**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра математического обеспечения и применения ЭВМ**

**отчет**

**по лабораторной работе №3**

**по дисциплине «Цифровая обработка сигналов»**

**Тема: Исследование результатов фильтрации дискретного сигнала**

|  |  |  |
| --- | --- | --- |
| Студент гр. 7383 |  | Прокопенко Н. |
| Студент гр. 7383 |  | Ханова Ю.А. |
| Преподаватель |  | Середа А.-В. И. |

Санкт-Петербург

2020

**Цель работы.**

Получение практических навыков выполнения фильтрации дискретных последовательностей с помощью рекурсивных и нерекурсивных фильтров, а также анализа получаемых результатов с помощью дискретного преобразования Фурье.

**Основные теоретические положения.**

Лабораторная работа потребует знаний:

* в области дискретизации непрерывного сигнала;
* фильтрации дискретного сигнала с помощью дискретных нерекурсивных и рекурсивных фильтров;
* дискретного преобразования Фурье (ДПФ) для дискретных последовательностей;

и умений:

* в организации вычислительных процессов;
* в проведении компьютерных расчетов с визуализацией получаемых результатов;
* проведения анализа полученных результатов и формулировка выводов.

**Постановка задачи.**

Для заданного дискретного сигнала применить соответствующие фильтры. Полученные результаты содержательно проинтерпретировать.

**Порядок выполнения работы.**

1. Сформировать дискретный сигнал посредством дискретизации с шагом непрерывного сигнала, представляющего собой линейную комбинацию косинусоид вида . Частота каждой из гармоник не должна превышать π. Всего одиннадцать гармоник с упорядоченными по возрастанию частотами от 0 до π, изменяющимися с шагом . Амплитуды гармоник представляют собой целые числа со значениями от до , определяемые случайным образом с помощью датчика равномерно распределенных случайных чисел. При необходимости нормализовать коэффициенты линейной комбинации посредством деления их на сумму полученных случайным образом амплитуд. Начальные фазы представляют собой случайные числа в промежутке от до . Дискретная последовательность должна включать в себя отсчета ().
2. Визуализировать исходные аналоговый и дискретизированный сигналы.
3. С помощью ДПФ найти дискретные отсчеты спектра дискретного сигнала и визуализировать их.
4. Для дискретного сигнала применить линейное сглаживание по -ти и -ти точкам, представить формулу для – передаточной функции (частотной характеристики) фильтра.
5. Визуализировать полученный после фильтрации дискретный сигнал совместно с исходным дискретным сигналом.
6. С помощью ДПФ найти дискретные отсчеты спектра дискретного сигнала после его фильтрации и визуализировать их совместно с отчетами спектра исходного дискретного сигнала.
7. Проанализировать результат на соответствие значениям . Сделать выводы.
8. Повторить пункты 4-7 для следующих фильтров:
   1. Сглаживание полиномом 2-ой степени по 5 и 9 узлам.
   2. Сглаживание полиномом 4-ой степени по 7 и 11 узлам.
   3. Дискретный фильтр, соответствующий численному дифференцированию 1-го порядка.
   4. Дискретный фильтр, соответствующий численному интегрированию (прямоугольников, трапеций, Симпсона).
9. Содержательно проинтерпретировать результаты выполнения лабораторной работы, сделать выводы.

**Ход работы.**

Для произведения вычислений и построения графиков была написана программа на языке Python. Код программы представлен в приложении А.

Был сформирован дискретный сигнал посредством дискретизации с шагом непрерывно го сигнала, представляющего собой линейную комбинацию косинусоид вида . Частота каждой из гармоник не должна превышать π. Всего одиннадцать гармоник с упорядоченными по возрастанию частотами от 0 до π, изменяющимися с шагом . Амплитуды гармоник представляют собой целые числа со значениями от до , определяемые случайным образом с помощью датчика равномерно распределенных случайных чисел. При необходимости нормализовать коэффициенты линейной комбинации посредством деления их на сумму полученных случайным образом амплитуд. Начальные фазы представляют собой случайные числа в промежутке от до . Дискретная последовательность должна включать в себя отсчета ().

Визуализируем исходные аналоговый и дискретизированный сигналы. Полученные графики представлен на рис. 1 и рис. 2 . На данном графике видно, что данный сигнал является периодическим с периодом , т.е.:

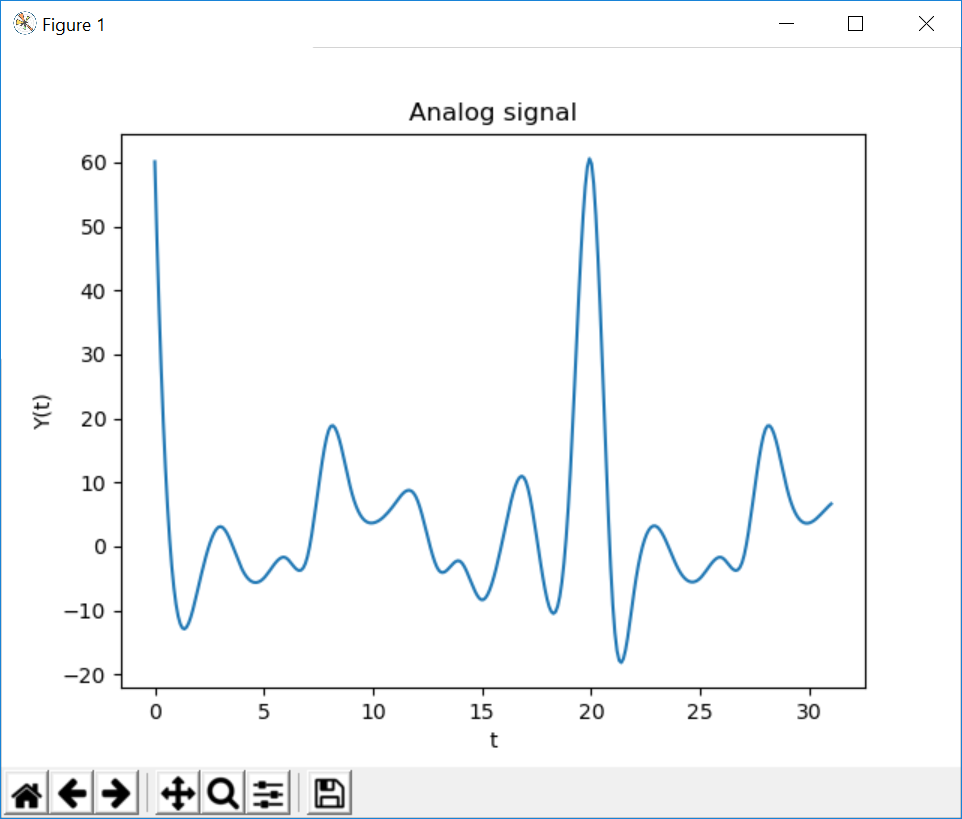


Рисунок 1 – Визуализированный аналоговый сигнал.



Рисунок 2 – Визуализированный дискретный сигнал

С помощью ДПФ найдём дискретные отсчеты спектра дискретного сигнала и визуализируем их. Рассмотрим фрагмент последовательности из отсчётов, например . Тогда дискретная функция:

тоже будет периодической, с периодом . Здесь – период дискретизации. Спектр также должен быть периодическим (с периодом ) и дискретным расстоянием между гармониками .

Поскольку периодическая функция, её можно разложить в ряд Фурье. Следующее выражение называется дискретным преобразованием Фурье:

Посчитаем дискретные отсчёты спектра и визуализируем их. Полученные графики дискретных отсчётов спектра дискретного сигнала для вещественной и мнимой частей представлены на рис. 3-4.

