

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра математического обеспечения и применения ЭВМ

ОТЧЕТ
по лабораторной работе №5
по дисциплине «Цифровая обработка сигналов»
Тема: Исследование результатов фильтрации дискретного сигнала с
помощью нерекурсивных фильтров, построенных на основе
полиномиальных приближений

Студент гр. 9381	_____	Колованов Р.А.
Студент гр. 9381	_____	Семенов А.Н.
Преподаватель	_____	Середа А.-В. И.

Санкт-Петербург
2022

Цель работы.

Получение практических навыков выполнения фильтрации дискретных последовательностей с помощью полиномиальных фильтров, а также анализа получаемых результатов с помощью дискретного преобразования Фурье (ДПФ).

Основные теоретические положения.

Дискретный сигнал: x_0, x_1, \dots, x_{N-1} , как правило, получается при дискретизации аналогового сигнала $s(t)$.

Будем считать, что отсчеты x_k , $k = 0, 1, \dots, N - 1$ дискретного сигнала получены в результате равномерной дискретизации сигнала $s(t)$ с шагом дискретизации, равным единице:

$$\begin{aligned}x_k &= s(t_k), k = 0, 1, \dots, N - 1; \\t_k - t_{k-1} &= T, k = 1, \dots, N - 1; T = 1.\end{aligned}$$

Спектр дискретного сигнала.

Представим дискретный сигнал в виде функции от времени:

$$s(t) = \sum_{k=-\infty}^{\infty} x_k \delta(t - k).$$

Тогда, пользуясь свойствами преобразования Фурье, спектр дискретного сигнала можно представить в виде периодической функции с периодом, равным 2π :

$$S(\omega) = \sum_{k=-\infty}^{\infty} x_k e^{-i\omega k}.$$

Отсюда дискретный сигнал может быть записан в виде:

$$s_d(t) = \frac{1}{T} \sum_{k=-\infty}^{\infty} s(t) e^{i\omega_k t}$$

А его спектр – в виде (см. рис. 0 а):

$$S_d(\omega) = \frac{1}{T} \sum_{k=-\infty}^{\infty} S\left(\omega - \frac{2\pi k}{T}\right)$$

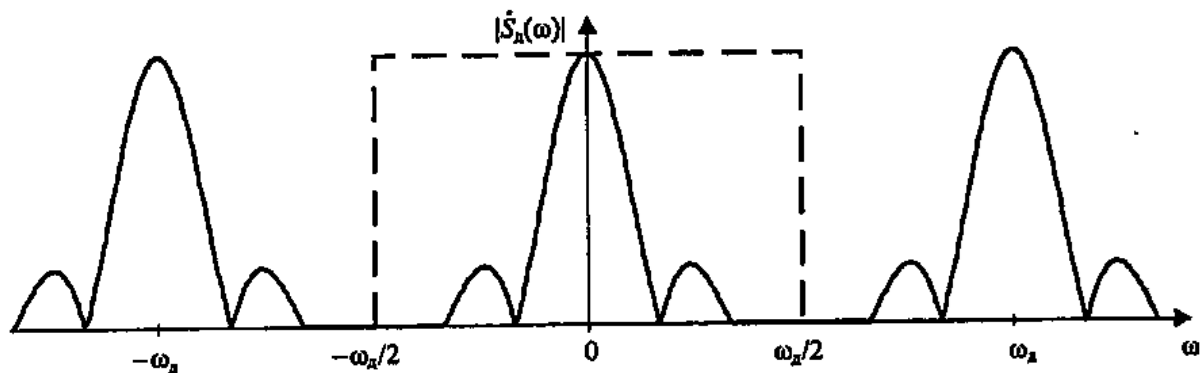


Рисунок 0 (а). График спектра дискретного сигнала.

Из рисунка 0 (а) можно заметить, что расстояние между копиями равно $\frac{2\pi}{T}$.

Теорема Котельникова.

Сигнал $s(t)$, не содержащий гармоник с частотами, превышающими некоторого значения $\hat{\omega} = 2\pi\hat{f}$, может быть представлен без потери информации своими дискретными отсчетами $s(kT)$, взятыми с интервалом T , удовлетворяющим условию:

$$T \leq \frac{1}{2\hat{f}} = \frac{\pi}{\hat{\omega}}$$

При этом восстановление исходного сигнала (рис. 0 б) осуществляется по формуле, представляемой собой разложение $s(t)$ в ряд по системе функций:

$$s(t) = \sum_{k=-\infty}^{\infty} s(kT)\varphi_k(t)$$

где базис Котельникова:

$$\varphi_k(t) = \frac{\sin\left(\pi\frac{t-kT}{T}\right)}{\left(\pi\frac{t-kT}{T}\right)}$$

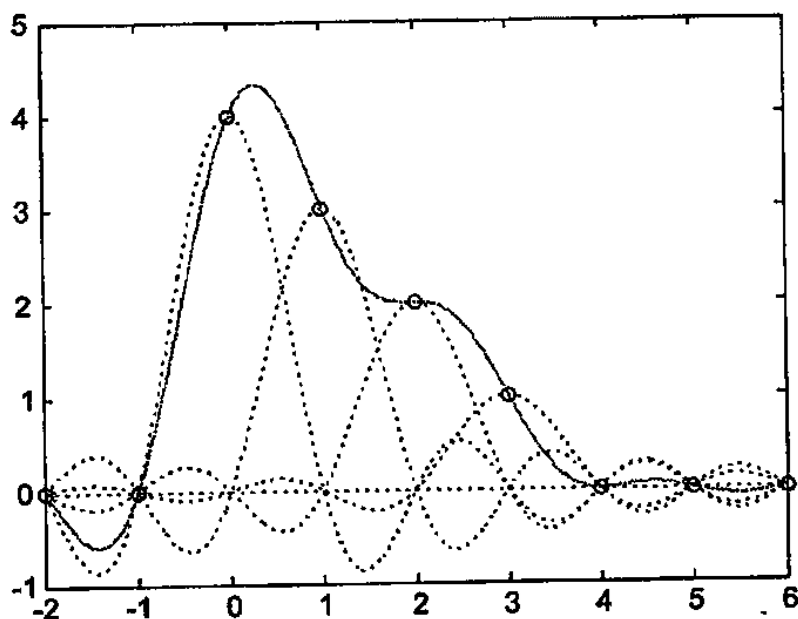


Рис. 0 (б). Восстановление сигнала по его дискретным отчетам.

Дискретное преобразование Фурье.

Пусть последовательность отсчетов $\{x_k\}$ является периодической с периодом N :

$$x_{k+N} = x_k \quad \forall k$$

Рассмотрим фрагмент последовательности из N отсчетов. Например, $\{x_k: k = 0, 1, 2, \dots, N - 1\}$. Тогда дискретная функция

$$s(t) = \sum_{k=-\infty}^{\infty} x_k \delta(t - kT)$$

– тоже будет периодической, с периодом NT . Здесь T – период дискретизации.

Спектр $s(t)$ также должен быть периодическим $T = \frac{2\pi}{T}$ и дискретным с расстоянием между гармониками $-\frac{2\pi}{NT}$.

Поскольку $s(t)$ – периодическая функция, ее можно разложить в ряд Фурье, коэффициенты которого вычисляются по формуле дискретного преобразования Фурье:

$$X(n) = \sum_{k=0}^{N-1} x_k e^{-i\frac{2\pi n}{N}k}, \quad n = 0, 1, 2, \dots, N-1$$

Обратное дискретное преобразование Фурье:

$$x_k = \frac{1}{N} \sum_{n=0}^{N-1} X(n) e^{i\frac{2\pi k}{N}n}, \quad k = 0, 1, 2, \dots, N-1$$

Свойства дискретного преобразования Фурье.

Пусть $\{x(k)\}$, $\{y(k)\}$ – дискретные последовательности с периодом N и ДПФ $\{x(k)\} = \{X(n)\}$, а ДПФ $\{y(k)\} = \{Y(n)\}$. Тогда верны следующие свойства:

- Линейность: $\text{ДПФ}[\alpha\{x(k)\} + \beta\{x(k)\}] = \alpha\{X(n)\} + \beta\{Y(n)\}$
- Задержка: $\{z(k)\} = \{x(k-1)\} \Rightarrow \{Z(n)\} = \{X(n)e^{-i\frac{2\pi n}{N}}\}$
- Симметрия: $X(N-n) = X(-n) = X^*(n)$
- ДПФ произведения:

$$Z(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k)Y(n-k), \quad n = 0, 1, 2, \dots, N-1; \quad Y(k) = Y(k \pm N)$$

- ДПФ вычисляет дискретные отсчеты спектра дискретного сигнала:

$$X(n) = S\left(\frac{2\pi n}{N}\right) = S\left(\omega_d \frac{n}{N}\right), \quad T = 1$$

Постановка задачи.

Для заданного дискретного сигнала применить фильтры, основанные на сглаживании полиномами. Полученные результаты содержательно проинтерпретировать.

Выполнение работы.

1. Сформируем дискретный сигнал посредством дискретизации с шагом $T = 1$ непрерывного сигнала, представляющего собой линейную комбинацию косинусоид вида $A_k \cos(\omega_k t + \varphi_k)$. Всего имеется одиннадцать гармоник с упорядоченными по возрастанию частотами от 0 до π , изменяющимися с шагом $\Delta\omega = 0.1\pi$. Амплитуды гармоник A_k представляют собой целые числа со значениями от 1 до 11, определяемые случайным образом с помощью датчика равномерно распределенных случайных чисел. Начальные фазы φ_k представляют собой случайные числа в промежутке от 0 до 0.5. Дискретная последовательность включает в себя 32 отсчета ($N = 31$).

