**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. 8382 |  | Облизанов А.Д. |
| Студентка гр. 8382 |  | Ивлева О.А. |
| Преподаватель |  | Сучков А.И. |

Санкт-Петербург

2021

## 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.

## Выполнение работы.

## Формирование дискретного сигнала

Был сформирован дискретный сигнал посредством дискретизации с шагом непрерывного сигнала, представляющего собой линейную комбинацию косинусоид. Коэффициенты линейной комбинации были нормализованы посредством деления их на сумму полученных случайным образом амплитуд. Дискретная последовательность включает в себя 32 отсчета. Аналоговый сигнал представлен на рис. 1, дискретный сигнал – на рис. 2.

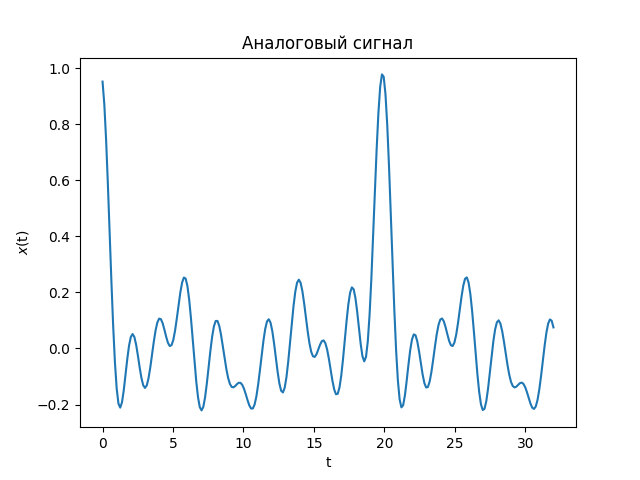


Рисунок 1 – Аналоговый сигнал

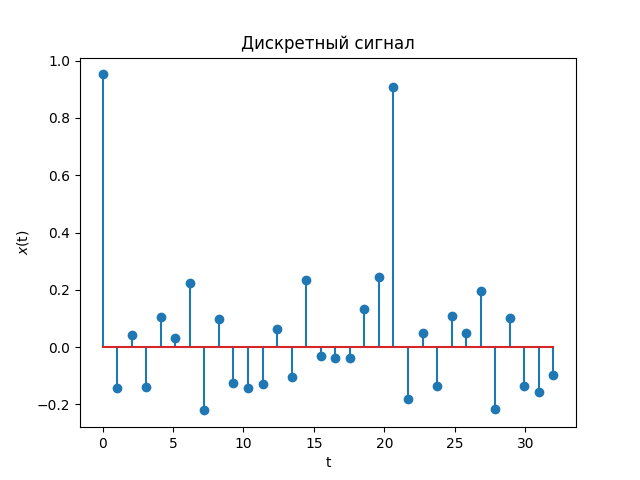


Рисунок 2 – Дискретный сигнал

## Нахождение дискретных отсчетов спектра дискретного сигнала с помощью ДПФ

Представим дискретный сигнал в виде функции от времени:

Тогда спектр дискретного сигнала:

С другой стороны, дискретный сигнал можно представить в виде:

Сумму можно представить в виде комплексного ряда Фурье, тогда

Полученный с помощью дискретного преобразования Фурье спектр дискретного сигнала представлен на рис. 3.

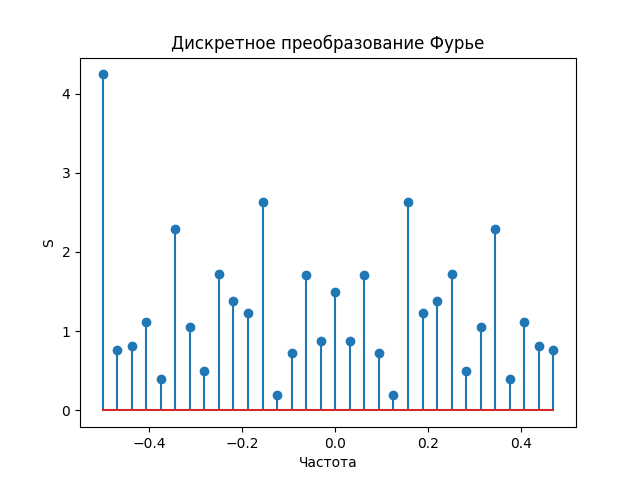


Рисунок 3 – Дискретные отсчеты спектра дискретного сигнала

Можно заметить, что спектр симметричен относительно нуля. Спектр представляет собой разложение исходного сигнала на линейную комбинацию простых синусоидальных функций и, соответственно, отражает амплитуды этих функций на разных частотах. Так как сигнал изначально является линейной комбинацией косинусоид, то в спектре представлены все частоты. Спектр имеет периодичность с шагом 1.

## Применение линейного сглаживания по 5-ти и 9-ти точкам

Линейное сглаживание по 5-ти и 9-ти точкам осуществляется с помощью полинома первой степени, а коэффициенты в передаточной функции одинаковы. Так, в z-области:

Общая формула передаточной функции в частотной области:

На рис. 4 представлено сравнение исходного сигнала и сигнала после применения линейного сглаживания по 5-ти точкам.

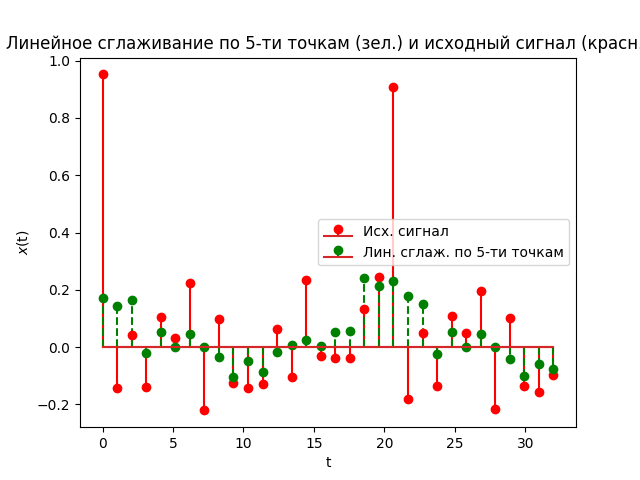


Рисунок 4 – Линейное сглаживание по 5-ти точкам

На рис. 5 представлено сравнение исходного сигнала и сигнала после применения линейного сглаживания по 9-ти точкам.

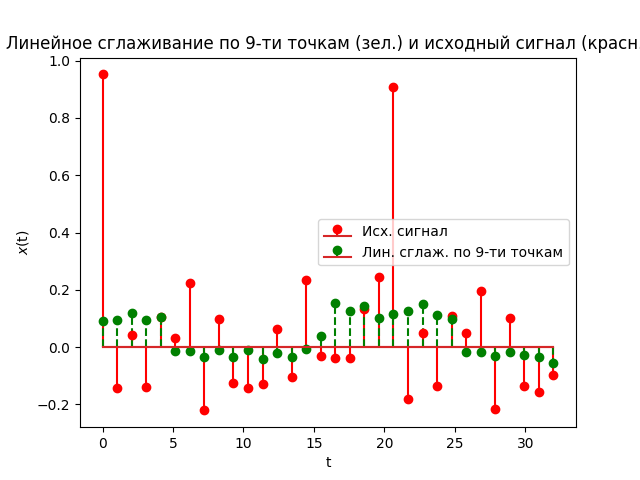


Рисунок 5 – Линейное сглаживание по 9-ти точкам

Аналогичным п.2 образом были получены спектры сигнала после фильтрации. На рис. 6 представлено сравнение спектров исходного сигнала, сигнала после линейного сглаживания по 5-ти и 9-ти точкам.

