

МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования

**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»**

Кафедра дискретной математики и информационных технологий

**АНАЛИЗ ЗВУКОВЫХ ВОЛН С ИСПОЛЬЗОВАНИЕМ РЯДОВ ФУРЬЕ И
PYTHON**

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студента 4 курса 421 группы
направления 09.03.01 — дискретной математики и информационных
технологий
факультета КНиИТ
Космынина Константина Алексеевича

Научный руководитель
профессор, д. ф.-м. н.

В. А. Молчанов

Заведующий кафедрой
доцент, к. ф.-м. н.

Л. Б. Тяпаев

Саратов 2025

ВВЕДЕНИЕ

Актуальность темы обусловлена широким применением анализа звуковых сигналов в науке и технике. Звуковые волны являются одним из важнейших видов сигналов, и их исследование необходимо для понимания музыкальных тонов, речи, а также для разработки современных технологий обработки аудио (сжатие, фильтрация, синтез звука и т.п.). Ряды Фурье представляют мощный математический аппарат для анализа периодических сигналов, позволяя разложить сложную звуковую волну на простые гармонические составляющие. Такой подход лежит в основе спектрального анализа звука, который широко применяется в акустике и цифровой обработке сигналов. Сочетание методов гармонического анализа с возможностями языка Python (и его библиотек) делает задачу исследования звуковых волн доступной и наглядной. Использование языка Python позволяет не только проводить математические расчёты, но и сразу воспроизводить звуковые сигналы, что обеспечивает наглядную проверку теоретических выводов на практике.

Цель работы — провести анализ звуковых волн с использованием разложения в ряд Фурье и методов программной обработки сигналов на языке Python. Для достижения цели необходимо решить следующие **задачи**:

- Изучить теоретические основы представления звуковых сигналов и рядов Фурье: понятие звуковой волны, гармоник, спектра, математический аппарат рядов Фурье.
- Проанализировать современные подходы к анализу звука и существующие решения, представленные в литературе и источниках (как отечественных, так и зарубежных).
- Разработать методологию экспериментального анализа звуковых волн с помощью Python: выбрать инструменты для генерации и воспроизведения звука, для вычисления коэффициентов Фурье.
- Реализовать программные примеры: генерация простейших звуковых сигналов, их воспроизведение; объединение нескольких сигналов и исследование результирующей волны; разложение сложной звуковой волны на гармоники (вычисление ряда Фурье) с идентификацией составляющих частот.

Объект исследования — звуковые волны как физическое и математическое явление, рассматриваемые в непрерывной и дискретной форме. **Предмет**

исследования — методы анализа звуковых волн на основе разложения в ряд Фурье и алгоритмы их компьютерной реализации с использованием Python.

Структура работы. Работа состоит из введения, трёх глав основной части, заключения, списка из 20 использованных источников и приложения с листингом программного кода. Во **введении** обоснована актуальность темы, сформулированы цель, задачи, объект и предмет исследования, описана структура работы. **Глава 1** представляет теоретический обзор: даются определения основных понятий (звук, звуковая волна, частота, гармоника), излагаются математические основы рядов Фурье, рассматриваются направления исследований и литературные источники по анализу звука (включая историческую справку и современные разработки), проводится их критический анализ. **Глава 2** посвящена методологии и практическому анализу: описаны методы исследования (математического и программного моделирования звуковых волн), процесс сбора и обработки данных (генерация тестовых сигналов, дискретизация); представлены программные реализации — определение и воспроизведение звуковых волн средствами Python, преобразование математического описания синусоидальной волны в слышимый звук, объединение (суммирование) нескольких волн для получения сложного звука, разложение звуковой волны в ряд Фурье для выявления её частотных составляющих. В этой главе приводятся результаты вычислительных экспериментов и примеры кода. **Глава 3** описывает разработку интерактивного программного комплекса, предназначенного для визуализации спектральных характеристик звуковых сигналов и интерактивного взаимодействия пользователя с результатами анализа; обоснован выбор архитектуры, рассмотрена структура приложения, приведены примеры пользовательских сценариев работы с комплексом. В **заключении** подведены итоги, сформулированы выводы о возможностях и ограничениях методов анализа звука с помощью рядов Фурье, а также намечены перспективы дальнейших исследований (в частности, применение быстрого преобразования Фурье, анализ реальных звуковых сигналов, улучшение качества синтеза звука и др.).

КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

Глава 1. Теоретический обзор представляет теоретические основы анализа звуковых волн с использованием математического аппарата рядов Фурье.

Раздел 1.1. Основные понятия звуковых волн и гармонического анализа вводит фундаментальные определения. Звуковая волна представляет собой механические колебания давления в среде, распространяющиеся в пространстве. Математически звук описывается функцией $x(t)$, характеризующей изменение давления во времени. Простейшая форма — гармоническое колебание:

$$x(t) = A \sin(2\pi ft + \varphi),$$

где A — амплитуда (громкость), f — частота в герцах (высота тона), φ — начальная фаза.

Сложные звуки представляются суперпозицией гармонических компонент. Для периодической функции с периодом T справедливо разложение в ряд Фурье:

$$x(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} \left(a_n \cos \frac{2\pi n}{T} t + b_n \sin \frac{2\pi n}{T} t \right),$$

где коэффициенты a_n и b_n определяются через интегралы. Синусоидальные составляющие с частотами n/T называются гармониками, их совокупность образует спектр сигнала.

Исторически теория восходит к работам Ж. Фурье (1822), Г. Гельмгольца (1863) и Г.С. Ома (1843). Гельмгольц экспериментально доказал, что тембр определяется набором и интенсивностью гармоник.

Раздел 1.2. Ряд Фурье и преобразование Фурье в обработке сигналов углубляет математический аппарат и рассматривает переход к цифровым методам анализа.

Для анализа непериодических сигналов используется интегральное преобразование Фурье — обобщение рядов на случай бесконечного периода. В цифровой обработке применяется дискретное преобразование Фурье (ДПФ), определяемое для N отсчётов сигнала как:

$$X[k] = \sum_{n=0}^{N-1} x[n] \cdot e^{-j2\pi kn/N}, \quad k = 0, 1, \dots, N-1,$$

где $X[k]$ — комплексные спектральные коэффициенты, $j = \sqrt{-1}$.

Согласно теореме Найквиста-Шеннона, для точного представления аналогового сигнала в цифровой форме частота дискретизации F_s должна превышать удвоенную максимальную частоту f_{max} в сигнале:

$$F_s > 2f_{max}.$$

Для качественного воспроизведения звука в полном слышимом диапазоне до 20 кГц используется стандартная частота дискретизации 44,1 кГц (точнее, 44100 Гц), принятая для компакт-дисков. При этой частоте шаг между отсчётами составляет $\Delta t = 1/44100 \approx 22,68$ мкс.

Революционным достижением стал алгоритм быстрого преобразования Фурье (БПФ, англ. FFT), предложенный Джеймсом Кули и Джоном Тьюки в 1965 году. Алгоритм снижает вычислительную сложность с $O(N^2)$ для прямого вычисления до $O(N \log N)$ за счёт рекурсивного разбиения задачи.

Интересно отметить, что БПФ находит применение не только в обработке сигналов, но и в вычислительной математике для эффективного умножения многочленов. При представлении многочленов степени n через их значения в $2n$ точках комплексной плоскости, БПФ позволяет выполнить умножение за $O(n \log n)$ операций вместо $O(n^2)$ при классическом подходе. Это имеет важное значение в компьютерной алгебре, криптографии и теории кодирования.

Глава 2. Методология и анализ звуковых волн представляет практическую реализацию теоретических методов с использованием языка программирования Python и специализированных библиотек.

Для исследования выбран Python версии со следующим набором библиотек:

1. NumPy — основа научных вычислений, предоставляющая эффективные операции с многомерными массивами.
2. Matplotlib — построение статических и интерактивных визуализаций.
3. PyGame — мультимедийная библиотека, модуль `pygame.sndarray` позволяет воспроизводить звук непосредственно из массивов NumPy.
4. SciPy — дополнительные функции для обработки сигналов.

