



НИУ ИТМО

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №5

“Связь непрерывного и дискретного”

Выполнил:

Александр Иванов, R3238

Преподаватель:

Перегудин А. А.

Санкт-Петербург, 2024

Содержание

1. Непрерывное и дискретное преобразование Фурье	3
1.1. Рассматриваемая функция	3
1.2. Нахождения образа аналитически	3
1.3. Нахождение образа численно	4
1.4. Восстановление функции	5
1.5. Увеличение промежутка интегрирования	7
1.6. Увеличение количества точек разбиения	9
2. Использование дискретного преобразования Фурье	12
2.1. Восстановление функции из ее образа	12

1. Непрерывное и дискретное преобразование Фурье

1.1. Рассматриваемая функция

Рассмотрим прямоугольную функцию, которая задается следующим образом:

$$f(t) = \begin{cases} 1, & |t| \leq 1/2, \\ 0, & |t| > 1/2 \end{cases} \quad (1)$$

Ее график приведен на рисунке 1.

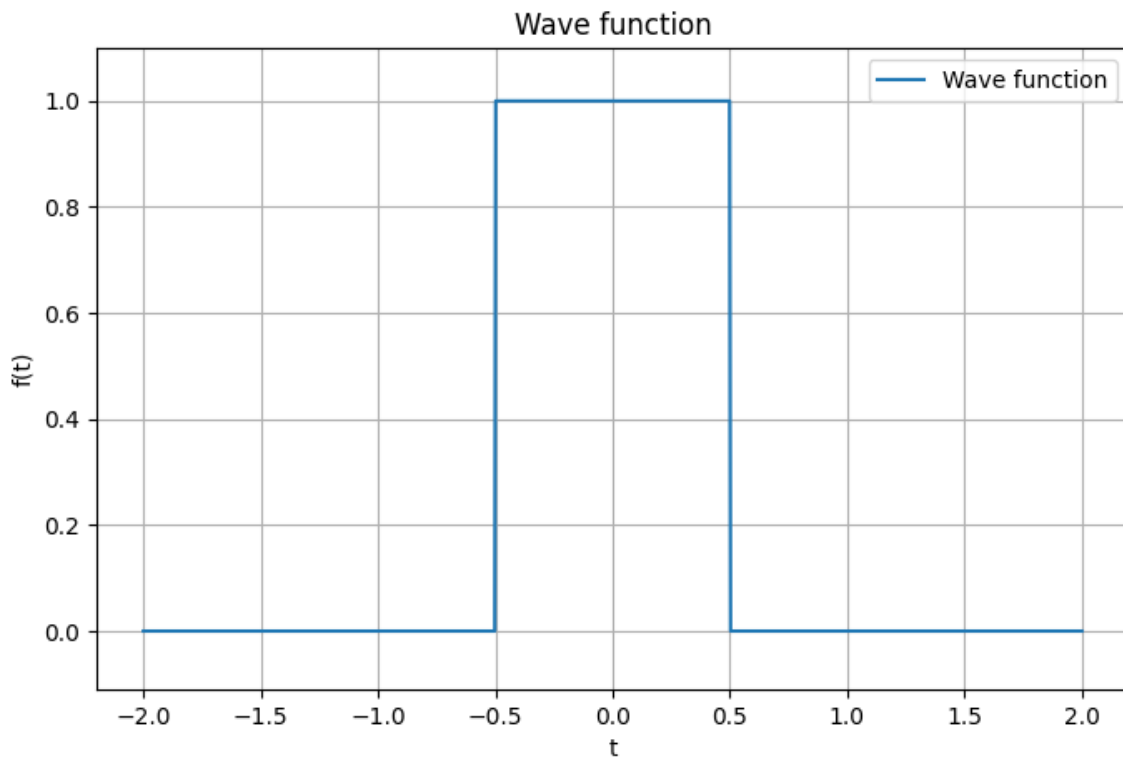


Рис. 1: График прямоугольной функции

1.2. Нахождения образа аналитически

$$F(v) = \hat{f} = \int_{-\infty}^{\infty} f(t) e^{-2\pi i v t} dt = \int_{-1/2}^{1/2} e^{-2\pi i v t} dt = \frac{e^{-\pi i v} - e^{\pi i v}}{-2\pi i v} = \frac{\sin(\pi v)}{\pi v} = \text{sinc}(\pi v) \quad (2)$$

График истинного образа функции приведен на рисунке 2.

