Numerische Implementierung der linearen FEM
Homework 2

Group: Diamond
Yahuan Shi/ 494877
Chih-Cheng Huang/ 494886
Xingyu Shang/ 498775
Haowang Zhang/ 498759

May 2024

1 Task 1

1.1 Caculation

The picture below shows the hand calculation process of a system of linear equations. The method of substitution and elimination is used. Infinite non-recurring decimals will appear during the calculation process, so the final result is not an exac value, but an approximation.
The solution is:

$$x = \begin{bmatrix} -0.0146 \\ 0.0642 \\ 0.2469 \end{bmatrix}$$

when$A = \begin{bmatrix} 1 & 2 & 3 \\ 1 & 2 & 4 \\ 1 & 1 & 1 \end{bmatrix}$ is, then

$$x = \begin{bmatrix} 8 \\ -5 \\ 1 \end{bmatrix}$$

In this situation,there is no need for approximation during this operation, and the solution is accurate.

calculate LGS by handwriting

$Ax = b$

$$A = \begin{bmatrix} 8 & 2 & 4 \\ 1 & 16 & 4 \\ 1 & 1 & 16 \end{bmatrix} \quad b = \begin{bmatrix} 1 \\ 2 \\ 4 \end{bmatrix}$$

the dimension of solution x is 3×1

let x be $x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$

then we can get:

$$\begin{cases} 8x_1 + 2x_2 + 4x_3 = 1 & ① \\ x_1 + 16x_2 + 4x_3 = 2 & ② \\ x_1 + x_2 + 16x_3 = 4 & ③ \end{cases}$$

②-③: $15x_2 - 4x_3 = -\frac{2}{3}$ ④

8×②-①: $126x_2 + 28x_3 = 15$ ⑤

the two equations ④⑤ can be eliminated:

$$x_2 = 0.0642 \qquad x_3 = 0.2469$$

finally, substitute the values of x_2 and x_3 back into equation ① to obtain $x_1 = -0.0146$

so: $$x = \begin{bmatrix} -0.0146 \\ 0.0642 \\ 0.2469 \end{bmatrix}$$

This is an approximate result, since infinite non-repeating decimals appear during the calculation process.

However, when matrix A is replaced by $\begin{bmatrix} 1 & 2 & 3 \\ 1 & 2 & 4 \\ 1 & 1 & 1 \end{bmatrix}$, this situation will not occur.

and x can be obtained using the same method

$$x = \begin{bmatrix} 8 \\ -5 \\ 1 \end{bmatrix}$$

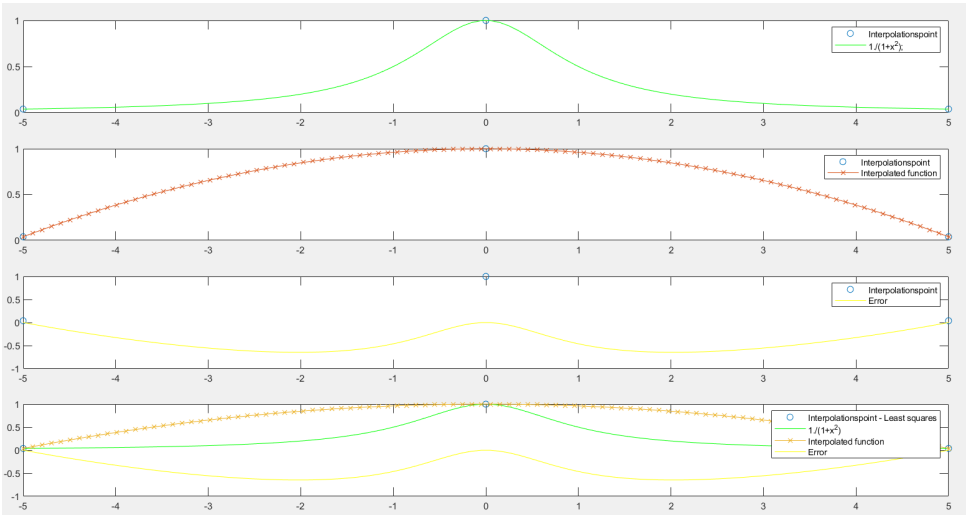
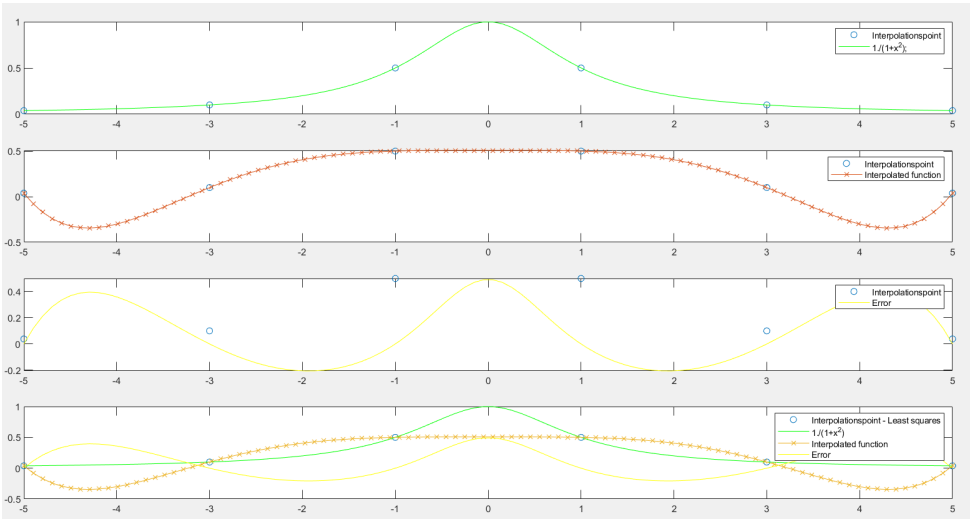
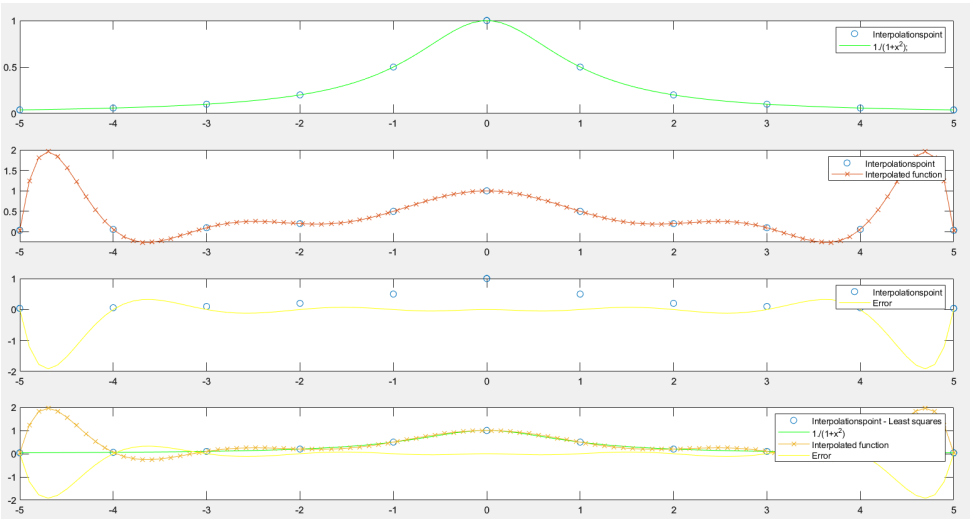
1.2 Backslash operator

