

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”

ЛАБОРАТОРНИЙ ПРАКТИКУМ
з дисципліни
ТЕХНОЛОГІЇ ОПРАЦЮВАННЯ СИГНАЛІВ
ТА ЗОБРАЖЕНЬ

Навчальний посібник

*Рекомендувала Науково-методична рада
Національного університету “Львівська політехніка”*

Львів
Видавництво Львівської політехніки
2021

Рецензенти:

Глухов В. С., доктор технічних наук, професор кафедри кафедри електронних обчислювальних машин;

Сало А. М., кандидат технічних наук, доцент, генеральний директор ТзОВ “Кіберсофт технології”;

Шаров Б. Г., кандидат технічних наук, ТзОВ “ЕЛСИ”

*Рекомендувала Науково-методична рада
Національного університету “Львівська політехніка”
як навчальний посібник для студентів галузі знань 12 “Інформаційні технології”
спеціальності 123 “Комп’ютерна інженерія”
(протокол № 59 від 20 жовтня 2021 р.)*

Лашко О. Л.

Л 32 Лабораторний практикум з дисципліни “Технології опрацювання сигналів та зображень” : навч. посібник / О. Л. Лашко. – Львів: Видавництво Львівської політехніки, 2021. – 62 с. – Режим доступу: <http://eom.lp.edu.ua/textbooks/nr-tosz.pdf> вільний. – Заголовок з екрана.

ISBN 978-966-941-691-9

Практикум містить комплекс з шести методичних вказівок до виконання лабораторних робіт з курсу “Технології опрацювання сигналів та зображень”.

УДК 004

ВСТУП

Цей лабораторний практикум виконується в рамках дисципліни “Технології опрацювання сигналів та зображень”, для студентів спеціалізації “Комп’ютерні системи та мережі” на кафедрі електронних обчислювальних машин.

Лабораторні роботи призначені для закріплення теоретичного матеріалу, розглянутого на лекціях та в ході самостійної роботи шляхом набуття навичок розв’язання практичних задач. Метою практикуму є ознайомлення студентів з основними методами та технічною реалізацією цифрової обробки інформації, що включає перетворення аналогових сигналів у дискретні, аналіз дискретних сигналів та цифрових систем, а також цифрову фільтрацію зображень; допомогти засвоїти основні методи та алгоритми опрацювання сигналів, принципи та шляхи проектування апаратних і програмних комп’ютерних засобів опрацювання сигналів, набути початкові практичні навички проектування таких засобів.

У результаті виконання лабораторних робіт студенти повинні знати: типи і особливості алгоритмів опрацювання сигналів, математичні основи їх реалізації, характеристики елементної бази та основи реалізації алгоритмів на її основі, особливості й етапи розробки цифрових пристроїв на процесорах і вузлах, орієнтованих на задачі опрацювання сигналів; мови програмування та мови опису апаратних засобів; пакети моделювання, опрацювання і фільтрації сигналів. Також студенти мають вміти: вибирати найефективніший метод і алгоритм, здійснювати математичне формування алгоритму та методу його розв’язання, розробляти блок-схеми алгоритмів, розробляти програмну реалізацію виконання алгоритмів, розраховувати технічні параметри апаратних і програмних засобів, проектувати структурні та функціональні схеми процесорів та вузлів опрацювання сигналів; моделювати і опрацьовувати сигнали різної форми.

Подані методичні вказівки призначені для допомоги студентам під час виконання індивідуальних завдань та ґрунтовного засвоєння матеріалу.

Лабораторний практикум охоплює основні розділи лекційного курсу і відповідає структурі робочої програми з вказаної дисципліни.

Лабораторна робота № 1

ВИКОНАННЯ ЕЛЕМЕНТАРНИХ ОПЕРАЦІЙ НАД СИГНАЛАМИ В СИСТЕМІ SCILAB

МЕТА РОБОТИ

Виконати операції над матрицями та векторами в системі SCILAB. Здобути навички роботи в цьому середовищі.

ТЕОРЕТИЧНІ ВІДОМОСТІ

Цифрова обробка сигналів – це область науки і техніки, в якій вивчаються загальні для різних технічних застосувань принципи, методи і алгоритми обробки сигналів засобами обчислювальної техніки.

Суть ЦОС як області науки пролягає у розв'язку на обчислювальній машині чотирьох основних задач:

- представлення сигналів в зручній для сприйняття формі;
- виділення із сигналів корисної інформації;
- внесення в сигнали корисної інформації;
- формування сигналів із заданими параметрами.

ЦОС характеризується тим, що маємо справу зі структурованими однотипними даними великого об'єму. Ці дані на математичному рівні представляються векторами, або, в загальному випадку – матрицями. Базовими операціями, що використовуються для розв'язку задач ЦОС, є перемноження елементів векторів, додавання елементів векторів, перетворення координат, обчислення тригонометричних функцій, пошук максимальних значень, операції інвертування матриць, табличне перетворення елементів матриць, сортування, отримання квадратного кореня, піднесення до степеня. Всі ці задачі можуть вирішуватися як в режимі реального часу (РРЧ), так і в нереальному режимі часу.

Для моделювання роботи систем ЦОС та їхніх частин обрано програмне середовище SCILAB, оскільки воно призначене для ефективною роботи з матрицями та має великий набір вбудованих функцій, що реалізують ключові операції обробки сигналів. Зокрема, SCILAB забезпечує одночасне звертання до всіх елементів матриць та векторів з метою виконання однотипних операцій. Це дозволяє здійснювати складні перетворення, не витрачаючи зусиль на дрібні тривіальні дії.

Система SCILAB поєднує розвинену мову програмування та широку бібліотеку чисельних алгоритмів, що охоплює багато областей науки і техніки.

Мова програмування SCILAB є високорівневим інтерпретатором. Це означає, що кожна команда виконується незалежно від інших і не потребує загальної компіляції. Так забезпечується простота написання та відлагодження програм, а також швидкість їх виконання. SCILAB дозволяє визначати користувачеві власні типи даних. При цьому стандартним операціям, наприклад, арифметичним операторам або операторам порівняння, можна надати особливого змісту для роботи з власними типами даних. Користувач може створювати власні модулі для вирішення конкретних задач або викликати функції, реалізовані на інших мовах програмування зокрема на С.

Найпростішим способом використання SCILAB є безпосереднє введення команд у консолі. Результат виконання при цьому відображається відразу ж після її введення і виконання (натисканням клавіші Enter). Таке покрокове виконання є ефективним для того, щоб зрозуміти поведінку готових програм, і дозволяє навчитися писати власні програми.

Послідовність команд, призначена для кількарядового використання повинна бути записана і збережена у файлі.

У пакеті SCILAB файли бувають двох типів: файл-програми – скрипти (мають розширення.sce) та файл-функції (мають розширення.sci), в яких описані функції, визначені самим користувачем.

Скрипт є послідовністю команд SCILAB, він не має вхідних і вихідних аргументів. Скрипт не вимагає жодних ключових слів для початку і кінця. Виконання команд, що містяться в скрипті може здійснюватися: з редактора файлів; з командного рядка, виставивши команду “ехес”; з іншого файлу. Фактично, створений файл стає командою, яку розуміє SCILAB. Всі змінні, оголошені в скрипті стають доступними в робочому середовищі після його виконання. Їх можна використовувати в інших програмах і командах (якщо примусово не очищати пам’ять).

Файл-функція – це файл з розширенням.sci, в якому описана деяка функція. Ім’я файлу повинно співпадати з іменем описаної в ньому функції.

Обов’язковим першим рядком такого файлу є заголовок функції, який містить: ключове слово “function”, список вихідних параметрів, знак “=”, ім’я функції та список вхідних параметрів. Декілька вихідних аргументів зручно використовувати для функцій, що повертають не одне, а кілька значень (так звані вектор-функції). Після заголовку йде тіло функції, в якому відбувається знаходження вихідних параметрів. Закінчуватися функція повинна ключовим словом “endfunction”.

Виклик функції відбувається після її виконання шляхом запуску з командного рядка за допомогою команди “ехес”.

КОНТРОЛЬНІ ЗАПИТАННЯ

1. Що таке SCILAB?
2. Які елементарні операції можна виконувати над сигналами?
3. Які засоби надає SCILAB для виводу графічних даних?
4. Яка різниця між файлами.sce та.sci в SCILAB, як вони використовуються?
5. Як оформити власну функцію в SCILAB?
6. Як вивести на екран результат обчислень?

ПОРЯДОК ВИКОНАННЯ РОБОТИ

1. Запустити систему SCILAB.
2. Ознайомитися з демонстраційною програмою, звернувши увагу на виконання елементарних операцій над матрицями і векторами (для цього слід в консолі SCILAB вибрати пункт меню “Довідка” – “Демонстрації SCILAB”).
3. В командному вікні SCILAB проілюструвати можливі операції над матрицями та векторами.
4. Вивчити можливості системи для роботи з графікою.

5. Написати власну програму (.sce), яка реалізовує завдання, задане варіантом.
6. Написати власну функцію (.sci), яка реалізовує завдання, задане варіантом.
7. Продемонструвати виконання написаних програм в SCILAB.

ЗМІСТ ЗВІТУ

1. Завдання.
2. Результати ознайомлення з демонстраційною програмою (перелік і короткий опис основних функцій).
3. Текст написаної програми та пояснення алгоритму її роботи.
4. Результат роботи програми.
5. Висновки.
6. Список використаної літератури.

ЗАВДАННЯ

| | |
|---|---|
| 1 | Задати цілочисельну матрицю розміром $n \times 2n$. Знайти матрицю, що отримується з цієї, шляхом перестановки рядків – першого з останнім, другого з передостаннім і т. д. |
| 1 | 2 |
| 2 | Задати квадратну матрицю A порядку n . Знайти вектор A^*v , при цьому елементи вектора v визначити за формулою: $b_i = \frac{1}{i^2 + 2}$. |
| 3 | Задати квадратну матрицю A розміром $n \times m$. Знайти вектор A^*v , при цьому елементи вектора v визначити за формулою: $b_i = \frac{3i^3}{13}$. |
| 4 | Задати квадратну матрицю A порядку n . Знайти матрицю A^*B , при цьому елементи матриці B визначити за формулою: $b_{ij} = \begin{cases} \frac{1}{i+j-1}, \text{при } i < j \\ 0, \text{при } i = j \\ -\frac{1}{i+j-1}, \text{при } i > j \end{cases}, (i, j = 1, \dots, n)$. |
| 5 | Задати дійсний вектор a_1, a_2, \dots, a_n та дійсну квадратну матрицю порядку n ($n \geq 6$). Побудувати дійсну матрицю розміром $n \times (n+1)$, вставивши у вихідну матрицю між 5 і 6 стовпцем новий стовпець з елементами a_1, a_2, \dots, a_n . |
| 6 | Задати квадратну матрицю A порядку n . Знайти матрицю A^*B , при цьому елементи матриці B визначити за формулою: $b_{ij} = \begin{cases} \frac{1}{i+j-1}, \text{при } i \leq j \\ \frac{1}{i+j+1}, \text{при } i > j \end{cases}, (i, j = 1, \dots, n)$. |
| 7 | Задати вектор a_1, a_2, \dots, a_{10} , що містить лише цілі числа та цілочисельну квадратну матрицю порядку n . Замінити нулями в матриці ті елементи з парною сумою індексів, для яких є рівні серед a_1, a_2, \dots, a_{10} . |

| 1 | 2 |
|----|--|
| 8 | Задати квадратну матрицю A порядку n . Знайти матрицю $A*B$, при цьому елементи матриці B визначити за формулою: $b_{ij} = \frac{1}{i+j-1}$, $(i, j = 1, \dots, n)$. |
| 9 | Задати матрицю розміру $n \times (n+1)$, елементами якої є дійсні числа $1/n, 2/n, 3/n, \dots, (n \times (n+1))/n$ та довільні дійсні числа a_1, a_2, \dots, a_{n+1} ; b_1, b_2, \dots, b_{n+1} , натуральні числа p, q ($p \leq n, q \leq n+1$). Утворити нову матрицю розміру $(n+1) \times (n+2)$ вставкою після рядка з номером p заданої матриці нового рядка з елементами a_1, a_2, \dots, a_{n+1} та подальшою вставкою після стовпця з номером q нового стовпця з елементами b_1, b_2, \dots, b_{n+1} . (n, p, q – змінні). |
| 10 | Задати дві дійсні квадратні матриці порядку n . Отримати нову матрицю шляхом віднімання від елементів кожного рядка першої матриці добутку елементів відповідного стовпця другої матриці. |
| 11 | Задати дійсну квадратну матрицю порядку n , всі елементи якої різні. Знайти в ній найменший по модулю елемент і утворити квадратну матрицю порядку $n-1$, шляхом викидання з початкової матриці того рядка та стовпця, на перетині яких розташований елемент зі знайденим значенням. |
| 12 | Задати вектор a_1, a_2, \dots, a_{11} , що містить лише різні цілі числа та цілочисельну квадратну матрицю порядку n . Замінити нулями в матриці ті елементи, для яких немає рівних серед a_1, a_2, \dots, a_{11} . |
| 13 | Задати квадратну матрицю A порядку n . Знайти вектор $17*A*v$, при цьому елементи вектора v визначити за формулою: $b_i = \begin{cases} \frac{i}{\sin(i/\pi)}, & \text{для } i - \text{парне} \\ \frac{i+1}{2 * \cos(i/\pi)}, & \text{для } i - \text{непарне} \end{cases} \quad (i, j = 1, \dots, n).$ |
| 14 | Задати вектор a_1, a_2, \dots, a_7 , що містить лише цілі числа та цілочисельну квадратну матрицю порядку n . Замінити нулями в матриці ті елементи з непарною сумою індексів, для яких є рівні серед a_1, a_2, \dots, a_7 . |
| 15 | Задати дійсну квадратну матрицю, порядку n , всі елементи якої різні. Знайти в ній найбільший по модулю елемент і утворити квадратну матрицю порядку $n-1$, шляхом викидання з початкової матриці того рядка та стовпця, на перетині яких розташований елемент зі знайденим значенням. |
| 16 | Задати цілочисельну матрицю розміру $n \times 3$ та цілі числа k, l ($1 \leq k \leq n; 1 \leq l \leq n; k \neq l$). Перетворити матрицю так, щоб рядок з початковим номером k безпосередньо знаходився за рядком з початковим номером l , зберігши порядок решти рядків. |
| 17 | Задати квадратні матриці A і B порядку n . Знайти матрицю $A(B-E)+C$, де E – одинична матриця порядку n , а елементи матриці C знаходяться за формулою $C_{ij} = \frac{1}{i+j}$ ($i, j = 1, \dots, n$). |
| 18 | Задати дві дійсні квадратні матриці порядку n . Отримати нову матрицю шляхом додавання до елементів кожного стовпця першої матриці добутку елементів відповідних рядків другої матриці. |
| 19 | Задати дві дійсні квадратні матриці порядку n . Отримати нову матрицю шляхом множення елементів кожного рядка першої матриці на найбільший з елементів відповідного стовпця другої матриці. |

