Summer 2020

An Adaptive Approach to Gibbs' Phenomenon

Jannatul Ferdous Chhoa

AN ADAPTIVE APPROACH TO GIBBS' PHENOMENON

by

Jannatul Ferdous Chhoa

A Dissertation
Submitted to the Graduate School,
the College of Arts and Sciences
and the School of Mathematics and Natural Sciences
of The University of Southern Mississippi
in Partial Fulfillment of the Requirements
for the Degree of Master of Science

Approved by:

Dr. James Lambers, Committee Chair
Dr. Haiyan Tian
Dr. Huiqing Zhu

Dr. James Lambers	Dr. Bernd Schroeder	Dr. Karen S. Coats
Committee Chair	Director of School	Dean of the Graduate School

August 2020

ABSTRACT

Gibbs' Phenomenon, an unusual behavior of functions with sharp jumps, is encountered while applying the Fourier Transform on them. The resulting reconstructions have high frequency oscillations near the jumps making the reconstructions far from being accurate. To get rid of the unwanted oscillations, we used the Lanczos sigma factor to adjust the Fourier series and we came across three cases. Out of the three, two of them failed to give us the right reconstructions because either it was removing the oscillations partially but not entirely or it was completely removing them but smoothing out the jumps a little too much. Even though the remaining one successfully removed the oscillations and gave us the right reconstruction, consistency needed to be gained. Taking this into account, we have developed an automated scheme that produces the right reconstruction each time. This scheme has been very efficient in reconstructing the signals quite accurately and consistently, leading to a new approach to signal processing.

ACKNOWLEDGMENTS

First of all, I would like to thank my advisor Dr. Lambers for his immense patience and kindness towards all his students. Without his careful guidance and intellectual help, this research would have been impossible. The credit for my learning growth at USM definitely goes to him. I would also like to express my gratitude to Dr. Zhu and Dr. Tian for their valuable suggestions on my thesis. I am also very grateful to my friends Eva, Damayanti, Oindrila apu, Chaity and Rashid bhaia who have become more like a family to me in Hattiesburg. I am thankful to my parents for supporting my dreams and letting me be who I want to be. Last but not the least, I would like to thank the love of my life for constantly motivating me, encouraging me and making me feel like his happiness lies in my success.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

Figure

LIST OF TABLES

Table

LIST OF ABBREVIATIONS

DFT - Discrete Fourier Transform
FFT - Fast Fourier Transform

NOTATION AND GLOSSARY

General Usage and Terminology

The notation used in this text represents fairly standard mathematical and computational usage. In many cases these fields tend to use different preferred notation to indicate the same concept, and these have been reconciled to the extent possible, given the interdisciplinary nature of the material. In particular, the notation for partial derivatives varies extensively, and the notation used is chosen for stylistic convenience based on the application. While it would be convenient to utilize a standard nomenclature for this important symbol, the many alternatives currently in the published literature will continue to be utilized.

The blackboard fonts are used to denote standard sets of numbers: $\mathbb{R}$ for the field of real numbers, $\mathbb{C}$ for the complex field, $\mathbb{Z}$ for the integers, and $\mathbb{Q}$ for the rationals. The capital letters, $A, B, \cdots$ are used to denote matrices, including capital greek letters, e.g., Λ for a diagnonal matrix. Functions which are denoted in boldface type typically represent vector valued functions, and real valued functions usually are set in lower case roman or greek letters. Caligraphic letters, e.g., $\mathcal{V}$, are used to denote spaces such as $\mathcal{V}$ denoting a vector space, $\mathcal{H}$ denoting a Hilbert space, or $\mathcal{F}$ denoting a general function space. Lower case letters such as i, j, k, l, m, n and sometimes p and d are used to denote indices.

Vectors are typset in square brackets, e.g., $[\cdot]$, and matrices are typeset in parenthesese, e.g., $(\cdot)$. In general the norms are typeset using double pairs of lines, e.g., $||\cdot||$, and the abolute value of numbers is denoted using a single pairs of lines, e.g., $|\cdot|$. Single pairs of lines around matrices indicates the determinant of the matrix.

Chapter 1
INTRODUCTION

Gibbs' phenomenon, named after Josiah W. Gibbs, is the strange behavior of functions with sudden discontinuities, that occurs near the discontinuities when the functions are represented by finite sums of Fourier series [1]. Even though Henry Wilbraham first observed it [2] in 1848, after almost 50 years, in 1898, it was rediscovered by Gibbs in a Fourier series of a different function. Gibbs gave a detailed mathematical analysis of this phenomenon in his two letters to *Nature* [3, 4], based on his observation of a sawtooth function. Later in 1906, Maxime Brocher made a significant contribution to their study by explaining the phenomenon from a more general point of view [5]. Brocher proved that the behavior is, indeed, a consequence of truncating the Fourier series of any function that has a jump discontinuity.

The phenomenon mainly occurs because of the nonuniform convergence of the Fourier series for functions with sudden discontinuities [7]. It causes ripples near the points of discontinuities, making the reconstructed function very inaccurate compared to the original one. Adding more and more Fourier coefficients to the series gradually reduces the width of the oscillation; however, the height of the oscillation remains constant and the magnitude of this height exceeds the value of the original function at the jump [6]. As a result, the finite sum that approximates the function fails to approach it uniformly over an interval that includes a point where the function is not continuous.

A function with a discontinuity in the spatial domain requires infinite frequency content; but practically it is impossible to sample infinite frequency content. As a consequence, Gibbs' phenomenon is encountered quite often in the field of applied science and engineering. In digital signal processing it commonly appears as ringing artifacts near instantaneous transitions that can cause unwanted echos in acoustics [8] or distortion near the edges in images [9]. It is also observed in the form of a series of lines in Magnetic Resonance Imaging [10] whenever there are tissue boundaries or tissue discontinuities. Other applications include, but are not limited to, vibration and stability of complex beams, pseudo-spectral time-domain analysis, techniques for numerical analysis and computation, and cubic-spline interpolation of discontinuous functions [11].

The three most well established approaches to get rid of Gibbs' phenomenon are: Fejer summation, Riesz summation and Sigma approximation. For this thesis, we worked with

Sigma Approximation that uses a factor, called a Lanczos sigma factor, which adjusts the Fourier series to suppress the unwanted oscillation [12]. When it comes to gaining convergence at the points of discontinuity to approximate a function, qualitatively Sigma approximation is better than Fejer summation [13]. Numerical experimentation have also indicated that Fejer and Riesz summation are not as effective at eliminating oscillations as Lanczos sigma factors.

In Chapter 2, we explain Gibbs' Phenomenon with a very simple example and describe the 'Goldilocks problem', a problem that we faced while using Lanczos sigma factors on the Fourier series. In Chapter 3, we develop our adaptive method by observing graphs of different functions in spectrum domain and solving a least squares problem for a general function. In Chapter 4, the graphical results along with numerical results are presented in an attempt to compare accuracy. Finally, conclusions and possible future work are discussed in Chapter 5.

Chapter 2

BACKGROUND

2.1 Fourier Series

The Fourier series of a 2π-periodic function $f(x)$, defined on the interval $[0,L]$, can be represented as

$$f(x) = \frac{1}{2}a_0 + \sum_{\omega=1}^{\infty}\left[a_\omega \cos\left(\frac{2\pi\omega}{L}x\right) + b_\omega \sin\left(\frac{2\pi\omega}{L}x\right)\right] \tag{2.1}$$

where the Fourier coefficients are:

$$a_\omega = \frac{2}{L}\int_0^L f(x)\cos\left(\frac{2\pi\omega}{L}x\right)dx, \quad \text{and} \quad b_\omega = \frac{2}{L}\int_0^L f(x)\sin\left(\frac{2\pi\omega}{L}x\right)dx.$$

Here, ω is an integer. The complex exponential form of (2.1) is:

$$f(x) = \frac{1}{\sqrt{L}}\sum_{\omega=-\infty}^{\infty}\hat{f}(\omega)e^{\frac{2\pi i\omega}{L}x} \tag{2.2}$$

where

$$\hat{f}(\omega) = \frac{1}{\sqrt{L}}\int_0^L f(x)e^{-\frac{2\pi i\omega}{L}x}dx \tag{2.3}$$

is the Fourier Transform of $f(x)$. One can get back the original function $f(x)$ by taking the Inverse Fourier Transform of $\hat{f}(\omega)$. But this is only possible when f is continuously differentiable and $\hat{f}$ is absolutely convergent; i.e. $\int_{-\infty}^{\infty}|f(x)|dx < \infty$.

Now if we define a grid on the interval $[0,L]$ that has N equally-spaced points $x_j = j\frac{L}{N}$, for $j = 0,...,N-1$, then we can approximate the Discrete Fourier series of $f(x)$ as

$$f_N(x) = \frac{1}{\sqrt{L}}\sum_{\omega=-N/2+1}^{N/2} e^{\frac{2\pi i\omega}{L}x}\tilde{f}(\omega) \tag{2.4}$$

where

$$\tilde{f}(\omega) = \frac{1}{\sqrt{L}}\sum_{j=0}^{N-1} e^{-\frac{2\pi i\omega}{L}x_j}f(x_j)\Delta x \tag{2.5}$$

is called the Discrete Fourier Transform (DFT) of $f_N(x)$.

A more efficient form of DFT of length N can be computed by writing it as a sum of two DFTs, each having a length of $N/2$. This newly computed Fourier Transform is

called the Fast Fourier Transform (FFT) that consists of two DFTs, one formed from the even-numbered points of the original set of N, and another is formed from the odd-numbered points of the set. Then the FFT can be defined as follows

$$\tilde{f}(\omega) = \frac{1}{2}\tilde{f}^e(\omega) + \frac{1}{2}e^{-2\pi i\omega/N}\tilde{f}^o(\omega) \tag{2.6}$$

where $\tilde{f}^e(\omega)$ is the $N/2$-point DFT of f obtained from its values at the even-numbered points of the N-point grid and $\tilde{f}^o(\omega)$ is the $N/2$-point DFT of f obtained from its values at the odd-numbered points of the grid. The computational expense of FFT is $O(N\log_2 N)$ whereas the computational expense of DFT is $O(N^2)$ which is a lot higher than the former one.