Параметры цифрового представления звука стандартизированы:

1. Частота дискретизации: $F_s = 44100$ Гц;
2. Разрядность квантования: 16 бит со знаком;

3. Диапазон значений: [-32768, +32767];
4. Формат: моно (одноканальный).

Раздел 2.1. Воспроизведение звуковых волн в Python демонстрирует процесс перехода от математического описания волны к реальному звуку.

Подраздел 2.1.1. Воспроизведение первого звука начинается с простейшего случая — генерации белого шума. Инициализация звуковой системы и создание шума выполняется следующим кодом:

```
import pygame, pygame.sndarray
pygame.mixer.init(frequency=44100, size=-16, channels=1)
import numpy as np

sample_rate = 44100
noise =
np.random.randint(-32768, 32767, size=sample_rate, dtype=np.int16)
noise_sound = pygame.sndarray.make_sound(noise)
noise_sound.play()
```

Белый шум характеризуется равномерным распределением энергии по всему частотному спектру. При воспроизведении он звучит как шипение, аналогичное фону ненастроенного радиоприёмника. Спектральная плотность мощности белого шума постоянна: $S(f) = \text{const}$.

Подраздел 2.1.2. Воспроизведение музыкальной ноты реализует генерацию периодических сигналов. Прямоугольная волна формируется чередованием двух уровней амплитуды, что даёт характерный «электронный» тембр с богатым набором гармоник.

Раздел 2.2. Преобразование синусоидальной волны в звук детально исследует синусоидальные сигналы как фундаментальные компоненты любого звука.

Подраздел 2.2.1. Дискретизация и воспроизведение синусоиды реализует генерацию чистого тона:

```
import numpy as np
import math

f = 440 # частота синусоиды, Гц
```

```

A = 10000 # амплитуда (в пределах 32767)
N = 44100 # число отсчётов на 1 секунду
n = np.arange(N)
sine_wave = A * np.sin(2 * math.pi * f * n / 44100)
sine_wave = sine_wave.astype(np.int16)

tone = pygame.sndarray.make_sound(sine_wave)
tone.play()

```

Результатом является чистый тон частотой 440 Гц (нота Ля первой октавы), служащий стандартом для настройки музыкальных инструментов. Спектр такого сигнала содержит единственную линию на частоте 440 Гц.

Раздел 2.3. Объединение звуковых волн исследует принцип суперпозиции — фундаментальное свойство линейных систем.

Подраздел 2.3.1. Сложение выборок звуковых волн показывает, что поэлементное суммирование массивов точно соответствует физическому наложению звуковых волн в пространстве. Это свойство линейности позволяет анализировать сложные звуки через их компоненты.

Подраздел 2.3.2. Изображение графика суммы двух синусоид визуализирует явление интерференции. При сложении синусоид с частотами ν_1 и ν_2 :

$$h(t) = A_1 \sin(2\pi\nu_1 t) + A_2 \sin(2\pi\nu_2 t)$$

Если частоты близки ($|\nu_1 - \nu_2| \ll \nu_1, \nu_2$), возникают биения — периодические изменения амплитуды с частотой $f_{beat} = |\nu_1 - \nu_2|$. Это явление используется для точной настройки музыкальных инструментов.

Подраздел 2.3.3. Построение линейной комбинации синусоид обобщает принцип на произвольное число компонент:

```

f1 = 440.0 # Hz
f2 = 660.0 # Hz
T = 1.0    # seconds
A1 = 10000
A2 = 10000

```

```

N = int(sample_rate * T)

```

```

n = np.arange(N)
wave1 = A1 * np.sin(2 * math.pi * f1 * n / sample_rate)
wave2 = A2 * np.sin(2 * math.pi * f2 * n / sample_rate)
wave_sum = wave1 + wave2
wave_sum = wave_sum.astype(np.int16)

chord = pygame.sndarray.make_sound(wave_sum)
chord.play()

```

Частоты 440 Гц и 660 Гц образуют музыкальный интервал квинту (отношение частот 3:2), дающий консонантное звучание.