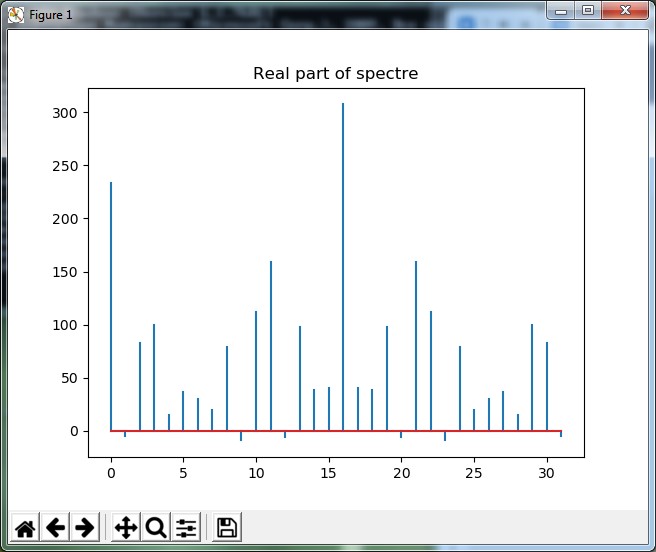


Рисунок 3 – Дискретные отсчёты спектра дискретного сигнала, вещественная часть.

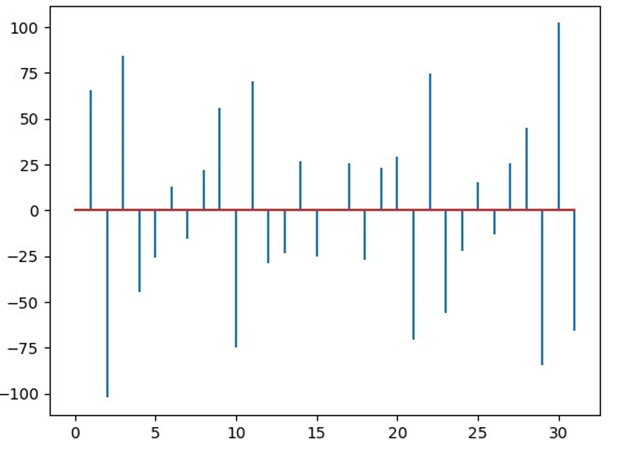


Рисунок 4 – Дискретные отсчёты спектра дискретного сигнала, мнимая часть.

Для дискретного сигнала применим линейное сглаживание по -ти и -ти точкам, представим формулу для – передаточной функции (частотной характеристики) фильтра.

Передаточные функции фильтра для 5 и 9 точек:

По данным передаточным функциям выведем формулу для линейного сглаживания по 5 точкам и 9 точкам:

Визуализируем полученный после фильтрации дискретный сигнал совместно с исходным дискретным сигналом. Полученный график представлен на рис. 5, где синим маркером представлен исходный график, а красным – после фильтрации.

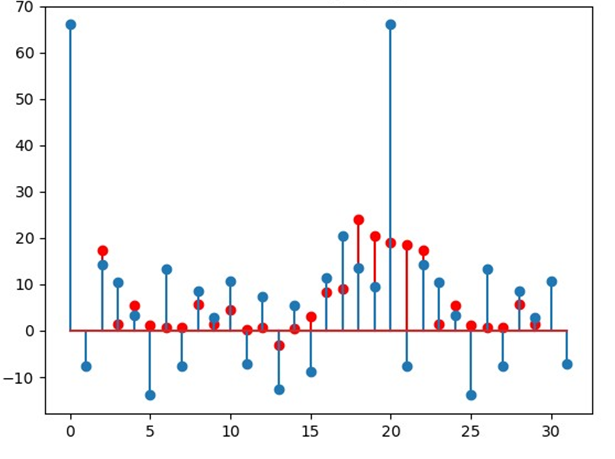


Рисунок 5 – Сглаженный полиномом по 5 точкам

С помощью ДПФ найдём дискретные отсчеты спектра дискретного сигнала после его фильтрации. Полученные графики представлены на рис. 6-7.

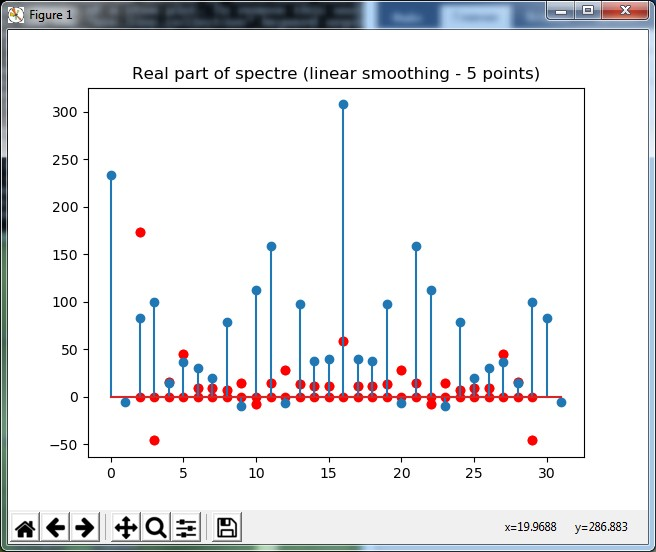


Рисунок 6 – Дискретные отсчёты спектра после фильтрации, вещественная часть.

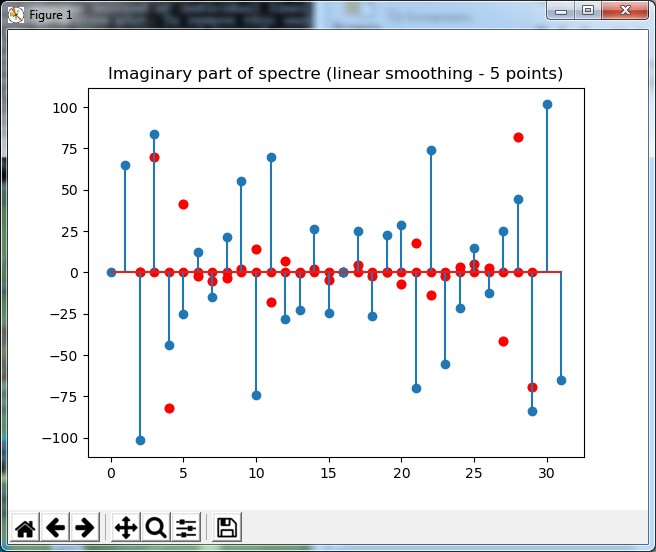


Рисунок 7 – Дискретные отсчёты спектра после фильтрации, мнимая часть

Аналогичные действия сделаем для 9 точек. Результаты представлены на рис. 8-10.