2.2 Convergence of Fourier Series

The Fourier series for an L-periodic function $f(x)$ converges to $f(x)$ at any point in $[0,L]$ at which f is continuously differentiable. If f has a jump discontinuity, then it converges to the average values of f on either side of the jump. Thus we can state the convergence theorem [7] at a point of discontinuity as:

If a function f is piecewise continuous and periodic as well as left and right differentiable at x then the Fourier Series of f at x converges to the value

$$\frac{1}{2}\left[f(x+0) + f(x-0)\right] \tag{2.7}$$

Here

$$f(x+0) = \lim_{h\to 0^+} f(x+h)$$

and

$$f(x-0) = \lim_{h\to 0^+} f(x-h)$$

are the right-hand limit and left-hand limit of f at x respectively.

2.3 Gibbs' Phenomenon

Continuing from the previous section, if f is not L-periodic, then there will be jump discontinuities as we extend the graph of $f(x)$ beyond $[0,L]$ and Fourier series will be converging again to (2.7) on either side of each discontinuity. Because of these discontinuities, more and more oscillatory behavior of the basis functions $e^{i\omega x}$ occurs as $|\omega|$ keeps increasing and thus Gibbs' Phenomenon is encountered as we truncate the Fourier series.

If f is k-times differentiable and its kth derivative is at least piece-wise continuous, then the coefficients of the complex exponential form of the Fourier series satisfy

$$| \hat{f}(\omega) | \leq \frac{C}{| \omega |^{k+1} + 1} \tag{2.8}$$

for some constant C that is independent of ω [15].

The above equation describes how rapidly the Fourier coefficients decay to zero. Very smooth functions will have very rapidly decaying Fourier coefficients resulting in the rapid convergence of the Fourier series. On the other hand, functions with discontinuities will have very slowly decaying Fourier coefficients causing the Fourier series to converge very slowly.

2.4 Example of a Square Wave Function

Now we illustrate a simple example of a square wave function where we set the period of the function to 2π and reconstruct it using the Fast Fourier Transform. First we are going to use the FFT to obtain the Fourier coefficients on a 2^{10}-point grid (Figure:2.1), and then we will evaluate the Fourier series, truncated to 2^{10} points, on a 2^{20}-point grid.(Figure:2.2)

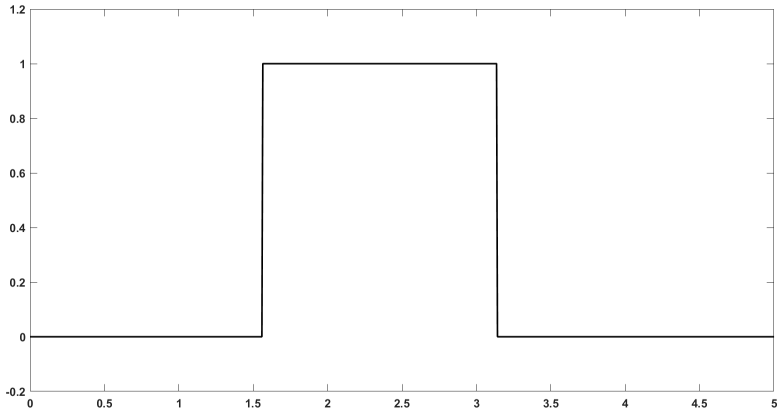

Figure 2.1: Original function using coarse grid (2^{10} grid points used)

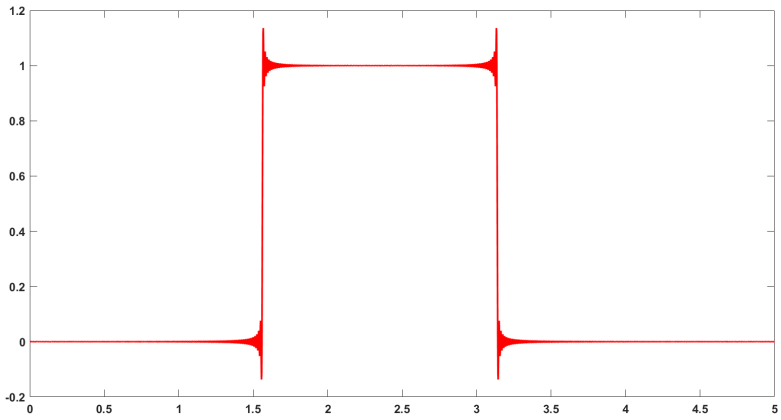

Figure 2.2: Reconstructed function using fine grid (2^{20} grid points used)

From the above figures we observe that, even though we have used a lot more Fourier coefficients to reconstruct our function, the reconstruction (Figure 2.2) is still very inaccurate compared to the original function (Figure 2.2). The oscillations appearing near the jumps in the reconstructed function are demonstrations of Gibbs' phenomenon.

2.5 Resolution of Gibbs' Phenomemon by Using Lanczos σ Factor

Truncating the Fourier series (2.1) up to N terms gives us

$$s(x) = \frac{1}{2}a_0 + \sum_{\omega=1}^{N-1}\left[a_\omega \cos\frac{2\pi\omega x}{L} + b_\omega \sin\frac{2\pi\omega x}{L}\right] + \frac{1}{2}a_N \cos\frac{2\pi N x}{L} \tag{2.9}$$

Averaging (2.9) over the interval $(x-\frac{L}{2N}, x+\frac{L}{2N})$ we get the Lanczos Sigma Approximation of the Fourier series [16]:

$$s(x) = \frac{1}{2}a_0 + \sum_{\omega=1}^{N-1}\left(\operatorname{sinc}\frac{\omega}{N}\right)^p\left[a_\omega \cos\frac{2\pi\omega x}{L} + b_\omega \sin\frac{2\pi\omega x}{L}\right] \tag{2.10}$$

In the above equation (2.10), $\operatorname{sinc}\frac{\omega}{N}$ is known as the Lanczos σ factor which gradually reduces the oscillations caused by Gibbs' Phenomenon as we keep increasing the value of p [12]. Here, the sinc function is defined as:

$$\operatorname{sinc} y = \begin{cases} \frac{\sin(\pi y)}{\pi y} & y \neq 0 \\ 1 & y = 0 \end{cases}$$

2.6 Goldilocks Problem

Revisiting the square wave example mentioned in Section 2.4: As we observed the oscillations near the jumps by using different values of p we came across three cases:

