МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Университет ИТМО

Факультет систем управления и робототехники

ОТЧЁТ
по дисциплине
"Частотные методы"

по теме:
ЛИНЕЙНАЯ ФИЛЬТРАЦИЯ

Студент:
Группа R3236 *Поляков А.А.*

Предподаватель:
к.т.н., доцент *Перегудин А.А.*

Санкт-Петербург
2024

СОДЕРЖАНИЕ

1 ИСХОДНЫЙ КОД

Онлайн версию кода здесь нет, потому что делал основные вычисления в live-script матлабовских, в репозитории можно найти исходники.

Особо много кода я не писал, скорее пытался красиво запрогать графики... Основные фишки кода были помечены сразу там в комментариях, поэтому прошу всех желающих заглянуть под капот непосредственно

2 ЗАДАНИЕ 1. СПЕКТРАЛЬНОЕ ДИФФЕРЕНЦИРОВАНИЕ

2.1 Лирическое вступление

В данном задании мы рассматриваем сигнал `y=sin(t)` с небольшим шумом уже знакомого вида - `a*(rand(size(t))-0.5)`

Для начала найдём численную производную от зашумлённого сигнала через поэлементную формулу:

$$\frac{y(k+1)-y(k)}{dt}$$

Найдём численную спектральную производную от зашумлённого сигнала, использую свойство Фурье-оператора:

$$\mathbb{F}(\frac{df}{dt}) = 2\pi i\nu\mathbb{F}(f)$$

То есть, сначала мы получаем Фурье-образ от обычного сигнала, потом домнажаем поэлементно на константы, это нас сразу сводит в Фурье-образ производной от сигнала, а теперь возвращаемся в мир обыкновенных сигналов с помощью обратного преобразования Фурье.

