

Санкт-Петербургский государственный политехнический
университет Петра Великого

**Высшая школа интеллектуальных систем и
суперкомпьютерных технологий**

Отчёт по лабораторной работе №10
Дисциплина: Телекоммуникационные технологии
Тема: Линейные стационарные системы

Выполнил студент гр. 3530901/80201

В.А. Пучкина

Преподаватель:

Н.В. Богач

Санкт-Петербург
2021

Содержание

1	Упражнение 10.1	5
1.1	Выстрел	5
1.2	Скрипка	6
2	Упражнение 10.2	9
3	Выводы	14

Список иллюстраций

1	Полученный сигнал выстрела.	5
2	Спектр сигнала выстрела.	6
3	Полученный сигнал скрипки.	7
4	Спектр сигнала скрипки.	7
5	Преобразованный сигнал скрипки.	8
6	Сигнал импульсной характеристики.	9
7	Передаточная функция.	10
8	Передаточная функция в логарифмическом масштабе.	10
9	Сигнал колокольчиков.	11
10	ДПФ записи колокольчиков.	12
11	Смоделированный сигнал.	12

Листинги

1	Чтение файла и дополнение сигнала выстрела нулями.	5
2	Получение спектра сигнала.	6
3	Чтение и преобразование сигнала скрипки.	6
4	Получение спектра сигнала скрипки.	6
5	Умножение ДПФ сигнала на передаточную функцию.	7
6	Чтение записи и формирование сигнала.	9
7	Получение передаточной функции.	10
8	Преобразование передаточной функции в логарифмический масштаб.	10
9	Чтение записи колокольчиков.	11
10	Получение ДПФ записи колокольчиков.	11
11	Умножение ДПФ записи на фильтр.	11
12	Использование свертки.	13

1 Упражнение 10.1

В разделе "Акустическая характеристика" умножение ДПФ сигнала на передаточную функцию соответствует круговой свёртке, но в предположении периодичности сигнала. В результате которой можно заметить, что на выходе, в начале фрагмента, слышна лишняя нота, "затекшая" из конца этого фрагмента. Решить эту проблему можно, если перед вычислением ДПФ добавить достаточно нулей в конец сигнала. Тогда эффекта "заворота" можно избежать.

В этом упражнении необходимо изменить пример `chap10.ipynb` и убедиться, что дополнение нулями устраняет лишнюю ноту в начале фрагмента. Будем действовать следующим образом: сократим сигнал до 2^{16} элементов, а после чего дополним его нулями до 2^{17} .

1.1 Выстрел

Начнём с сигнала выстрела.

```
1 shot = read_wave('resources/Sounds/task1_kleeb_gunshot.wav')
2 shot.make_audio()
3 shot = shot.segment(start=0.12)
4 shot.shift(-0.12)
5 shot.truncate(2**16)
6 shot.zero_pad(2**17)
7 shot.normalize()
8 shot.plot()
9 decorate(xlabel='Time (s)')
```

Листинг 1: Чтение файла и дополнение сигнала выстрела нулями.

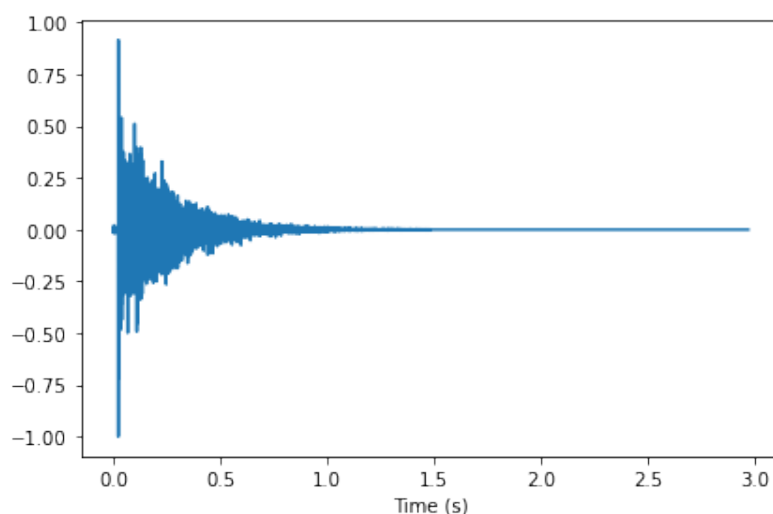


Рис. 1: Полученный сигнал выстрела.