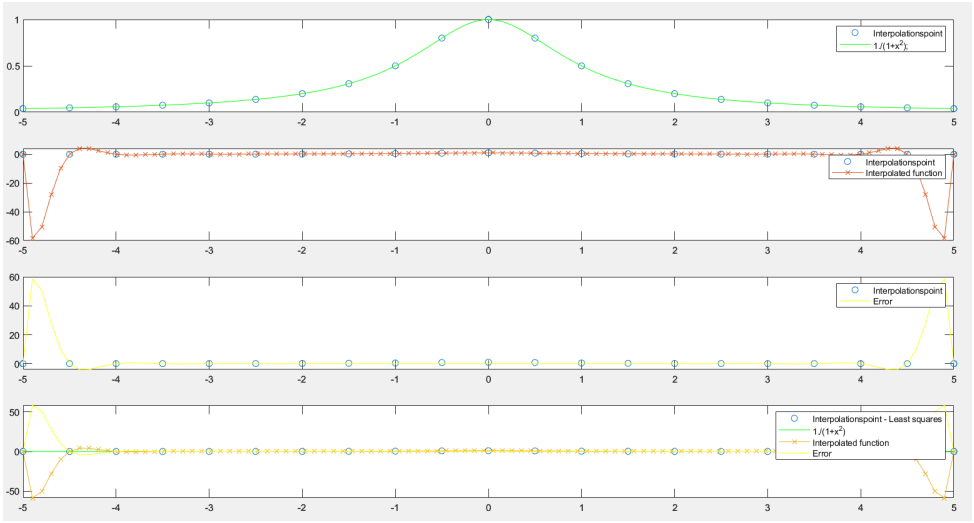
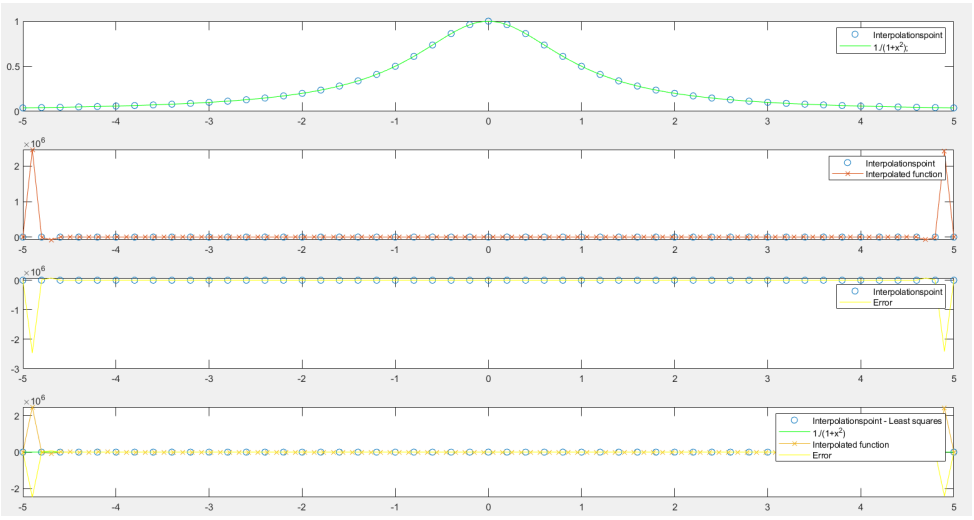
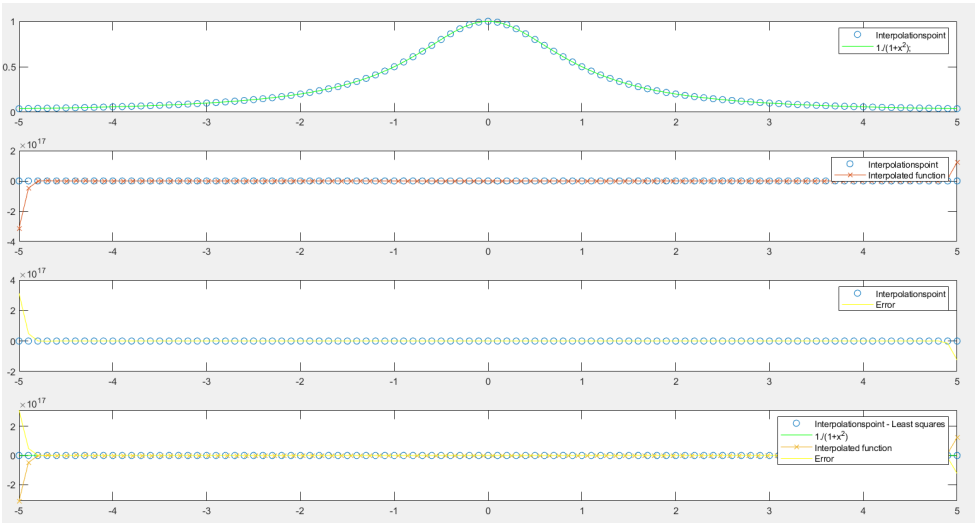
when we calculate $\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} x = \begin{bmatrix} 1 \\ 2 \\ 4 \end{bmatrix}$ by using Backslash-Operator $A \setminus b$ in Matlab.
It will be displayed in the matlab command line: "Warning: The matrix is close to singular values, or is poorly scaled. Results may not be accurate. RCOND = 1.541976e-18."
If we calculate the determinant of this matrix, we will find that the determinant is close to 0, so it can be regarded as a singular matrix. Therefore, there is no inverse matrix, but in matlab for this situation, we can find a least squares solution, although this is not an exact solution.

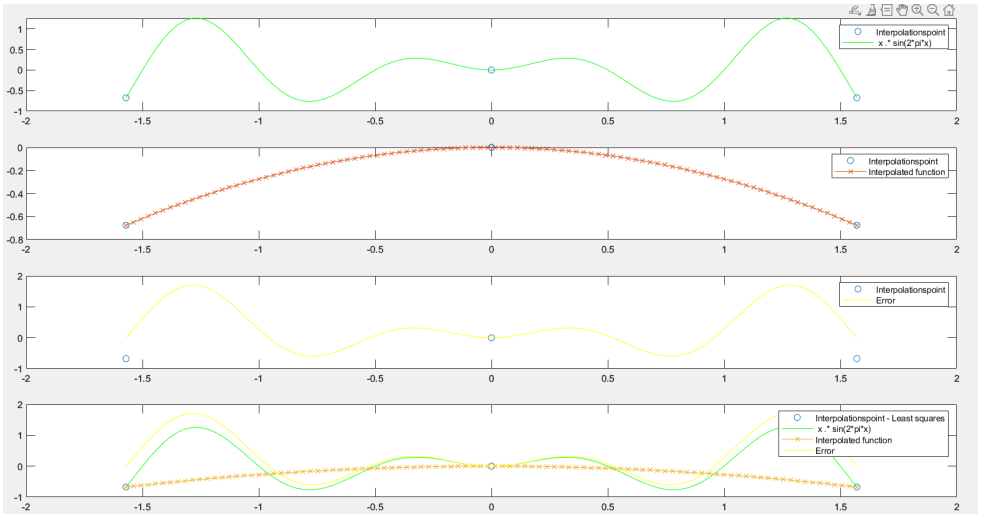
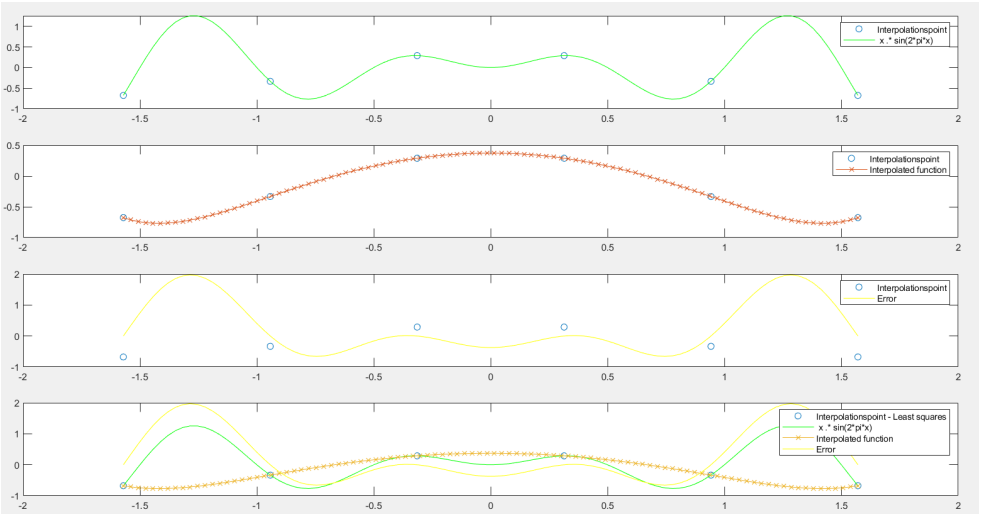
2 Task 2: Interpolation

