



НИУ ИТМО

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №3

“Жесткая фильтрация”

Выполнил:

Александр Иванов, R3238

Преподаватель:

Перегудин А.А.

Санкт-Петербург, 2024

Содержание

1. Задание 1. Жесткие фильтры	4
1.1. Исходная функция	4
1.2. Убираем высокие частоты	4
1.2.1. Рассматриваемая функция	4
1.2.2. Графики рассматриваемой и зашумленной функции	4
1.2.3. Нахождение образов исходной и зашумленной функций	6
1.2.4. Фильтрация высоких частот	7
1.2.5. Сравнение модулей образов	9
1.2.6. Исследование влияния частоты среза на фильтрацию	11
1.2.7. Исследование влияния параметра b на фильтрацию	13
1.3. Убираем специфические частоты	17
1.3.1. Рассматриваемая функция	17
1.3.2. Графики рассматриваемой и зашумленной функции	18
1.3.3. Нахождение образов исходной и зашумленной функций	19
1.3.4. Фильтрация высоких частот	20
1.3.5. Сравнение модулей образов	25
1.3.6. Только гармонический шум	26
1.3.7. Исследование влияния параметров b , c , d на фильтрацию	30
1.4. Убираем низкие частоты	32
1.4.1. Рассматриваемая функция	32
1.4.2. Графики рассматриваемой и зашумленной функции	33
1.4.3. Нахождение образов исходной и зашумленной функций	34
1.4.4. Фильтрация низких частот	35

2. Задание 2. Фильтрация звука	37
2.1. Чтение исходных данных	37
2.2. Нахождение образа исходного сигнала	37
2.3. Определение частот с шумом	39
2.4. Обратное преобразование	39

1. Задание 1. Жесткие фильтры

1.1. Исходная функция

Для выполнения задания рассмотрим функцию $g(t)$, которая задается следующим образом:

$$g(t) = \begin{cases} a, & t \in [t_1, t_2] \\ 0, & otherwise \end{cases} \quad (1)$$

Кроме того, зададим *зашумленную* версию функции $u(t)$

$$u(t) = g(t) + b(\text{rand}(t) - 0.5) + c(\sin(d \cdot t)) \quad (2)$$

Для расчетов будем использовать массив из 1000 точек, соответствующих значениям функции $u(t)$ на интервале $[-T/2, T/2]$.

1.2. Убираем высокие частоты

1.2.1. Рассматриваемая функция

Рассмотрим функцию $g(t)$ при параметрах $a = 3$, $t_1 = -2$, $t_2 = 4$ (см. рисунок 1) и ее *зашумленную* версию $u(t)$ с параметрами $b = 1$, $c = 0$, $d = 0$ (см. рисунок 2). на промежутке $[-T/2, T/2]$ с $T = 10$.

1.2.2. Графики рассматриваемой и зашумленной функции

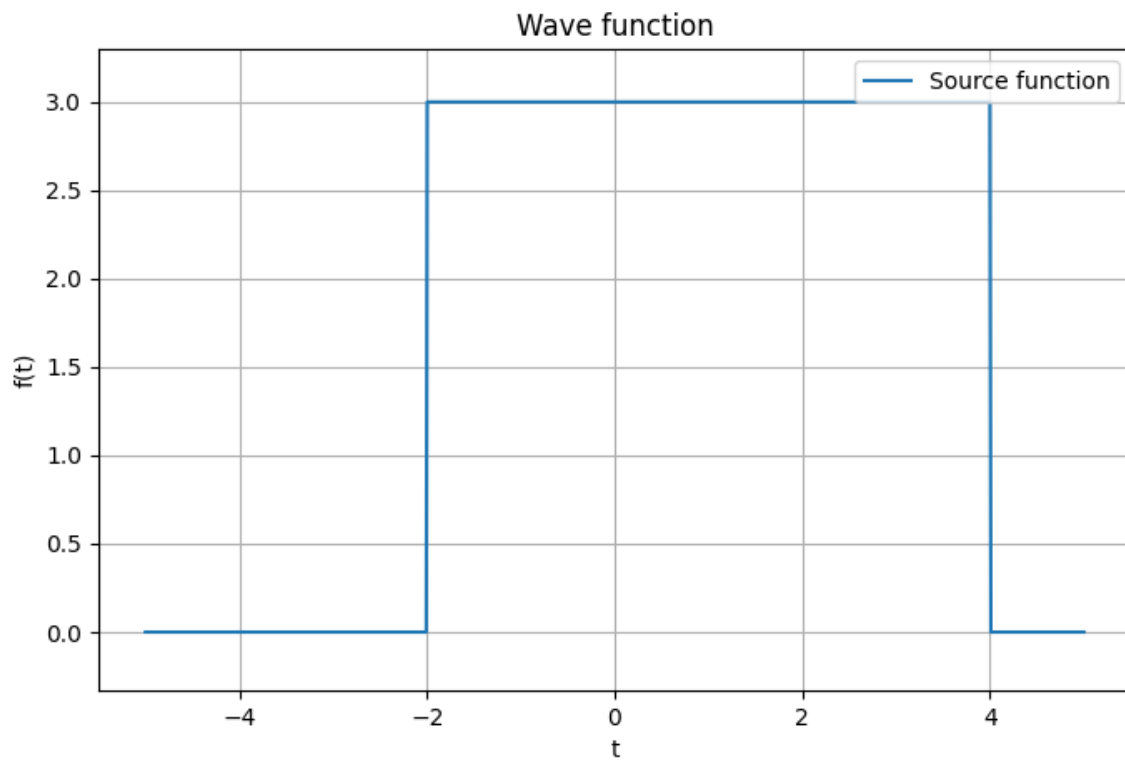


Рис. 1: Функция $g(t)$ с параметрами $a = 3$, $t_1 = -2$, $t_2 = 4$

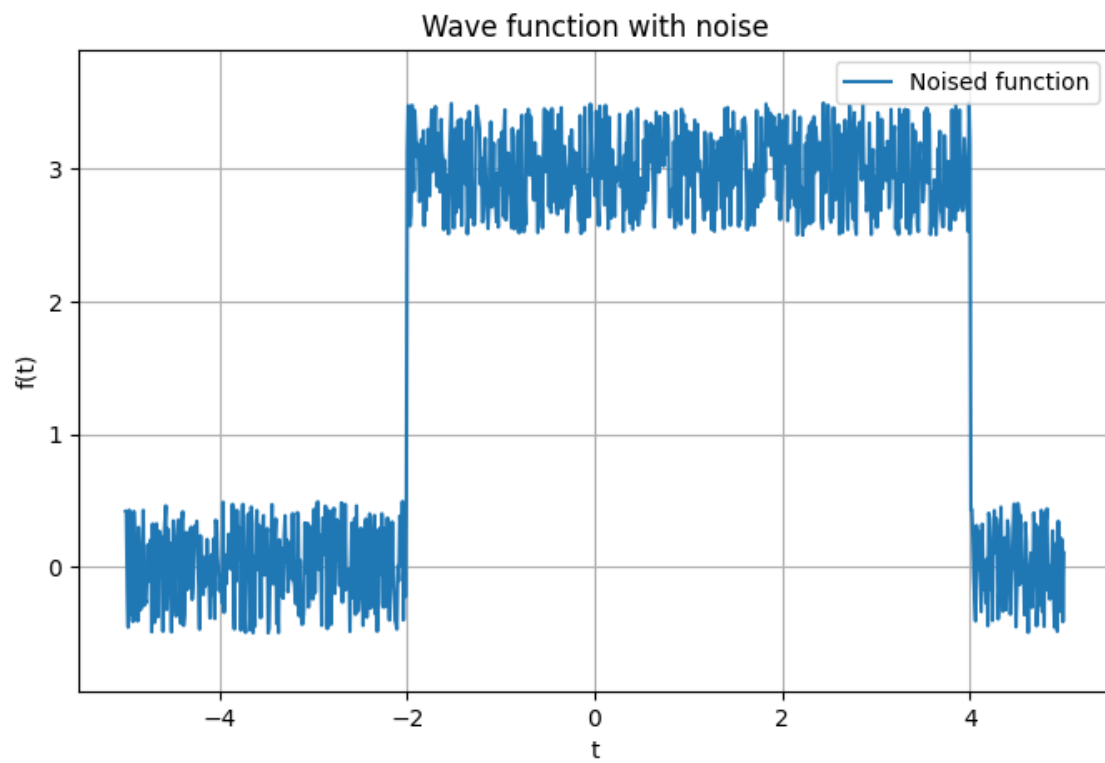


Рис. 2: Функция $u(t)$ с параметрами $b = 1$, $c = 0$, $d = 0$

1.2.3. Нахождение образов исходной и зашумленной функций

Найдем образы исходной (см. рисунок 3) и зашумленной (см. рисуну 4) функций с помощью преобразования Фурье.

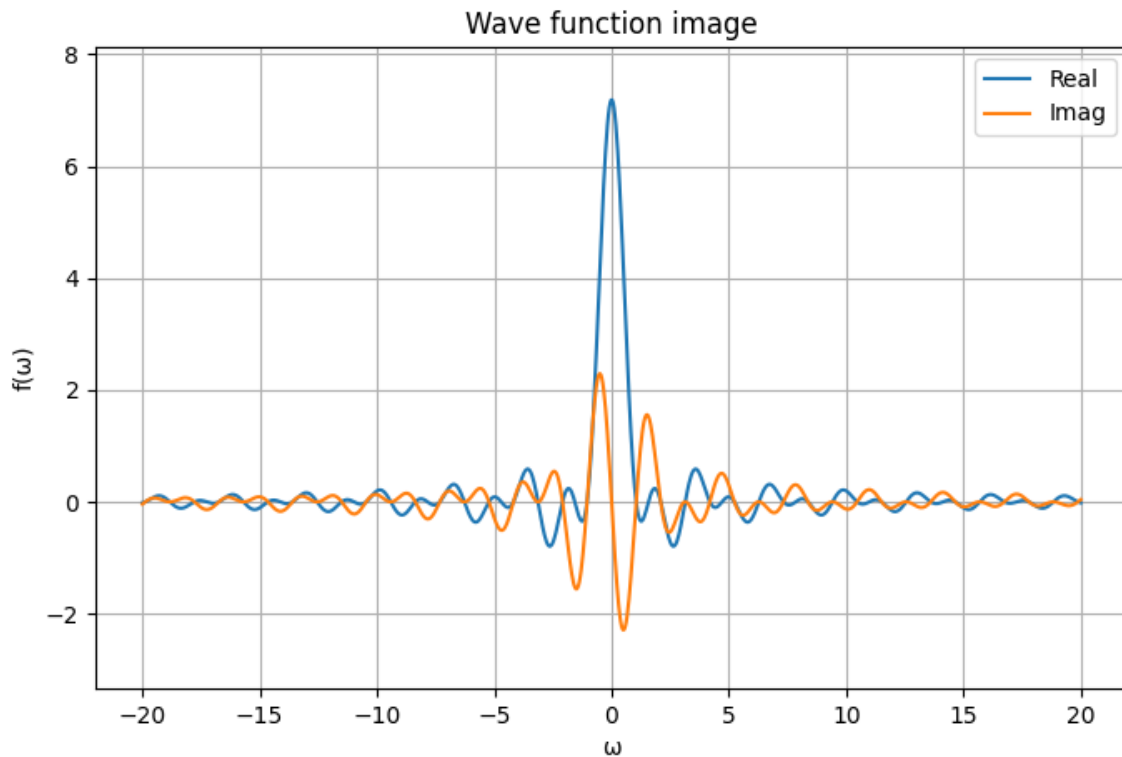


Рис. 3: Образ функция $g(t)$ с параметрами $a = 3$, $t_1 = -2$, $t_2 = 4$

Разница между образами исходной и зашумленной функций не заметна на графиках. Это может быть связано с тем, что высокочастотные компоненты функции $u(t)$ не сильно отличаются от нуля.

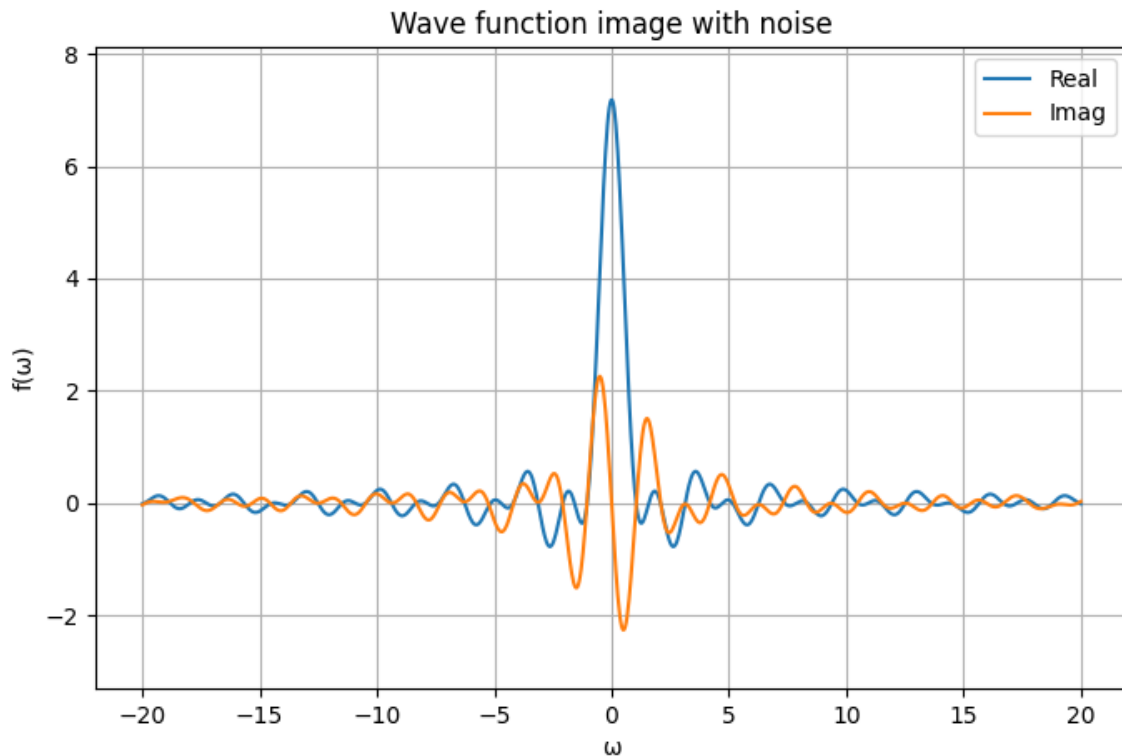


Рис. 4: Образ функция $u(t)$ с параметрами $b = 1$, $c = 0$, $d = 0$

1.2.4. Фильтрация высоких частот

Обрежем образ функции $u(t)$, убрав компоненты, соответствующие высоким частотам (см. рисунок 5).

Как и ожидалось, высокочастотные компоненты ($|\omega| > \omega_0$) стали равны нулю. Теперь выполним обратное преобразование Фурье (используя обрезанный образ), получив при этом фильтрованную версию функции (см. рисунок 6).

Видим, что шумов стало меньше, но функция не идентична исходной, так как удаление высокочастотных компонент сказалось не только на шумах, но и на самой функции. Сравнительные графики исходной и фильтрованной функций представлены на рисунке 7.

Таким образом, фильтрация высоких частот позволяет убрать шумы, не сильно влияя на саму функцию.

