

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

ОТЧЕТ
по лабораторной работе №2
по дисциплине «Цифровая обработка сигналов»
Тема: Частотный анализ полиномиальных приближений

Студентка гр. 8382

Звегинцева Е.Н.

Студент гр. 8382

Мирончик П.Д.

Преподаватель

Середа В.И.

Санкт-Петербург

2021

Цель работы.

Анализ частотных характеристик известных формул полиномиального сглаживания временных рядов и формул численного интегрирования.

Основные теоретические положения.

В качестве временного ряда рассматривается дискретный сигнал с шагом дискретизации, равным единице.

Под полиномиальным сглаживанием понимается аппроксимация в смысле МНК значений конечного (нечетного) числа элементов сглаживаемого ряда полиномом заданного порядка с присвоением среднему из этих элементов значения сглаживающего полинома в центре выбранного временного отрезка. Такой подход соответствует так называемому сглаживанию в скользящем окне.

В качестве исследуемых формул численного интегрирования используются квадратурные формулы Ньютона-Котеса.

Постановка задачи.

Получить формулы для передаточных функций нерекурсивных фильтров, соответствующих полиномиальному сглаживанию дискретного сигнала для полиномов различного порядка и построить графики $\tilde{H}(f)$. Проинтерпретировать частотные свойства передаточных функций. Провести сопоставительный анализ частотных характеристик передаточных функций для различных степеней полиномов. Получить формулы для передаточных функций рекурсивных фильтров, соответствующих квадратурным формулам Ньютона-Котеса различного порядка. Проинтерпретировать частотные свойства передаточных функций. Провести сопоставительный анализ частотных характеристик передаточных функций для различных квадратурных формул.

Порядок выполнения работы.

1. Вывести формулы для передаточной функции нерекурсивного фильтра, соответствующего сглаживанию прямой линией по 3, 5, 7 и 9 точкам. Построить графики $\tilde{H}(f)$. Проинтерпретировать частотные свойства передаточных функций для различного количества точек.
2. Вывести формулы для передаточной функции нерекурсивного фильтра, соответствующего сглаживанию полиномом второй степени по 7, 9, 11 и 13 точкам. Построить графики $\tilde{H}(f)$. Проинтерпретировать частотные свойства передаточных функций для различного количества точек.
3. Вывести формулы для передаточной функции нерекурсивного фильтра, соответствующего сглаживанию полиномом четвёртой степени по 9, 11, 13 и 15 точкам. Построить графики $\tilde{H}(f)$. Проинтерпретировать частотные свойства передаточных функций для различного количества точек.
4. Вывести формулы для передаточной функции нерекурсивного фильтра, соответствующего сглаживанию по формулам Спенсера по 13, 15 и 21 точкам. Построить графики $\tilde{H}(f)$. Проинтерпретировать частотные свойства передаточных функций для различного количества точек.
5. Построить графики из предыдущих пунктов в логарифмической шкале (Дб). Объясните, чем отличаются данные графики от полученных ранее и объясните их смысл.
6. Провести сопоставительный анализ свойств передаточных функций, полученных при выполнении пп. 1-4.
7. Вывести формулы передаточных функций рекурсивных фильтров, соответствующих квадратурным формулам прямоугольников, трапеций и Симпсона. Построить графики передаточных функций и графики отношения вычисляемого в результате фильтрации значения к истинному. Проинтерпретировать частотные свойства полученных передаточных функций.

8. Вывести формулу передаточной функции рекурсивного фильтра, соответствующего квадратурной формуле:

$$y_{n+2} = y_{n-1} + \frac{1}{8}(x_{n+2} + 3x_{n+1} + 3x_n + x_{n-1}).$$

Построить график передаточной функции и график отношения вычисляемого в результате фильтрации значения к истинному. Проинтерпретировать частотные свойства передаточной функции.

9. Провести сопоставительный анализ частотных характеристик передаточных функций, полученных при выполнении пп. 7 и 8.

Ход работы.

В ходе работы были выполнены следующие действия:

1) Входной сигнал: $s(t)$.

Выходной сигнал: $y(t) = A + Bt$.

Если осуществлять приближение по МНК прямой линией по m точкам, то необходимо минимизировать следующее выражение для нечетных чисел:

$$F(A, B) = \sum_{k=-m}^m (s_k - y_k)^2 = \sum_{k=-m}^m (s_k - A - Bk)^2 \Rightarrow \min$$

Продифференцируем и приравняем к нулю, чтобы найти минимум.

Посчитаем частные производные по A , получим:

$$\begin{aligned} \frac{\delta F(A, B)}{\delta A} = 0 &\Leftrightarrow -2 \sum_{k=-m}^m (s_k - A - Bk) = 0 \Leftrightarrow \\ &\Leftrightarrow - \sum_{k=-m}^m s_k + \sum_{k=-m}^m A + \sum_{k=-m}^m Bk = 0 \end{aligned}$$

Тогда:

$$A = \frac{1}{2m+1} \sum_{k=-m}^m s_k$$

В итоге получаем:

$$y_0 = A = \frac{1}{2m+1} \sum_{k=-m}^m s_k = \frac{1}{2m+1} (s_{-m} + s_{-m+1} + \dots + s_{m-1} + s_m)$$

В общем случае:

$$\begin{aligned} y_n &= \frac{1}{2m+1} \sum_{k=-m}^m s_k = \\ &= \frac{1}{2m+1} (s_{-m+n} + s_{-m+1+n} + \dots + s_{m-1+n} + s_{m+n}) \end{aligned}$$

$$s_n = e^{i\omega n}$$

$$y_n = \frac{1}{2m+1} (e^{-mi\omega} + e^{(-m+1)i\omega} + \dots + e^{(m-1)i\omega} + e^{mi\omega}) = H(\omega)$$

$$H(\omega) = \frac{1}{2m+1} (1 + 2 \cos(\omega) + 2 \cos(2\omega) + \dots + 2 \cos(m\omega))$$

$$H(\omega) = \frac{\sin(\frac{(2m+1)\omega}{2})}{(2m+1)\sin(\frac{\omega}{2})}$$

$$H(\omega) = H(2\pi f) = \tilde{H}(f)$$

Формула для передаточной функции нерекурсивного фильтра, соответствующего сглаживанию прямой линией по 3 точкам:

$$H_3(f) = \frac{\sin(3\pi f)}{3 \sin(\pi f)}$$

Формула для передаточной функции нерекурсивного фильтра, соответствующего сглаживанию прямой линией по 5 точкам:

$$H_5(f) = \frac{\sin(5\pi f)}{5 \sin(\pi f)}$$

Формула для передаточной функции нерекурсивного фильтра, соответствующего сглаживанию прямой линией по 7 точкам:

$$H_7(f) = \frac{\sin(7\pi f)}{7 \sin(\pi f)}$$

Формула для передаточной функции нерекурсивного фильтра, соответствующего сглаживанию прямой линией по 9 точкам:

$$H_9(f) = \frac{\sin(9\pi f)}{9 \sin(\pi f)}$$

Графики для передаточных функций на интервале $f \in [0; 1]$ представлены рис. 1.

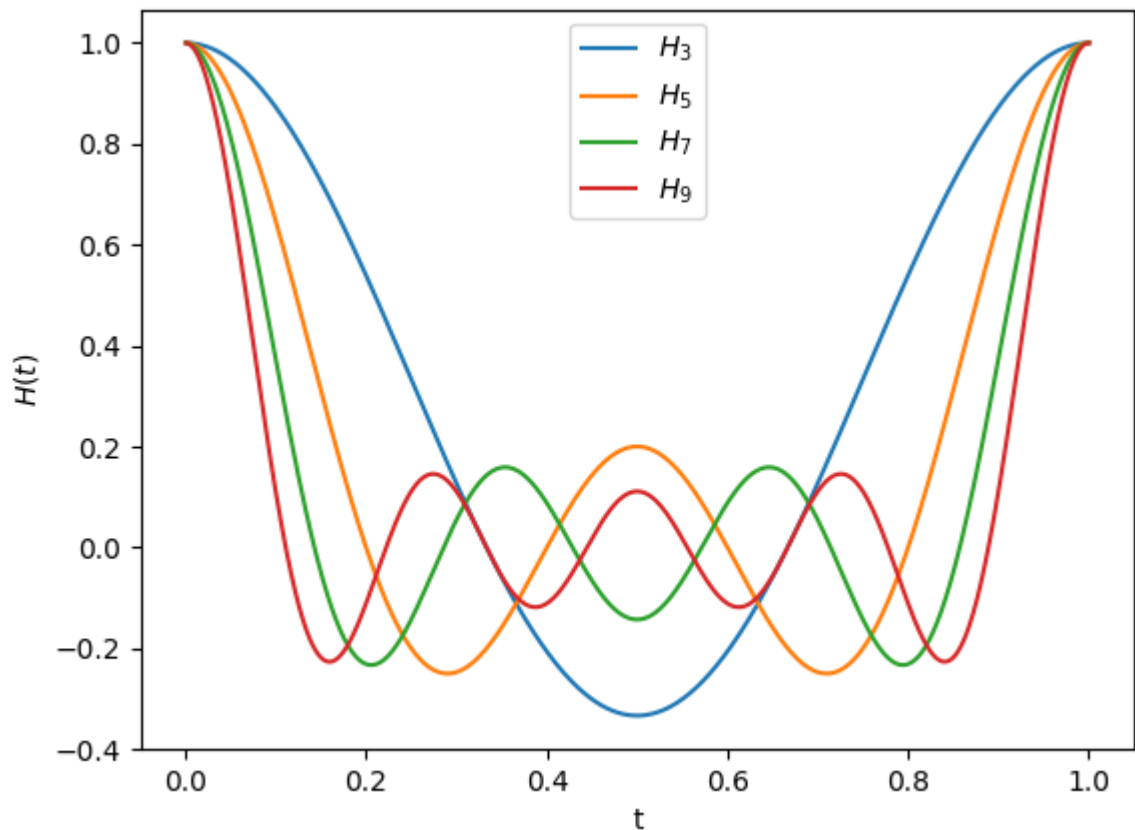


Рисунок 1 – Сглаживание прямой по методу наименьших квадратов.

Исходя из графика, можно сделать наблюдение, что сколько было использовано точек для построения передаточной функции, столько она имеет экстремумов (с увеличением количества увеличиваются колебания и порядок гладкости).

2) Входной сигнал: $s(t)$.

Выходной сигнал: $y(t) = A + Bt + Ct^2$.