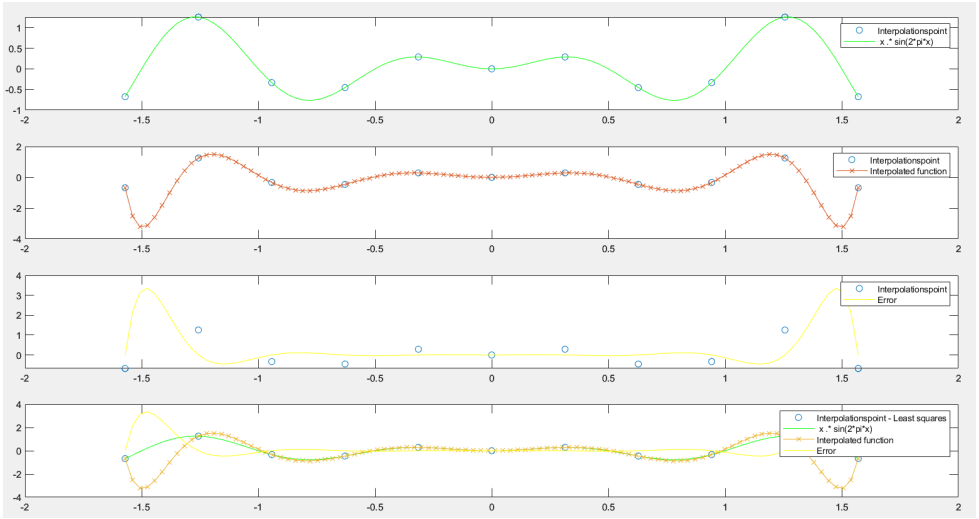
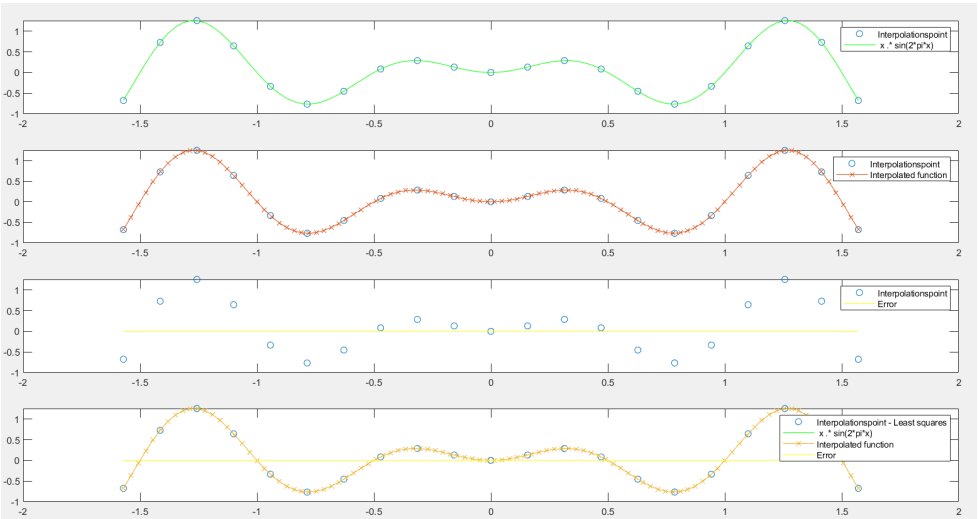
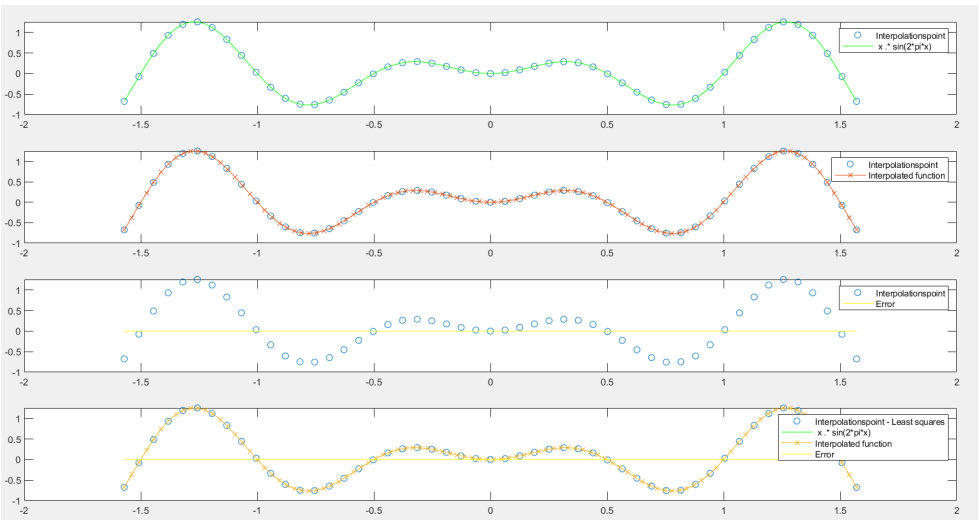
2.1 Polynominterpolation

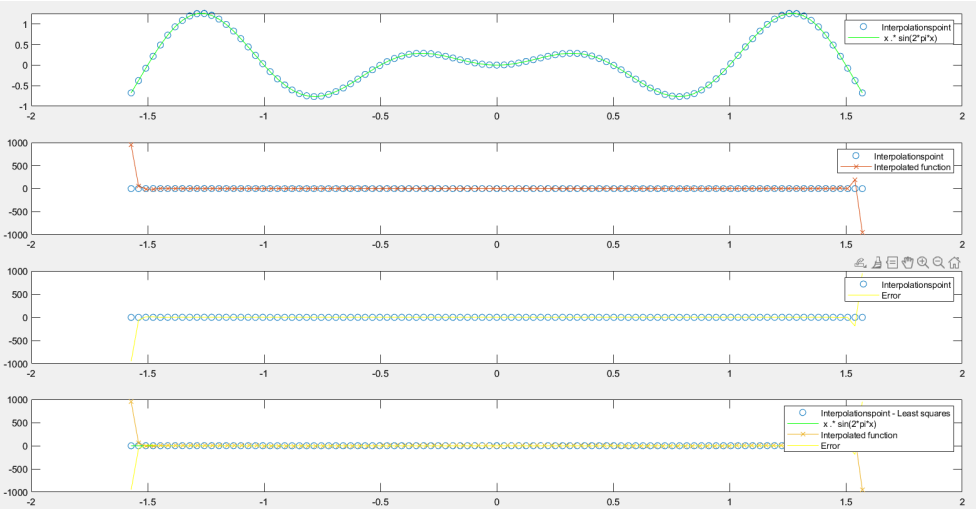
This is our analysis of polynomial interpolation. The next page is the corresponding simulation results:
By outputting the interpolation points, interpolation function, and error of the two functions, we can observe that: when the step size is larger, the interpolation points are also smaller, and the interpolation points increase as the step size decreases; when the interpolation points are smaller, the interpolating polynomial is also less accurate, and the interpolated results deviate from the original function with a larger error, especially in the area where the curve of the function is changing more rapidly. When the step size is smaller and the interpolation points are denser, such as in function $y_2(x) = x\sin(2\pi x)$, when the step size is 0.01π, the accuracy of the interpolating polynomial is higher, and the interpolation result is very close to the original function, the error is significantly reduced, and the interpolation result is very accurate.
So we can conclude:
① The smaller the step size, the more accurate the interpolation result is, the closer it is to the original function, and the smaller the error is.
② When there are fewer interpolation points, high order polynomial fitting may lead to overfitting problem with very large error.
Therefore, I have two suggestions:
① Reduce the step size and increase the number of interpolation points in the complex polynomial fitting process.
② try different interpolation methods to find more accurate interpolation result

Step Size	$y_1(x) = \frac{1}{1+x^2}$
5	
2	
1	

0.5	 Interpolationspoint 1./(1+x^2); Interpolationspoint Interpolated function Interpolationspoint Error Interpolationspoint - Least squares 1./(1+x^2) Interpolated function Error
0.2	 Interpolationspoint 1./(1+x^2); Interpolationspoint Interpolated function Interpolationspoint Error Interpolationspoint - Least squares 1./(1+x^2) Interpolated function Error
0.1	 Interpolationspoint 1./(1+x^2); Interpolationspoint Interpolated function Interpolationspoint Error Interpolationspoint - Least squares 1./(1+x^2) Interpolated function Error

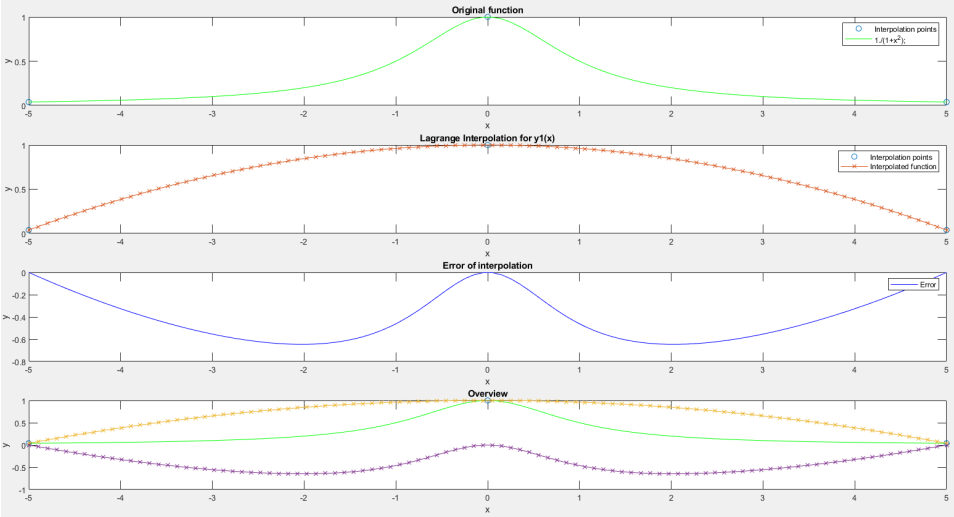
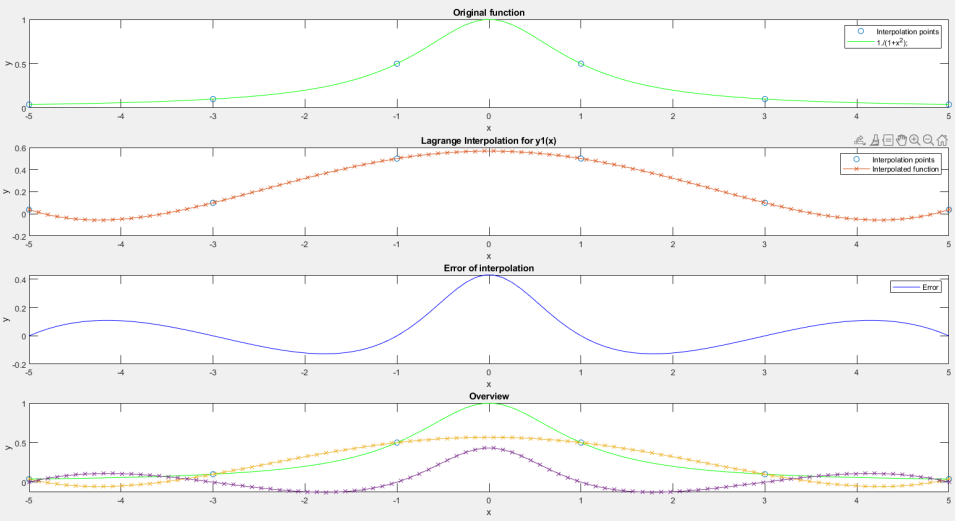
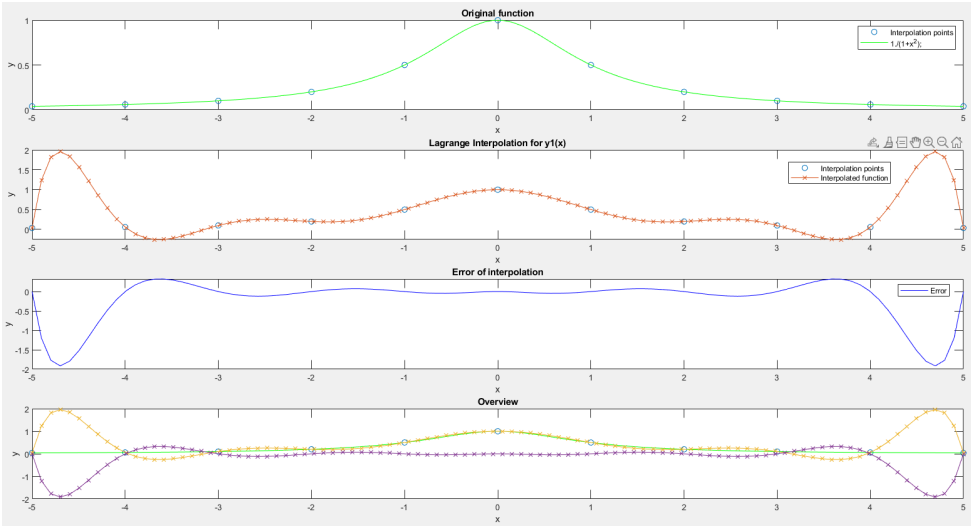
Step Size	$y_2(x) = xsin(2\pi x)$
0.5π	
0.2π	

