**Лабораторная работа 2**

Тема: "Имитационное моделирование. Моделирование частотной характеристики. Моделирование случайных событий"

Цели работы:

1. изучение методов частотного анализа дискретных сигналов с помощью дискретного преобразования Фурье (MS Excel| Google sheets и python);
2. изучение методов и алгоритмов моделирования случайных чисел (MS Excel | Google sheets и python).

Часть 1. Дискретное преобразование Фурье

Содержание

[Задача 1. Дискретное преобразование Фурье 1](#_Toc96075452)

[Варианты задания 1](#_Toc96075453)

[Примечания 2](#_Toc96075454)

[Теоретические данные 2](#_Toc96075455)

[Пример выполнения работы 5](#_Toc96075456)

# Задача 1. Дискретное преобразование Фурье

1. Согласно варианту задания (табл. 1) средствами MS Excel или Google sheets задать функцию тестового сигнала.
2. Провести процедуру прямого и обратного с использованием алгоритма быстрого преобразования Фурье (БПФ).
3. Вывести результаты в виде графиков:
   1. Функция сигнала (в зависимости от числа отсчетов);
   2. Прямое преобразование Фурье (в зависимости от частоты);
   3. Обратное преобразование Фурье (в зависимости от числа отсчетов).
4. Согласно варианту задания (табл. 1) средствами python 3 и библиотек scipy провести аналогичную процедуру:
   1. Задать входные данные по варианту;
   2. Провести процедуру прямого и обратного БПФ;
   3. Вывести результаты в виде 3 графиков (библиотеки matplotlib или plotly)

# Варианты задания

Таблица 1

Варианты данных для задачи преобразования Фурье

|  |  |  |
| --- | --- | --- |
| Вариант | Тестовый сигнал | Коэффициенты |
| 1, 10, 19 |  |  |
| 2, 11, 20 |  |  |
| 3, 12, 21 |  |  |
| 4, 13, 22 |  |  |
| 5, 14, 23 |  |  |
| 6, 15, 24 |  |  |
| 7, 16, 25 |  |  |
| 8, 17, 26 |  |  |
| 9, 18, 27 |  |  |

К заданию в elearning прикрепляется решение задания:

1. архив с файлами \*.xlsx и папкой с \*.py

или

1. ссылка на открытый Google Sheets документ и программу на диске в файле \*.txt.

# Примечания

1. Предварительно установить надстройку Анализ данных в MS Excel или расширение XLMiner Analysys ToolPak в Google
2. Предварительно установить python версии 3.\* (3.7.5 или 3.8, например) и библиотеки matplotlib (или plotly), scipy, numpy через pip install. Решение может быть сделано в Google Colab, там эти библиотеки есть.

# Теоретические данные

Многие сигналы удобно анализировать, раскладывая их на синусоиды (гармоники). Тому есть несколько причин. Например, подобным образом работает человеческое ухо. Оно раскладывает звук на отдельные колебания различных частот. Кроме того, синусоиды являются «собственными функциями» линейных систем (т.к. они проходят через линейные системы, не изменяя формы, а могут изменять лишь фазу и амплитуду).

Пусть дискретный сигнал имеет период *N* точек. В этом случае его можно представить в виде конечного ряда (т.е. линейной комбинации) дискретных синусоид:

 (ряд Фурье).

Эквивалентная запись (каждый косинус раскладываем на синус и косинус, но теперь – без фазы):

 (ряд Фурье).

Базисные синусоиды имеют кратные частоты. Первый член ряда (*k*=0) – это константа, **называемая постоянной составляющей** (*DC offset*) сигнала. Самая первая синусоида (*k*=1) имеет такую частоту, что ее период совпадает с периодом самого исходного сигнала. Самая высокая составляющая (*k*=*N/2*) имеет такую частоту, что ее период равен двум отсчетам. Коэффициенты  и  называются **спектром** сигнала (*spektrum*). Они показывают амплитуды синусоид, из которых состоит сигнал. Шаг по частоте между двумя соседними синусоидами из разложения Фурье называется **частотным разрешением** спектра.

На рис. 1 показаны синусоиды, по которым происходит разложение дискретного сигнала из 8 точек. Каждая из синусоид состоит из 8 точек, то есть является обычным дискретным сигналом. Непрерывные синусоиды показаны на рисунке для наглядности.