| 1 | 2 |
|----|--|
| 20 | Задати квадратну матрицю A порядку n . Знайти вектор $A*v$, при цьому елементи вектора v визначити за формулою: $b_i = \begin{cases} \frac{1}{i^2+2}, & \text{якщо } i - \text{парне} \\ \frac{1}{i}, & \text{якщо } i - \text{непарне} \end{cases} (i, j = 1, \dots, n).$ |
| 21 | Задати квадратну матрицю A порядку n . Знайти матрицю $A+B$, при цьому елементи матриці B визначити за формулою: $b_{ij} = \begin{cases} \frac{i+j}{\cos(i*j*\pi/13)}, & \text{при } i \leq j \\ \frac{i+j}{\sin(i*j*\pi/12)}, & \text{при } i > j \end{cases}, (i, j = 1, \dots, n).$ |
| 22 | Задати квадратні матриці A і B порядку n . Знайти матрицю $A(C-E)+B*C$, де E – одинична матриця порядку n , а елементи матриці C знаходяться за формулою $C_{ij} = \frac{29}{\cos(i*j/\pi)} (i, j = 1, \dots, n).$ |
| 23 | Задати дійсний вектор a_1, a_2, \dots, a_n та дійсну квадратну матрицю порядку n ($n \geq 6$). Побудувати дійсну матрицю розміром $(n+1) \times n$, вставивши у вихідну матрицю між 5 і 6 рядком новий рядок з елементами a_1, a_2, \dots, a_n . |
| 24 | Задати дві цілочисельні квадратні матриці порядку n . Отримати нову матрицю шляхом множення елементів кожного стовпця першої матриці на найменший з елементів відповідного рядка другої матриці. |
| 25 | Задати квадратну матрицю A порядку n . Знайти матрицю $A+B$, при цьому елементи матриці B визначити за формулою: $b_{ij} = \frac{\sin((i+j)/\pi)}{i*j}, (i, j = 1, \dots, n).$ |
| 26 | Задати вектор a_1, a_2, \dots, a_{25} , що містить лише різні цілі числа та цілочисельну квадратну матрицю порядку n . Замінити одиницями в матриці ті елементи з парною сумою індексів, для яких немає рівних серед a_1, a_2, \dots, a_{25} . |
| 27 | Задати матрицю розміром $n \times (n+1)$, елементами якої є дійсні числа $1, 2, 3, \dots$ ($n \times (n+1)$). Також задати довільні дійсні числа $a_1, a_2, \dots, a_{n+1}; b_1, b_2, \dots, b_{n+1}$, і натуральні числа p, q ($p \leq n, q \leq n+1$). Утворити нову матрицю розміром $(n+1) \times (n+2)$ вставкою після рядка з номером p заданої матриці нового рядка з елементами a_1, a_2, \dots, a_{n+1} та подальшою вставкою після стовпця з номером q нового стовпця з елементами b_1, b_2, \dots, b_{n+1} , n, p, q – змінні. |
| 28 | Задати вектор a_1, a_2, a_3 , що містить лише різні цілі числа та цілочисельну квадратну матрицю порядку n . Знайти в матриці ті елементи з непарною сумою індексів, для яких є рівні серед a_1, a_2, a_3 і замінити їх на середнє арифметичне значень a_1, a_2, a_3 |
| 29 | Задати цілочисельну матрицю розміром $n \times m$. Знайти матрицю, що отримується з даної шляхом перестановки стовпців – першого з останнім, другого з передостаннім і т. д. |
| 30 | Задати цілочисельну матрицю розміром $5 \times n$ та цілі числа k, l ($1 \leq k \leq n; 1 \leq l \leq n; k \neq l$). Перетворити матрицю так, щоб стовпець з початковим номером k безпосередньо знаходився за стовпцем з початковим номером l , зберігши порядок решти рядків. |

ПРИКЛАД ВИКОНАННЯ

1. Завдання

Задати квадратні матриці A і B порядку n . Знайти матрицю X :

$$X = 13 \cdot A \cdot (C + E) - B \cdot (C + A),$$

де E – одинична матриця порядку n ; елементи матриці C знаходяться за формулою:

$$C_{ij} = \frac{1}{\sin(i + 2 * j / \pi)} \quad (i, j = 1, \dots, n).$$

2. Результати ознайомлення з програмним середовищем

Виходячи з поставленого завдання та можливостей системи Scilab, для вводу даних з клавіатури оберемо функцію “input”, яка дозволить в інтерактивному режимі задати розмірність матриць A , B , C , E .

Далі використаємо можливість діалогу для вибору способу задання елементів матриць A та B . Для цього найдоцільніше скористатися оператором “select”, який забезпечить порівняння введеного символу з еталоном та виконання відповідних дій. Так, якщо на запитання про введення матриць за допомогою випадкового заповнення буде дано ствердну відповідь (символ “Y” або “y”), то відбудеться виклик вбудованої функції “rand”. Якщо ж відповідь буде негативна (символ “n”), то користувач отримає можливість ввести елементи обох матриць з клавіатури за правилами Scilab (в квадратних дужках, розділяючи рядки знаком “;”, а стовпці – “,”). Якщо ж введена відповідь буде інакшою, то виведеться повідомлення про помилку і програма завершиться.

Наступним кроком є утворення одиничної матриці, тобто такої матриці, головна діагональ якої містить одиниці, а решта елементів – нульові. Для цього слід використати функцію “eye” зі змінною n , як параметру розмірності.

Для утворення матриці C треба скористатися заданою формулою обчислення її елементів та оператором циклу “if” “end”. У класичному випадку (наприклад у мові С), варто було б організувати вкладений цикл за номером рядка та номером стовпця (i та j , відповідно). Однак, оскільки Scilab орієнтований на роботу з масивами даних, достатньо утворити цикл лише за одним параметром, а інший задати як вектор з n значень. При цьому при кожному входженні в цикл буде утворюватися не один елемент матриці, а цілий рядок, або стовпець (залежнозалежно від того, за яким параметром організований цикл).

На завершальному етапі треба знайти матрицю X . Для цього досить записати задану формулу її обчислення так, ніби працюємо зі скалярними значеннями, а Scilab забезпечить коректний обрахунок шуканої матриці.

Зауважимо, що спосіб відображення даних на екрані неграфічних даних в системі Scilab – використання функції “disp”. Наприклад, команда: “disp (c)” приведе до виводу значення змінної c у вигляді: <значення c >.

Для того, щоб не відбувалося звертання до заповнених областей пам’яті на початку програми, треба скористатися командою “clear” з параметром “all”, що дозволить очистити всю доступну пам’ять. Для очищення командного вікна використовується команда “clc”. При використанні графічних вікон для їх автоматичного закриття слід звернутися до команди “close”.

Нижче приведено текст програми, що реалізовує завдання. Ця програма повинна міститися у файлі <назва>.sce, який потрібно запускати з середовища Scilab. Виконання та відображення буде відбуватися в командному вікні.

3. Текст програми

| | |
|--|--|
| clc | // очистка екрану |
| clear all | // очистка пам'яті |
| n=input ('n=') | // ввід розміру матриць |
| choice=input ("input matrix randomize? Y/N ", "s"); | // запит на спосіб введення елементів матриць // випадковим чином |
| select choice | |
| case 'y' then | // при ствердній відповіді |
| a=10.*rand (n,n); | // генерація матриць A і B випадковим чином |
| b=10.*rand (n,n); | // в обраному діапазоні |
| case "n" then | // при негативній відповіді |
| a=input ('input matrix A =') | // запит на ввід обох матриць в ручному |
| b=input ('input matrix B =') | // режимі |
| else | // при іншій відповіді |
| error ("Wrong answer") | // повідомлення про помилку |
| end | // завершення оператора "select" |
| disp (a, 'a=') | // відображення значень у командному вікні |
| disp (b, "b=") | |
| e=eye (n,n) | // утворення та вивід одиничної матриці, введеного |
| disp (e, "e=") | // розміру, за допомогою вбудованої функції |
| i=[1:n] *ones (1,n); | // визначення матриці, що містить номер рядка |
| disp (i, "i=") | |
| j=i'; | // визначення матриці, що містить номер стовпця |
| c=1.0./sin (i+2*j/%pi); | // побудова матриці C за заданою формулою і вивід // її на |
| disp (c, "c=") | екран |
| x=13*a* (c+e)-b* (c+a); | // Обрахунок і вивід на екран шуканої матриці X |
| disp (x, "x=") | |

4. Результат виконання приведеної програми

```

>>n=3
n =
    3
>>input matrix randomize? Y/N  y
a =
    6    7    6
    5    8    9
    7   10    2
b =
   13   12    1
    4   10   12
    4    2    3
e =
    1    0    0
    0    1    0

```

```
0 0 1
c =
1.0022 1.3102 4.3542
2.0670 -7.6181 -1.4391
-2.1050 -1.1048 -1.0198
x =
-0.6491 -707.7388 -19.1921
-58.5927 -876.1858 2.2501
339.3545 -830.5394 149.1551
```

Висновки

В ході даної лабораторної роботи відбулося ознайомлення з можливостями програмного середовища SCILAB, освоєні основні програмні конструкції та продемонстровані засоби графічного та неграфічного відображення інформації. Крім того, виконано поставлене індивідуальне завдання, що закріпило набуті навички на практиці.

Лабораторна робота № 2

ДИСКРЕТИЗАЦІЯ І КВАНТУВАННЯ СИГНАЛІВ

МЕТА РОБОТИ

Дослідити процес дискретизації і квантування сигналів, оцінити похибку оцифрування.

ТЕОРЕТИЧНІ ВІДОМОСТІ

Аналоговим сигналом будемо називати функцію $f(t)$ деякого аргументу, яка описує певний фізичний процес, або явище. При цьому: $-\infty \leq t \leq \infty$, $-\infty \leq f(t) \leq \infty$, t – час, швидкість, напруга, віддаль тощо.

Дискретний сигнал – це сигнал, що описується функцією $f(\Delta t_i)$, визначеною тільки в конкретні значення аргументу.

Цифровий сигнал – це дискретизований сигнал, який набуває значень зі скінченної кількості рівнів квантування.

Теорема Котельникова

Якщо сигнал $x(t)$ обмежений смугою $(0, f_{\text{граничне}})$, то він може бути відтворений з як завгодно великою точністю за відліками, що взяті з частотою дискретизації $2f_2$:

$$\Delta t \leq \frac{1}{2f_2}, \quad (1)$$

де $f_2 = \frac{\omega_2}{2\pi}$ – гранична частота.

Вибір надто малого Δt призводить до надлишковості обчислень, вибір надто великого Δt призводить до втрати точності через явище підміни частот. Отже, Δt треба обирати обдуманно і обгрунтовано.

При застосуванні складних методів обробки, частоту дискретизації збільшують принаймні в 4–8 разів. Тобто:

$$f_d = 2f_2 * \text{coef},$$

де $\text{coef} \geq 1$ – ціле число. Переважно, при цифровій обробці, він обирається як степінь числа два.

Крок і рівні квантування знаходяться, виходячи з заданої амплітуди сигналу. Для цього використовується формула:

$$\Delta k = (\max_A - \min_A) / (M - 1), \quad (2)$$

де \max_A і \min_A – максимальне і мінімальне значення амплітуди, відповідно; M – кількість рівнів квантування, яка, як правило, пов'язана з розрядністю обчислювального пристрою.

Оцінка похибки квантування може здійснюватися за такими критеріями:

абсолютна похибка:
$$\Delta = \max_{n=0}^{N-1} |x_u(n) - x_d(n)|, \quad (3)$$

де $x_u(n)$ – значення цифрового сигналу в точці n ; $x_d(n)$ – значення дискретного сигналу в точці n ; N – кількість відліків;

середня похибка:
$$\bar{x} = \frac{1}{N} \left[\sum_{n=0}^{N-1} x_u(n) - \sum_{n=0}^{N-1} x_d(n) \right]; \quad (4)$$

дисперсія:
$$D = \frac{1}{N} \sum_{n=0}^{N-1} \{x_u(n) - x_d(n)\}^2. \quad (5)$$

КОНТРОЛЬНІ ЗАПИТАННЯ

1. Дайте визначення сигналу.
7. Для чого виконується дискретизація і квантування сигналу?
8. Про що говорить теорема Котельникова?
9. Поясніть виникнення похибки при перетворенні сигналу в цифрову форму.
10. Які критерії оцінки похибки ви знаєте?