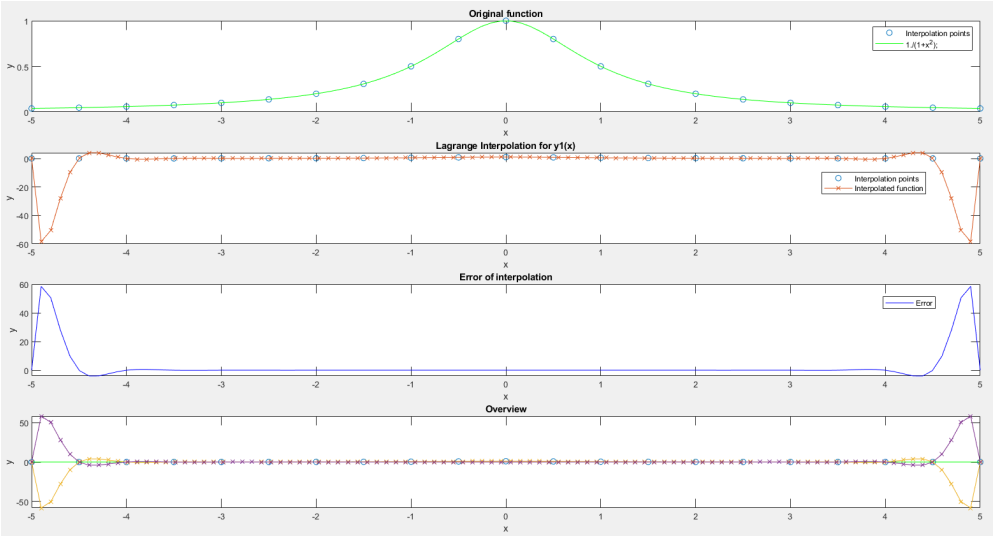
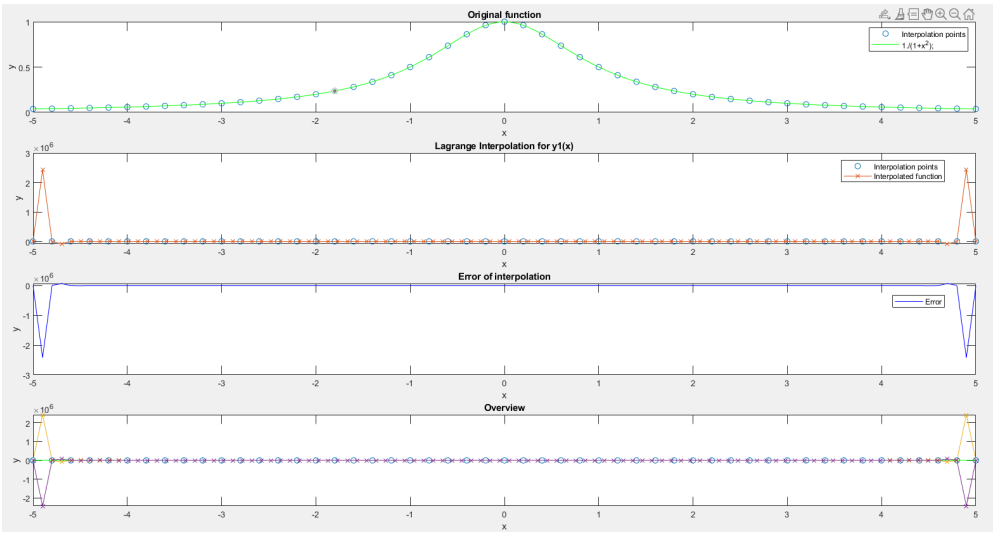
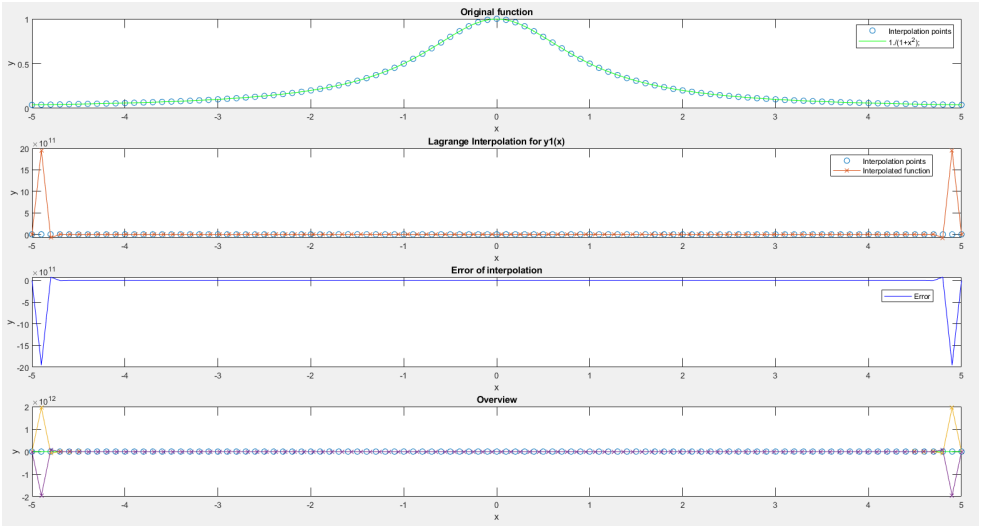
0.1π	
0.05π	
0.02π	