Теперь посмотрим на его спектр.

```
1 transfer = shot.make_spectrum()  
2 transfer.plot()  
3 decorate(xlabel='Frequency (Hz)', ylabel='Amplitude')
```

Листинг 2: Получение спектра сигнала.

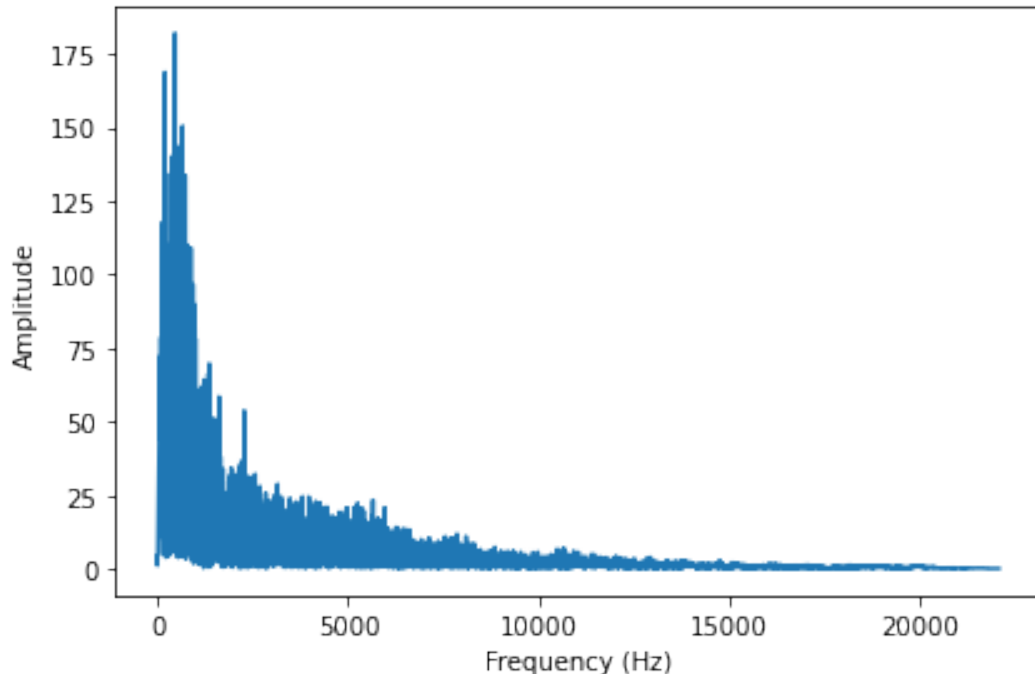


Рис. 2: Спектр сигнала выстрела.

1.2 Скрипка

Теперь возьмём сигнал скрипки.

```
1 violin = read_wave('resources/Sounds/task1_jcveliz_violin_origional.wav')  
2 violin.make_audio()  
3  
4 violin = violin.segment(start=0.11)  
5 violin.shift(-0.11)  
6 violin.truncate(2**16)  
7 violin.zero_pad(2**17)  
8 violin.normalize()  
9 violin.plot()  
10 decorate(xlabel='Time (s)')
```

Листинг 3: Чтение и преобразование сигнала скрипки.

Посмотрим на спектр полученного сигнала.

```
1 violin_spectrum = violin.make_spectrum()  
2 violin_spectrum.plot()  
3 decorate(xlabel='Frequency (Hz)', ylabel='Amplitude')
```

Листинг 4: Получение спектра сигнала скрипки.

