

Министерство образования и науки Российской Федерации
Федеральное государственное автономное образовательное учреждение высшего
образования

САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ ИТМО

Факультет систем управления и робототехники

Лабораторная работа № 5
"Связь непрерывного и дискретного"
по дисциплине Частотные методы

Выполнил: студент гр. **R3242**

Мареев П. А.

Преподаватель: *Перегудин Алексей Алексеевич*

Санкт-Петербург, 2025

Содержание

1	Задание 1. Непрерывное и дискретное преобразование Фурье.	3
1.1	Численное интегрирование.	5
1.2	Использование DFT.	22
1.3	Мои объяснения.	31
1.4	Приближение непрерывного с помощью DFT.	32
1.5	Выводы.	43
2	Задание 2. Сэмплирование.	44
2.1	Выводы.	53
3	Выводы по лабораторной работе.	54

Вступление

Современные технологии обработки сигналов, телекоммуникаций и анализа данных опираются на математические методы, связывающие непрерывные и дискретные представления информации. Лабораторная работа посвящена изучению взаимосвязи непрерывного и дискретного подходов в контексте преобразований Фурье и теории сэмплирования. Эти методы являются ключевыми для корректного преобразования аналоговых сигналов в цифровую форму, их восстановления и анализа в частотной области.

Основная цель работы – исследовать особенности численного интегрирования, дискретного преобразования Фурье (DFT) и их связь с аналитическими решениями. Практическая часть включает реализацию алгоритмов в среде MATLAB, что позволяет наглядно оценить точность, быстродействие и ограничения различных методов. Особое внимание уделяется аппроксимации непрерывных преобразований с помощью дискретных аналогов, а также анализу ошибок, возникающих при дискретизации и восстановлении сигналов.

В рамках первого задания рассматривается прямоугольная функция и ее Фурье-образ. Сравниваются результаты, полученные через численное интегрирование (функция `trapz`) и унитарное DFT (с использованием `fft` и `ifft`). Это позволяет выявить компромисс между точностью и вычислительной эффективностью. Второе задание фокусируется на теореме Найквиста-Шеннона-Котельникова, демонстрируя ее применение для восстановления непрерывных сигналов из дискретных выборок. Исследуется влияние шага сэмплирования на качество интерполяции, а также связь между временными и частотными характеристиками сигналов.

Работа способствует формированию навыков анализа спектральных свойств сигналов, критической оценки алгоритмов и понимания математических основ цифровой обработки данных.

1 Задание 1. Непрерывное и дискретное преобразование Фурье.

Прежде, чем приступить к выполнению задания, рассмотрим знакомую прямоугольную функцию $\Pi : \mathbb{R} \rightarrow \mathbb{R}$:

$$\Pi(t) = \begin{cases} 1, & |t| \leq \frac{1}{2}, \\ 0, & |t| > \frac{1}{2}. \end{cases}$$

Ее Фурье-образом будет являться аналитическое выражение:

$$\hat{\Pi}(\nu) = \int_{-\infty}^{+\infty} \Pi(t) e^{-2\pi i \nu t} dt.$$

Приведем график исходной функции, построенный при помощи программы.

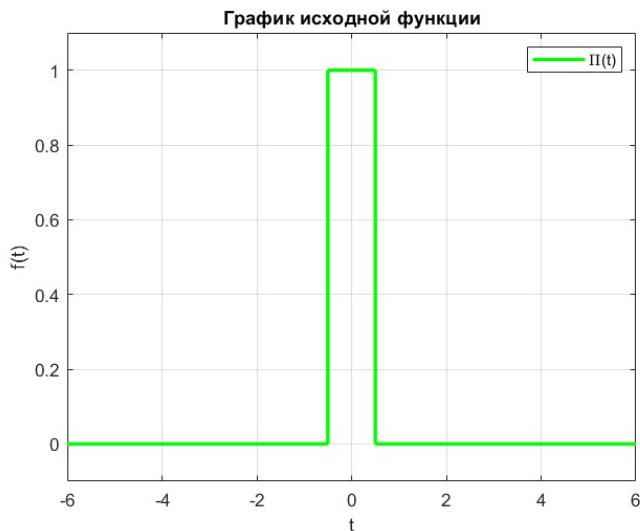


Рис. 1. График исходной функции $\Pi(t)$.

Вручную аналитически вычислим образ функции $\Pi(t)$.

$$\begin{aligned}\hat{\Pi}(\nu) &= \int_{-\infty}^{+\infty} \Pi(t) e^{-2\pi i \nu t} dt = \int_{-\frac{1}{2}}^{\frac{1}{2}} e^{-2\pi i \nu t} dt = \left. \frac{e^{-2\pi i \nu t}}{-2\pi i \nu} \right|_{-\frac{1}{2}}^{\frac{1}{2}} = \\ &= -\frac{e^{-\pi i \nu} - e^{\pi i \nu}}{2\pi i \nu} = \frac{e^{\pi i \nu} - e^{-\pi i \nu}}{2\pi i \nu} = \frac{\sin(\pi \nu)}{\pi \nu} = \text{sinc}(\nu)\end{aligned}$$

Построим график Фурье-образа исходной функции, построенный при помощи [программы](#).

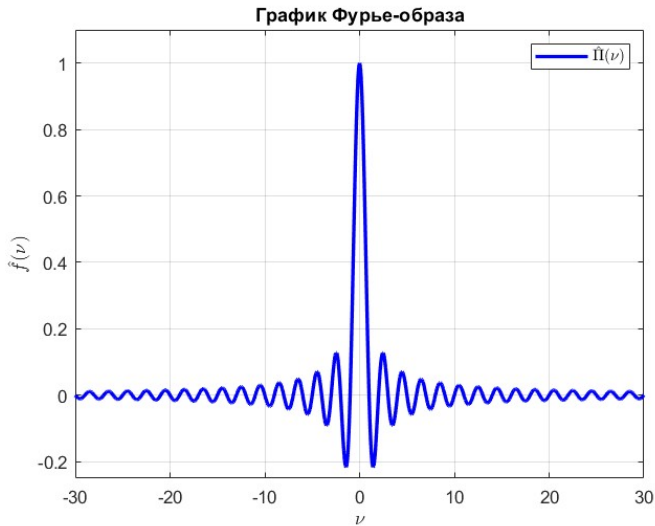


Рис. 2. График Фурье-образа $\hat{\Pi}(\nu)$.

1.1 Численное интегрирование.

Зададим функцию $\Pi(t)$ в MATLAB. Найдем ее Фурье-образ с помощью численного интегрирования (функция `trapz`). Вновь используя численное интегрирование, выполним обратное преобразование Фурье от найденного Фурье-образа для восстановления исходной функции. Схематично представим требуемые действия:

$$\Pi(t) \xrightarrow{\text{trapz}} \hat{\Pi}(\nu) \xrightarrow{\text{trapz}} \Pi(t)$$

Вспомним, как связаны ширина диапазона частот и шаг частоты со значением интервала времени и шага дискретизации:

$$V \sim \frac{1}{dt} \quad dv \sim \frac{1}{T}$$

То есть, можно сформулировать нехитрые следствия на основе этих соотношений:

- С увеличением значения dt диапазон частот V сужается, а с уменьшением – расширяется.
- С увеличением значения T шаг частоты dv становится меньше, а с уменьшением – возрастает.

Таким образом, для получения качественных результатов требуется обеспечить согласованность параметров в обеих областях.

Реализуем [программу](#) для выполнения этого задания при помощи прикладного пакета MATLAB. Исследуем влияние параметров T (промежуток интегрирования по времени), Δt (шаг дискретизации), V (промежуток интегрирования по частоте) и $\Delta \nu$ (шаг частот), выбрав наиболее показательные значения.

Зададимся множеством значений промежутка интегрирования по времени T :

$$T = \{5, 20, 100\}. \quad (1)$$

Выберем теперь значения остальных параметров:

$$\Delta t = dt = 0.001 \quad V = 12 \quad \Delta \nu = dv = 0.01.$$

Рассмотрим последовательно все значения параметра T из множества (1). Построим на своих частотах сравнительные графики аналитического образа $\hat{\Pi}(\nu)$ и функций найденных с помощью численного интегрирования. Дополнительно построим сравнительные графики исходной функций $\Pi(t)$ и восстановленной функции $\Pi(t)$ для выбранных значений $\Delta t, V, \Delta \nu$.

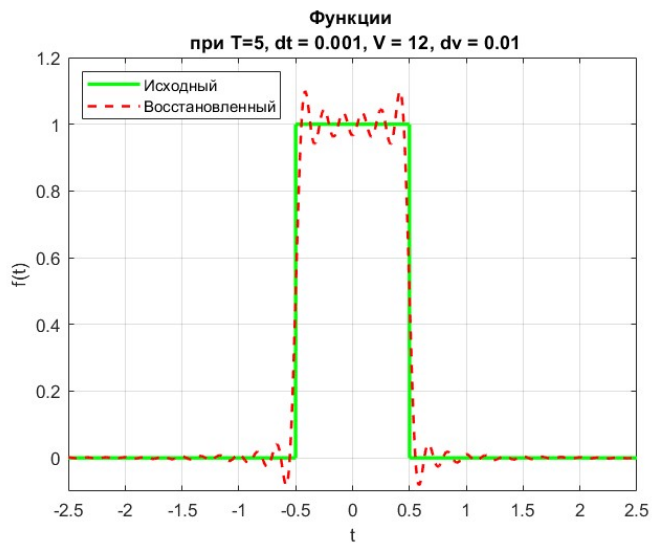


Рис. 3. Исходный и восстановленный сигналы.

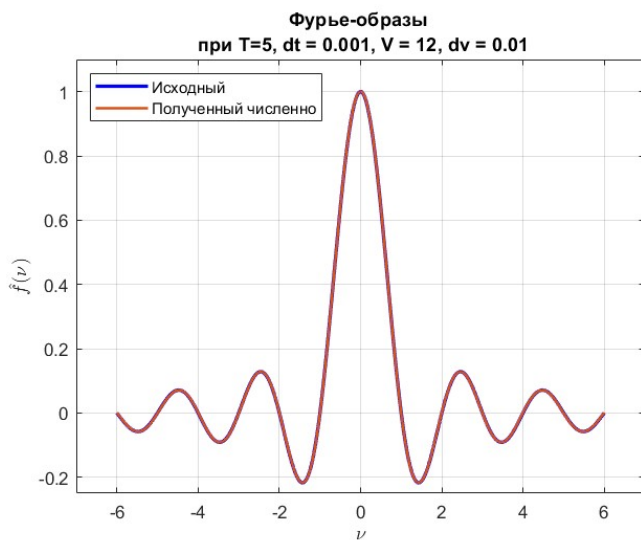


Рис. 4. Фурье-образы: исходный и полученный численно.

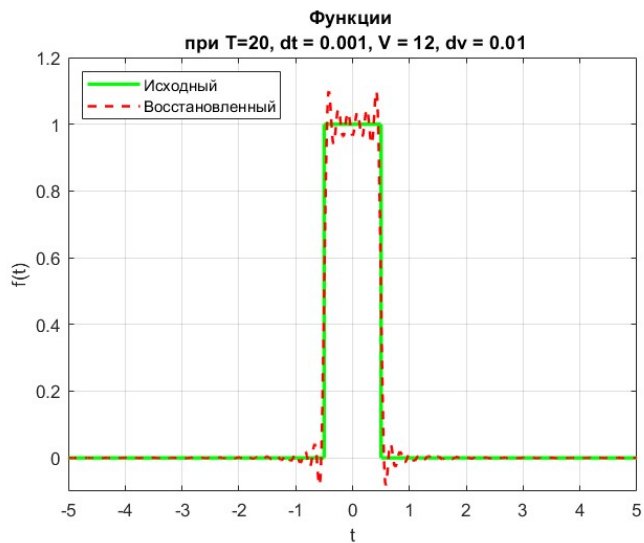


Рис. 5. Исходный и восстановленный сигналы.

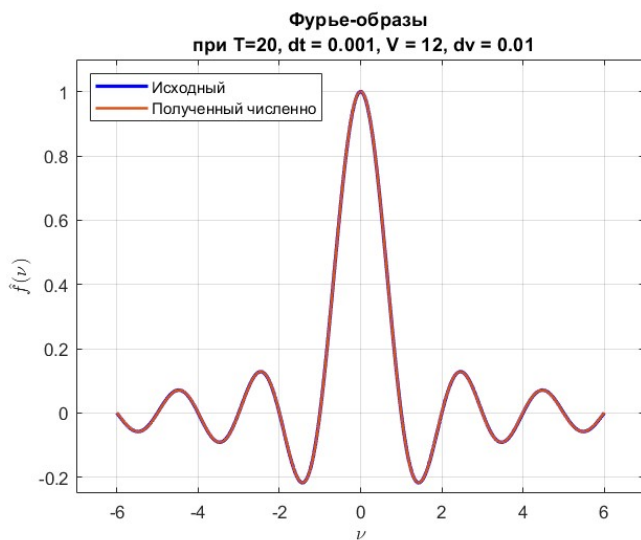


Рис. 6. Фурье-образы: исходный и полученный численно.

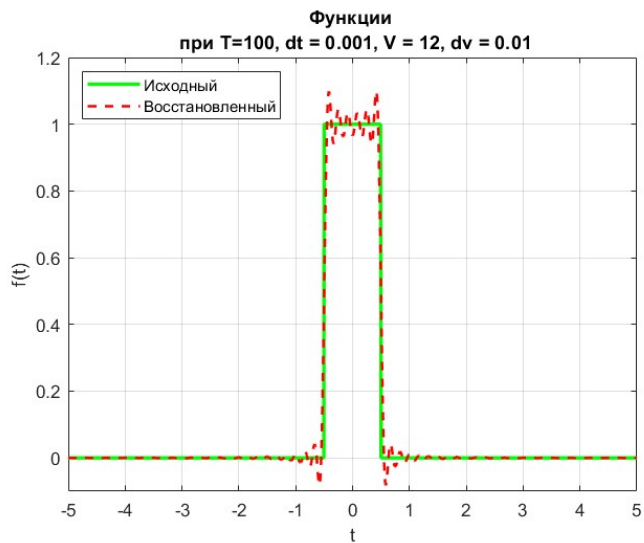


Рис. 7. Исходный и восстановленный сигналы.

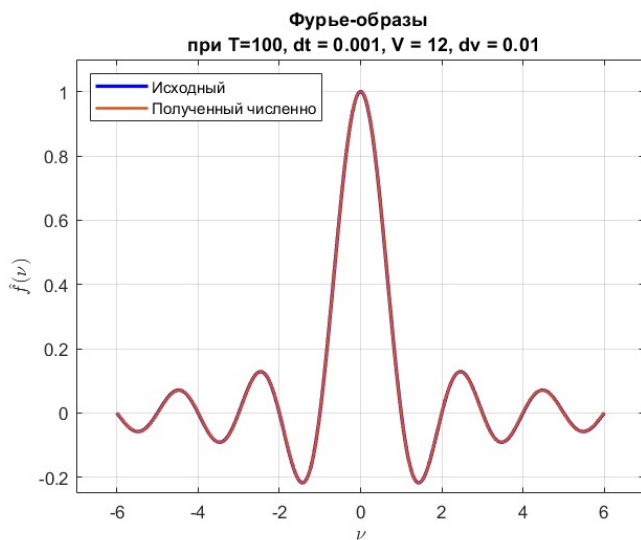


Рис. 8. Фурье-образы: исходный и полученный численно.