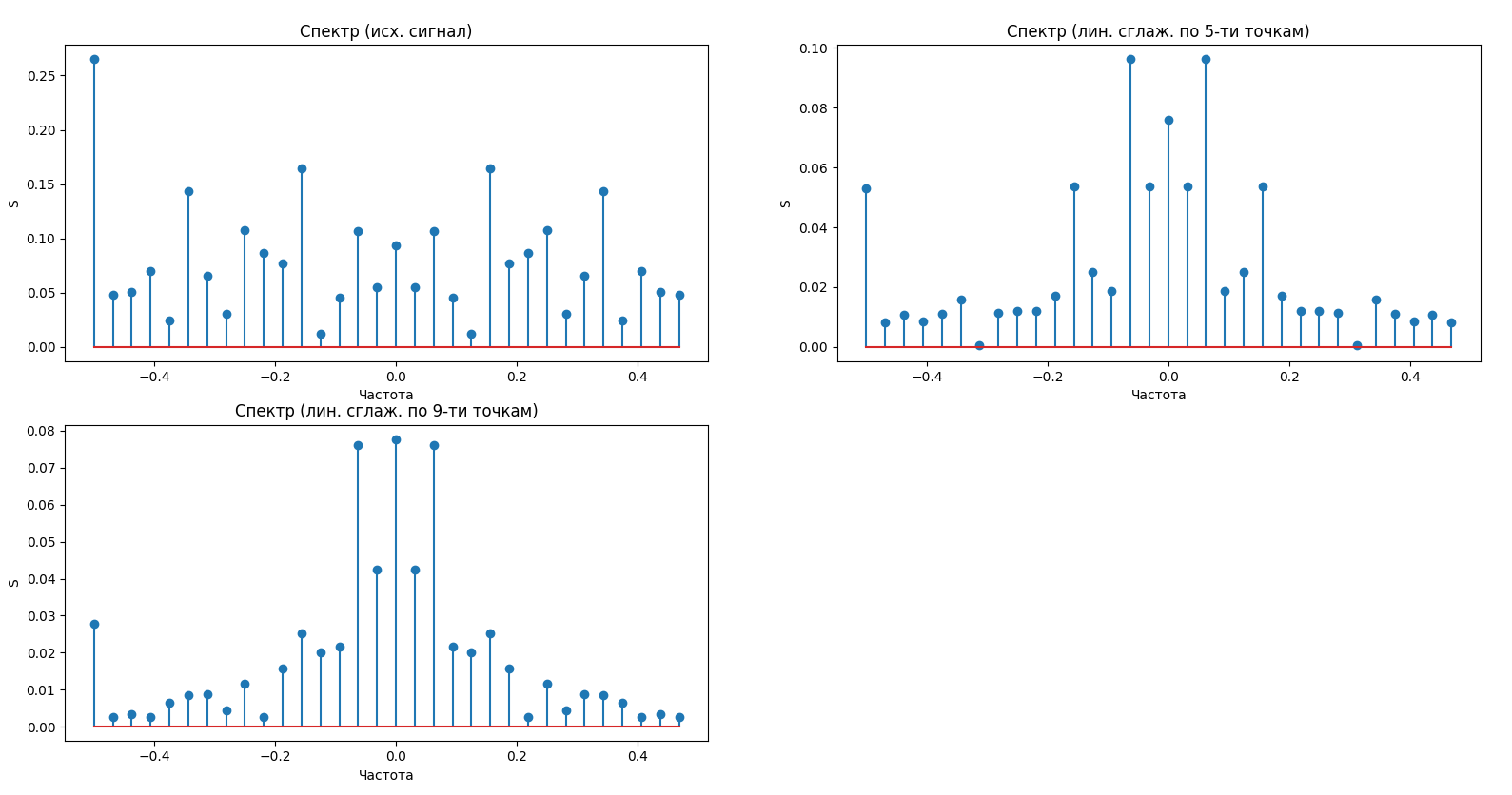


Рисунок 6 – Дискретные отсчеты спектра после линейного сглаживания

Из спектра видно, что без ослабления пропускается только сигнал постоянного уровня (нулевой частоты). С увеличением числа точек полоса пропускания становится уже.

График передаточной функции линейного сглаживания представлен на рис. 7.

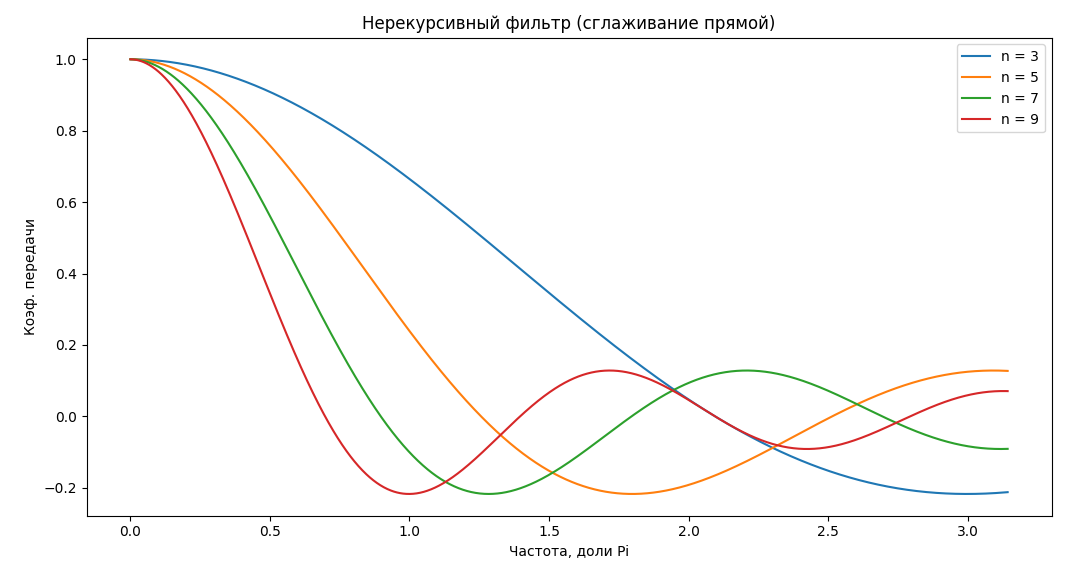


Рисунок 7 – График передаточной функции линейного сглаживания

График передаточной функции подтверждает ранее сделанные выводы относительно полосы пропускания и ослабления уровня.

## Сглаживание полиномом 2-й степени по 5 и 9 узлам

Формулы передаточных функций для сглаживания полиномом второй степени:

На рис. 8 представлено сравнение исходного сигнала и сигнала после применения сглаживания полиномом 2-й степени по 5-ти и 9-ти точкам.