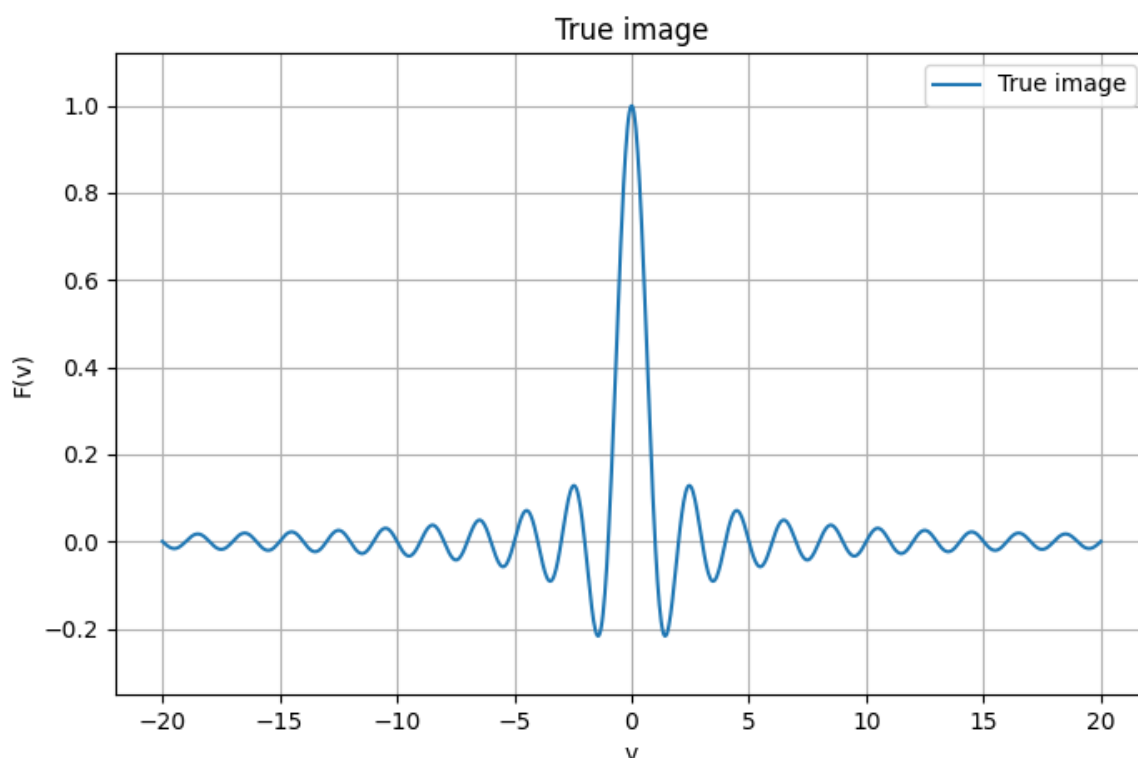


Рис. 2: Истинный образ функции

1.3. Нахождение образа численно

Теперь, для того, чтобы найти образ данной функции, воспользуемся численным интегрированием. Так как такой метод не может обеспечить интегрирование по бесконечному интервалу, то будем интегрировать по ограниченному интервалу. Для начала рассмотрим разбиение из 1000 точек и интегрирование по интервалу $[-20, 20]$.

График образа функции $f(t)$, полученный численным методом, приведен на рисунке 3.

Рассмотрим сравнительный график истинного и численного образа функции на рисунке 4.

Графики истинного и численного образа функции совпадают, что говорит о корректности численного метода. Для более подробного анализа рассмотрим графики разности истинного и численного образа функции (*ошибки*) на рисунке 5. На рисунке 5 видно, что ошибка

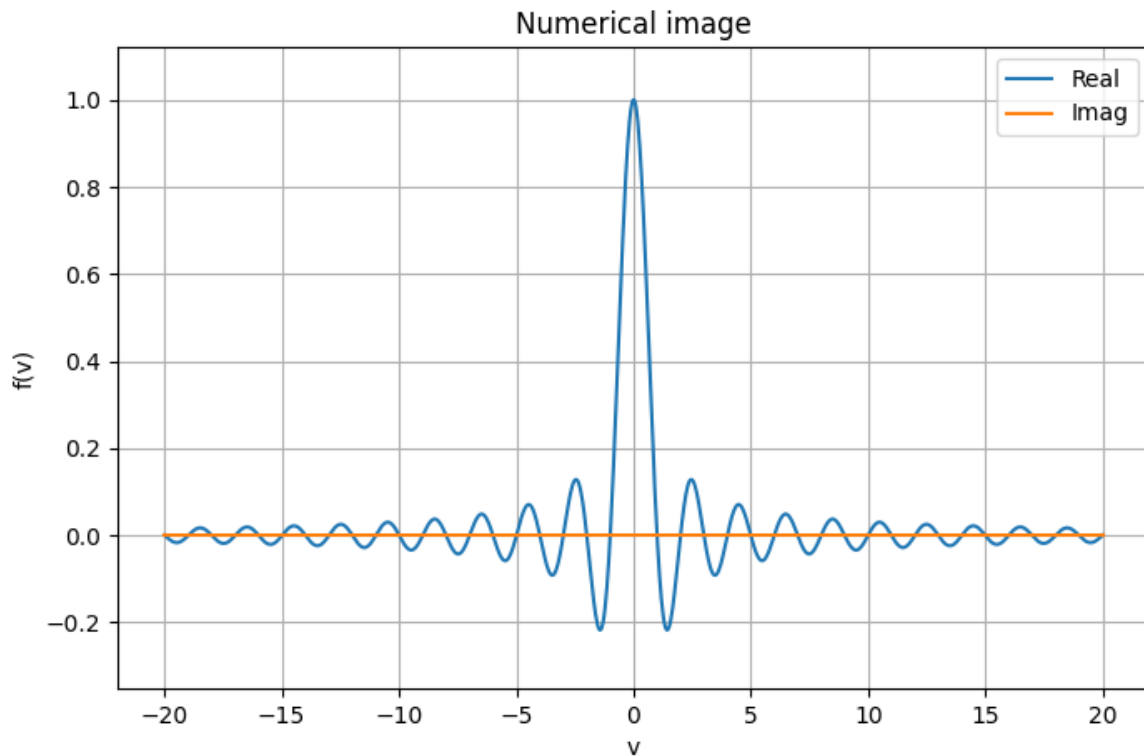


Рис. 3: Образ функции, полученный численным методом (разбиение на 1000 точек, интервал $[-20, 20]$)

представляет из себя гармоническую функцию с амплитудой $\approx 10^{-3}$, что говорит о том, что численный метод хорошо смог приблизиться к истинному образу функции.

1.4. Восстановление функции

Теперь, имея образ функции, можно восстановить исходную функцию. Применим обратное преобразование Фурье к численному образу функции и сравним результат с исходной функцией. Графики исходной функции и восстановленной функции приведены на рисунке 6. На рисунке 6 видно, что исходная и восстановленная функции не полностью совпадают, в точках разрыва заметны наиболее сильные отклонения. Это связано, по большей части, с тем, что интегрирование численными методами невозможно провести на бесконечном интервале, а также с тем, что восстановление функции происходит с использованием образа функции, который также был получен численным методом.