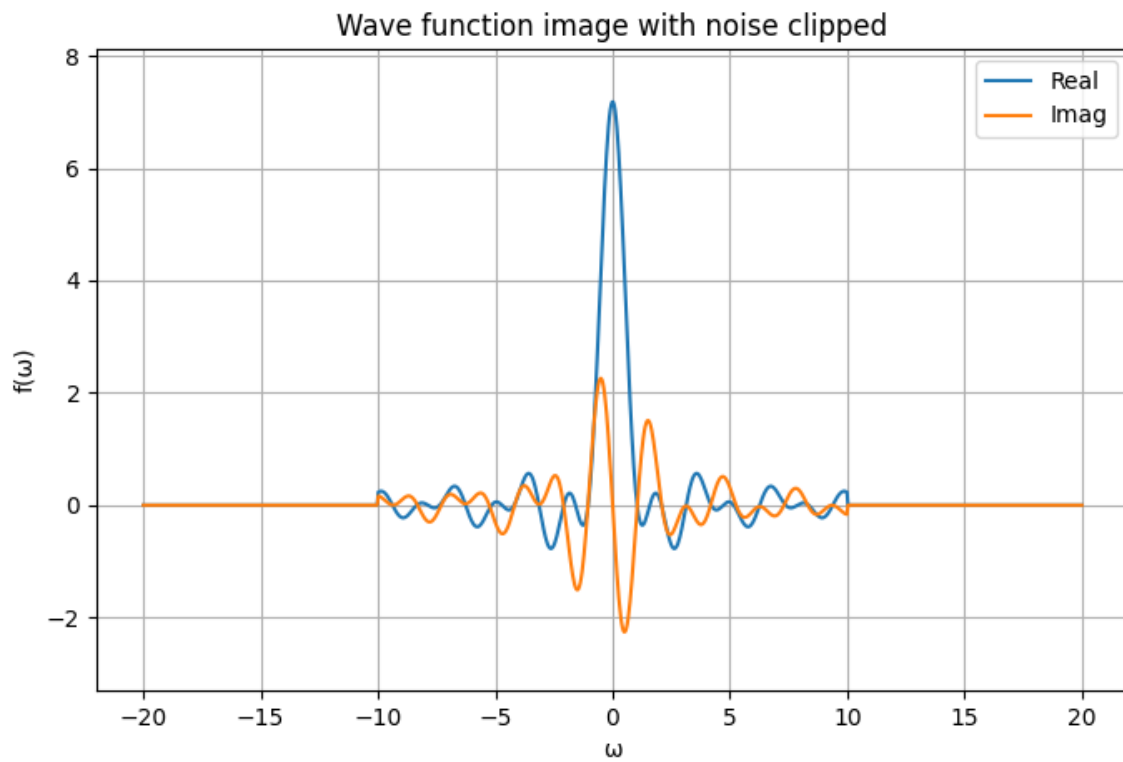


Рис. 5: Обрезанный образ функция $u(t)$ при $\omega_0 = 10$

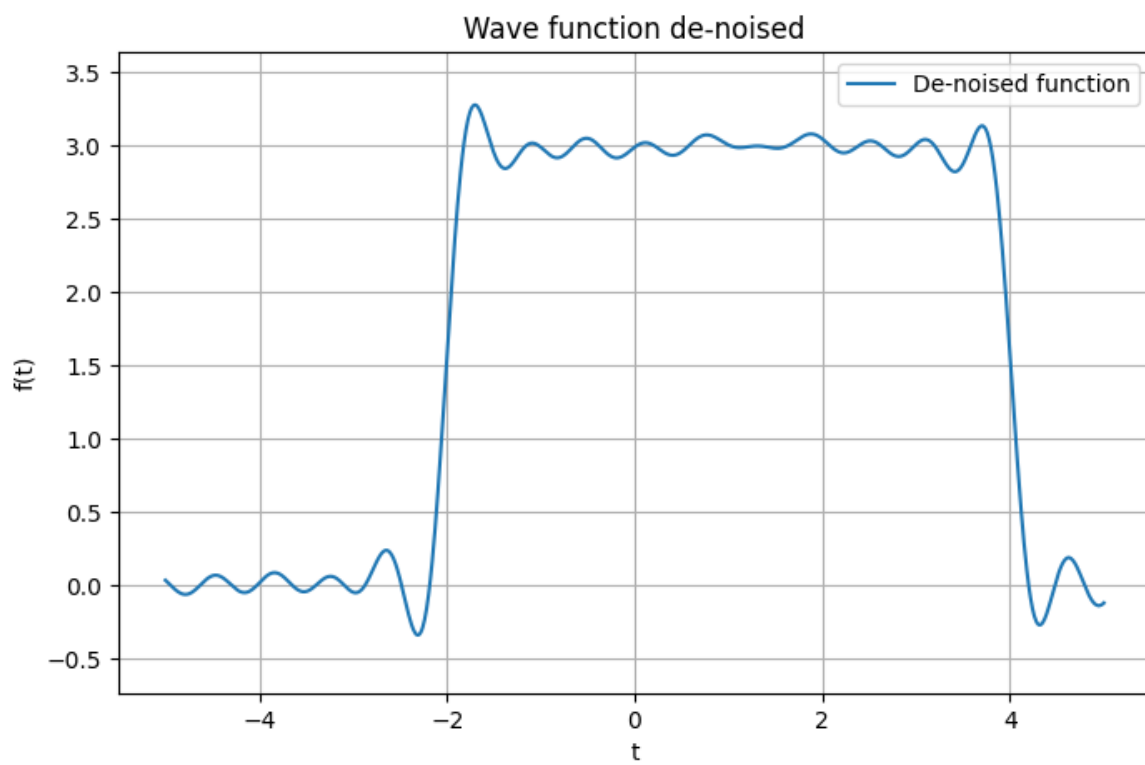


Рис. 6: Фильтрованная функция $u(t)$ при $\omega_0 = 10$

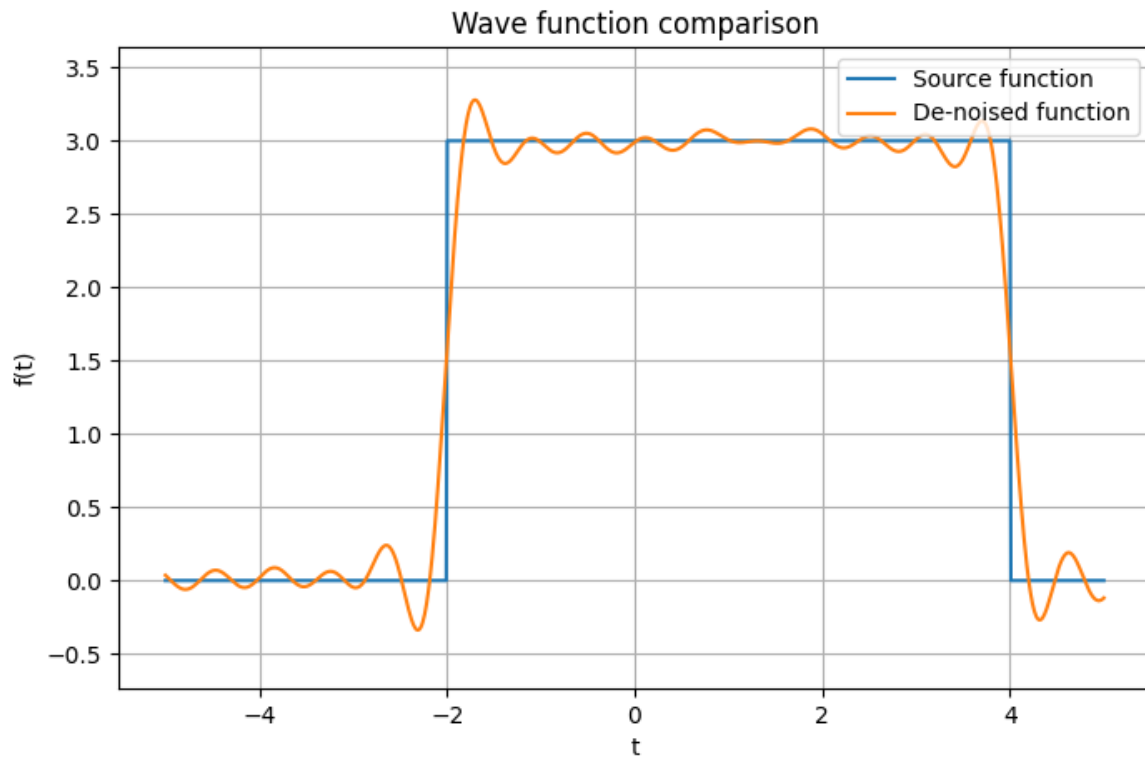


Рис. 7: Исходная и фильтрованная функция при $\omega_0 = 10$

1.2.5. Сравнение модулей образов

Сравним модули образов исходной и фильтрованной функций (см. рисунок 8 и 9).

Видим, что практически частоты выше ω_0 обнулились, при этом низкочастотные компоненты остались практически неизменными.

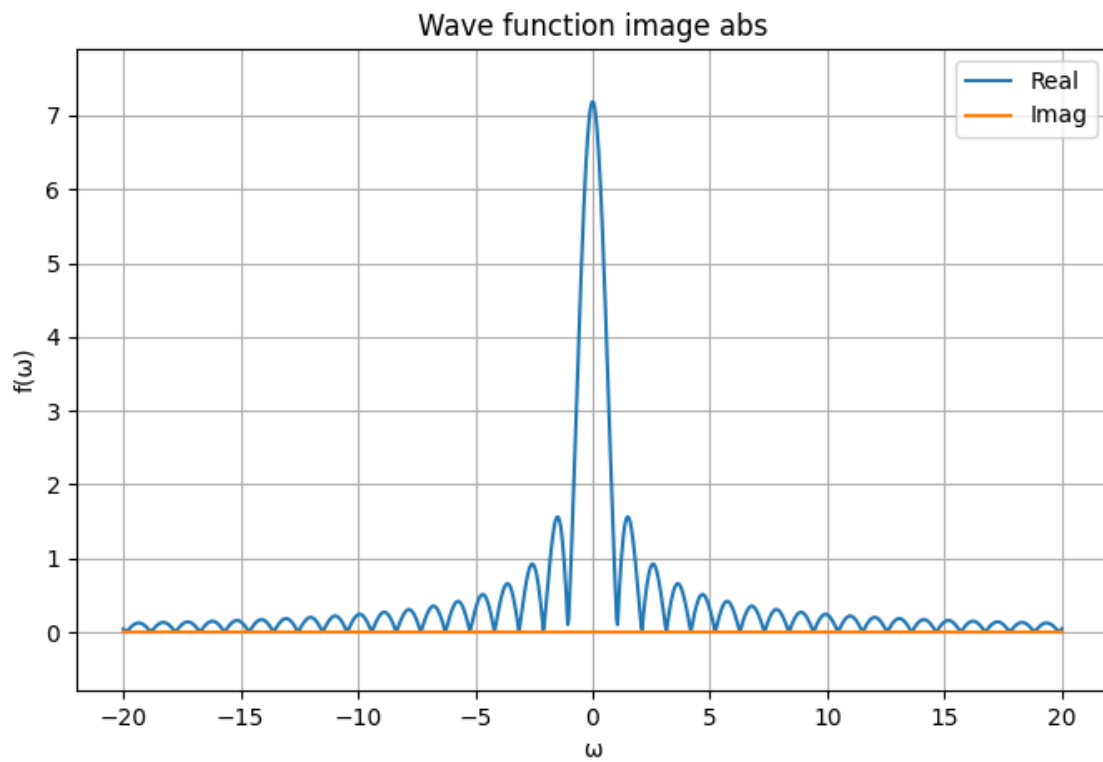


Рис. 8: Модуль образа исходной функции $g(t)$

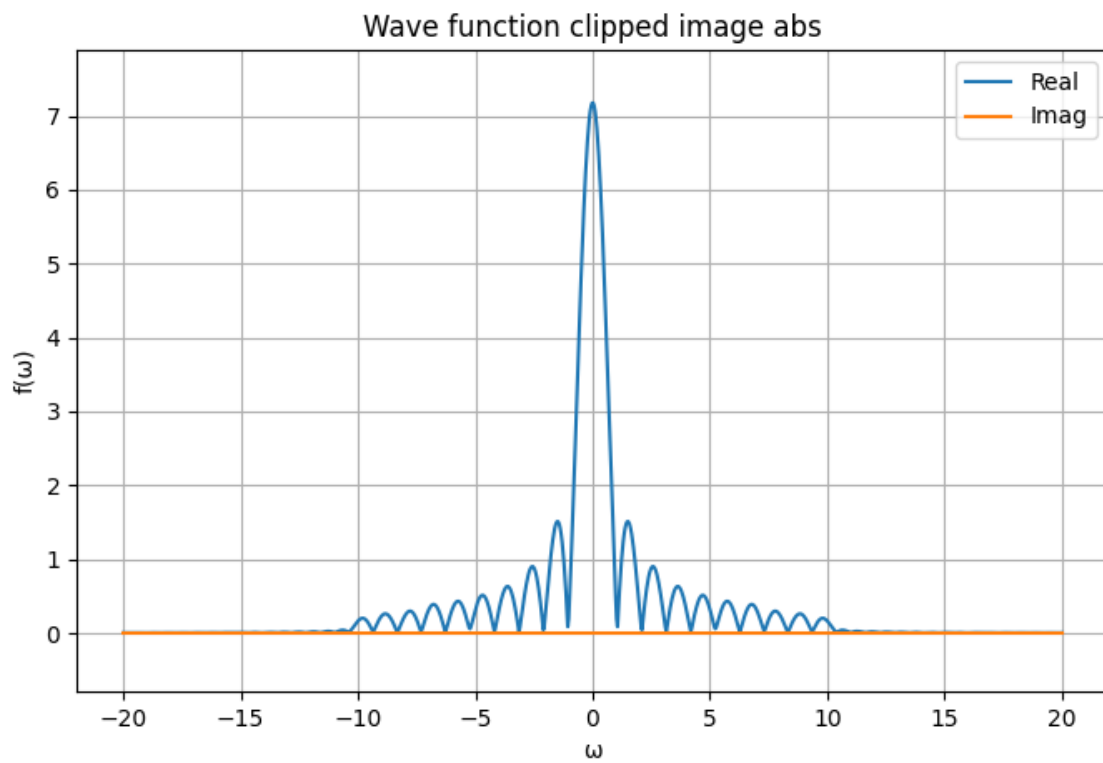


Рис. 9: Модуль образа фильтрованной функции $u(t)$ при $\omega_0 = 10$

1.2.6. Исследование влияния частоты среза на фильтрацию

Рассмотрим несколько значений частоты среза ω_0 при $b = 1$ и сравним результаты фильтрации.

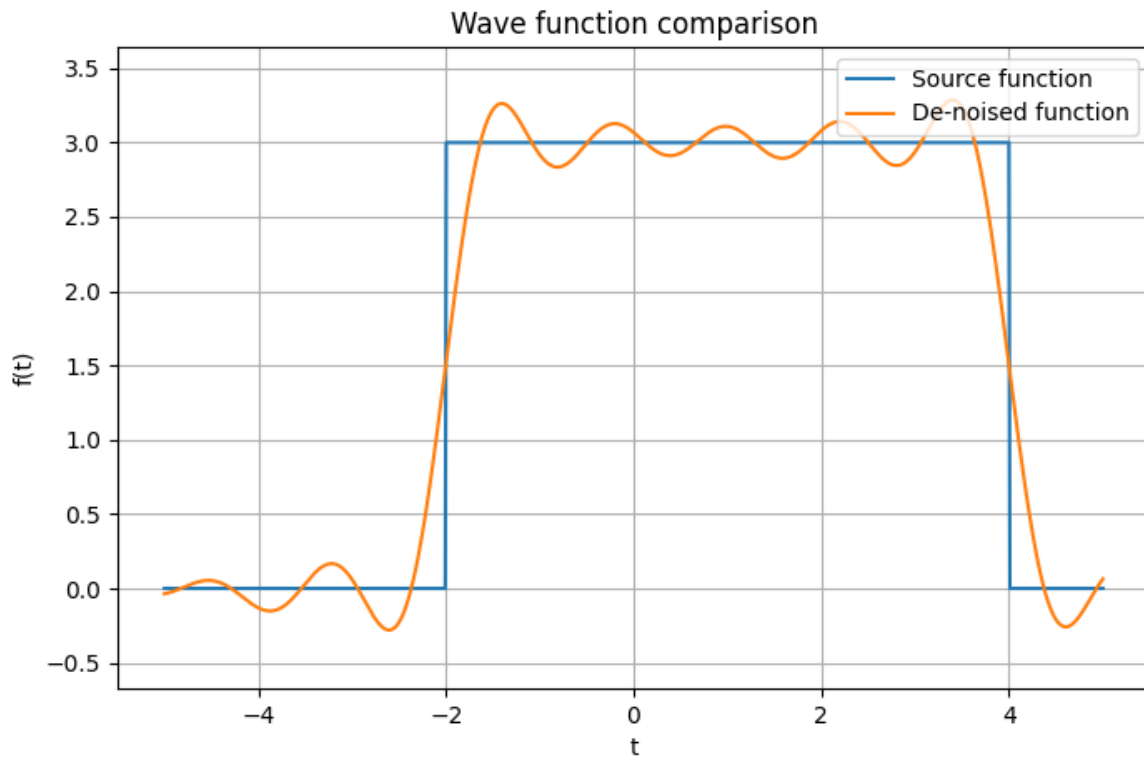


Рис. 10: Исходная и фильтрованная функция при $\omega_0 = 5$

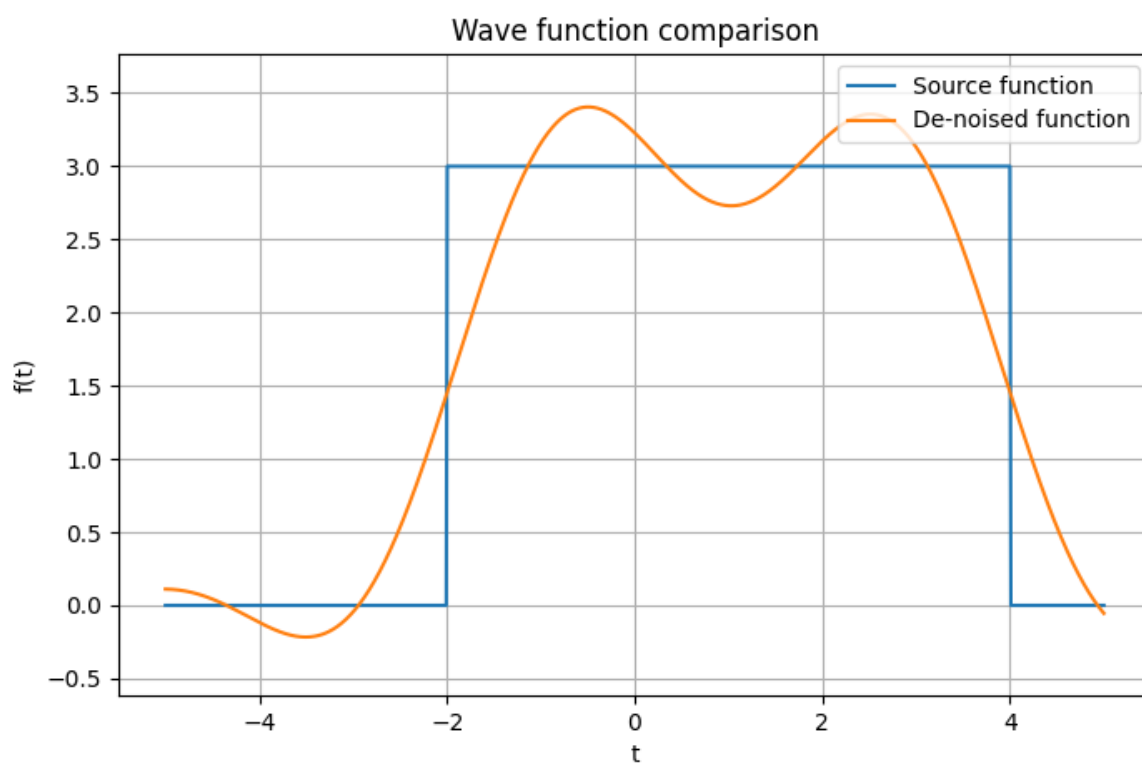


Рис. 11: Исходная и фильтрованная функция при $\omega_0 = 2$

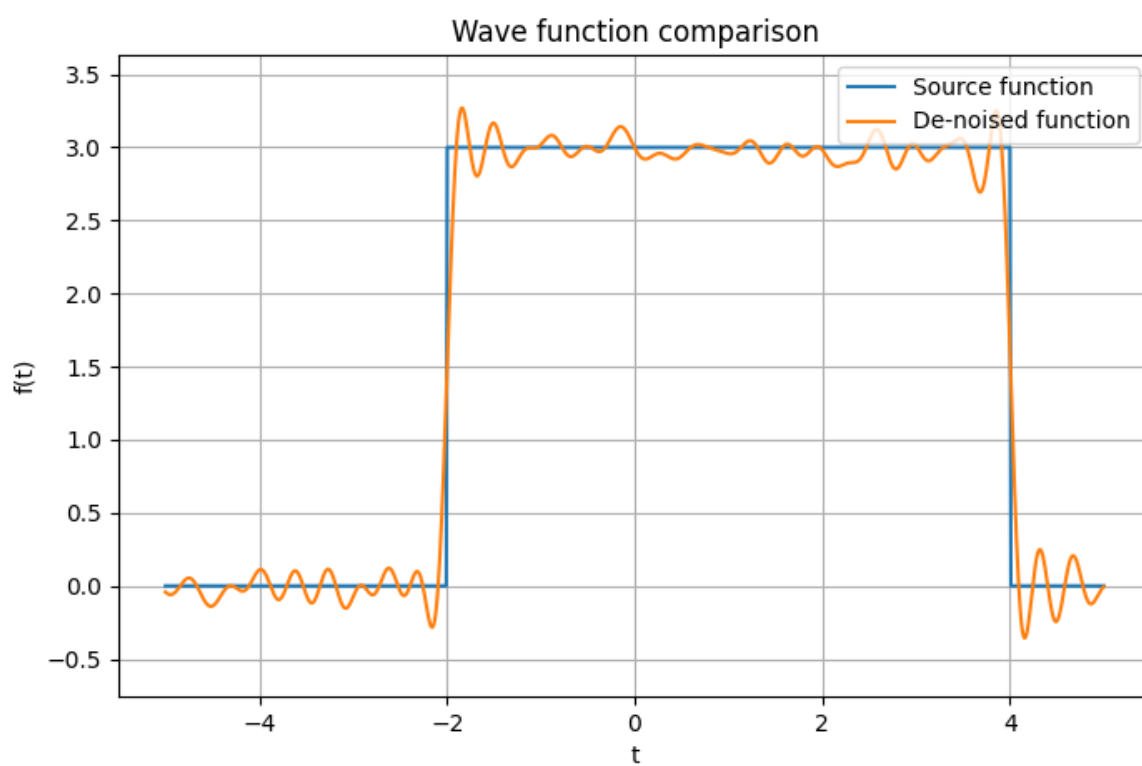


Рис. 12: Исходная и фильтрованная функция при $\omega_0 = 20$

Видим, что при уменьшении ω_0 фильтрация становится более жесткой, что приводит к более сильному удалению шумов, но и к более сильному искажению самой функции.

1.2.7. Исследование влияния параметра b на фильтрацию

Параметр b влияет на амплитуду шумов в зашумленной функции. Таким образом, при его увеличении шумы становятся более заметными.

Рассмотрим несколько значений параметра b при $\omega_0 = 10$ и сравним результаты фильтрации.