Если осуществлять приближение по МНК по m точкам, то необходимо минимизировать следующее выражение для нечетных чисел:

$$F(A, B, C) = \sum_{k=-m}^m (s_k - y_k)^2 = \sum_{k=-m}^m (s_k - A - Bk - Ck^2)^2 \Rightarrow \min$$

Переменными в этой задаче являются коэффициенты A и C . Продифференцируем и приравняем к нулю, чтобы найти минимум. Посчитаем частные производные по A и C , получим:

$$\begin{aligned}
\begin{cases} \frac{\delta F(A, B, C)}{\delta A} = 0 \\ \frac{\delta F(A, B, C)}{\delta C} = 0 \end{cases} &\Leftrightarrow \begin{cases} -2 \sum_{k=-m}^m (s_k - A - Bk - Ck^2) = 0 \\ -2 \sum_{k=-m}^m (k^2 s_k - k^2 A - Bk^3 - Ck^4) = 0 \end{cases} \Leftrightarrow \\
&\Leftrightarrow \begin{cases} - \sum_{k=-m}^m s_k + \sum_{k=-m}^m A + \sum_{k=-m}^m Bk + \sum_{k=-m}^m Ck^2 = 0 \\ - \sum_{k=-m}^m k^2 s_k + \sum_{k=-m}^m k^2 A + \sum_{k=-m}^m Bk^3 + \sum_{k=-m}^m Ck^4 = 0 \end{cases}
\end{aligned}$$

Тогда система нормальных уравнений:

$$\begin{cases} (2m+1)A + \frac{m(m+1)(2m+1)}{3}C = \sum_{k=-m}^m s_k \\ \frac{m(m+1)(2m+1)}{3}A + \frac{m(m+1)(2m+1)(3m^2+3m-1)}{15}C = \sum_{k=-m}^m k^2 s_k \end{cases}$$

Подставим $\frac{m(m+1)(2m+1)}{3}C$ во второе уравнение:

$$\begin{aligned}
&\frac{m(m+1)(2m+1)}{3}A + \frac{3m^2+3m-1}{5} \left(\sum_{k=-m}^m s_k - (2m+1)A \right) \\
&= \sum_{k=-m}^m k^2 s_k \Leftrightarrow \\
&\left(\frac{m(m+1)(2m+1)}{3} - \frac{(3m^2+3m-1)(2m+1)}{5} \right) A \\
&= \sum_{k=-m}^m k^2 s_k - \frac{3m^2+3m-1}{5} \sum_{k=-m}^m s_k
\end{aligned}$$

Тогда:

$$A = \frac{\sum_{k=-m}^m k^2 s_k - \frac{3m^2+3m-1}{5} \sum_{k=-m}^m s_k}{\frac{m(m+1)(2m+1)}{3} - \frac{(3m^2+3m-1)(2m+1)}{5}}$$

В итоге получаем:

$$y_0 = A = \frac{\sum_{k=-m}^m k^2 s_k - \frac{3m^2 + 3m - 1}{5} \sum_{k=-m}^m s_k}{\frac{m(m+1)(2m+1)}{3} - \frac{(3m^2 + 3m - 1)(2m+1)}{5}}$$

Для 7 точек:

$$\begin{aligned} y_7 = A &= \frac{\sum_{k=-3}^3 k^2 s_k - 7 \sum_{k=-3}^3 s_k}{28 - 49} = \frac{1}{21} \left(7 \sum_{k=-3}^3 s_k - \sum_{k=-3}^3 k^2 s_k \right) \\ &= \frac{1}{21} (7s_{-3} + 7s_{-2} + 7s_{-1} + 7s_0 + 7s_1 + 7s_2 + 7s_3 - 9s_{-3} - 4s_{-2} \\ &\quad - s_1 - 4s_2 - 9s_3) \\ &= \frac{1}{21} (-2s_{-3} + 3s_{-2} + s_{-1} + 7s_0 + 6s_1 + 3s_2 - 2s_3) \\ y_7 &= \frac{1}{21} (-2s_{-3} + 3s_{-2} + s_{-1} + 7s_0 + 6s_1 + 3s_2 - 2s_3) \end{aligned}$$

В общем случае:

$$y_n = \frac{1}{21} (-2s_{n-3} + 3s_{n-2} + s_{n-1} + 7s_n + 6s_{n+1} + 3s_{n+2} - 2s_{n+3})$$

$$s_n = e^{i\omega n}$$

$$\begin{aligned} y_n &= \frac{1}{21} (-2e^{-3i\omega} + 3e^{-2i\omega} + 6e^{-i\omega} + 7 + 6e^{i\omega} + 3e^{2i\omega} - 2e^{3i\omega}) e^{i\omega n} \\ &= H(\omega) e^{i\omega n} \end{aligned}$$

$$H(\omega) = \frac{1}{21} (-2e^{-3i\omega} + 3e^{-2i\omega} + 6e^{-i\omega} + 7 + 6e^{i\omega} + 3e^{2i\omega} - 2e^{3i\omega})$$

$$H(\omega) = \frac{1}{21} (7 + 12 \cos \omega + 6 \cos 2\omega - 4 \cos 3\omega)$$

$$H(\omega) = H(2\pi f) = \tilde{H}(f)$$

Формула для передаточной функции нерекурсивного фильтра, соответствующего сглаживанию полиномом второй степени по 7 точкам:

$$H_7(f) = \frac{1}{21} (7 + 12 \cos(2\pi f) + 6 \cos(4\pi f) - 4 \cos(6\pi f))$$

Аналогично посчитаем для 9, 11, 13 точек.

Формула для передаточной функции нерекурсивного фильтра, соответствующего сглаживанию полиномом второй степени по 9 точкам:

$$H_9(f) = \frac{1}{231} (59 + 108 \cos(2\pi f) + 78 \cos(4\pi f) + 28 \cos(6\pi f) - 42 \cos(8\pi f))$$

Формула для передаточной функции нерекурсивного фильтра, соответствующего сглаживанию полиномом второй степени по 11 точкам:

$$H_{11}(f) = \frac{1}{429} (89 + 168 \cos(2\pi f) + 138 \cos(4\pi f) + 88 \cos(6\pi f) + 18 \cos(8\pi f) - 72 \cos(10\pi f))$$

Формула для передаточной функции нерекурсивного фильтра, соответствующего сглаживанию полиномом второй степени по 13 точкам:

$$H_{13}(f) = \frac{1}{143} (25 + 48 \cos(2\pi f) + 42 \cos(4\pi f) + 32 \cos(6\pi f) + 18 \cos(8\pi f) - 22 \cos(10\pi f))$$

Графики для передаточных функций на интервале $f \in [0; 1]$ представлены рис. 2.

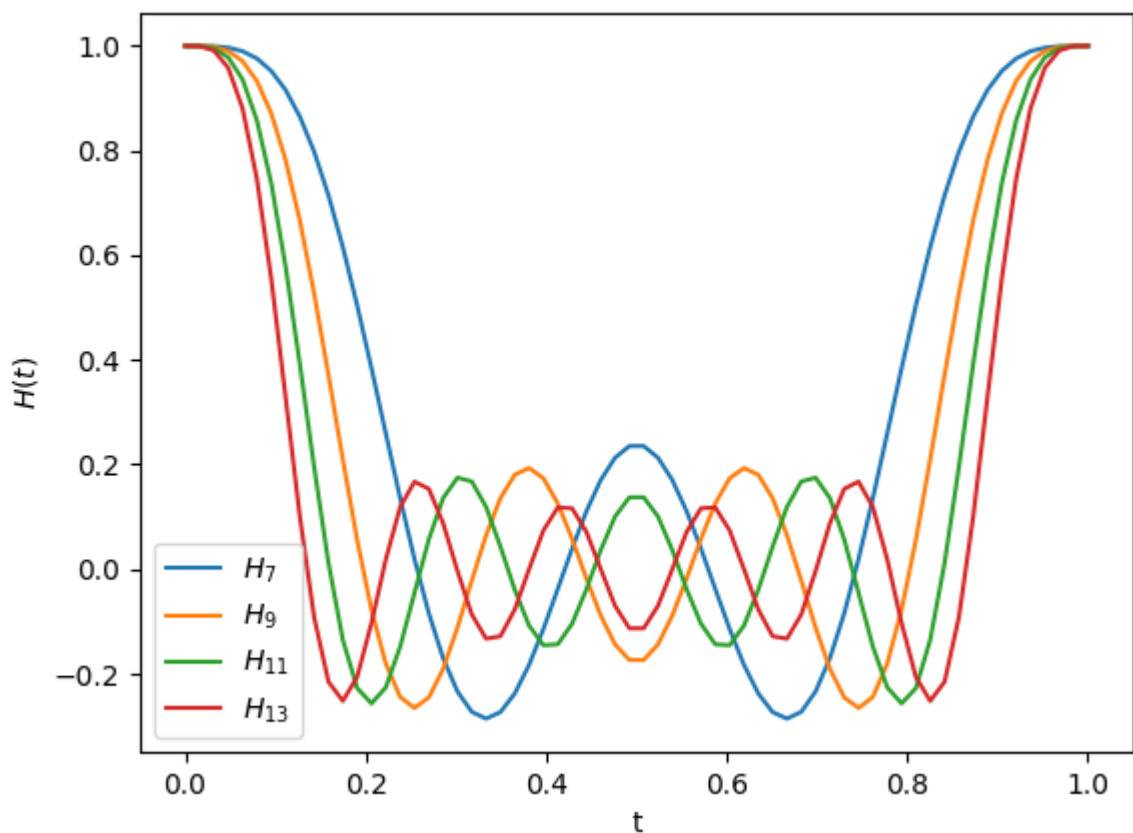


Рисунок 2 – Графики передаточной функции при сглаживании полиномом второй степени по 7, 9, 11 и 13 точкам

3) Входной сигнал: $s(t)$.

Выходной сигнал: $y(t) = A + Bt + Ct^2 + Dt^3 + Et^4$.

Если осуществлять приближение по МНК по m точкам, то необходимо минимизировать следующее выражение для нечетных чисел:

$$F(A, B, C) = \sum_{k=-m}^m (s_k - y_k)^2 = \sum_{k=-m}^m (s_k - A - Bk - Ck^2 + Dk^3 + Ek^4)^2$$

$$\Rightarrow \min$$

Посчитаем частные производные по A , C и E , получим:

$$\begin{cases} \frac{\delta F(A, B, C, D, E)}{\delta A} = 0 \\ \frac{\delta F(A, B, C, D, E)}{\delta C} = 0 \\ \frac{\delta F(A, B, C, D, E)}{\delta E} = 0 \end{cases} \Leftrightarrow \begin{cases} -2 \sum_{k=-m}^m (s_k - A - Bk - Ck^2 - Dk^3 - Ek^4) = 0 \\ -2 \sum_{k=-m}^m (k^2 s_k - k^2 A - Bk^3 - Ck^4 - Dk^5 - Ek^6) = 0 \\ -2 \sum_{k=-m}^m (k^4 s_k - k^4 A - Bk^5 - Ck^6 - Dk^7 - Ek^8) = 0 \end{cases}$$

$$\Leftrightarrow \begin{cases} - \sum_{k=-m}^m s_k + \sum_{k=-m}^m A + \sum_{k=-m}^m Bk + \sum_{k=-m}^m Ck^2 + \sum_{k=-m}^m Dk^3 + \sum_{k=-m}^m Ek^4 = 0 \\ - \sum_{k=-m}^m k^2 s_k + \sum_{k=-m}^m k^2 A + \sum_{k=-m}^m Bk^3 + \sum_{k=-m}^m Ck^4 + \sum_{k=-m}^m Dk^5 + \sum_{k=-m}^m Ek^6 = 0 \\ - \sum_{k=-m}^m k^4 s_k + \sum_{k=-m}^m k^4 A + \sum_{k=-m}^m Bk^5 + \sum_{k=-m}^m Ck^6 + \sum_{k=-m}^m Dk^7 + \sum_{k=-m}^m Ek^8 = 0 \end{cases}$$

$$\begin{cases} - \sum_{k=-m}^m s_k + \sum_{k=-m}^m A + \sum_{k=-m}^m Ck^2 + \sum_{k=-m}^m Ek^4 = 0 \\ - \sum_{k=-m}^m k^2 s_k + \sum_{k=-m}^m k^2 A + \sum_{k=-m}^m Ck^4 + \sum_{k=-m}^m Ek^6 = 0 \\ - \sum_{k=-m}^m k^4 s_k + \sum_{k=-m}^m k^4 A + \sum_{k=-m}^m Ck^6 + \sum_{k=-m}^m Ek^8 = 0 \end{cases}$$