Подраздел 2.3.4. Построение прямоугольной волны через сумму синусоид иллюстрирует ряд Фурье на практике. Прямоугольная волна аппроксимируется суммой нечётных гармоник с амплитудами $\frac{4}{\pi(2n-1)}$.

Раздел 2.4. Разложение звуковых сигналов в ряд Фурье решает обратную задачу — определение спектрального состава заданного сигнала.

Подраздел 2.4.1. Поиск компонент вектора с помощью внутреннего произведения устанавливает аналогию между разложением по базису в конечномерном пространстве и разложением функций. В $\mathbb{R}^3$ для вектора $\mathbf{v} = (v_1, v_2, v_3)$ координаты находятся как:

$$v_i = \mathbf{v} \cdot \mathbf{e}_i$$

Аналогично для функций вводится внутреннее произведение.

Подраздел 2.4.2. Определение внутреннего произведения периодических функций. Для функций с периодом 1 определяется:

$$\langle f, g \rangle = 2 \int_0^1 f(t)g(t)dt$$

Множитель 2 обеспечивает нормировку базисных функций. При численной реализации интеграл аппроксимируется суммой Римана:

$$\langle f, g \rangle \approx 2 \sum_{k=0}^{N-1} f(t_k)g(t_k)\Delta t$$

Подраздел 2.4.3. Определение функции для поиска коэффициентов

Фурье реализует вычисление:

```
def fourier_coefficients(f, N):  
    a0 = inner_product(f, const)  
    an = [inner_product(f, c(n)) for n in range(1, N+1)]  
    bn = [inner_product(f, s(n)) for n in range(1, N+1)]  
    return a0, an, bn
```

, где const , $c(n)$, $s(n)$ — базисные функции.

Подраздел 2.4.4. Поиск коэффициентов Фурье для прямоугольной волны. Экспериментально подтверждено, что разложение прямоугольной волны в ряд Фурье содержит только нечётные гармоники основной частоты. Коэффициенты при чётных гармониках ($n = 2, 4, 6, \dots$) равны нулю. Это согласуется с теоретическим результатом для симметричных относительно начала координат функций.

Подраздел 2.4.5. Коэффициенты Фурье для других волнообразных функций демонстрирует применение разработанного метода к анализу треугольных, пилообразных и произвольных периодических сигналов. Установлена зависимость между гладкостью анализируемой функции и скоростью убывания её коэффициентов Фурье.

Глава 3. Разработка интерактивного программного комплекса представляет современную веб-реализацию инструментов для спектрального анализа звуковых сигналов.

Раздел 3.1. Введение в практическую часть обосновывает выбор веб-платформы. Преимущества включают: кроссплатформенность, отсутствие необходимости установки, простоту распространения и обновления. Веб-приложение делает методы анализа доступными широкой аудитории.

Раздел 3.2. Технологический стек и архитектура приложения представляет используемые технологии:

- TypeScript для типобезопасности;
- React для компонентного интерфейса;
- Next.js для серверно-клиентной архитектуры;
- Plotly.js для интерактивных графиков;
- Vercel для автоматического развёртывания.

Архитектура включает три основных модуля: анализатор произвольных сигналов, анализатор музыкальных нот и тюнер инструментов.

Раздел 3.3. Описание реализации детализирует ключевые компоненты. Модуль БПФ реализует алгоритм Кули-Тьюки с сложностью $O(N \log N)$. Поддерживается реконструкция сигнала по ограниченному числу гармоник для демонстрации сходимости ряда Фурье.

Раздел 3.4. Настройка звукового сигнала на основе преобразования Фурье представляет интерактивный тюнер. Алгоритм анализирует спектр в реальном времени, определяет основную частоту и сопоставляет с ближайшей нотой. Точность определения частоты повышается параболической интерполяцией спектрального максимума.

Раздел 3.5. Развёртывание и эксплуатация описывает процесс публикации на платформе Vercel. Continuous Deployment обеспечивает автоматическое обновление при изменениях в репозитории.