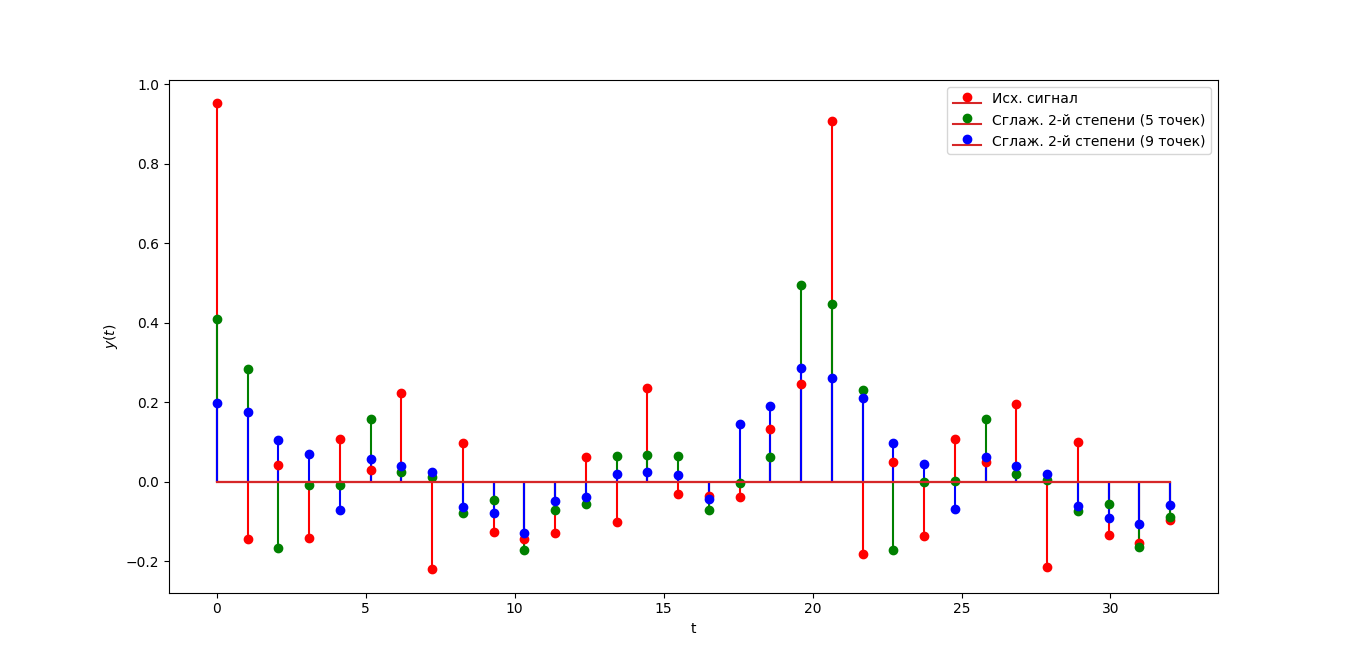


Рисунок 8 – Сглаживание полиномом 2-й степени по 5-ти и 9-ти точкам

Аналогичным п.2 образом были получены спектры сигнала после фильтрации. На рис. 9 представлено сравнение спектров исходного сигнала, сигнала после сглаживания полиномом 2-й степени по 5-ти и 9-ти точкам.

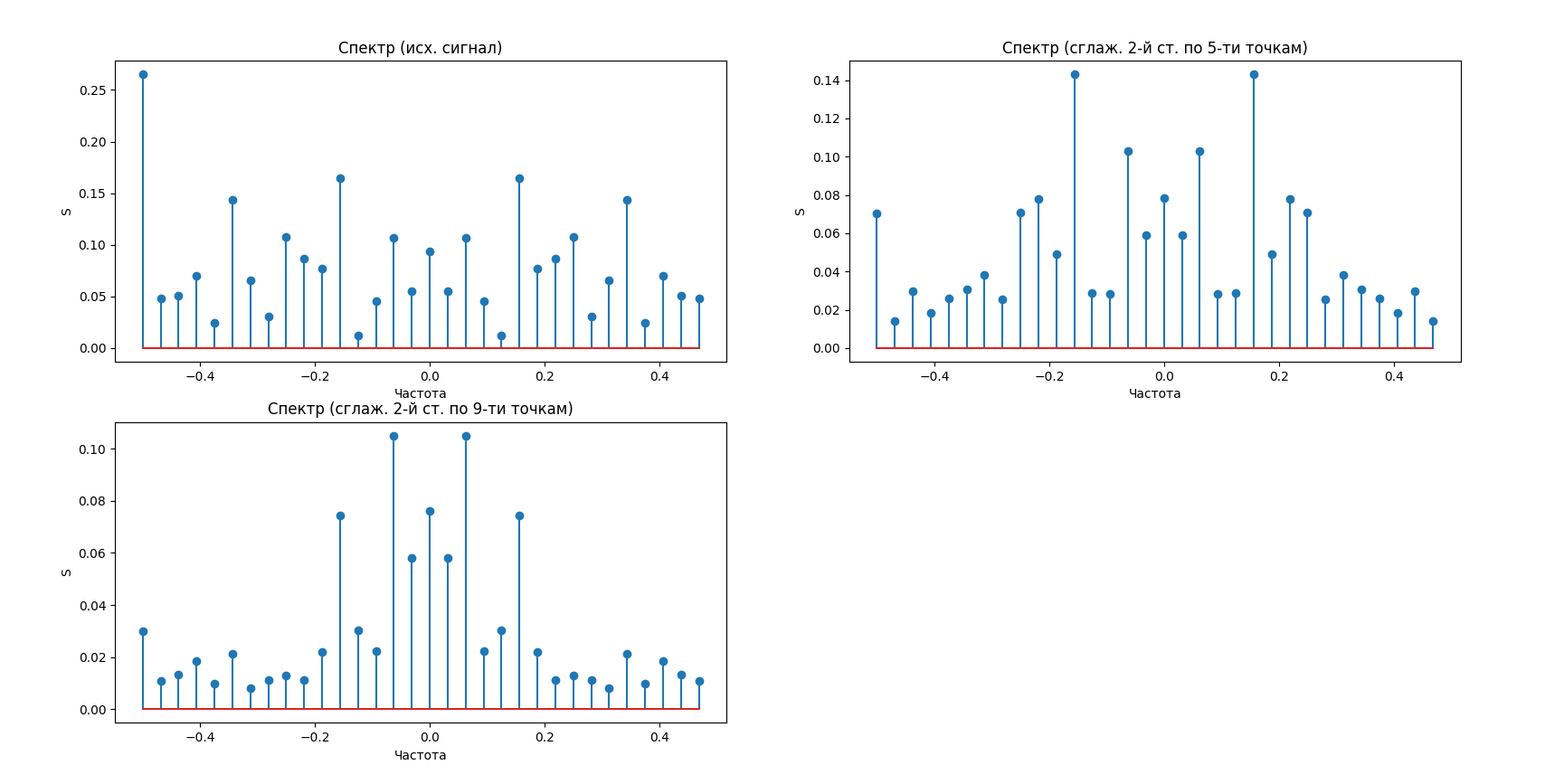


Рисунок 9 – Спектры сигналов после сглаживание фильтром 2-го порядка

На рис. 10 представлены графики передаточных функций фильтра сглаживания полиномом второй степени.

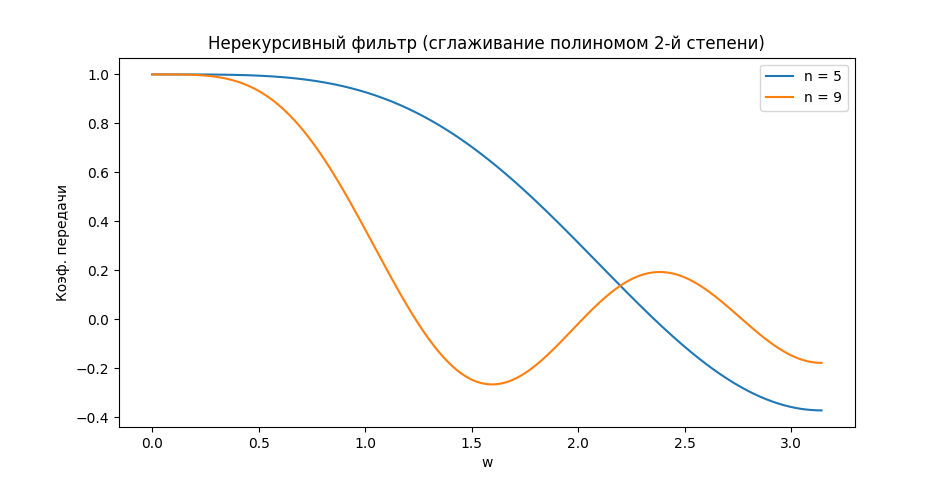


Рисунок 10 – Передаточная функция фильтра 2-го порядка

Из графика видно, что при увеличении числа точек увеличивается крутизна среза и уменьшается полоса пропускания. Также присутствует более «пологая» часть в полосе пропускания около 0, в отличие от линейного фильтра, поэтому пропускаются сигналы низкой частоты почти без потери амплитуды в более широком диапазоне, что видно на графиках спектра.