Система нормальных уравнений:

$$\left\{ \begin{aligned} & (2m+1)A + \frac{m(m+1)(2m+1)}{3}C + \frac{m(m+1)(2m+1)(3m^2+3m-1)}{15}F = \\ & \quad = \sum_{k=-m}^m s_k \\ & \frac{m(m+1)(2m+1)}{3}A + \frac{m(m+1)(2m+1)(3m^2+3m-1)}{15}C + \\ & \quad + \frac{m(m+1)(2m+1)(1-3m+6m^3+3m^4)}{21}F = \sum_{k=-m}^m k^2 s_k \\ & \frac{m(m+1)(2m+1)(3m^2+3m-1)}{15}A + \frac{m(m+1)(2m+1)(1-3m+6m^3+3m^4)}{21}C + \\ & \quad + \frac{m(m+1)(2m+1)(-3+9m-m^2-15m^3+5m^4+15m^5+5m^6)}{45} = \sum_{k=-m}^m k^4 s_k \end{aligned} \right.$$

Выразим C из первого уравнения системы нормальных уравнений:

$$C = \frac{-15A - 30Am + Em - 10Em^3 - 15Em^4 - 6Em^5 - 15 \sum_{k=-m}^m s_k}{5m(m+1)(2m+1)}$$

Подставим теперь C во второе и третье уравнение:

$$\left\{ \begin{aligned} & \frac{(-3-2m+12m^2+8m^3)(-35A+3Em(-2-m+2m^2+m^3))}{525} + \\ & \quad + \frac{3m^2+3m-1}{5} \sum_{k=-m}^m s_k = \sum_{k=-m}^m k^2 s_k \\ & \quad - \frac{(-3-2m+12m^2+8m^3)}{315} * \\ & \quad * (3A(6m^2+6m-5) + Em(-6+m+12m^2+m^3-6m^4-2m^5)) + \\ & \quad + \frac{1-3m+6m^3+3m^4}{7} \sum_{k=-m}^m s_k = \sum_{k=-m}^m k^4 s_k \end{aligned} \right.$$

Выразим из 2 уравнения E и подставим в 3 уравнение:

$$\left\{ \begin{aligned} & E = 35 \cdot \frac{A(-3-2m+12m^2+8m^3) + (3-9m-9m^2) \sum_{k=-m}^m s_k + 15 \sum_{k=-m}^m k^2 s_k}{3m(-2-m+2m^2+m^3)(-3-2m+12m^2+8m^3)} \\ & \quad - \frac{4A(45+18m-200m^2-80m^3+80m^4+32m^5)}{3m(-2-m+2m^2+m^3)(-3-2m+12m^2+8m^3)} - \\ & \quad - 15 \cdot (12-50m-35m^2+30m^3+15m^4) \sum_{k=-m}^m s_k + \\ & \quad + 525 \cdot (-3+2m+2m^2) \sum_{k=-m}^m k^2 s_k = 945 \sum_{k=-m}^m k^4 s_k \end{aligned} \right.$$

И выразим из 3 уравнения A :

$$A = 15 \left(\frac{(12 + 5m(1 + m)(-10 + 3m(1 + m))) \sum_{k=-m}^m s_k}{4(-3 + 2m)(-1 + 2m)(1 + 2m)(3 + 2m)(5 + 2m)} - \frac{35(-3 + 2m(1 + m)) \sum_{k=-m}^m k^2 s_k - 63 \sum_{k=-m}^m k^4 s_k}{4(-3 + 2m)(-1 + 2m)(1 + 2m)(3 + 2m)(5 + 2m)} \right)$$

В итоге получаем:

$$y_0 = A = 15 \left(\frac{(12 + 5m(1 + m)(-10 + 3m(1 + m))) \sum_{k=-m}^m s_k}{4(-3 + 2m)(-1 + 2m)(1 + 2m)(3 + 2m)(5 + 2m)} - \frac{35(-3 + 2m(1 + m)) \sum_{k=-m}^m k^2 s_k - 63 \sum_{k=-m}^m k^4 s_k}{4(-3 + 2m)(-1 + 2m)(1 + 2m)(3 + 2m)(5 + 2m)} \right)$$

Для 9 точек:

$$y_9 = \frac{1}{429} \left(179 \sum_{k=-4}^4 s_k - \frac{1}{4} \left(185 \sum_{k=-4}^4 k^2 s_k - 9 \sum_{k=-4}^4 k^4 s_k \right) \right)$$

$$y_9 = \frac{1}{429} (15s_{-4} - 55s_{-3} + 30s_{-2} + 135s_{-1} + 179s_0 + 135s_1 + 30s_2 - 55s_3 + 15s_4)$$

В общем случае:

$$y_n = \frac{1}{429} (15s_{n-4} - 55s_{n-3} + 30s_{n-2} + 135s_{n-1} + 179s_n + 135s_{n+1} + 30s_{n+2} - 55s_{n+3} + 15s_{n+4})$$

$$s_n = e^{i\omega n}$$

$$y_n = \frac{1}{429} (15e^{-4i\omega} - 55e^{-3i\omega} + 30e^{-2i\omega} + 135e^{-i\omega} + 179 + 135e^{i\omega} + 30e^{2i\omega} - 55e^{3i\omega} + 15e^{4i\omega}) e^{i\omega n} = H(\omega) e^{i\omega n}$$

$$H(\omega) = \frac{1}{429} (15e^{-4i\omega} - 55e^{-3i\omega} + 30e^{-2i\omega} + 135e^{-i\omega} + 179 + 135e^{i\omega} + 30e^{2i\omega} - 55e^{3i\omega} + 15e^{4i\omega})$$

$$H(\omega) = \frac{1}{429} (179 + 270 \cos \omega + 60 \cos 2\omega - 110 \cos 3\omega + 30 \cos 4\omega)$$

$$H(\omega) = H(2\pi f) = \tilde{H}(f)$$

Формула для передаточной функции нерекурсивного фильтра, соответствующего сглаживанию полиномом четвертой степени по 9 точкам:

$$H_9(f) = \frac{1}{429} (179 + 270 \cos(2\pi f) + 60 \cos(4\pi f) - 110 \cos(6\pi f) + 30 \cos(8\pi f))$$

Аналогично посчитаем для 11, 13, 15 точек.

Формула для передаточной функции нерекурсивного фильтра, соответствующего сглаживанию полиномом четвертой степени по 11 точкам:

$$H_{11}(f) = \frac{1}{429} (143 + 240 \cos(2\pi f) + 120 \cos(4\pi f) - 20 \cos(6\pi f) - 90 \cos(8\pi f) + 36 \cos(10\pi f))$$

Формула для передаточной функции нерекурсивного фильтра, соответствующего сглаживанию полиномом четвертой степени по 13 точкам:

$$H_{13}(f) = \frac{1}{2431} (677 + 1200 \cos(2\pi f) + 780 \cos(4\pi f) + 220 \cos(6\pi f) - 270 \cos(8\pi f) - 396 \cos(10\pi f) + 220 \cos(12\pi f))$$

Формула для передаточной функции нерекурсивного фильтра, соответствующего сглаживанию полиномом четвертой степени по 15 точкам:

$$H_{15}(f) = \frac{1}{46189} (11063 + 20250 \cos(2\pi f) + 15000 \cos(4\pi f) + 7510 \cos(6\pi f) - 330 \cos(8\pi f) - 5874 \cos(10\pi f) + 5720 \cos(12\pi f) + 4290 \cos(14\pi f))$$

Графики для передаточных функций на интервале $f \in [0; 1]$ представлены рис. 3.

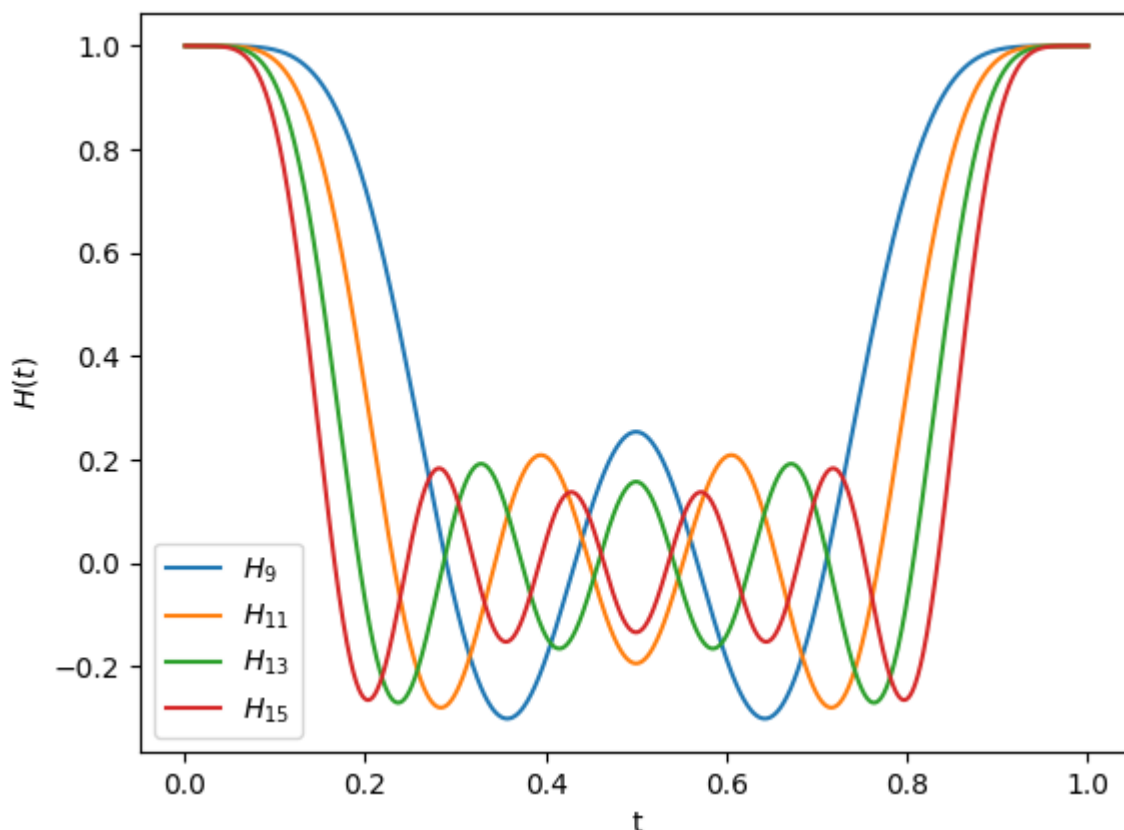


Рисунок 3 – Графики передаточной функции при сглаживании полиномом четвертой степени по 9, 11, 13 и 15 точкам

4) Графики для передаточных функций на интервале $f \in [0; 1]$ представлены рис. 4.