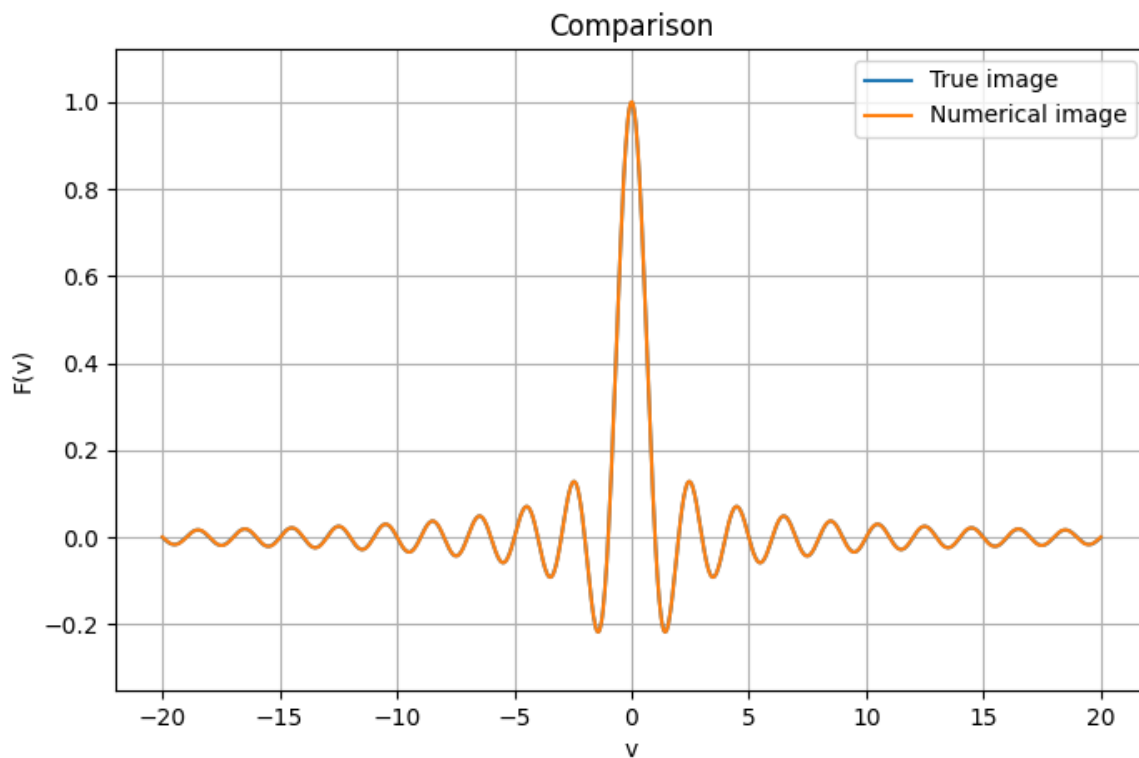


Рис. 4: Сравнение истинного и численного образа функции (разбиение на 1000 точек, интервал $[-20, 20]$)

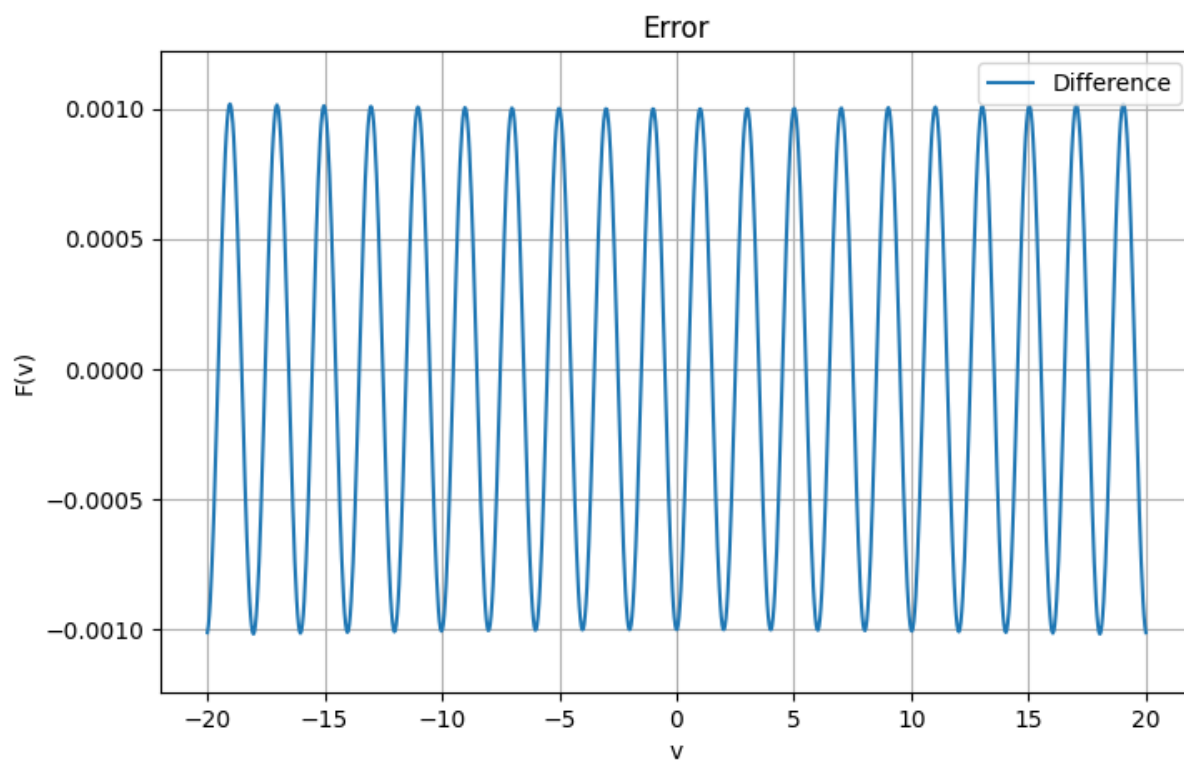


Рис. 5: График ошибки (разбиение на 1000 точек, интервал $[-20, 20]$)

