Національний технічний університет України

«Київський політехнічний інститут»

Факультет інформатики і обчислювальної техніки

Кафедра обчислювальної техніки

**Лабораторна робота №2.1-2.2**

**з курсу: «Інтелектуальні вбудовані системи»**

*Виконав:*

студент групи ІП-84

Шмалько Б. І.

Залікова книжка №8425

*Перевірив:*

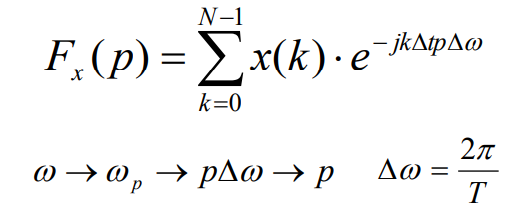
Регіда П. Г.

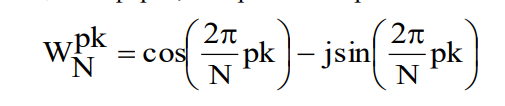
Київ 2020 р.

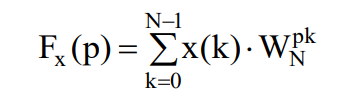
**Основні теоретичні відомості**

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)





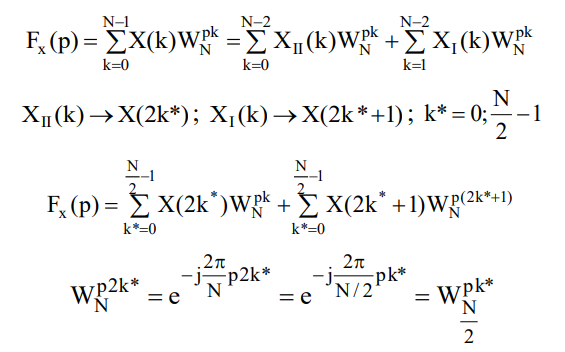


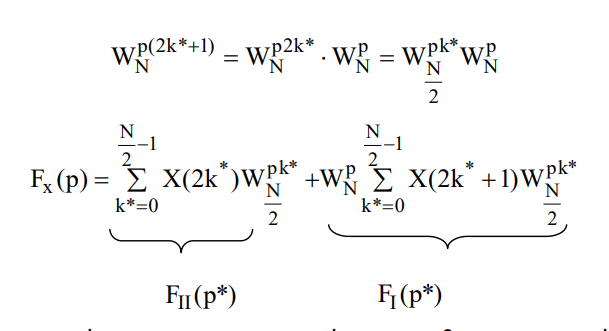
Швидкі алгоритми ПФ отримали назву схеми Кулі-Тьюкі. Всі ці алгоритми

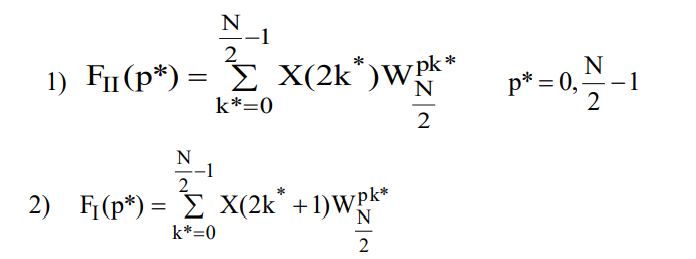
використовують регулярність самої процедури ДПФ і те, що будь-який складний коефіцієнт W. Існують формальні підходи для отримання регулярних графів ДПФ. Всі отримані алгоритми поділяються на 2 класи:

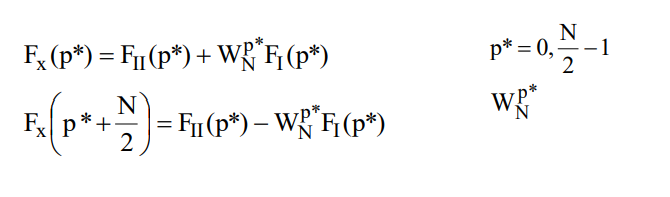
1) На основі реалізації принципу зріджені за часом Хк

2)на основі реалізації принципу зріджені відліків шуканого спектру F(p). можна розкласти на прості комплексні коефіцієнти









**Завдання на лабораторну роботу**

Для згенерованого випадкового сигналу з Лабораторної роботи N 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.