## Сглаживание полиномом 4-й степени по 7 и 11 узлам

Формулы передаточных функций для сглаживания полиномом четвертой степени:

На рис. 11 представлено сравнение исходного сигнала и сигнала после применения сглаживания полиномом 4-й степени по 9-ти и 11-ти точкам.

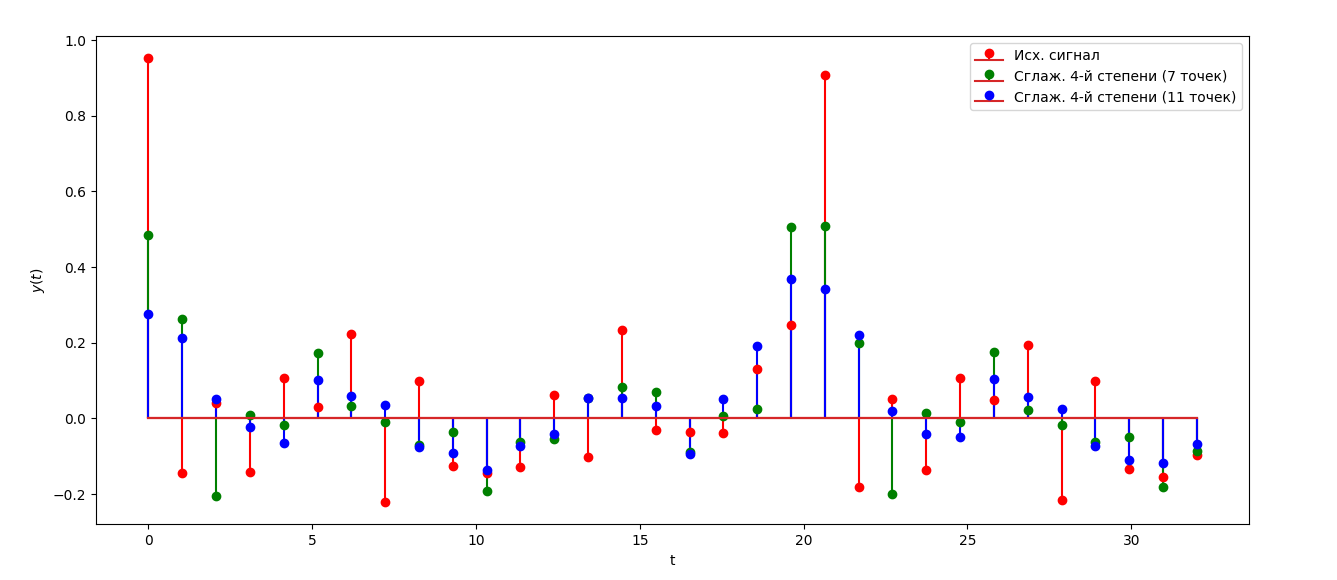


Рисунок 11 – Сглаживание полиномом 4-й степени по 9-ти и 11-ти точкам

Аналогичным п.2 образом были получены спектры сигнала после фильтрации. На рис. 12 представлено сравнение спектров исходного сигнала, сигнала после сглаживания полиномом 4-й степени по 7-ти и 11-ти точкам.

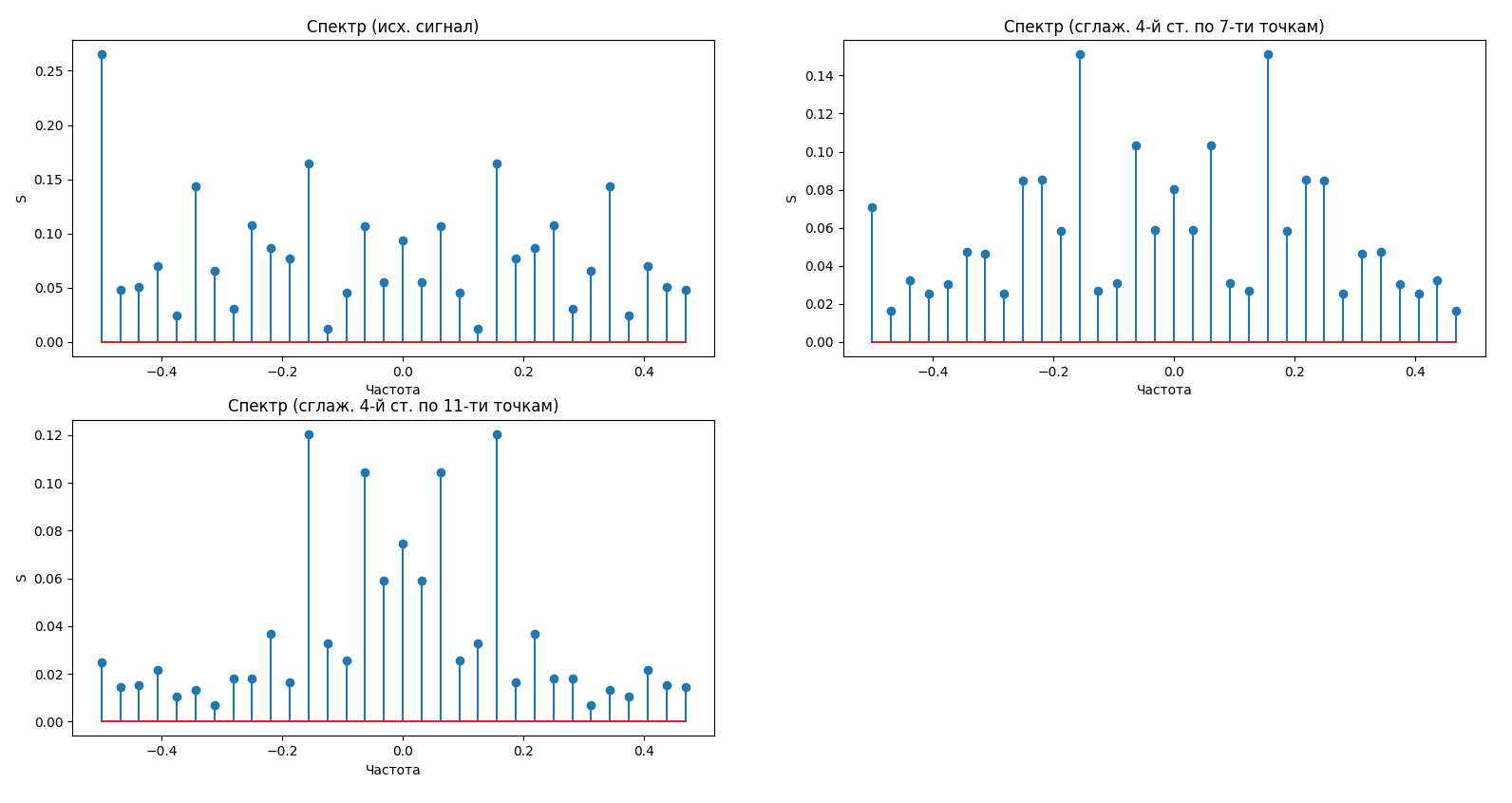


Рисунок 12 – Спектры сигналов после сглаживание фильтром 4-го порядка

На рис. 13 представлены графики передаточных функций фильтра сглаживания полиномом четвертой степени.