2.2 Графики компонент Фурье-образа сигнала и его спектральной производной

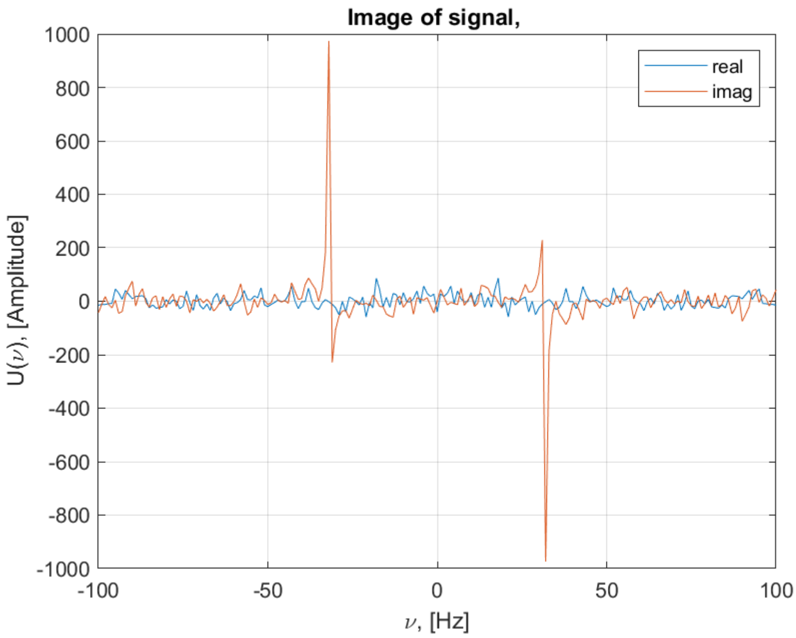

Рисунок 1 — Фурье-образ численной производной сигнала

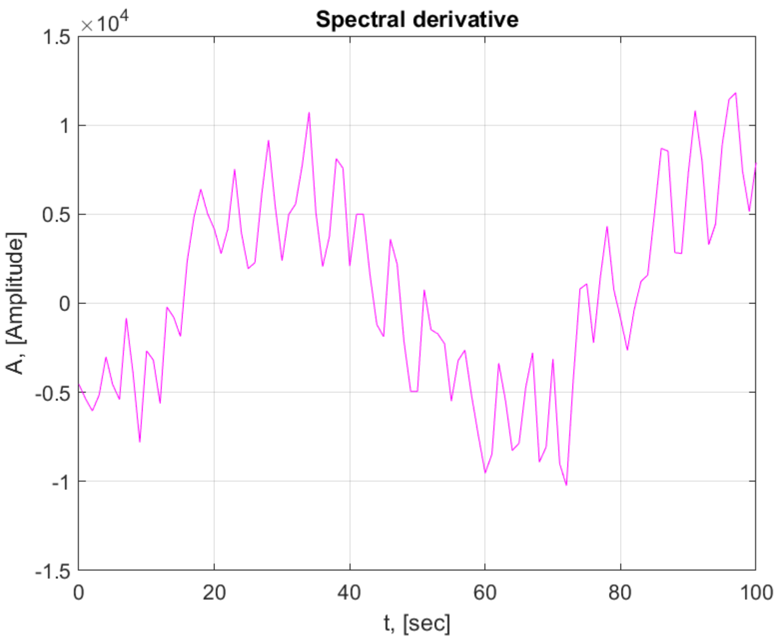

Рисунок 2 — Спектральная производная

2.3 Графики для сравнения

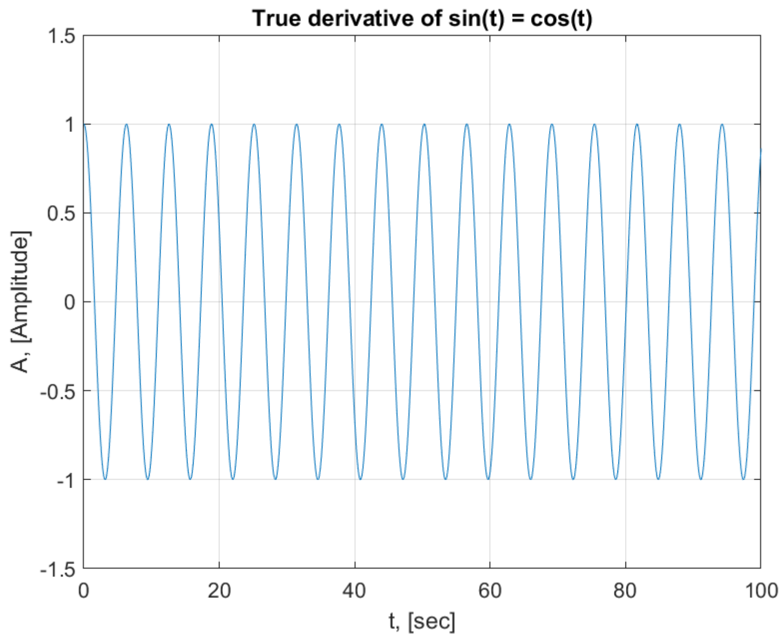

Рисунок 3 — Истинная производная

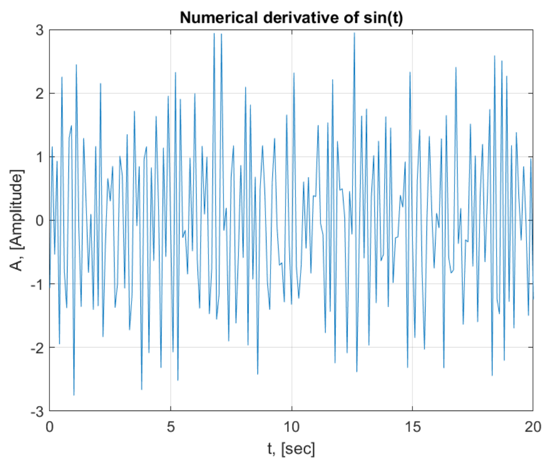

Рисунок 4 — Численная производная

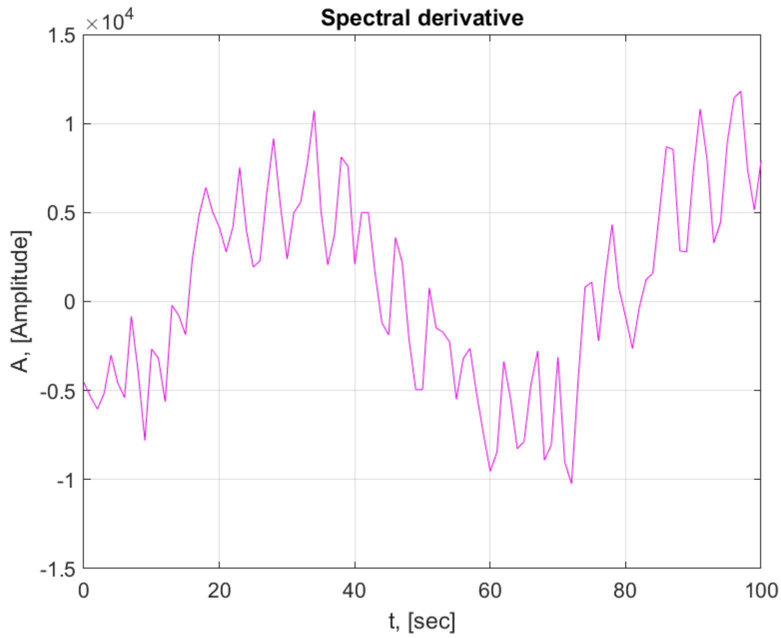

Рисунок 5 — Спектральная производная

2.4 Делаем выводы

Если не приближать и не рассматривать эти три графика вблизи, то кажется, что численная и спектральная - *просто гармонический шум*... Если подобрать удачные отрезки рассмотрения, то прослеживается тот факт, что они очень сильно хотят напоминать график оригинальной производной, но не могут. Думаю, что основное достижение сейчас - это то, что **мы смогли получить производную** без численного приближения только с помощью оператора Фурье и прямого, обратного преобразования.

3 ЗАДАНИЕ 1. ЛИНЕЙНЫЕ ФИЛЬТРЫ

Возьмём шумный сигнал из прошлой лабы, но будем применять оружие покрупнее:

```
u = g + b*(rand(size(t))-0.5) + c*sin(d*t);
```

3.1 Фильтр первого порядка

Линейный фильтр первого порядка мы зададим следующим образом:

$$W_1(p) = \frac{1}{Tp + 1}$$

Берём $d = c = 0$, также зададим постоянную времени $T > 0$ для фильтра. Тогда в этом пункте мы будем работать со следующей версией шумного сигнала:

```
u = g + b*(rand(size(t))-0.5)
```

…из чего сразу следует, что у нас добавляется только "случайный"шум.

3.1.1 Испытания

Построим сравнительные графики исходного и фильтрованного сигналов, графики модулей их Фурье-образов, а также АЧХ и ФЧХ фильтра. Чтобы возможно упростить общий анализ, я свёл все эти графики в подграфики.

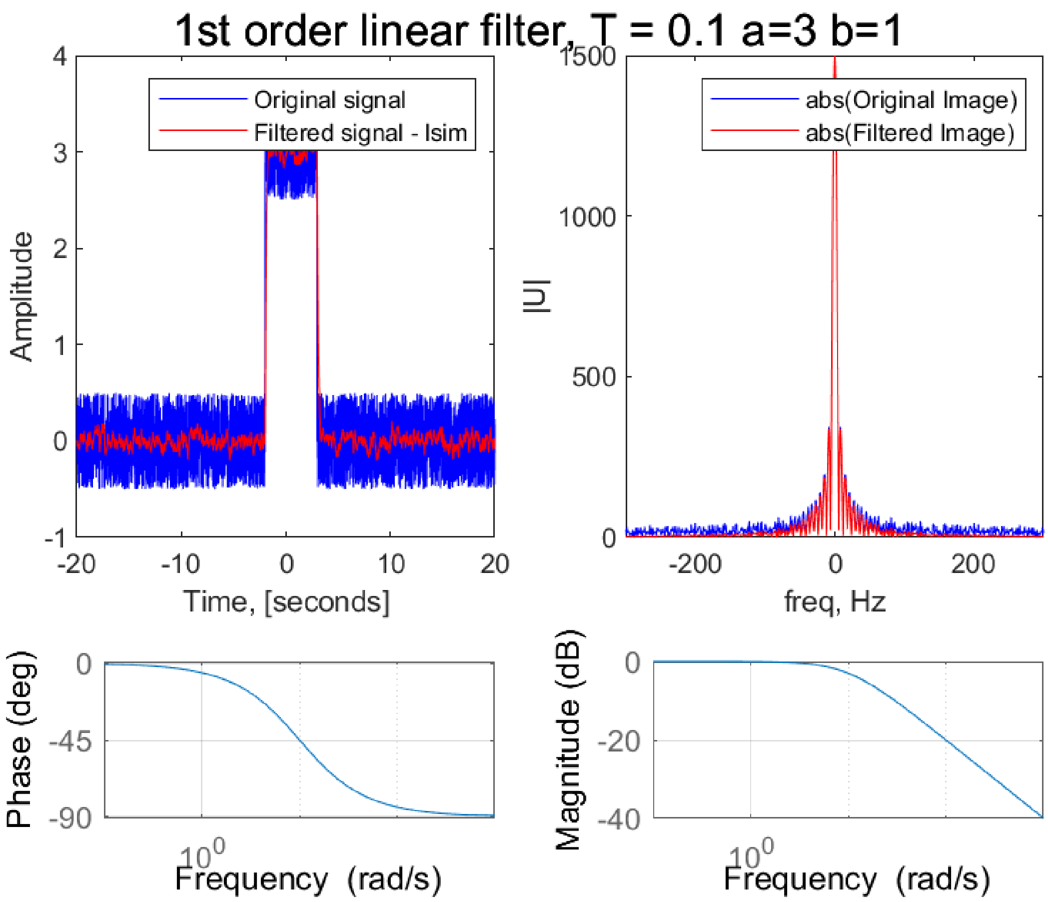

Рисунок 6 — Испытание 1

3.1.2 Выводы

Давайте исследовать влияние постоянной времени T и значения параметра a на эффективность фильтрации. Возможно это не слишком заметно, но параметр a у нас влияет на добавление белого шума и как бы мы не пытались увеличивать его амплитуду, - фильтр все равно более менее хорошо справлялся с таким шумом. Но фильтр в первую очередь справлялся из-за хорошо подобранной постоянной времени - больше единицы ставить не было смысла, потому что результат плачевный. Поэтому подбирая на ощупь в пределах $[0; 1]$ с шагом 0.1 можно было достичь приемлимых результатов.