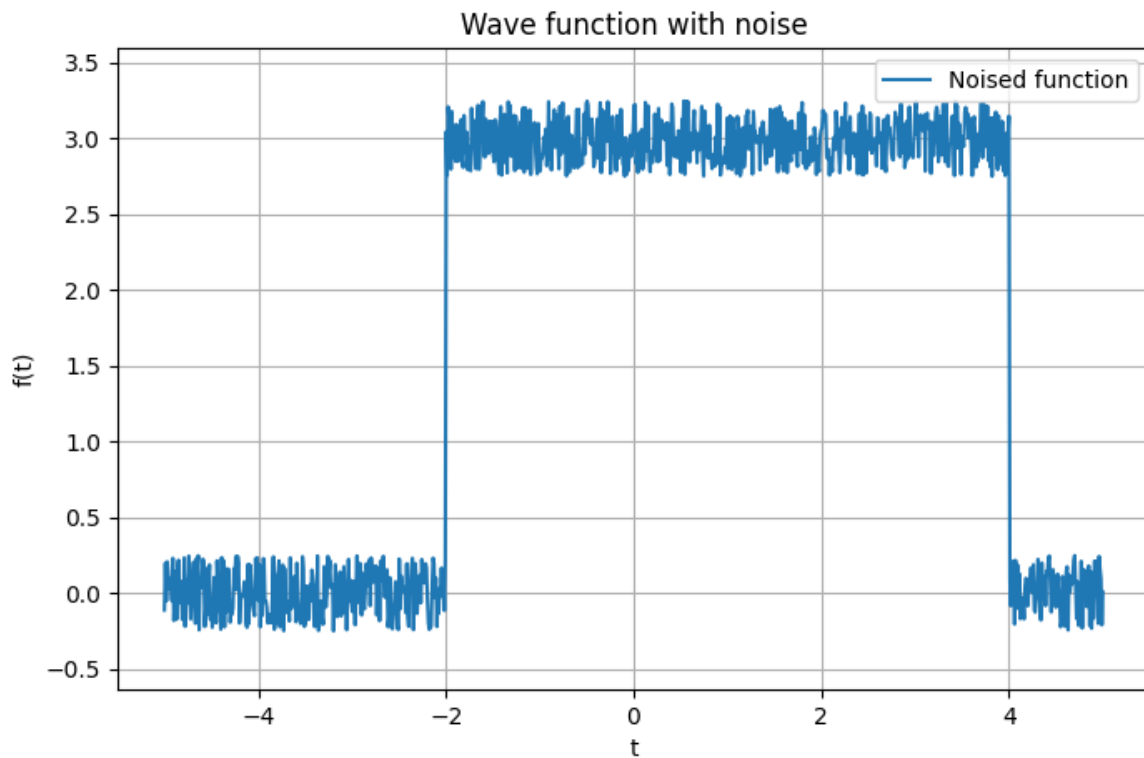


Рис. 13: Функция $u(t)$ с параметрами $b = 0.5$, $c = 0$, $d = 0$

Видно, что при увеличении значения параметра b шумы становятся более заметными, что приводит к более сильному искажению функции при фильтрации, но даже при значении $b = 4$, когда исходная функция практически неразличима за шумом, фильтрация все равно убирает шумы и восстанавливает функцию. Это связано с тем, что муш имеет только высокочастотные компоненты, которые удаляются при фильтрации.

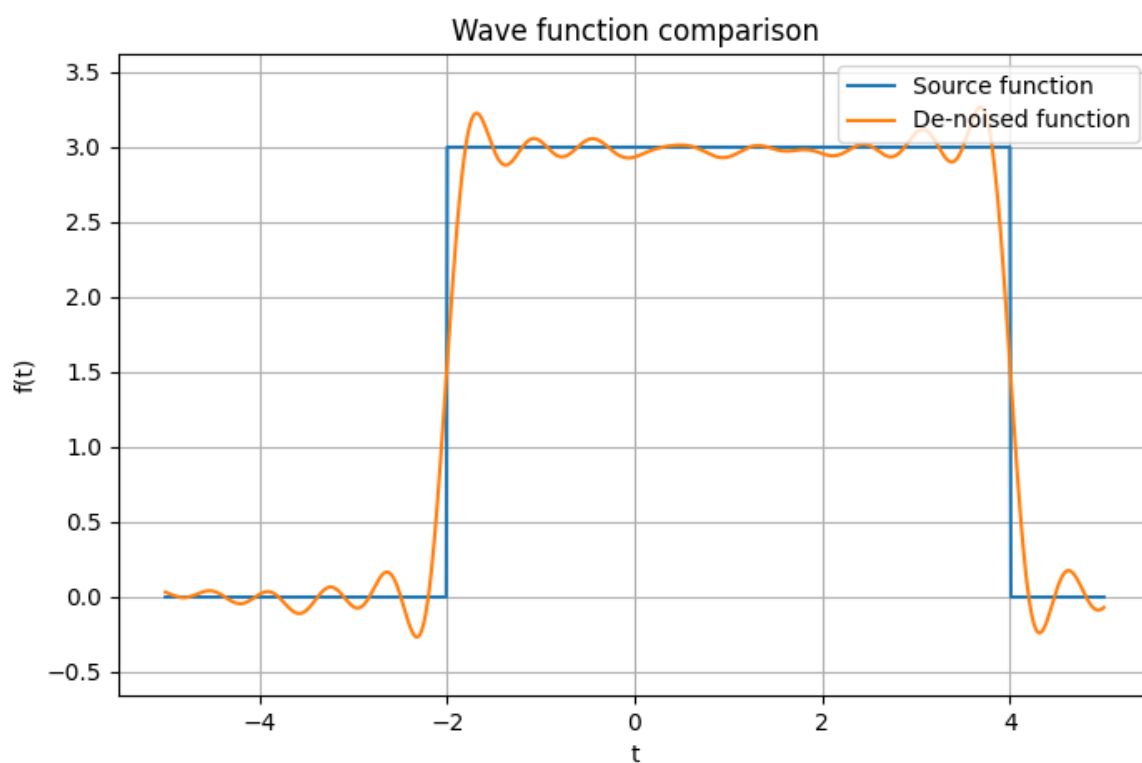


Рис. 14: Исходная и фильтрованная функция при $\omega_0 = 10$

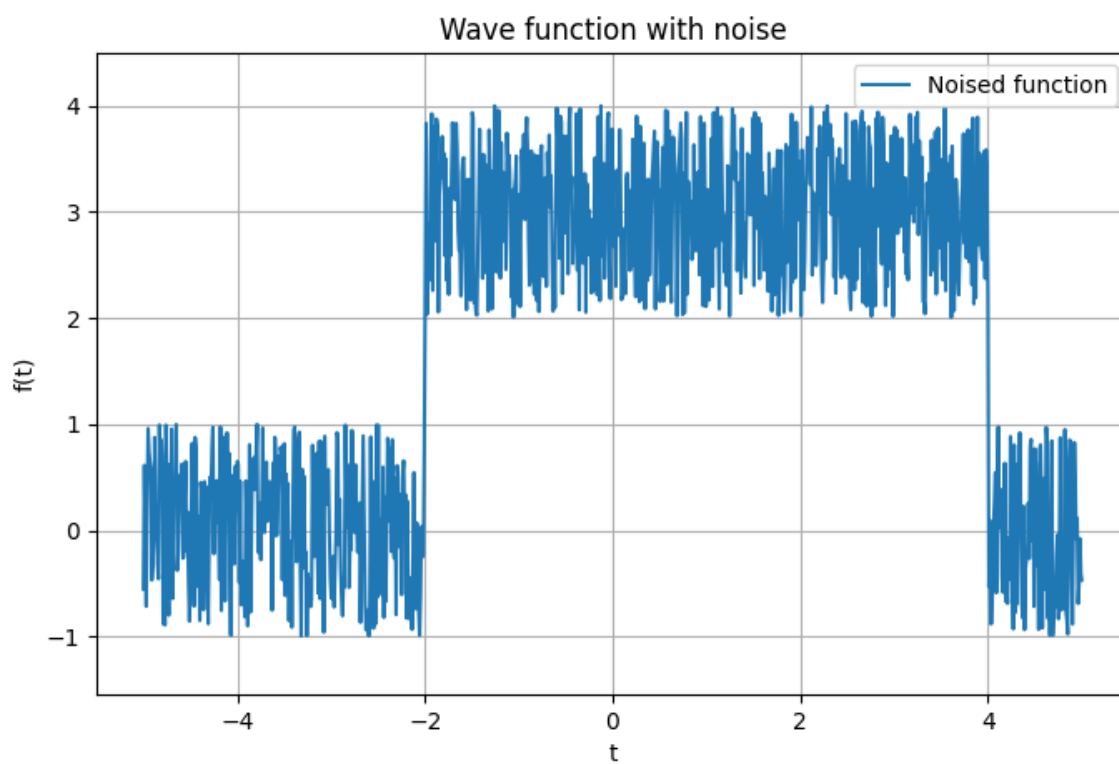


Рис. 15: Функция $u(t)$ с параметрами $b = 2$, $c = 0$, $d = 0$

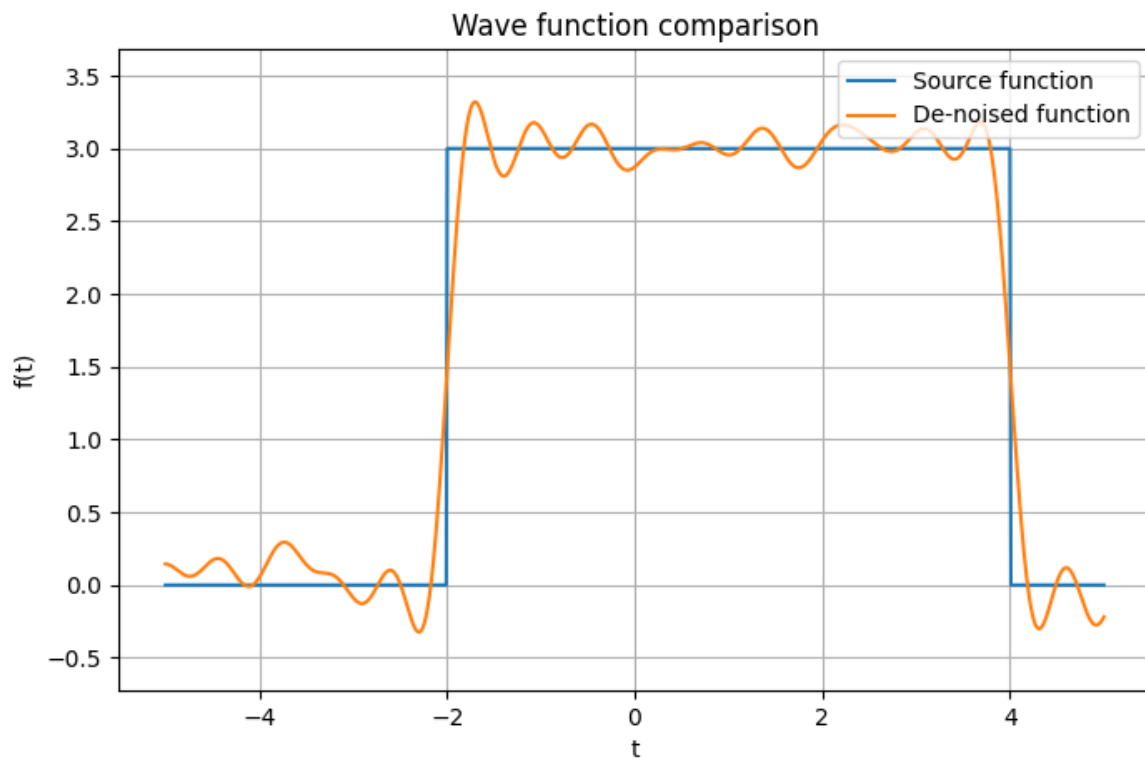


Рис. 16: Исходная и фильтрованная функция при $\omega_0 = 10$

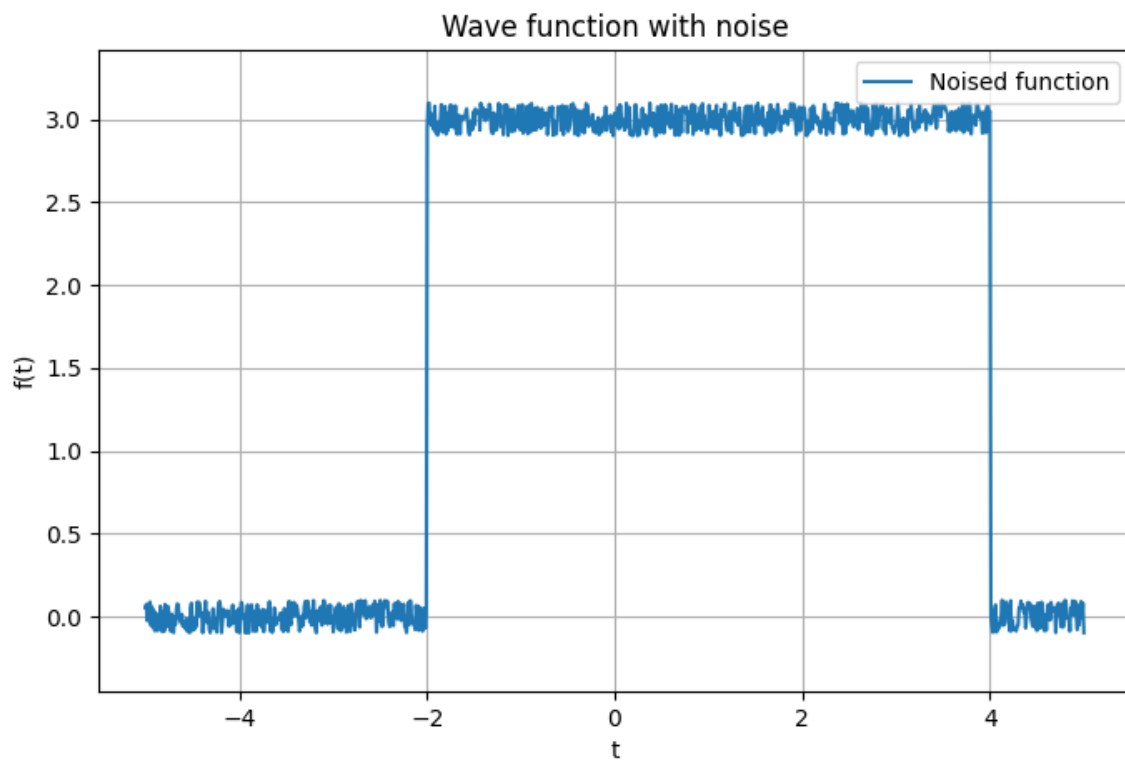


Рис. 17: Функция $u(t)$ с параметрами $b = 0.2$, $c = 0$, $d = 0$

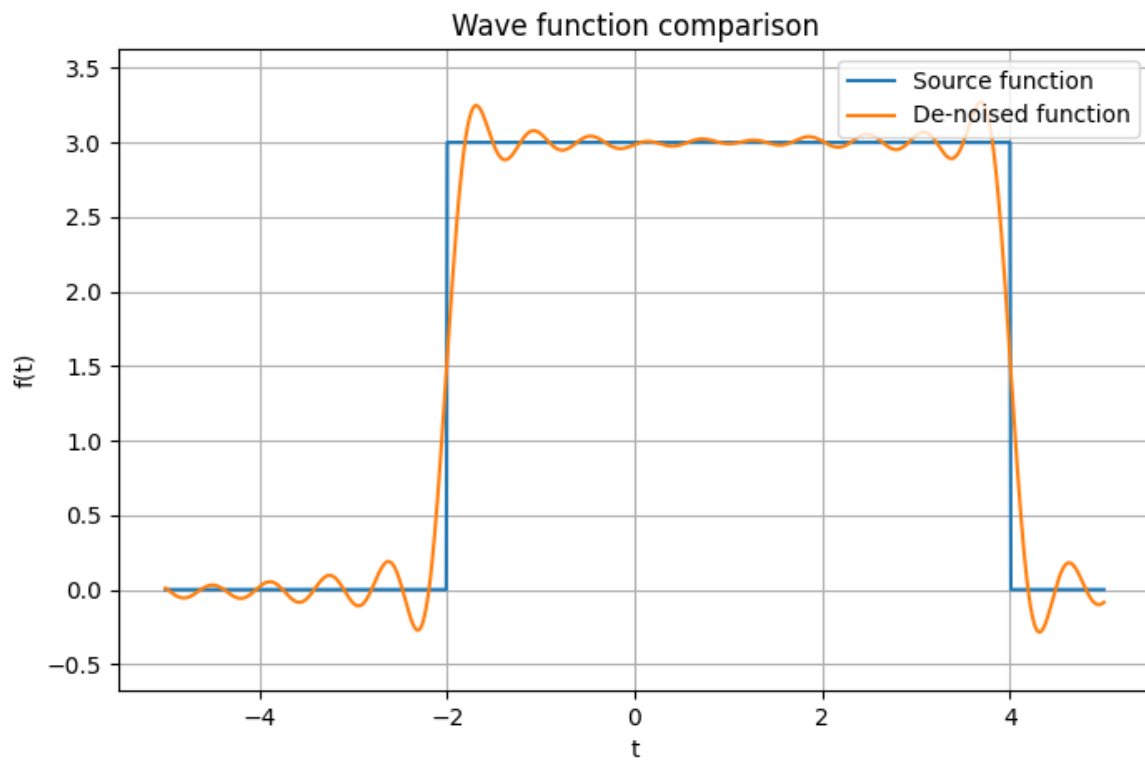


Рис. 18: Исходная и фильтрованная функция при $\omega_0 = 10$

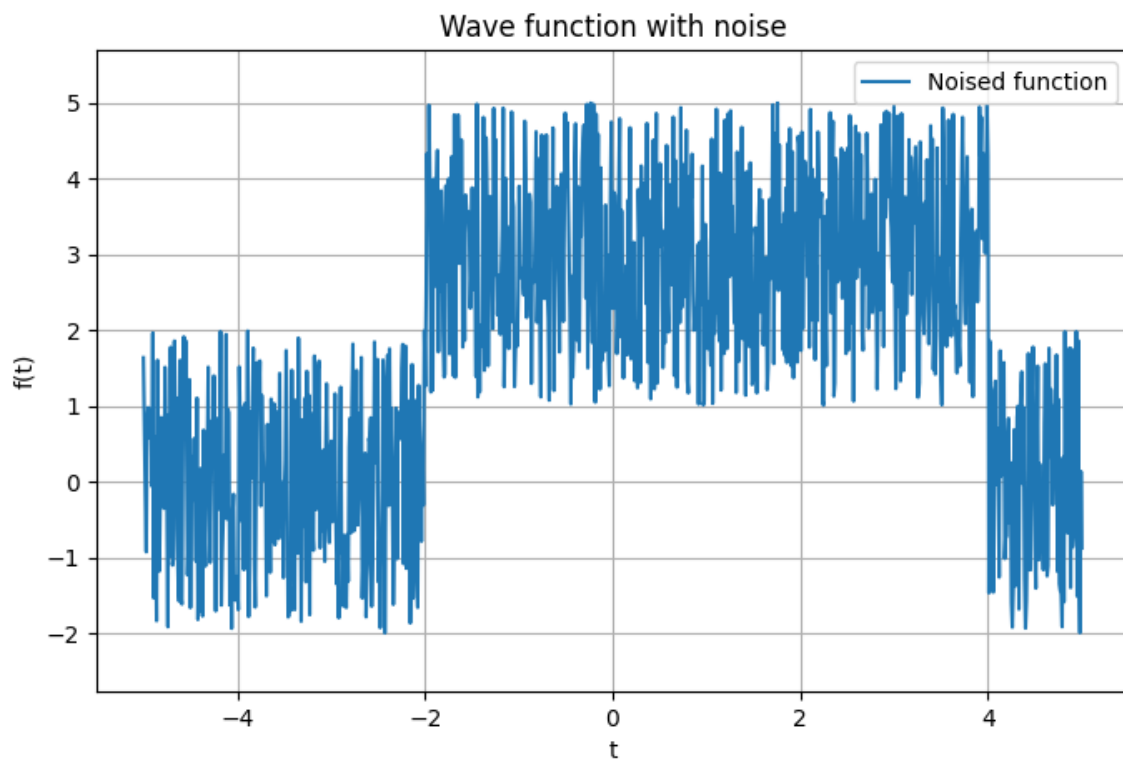


Рис. 19: Функция $u(t)$ с параметрами $b = 4$, $c = 0$, $d = 0$

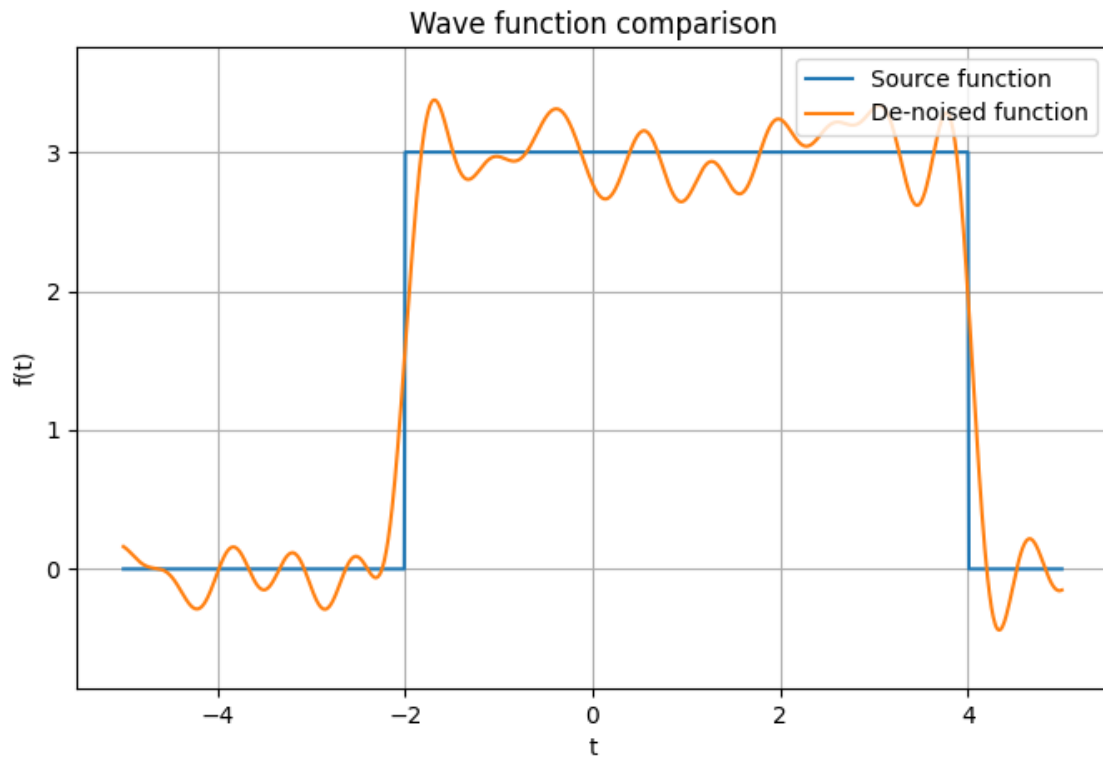


Рис. 20: Исходная и фильтрованная функция при $\omega_0 = 10$

1.3. Убираем специфические частоты

1.3.1. Рассматриваемая функция

Рассмотрим функцию $g(t)$ при параметрах $a = 3$, $t_1 = -2$, $t_2 = 4$ (см. рисунок 21) и ее *зашумленную* версию $u(t)$ с параметрами $b = 0.5$, $c = 0.8$, $d = 8$ (см. рисунок 22). на промежутке $[-T/2, T/2]$ с $T = 10$.