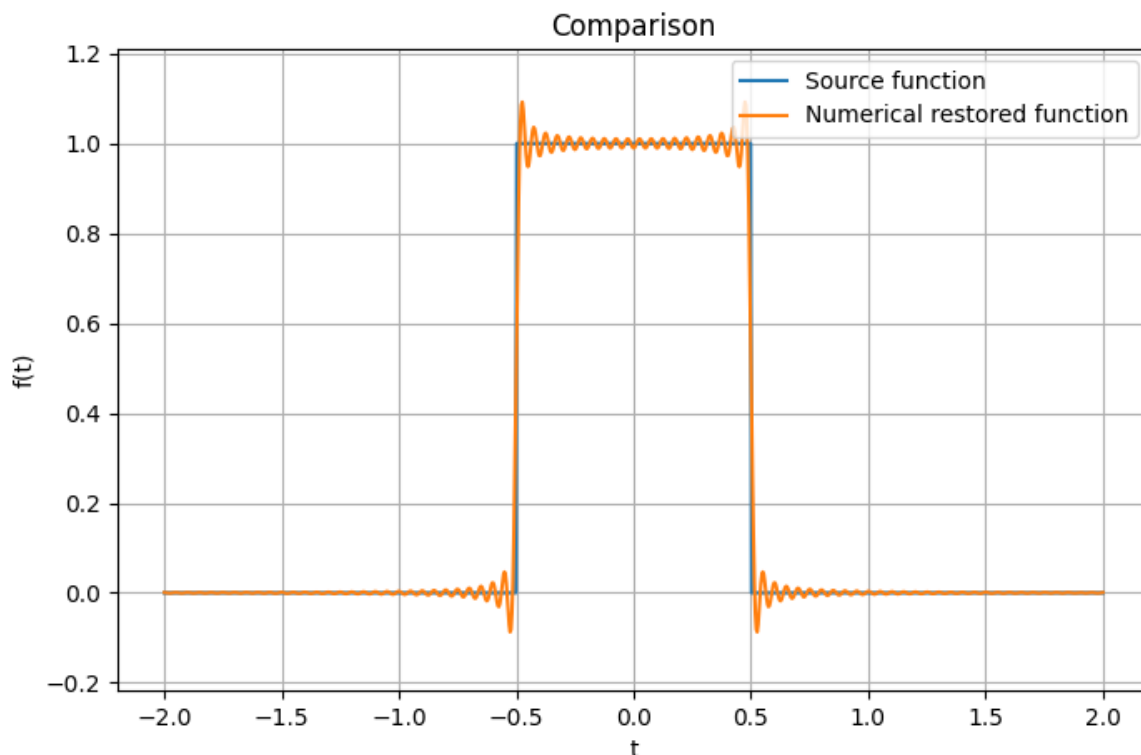


Рис. 6: Сравнение исходной и восстановленной функции (разбиение на 1000 точек, интервал $[-20, 20]$)

1.5. Увеличение промежутка интегрирования

Для того, чтобы уменьшить ошибку восстановления функции, увеличим промежуток интегрирования до $[-80, 80]$. Количество точек разбиения оставим прежним.

Для анализа сразу рассмотрим график ошибки на рисунке 7.

Ошибка, как ни странно, выросла. Это может быть связано с тем, что точность вычисления интеграла при увеличении промежутка интегрирования, при этом с прежним количеством точек разбиения, уменьшается. Тем не менее, рассмотрим сравнительные графики исходной и восстановленной функции на рисунке 8.

На рисунке 8 видно, что восстановленная функция стала более похожа на исходную, чем в предыдущем случае, несмотря на то, что ошибка в образе функции увеличилась. Это можно связать с тем, что вклад ошибки от малого количества точек разбиения меньше, чем вклад ошибки от малого промежутка интегрирования функции $\text{sinc}(\pi v)$, которая, хоть и стремится к нулю на бесконечности, но не равна ему на конечных интервалах.

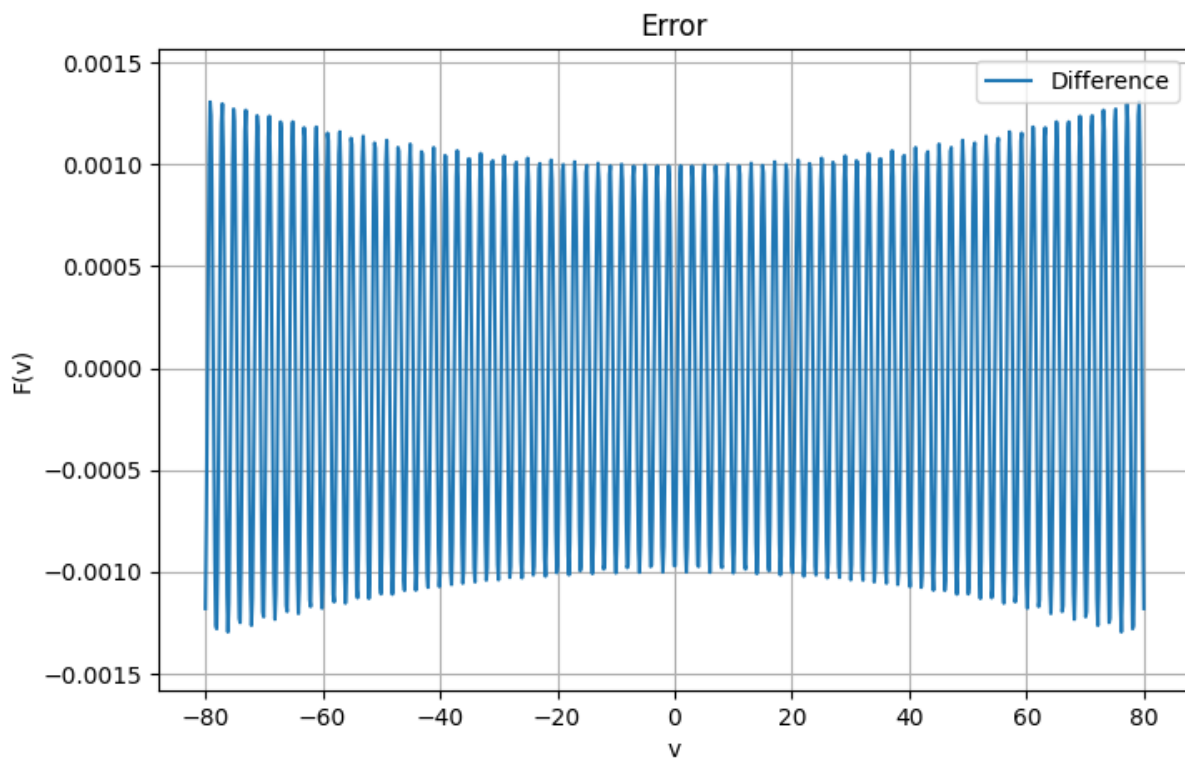


Рис. 7: График ошибки (разбиение на 1000 точек, интервал $[-80, 80]$)