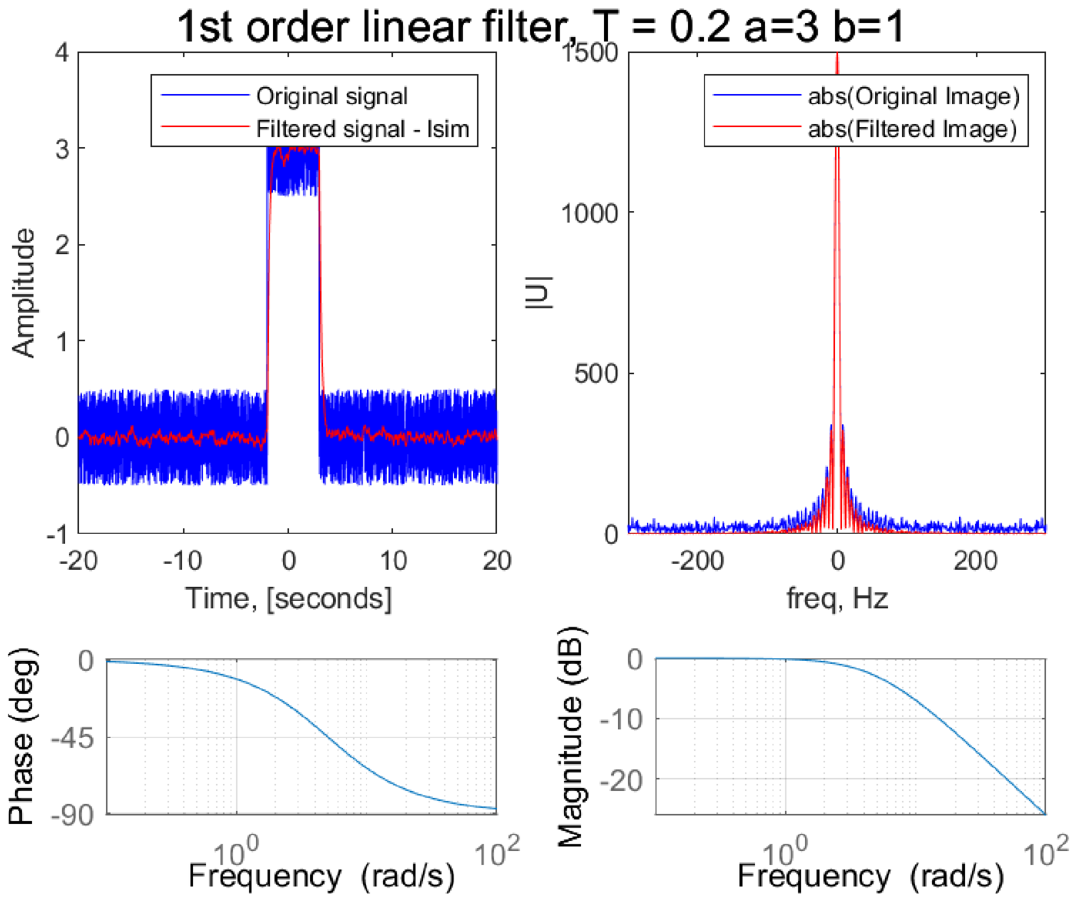

Рисунок 7 — Испытание 2

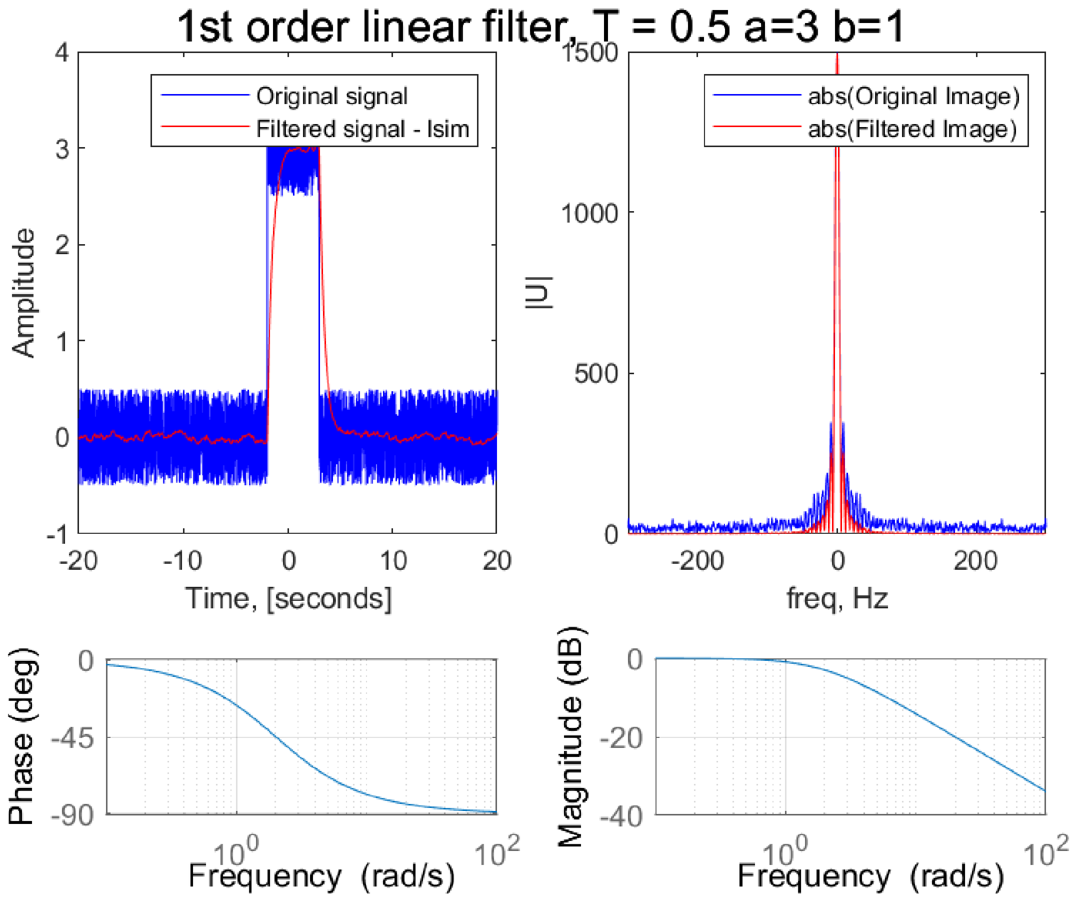

Рисунок 8 — Испытание 3

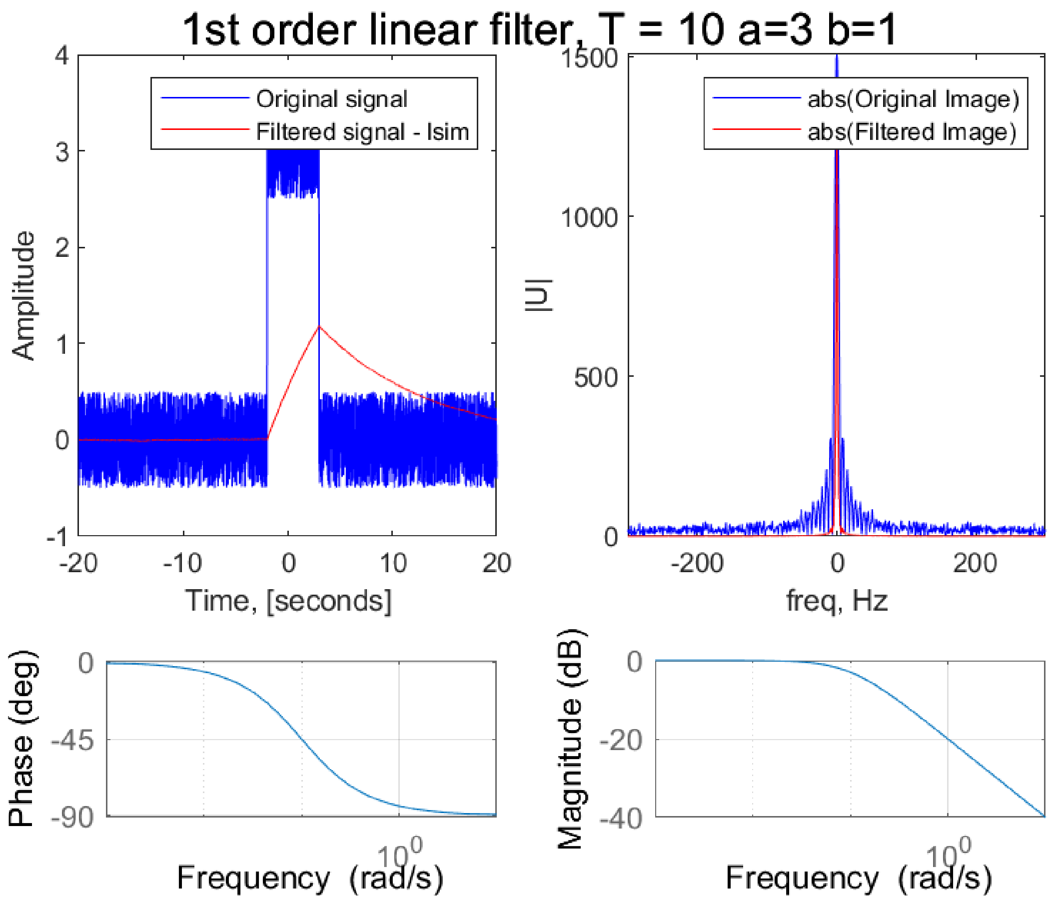

Рисунок 9 — Испытание 4

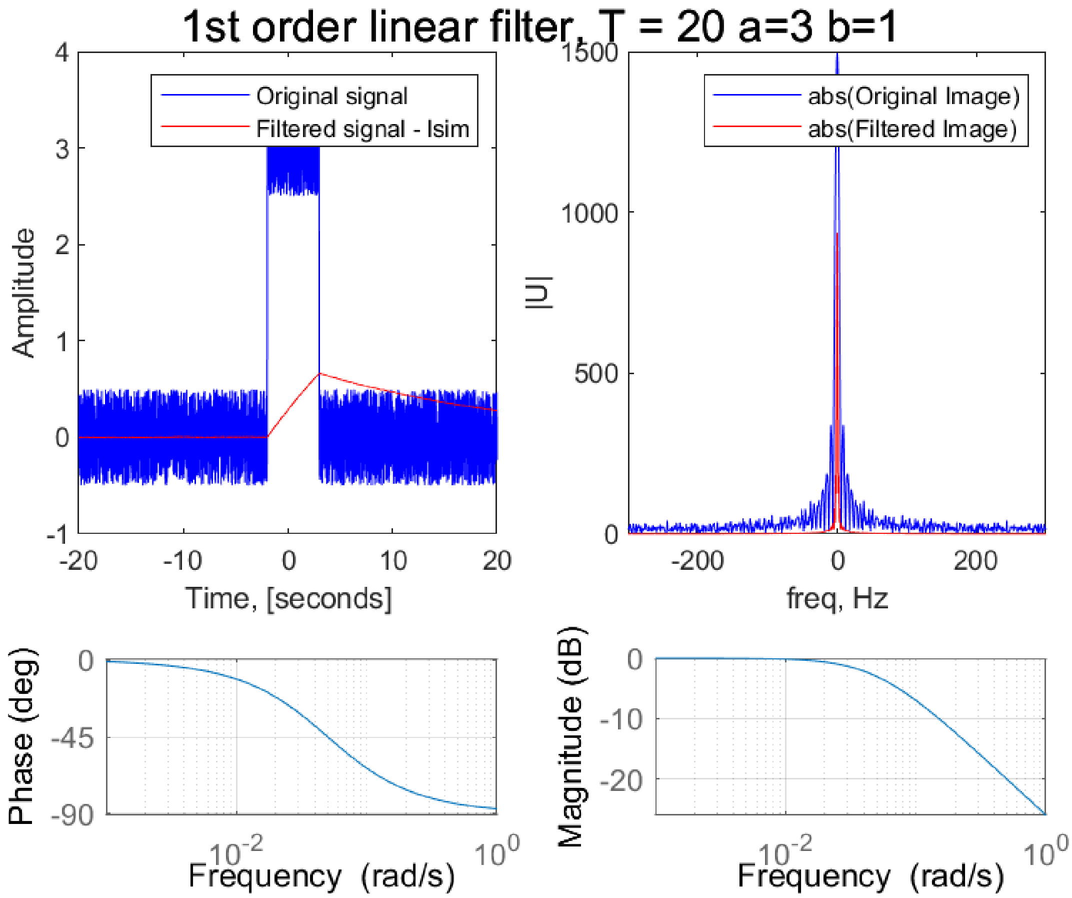