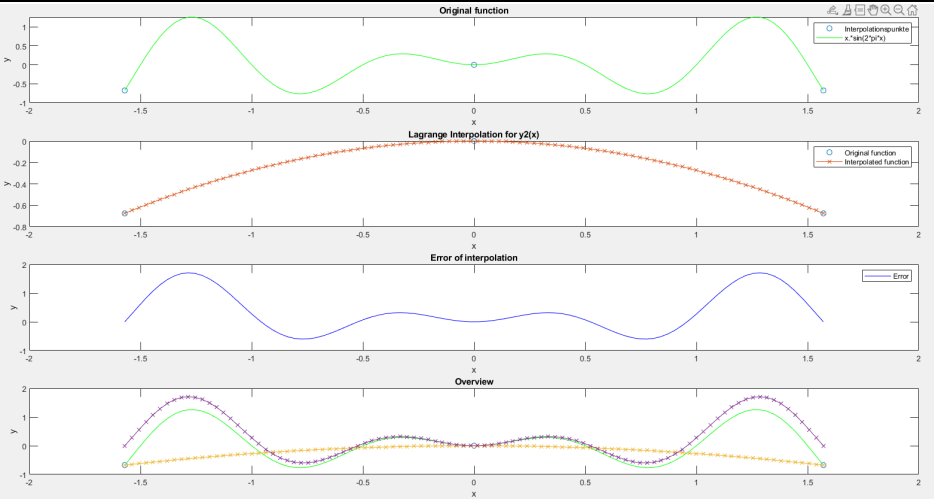
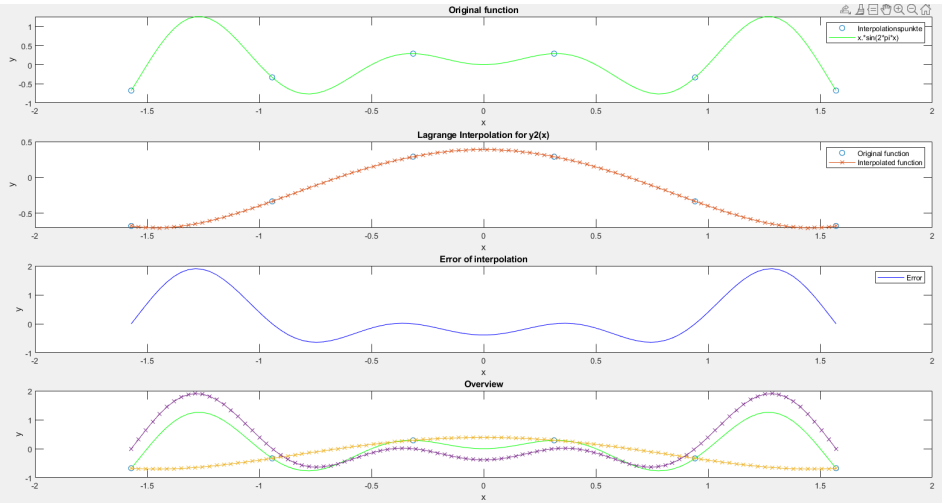
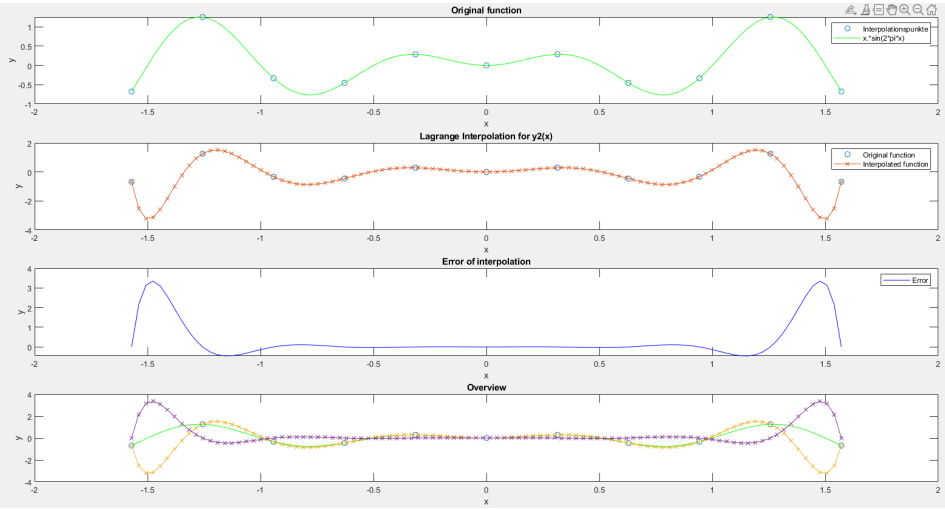
0.01π	 Interpolationspoint x .* sin(2*pi*x) Interpolationspoint Interpolated function Interpolationspoint Error Interpolationspoint - Least squares x .* sin(2*pi*x) Interpolated function Error

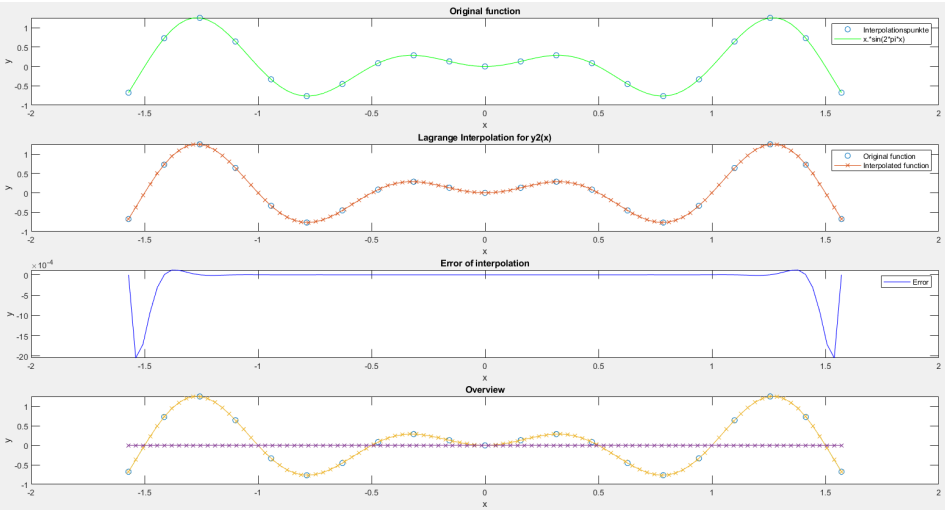
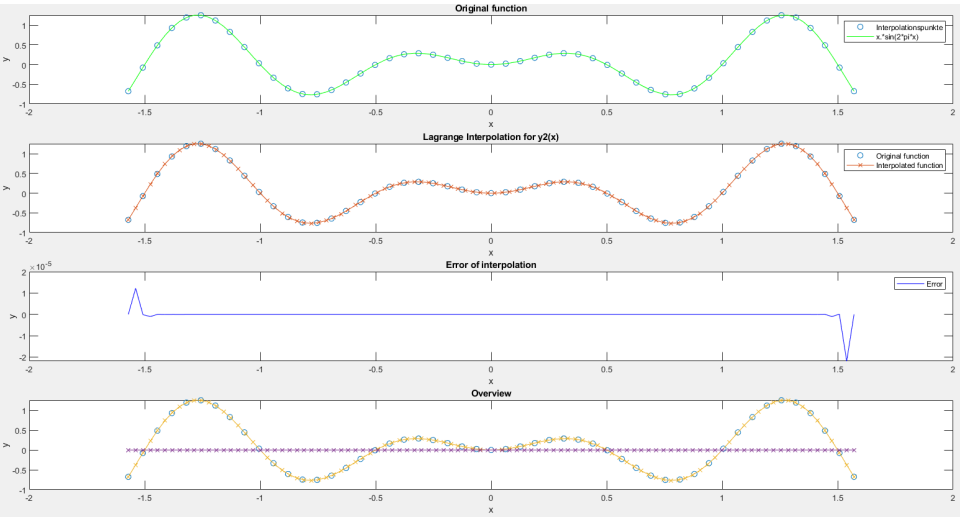
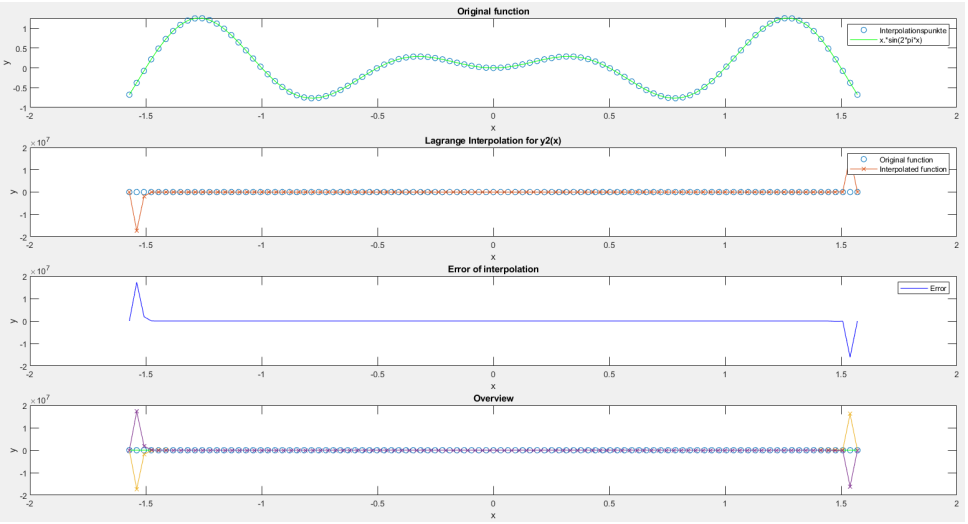
2.2 Lagrange-Interpolation

As we can see from the diagrams of both functions below, the smaller the step size is, the more points there are for the Lagrange interpolation. More interpolation points lead to a more accurate approximation to original function.
Furthermore, it is noted that values of error reduce as the number of interpolation points increase in most areas of the interval. However, high-degree polynomial interpolation tends to oscillate near the endpoints of the interval when interpolating equally spaced data points. It can be described as Runge's phenomenon. The higher degree interpolation function is, more severe the Runge's phenomenon occurs near the endpoints of the interval. It is especially to be noticed that the Lagrange interpolations of funtion y_2 with step size of $0{,}02\pi$ and $0{,}01\pi$ oscillate in different directions.
As a result, the error of Lagrange interpolation of funtion y_2 for both step sizes mentioned above also oscillate in different directions. Generally, the more interpolation points there are which means the higher degree Lagrange interpolation is, the better approximation Lagrange interpolation can offer in comparison to the original function.
In order to mitigate Runge's phenomenon, various techniques can be employed, such as using lower-degree polynomials for interpolation, employing non-equally spaced interpolation points or using alternative interpolation methods. These approaches aim to reduce the oscillations and inaccuracies associated with high-degree polynomial interpolation.