Проанализируем полученные графические результаты. На рисунке 3 при $T = 5$ восстановленный сигнал сильно искажен из-за потерь информации ввиду малого значения параметра. При этом Фурье-образ на рисунке 4 найден корректно и совпадает с исходным.

На рисунке 5 при $T = 20$ искажения так же присутствуют, восстановленный сигнал едва похож на исходный. Но Фурье-образы на рисунке 6 снова совпали.

На рисунке 7 при $T = 100$ сигнал восстановлен без особой точности с использованием сравнительно большого количества вычислительных ресурсов. Фурье-образы на рисунке 8 совпадают полностью.

Заметим, что совпадение Фурье-образов говорит о некой «самостоятельности» параметров: каждый из них отвечает за определенную характеристику сигнала в соответствующей области.

Выберем множество значений шага дискретизации Δt :

$$\Delta t = dt = \{1, 0.5, 0.01\}. \quad (2)$$

Определим теперь значения остальных параметров:

$$T = 12 \quad V = 12 \quad \Delta\nu = d\nu = 0.01.$$

Рассмотрим последовательно все значения параметра Δt из множества (2). Построим на своих частотах сравнительные графики аналитического образа $\hat{P}(\nu)$ и функций найденных с помощью численного интегрирования. Дополнительно построим сравнительные графики исходной функций $\Pi(t)$ и восстановленной функции $\Pi(t)$ для выбранных значений $\Delta t, V, \Delta\nu$.

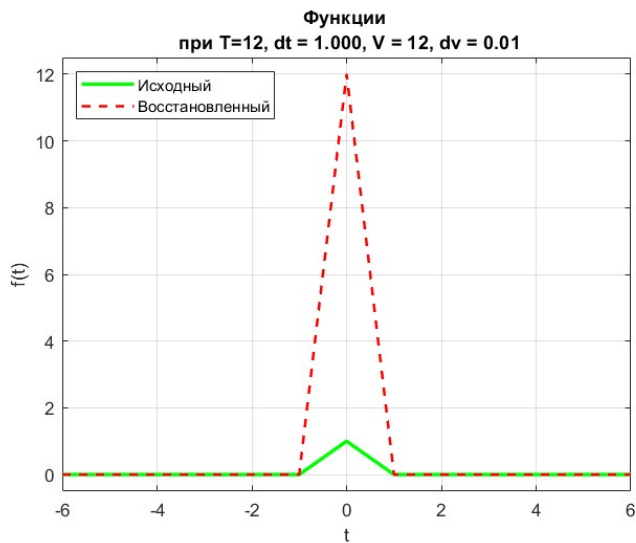


Рис. 9. Исходный и восстановленный сигналы.

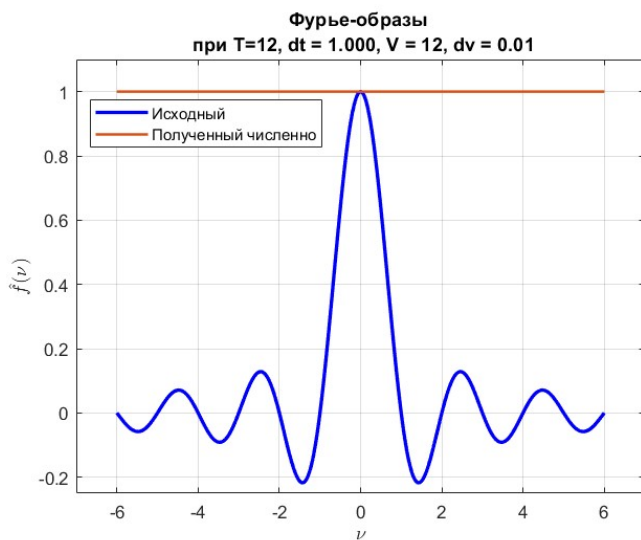


Рис. 10. Фурье-образы: исходный и полученный численно.

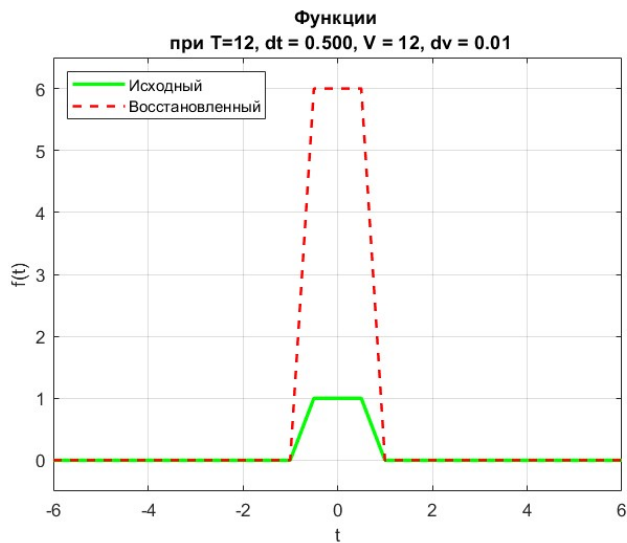


Рис. 11. Исходный и восстановленный сигналы.

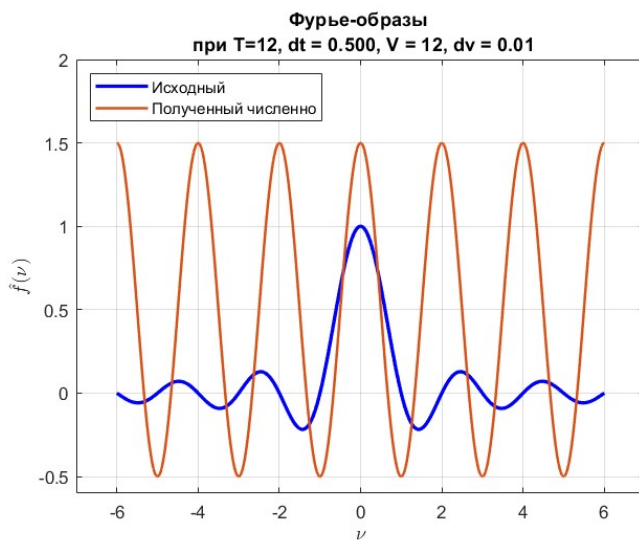


Рис. 12. Фурье-образы: исходный и полученный численно.

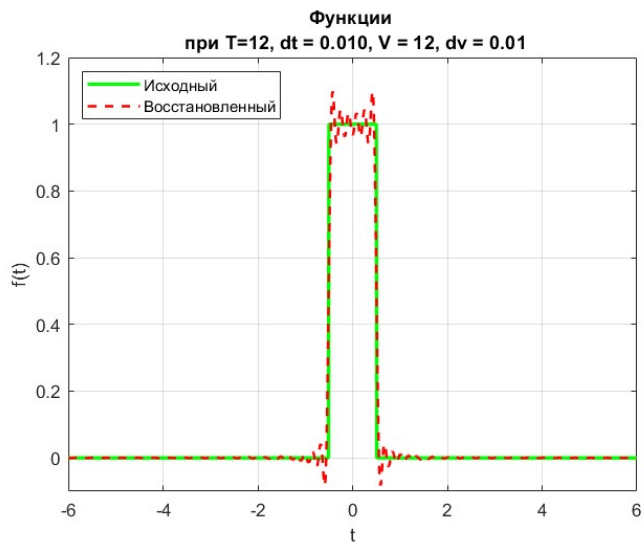


Рис. 13. Исходный и восстановленный сигналы.

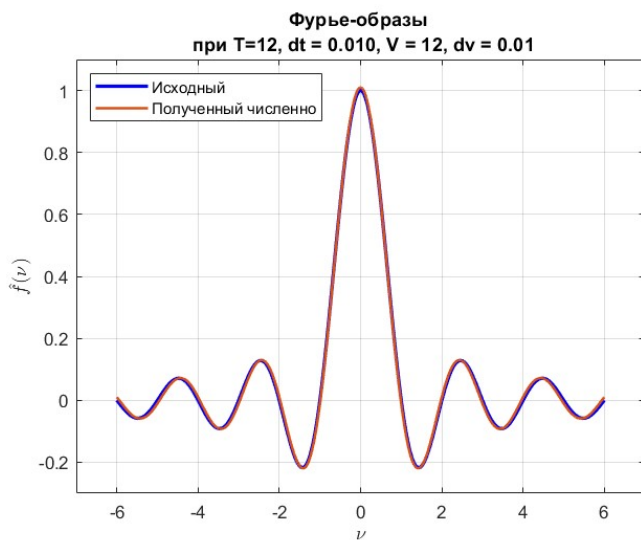


Рис. 14. Фурье-образы: исходный и полученный численно.

Охарактеризуем полученные графики. На рисунке 9 использован сравнительно большой шаг, поэтому какая-либо точность фактически отсутствует. Фурье-образы на рисунке 10 не совпадают из-за слишком большого шага, а значит, возникновения ошибок.

На рисунке 11 при малом увеличении точности удастся заметить очертания исходного сигнала. При этом Фурье-образы на рисунке 12 сильно отличаются: возникла лишняя периодичность при численном интегрировании и довольно большом шаге.

На рисунке 13 шаг позволяет достичь приемлемой точности: функцию удастся восстановить с малыми искажениями. Кроме того, на рисунке 14 видно полное совпадение Фурье-образов.

Становится понятно, что шаг дискретизации Δt играет важную роль: он влияет на качество представления сигнала и во временной, и в частотной области.

Определим множество значений промежутка интегрирования по частоте V :

$$V = \{3, 15, 50\}. \quad (3)$$

Установим теперь значения остальных параметров:

$$T = 12 \quad \Delta t = dt = 0.01 \quad \Delta \nu = dv = 0.01.$$

Рассмотрим последовательно все значения параметра V из множества (3). Построим на своих частотах сравнительные графики аналитического образа $\hat{P}(\nu)$ и функций найденных с помощью численного интегрирования. Дополнительно построим сравнительные графики исходной функций $P(t)$ и восстановленной функции $\Pi(t)$ для выбранных значений $\Delta t, V, \Delta \nu$.

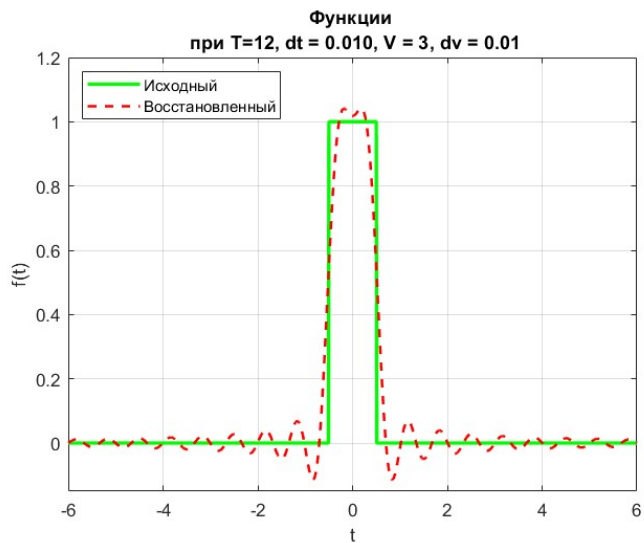


Рис. 15. Исходный и восстановленный сигналы.

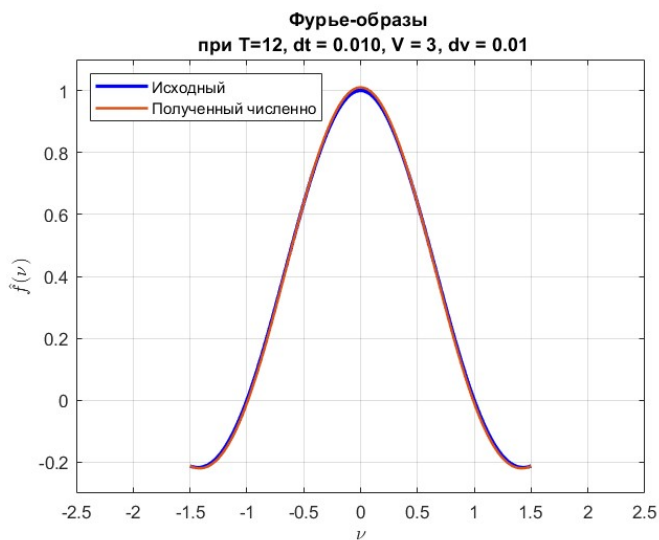


Рис. 16. Фурье-образы: исходный и полученный численно.

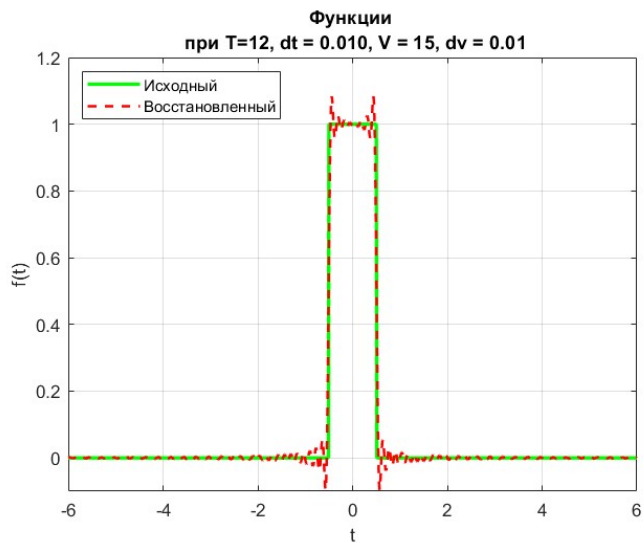


Рис. 17. Исходный и восстановленный сигналы.

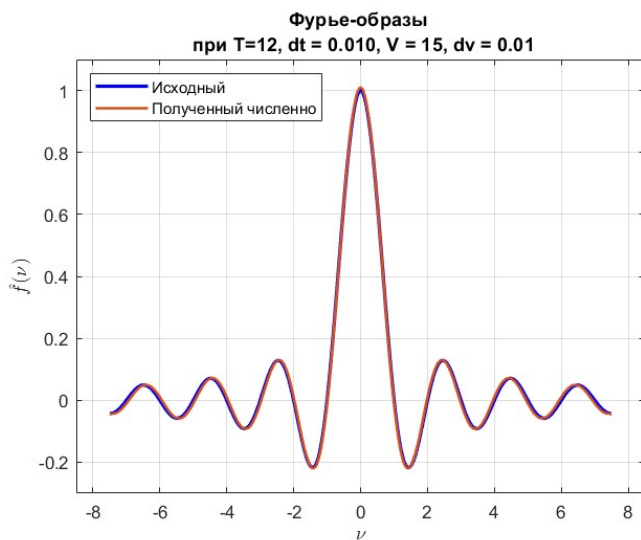


Рис. 18. Фурье-образы: исходный и полученный численно.