1.3.2. Графики рассматриваемой и зашумленной функции

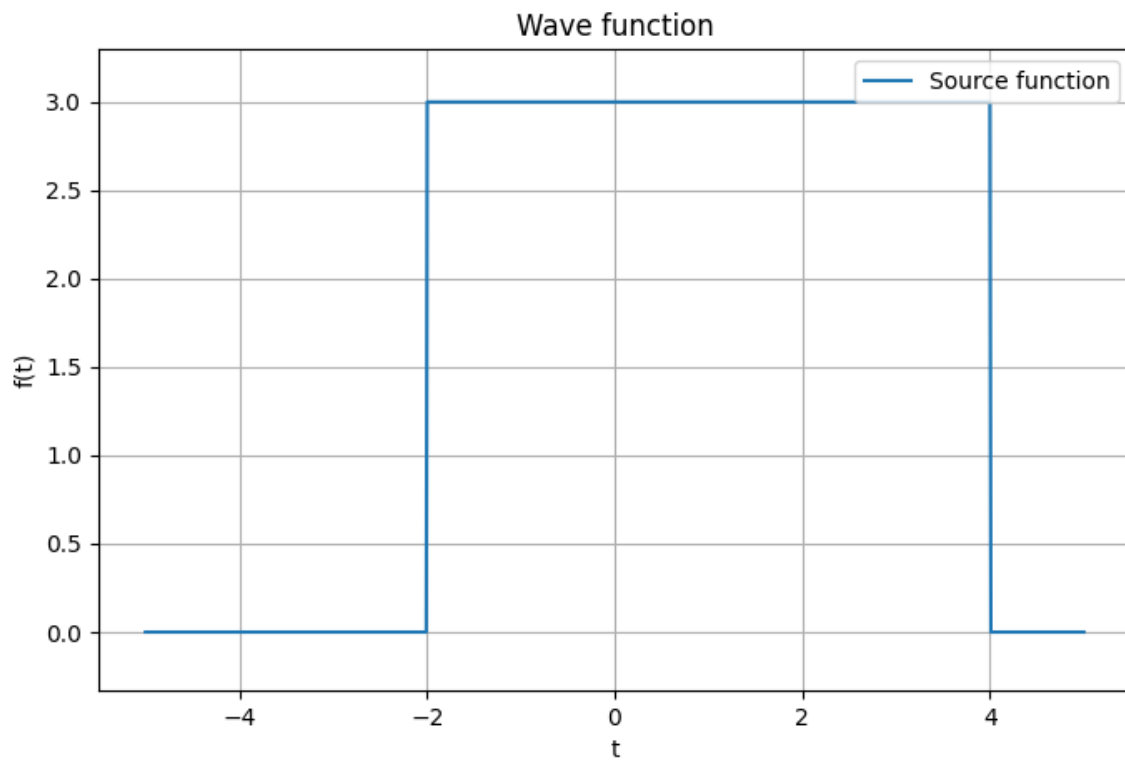


Рис. 21: Функция $g(t)$ с параметрами $a = 3$, $t_1 = -2$, $t_2 = 4$

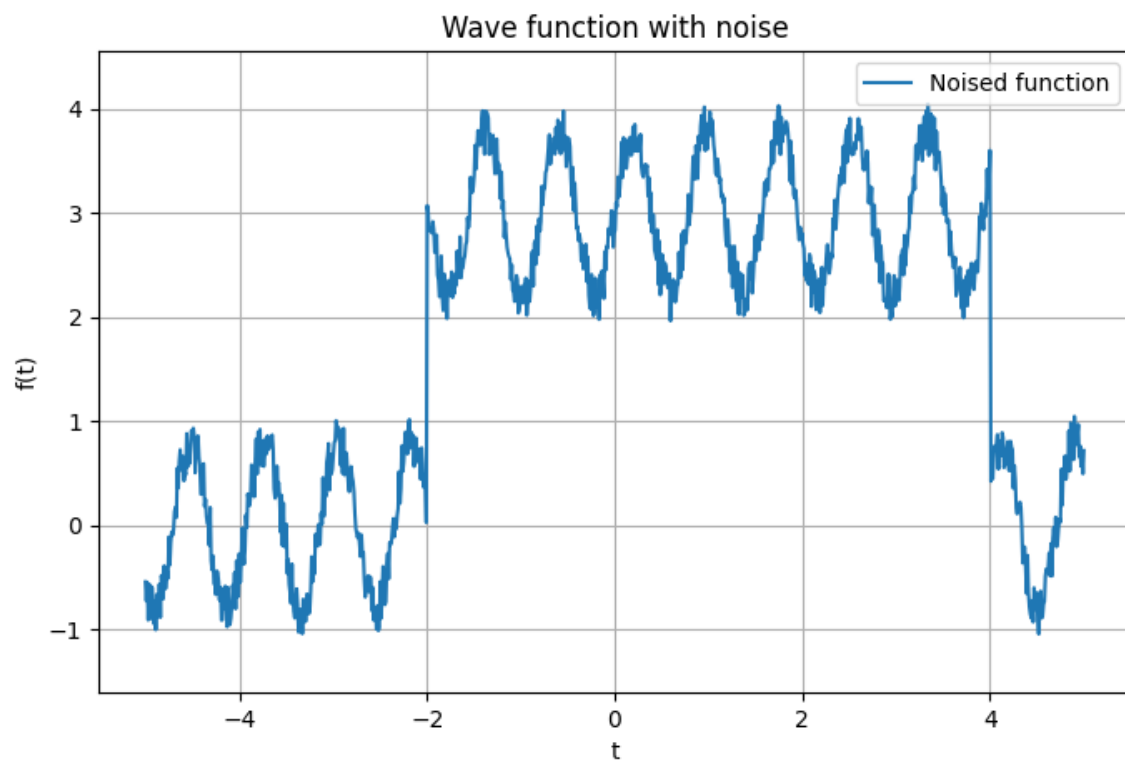


Рис. 22: Функция $u(t)$ с параметрами $b = 0.5$, $c = 0.8$, $d = 8$

Теперь в функции присутствуют обе компоненты шумов – случайная и гармоническая. Для начала, попробуем избавиться от случайной компоненты уже известным образом.

1.3.3. Нахождение образов исходной и зашумленной функций

Найдем образы исходной (см. рисунок 23) и зашумленной (см. рисунок 24) функций с помощью преобразования Фурье.

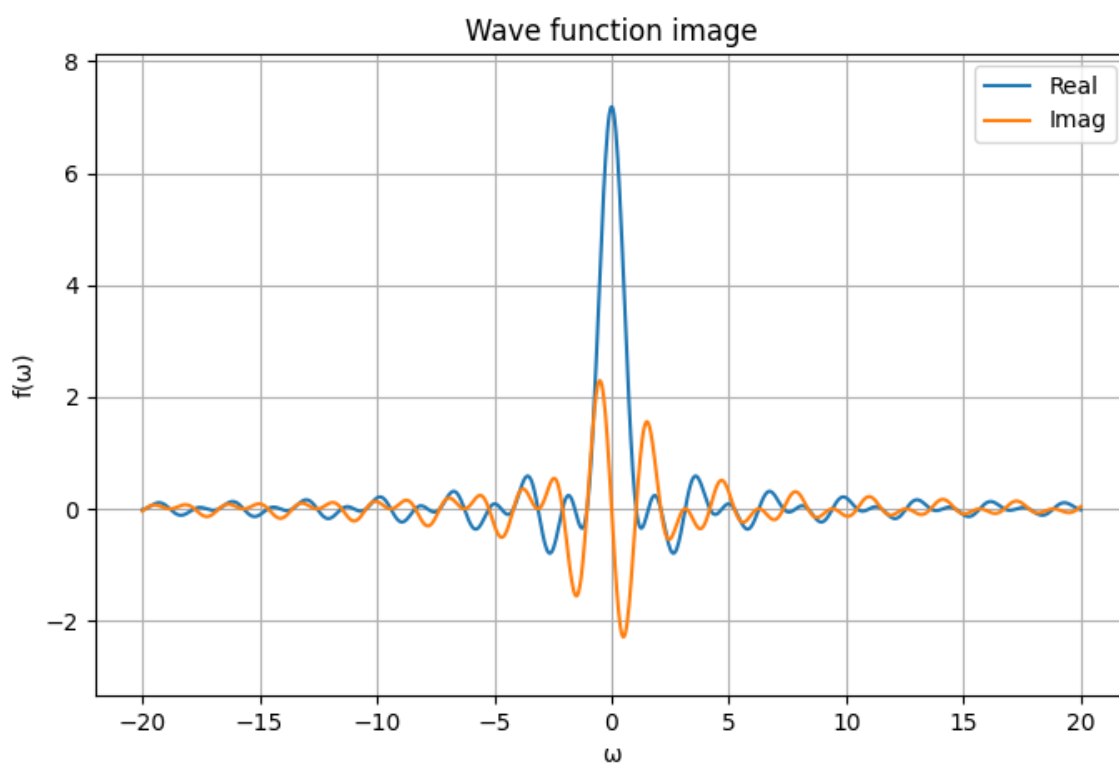


Рис. 23: Образ функция $g(t)$ с параметрами $a = 3$, $t_1 = -2$, $t_2 = 4$

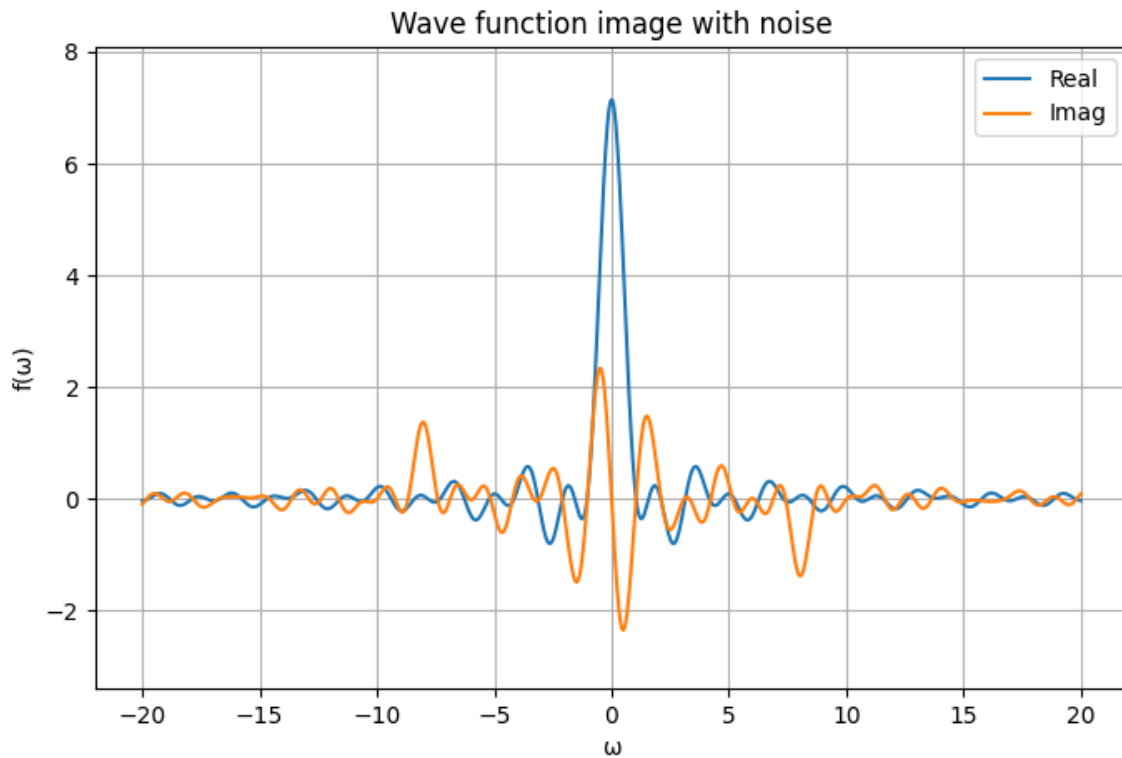


Рис. 24: Образ функция $u(t)$ с параметрами $b = 0.5$, $c = 0.8$, $d = 8$

1.3.4. Фильтрация высоких частот

Обрежем образ функции $u(t)$, убрав компоненты, соответствующие высоким частотам (см. рисунок 25).

Теперь выполним обратное преобразование Фурье (используя обрезанный образ), получив при этом фильтрованную версию функции (см. рисунок 26).

Видим, что соответствующие случайному шуму выбросы исчезли, но в функции все еще присутствует гармоническая составляющая шума. Сравнительные графики исходной и фильтрованной функций представлены на рисунке 27.