Исходный аналоговый сигнал:

$$s(t) = \sum_{k=0}^{10} A_k \cos(\omega_k t + \varphi_k).$$

Сформированный дискретный сигнал:

$$x_k = s(t_k), k = 0 \dots N, N = 31, t_k - t_{k-1} = T = 1.$$

При помощи генератора псевдослучайных чисел были сгенерированы значения коэффициентов A_k и φ_k (коэффициенты A_k были нормализованы посредством деления их на сумму полученных A_k):

k	A_k	ω_k	φ_k
0	0.02040816	0	0.16838626
1	0.12244898	0.1π	0.18390124
2	0.10204082	0.2π	0.48714459

3	0.18367347	0.3π	0.42935826
4	0.16326531	0.4π	0.15484798
5	0.08163265	0.5π	0.47822058
6	0.02040816	0.6π	0.08505703
7	0.04081633	0.7π	0.4713078
8	0.02040816	0.8π	0.0025991
9	0.14285714	0.9π	0.06435555
10	0.10204082	π	0.06435555

2. Визуализируем исходные аналоговый и дискретизированный сигналы.

Графики исходного аналогового и дискретного сигналов приведены на рисунках 1 и 2.

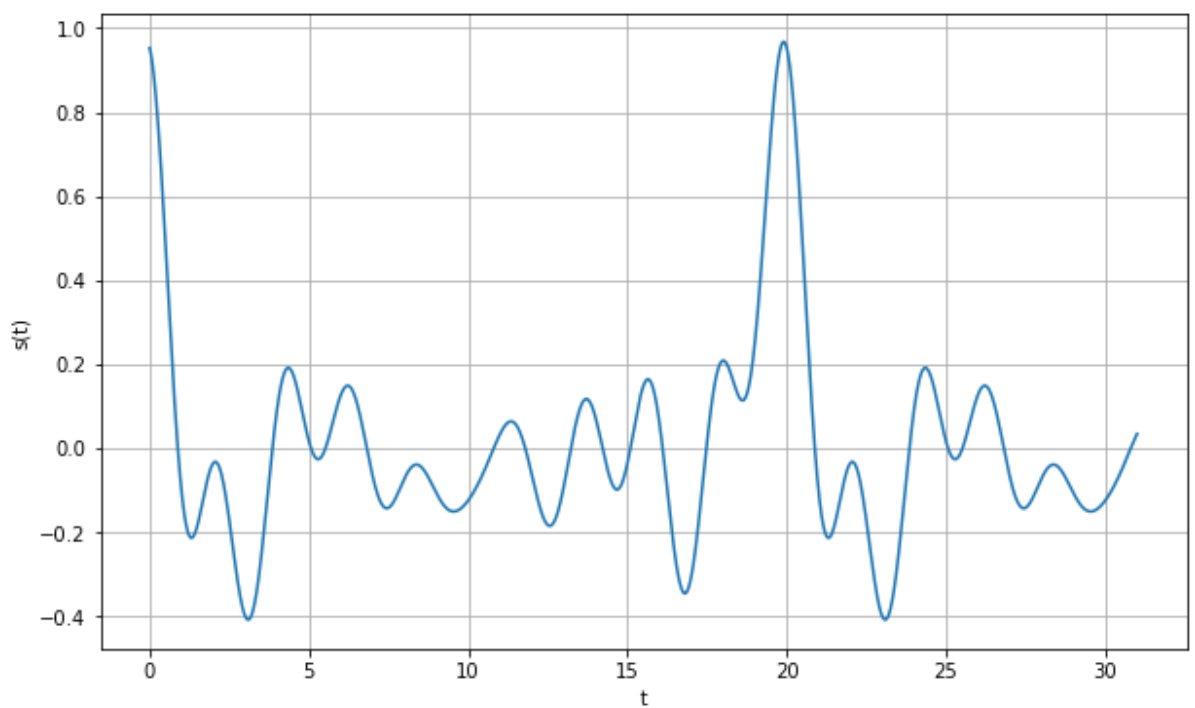


Рисунок 1 – Исходный аналоговый сигнал.

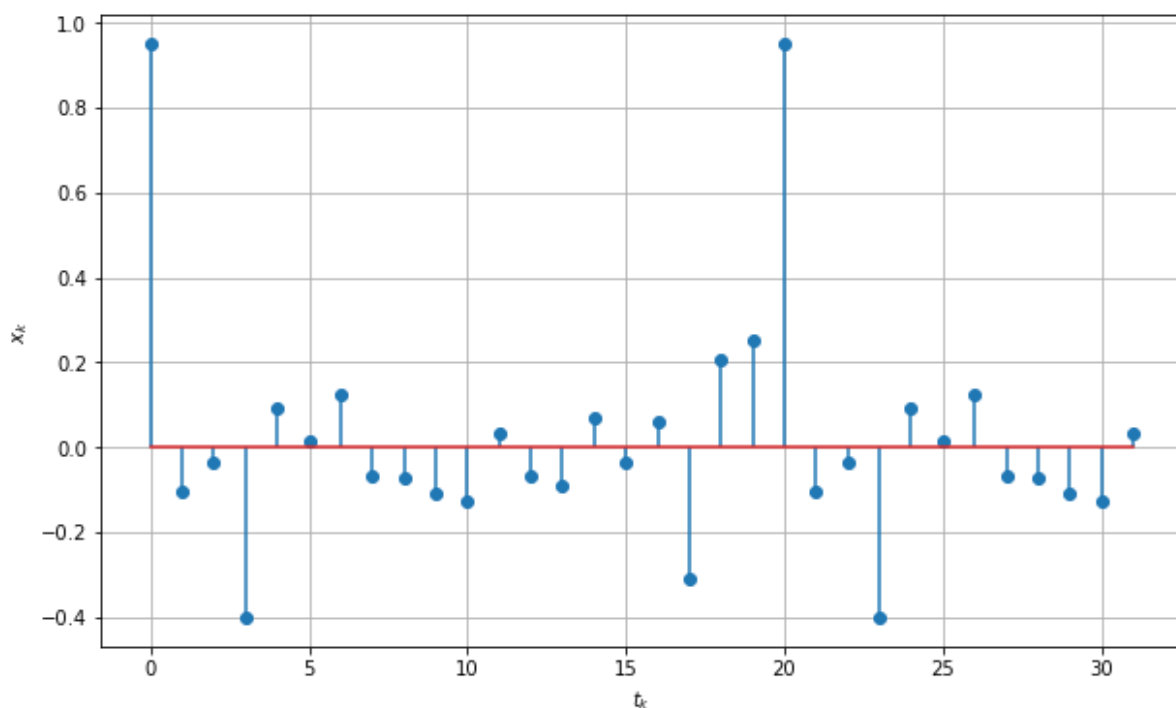


Рисунок 2 – Исходный дискретный сигнал.

3. При помощи дискретного преобразования Фурье найдем дискретные отсчеты спектра дискретного сигнала и визуализируем их.

Пусть последовательность отсчетов $\{x_k\}$ является периодической с периодом N , то есть $x_{k+N} = x_k \forall k$. Рассмотрим фрагмент последовательности $\{x_k: k = 0, 1, 2, \dots, N - 1\}$ из N отсчетов. Тогда дискретная функция

$$s(t) = \sum_{k=-\infty}^{\infty} x_k \delta(t - kT)$$

также будет периодической, с периодом NT , где T – период дискретизации.

Спектр $s(t)$ также должен быть периодическим $T = \frac{2\pi}{T}$ и дискретным с расстоянием между гармониками $-\frac{2\pi}{NT}$.

Поскольку $s(t)$ – периодическая функция, ее можно разложить в ряд Фурье, коэффициенты которого вычисляются по формуле дискретного преобразования Фурье:

$$X(n) = \sum_{k=0}^{N-1} x_k e^{-i\frac{2\pi n}{N}k}, \quad n = 0, 1, 2, \dots, N-1$$

Полученные при помощи дискретного преобразования Фурье дискретные отсчеты спектра исходного дискретного сигнала представлены на рисунке 3.

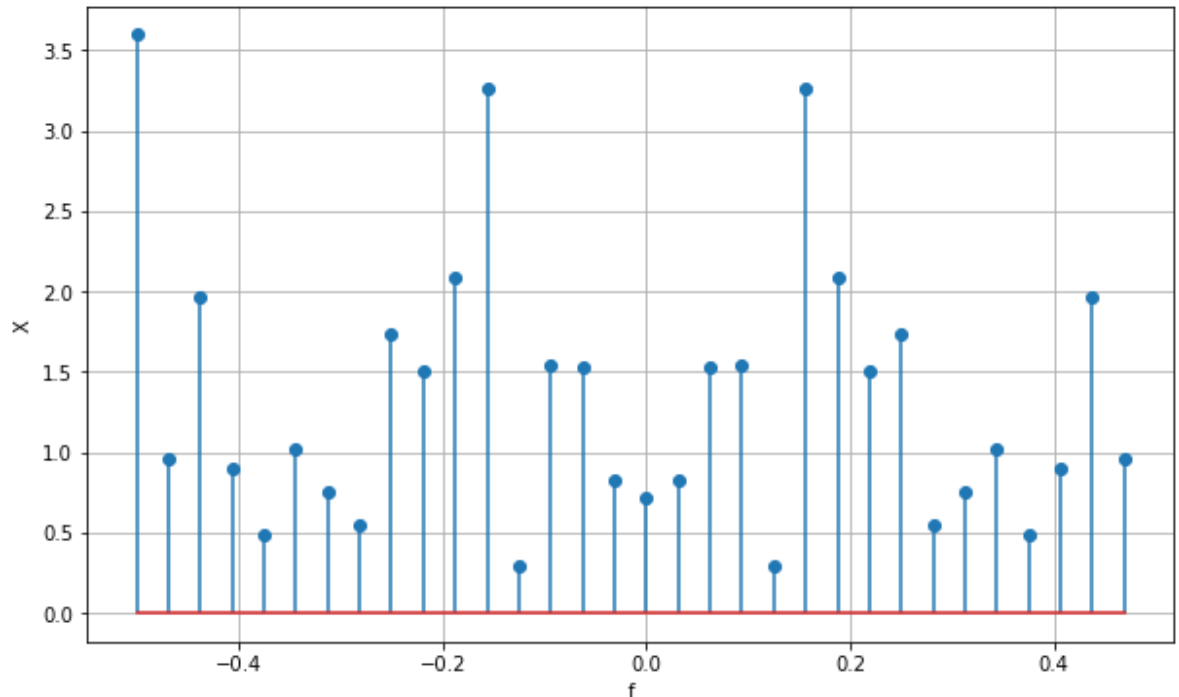


Рисунок 3 – Дискретные отсчеты спектра исходного дискретного сигнала.