ПОРЯДОК ВИКОНАННЯ РОБОТИ

1. Знайти в аналітичному виді крок дискретизації заданого варіантом сигналу за формулою (1) та період цього сигналу.
2. За допомогою системи SCILAB задати вхідний сигнал, утворивши дискретну послідовність $x(n\Delta t)$ та вивести його графік на одному періоді. При цьому частоту дискретизації збільшити в 2 рази, тобто $coef=2^1$.
3. Оцифрувати отриману дискретну послідовність, для 16 рівнів квантування (тобто $M=2^4$), попередньо знайшовши максимальну амплітуду сигналу. Вивести графік цифрового сигналу.
8. Змінюючи коефіцієнт дискретизації та кількість рівнів квантування, дослідити похибку оцифрування за формулами (3), (4), (5).
9. Зробити висновок про крок дискретизації та кількість рівнів квантування, які доцільно застосовувати для заданого варіантом сигналу.

ЗМІСТ ЗВІТУ

1. Завдання (аналітичний запис сигналу) згідно з варіантом.
2. Аналітичний розрахунок кроку дискретизації та періоду сигналу.
3. Текст програми на SCILAB.
4. Значення похибок при різних коефіцієнтах збільшення частоти дискретизації та різних кількостях рівнів квантування (у табличному вигляді).
5. Графіки дискретного та цифрового сигналу для вибраних параметрів оцифрування (рисунок).
10. Висновки.
11. Список використаної літератури.

ЗАВДАННЯ

Форма сигналу:

$$x(t) = A_1 \cos(\omega_1 t + \varphi_1) - A_2 \sin(\omega_2 t + \varphi_2) + A_3 \sin(\omega_3 t + \varphi_3) - A_4 \cos(\omega_4 t + \varphi_4).$$

| Варіант № | Параметри сигналу | | | | | | | | | | | |
|-----------|-------------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| | A ₁ | A ₂ | A ₃ | A ₄ | ω ₁ | ω ₂ | ω ₃ | ω ₄ | φ ₁ | φ ₂ | φ ₃ | φ ₄ |
| 1 | -24 | -19 | 22 | 0,32 | 8 | 2 | 1/5 | 12 | π/4 | 0 | π/7 | π/5 |
| 2 | 25 | 18 | 0,73 | -7 | 4 | 12 | 2 | 1/3 | π/5 | π/2 | 0 | π/4 |
| 3 | 16 | 0,45 | -5 | 18 | 14 | 9 | 1/8 | 10 | π/7 | π/5 | π/3 | 0 |
| 4 | 0,48 | 23 | -5 | -17 | 13 | 1/7 | 15 | 5 | π/7 | π/5 | 0 | π/4 |
| 5 | 19 | -0,27 | 1 | 21 | 1/6 | 14 | 15 | 15 | π/7 | 0 | π/3 | π/5 |
| 6 | 30 | -29 | 0,20 | -2 | 7 | 1/5 | 7 | 2 | 0 | π/2 | π/3 | π/7 |
| 7 | 13 | -26 | 26 | 0,36 | 12 | 6 | 1/4 | 2 | π/2 | 0 | π/3 | π/7 |
| 8 | 22 | -13 | -0,62 | 8 | 4 | 15 | 11 | 1/3 | π/2 | π/4 | 0 | π/7 |
| 9 | -22 | 0,68 | 15 | -4 | 1/2 | 11 | 14 | 6 | π/4 | π/5 | π/2 | 0 |
| 10 | 0,81 | 23 | -27 | -22 | 7 | 1/9 | 12 | 10 | π/4 | π/5 | 0 | π/2 |
| 11 | 2 | 0,76 | -30 | -8 | 4 | 10 | 1/8 | 13 | π/4 | 0 | π/5 | π/2 |
| 12 | 30 | -26 | 0,81 | 27 | 4 | 1 | 8 | 1/7 | 0 | π/4 | π/7 | π/2 |
| 13 | -24 | -27 | -2 | 0,88 | 1/6 | 5 | 11 | 11 | π/4 | 0 | π/3 | π/2 |
| 14 | -3 | 6 | 15 | -0,12 | 6 | 1/5 | 3 | 8 | π/5 | π/6 | 0 | π/7 |
| 15 | 15 | 14 | 0,23 | 26 | 10 | 13 | 1/3 | 7 | π/5 | π/7 | π/2 | 0 |
| 16 | -7 | 0,01 | 13 | 17 | 14 | 7 | 11 | 1/2 | π/5 | π/7 | 0 | π/3 |
| 17 | 0,59 | -2 | -12 | -11 | 1/9 | 3 | 9 | 1 | π/5 | 0 | π/2 | π/3 |
| 18 | 29 | 1 | 20 | 0,89 | 3 | 1/8 | 2 | 12 | 0 | π/7 | π/2 | π/3 |
| 19 | 0,13 | 27 | -2 | 19 | 1 | 2 | 1/7 | 9 | π/5 | 0 | π/2 | π/3 |
| 20 | -29 | -0,91 | -12 | -17 | 8 | 7 | 11 | 1/6 | π/2 | π/4 | 0 | π/6 |
| 21 | -9 | 1 | 0,72 | 27 | 3 | 5 | 1/5 | 13 | π/2 | π/4 | π/6 | 0 |
| 22 | 29 | 2 | -19 | 0,76 | 12 | 1/4 | 6 | 8 | π/2 | π/4 | 0 | π/6 |
| 23 | -9 | 18 | 6 | 0,28 | 1/3 | 10 | 12 | 14 | π/2 | 0 | π/7 | π/4 |
| 24 | 11 | -22 | -0,93 | -26 | 4 | 1/2 | 14 | 5 | 0 | π/4 | π/5 | π/2 |
| 25 | -12 | 0,38 | -2 | 2 | 4 | 9 | 1/8 | 15 | π/3 | 0 | π/5 | π/2 |
| 26 | 0,51 | -12 | -18 | 3 | 12 | 7 | 14 | 1/7 | π/3 | π/2 | 0 | π/4 |
| 27 | 0,83 | -6 | -20 | -11 | 14 | 2 | 11 | 1/6 | π/3 | π/2 | π/5 | 0 |
| 28 | -30 | 0,71 | 17 | 15 | 13 | 6 | 1/5 | 2 | π/3 | π/2 | 0 | π/4 |
| 29 | 17 | 21 | -0,46 | -17 | 1 | 1/4 | 1 | 10 | π/3 | 0 | π/5 | π/4 |
| 30 | 28 | -18 | 13 | 0,51 | 1 | 6 | 10 | 1/3 | 0 | π/2 | π/5 | π/4 |

ПРИКЛАД ВИКОНАННЯ

1. Завдання

$$x(t) = A_1 \cos(\omega_1 t + \varphi_1) - A_2 \sin(\omega_2 t + \varphi_2) + A_3 \sin(\omega_3 t + \varphi_3) - A_4 \cos(\omega_4 t + \varphi_4),$$

при цьомуцьому, згідно з варіантом:

| № | A ₁ | A ₂ | A ₃ | A ₄ | ω ₁ | ω ₂ | ω ₃ | ω ₄ | φ ₁ | φ ₂ | φ ₃ | φ ₄ |
|----|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| 31 | 1 | 1 | 1 | 1 | 1 | 4 | 1/3 | 13 | 2π | π/2 | 0 | 0 |

Тобто, аналітичний запис сигналу такий:

$$x(t) = 1 \cos(t + 2\pi) - 1 \sin(4t + \pi/2) + 1 \sin\left(\frac{1}{3}t\right) - 1 \cos(13t).$$

2. Аналітичний розрахунок кроку дискретизації та періоду сигналу

Згідно з теоремою Котельникова: $\Delta t \leq \frac{1}{2f_2}$, де $f_2 = \frac{\omega_2}{2\pi}$ – гранична частота.

Оскільки, заданий сигнал містить різні частоти, то граничною буде **найбільша** з них:

$$\omega_2 = \max(\omega_1, \omega_2, \omega_3, \omega_4) = \max\left(1, 4, \frac{1}{3}, 13\right) = 13. \text{ Отже: } f_2 = \omega_2 / 2\pi = 13 / 2\pi.$$

Підставивши отримане значення у теорему Котельникова, маємо крок дискретизації:

$$\Delta t \leq \pi / 13.$$

Для знаходження періоду заданого сигналу треба знайти найменше спільне кратне між періодами всіх окремих складових сигналу. Таких частин є чотири (чотири доданки присутні в аналітичному представленні сигналу):

$$x_1(t) = 1 \cos(t + 2\pi); \quad x_2(t) = -1 \sin(4t + \pi/2);$$

$$x_3(t) = 1 \sin\left(\frac{1}{3}t\right); \quad x_4(t) = -1 \cos(13t).$$

Як відомо, амплітуда та фаза не впливають на період сигналу, тому до уваги треба брати лише частоту. Отже, складові заданого сигналу мають такі періоди: $T_1 = 2\pi$;

$$T_2 = \frac{2\pi}{4} = \frac{\pi}{2}; \quad T_3 = \frac{2\pi}{1/3} = 6\pi; \quad T_4 = \frac{2\pi}{13}.$$

Очевидно, що найменше спільне кратне становить 6π (воно ділиться без остачі на решту періодів). Таким чином період заданого складеного сигналу становить:

$$T = 6\pi.$$

3. Текст програми

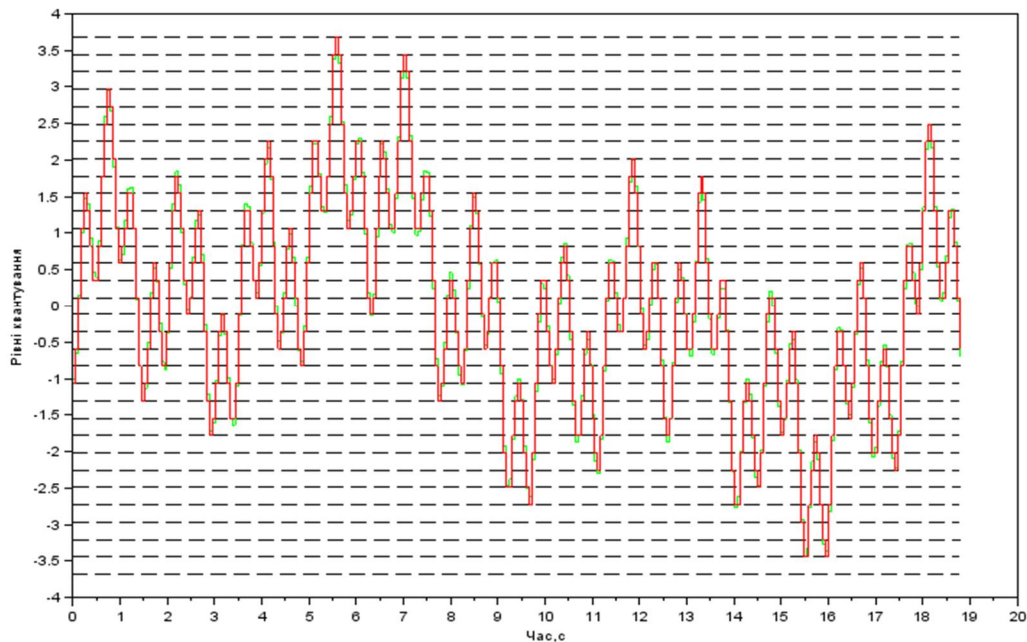
| | |
|---|---|
| clear all | //очистка пам'яті |
| clc | //закриття всіх графічних вікон |
| close () | //очистка екрану |
| A1=1; A2=1; A3=1; A4=1; | //амплітуда |
| w1=1; w2=4; w3=1/3; w4=13; | //частота |
| phi1=2*%pi; phi2=%pi/2; phi3=0; phi4=0; | //фаза |
| M=2^5; | //кількість рівнів квантування |
| koef=2^0; | //коефіцієнт кількості відліків |
| w_gr=max ([w1,w2,w4,w3]); | //гранична кругова частота |
| f_gr=w_gr/ (2*%pi); | //гранична лінійна частота |
| dt=1/ (2*f_gr*koef); | //дискрет часу за теоремою //Котельникова |

| | |
|---|---|
| $T=6*\pi$; | //період з аналітичних розрахунків |
| $t=0:dt:T-dt$; | //вектор часу для одного періоду |
| $x=A1*\cos(w1*t+phi1)-A2*\sin(w2*t+phi2)+A3*\sin(w3*t+phi3)-A4*\cos(w4*t+phi4)$; | //вектор дискретного сигналу |
| $maxA=\max(\text{abs}(x))$ | //максимальне значення амплітуди |
| $minA=-maxA$ | //мінімальне значення амплітуди |
| $N=\text{length}(x)$; | //довжина вектору сигналу |
| $k=(maxA-minA)/(M-1)$; | //квант амплітуди |
| $K=\text{minA}:k:maxA$; | //вектор рівнів квантування |
| $y=\text{floor}(x/k)*k$; if <u>modulo</u> (M,2)==0 $y=y+k/2$; end; | //округлення дискретного значення //сигналу до найближчого рівня //квантування, а отже, отримання //квантованого, тобто цифрового //сигналу |
| $KK=\text{ones}(N,1)*K$; plot (t,KK, 'k--') ff=gca () ff.auto_ticks=["on","on","on"] xlabel ('Час,c'); ylabel ('Рівні квантування') | //відображення рівнів квантування |
| plot2d (t,x,3) | //графік дискретного сигналу |
| plot2d2 (t,y,5) | //графік квантованого сигналу |
| $a=\max(\text{abs}(y-x))$ disp (a,"a=") | //абсолютна похибка |
| $b=(1/N)*(\text{sum}(y)-\text{sum}(x))$ disp (b,"b=") | //середня похибка |
| $d=(1/N)*\text{sum}((y-x).^2)$ disp (d,"d=") | //дисперсія |

4. Оцінка похибки оцифрування

| <i>Koef</i> | <i>M</i> | <i>A</i> | <i>B</i> | <i>D</i> |
|-------------|----------|----------|------------|----------|
| 1 | 8 | 2.8810 | 0.1483 | 3.2191 |
| | 32 | 0.6482 | 0.0335 | 0.1537 |
| | 256 | 0.0789 | -0.0020 | 0.0020 |
| 2 | 8 | 2.8810 | -0.0371 | 2.8944 |
| | 32 | 0.6482 | 0.0084 | 0.1545 |
| | 256 | 0.0791 | 0.0010 | 0.0021 |
| 4 | 8 | 2.8810 | -0.1298 | 2.7520 |
| | 32 | 0.6516 | 10^{-15} | 0.1491 |
| | 256 | 0.0791 | 0.0015 | 0.0020 |
| 8 | 8 | 2.8810 | -0.1112 | 2.7459 |
| | 32 | 0.6529 | -0.0021 | 0.1445 |
| | 256 | 0.0792 | 10^{-4} | 0.0021 |

5. Графіки дискретного та квантованого сигналу для таких параметрів: $M=32$; $koef=4$.