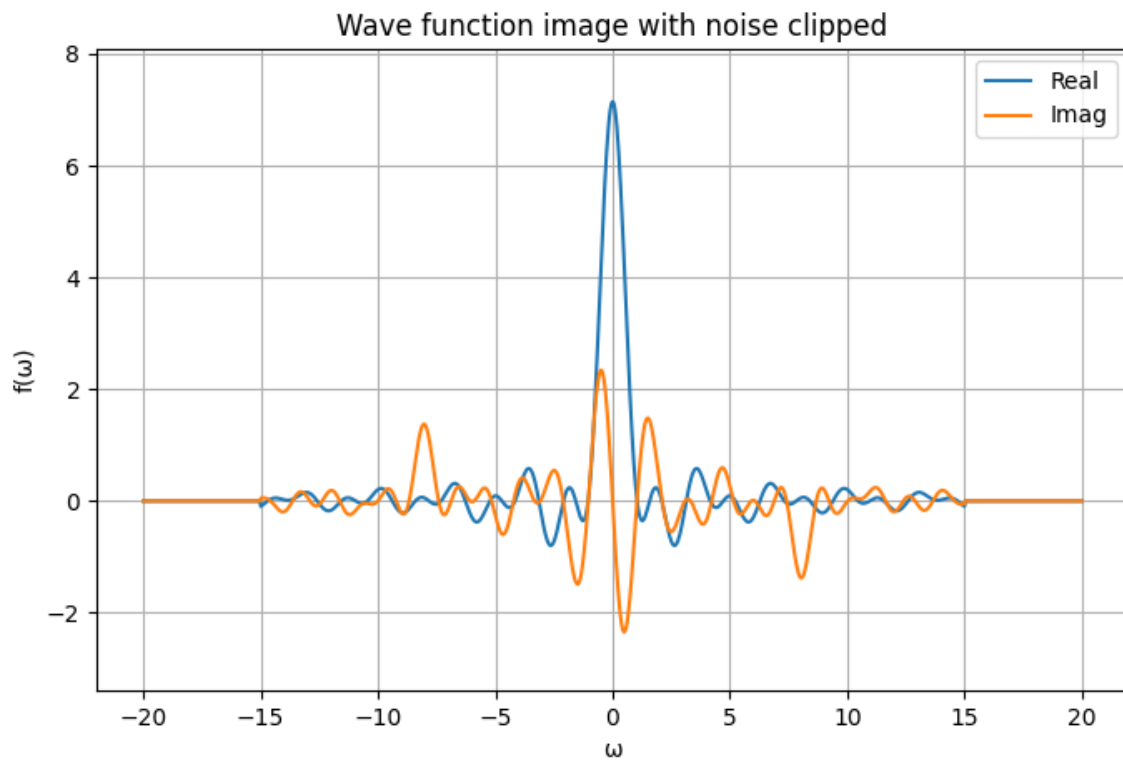


Рис. 25: Обрезанный образ функция $u(t)$ при $\omega_0 = 15$

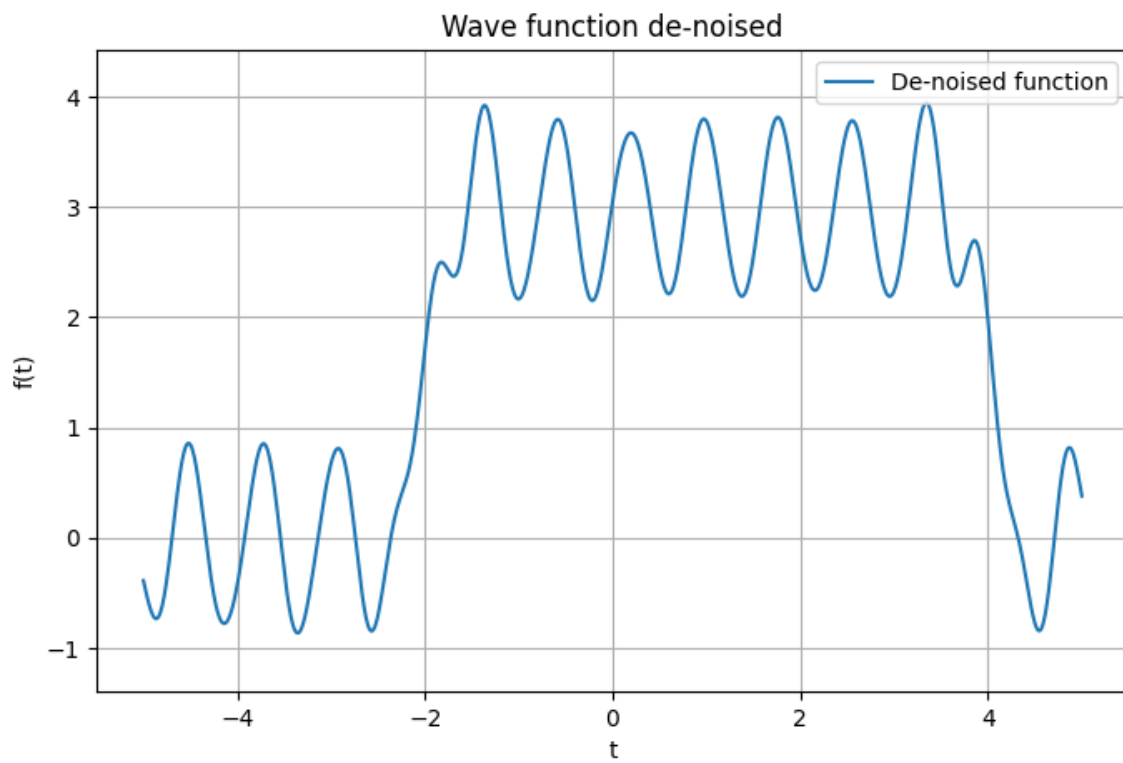


Рис. 26: Фильтрованная функция $u(t)$ при $\omega_0 = 15$

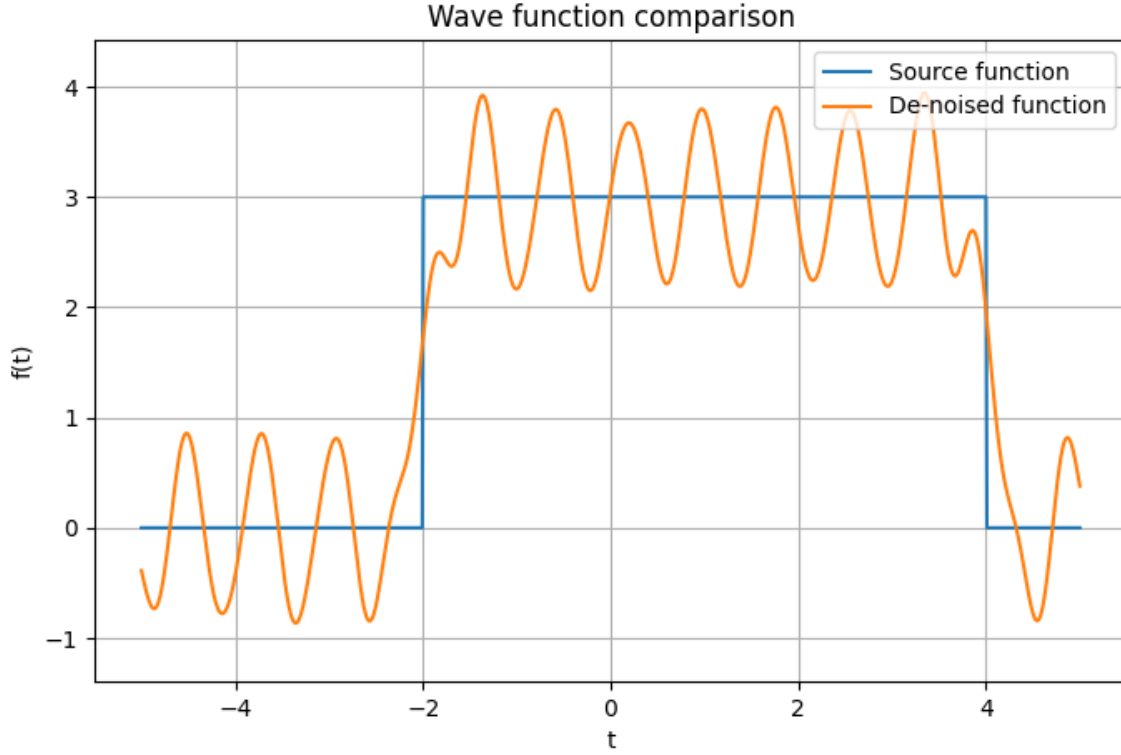


Рис. 27: Исходная и фильтрованная функция при $\omega_0 = 15$

Теперь попробуем избавиться от гармонической составляющей шума. Обратим внимание на образ функции $u(t)$ (см. рисунок 24). На нем четко видны пики при $\omega = 8$, что соотносится с тем, что мы положили значение параметра d , то есть частоты гармонического шума равным 8.

Для того, чтобы убрать эту компоненту шума обнулим образ функции еще и в небольшой окрестности пиков. В данном случае можно задать *обрезанный* образ следующим образом:

$$\hat{U}'(\omega) = \begin{cases} 0, & |\omega| > \omega_0 \\ 0, & \omega \in [8 - \Delta, 8 + \Delta] \\ 0, & \omega \in [-8 - \Delta, -8 + \Delta] \\ \hat{U}(\omega), & otherwise \end{cases} \quad (3)$$

где Δ – некоторая малая величина, определяющая ширину окрестности пиков. Для примера рассмотрим случай $\Delta = 0.7$.

График полученного образа приведен на рисунке 28.

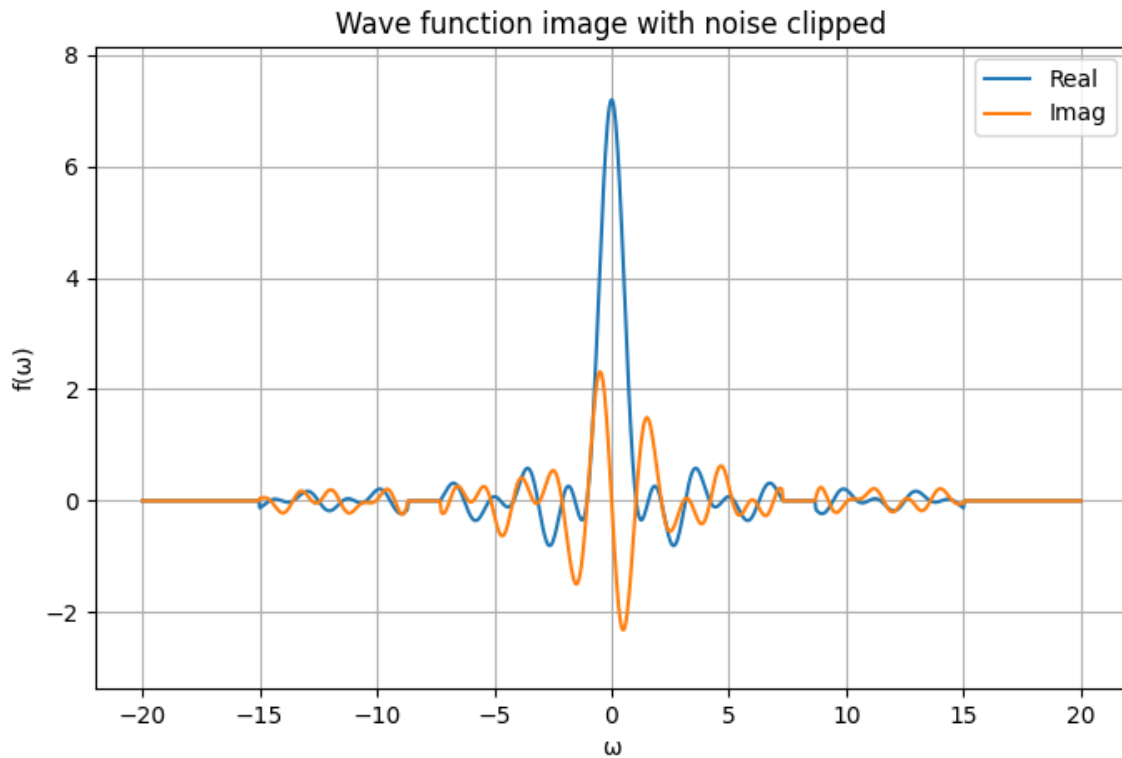


Рис. 28: Обрезанный образ функция $u(t)$ при $\omega_0 = 15$ и $\Delta = 0.7$, $d = 8$

Теперь выполним обратное преобразование Фурье (используя обрезанный образ), получив при этом фильтрованную версию функции (см. рисунок 29).

Сравнительные графики исходной и фильтрованной функций представлены на рисунке 30.

Видно, что гармоническую составляющую шума удалось убрать, но при этом исходная функция тоже несколько исказилась.

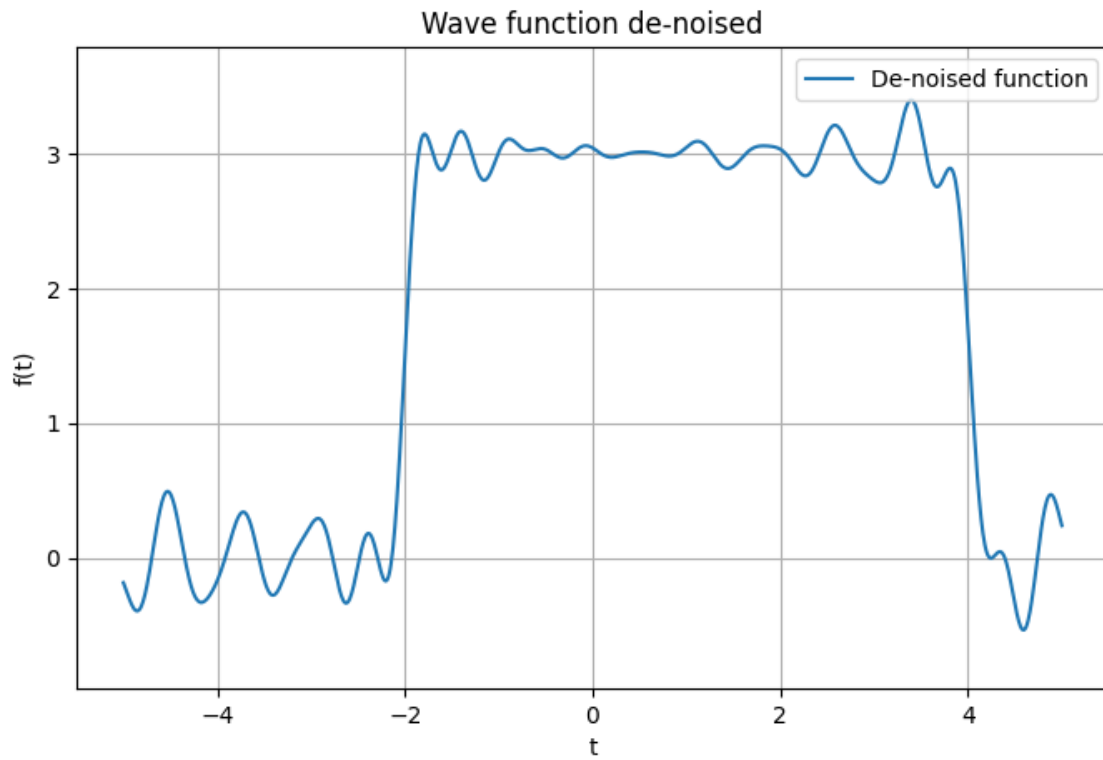


Рис. 29: Фильтрованная функция $u(t)$ при $\omega_0 = 15$ и $\Delta = 0.7$, $d = 8$

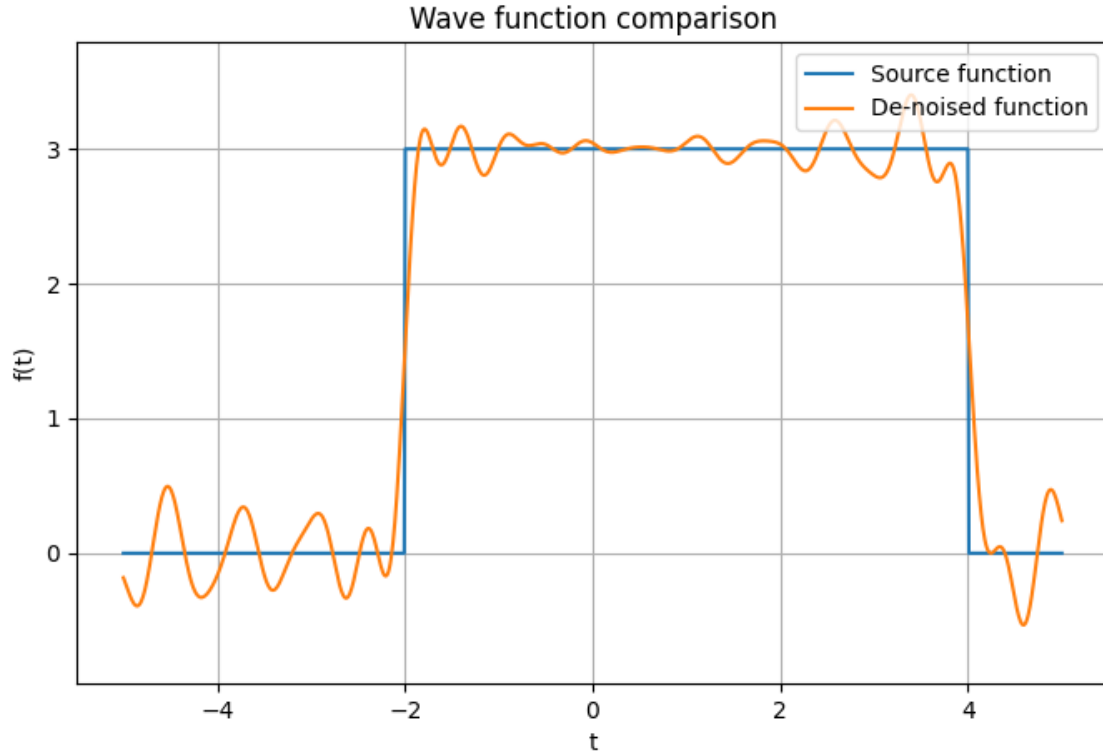


Рис. 30: Исходная и фильтрованная функция при $\omega_0 = 15$ и $\Delta = 0.7$, $d = 8$

1.3.5. Сравнение модулей образов

Сравним модули образов исходной и фильтрованной функций (см. рисунок 31 и 32).

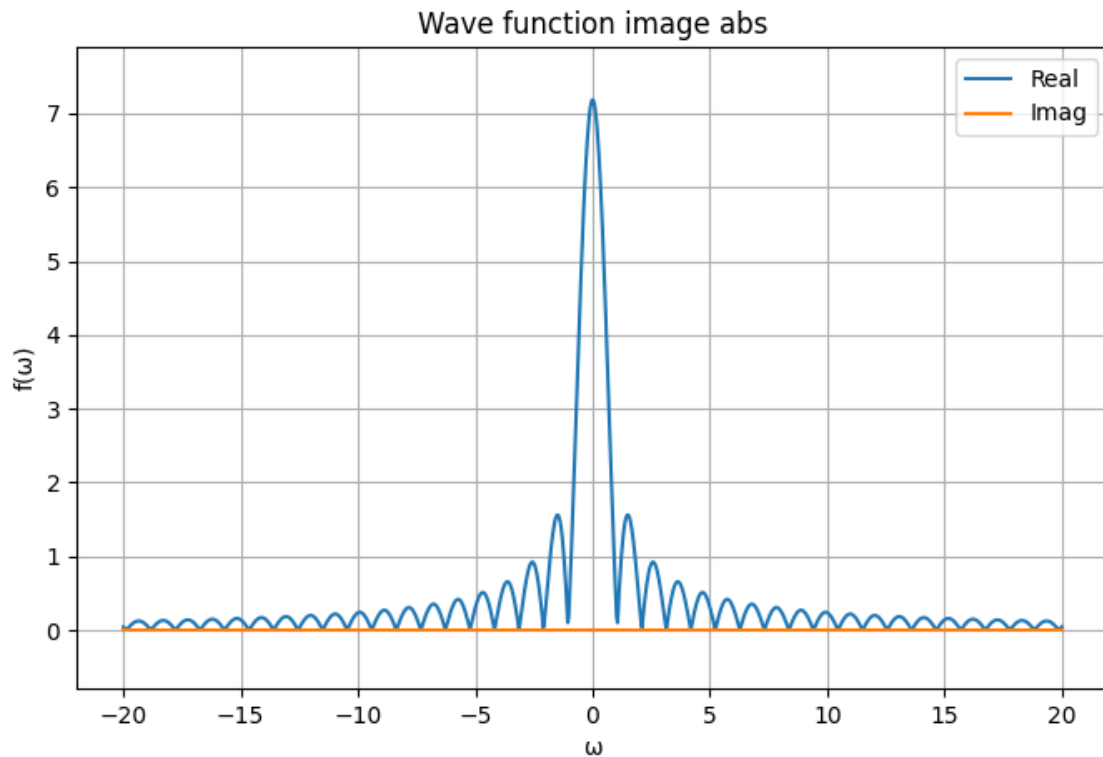


Рис. 31: Модуль образа исходной функции $g(t)$

Видим, что практически частоты выше ω_0 обнулились, при этом низкочастотные компоненты остались практически неизменными. На графике модуля образа *выбросы*, соответствующие гармоническому шуму не заметы, так как они проявляются только в мнимой части образа из-за того, что гармонической составляющей шума соответствует нечетная функция – синус.

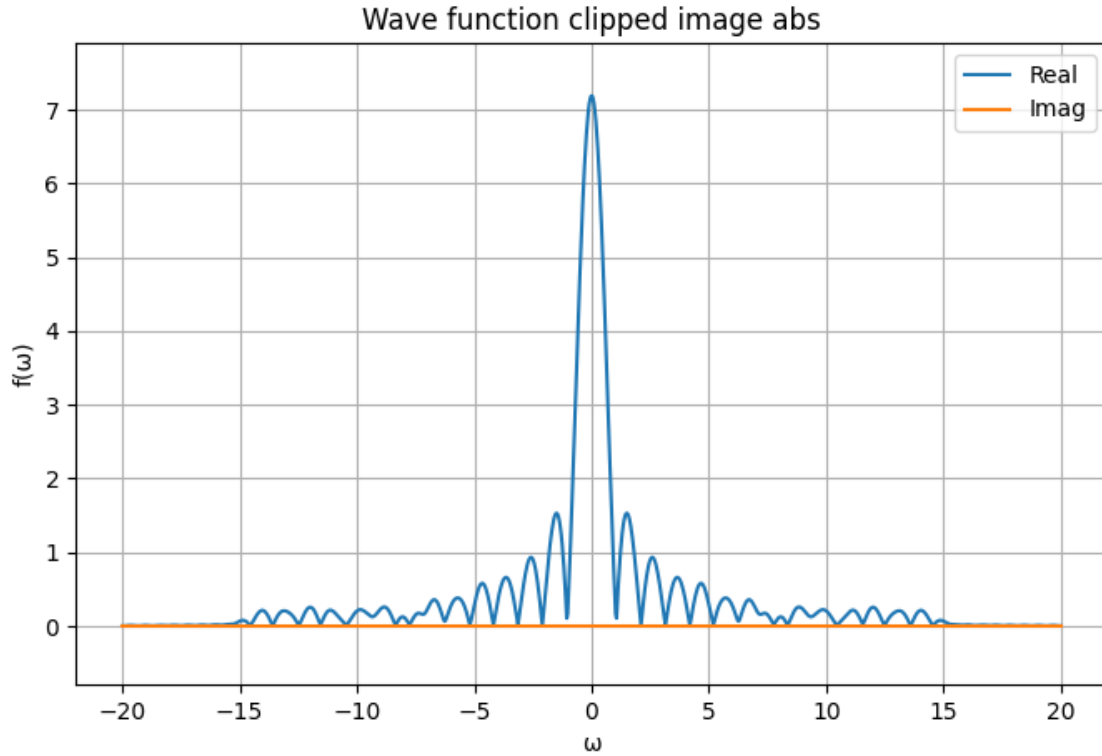


Рис. 32: Модуль образа фильтрованной функции $u(t)$ при $\omega_0 = 15$

Таким образом, удалось убрать обе компоненты шума, не сильно изменив при этом вид исходной функции.

1.3.6. Только гармонический шум

Рассмотрим функцию $g(t)$ при параметрах $a = 3$, $t_1 = -2$, $t_2 = 4$ и ее *зашумленную* версию $u(t)$ с параметрами $b = 0$, $c = 1$, $d = 10$ (см. рисунок 33). на промежутке $[-T/2, T/2]$ с $T = 10$. Такой шум имеет только гармоническую составляющую. Такую составляющую можно убрать, обрезав образ функции в окрестности пика, соответствующего нужной частоте.