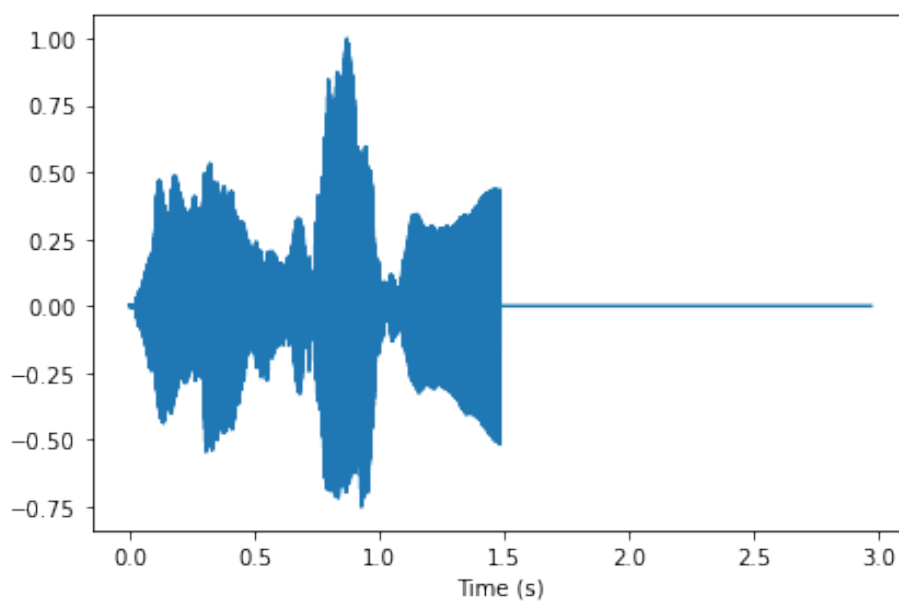


Рис. 3: Полученный сигнал скрипки.

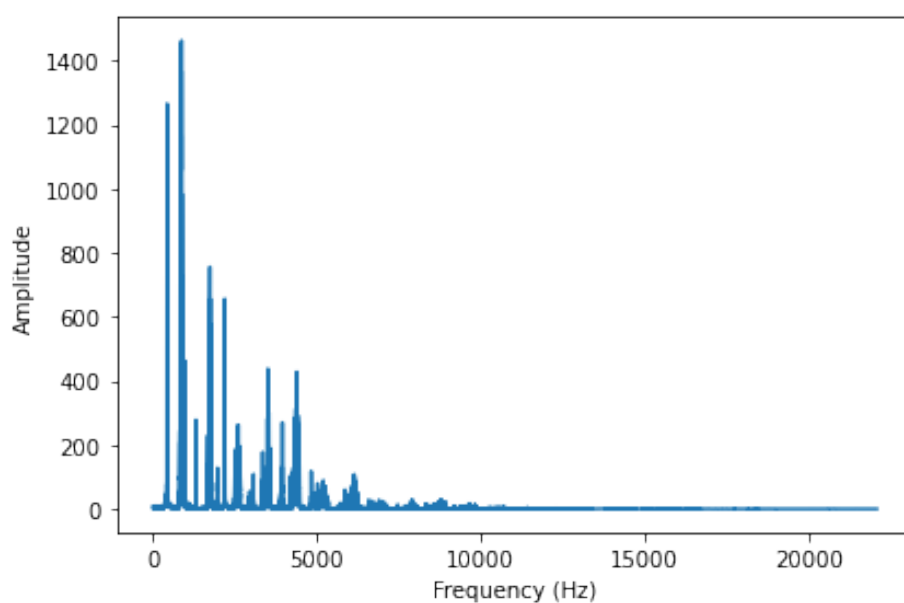


Рис. 4: Спектр сигнала скрипки.