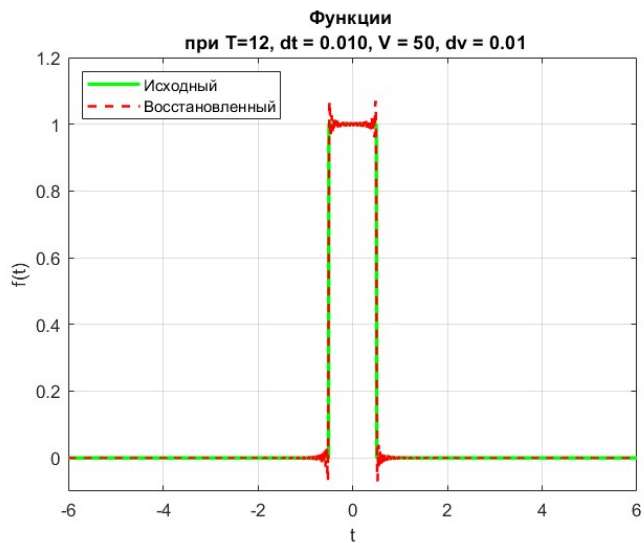


Рис. 19. Исходный и восстановленный сигналы.

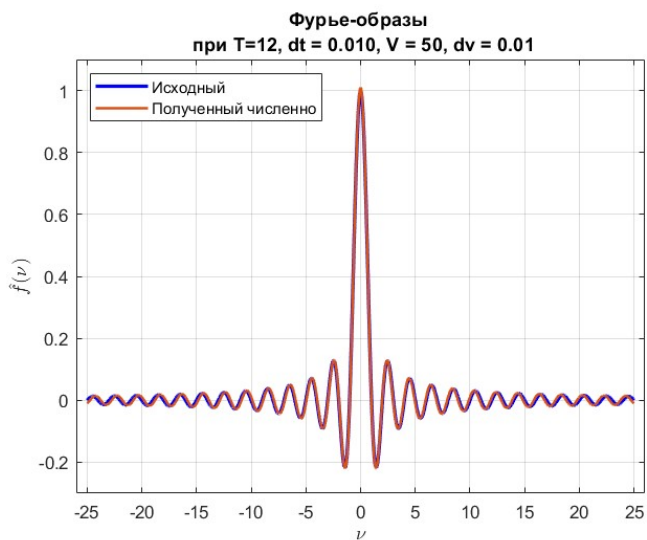


Рис. 20. Фурье-образы: исходный и полученный численно.

Проанализируем полученные результаты. На рисунке 15 восстановленный сигнал из-за осцилляций отличается от исходного. На рисунке 16 Фурье-образы всюду совпадают.

На рисунке 17 восстановленный сигнал совпадает с исходным, но колебания отчетливо заметны. При этом, на рисунке 18 Фурье-образы совпадают полностью.

На рисунке 19 восстановленный сигнал имеет искажения, которые можно явно обнаружить. Однако Фурье-образы на рисунке 20 полностью совпадают.

Следовательно, промежуток интегрирования по частоте V влияет на степень искаженности восстановленного сигнала.

Предложим множество значений шага частот $\Delta\nu$:

$$\Delta\nu = d\nu = \{0.75, 0.01, 0.0001\}. \quad (4)$$

Приведем теперь значения остальных параметров:

$$T = 12 \quad \Delta t = dt = 0.01 \quad V = 12.$$

Рассмотрим последовательно все значения параметра Δt из множества (4). Построим на своих частотах сравнительные графики аналитического образа $\hat{P}(\nu)$ и функций найденных с помощью численного интегрирования. Дополнительно построим сравнительные графики исходной функций $P(t)$ и восстановленной функции $\hat{P}(t)$ для выбранных значений $\Delta t, V, \Delta\nu$.

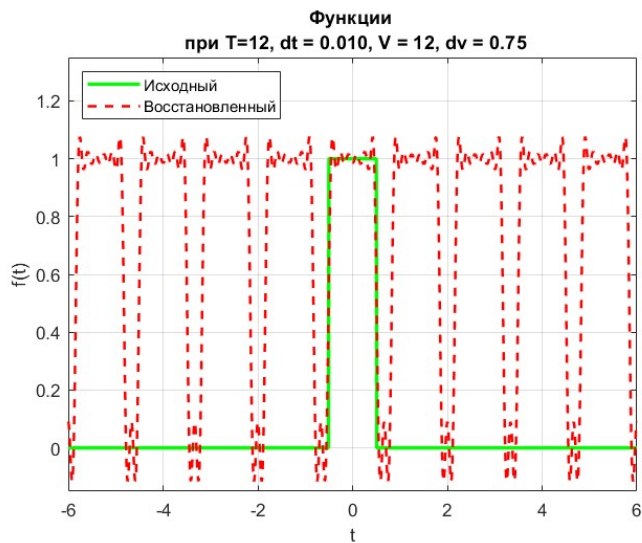


Рис. 21. Исходный и восстановленный сигналы.

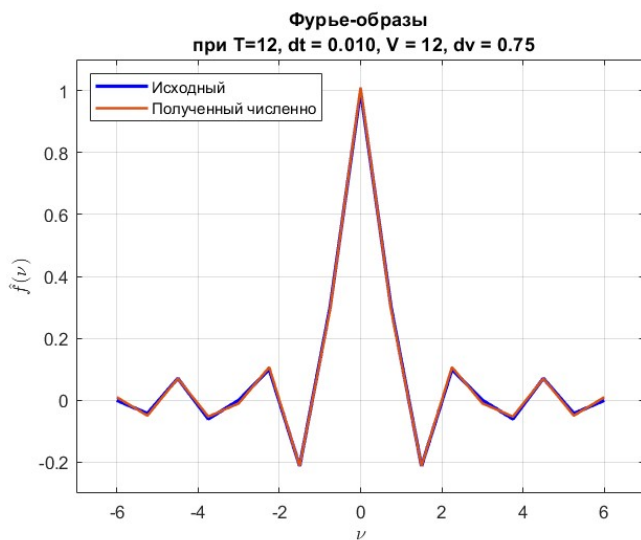


Рис. 22. Фурье-образы: исходный и полученный численно.

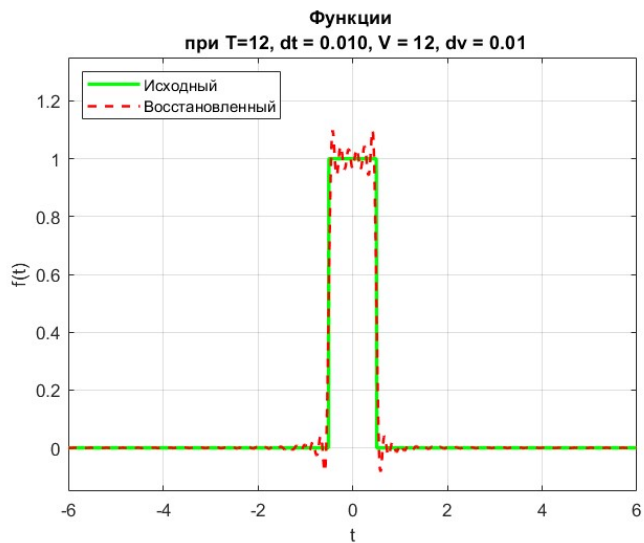


Рис. 23. Исходный и восстановленный сигналы.

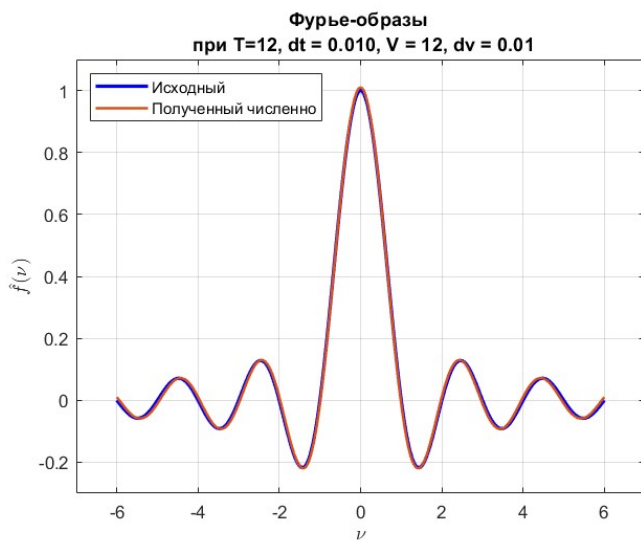


Рис. 24. Фурье-образы: исходный и полученный численно.

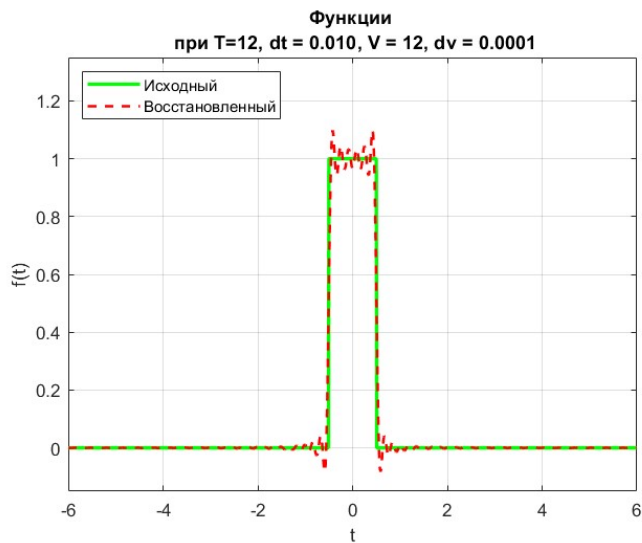


Рис. 25. Исходный и восстановленный сигналы.

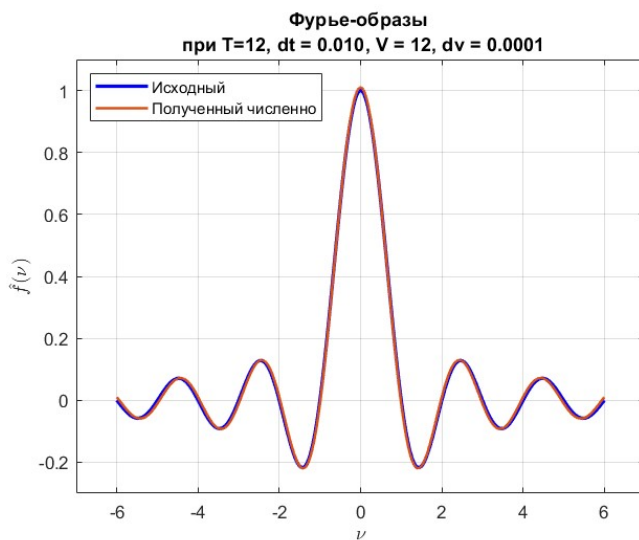


Рис. 26. Фурье-образы: исходный и полученный численно.

Охарактеризуем полученные графики. На графике 21 восстановленный сигнал имеет лишнюю периодичность из-за довольно большого значения шага $\Delta\nu$. Фурье-образы на рисунке 22 выглядят как ломаные, преимущественно совпадают, но не похожи на график функции $\text{sinc}(\nu)$.

На рисунке 23 периодичность отсутствует, а искажения восстановленного сигнала уменьшаются, но все еще заметны. При этом Фурье-образы на рисунке 24 совпадают, лучше приближаясь к графику функции $\text{sinc}(\nu)$.

На рисунке 25 восстановленный сигнал имеет колебания, видный его отличия от исходного. Фурье-образы на рисунке 26 совпадают и представляют собой графики функции $\text{sinc}(\nu)$.

Значит, шаг дискретизации по частоте регулирует искажения восстановленного сигнала и точность при построении Фурье-образа.

Таким образом, можно установить, как влияет каждый из параметров на эффективность использования функции `trapz` для численного интегрирования:

1. T

Влияет на точность моделирования: при малом T сигнал будет искажен, при большом T точность будет улучшена, искажения – уменьшены, но потребуются больше вычислительных ресурсов.

2. Δt

Определяет «полноценность» моделирования: большие значения Δt приводят к возникновению периодичности, а слишком малые увеличивают время моделирования без существенного роста точности.

3. V

Воздействует на достоверность моделирования: при малом V в сигнале появляются осцилляции, а большое V требует оптимального шага $\Delta\nu$ и сравнительно много вычислительных затрат.

4. $\Delta\nu$

Регулирует детализацию моделирования: при большом шаге $\Delta\nu$ возникают искажения, а при слишком малом растут вычислительные затраты без увеличения точности.

Таким образом, корректная настройка параметров для функция `trapz` позволяет найти Фурье-образ с довольно высокой точностью при оптимальных затратах.

Следовательно, для обеспечения высокой точности, функции `trapz` требуется много вычислительных ресурсов, что снижает ее быстродействие.

1.2 Использование DFT.

Найдем Фурье-образ функции $\Pi(t)$ с помощью дискретного преобразования Фурье (конструкция `fftshift(fft())`), используя его так, чтобы преобразование было унитарным. Выполним обратное преобразование от найденного Фурье-образа с помощью обратного дискретного преобразования (конструкция `ifft(ifftshift())`). Схематично представим требуемые действия:

$$\Pi(t) \xrightarrow{\text{fftshift(fft())}} \hat{\Pi}(\nu) \xrightarrow{\text{ifft(ifftshift())}} \Pi(t)$$

Проведем исследования и выберем наиболее показательные значения T (промежуток времени) и Δt (шаг дискретизации), а также найдем соответствующие им $V, \Delta\nu$.

Отметим, что при использовании DFT параметры $T, \Delta t, V, \Delta\nu$ связаны следующими соотношениями:

$$V = \frac{1}{\Delta t} = \frac{1}{dt} \quad \Delta\nu = d\nu = \frac{1}{T}$$

Иначе говоря, выбор значений параметров T и Δt задает характеристики в частотной области.

Выберем множество значений промежутка времени T :

$$T = \{2, 13, 90\}. \quad (5)$$

Зафиксируем величину шага дискретизации Δt :

$$\Delta t = dt = 0.01$$

Рассмотрим последовательно все значения параметра T из множества (5). Реализуем программу при помощи пакета MATLAB и построим на своих частотах сравнительные графики аналитического образа $\hat{\Pi}(\nu)$ и функций найденных с помощью унитарного `fftshift(fft())`. Построим сравнительные графики исходной функции $\Pi(t)$ и восстановленных функций $\Pi(t)$ для выбранных значений T и Δt .

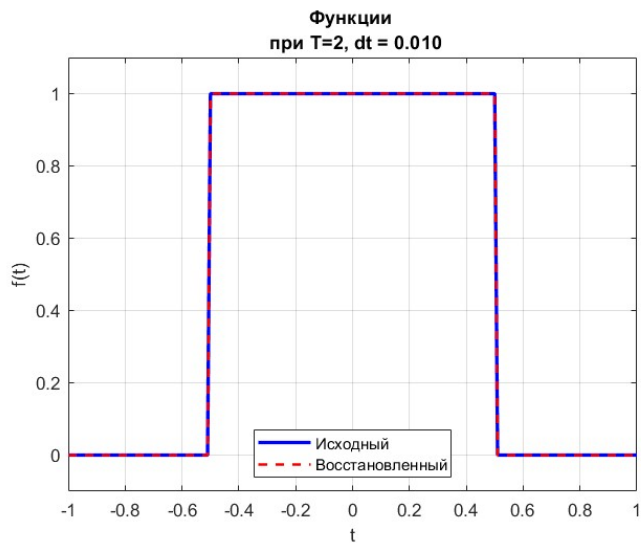


Рис. 27. Исходный и восстановленный сигналы.