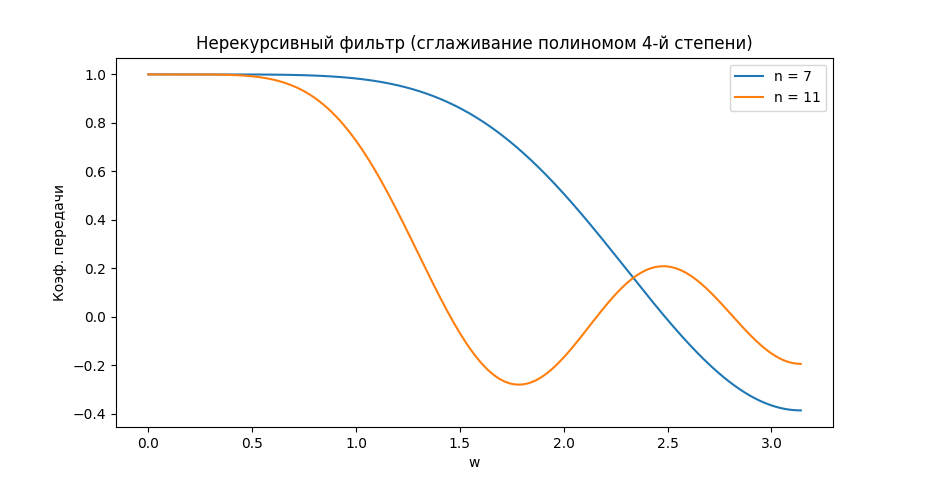


Рисунок 13 – Передаточная функция фильтра 4-го порядка

Полученные спектры после фильтрации схожи с таковыми при сглаживании полиномом 2-й степени, однако здесь наблюдается еще более широкая пологая часть пропускания низких частот почти без ослабления. При сглаживании по 11-ти точкам можно увидеть, что полоса пропускания становится уже, что видно на спектре: большее число дискретных отсчетов ослабляется.

## Дискретный фильтр, соответствующий численному дифференцированию 1-го порядка.

Передаточная функция имеет вид:

На рис. 14 представлено сравнение исходного сигнала и сигнала после применения сглаживания фильтром дифференцирования первого порядка:

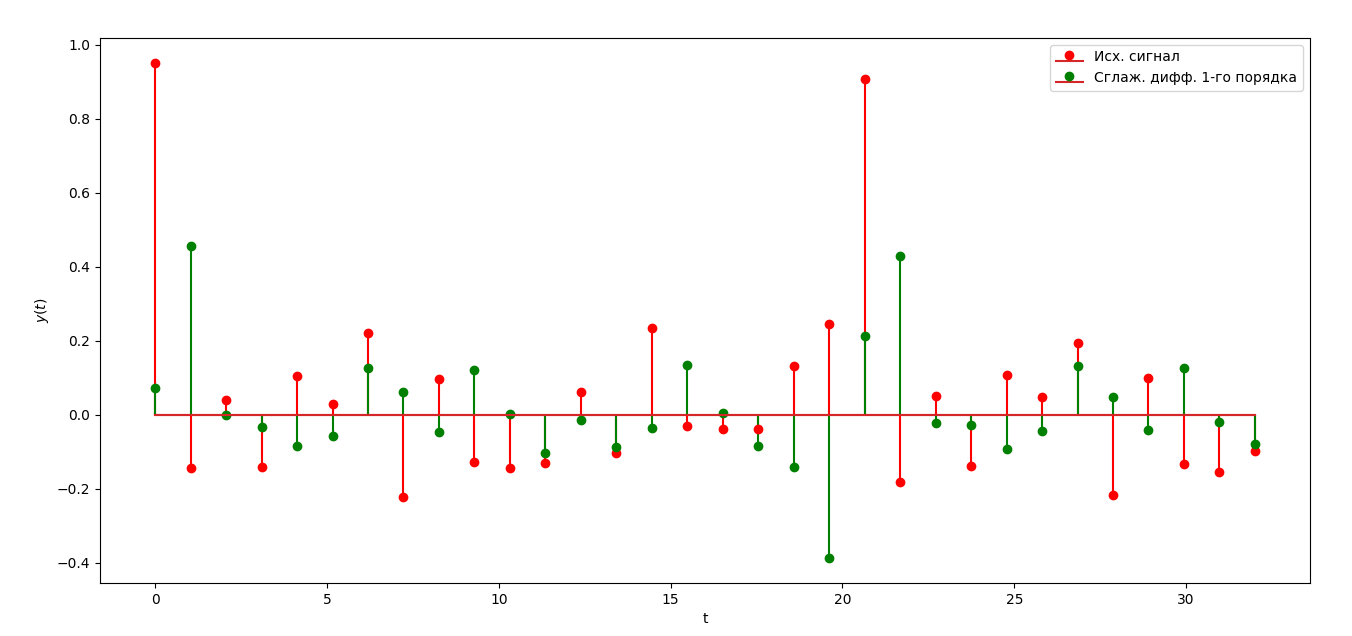


Рисунок 14 – Сглаживание фильтром дифференцирования первого порядка

Аналогичным п.2 образом был получен спектр сигнала после фильтрации. Сравнение спектров исходного и фильтрованного сигнала представлено на рис. 15.

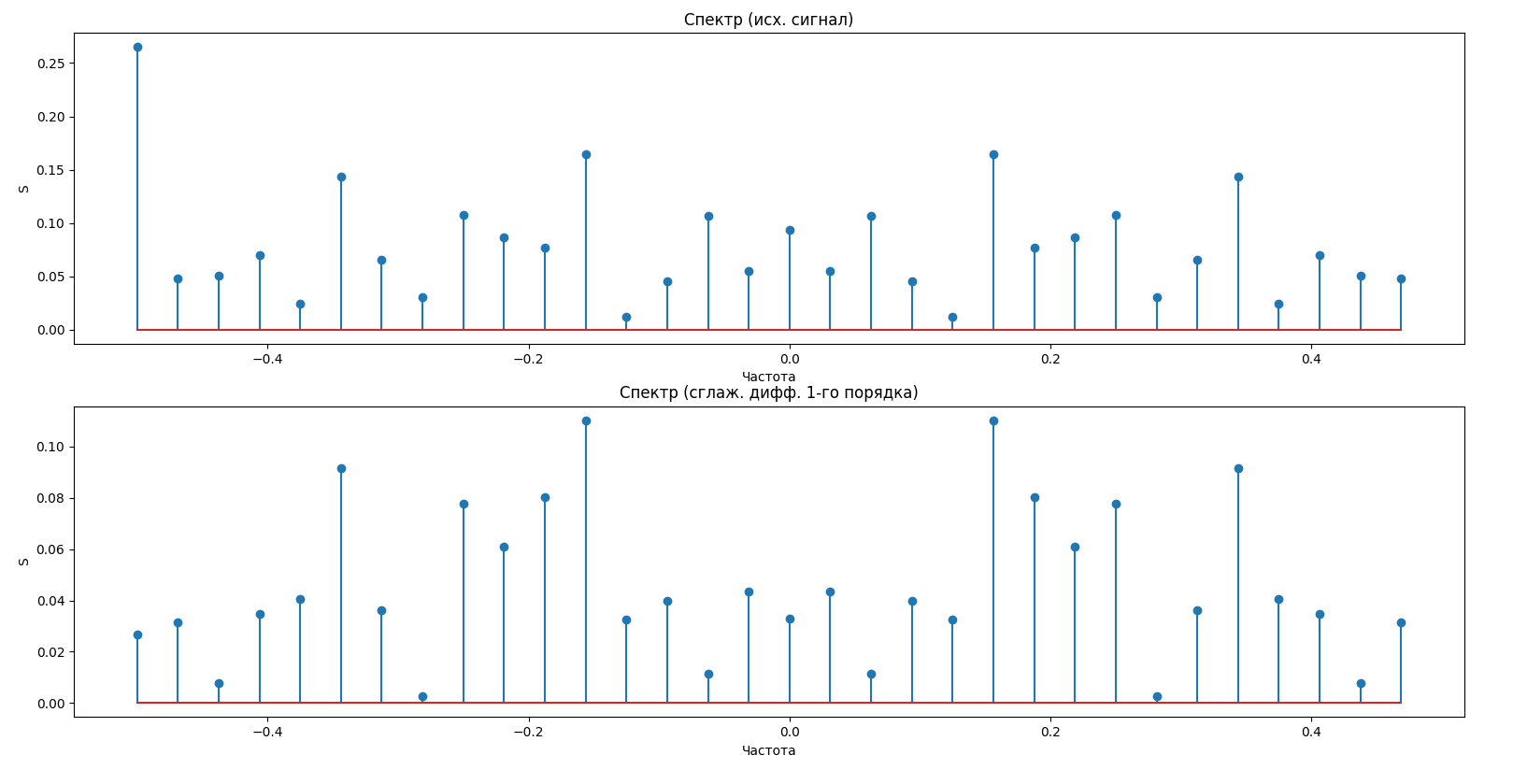


Рисунок 15 – Сравнение спектров исходного и фильтрованного сигнала

График передаточной функции представлен на рис. 16.