$$y_n = \frac{1}{320} (-3s_{n-7} - 6s_{n-6} - 5s_{n-5} + 3s_{n-4} + 21s_{n-3} + 46s_{n-2} - 67s_{n-1} + 74s_n + 67s_{n+1} + 46s_{n+2} + 21s_{n+3} + 3s_{n+4} - 5s_{n+5} - 6s_{n+6} - 3s_{n+7})$$

$$y_n = \frac{1}{350} (-s_{n-10} - 3s_{n-6} - 5s_{n-8} - 5s_{n-7} - 2s_{n-6} + 6s_{n-5} + 18s_{n-4} + 33s_{n-3} + 47s_{n-2} + 57s_{n-1} + 60s_n + 57s_{n+1} + 47s_{n+2} + 33s_{n+3} + 18s_{n+4} + 6s_{n+5} - 2s_{n+6} - 5s_{n+7} - 5s_{n+8} - 3s_{n+9} - s_{n+10})$$

Соответствующие передаточные функции:

$$H_{15}(f) = \frac{1}{320} (74 + 134 \cos(2\pi f) + 92 \cos(4\pi f) + 42 \cos(6\pi f) + 6 \cos(8\pi f) - 10 \cos(10\pi f) - 12 \cos(12\pi f) - 6 \cos(14\pi f))$$

$$H_{21}(f) = \frac{1}{350} (60 + 114 \cos(2\pi f) + 94 \cos(4\pi f) + 66 \cos(6\pi f) + 36 \cos(8\pi f) + 12 \cos(10\pi f) - 4 \cos(12\pi f) - 10 \cos(14\pi f) - 10 \cos(16\pi f) - 6 \cos(18\pi f) - 2 \cos(20\pi f))$$

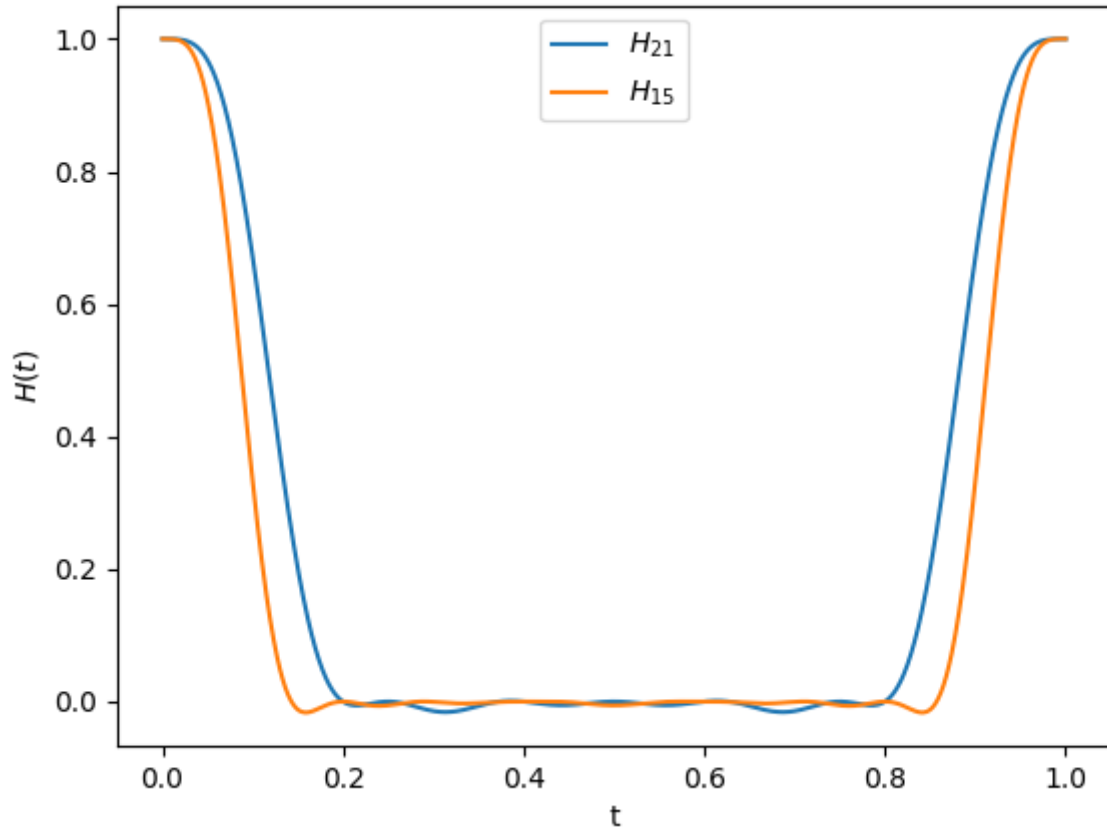


Рисунок 4 – Графики передаточной функции при сглаживании по формулам Спенсера по 15, 21 точкам

5) Кривые, построенные ранее, недостаточно информативны, так как значения на высоких частотах настолько малы, что невозможно решить насколько они хороши. Поэтому лучше использовать логарифмы чисел $H(w)$. Для этой цели используются децибелы:

$$\text{Значение в Дб} = 20 * \lg \left(\frac{|y_n|}{|s_n|} \right)$$

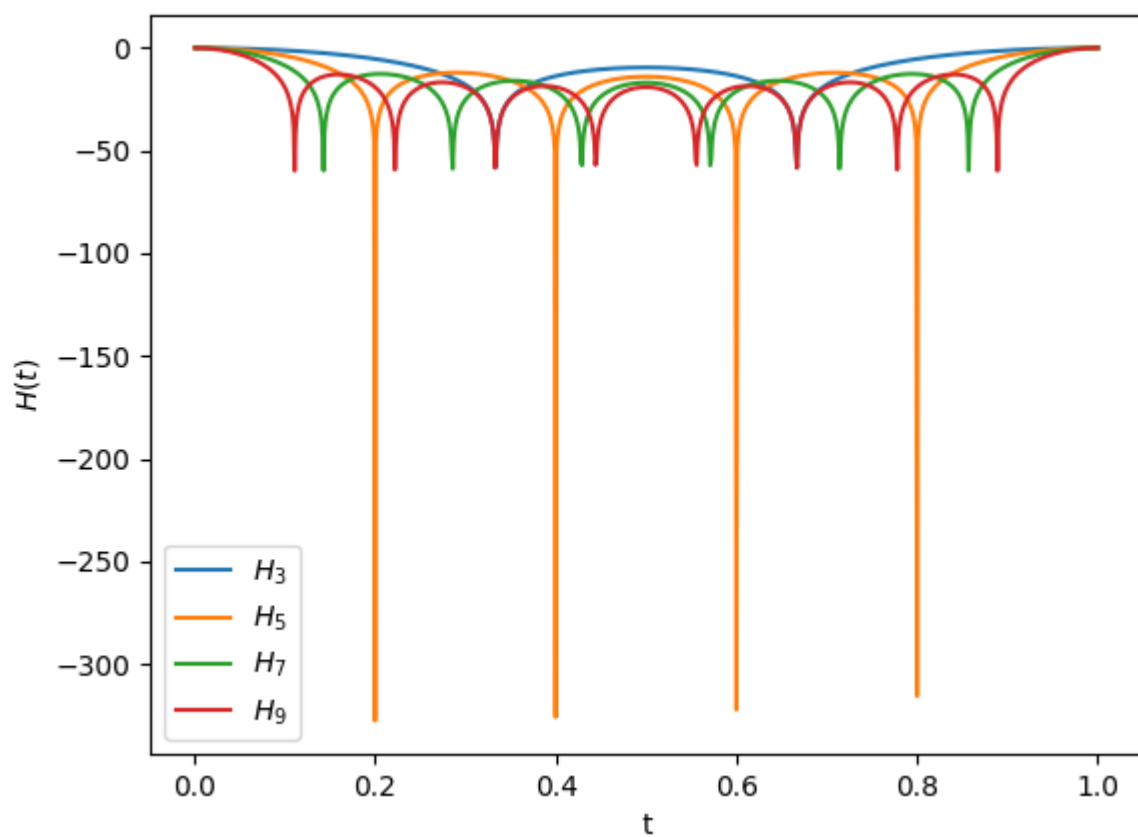


Рисунок 5 – Сглаживание прямой по методу наименьших квадратов в логарифмической шкале.

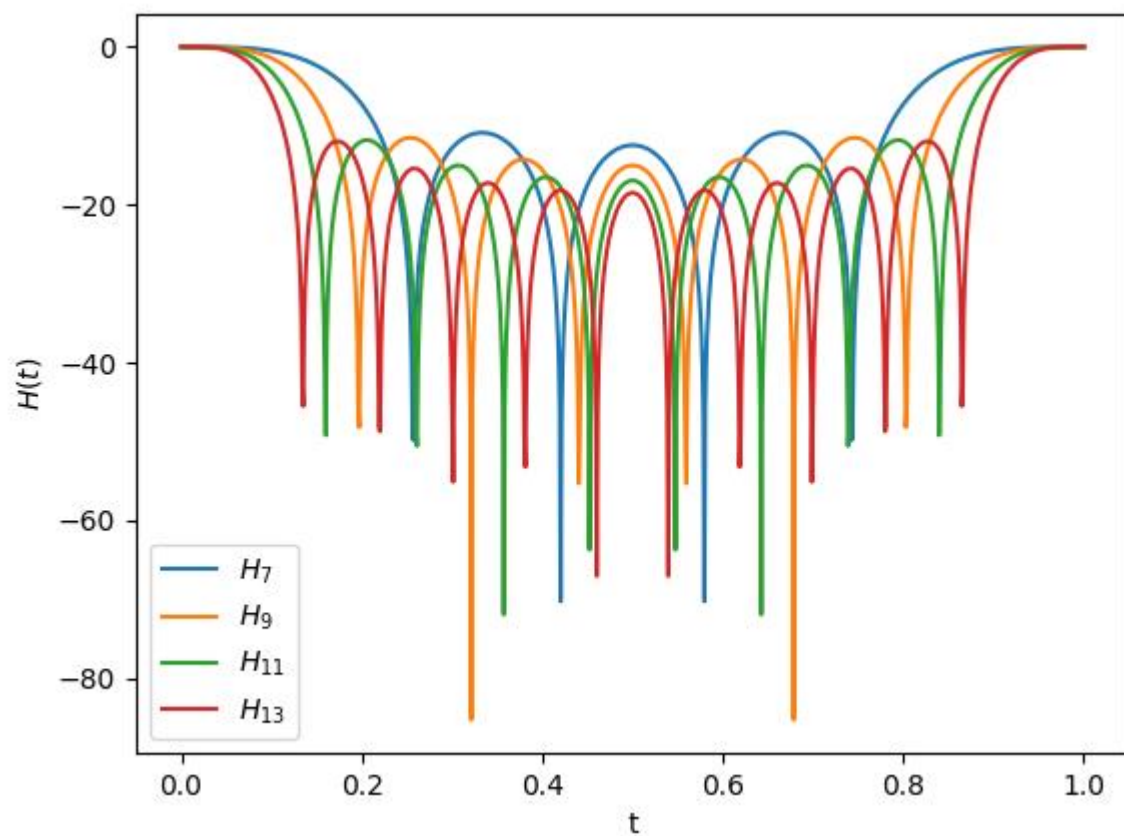


Рисунок 6 – Сглаживание полиномом второй степени по методу наименьших квадратов в логарифмической шкале.