Висновки

В цій лабораторній роботі проведено оцифрування сигналу, заданого аналітичним виразом: $x(t) = \cos(t + 2\pi) - \sin(4t + \pi/2) + \sin\left(\frac{1}{3}t\right) - \cos(13t)$. Для цього визначено крок дискретизації та період досліджуваного сигналу. Вони становлять відповідно: $\Delta t \leq \pi/13$; $T = 6\pi$.

Здійснено оцінку точності оцифрування за критеріями абсолютної, середньої похибки та дисперсії, залежно від частоти дискретизації та кількості рівнів квантування. З отриманих результатів видно, що перший параметр практично не впливає на точність оцифрування, тоді як зі збільшенням другого, точність оцифрування збільшується.

Отже, для коректного представлення сигналу потрібно забезпечити частоту дискретизації не менше ніж $13/\pi$, тобто 5 відліків в одиницю часу та максимально можливу кількість рівнів квантування.

Лабораторна робота № 3

ЗГОРТКА ТА КОРЕЛЯЦІЯ СИГНАЛІВ

МЕТА РОБОТИ

Обрахувати лінійну дискретну згортку заданих сигналів. Знайти автокореляційну та взаємкореляційну характеристики сигналів та дослідити їхні властивості.

ТЕОРЕТИЧНІ ВІДОМОСТІ

Лінійна згортка (*convolution*) – основна операція ЦОС, особливо в режимі реального часу. Математично – вираз згортки у вигляді суми, аналогічний інтегралу згортки, що використовується в теорії лінійних аналогових систем. Проте, дискретна згортка має не лише теоретичне значення за аналогією з інтегралом згортки, а і безпосередньо застосовується при реалізації дискретних систем.

В загальному випадку, для двох послідовностей $x(n)$ та $h(n)$ згортка $y(n)$ визначається виразом:

$$y(n) = x(n) \otimes h(n) = \sum_{k=-\infty}^{\infty} x(k) \cdot h(n-k) \text{ або } y(n) = h(n) * x(n) = \sum_{k=-\infty}^{\infty} x(n-k) \cdot h(k),$$

де \otimes або $*$ – умовні позначення операції згортки.

У системах обробки сигналів перша послідовність $x(n)$ являє собою оброблювані дані (наприклад сигнал на вході системи), а друга – $h(n)$, – називається ядром згортки (*kernel*).

Всі сигнали, що обробляються на комп'ютері, мають скінченну тривалість. Тому доцільно розглядати згортку для вхідних послідовностей $x(n)$ та $h(n)$ довжиною N та M відліків, відповідно. Якщо сигнал $x(n)$ відмінний від нуля лише на відрізку від 0 до $N-1$ (що відповідає довжині N), а ядро згортки визначено на відрізку $-m_1$ до $+m_2$ включно, тобто $M = m_1 + m_2 + 1$, то при підстановці у формулу згортки, отримаємо сигнал $y(n)$, який буде відмінний від нуля на відрізку від $-m_1$ до $N-1 + m_2$, включно. Отже, довжина результуючого сигналу буде **$L=N+M-1$** відлік, тобто сума довжини вхідного сигналу та ядра згортки мінус один. Тому у формулі згортки межі підсумовування зміняться:

$$y(n) = x(n) \otimes h(n) = \sum_{k=0}^{L-1} x(k) \cdot h(n-k) = \sum_{k=0}^{L-1} x(n-k) \cdot h(k).$$

В системі Scilab згортку можна обчислити за допомогою вбудованої функції `conv` ($A, B [, \text{shape}]$) або `convol` (h, x), особливості використання яких описані в довідці.

Кореляційний аналіз сигналів є однією з найважливіших операцій ЦОС. Він застосовується в таких методах обробки сигналів, як виявлення прихованих періодичностей, визначення віддалі до об'єкту локації (може бути радіолокація і ехолокація), визначення швидкості рухомого об'єкту, згладжування шумів тощо.

Коефіцієнт кореляції $r = \sum_{n=0}^{N-1} x(n) \cdot y(n)$ засвідчує міру подібності, міру взаємної залежності між сигналами $x(n)$ і $y(n)$.

$$1. \text{ Для } y(n) = x(n) + \varepsilon(n), \text{ маємо } r = \sum_{n=0}^{N-1} (x(n))^2 + \varepsilon(n) \cdot x(n).$$

При цьому вважають, що якщо r велике, то між $x(n)$ і $y(n)$ є кореляція.

$$2. \text{ Для } y(n) = -x(n) + \varepsilon(n) \text{ маємо } r = \sum_{n=0}^{N-1} -(x(n))^2 + \varepsilon(n) \cdot x(n).$$

При цьому вважають, що якщо r велике по модулю, але від'ємне, то між $x(n)$ і $y(n)$ є кореляція, що розвивається в протилежному напрямку.

3. Якщо $x(n)$ і $y(n)$, або хоча б одне з них є реалізаціями випадкового процесу з рівномірним розподілом на інтервалі $[-1, 1]$: $x(n) \in [-1, 1]$ $y(n) \in [-1, 1]$, при цьомуцьому імовірність того, що $x(n)$ чи $y(n)$ набуде якогось значення $a \in [-1, 1]$ є однаковою, то маємо $r = \sum_{n=0}^{N-1} x(n) \cdot y(n)$. При цьому вважають, що якщо r прямує до нуля, то процеси некорельовані, якщо ж $r \neq 0$, то ці два процеси корелюються і між ними існує зв'язок.

Автокореляційною функцією (АКФ) називають скалярний добуток сигналу та його копії (здебільшого затриманої в часі). Для неперіодичних сигналів АКФ обчислюють за виразами, (1) для аналогового і (2) для цифрового представлення

$$b_{xx}(\tau) = \int_{-\infty}^{\infty} x(t) \cdot x(t + \tau) dt, \quad (1)$$

$$b_{xx}(n) = \sum_{k=-\infty}^{\infty} x(k) \cdot x(k + n), \quad (2)$$

де x – досліджуваний сигнал.

Автокореляційна функція застосовується для визначення часової затримки при проходженні сигналів крізь досліджувану систему.

Взаємкореляційною функцією (ВКФ) називають скалярний добуток двох сигналів. Для неперіодичних сигналів ВКФ обчислюють за виразами, (3) для аналогового і (4) для цифрового представлення. Взаємкореляційна функція застосовується для визначення подібності сигналів та розміщення їх на осі часу.

$$b_{xy}(\tau) = \int_{-\infty}^{\infty} x(t) \cdot y(t + \tau) dt, \quad (3)$$

$$b_{xy}(n) = \sum_{k=-\infty}^{\infty} x(k) \cdot y(k + n), \quad (4)$$

де x, y – досліджувані сигнали.

Бачимо, що формула кореляції дуже подібна до формули згортки. Якщо визначити ядро згортки як $h(k) = y(-k)$, то вони співпадають. Отже, кореляцію можна обраховувати як згортку, взявши в якості ядра згортки, сигнал y , симетрично відбитий відносно нульового відліку.

В системі Scilab кореляційну функцію можна знайти, використовуючи вбудовані функції `corr()`, або `xcorr()`, особливості використання яких описані в довідці.

КОНТРОЛЬНІ ЗАПИТАННЯ

1. Дайте означення автокореляційної функції.
2. Дайте означення взаємкореляційної функції.
3. Поясніть застосування автокореляційної функції.
4. Поясніть застосування взаємкореляційної функції.
5. Що визначає коефіцієнт кореляції?
6. Які функції системи SCILAB забезпечують обчислення кореляційних характеристик сигналів?
7. Які функції системи SCILAB забезпечують обчислення згортки сигналів?

ПОРЯДОК ВИКОНАННЯ РОБОТИ

1. За допомогою системи SCILAB утворити вхідні дискретні послідовності $x(n)$ та $y(n)$, $N = 2^{10}$, та вивести їхні графіки на спільній часовій осі.
2. Обчислити згортку заданих сигналів та вивести графік результуючого сигналу, врахувавши нормуючий коефіцієнт.
3. Обчислити автокореляційні функції заданих сигналів та вивести їх у різних графічних вікнах.
4. Обчислити взаємкореляційну функцію заданих сигналів та вивести її графік на окремому рисунку.
5. Обчислити взаємкореляційну функцію заданих сигналів, використовуючи функцію згортки. Вивести графік на окремому рисунку. Порівняти з отриманим у попередньому пункті.
6. Проаналізувати отримані результати та зробити висновок про кореляційні характеристики заданих сигналів.

ЗМІСТ ЗВІТУ

1. Завдання.
2. Аналітичне представлення заданих сигналів.
3. Текст програми для виконання в SCILAB.
4. Графіки вхідних сигналів на спільній часовій осі.
5. Графік результату згортки.
6. Графіки автокореляційних функцій для кожного із заданих сигналів.
7. Графік взаємкореляційної функції заданих сигналів відносно часової осі.
8. Графік взаємкореляційної функції, знайденої з використання згортки.
9. Висновки.

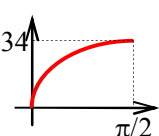
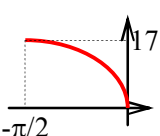
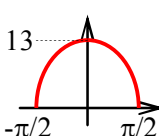
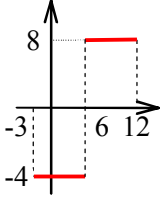
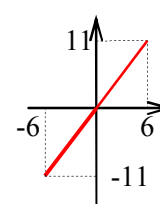
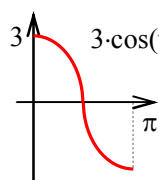
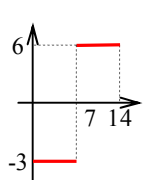
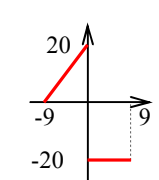
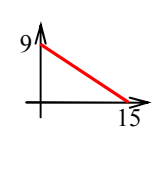
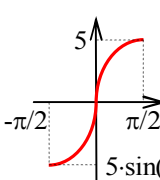
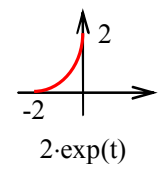
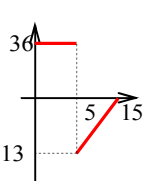
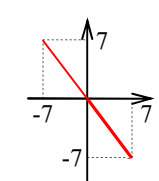
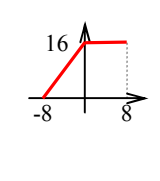
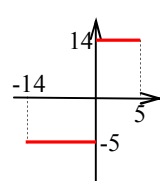
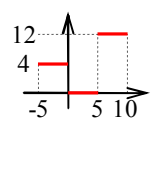
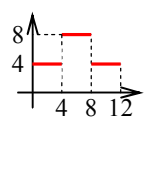
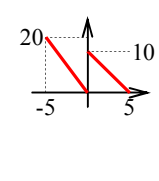
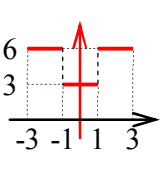
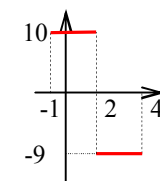
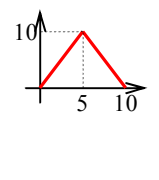
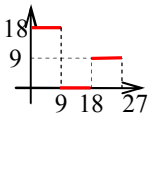
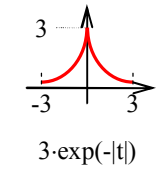
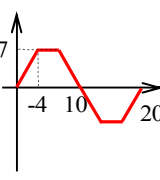
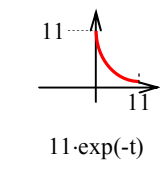
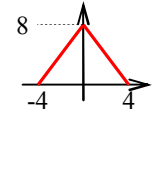
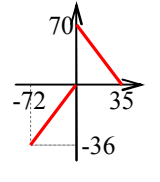
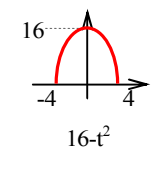
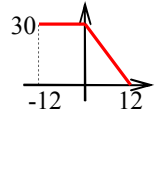
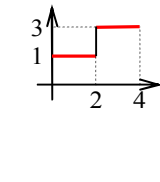
ЗАВДАННЯ

Таблиця 1

| № | $x(t)$ | $y(t)$ | № | $x(t)$ | $y(t)$ |
|----|--------|--------|----|--------|--------|
| 1 | 1 | 30 | 16 | 16 | 1 |
| 2 | 2 | 28 | 17 | 17 | 3 |
| 3 | 3 | 26 | 18 | 18 | 5 |
| 4 | 4 | 24 | 19 | 19 | 7 |
| 5 | 5 | 22 | 20 | 20 | 9 |
| 6 | 6 | 20 | 21 | 21 | 11 |
| 7 | 7 | 18 | 22 | 22 | 13 |
| 8 | 8 | 16 | 23 | 23 | 15 |
| 9 | 9 | 14 | 24 | 24 | 17 |
| 10 | 10 | 12 | 25 | 25 | 19 |
| 11 | 11 | 10 | 26 | 26 | 21 |
| 12 | 12 | 8 | 27 | 27 | 23 |
| 13 | 13 | 6 | 28 | 28 | 25 |
| 14 | 14 | 4 | 29 | 29 | 27 |
| 15 | 15 | 2 | 30 | 30 | 29 |

Форма вхідних сигналів $x(t)$ та $y(t)$ визначена в табл. 2.