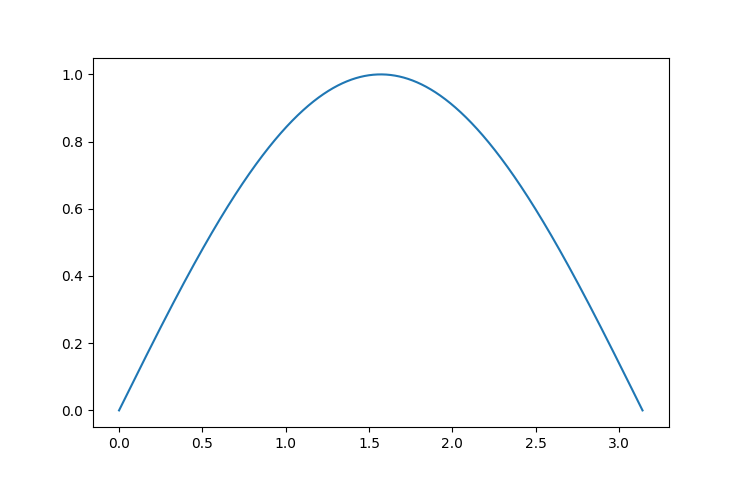


Рисунок 16 – Передаточная функция

По графику спектра видно, что средние частоты действительно усиливаются, при этом низкие и высокие частоты сглаживаются, что подтверждается графиком передаточной функции.

## Дискретный фильтр, соответствующий численному интегрированию (прямоугольников, трапеций, Симпсона).

Передаточная функция для рекурсивного фильтра, соответствующего численному интегрированию трапеций:

Передаточная функция для рекурсивного фильтра, соответствующего численному интегрированию прямоугольников:

Передаточная функция для рекурсивного фильтра, соответствующего численному интегрированию по формуле Симпсона:

На рис. 17 представлено сравнение исходного сигнала и сигнала после применения сглаживания фильтрами интегрирования.

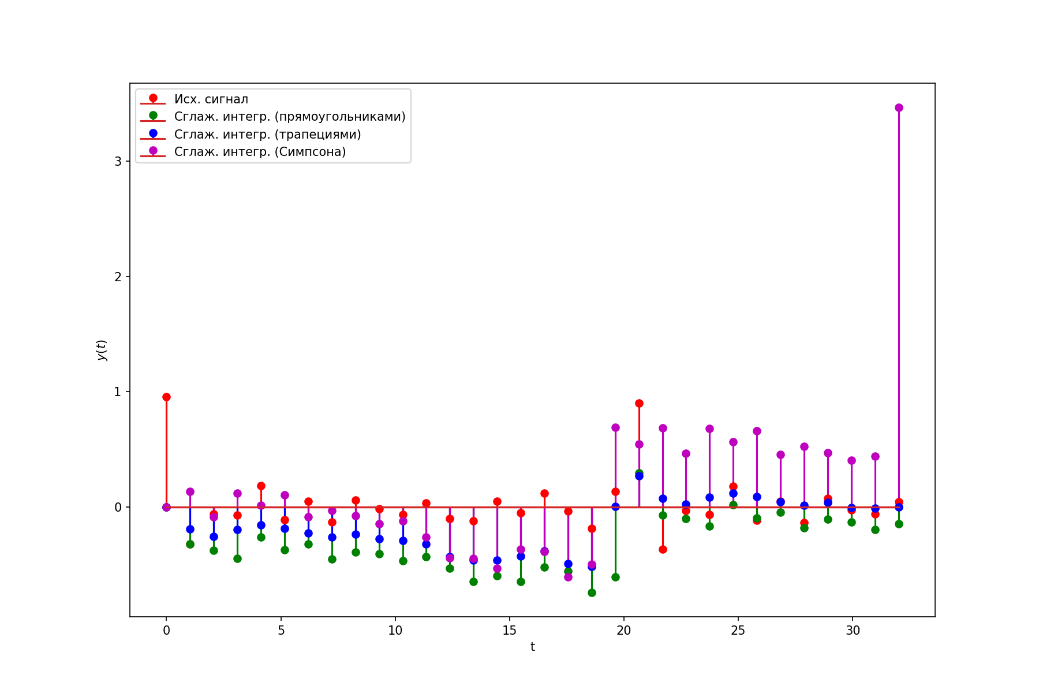


Рисунок 17 – Сравнение исходного и отфильтрованного сигналов

Аналогичным п.2 образом был получен спектр сигнала после фильтрации. Сравнение спектров исходного и фильтрованного сигнала представлено на рис. 18.