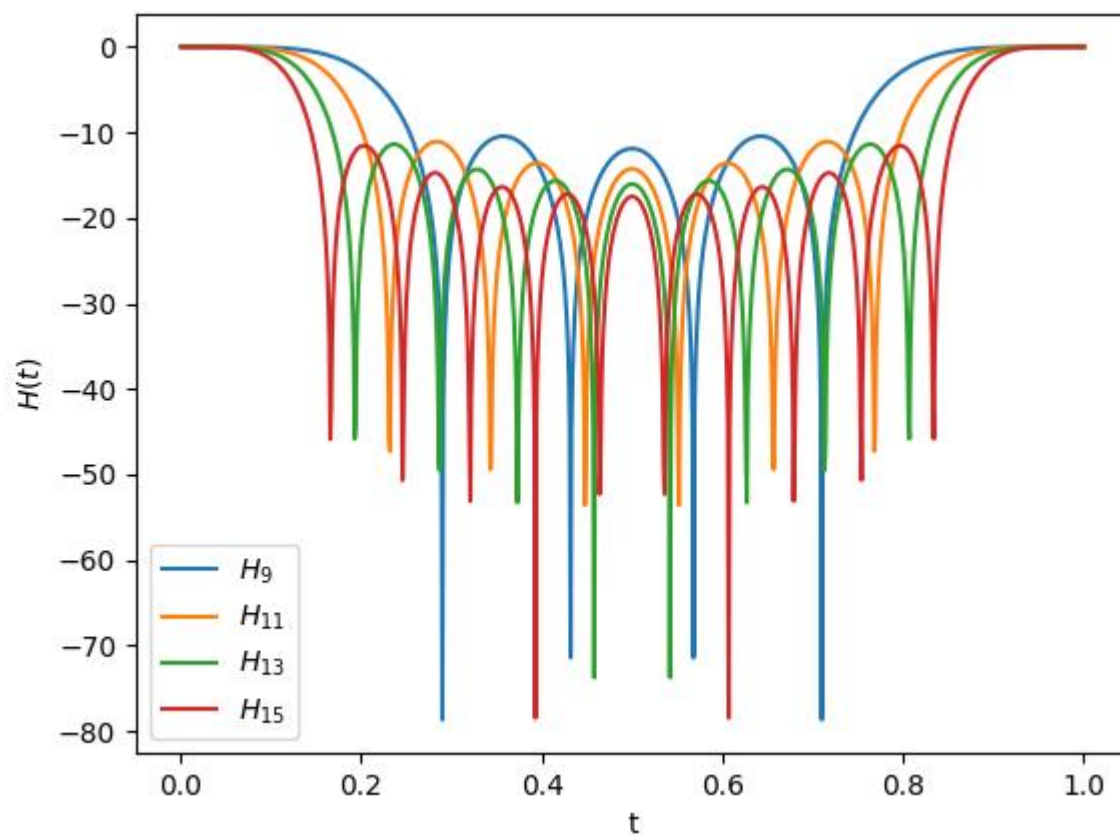


Рисунок 7 – Сглаживание полиномом четвертой степени по методу наименьших квадратов в логарифмической шкале.

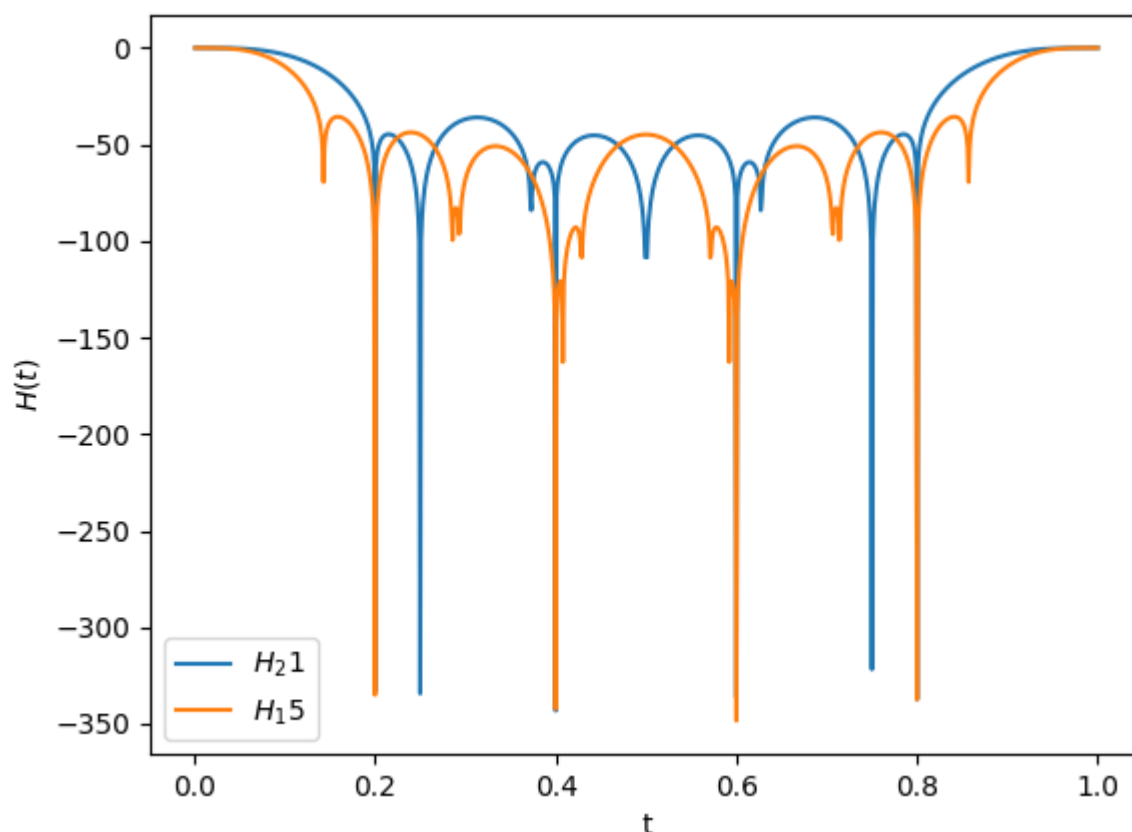


Рисунок 8 – Сглаживание по формулам Спенсера в логарифмической шкале.

Графики на рис. 5-8 позволяют сделать заключение о том, что шум является высокочастотным, а сигнал низкочастотным. Теперь ясно, что формулы сглаживания пропускают одни частоты, а другие устраняют.

6) Проведем сопоставительный анализ свойств передаточных функций, полученных при выполнении пп. 1-4.

Были получены графики передаточных функций нерекурсивных фильтров, соответствующих сглаживанию прямой линией, полиномом второй степени, полиномом четвёртой степени и формулам Спенсера (рис. 1-4). Графики соответствуют ожиданиям.

7) Выведем формулы передаточных функций рекурсивных фильтров, соответствующих квадратурным формулам прямоугольников, трапеций и Симпсона. Построим графики передаточных функций и графики отношения вычисляемого в результате фильтрации значения к истинному. Проинтерпретируем частотные свойства полученных передаточных функций.

Формула прямоугольников:

$$y_{n+1} = y_n + s_{n+\frac{1}{2}}, y_0 = 0$$

Пусть $s_n = e^{i\omega n}$ и $y_n = H(\omega)e^{i\omega n}$, тогда:

$$\begin{cases} y_{n+1} = H(\omega)e^{i\omega n} + e^{i\omega(n+\frac{1}{2})} \\ y_{n+1} = H(\omega)e^{i\omega(n+1)} \end{cases}$$

$$H(\omega)(e^{i\omega n}e^{i\omega}) = H(\omega)e^{i\omega n} + e^{i\omega n}e^{\frac{1}{2}i\omega}$$

$$H(\omega)(e^{i\omega n}e^{i\omega} - e^{i\omega n}) = e^{i\omega n}e^{\frac{1}{2}i\omega}$$

$$H(\omega)(e^{i\omega} - 1) = e^{\frac{1}{2}i\omega}$$

$$H(\omega) = \frac{1}{e^{\frac{i\omega}{2}} - e^{-\frac{i\omega}{2}}} = \frac{1}{2i\sin\frac{\omega}{2}}$$

$$\tilde{H}(f) = \frac{1}{2i\sin(\pi f)}$$

Точное значение интеграла $e^{i\omega t}$ равно $\frac{e^{i\omega t}}{i\omega}$, тогда отношение значений:

$$\gamma = \frac{\text{Вычисленное}}{\text{Точное}} = \frac{i\omega}{2i\sin\frac{\omega}{2}} = \frac{\frac{\omega}{2}}{\sin\frac{\omega}{2}} = 1 + \frac{x^2}{24} + \frac{7x^4}{5760} + \dots$$

$$\gamma = \frac{\pi f}{\sin(\pi f)} = 1 + \frac{\pi^2 f^2}{6} + \frac{7\pi^4 f^4}{360} + \dots$$

Формула трапеций:

$$y_{n+1} = y_n + \frac{1}{2}(s_n + s_{n+1}), y_0 = 0$$

Пусть $s_n = e^{i\omega n}$ и $y_n = H(\omega)e^{i\omega n}$, тогда:

$$\begin{cases} y_{n+1} = H(\omega)e^{i\omega n} + \frac{e^{i\omega n} + e^{i\omega(n+1)}}{2} \\ y_{n+1} = H(\omega)e^{i\omega(n+1)} \end{cases}$$

$$H(\omega)(e^{i\omega n}e^{i\omega}) = H(\omega)e^{i\omega n} + e^{i\omega n}\frac{1 + e^{i\omega}}{2}$$

$$H(\omega)(e^{i\omega} - 1) = \frac{1 + e^{i\omega}}{2}$$

$$H(\omega) = \frac{1 + e^{i\omega}}{2(e^{i\omega} - 1)} = \frac{\cos \frac{\omega}{2}}{2i \sin \frac{\omega}{2}}$$

$$\tilde{H}(f) = \frac{\cos(\pi f)}{2i \sin(\pi f)}$$

Точное значение интеграла $e^{i\omega t}$ равно $\frac{e^{i\omega t}}{i\omega}$, тогда отношение значений:

$$\gamma = \frac{\text{Вычисленное}}{\text{Точное}} = \cos \frac{\omega}{2} \frac{\frac{\omega}{2}}{\sin \frac{\omega}{2}} = 1 - \frac{\omega^2}{12} + \frac{\omega^4}{720} + \dots$$

$$\gamma = \cos(\pi f) \frac{\pi f}{\sin(\pi f)} = 1 - \frac{\pi^2 \omega^2}{3} + \frac{\pi^4 \omega^4}{45} + \dots$$

Формула Симпсона:

$$y_{n+1} = y_{n-1} + \frac{1}{3}(s_{n-1} + 4s_n + s_{n+1}), y_0 = 0$$

Пусть $s_n = e^{i\omega n}$ и $y_n = H(\omega)e^{i\omega n}$, тогда:

$$\begin{cases} y_{n+1} = H(\omega)e^{i\omega(n-1)} + \frac{e^{i\omega(n-1)} + 4e^{i\omega n} + e^{i\omega(n+1)}}{3} \\ y_{n+1} = H(\omega)e^{i\omega(n+1)} \end{cases}$$

$$H(\omega)(e^{i\omega n}e^{i\omega}) = H(\omega)e^{i\omega n}e^{-i\omega} + e^{i\omega n} \frac{e^{-i\omega} + 4 + e^{i\omega}}{3}$$

$$H(\omega)(e^{i\omega} - e^{-i\omega}) = \frac{e^{-i\omega} + 4 + e^{i\omega}}{3}$$

$$H(\omega) = \frac{e^{-i\omega} + 4 + e^{i\omega}}{3(e^{i\omega} - e^{-i\omega})} = \frac{\cos \omega + 2}{3i \sin \omega}$$

$$\tilde{H}(f) = \frac{\cos(2\pi f) + 2}{3i \sin(2\pi f)}$$

Точное значение интеграла $e^{i\omega t}$ равно $\frac{e^{i\omega t}}{i\omega}$, тогда отношение значений:

$$\gamma = \frac{\text{Вычисленное}}{\text{Точное}} = \frac{(\cos \omega + 2)i\omega}{3i \sin \omega} = \frac{\cos \omega + 2}{3} \cdot \frac{\omega}{\sin \omega} = 1 + \frac{\omega^4}{180} + \dots$$

$$\gamma = \frac{\cos(2\pi f) + 2}{3} \cdot \frac{2\pi f}{\sin(2\pi f)} = 1 + \frac{4\pi^4 f^4}{45} + \dots$$

Графики передаточных функций и графики отношения вычисляемого в результате фильтрации значения к истинному для формул прямоугольников, трапеций и Симпсона представлены на рис. 9, 10, 11 соответственно.