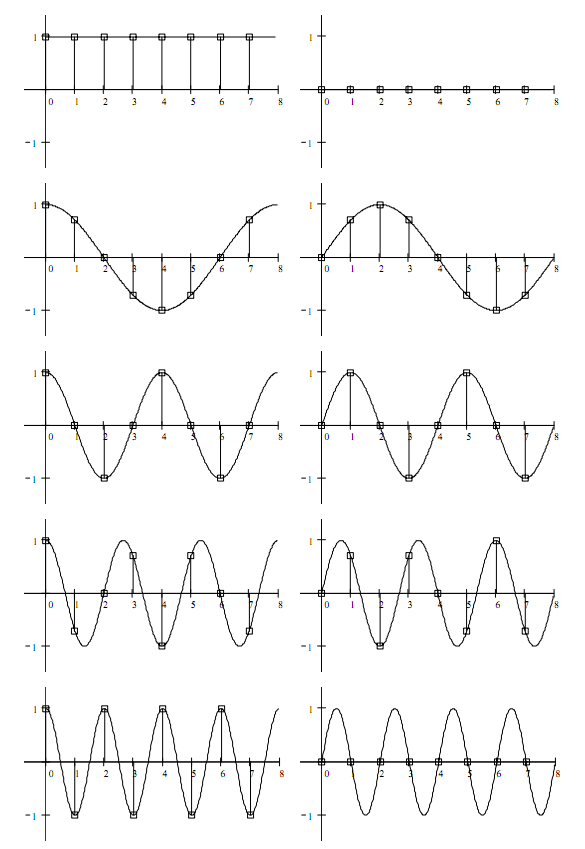


Рис. 1. Базовые функции ряда Фурье для 8-точечного дискретного сигнала.

Слева - косинусы, справа - синусы. Частоты увеличиваются сверху вниз.

Зная коэффициенты  и , можно однозначно восстановить исходный сигнал, вычислив сумму ряда Фурье в каждой точке. Разложение сигнала на синусоиды (т.е. получение коэффициентов) называется **прямым преобразование Фурье.** Обратный процесс – синтез сигнала по синусоидам – называется **обратным преобразованием Фурье**.

Рассмотрим алгоритм прямого преобразования Фурье, т.е нахождения коэффициентов  и 



Подробнее:

1. <http://www.phys.nsu.ru/cherk/Vestnik_Fourier_my_15_09.pdf>

# Пример выполнения работы

Задан тестовый сигнал

,

где



Воспользуемся библиотекой scipy. В этом пакете прямое и обратное преобразование Фурье осуществляется встроенными функциями ():

* **fft(*y*)** – вектор прямого преобразования Фурье;
* **ifft(*v*)** – вектор обратного преобразования Фурье,

здесь ***y*** - вектор действительных данных, взятых через равные промежутки значений аргумента; ***v*** - вектор действительных данных Фурье-спектра, взятых через равные промежутки значений частоты.

Подробнее:

1. <https://docs.scipy.org/doc/scipy/tutorial/fft.html>
2. <http://old.pynsk.ru/posts/2015/Nov/09/matematika-v-python-preobrazovanie-fure/#.Yg9RGdJn1OQ>

На рис. 2 показан фрагмент, на котором проиллюстрирована математическая реализация данного примера.

|  |
| --- |
|  |
| *a* |
|  |
| *б* |
|  |
| *В* |

Рис. 2. Пример кода: a - в Google Colab; b – в Google Sheets; в – настройка пакета в Google Sheets

В результате решения задачи получили графики (рис. 3): исходного сигнала(*а*), прямого преобразования Фурье (*б*), обратного преобразования Фурье (*в*). Обратить внимание, что обратное преобразование вернуло функцию обратно. Также нужно будет отобразить графики на одной Figure с подписями осей.

|  |
| --- |
|  |
| *а* |
|  |
| *б* |
|  |
| *в* |

Рис.5. Пример отображения графиков по-отдельности с помощью пакета plotly: а - Функция сигнала (в зависимости от числа отсчетов); б - Прямое преобразование Фурье (в зависимости от частоты); в - Обратное преобразование Фурье (в зависимости от числа отсчетов).