

НАУЧНО ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО
ФАКУЛЬТЕТ СИСТЕМ УПРАВЛЕНИЯ И РОБОТОТЕХНИКИ

Отчет по лабораторной работе №3
Жёсткая фильтрация

Выполнил студент группы R3380
Преподаватели

Мовчан И. Е.
Пашенко А.В., Перегудин А. А.

Санкт-Петербург
2025

Содержание

1 Жёсткие фильтры	2
1.1 Убираем высокие частоты	3
1.2 Убираем специфические частоты	8
1.3 Убираем низкие частоты?	23
2 Фильтрация звука	27
3 Общие выводы	29
4 Приложение	31

1 Жёсткие фильтры

Перед выполнением зададимся числами $a = 1$, $t_1 = -3$, $t_2 = 3$ (отметим, что $t_1 < t_2$), составив тем самым прямоугольную волну

$$g(t) = \begin{cases} a, & t \in [t_1, t_2], \\ 0, & \text{иначе;} \end{cases} \Rightarrow g(t) = \begin{cases} 1, & t \in [-3, 3], \\ 0, & \text{иначе.} \end{cases}$$

Кроме этого, выберем времени большой интервал по времени T и достаточно малым шагом дискретизации Δt (для улучшения частотных значений), задав массив времени t с данными параметрами и значения функции $g(t)$ на этом же отрезке.

Рассмотрим зашумлённую версию функции:

$$u(t) = g(t) + b\xi(t) + c \sin(dt),$$

где $\xi \sim U[-1, 1]$ - равномерное распределение, представляющее белый шум, а b, c, d - параметры возмущений. Сигнал $u(t)$ мы и будем очищать от этих сторонних воздействий различной природы.

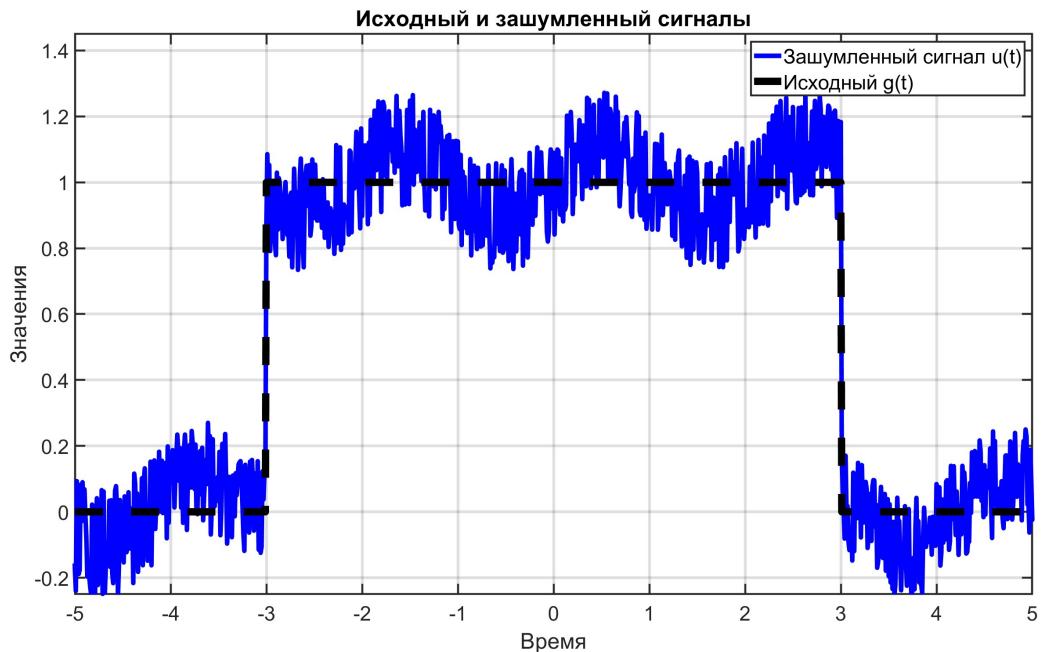


Рис. 1: Исходный и зашумленный сигналы при $b = 0.35$, $c = 0.1$, $d = 3$

1.1 Убираем высокие частоты

Положим $c = 0$ (уберем стабильную синусоидальную помеху), а $b = 0.35$ (то есть будем исследовать то, как можно подавить белый шум; для личного удобства далее в качестве параметра b будет рассматриваться максимально возможный «разнос» случайного возмущения, то есть параметр в два раза больший $= 2b$ исходного). Найдём Фурье-образ зашумлённого сигнала $\hat{u}(\nu)$ и обнулим его значения для всего диапазона частот вне отрезка $[-\nu_0, \nu_0]$, где $\nu_0 = 2$, сделаем обратное преобразование и посмотрим, что изменится. Результаты работы представлены на рисунках 2 и 3.

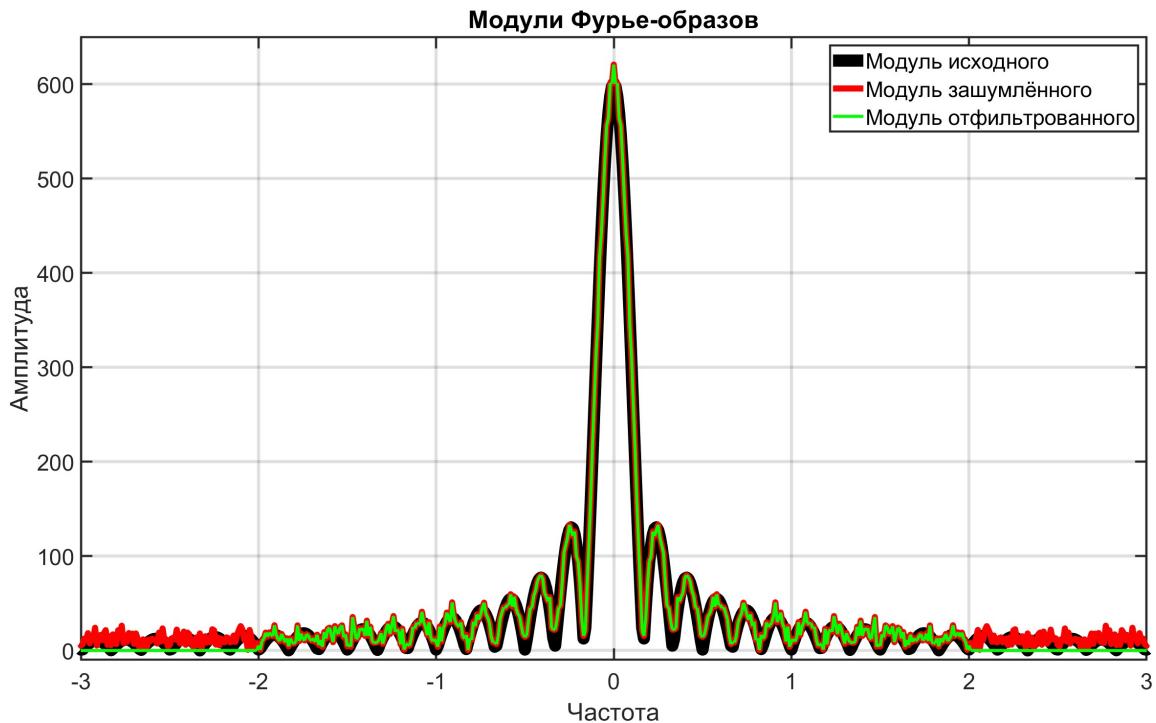


Рис. 2: Модули образов Фурье сигналов при $b = 0.35$, $\nu_0 = 2$

Как видим, модуль зашумлённой функции на высоких частотах представляет из себя еле разборчивое «месиво», не вносящее никакого положительного вклада в сигнал. Это признаки проявления белого шума, который мы и пытаемся подавить, сведя его проявление в образах к 0. Можно заметить, что стороннее воздействие также

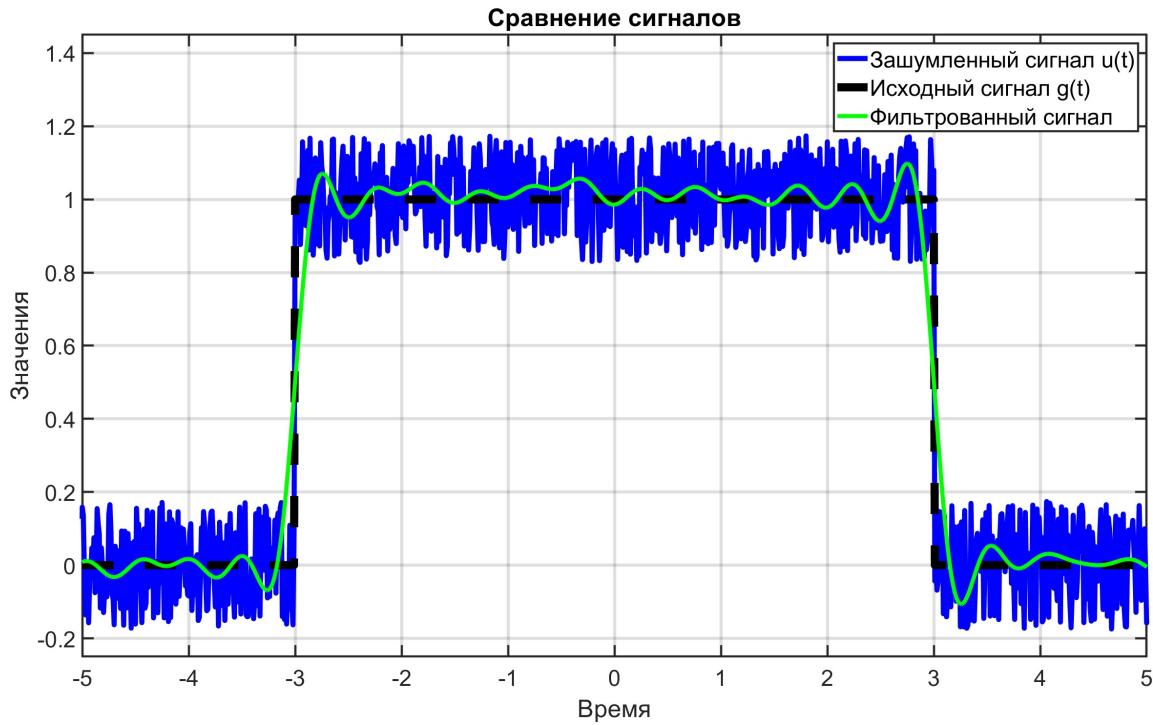


Рис. 3: Восстановление отфильтрованной функции при $b = 0.35$, $\nu_0 = 2$

проводится и на низких частотах, только тут оно менее значимо, так как его амплитуда в сравнении с истинным сигналом минимальна, соответственно, минимальны и их вклад с искажениями (ровно противоположная ситуация наблюдается с высокими частотами, обладающими малыми амплитудами, для которых сторонние выбросы достаточно значимы). Итак, идея метода фильтрации белого шума - оставить неизменными низкие частоты, так как там он не мало что меняет, и подавить до 0 высокие, которые и так не несут в себе никакого смысла, так как полностью зашумлены. То есть мы применяем *фильтр низких частот* (на рисунке 3 как раз продемонстрирована его хорошая работа - случайные частые выбросы исчезли, превратившись в плавные сдвиги возле истинных значений сигналов, представление функции явно сдвинулось в лучшую сторону, став более гладким).

Параметр b влияет на уровень разноса шума, ν_0 - на «грубость» его отсечения в частотной области. Соответственно, чем выше b , тем более мелкие отрезки нам приходится использовать, уменьшая ν_0 ,

так как шум начинает проявляться на всë более большом диапазоне высоких частот (доходя и до низких), портя их амплитуды. Увеличение ν_0 также приводит к более жёсткой фильтрации, его уменьшение - к мягкости (зашумлённый сигнал практически мало трогается, так как урезаются не все шумовые компоненты в частотной облатси). Докажем сказанное, проварыировав каждый из параметров. Результаты работы при параметрах $b = 0.35$ и $\nu_0 = 8$ представлены на рисунках 4 и 5 (мягкая фильтрация, значима роль сторонних воздействий), при $b = 2$, $\nu_0 = 2$ - на рисунках 6 и 7, а при $b = 1$, $\nu_0 = 1$ на рисунках 8 9. Можно видеть, что сказанное действительно подтвердилось. Также отметим, что чем больше влияние рассматриваемого шума, тем проблематичнее выделить главный вид сигнала (вплоть до практически полной невозможности восстановления).