Теперь умножим ДПФ сигнала на передаточную функцию и преобразуем полученный результат обратно в сигнал.

```

1 output = (violin_spectrum * transfer).make_wave()
2 output.normalize()
3 output.plot()
4 output.make_audio()

```

Листинг 5: Умножение ДПФ сигнала на передаточную функцию.

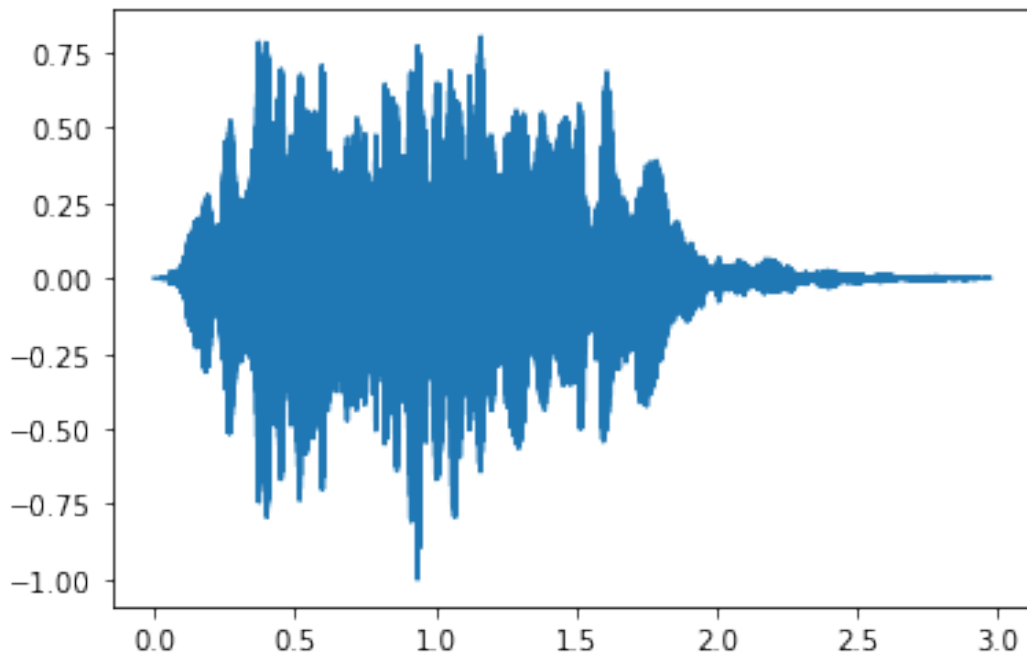


Рис. 5: Преобразованный сигнал скрипки.

При прослушивании полученного сигнала можно заметить, что лишней ”затёкшей” ноты в начале нет.

В данном упражнении был изучен способ устранения лишней ноты в начале фрагмента, возникающей в результате использования круговой свёртке. Это можно сделать, добавив достаточное количество нулей в конец сигнала, перед вычислением ДПФ.

2 Упражнение 10.2

В данном упражнении необходимо просмотреть коллекцию импульсных характеристик на онлайн-ресурсе [Open AIR](#) и скачать одну из них. Затем необходимо найти и скачать короткие записи с той же частотой дискретизации, что и у найденной импульсной характеристики. После этого необходимо смоделировать двумя способами звучание записи в том пространстве, где была измерена импульсная характеристика, как сверткой самой записи с импульсной характеристикой, так и умножением ДПФ записи на вычисленный фильтр, соответствующий импульсной характеристике.

Итак, была выбрана [эта](#) импульсная характеристика и [эта](#) запись звона колокольчиков.

Теперь считаем импульсную характеристику. Это запись первой баптистской церкви в Нэшвилле.

```
1 response = read_wave('resources/Sounds/task2_1st_baptist_nashville_balcony.wav')
2 response = response.segment(duration=3)
3
4 response.normalize()
5 response.plot()
6 decorate(xlabel='Time (s)')
7 response.make_audio()
```

Листинг 6: Чтение записи и формирование сигнала.