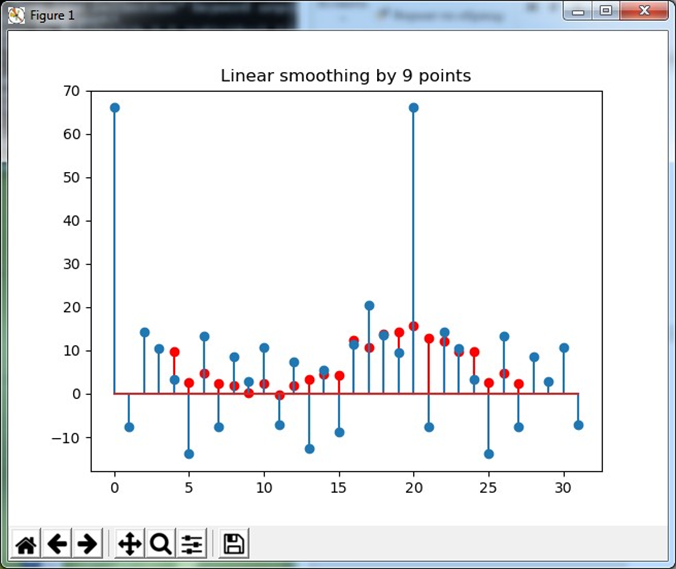


Рисунок 8 – Сглаженный полиномом первой степени по 9 точкам

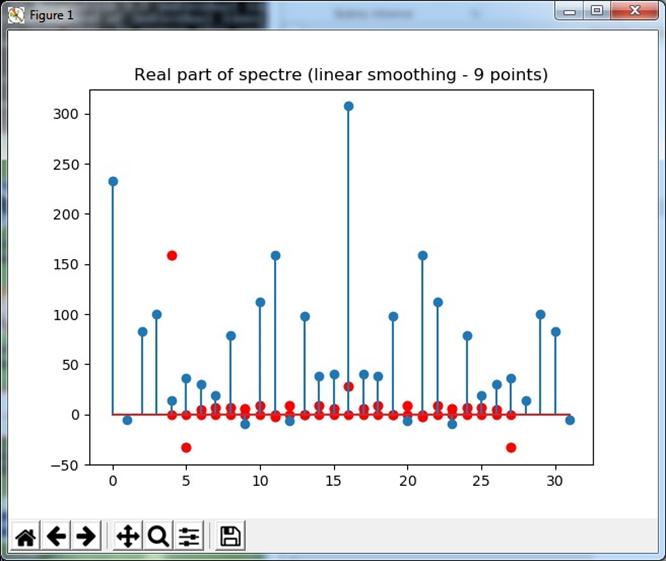


Рисунок 9 – Дискретные отсчёты спектра после фильтрации, вещественная часть

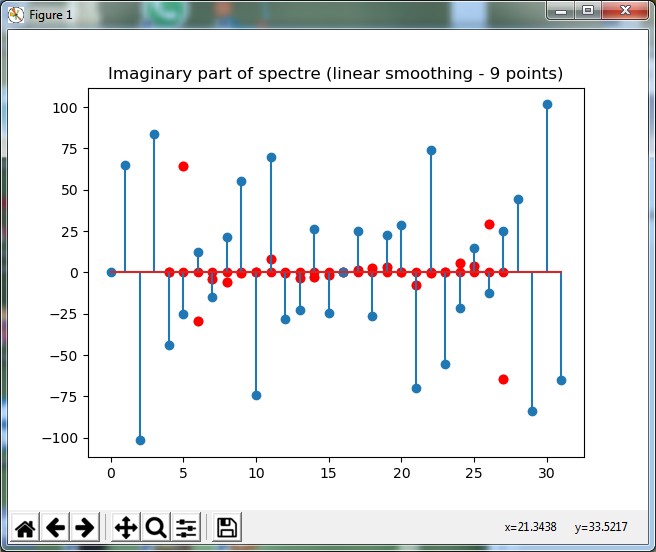


Рисунок 10 – Дискретные отсчёты спектра после фильтрации, мнимая часть

Для дискретного сигнала применим сглаживание полиномом второй степени по и точкам:

Представим формулу для – передаточной функции (частотной характеристики) фильтра:

Визуализируем полученный после фильтрации дискретный сигнал совместно с исходным дискретным сигналом, полученные графики представлены на рис. 11-12.

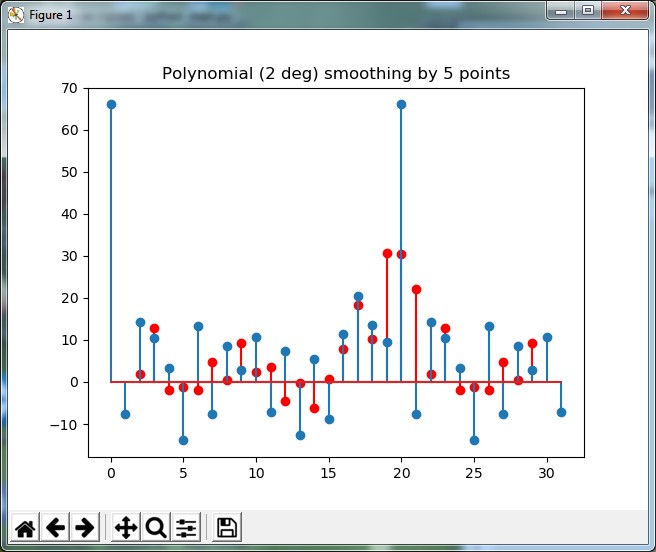


Рисунок 11 – Сглаженный полиномом второй степени по 5 точкам

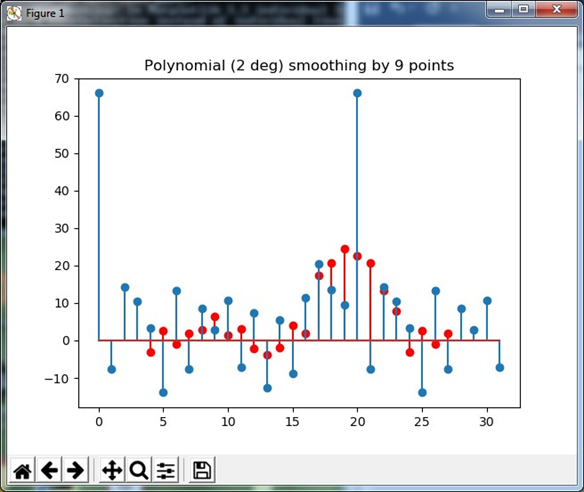


Рисунок 12 – Сглаженный полиномом второй степени по 5 точкам

С помощью ДПФ найдём дискретные отсчеты спектра дискретного сигнала после его фильтрации и визуализируем их совместно с отсчетами спектра исходного дискретного сигнала полученные графики представлены на рис. 13-16.

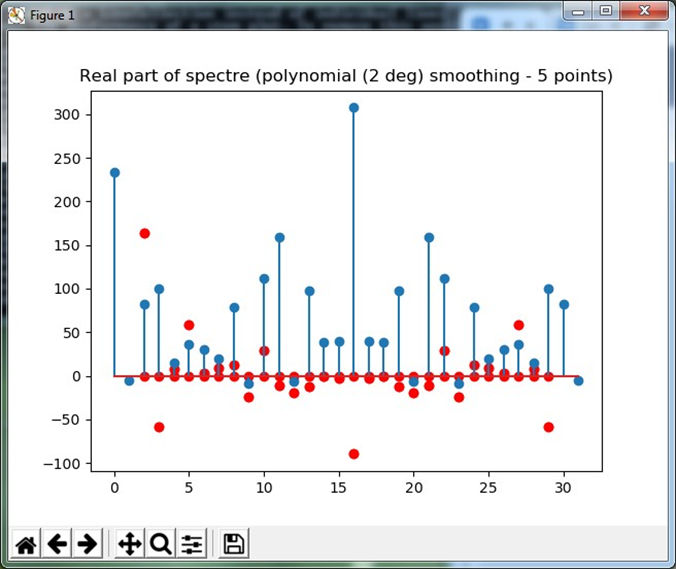


Рисунок 13 – Дискретные отсчёты спектра после фильтрации, вещественная часть (5 точек)



Рисунок 14 – Дискретные отсчёты спектра после фильтрации, мнимая часть (5 точек)

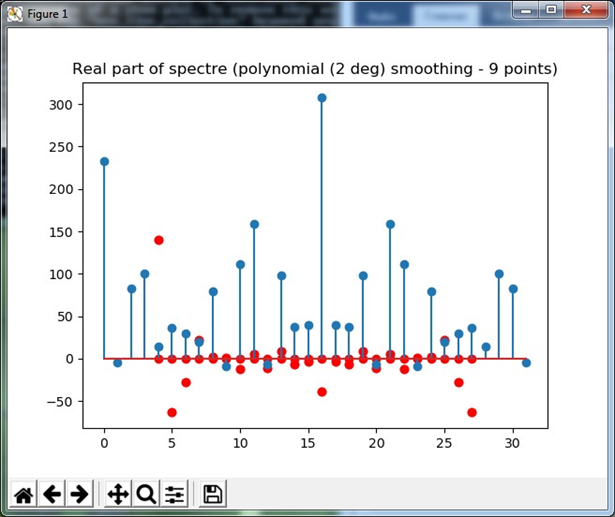


Рисунок 15 – Дискретные отсчёты спектра после фильтрации, вещественная часть (9 точек)

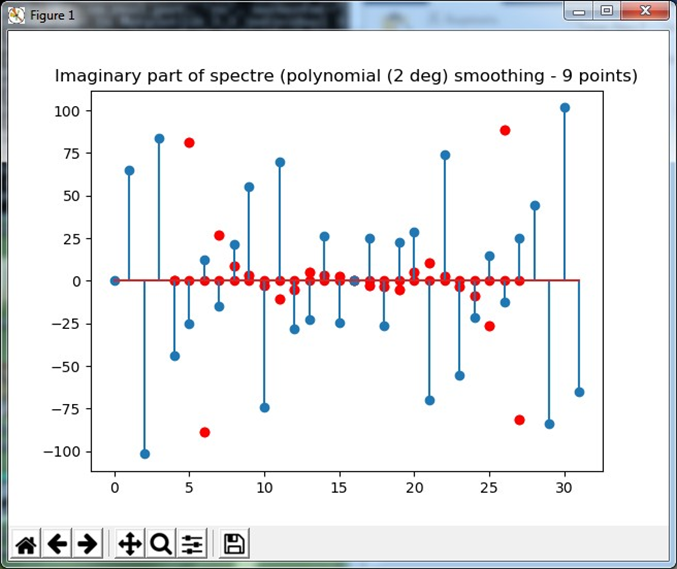


Рисунок 16 – Дискретные отсчёты спектра после фильтрации, мнимая часть (9 точек)

Для дискретного сигнала применим сглаживание полиномом четвёртой степени по и узлам:

Представим формулу для – передаточной функции (частотной характеристики) фильтра:

Визуализируем полученный после фильтрации дискретный сигнал совместно с исходным дискретным сигналом. Результаты представлены на рис. 17-18.