Step Size	$y_1(x) = \frac{1}{1+x^2}$
5	
2	
1	

0.5	 Original function Interpolation points 1./(1+x^2); Lagrange Interpolation for y1(x) Interpolation points Interpolated function Error of interpolation Error Overview x y
0.2	 Original function Interpolation points 1./(1+x^2); Lagrange Interpolation for y1(x) Interpolation points Interpolated function Error of interpolation Error Overview x y
0.1	 Original function Interpolation points 1./(1+x^2); Lagrange Interpolation for y1(x) Interpolation points Interpolated function Error of interpolation Error Overview x y

Step Size	$y_2(x) = xsin(2\pi x)$
0.5π	
0.2π	
0.1π	

0.05π	
0.02π	
0.01π	

2.3 stückweise mit LagrangePolynomen 2. Ordnung

When using second-order Lagrangian polynomials for piecewise interpolation, changes in step size will affect the accuracy and smoothness of the interpolation results. The smaller the step size, the closer the interpolation result is to the original function and the smoother the curve. The larger the step size, the more oscillation or distortion may occur in the interpolation result.
Below is a more detailed introduction to the impact of step size changes on the interpolation results.

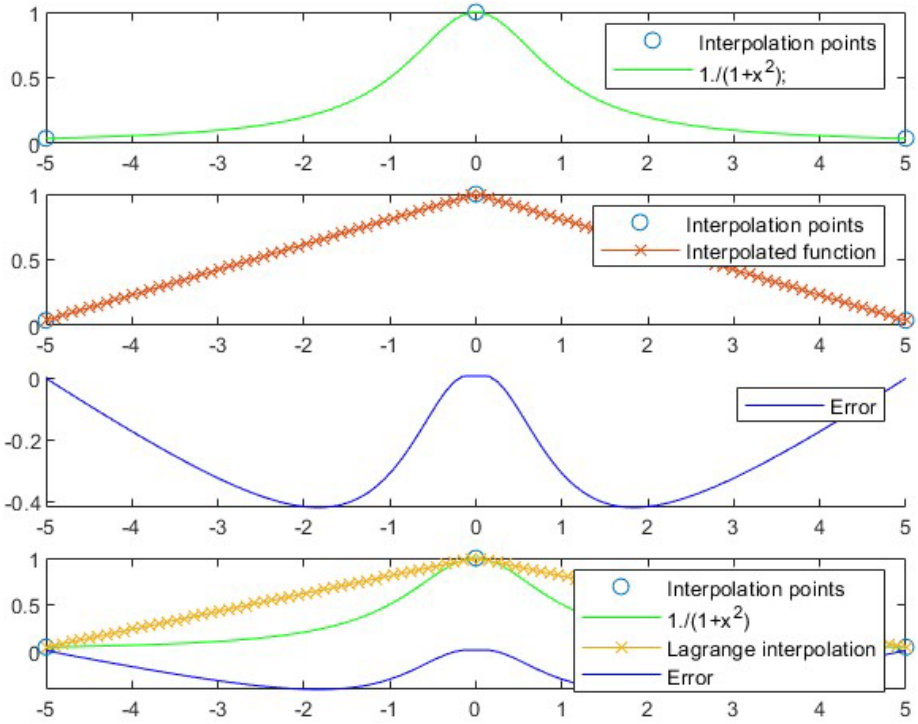
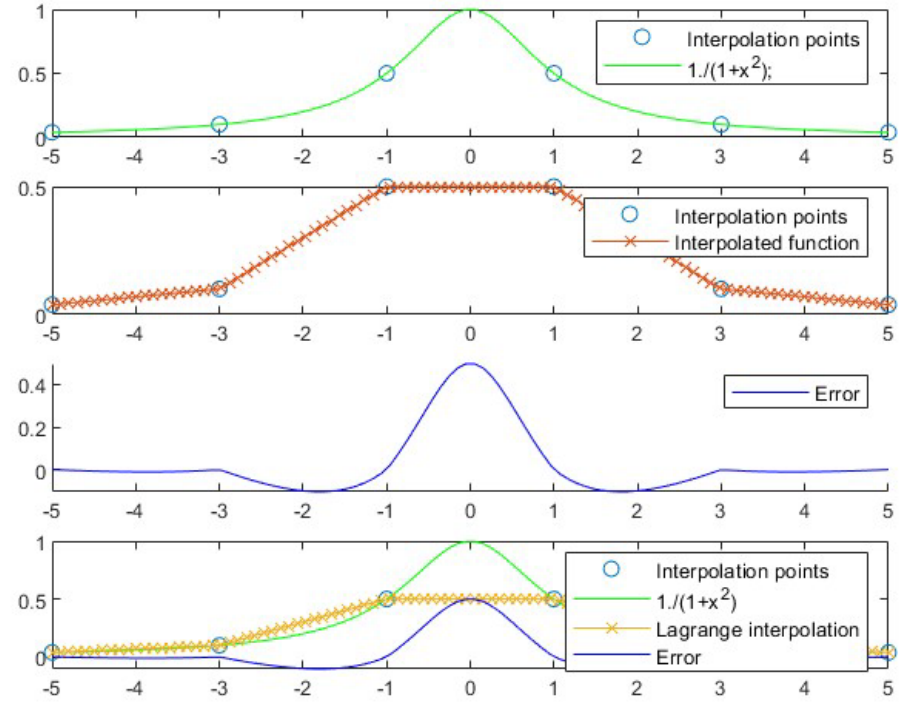
Accuracy: Smaller step sizes generally produce more accurate interpolation results because more interpolation points mean more data is used to fit the interpolation polynomial, and therefore the interpolation results are closer to the true function values.

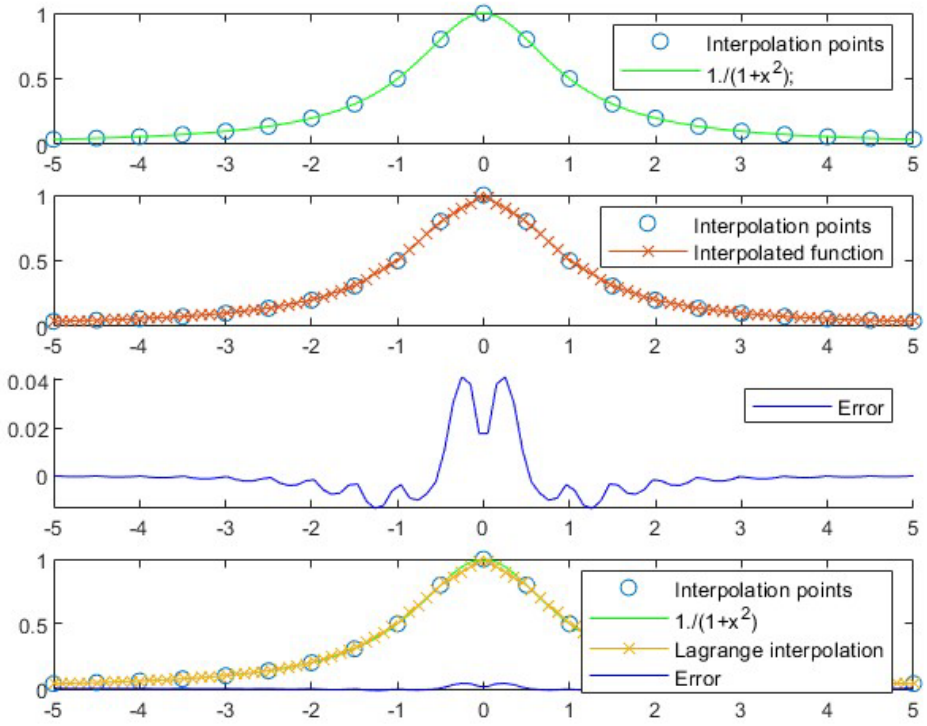
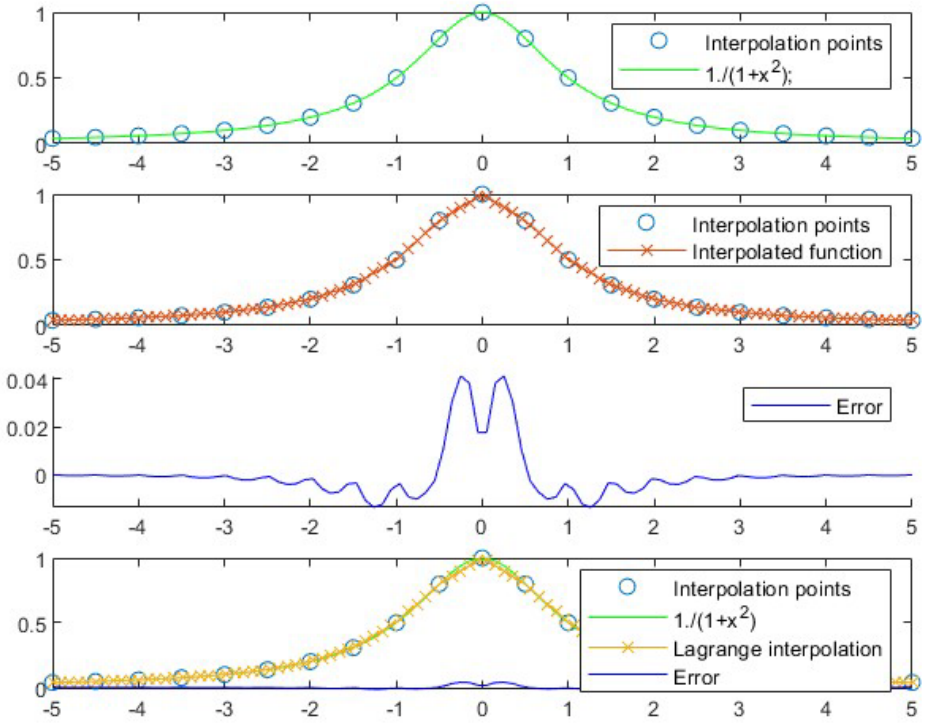
Oscillation and distortion: When the step size is large, the interpolation polynomial may not be able to approximate the shape of the original function well, resulting in oscillation or distortion in the interpolation result, especially where the original function has sharp changes or large curvature. A large step size may cause the interpolation result to deviate further from the original function.