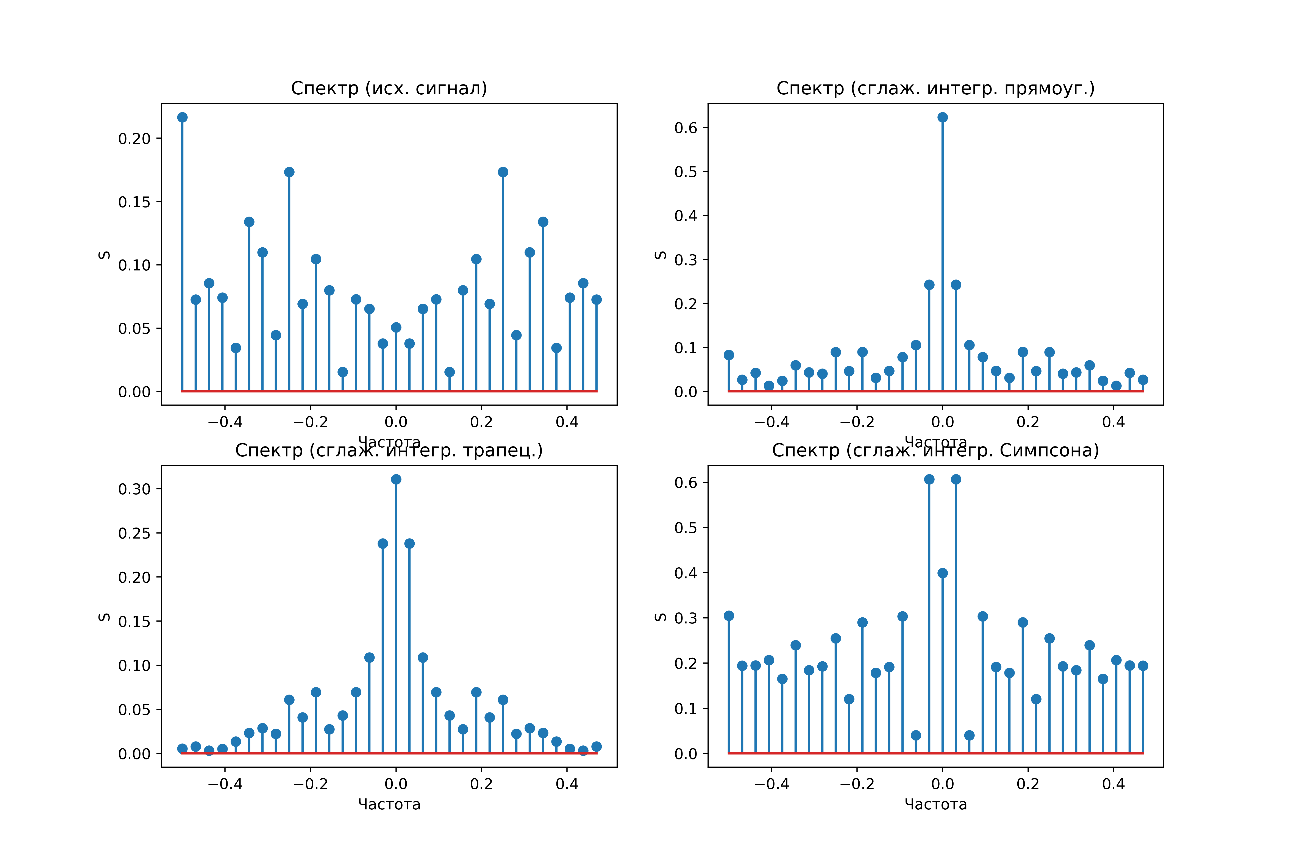


Рисунок 18 – Спектр исходного и отфильтрованных сигналов

Из спектров можно увидеть отличие сглаживаний интегрированием с помощью различных формул.

* Сглаживание интегрированием по формулам прямоугольников значительно усиливает низкие частоты в небольшом диапазоне
* Сглаживание интегрированием по формулам трапеции усиливает низкие частоты, но в большем диапазоне и с меньшим усилением
* Сглаживание интегрированием по формулам Симпсона усиливает весь спектр частот, а в особенности низкие частоты

Исходный код программы, разработанной для генерации сигналов, применения фильтров и построения графиков, представлен в Приложении А.

## Выводы

В результате выполнения лабораторной работы был сгенерирован аналоговый сигнал, дискретизирован, построен спектр дискретного сигнала. Было выяснено, что спектр дискретного сигнала симметричен относительно 0, в спектре (представление в виде дискретных отсчетов) представлено множество частот.

Были применены фильтры: линейного сглаживания, сглаживания полиномом 2-й и 4-й степени, построены графики сигнала и спектра, передаточной функции. В результате по спектру было определено, что обеспечивается фильтрация высоких частот, ширина полосы пропускания уменьшается с увеличением числа точек, равномерность передачи сигнала в полосе пропускания увеличивается при использовании полиномов более высоких порядков.

Был применен дискретный фильтр, соответствующий численному дифференцированию, построен график сигнала, спектр, передаточная функция. В результате графики спектра и передаточной функции показали, что фильтр имеет полосу пропускания в области средних частот, уменьшает амплитуду низких и высоких частот.

Были применены фильтры, соответствующие численному интегрированию по формулам прямоугольников, трапеций, Симпсона. Фильтры отличаются усилением сигнала в области низких частот, различной полосой пропускания.

# Приложение А.

# Исходный код программы

import numpy as np

from scipy.fftpack import fft, fftfreq

import matplotlib.pyplot as plt

fig = plt.gcf()

fig.set\_size\_inches(12, 8)

def rect(orig):

output = np.empty(len(orig))

output[0] = 0

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

output[i] = output[i - 1] + orig[i]

return output

def simpson(orig):

output = np.empty(len(orig))

output[0] = 0

for i in range(1, len(orig) - 1):

output[i] = output[i - 1] + (orig[i - 1] + orig[i] + 4 \* orig[i + 1]) / 3

return output

def trap(orig):

output = np.empty(len(orig))

output[0] = 0

for i in range(1, len(orig) - 1):

output[i] = output[i - 1] + (orig[i] + orig[i + 1]) / 2

return output

def get\_analog(n):

ws = np.arange(0, np.pi + 0.1 \* np.pi, 0.1 \* np.pi)

us = np.random.random(size=11) \* 0.5

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

x = np.linspace(0, 32, n)

y = np.zeros(n)

for w, u, A in zip(ws, us, As):

y += (A \* np.cos(w \* x + u))

y = y / np.sum(As)

return x, y

def get\_discrete(y):

out\_x = np.linspace(0, 32, 32)

out\_y = []

for idx in range(32):

out\_y.append(y[idx \* 8])

return out\_x, np.array(out\_y)

def gen\_analog\_make\_discrete(numfig):

x, y = get\_analog(256)

plt.plot(x, y, numfig)

plt.ylabel(r'$x$(t)')

plt.xlabel('t')

plt.title('Аналоговый сигнал')

plt.savefig(str(numfig) + ".png", dpi=150)

numfig += 1

plt.clf()

x\_disc, y\_disc = get\_discrete(y)

plt.stem(x\_disc, y\_disc)

plt.ylabel(r'$x$(t)')

plt.xlabel('t')

plt.title('Дискретный сигнал')

plt.savefig(str(numfig) + ".png", dpi=150)

numfig += 1

plt.clf()

return x\_disc, y\_disc

def spectrum(y, numfig):

yf = fft(y)

xf = fftfreq(32, 1)

plt.stem(xf, np.abs(yf))

plt.ylabel(r'S')

plt.xlabel('Частота')

plt.title('Дискретное преобразование Фурье')

plt.savefig(str(numfig) + ".png", dpi=150)

numfig += 1

plt.clf()

return

def linear\_avg(x, y, numfig):

lin\_avg\_5 = np.convolve(y, np.ones(5), 'same') / 5

lin\_avg\_9 = np.convolve(y, np.ones(9), 'same') / 9

plt.stem(x, y, 'r', markerfmt='ro', label='Исх. сигнал')

plt.stem(x, lin\_avg\_5, 'g--', markerfmt='go', label='Лин. сглаж. по 5-ти точкам')

plt.ylabel(r'$x$(t)')

plt.xlabel('t')

plt.legend()

plt.title('Линейное сглаживание по 5-ти точкам (зел.) и исходный сигнал (красн.)')

plt.savefig(str(numfig) + ".png", dpi=150)

numfig += 1

plt.clf()

plt.stem(x, y, 'r', markerfmt='ro', label='Исх. сигнал')

plt.stem(x, lin\_avg\_9, 'g--', markerfmt='go', label='Лин. сглаж. по 9-ти точкам')

plt.ylabel(r'$x$(t)')

plt.xlabel('t')

plt.legend()

plt.title('Линейное сглаживание по 9-ти точкам (зел.) и исходный сигнал (красн.)')

plt.savefig(str(numfig) + ".png", dpi=150)

numfig += 1

plt.clf()

return

def linear\_avg\_spectrum(x, y, numfig):

lin\_avg\_5 = np.convolve(y, np.ones(5), 'same') / 5

lin\_avg\_9 = np.convolve(y, np.ones(9), 'same') / 9

xf = fftfreq(32, 1)

f = fft(y)

f = 2 \* np.abs(f) / len(f)

f\_5 = fft(lin\_avg\_5)

f\_5 = 2 \* np.abs(f\_5) / len(f\_5)

f\_9 = fft(lin\_avg\_9)

f\_9 = 2 \* np.abs(f\_9) / len(f\_9)

plt.subplot(221)

plt.stem(xf, f)

plt.ylabel(r'S')

plt.xlabel('Частота')

plt.title('Спектр (исх. сигнал)')

plt.subplot(222)

plt.stem(xf, f\_5)

plt.ylabel(r'S')

plt.xlabel('Частота')

plt.title('Спектр (лин. сглаж. по 5-ти точкам)')

plt.subplot(223)

plt.stem(xf, f\_9)

plt.ylabel(r'S')

plt.xlabel('Частота')

