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 №2.2

з дисципліни

«Інтелектуальні вбудовані системи»

на тему

«ДОСЛІДЖЕННЯ АЛГОРИТМУ ШВИДКОГО ПЕРЕТВОРЕННЯ ФУР'Є З ПРОРІДЖУВАННЯМ ВІДЛІКІВ СИГНАЛІВ У ЧАСІ»

|  |  |  |
| --- | --- | --- |
|  |  |  |

Виконав: Перевірив:

студент групи ІП-83 викладач  
Фібрук Руслан Сергійович Регіда Павло Геннадійович  
номер залікової книжки: 8320

Київ 2021

**Основні теоретичні відомості**

Швидкі алгоритми ПФ отримали назву схеми Кулі-Тьюкі. Всі ці алгоритми використовують регулярність самої процедури ДПФ і те, що будь-який складний коефіцієнт pk WN можна розкласти на прості комплексні коефіцієнти. Для стану таких груп коефіцієнтів процедура ДПФ повинна стати багаторівневою, не порушуючи загальних функціональних зв'язків графа процедури ДПФ. Існують формальні підходи для отримання регулярних графів ДПФ. Всі отримані алгоритми поділяються на 2 класи: 1) На основі реалізації принципу зріджені за часом Хк 2)на основі реалізації принципу зріджені відліків шуканого спектру F(p). Найпростіший принцип зріджені - поділу на парні/непарні пів-послідовності, які потім обробляють паралельно. А потім знаходять алгоритм, як отримати шуканий спектр. Якщо нам вдасться ефективно розділити, а потім алгоритм отримання спектра, то ми можемо перейти від N ДПФ до N/2 ДПФ.

В процесорах цифрової обробки сигналів спеціальне АЛУ орієнтоване на виконання реалізації ШПФ або за часом, або по частоті. Функції, реалізовані базовими операціями БПФ, визначаються базовими алгоритмами. Операнди представлені як комплексні величини. Особливість операції - 2 операнда на вході, на виході так само 2 результату.

**Завдання на лабораторну роботу**

Для згенерованого випадкового сигналу з Лабораторної роботи N 1 відповідно до заданого варіантом (Додаток 1) побудувати його спектр, використовуючи процедуру швидкого перетворення Фур'є з проріджуванням відліків сигналу за часом. Розробити відповідну програму і вивести отримані значення і графіки відповідних параметрів.

Варiант**:** 20

Число гармонік в сигналі: 6

Гранична частота: 1700

Кількість дискретних відліків: 1024

**Лістинг програми**

import numpy as np

import matplotlib.pyplot as plt

HARMONICS\_COUNT = 6

MAX\_FREQUENCY = 1700

DISCRETE\_TIMES\_COUNT = 1024

def rand\_sig(harmonics\_count, max\_freq, discr\_times\_count):

sig = np.zeros(discr\_times\_count)

freq\_start = max\_freq / harmonics\_count

for harmonic\_index in range(harmonics\_count):

amplitude = np.random.uniform(0.0, 1000.0)

phase = np.random.uniform(-np.pi / 2, np.pi / 2)

freq = freq\_start \* (harmonic\_index + 1)

for time in range(discr\_times\_count):

sig[time] += amplitude \* np.sin(freq \* time + phase)

return sig

def fast\_fourier\_transform(sig):

if len(sig) <= 1: return sig

res = np.zeros(len(sig))

x\_even = fast\_fourier\_transform(sig[::2])

x\_odd = fast\_fourier\_transform(sig[1::2])

for p in range(int(len(sig) / 2)):

angle = 2 \* np.pi \* p / len(sig)

turn\_coef = complex(np.cos(angle), -np.sin(angle))

res[p] = x\_even[p] + turn\_coef \* x\_odd[p]

res[p + int(len(sig) / 2)] = x\_even[p] - turn\_coef \* x\_odd[p]

return res

sig = rand\_sig(HARMONICS\_COUNT, MAX\_FREQUENCY, DISCRETE\_TIMES\_COUNT)