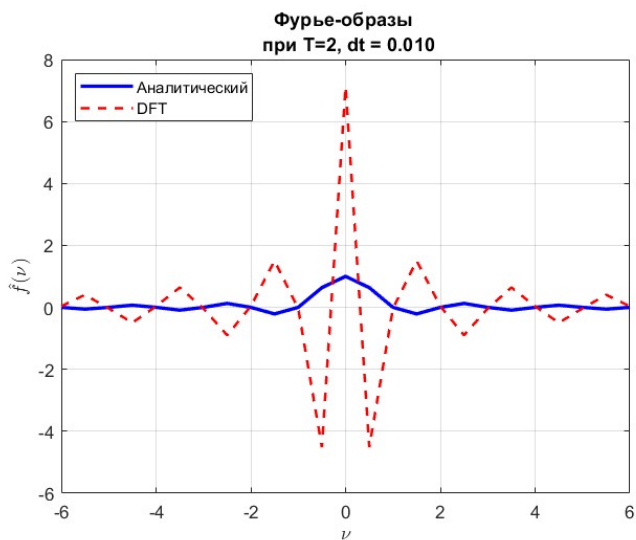


Рис. 28. Фурье-образы: аналитический и полученный при помощи DFT.

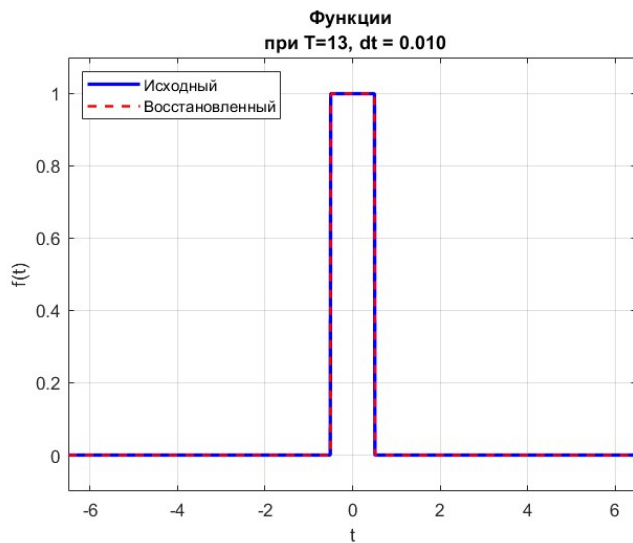


Рис. 29. Исходный и восстановленный сигналы.

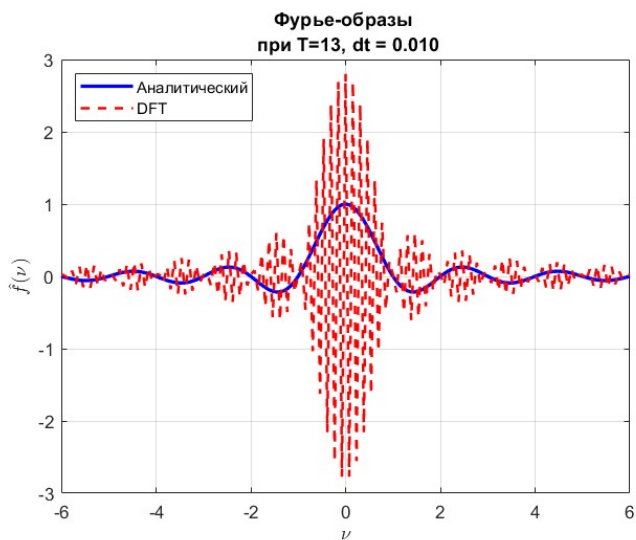


Рис. 30. Фурье-образы: аналитический и полученный при помощи DFT.

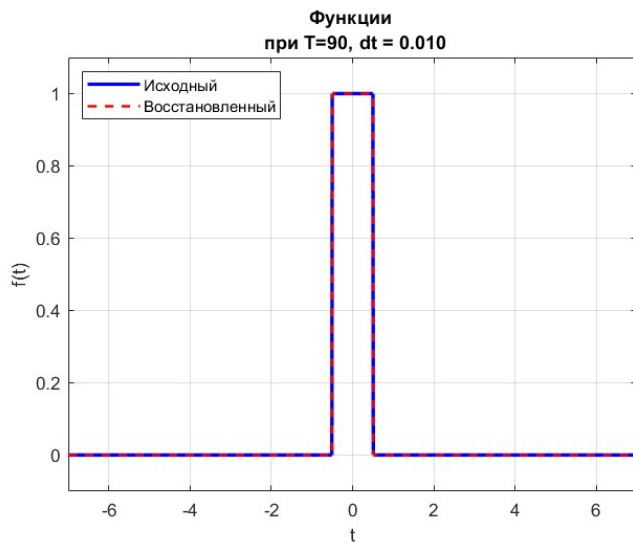


Рис. 31. Исходный и восстановленный сигналы.

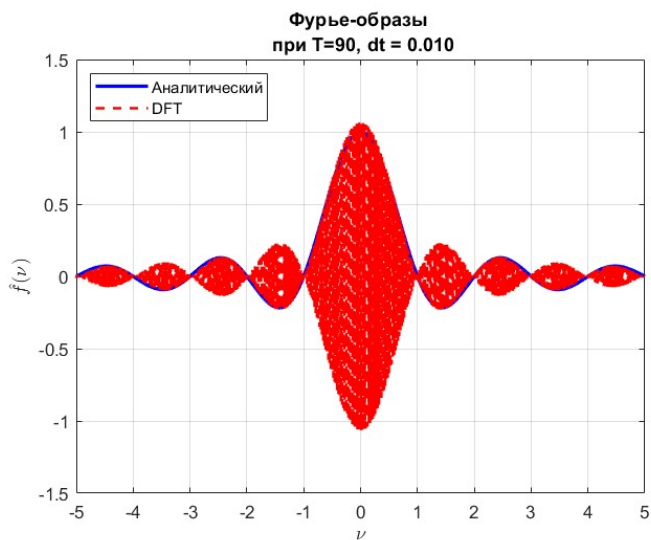


Рис. 32. Фурье-образы: аналитический и полученный при помощи DFT.

Проанализируем приведенные графики. На рисунке 27 исходный сигнал полностью совпадает с восстановленным, искажения отсутствуют. При этом Фурье-образы на рисунке 28 имеют структуру ломаных, плохо совпадают и фактически не похожи на график функции $\text{sinc}(\nu)$.

На рисунке 29 сигнал восстановлен без искажений. Найденный при помощи DFT Фурье-образ напоминает график функции $\text{sinc}(\nu)$, но не совпадает с аналитическим.

На рисунке 31 восстановленный сигнал полностью совпадает с исходным. При этом Фурье-образы на рисунке 32 похожи на график функции $\text{sinc}(\nu)$ и практически совпадают.

Выберем множество значений шага дискретизации Δt :

$$\Delta t = dt = \{0.1, 0.05, 0.001\}. \quad (6)$$

Зафиксируем величину промежутка времени T :

$$T = 12$$

Рассмотрим последовательно все значения параметра Δt из множества (6). С помощью имеющейся программы построим на своих частотах сравнительные графики аналитического образа $\hat{\Pi}(\nu)$ и функций найденных с помощью унитарного $\text{fftshift}(\text{fft}())$. Построим сравнительные графики исходной функции $\Pi(t)$ и восстановленных функций $\hat{\Pi}(t)$ для выбранных значений T и Δt .

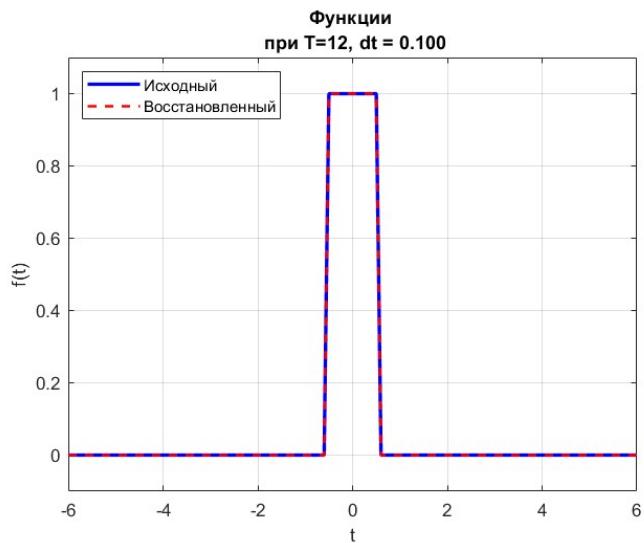


Рис. 33. Исходный и восстановленный сигналы.

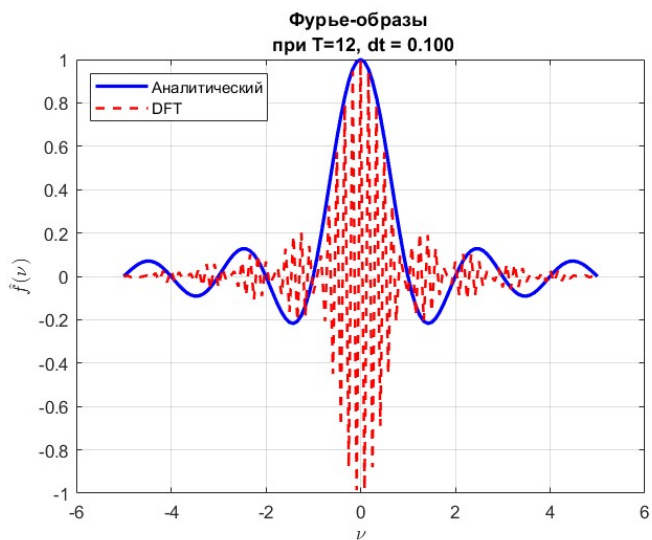


Рис. 34. Фурье-образы: аналитический и полученный при помощи DFT.

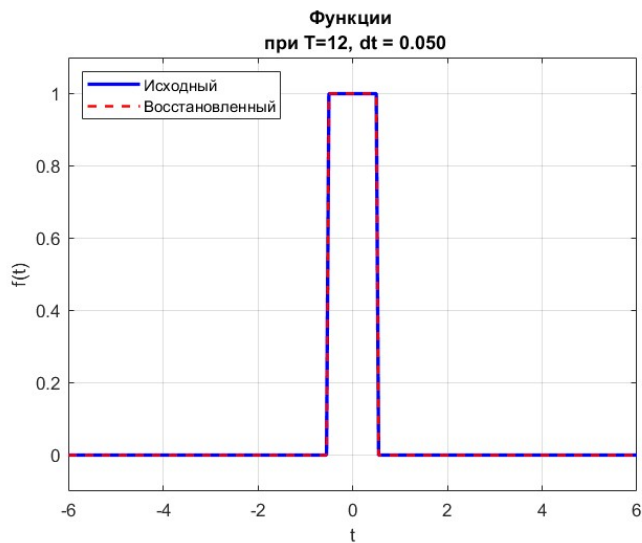


Рис. 35. Исходный и восстановленный сигналы.

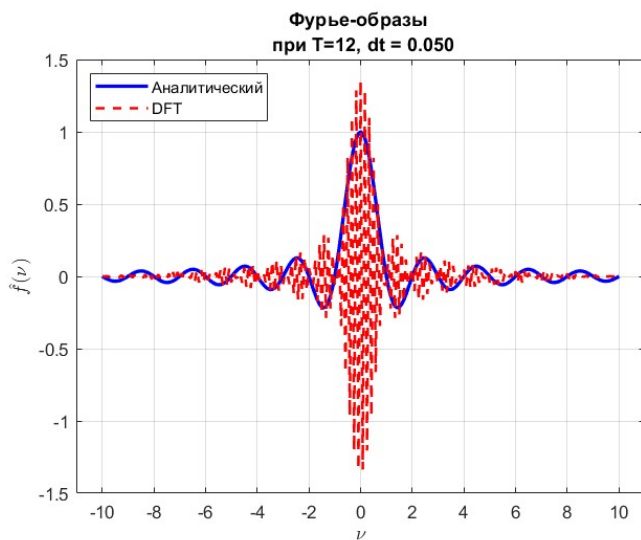


Рис. 36. Фурье-образы: аналитический и полученный при помощи DFT.

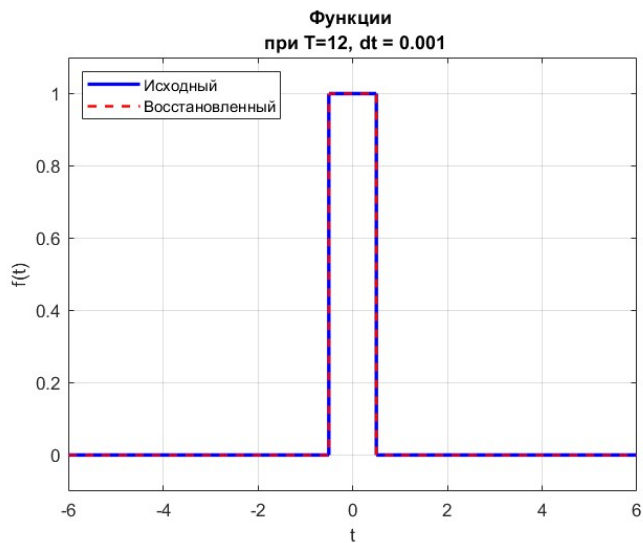


Рис. 37. Исходный и восстановленный сигналы.

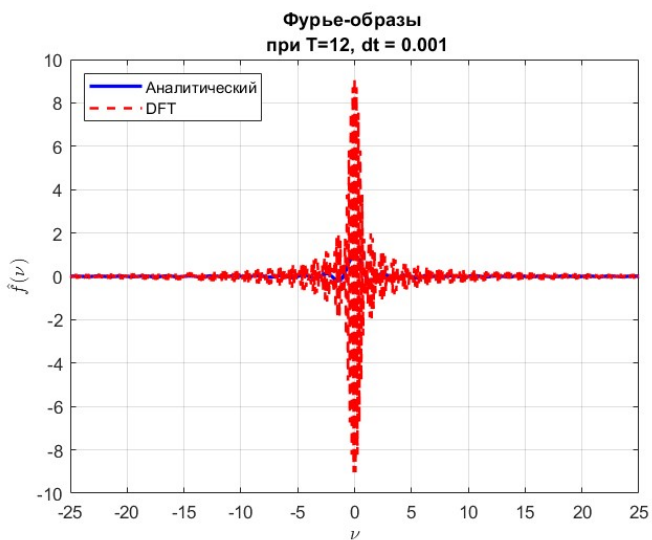


Рис. 38. Фурье-образы: аналитический и полученный при помощи DFT.

Оценим полученные результаты. На рисунке 33 исходный и восстановленный сигналы полностью совпадают. Фурье-образы на рисунке 34 похожи на график функции $\text{sinc}(\nu)$, но кое-где не совпадают из-за наличия искажений.

На рисунке 35 восстановленный сигнал не имеет искажений, он полностью совпадает с исходным. При этом, Фурье-образы на рисунке 36 хоть и стали больше похожи на график функции $\text{sinc}(\nu)$, но все еще не совпадают полностью.

На рисунке 37 искажений в восстановленном сигнале нет, он всюду совпадает с исходным. При этом, Фурье-образы на рисунке 38 преимущественно не совпадают, но имеют отчетливое сходство с графиком функции $\text{sinc}(\nu)$.

Определим, какое влияние оказывает каждый из параметров:

1. T

Влияет на достоверность в частотной области: малые значения параметра приводят к искажениям при моделировании в частотной области, а большие реализуют более точное совпадение с графиком исходного Фурье-образа.

2. Δt

Определяет степень детализации сигнала: большие значения параметра вызывают искажения компонент сигнала, а малое дает возможность получить сравнительно качественный результат моделирования.