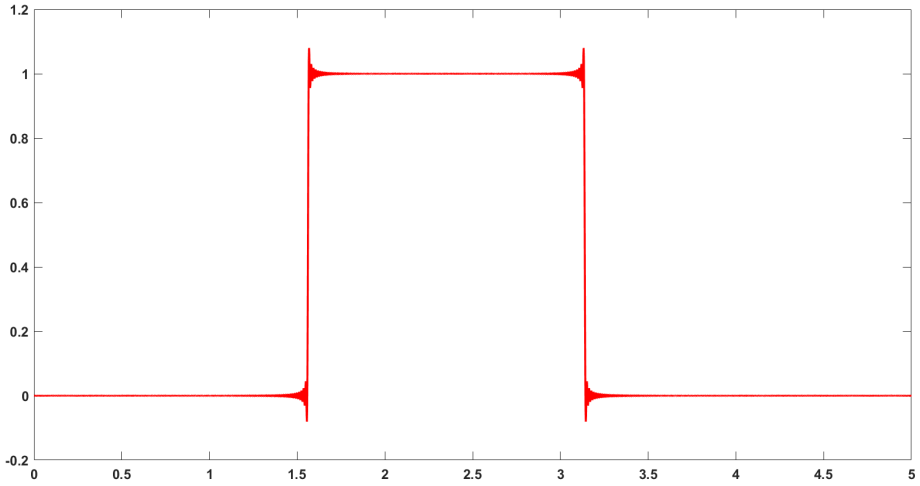

Figure 2.3: Case 1

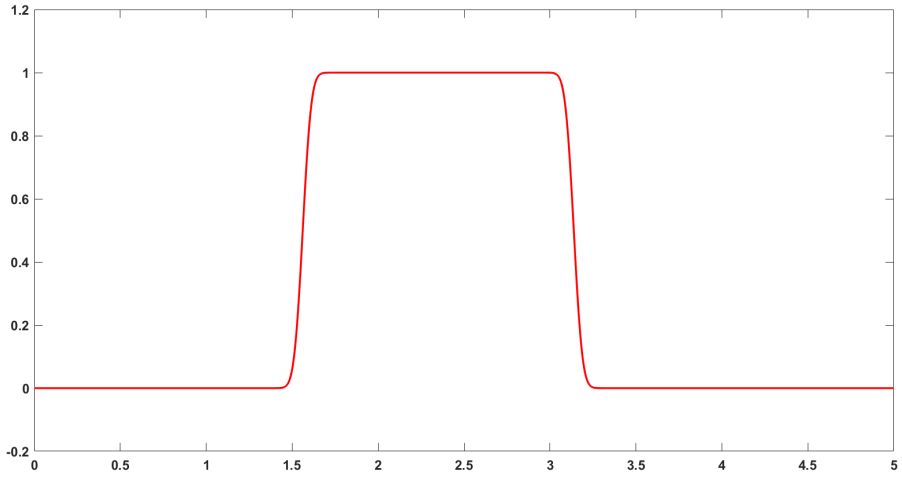

Figure 2.4: Case 2

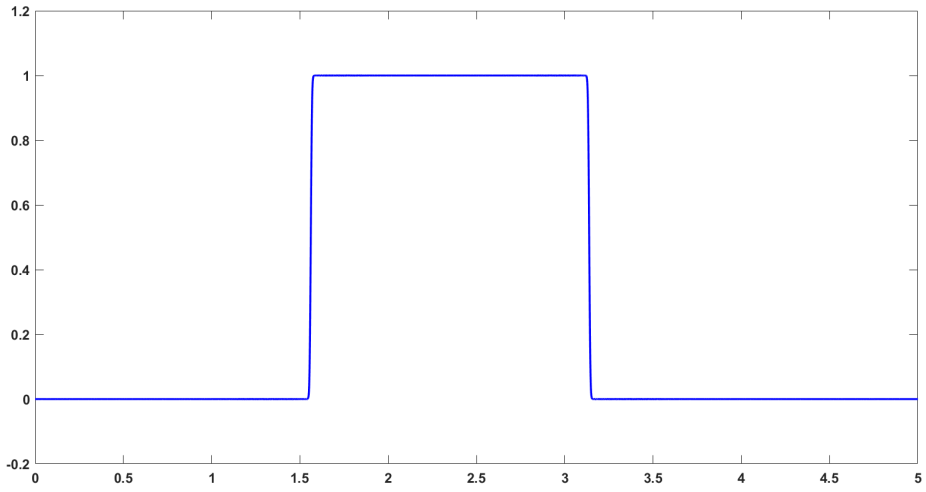

Figure 2.5: Case 3

- **Case 1:** When p is too small, oscillations are removed partially but not entirely. (Figure 2.3)
- **Case 2:** When the value of p is too large, oscillations are removed completely but smoothing out the jumps too much. (Figure 2.4)
- **Case 3:** When p is sufficient, oscillations are successfully removed. (Figure 2.5)

This observation has led us to a 'Goldilocks problem' which we will try to solve in the next Chapter by using a least squares approach. Clearly getting the right reconstruction through trial and error method is not convenient and so we aim to develop an adaptive scheme to overcome this problem.

Chapter 3

SCHEME DEVELOPMENT

Our goal in this chapter is to find the right value of p in equation (2.10) in order to get rid of Gibbs' phenomenon and obtain the accurate reconstruction of the function.

3.1 Decay Rate Observation

First we observe the decay rate of the Fourier coefficients for the square wave function, the one that is damped by sufficiently enough Lanczos sigma factors to remove oscillations without smoothing, which was our third case in Section 2.6. So as to see that, we will be plotting the absolute value of the FFT of the function against its frequency domain both in logarithmic scale.

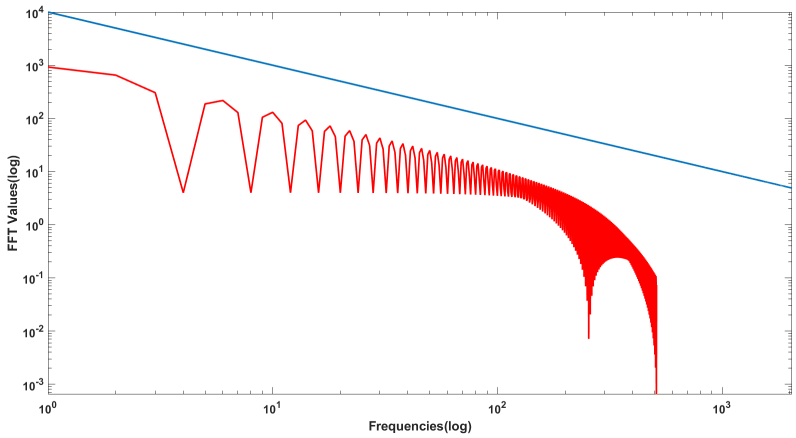

Figure 3.1: Observing decay rate of the low frequency part of the square wave function.

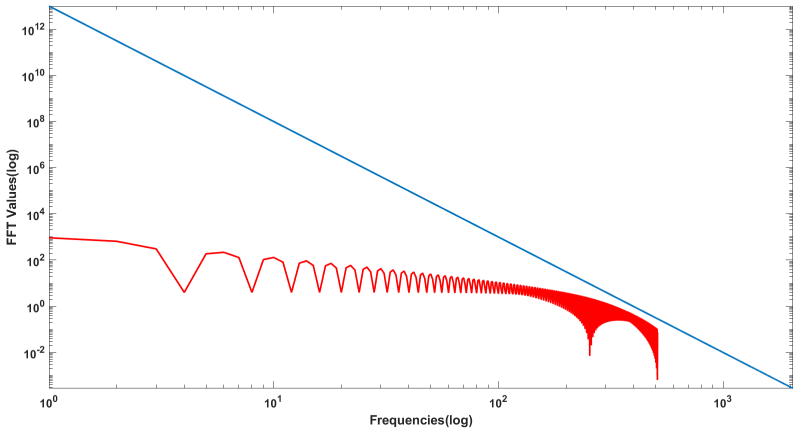

Figure 3.2: Observing decay rate of the high frequency part of the square wave function.

As it can be seen from the above figures, the decay rate for the low frequency part is different from the decay rate for the high frequency part. From our observation, the slope of the blue line parallel to the low frequency part (Figure 3.1) is -1 and that of the high frequency part (Figure 3.2) is -5. This was expected since from equation (2.8) we know that the Fourier coefficients of the high frequency part are supposed to decay faster than that of the low frequency part. Furthermore, we also investigated the decay rates of other functions in similar way and we were left with the same results every time getting a decay rate of 1 and 5 for the low frequency and high frequency part of the Fourier coefficients respectively. This data will be particularly helpful in the construction of our least squares problem in Section 3.3.

3.2 Peak Pattern Observation

Next, we will be taking a closer look at the graphs of the Fourier coefficients of different functions with different number of discontinuities in the frequency domain, again using enough Lanczos σ factors that give the right reconstruction of each of them. These graphs will be zoomed towards the high frequency part. The functions that we will be using are defined as follows, where N is the number of equally spaced points in spatial domain:

$$f1(x) = x^2, \qquad 1 \leq x \leq N \tag{3.1}$$

$$f2(x) = \begin{cases} x+5, & 1 \leq x \leq N/2 \\ x-2\pi+5, & N/2 < x \leq N \end{cases} \tag{3.2}$$

$$f3(x) = \begin{cases} x+5, & 1 \le x \le N/2 \\ x^2, & N/2 < x \le N \end{cases} \tag{3.3}$$

$$f4(x) = \begin{cases} x+5, & 1 \le x \le N/4 \\ x+9, & N/4 < x \le N/2 \\ x-2\pi+5, & N/2 < x \le N \end{cases} \tag{3.4}$$

$$f5(x) = \begin{cases} x+5, & 1 \le x \le N/4 \\ x+9, & N/4 < x \le N/2 \\ x, & N/2 < x \le N \end{cases} \tag{3.5}$$

$$f6(x) = \begin{cases} x+5, & 1 \le x \le N/8 \\ x+9, & N/8 < x \le N/4 \\ x, & N/4 < x \le N/2 \\ x-2\pi+5, & N/2 < x \le N \end{cases} \tag{3.6}$$

$$f7(x) = \begin{cases} x+5, & 1 \le x \le N/8 \\ x+9, & N/8 < x \le N/4 \\ x, & N/4 < x \le N/2 \\ x^2, & N/2 < x \le N \end{cases} \tag{3.7}$$

$$f8(x) = \begin{cases} x+5, & 1 \le x \le N/16 \\ x+9, & N/16 < x \le N/8 \\ x, & N/8 < x \le N/4 \\ x^2, & N/4 < x \le N/2 \\ x-2\pi+5, & N/2 < x \le N \end{cases} \tag{3.8}$$