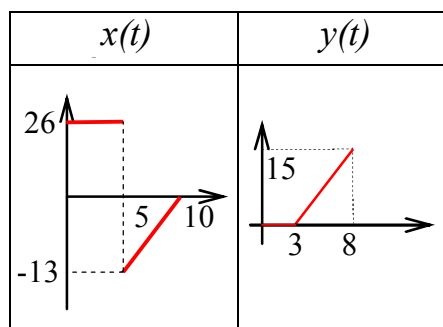
Таблица 2

| № | форма | № | форма | № | форма | № | форма | № | форма |
|---|---|----|--|----|---|----|---|----|--|
| 1 |  $34 \cdot \sin(t)$ | 7 |  $17 \cdot \sin(-t)$ | 13 |  $13 \cdot \cos(t)$ | 19 |  | 24 |  |
| 2 |  $3 \cdot \cos(t)$ | 8 |  | 14 |  | 20 |  | 25 |  $5 \cdot \sin(t)$ |
| 3 |  $2 \cdot \exp(t)$ | 9 |  | 15 |  | 21 |  | 26 |  |
| 4 |  | 10 |  | 16 |  | 22 |  | 27 |  |
| 5 |  | 11 |  | 17 |  $3 \cdot \exp(-t)$ | 23 |  | 28 |  $11 \cdot \exp(-t)$ |
| 6 |  | 12 |  | 18 |  $16 - t^2$ | 29 |  | 30 |  |

ПРИКЛАД ВИКОНАННЯ

1. Завдання

Сигнали задано так:



2. Аналітичне представлення сигналів

Оскільки сигнали визначені на різних часових проміжках, для коректного опрацювання їх треба привести до спільного вектора часу, тобто доповнити нулями коротший із сигналів.

Для сигналу X

Виходячи із заданого графіка, сигнал описується фінітною функцією тривалістю 10 с. Перші 5 с сигнал має сталу амплітуду 26 од. Наступні 5 с – амплітуда змінюється лінійно від -13 до 0, тобто функція, що його описує відповідає рівнянню $x = kt + b$. Для знаходження коефіцієнтів слід розв'язати систему:

$$\begin{cases} -13 = 5k + b \\ 0 = 10k + b \end{cases}$$

Отримуємо: $\begin{cases} -13 = 5k + b \\ b = -10k \end{cases} ; \begin{cases} -13 = 5k - 10k \\ b = -10k \end{cases} ; \begin{cases} -13 = -5k \\ b = -10k \end{cases} ; \begin{cases} k = 13/5 \\ b = -26 \end{cases}$

Тобто, рівняння прямої для заданого часового проміжку становить:

$$x = (13/5)t - 26.$$

Отже, аналітичний опис заданого графіком сигналу буде:

$$x = \begin{cases} 26 & 0 \leq t < 5 \\ (13/5)t - 26 & 5 \leq t < 10 \end{cases}$$

Для сигналу Y

Виходячи із заданого графіка, сигнал описується фінітною функцією, тривалістю 8 с. Перші 3 с сигнал має сталу амплітуду 0 од. Наступні 5 с – амплітуда змінюється лінійно від 0 до 15, тобто функція, що його описує, відповідає рівнянню $x = kt + b$. Для знаходження коефіцієнтів треба розв'язати систему:

$$\begin{cases} 0 = 3k + b \\ 15 = 8k + b \end{cases}$$

Отримуємо: $\begin{cases} b = -3k \\ 15 = 8k - 3k \end{cases} ; \begin{cases} b = -3k \\ 15 = 5k \end{cases} ; \begin{cases} b = -3k \\ k = 3 \end{cases} ; \begin{cases} k = 3 \\ b = -9 \end{cases}$

Тобто, рівняння прямої для заданого часового проміжку становить:

$$y = 3t - 9.$$

Врахувавши, що тривалість цього сигналу менша, аніж попереднього, доповнимо його нульовими значеннями.

Отже, аналітичний опис заданого графіком сигналу буде:

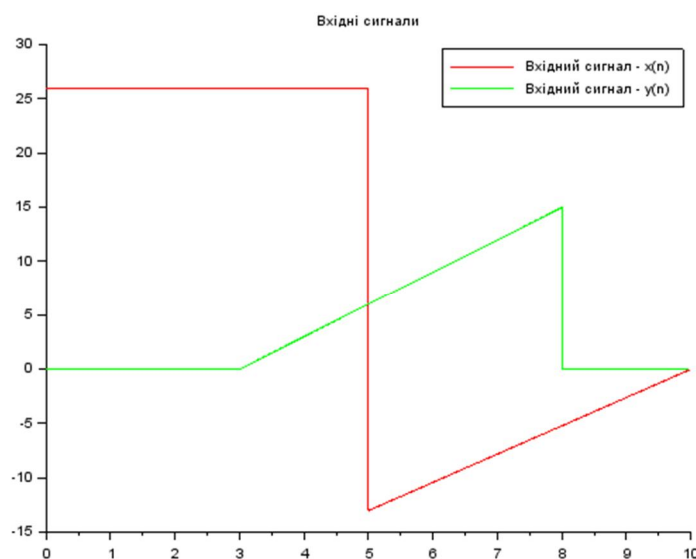
$$y = \begin{cases} 0 & 0 \leq t < 3 \\ 3t - 9 & 3 \leq t < 8 \\ 0 & 8 \leq t < 10 \end{cases}.$$

3. Текст програми для виконання в SCILAB

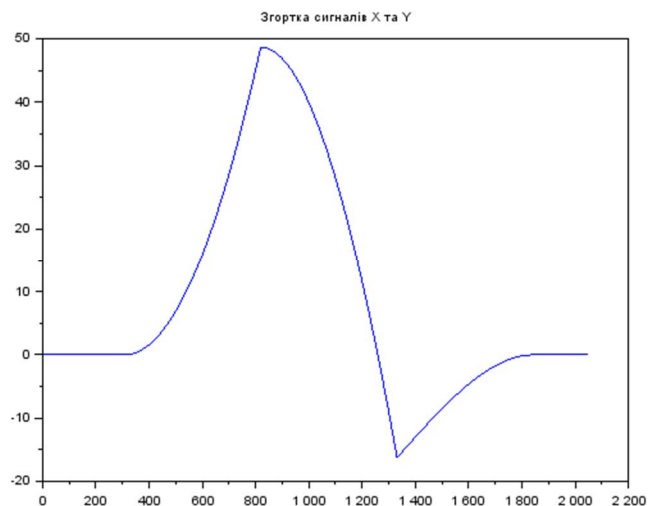
| | |
|--|---|
| <code>clc</code> | <i>//очистка екрану</i> |
| <code>clear</code> | <i>//очистка пам'яті</i> |
| <code>close;close;close;close;close;close;</code> | <i>//закриття всіх графічних вікон</i> |
| <code>m=10;N=2^m;</code> | <i>//кількість точок</i> |
| <code>T1=10;</code> | <i>//інтервал дослідження першої функції</i> |
| <code>dt1=T1/N;</code> | <i>//крок дискретизації першої функції</i> |
| <code>t1=0:dt1:T1-dt1;</code> | <i>//вектор часу для першої функції</i> |
| <code>t11=[t1 (1:N/2)];</code> | <i>//вектор першої частини часу для першої функції</i> |
| <code>t12=[t1 (N/2+1:N)];</code> | <i>//вектор другої частини часу для першої функції</i> |
| <code>x11=26*ones (1,N/2);</code> | <i>//обрахунок першої частини значень першої функції</i> |
| <code>x12=13/5*t12-26;</code> | <i>//обрахунок другої частини значень першої функції</i> |
| <code>x=[x11 x12];</code> | <i>//обчислення першої функції</i> |
| <code>T2=T1;</code> | <i>//інтервал дослідження другої функції</i> |
| <code>dt2=T2/N;</code> | <i>//крок дискретизації другої функції</i> |
| <code>t21=0:dt2:3-dt2;</code> | <i>//вектор першої частини часу для другої функції</i> |
| <code>t22=3:dt2:8-dt2;</code> | <i>//вектор другої частини часу для другої функції</i> |
| <code>t23=8:dt2:T2-dt2;</code> | <i>//вектор третьої частини часу для другої функції</i> |
| <code>t2=[t21 t22 t23]</code> | <i>//вектор часу для другої функції</i> |
| <code>y21=0*t21;</code> | <i>//обрахунок першої частини значень другої функції</i> |
| <code>y22=3*t22-9;</code> | <i>//обрахунок другої частини значень другої функції</i> |
| <code>y23=0*t23;</code> | <i>//обрахунок третьої частини значень другої функції</i> |
| <code>y=[y21 y22 y23];</code> | <i>//обчислення другої функції</i> |
| <code>figure (1,'BackgroundColor',[1,1,1]);</code> | |
| <code>plot2d2 (t1,x,5,);</code> | |
| <code>plot2d2 (t2,y,3)</code> | <i>//вивід функцій на спільному графіку</i> |
| <code>legend ('Вхідний сигнал – x (n)', 'Вхідний сигнал – y (n)')</code> | |
| <code>title ('Вхідні сигнали')</code> | |

| | |
|---|---|
| <code>Bxx= (1/N)*xcorr (x,x);</code> | //знаходження автокореляційної функції першого сигналу |
| <code>figure (2,'BackgroundColor',[1,1,1]);</code> | //вивід автокореляційної функції першого сигналу на графіку |
| <code>plot (Bxx);</code> | |
| <code>title ('Автокореляційна функція сигналу X')</code> | |
| <code>Byy= (1/N)*xcorr (y,y);</code> | //знаходження автокореляційної функції другого сигналу |
| <code>figure (3,'BackgroundColor',[1,1,1]);</code> | //вивід автокореляційної функції другого сигналу на графіку |
| <code>plot (Byy);</code> | |
| <code>title ('Автокореляційна функція сигналу Y')</code> | |
| <code>Bxy= (1/ (2*N))*xcorr (x,y);</code> | //знаходження взаємкореляційної функції |
| <code>figure (4,'BackgroundColor',[1,1,1]);</code> | //вивід взаємкореляційної функції двох сигналів на графіку |
| <code>plot (Bxy);</code> | |
| <code>title ('Взаємкореляційна функція сигналів X та Y')</code> | |
| <code>Cxy=conv (x,y);</code> | //знаходження згортки заданих сигналів |
| <code>figure (5,'BackgroundColor',[1,1,1]);</code> | //вивід графіка згортки сигналів |
| <code>plot (Cxy./ (2*N));</code> | |
| <code>title ('Згортка сигналів X та Y')</code> | |
| <code>y_inv=y (\$:-1:1);</code> | //знаходження оберненої послідовності |
| <code>Cxy1=conv (x,y_inv);</code> | //знаходження згортки сигналу x та оберненого сигналу y |
| <code>figure (6,'BackgroundColor',[1,1,1]);</code> | //вивід графіка згортки цих сигналів |
| <code>plot (Cxy1./ (2*N));</code> | |
| <code>title ('Згортка сигналів X та Y_inv')</code> | |
| <code>[d,d1]=max (Bxy);</code> | //знаходження піку взаємкореляційної функції |
| <code>disp ('Максимальне значення ВКФ становить',d);</code> | //відображення в консолі знайдених параметрів |
| <code>disp ('Максимум досягається у точці з індексом: ',d1);</code> | |

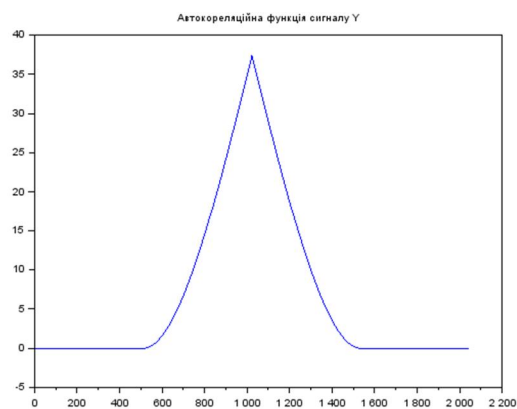
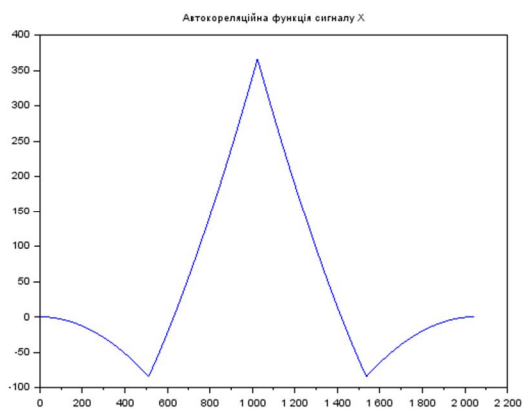
4. Графіки вхідних сигналів на спільній часовій осі