Следовательно, промежуток времени T и шаг дискретизации Δt определяют точность, которую удастся достичь при использовании DFT.

Таким образом, использование функции DFT дает возможность получить результат быстро, но без особой точности. Для ее повышения требуется корректно выбрать оптимальные значения T и Δt .

1.3 Мои объяснения.

В процессе выполнения предыдущих пунктов удалось заметить, что функция `trapz` работает долго, а `fft` – быстро. Кроме того, приблизиться к истинному Фурье образу получилось только у функции `trapz`.

Почему так?

[Документация](#) MATLAB позволяет узнать, что функция `trapz` использует трапециевидный метод для интегрирования:

$$\begin{aligned} \int_a^b f(x) dx &\approx \frac{b-a}{2N} \sum_{n=1}^N (f(x_n) + f(x_{n+1})) \\ &= \frac{b-a}{2N} [f(x_1) + 2f(x_2) + \dots + 2f(x_n) + 2f(x_{n+1})], \end{aligned}$$

где интервал между всеми точками равен скалярному значению $\frac{b-a}{2N}$. Именно вычисление значения интеграла при помощи численной схемы, по точкам позволяет приблизиться к кардинальному синусу – истинному Фурье-образу. А значит, такая высокая точность требует много вычислительных ресурсов.

Для функции `fft` [документация](#) также дает информацию о ее реализации:

$$\begin{aligned} Y(k) &= \sum_{j=1}^n X(j) W_n^{(j-1)(k-1)} \\ X(j) &= \frac{1}{n} \sum_{k=1}^n Y(k) W_n^{-(j-1)(k-1)}, \end{aligned} \tag{7}$$

где $Y = \text{fft}(X)$, $X = \text{ifft}(Y)$ длины n , а $W_n = e^{-\frac{2\pi i}{n}}$ один из корней степени n из единицы. Фурье-образы, которые были найдены при помощи данной функции, далеки от истинного. Однако при этом унитарность преобразования при использовании `ifft` позволяет избавиться от искажений и колебаний в восстановленном сигнале.

Почему обратное преобразование в одном из случаев работает лучше?

Такое поведение `ifft` объясняется [унитарностью](#) – ключевым свойством [матрицы DFT](#). Кроме того, соотношения (7) говорят о четкой противоположности преобразований. Следовательно, при восстановлении сигнала удастся свести искажения к минимуму.

1.4 Приближение непрерывного с помощью DFT.

Давайте исправим ситуацию и попробуем совместить достоинства обоих подходов: точность и быстродействие. Найдем способ получить правильный Фурье-образ, соответствующий непрерывному преобразованию Фурье, используя функцию `fft` и не прибегая к численному интегрированию. Найдем способ восстановить исходный сигнал по полученному Фурье-образу – тоже с помощью `fft`. Представим схему желаемого успеха:

$$\Pi(t) \xrightarrow{\text{умное использование fft}} \hat{\Pi}(\nu) \xrightarrow{\text{умное использование ifft}} \Pi(t)$$

Воспользуемся предложенными идеями. Отметим, что любой интеграл по конечному промежутку можно аппроксимировать с помощью суммы Римана:

$$\int_a^b f(t)dt \approx \sum_{n=0}^{N-1} f(t_n)\Delta t, \quad \text{где} \quad \Delta t = \frac{b-a}{N-1}$$

Тогда имеем следующее:

$$\int_{\frac{T}{2}}^{-\frac{T}{2}} f(t)e^{-2\pi i\nu t} dt \approx \sum_{n=0}^{N-1} f(t_n)e^{-2\pi i\nu_m t_n} \Delta t, \quad \Delta t = \frac{\frac{T}{2} - (-\frac{T}{2})}{N-1} = \frac{T}{N-1} \quad (8)$$

где ν_m – точки, взятые в частотной области для аппроксимации.

Запишем, как задается преобразование Фурье найденное на конечном промежутке T через сумму Римана, умноженную на некую константу c_m :

$$\hat{f}(\nu_m) \approx c_m \sum_{n=0}^{N-1} f(t_n)e^{-2\pi i \frac{nm}{N}}, \quad (9)$$

Сопоставим формулы (8) и (9) и получим вид искомых коэффициентов:

$$c_m = \Delta t \cdot e^{-2\pi i a \nu_m}$$

Следовательно, DFT аппроксимирует интеграл как сумму Римана. А значит, можно определить коэффициенты, используемые для обратного преобразования:

$$c_m^{-1} = \frac{1}{\Delta t} \cdot e^{2\pi i a \nu_m}$$

Тогда программно преобразование Фурье при известных c_m можно задать через `fft` как:

$$\mathcal{F}\{f\} \rightarrow \text{fftshift}(c.*\text{fft}(f))$$

Тогда обратное преобразование Фурье можно найти как:

$$\mathcal{F}^{-1}\{f\} \rightarrow \text{ifft}(\text{ifftshift}(f_hat)./c)$$

Таким образом, предложенный вид коэффициентов c_m определяет новый метод, который не просто вычисляет сумму, а приводит ее в соответствие с интегралом.

Отметим, что коэффициент c_m выполняет две ключевые функции:

1. Масштабирование амплитуды.

Умножение на Δt превращает сумму в приближение интеграла.

2. Сохранение размерности.

Умножение позволяет реализовать правильный физический смысл при работе в частотной области.

Проведем исследования нового метода и выберем наиболее показательные значения T (промежуток интегрирования по времени) и Δt (шаг дискретизации), а также найдем соответствующие им $V, \Delta\nu$.

Реализуем [программу](#) для выполнения данного задания при помощи средств пакета MATLAB.

Выберем множество значений промежутка времени T :

$$T = \{4, 10, 40\}. \quad (10)$$

Зафиксируем величину шага дискретизации Δt :

$$\Delta t = dt = 0.01$$

Рассмотрим последовательно все значения параметра T из множества (10). Построим на своих частотах сравнительные графики аналитического образа $\hat{\Pi}(\nu)$ и функций найденных с помощью разработанного алгоритма. Построим сравнительные графики исходной функций $\Pi(t)$ и восстановленных функций $\Pi(t)$ для выбранных значений T и Δt .

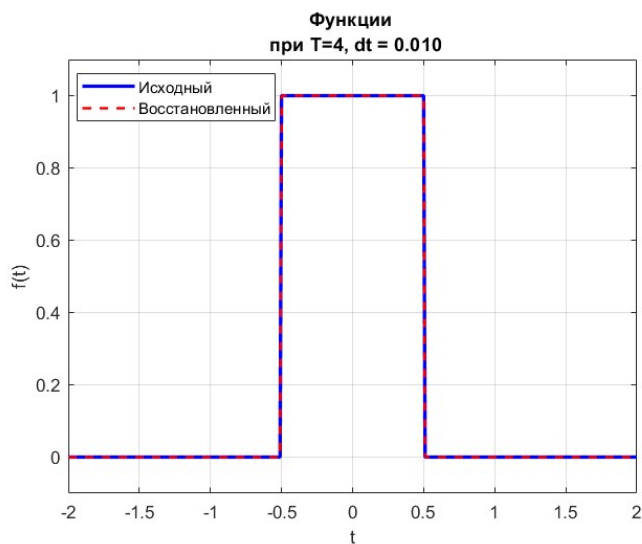


Рис. 39. Исходный и восстановленный сигналы.

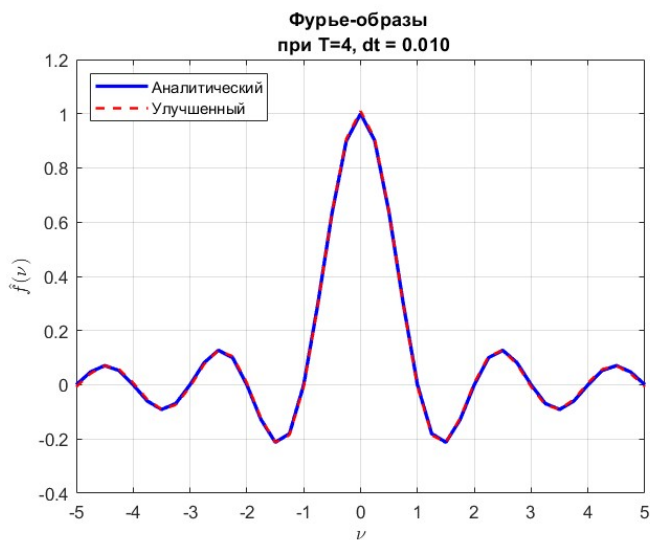


Рис. 40. Фурье-образы: аналитический и полученный при помощи разработанного алгоритма.

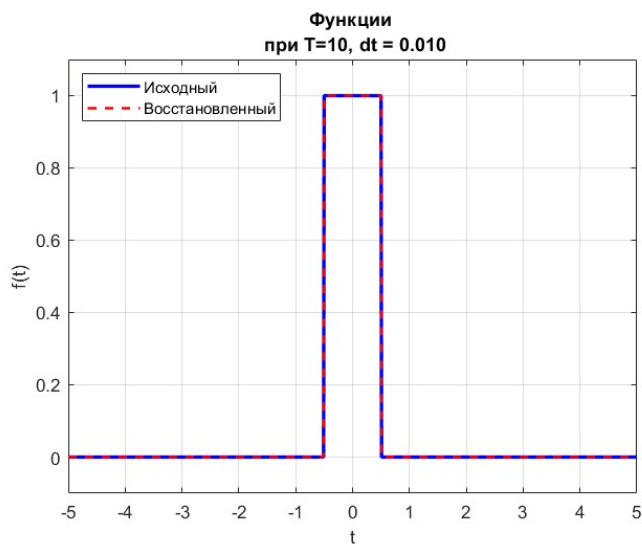


Рис. 41. Исходный и восстановленный сигналы.

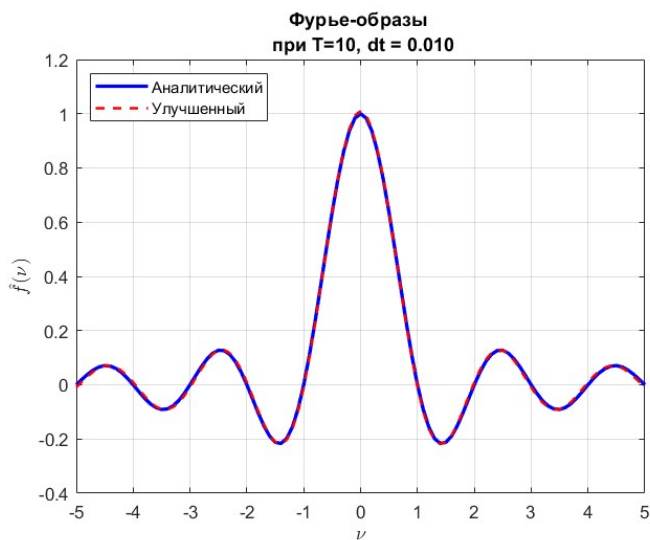


Рис. 42. Фурье-образы: аналитический и полученный при помощи разработанного алгоритма.

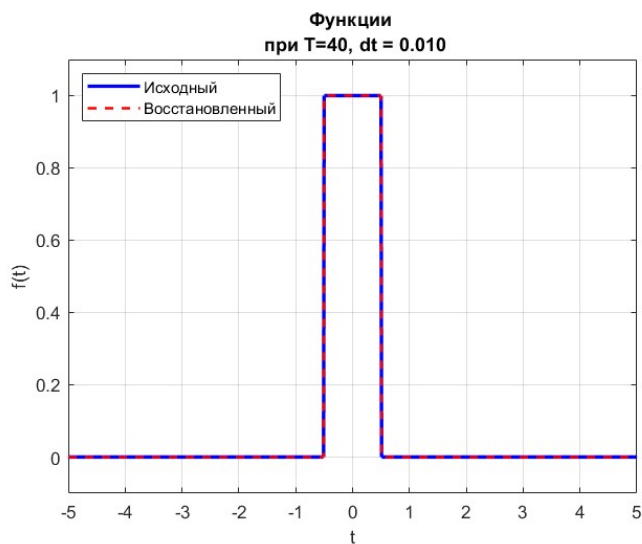


Рис. 43. Исходный и восстановленный сигналы.

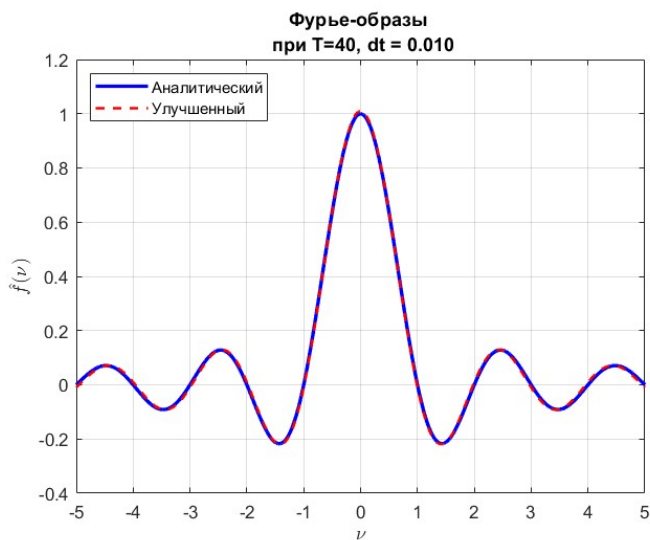


Рис. 44. Фурье-образы: аналитический и полученный при помощи разработанного алгоритма.

Проанализируем полученные результаты. На рисунке 43 исходный и восстановленный сигнал полностью совпадают, искажения отсутствуют. На рисунке 40 Фурье-образы совпадают, имеют вид графика функции $\text{sinc}(\nu)$, но видна явная дискретизация: заметны участки ломаных.

На рисунке 41 восстановленный сигнал не имеет искажений и полностью совпадает с исходным. На рисунке 42 Фурье-образы совпадают и представляют собой график функции $\text{sinc}(\nu)$ без видимой дискретизации.

На рисунке 43 восстановленный сигнал не искажен и полностью совпадает с исходным. Фурье-образы на рисунке 44 совпадают между собой и являются графиком функции $\text{sinc}(\nu)$.

Таким образом, величина промежутка времени T при умном использовании `fft` влияет на возможность приблизиться к аналитическому Фурье-образу: с увеличением значения при должном шаге дискретизации это удастся сделать.

Определим множество значений шага дискретизации Δt :

$$\Delta t = dt = \{0.1, 0.01, 0.001\}. \quad (11)$$

Зафиксируем величину промежутка времени T :

$$T = 12$$

Рассмотрим последовательно все значения параметра T из множества (11). Построим на своих частотах сравнительные графики аналитического образа $\hat{\Pi}(\nu)$ и функций найденных с помощью разработанного алгоритма. Построим сравнительные графики исходной функций $\Pi(t)$ и восстановленных функций $\Pi(t)$ для выбранных значений T и Δt .