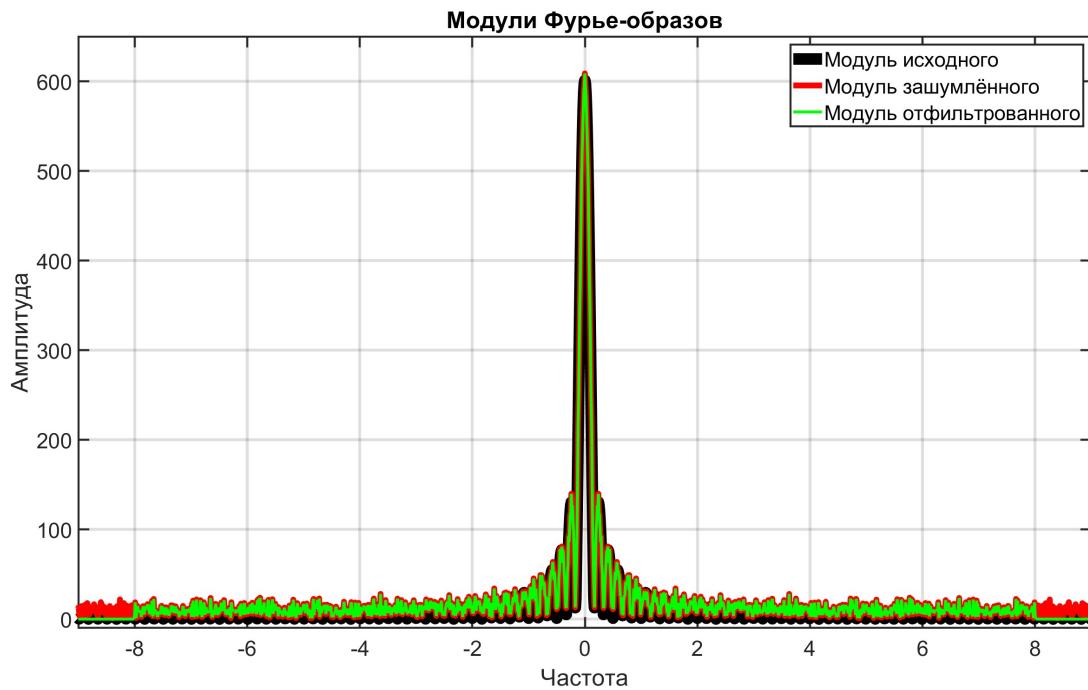


Рис. 4: Модули образов Фурье сигналов при $b = 0.35$, $\nu_0 = 8$

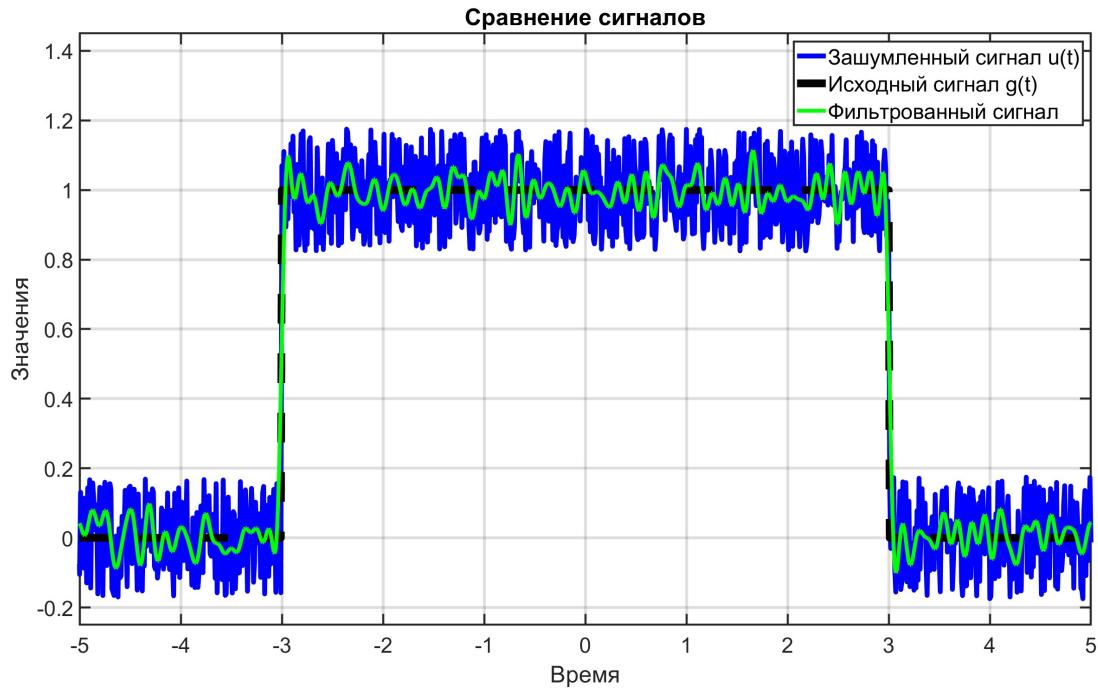


Рис. 5: Восстановление отфильтрованной функции при $b = 0.35$, $\nu_0 = 8$

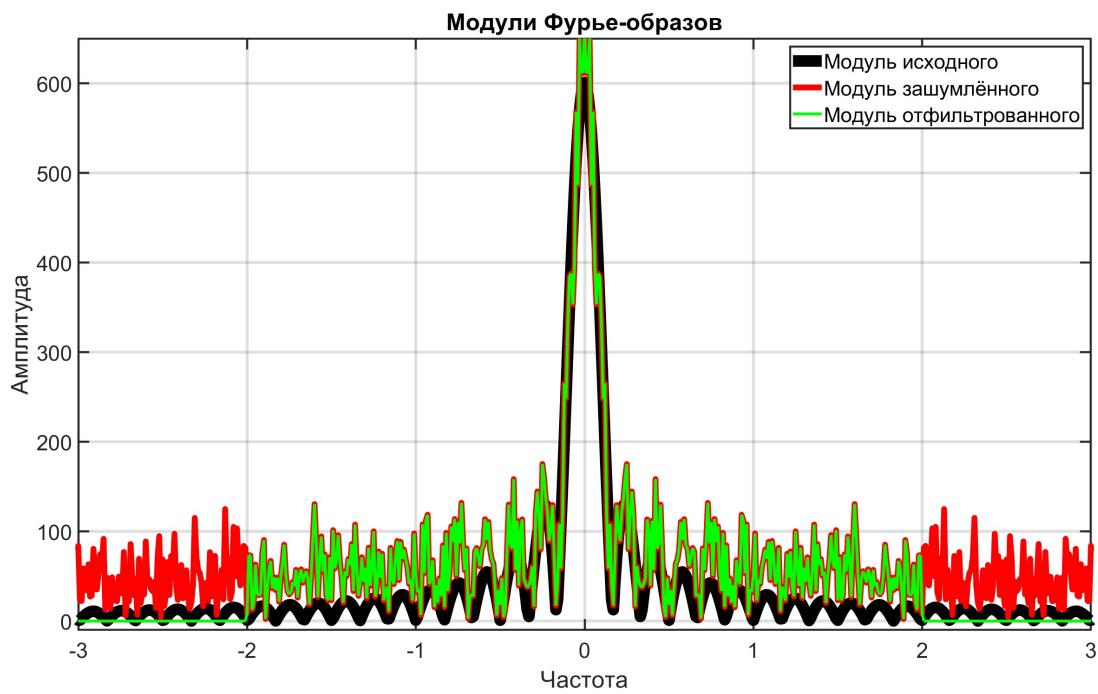


Рис. 6: Модули образов Фурье сигналов при $b = 1$, $\nu_0 = 2$

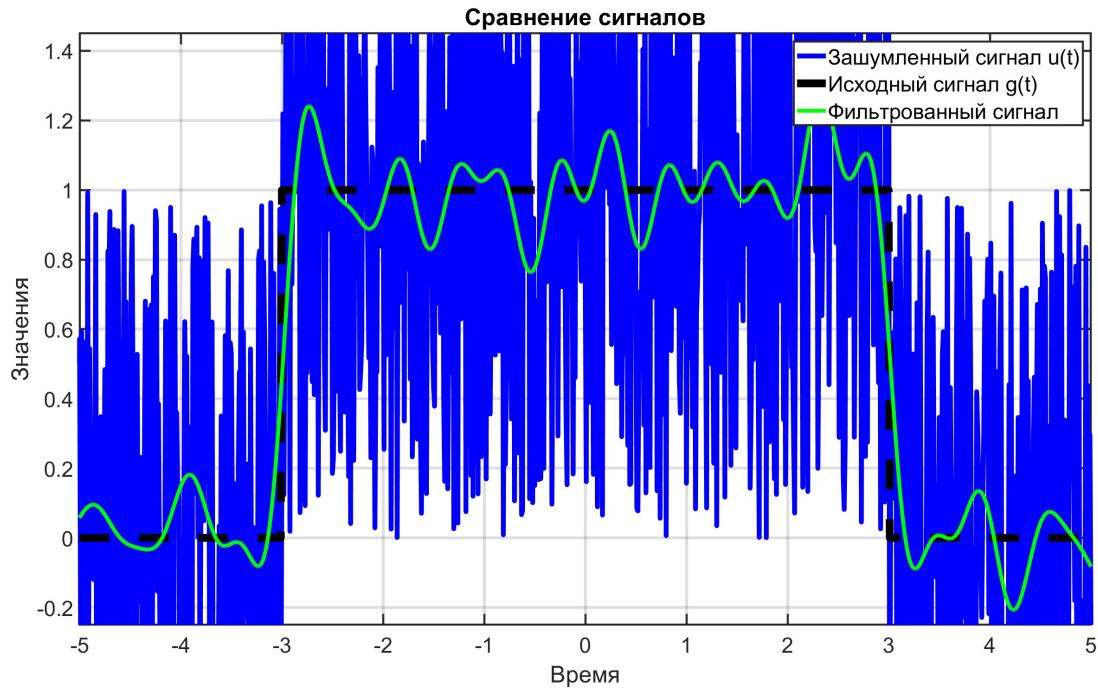


Рис. 7: Восстановление отфильтрованной функции при $b = 1$, $\nu_0 = 2$

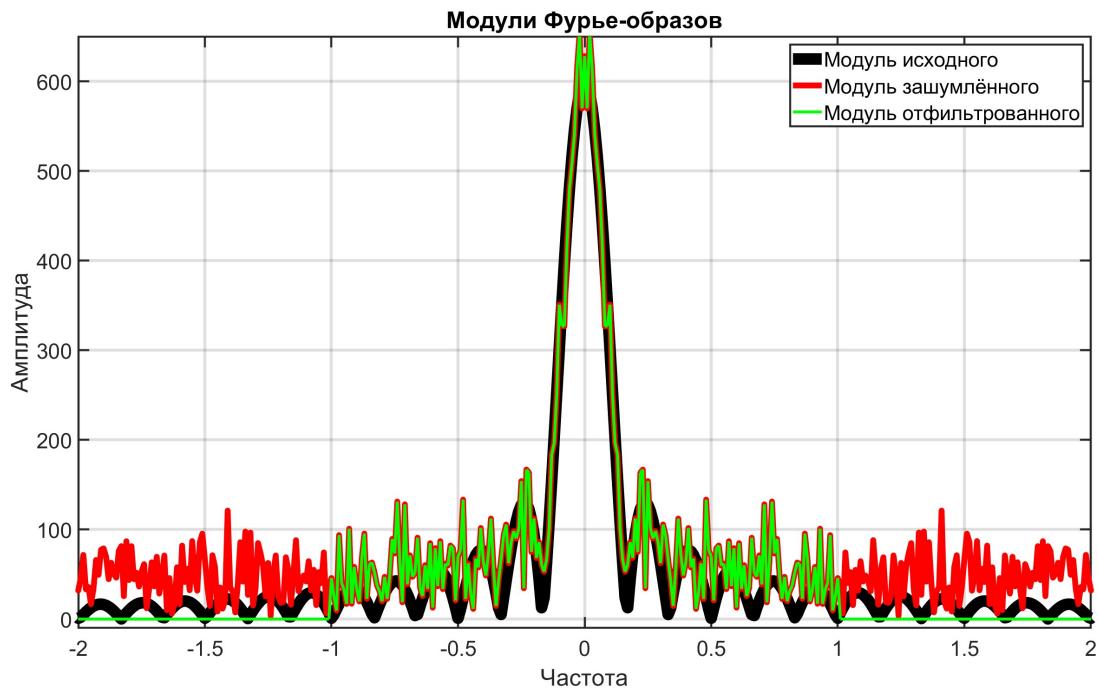
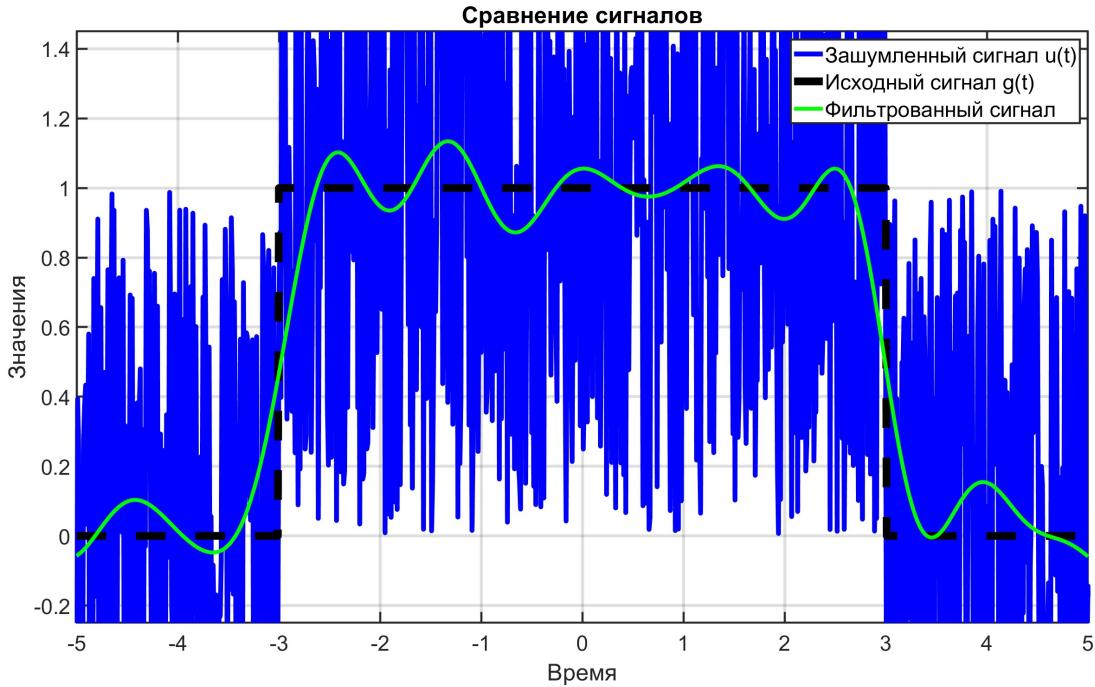


Рис. 8: Модули образов Фурье сигналов при $b = 2$, $\nu_0 = 1$

Рис. 9: Восстановление отфильтрованной функции при $b = 2$, $\nu_0 = 1$

1.2 Убираем специфические частоты

Примем теперь все параметры $b = 0.35$, $c = 0.1$ и $d = 3$ ненулевыми (получим график зашумлённого сигнала, как на рисунке 1). Подберём такой совмещенный фильтр, который независимым образом подавляет как специфические помехи в виде стороннего синусоидального воздействия (гармоники), так и случайные шумы. Данное достигается при последовательном использовании уже изученного *фильтра низких частот* с параметром ν_0 и *полосового фильтра*, давящий конкретные симметричные относительно 0 отрезки $[\pm\nu_1 - h, \pm\nu_1 + h]$, где h - ширина подавления, а $\pm\nu_1$ - точки, возле которых уменьшение амплитуд и происходит.

Всё дело в том, что гармоника при преобразовании Фурье превращается в дельта-функцию, разнесенную на соответствующие частоты, симметричные относительно 0 (дельта-функции локализованы в одной точке, поэтому их нужно подавлять таким же локализированным на конкретных отрезках полосовым фильтром), а случайный

шум, как уже было сказано, проявляется в основном на высоких частотах (и их мы будем гасить с помощью изученного в предыдущем пункте метода). Именно поэтому последовательное заглушение сначала одного воздействия, а после другого (без разницы, в каком порядке, ведь итог один) как раз даёт необходимую нам фильтрацию.

Результаты работы при параметрах фильтра $\nu_0 = 2$, $\nu_1 = 0.5$, $h = 0.1$ и параметрах сигнала $b = 0.35$, $c = 0.1$ и $d = 3$ приведены на рисунках 10 и 11. Видим, что подавились частота, находящиеся за пределами диапазона $[-\nu_0, \nu_0] = [-2, 2]$ либо частоты, входящие в отрезки $[\pm\nu_1 - h, \pm\nu_1 + h] = [\pm0.5 - 0.1, \pm0.5 + 0.1]$. Соответственно, те дельта-пики, которые давало гармоническое воздействие были полностью подавлены (в ширине h), а испорченные амплитуды высоких частот были полностью занулены. На рисунке 11 вышла хоть и не лучшая, но вполне неплохая фильтрация.