Рисунок 10 — Испытание 5

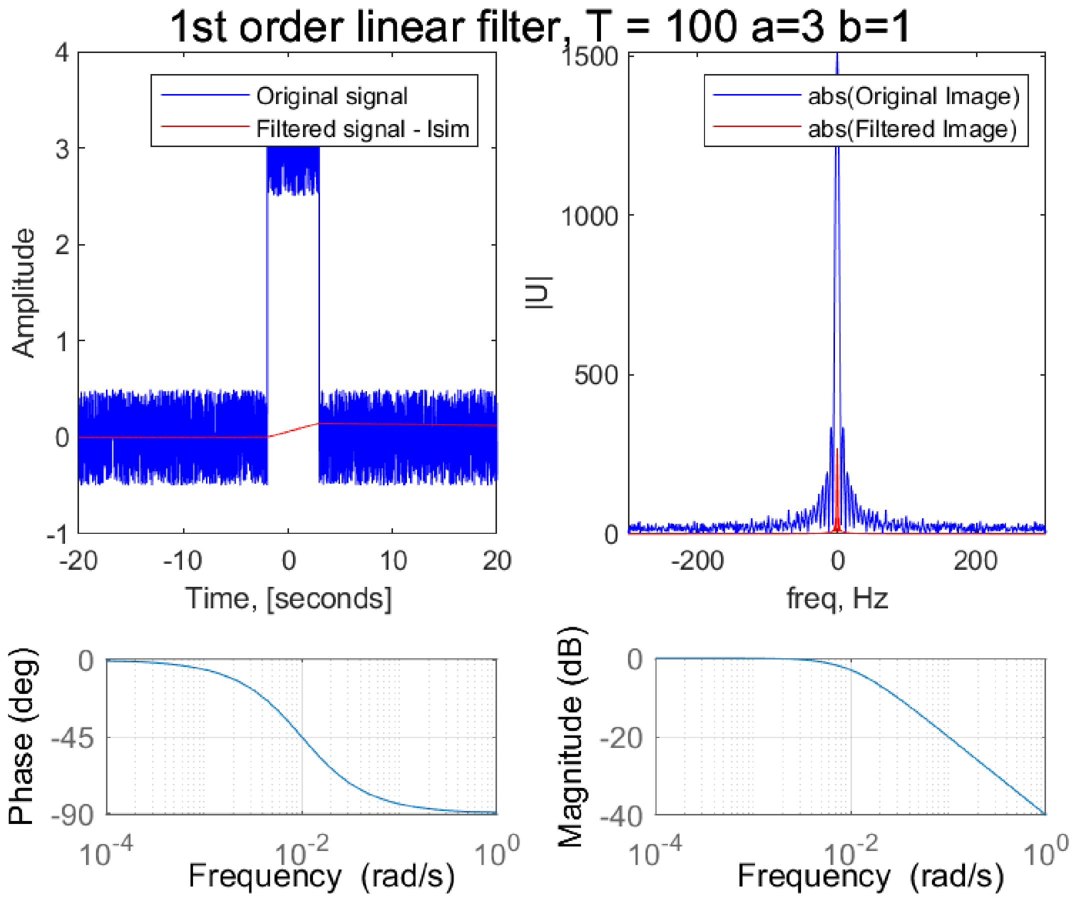

Рисунок 11 — Испытание 6

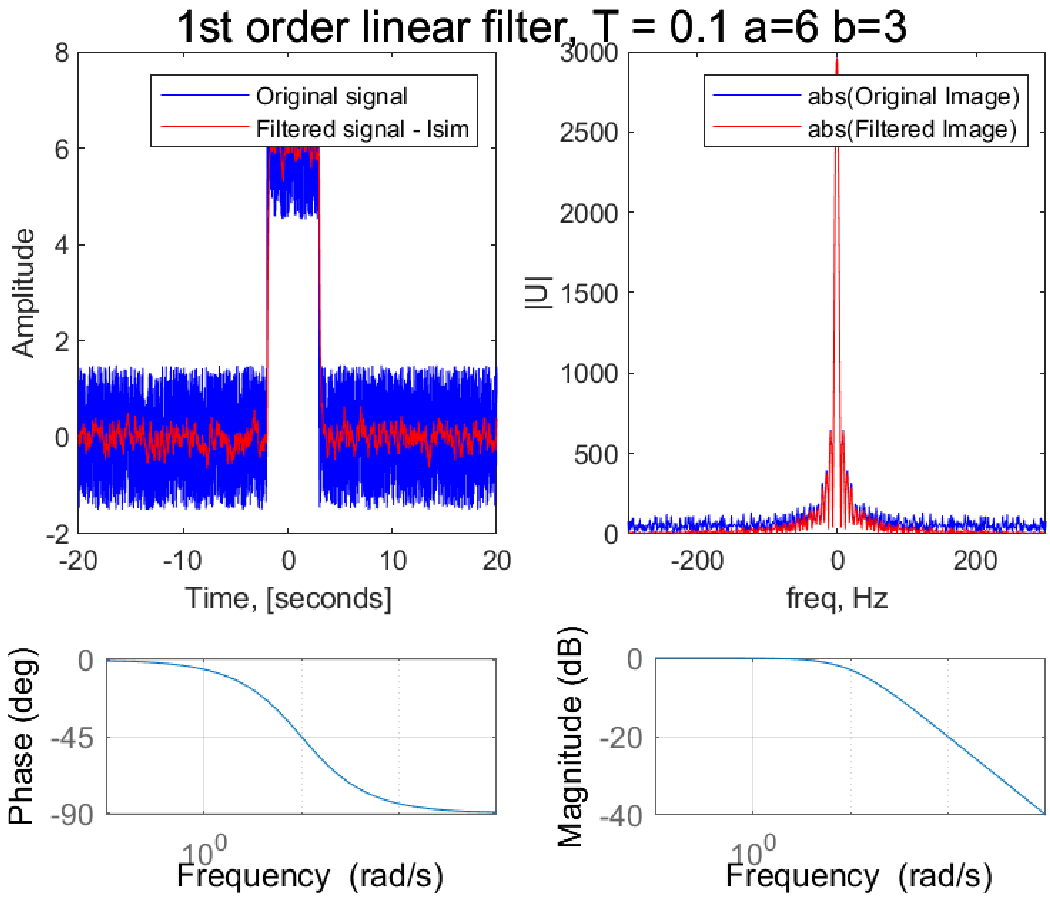

Рисунок 12 — Испытание 7

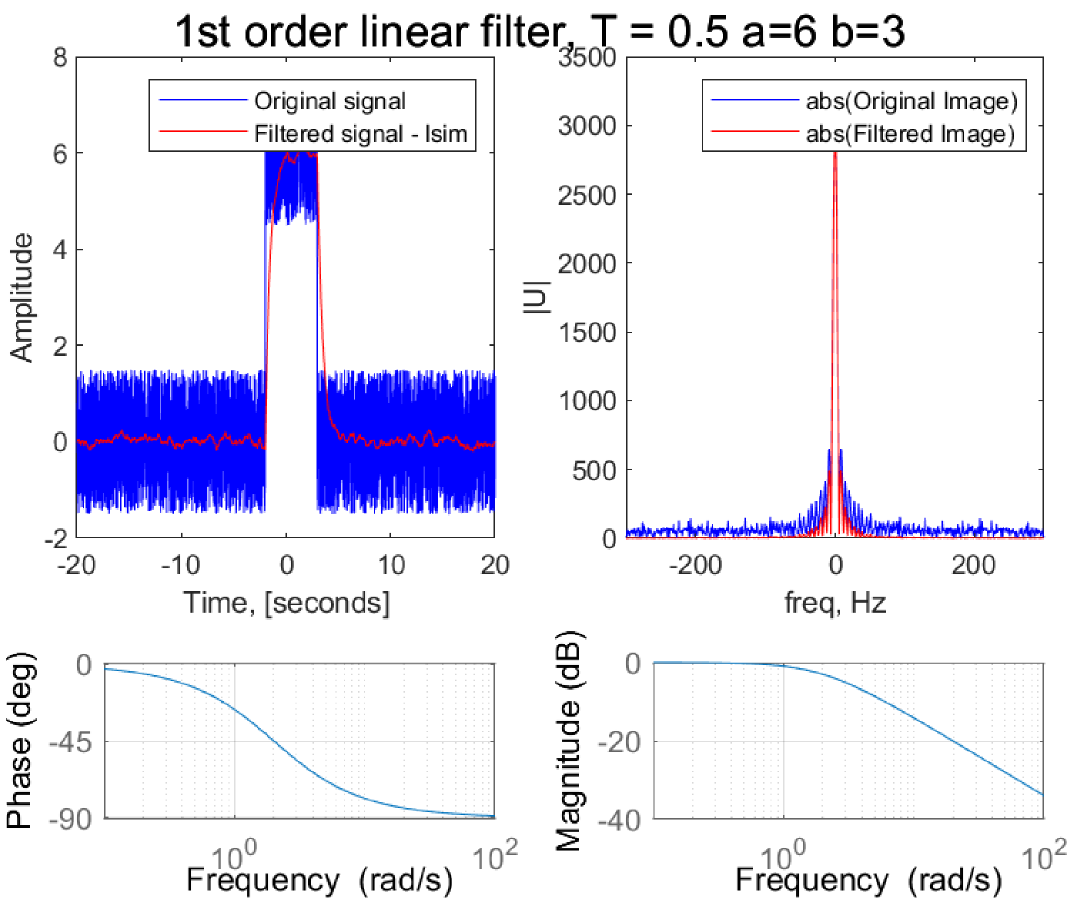

Рисунок 13 — Испытание 8

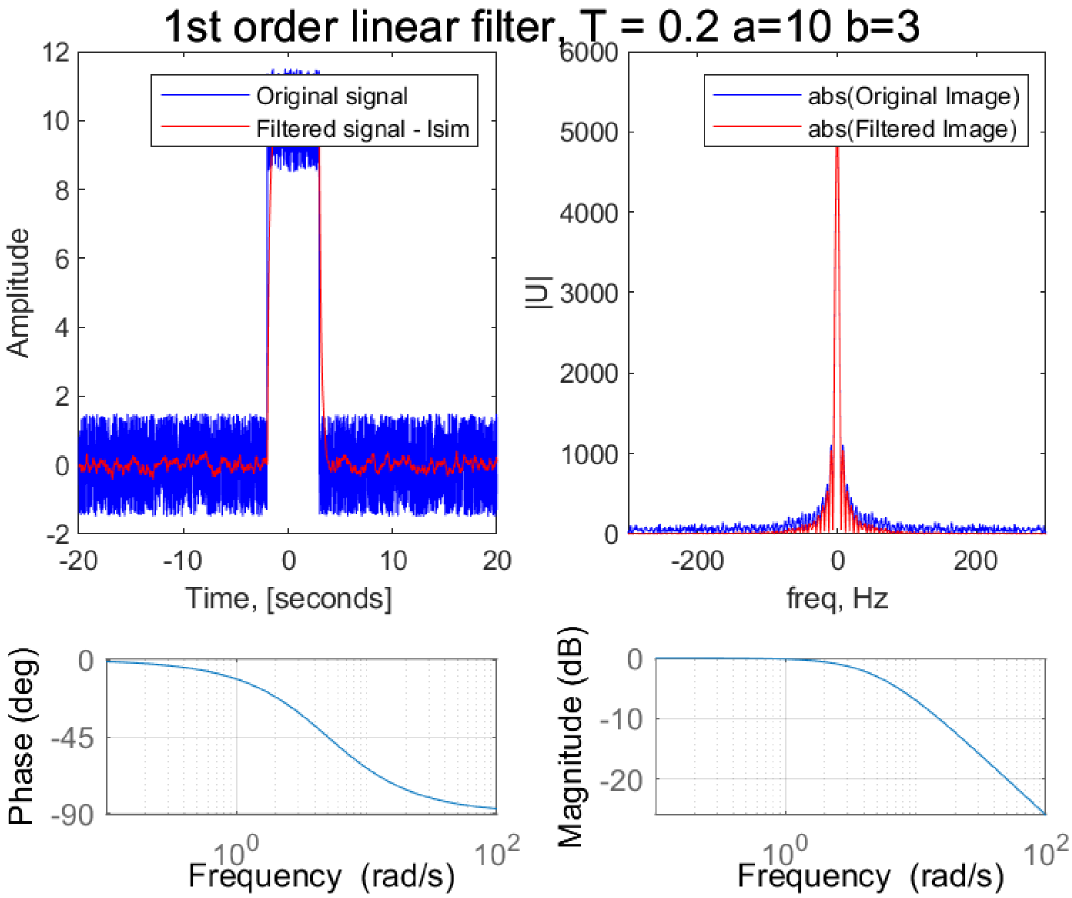

Рисунок 14 — Испытание 9

3.2 Специальный фильтр

Выберем только $b = 0$, остальные будут как-то заданы. Теперь мы уже имеем дело с двумя компонентами шума - случайным и гармоническим:

```
u = g + b*(rand(size(t))-0.5) + c*sin(d*t);
```