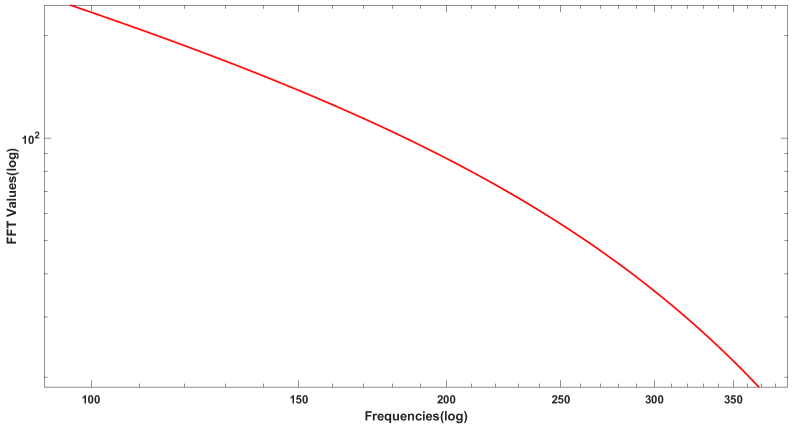

Figure 3.3: Peak pattern for function f1

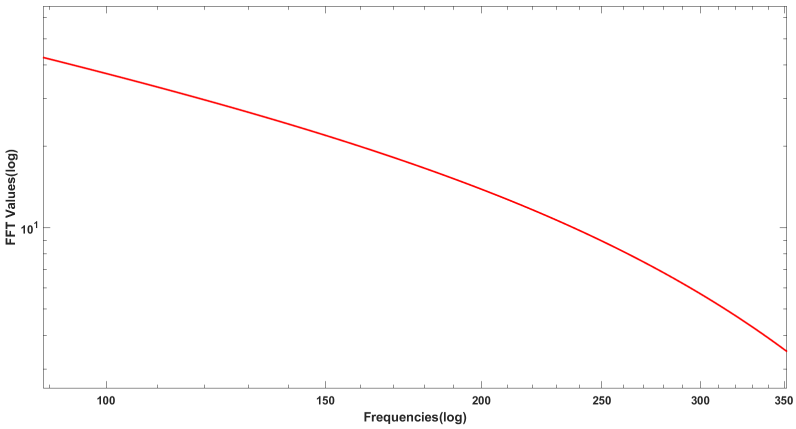

Figure 3.4: Peak pattern for function f2

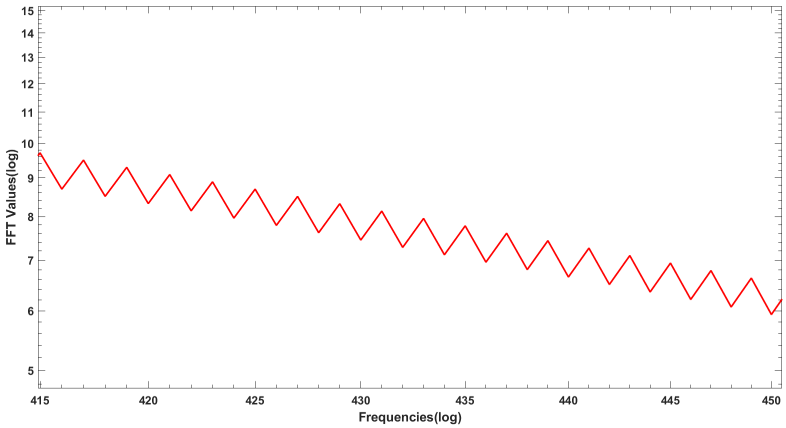

Figure 3.5: Peak pattern for function f3

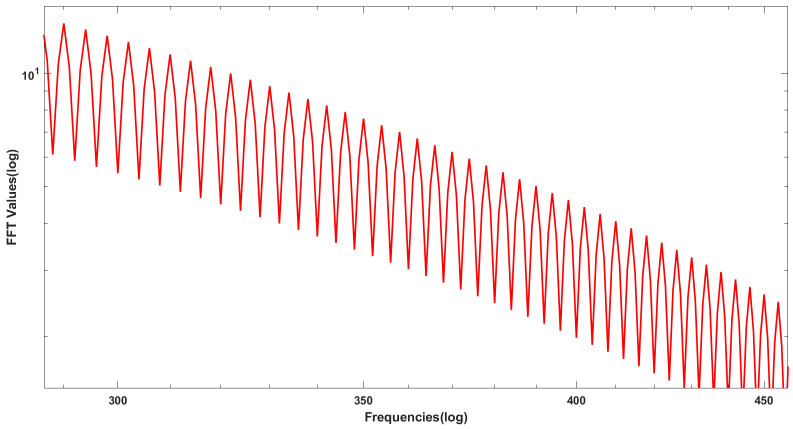

Figure 3.6: Peak pattern for function f4

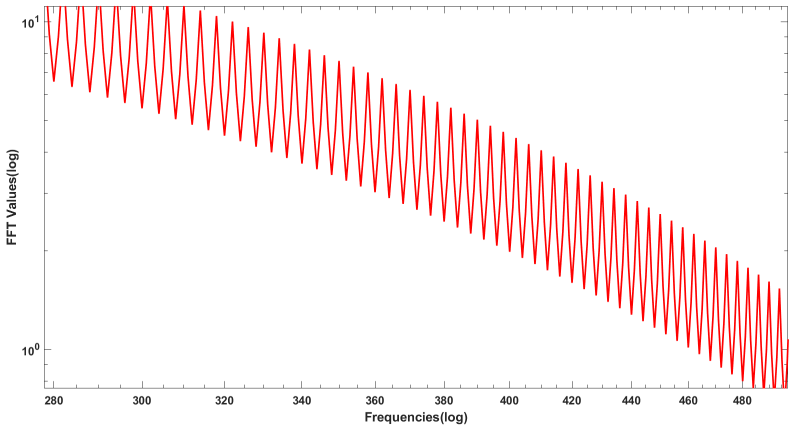

Figure 3.7: Peak pattern for function f5

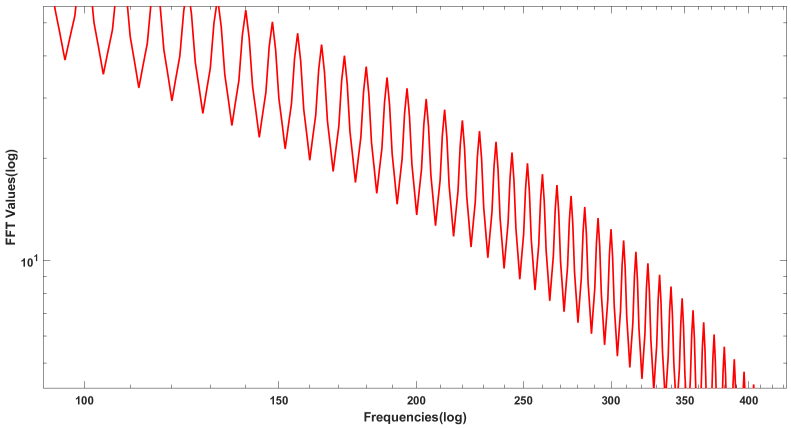

Figure 3.8: Peak pattern for function f6

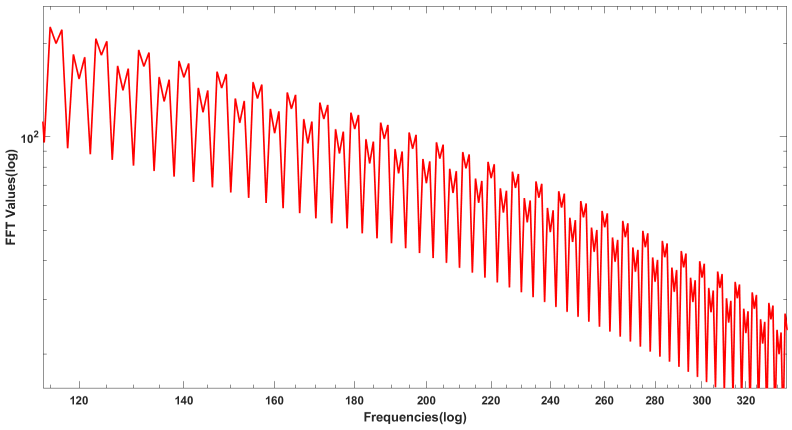

Figure 3.9: Peak pattern for function f7

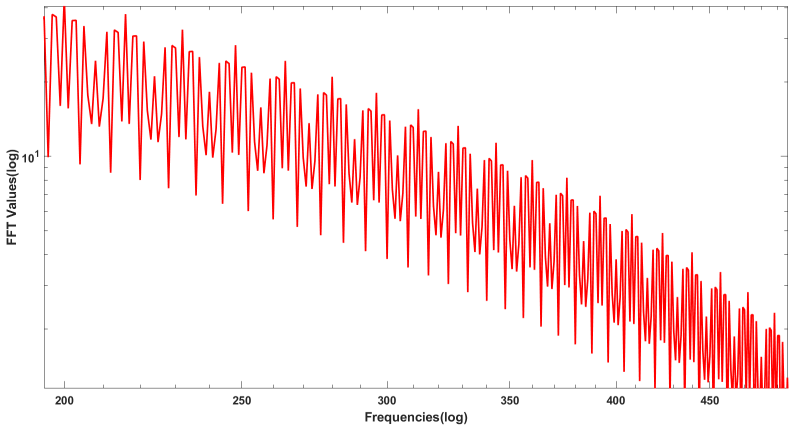

Figure 3.10: Peak pattern for function f8

As we take a closer look at the high frequency parts of the graphs, a clear pattern among the peaks becomes visible for each function. The peaks are uniformly spaced in each graph. Now what we are interested in is that if this spacing between the peaks is related to any certain properties of the functions. Other than the regular discontinuity, there is another type of discontinuity that particularly occurs in some of these functions, that do not have the same value at the end points, because of extending each function in order to make them periodic. For simplicity, we define the discontinuities that are occurring because of unequal function values on the boundaries as exterior discontinuities and the rest of the discontinuities as interior discontinuities. We have been keeping track of the number of both of these types of discontinuities of each function that we observed and the following data successfully led us to establish an important relation between the number of interior discontinuities of each function and the corresponding spacing between the peaks. Here, we denote D, D_e and D_i as the number of total discontinuities, exterior discontinuities and interior discontinuities of the functions respectively.

Table 3.1: Number of Spacing

Function Names	D	D_e	D_i	Spacing
f1	1	1	0	1
f2	1	0	1	1
f3	2	1	1	2
f4	2	0	2	4
f5	3	1	2	4
f6	3	0	3	8
f7	4	1	3	8
f8	4	0	4	16

From the above table it is clear that, if we raise 2 to the power of number of interior discontinuities of the function we get the spacing for the corresponding function. So,

$$spacing = 2^{D_i} \tag{3.9}$$

And based on this we start developing our algorithm to get the right value of p. The only exception to this pattern is for the function f2 that has 1 interior discontinuity but the spacing is 1 instead of 2^1.

Also, for the high frequency part of the function, we are only interested in the Fourier coefficients at the peaks, so in order to get those values we use our MATLAB function `shiftf.m` found in A.10. What this function does is, it takes the required number of spacing for each function and finds the highest FFT value in between each spacing in the frequency domain. This set of highest values are basically the Fourier coefficients at the peaks, indicated by blue circles in the following figure.