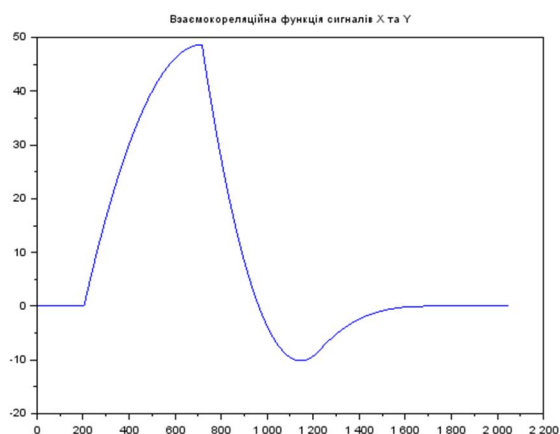
5. Графіки результату згортки сигналів



6. Графіки автокореляційних функцій для кожного із заданих сигналів

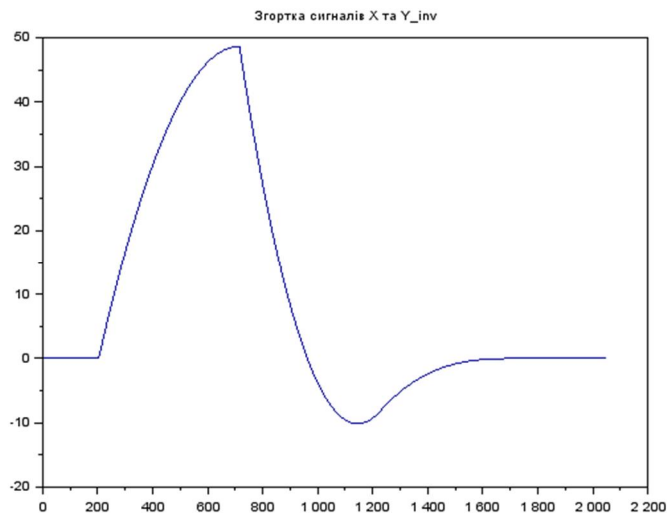


7. Графік взаємкореляційної функції заданих сигналів



Максимальне значення ВКФ становить – 48.654785.
Максимум досягається у точці з індексом – 716.

8. Графік взаємкореляційної функції, знайденої за допомогою згортки



Висновки

З графіка взаємкореляційної функції видно, що вона досягає максимуму в дискрет часу $n=716$. Тому можна стверджувати, що вхідні функції є найбільш корельовані саме в цій точці. Також з форми взаємкореляційної функції видно, що коли один із вхідних сигналів рівний нулю, вона також рівна нулю.

Порівнюючи результати взаємкореляційної функції, обчисленої безпосередньо та знайденої за допомогою згортки, бачимо, що вони повністю збігаються. Це доводить можливість застосування математичного апарату згортки для знаходження кореляційних характеристик сигналів. При цьому одну з послідовностей слід використовувати у зворотному порядку слідування відліків.

Лабораторна робота № 4

ВИЗНАЧЕННЯ І ДОСЛІДЖЕННЯ СПЕКТРАЛЬНИХ ХАРАКТЕРИСТИК СИГНАЛІВ

МЕТА РОБОТИ

Визначити спектр заданого сигналу засобами системи SCILAB. Дослідити властивості сигналу, провівши фільтрацію в частотній області.

ТЕОРЕТИЧНІ ВІДОМОСТІ

Для знаходження частотних характеристик сигналу в практиці ЦОС використовується Дискретне Перетворення Фур'є (ДПФ), а для відтворення сигналу за його спектром – Обернене Дискретне Перетворення Фур'є (ОДПФ).

Прямим та оберненим дискретним перетворенням Фур'є (ДПФ) називають пару взаємооднозначних лінійних перетворень виду (1), (2).

$$\text{Пряме (ДПФ):} \quad X(k) = \sum_{n=0}^{N-1} x(n)W_N^{nk} \quad k = 0,1,\dots,(N-1), \quad (1)$$

$$\text{Обернене (ОДПФ):} \quad x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k)W_N^{-nk} \quad n = 0,1,\dots,(N-1), \quad (2)$$

де $x(n)$, $n = 0,1,\dots,(N-1)$ – послідовність відліків в часовій області; $X(k)$, $k = 0,1,\dots,(N-1)$ – дискретні коефіцієнти Фур'є – один період послідовності в частотній області; k – номер відліку послідовності $X(k)$, що відповідає частоті $k\Delta\omega$; $\Delta\omega = 2\pi/T$, T – тривалість сигналу; W_N^{nk} – повертаючий множник

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

При цьому величина f_d/N називається роздільною здатністю за частотою.

Пряме дискретне перетворення Фур'є (1) призначено для виконання Фур'є-аналізу, тобто визначає спектральні компоненти (складові) $X(k)$ сигналу $x(n)$. Обернене перетворення Фур'є (2) забезпечує Фур'є – синтез сигналу $x(n)$ за заданим набором спектральних компонент $X(k)$. У загальному випадку послідовності $x(n)$ і $X(k)$ – комплексні. Якщо ж $x(n)$ – дійсна послідовність, то $X(k)$ є комплексно спряженою: $\text{Re}\{X(k)\} = \text{Re}\{X(N-k)\}$, $\text{Im}\{X(k)\} = -\text{Im}\{X(N-k)\}$. Для дійсних сигналів спектральні компоненти з номерами $k=N/2+1, N/2+2, \dots, N-1$ відповідають від'ємним частотам і не мають фізичного змісту.

Швидким перетворенням Фур'є (ШПФ) називають групу алгоритмів, що суттєво зменшують обчислювальні затрати при обчисленні прямого чи оберненого перетворень у порівнянні з безпосереднім способом, що ґрунтується на формулах (1) чи (2). Серед

відомих алгоритмів ШПФ найпростішу структуру має алгоритм Кулі–Тьюкі за основою два (ШПФ2). Його основна ідея полягає в рекурсивному (при $N=2^m$) зведенні K -точкових ($K=N, N/2, \dots, 4, 2$) перетворень до двох $K/2$ -точкових. При часовому прорідженні з цією метою застосовується формула розкладу:

$$X(k) = X_1(k) + W_N^k X_2(k), \quad k=0,1,2,\dots,K-1, \quad (3)$$

де $X(k) = \text{ДПФ}_K \{x(n)\}$, $X_1(k) = \text{ДПФ}_{K/2} \{x(2n)\}$, $X_2(k) = \text{ДПФ}_{K/2} \{x(2n+1)\}$.

Якщо обчислюється $N=2^m$ – точкове перетворення комплексної послідовності, то кількість операцій комплексного множення μ_N і додавання α_N в алгоритмі ШПФ рівні: $\mu_N = N/2 \log_2 N$, $\alpha_N = N/2 \log_2 N$. Порівняно з безпосереднім способом обчислення перетворень (1) чи (2), який потребує N^2 комплексних множень і $N(N-1)$ комплексних додавань, обчислювальні затрати суттєво скорочуються – приблизно в $N/\log_2 N$ разів (наприклад, при $N = 1024$ – в сотні разів).

Крім нормуючого постійного множника $1/N$, в оберненому ДПФ маємо комплексно-спряжені повертаючі множники $W_N^{-kn} = \{W_N^{kn}\}^*$. Для розробки алгоритму швидкого оберненого ДПФ використовуємо рівність $x(n) = \text{ДПФ}_N^{-1} \{X(k)\} = 1/N [\text{ДПФ}_N \{X^*(k)\}]^*$, що отримується з (2) в результаті операції комплексного спряження. Інакше кажучи, для обчислення оберненого ДПФ послідовності $X(k)$ за допомогою алгоритму прямого ШПФ достатньо: знайти комплексно спряжену послідовність $X^*(k)$; обчислити її пряме ДПФ – $x_1(n)$; виконати операції комплексного спряження і множення на нормуючий множник отриманої послідовності: $x(n) = 1/N \{x_1(n)\}^*$.

Фур'є-образ $X(k)$ називають комплексним спектром або просто *спектром* дискретного сигналу.

Комплексну функцію $X(k)$ можна виразити через її модуль і аргумент (як і довільне комплексне число):

$$X(k) = |X(k)| \cdot e^{j \arg(X(k))},$$

де $|X(k)| = \sqrt{\text{Re}^2(X(k)) + \text{Im}^2(X(k))}$ називають амплітудним спектром;

$\arg(X(k)) = -\arctg \frac{\text{Im}(X(k))}{\text{Re}(X(k))}$ називають фазовим спектром.

Фільтрація сигналів. Математично процес фільтрації полягає у перетворенні спектру сигналу відповідно з заданою функцією, що визначає спектральну характеристику фільтра. Залежно від призначення фільтра, а отже і загального виду його частотної характеристики, виділяють такі основні, найбільш розповсюджені, типи фільтрів частотної селекції (вибіркові фільтри):

- НЧ-фільтри (*low-pass filters*) – фільтр низьких частот – пропускає всі частоти нижче заданої;
- ВЧ-фільтри (*high-pass filters*) – фільтр високих частот пропускає всі частоти вище заданої;

- смугові фільтри (*band-pass filters*) – пропускають лише визначену смугу частот;
- режекторні фільтри (*band-reject filters*) – пригашають визначену смугу частот.

Апроксимація сигналів. Апроксимація передбачає наближення частотно-фільтрованого сигналу до оригіналу в часовій області шляхом виконання оберненого перетворення Фур'є.

Теорема Парсеваля

Теорема встановлює зв'язок між енергією дискретного сигналу, обчисленою у часовій та у частотній областях:

$$E = \sum_{n=0}^{N-1} |x(n)|^2 = \frac{1}{N} \sum_{k=0}^{N-1} |X(k)|^2,$$

Де $\sum_{n=0}^{N-1} |x(n)|^2$ – енергія сигналу, обчислена в часовій області; $\frac{1}{N} \sum_{k=0}^{N-1} |X(k)|^2$ – енергія сигналу, обчислена в частотній області.

Тобто, якщо аналіз сигналу проводиться в частотній області, ця рівність дозволяє обчислити енергію сигналу безпосередньо за відліками ДПФ, не виконуючи ОДПФ.

КОНТРОЛЬНІ ЗАПИТАННЯ

1. Дайте означення прямого і оберненого (ДПФ). Поясніть зміст Фур'є – аналізу і Фур'є – синтезу сигналів.
2. Котрі спектральні компоненти дійсного сигналу стосуються від'ємних частот, чи мають вони фізичний зміст?
3. Як співвідносяться ДПФ і ШПФ? Як обчислити обернене ДПФ за допомогою алгоритму прямого ШПФ?
4. Що зв'язує рівність Парсеваля?
5. Дайте коротке визначення фільтрації і апроксимації сигналів.
6. Які типи селективної частотної фільтрації вам відомі?
7. Які функції системи SCILAB забезпечують обчислення спектральних характеристик сигналів?

ПОРЯДОК ВИКОНАННЯ РОБОТИ

1. За допомогою системи SCILAB утворити дискретну послідовність заданого сигналу для $N=256$ та обчислити її ДПФ за допомогою алгоритму ШПФ (функція "fft ()"). Вивести графічно обидві послідовності (для частотної характеристики – на різних графіках дійсну та уявну складові).

$$X(k) = ШПФ_N \{x(n)\}.$$

2. Модифікувати отриманий дискретний спектр сигналу, вилучивши певну частину ($p=N/4$) спектральних коефіцієнтів Вивести модифіковані амплітудні спектри на графіки.

2.1. Відкидання високочастотних складових:

$$X_{m1}(k) = \begin{cases} 0 & ; \quad N/2-p < k < N/2 + p \\ X(k) & . \end{cases}$$

2.2. Відкидання низькочастотних складових:

$$X_{m2}(k) = \begin{cases} X(k) & ; \quad N/2-p < k < N/2 + p \\ 0 & . \end{cases}$$

2.3. Відкидання смуги середніх частотних складових:

$$X_{m3}(k) = \begin{cases} 0 & ; \quad N/4-p/2 < k < N/4 + p/2, \quad 3N/4-p/2 < k < 3N/4 + p/2 \\ X(k) & . \end{cases}$$

2.4. Відкидання всіх частот, крім середніх:

$$X_{m4}(k) = \begin{cases} X(k) & ; \quad N/4-p/2 < k < N/4 + p/2, \quad 3N/4-p/2 < k < 3N/4 + p/2 \\ 0 & . \end{cases}$$

3. Знайти обернені перетворення (функція “ifft ()”) отриманих в п.1 та в п.2 дискретних спектрів та порівняти їх, вивівши на спільний графік.

$$x_1(n) = \text{ОШПФ}_N \{X(k)\}; \quad x_{m_i}(n) = \text{ОШПФ}_N \{X_{m_i}(k)\}.$$

4. Змінюючи кількість відкинутих спектральних значень $p=N/32, p=N/16, p=N/8$, дослідити зміни відновленого сигналу.

5. Зробити висновок про можливість апроксимації заданого сигналу з середньою точністю 1 % та 5 %.