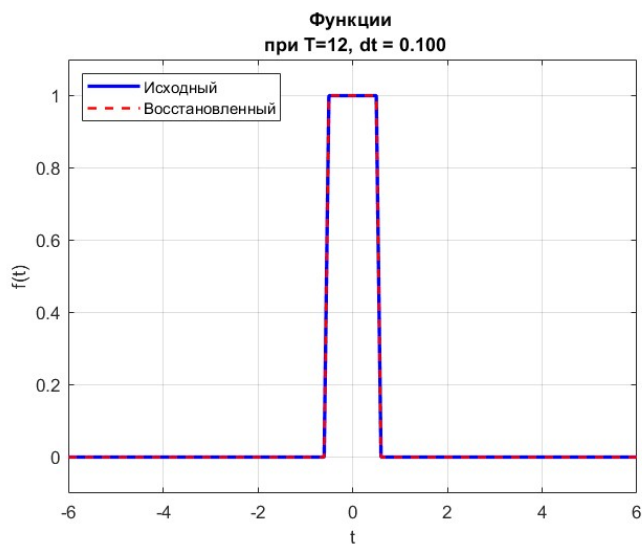


Рис. 45. Исходный и восстановленный сигналы.

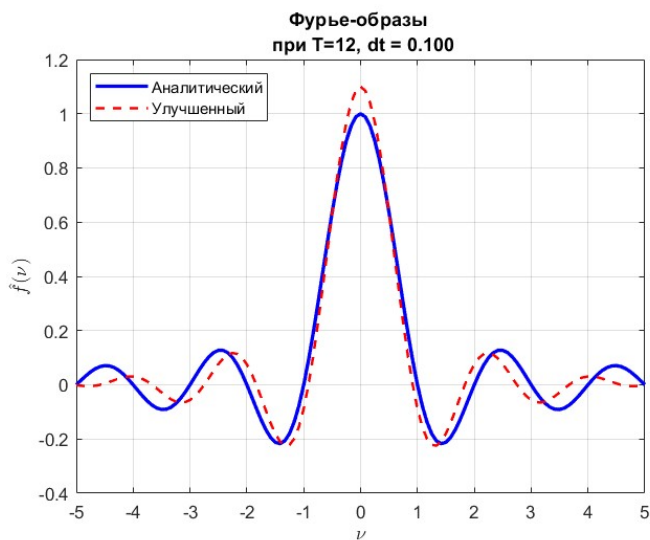


Рис. 46. Фурье-образы: аналитический и полученный при помощи разработанного алгоритма.

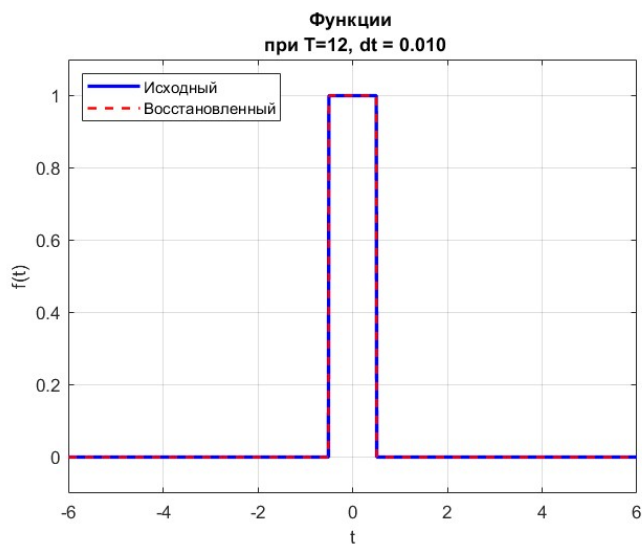


Рис. 47. Исходный и восстановленный сигналы.

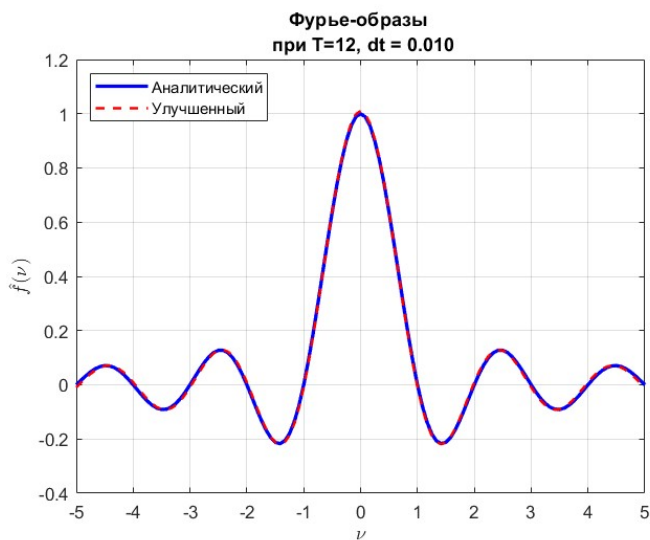


Рис. 48. Фурье-образы: аналитический и полученный при помощи разработанного алгоритма.

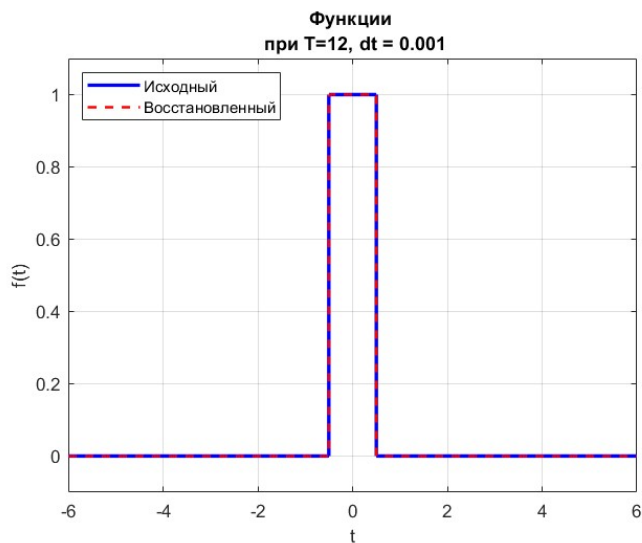


Рис. 49. Исходный и восстановленный сигналы.

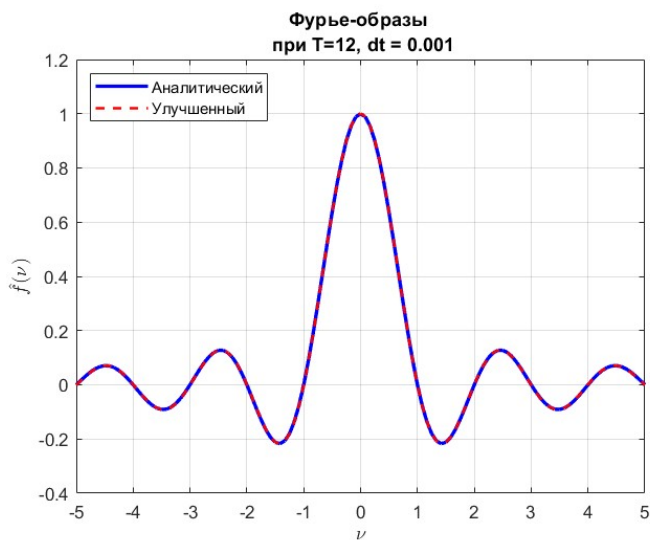


Рис. 50. Фурье-образы: аналитический и полученный при помощи разработанного алгоритма.

Охарактеризуем приведенные графики. На рисунке 45 сигнал восстановлен без искажений. На рисунке 46 Фурье-образы не совпадают: большой шаг привел к появлению искажений.

На рисунке 47 восстановленный сигнал полностью совпадает с исходным. Фурье-образы на рисунке 48 полностью совпадают, дискретизация не заметна, искажений нет.

На рисунке 49 исходный и восстановленный сигналы полностью совпадают. Фурье-образы на рисунке 50 полностью совпадают и представляют собой график функции $\text{sinc}(\nu)$.

Следовательно, шаг дискретизации Δt определяет точность, с которой возможно приблизиться к аналитическому Фурье-образу.

Для более наглядного сравнения выберем 2 пары значений параметров T и Δt и построим сравнительные графики Фурье-образов, которые получены при помощи разработанного метода и алгоритмов из двух предыдущих пунктов:

$$T = 14 \quad \Delta t = 0.1$$

$$T = 6 \quad \Delta t = 0.01$$

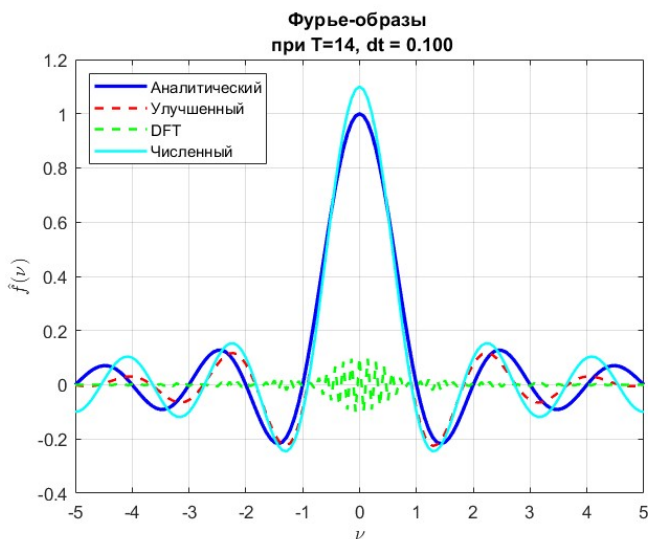


Рис. 51. Сравнение работы методов.

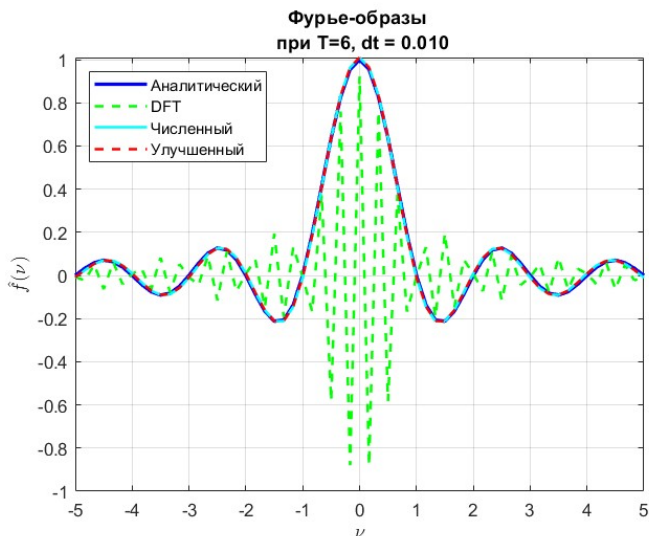


Рис. 52. Сравнение работы методов.

Сравним приведенные графики. На рисунке 51 видно, что улучшенный алгоритм точнее всех остальных приблизил аналитический Фурье-образ: у численного метода есть погрешности, а DFT слишком не точен.

На рисунке 52 снова улучшенный алгоритм справился лучше всех: у численного метода заметны погрешности, а DFT имеет неточности.

Отметим, когда новый метод выигрывает, а когда нет. Выполнение данного задания, позволило понять, что разработанный алгоритм подходит для обработки данных, которая выполняется быстрее, чем при использовании `trapz`, для задач, где важны и точность, и скорость.

Однако есть ситуации, в которых новый метод проигрывает. Например, обработка непериодических сигналов, сигналов с грубой дискретизацией.

Сформулируем выводы о влиянии величины шага и размера промежутка интегрирования.

1. Δt

Определяет степень детализации: с увеличением значения приближение к аналитическому Фурье-образу становится точнее.

2. T

Регулирует искажения: при увеличении значения удастся лучше приблизиться к аналитическому Фурье-образу.

Таким образом, параметры метода определяют его эффективность и границы применимости.

Следовательно, правильная настройка параметров улучшенного метода позволяет быстро получить результат со сравнительно высокой точностью и оптимальным использованием вычислительных ресурсов.

1.5 Выводы.

Сформулируем общие наблюдения по результатам выполнения данного задания.

Функция `trapz` позволяет детальнее исследовать характеристики сигнала, но требует использования большого количества вычислительных ресурсов.

Быстродействие `fft` вынуждает отказаться от точности вычислений: метод применим в задачах, где допустимы погрешности при быстром получении результата.

Улучшенный метод позволяет совместить достоинства обоих подходов: реализовать высокую точность с сохранением быстродействия.

Для эффективного моделирования важно выбирать метод, который подходит под данную задачу. Это позволит получить корректные результаты с высоким уровнем достоверности.

Разные функции способны обрабатывать различные виды сигналов, что позволяет успешно проводить их анализ, а также применять на практике.

2 Задание 2. Сэмплирование.

Зададимся параметрами:

$$\begin{aligned} a_1 &= 2 & a_2 &= 1 \\ \omega_1 &= 5 & \omega_2 &= 20\pi \\ \varphi_1 &= \frac{\pi}{6} & \varphi_2 &= \frac{\pi}{4} \\ b &= 6\pi \end{aligned}$$

и рассмотрим функции:

$$\begin{aligned} y_1(t) &= a_1 \cdot \sin(\omega_1 t + \varphi_1) + a_2 \cdot \sin(\omega_2 t + \varphi_2) \\ y_2(t) &= \text{sinc}(bt) \end{aligned}$$

Запишем конкретные функции, которые используются в данном задании:

$$\begin{aligned} y_1(t) &= 2 \cdot \sin\left(5t + \frac{\pi}{6}\right) + 1 \cdot \sin\left(20\pi t + \frac{\pi}{4}\right) \\ y_2(t) &= \text{sinc}(6\pi t) \end{aligned}$$

Отсюда найдем значения B – максимальную частоту в сигнале для каждой из функций:

$$\begin{aligned} B_1 &= \frac{20\pi}{2\pi} = 10 \text{ Гц} \\ B_2 &= \frac{6\pi}{2\pi} = 3 \text{ Гц} \end{aligned}$$

Следовательно, по [теореме](#) Найквиста-Шеннона-Котельникова можно найти значения шага дискретизации Δt для каждой функции, при которых функцию можно будет восстановить:

$$\begin{aligned} y_1 : \quad \Delta t &< \frac{1}{2B_1} = 0.05 \\ y_2 : \quad \Delta t &< \frac{1}{2B_2} = 0.16 \end{aligned} \tag{12}$$

Таким образом, для успешного восстановления функции y_1 после сэмплирования требуется $\Delta t < 0.05$, а для восстановления функции y_2 необходимо $\Delta t < 0.16$.

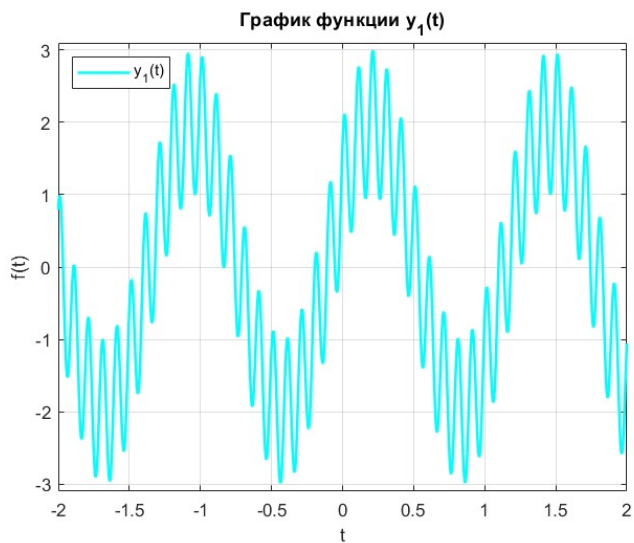


Рис. 53. График функции y_1 .