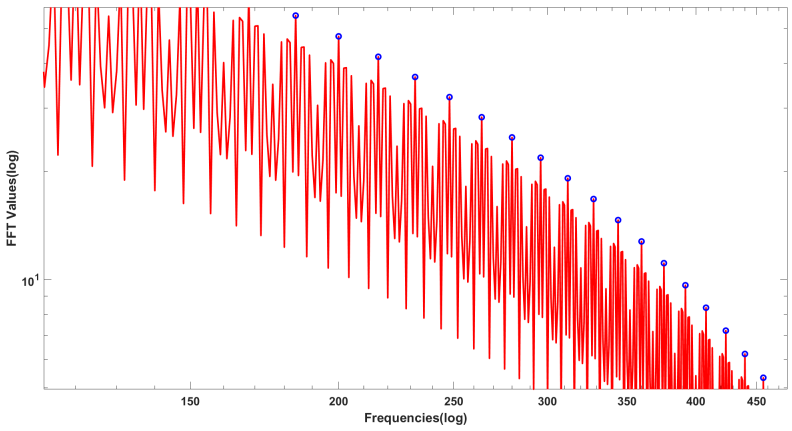

Figure 3.11: Peak pattern for function f8 indicating Fourier coefficients at the peaks

3.3 Least Squares Approach

In this Section we will be developing a least squares problem by first splitting $\hat{f}$ into two parts based on its frequency. Here, $\hat{f}_1$ comes from the low frequency part and $\hat{f}_2$ comes from the high frequency part. As we discussed in the previous section, our FFT of $\hat{f}_2$ will be taking the coefficients at the peaks only.

$$\hat{f} = \begin{pmatrix} \hat{f}_1 \\ \hat{f}_2 \end{pmatrix} \tag{3.10}$$

Based on the fact that the FFT values are inversely propotional to the frequency (Equation:2.8) and using the values of the decay rates that we found in Section 3.1, we can approximate $\hat{f}_1$ and $\hat{f}_2$ using Lanczos σ factors as follows

$$\hat{f}_1.\sigma_1^p \sim c_1\omega_1^{-1} \tag{3.11}$$

and

$$\hat{f}_2.\sigma_2^p \sim c_2\omega_2^{-5} \tag{3.12}$$

where c_1 and c_2 are the constant factors(scalars) and ω_1 and ω_2 are the frequencies of $\hat{f}_1$ and $\hat{f}_2$ respectively. Now taking natural logs of both sides of (3.11) and (3.12) will give us

$$\ln \hat{f}_1 + p\ln\sigma_1 = \ln c_1 - \ln \omega_1 \tag{3.13}$$

and

$$\ln \hat{f}_2 + p\ln\sigma_2 = \ln c_2 - 5\ln \omega_2 \tag{3.14}$$

Rearranging (3.13) and (3.14) and replacing $\ln c_1$ and $\ln c_2$ by all-ones vectors times constants C_1 and C_2 respectively we obtain

$$p\ln\sigma_1 - eC_1 = -\ln \hat{f}_1 - \ln \omega_1 \tag{3.15}$$

and

$$p\ln\sigma_2 - eC_2 = -\ln \hat{f}_2 - 5\ln \omega_2 \tag{3.16}$$

where e is an all-ones vector.

The matrix-vector form of (3.15) and (3.16):

$$\begin{pmatrix} \ln\sigma_1 & -e & 0 \\ \ln\sigma_2 & 0 & -e \end{pmatrix} \begin{bmatrix} p \\ C_1 \\ C_2 \end{bmatrix} = \begin{bmatrix} -\ln \hat{f}_1 - 1\ln \omega_1 \\ -\ln \hat{f}_2 - 5\ln \omega_2 \end{bmatrix}, \tag{3.17}$$

which is a least squares problem. Solving this system (3.17) we can find the desired value of p. Our MATLAB code `solvesigmafft.m` in A.11 is based on the above algorithm that gives us the right value of p depending on the number of interior discontinuities of the function where spacing is determined from using the formula in the equation (3.9).

Chapter 4

RESULTS

4.1 Reconstructed Functions Using Our Scheme

The output for different functions using our MATLAB code `sigmafft.m` found in A.12 gives us the reconstructed function using our scheme, and the value of the power p that comes from the `solvesigmafft.m` code in A.11 after inputting the required spacing value for the given function. The following table is an extension of the Table 3.1, that includes the required p values for each function:

Table 4.1: Powers of σ

Function Names	D	D_e	D_i	Spacing	p
f1	1	1	0	1	28.7451
f2	1	0	1	1	28.7521
f3	2	1	1	2	28.2674
f4	2	0	2	4	27.4469
f5	3	1	2	4	27.4266
f6	3	0	3	8	25.8519
f7	4	1	3	8	25.8147
f8	4	0	4	16	24.5590