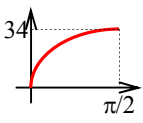
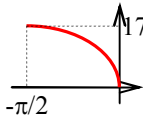
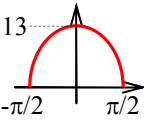
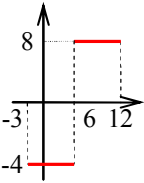
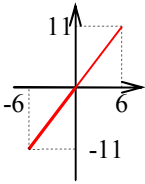
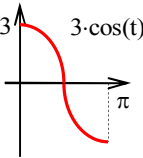
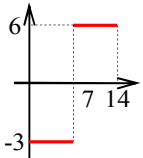
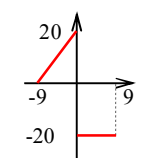
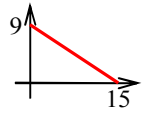
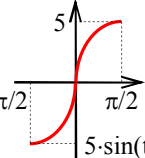
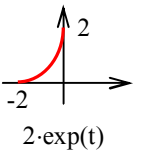
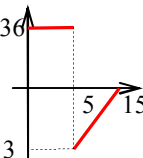
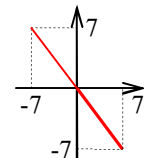
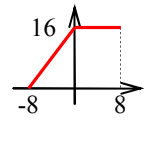
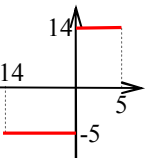
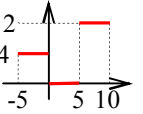
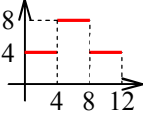
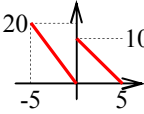
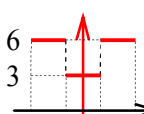
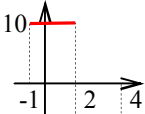
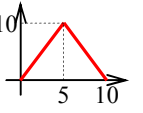
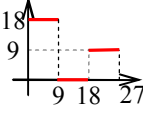
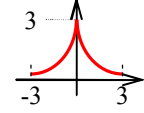
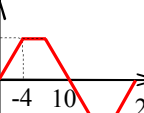
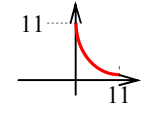
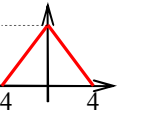
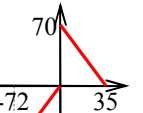
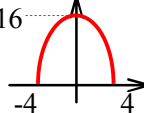
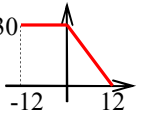
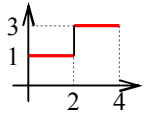
6. Переконалися у вірності теореми Парсеваля.

7. Дослідити втрату енергії за різних параметрів фільтрації.

ЗМІСТ ЗВІТУ

1. Завдання.
2. Аналітичне представлення заданого сигналу.
3. Текст програми для виконання в SCILAB.
4. Графік вхідної послідовності для 256 точок та її частотного спектру.
5. Графік амплітудного спектру вхідного сигналу та модифікованих амплітудних спектрів.
6. Графіки відновлених сигналів за всіма спектральними послідовностями.
7. Значення середніх похибок відновлених сигналів (у відсотках).
8. Числові значення енергії, обраховані в різних областях представлення сигналу.
9. Висновки.

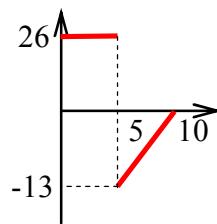
ЗАВДАННЯ

| № | форма | № | форма | № | форма | № | форма | № | форма |
|---|---|----|--|----|---|----|---|----|--|
| 1 |  $34 \cdot \sin(t)$ | 7 |  $17 \cdot \sin(-t)$ | 13 |  $13 \cdot \cos(t)$ | 19 |  | 24 |  |
| 2 |  $3 \cdot \cos(t)$ | 8 |  | 14 |  | 20 |  | 25 |  $5 \cdot \sin(t)$ |
| 3 |  $2 \cdot \exp(t)$ | 9 |  | 15 |  | 21 |  | 26 |  |
| 4 |  | 10 |  | 16 |  | 22 |  | 27 |  |
| 5 |  | 11 |  | 17 |  $3 \cdot \exp(- t)$ | 23 |  | 28 |  $11 \cdot \exp(-t)$ |
| 6 |  | 12 |  | 18 |  $16 \cdot t^2$ | 29 |  | 30 |  |

ПРИКЛАД ВИКОНАННЯ

1. Завдання

Сигнал задано так:



2. Аналітичне представлення сигналу

Виходячи із заданого графіка, сигнал описується фінитною функцією, тривалістю 10 с. Перші 5 с сигнал має сталу амплітуду 26 од. Наступні 5 с – амплітуда змінюється лінійно від -13 до 0, тобто функція, що його описує, відповідає рівнянню $x = kt + b$. Для знаходження коефіцієнтів треба розв'язати систему:

$$\begin{cases} -13 = 5k + b \\ 0 = 10k + b \end{cases}$$

Отримуємо: $\begin{cases} -13 = 5k + b \\ b = -10k \end{cases}; \begin{cases} -13 = 5k - 10k \\ b = -10k \end{cases}; \begin{cases} -13 = -5k \\ b = -10k \end{cases}; \begin{cases} k = 13/5 \\ b = -26 \end{cases}$

Тобто рівняння прямої для заданого часового проміжку становить:

$$x = (13/5)t - 26.$$

Отже, аналітичний опис заданого графіком сигналу буде:

$$x = \begin{cases} 26 & 0 \leq t < 5 \\ (13/5)t - 26 & 5 \leq t < 10 \end{cases}$$

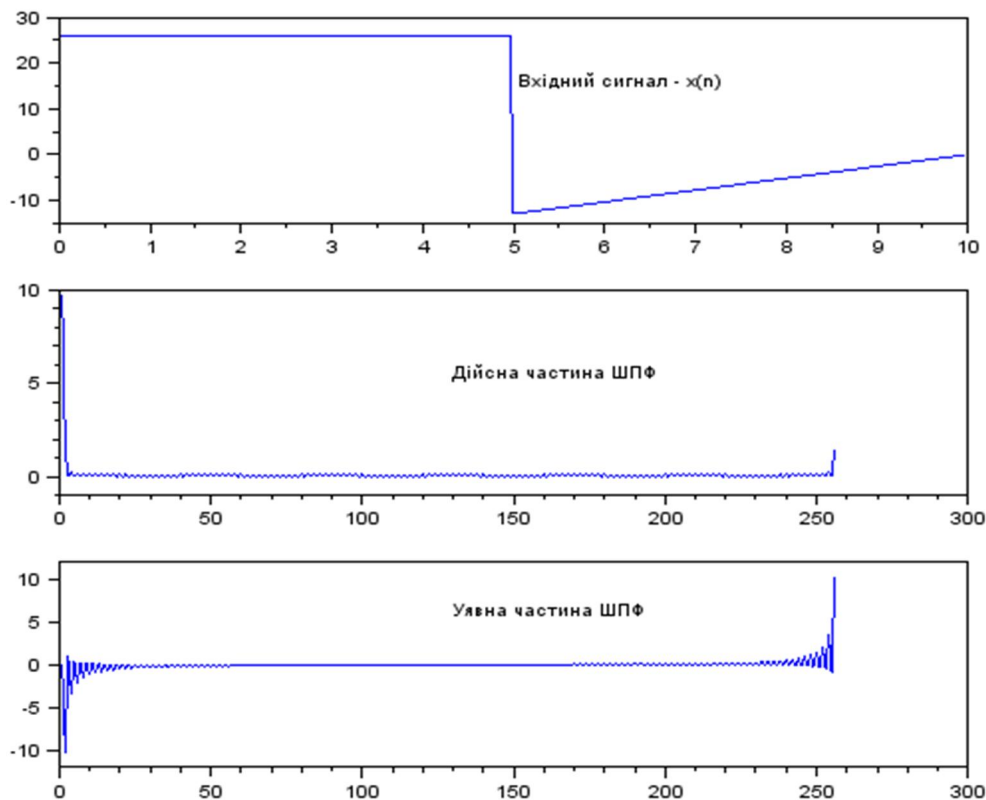
3. Текст програми

| | |
|---------------------------|---|
| clc | //очистка екрану |
| clear | //очистка пам'яті |
| close, close | //закриття всіх графічних вікон |
| T=10;m=8;N=2^m; | //задання параметрів вхідного сигналу |
| p=N/4;k1=N/2-p; k2=N/2+p; | //визначенн параметрів фільтрації |
| dt=T/N; | //знаходження кроку дискретизації |
| t1=0:dt:5-dt;; | //задання першого часового проміжку сигналу |
| t2=5:dt:10-dt; | //задання другого часового проміжку сигналу |
| t=[t1,t2]; | //об'єднання двох часових проміжків |
| x1=26+0*t1; | //опис першої частини сигналу |
| x2= (13/5)*t2-26; | //опис другої частини сигналу |
| x=[x1,x2]; | //об'єднання двох частин сигналу |

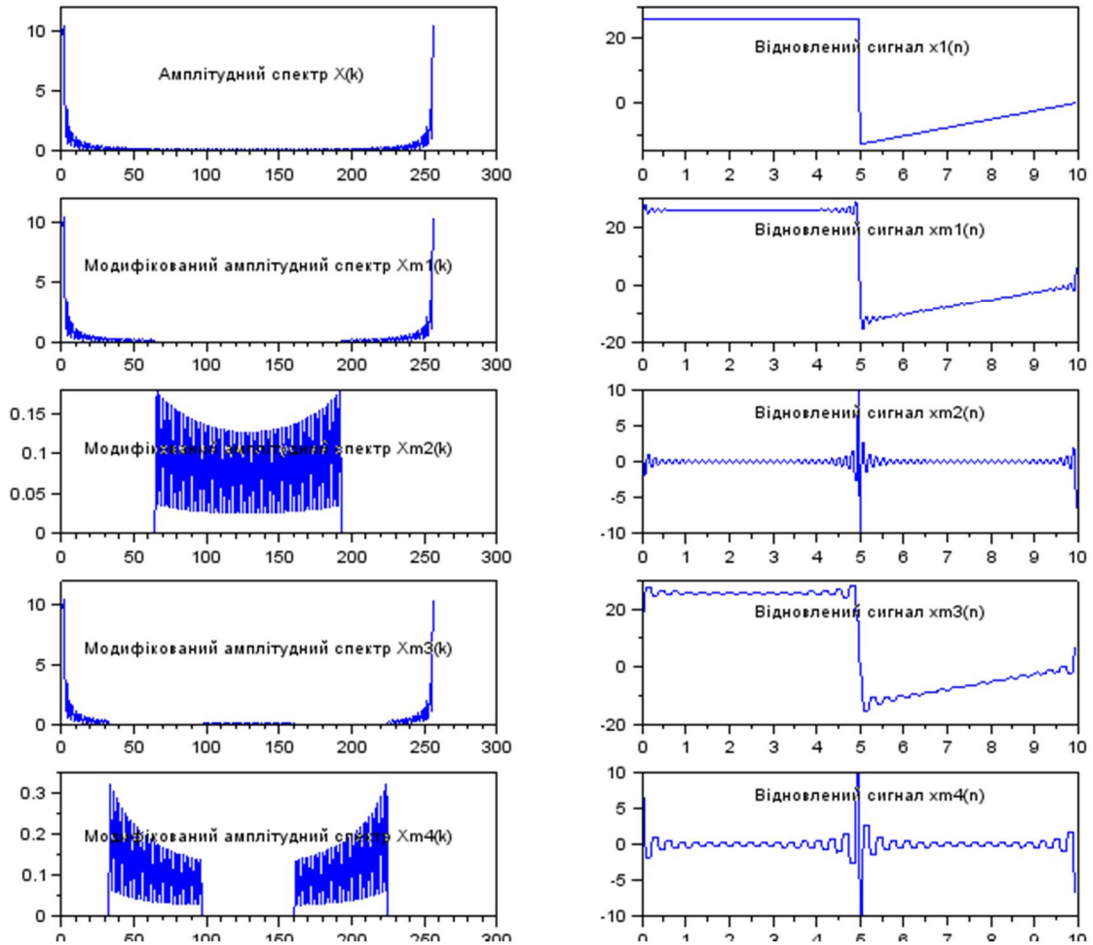
| | |
|---|--|
| $dw=2*\%pi/T;$ | //визначення роздільної здатності за частотою |
| $w=1:1:N;$ | //знаходження вектора частот |
| $Sx = \text{fft} (x)/ (N);$ | //обрахунок спектральних коефіцієнтів за алгоритмом ШПФ |
| <u>figure</u> (1,'BackgroundColor',[1,1,1]); | |
| <u>subplot</u> (3,1,1), <u>plot</u> (t, x), <u>title</u> ('Вхідний сигнал – x (n)'), 'position',[T/2,max (x)/2]); | //відображення в першому графічному вікні вхідного //сигналу та дійсної і уявної частини його //частотного спектру |
| <u>subplot</u> (3,1,2), <u>plot</u> (w, real (Sx)), <u>title</u> ('Дійсна частина ШПФ', 'position',[N/2,max (real (Sx))/2]); | |
| <u>subplot</u> (3,1,3), <u>plot</u> (w, imag (Sx)), <u>title</u> ('Уявна частина ШПФ', 'position',[N/2,max (imag (Sx))/2]); | |
| $Sx1 = [Sx (1 : k1) \text{zeros} (1, k2 - k1) Sx (k2 + 1 : N)];$ | |
| $Sx2 = [\text{zeros} (1, p) Sx (p+1 :N-p) \text{zeros} (1, p)];$ | //відкидання "р" низьких частот спектру |
| $Sx3 = [Sx (1 : N/4-p/2),\text{zeros} (1, p),Sx ((N/4+p/2)+1:3*N/4-p/2), \text{zeros} (1, p), Sx (3*N/4+p/2+1:N)];$ | //відкидання "р" середніх частот спектру |
| $Sx4 = [\text{zeros} (1, N/4-p/2),Sx ((N/4-p/2)+1 :N/4+p/2), \text{zeros} (1, N/2-p), Sx (3*N/4-p/2+1:3*N/4+p/2),\text{zeros} (1,N/4-p/2)];$ | //відкидання "р" низьких і "р" високих частот //спектру |
| <u>figure</u> (2,'BackgroundColor',[1,1,1]); | |
| <u>subplot</u> (5,2,1), <u>plot</u> (w, abs (Sx)), <u>title</u> ('Амплітудний спектр X (k)', 'position',[N/4,max (abs (Sx))/2]);; | //відображення в другому графічному вікні //немодифікованого та модифікованих амплітудних //спектрів |
| <u>subplot</u> (5,2,3), <u>plot</u> (w, abs (Sx1)), <u>title</u> ('Модифікований амплітудний спектр Xm1 (k)', 'position',[N/20,max (abs (Sx1))/2]); | |
| <u>subplot</u> (5,2,5), <u>plot</u> (w, abs (Sx2)), <u>title</u> ('Модифікований амплітудний спектр Xm2 (k)', 'position',[N/20,max (abs (Sx2))/2]); | |
| <u>subplot</u> (5,2,7), <u>plot</u> (w, abs (Sx3)), <u>title</u> ('Модифікований амплітудний спектр Xm3 (k)', 'position',[N/20,max (abs (Sx3))/2]); | |
| <u>subplot</u> (5,2,9), <u>plot</u> (w, abs (Sx4)), <u>title</u> ('Модифікований амплітудний спектр Xm4 (k)', 'position',[N/20,max (abs (Sx4))/2]); | |
| $x1 = N*\text{ifft} (Sx);$ | //відтворення вхідного сигналу за його //не-модифікованим спектром, шляхом знаходження //оберненого перетворення Фур'є |
| $xm1 = N*\text{ifft} (Sx1);$ | //відтворення вхідного сигналу за його //модифікованими спектрами |
| $xm2 = N*\text{ifft} (Sx2);$ | |
| $xm3 = N*\text{ifft} (Sx3);$ | |
| $xm4 = N*\text{ifft} (Sx4);$ | |
| <u>subplot</u> (5, 2, 2), <u>plot</u> (t, x1), <u>title</u> ('Відновлений сигнал x1 (n)'), 'position',[T/4,max (abs (x1))/2]); | //відображення в другому графічному вікні сигналів, //відтворених за різними спектрами |
| <u>subplot</u> (5, 2, 4), <u>plot</u> (t, real (xm1)), <u>title</u> ('Відновлений сигнал xm1 (n)'), 'position',[T/4,max (abs (xm1))/2]); | |
| <u>subplot</u> (5, 2, 6), <u>plot</u> (t, real (xm2)), <u>title</u> ('Відновлений сигнал xm2 (n)'), 'position',[T/4,max (abs (xm2))/2]); | |