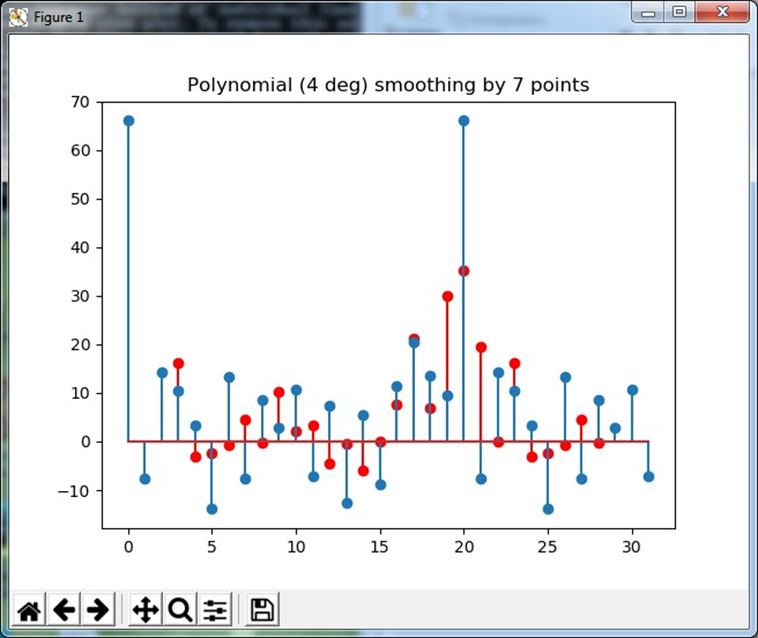


Рисунок 17 – Сглаженный полиномом четвертой степени по 7 точкам

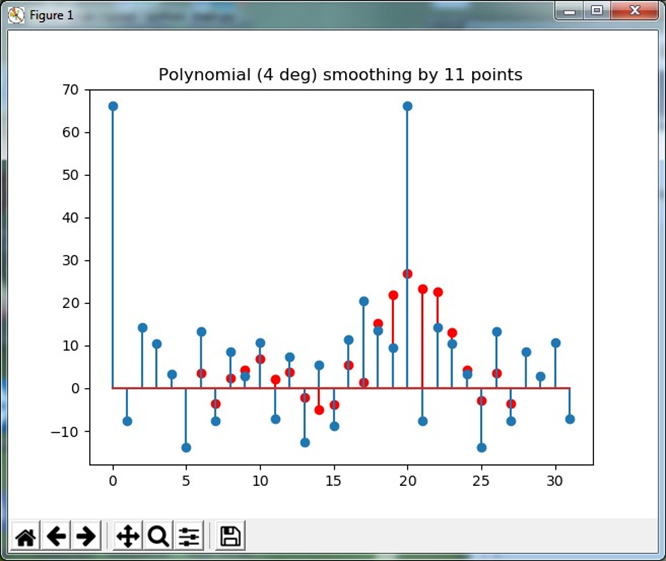


Рисунок 18 – Сглаженный полиномом четвертой степени по 11 точкам

С помощью ДПФ найдём дискретные отсчеты спектра дискретного сигнала после его фильтрации и визуализируем их совместно с отсчетами спектра исходного дискретного сигнала, результаты представлены на рис. 19- 22.

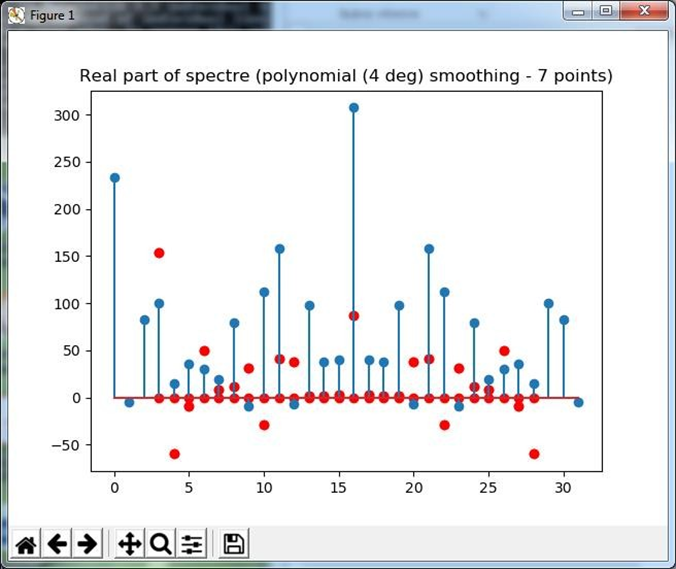


Рисунок 19 – Дискретные отсчёты спектра после фильтрации, вещественная часть (7 точек)

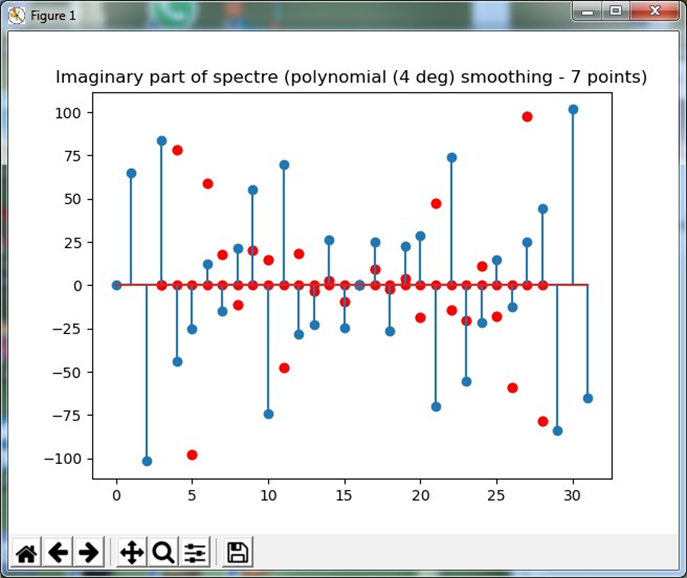


Рисунок 20 – Дискретные отсчёты спектра после фильтрации, мнимая часть (7 точек)

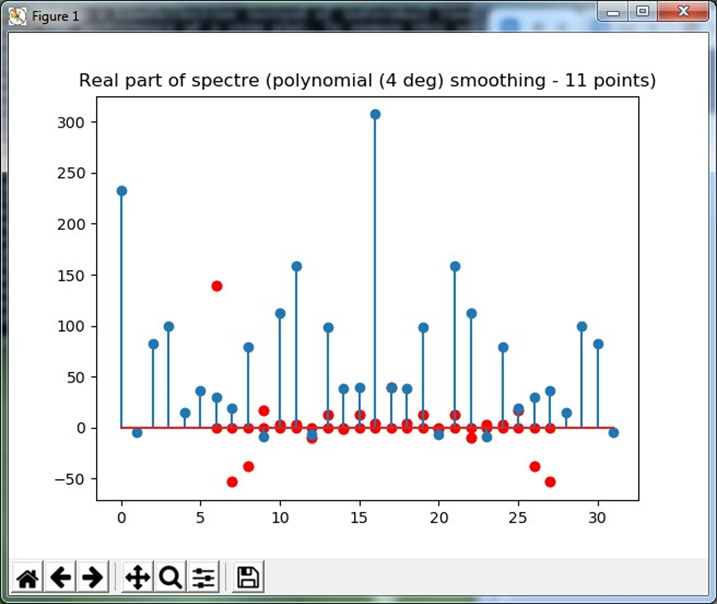


Рисунок 21 – Дискретные отсчёты спектра после фильтрации, вещественная часть (11 точек)