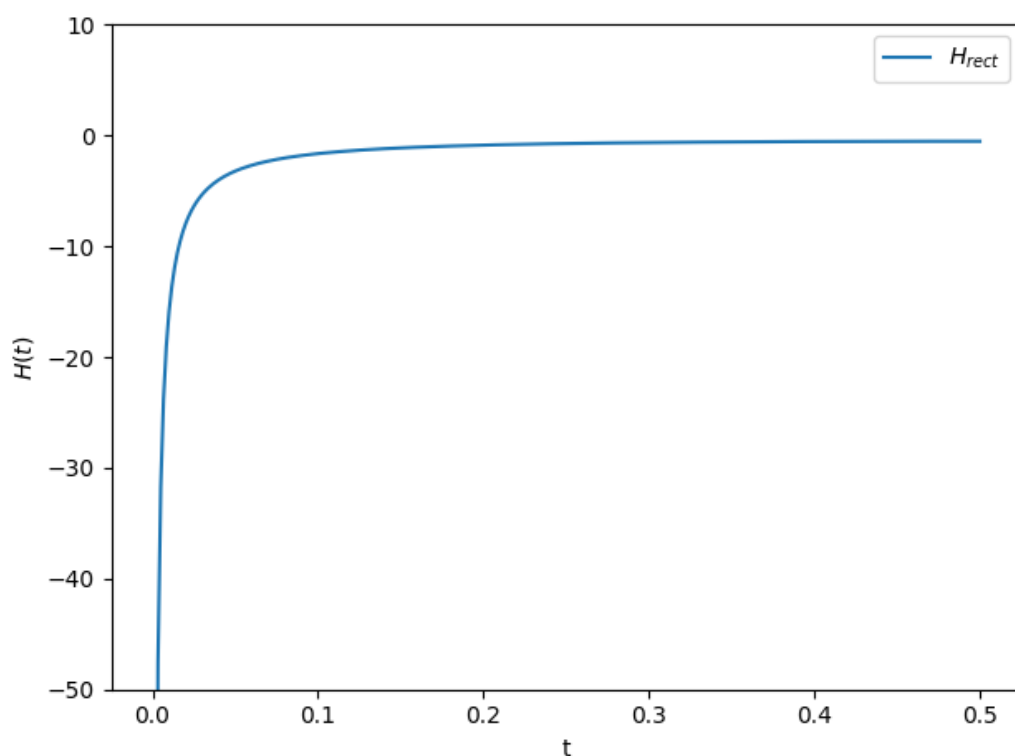


Рисунок 9 – График отношения вычисляемого в результате фильтрации значения к истинному для формул прямоугольников

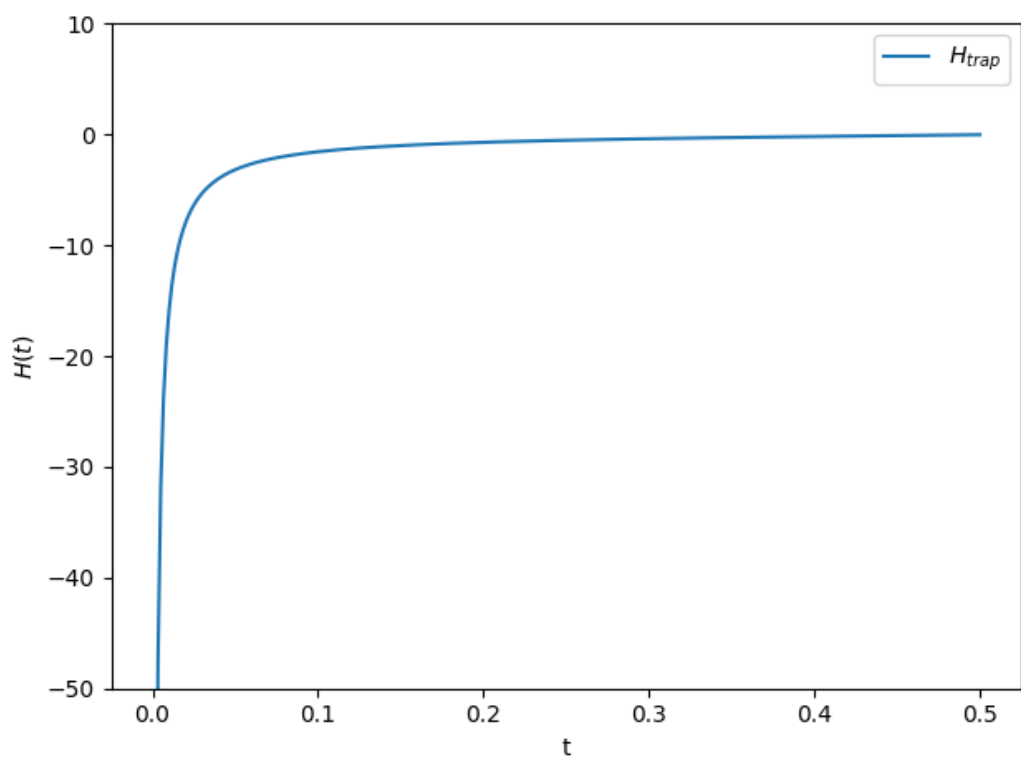


Рисунок 10 – График отношения вычисляемого в результате фильтрации значения к истинному(b) для формул трапеций

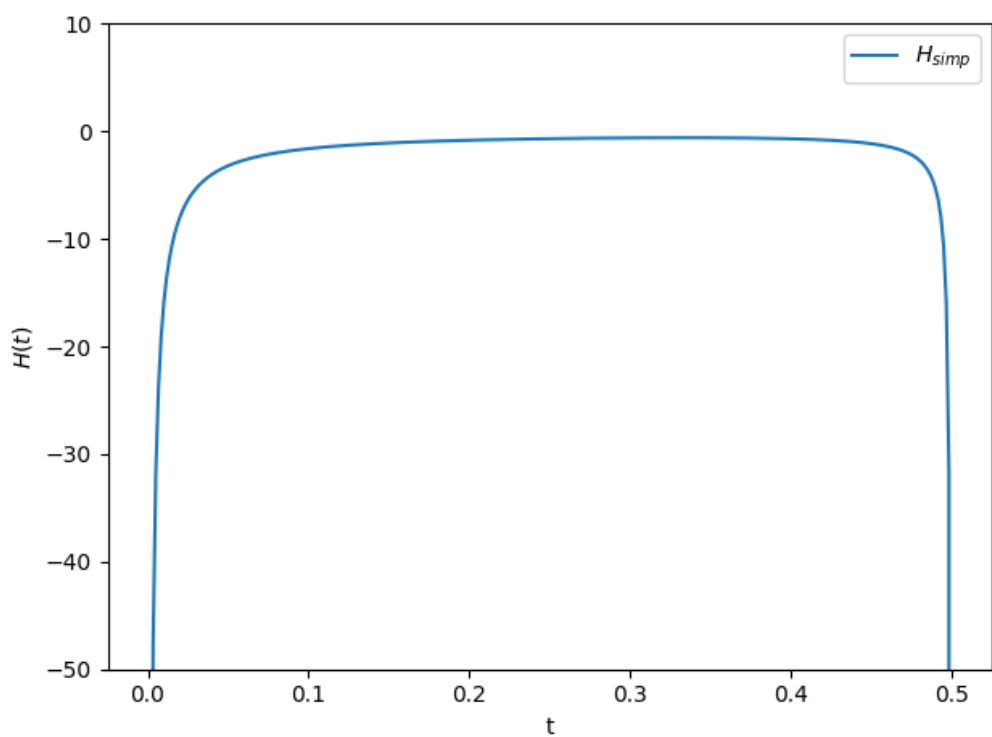


Рисунок 11 – График отношения вычисляемого в результате фильтрации значения к истинному(b) для формул Симпсона

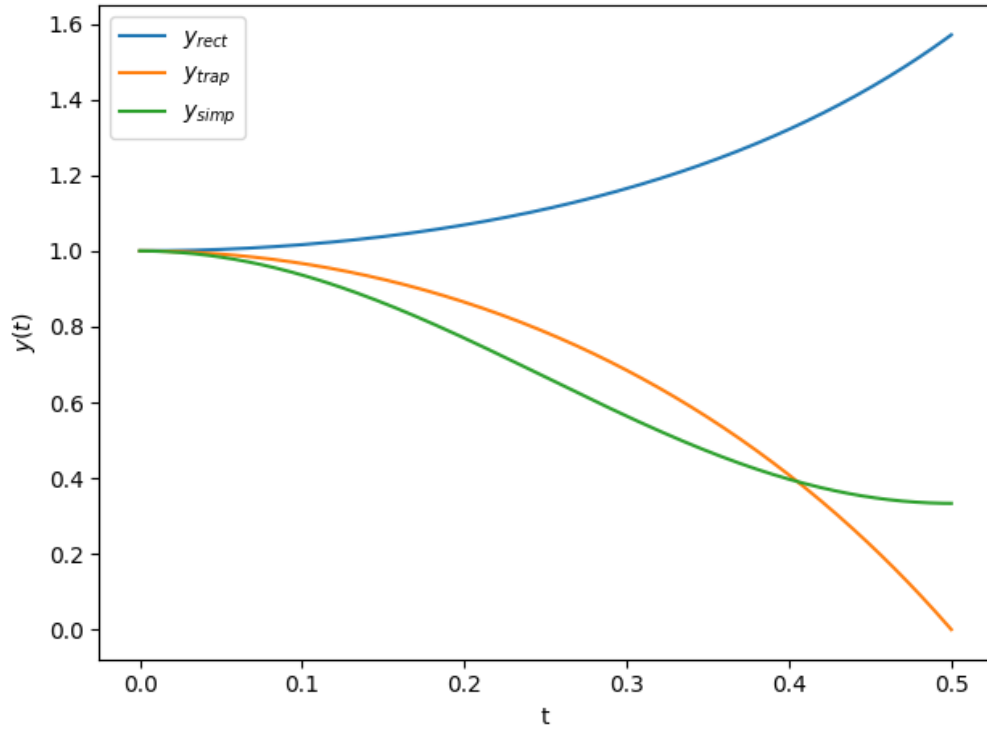


Рисунок 12 – Графики отношения вычисляемого в результате фильтрации значения к истинному (b) для формул прямоугольников, для формул трапеций и для формул Симпсона