Спектр симметричен относительно нуля, он представляет собой разложение исходного сигнала на линейную комбинацию простых синусоидальных функций и отражает амплитуды этих функций на разных частотах. Спектр имеет периодичность с шагом 1.

4. Для дискретного сигнала применим линейное сглаживание по 5-ти и 9-ти точкам, представим формулы для передаточных функций $H(\omega)$ – частотной характеристики фильтра.

Формулы для линейного сглаживания по 5 точкам и 9 точкам:

$$y_{n,5} = \frac{1}{5} \sum_{k=-2}^{k=2} x_{n+k}$$

$$y_{n,9} = \frac{1}{9} \sum_{k=-4}^{k=4} x_{n+k}$$

Передаточные функции для линейного сглаживания по 5-ти и 9-ти точкам:

$$H_5(\omega) = \frac{1}{5} (1 + 2\cos(\omega) + 2\cos(2\omega))$$

$$H_9(\omega) = \frac{1}{9} (1 + 2\cos(\omega) + 2\cos(2\omega) + 2\cos(3\omega) + 2\cos(4\omega))$$

Графики передаточных функций для линейного сглаживания по 5 и 9 точкам представлены на рисунке 4.

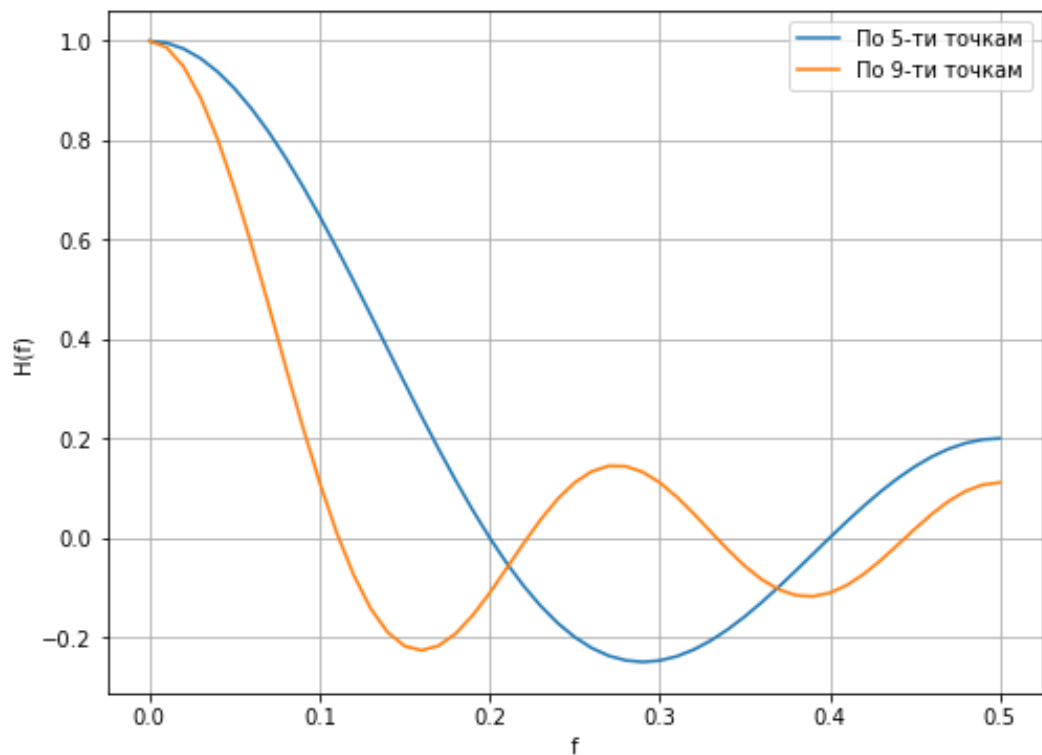


Рисунок 3 – Передаточная функция для линейного сглаживания по 5 и 9 точкам.

5. Визуализируем полученные после фильтрации дискретные сигналы совместно с исходным дискретным сигналом.

Графики исходного сигнала и сигнала после применения линейного сглаживания по 5-ти и по 9-ти точкам представлены на рисунке 5 и 6.

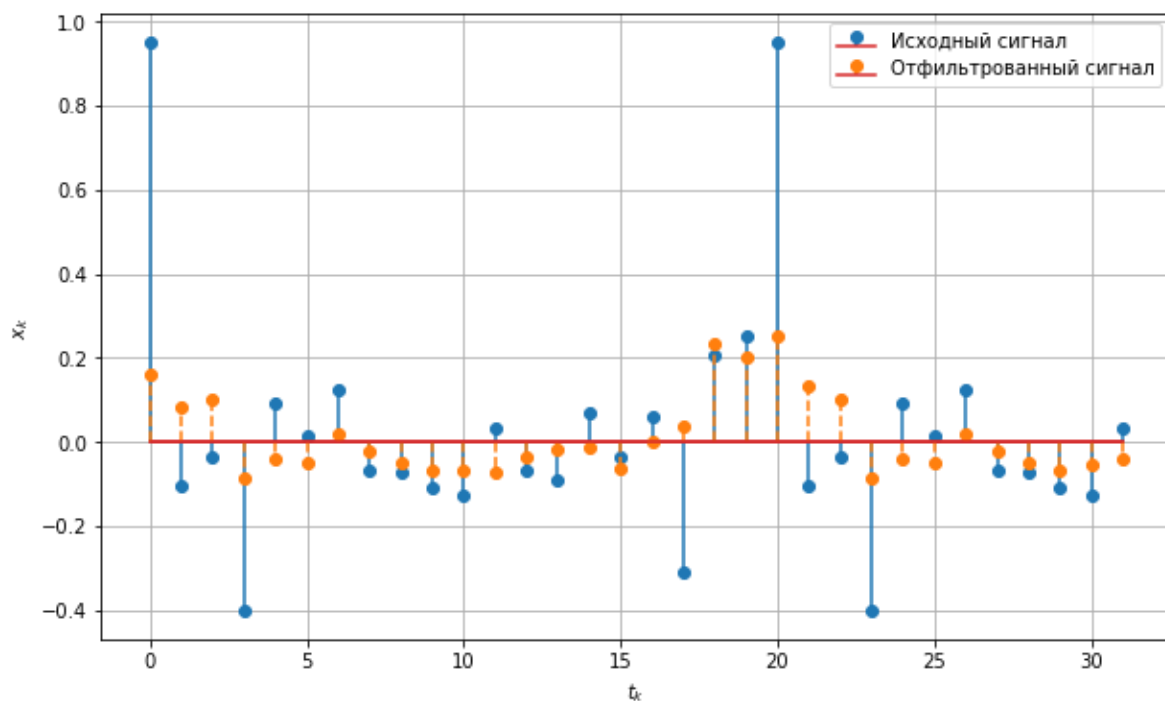


Рисунок 5 – Исходный дискретный сигнал и сигнал после применения сглаживания по 5-ти точкам.

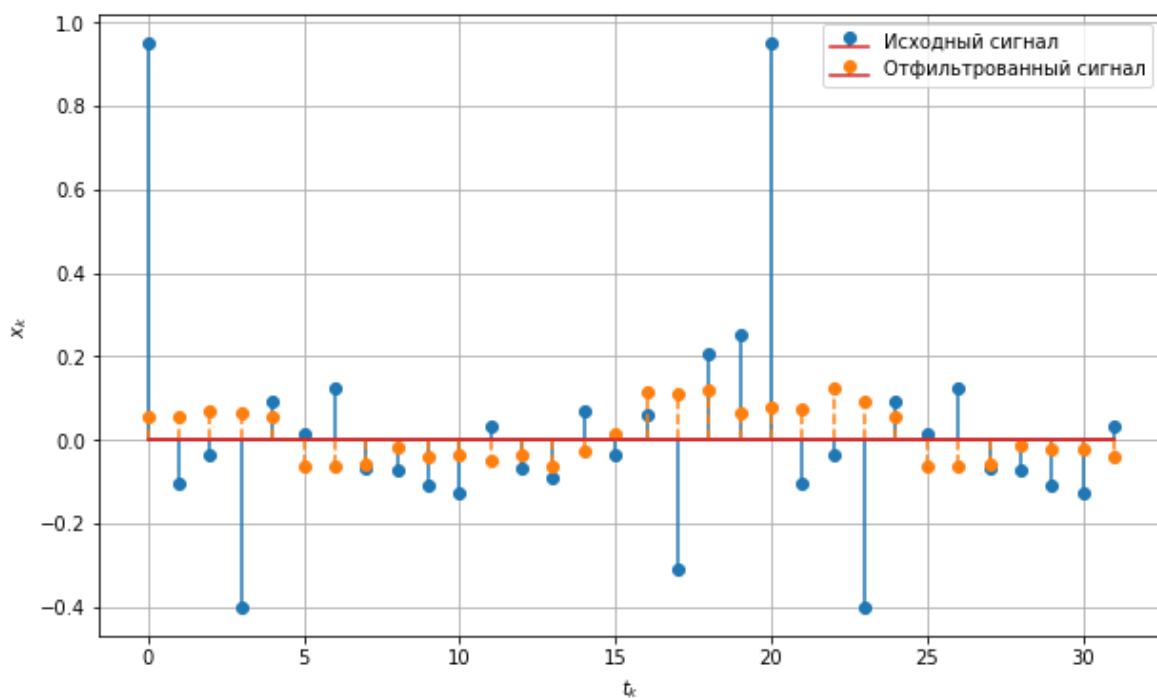


Рисунок 6 – Исходный дискретный сигнал и сигнал после применения сглаживания по 9-ти точкам.

