Міністерство освіти і науки України
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра обчислювальної техніки

Лабораторна робота №2.1

з дисципліни
«Інтелектуальні вбудовані системи»

на тему
«Дослідження параметрів алгоритму дискретного перетворення Фур'є»

Виконав:
студент групи ІП-84
Тимофеєнко Павло Вікторович
номер залікової книжки: 8523

Перевірив:
викладач
Регіда Павло Геннадійович

Київ 2021

Основні теоретичні відомості

В основі спектрального аналізу використовується реалізація так званого дискретного перетворювача Фур'є (ДПФ) з неформальним (не формульним) поданням сигналів, тобто досліджувані сигнали представляються послідовністю відліків $x(k)$

$$F_x(p) = \sum_{k=0}^{N-1} x(k) \cdot e^{-jk\Delta tp\Delta\omega}$$

$$\omega \to \omega_p \to p\Delta\omega \to p \qquad \Delta\omega = \frac{2\pi}{T}$$

На всьому інтервалі подання сигналів **T**, **2π** - один період низьких частот. Щоб підвищити точність треба збільшити інтервал **T.**

$$t \to t_k \to k\Delta t \to k; \quad \Delta t = \frac{T}{N} = \frac{1}{k_{зн.}} \cdot f'гр.$$

ДПФ - проста обчислювальна процедура типу звірки (тобто Σ-е парних множень), яка за складністю також має оцінку $\mathbf{N^2 + N}$. Для реалізації ДПФ необхідно реалізувати поворотні коефіцієнти ДПФ:

$$W_N^{pk} = e^{-jk\Delta t\Delta\omega p}$$

Ці поворотні коефіцієнти записуються в ПЗУ, тобто є константами.

$$W_N^{pk} = e^{-jk\frac{T}{N}p\frac{2\pi}{T}} = e^{-j\frac{2\pi}{N}pk}$$

W_N^{pk} не залежать від **T**, а лише від розмірності перетворення **N.** Ці коефіцієнти подаються не в експоненційній формі, а в тригонометричній.

$$W_N^{pk} = \cos\left(\frac{2\pi}{N}pk\right) - j\sin\left(\frac{2\pi}{N}pk\right)$$

Ці коефіцієнти повторюються (тому і **p** до **N-1**, і **k** до **N-1**, а **(N-1) • (N-1)**) з періодом **N(2π)**.. Т.ч. в ПЗУ треба зберігати N коефіцієнтів дійсних і уявних частин. Якщо винести знак коефіцієнта можна зберігати **N/2** коефіцієнтів.

2π/N- деякий мінімальний кут, на який повертаються ці коефіцієнти. У ПЗУ окремо зберігаються дійсні та уявні частини компілюють коефіцієнтів. Більш загальна форма ДПФ представляється як:

$$F_x(p) = \sum_{k=0}^{N-1} x(k) \cdot W_N^{pk}$$

Завдання на лабораторну роботу

Для згенерованого випадкового сигналу з лабораторної роботи №1 відповідно до заданого варіантом (Додаток 1) побудувати його спектр, використовуючи процедуру дискретного перетворення Фур'є. Розробити відповідну програму і вивести отримані значення і графіки відповідних параметрів.

Варіант-23

Число гармонік в сигналі n	Гранична частота	Кількість дискретних відліків, N
8	1500	1024

Лістинг програми

Index.py

```
import matplotlib.pyplot as plt
import numpy as np
import rsg
import dft

plt.style.use('bmh')

harmonics_count = 8
frequency = 1500
N = 1024

signals = rsg.generateSignal(harmonics_count,frequency,N)
ADFT = np.abs(dft.DFT(signals))

fig, (ax1, ax2) = plt.subplots(nrows=2, ncols=1)
fig.subplots_adjust(hspace=0.5)
fig.set_size_inches(12,6)

ax1.plot(signals)
ax1.set_xlim(0, int(N/4))
ax1.set(xlabel='Time', ylabel='Signal(t)',
        title='Random generated signals')

ax2.plot(ADFT)
ax2.set_xlim(0, int(N/4))
ax2.set(xlabel='Time', ylabel='Freq (Hz)',
```

```
        title='Frequency spectrum')

fig.savefig("plot.png")
plt.show()
```

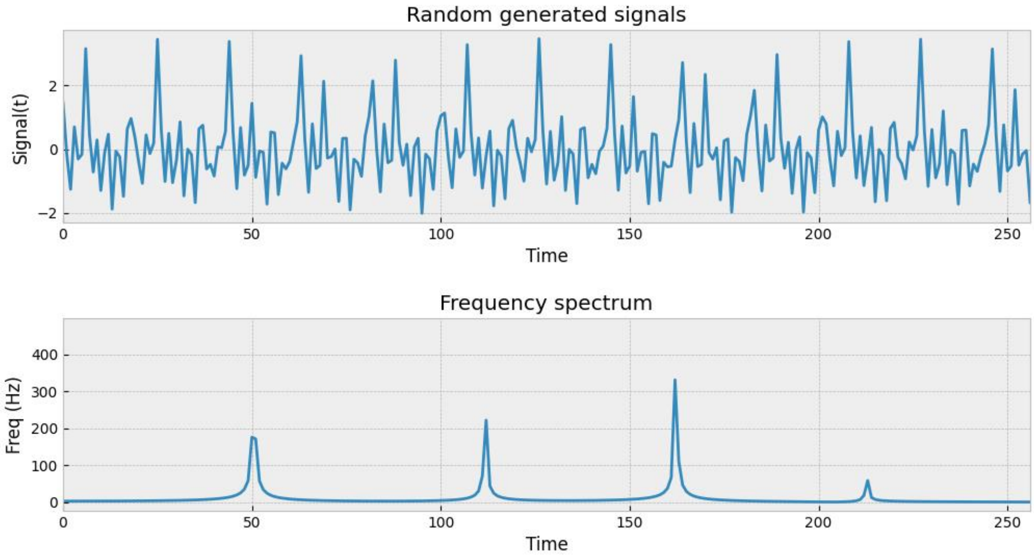
rsg.py

```
import random
import math

def generateSignal(harmonics_count,frequency,N):
    signal = [0] * N

    for i in range (harmonics_count):

        W = (frequency / harmonics_count) * (i+1)
        Amplitude = random.random()
        Phase = random.random()

        for t in range(N):
            signal[t] += (Amplitude * math.sin(W * t + Phase))

    return signal
```

dft.py

```
import math

def DFT(signals):

    L = len(signals)
    DFT = [0] * L

    for p in range(L):
        for k in range(L):
            DFT[p] += signals[k] * (math.cos(2.0 * math.pi * p * k / L) - math.si
n(2.0 * math.pi * p * k / L) * 1j)

    return DFT
```

Результати роботи програми



Висновок

Був ознайомлений з принципами реалізації спектрального аналізу випадкових сигналів на основі алгоритму перетворень Фур’є. Вивчив та дослідив особливості даного алгоритму з використанням засобів моделювання і сучасної програмної оболонки Python.