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.

**Варіант-25**

|  |  |  |
| --- | --- | --- |
| Число гармонік в сигналі n | Гранична частота, ωгр | Кількість дискретних відліків, N |
| 12 | 2700 | 64 |

**Лістинг програми**

**const** DiscretFourier = x **=>** {

**const** N = x.length

**const** result = []

**let** Freal, Fimage

    for(**let** p = 0; p < N; p++){

        Freal = 0, Fimage = 0

        for(**let** k = 0; k < N; k++){

            Freal += x[k] \* Math.cos(2\*Math.PI\*p\*k/N)

            Fimage += x[k] \* -Math.sin(2\*Math.PI\*p\*k/N)

        }

        result.push(Math.sqrt(Freal\*\*2 + Fimage\*\*2))

    }

    return result

}

**const** FastFourier = x **=>** {

**const** N = x.length

**const** result = []

**const** cache = [[],[]]

  for (**let** i = 0; i < N/2; i++){

**let** arg = -2\*Math.PI\*i/N

    cache[0][i] = Math.cos(arg)

    cache[1][i] = Math.sin(arg)

    cache[0][i + N/2] = -cache[0][i]

    cache[1][i + N/2] = -cache[1][i]

}

  for(**let** p = 0; p < N/2; p++){

**let** Freal0 = 0, Fimage0 = 0, Freal1 = 0, Fimage1 = 0

    for(**let** k = 0; k < N/2; k++){

          Freal0 += x[2\*k] \* cache[0][p\*2\*k % N]

          Fimage0 += x[2\*k] \* cache[1][p\*2\*k % N]

          Freal1 += x[2\*k+1] \* cache[0][p\*(2\*k+1) % N]

          Fimage1 += x[2\*k+1] \* cache[1][p\*(2\*k+1) % N]

        }

    result[p] = Math.sqrt((Freal0+Freal1)\*\*2 + (Fimage0+Fimage1)\*\*2)

    result[N/2 + p] = Math.sqrt((Freal0-Freal1)\*\*2 + (Fimage0-Fimage1)\*\*2)