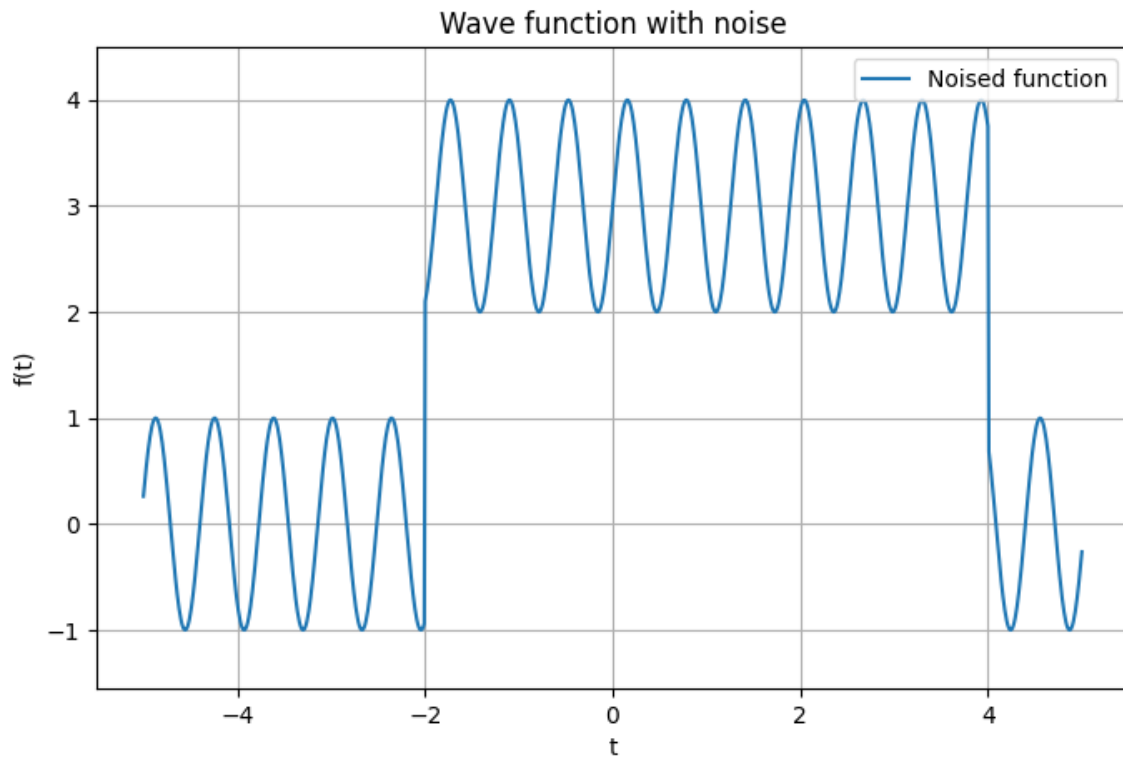


Рис. 33: Функция $u(t)$ с параметрами $b = 0$, $c = 1$, $d = 10$

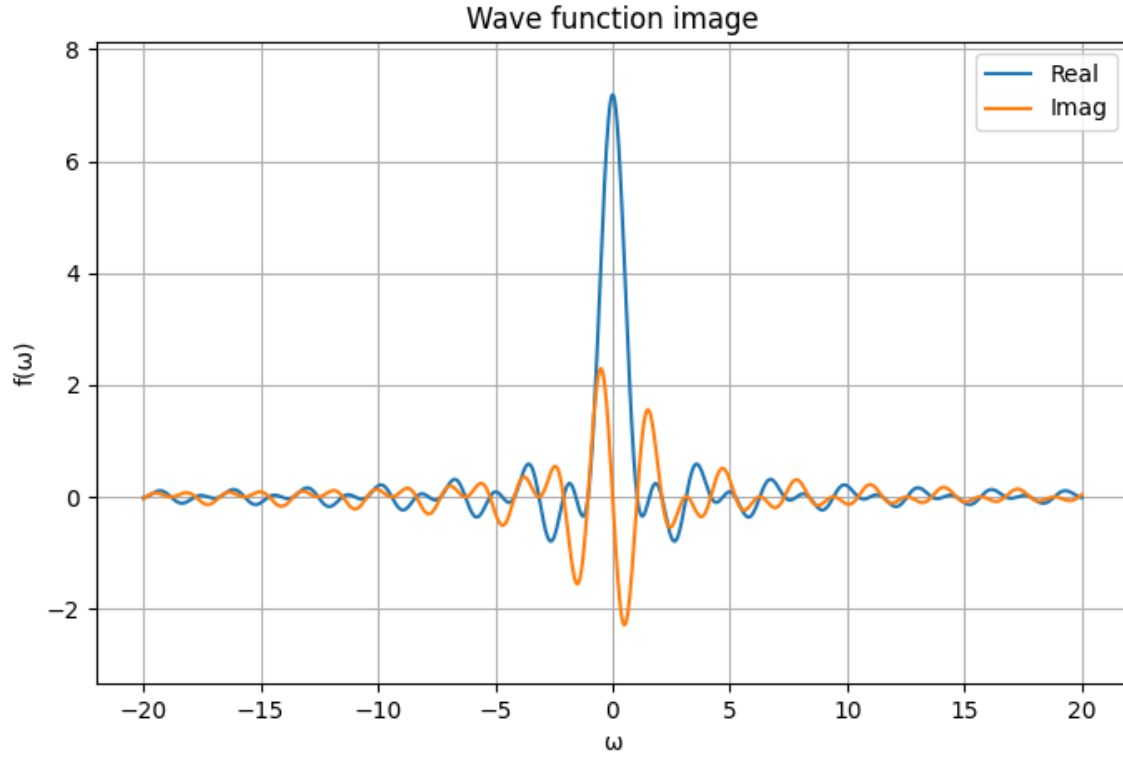


Рис. 34: Образ функция $g(t)$ с параметрами $a = 3$, $t_1 = -2$, $t_2 = 4$

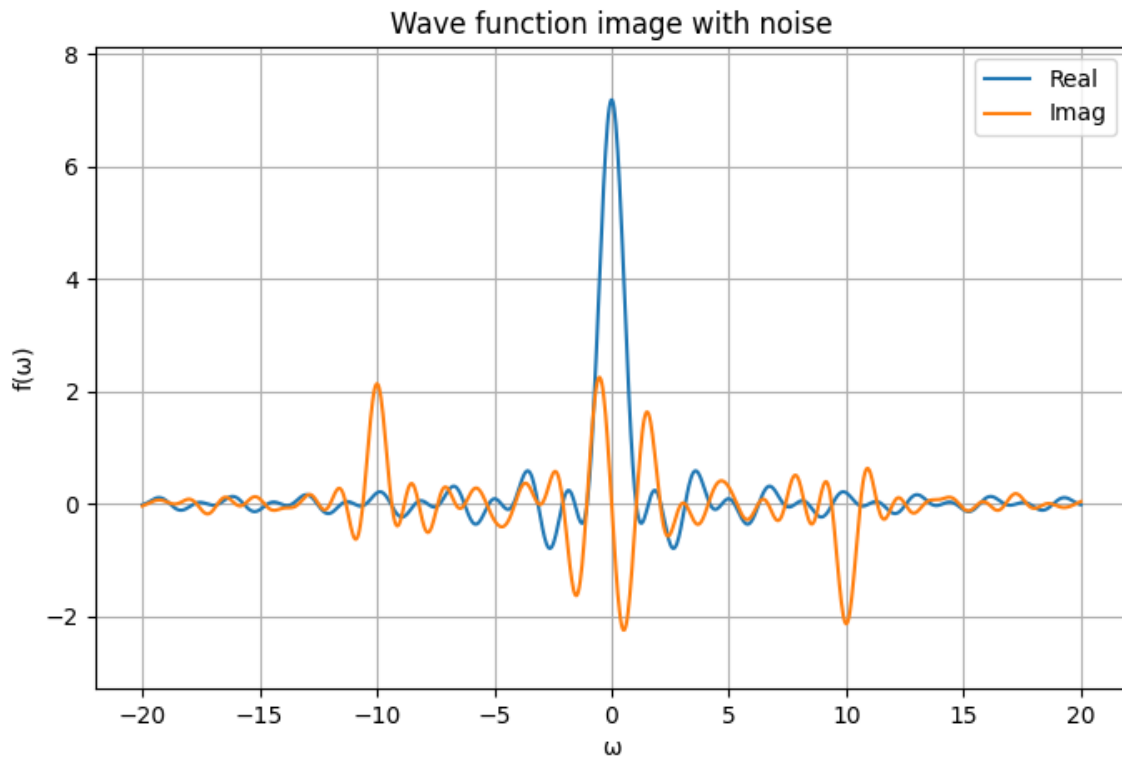


Рис. 35: Образ функция $u(t)$ с параметрами $b = 0$, $c = 1$, $d = 10$

На графике образа функции с таким видом шума видно, что пики при $|\omega| = 10$ (см. рисунок 35)

Обнулим образ в окрестности пика, соответствующего частоте 10. Для примера рассмотрим случай $\Delta = 1$.

Сравнительные графики исходной и фильтрованной функций представлены на рисунке 38.

Видно, что гармоническую составляющую шума удалось убрать, но при этом исходная функция тоже несколько исказилась.

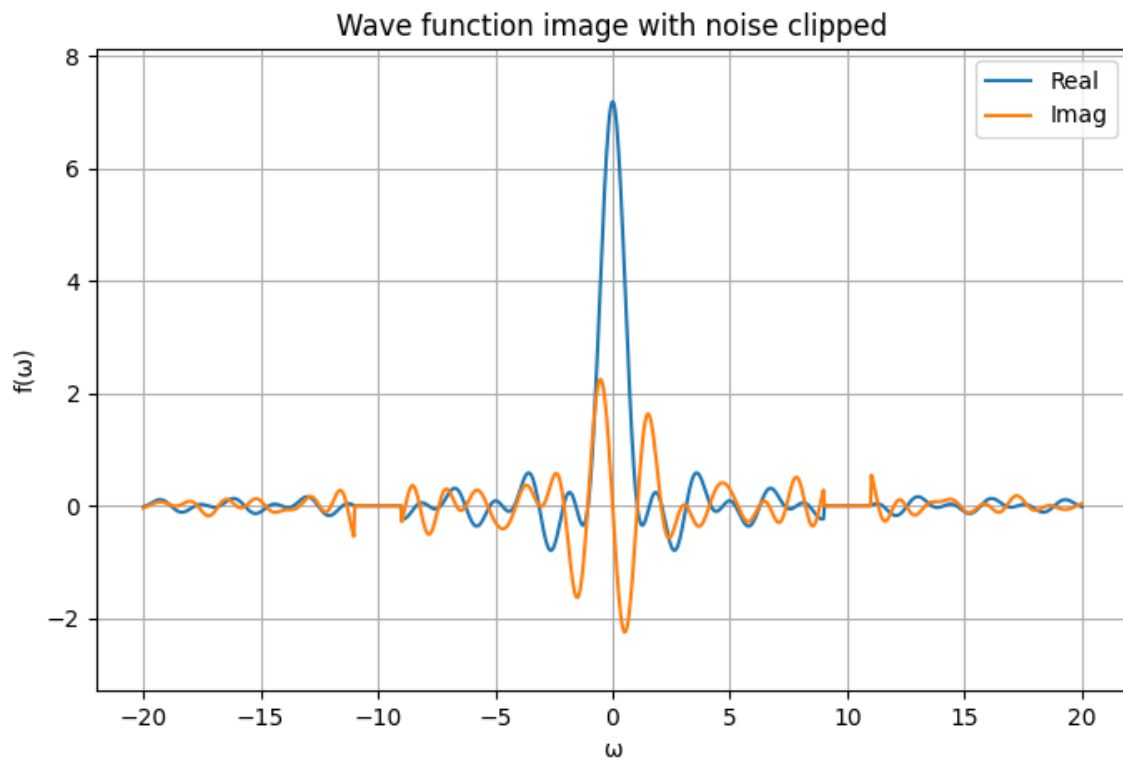


Рис. 36: Обрезанный образ функция $u(t)$

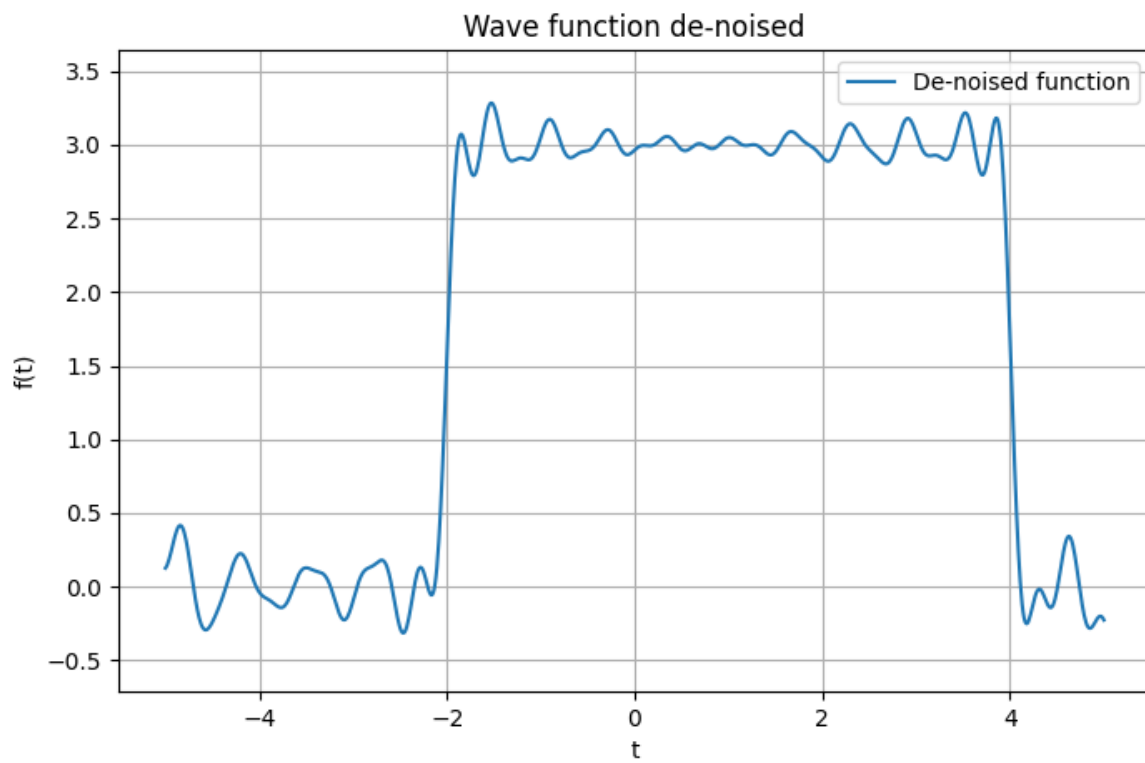


Рис. 37: Фильтрованная функция $u(t)$ при $\omega_0 = 15$

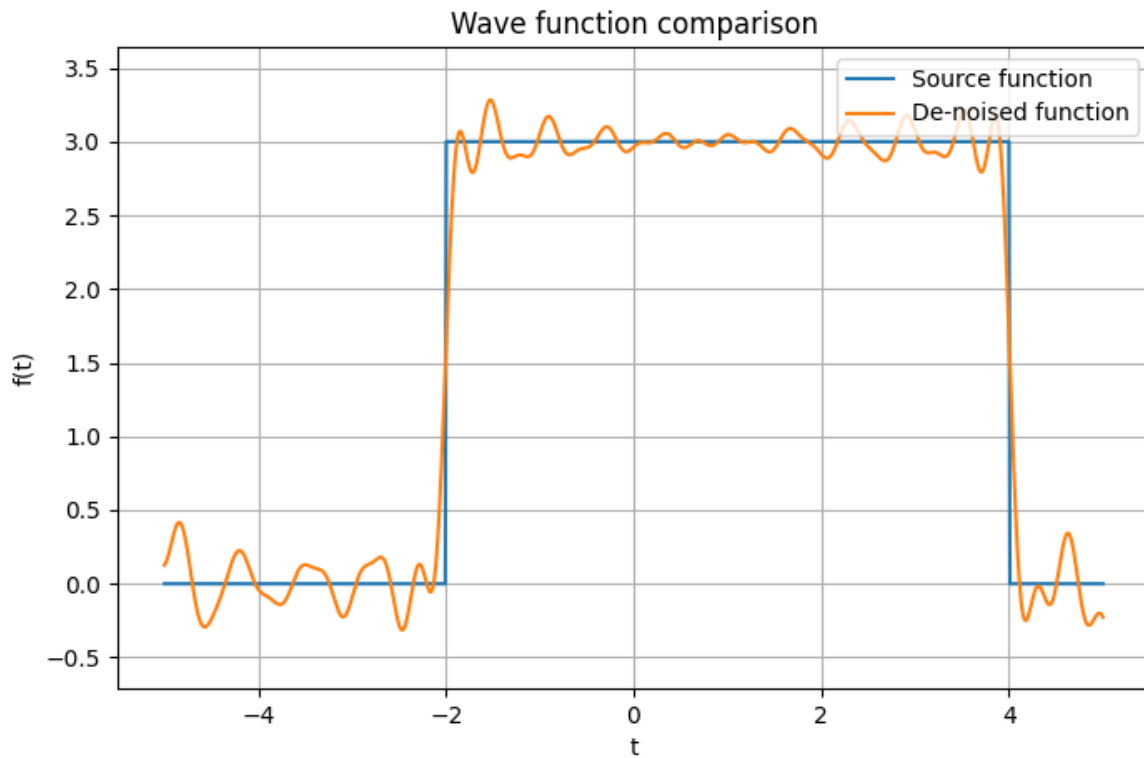


Рис. 38: Исходная и фильтрованная функция при $\omega_0 = 15$

1.3.7. Исследование влияния параметров b , c , d на фильтрацию

Из формулы (2) понятно, что компонента b отвечает за случайный шум, а c и d – за гармонический, при этом c – амплитуда, а d – частота.

Здесь я приведу только сравнительные графики исходной и фильтрованной функций при различных значениях параметров b , c , d .

Очевидно, что, чем меньше шума было внесено в функцию, тем лучше будет получаться его убрать. Но даже при довольно больших значениях параметров b , c , d удалось убрать шум, не сильно исказив исходную функцию.