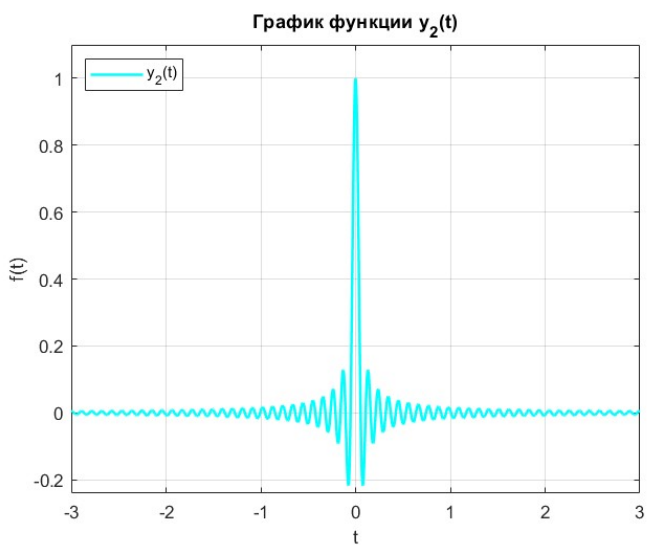


Рис. 54. График функции y_2 .

Приведем графики исходных функций без видимой дискретизации на рисунках 53 и 54.

Отметим, что величина промежутка интегрирования по времени T влияет на искажения в сигнале. Иначе говоря, с помощью этого параметра можно регулировать потери информации и достоверность сигнала. Для удобства выберем оптимальное значение, которое позволит получить убедительные результаты.

Теперь зададим сэмплированный вариант исходных функций: рассмотрим разреженный вариант массива времени и соответствующий ему массив значений. Построим дискретный график поверх непрерывного.

Приведем интерполяционную формулу, которую применим к сэмплированным данным:

$$f(t) = \sum_{n=-\infty}^{\infty} f(t_n) \cdot \text{sinc}(2B(t - t_n)), \quad t_n = \frac{n}{2B}$$

Затем построим график восстановленной функции поверх исходной.

Реализуем [программу](#) для выполнения данного задания с помощью средств пакета MATLAB.

Приведем графики сэмплированной и восстановленной функций.



Рис. 55. Исходный и сэмплированный сигналы y_1 при $\Delta t = 0.1$.

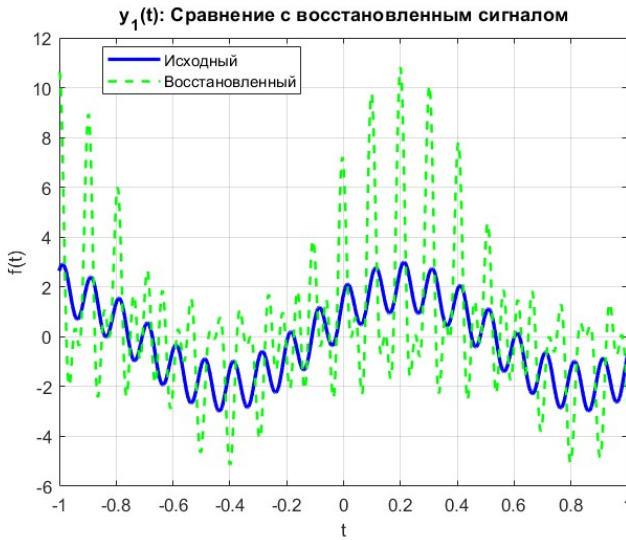


Рис. 56. График исходной и восстановленной функции y_1 при $\Delta t = 0.1$.

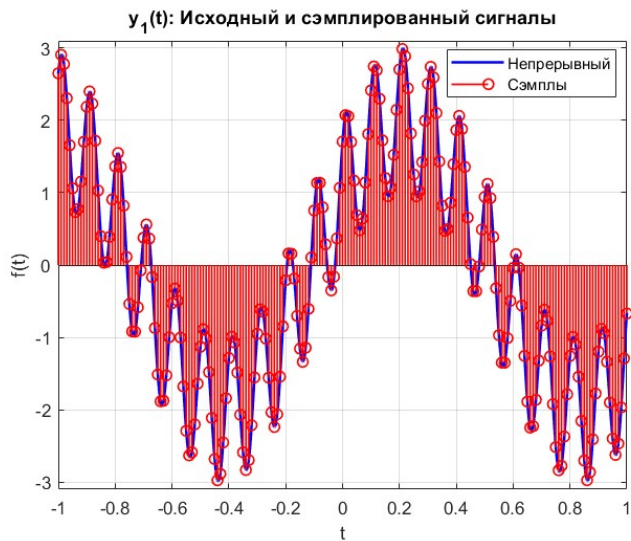


Рис. 57. Исходный и сэмплированный сигналы y_1 при $\Delta t = 0.01$.

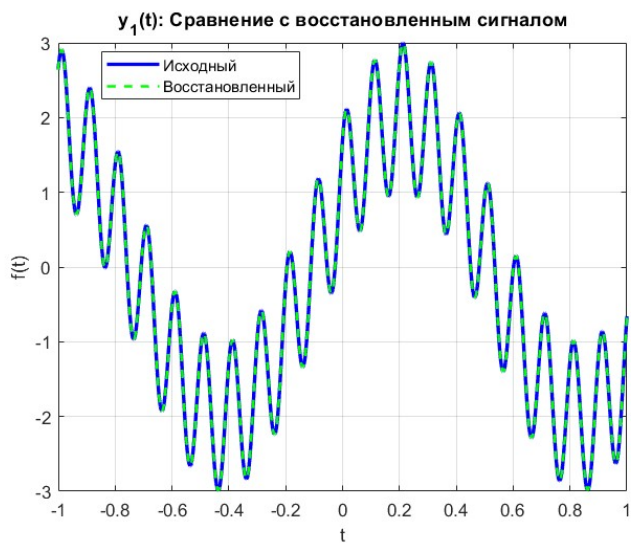


Рис. 58. График исходной и восстановленной функции y_1 при $\Delta t = 0.01$.

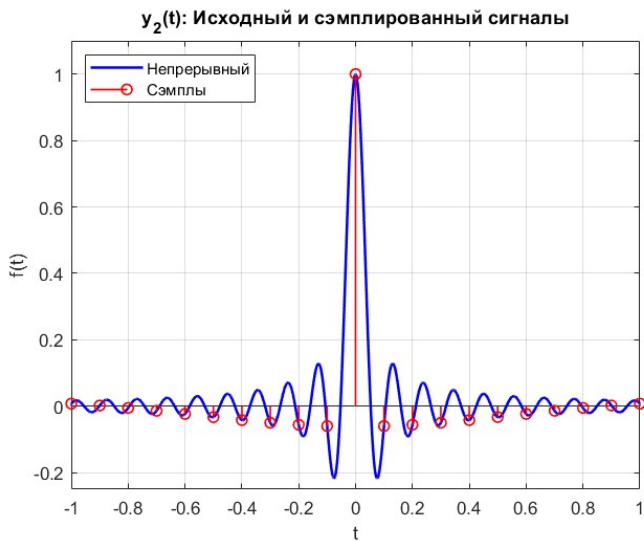


Рис. 59. Исходный и сэмплированный сигналы y_2 при $\Delta t = 0.1$.

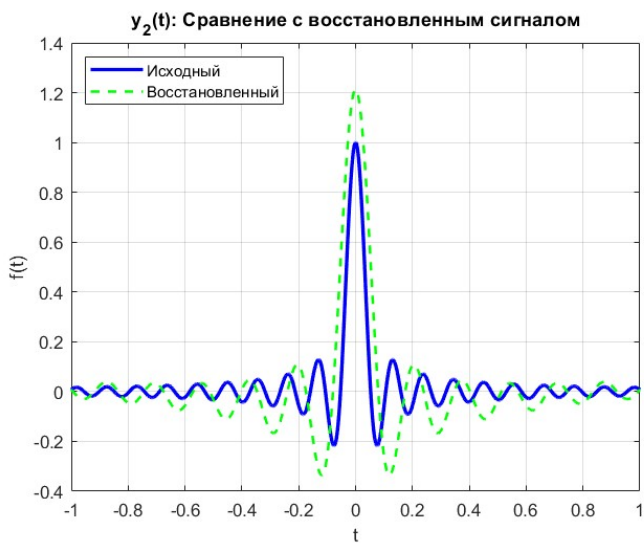


Рис. 60. График исходной и восстановленной функции y_2 при $\Delta t = 0.1$.

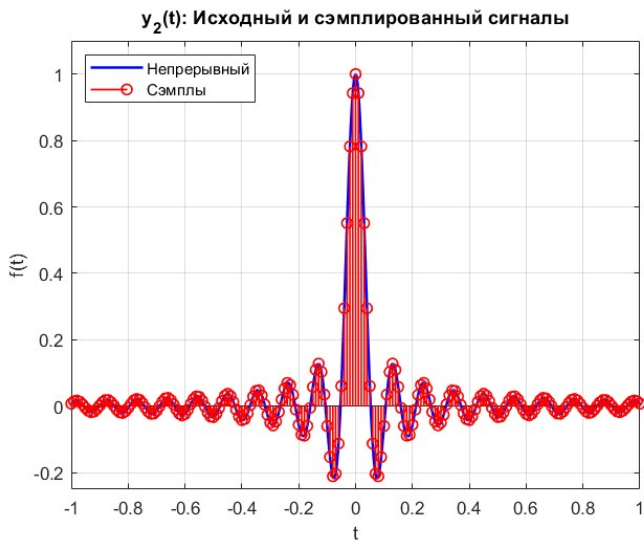


Рис. 61. Исходный и сэмплированный сигналы y_2 при $\Delta t = 0.01$.

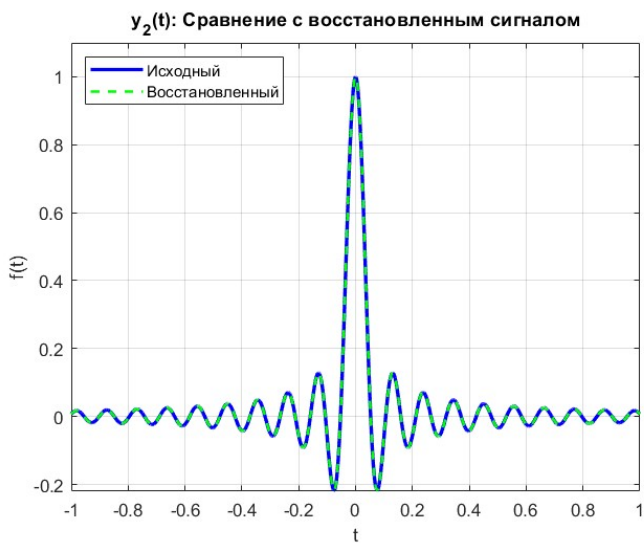


Рис. 62. График исходной и восстановленной функции y_2 при $\Delta t = 0.01$.

Проанализируем полученные результаты. При грубой дискретизации на рисунке 55 остается совсем мало информации о сигнале. Именно поэтому на рисунке 56 видно, что сигнал не удалось восстановить.

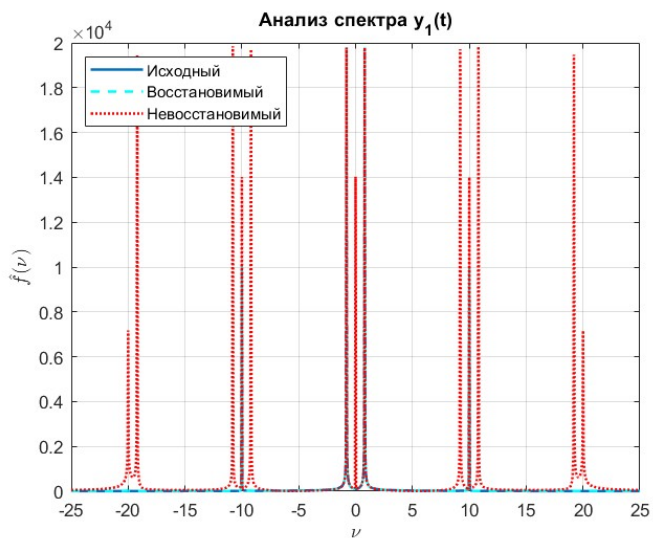
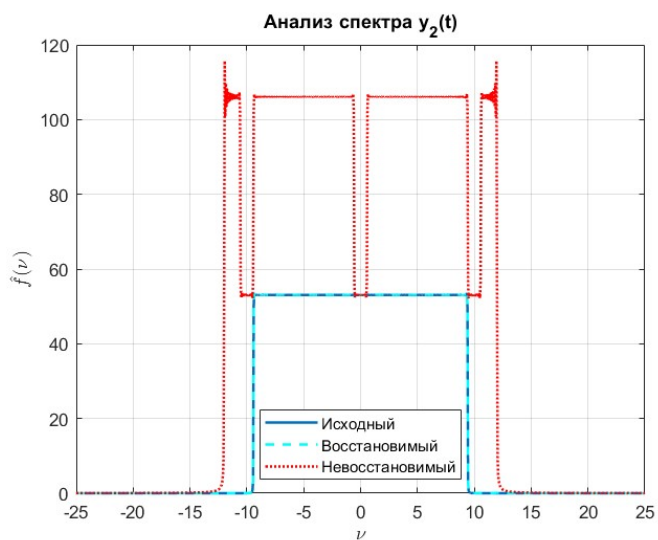
Однако при качественной дискретизации на рисунке 57, которая выполнена в соответствии с теоретическими расчетами (12) удастся сохранить достаточное для восстановления количество информации о сигнале. На рисунке 58 видно успешное восстановление сигнала, что соответствует теореме.

На рисунке 59 использовано достаточно мало точек для дискретизации, очень много информации о сигнале упущено. Из-за этого на рисунке 60 сигналы не совпадают – восстановление не удалось выполнить.

На рисунке 61 при дискретизации, основанной на расчетах (12), остается требуемое количество информации о сигнале. То есть, на рисунке 62 сигналы совпадают, а значит, восстановление было выполнено успешно.

Таким образом, все полученные результаты согласуются с теоремой Найквиста-Шеннона-Котельникова. Ее использование в процессе сэмплирования и восстановления сигнала позволяет получить достоверный результат при оптимальных вычислительных затратах.

Для наглядности анализа приведем Фурье-образы исходных функций.

Рис. 63. График функции y_1 .Рис. 64. График функции y_2 .

На рисунках 63 и 64 видно, что при качественной дискретизации Фурье-образы восстанавливаемого сигнала – того который можно восстановить в соответствии с теоремой Найквиста-Шеннона-Котельникова – полностью совпадают с Фурье-образами исходных функций. Соответственно, сигнал, который не удастся восстановить из-за грубой дискретизации – невозможный, имеет искажения и его Фурье-образ практически не совпадает с Фурье-образом исходной функции.

2.1 Выводы.

Сформулируем общие наблюдения на основе результатов выполнения данного задания.

Теорема Найквиста-Шеннона-Котельникова дает исчерпывающий ответ на вопрос восстановления сигнала при его сэмплировании. Выбор степени дискретизации на ее основании позволяет эффективно и удобно работать с различными сигналами.

Кроме того, дискретизация, основанная на рассмотренной теореме, позволяет избежать алиасинга – эффекта наложения спектров сигнала. Благодаря ей удастся выполнить восстановление без искажений.

Теорема Найквиста-Шеннона-Котельникова находит широкое практическое применение при работе с различными сигналами, информацию и качество которых необходимо сохранить на требуемом уровне.