8) Выведем формулу передаточной функции рекурсивного фильтра, соответствующего квадратурной формуле. Построим график передаточной функции и график отношения вычисляемого в результате фильтрации значения к истинному. Проинтерпретируем частотные свойства передаточной функции.

$$y_{n+2} = y_{n-1} + \frac{1}{8}(s_{n+2} + 3s_{n+1} + 3s_n + s_{n-1})$$

Пусть $s_n = e^{i\omega n}$ и тогда $H(\omega)e^{i\omega n}$, тогда:

$$\begin{cases} y_{n+2} = H(\omega)e^{i\omega(n-1)} + \frac{e^{i\omega(n+2)} + 3e^{i\omega(n+1)} + 3e^{i\omega n} + e^{i\omega(n-1)}}{8} \\ y_{n+2} = H(\omega)e^{i\omega(n+2)} \end{cases}$$

$$H(\omega)(e^{i\omega n}e^{2i\omega}) = H(\omega)e^{i\omega n}e^{-i\omega} + e^{i\omega n} \frac{e^{2i\omega} + 3e^{i\omega} + 3 + e^{-i\omega}}{8}$$

$$H(\omega)(e^{2i\omega} - e^{-i\omega}) = \frac{e^{2i\omega} + 3e^{i\omega} + 3 + e^{-i\omega}}{8}$$

$$H(\omega) = \frac{e^{2i\omega} + 3e^{i\omega} + 3 + e^{-i\omega}}{8(e^{2i\omega} - e^{-i\omega})} \cdot \frac{e^{-\frac{i\omega}{2}}}{e^{-\frac{i\omega}{2}}}$$

$$H(\omega) = \frac{e^{\frac{3i\omega}{2}} + 3e^{\frac{i\omega}{2}} + 3e^{-\frac{i\omega}{2}} + e^{-\frac{3i\omega}{2}}}{8(e^{\frac{3i\omega}{2}} - e^{-\frac{3i\omega}{2}})} = \frac{2\cos\frac{3\omega}{2} + 6\cos\frac{\omega}{2}}{16i\sin\frac{3\omega}{2}}$$

$$\tilde{H}(f) = \frac{\cos(3\pi f) + 3\cos(\pi f)}{8i\sin(3\pi f)}$$

Точное значение интеграла $e^{i\omega t}$ равно $\frac{e^{i\omega t}}{i\omega}$, тогда отношение значений:

$$\gamma = \frac{\text{Вычисленное}}{\text{Точное}} = \omega \frac{\cos\frac{3\omega}{2} + 3\cos\frac{\omega}{2}}{8\sin\frac{3\omega}{2}} = \frac{1}{12} \left(\cos\frac{3\omega}{2} + 3\cos\frac{\omega}{2} \right) \cdot \frac{\frac{3\omega}{2}}{\sin\frac{3\omega}{2}}$$

$$= \frac{1}{3} + \frac{\omega^4}{240} + \dots$$

$$\gamma = \frac{1}{12} (\cos(3\pi f) + 3\cos(\pi f)) \frac{3\pi f}{\sin(3\pi f)} = \frac{1}{3} + \frac{\pi^4 f^4}{15} + \dots$$

Графики передаточных функций и график отношения вычисляемого в результате фильтрации значения к истинному представлены на рис. 13 и на рис. соответственно.

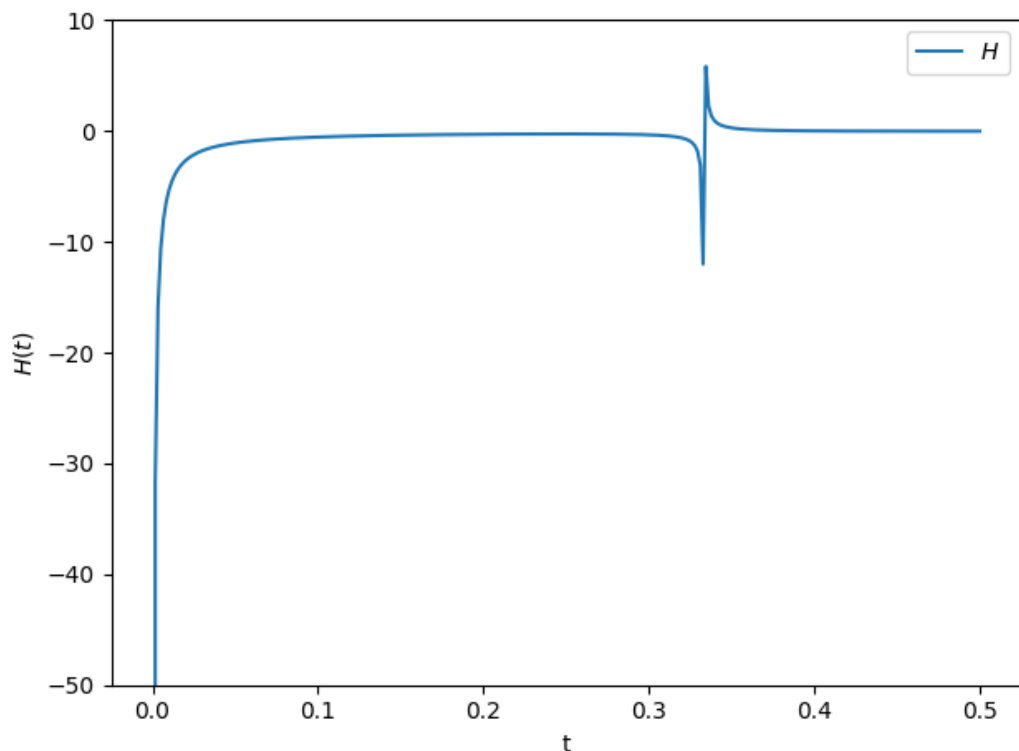


Рисунок 13 – График передаточной функции

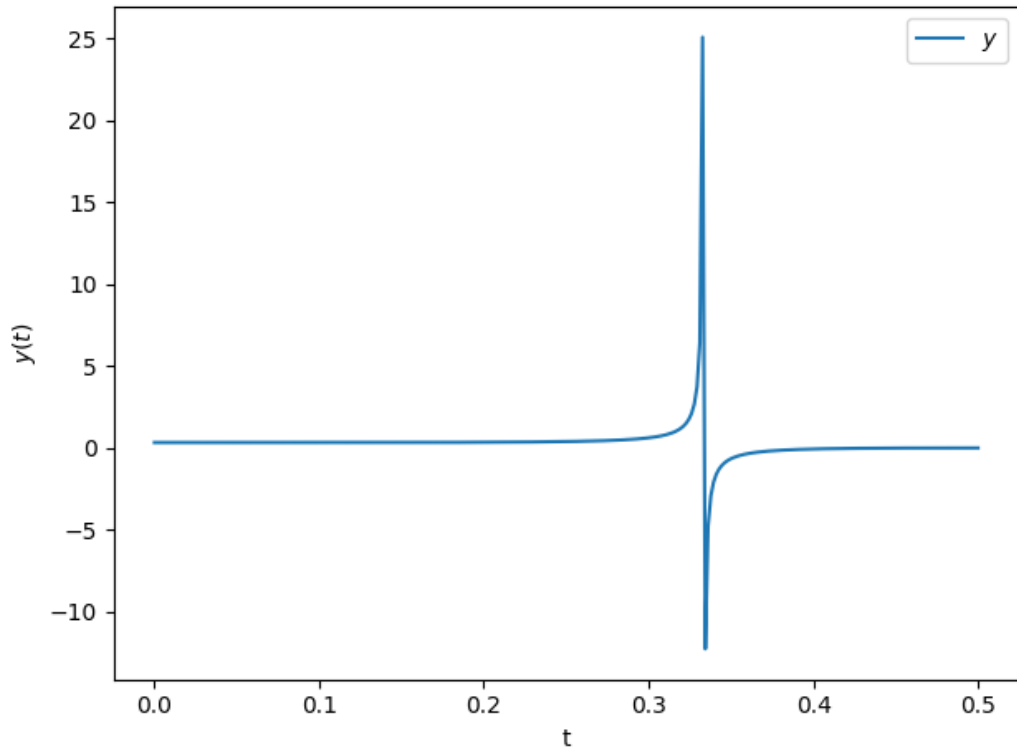


Рисунок 14 – График отношения вычисляемого в результате фильтрации значения к истинному

9) Проведем сопоставительный анализ частотных характеристик передаточных функций, полученных при выполнении пп. 7 и 8.

Были получены графики передаточных функций рекурсивных фильтров, соответствующих квадратурным формулам прямоугольников, трапеций, Симпсона, а также квадратурной формуле, полученной в п.8. Графики представлены на рис. 9 - 14 и соответствуют ожиданиям.

Выводы.

В ходе выполнения лабораторной работы был проведен анализ частотных характеристик известных формул полиномиального сглаживания временных рядов и формул численного интегрирования, были построены соответствующие графики.