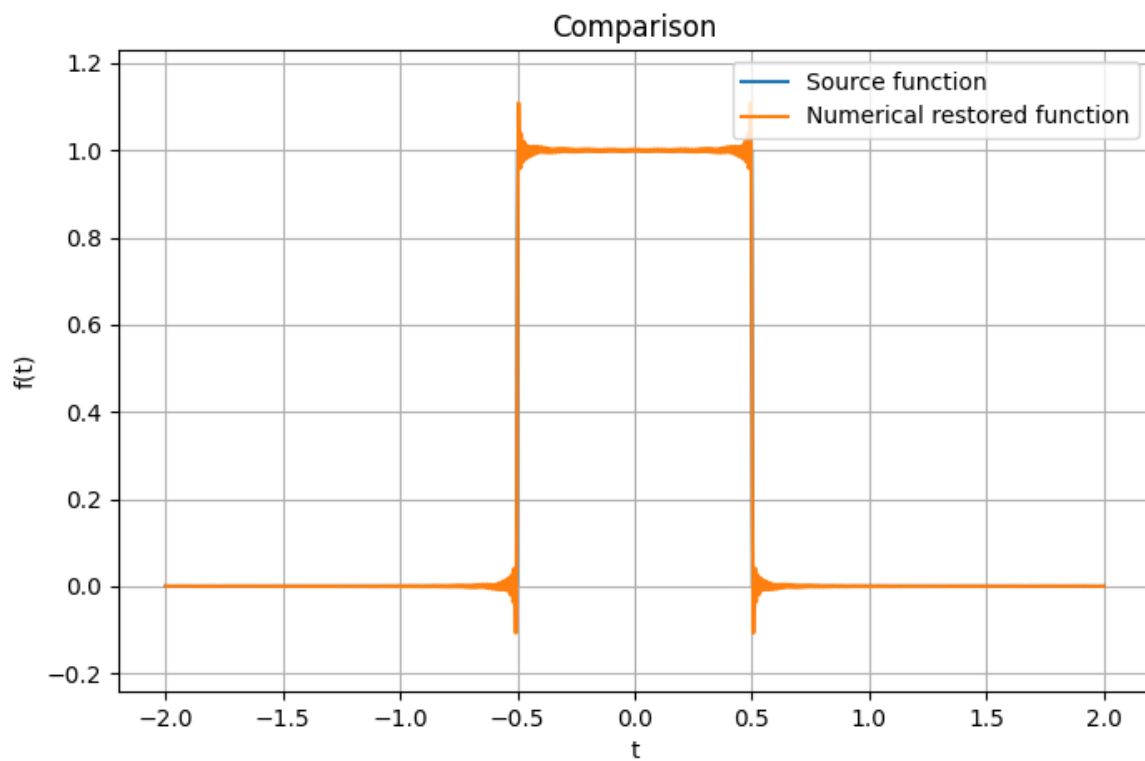


Рис. 8: Сравнение исходной и восстановленной функции (разбиение на 1000 точек, интервал $[-80, 80]$)

1.6. Увеличение количества точек разбиения

Теперь, кроме увеличения промежутка интегрирования, увеличим количество точек разбиения до 10000. Сразу же перейдем к анализу графика ошибки на рисунке 9.

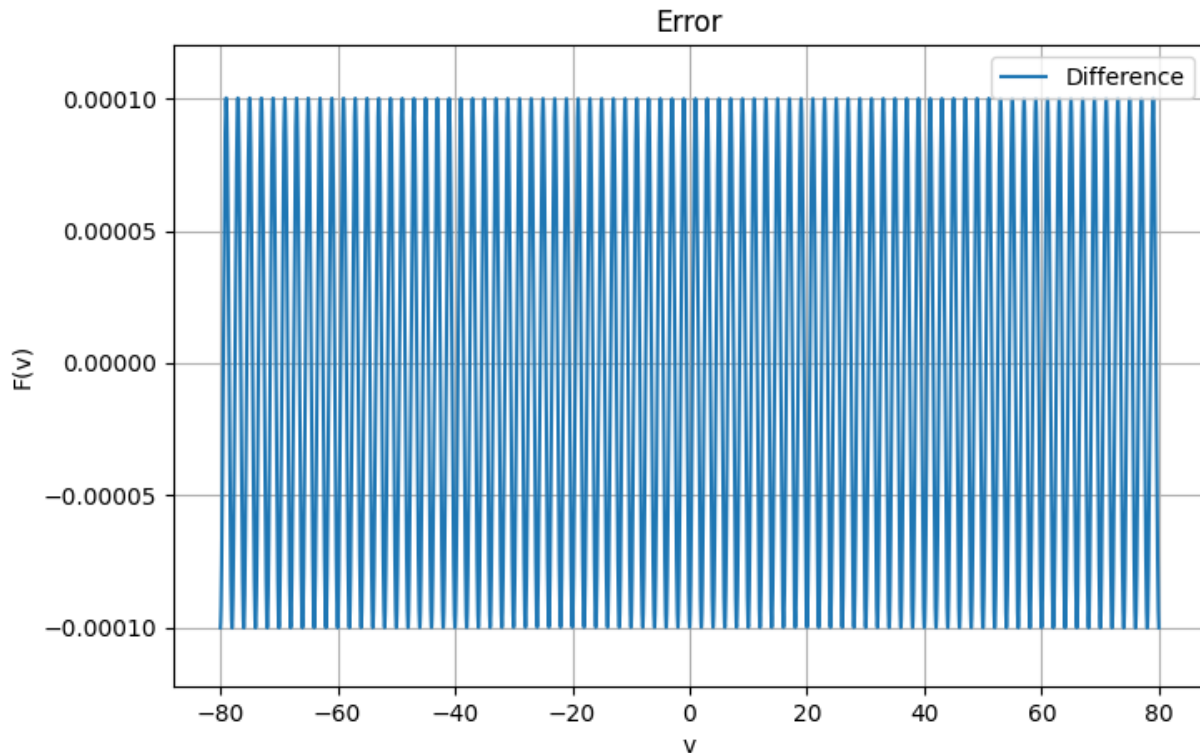


Рис. 9: График ошибки (разбиение на 10000 точек, интервал $[-80, 80]$)