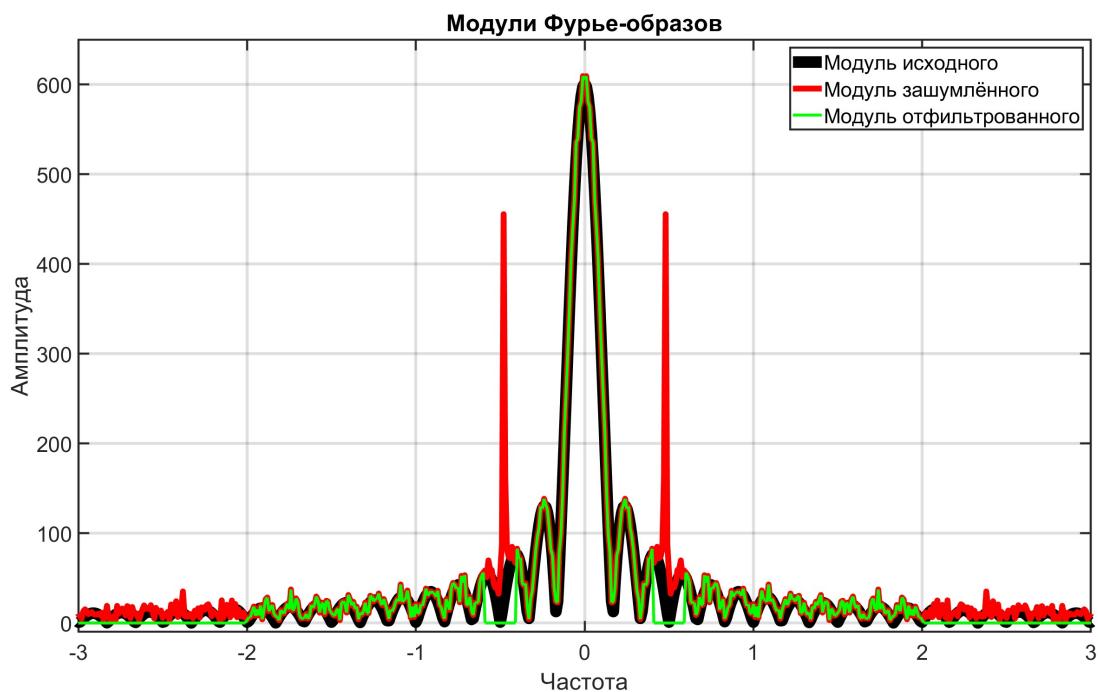


Рис. 10: Образы сигналов при $\nu_0 = 2$, $\nu_1 = 0.5$, $h = 0.1$, $b = 0.35$, $c = 0.1$ и $d = 3$

Давайте разберёмся с каждым из параметров по отдельности. Начнём с уже знакомого нам ν_0 . Как и прежде, он задаёт отрезок низких частот, который мы оставляем неизменным, и отвечает за си-

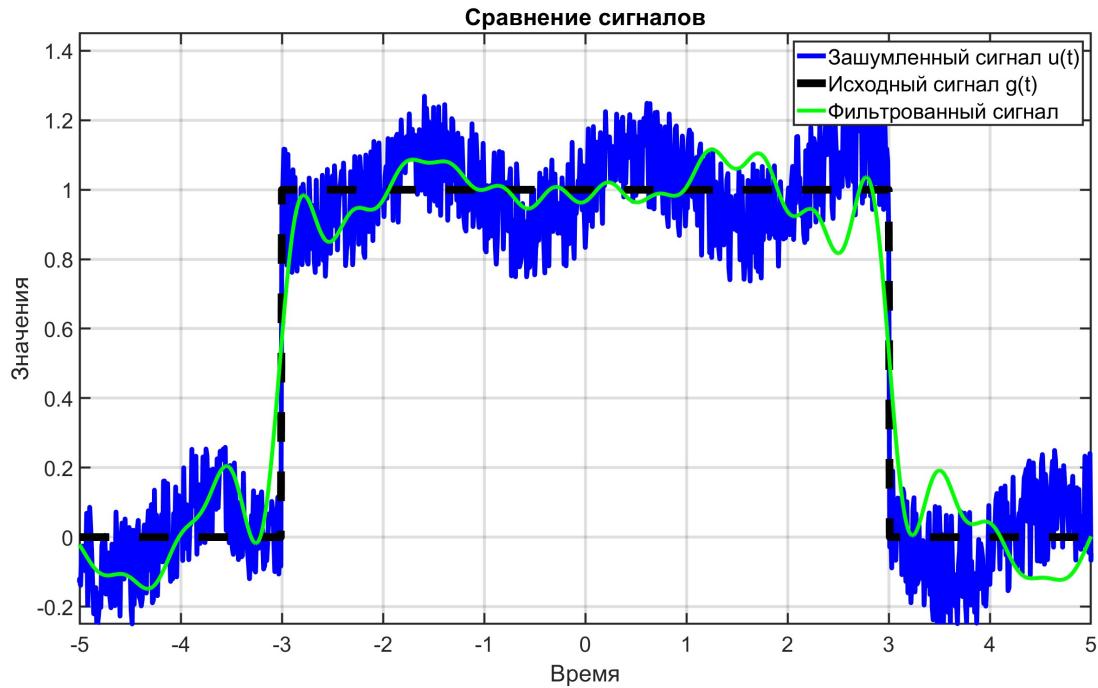


Рис. 11: Фильтрация при $\nu_0 = 2$, $\nu_1 = 0.5$, $h = 0.1$, $b = 0.35$, $c = 0.1$ и $d = 3$

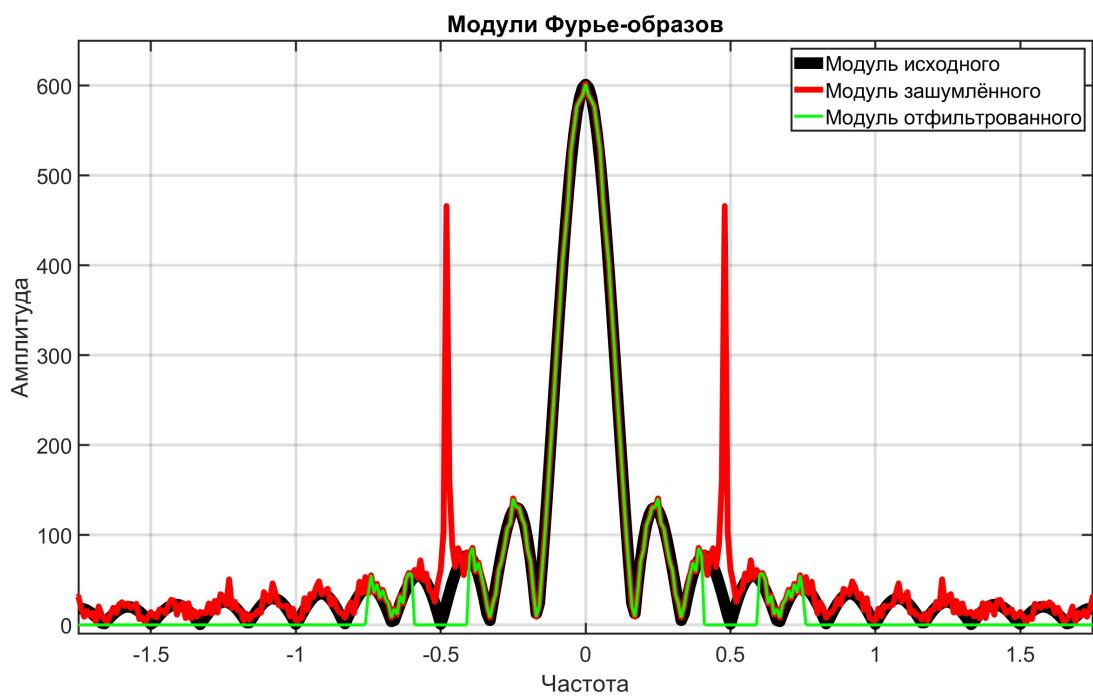


Рис. 12: Образы сигналов при $\nu_0 = 0.75$, $\nu_1 = 0.5$, $h = 0.1$, $b = 0.35$, $c = 0.1$, $d = 3$

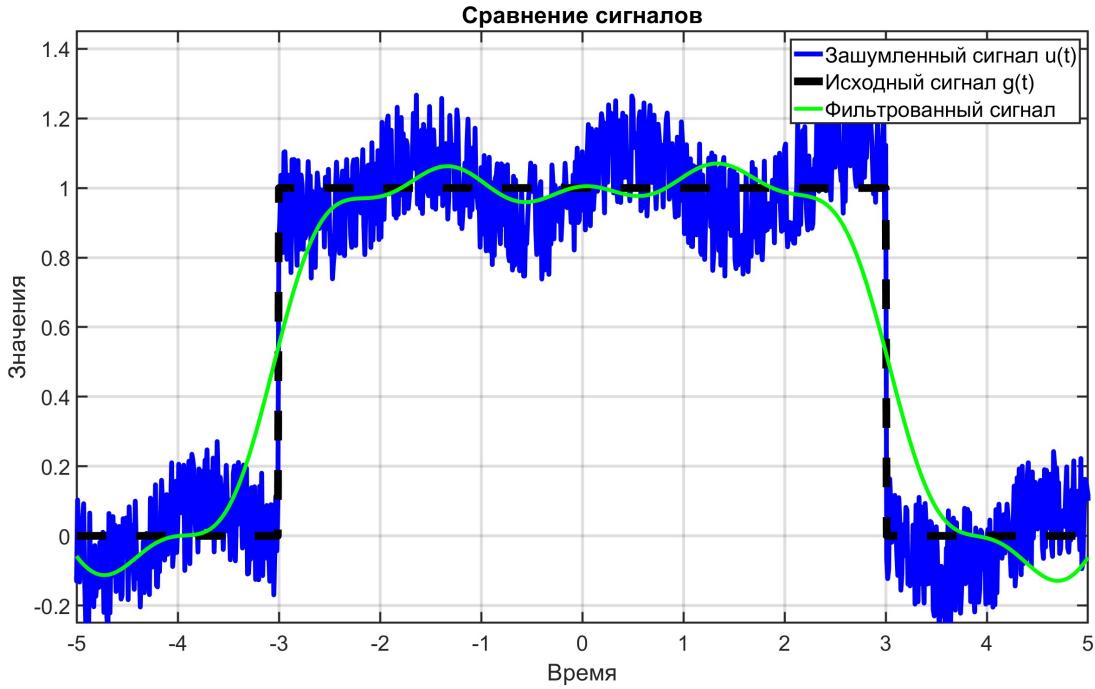


Рис. 13: Фильтрация при $\nu_0 = 0.75$, $\nu_1 = 0.5$, $h = 0.1$, $b = 0.35$, $c = 0.1$ и $d = 3$

лу, с которой будут подавляться случайные шумы, присутствующие в обрабатываемом сигнале. Уменьшение параметра приводит к более сильной фильтрации, а также большему урезания важных низкочастотных компонент, дающих общий вид функции, что и наблюдаем на рисунках 12 и 13, построенные при параметрах $\nu_0 = 0.75$, $\nu_1 = 0.5$, $h = 0.1$, $b = 0.35$, $c = 0.1$ и $d = 3$ (исчезают свойственные резкие пики сигнала, однако влияние белого шума практически незаметно). Стоит также отметить, что с помощью низкочастотных фильтров мы можем « зайти на территорию полосовых», убрав дельта-скачки в частотной области за счёт увеличения подавляемой области высоких частот (однако этот метод хоть и имеет место на существование, но является уж слишком варварским и грубым).

Далее на очереди - параметры ν_1 и h . С их помощью мы способны регулировать область работы полосового фильтра, расширяя её за счёт увеличения h и двигая изменением ν_1 . Может возникнуть ситуация, где мы знаем частоту сторонней гармоники, но с какой-то определенной погрешностью (то есть имеем $x \pm \Delta x$), - изменения-

ми параметров ν_1 и h можно точно добиться подавления именно той частотной области, в которой со 100-процентной вероятностью находится стороннее воздействие, дельта-функция (например, выбрав ширину h чуть выше Δx , а $\nu_1 = x$). Существует некоторый неизбежный недостаток метода в виде нежелательного подавления важных низких частот (имеет места при хоть сколько-то большой ширине). Избежать его возможно, только если мы по какой-то случайности чётко знаем частоту гармоники (в этом случае можно взять очень малое h , которое оставит практически все амплитуды неизменными, ν_1 , которое в точности задаёт точку дельта-пика образа синусоиды, и точечно в спектре очистить функцию). Итак, в идеале необходимо выбирать маленькую ширину h и точечно находить $\nu_1 = \frac{d}{2\pi}$, где d - частота синусоиды $\sin(dt)$ (у нас в идеале должно быть $\nu_1 = \frac{3}{2\pi} \approx 0.48$), а в случае проблем увеличивать h либо двигать ν_1 .

Подкрепим наши слова графиками: результаты работы при $\nu_1 = 1.5$ изображены на рисунках 14 и 15 (произошло смещение подавления - частоты, в которых находится образ Фурье гармоники, остаются нетронутыми; на выходе имеем ту же характерную синусоиду, не измененную ни по амплитуде, ни по фазе). На рисунках 16 и 17 находятся графики модулей образов исследуемых сигналов и сравнения их во временной области при параметрах $\nu_0 = 2$, $\nu_1 = 0.5$, $h = 0.25$, $b = 0.35$, $c = 0.1$ и $d = 3$ (увеличение h привело к захвату важных низких частот, ожидали возникли искажениям функции в общем виде функции - пропали пики, в целом потерялась структура). Также отдельно была рассмотрена ситуация, когда $\nu_1 = \frac{d}{2\pi}$, а $h = 0.01$ (при этом $\nu_0 = 2$, $b = 0.35$, $c = 0.1$ и $d = 3$), то есть происходит точечное уничтожение выброса в частотной области. Результаты на рисунках 18 и 19. Заметно, что результат практически идентичен, полученному на рисунке 3 при тех же прочих параметрах, где лишние гармоники попросту отсутствовали.

Дальше исследуем влияние каждого из параметров, задающих шумовое воздействие. Увеличение b , как уже было изучено, влечёт большие амплитуды белого шума (графики образов сигналов и их вида во временной области при изменении параметра до $b = 2$ представлены на рисунках 20 и 21 соответственно; прочие параметры при

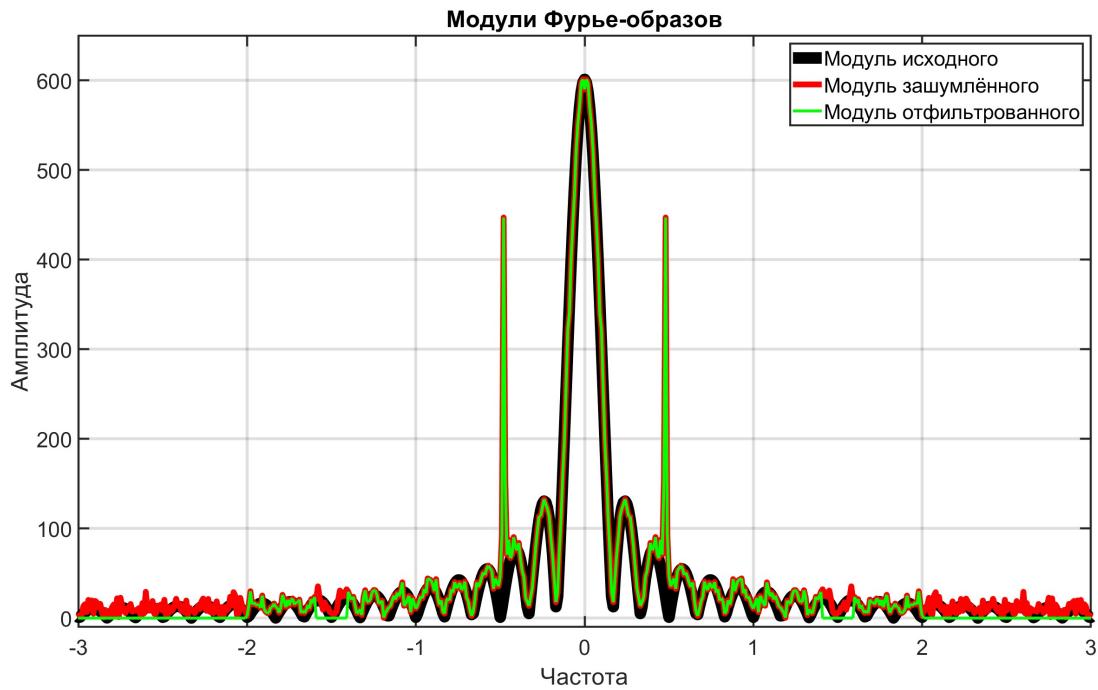


Рис. 14: Образы сигналов при $\nu_0 = 2$, $\nu_1 = 0.5$, $h = 0.1$, $b = 0.35$, $c = 0.1$ и $d = 3$