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД

```
import numpy as np
import matplotlib.pyplot as plt
import math

# task1
"""
h_3 = lambda x: np.sin(3*math.pi*x)/(3*np.sin(x*math.pi))
h_5 = lambda x: np.sin(5*math.pi*x)/(5*np.sin(x*math.pi))
h_7 = lambda x: np.sin(7*math.pi*x)/(7*np.sin(x*math.pi))
h_9 = lambda x: np.sin(9*math.pi*x)/(9*np.sin(x*math.pi))

t = np.linspace(0.001, 1.0, 64)

plt.plot(t, h_3(t), label=r'$H_3$')
plt.plot(t, h_5(t), label=r'$H_5$')
plt.plot(t, h_7(t), label=r'$H_7$')
plt.plot(t, h_9(t), label=r'$H_9$')
plt.xlabel('t')
plt.ylabel(r'$H(t)$')
plt.legend()
plt.show()
"""

# task2
"""
h_7 = lambda x: 1 / 21 * (7 + 12 * np.cos(x) + 6 * np.cos(2 * x) - 4 *
np.cos(3 * x))
h_9 = lambda x: 1 / 231 * (59 + 108 * np.cos(x) + 78 * np.cos(2 * x) +
28 * np.cos(3 * x) - 42 * np.cos(4 * x))
h_11 = lambda x: 1 / 429 * (
89 + 168 * np.cos(x) + 138 * np.cos(2 * x) + 88 * np.cos(3
* x) + 18 * np.cos(4 * x) - 72 * np.cos(
5 * x))
h_13 = lambda x: 1 / 143 * (
25 + 48 * np.cos(x) + 42 * np.cos(2 * x) + 32 * np.cos(3 *
x) + 18 * np.cos(4 * x) - 22 * np.cos(6 * x))

t = np.linspace(0, 2 * math.pi, 64)

plt.plot(t, h_7(t), label=r'$H_7$')
plt.plot(t, h_9(t), label=r'$H_9$')
plt.plot(t, h_11(t), label=r'$H_{11}$')
plt.plot(t, h_13(t), label=r'$H_{13}$')
plt.xlabel('t')
plt.ylabel(r'$H(t)$')
plt.legend()
plt.show()
```

```

"""

# task3
"""
h_9 = lambda x: 1 / 429 * (179 + 270 * np.cos(x) + 60 * np.cos(2 * x) -
110 * np.cos(3 * x) + 30 * np.cos(4 * x))
h_11 = lambda x: 1 / 429 * (143 + 240 * np.cos(x) + 120 * np.cos(2 * x)
- 20 * np.cos(3 * x) - 90 * np.cos(4 * x) + 36 * np.cos(5 * x))
h_13 = lambda x: 1 / 2431 * (677 + 1200 * np.cos(x) + 780 * np.cos(2 *
x) + 220 * np.cos(3 * x) - 270 * np.cos(4 * x) - 396 * np.cos(5 * x) +
220 * np.cos(6 * x))
h_15 = lambda x: 1 / 46189 * (11063 + 20250 * np.cos(x) + 15000 *
np.cos(2 * x) + 7510 * np.cos(3 * x) - 330 * np.cos(4 * x) - 5874 *
np.cos(5 * x) - 5720 * np.cos(6 * x) + 4290 * np.cos(7 * x))

t = np.linspace(0, 2 * math.pi, 64)

plt.plot(t, h_9(t), label=r'$H_9$')
plt.plot(t, h_11(t), label=r'$H_{11}$')
plt.plot(t, h_13(t), label=r'$H_{13}$')
plt.plot(t, h_15(t), label=r'$H_{15}$')
plt.xlabel('t')
plt.ylabel(r'$H(t)$')
plt.legend()
plt.show()
"""

# task4
"""
h_21 = lambda x: 1/320 * (74 + 134*np.cos(2*math.pi*x) +
92*np.cos(4*math.pi*x) + 42*np.cos(6*math.pi*x) + 6*np.cos(8*math.pi*x)
- 10*np.cos(10*math.pi*x) - 12*np.cos(12*math.pi*x) -
6*np.cos(14*math.pi*x))
h_15 = lambda x: 1/350 * (60 + 114*np.cos(2*math.pi*x) +
94*np.cos(4*math.pi*x) + 66*np.cos(6*math.pi*x) +
36*np.cos(8*math.pi*x) + 12*np.cos(10*math.pi*x) -
4*np.cos(12*math.pi*x) - 10*np.cos(14*math.pi*x) -
10*np.cos(16*math.pi*x) - 6*np.cos(18*math.pi*x) -
2*np.cos(20*math.pi*x))

t = np.linspace(0, 1, 64)

plt.plot(t, h_21(t), label=r'$H_{21}$')
plt.plot(t, h_15(t), label=r'$H_{15}$')
plt.xlabel('t')
plt.ylabel(r'$H(t)$')
plt.legend()
plt.show()
"""

```

```

#task5
"""
def task1_inner():
    h_3 = lambda x: np.sin(3 * math.pi * x) / (3 * np.sin(x * math.pi))
    h_5 = lambda x: np.sin(5 * math.pi * x) / (5 * np.sin(x * math.pi))
    h_7 = lambda x: np.sin(7 * math.pi * x) / (7 * np.sin(x * math.pi))
    h_9 = lambda x: np.sin(9 * math.pi * x) / (9 * np.sin(x * math.pi))

    t = np.linspace(0.001, 1, 64)

    plt.plot(t, 20*np.log10(np.abs(h_3(t))), label=r'$H_3$')
    plt.plot(t, 20*np.log10(np.abs(h_5(t))), label=r'$H_5$')
    plt.plot(t, 20*np.log10(np.abs(h_7(t))), label=r'$H_7$')
    plt.plot(t, 20*np.log10(np.abs(h_9(t))), label=r'$H_9$')
    plt.xlabel('t')
    plt.ylabel(r'$H(t)$')
    plt.legend()
    plt.show()

def task2_inner():
    h_7 = lambda x: 1 / 21 * (7 + 12 * np.cos(x) + 6 * np.cos(2 * x) -
4 * np.cos(3 * x))
    h_9 = lambda x: 1 / 231 * (59 + 108 * np.cos(x) + 78 * np.cos(2 *
x) + 28 * np.cos(3 * x) - 42 * np.cos(4 * x))
    h_11 = lambda x: 1 / 429 * (
        89 + 168 * np.cos(x) + 138 * np.cos(2 * x) + 88 * np.cos(3
* x) + 18 * np.cos(4 * x) - 72 * np.cos(
        5 * x))
    h_13 = lambda x: 1 / 143 * (
        25 + 48 * np.cos(x) + 42 * np.cos(2 * x) + 32 * np.cos(3 *
x) + 18 * np.cos(4 * x) - 22 * np.cos(6 * x))

    t = np.linspace(0.001, 2*math.pi, 64)

    plt.plot(t, 20*np.log10(np.abs(h_7(t))), label=r'$H_7$')
    plt.plot(t, 20*np.log10(np.abs(h_9(t))), label=r'$H_9$')
    plt.plot(t, 20*np.log10(np.abs(h_11(t))), label=r'$H_{11}$')
    plt.plot(t, 20*np.log10(np.abs(h_13(t))), label=r'$H_{13}$')
    plt.xlabel('t')
    plt.ylabel(r'$H(t)$')
    plt.legend()
    plt.show()

def task3_inner():
    h_9 = lambda x: 1 / 429 * (
        179 + 270 * np.cos(x) + 60 * np.cos(2 * x) - 110 *
np.cos(3 * x) + 30 * np.cos(4 * x))
    h_11 = lambda x: 1 / 429 * (

```

```

        143 + 240 * np.cos(x) + 120 * np.cos(2 * x) - 20 *
np.cos(3 * x) - 90 * np.cos(4 * x) + 36 * np.cos(
        5 * x))
    h_13 = lambda x: 1 / 2431 * (677 + 1200 * np.cos(x) + 780 * np.cos(2
* x) + 220 * np.cos(3 * x) - 270 * np.cos(
        4 * x) - 396 * np.cos(5 * x) + 220 * np.cos(6 * x))
    h_15 = lambda x: 1 / 46189 * (
        11063 + 20250 * np.cos(x) + 15000 * np.cos(2 * x) + 7510
* np.cos(3 * x) - 330 * np.cos(
        4 * x) - 5874 * np.cos(5 * x) - 5720 * np.cos(6 * x) + 4290
* np.cos(7 * x))

    t = np.linspace(0.001, 2*math.pi, 64)

    plt.plot(t, 20*np.log10(np.abs(h_9(t))), label=r'$H_9$')
    plt.plot(t, 20*np.log10(np.abs(h_11(t))), label=r'$H_{11}$')
    plt.plot(t, 20*np.log10(np.abs(h_13(t))), label=r'$H_{13}$')
    plt.plot(t, 20*np.log10(np.abs(h_15(t))), label=r'$H_{15}$')
    plt.xlabel('t')
    plt.ylabel(r'$H(t)$')
    plt.legend()
    plt.show()

def task4_inner():
    h_21 = lambda x: 1 / 320 * (74 + 134 * np.cos(2 * math.pi * x) + 92
* np.cos(4 * math.pi * x) + 42 * np.cos(
        6 * math.pi * x) + 6 * np.cos(8 * math.pi * x) - 10 * np.cos(10
* math.pi * x) - 12 * np.cos(
        12 * math.pi * x) - 6 * np.cos(14 * math.pi * x))
    h_15 = lambda x: 1 / 350 * (60 + 114 * np.cos(2 * math.pi * x) + 94
* np.cos(4 * math.pi * x) + 66 * np.cos(
        6 * math.pi * x) + 36 * np.cos(8 * math.pi * x) + 12 * np.cos(10
* math.pi * x) - 4 * np.cos(
        12 * math.pi * x) - 10 * np.cos(14 * math.pi * x) - 10 * np.cos(16
* math.pi * x) - 6 * np.cos(
        18 * math.pi * x) - 2 * np.cos(20 * math.pi * x))

    t = np.linspace(0.001, 1, 64)

    plt.plot(t, 20*np.log10(np.abs(h_21(t))), label=r'$H_{21}$')
    plt.plot(t, 20*np.log10(np.abs(h_15(t))), label=r'$H_{15}$')
    plt.xlabel('t')
    plt.ylabel(r'$H(t)$')
    plt.legend()
    plt.show()

task1_inner()
task2_inner()
task3_inner()
task4_inner()

```

```

"""

# task7
"""
h_rect = lambda x: (1 / (2j*np.sin(math.pi*x))).imag
h_trap = lambda x: (np.cos(math.pi*x) / (2j*np.sin(math.pi*x))).imag
h_simp      =      lambda      x:      ((np.cos(2*math.pi*x)+2) /
(3j*np.sin(2*math.pi*x))).imag

y_rect = lambda x: math.pi*x / (np.sin(math.pi * x))
y_trap = lambda x: np.cos(math.pi * x)*(math.pi*x/np.sin(x*math.pi))
y_simp = lambda x: (np.cos(2*math.pi*x) + 2)/3

t = np.linspace(1e-10, 0.5, 300)

plt.plot(t, h_rect(t), label=r'$H_{rect}$')
plt.xlabel('t')
plt.ylabel(r'$H(t)$')
plt.legend()
x1, x2, y1, y2 = plt.axis()
plt.axis((x1, x2, -50, 10.))
plt.show()

plt.plot(t, h_trap(t), label=r'$H_{trap}$')
plt.xlabel('t')
plt.ylabel(r'$H(t)$')
plt.legend()
x1, x2, y1, y2 = plt.axis()
plt.axis((x1, x2, -50, 10.))
plt.show()

plt.plot(t, h_simp(t), label=r'$H_{simp}$')
plt.xlabel('t')
plt.ylabel(r'$H(t)$')
plt.legend()
x1, x2, y1, y2 = plt.axis()
plt.axis((x1, x2, -50, 10.))
plt.show()

plt.plot(t, y_rect(t), label=r'$y_{rect}$')
plt.plot(t, y_trap(t), label=r'$y_{trap}$')
plt.plot(t, y_simp(t), label=r'$y_{simp}$')
plt.xlabel('t')
plt.ylabel(r'$y(t)$')
plt.legend()
plt.show()
"""

```



```

# task8
"""
h                                =                                lambda                                x:
((np.cos(3*math.pi*x)+3*np.cos(math.pi*x))/(8j*np.sin(3*math.pi*x))).i
mag
y                                =                                lambda                                x:
(1/12)*(np.cos(3*math.pi*x)+3*np.cos(math.pi*x))*((3*math.pi*x)/np.sin
(3*math.pi*x))

t = np.linspace(1e-10, 0.5, 300)

plt.plot(t, h(t), label=r'$H$', )
plt.xlabel('t')
plt.ylabel(r'$H(t)$')
plt.legend()
x1, x2, y1, y2 = plt.axis()
plt.axis((x1, x2, -50, 10.))
plt.show()

plt.plot(t, y(t), label=r'$y$')
plt.xlabel('t')
plt.ylabel(r'$y(t)$')
plt.legend()
plt.show()
"""

```