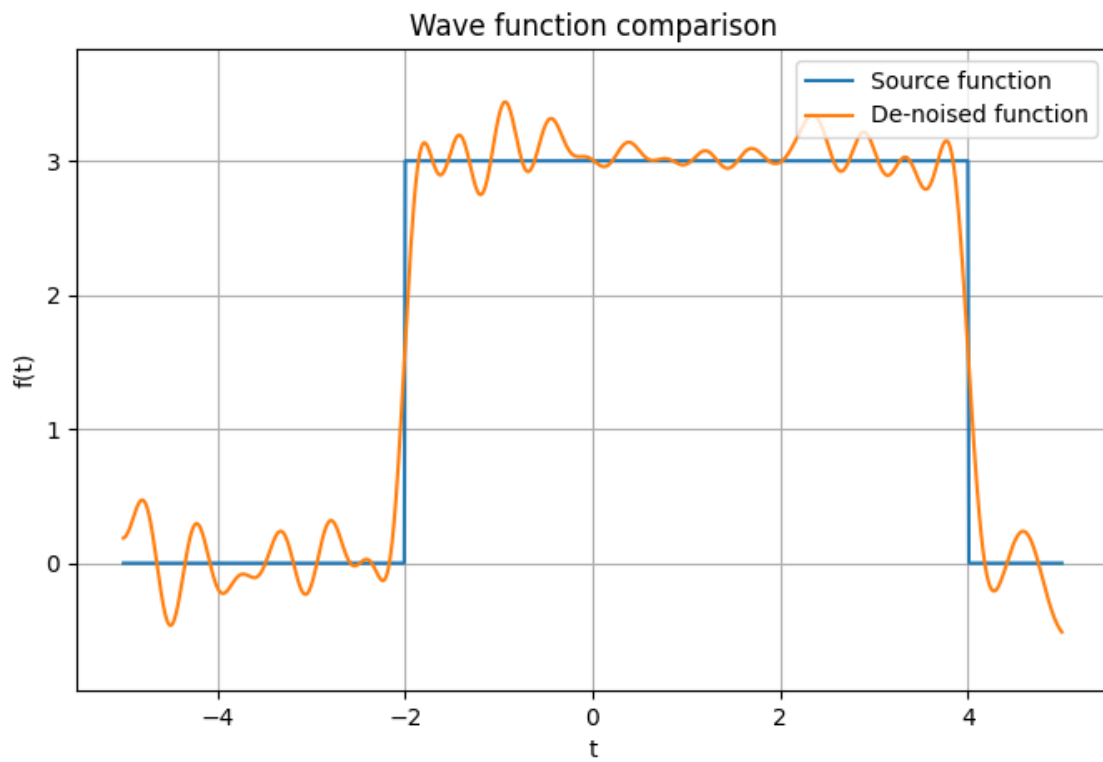


Рис. 39: Исходная и фильтрованная функция при $b = 2$, $c = 1$, $d = 10$

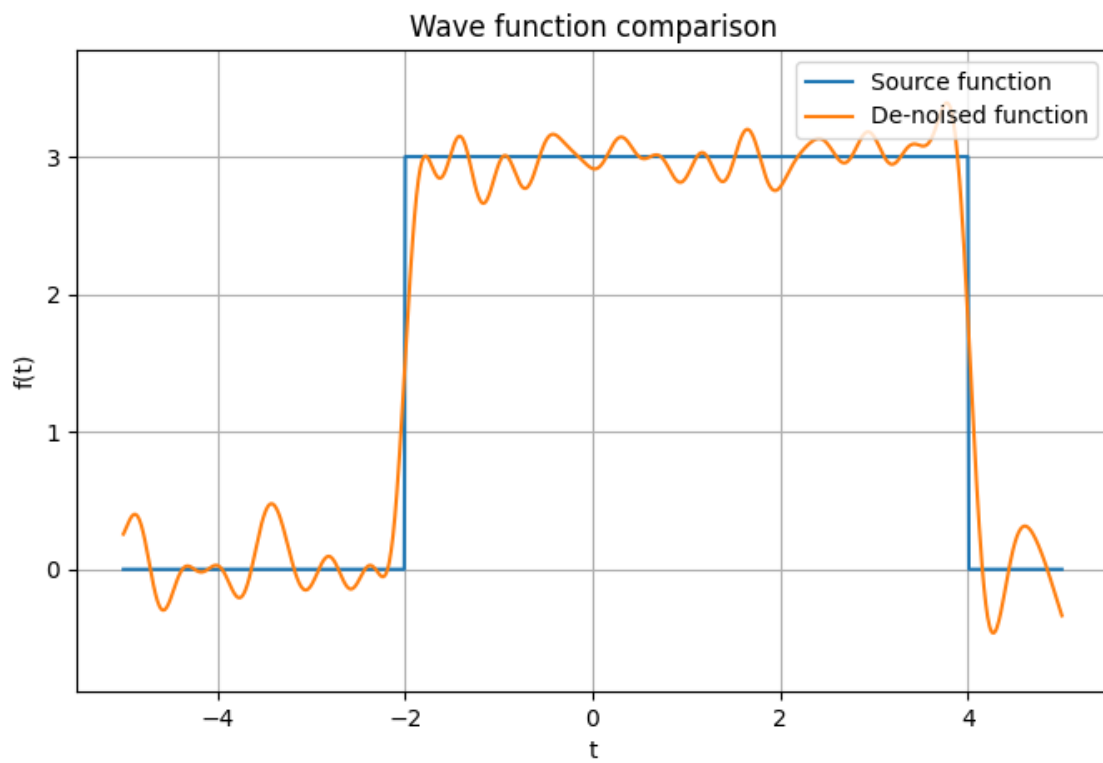


Рис. 40: Исходная и фильтрованная функция при $b = 0.5$, $c = 3$, $d = 10$

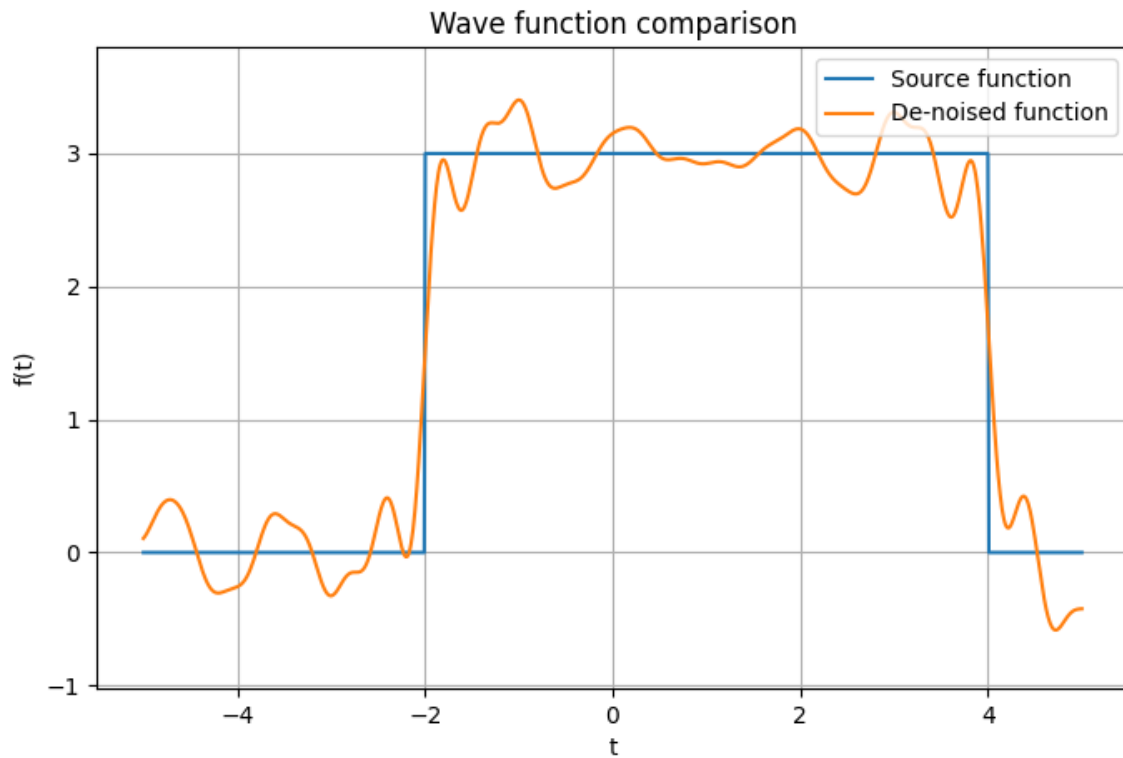


Рис. 41: Исходная и фильтрованная функция при $b = 0.5$, $c = 1$, $d = 5$

1.4. Убираем низкие частоты

1.4.1. Рассматриваемая функция

Рассмотрим функцию $g(t)$ при параметрах $a = 3$, $t_1 = -2$, $t_2 = 4$ (см. рисунок 42) и ее *зашумленную* версию $u(t)$ с параметрами $b = 0.3$, $c = 1$, $d = 10$ (см. рисунок 43). на промежутке $[-T/2, T/2]$ с $T = 10$.

1.4.2. Графики рассматриваемой и зашумленной функции

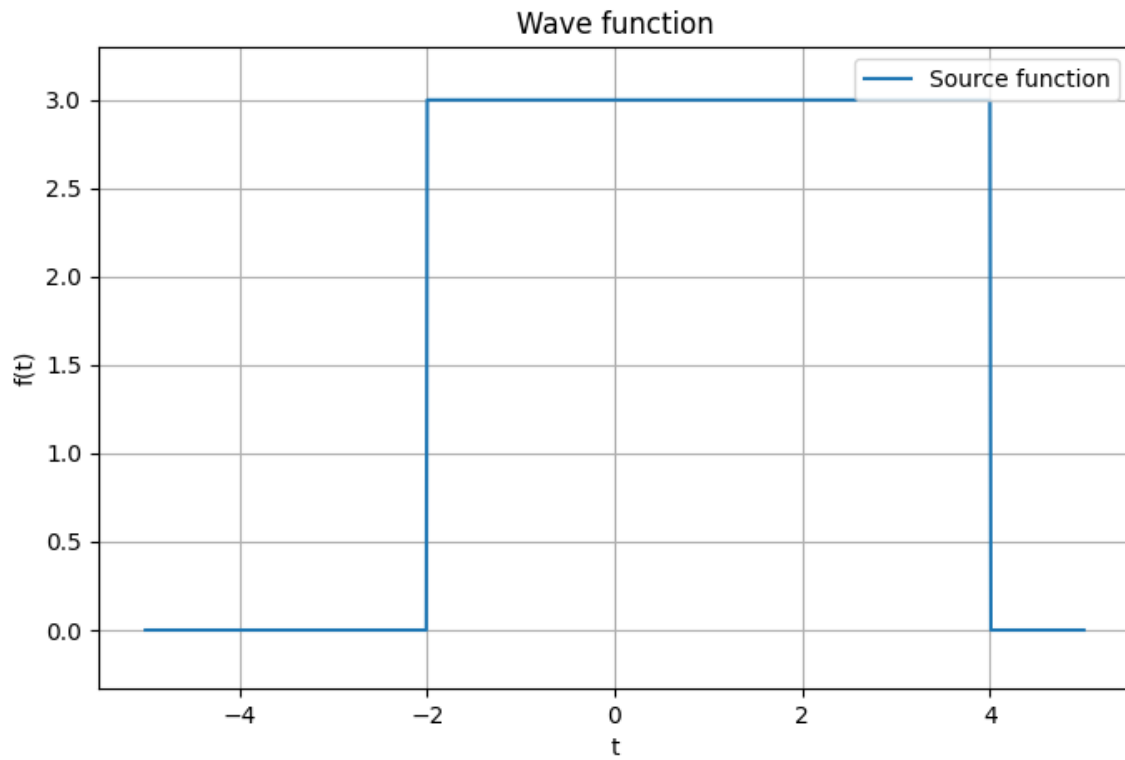


Рис. 42: Функция $g(t)$ с параметрами $a = 3$, $t_1 = -2$, $t_2 = 4$

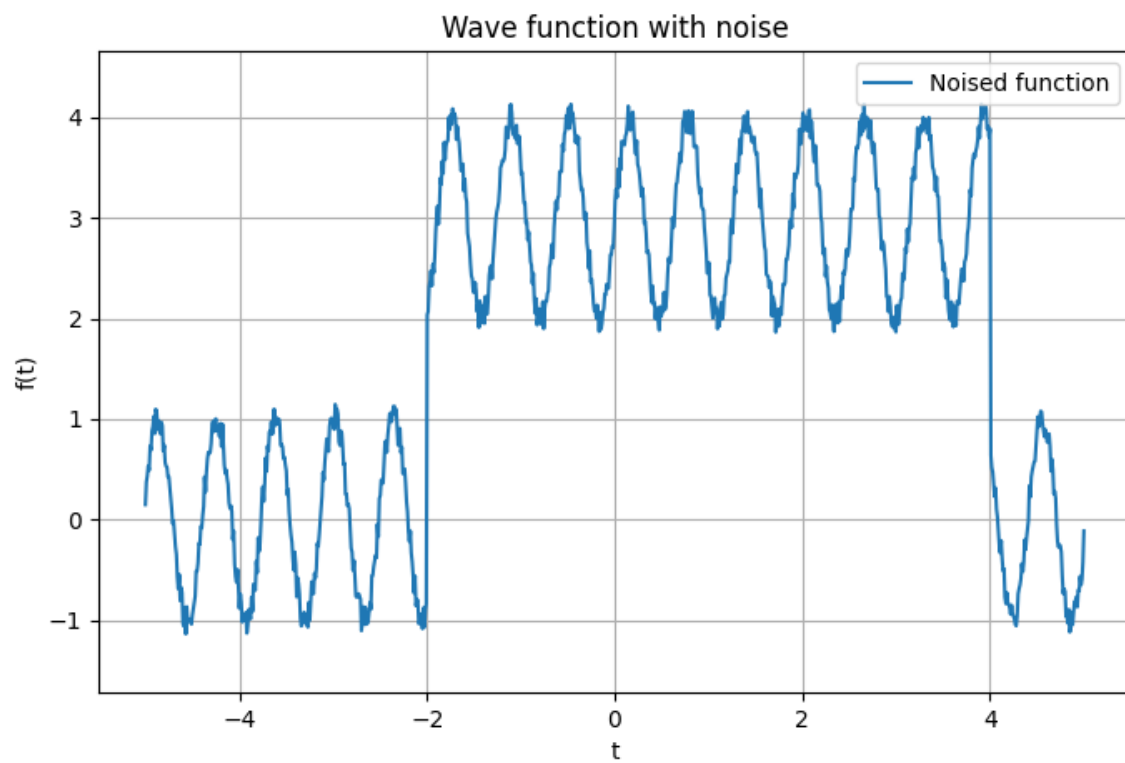


Рис. 43: Функция $u(t)$ с параметрами $b = 0.3$, $c = 1$, $d = 10$

1.4.3. Нахождение образов исходной и зашумленной функций

Найдем образы исходной (см. рисунок 44) и зашумленной (см. рисунок 45) функций с помощью преобразования Фурье.

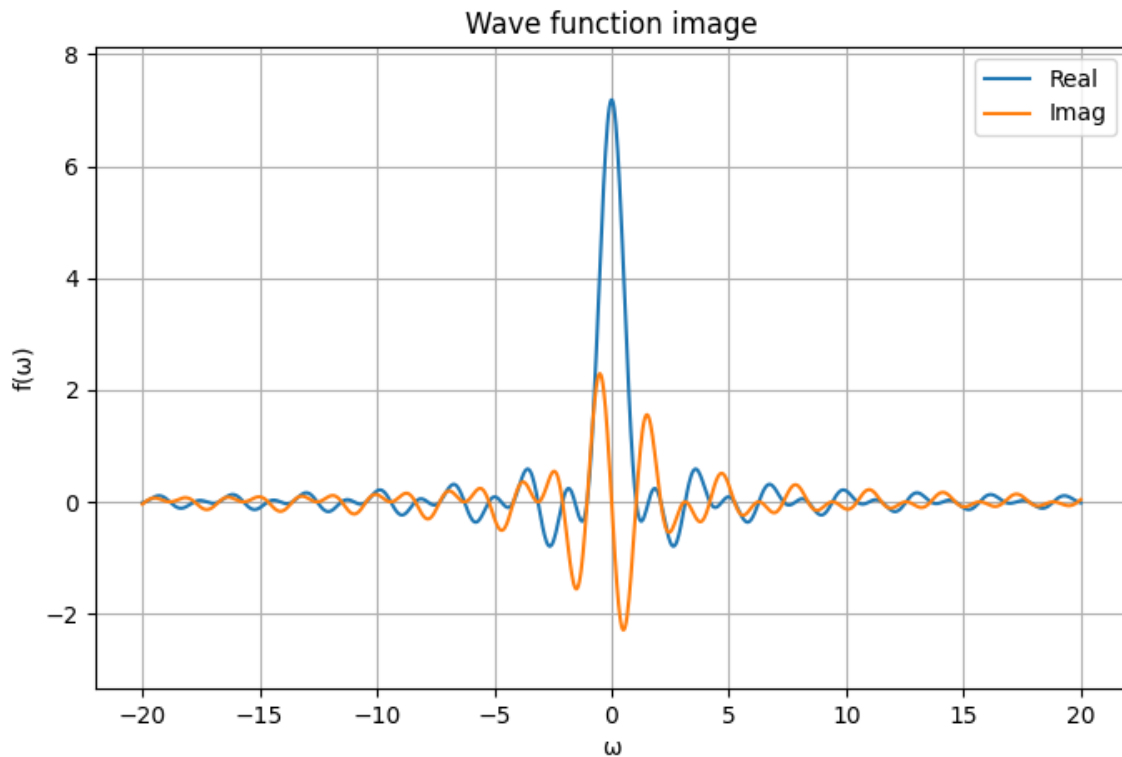


Рис. 44: Образ функция $g(t)$ с параметрами $a = 3$, $t_1 = -2$, $t_2 = 4$

Как и в прошлом случае, видим, что на графике образа зашумленной функции появились пики, которые отсутствуют на графике образа исходной функции.

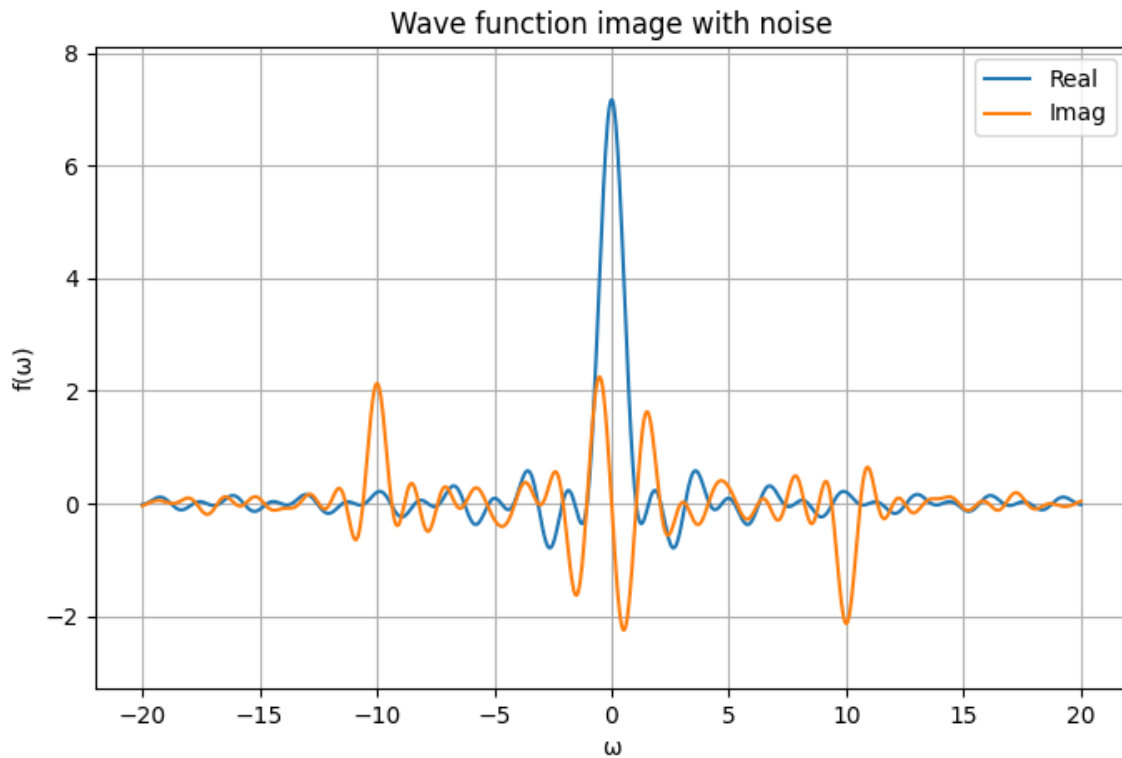


Рис. 45: Образ функция $u(t)$ с параметрами $b = 0.3$, $c = 1$, $d = 10$

1.4.4. Фильтрация низких частот

Обрежем образ функции $u(t)$, убрав компоненты, соответствующие низким частотам (см. рисунок 46).

Как и ожидалось, низкочастотные компоненты ($|\omega| < \omega_0$) стали равны нулю.

Теперь выполним обратное преобразование Фурье (используя обрезанный образ), получив при этом фильтрованную версию функции (см. рисунок 47).

Сравнительные графики исходной и фильтрованной функций представлены на рисунке 48.

Видно, что от исходной функции остался только шум, что логично, ведь мы убрали низкие частоты, которые, в большинстве, и соответствовали самой функции. Амплитуда шума осталась неизменной.