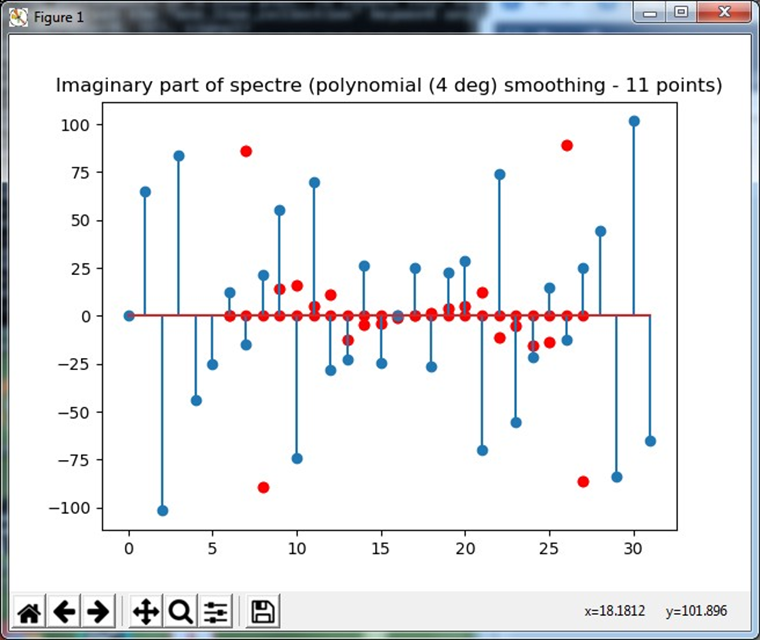


Рисунок 22 – Дискретные отсчёты спектра после фильтрации, мнимая часть (11 точек)

Для дискретного сигнала применим дискретный фильтр, соответствующий численному дифференцированию 1-го порядка:

Представим формулу для – передаточной функции (частотной характеристики) фильтра:

Визуализируем полученный после фильтрации дискретный сигнал совместно с исходным дискретным сигналом, полученный график представлен на рис. 23.

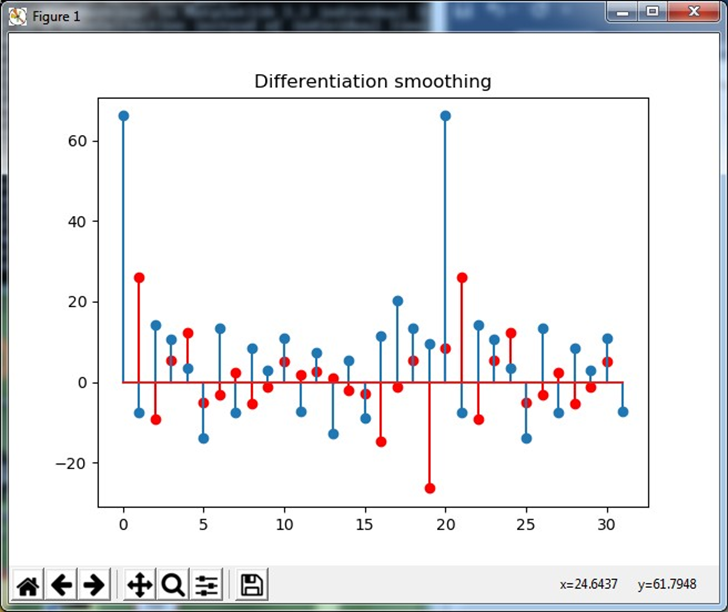


Рисунок 23 – Дискретный сигнал после применения фильтра, соответствующего численному дифференцированию первого порядка

С помощью ДПФ найдём дискретные отсчеты спектра дискретного сигнала после его фильтрации и визуализируем их совместно с отсчетами спектра исходного дискретного сигнала полученные графики представлены на рис. 24-25.

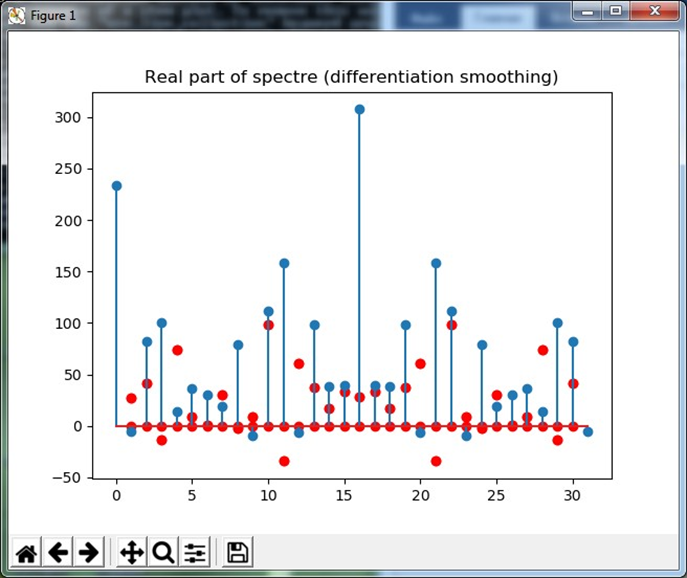


Рисунок 24 – Дискретные отсчёты спектра после применения фильтра соответствующего численному дифференцированию, вещественная часть

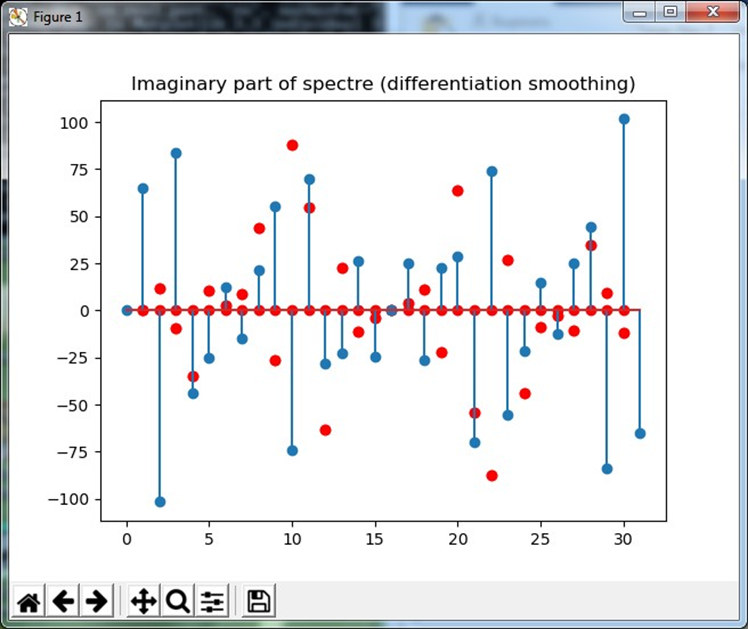


Рисунок 25 – Дискретные отсчёты спектра после применения фильтра соответствующего численному дифференцированию, мнимая часть

Для дискретного сигнала применим дискретный фильтр, соответствующий численному интегрированию (средних прямоугольников , трапеций , Симпсона ):

Представим формулу для – передаточной функции (частотной характеристики) фильтра.

Визуализируем полученные после фильтрации дискретные сигналы совместно с исходным дискретным сигналом, полученный результат представлен на рис. 26-28.