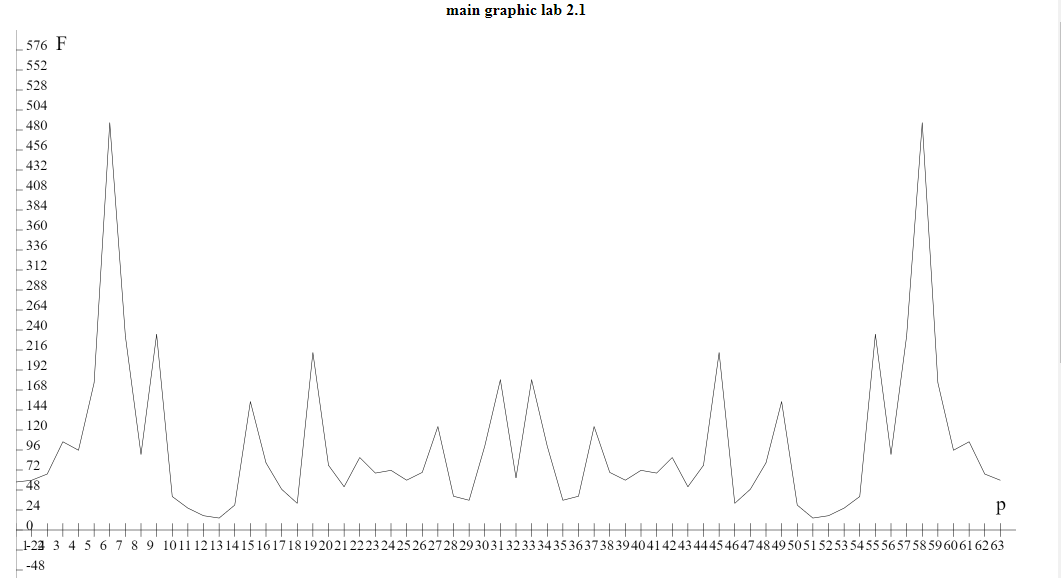
}

return result

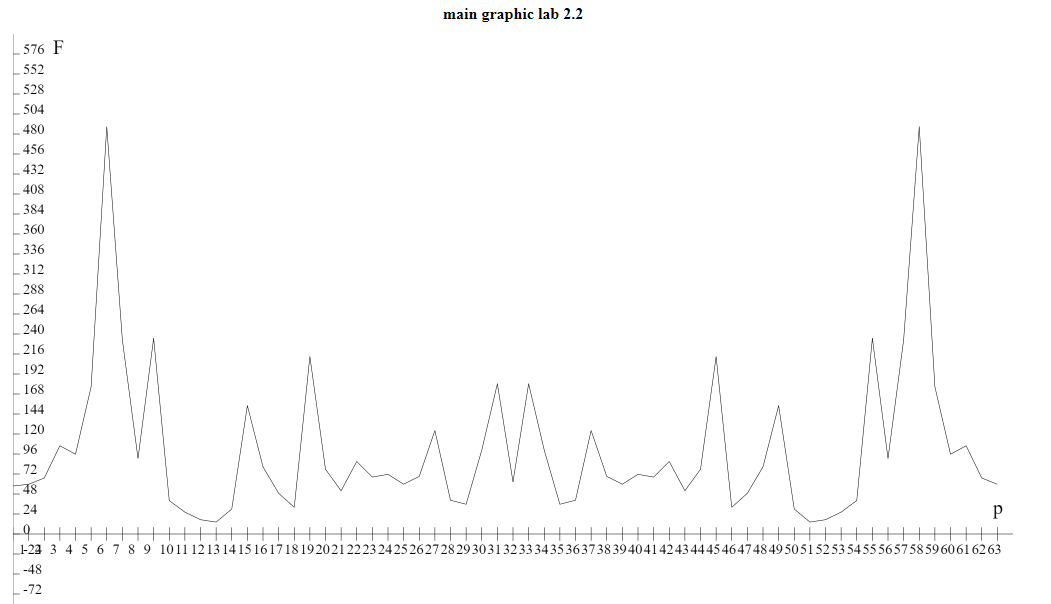
}

**Результати роботи програми**

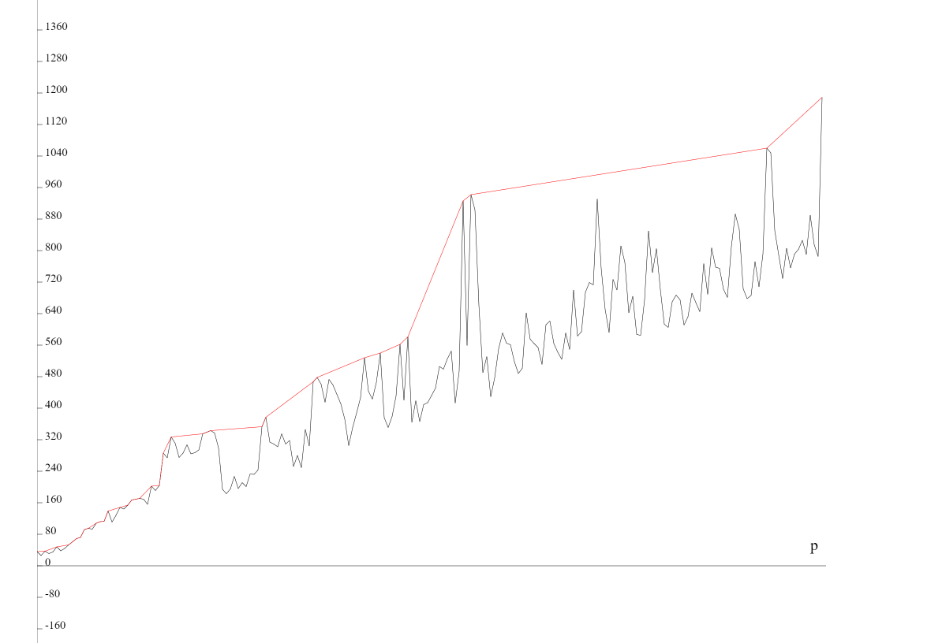
***Графік спектру для дискретного перетворення Фур’є:***