Раздел 3.6. Выводы по практической части подводит итоги разработки. Созданный комплекс успешно реализует поставленные задачи, обеспечивая наглядную визуализацию спектрального анализа. Интерактивность позволяет в реальном времени наблюдать связь между параметрами сигнала и его спектром.

Программный комплекс демонстрирует практическое применение теории рядов Фурье, соединяя математические методы с современными веб-технологиями. Он может использоваться как в образовательных целях для изучения спектрального анализа, так и в качестве основы для специализированных приложений обработки звука.

ЗАКЛЮЧЕНИЕ

Выполненная бакалаврская работа достигла поставленной цели — проведён комплексный анализ звуковых волн на основе разложения в ряды Фурье, реализованы программные эксперименты на языке Python и разработан интерактивный программный комплекс, обеспечивающий наглядную визуализацию спектральных характеристик аудио-сигналов.

В результате исследования получены следующие основные **выводы**:

1. **Разложение звуковой волны в ряд Фурье** является эффективным способом описания её частотного содержания. Любая периодическая звуковая волна может быть представлена в виде суммы синусоидальных колебаний (гармоник). В работе на конкретных примерах показано, что сложные по форме сигналы (сумма тонов, прямоугольная волна) разлагаются на гармонические составляющие, совпадающие с теоретически предсказанными (например, прямоугольная волна содержит только нечётные гармоники с затухающими амплитудами).
2. **Методы вычислительного анализа на Python** подтвердили свою пригодность. С помощью библиотеки NumPy был выполнен расчёт дискретных коэффициентов Фурье, а библиотека PyGame позволила воспроизвести сгенерированные сигналы и убедиться, что математическое описание адекватно реальному звуку (например, синусоида 440 Гц действительно звучит как нота ля). Таким образом, Python может служить удобной платформой для экспериментальной работы со звуком, совмещая анализ и синтез.
3. **Линейность звуковых волн** экспериментально проиллюстрирована. При суммировании двух синусоид образуется сигнал, спектр которого содержит только исходные частоты, а во временной области наблюдаются явления интерференции (биения), объяснимые через суперпозицию гармоник. Это показывает, что принцип суперпозиции выполняется и для акустических сигналов: сложный звук не мешает компонентам сохранять свою индивидуальность в спектре.
4. **Связь временного и частотного описаний** звука: исследования подтвердили, что анализ в частотной области часто проще, чем во временной. Например, определение высоты сложного звука на слух (выделение отдельных нот в аккорде) — нетривиальная задача, тогда как алгоритм Фурье

легко выделяет две чёткие частоты. В то же время некоторые особенности сигнала (временные переходные процессы, разрывы) проявляются как широкополосное распределение по гармоникам (эффект Гиббса для прямоугольной волны). Следовательно, для полного понимания сигнала ценно использовать оба представления.

5. **Разработан интерактивный программный комплекс**, реализующий графический интерфейс для загрузки или генерации звуковых файлов, вычисления их спектров в реальном времени и визуального представления результатов в виде амплитудно-частотных диаграмм. Архитектура комплекса основана на модульном подходе, что упрощает расширение функциональности.
6. **Практические навыки и инструменты:** созданные функции генерации сигналов, построения спектров и разработанный программный комплекс образуют завершённый программный пакет, пригодный для учебных и исследовательских целей.

Научная новизна работы заключается в интеграции классических методов гармонического анализа с современными средствами интерактивной визуализации. Предложенный программный комплекс демонстрирует синергию математических алгоритмов и пользовательских интерфейсов, открывая новые возможности для популяризации спектрального анализа в учебном процессе и самообразовании.

В заключение, проведённое исследование демонстрирует единство математического аппарата Фурье и физической природы акустических колебаний. Интерактивный программный комплекс позволяет «увидеть» звук, мгновенно преобразуя его в спектр и визуализируя скрытые характеристики. Сочетание теории, программных вычислений и интерактивных средств сделало изучение звука наглядным и мотивирует к дальнейшему развитию компетенций на стыке программирования, математики и акустики.