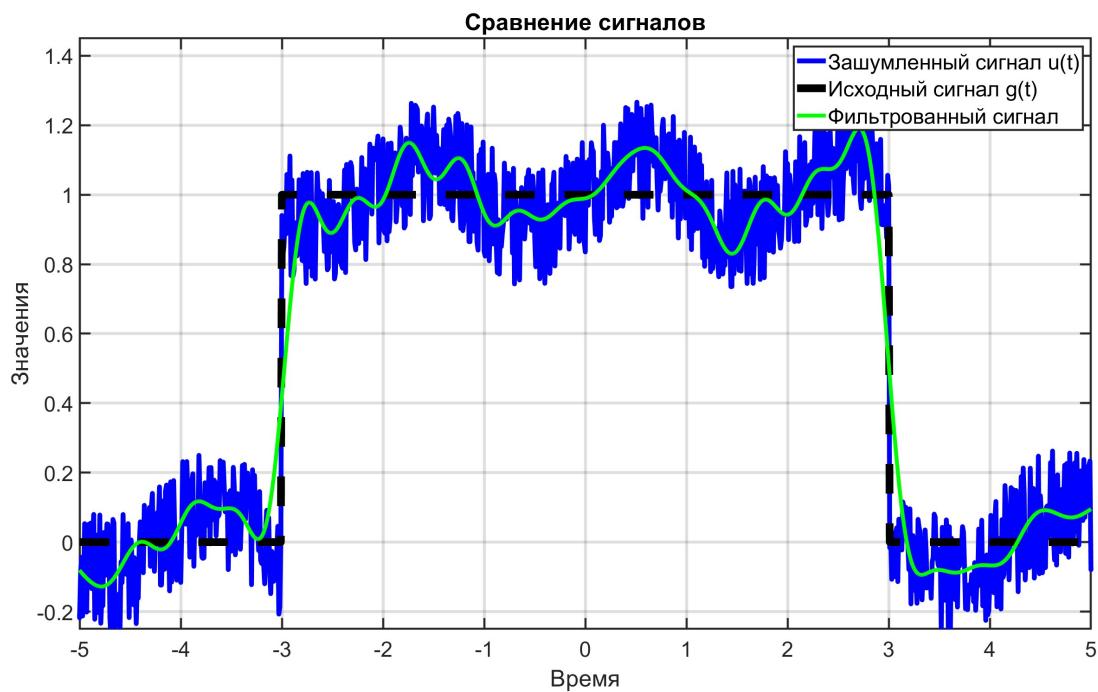


Рис. 15: Фильтрация при $\nu_0 = 2$, $\nu_1 = 0.5$, $h = 0.1$, $b = 0.35$, $c = 0.1$ и $d = 3$

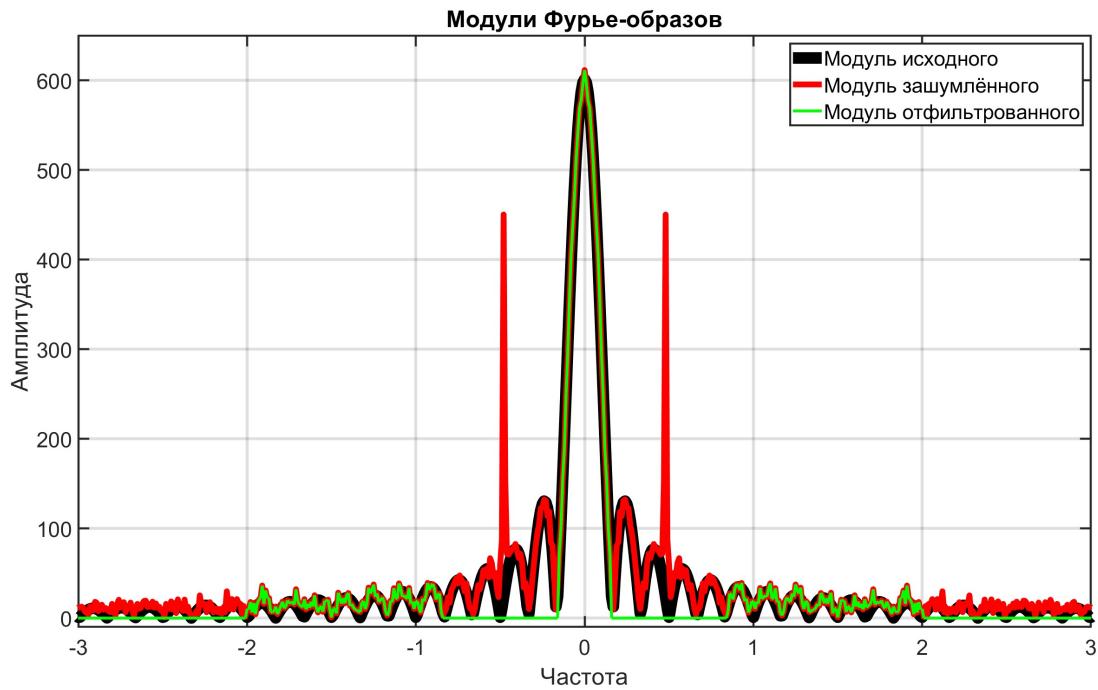


Рис. 16: Образы сигналов при $\nu_0 = 2$, $\nu_1 = 0.5$, $h = 0.35$, $b = 0.35$, $c = 0.1$ и $d = 3$

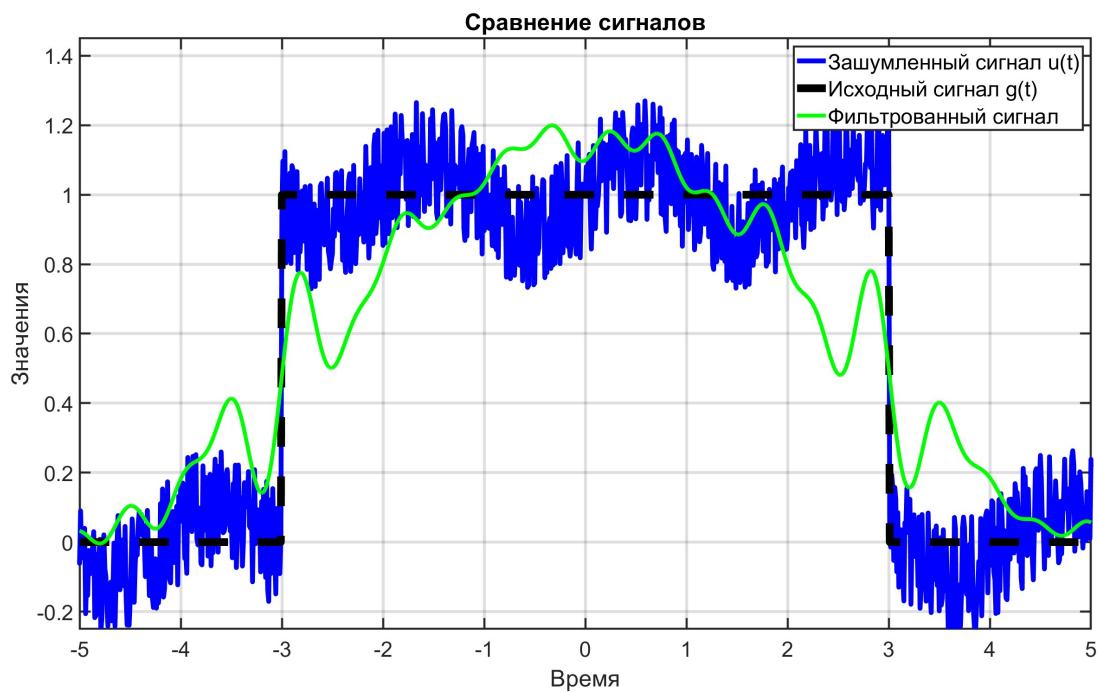


Рис. 17: Фильтрация при $\nu_0 = 2$, $\nu_1 = 0.5$, $h = 0.35$, $b = 0.35$, $c = 0.1$ и $d = 3$

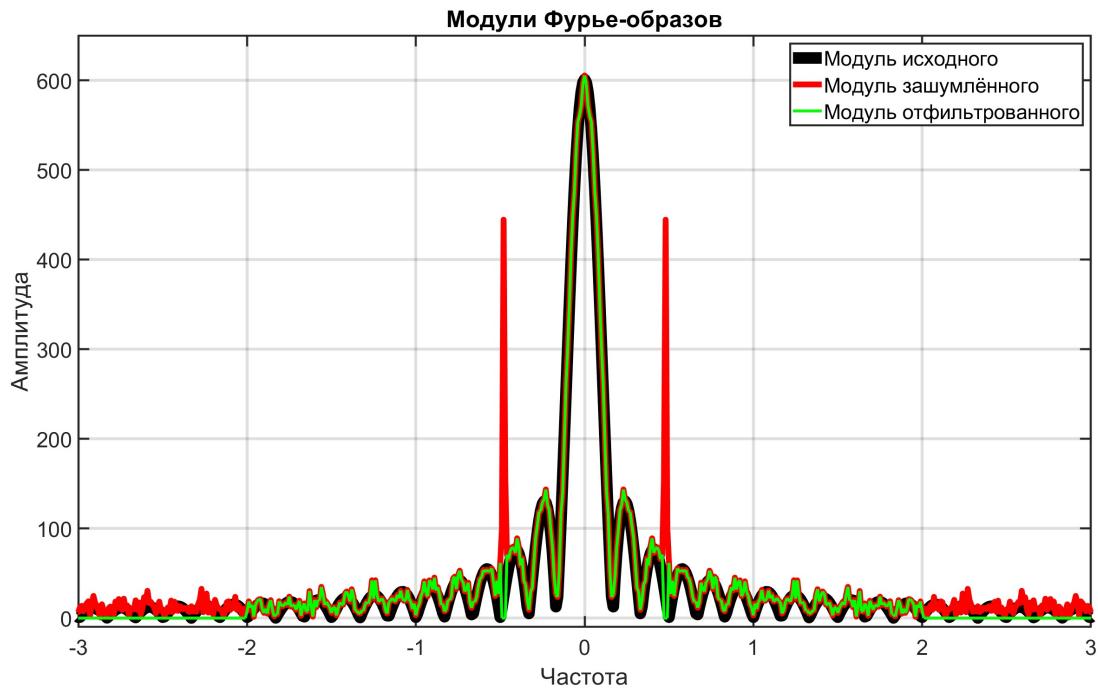


Рис. 18: Образы при $\nu_0 = 2$, $\nu_1 \approx 0.48$, $h = 0.015$, $b = 0.35$, $c = 0.1$ и $d = 3$

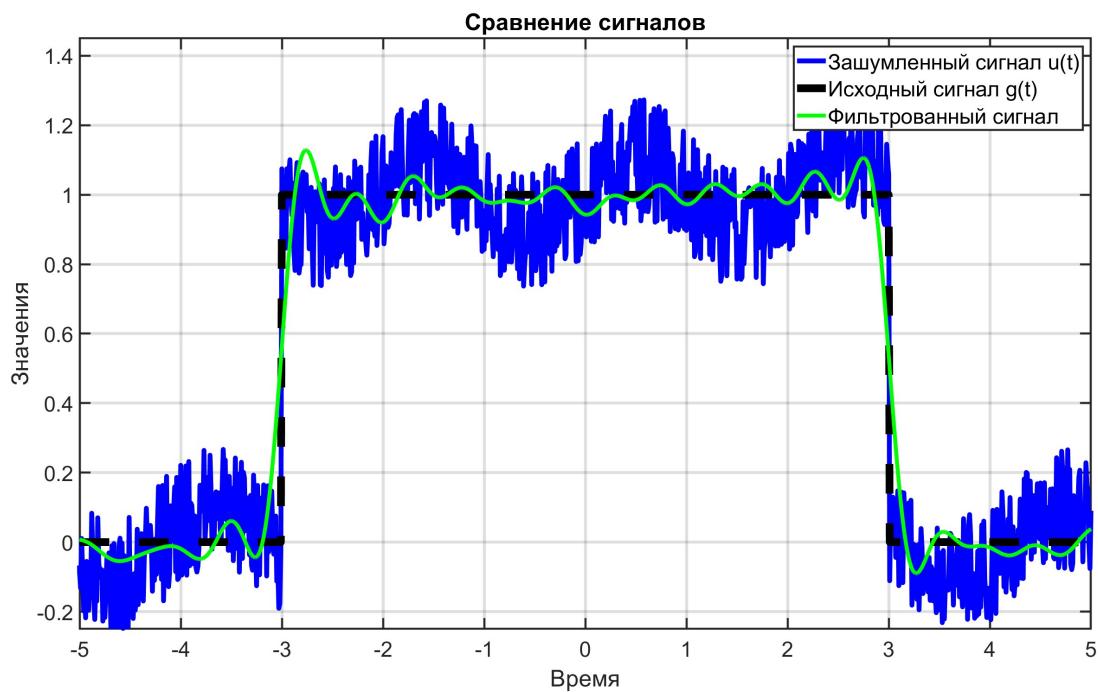


Рис. 19: Фильтрация при $\nu_0 = 2$, $\nu_1 \approx 0.48$, $h = 0.01$, $b = 0.35$, $c = 0.1$ и $d = 3$

этом равны $\nu_0 = 2$, $\nu_1 = 0.5$, $h = 0.1$, $c = 0.1$ и $d = 3$). Всё как раньше: при мелких b бы волны задавать малые диапазоны глушищихся частот, так как шум вносит свой вклад лишь на самых мелких амплитудах, при больших - вынуждены расширяться (рассмотрено ранее). Как и прежде, при высоких значениях шумов восстановление явно хромает (страдают даже низкие частоты с высокой амплитудой).

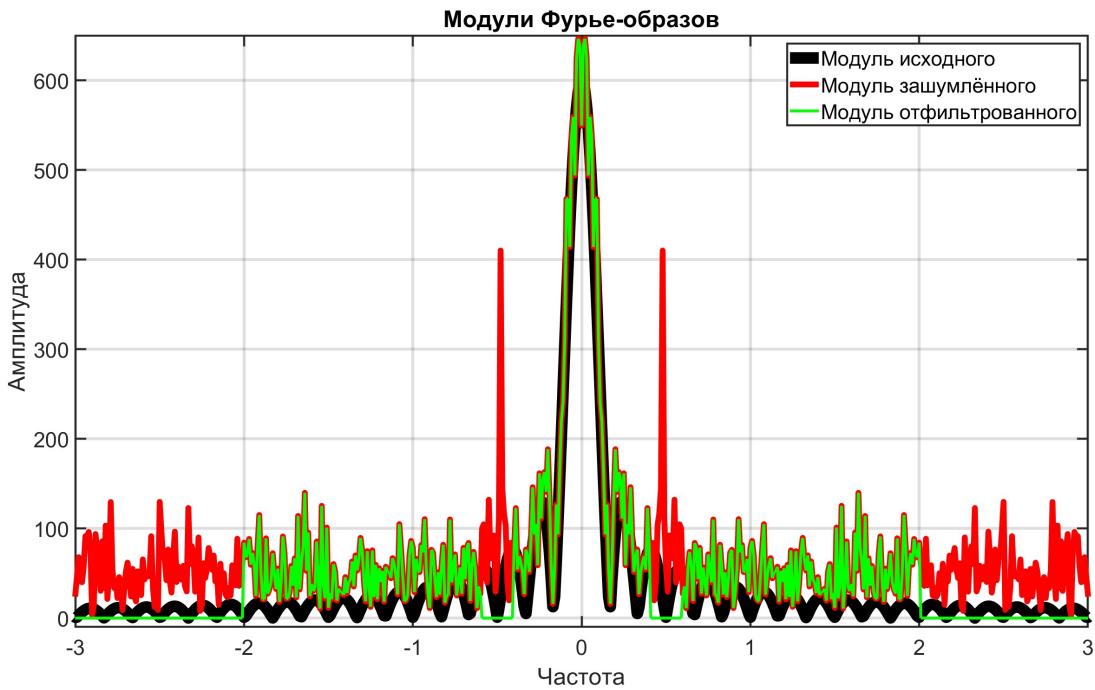


Рис. 20: Образы при $\nu_0 = 2$, $\nu_1 = 0.5$, $h = 0.1$, $b = 2$, $c = 0.1$ и $d = 3$

Отдельного внимания заслуживает случай $b = 0$. В этой ситуации случайный шум отсутствует (а значит, низкочастотных фильтр нам ни к чему), и можно практически неизменно восстановить исходный сигнал, точечно подавив присутствующую гармонику. Результаты при параметрах $\nu_0 = 2$, $\nu_1 \approx 0.48$, $h = 0.01$, $b = 0$, $c = 0.1$ и $d = 3$ - на рисунках 22 и 23.