3.2.1 Испытания

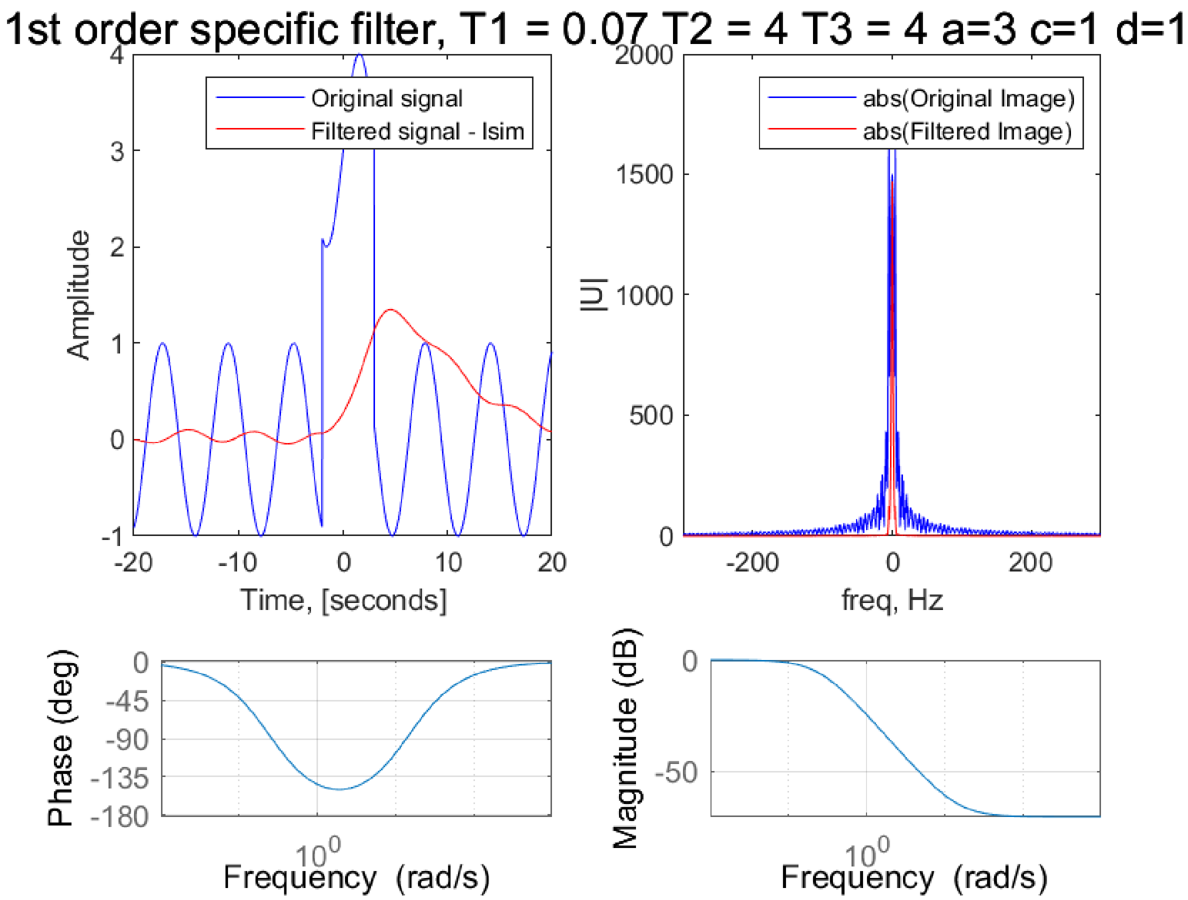

Рисунок 15 — Испытание 1

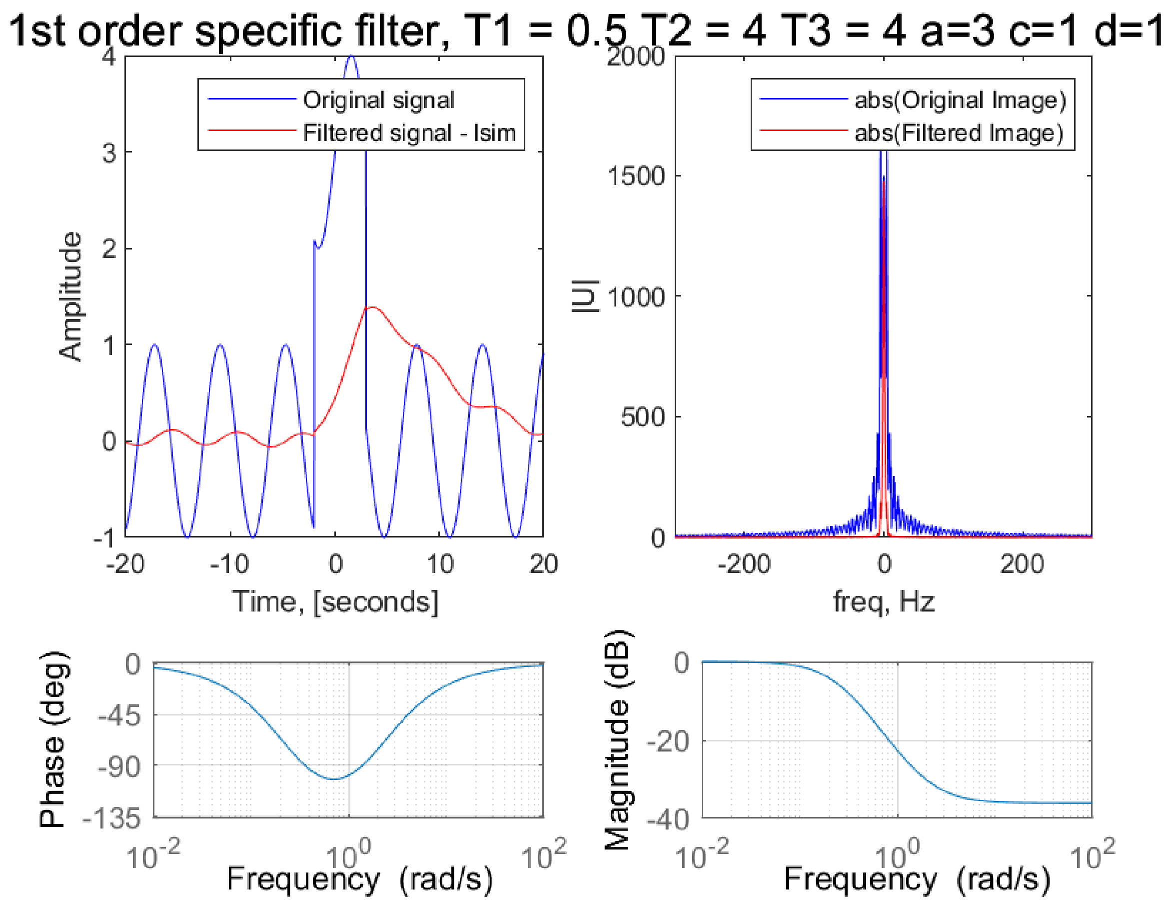

Рисунок 16 — Испытание 2

3.2.2 Выводы

Чисто эмпирическим путём удалось выяснить, что похоже, равенство $T_2 = T_3$ - даёт очень хорошие результаты. Также при этом T_1 должен быть меньше двух остальных коэффциентов, и не сллишком равен им... Поэтому все испытания проводились примерно с таким соотношением.

При большом c мы получаем гармонический шум с большой амплитудой, коэффициенты фильтрации для которого подбираются на глаз куда сложнее, нежели для амплитуд небольших. То же самое было и с параметром d, но дело не в этом. При небольшом c результат фильтрации выходит самым гладким и точным, а при увеличении потери от оригинала как будто значительно увеличиваются.