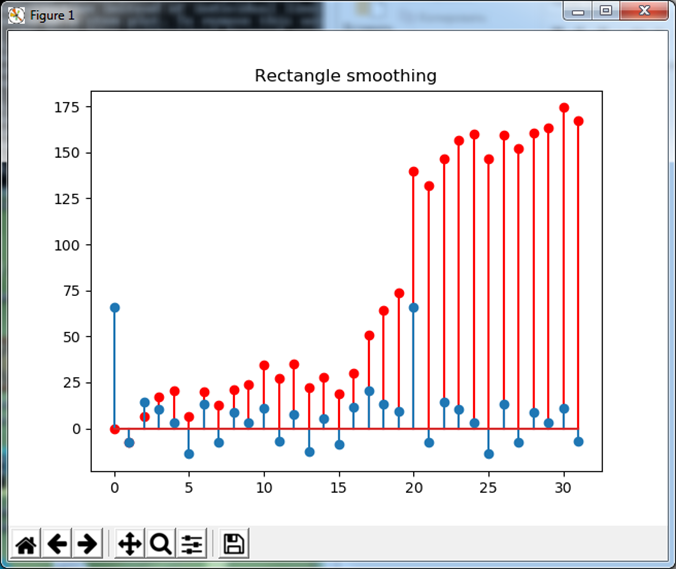


Рисунок 26 – Дискретный сигнал после применения к нему фильтров, соответствующих численному интегрированию (формула средних

прямоугольников)

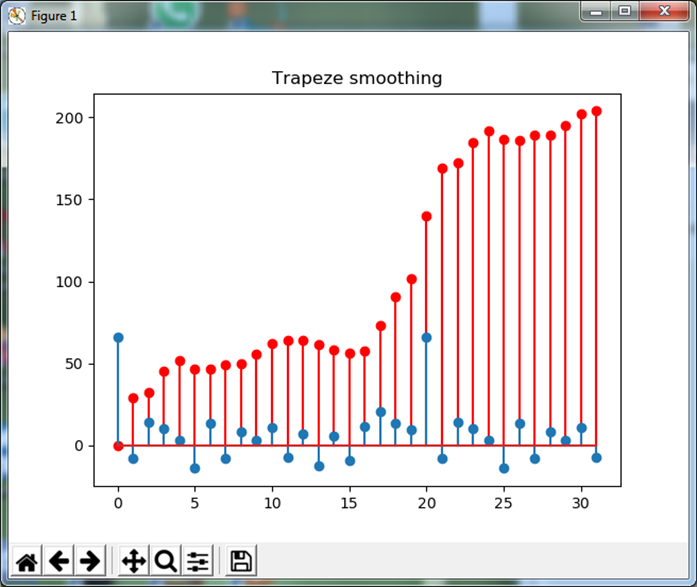


Рисунок 27 – Дискретный сигнал после применения к нему фильтров, соответствующих численному интегрированию (формула трапеций)

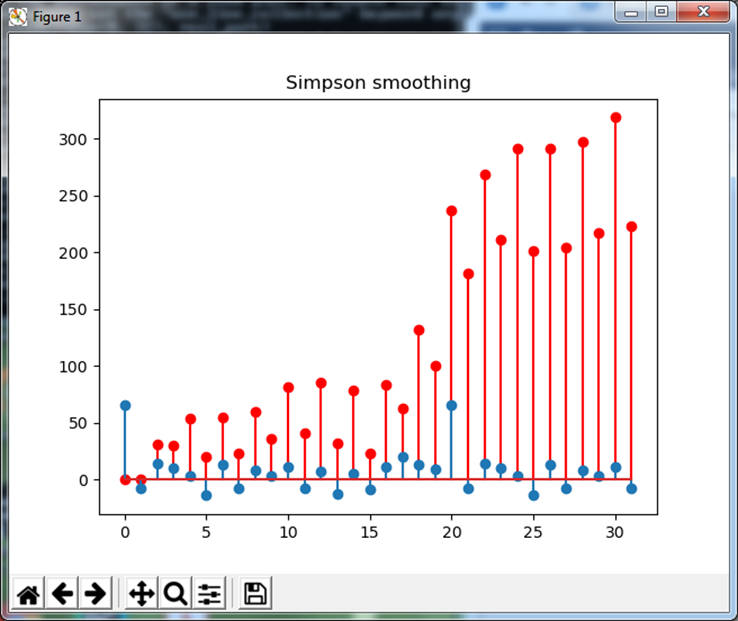


Рисунок 28 – Дискретный сигнал после применения к нему фильтров, соответствующих численному интегрированию (формула Симпсона)

С помощью ДПФ найдём дискретные отсчеты спектра дискретного сигнала после его фильтрации и визуализируем их совместно с отсчетами спектра исходного дискретного сигнала. Результаты представлены на рис. 29- 34.

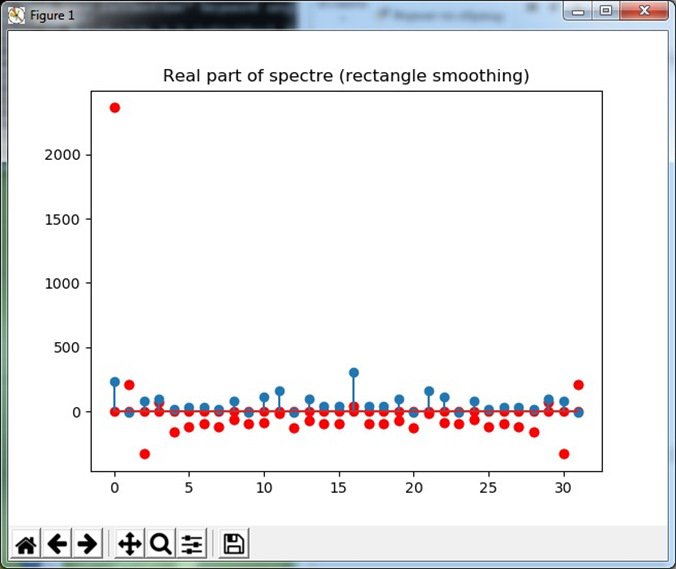


Рисунок 29 – Дискретные отсчёты спектра после фильтрации, вещественная часть (формула средних прямоугольников)

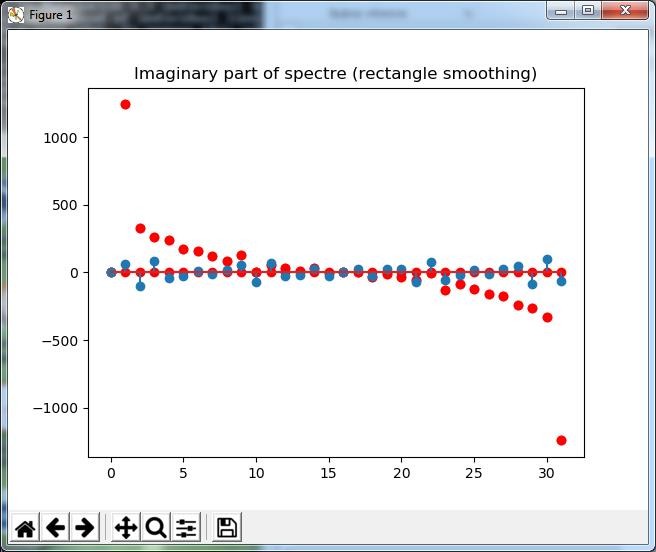


Рисунок 30 – Дискретные отсчёты спектра после фильтрации, мнимая часть (формула средних прямоугольников)

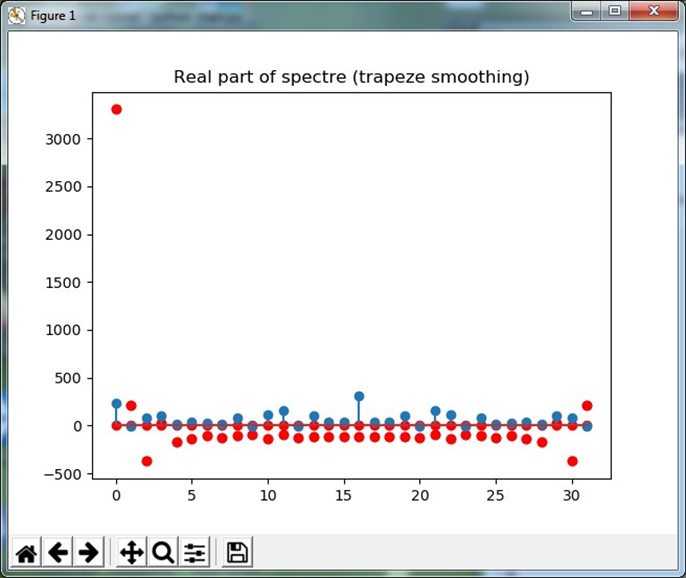


Рисунок 31 – Дискретные отсчёты спектра после фильтрации, вещественная часть (формула трапеций)

