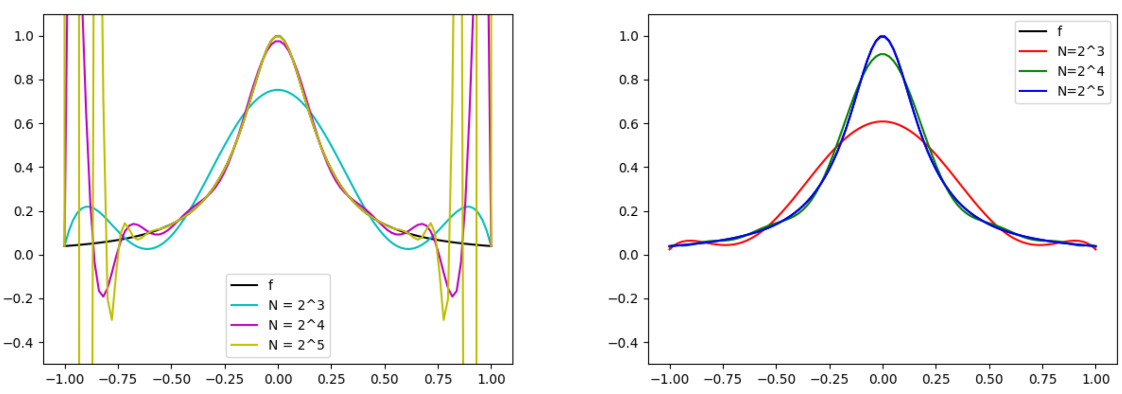
   For the corresponding linear system of the trigonometric interpolation, I unwind the trigonometric polynominal $T(x)$ and substitute elements in $\underline{x}$ into the linear system, then transform it into matrix-vector form

$$p(x_1) = a_1 cos(\pi x_1) + a_2 cos(2\pi x_1) + ... + a_{n/2} cos((n/2)\pi x_1) + b_1 sin(\pi x_1) + b_2 sin(2\pi x_1) + ... + b_{n/2} sin((n/2)\pi x_1)$$
$$p(x_2) = a_1 cos(\pi x_2) + a_2 cos(2\pi x_2) + ... + a_{n/2} cos((n/2)\pi x_2) + b_1 sin(\pi x_2) + b_2 sin(2\pi x_2) + ... + b_{n/2} sin((n/2)\pi x_2)$$
$$\vdots$$
$$p(x_n) = a_1 cos(\pi x_n) + a_2 cos(2\pi x_n) + ... + a_{n/2} cos((n/2)\pi x_n) + b_1 sin(\pi x_n) + b_2 sin(2\pi x_n) + ... + b_{n/2} sin((n/2)\pi x_n)$$

$$\rightarrow \begin{bmatrix} cos(\pi x_1) & cos(2\pi x_1) & \cdots & cos((n/2)\pi x_1) & sin(\pi x_1) & sin(2\pi x_1) & \cdots & sin((n/2)\pi x_1) \\ cos(\pi x_2) & cos(2\pi x_2) & \cdots & cos((n/2)\pi x_2) & sin(\pi x_2) & sin(2\pi x_2) & \cdots & sin((n/2)\pi x_2) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ cos(\pi x_n) & cos(2\pi x_n) & \cdots & cos((n/2)\pi x_n) & sin(\pi x_n) & sin(2\pi x_n) & \cdots & sin((n/2)\pi x_n) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ b_{n/2} \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