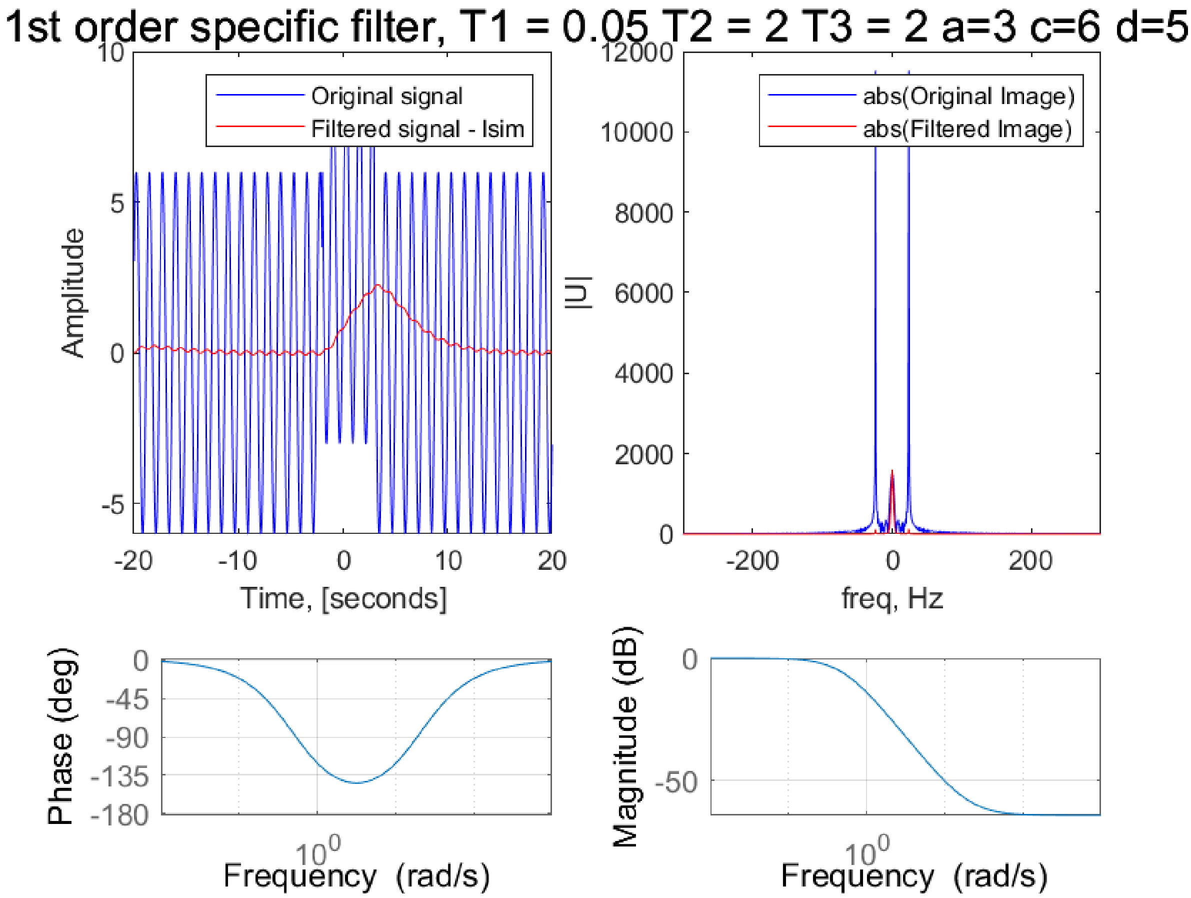

Рисунок 17 — Испытание 3

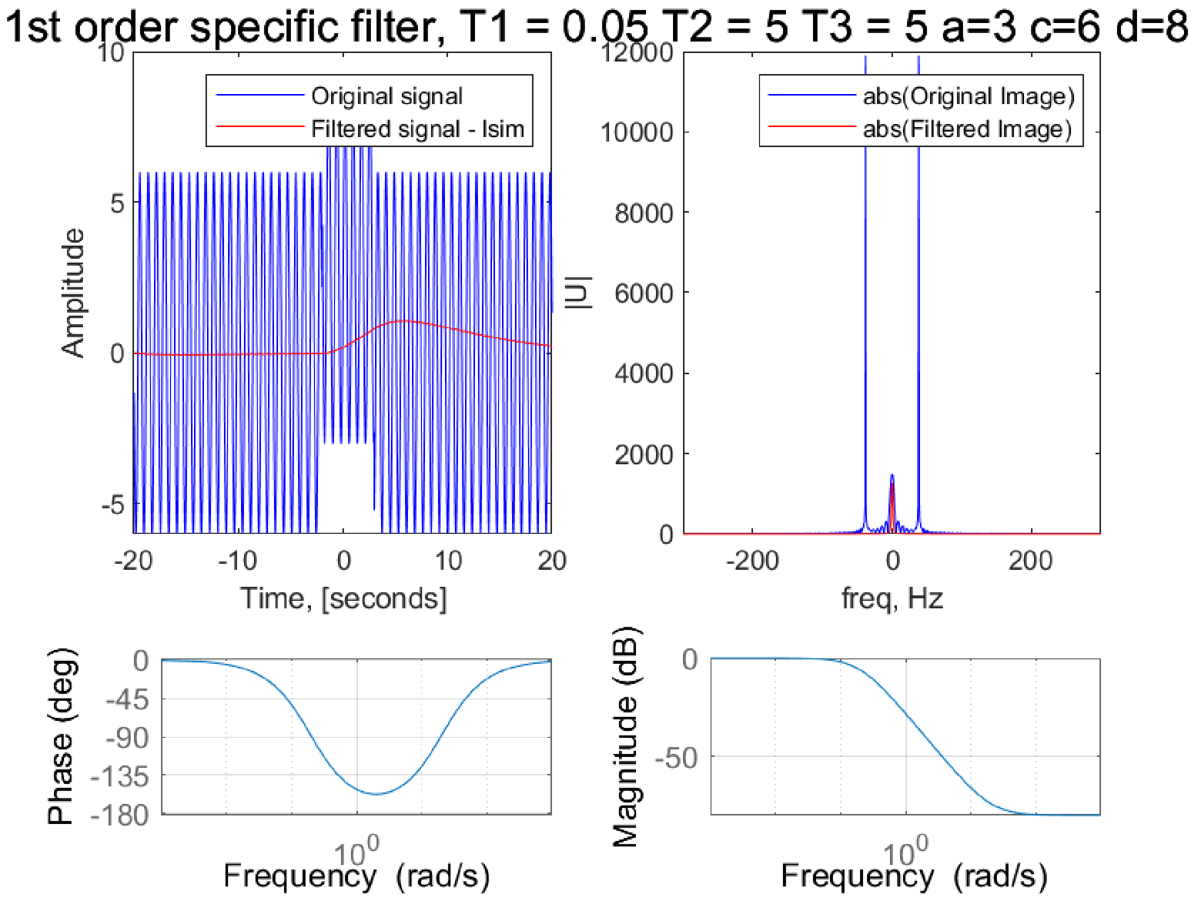

Рисунок 18 — Испытание 4

4 ЗАДАНИЕ 2. ГЛАДИМ БИРЖЕВЫЕ ДАННЫЕ

Выбираем какую-нибудь котировку ценной бумаги, у меня матлаб ругался на акции РосНефти по неизвестной мне причине, поэтому пришлось брать базовую базу, а именно всеми любимые зелёные бумажки (Сбербанк). На этом сайте задали временной промежуток в 4 года и скачали `.csv` файл, с которым будем работать далее.

4.1 Сравнительные графики исходного и фильтрованного сигналов