Перейдём теперь к исследованию гармоники. Параметр d задаёт её частоту, c - амплитуду. Уже было получено, что синусоидальное воздействие отображается в частотной области в виде дельта-пика с аргументом, равным частоте гармоники, то есть d . Получается, изменение d «двигает» эту вершину, а при большом значении это-

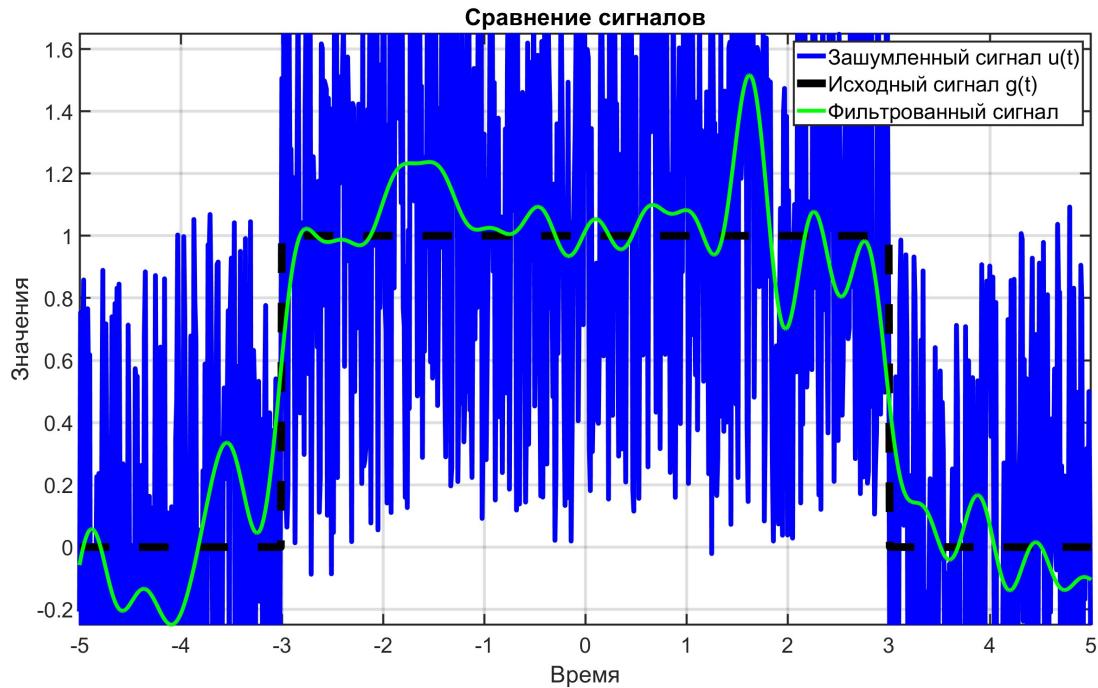


Рис. 21: Фильтрация при $\nu_0 = 2$, $\nu_1 = 0.5$, $h = 0.1$, $b = 2$, $c = 0.1$ и $d = 3$

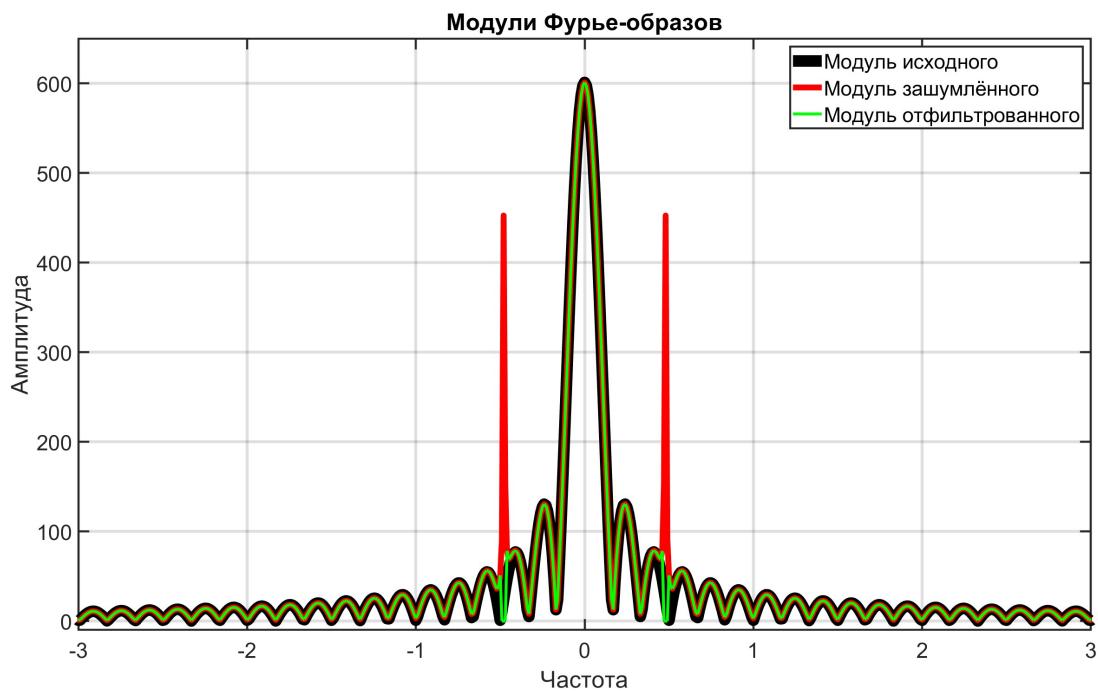


Рис. 22: Образы при $\nu_0 = 2$, $\nu_1 \approx 0.48$, $h = 0.01$, $b = 0$, $c = 0.1$ и $d = 3$

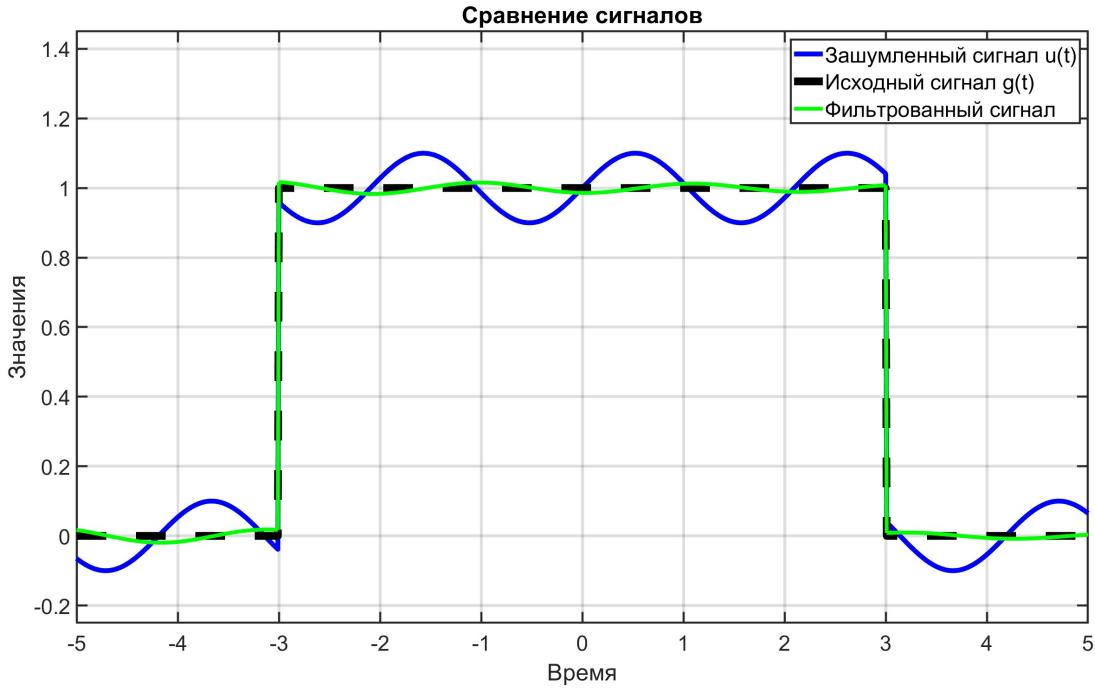


Рис. 23: Фильтрация при $\nu_0 = 2$, $\nu_1 \approx 0.48$, $h = 0.01$, $b = 0$, $c = 0.1$ и $d = 3$

го параметра мы можем даже перейти в случай высокочастотной помехи, которую будет подавлять уже не полосовой фильтр, а низкочастотный. Итак, графики при $d = 6$ представлены на рисунках 24 и 25 (на образе произошло смещение дельта-функции, поэтому полосовой фильтр не смог подавить возникающий выброс; синусоидальное воздействие при фильтрации сохранилось). Также зададимся достаточно высоким $d = 15$, и попробуем использовать только низкочастотный фильтр вместе с параметрами $\nu_0 = 2$, $b = 0.35$, $c = 0.1$ (рисунки 26 и 27). Видим, что пики задаются уже на высоких частотах, а значит, использование полосового фильтра не так уж и необходима.

Параметр c влияет на амплитуду сторонней гармоники. По рисункам 28 и 29, построенных при параметрах $\nu_0 = 2$, $\nu_1 = 0.5$, $h = 0.1$, $b = 0.35$, $c = 0.75$ и $d = 3$, и рисунках 30 и 31, построенных при параметрах $\nu_0 = 2$, $\nu_1 = 0.5$, $h = 0.1$, $b = 0.35$, $c = 0.05$ и $d = 3$. Видим, что меньшие амплитуды дают и меньшие пики в частотной области, которые легче и в подавлении (то есть при больших c мы

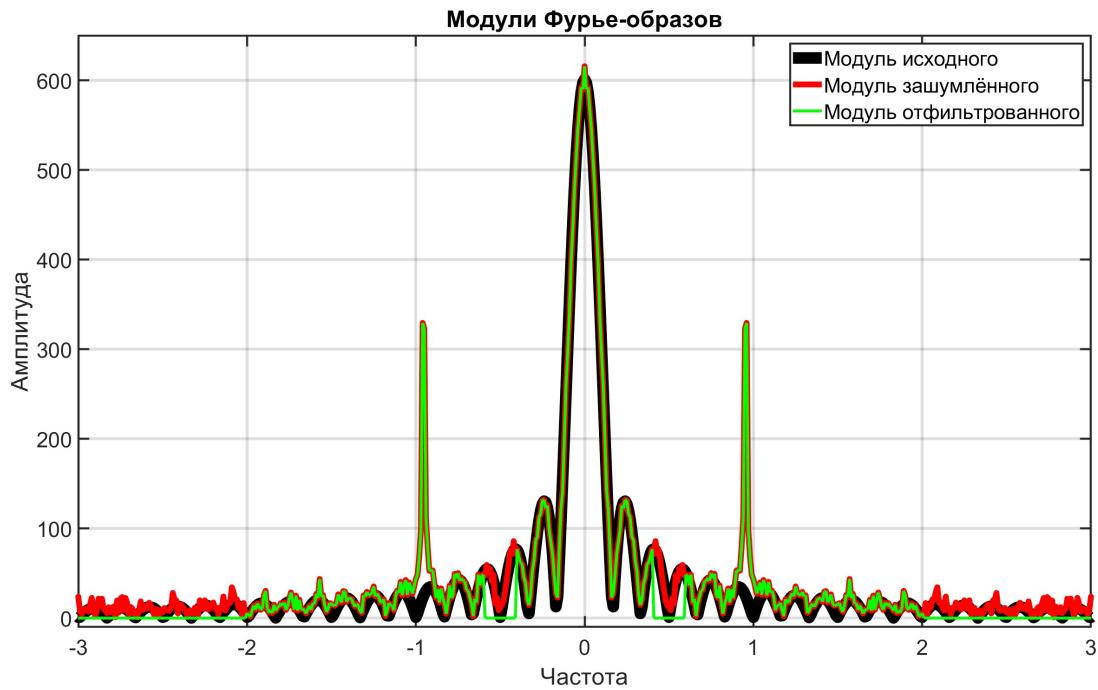


Рис. 24: Образы при $\nu_0 = 2$, $\nu_1 = 0.5$, $h = 0.1$, $b = 0.35$, $c = 0.1$ и $d = 6$

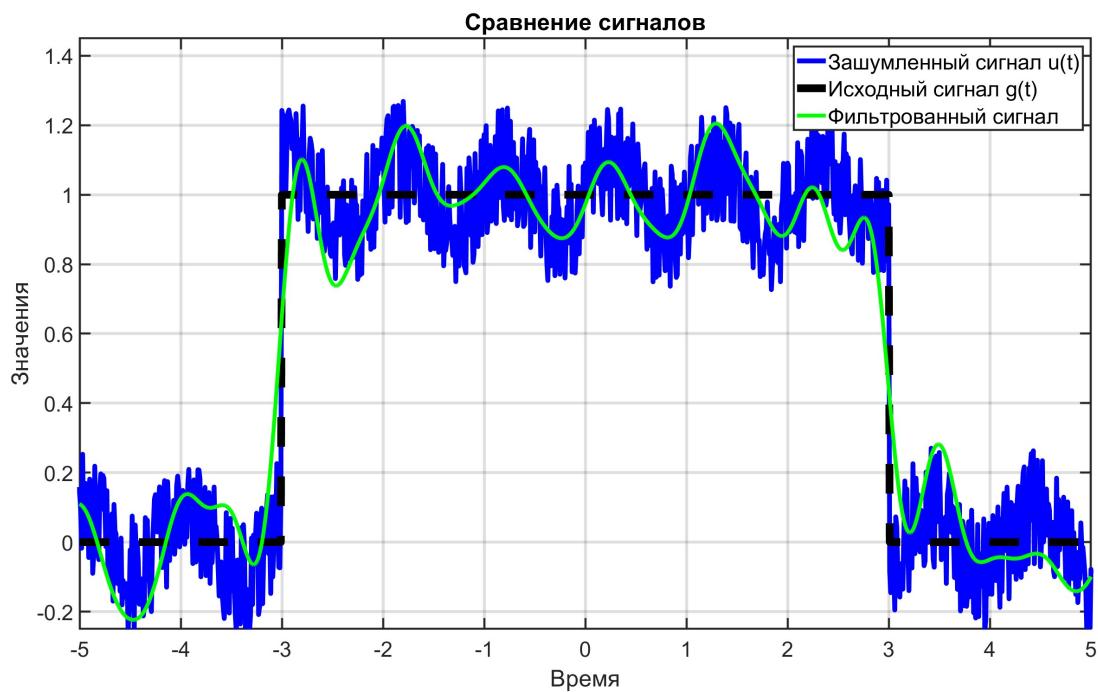
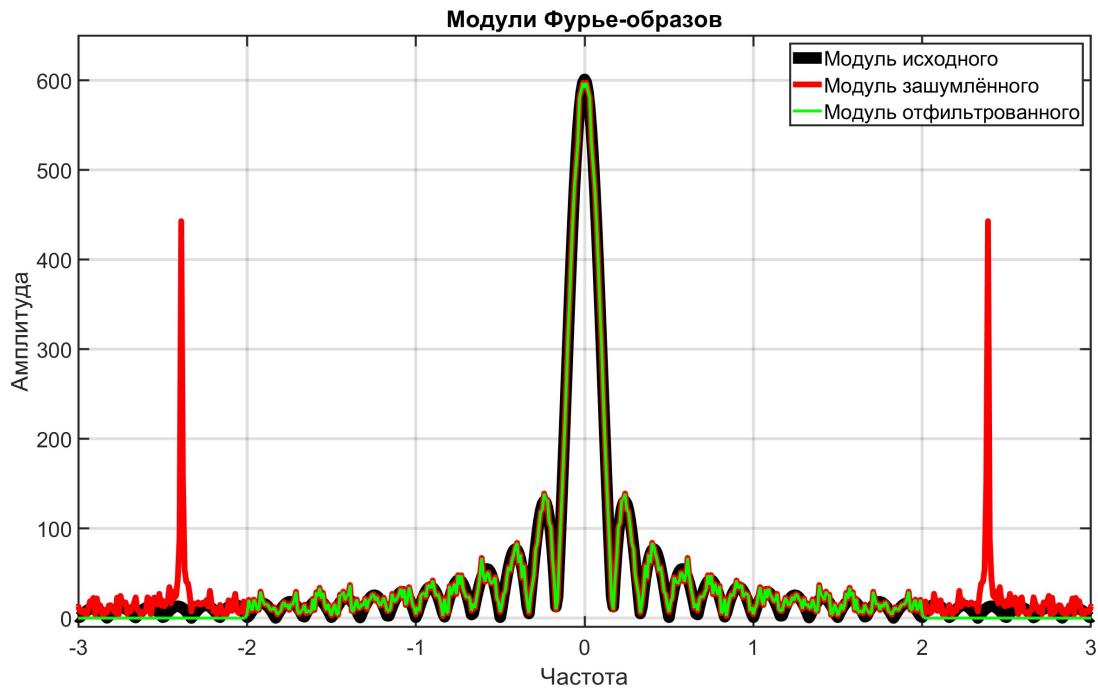
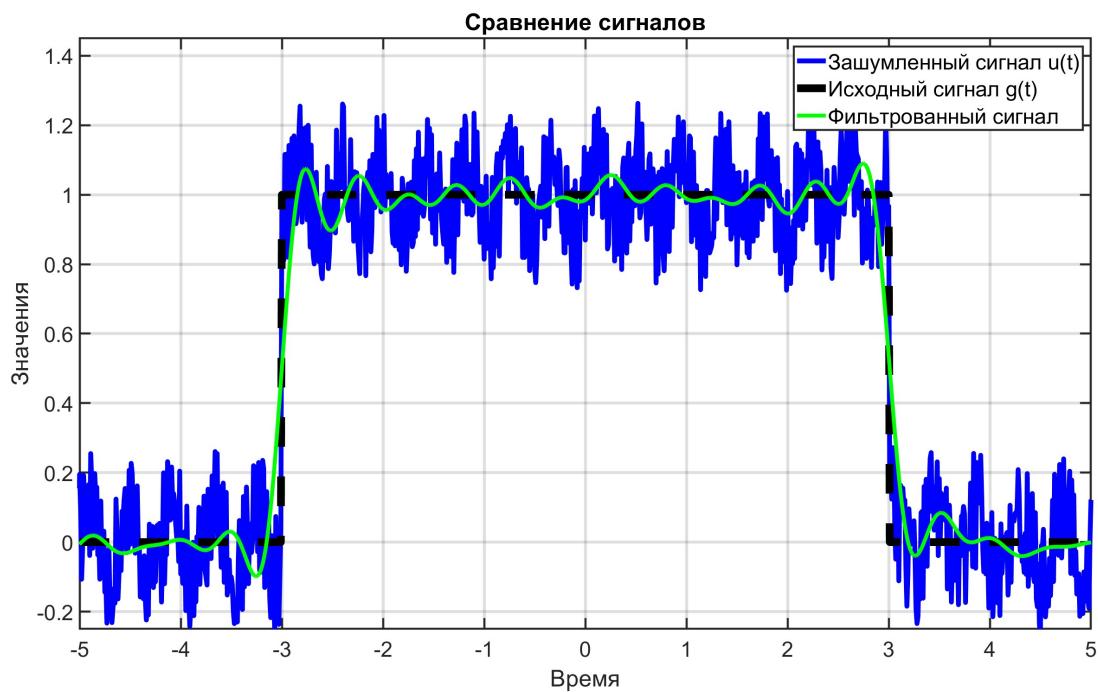