FFT = fast\_fourier\_transform(sig)

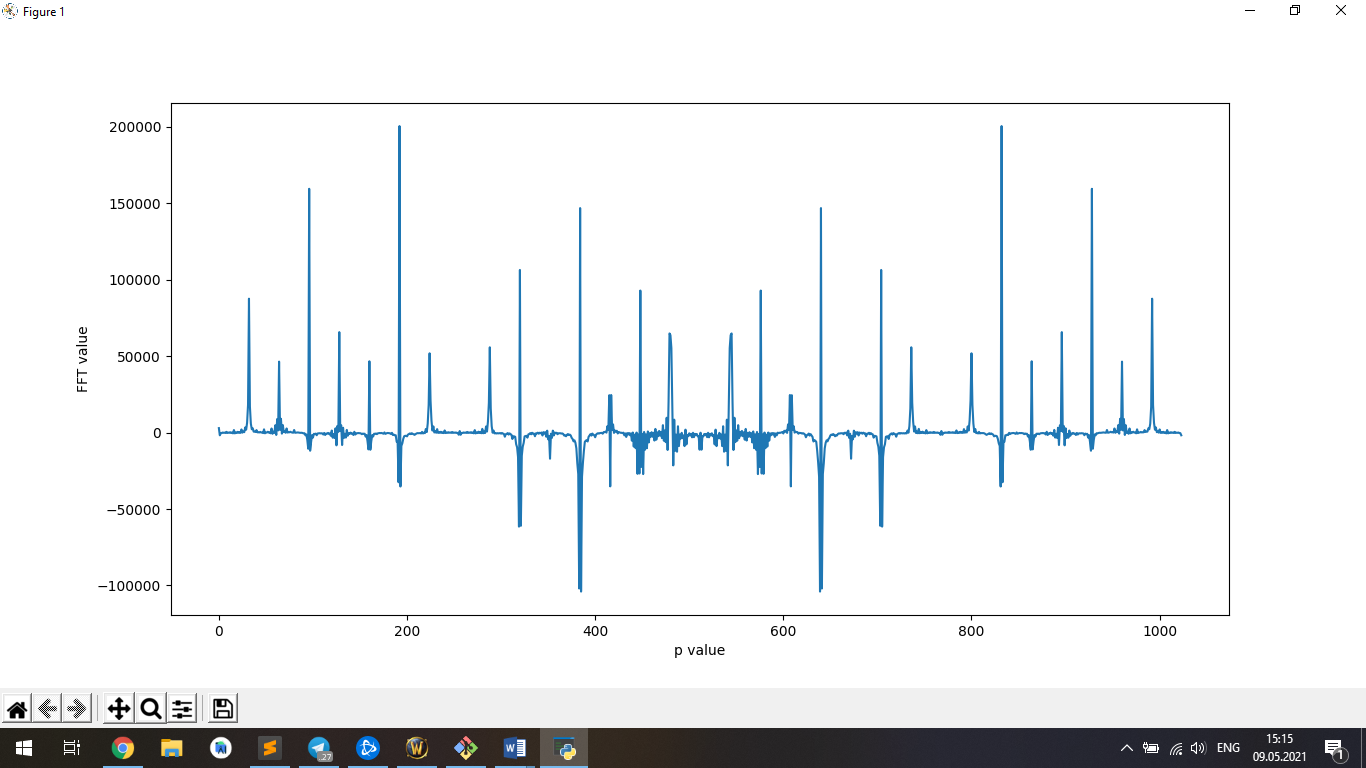
plt.plot(range(DISCRETE\_TIMES\_COUNT), FFT)

plt.xlabel("p value")

plt.ylabel("FFT value")

plt.show()

**Результати роботи програми**



**Висновки**

При виконанні цієї лабораторної роботи ми реалізували алгоритм швидкого перетворення Фур'є та отримали його результат для згенерованого випадкового сигналу.