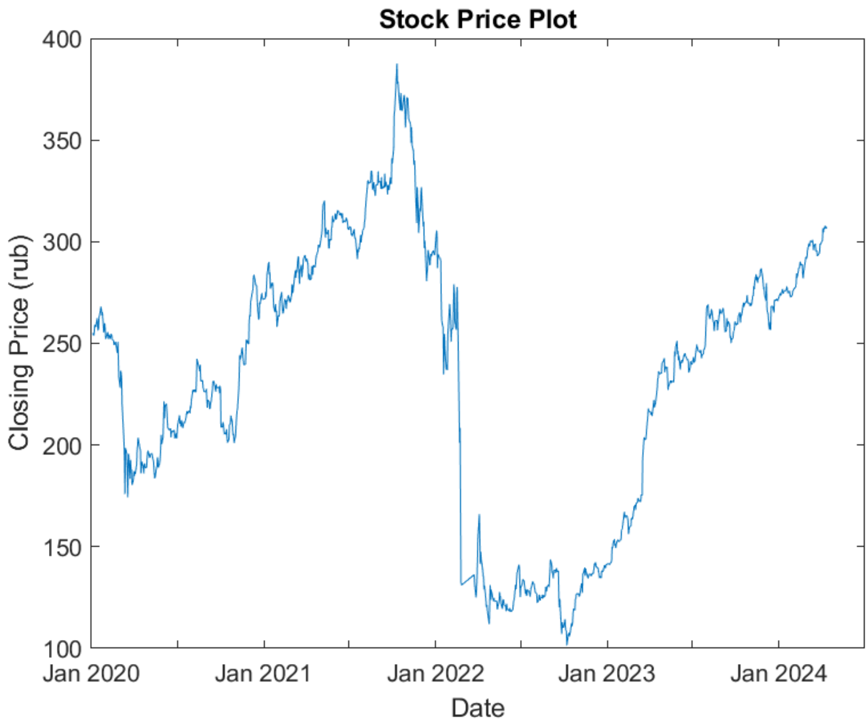

Рисунок 19 — Изначальный график котировки

Получается, что каждый из подграфиков ниже нужно читать следующим образом: сначала мы сглаживаем ”днями то есть минимальный единичный отрезок при фильтрации на оси икс - это день, потом неделя, месяц, год. Так как данные у меня взяты за 4 года, то при апроксимации по годам итоговый график совсем плывёт и почти ничего не показывает, потому что для него слишком мало данных - всего 4 точки(4 года).

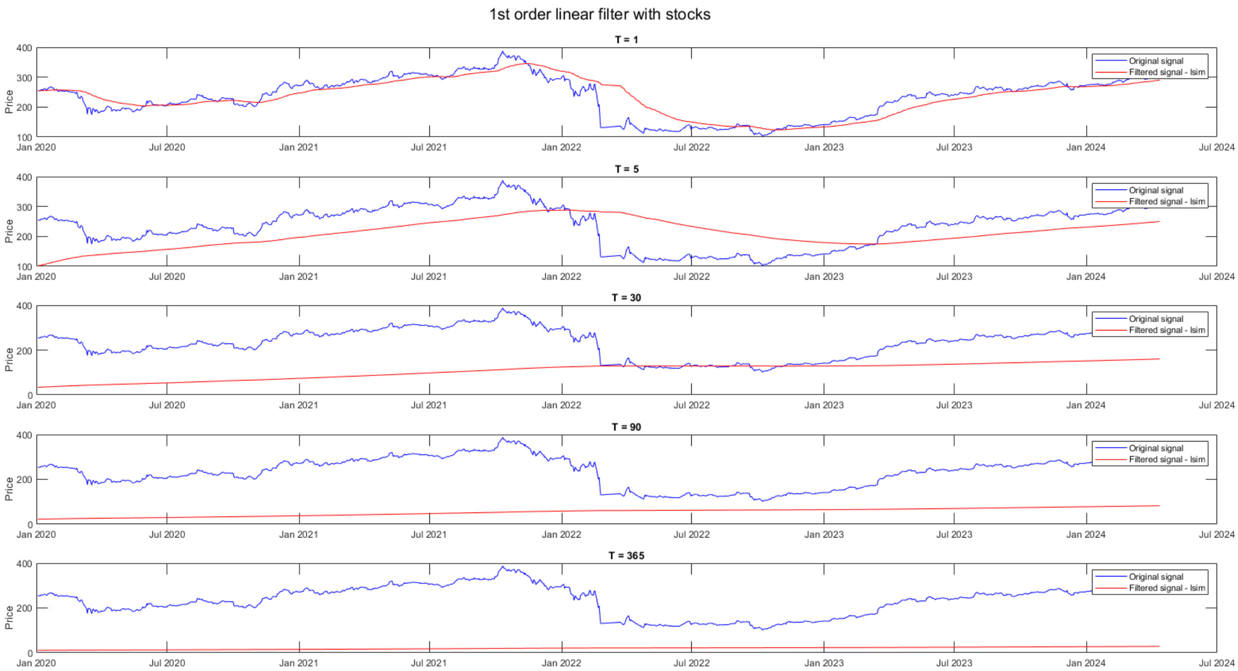

Рисунок 20 — Сглаживания по всем T