6. При помощи ДПФ найдем дискретные отсчеты спектра дискретного сигнала после его фильтрации и визуализируем их совместно с отсчетами спектра исходного дискретного сигнала.

Графики дискретных отсчетов спектра, полученных при помощи ДПФ, для исходного сигнала и сигнала после фильтрации представлены на рисунке 7 и 8.

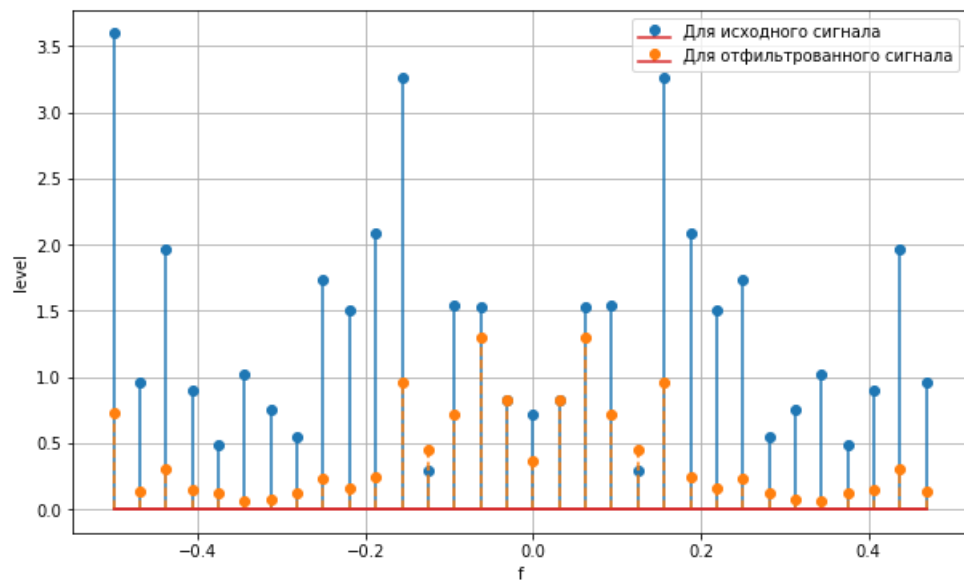


Рисунок 7 – Дискретные отсчеты спектра для исходного дискретного сигнала и сигнала после применения сглаживания по 5-ти точкам.

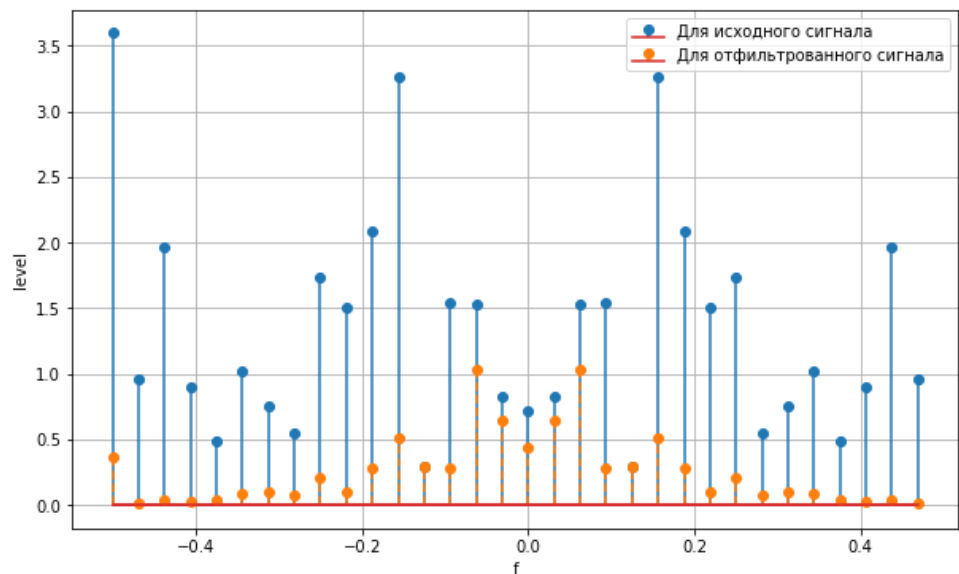


Рисунок 8 – Дискретные отсчеты спектра для исходного дискретного сигнала и сигнала после применения сглаживания по 9-ти точкам.

7. Проанализируем результаты на соответствие значениям соответствующих передаточных функций $H(\omega)$:

Из полученного спектра видно, что без ослабления пропускается только сигнал постоянного уровня (нулевой частоты). Сигналы с частотами, близкими к 0, ослабевают не сильно, а сигнал с большими частотами значительно ослабевает. С увеличением числа точек полоса пропускания становится меньше. Графики передаточных функций, представленные на рисунке 3, подтверждают данный вывод.

8. Повторим действия из пунктов 4 – 7 для других фильтров.

А. Фильтр сглаживания полиномом 2-ой степени по 5 и 9 узлам:

Формулы для сглаживания полиномом второй степени по 5-ти точкам и 9-ти точкам:

$$y_{n,5} = \frac{1}{35}(-3x_{n-2} + 12x_{n-1} + 17x_n + 12x_{n+1} - 3x_{n+2})$$

$$y_{n,9} = \frac{1}{231}(-21x_{n-4} + 14x_{n-3} + 39x_{n-2} + 54x_{n-1} + 59x_n + 54x_{n+1} + 39x_{n+2} + 14x_{n+3} - 21x_{n+4})$$

Передаточные функции для сглаживания полиномом второй степени по 5-ти точкам и 9-ти точкам:

$$H_5(\omega) = \frac{(17 + 24 \cos(\omega) - 6 \cos(2\omega))}{35}$$

$$H_9(\omega) = \frac{(59 + 108 \cos(\omega) + 78 \cos(2\omega) + 28 \cos(3\omega) - 42 \cos(4\omega))}{231}$$

Графики передаточных функций для сглаживания полиномом второй степени по 5 и 9 точкам представлены на рисунке 9.

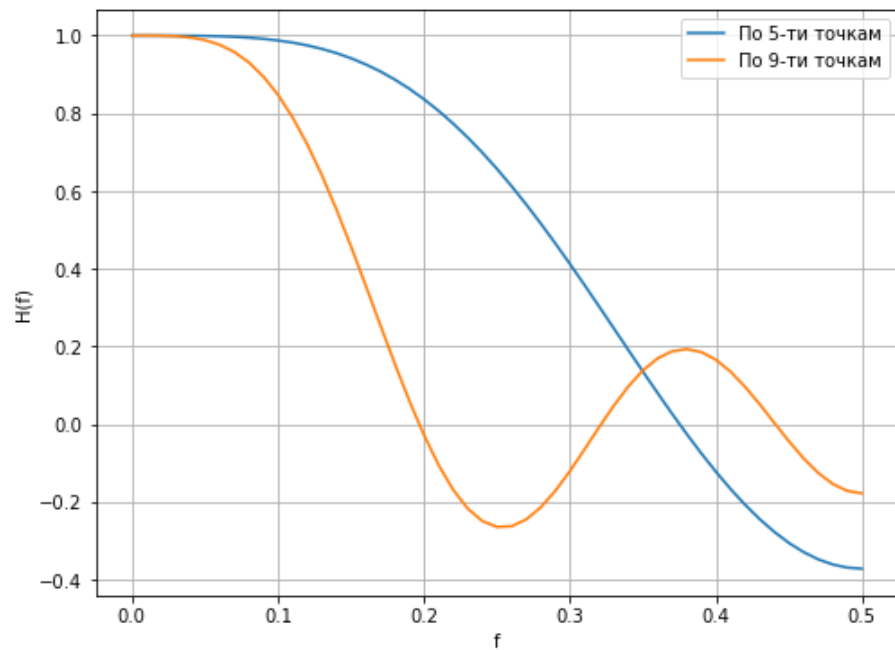


Рисунок 9 – Передаточная функция для сглаживания полиномом второй степени по 5 и 9 точкам.

Визуализируем полученные после фильтрации дискретные сигналы совместно с исходным дискретным сигналом.

Графики исходного сигнала и сигнала после применения сглаживания полиномом второй степени по 5-ти и по 9-ти точкам представлены на рисунке 10 и 11.