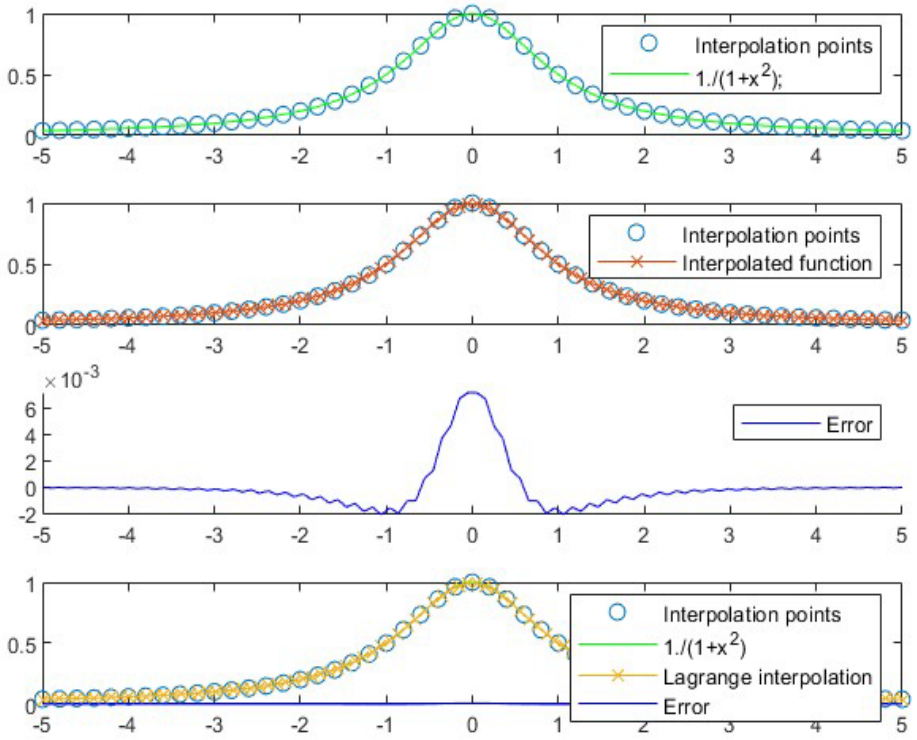
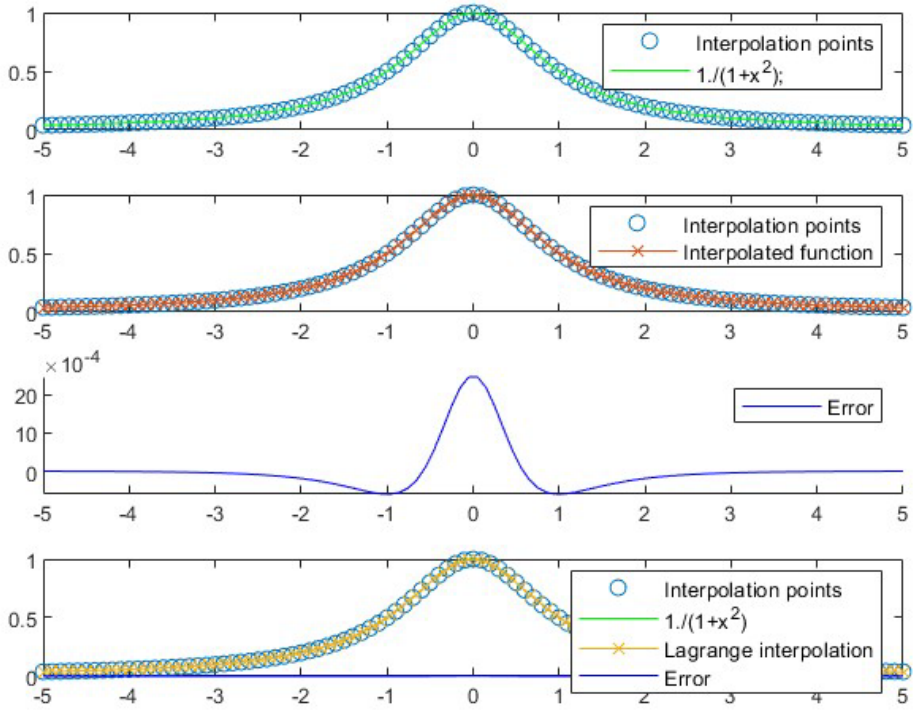
Computational cost: Smaller step sizes require more interpolation points and calculations, and therefore may require longer computation time.

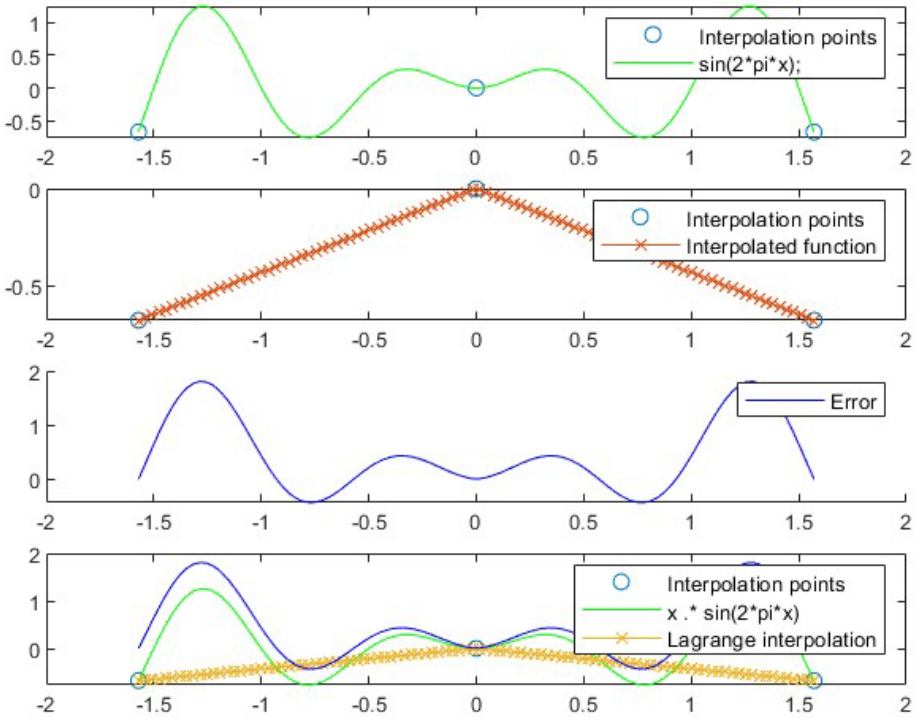
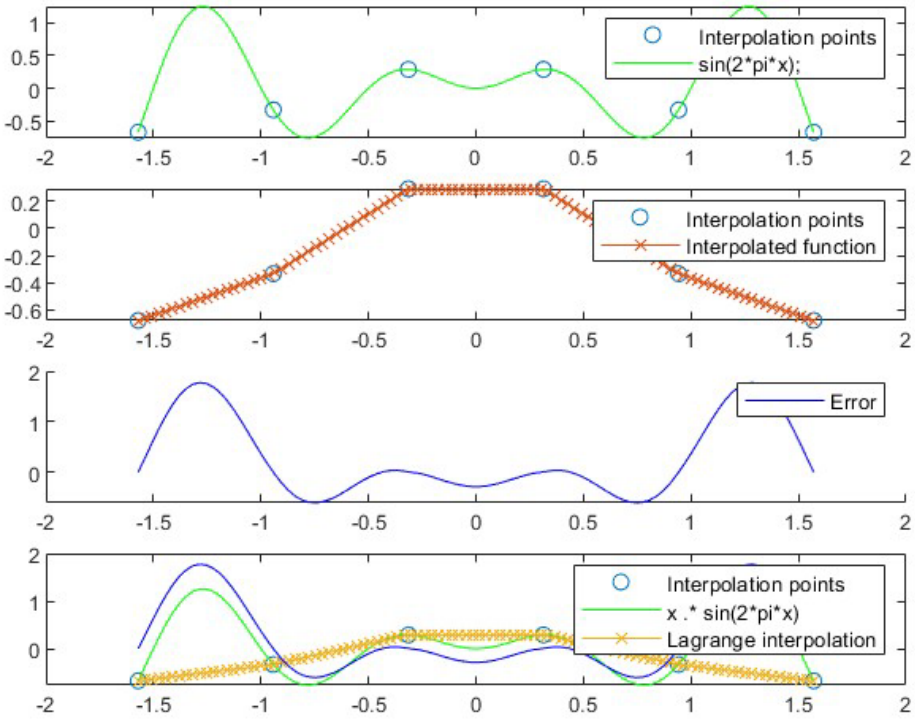
Smoothness: Smaller step sizes generally produce smoother interpolation curves because the interpolation polynomial between adjacent interpolation points is more linear.
Therefore, when choosing the step size, factors such as accuracy, computational cost, and smoothness need to be weighed and adjusted according to specific application requirements and data characteristics. Typically, the best step size value can be determined by trying different step sizes and observing changes in the interpolation results. Use more stable interpolation methods; Consider using methods such as spline interpolation, which use low-order polynomials over multiple small intervals to provide smoother and more robust interpolation results.