******

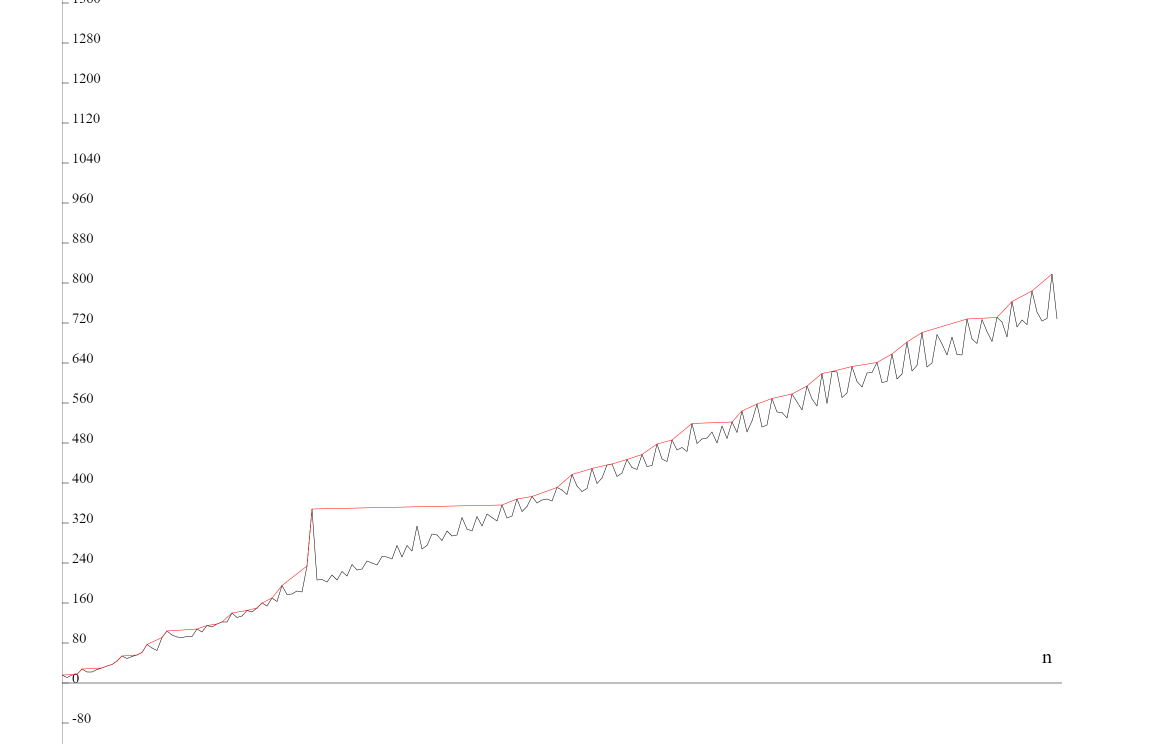
***Графік спектру для швидкого перетворення Фур’є:***

******

***Графік складності алгоритму для дискретного перетворення Фур’є:***

******

***Графік складності алгоритму для швидкого перетворення Фур’є:***

******

**Висновки**

В даній лабораторній роботі ми дослідили принципами реалізації спектрального аналізу випадкових сигналів на основі алгоритму перетворення Фур'є, принципами реалізації прискореного спектрального

аналізу випадкових сигналів на основі алгоритму швидкого перетворення Фур'є. На основі знайдених даних знайшли спектр. Побудували графіки для , дискретного перетворення Фур’є, швидкого перетворення Фур’є, а також складності цих алгоритмів. Дані знання знадобляться нам у нашій подальшій кар’єрі.