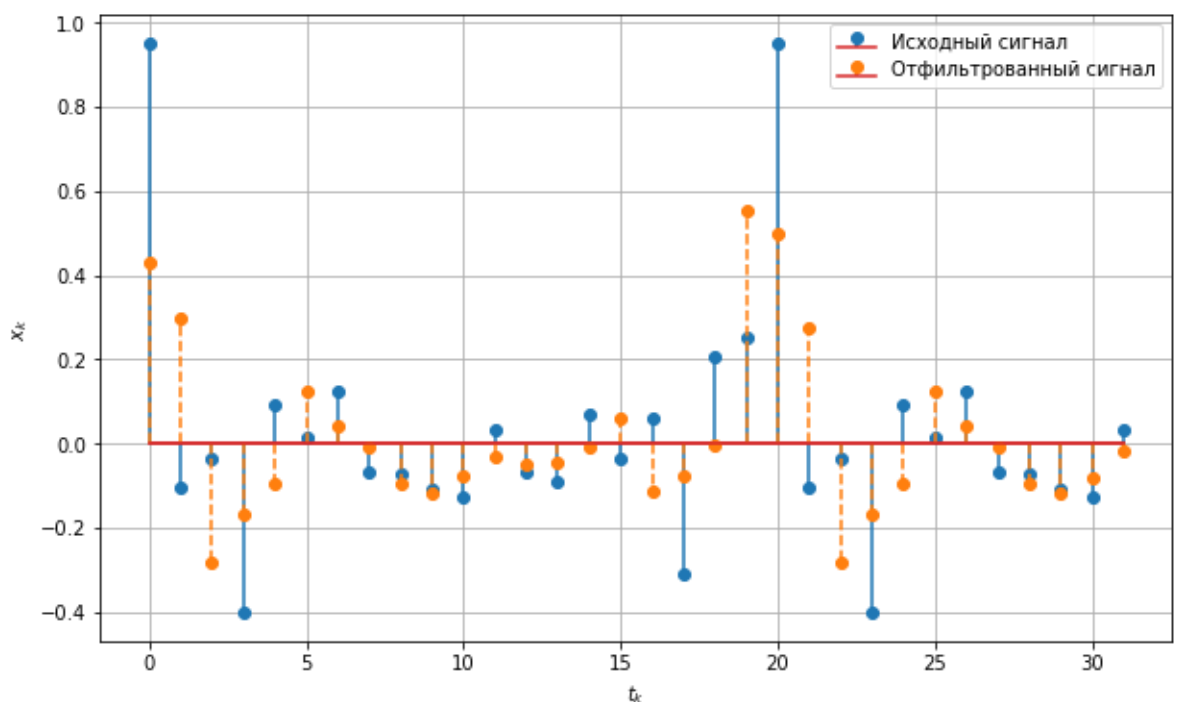


Рисунок 10 – Исходный дискретный сигнал и сигнал после применения сглаживания полиномом второй степени по 5-ти точкам.

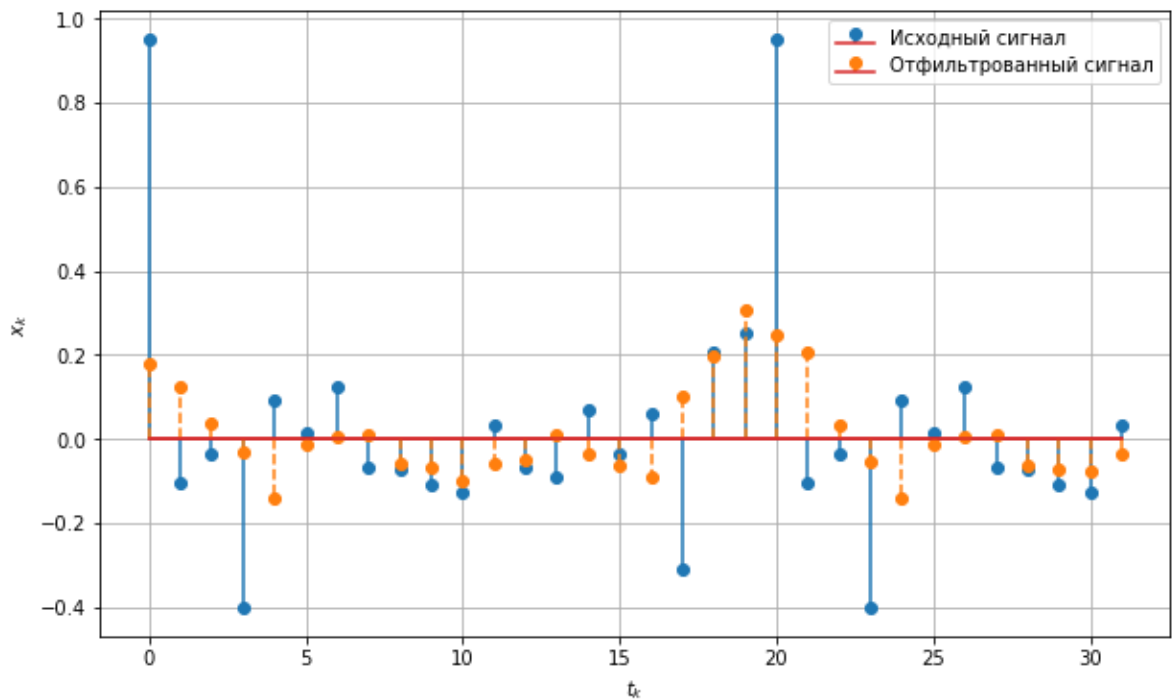


Рисунок 11 – Исходный дискретный сигнал и сигнал после применения сглаживания полиномом второй степени по 9-ти точкам.

При помощи ДПФ найдем дискретные отсчеты спектра дискретного сигнала после его фильтрации и визуализируем их совместно с отсчетами спектра исходного дискретного сигнала.

Графики дискретных отсчетов спектра, полученных при помощи ДПФ, для исходного сигнала и сигнала после фильтрации представлены на рисунке 12 и 13.

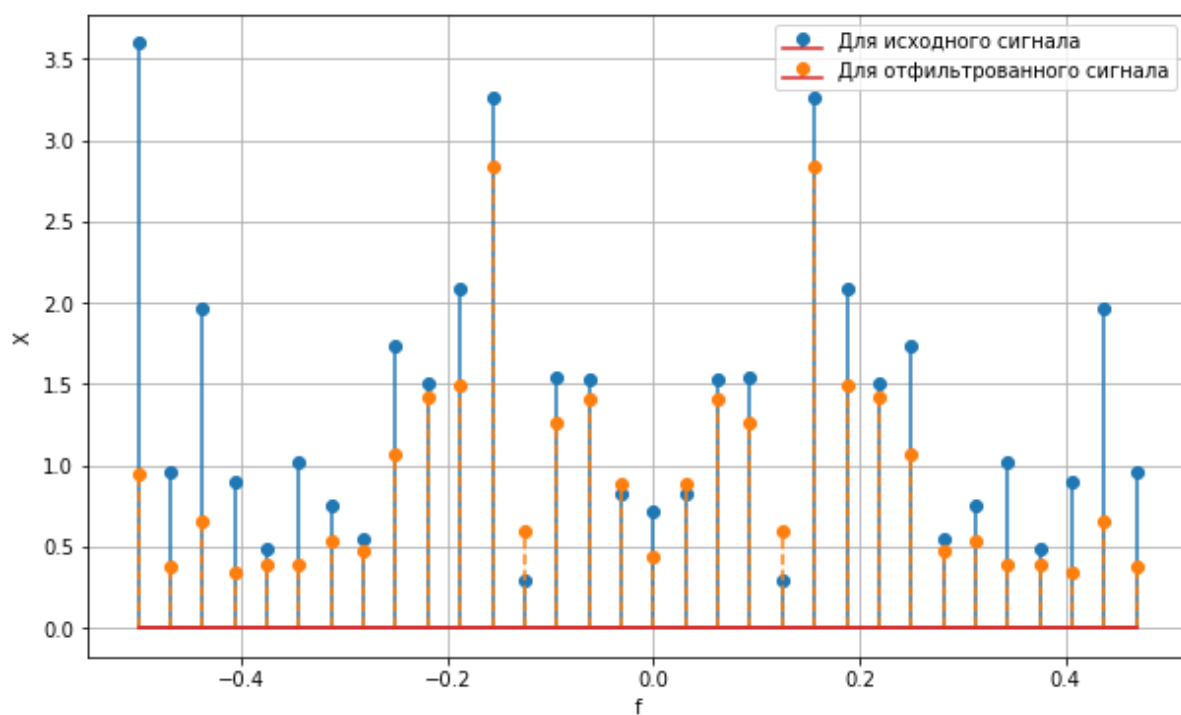


Рисунок 12 – Дискретные отсчеты спектра для исходного дискретного сигнала и сигнала после применения сглаживания полиномом второй степени по 5-ти точкам.

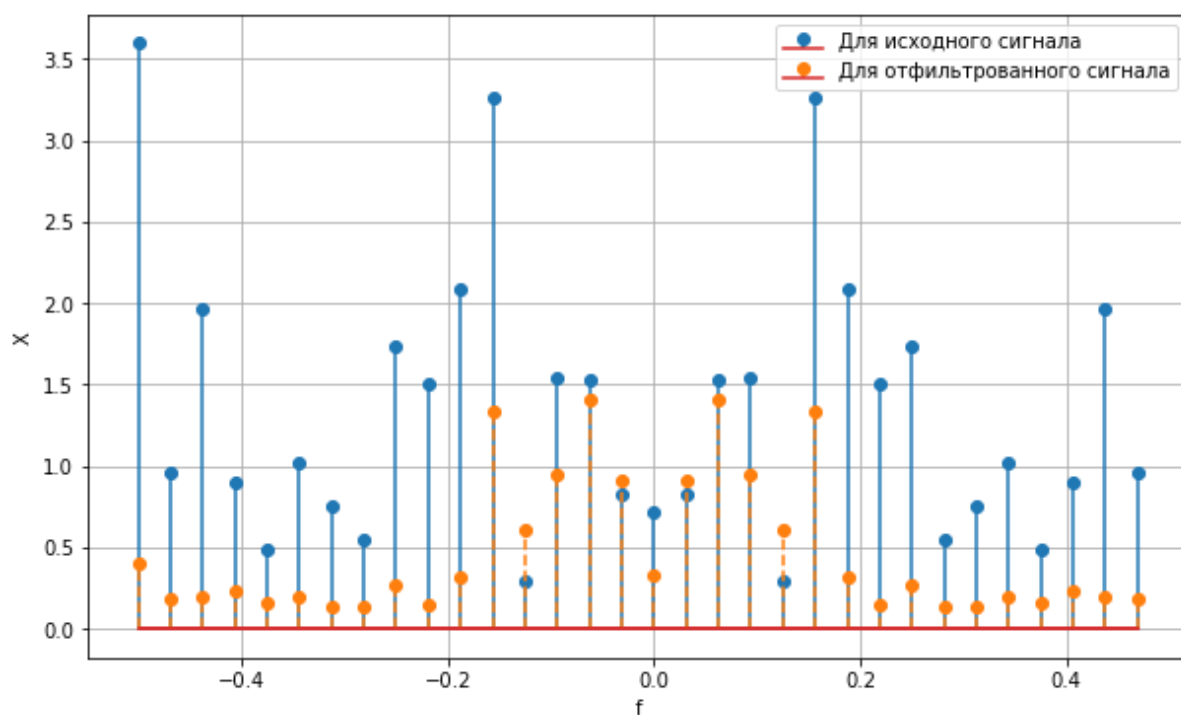


Рисунок 13 – Дискретные отсчеты спектра для исходного дискретного сигнала и сигнала после применения сглаживания полиномом второй степени по 9-ти точкам.