Рис. 25: Фильтрация при $\nu_0 = 2$, $\nu_1 = 0.5$, $h = 0.1$, $b = 0.35$, $c = 0.1$ и $d = 6$

Рис. 26: Образы при $\nu_0 = 2$, $b = 0.35$, $c = 0.1$ и $d = 15$ Рис. 27: Фильтрация при $\nu_0 = 2$, $b = 0.35$, $c = 0.1$ и $d = 15$

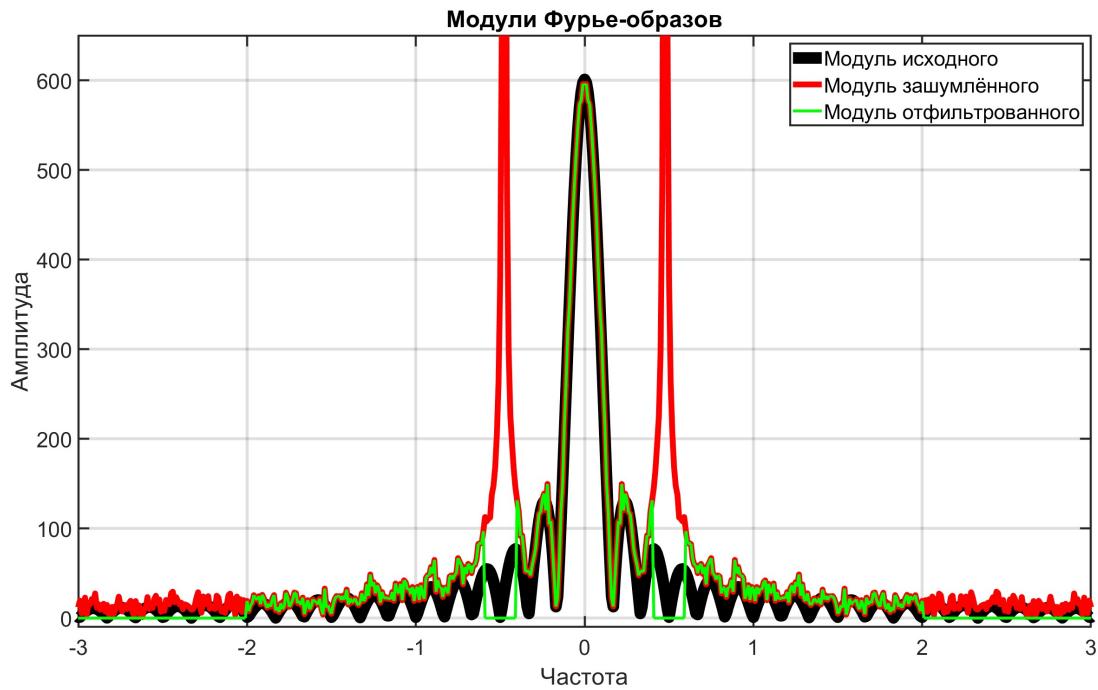


Рис. 28: Образы при $\nu_0 = 2$, $\nu_1 = 0.5$, $h = 0.1$, $b = 0.35$, $c = 0.75$ и $d = 3$

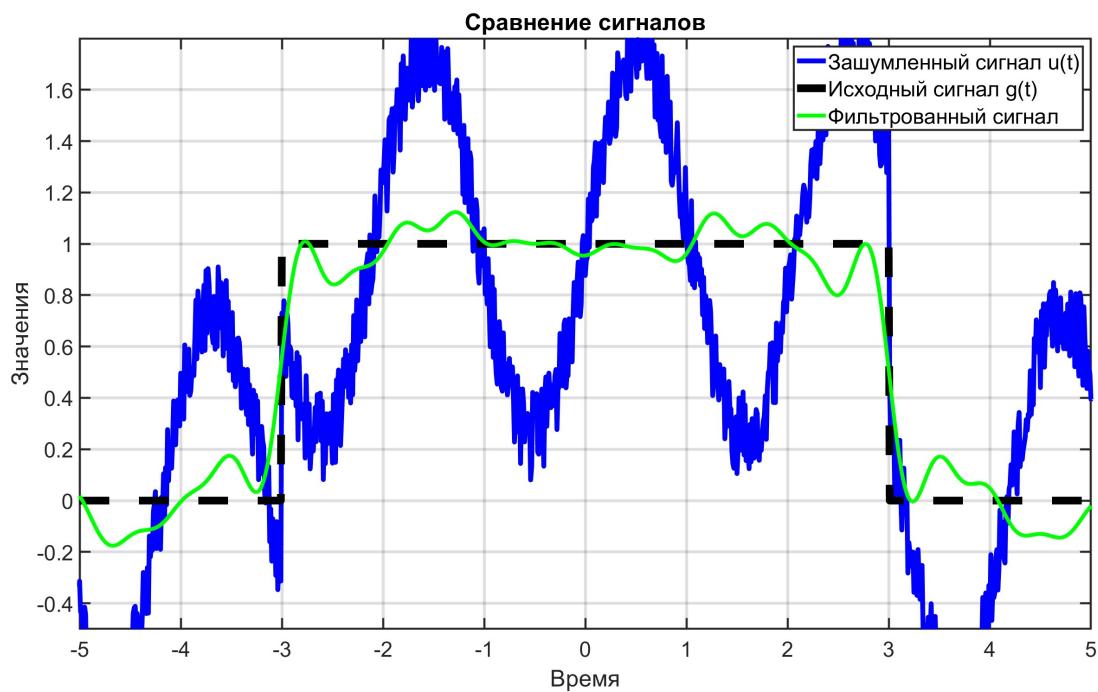


Рис. 29: Фильтрация при $\nu_0 = 2$, $\nu_1 = 0.5$, $h = 0.1$, $b = 0.35$, $c = 0.75$ и $d = 3$

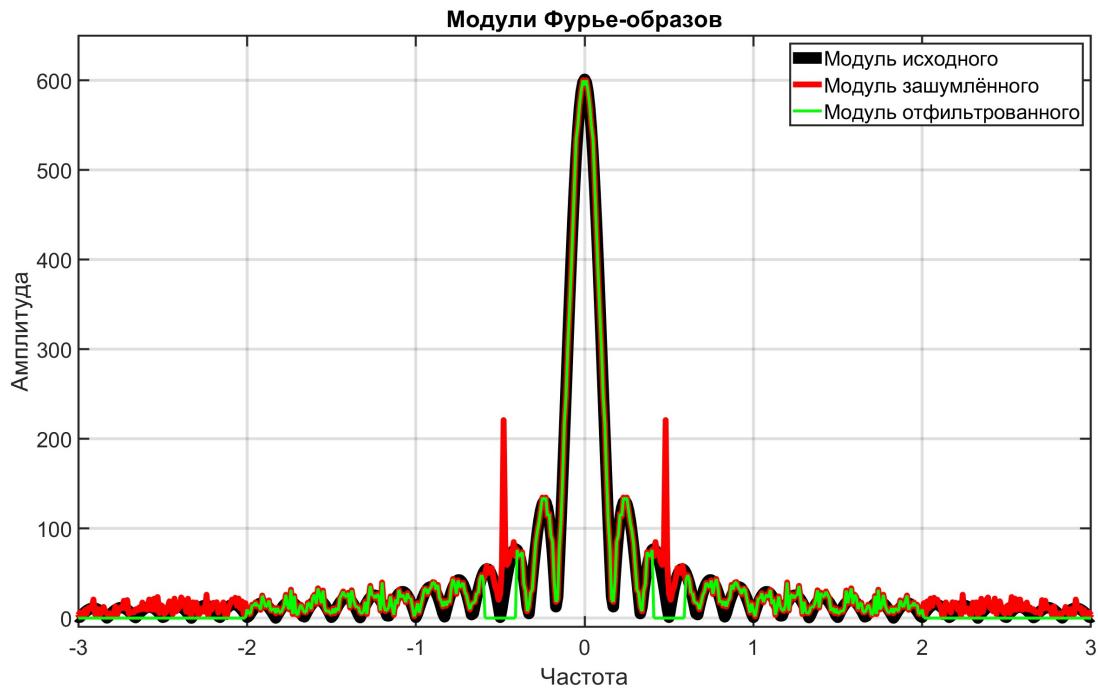


Рис. 30: Образы при $\nu_0 = 2$, $\nu_1 = 0.5$, $h = 0.1$, $b = 0.35$, $c = 0.05$ и $d = 3$

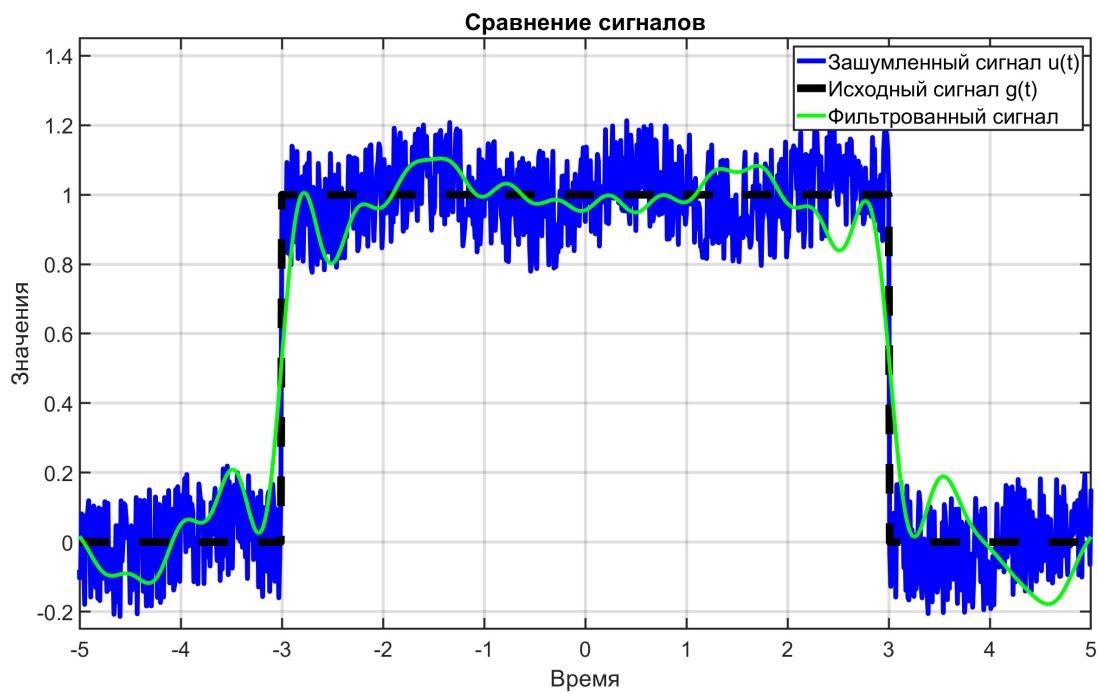


Рис. 31: Фильтрация при $\nu_0 = 2$, $\nu_1 = 0.5$, $h = 0.1$, $b = 0.35$, $c = 0.05$ и $d = 3$

можем быть вынуждены расширять и параметр ширины полосового фильтра h , из-за приближений **fft** дельта-функций). В остальном работа фильтров при вариации параметра c остаётся той же. Можно также заметить, что вариация c не влечёт никакого изменения в белом шуме, что подтверждает его независимость с гармоникой.

1.3 Убираем низкие частоты?

Рассмотрим фильтр, который обнуляет Фурье-образ на всех частотах в некоторой окрестности $[-\hat{\nu}, \hat{\nu}]$ точки $\nu = 0$. Зададимся всё теми же параметрами обрабатываемого сигнала $b = 0.35$, $c = 0.05$ и $d = 3$ и $\hat{\nu} = 1$. Посмотрим, что из этого выйдет. Результаты представлены на рисунках 32 и 33.