plt.title('Спектр (лин. сглаж. по 9-ти точкам)')

plt.savefig(str(numfig) + ".png", dpi=150)

numfig += 1

plt.clf()

return

def square\_avg(x, y, numfig):

sq\_avg\_5 = np.convolve(y, np.array([-3, 12, 17, 12, -3]), 'same') / 35

sq\_avg\_9 = np.convolve(y, np.array([-21, 14, 39, 54, 59, 54, 39, 14, -21]), 'same') / 231

plt.stem(x, y, 'r', markerfmt='ro', label='Исх. сигнал')

plt.stem(x, sq\_avg\_5, 'g', markerfmt='go', label='Сглаж. 2-й степени (5 точек)')

plt.stem(x, sq\_avg\_9, 'b', markerfmt='bo', label='Сглаж. 2-й степени (9 точек)')

plt.xlabel('t')

plt.ylabel(r'$y(t)$')

plt.legend()

plt.savefig(str(numfig) + ".png", dpi=150)

numfig += 1

plt.clf()

xf = fftfreq(32, 1)

f = fft(y)

f = 2 \* np.abs(f) / len(f)

f\_5 = fft(sq\_avg\_5)

f\_5 = 2 \* np.abs(f\_5) / len(f\_5)

f\_9 = fft(sq\_avg\_9)

f\_9 = 2 \* np.abs(f\_9) / len(f\_9)

plt.subplot(221)

plt.stem(xf, f)

plt.ylabel(r'S')

plt.xlabel('Частота')

plt.title('Спектр (исх. сигнал)')

plt.subplot(222)

plt.stem(xf, f\_5)

plt.ylabel(r'S')

plt.xlabel('Частота')

plt.title('Спектр (сглаж. 2-й ст. по 5-ти точкам)')

plt.subplot(223)

plt.stem(xf, f\_9)

plt.ylabel(r'S')

plt.xlabel('Частота')

plt.title('Спектр (сглаж. 2-й ст. по 9-ти точкам)')

plt.savefig(str(numfig) + ".png", dpi=150)

numfig += 1

plt.clf()

def quad\_avg(x, y, numfig):

quad\_avg\_7 = np.convolve(y, np.array([5, -30, 75, 131, 75, -30, 5]), 'same') / 231

quad\_avg\_11 = np.convolve(y, np.array([13, -45, -10, 60, 120, 143, 120, 60, -10, -45, 13]), 'same') / 429

plt.stem(x, y, 'r', markerfmt='ro', label='Исх. сигнал')

plt.stem(x, quad\_avg\_7, 'g', markerfmt='go', label='Сглаж. 4-й степени (7 точек)')

plt.stem(x, quad\_avg\_11, 'b', markerfmt='bo', label='Сглаж. 4-й степени (11 точек)')

plt.xlabel('t')

plt.ylabel(r'$y(t)$')

plt.legend()

plt.savefig(str(numfig) + ".png", dpi=150)

numfig += 1

plt.clf()

xf = fftfreq(32, 1)

f = fft(y)

f = 2 \* np.abs(f) / len(f)

f\_7 = fft(quad\_avg\_7)

f\_7 = 2 \* np.abs(f\_7) / len(f\_7)

f\_11 = fft(quad\_avg\_11)

f\_11 = 2 \* np.abs(f\_11) / len(f\_11)

plt.subplot(221)

plt.stem(xf, f)

plt.ylabel(r'S')

plt.xlabel('Частота')

plt.title('Спектр (исх. сигнал)')

plt.subplot(222)

plt.stem(xf, f\_7)

plt.ylabel(r'S')

plt.xlabel('Частота')

plt.title('Спектр (сглаж. 4-й ст. по 7-ти точкам)')

plt.subplot(223)

plt.stem(xf, f\_11)

plt.ylabel(r'S')

plt.xlabel('Частота')

plt.title('Спектр (сглаж. 4-й ст. по 11-ти точкам)')

plt.savefig(str(numfig) + ".png", dpi=150)

numfig += 1

plt.clf()

return

def diff\_avg(x, y, numfig):

diff\_avg = np.convolve(y, np.array([-1, 0, 1]), 'same') / 2

plt.stem(x, y, 'r', markerfmt='ro', label='Исх. сигнал')

plt.stem(x, diff\_avg, 'g', markerfmt='go', label='Сглаж. дифф. 1-го порядка')

plt.xlabel('t')

plt.ylabel(r'$y(t)$')

plt.legend()

plt.savefig(str(numfig) + ".png", dpi=150)

numfig += 1

plt.clf()

xf = fftfreq(32, 1)

f = fft(y)

f = 2 \* np.abs(f) / len(f)

f\_1 = fft(diff\_avg)

f\_1 = 2 \* np.abs(f\_1) / len(f\_1)

plt.subplot(211)

plt.stem(xf, f)

plt.ylabel(r'S')

plt.xlabel('Частота')

plt.title('Спектр (исх. сигнал)')

plt.subplot(212)

plt.stem(xf, f\_1)

plt.ylabel(r'S')

plt.xlabel('Частота')

plt.title('Спектр (сглаж. дифф. 1-го порядка)')

plt.savefig(str(numfig) + ".png", dpi=150)

numfig += 1

plt.clf()

return

def integral\_avg(x, y, numfig):

y\_rect = rect(y)

y\_trap = trap(y)

y\_simps = simpson(y)

plt.stem(x, y, 'r', markerfmt='ro', label='Исх. сигнал')

plt.stem(x, y\_rect, 'g', markerfmt='go', label='Сглаж. интегр. (прямоугольниками)')

plt.stem(x, y\_trap, 'b', markerfmt='bo', label='Сглаж. интегр. (трапециями)')

plt.stem(x, y\_simps, 'm', markerfmt='mo', label='Сглаж. интегр. (Симпсона)')

plt.xlabel('t')

plt.ylabel(r'$y(t)$')

plt.legend()

plt.savefig(str(numfig) + ".png", dpi=150)

numfig += 1

plt.clf()

xf = fftfreq(32, 1)

f = fft(y)

f = 2 \* np.abs(f) / len(f)

f\_rect = fft(y\_rect)

f\_rect = 2 \* np.abs(f\_rect) / len(f\_rect)

f\_trap = fft(y\_trap)

f\_trap = 2 \* np.abs(f\_trap) / len(f\_trap)

f\_simps = fft(y\_simps)

f\_simps = 2 \* np.abs(f\_simps) / len(f\_simps)

plt.subplot(221)

plt.stem(xf, f)

plt.ylabel(r'S')

plt.xlabel('Частота')

plt.title('Спектр (исх. сигнал)')

plt.subplot(222)

plt.stem(xf, f\_rect)

plt.ylabel(r'S')

plt.xlabel('Частота')

plt.title('Спектр (сглаж. интегр. прямоуг.)')

plt.subplot(223)

plt.stem(xf, f\_trap)

plt.ylabel(r'S')

plt.xlabel('Частота')

plt.title('Спектр (сглаж. интегр. трапец.)')

plt.subplot(224)

plt.stem(xf, f\_simps)

plt.ylabel(r'S')

plt.xlabel('Частота')

plt.title('Спектр (сглаж. интегр. Симпсона)')

plt.savefig(str(numfig) + ".png", dpi=500)

numfig += 1

plt.clf()

return

numfig = 0

xx, yy = gen\_analog\_make\_discrete(numfig)

numfig = 10

spectrum(yy, numfig)

numfig = 20

linear\_avg(xx, yy, numfig)

numfig = 30

linear\_avg\_spectrum(xx, yy, numfig)

numfig = 40

square\_avg(xx, yy, numfig)

numfig = 50

quad\_avg(xx, yy, numfig)

numfig = 60

diff\_avg(xx, yy, numfig)

numfig = 70

integral\_avg(xx, yy, numfig)