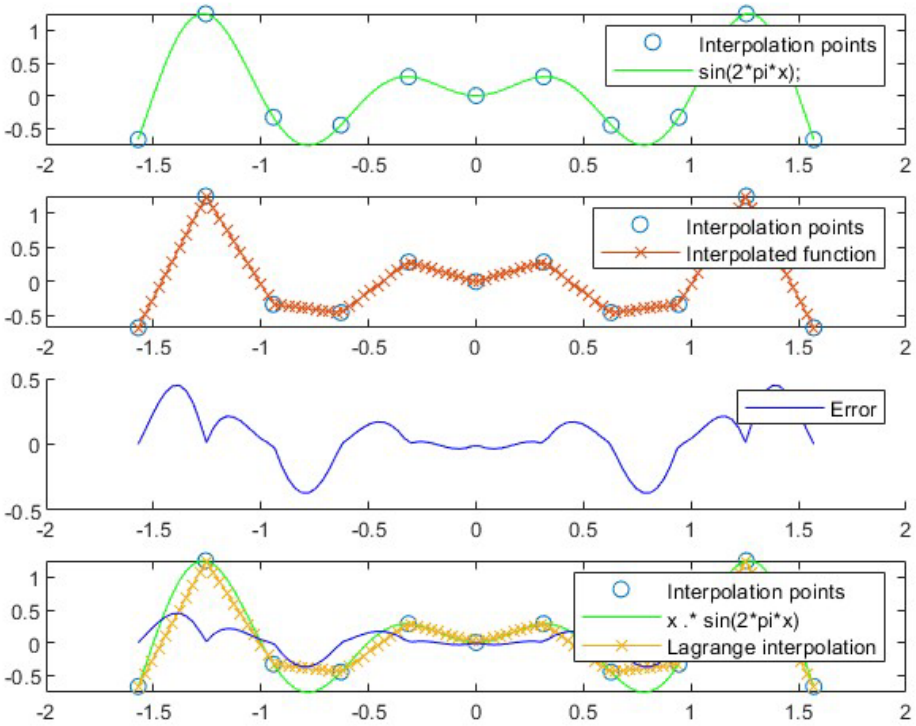
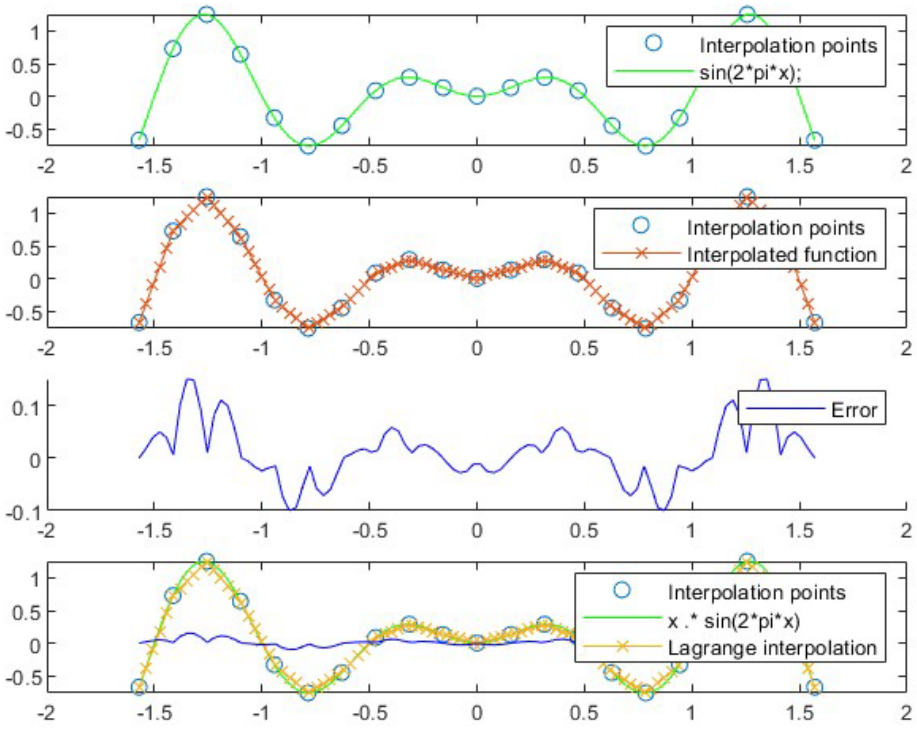
The following is the simulation result:

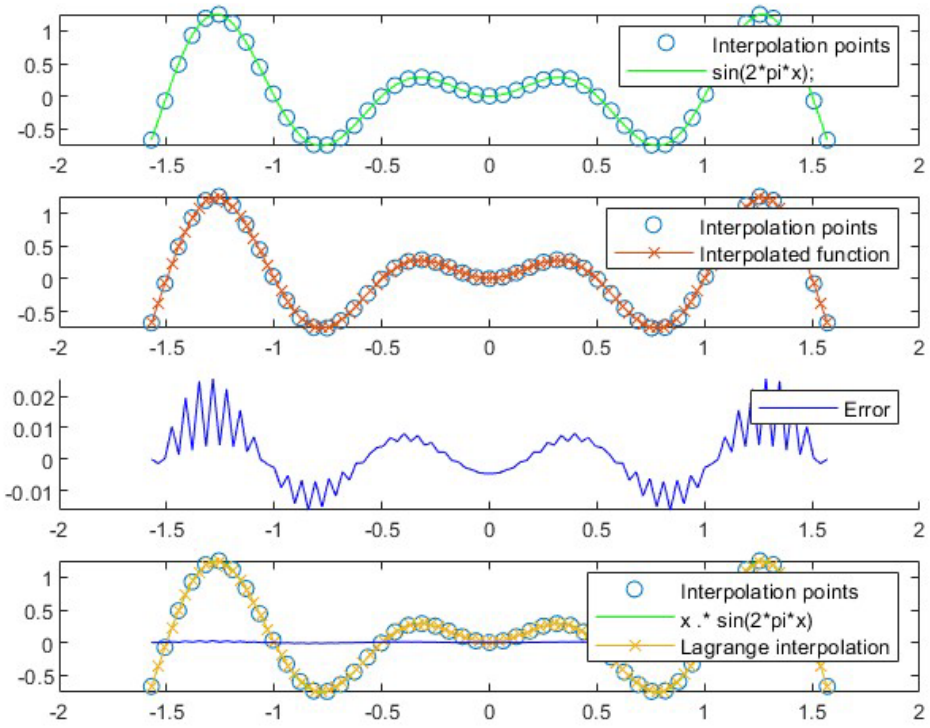
Step Size	$y_1(x) = \frac{1}{1+x^2}$
5	
2	

1	 Interpolation points 1./(1+x^2); Interpolation points Interpolated function Error Interpolation points 1./(1+x^2) Lagrange interpolation Error
0.5	 Interpolation points 1./(1+x^2); Interpolation points Interpolated function Error Interpolation points 1./(1+x^2) Lagrange interpolation Error

0.2	 Interpolation points 1./(1+x^2); Interpolation points Interpolated function ×10^-3 Error Interpolation points 1./(1+x^2) Lagrange interpolation Error
0.1	 Interpolation points 1./(1+x^2); Interpolation points Interpolated function ×10^-4 Error Interpolation points 1./(1+x^2) Lagrange interpolation Error

Step Size	$y_2(x) = xsin(2\pi x)$
0.5π	
0.2π	

0.1π	 Interpolation points sin(2*pi*x); Interpolation points Interpolated function Error Interpolation points x .* sin(2*pi*x) Lagrange interpolation
0.05π	 Interpolation points sin(2*pi*x); Interpolation points Interpolated function Error Interpolation points x .* sin(2*pi*x) Lagrange interpolation

$\frac{0.02}{\pi}$	
$\frac{0.01}{\pi}$