Проанализируем результаты на соответствие значениям соответствующих передаточных функций $H(\omega)$:

Из графиков передаточных функций на рисунке 9 видно, что при увеличении числа точек увеличивается крутизна среза и уменьшается полоса пропускания частот.

Также присутствует более продолговатая часть в полосе пропускания около нулевых частот, в отличие от линейного фильтра, поэтому сигналы низкой частоты пропускаются с малой потерей амплитуды в более широком диапазоне по сравнению с линейным фильтром, что видно на графиках спектра.

Видно, что графики передаточных функций, представленные на рисунке 9, объясняют изменение амплитуд в спектре сигнала.

Б. Фильтр сглаживания полиномом 4-ой степени по 7 и 11 узлам:

Формулы для сглаживания полиномом 4-ой степени по 7-ми точкам и 11-ти точкам:

$$y_{n,7} = \frac{1}{231}(5x_{n-3} - 30x_{n-2} + 75x_{n-1} + 131x_n + 75x_{n+1} - 30x_{n+2} + 5x_{n+3})$$

$$y_{n,11} = \frac{1}{429}(18x_{n-5} - 45x_{n-4} - 10x_{n-3} + 60x_{n-2} + 120x_{n-1} + 143x_n + \\ + 120x_{n+1} + 60x_{n+2} + 10x_{n+3} - 45x_{n+4} + 18x_{n+5})$$

Передаточные функции для сглаживания полиномом 4-ой степени по 7-ми точкам и 11-ти точкам:

$$H_7(\omega) = \frac{(131 + 150 \cos(\omega) - 60 \cos(2\omega) + 10 \cos(3\omega))}{231}$$

$$H_{11}(\omega) = \frac{(143 + 240 \cos(\omega) + 120 \cos(2\omega) - 20 \cos(3\omega) - 90 \cos(4\omega) + 36 \cos(5\omega))}{429}$$

Графики передаточных функций для сглаживания полиномом 4-ой степени по 7-ми точкам и 11-ти точкам представлены на рисунке 14.

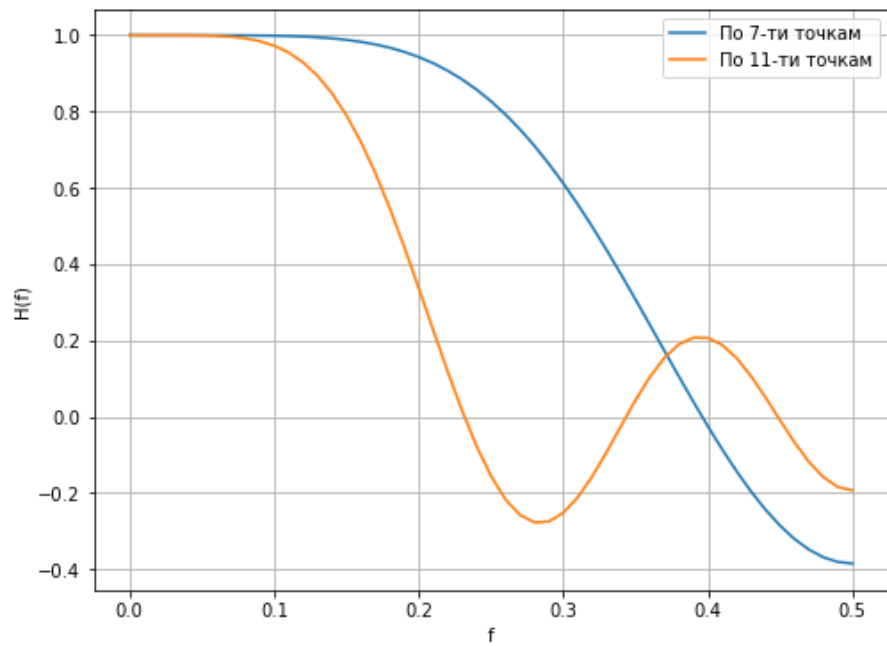


Рисунок 14 – Передаточная функция для сглаживания полиномом 4-ой степени по 7-ми точкам и 11-ти точкам.

Визуализируем полученные после фильтрации дискретные сигналы совместно с исходным дискретным сигналом.

Графики исходного сигнала и сигнала после применения сглаживания полиномом 4-ой степени по 7-ми точкам и 11-ти точкам представлены на рисунке 15 и 16.

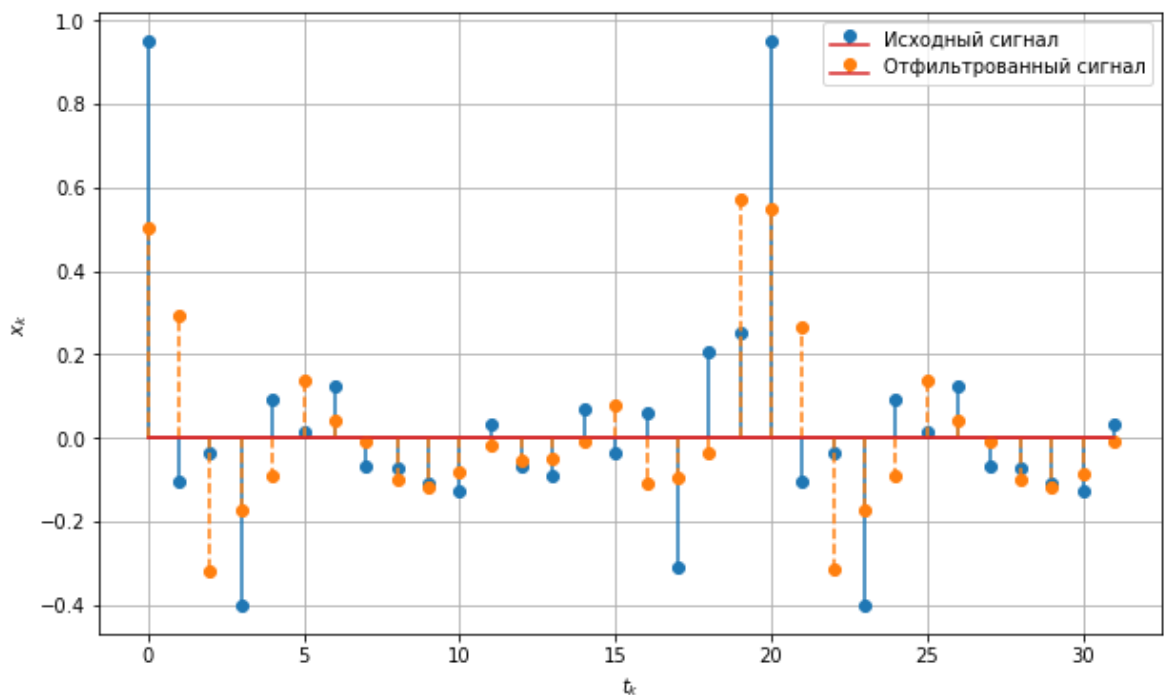


Рисунок 15 – Исходный дискретный сигнал и сигнал после применения сглаживания полиномом 4-ой степени по 7-ти точкам.

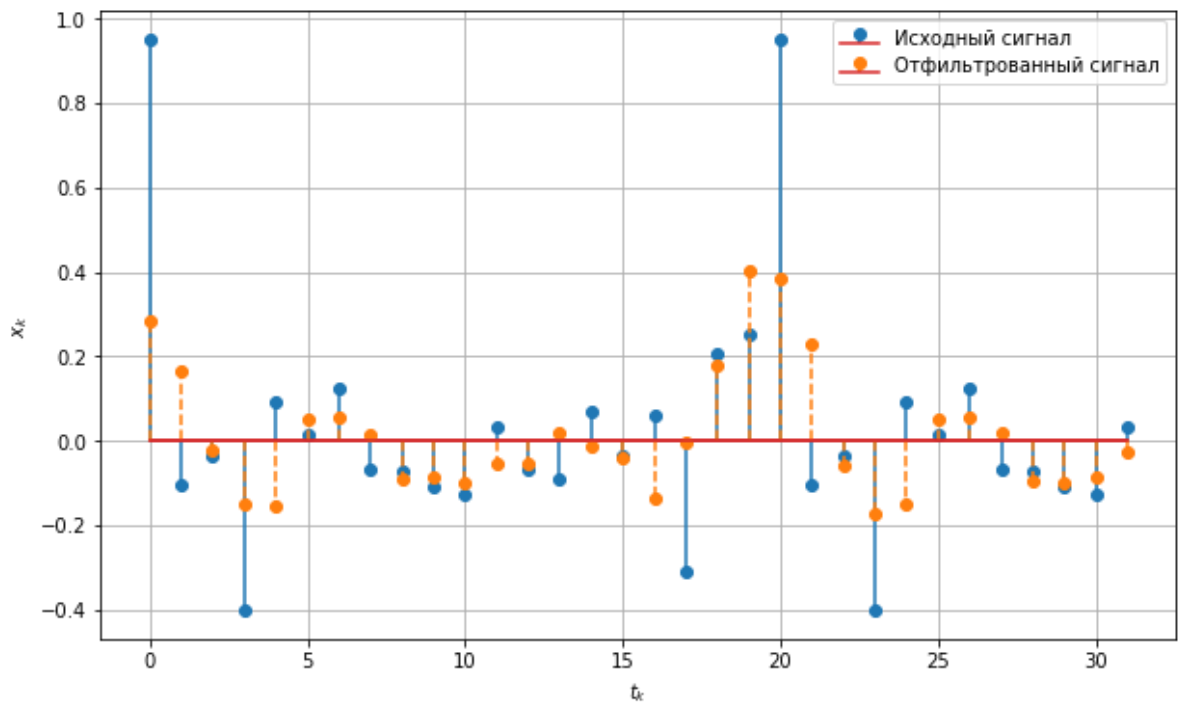


Рисунок 16 – Исходный дискретный сигнал и сигнал после применения сглаживания полиномом 4-ой степени по 11-ти точкам.