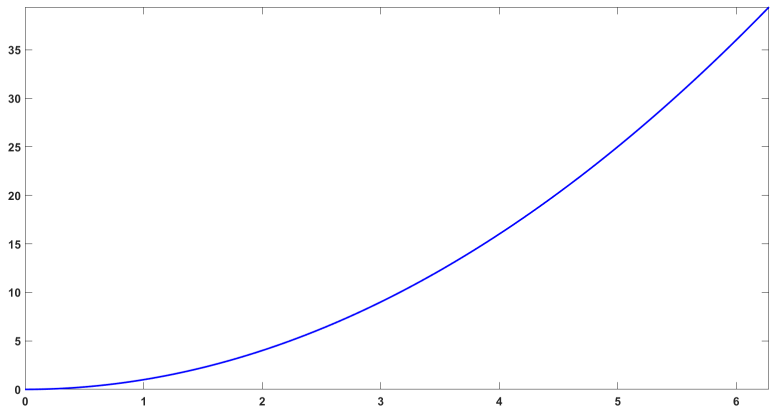

Figure 4.1: Original f1 function

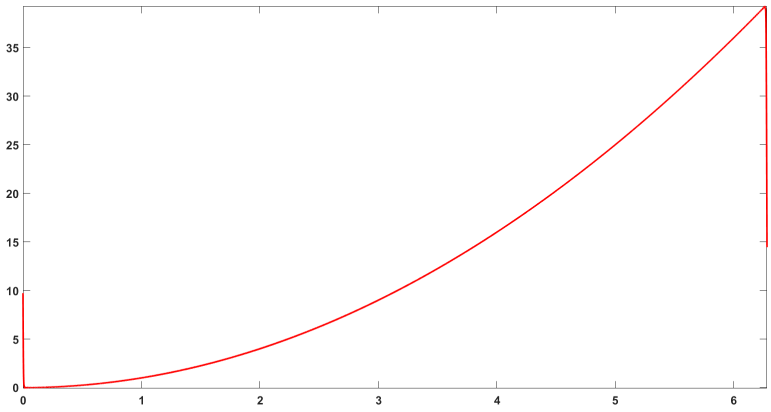

Figure 4.2: Reconstructed f1 function

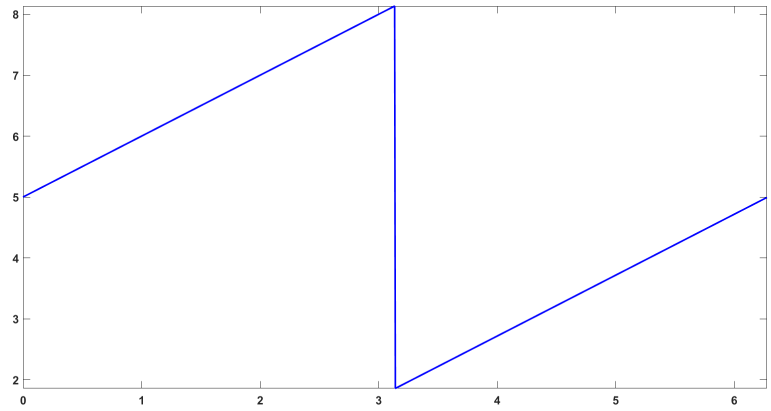

Figure 4.3: Original f2 function

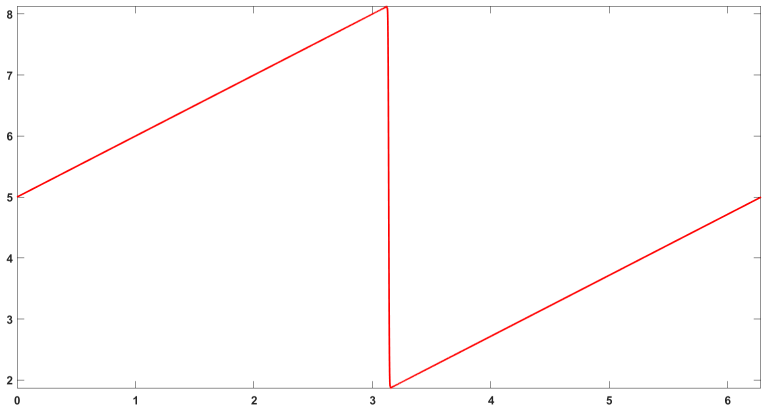

Figure 4.4: Reconstructed f2 function

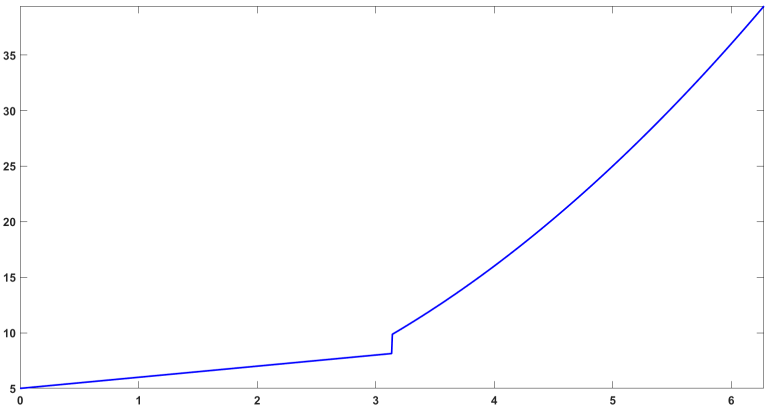

Figure 4.5: Original f3 function

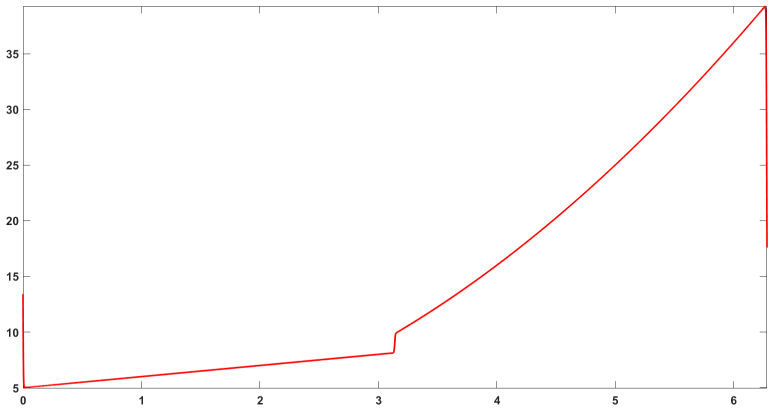

Figure 4.6: Reconstructed f3 function

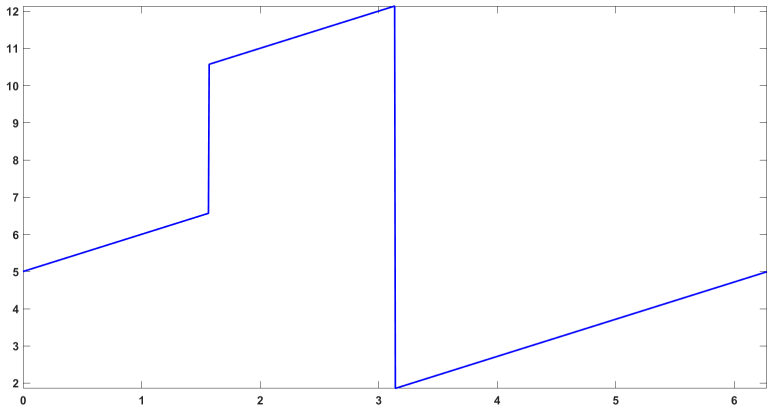

Figure 4.7: Original f4 function

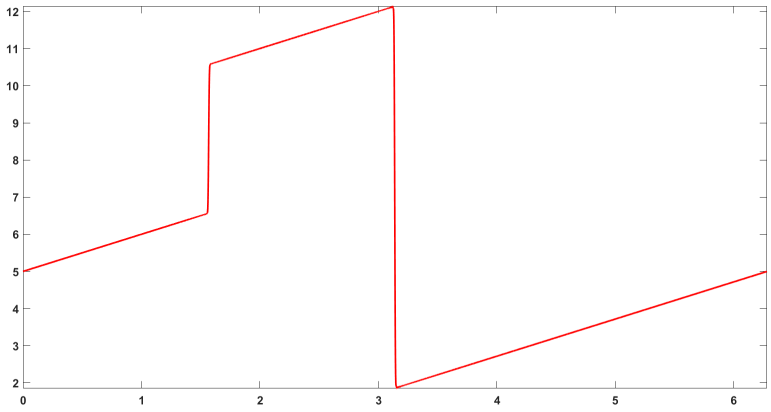

Figure 4.8: Reconstructed f4 function

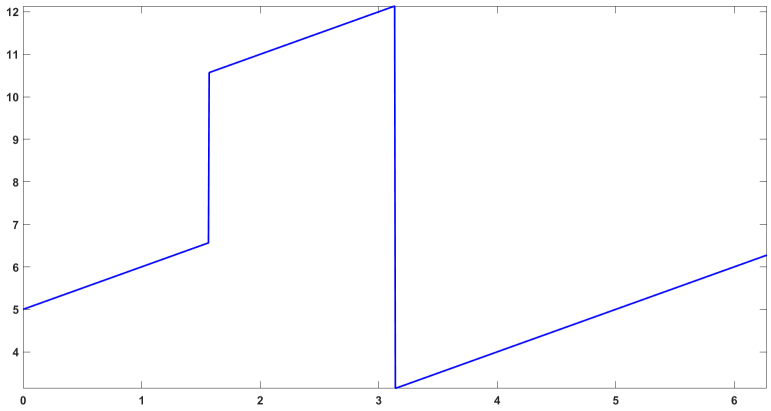

Figure 4.9: Original f1 function

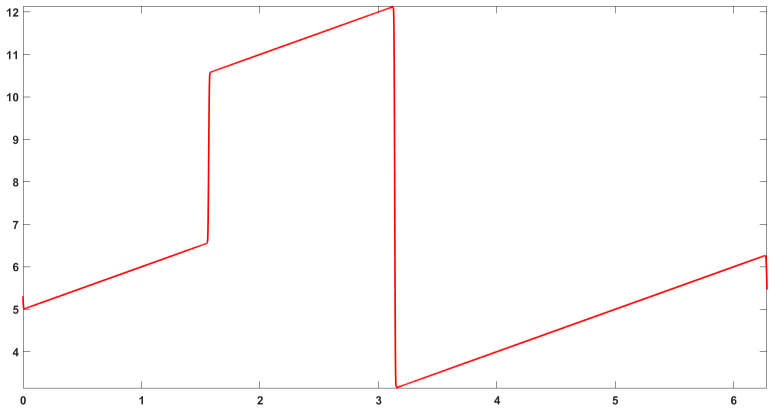

Figure 4.10: Reconstructed f5 function

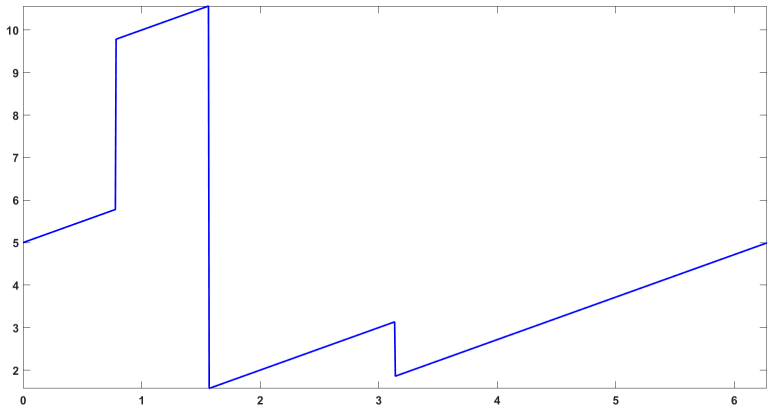

Figure 4.11: Original f6 function

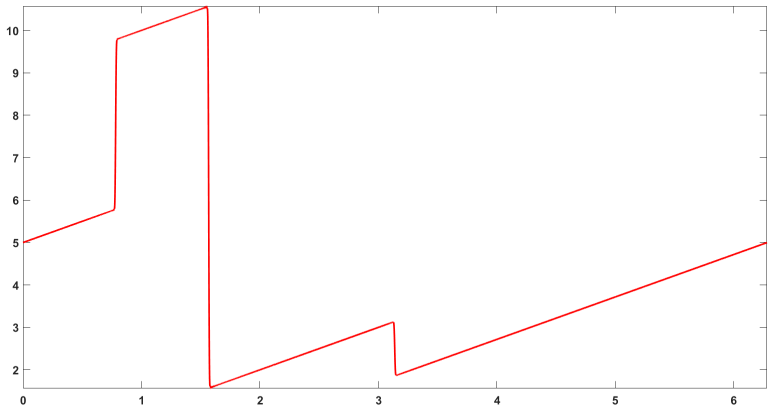

Figure 4.12: Reconstructed f6 function

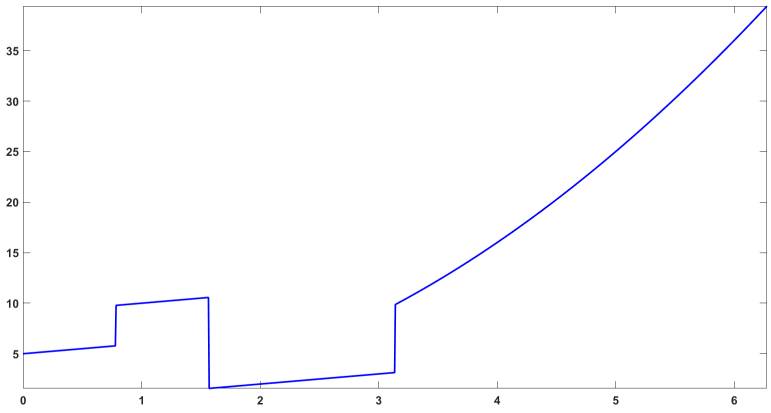

Figure 4.13: Original f7 function

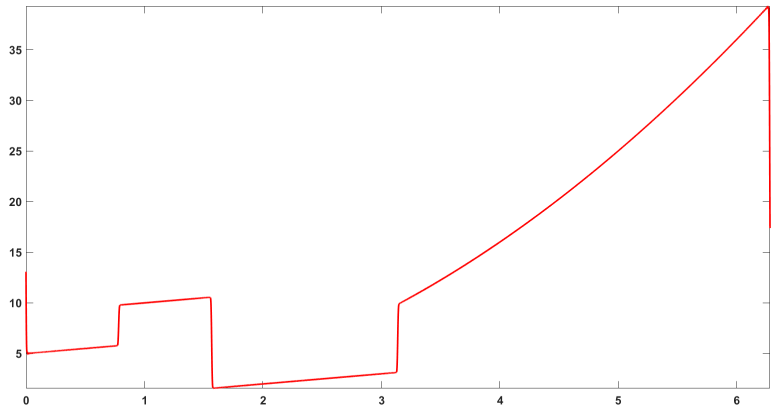

Figure 4.14: Reconstructed f7 function

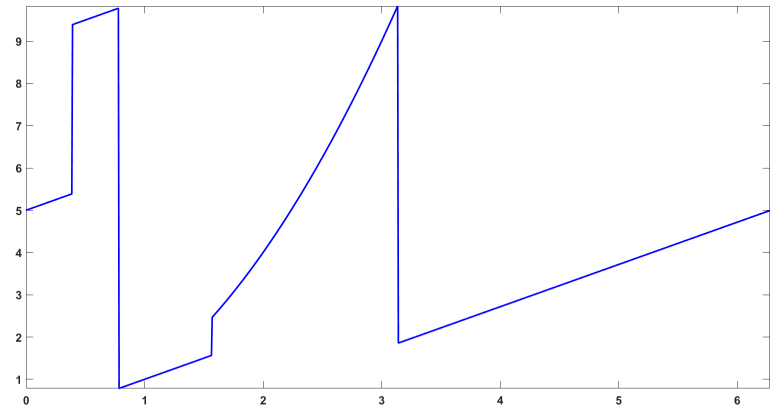

Figure 4.15: Original f8 function

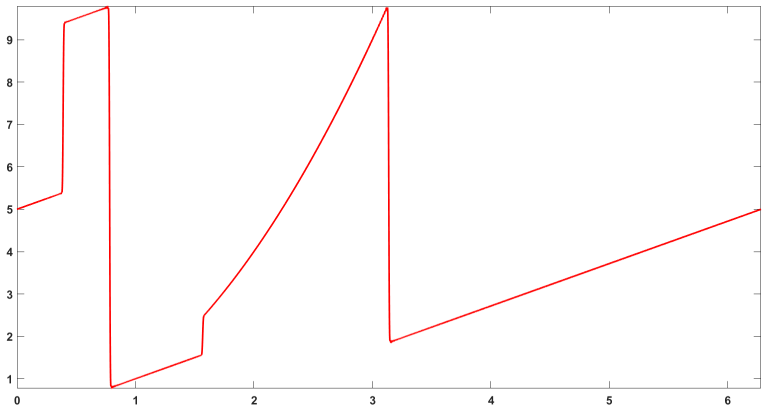

Figure 4.16: Reconstructed f8 function

From all of the graphs above, we see that our scheme has successfully removed Gibbs' oscillations from the reconstructed functions.

4.2 Error and Total Variation Comparison

To calculate errors we use both the 2-norm (Euclidean norm) and 1-norm. The norms are defined as follows:

$$\|x\|_2 = \sqrt{\left(\sum_{i=1}^{n} |\, x_i \,|^2\right)} \tag{4.1}$$

$$\|x\|_1 = \sum_{i=1}^{n} |\, x_i \,| \tag{4.2}$$

where $\|x\|_2$ and $\|x\|_1$ are the 2-norm and 1-norm of the vector x of length n respectively.

In Table 4.2 we are comparing the errors in percentage, calculated before using Sigma approximation (e_b) and after using Sigma approximation with our scheme (e_a), in both 2-norm and 1-norm. The following formula is being used to calculate the errors:

$$Error = \frac{\|f_r - f_{apx}\|}{\|f_r\|} \times 100\% \tag{4.3}$$

where f_r and f_{apx} are the original function and reconstructed function respectively.

Table 4.2: Error Comparison

Function Names	e_b (2-norm)	e_a (2-norm)	e_b (1-norm)	e_a (1-norm)
f1	3.5774%	3.4407%	0.5099%	0.2059%
f2	1.8903%	1.8172%	0.2137%	0.0863%
f3	3.0684%	2.9456%	0.4068%	0.1669%
f4	2.5756%	2.4630%	0.0036%	0.0016%
f5	2.1994%	2.1031%	0.3201%	0.1439%
f6	3.2227%	3.0626%	0.4700%	0.2206%
f7	3.2629%	3.1022%	0.5691%	0.2551%
f8	4.2148%	3.9877%	0.7107%	0.3363%

From the above table it is clear that the errors between the reconstructed functions with and without using σ factors do not vary that much in the Euclidean norm. On the other hand, when the errors are calculated in the 1-norm there are significant differences between the errors of functions with and without using σ factors. In fact, the errors have decreased by almost 50% while using our scheme in the 1-norm. This discrepancy in the behavior of the errors in different norms can be explained from plotting them. To observe that, we graphically represent the errors for the function f4 with and without using σ factors by simply subtracting the reconstructed function from the original function in the spatial domain:

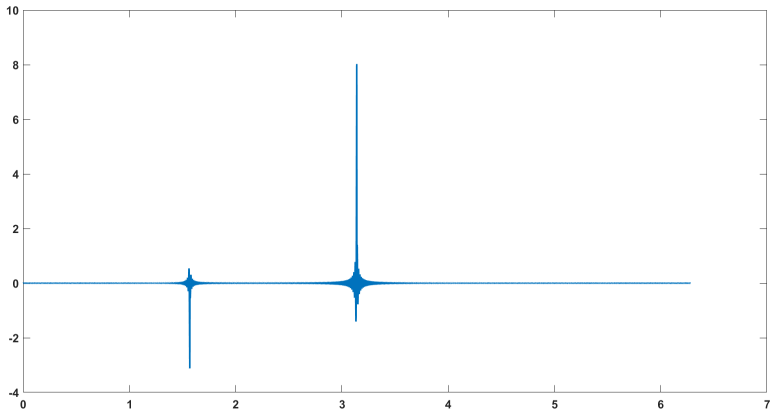

Figure 4.17: Error of function f4 without using Sigma approximation

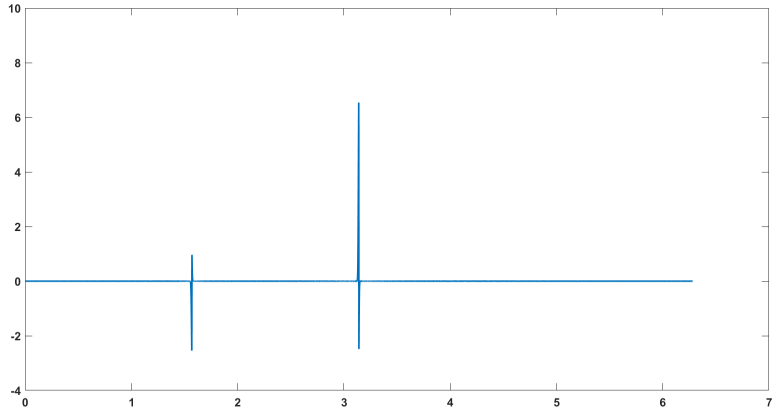

Figure 4.18: Error of function f4 using Sigma approximation