3 Выводы по лабораторной работе.

Сформулируем общие выводы, которые можно сделать на основании результатов выполнения лабораторной работы:

1. Численное интегрирование с использованием функции *trapz* обеспечивает высокую точность вычисления Фурье-образов, однако требует значительных вычислительных ресурсов, что ограничивает ее применение в задачах, требующих быстрой обработки данных.
2. Дискретное преобразование Фурье DFT демонстрирует высокое быстродействие, но уступает в точности при аппроксимации непрерывных сигналов, особенно при недостаточно мелком шаге дискретизации Δt .
3. Разработанный улучшенный метод, сочетающий масштабирование и коррекцию фазовых коэффициентов, позволяет приблизить результаты DFT к аналитическому решению, сохраняя при этом вычислительную эффективность.
4. Теорема Найквиста-Шеннона-Котельникова подтверждает свою практическую значимость: выбор шага дискретизации $\Delta t < \frac{1}{2B}$ гарантирует возможность точного восстановления сигнала без наложения спектров.
5. Параметр T (промежуток интегрирования по времени) существенно влияет на точность моделирования: увеличение T снижает искажения в частотной области, но повышает количество требуемых вычислительных ресурсов.
6. Шаг дискретизации Δt играет ключевую роль в детализации сигнала: слишком большой шаг приводит к потере информации, а слишком мелкий – к неоправданному росту вычислительной нагрузки без значительного улучшения точности.

7. Восстановление сигнала через обратное DFT демонстрирует устойчивость к искажениям даже при наличии неточностей в частотной области благодаря унитарности преобразования.
8. Анализ спектров сигналов $y_1(t)$ и $y_2(t)$ подтвердил, что соблюдение условия Найквиста обеспечивает совпадение Фурье-образов восстановленного и исходного сигналов, а его нарушение приводит к существенным искажениям.
9. Практические эксперименты подчеркивают необходимость компромисса между точностью и быстродействием: выбор используемого метода обработки должен определяться требованиями конкретной задачи.
10. Корректный выбор параметров T , Δt , V и $\Delta\nu$ позволяет достичь баланса между вычислительной эффективностью и достоверностью результатов, что критически важно в прикладных задачах цифровой обработки сигналов.

Приложение

```
1 % Параметры временной и частотной осей
2 T = 12; % Интервал времени
3 dt = 0.001; % Маленький шаг дискретизации
4 t = -T/2 : dt : T/2; % Набор временных шагов
5 V = 1/dt; % Ширина диапазона частот
6 dv = 1/T; % Шаг частоты
7 v = -V/2 : dv : V/2; % Набор частот
8
9 g = zeros(size(t)); % Инициализация нулями
10 g(abs(t) <= 0.5) = 1; % Значения функции g(t)
11
12 % Получение истинного Фурье-образа исходной функции
13 image = sinc(v);
14
15 % Построение графика во временной области
16 figure;
17 plot(t, g, 'LineWidth', 2, 'Color','green');
18 legend('\Pi(t)');
19 grid on;
20 xlabel('t');
21 ylabel('f(t)');
22 title('График исходной функции');
23 xlim([-T/2, T/2]);
24 ylim([-0.1, 1.1]);
25
26 % Построение графика в частотной области
27 figure;
28 plot(v, image, 'LineWidth', 2, 'Color','b');
29 legend('$$\hat{\Pi}(\nu)$$', 'Interpreter','latex');
30 grid on;
31 xlabel('$$\nu$$', 'Interpreter','latex');
32 ylabel('$$\hat{f}(\nu)$$', 'Interpreter','latex');
33 title('График Фурье-образа');
34 xlim([-30, 30]);
35 ylim([-0.25, 1.1]);
```

Листинг 1. Программа, используемая для выполнения предварительного задания.

```

1 T = 5; % Интервал времени
2 dt = 0.001; % Шаг дискретизации
3 t = -T/2 : dt : T/2; % Набор временных шагов
4 V = 12; % Ширина диапазона частот
5 dv = 0.01; % Шаг частоты
6 v = -V/2 : dv : V/2; % Набор частот
7 str_1 = sprintf('Функции \n при T=%d, dt = %.3f, V = %d, dv = %.2f',
8               , T, dt, V, dv);
9 str_2 = sprintf('Фурье-образы \n при T=%d, dt = %.3f, V = %d, dv =
10               %.2f', T, dt, V, dv);
11 g = zeros(size(t)); % Инициализация нулями
12 g(abs(t) <= 0.5) = 1; % Значения функции g(t)
13
14 % Численное преобразование Фурье (прямое)
15 g_hat = zeros(size(v));
16
17 % Получение истинного Фурье-образа исходной функции
18 image = sinc(v);
19
20 % Вычисление интеграла для каждой частоты
21 for k = 1:length(v)
22     integrand = g .* exp(-1i*2*pi*v(k)*t);
23     g_hat(k) = trapz(t, integrand); % Численное интегрирование
24 end
25
26 % Визуализация Фурье-образа
27 figure;
28 plot(v, image, 'LineWidth', 2, 'Color','b'); hold on;
29 plot(v, real(g_hat), 'LineWidth', 1.5);
30 xlabel('$$\nu$$', 'Interpreter','latex');
31 ylabel('$$\hat{f}(\nu)$$', 'Interpreter','latex');
32 legend('Исходный', 'Полученный численно', 'Location', 'northwest');
33 title(str_2);
34 xlim([-V/2 - 1, V/2 + 1]);
35 ylim([-0.25, 1.1]);
36 grid on;
37
38 % Обратное преобразование Фурье
39 g_recovered = zeros(size(t));
40
41
42
43
44
45
46
47
48

```

```
49 % Вычисление интеграла для каждого момента времени
50 for m = 1:length(t)
51     integrand = g_hat .* exp(1i*2*pi*v*t(m));
52     g_recovered(m) = trapz(v, integrand); % Численное интегрирование
53 end
54
55 g_recovered = real(g_recovered); % Восстановление исходного сигнала
56
57 % Сравнение исходного и восстановленного сигналов
58 figure;
59 plot(t, g, 'g', 'LineWidth', 2); hold on;
60 plot(t, g_recovered, 'r--', 'LineWidth', 1.5);
61 xlabel('t');
62 ylabel('f(t)');
63 legend('Исходный', 'Восстановленный', 'Location', 'northwest');
64 title(str_1);
65 xlim([-T/2, T/2]);
66 ylim([-0.1, 1.2]);
67 grid on;
```

Листинг 2. Программа, используемая для выполнения задания 1.1.

```

1 % Параметры временной и частотной осей
2 T = 10; % Интервал времени
3 dt = 0.01; % Шаг дискретизации
4 t = -T/2 : dt : T/2; % Набор временных шагов
5 V = 1/dt; % Ширина диапазона частот
6 dv = 1/T; % Шаг частоты
7 v = -V/2 : dv : V/2; % Набор частот
8 N = length(t); % Количество точек, используемых для дискретизации
9 str_1 = sprintf('Функции \n при T=%d, dt = %.3f', T, dt);
10 str_2 = sprintf('Фурье-образы \n при T=%d, dt = %.3f', T, dt);
11
12 g = zeros(size(t)); % Инициализация нулями
13 g(abs(t) <= 0.5) = 1; % Значения функции g(t)
14
15 % Применение унитарного DFT
16 g_hat_fft = fftshift(fft(g)) / sqrt(N);
17
18 % Аналитический Фурье-образ
19 g_hat_analytical = sinc(v);
20
21 % Обратное преобразование
22 g_recovered = ifft(ifftshift(g_hat_fft)) * sqrt(N);
23
24 % Построение графиков
25 figure;
26 plot(v, g_hat_analytical, 'b', 'LineWidth', 2, 'DisplayName', 'Анал
    итический');
27 hold on;
28 plot(v, real(g_hat_fft), 'r--', 'LineWidth', 1.5, 'DisplayName', '
    DFT');
29 xlabel('$$\nu$$', 'Interpreter','latex');
30 ylabel('$$\hat{f}(\nu)$$', 'Interpreter','latex');
31 title(str_2);
32 legend('Location','best');
33 grid on;
34 xlim([-6, 6]);
35
36 figure;
37 plot(t, g, 'b', 'LineWidth', 2);
38 hold on;
39 plot(t, real(g_recovered), 'r--', 'LineWidth', 1.5);
40 xlabel('t');
41 ylabel('f(t)');
42 title(str_1);
43 legend('Исходный', 'Восстановленный', 'Location', 'best');
44 grid on;
45 xlim([-T/2, T/2]);
46 ylim([-0.1, 1.1]);

```

Листинг 3. Программа, используемая для применения DFT.

```

1 % Инициализация параметров
2 T = 12; % Интервал времени
3 dt = 0.01; % Шаг дискретизации
4 t = -T/2 : dt : T/2; % Набор временных шагов
5 V = 1/dt; % Ширина диапазона частот
6 dv = 1/T; % Шаг частоты
7 v = -V/2 : dv : V/2; % Набор частот
8 str_1 = sprintf('Функции \n при T=%d, dt = %.3f', T, dt);
9 str_2 = sprintf('Фурье-образы \n при T=%d, dt = %.3f', T, dt);
10
11 % Прямоугольная функция
12 g = double(abs(t) <= 0.5);
13
14 % Коэффициенты c_m для связи FFT с непрерывным преобразованием
15 c = dt * exp(-2 * 1i * pi .* v * (-T/2)); % Масштабирующий коэффициент
16
17 % Прямое преобразование Фурье
18 g_hat_fft = fftshift(fft(g)) .* c;
19
20 % Аналитический Фурье-образ
21 g_hat_analytical = sinc(v);
22
23 % Обратное преобразование
24 g_recovered = ifft(ifftshift(g_hat_fft ./ c));
25
26 % Сравнение Фурье-образов
27 figure;
28 plot(v, g_hat_analytical, 'b', 'LineWidth', 2); hold on;
29 plot(v, real(g_hat_fft), 'r--', 'LineWidth', 1.5);
30 xlabel('\nu (Гц)');
31 legend('Аналитический', 'Улучшенный', 'Location', 'northwest');
32 title(str_2);
33 grid on;
34 xlim([-5 5]);
35
36 % Сравнение функций
37 figure;
38 plot(t, g, 'b', 'LineWidth', 2); hold on;
39 plot(t, real(g_recovered), 'r--', 'LineWidth', 1.5);
40 xlabel('Время (с)');
41 legend('Исходный', 'Восстановленный', 'Location', 'northwest');
42 title(str_1);
43 xlim([-T/2, T/2]);
44 ylim([-0.1, 1.1]);
45 grid on;

```

Листинг 4. Программа, которая реализует приближение непрерывного с помощью DFT.

```
1 % Инициализация параметров
2 a_1 = 2;
3 a_2 = 1;
4 omega_1 = 5;
5 omega_2 = 20*pi;
6 phi_1 = pi/6;
7 phi_2 = pi/4;
8 b = 6*pi;
9 B_1 = 20;
10 B_2 = 6;
11
12 % Параметры временной и частотной осей
13 T = 20; % Интервал времени
14 dt = 0.001; % Маленький шаг дискретизации
15 t = -T/2 : dt : T/2; % Набор временных шагов
16 V = 1/dt; % Ширина диапазона частот
17 dv = 1/T; % Шаг частоты
18 v = -V/2 : dv : V/2; % Набор частот
19
20 y1 = a_1 * sin(omega_1 .* t + phi_1) + a_2 * sin(omega_2 .* t +
    phi_2); % Функция y_1
21 y2 = sinc(b .* t); % Функция y_2
22
23 % Получение непрерывных графиков
24 figure;
25 plot(t, y_1, 'LineWidth', 1.5, 'Color', 'cyan');
26 legend('y_{1}(t)', 'Location', 'northwest');
27 grid on;
28 xlabel('t');
29 ylabel('f(t)');
30 title('График функции y_1(t)');
31 xlim([-2, 2]);
32 ylim([-3.1, 3.1]);
33
34 figure;
35 plot(t, y_2, 'LineWidth', 1.5, 'Color', 'cyan');
36 legend('y_2(t)', 'Location', 'northwest');
37 grid on;
38 xlabel('t');
39 ylabel('f(t)');
40 title('График функции y_2(t)');
41 ylim([-0.24, 1.1]);
42 xlim([-3, 3]);
43
44 % Получение графиков при сэмплировании
45 dt1 = 0.01; % Разреженная сетка
46
47 % Создание временной сетки
48 t_sparse = -T/2:dt1:T/2; % Разреженная сетка
49
```

```
50 % Создание сэмплованных сигналов
51 y1_sparse = a_1*sin(omega_1 .* t_sparse + phi_1) + a_2*sin(omega_2
    .* t_sparse + phi_2);
52 y2_sparse = sinc(b .* t_sparse);
53
54 % % Графики для y1(t)
55 figure;
56 plot(t, y1, 'b', 'LineWidth', 1.5);
57 hold on;
58 stem(t_sparse, y1_sparse, 'r', 'LineWidth', 1);
59 title('y_1(t): Исходный и сэмплованный сигналы');
60 legend('Непрерывный', 'Сэмплы', 'Location', 'best');
61 xlabel('t');
62 ylabel('f(t)');
63 xlim([-1 1]);
64 ylim([-3.1, 3.1]);
65 grid on;
66
67 % Графики для y2(t)
68 figure;
69 plot(t, y2, 'b', 'LineWidth', 1.5);
70 hold on;
71 stem(t_sparse, y2_sparse, 'r', 'LineWidth', 1);
72 xlabel('t');
73 ylabel('f(t)');
74 title('y_2(t): Исходный и сэмплованный сигналы');
75 legend('Непрерывный', 'Сэмплы', 'Location', 'northwest');
76 xlim([-1 1]);
77 ylim([-0.25, 1.1]);
78 grid on;
79
80 % Восстановление сигналов
81
82 % Восстановление y1(t)
83 y1_recovered = zeros(size(t));
84
85 for idx = 1:length(t)
86     t_current = t(idx);
87     y1_recovered(idx) = sum(y1_sparse .* sinc((t_current -
        t_sparse)*2*B_1) * 2 * B_1 * dt1);
88 end
89
90 % Восстановление y2(t)
91 y2_recovered = zeros(size(t));
92 for idx = 1:length(t)
93     t_current = t(idx);
94     y2_recovered(idx) = sum(y2_sparse .* sinc((t_current - t_sparse)
        *4*B_2) * 4 * B_2 * dt1);
95 end
96
```

```
97 % Визуализация восстановленных сигналов
98 figure;
99 plot(t, y1, 'b', 'LineWidth', 2); hold on;
100 plot(t, y1_recovered, 'g--', 'LineWidth', 1.5);
101 title('y_1(t): Сравнение с восстановленным сигналом');
102 xlabel('t');
103 ylabel('f(t)');
104 legend('Исходный', 'Восстановленный', 'Location', 'best');
105 xlim([-1 1]);
106 grid on;
107
108 figure;
109 plot(t, y2, 'b', 'LineWidth', 2); hold on;
110 plot(t, y2_recovered, 'g--', 'LineWidth', 1.5);
111 title('y_2(t): Сравнение с восстановленным сигналом');
112 xlabel('t');
113 ylabel('f(t)');
114 legend('Исходный', 'Восстановленный', 'Location', 'northwest');
115 xlim([-1 1]);
116 ylim([-0.22, 1.1]);
117 grid on;
```

Листинг 5. Программа, используемая для выполнения задания 2.