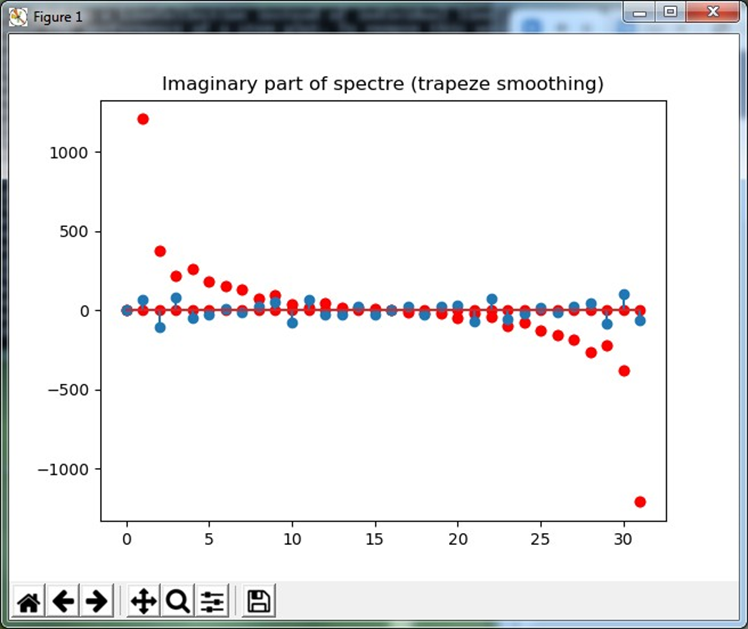


Рисунок 32 – Дискретные отсчёты спектра после фильтрации, мнимая часть (формула трапеций)

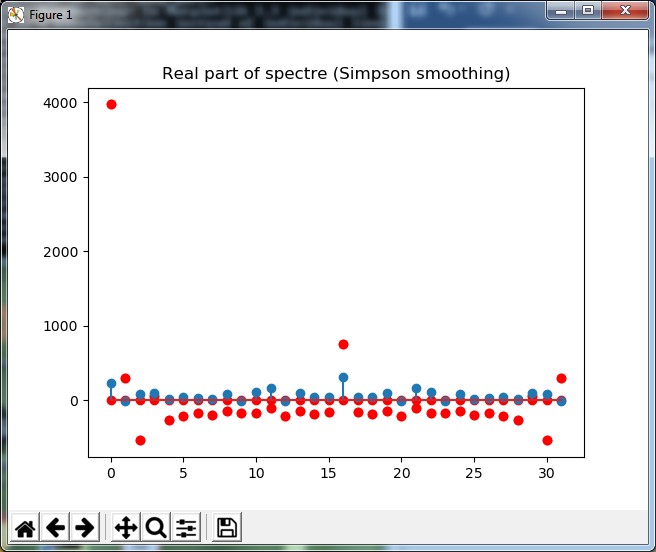


Рисунок 33 – Дискретные отсчёты спектра после фильтрации, вещественная часть (формула Симпсона)

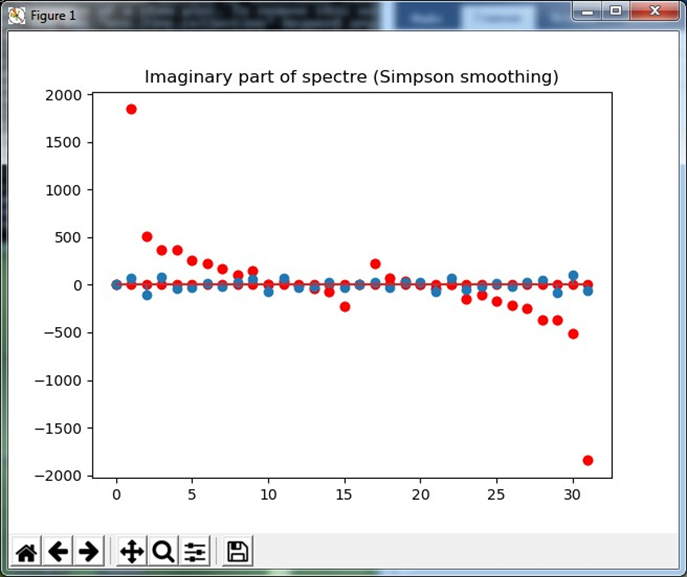


Рисунок 34 – Дискретные отсчёты спектра после фильтрации, мнимая часть (формула Симпсона)

**Выводы.**

В ходе выполнения лабораторной работы получены навыки фильтрации дискретных последовательностей с помощью линейного и полиномиального сглаживаний, а также с помощью фильтров, соответствующих численному дифференцированию и интегрированию. Была написана программа, с помощью которой были выведены графики фильтрации.

ДПФ используется для того, чтобы вычислить спектр сигнала по его дискретным отсчетам. Т.к. спектр линейного сигнала ограничен частотой, меньшей половине частоты дискретизации, то он может быть представлен суммой конечного числа гармоник, а именно – число гармоник равно половине числа отсчетов дискретного сигнала.

**ПРИЛОЖЕНИЕ А**

**ИСХОДНЫЙ КОД**

import math

import matplotlib.pyplot as plt

import numpy as np

from scipy.fft import fft

from scipy.interpolate import interp1d

def gen\_signal():

harmonics = []

amplitudes = np.random.randint(1, 11, 11)

frequencies = [0.1 \* math.pi \* i for i in range(0, 11)]

phases = np.random.uniform(0, 0.5, 11)

for a, f, p in zip(amplitudes, frequencies, phases):

harmonics.append([a \* np.cos(f \* t + p) for t in range(0, 32)])

print("Amplitudes: " + str(amplitudes))

print("Frequencies: " + str(frequencies))

print("Phases: " + str(phases))

return [sum(y) for y in zip(\*harmonics)]

def plot\_signal(signal, interpolate=False):

plt.xlabel('t')

plt.ylabel('Y(t)')

x = np.linspace(0, 31, num=32, endpoint=True)

if interpolate:

plt.title('Analog signal')

interpolated = interp1d(x, signal, kind='cubic')

x = np.linspace(0, 31, num=320, endpoint=True)

signal = interpolated(x)

plt.plot(x, signal)

else:

plt.title('Discrete signal')

plt.stem(x, signal, markerfmt=' ')

plt.show()

def spectre(signal):

dft = fft(signal)

real\_part = [y.real for y in dft]

imag\_part = [y.imag for y in dft]

return real\_part, imag\_part

def split(signal, n):

for i in range(0, len(signal) - n + 1):

yield signal[i:i + n]

def linear(signal):

smoothed = []

for part in split(signal, 5):

smoothed.append(sum(part) / 5)

plt.title('Linear smoothing by 5 points')

plt.stem(range(2, 30), smoothed, 'r', markerfmt='ro')

plt.stem(range(0, 32), signal)

plt.show()

real\_part, imag\_part = spectre(signal)

sm\_real\_part, sm\_imag\_part = spectre(smoothed)

plt.title('Real part of spectre (linear smoothing - 5 points)')

plt.stem(range(2, 30), sm\_real\_part, 'ro', markerfmt='ro')

plt.stem(range(0, 32), real\_part)

plt.show()

plt.title('Imaginary part of spectre (linear smoothing - 5 points)')

plt.stem(range(2, 30), sm\_imag\_part, 'ro', markerfmt='ro')

plt.stem(range(0, 32), imag\_part)

plt.show()

smoothed = []

for part in split(signal, 9):

smoothed.append(sum(part) / 9)

plt.title('Linear smoothing by 9 points')

plt.stem(range(4, 28), smoothed, 'r', markerfmt='ro')

plt.stem(range(0, 32), signal)

plt.show()

sm\_real\_part, sm\_imag\_part = spectre(smoothed)

plt.title('Real part of spectre (linear smoothing - 9 points)')

plt.stem(range(4, 28), sm\_real\_part, 'ro', markerfmt='ro')

plt.stem(range(0, 32), real\_part)

plt.show()

plt.title('Imaginary part of spectre (linear smoothing - 9 points)')

plt.stem(range(4, 28), sm\_imag\_part, 'ro', markerfmt='ro')

plt.stem(range(0, 32), imag\_part)

plt.show()

def deg2(signal):

real\_part, imag\_part = spectre(signal)

smoothed = []

for part in split(signal, 5):