Ошибка уменьшилась, что говорит о том, что увеличение количества точек разбиения дает более точное интегрирование, как и предполагалось. Теперь ошибка по модулю не превышает 10^{-4} : при увеличении числа точек разбиения на порядок, ошибка уменьшилась на порядок.

При этом, при увеличении количества точек разбиения, сильно увеличивается время вычисления образа функции. Это, очевидно, связано с необходимостью проведения большего числа операций с плавающей точкой.

Рассмотрим сравнительные графики исходной и восстановленной функции на рисунке 10.

В данном случае уже довольно трудно судить о величине ошибки, поэтому рассмотрим график разности исходной и восстановленной функции на рисунке 11.

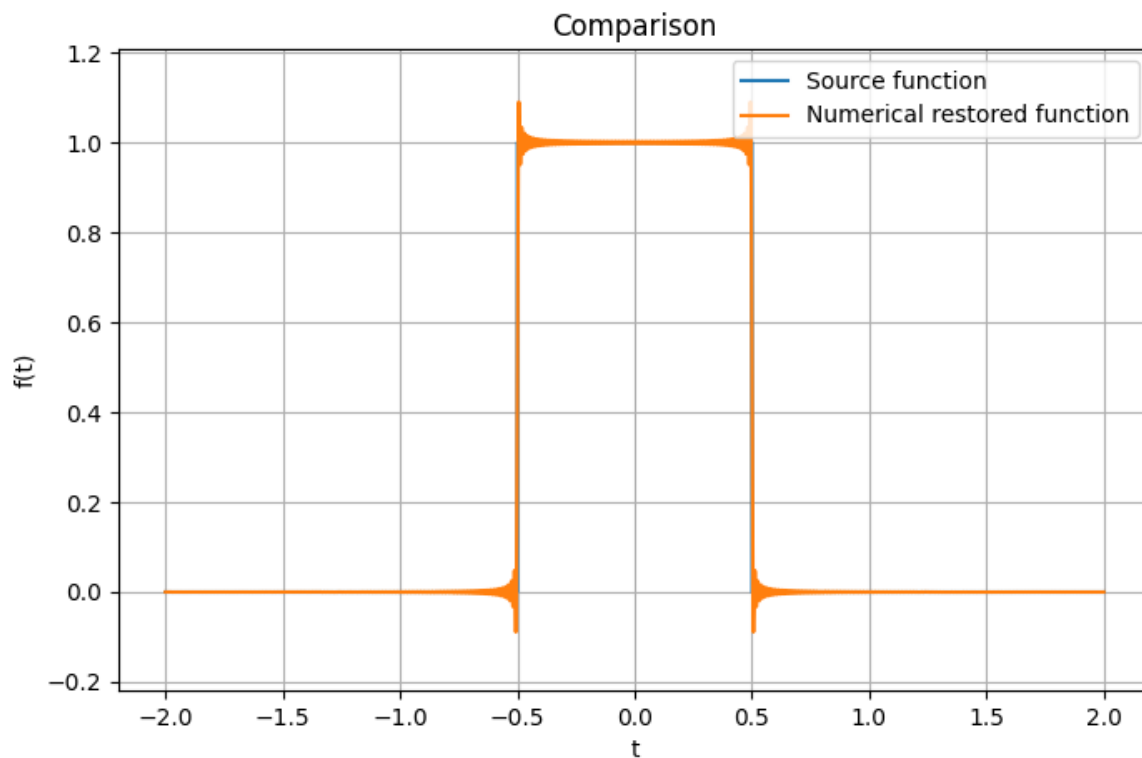


Рис. 10: Сравнение исходной и восстановленной функции (разбиение на 10000 точек, интервал $[-80, 80]$)

И ошибки для прошлого случая на рисунке 9.

Видно, что как по величине, так и по форме, ошибка восстановления функции уменьшилась. Это говорит о том, что увеличение количества точек разбиения дает более точное восстановление функции, что и ожидалось.