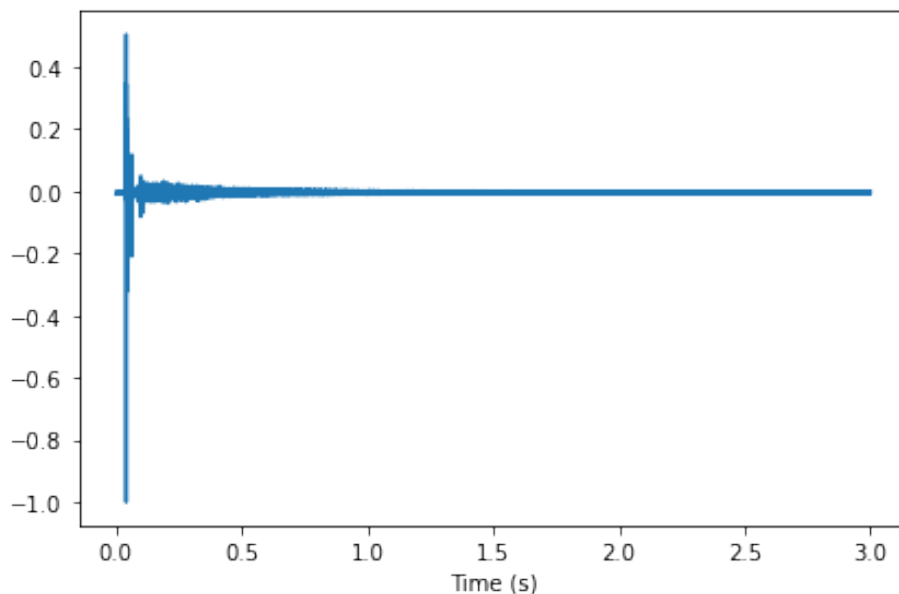


Рис. 6: Сигнал импульсной характеристики.

Теперь получим ДПФ импульсной характеристики - передаточную функцию.

```

1 transfer = response.make_spectrum()
2 transfer.plot()
3 decorate(xlabel='Frequency (Hz)', ylabel='Amplitude')

```

Листинг 7: Получение передаточной функции.

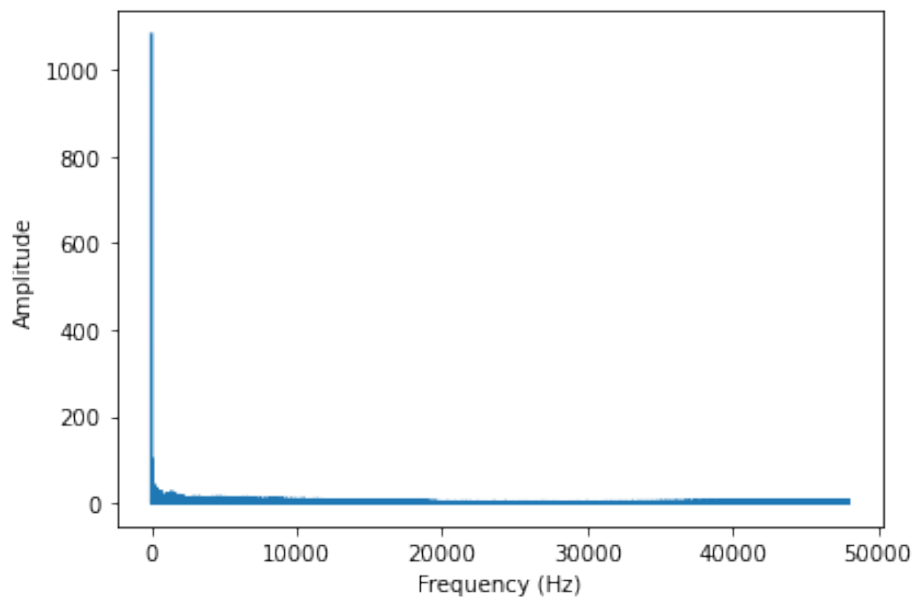


Рис. 7: Передаточная функция.