При помощи ДПФ найдем дискретные отсчеты спектра дискретного сигнала после его фильтрации и визуализируем их совместно с отсчетами спектра исходного дискретного сигнала.

Графики дискретных отсчетов спектра, полученных при помощи ДПФ, для исходного сигнала и сигнала после фильтрации представлены на рисунке 17 и 18.

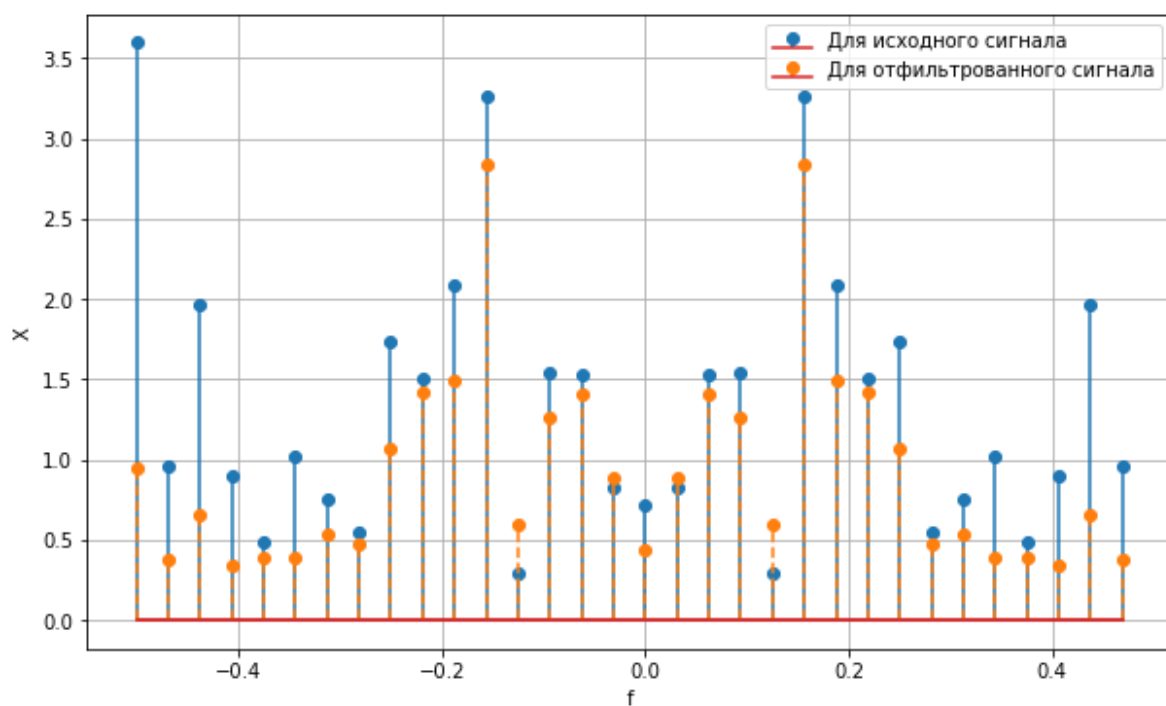


Рисунок 17 – Дискретные отсчеты спектра для исходного дискретного сигнала и сигнала после применения сглаживания полиномом 4-ой степени по 7-ти точкам.

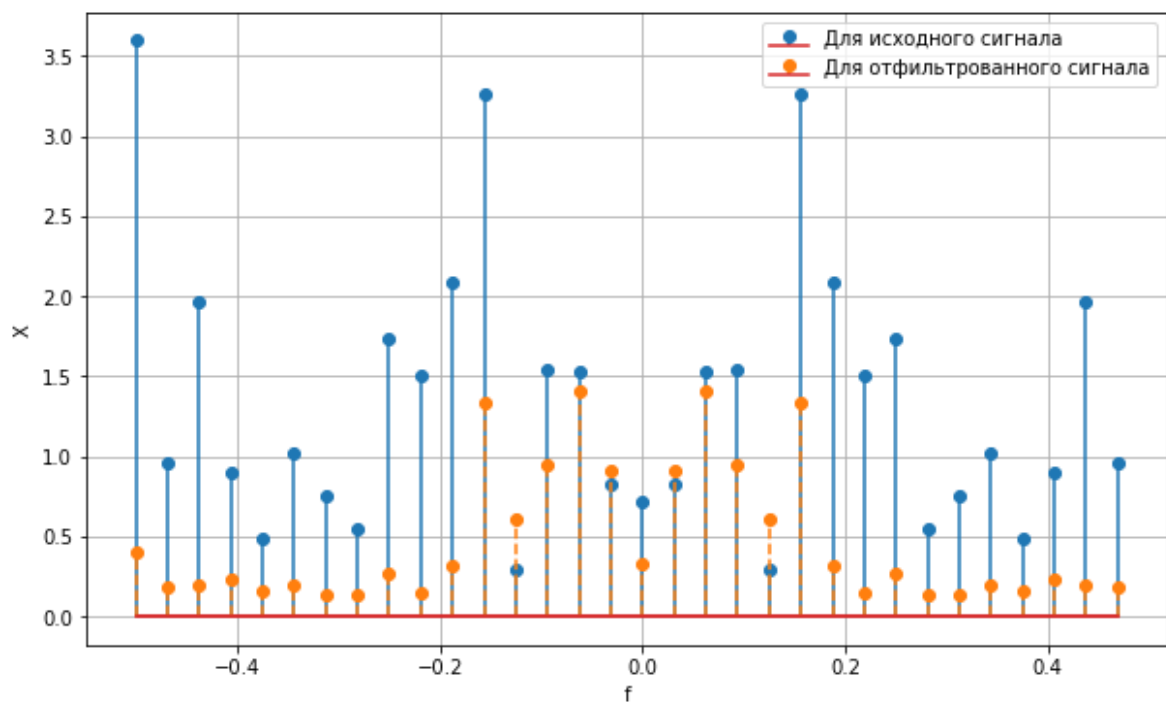


Рисунок 18 – Дискретные отсчеты спектра для исходного дискретного сигнала и сигнала после применения сглаживания полиномом 4-ой степени по 11-ти точкам.

Проанализируем результаты на соответствие значениям соответствующих передаточных функций $H(\omega)$:

Из графика передаточных функций на рисунке 14 видно, что при увеличении числа точек увеличивается крутизна среза и уменьшается полоса пропускания частот. Полученные передаточные функции схожи с передаточными функциями для полинома второй степени, только в данном случае полоса пропускания еще шире.

Здесь также присутствует более продолговатая часть в полосе пропускания около нулевых частот, в отличие от линейного фильтра, поэтому сигналы низкой частоты пропускаются с малой потерей амплитуды в более широком диапазоне по сравнению с линейным фильтром, что видно на графиках спектра.

Видно, что графики передаточных функций, представленные на рисунке 14, объясняют изменение амплитуд в спектре сигнала.

Выводы.

В ходе выполнения лабораторной работы была проведена фильтрация дискретных последовательностей при помощи полиномиальных фильтров, а также был проведен анализ получаемых результатов при помощи дискретного преобразования Фурье (ДПФ).

В результате был сгенерирован аналоговый сигнал, после чего он был дискретизирован. Для полученного дискретного сигнала был построен спектр, представленный в виде набора дискретных отсчетов. Было определено, что спектр показывает наличие в исходном сигнале множества различных частот.

Были применены фильтры линейного сглаживания, сглаживания полиномом 2-й и 4-й степени по различным количествам точек (узлов), построены графики исходного и отфильтрованного сигнала и спектра. В результате по спектру было определено, что рассматриваемые фильтры обеспечивают фильтрацию (уменьшение амплитуды) высоких частот, при этом ширина полосы пропускания уменьшается с увеличением числа точек. Равномерность передачи сигнала в полосе пропускания увеличивается при использовании полиномов более высоких порядков.