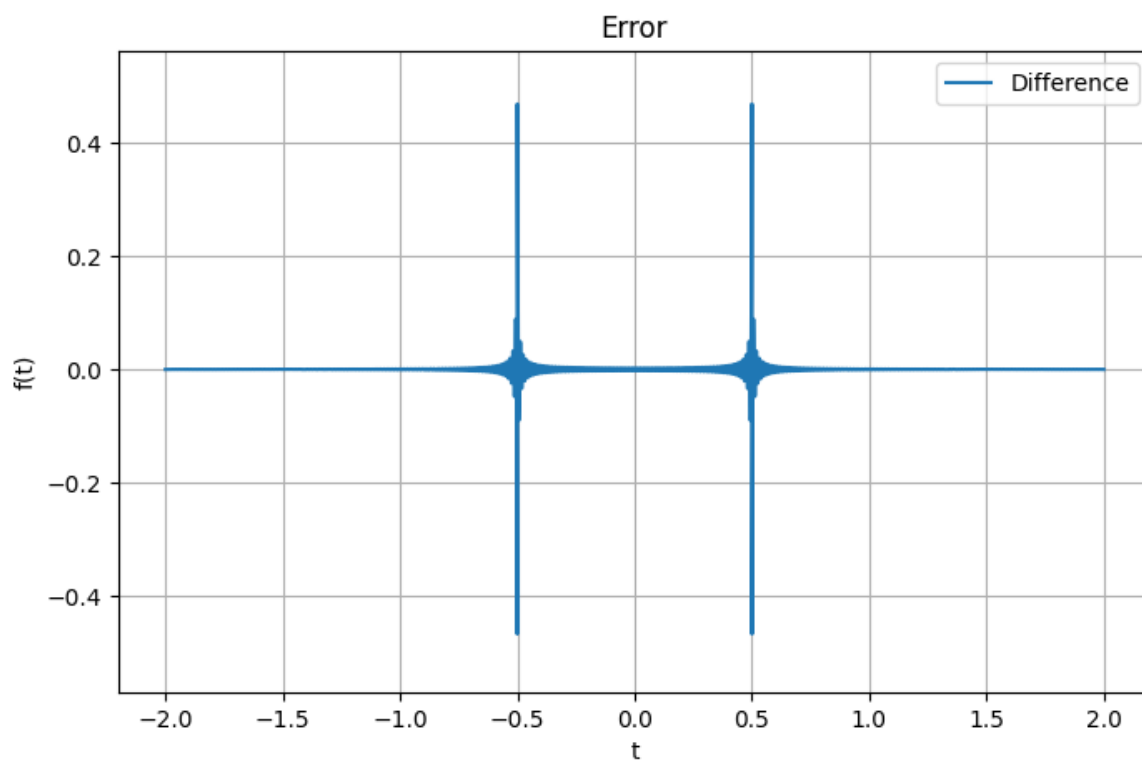


Рис. 11: График разности исходной и восстановленной функции (разбиение на 10000 точек, интервал $[-80, 80]$)

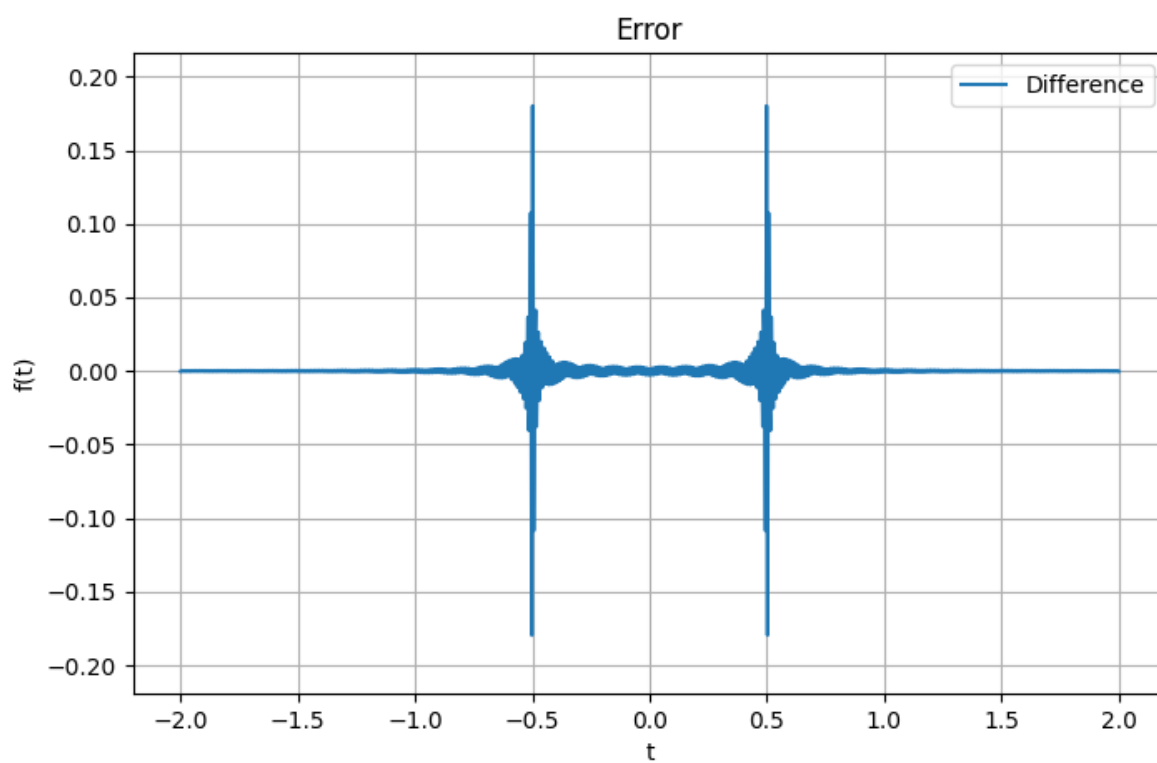


Рис. 12: График разности исходной и восстановленной функции (разбиение на 1000 точек, интервал $[-80, 80]$)

2. Использование дискретного преобразования Фурье

В библиотеке `numpy.fft` есть функция `fft`, которая реализует быстрое преобразование Фурье. Задается она, согласно документации, следующим образом:

$$F(k) = \sum_{n=0}^{N-1} f(n) e^{-2\pi i k n / N} x(n) \quad (3)$$

И обратное преобразование Фурье `ifft`:

$$f(n) = \frac{1}{N} \sum_{k=0}^{N-1} F(k) e^{2\pi i k n / N} \quad (4)$$

Для того, чтобы сделать это преобразование унитарным, необходимо умножить результат прямого преобразования на $\frac{1}{\sqrt{N}}$, а результат обратного преобразования на \sqrt{N} .

2.1. Восстановление функции из ее образа

Посмотрим на то, как будет выглядеть восстановленная из образа функции при использовании `ifft` на рисунке 13.

На рисунке 13 видно, что исходная и восстановленная функции не совпадают. Посмотрим на график разности исходной и восстановленной функции на рисунке 14.

Видно, что ошибка по модулю не превышает $4 \cdot 10^{-16}$, что говорит о том, что восстановленная функция практически идеально совпадает с исходной. При этом скорость работы алгоритма быстрого преобразования Фурье значительно выше, чем обычного преобразования Фурье.