Посмотрим на неё в логарифмическом масштабе.

```

1 transfer.plot()
2 decorate(xlabel='Frequency (Hz)', ylabel='Amplitude', xscale='log', yscale='log')

```

Листинг 8: Преобразование передаточной функции в логарифмический масштаб.

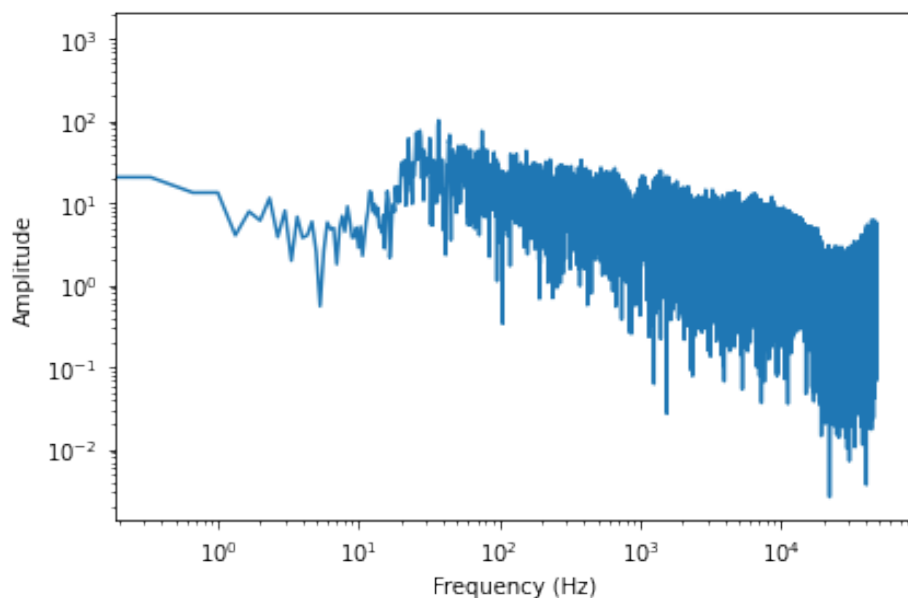


Рис. 8: Передаточная функция в логарифмическом масштабе.

Теперь считаем запись колокольчиков.

```
1 bell_wave = read_wave('resources/Sounds/task2_stomachache_cowbell.wav')
2 bell_wave = bell_wave.segment(start=0)
3
4 bell_wave.truncate(len(response))
5 bell_wave.normalize()
6 bell_wave.plot()
7 decorate(xlabel='Time (s)')
8 bell_wave.make_audio()
```

Листинг 9: Чтение записи колокольчиков.