The difference between the above two graphs is quite prominent; the errors are distributed differently across the spatial domain. This actually explains why the errors improve when the 1-norm is being used instead of the 2-norm. Even though in Figure (4.18) the errors are still large at the discontinuities, the errors away from the discontinuities are very small compared to that in Figure (4.17). In fact, the errors away from the discontinuities almost vanished while using σ factors. The reason why the error is still large at the discontinuity is because there is at least a little smoothing, even if not excessive, so the jump in the reconstructed function, while appearing sharp, is not as sharp.

Now to have a better understanding of the difference between the reconstructed functions with and without using our scheme, we calculate the measure of the oscillations of the functions by simply computing the total variation of the functions in the spatial domain. The total variation of a function f over the interval (a,b) is basically the least upper bound of $\sum_{j=1}^{n} \mid f(x_{j+1}) - f(x_j) \mid$ taken over all partitions within the interval (a,b), where n is the total number of partitions. Using our code `sigmafftvariation.m` found in A.14 we get the total variation of the original function, denoted by v_1, total variation of the reconstructed function without σ factors, denoted by v_2 and total variation of the reconstructed functions using our scheme, denoted by v_3 in the following table:

Table 4.3: Total Variation Comparison

Function Names	v_1	v_2	v_3
f1	39.4013	207.8533	75.8488
f2	12.5480	34.3096	12.6900
f3	34.4013	186.5956	65.9719
f4	20.5480	69.0726	21.1573
f5	19.2648	66.8860	20.8430
f6	20.5357	63.5346	21.2170
f7	52.3891	246.9639	85.7230
f8	33.9270	95.7514	35.1775

From Table 4.3 we can conclude that the total variation of the reconstructed functions using our scheme is a lot less than that of the reconstructions without using σ factors. As a matter of fact, the total variation using our scheme is very close to the total variation of the original functions that do not have exterior discontinuities. However, the values are higher for function with exterior discontinuities and the reason behind this is when the discontinuity is on the boundary, the total variation of the original function does not include the change resulting from this discontinuity, but the reconstructed function does since there is still a little smoothing that would lead to some variation.

Chapter 5

CONCLUSIONS

Our research has been successful in developing an adaptive algorithm for a preexisting method that was lacking efficiency. Our algorithm has proven to be effective in removing the Gibbs' oscillations from most places and the reconstructed functions have been quite accurate. In terms of complexity, our method is much simpler than the ones that use an alternative basis to resolve Gibbs' phenomenon such as the post-processing Galerkin approach, used in [17], which deals with 'reprojection' leading to a higher-order reconstruction. Moreover instead of relying on knowing the precise location of the discontinuities, our method only relies on the number of discontinuities.

Although our research was intended to achieve efficiency, further improvement of the algorithm can be directed towards gaining more precision. Future work should also include additional investigation to determine why the only exception to our algorithm occurs for a function with 1 interior discontinuity, and if there are any other exceptions. For continuation of this work, an adaptive scheme with a similar approach can be developed for 2 and 3 dimensional cases.