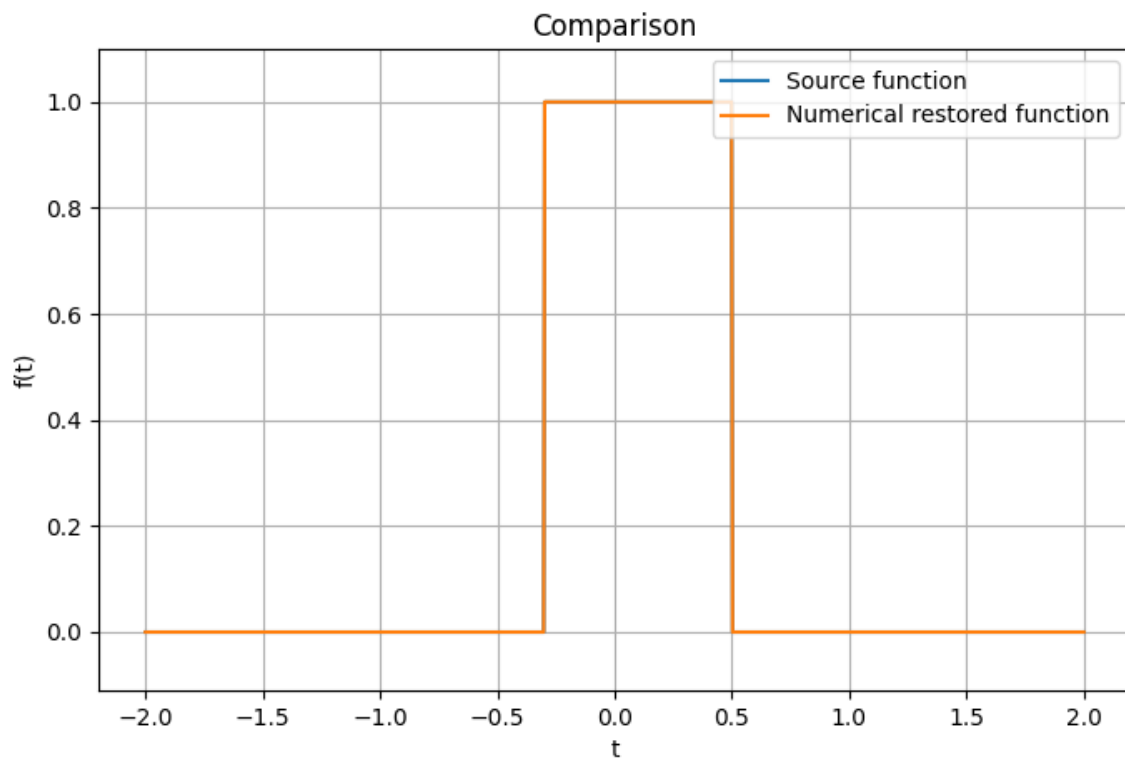


Рис. 13: Сравнение исходной и восстановленной функции (разбиение на 1000 точек)

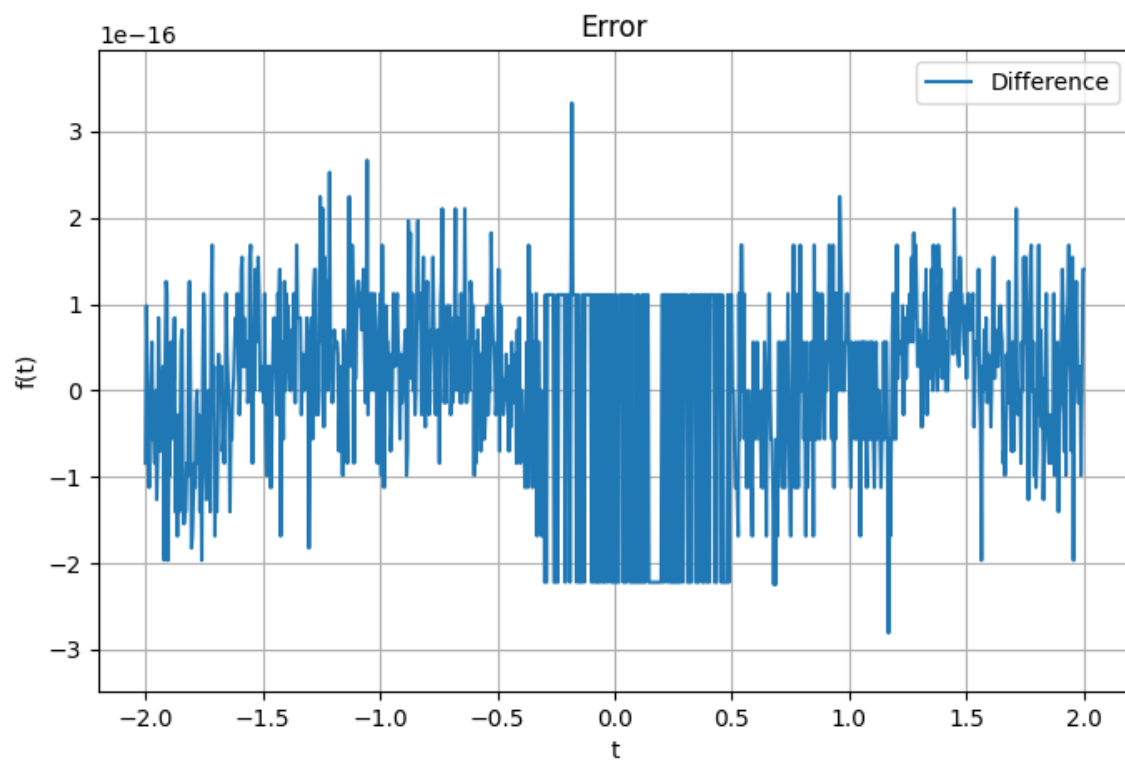


Рис. 14: Разность исходной и восстановленной функции (разбиение на 1000 точек)