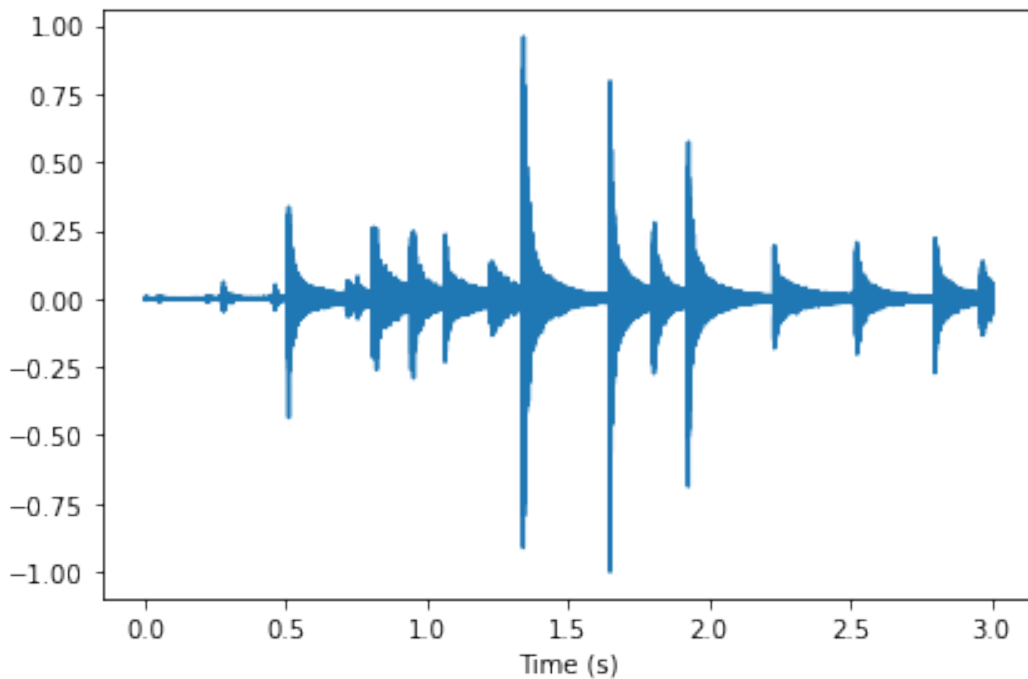


Рис. 9: Сигнал колокольчиков.

Вычислим ДПФ записи колокольчиков.

```
1 bell_spectrum = bell_wave.make_spectrum()
2 bell_spectrum.plot()
3 decorate(xlabel='Frequency (Hz)', ylabel='Amplitude')
```

Листинг 10: Получение ДПФ записи колокольчиков.

Теперь смоделируем звучание колокольчиков в пространстве, умножив ДПФ нашей записи на вычисленный фильтр импульсной характеристики.

```
1 output = (bell_spectrum * transfer).make_wave()
2 output.normalize()
3 output.plot()
4 output.make_audio()
```

Листинг 11: Умножение ДПФ записи на фильтр.

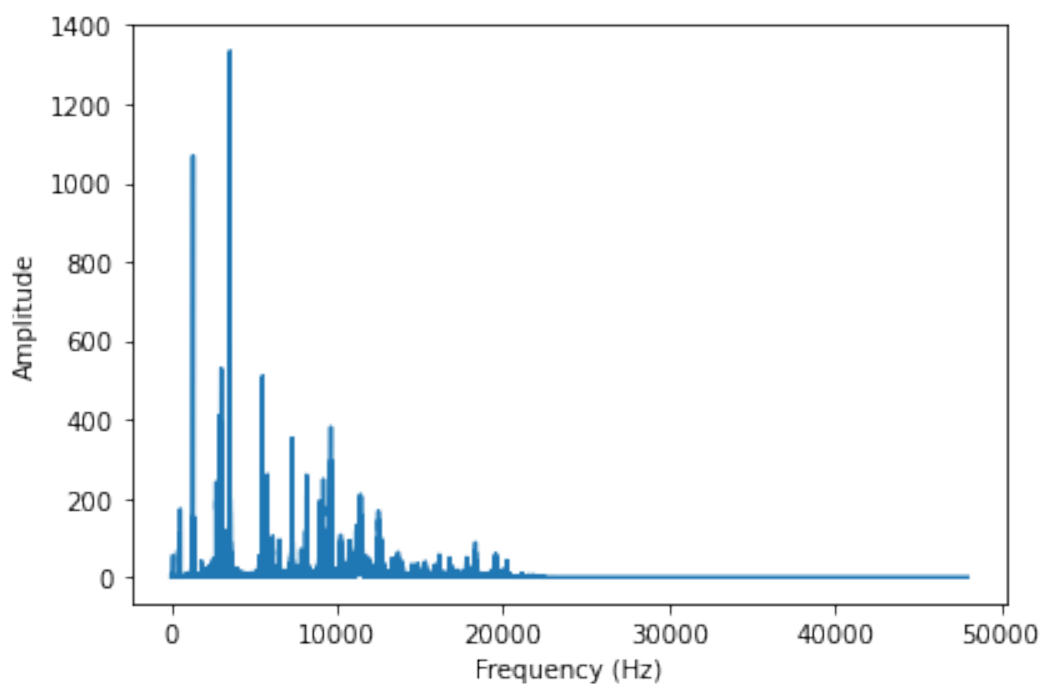


Рис. 10: ДПФ записи колокольчиков.