smoothed.append((-3 \* part[0] + 12 \* part[1] + 17 \* part[2] + 12 \* part[3]- 3 \* part[4]) / 35)

plt.title('Polynomial (2 deg) smoothing by 5 points')

plt.stem(range(2, 30), smoothed, 'r', markerfmt='ro')

plt.stem(range(0, 32), signal)

plt.show()

sm\_real\_part, sm\_imag\_part = spectre(smoothed)

plt.title('Real part of spectre (polynomial (2 deg) smoothing - 5 points)')

plt.stem(range(2, 30), sm\_real\_part, 'ro', markerfmt='ro')

plt.stem(range(0, 32), real\_part)

plt.show()

plt.title('Imaginary part of spectre (polynomial (2 deg) smoothing - 5 points)')

plt.stem(range(2, 30), sm\_imag\_part, 'ro', markerfmt='ro')

plt.stem(range(0, 32), imag\_part)

plt.show()

smoothed = []

for part in split(signal, 9):

smoothed.append((-21 \* part[0] + 14 \* part[1] + 39 \* part[2] + 54 \* part[3] + 59 \* part[4] + 54 \* part[5]+ 39 \* part[6] + 14 \* part[7] - 21 \* part[8]) / 231)

plt.title('Polynomial (2 deg) smoothing by 9 points')

plt.stem(range(4, 28), smoothed, 'r', markerfmt='ro')

plt.stem(range(0, 32), signal)

plt.show()

sm\_real\_part, sm\_imag\_part = spectre(smoothed)

plt.title('Real part of spectre (polynomial (2 deg) smoothing - 9 points)')

plt.stem(range(4, 28), sm\_real\_part, 'ro', markerfmt='ro')

plt.stem(range(0, 32), real\_part)

plt.show()

plt.title('Imaginary part of spectre (polynomial (2 deg) smoothing - 9 points)')

plt.stem(range(4, 28), sm\_imag\_part, 'ro', markerfmt='ro')

plt.stem(range(0, 32), imag\_part)

plt.show()

def deg4(signal):

real\_part, imag\_part = spectre(signal)

smoothed = []

for part in split(signal, 7):

smoothed.append((5 \* part[0] - 30 \* part[1] + 75 \* part[2] + 131 \* part[3] + 75 \* part[4] - 30 \* part[5] + 5 \* part[6]) / 231)

plt.title('Polynomial (4 deg) smoothing by 7 points')

plt.stem(range(3, 29), smoothed, 'r', markerfmt='ro')

plt.stem(range(0, 32), signal)

plt.show()

sm\_real\_part, sm\_imag\_part = spectre(smoothed)

plt.title('Real part of spectre (polynomial (4 deg) smoothing - 7 points)')

plt.stem(range(3, 29), sm\_real\_part, 'ro', markerfmt='ro')

plt.stem(range(0, 32), real\_part)

plt.show()

plt.title('Imaginary part of spectre (polynomial (4 deg) smoothing - 7 points)')

plt.stem(range(3, 29), sm\_imag\_part, 'ro', markerfmt='ro')

plt.stem(range(0, 32), imag\_part)

plt.show()

smoothed = []

for part in split(signal, 11):

smoothed.append((18 \* part[0] - 45 \* part[1] - 10 \* part[2] + 60 \* part[3] + 120 \* part[4] + 143 \* part[5] + 120 \* part[6] + 60 \* part[7] - 10 \* part[8] - 45 \* part[9] + 18 \* part[10]) / 429)

plt.title('Polynomial (4 deg) smoothing by 11 points')

plt.stem(range(6, 28), smoothed, 'r', markerfmt='ro')

plt.stem(range(0, 32), signal)

plt.show()

sm\_real\_part, sm\_imag\_part = spectre(smoothed)

plt.title('Real part of spectre (polynomial (4 deg) smoothing - 11 points)')

plt.stem(range(6, 28), sm\_real\_part, 'ro', markerfmt='ro')

plt.stem(range(0, 32), real\_part)

plt.show()

plt.title('Imaginary part of spectre (polynomial (4 deg) smoothing - 11 points)')

plt.stem(range(6, 28), sm\_imag\_part, 'ro', markerfmt='ro')

plt.stem(range(0, 32), imag\_part)

plt.show()

def diff(signal):

real\_part, imag\_part = spectre(signal)

smoothed = []

for part in split(signal, 3):

smoothed.append((part[0] - part[2])/2)

plt.title('Differentiation smoothing')

plt.stem(range(1, 31), smoothed, 'r', markerfmt='ro')

plt.stem(range(0, 32), signal)

plt.show()

sm\_real\_part, sm\_imag\_part = spectre(smoothed)

plt.title('Real part of spectre (differentiation smoothing)')

plt.stem(range(1, 31), sm\_real\_part, 'ro', markerfmt='ro')

plt.stem(range(0, 32), real\_part)

plt.show()

plt.title('Imaginary part of spectre (differentiation smoothing)')

plt.stem(range(1, 31), sm\_imag\_part, 'ro', markerfmt='ro')

plt.stem(range(0, 32), imag\_part)

plt.show()

def intr(signal):

real\_part, imag\_part = spectre(signal)

smoothed = [0]

for i in range(1, len(signal)):

smoothed.append(smoothed[i-1] + signal[i])

plt.title('Rectangle smoothing')

plt.stem(range(0, 32), smoothed, 'r', markerfmt='ro')

plt.stem(range(0, 32), signal)

plt.show()

sm\_real\_part, sm\_imag\_part = spectre(smoothed)

plt.title('Real part of spectre (rectangle smoothing)')

plt.stem(range(0, 32), sm\_real\_part, 'ro', markerfmt='ro')

plt.stem(range(0, 32), real\_part)

plt.show()

plt.title('Imaginary part of spectre (rectangle smoothing)')

plt.stem(range(0, 32), sm\_imag\_part, 'ro', markerfmt='ro')

plt.stem(range(0, 32), imag\_part)

plt.show()

smoothed = [0]

for i in range(1, len(signal)):

smoothed.append(smoothed[i - 1] + 1/2 \* (signal[i-1] + signal[i]))

plt.title('Trapeze smoothing')

plt.stem(range(0, 32), smoothed, 'r', markerfmt='ro')

plt.stem(range(0, 32), signal)

plt.show()

sm\_real\_part, sm\_imag\_part = spectre(smoothed)

plt.title('Real part of spectre (trapeze smoothing)')

plt.stem(range(0, 32), sm\_real\_part, 'ro', markerfmt='ro')

plt.stem(range(0, 32), real\_part)

plt.show()

plt.title('Imaginary part of spectre (trapeze smoothing)')

plt.stem(range(0, 32), sm\_imag\_part, 'ro', markerfmt='ro')

plt.stem(range(0, 32), imag\_part)

plt.show()

smoothed = [0, 0]

for i in range(2, len(signal)):

smoothed.append(smoothed[i - 2] + 1/3 \* (signal[i-2] + 4\*signal[i-1] + 4\*signal[i]))

plt.title('Simpson smoothing')

plt.stem(range(0, 32), smoothed, 'r', markerfmt='ro')

plt.stem(range(0, 32), signal)

plt.show()

sm\_real\_part, sm\_imag\_part = spectre(smoothed)

plt.title('Real part of spectre (Simpson smoothing)')

plt.stem(range(0, 32), sm\_real\_part, 'ro', markerfmt='ro')

plt.stem(range(0, 32), real\_part)

plt.show()

plt.title('Imaginary part of spectre (Simpson smoothing)')

plt.stem(range(0, 32), sm\_imag\_part, 'ro', markerfmt='ro')

plt.stem(range(0, 32), imag\_part)

plt.show()

# task 1

signal = gen\_signal()

# task 2

plot\_signal(signal, True)

plot\_signal(signal, False)

# task 3

real\_part, imag\_part = spectre(signal)

plt.title('Real part of spectre')

plt.stem(range(0, 32), real\_part, markerfmt=' ')

plt.show()

plt.title('Imaginary part of spectre')

plt.stem(range(0, 32), imag\_part, markerfmt=' ')

plt.show()

# task 4,5,6

linear(signal)

# task 8a

deg2(signal)

# task 8b

deg4(signal)

# task 8c

diff(signal)

# task 8d

intr(signal)