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД ПРОГРАММЫ

```
import math
import numpy as np
import matplotlib.pyplot as plt
from scipy.fftpack import fft, fftfreq, ifft, fftshift, rfft

N = 31
T = 1

analog_t_values = np.arange(0, N + 0.01, 0.01)
t_values = np.arange(0, N + T, T)

w_k = np.arange(0, np.pi + 0.1 * np.pi, 0.1 * np.pi)
A_k = np.random.randint(1, 11, 11)
A_k = A_k / sum(A_k)
fi_k = np.random.random(size=11) / 2

print(w_k)
print(A_k)
print(fi_k)

# [0.          0.31415927  0.62831853  0.9424778   1.25663706  1.57079633  1.88495559
#  2.19911486  2.51327412  2.82743339  3.14159265]
# [0.02040816  0.12244898  0.10204082  0.18367347  0.16326531  0.08163265  0.02040816
#  0.04081633  0.02040816  0.14285714  0.10204082]
# [0.16838626  0.18390124  0.48714459  0.42935826  0.15484798  0.47822058  0.08505703
#  0.4713078   0.0025991   0.06435555  0.11157867]

def s(t):
    r = 0
    for i in range(11):
        r += A_k[i] * math.cos(w_k[i] * t + fi_k[i])
    return r

plt.figure(figsize=(10, 6))
plt.plot(analog_t_values, [s(t) for t in analog_t_values])
plt.ylabel("s(t)")
plt.xlabel("t")
plt.grid()
```

```

plt.show()

plt.figure(figsize=(10, 6))
plt.stem(t_values, [s(t) for t in t_values])
plt.ylabel(r"$x_k$")
plt.xlabel(r"$t_k$")
plt.grid()
plt.show()

def dft(x):
    x = np.asarray(x, dtype=float)
    N = x.shape[0]
    n = np.arange(N)
    k = n.reshape((N, 1))
    M = np.exp(-2j * np.pi * k * n / N)
    return np.dot(M, x)

f_values = fftfreq(N + 1, T)
X_values = np.abs(dft([s(t) for t in t_values]))

plt.figure(figsize=(10, 6))
plt.stem(f_values, X_values)
plt.ylabel("X")
plt.xlabel("f")
plt.grid()
plt.show()

x_values = [s(t) for t in t_values]
x_values_5 = np.convolve(x_values, np.ones(5), 'same') / 5
x_values_9 = np.convolve(x_values, np.ones(9), 'same') / 9

plt.figure(figsize=(10, 6))
plt.stem(t_values, x_values, label="Исходный сигнал")
plt.stem(t_values, x_values_5, linefmt="C1--", markerfmt="C1o",
label="Отфильтрованный сигнал")
plt.ylabel(r"$x_k$")
plt.xlabel(r"$t_k$")
plt.legend()
plt.grid()
plt.show()

plt.figure(figsize=(10, 6))

```



```

plt.stem(t_values, x_values, label="Исходный сигнал")
plt.stem(t_values, x_values_9, linefmt="C1--", markerfmt="C1o",
label="Отфильтрованный сигнал")
plt.ylabel(r"$x_k$")
plt.xlabel(r"$t_k$")
plt.legend()
plt.grid()
plt.show()

f_values = fftfreq(N + 1, T)
X_values_5 = np.abs(dft(x_values_5))

plt.figure(figsize=(10, 6))
plt.stem(f_values, X_values, label="Для исходного сигнала")
plt.stem(f_values, X_values_5, linefmt="C1--", markerfmt="C1o", label="Для
отфильтрованного сигнала")
plt.ylabel("X")
plt.xlabel("f")
plt.legend()
plt.grid()
plt.show()

f_values = fftfreq(N + 1, T)
X_values_9 = np.abs(dft(x_values_9))

plt.figure(figsize=(10, 6))
plt.stem(f_values, X_values, label="Для исходного сигнала")
plt.stem(f_values, X_values_9, linefmt="C1--", markerfmt="C1o", label="Для
отфильтрованного сигнала")
plt.ylabel("X")
plt.xlabel("f")
plt.legend()
plt.grid()
plt.show()

def h_5(f):
    return (1 + 2 * math.cos(2 * math.pi * f) + 2 * math.cos(4 * math.pi * f)) /
5

def h_9(f):
    return (1 + 2 * math.cos(2 * math.pi * f) + 2 * math.cos(4 * math.pi * f) +
2 * math.cos(6 * math.pi * f) + 2 * math.cos(8 * math.pi * f)) / 9

```

```

f_values_for_h = np.arange(0, 0.5 + 0.01, 0.01)

plt.figure(figsize=(8, 6))
plt.plot(f_values_for_h, [h_5(f) for f in f_values_for_h], label="По 5-ти точкам")
plt.plot(f_values_for_h, [h_9(f) for f in f_values_for_h], label="По 9-ти точкам")
plt.ylabel("H(f)")
plt.xlabel("f")
plt.legend()
plt.grid()
plt.show()

def h_5(f):
    return (17 + 24 * math.cos(2 * math.pi * f) - 6 * math.cos(2 * 2 * math.pi *
f)) / 35

def h_9(f):
    return (59 + 108 * math.cos(2 * math.pi * f) + 78 * math.cos(2 * 2 * math.pi
* f) + 28 * math.cos(3 * 2 * math.pi * f) - 42 * math.cos(4 * 2 * math.pi * f))
/ 231

f_values_for_h = np.arange(0, 0.5 + 0.01, 0.01)

plt.figure(figsize=(8, 6))
plt.plot(f_values_for_h, [h_5(f) for f in f_values_for_h], label="По 5-ти точкам")
plt.plot(f_values_for_h, [h_9(f) for f in f_values_for_h], label="По 9-ти точкам")
plt.ylabel("H(f)")
plt.xlabel("f")
plt.legend()
plt.grid()
plt.show()

x_values = [s(t) for t in t_values]
x_values_5 = np.convolve(x_values, np.array([-3, 12, 17, 12, -3]), 'same') / 35
x_values_9 = np.convolve(x_values, np.array([-21, 14, 39, 54, 59, 54, 39, 14, -
21]), 'same') / 231

plt.figure(figsize=(10, 6))
plt.stem(t_values, x_values, label="Исходный сигнал")
plt.stem(t_values, x_values_5, linefmt="C1--", markerfmt="C1o",
label="Отфильтрованный сигнал")
plt.ylabel(r"$x_k$")

```

```

plt.xlabel(r"$t_k$")
plt.legend()
plt.grid()
plt.show()

plt.figure(figsize=(10, 6))
plt.stem(t_values, x_values, label="Исходный сигнал")
plt.stem(t_values, x_values_9, linefmt="C1--", markerfmt="C1o",
label="Отфильтрованный сигнал")
plt.ylabel(r"$x_k$")
plt.xlabel(r"$t_k$")
plt.legend()
plt.grid()
plt.show()

f_values = fftfreq(N + 1, T)
X_values_5 = np.abs(dft(x_values_5))

plt.figure(figsize=(10, 6))
plt.stem(f_values, X_values, label="Для исходного сигнала")
plt.stem(f_values, X_values_5, linefmt="C1--", markerfmt="C1o", label="Для
отфильтрованного сигнала")
plt.ylabel("X")
plt.xlabel("f")
plt.legend()
plt.grid()
plt.show()

f_values = fftfreq(N + 1, T)
X_values_9 = np.abs(dft(x_values_9))

plt.figure(figsize=(10, 6))
plt.stem(f_values, X_values, label="Для исходного сигнала")
plt.stem(f_values, X_values_9, linefmt="C1--", markerfmt="C1o", label="Для
отфильтрованного сигнала")
plt.ylabel("X")
plt.xlabel("f")
plt.legend()
plt.grid()
plt.show()

def h_7(f):

```

```

    return (131 + 150 * math.cos(2 * math.pi * f) - 60 * math.cos(2 * 2 * math.pi
* f) + 10 * math.cos(3 * 2 * math.pi * f)) / 231

def h_11(f):
    return (143 + 240 * math.cos(2 * math.pi * f) + 120 * math.cos(2 * 2 * math.pi
* f) - 20 * math.cos(3 * 2 * math.pi * f) - 90 * math.cos(4 * 2 * math.pi * f) +
36 * math.cos(5 * 2 * math.pi * f)) / 429

f_values_for_h = np.arange(0, 0.5 + 0.01, 0.01)

plt.figure(figsize=(8, 6))
plt.plot(f_values_for_h, [h_7(f) for f in f_values_for_h], label="По 7-ти точкам")
plt.plot(f_values_for_h, [h_11(f) for f in f_values_for_h], label="По 11-ти
точкам")
plt.ylabel("H(f)")
plt.xlabel("f")
plt.legend()
plt.grid()
plt.show()

x_values = [s(t) for t in t_values]
x_values_7 = np.convolve(x_values, np.array([5, -30, 75, 131, 75, -30, 5]),
'same') / 231
x_values_11 = np.convolve(x_values, np.array([13, -45, -10, 60, 120, 143, 120,
60, -10, -45, 13]), 'same') / 429

plt.figure(figsize=(10, 6))
plt.stem(t_values, x_values, label="Исходный сигнал")
plt.stem(t_values, x_values_7, linefmt="C1--", markerfmt="C1o",
label="Отфильтрованный сигнал")
plt.ylabel(r"$x_k$")
plt.xlabel(r"$t_k$")
plt.legend()
plt.grid()
plt.show()

plt.figure(figsize=(10, 6))
plt.stem(t_values, x_values, label="Исходный сигнал")
plt.stem(t_values, x_values_11, linefmt="C1--", markerfmt="C1o",
label="Отфильтрованный сигнал")
plt.ylabel(r"$x_k$")
plt.xlabel(r"$t_k$")

```

```

plt.legend()
plt.grid()
plt.show()

f_values = fftfreq(N + 1, T)
X_values_7 = np.abs(dft(x_values_7))

plt.figure(figsize=(10, 6))
plt.stem(f_values, X_values, label="Для исходного сигнала")
plt.stem(f_values, X_values_5, linefmt="C1--", markerfmt="C1o", label="Для
отфильтрованного сигнала")
plt.ylabel("X")
plt.xlabel("f")
plt.legend()
plt.grid()
plt.show()

f_values = fftfreq(N + 1, T)
X_values_11 = np.abs(dft(x_values_11))

plt.figure(figsize=(10, 6))
plt.stem(f_values, X_values, label="Для исходного сигнала")
plt.stem(f_values, X_values_9, linefmt="C1--", markerfmt="C1o", label="Для
отфильтрованного сигнала")
plt.ylabel("X")
plt.xlabel("f")
plt.legend()
plt.grid()
plt.show()

```