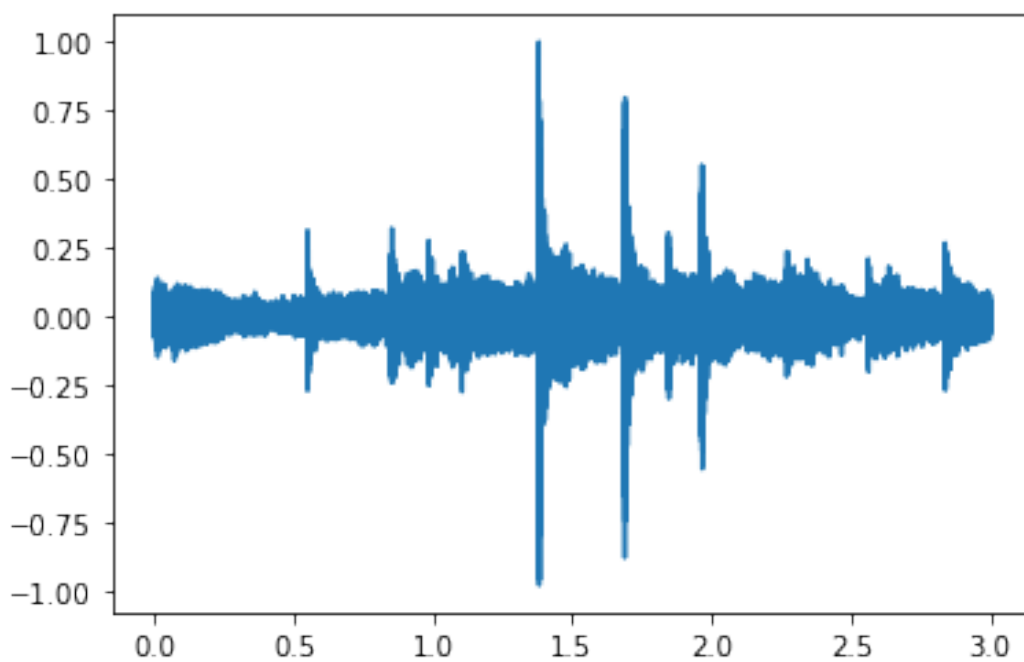


Рис. 11: Смоделированный сигнал.

Прослушав полученный сигнал, можно заметить, что звук действительно как будто переместился в большое пространство, появились эхо и небольшой гул на фоне.

Теперь используем свертку.

```
1 convolved2 = bell_wave.convolve(response)
2 convolved2.normalize()
3 convolved2.make_audio()
4
```

Листинг 12: Использование свертки.

Сигнал, полученный таким образом, звучит так же, как и предыдущий.

В данном упражнении было смоделировано звучание записи колокольчиков в выбранном пространстве, где была измерена импульсная характеристика, двумя способами: умножением ДПФ на вычисленный фильтр и сверткой. При сравнении полученных записей был сделан вывод, что полученные записи звучат одинаково.

3 Выводы

В ходе выполнения данной лабораторной работы был изучен вариант решения проблемы возникновения ”затекшей” ноты в начале фрагмента при использовании свертки. Был сделан вывод, что дополнение нулями устраняет эту лишнюю ноту.

Кроме того, было смоделировано звучание записи колокольчиков в выбранном пространстве двумя способами: умножением ДПФ на вычисленный фильтр и сверткой. Был сделан вывод, что записи, полученные этими способами, звучат одинаково.