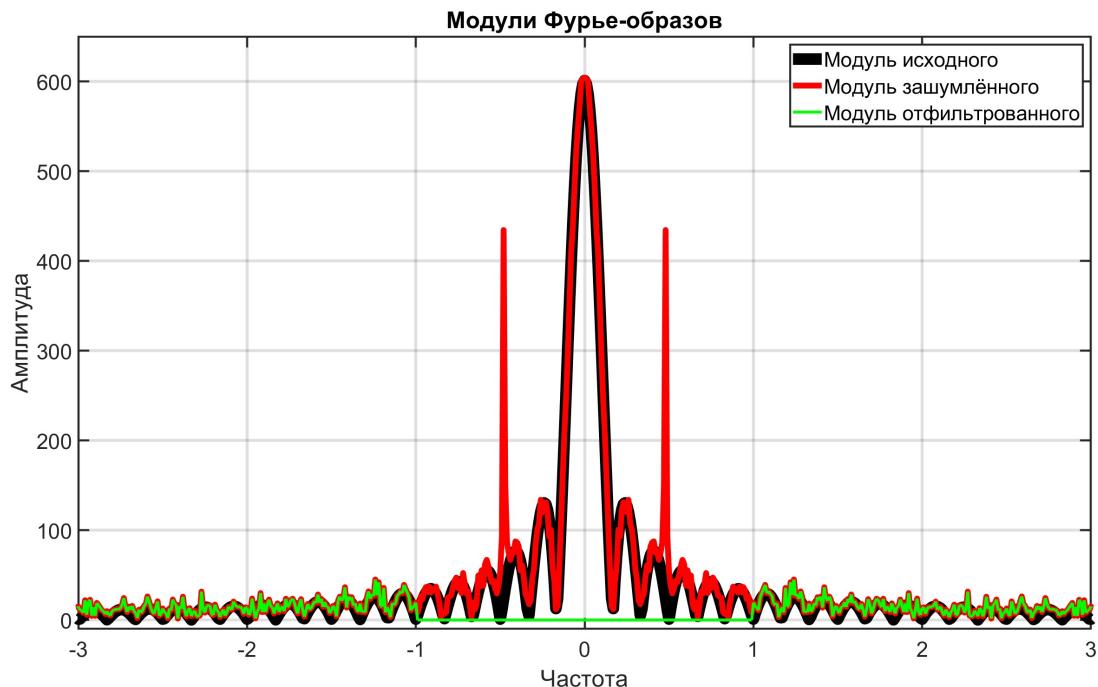
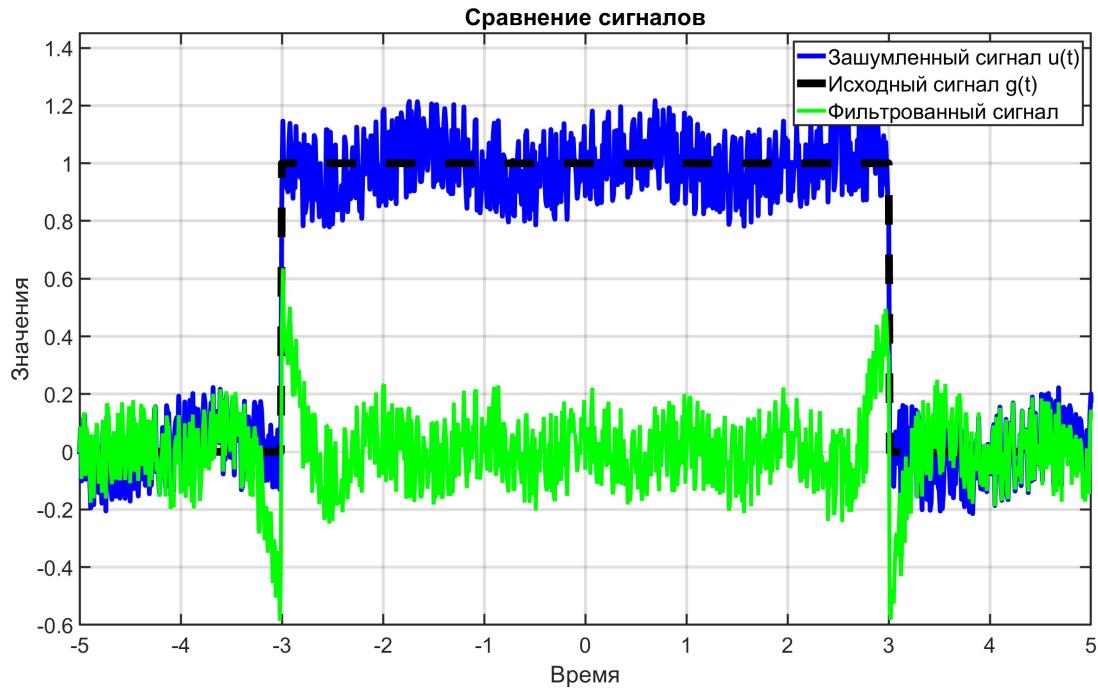
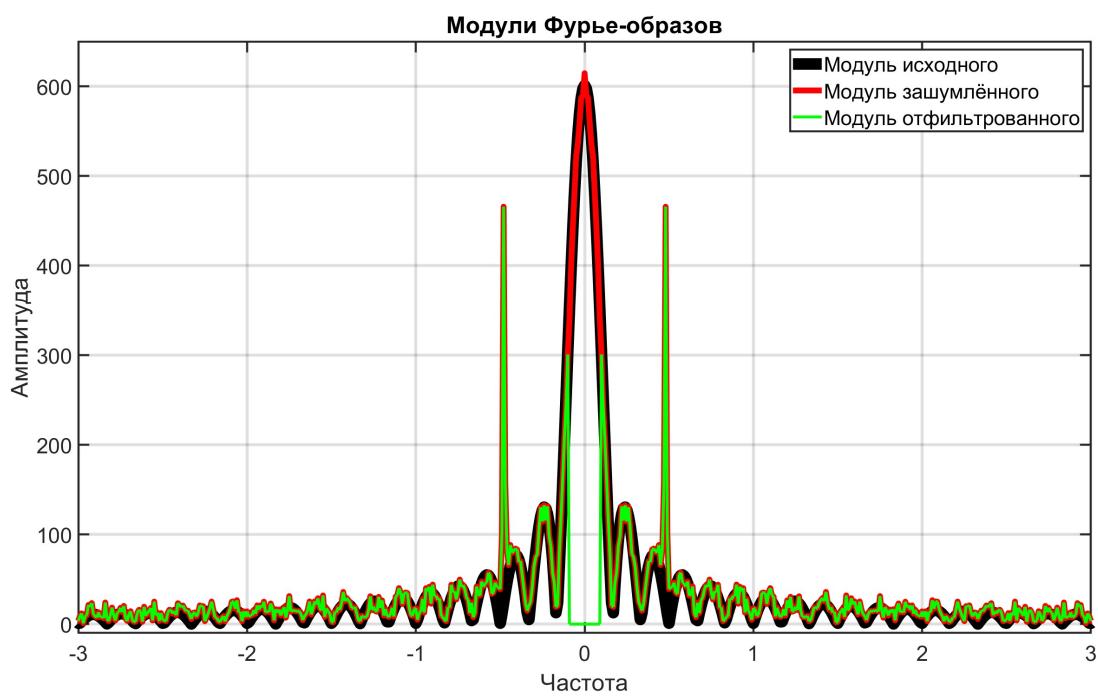
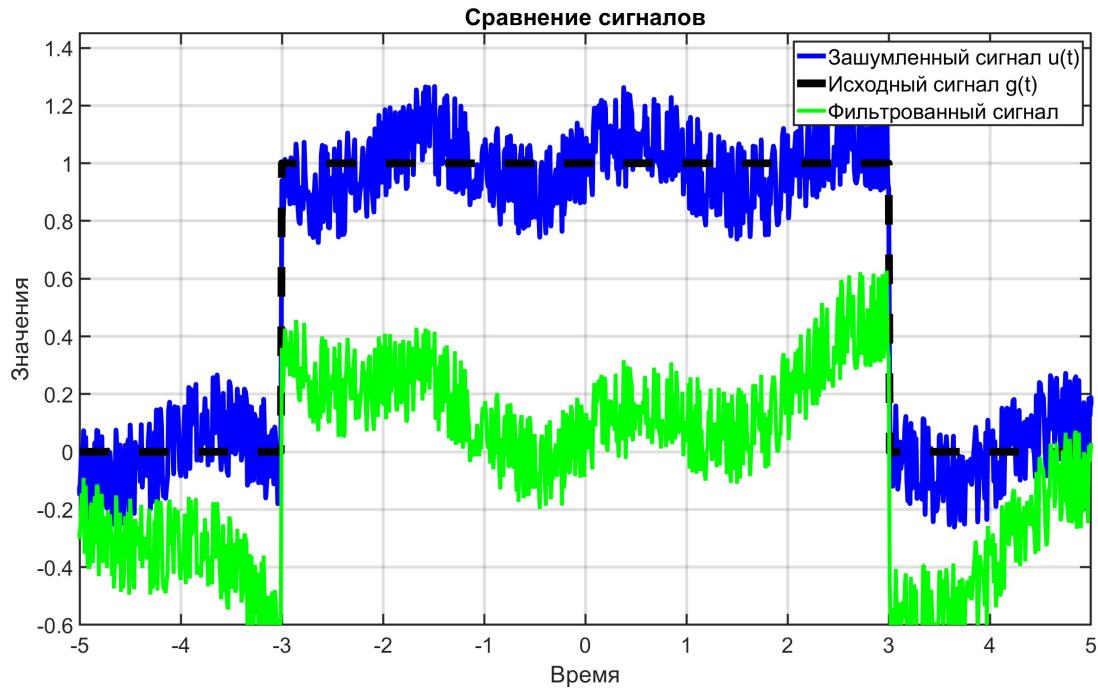
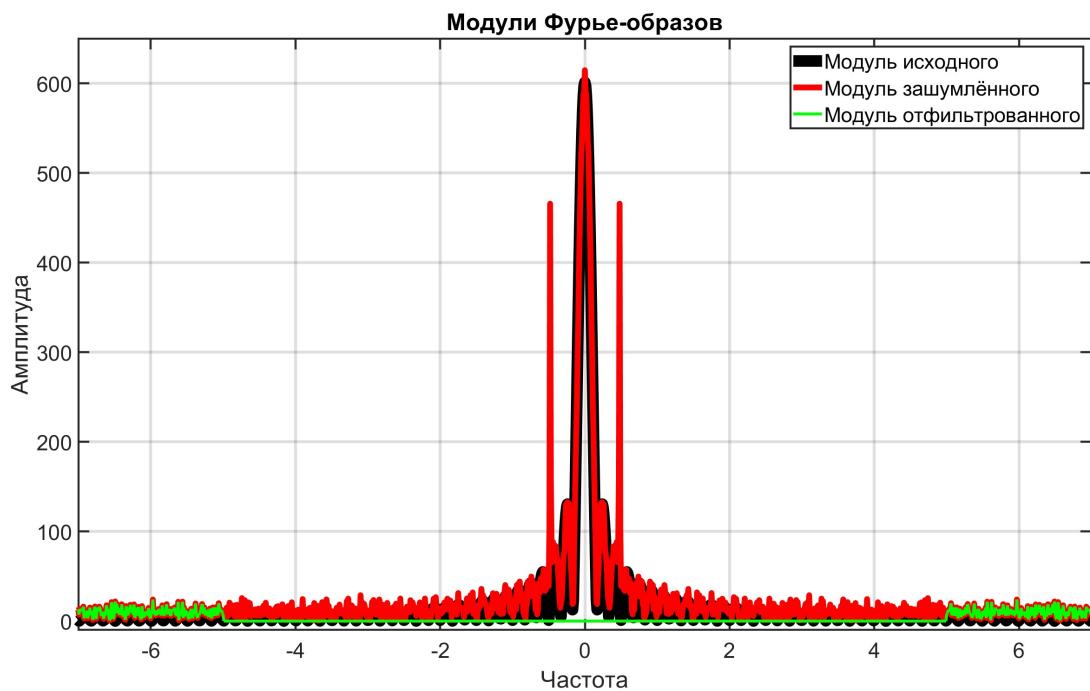


Рис. 32: Образы при $\hat{\nu} = 1$, $b = 0.35$, $c = 0.1$ и $d = 3$

Как видим, выделяются только высокочастотные составляющие сигнала (в нашем случае выделились белый шум и резкие переходы). Вариация параметра $\hat{\nu}$ от 0.1 до 5 на рисунках 34 и 35, а также 36 и 37 (при прочих неизменных). Соответственно, чем больше $\hat{\nu}$, тем

Рис. 33: Фильтрация при $\hat{\nu} = 1$, $b = 0.35$, $c = 0.1$ и $d = 3$ Рис. 34: Образы при $\hat{\nu} = 0.1$, $b = 0.35$, $c = 0.1$ и $d = 3$

Рис. 35: Фильтрация при $\hat{\nu} = 0.1$, $b = 0.35$, $c = 0.1$ и $d = 3$ Рис. 36: Образы при $\hat{\nu} = 0.25$, $b = 0.35$, $c = 0.1$ и $d = 3$

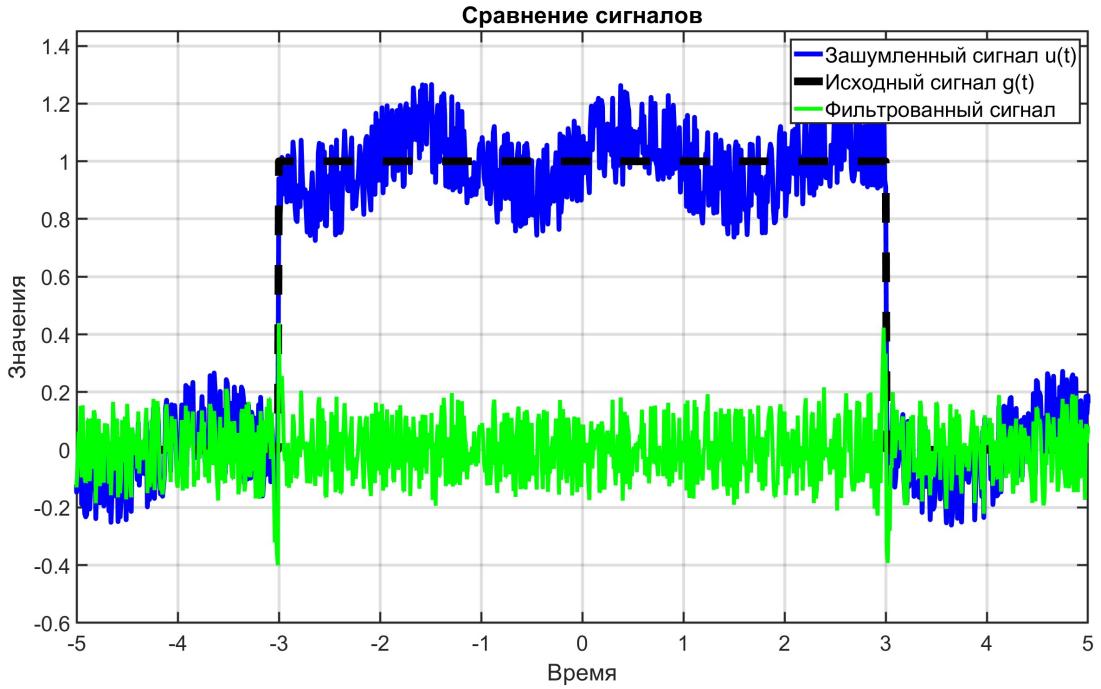


Рис. 37: Фильтрация при $\hat{\nu} = 0.25$, $b = 0.35$, $c = 0.1$ и $d = 3$

больший диапазон подавления низких частот применяется. Данный фильтр может использоваться в случае наличия низкочастотных помех (расположения самого сигнала при этом должно находиться на более высоких частотах, так как иначе теряется информация и появляются искажения в исходных составляющих) или же просто нежелательных компонент, которые мы хотим подавить, получив некоторую обработку.

Таким образом, показано, что фильтр низких частот эффективно устраниет высокочастотный шум, при этом параметр ν_0 критически влияет на степень сглаживания (увеличение влечёт большую силу фильтрации), а параметр b — на форму обрабатываемого сигнала. Применение полосовых фильтров вместе с низкочастотными позволило подавить все существующие помехи, включая гармонику и случайный шум. Отдельный случай $b = 0$ в данном контексте также продемонстрировал отсутствие шумовой составляющей и подчеркнул необходимость точной настройки фильтра (так как в этом случае мы смогли практически без искажений восстановить исходный

сигнал). Фильтрация же низких частот показала, что даже удаление небольшого диапазона около нуля может сильно искажить общий вид сигнала, особенно при наличии медленно меняющихся компонент.