Appendix A

MATLAB CODES

A.1 f1.m

```
function f=f1(x)
f=x.^2;
end
```

A.2 f2.m

```
function f=f2(x,N)
xa=x(1:N/2);
xb=x((N/2)+1:end);
f=[xa+5;xb-2*pi+5];
end
```

A.3 f3.m

```
function f=f3(x,N)
xa=x(1:N/2);
xb=x((N/2)+1:end);
f=[xa+5;xb.^2];
end
```

A.4 f4.m

```
function f=f4(x,N)
xa=x(1:N/4);
xb=x((N/4)+1:N/2);
xc=x((N/2)+1:end);
```

```
f=[xa+5;xb+9;xc-2*pi+5];
end
```

A.5 f5.m

```
function f=f5(x,N)
xa=x(1:N/4);
xb=x((N/4)+1:N/2);
xc=x((N/2)+1:end);
f=[xa+5;xb+9;xc];
end
```

A.6 f6.m

```
function f=f6(x,N)
xa=x(1:N/8);
xb=x((N/8)+1:N/4);
xc=x((N/4)+1:N/2);
xd=x((N/2)+1:end);
f=[xa+5;xb+9;xc;xd-2*pi+5];
end
```

A.7 f7.m

```
function f=f7(x,N)
xa=x(1:N/8);
xb=x((N/8)+1:N/4);
xc=x((N/4)+1:N/2);
xd=x((N/2)+1:end);
f=[xa+5;xb+9;xc;xd.^2];
end
```

A.8 f8.m

```
function f=f8(x,N)
xa=x(1:N/16);
xb=x((N/16)+1:N/8);
xc=x((N/8)+1:N/4);
xd=x((N/4)+1:N/2);
xe=x((N/2)+1:end);
f=[xa+5;xb+9;xc;xd.^2;xe-2*pi+5];
end
```

A.9 sigmafftdecayrate.m

```
function sigma=sigmafftdecayrate(power,decayrate,scaling,f)
p=10;
N=2^p;
dx=2*pi/N;
x=dx*(0:N-1)';
f=f(x,N);
Tf=fft(f);
q=12;
N2=2^q;
w=[0:N2/2,-N2/2+1:-1];
w=w(2:N2/2);
w1=(10^scaling)*w.^(decayrate);
sigma=rand(N2,1);
for i=1:N2
sigma(i)=abs(sinc((i-N2/2)/(N2/2+1)));
end
sigma=sigma([N2/2:end,1:N2/2-1]);
sigma(1)=1;
Tf2=(N2/N)*sigma.^power.*[Tf(1:N/2);zeros(N2-N+1,1);Tf(N/2+2:N)];
Tf3=Tf2(2:N2/2);
figure(1)
loglog(w,abs(Tf3),'r')
hold on
loglog(w,w1)
axis tight
hold off
end
```

A.10 shiftf.m

```
function shift=shiftf(spacing,f)
p=10;
N=2^p;
f=f(x,N);
Tf=fft(f);
ww=[0:N/2,-N/2+1:-1];
ww=ww(2:N/2);
Tf=Tf(2:N/2);
ss=zeros(1,spacing);
for i=1:spacing
s=abs(Tf(3*N/8+i-1:spacing:N/2-1));
```

```
ss(i)=s(1);
end
[~,shift]=max(ss(:));
shift=shift-1;
end
```

A.11 solvesigmafft.m

```
function power=solvesigmafft(spacing,f)
p=10;
N=2^p;
f=f(x,N);
w=[0:N/2,-N/2+1:-1];
w1=w(2:N/8);
shift=shiftf(spacing);
w2=w(3*N/8+shift:spacing:N/2);
Tf=fft(f);
q=12;
N2=2^q;
sigma=rand(N,1);
for i=1:N2
sigma(i)=abs(sinc((i-N2/2)/(N2/2+1)));
end
sigma=sigma([N2/2:end,1:N2/2-1]);
plot(sigma)
sigma(1)=1;
sigma1=sigma(2:N/8);
sigma2=sigma(3*N/8+shift:spacing:N/2);
F1=abs(Tf(2:N/8));
F2=abs(Tf(3*N/8+shift:spacing:N/2));
m1=length(F1);
e1=ones(m1,1);
z1=zeros(m1,1);
m2=length(F2);
e2=ones(m2,1);
z2=zeros(m2,1);
A=[log(sigma1) -e1 z1;log(sigma2) z2 -e2];
b=[-log(F1)-log((w1)');-log(F2)-5*log((w2)')];
X=A\b;
power=X(1);
end
```

A.12 sigmafft.m

```
function sigma=sigmafft(power,f)
p=10;
N=2^p;
dx=2*pi/N;
x=dx*(0:N-1)';
f=f(x,N);
w=[0:N/2,-N/2+1:-1];
Tf=fft(f);
q=12;
N2=2^q;
dx2=2*pi/N2;
x2=dx2*(0:N2-1);
sigma=rand(N2,1);
```

```
for i=1:N2
sigma(i)=abs(sinc((i-N2/2)/(N2/2+1)));
end
sigma=sigma([N2/2:end,1:N2/2-1]);
sigma(1)=1;
Tf2=(N2/N)*sigma.^power.*[Tf(1:N/2);zeros(N2-N+1,1);Tf(N/2+2:N)];
f2=real(ifft(Tf2));
figure(1)
plot(x,f,'b')
axis tight
figure(2)
plot(x2,f2,'r')
xlim([0 5])
axis tight
end
```

A.13 sigmaffterror.m

```
function [errornormb,errornorma,errornormb1,errornorma1]=sigmaffterror(power.f)
p=10;
N=2^p;
dx=2*pi/N;
x=dx*(0:N-1)';
f=f(x,N);
Tf=fft(f);
q=12;
N2=2^q;
dx2=2*pi/N2;
x2=dx2*(0:N2-1)';
fr=f(x2,N2);
Tfb=(N2/N)*[Tf(1:N/2);zeros(N2-N+1,1);Tf(N/2+2:end)];
fb=ifft(Tfb);
sigma=rand(N2,1);
for i=1:N2
sigma(i)=abs(sinc((i-N2/2)/(N2/2+1)));
end
sigma=sigma([N2/2:end,1:N2/2-1]);
sigma(1)=1;
Tf2=(N2/N)*sigma.^power.*[Tf(1:N/2);zeros(N2-N+1,1);Tf(N/2+2:N)];
fa=real(ifft(Tf2));
errorb=fr-fb;
errora=fr-fa;
errornormb=100*norm(errorb)/norm(fr);
errornorma=100*norm(errora)/norm(fr);
errornormb1=100*norm(errorb,1)/norm(fr,1);
errornorma1=100*norm(errora,1)/norm(fr,1);
figure(1)
plot(x2,errorb)
figure(2)
plot(x2,errora)
end
```

A.14 sigmafftvariation.m

```
function [errornormb,errornorma,errornormb1,errornorma1]=sigmaffterror(power.f)
p=10;
N=2^p;
```

```
dx=2*pi/N;
x=dx*(0:N-1)';
f=f(x,N);
Tf=fft(f);
q=12;
N2=2^q;
dx2=2*pi/N2;
x2=dx2*(0:N2-1)';
fr=f(x2,N2);
Tfb=(N2/N)*[Tf(1:N/2);zeros(N2-N+1,1);Tf(N/2+2:end)];
fb=ifft(Tfb);
sigma=rand(N2,1);
for i=1:N2
sigma(i)=abs(sinc((i-N2/2)/(N2/2+1)));
end
sigma=sigma([N2/2:end,1:N2/2-1]);
sigma(1)=1;
Tf2=(N2/N)*sigma.^power.*[Tf(1:N/2);zeros(N2-N+1,1);Tf(N/2+2:N)];
fa=real(ifft(Tf2));
errorb=fr-fb;
errora=fr-fa;
errornormb=100*norm(errorb)/norm(fr);
errornorma=100*norm(errora)/norm(fr);
errornormb1=100*norm(errorb,1)/norm(fr,1);
errornorma1=100*norm(errora,1)/norm(fr,1);
figure(1)
plot(x2,errorb)
figure(2)
plot(x2,errora)
end
```

BIBLIOGRAPHY

[1] Jerri, A. J.: *The Gibbs phenomenon in Fourier analysis, splines, and wavelet approximations.* Springer (1998)

[2] Wilbraham, Henry: *On a certain periodic function.* The Cambridge and Dublin Mathematical Journal, **3** (1848) 198–201.

[3] GibbS, J.: Fourier's Series. *Nature* **59** (1898) 202.

[4] Gibbs, J.: Fourier's Series. *Nature* **59** (1899) 606.

[5] Bocher, M.: Introduction to the theory of Fourier's series. *Annals of Mathematics* **7** (1906) 81-152.

[6] Korner, T. W.: *Fourier Analysis*. Cambridge University Press (1989) 62-66.

[7] Lin, C.C. and Segel, L. A.: *Mathematics Applied to Deterministic Problems in the Natural Sciences*. Classics in Applied Mathematics **1** SIAM, Philadelphia (1988)

[8] Hänsler, E. and Schmidt, G.: *Acoustic Echo and Noise Control: A Practical Approach*. John Wiley & Sons (2005)

[9] Allen, R. L. and Mills, D.: *Signal Analysis: Time, Frequency, Scale, and Structure*. Wiley-IEEE Press (2004)

[10] Gallagher, T. A., Nemeth, A. J. and Hacein-Bey, L.: An Introduction to the Fourier Transform: Relationship to MRI. *American Journal of Roentgenology* (2008)

[11] Libii, J. N.: Gibbs Phenomenon In Engineering Systems.*Annual Conference, Portland, Oregon, 2005, June*. ASEE Conferences (2005)

[12] Lanczos, C.: *Applied Analysis*. Princeton, New Jersey: Van Nostrand (1956)

[13] Gunawan, G., Harahap, E. and Suwanda: Transformation of the Mean Value of Integral On Fourier Series Expansion. *Journal of Physics: Conference Series* (2009)

[14] Lambers, J. V. and Sumner, A. C.: *Explorations in Numerical Analysis*. World Scientific, New Jersey (2018)

[15] Gustafsson, B., Kreiss, H.-O. and Oliger, J. E.: *Time-Dependent Problems and Difference Methods*. John Wileym & Sons, New York (1995)

[16] Hamming, R. W.: *Numerical Methods for Scientists and Engineers.* **2** New York: Dover (1986) 534-536.

[17] Gottlieb, S., Jung, J.-H. and Kim, S. : A Review of David Gottlieb's Work on the Resolution of the Gibbs Phenomenon. *Communications in Computational Physics* **9**(3) (2011) 497-519.