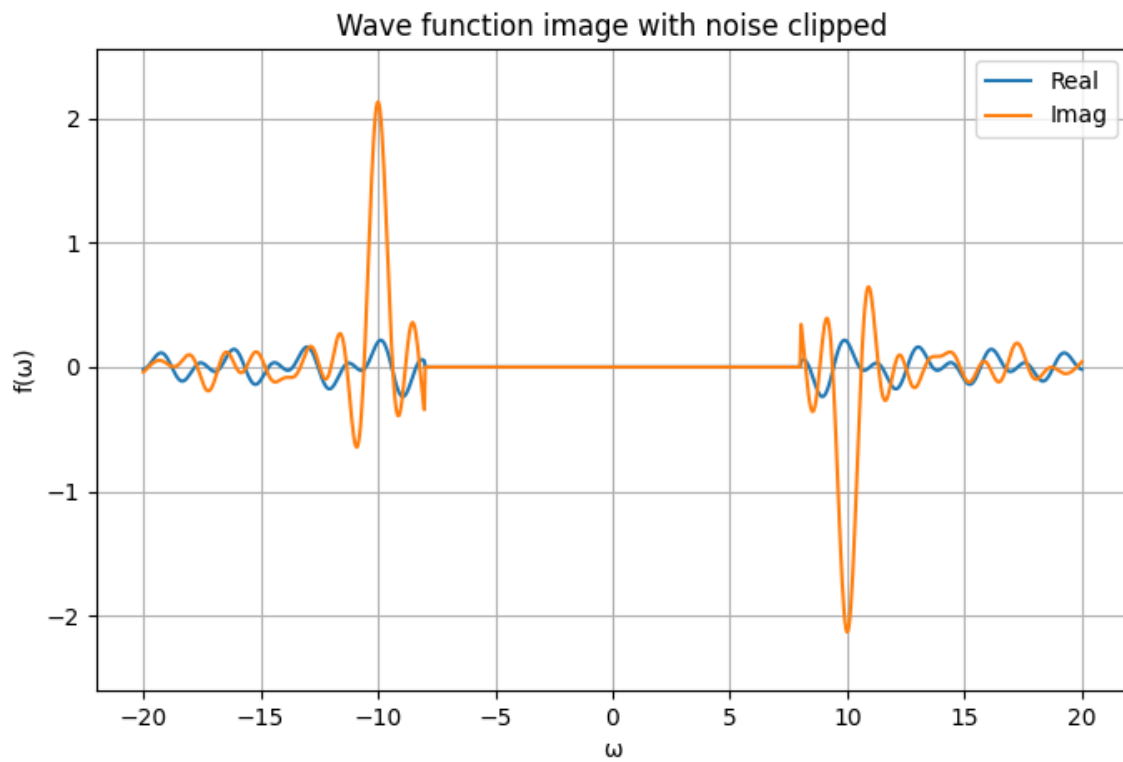


Рис. 46: Обрезанный образ функция $u(t)$ при $\omega_0 = 8$

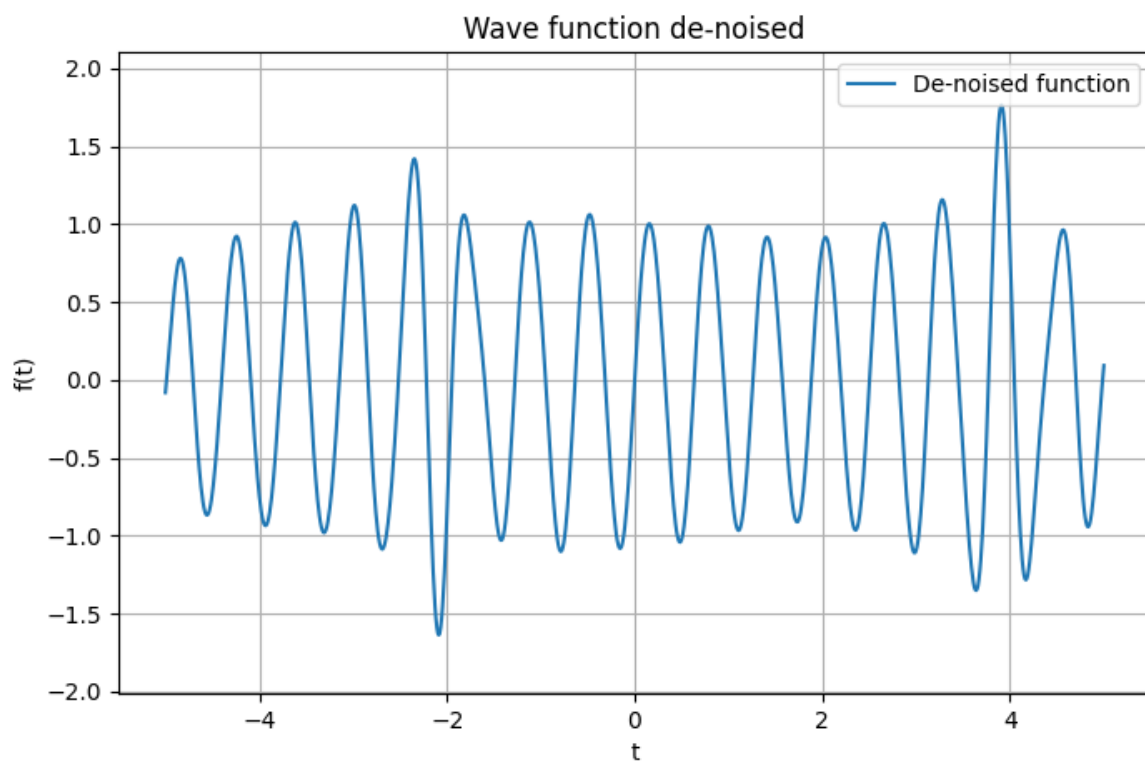


Рис. 47: Фильтрованная функция $u(t)$ при $\omega_0 = 8$

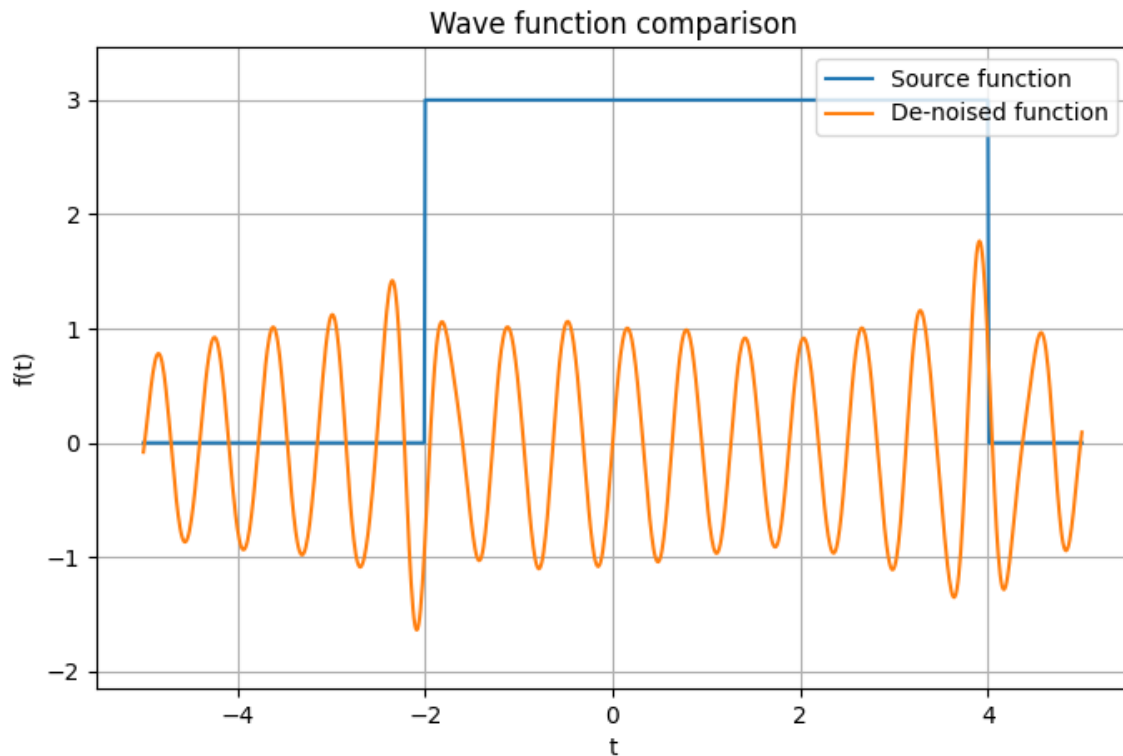


Рис. 48: Исходная и фильтрованная функция при $\omega_0 = 8$

2. Задание 2. Фильтрация звука

2.1. Чтение исходных данных

Для начала, прочитаем `.wav` файл в виде массива точек и выведем его график (см. рисунок 49).

2.2. Нахождение образа исходного сигнала

Образ исходного сигнала находится с помощью преобразования Фурье. Для этого используется функция `fft` из библиотеки `numpy`. После нахождения образа сигнала, мы можем построить его график (см. рисунок 50).

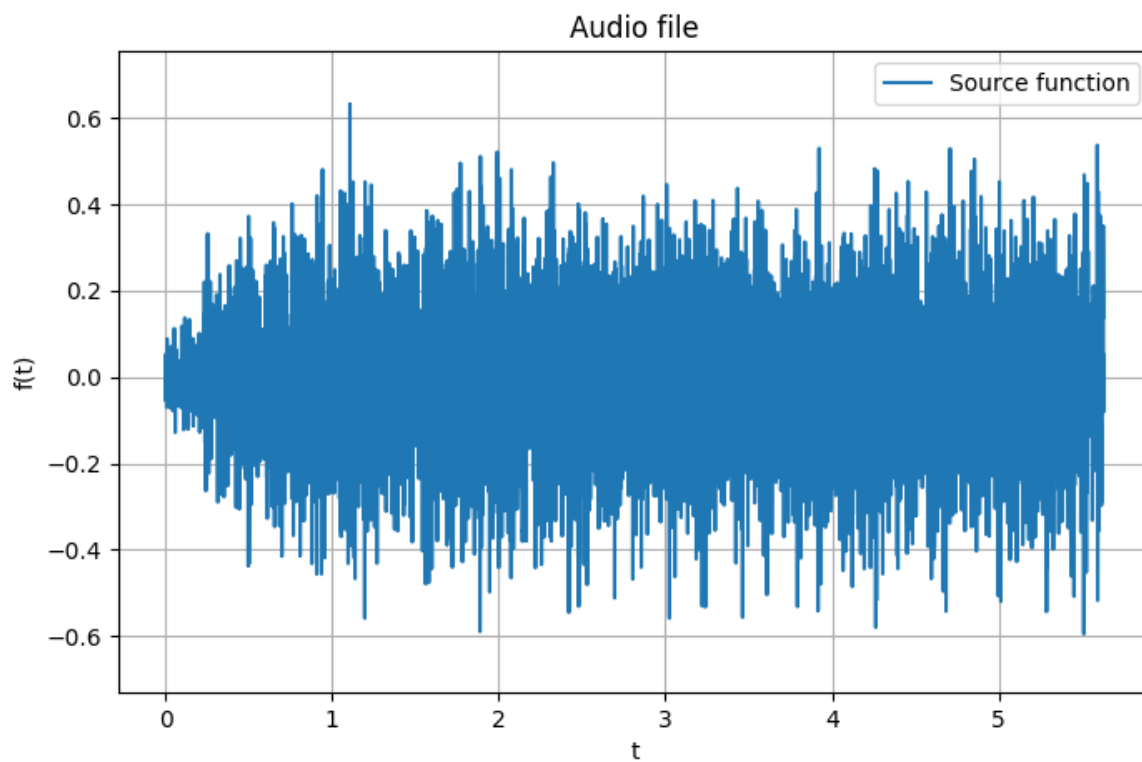


Рис. 49: График исходного .wav файла

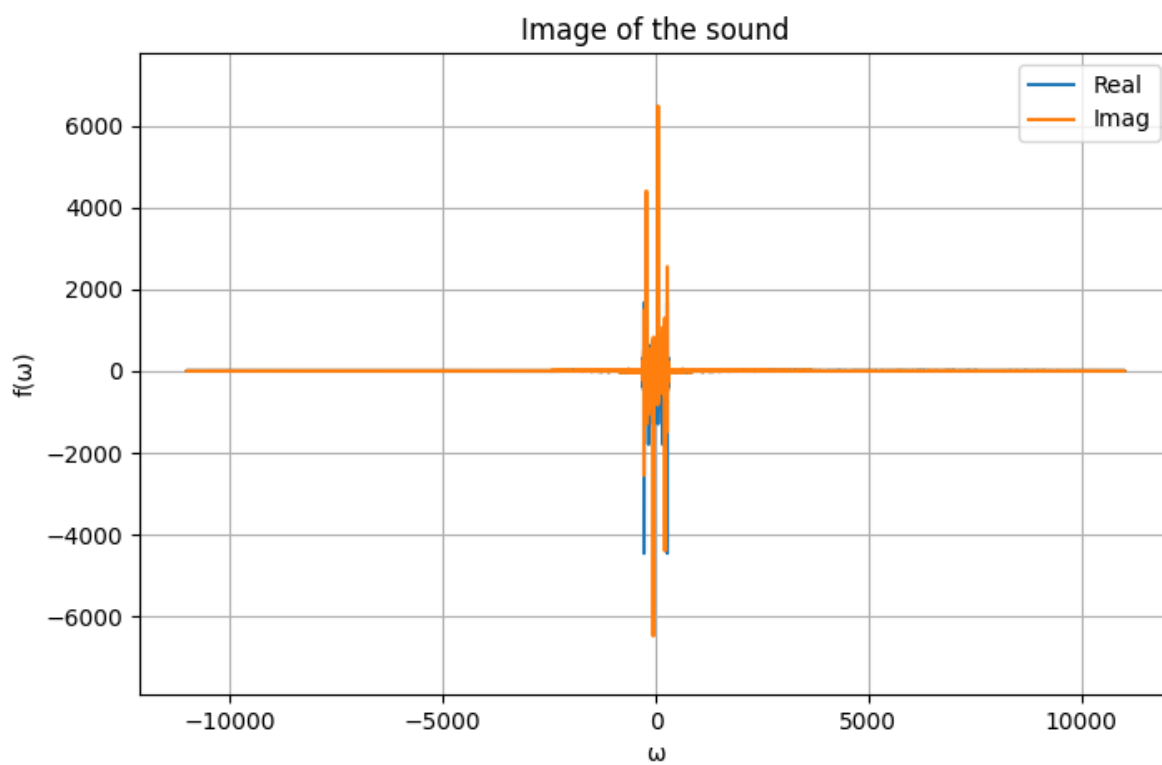


Рис. 50: График образа исходного сигнала

2.3. Определение частот с шумом

Заметим, что на графике образа есть ярко выраженные пики в окрестности нуля. Остальная часть образа кажется равной нулю, но это не так, там есть малые значения, неразличимые из-за масштаба. Обнулим все частоты, сильно *выбивающиеся* из общей картины ($\omega = \pm 300$), и построим график (см. рисунок 51).

Теперь стали заметны остальные значения. Максимум функции снизился ≈ 6000 до ≈ 40 . Теперь частоты в образе распределены гораздо более равномерно, что похоже на образ аудио с голосом.

2.4. Обратное преобразование

Теперь, когда мы нашли образ сигнала и обрезаем его, мы можем восстановить исходный сигнал с помощью обратного преобразования Фурье. Для этого используется функция `ifft` из библиотеки `numpy`. После нахождения обратного преобразования, мы можем построить его график (см. рисунок 52). График звуковой волны стал менее равномерным, что больше похоже на запись голоса. Сохраним полученный сигнал как аудио и послушаем его.

Запись стала чистой, все шумы ушли, при этом речь диктора не исказилась, по крайне мере, на слух. Из этого можно сделать вывод, что шумы были исключительно низкочастотными и от них удалось полностью избавиться.

Отфильтрованная запись доступна по [ссылке](#)

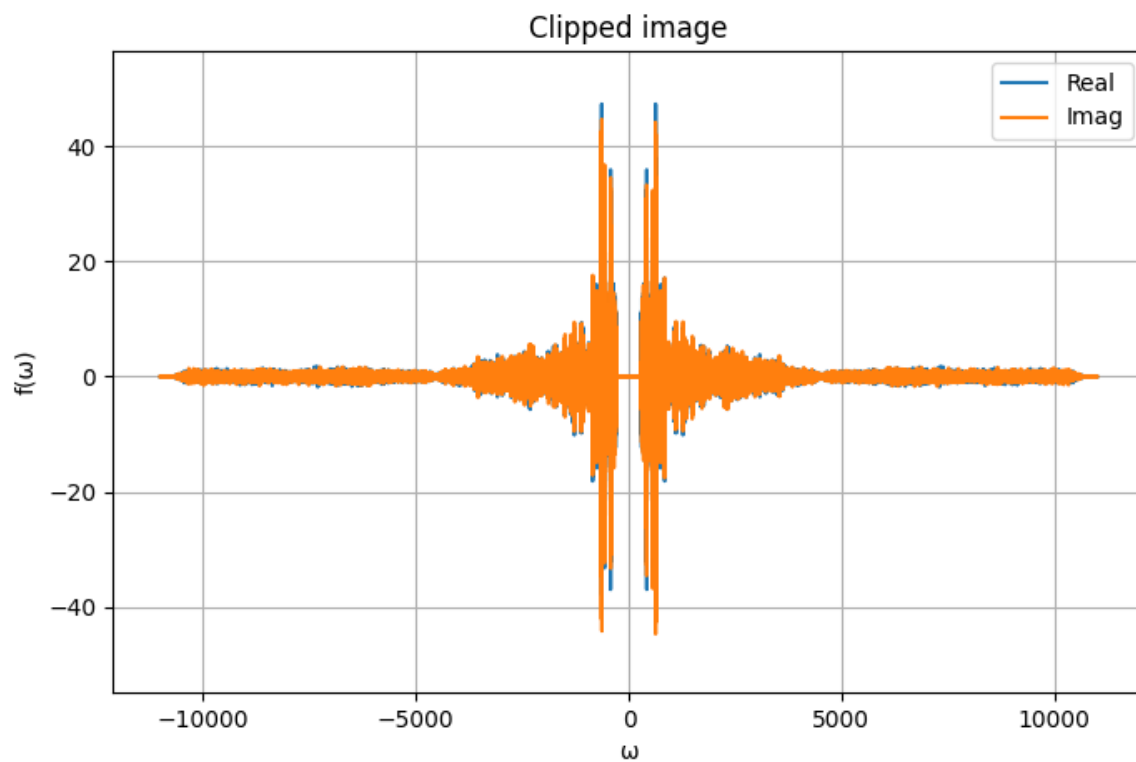


Рис. 51: График обрезанного образа исходного сигнала

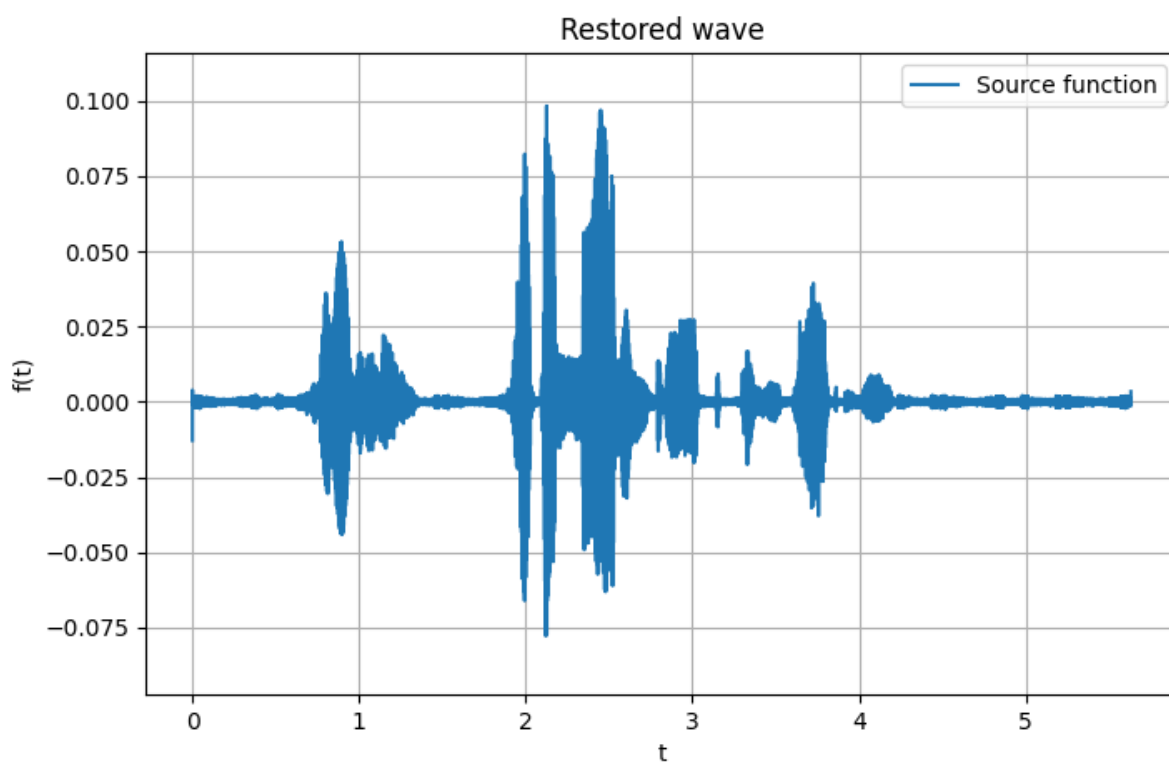


Рис. 52: График фильтрованной версии исходного сигнала