3. Solve the matrix equation, and the resulted vector includes the coefficients of the interpolation function.

3 Computational Results

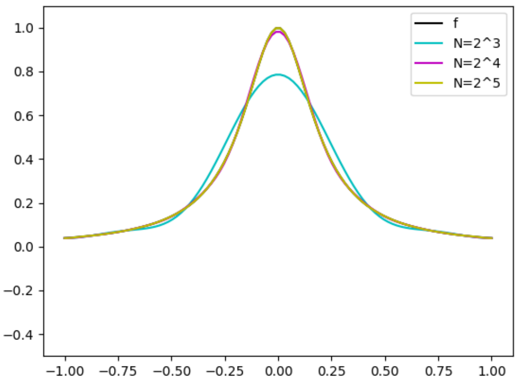


(a) Using uniform nodes

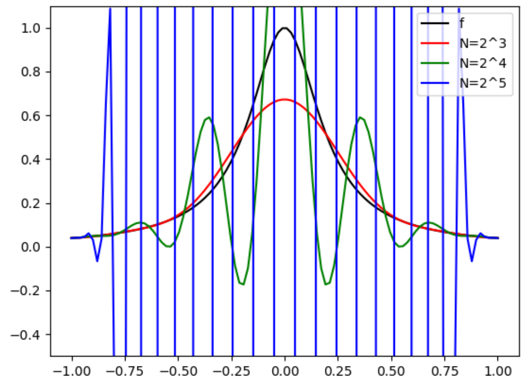
(b) Using Chebyshev nodes

Figure 1: Polynomial interpolation on polynomials with N-1 degrees in comparison to original Runge's function

The determinant of the V matrices of the x vectors using uniform grid are 7.3187×10^{-05} (N=2^3), 8.8861×10^{-29} (N=2^4), and 1.8523×10^{-141} (N=2^5). The determinant of the V matrices of the x vectors using Chebyshev nodes are 0.0002 (N=2^3), 5.8491×10^{-25} (N=2^4), and 2.7383×10^{-121} (N=2^5)



(a) Using uniform nodes



(b) Using Chebyshev nodes

Figure 2: Trigonometric interpolation with N trigonometric terms in comparison to original Runge's function

The determinant of the A matrices of the x vectors using uniform grid are 1.9449×10^{-13} (N=2^3), 1.8994×10^{-08} (N=2^4), and 58603.2336 (N=2^5). The determinant of the V matrices of the x vectors using Chebyshev nodes are 7.6428 (N=2^3), 0.4045 (N=2^4), and 2.7461×10^{-14} (N=2^5)

4 Summary and Conclusions

Using Chebyshev nodes proved to mitigate the oscillations happening over using uniform grid nodes in the classic polynomial interpolation, as we can see in Figure 1. It improved the quality of the interpolation at the edges by a considerable margin, but loses some of its accuracy at the center (compare the interpolation graphs of N=2^4 on both sides). The Vandermonde matrices' determinants are all larger comparing to those using uniform grid nodes with the same value of N, as we can see from the determinant values above.

However, the same thing does not happen in Figure 2, but opposite. Using trigonometric polynomials to interpolate Runge's function seems to be clean and effective when performing on the uniform grid, but have lots of oscillation on a collection of Chebyshev nodes. This is because Chebyshev nodes are expected to mitigate oscillations on polynomial interpolation only and not trigonometric interpolation. [6] The A matrices' determinants on those using Chebyshev nodes are now still bigger comparing to those using uniform grid nodes, with the same value of N=2^3 and N=2^4, as we can see from the determinant values above, but not in the case of N=2^5 where the determinant of the former is way larger than that of the latter. The values seem to increase as N increases in the uniform case, unlike those of the other and of the cases using polynomial interpolation in Figure 1.

References

[1] The Editors of Encyclopaedia Britannica. *interpolation*. Encyclopedia Britannica. URL: `https://www.britannica.com/science/interpolation`.

[2] Tobin A Driscoll and Richard J Braun. *Fundamentals of numerical computation*. Society For Industrial and Applied Mathematics. C, 2018.

[3] Dan Margalit and Joseph Rabinoff. *Interactive Linear Algebra*. 2019. URL: `https://textbooks.math.gatech.edu/ila/ila.pdf`.

[4] John H Mathews and Kurtis D Fink. *Numerical Methods Using MATLAB*. 3rd ed. Prentice Hall, 1999, pp. 232–236.

[5] *Polynomial interpolation*. Wikipedia, Oct. 2022. URL: `https://en.wikipedia.org/wiki/Polynomial_interpolation#Interpolation_theorem` (visited on 11/06/2022).

[6] *Runge's Phenomenon*. Wikipedia, Sept. 2021. URL: `https://en.wikipedia.org/wiki/Runge%27s_phenomenon`.

[7] Eric W. Weisstein. *Vandermonde Matrix*. mathworld.wolfram.com. URL: `https://mathworld.wolfram.com/VandermondeMatrix.html`.