2 Фильтрация звука

Скачаем файл **МИНА.wav**. При его прослушивании был замечен голос и шумы, одним из которых был явный низкий гул. Сделаем так, чтобы остался только голос. Для этого для начала найдём образ Фурье аудиофайла:

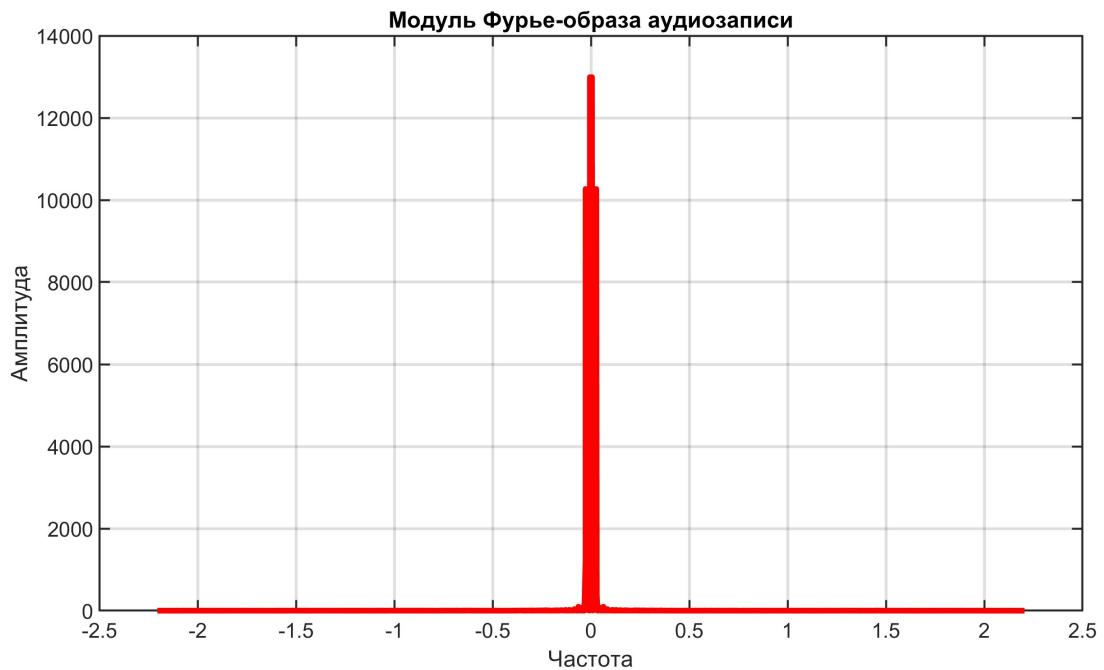


Рис. 38: Модуль Фурье-образа аудиозаписи

В центре видны явные отклонения – голос такими амплитудами частот не обладает. Рисунок 39 показывает, что данные помехи находятся на отрезке $[-300, 300]$ ($\hat{\nu} = 300$), то есть мы можем их подавить изученным в предыдущем пункте фильтром верхних частот. Сделав данное, мы уберём тот самый постоянный низкий гул.

Далее, на записи были слышны высокочастотные помехи (что-то вроде жужжания сверчков). Подтверждают это и графики: на исход-

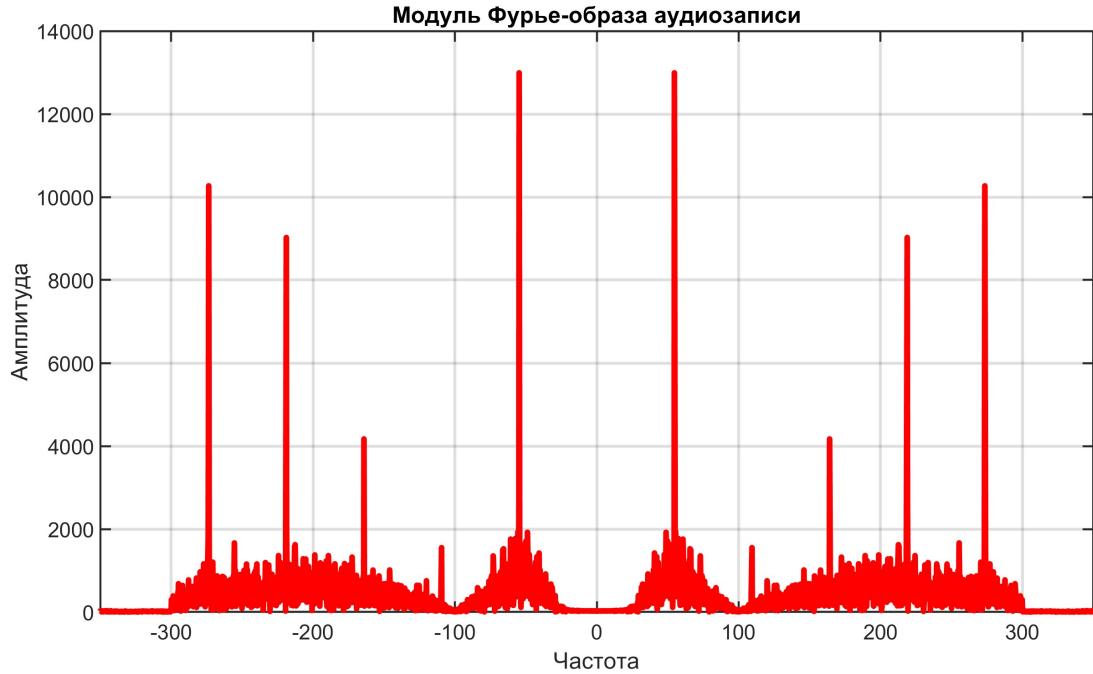


Рис. 39: Модуль Фурье-образа аудиозаписи (приближение в центре)

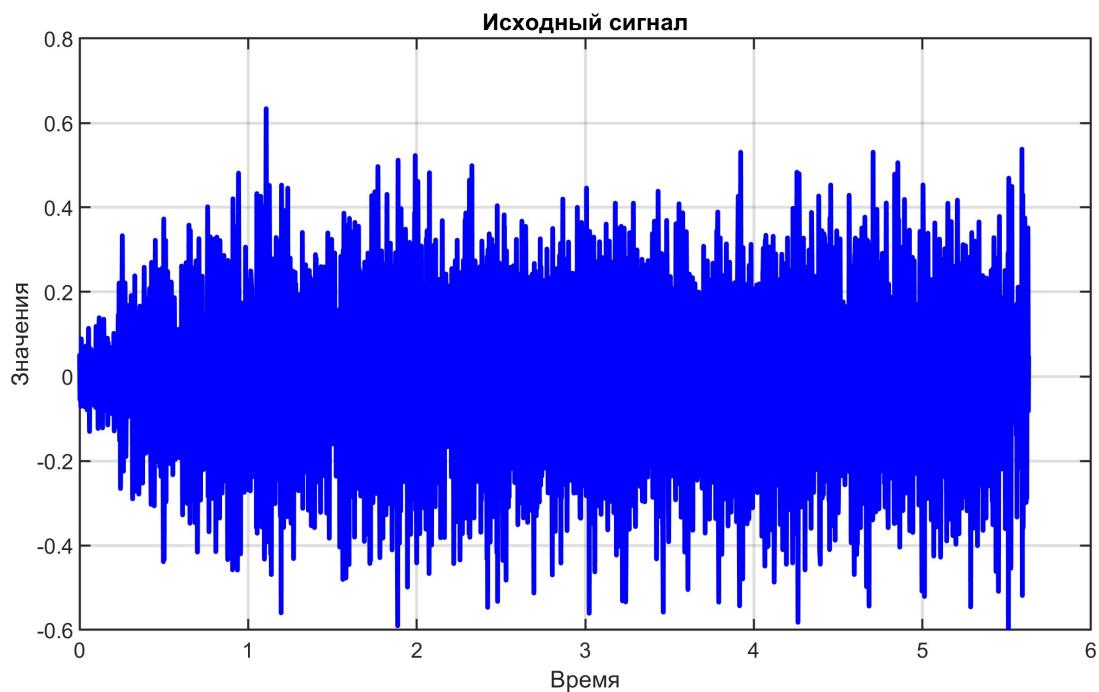


Рис. 40: График исходной аудиозаписи

ная запись явно сильно колеблется (рисунок 40). Экспериментально было получено, что фильтр низких частот при $\nu_0 = 5000$ успешно их уничтожает, поэтому будем использовать его.

По итогу, обработка аудиозаписи свелась к последовательному применению фильтра низких частот ($\hat{\nu} = 300$), а после фильтра верхних частот ($\nu_0 = 5000$). В конце остался только чистый голос мужчины, говорящего: «муха - это маленькая птичка». Сравнение образов фильтрованной записи и исходной приведено на рисунке 41, сравнение фильтрации во временной области - на рисунке 42.

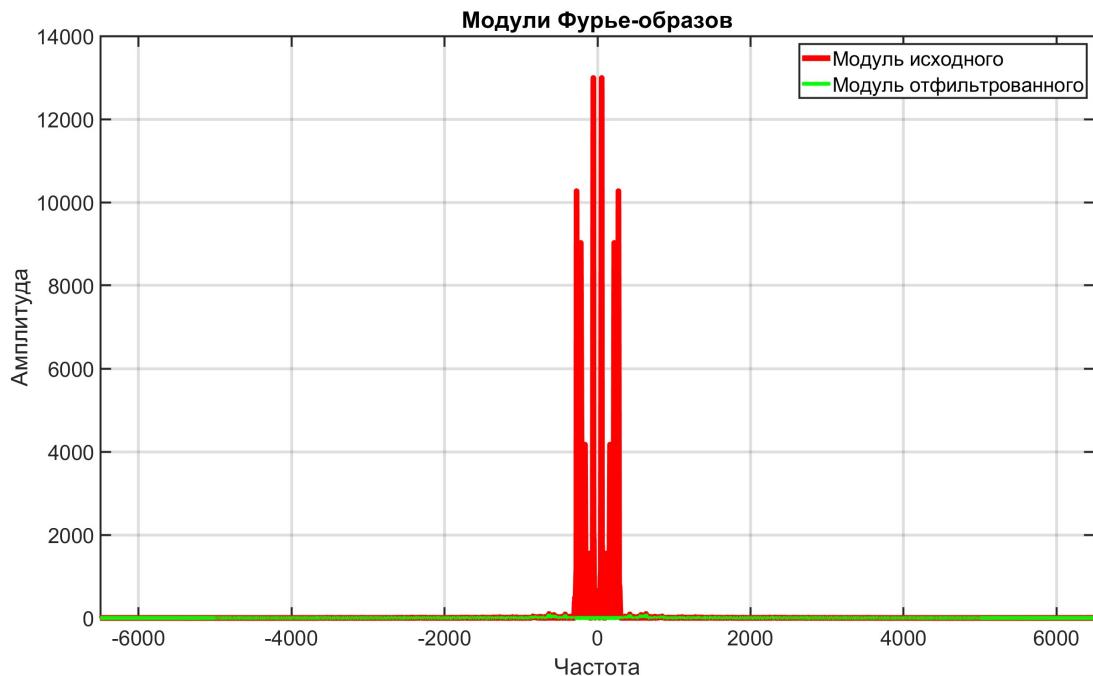


Рис. 41: Сравнение модулей Фурье-образов

3 Общие выводы

В ходе лабораторной работы было проведено исследование различных методов жесткой фильтрации сигналов на основе преобразования Фурье. Были изучены эффекты удаления высоких, специфических и низких частот, а также проанализировано влияние параметров фильтров на качество восстановления сигнала. На каждом

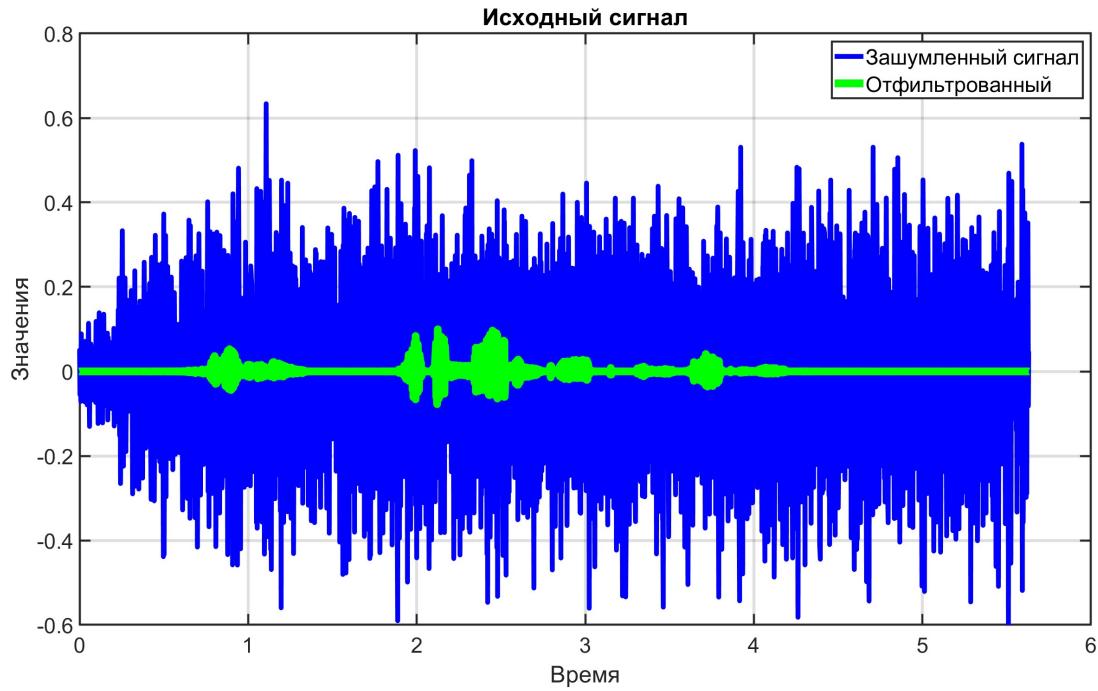


Рис. 42: Сравнение сигналов во временной области

этапе также выполнено сравнение образов исходного, зашумленного и отфильтрованного сигналов, что позволило более точно выявить оптимальные значения параметров фильтрации. Кроме того, успешно выполнена фильтрация звукового сигнала, при которой удалось отделить голос от шума высокого и низкого шумов. В общем, работа показала эффективность жёсткой фильтрации для подавления помех самой разной природы.

4 Приложение

Листинг 1: Код для жёсткой фильтрации

```

1 a = 1;
2 t1 = -3;
3 t2 = 3;
4 T = 100;
5 dt = 0.01 ;
6 t = -T/2:dt:T/2;
7 N = length(t);
8 g = zeros(size(t));
9 g(t ≥ t1 & t ≤ t2) = a;
10
11 c = 0.1 ;
12 b = 0.35 ;
13 d = 3;
14 u = g + b*(rand(size(t))-0.5) + c*sin(d*t);
15
16 %%%
17 G = fftshift(fft(g));
18 U = fftshift(fft(u));
19 V = 1/dt;
20 dnu = 1/T;
21 nu = -V/2:dnu:V/2;
22 U_filt = U;
23
24 %%%
25 n0 = 2;
26 U_filt(nu < -n0 | nu > n0 ) = 0;
27
28 %%%
29 n1 = 0.5 ;
30 h = 0.1 ;
31 U_filt(nu < n1 + h & nu > n1 - h ) = 0;
32 U_filt(nu < -n1 + h & nu > -n1 - h ) = 0;
33
34 %%%
35 hatnu = 5;
36 U_filt(nu > -hatnu & nu < hatnu) = 0;
37
38 %%%
39 u_filtered = ifft(ifftshift(U_filt));

```

Листинг 2: Код для фильтрации аудиозаписи

```
1 [y, f] = audioread('MUHA.wav');
2 %sound(y, f);
3
4 dt = 1/f;
5 T = length(y)*dt;
6 t = 0:dt:T-dt;
7
8 Y = fftshift(fft(y));
9 magY = abs(Y);
10 N = length(Y);
11 nu = (-N/2:N/2-1)*(f/N);
12
13 hatnu = 5;
14 n1 = 300;
15 n2 = 5000;
16 Y_filtered = Y;
17 Y_filtered(nu > -n1 & nu < n1) = 0;
18 Y_filtered(nu < -n2 | nu > n2) = 0;
19
20 y_filtered = real(ifft(ifftshift(Y_filtered)));
21 sound(y_filtered, f);
```