| | |
|--|--|
| <code>subplot (5, 2, 8), plot (t, real (xm3)), title ('Відновлений сигнал xm3 (n)', 'position', [T/4, max (abs (xm3))/2]);</code> | |
| <code>subplot (5, 2, 10), plot (t, real (xm4)), title ('Відновлений сигнал xm4 (n)', 'position', [T/4, max (abs (xm4))/2]);</code> | |
| <code>onePr=100/ (N*max (abs (26),abs (-13)));</code> | |
| <code>bm1=onePr* (sum (abs (x-xm1)))</code> | |
| <code>bm2=onePr* (sum (abs (x-xm2)))</code> | |
| <code>bm3=onePr* (sum (abs (x-xm3)))</code> | |
| <code>bm4=onePr* (sum (abs (x-xm4)))</code> | <i>//обрахунок і відображення в консолі</i> |
| <code>disp (N,"N=")</code> | <i>значення //похибок для різних типів</i> |
| <code>disp (p,"p=")</code> | <i>проведеної фільтрації</i> |
| <code>disp (bm1,"bm1=")</code> | |
| <code>disp (bm2,"bm2=")</code> | |
| <code>disp (bm3,"bm3=")</code> | |
| <code>disp (bm4,"bm4=")</code> | |
| <code>E_time=sum (abs (x).^2)/N;</code> | <i>//обрахунок енергії в часовій області дослідження</i> |
| <code>disp ("E_time",E_time);</code> | <i>//відображення в консолі</i> |
| <code>E_spektr=sum (abs (Sx).^2);</code> | <i>// обрахунок енергії в частотній обл. дослідження</i> |
| <code>disp ("E_spektr",E_spektr);</code> | <i>// відображення в консолі</i> |

4. Графіки вхідної послідовності для 256 точок та її частотного спектру



5, 6. Графіки амплітудних спектрів та відновлених сигналів (для $p = N/4$)



7. Значення похибок відновлених сигналів (у відсотках)

| $N = 256$ | $\epsilon (x_{m1}), \%$ | $\epsilon (x_{m2}), \%$ | $\epsilon (x_{m3}), \%$ | $\epsilon (x_{m4}), \%$ |
|-------------|-------------------------|-------------------------|-------------------------|-------------------------|
| $p=N/4=64$ | 1.7717912 | 62.510928 | 3.112073 | 62.374876 |
| $p=N/8=32$ | 1.2528406 | 62.31915 | 2.2005371 | 62.586001 |
| $p=N/16=16$ | 1.0524822 | 61.96566 | 1.8231564 | 62.613953 |
| $p=N/32=8$ | 0.9029482 | 61.295403 | 1.5299611 | 62.590289 |

8. Значення енергії, обраховані в різних областях представлення сигналу

$E_{time} = 366.49760.$

$E_{spektr} = 366.49760.$

Значення енергії сигналів за $p=N/4$

| | 1 | 2 | 3 | 4 |
|-----------------|-----------|-----------|-----------|-----------|
| $E_{timeXmn}$ | 365.12774 | 1.3698595 | 363.75860 | 2.7390027 |
| $E_{spektrSxn}$ | 365.12774 | 1.3698595 | 363.75860 | 2.7390027 |
| E_{diff} | 1.36986 | 365.12774 | 2.739 | 363.75859 |

Висновки

В цій лабораторній роботі було отримано спектральну характеристику заданого фінітного сигналу. Це зроблено за допомогою системи SCILAB, а саме, шляхом використання вбудованої функції $\text{fft}()$, яка дозволяє обчислювати швидке перетворення Фур'є N – точкової послідовності. Графік частотного складу сигналу приведено в п. 4.

Крім того, проводилась частотна фільтрація сигналу, тобто відкидання різних частин спектральних коефіцієнтів, а далі – відновлення сигналу за допомогою оберненого швидкого перетворення Фур'є.

Як показало дослідження, при відкиданні спектральних складових, що відповідають за високі частоти, відтворений сигнал спотворюється незначно, тоді як відкидання низькочастотної складової призводить до повної втрати початкового сигналу, за винятком точки перепаду амплітуди. Тому, для апроксимації заданого сигналу слід використовувати лише низькочастотну складову. Так, для апроксимації з середньою точністю 5 % достатньо взяти першу чверть спектральних коефіцієнтів, а для апроксимації з середньою точністю 1 % можна відкидати лише $N/16$ спектральних складових. Якщо ж основною задачею при дослідженні сигналу є виявлення різких перепадів амплітуди, то варто використовувати високочастотну частину спектру.

Знайдені числові значення енергії сигналу в часовій та частотній області – повністю збігаються, що підтверджує вірність теореми Парсеваля. При фільтрації сигналу виявлено, що максимальні втрати енергії спостерігаються при відкиданні низьких частот та при смуговій фільтрації.

Лабораторна робота № 5

ДОСЛІДЖЕННЯ ЛІНІЙНИХ ДИСКРЕТНИХ СИСТЕМ У ЧАСОВІЙ ТА ЧАСТОТНІЙ ОБЛАСТЯХ

МЕТА РОБОТИ

Визначити характеристики лінійної дискретної системи в часовій та частотній області. Дослідити властивості заданої ЛДС.

ТЕОРЕТИЧНІ ВІДОМОСТІ

Математичне моделювання обробки сигналів лінійною дискретною системою (ЛДС) включає в себе:

- розрахунок характеристик ЛДС в часовій області, z -області, та частотній області;
- розрахунок реакції ЛДС на вхідний сигнал по різницевому рівнянню;
- аналіз вхідних сигналів та реакцій ЛДС в часовій та частотних областях.

В Scilab математичною моделлю ЛДС вважають співвідношення між входом та виходом у вигляді різницевого рівняння або системи рівнянь, які дозволяють розрахувати реакцію системи на заданий вхідний вплив.

В **часовій області** основною характеристикою ЛДС є імпульсна характеристика $h[n]$, а моделювання роботи ЛДС (розрахунок реакції) виконується з використанням одного з таких співвідношень між входом та виходом.

1. Різницевого рівняння

$$y[n] = b_0x[n] + b_1x[n-1] + \dots + b_{N-1}x[n-(N-1)] - a_1y[n-1] - a_2y[n-2] - \dots - a_{M-1}y[n-(M-1)],$$

яке задається векторами коефіцієнтів $b = [b_0, b_1, \dots, b_{N-1}]$ та $a = [a_0, a_1, \dots, a_{M-1}]$.

В Scilab всі функції аналізу систем побудовані для випадку, коли перший елемент вектору коефіцієнтів a дорівнює одиниці: $a_0 = 1$.

2. Формули згортки

$$y[n] = \sum_{k=0}^{\infty} x[k]h[n-k] = \sum_{k=0}^{\infty} h[k]x[n-k].$$

3. Системи рівнянь змінних стану.

В **z -області** основною характеристикою ЛДС є передавальна функція:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{n=0}^{N-1} b_n z^{-n}}{\sum_{m=0}^{M-1} a_m z^{-m}} = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_{N-1} z^{-(N-1)}}{1 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_{M-1} z^{-(M-1)}},$$

яка, подібно до різницевого рівняння, задається векторами коефіцієнтів b та a та може мати різні види математичного запису.

В частотній області основною характеристикою ЛДС є комплексна частотна характеристика (КЧХ), а також її модуль (амплітудно-частотна характеристика, АЧХ) та аргумент (фазо-частотна характеристика, ФЧХ):

$$H(e^{j\omega}) = |H(e^{j\omega})| e^{j \arg\{H(e^{j\omega})\}} = K(\omega) e^{j\Phi(\omega)}.$$

КЧХ є Фур'є-відображенням імпульсної характеристики $h(n)$, яке визначається за допомогою перетворення Фур'є (приведено для нормованих частоти та часу):

$$H(\hat{\omega}) = \sum_{n=0}^{\infty} h(n) e^{-j\hat{\omega}n}.$$

За відомою КЧХ, імпульсна характеристика знаходиться за допомогою оберненого перетворення Фур'є.

Оскільки між перетворенням Фур'є та z-перетворенням існує однозначний зв'язок, то маючи передавальну функцію, можна автоматично отримати частотну характеристику, шляхом підстановки $z = e^{j\hat{\omega}}$:

$$H(\hat{\omega}) = \frac{\sum_{i=0}^{N-1} b_i e^{-ji\hat{\omega}}}{1 + \sum_{k=1}^{M-1} a_k e^{-jk\hat{\omega}}}. \quad (1)$$

Важливим є те, що частотна характеристика, як і передавальна функція, залежить виключно від внутрішніх параметрів ЛДС і не залежить від впливу та від відгуку.

КОНТРОЛЬНІ ЗАПИТАННЯ

1. Що таке ЛДС?
2. Що таке ІХ?
3. Що таке АЧХ?
4. Що таке ФЧХ?
5. Що таке ШПФ?
6. Які властивості ЛДС ви знаєте?
7. В чому полягає властивість однорідності ЛДС?
8. В чому полягає властивість адитивності ЛДС?
9. Яка умова стійкості ЛДС (в часовій області)?
10. Яка умова стійкості ЛДС (в частотній області)?

ПОРЯДОК ВИКОНАННЯ РОБОТИ

Визначення характеристик ЛДС з використанням різницевого рівняння

1. Записати різницеве рівняння для заданої ЛДС. Задати в програмі вектори a та b коефіцієнтів рекурсивної та нерекурсивної частини ЛДС (згідно з таблицею варіантів).

2. Сформувати відліки синусоїдного сигналу частоти 10 Гц тривалістю 1 с. амплітуди 1 В, дискретизованого з частотою 256 Гц. Розрахувати реакцію системи на отриманий сигнал (функція filter) для двох випадків:

- 2.1. Нульові початкові умови.

2.2. Випадкові початкові умови (скористатися функцією `rand`).

Для пунктів 2.1 та 2.2 побудувати графіки вхідного та вихідного сигналів в одному вікні. А також побудувати в окремому вікні перші 100 мс вхідного та вихідного сигналу. Зробити висновки щодо вигляду вихідного сигналу відносно вхідного (форма, амплітуда, спотворення, підсилення).

3. Визначити коефіцієнт передачі напруги ЛДС на частоті 10 Гц, а також різницю фаз між вихідним і вхідним сигналом, виходячи з дослідження графіків.

4. Сформувати два синусоїдальних сигнали частоти 7 та 70 Гц, тривалістю 1 с. Проілюструвати властивість адитивності системи, визначивши реакцію системи спочатку на кожний з сигналів окремо, а потім на суму цих сигналів. Проілюструвати властивість однорідності системи. Навести необхідні графіки.

5. Розрахувати за допомогою функції `filter` перші 30 відліків імпульсної характеристики системи, подавши на вхід системи одиничний імпульс (за нульових початкових умов). Побудувати графіки вхідного та вихідного сигналу.

6. Розрахувати 128 відліків імпульсної характеристики. Порівняти результати з результатами п. 5, побудувати графіки, зробити висновок щодо стійкості системи.

Моделювання роботи ЛДС з використанням рівняння згортки

7. Розрахувати реакцію системи на сигнал з п. 2.1 з використанням функції розрахунку згортки `conv`. Побудувати графіки вхідного та вихідного сигналу, аналогічні до п. 2.1 (з нульовими початковими умовами). Всі отримані в п. 7 результати порівняти з п. 2.1. Зробити висновки.

Дослідження роботи ЛДС в частотній області

8. Обчислити комплексну частотну характеристику системи для частоти дискретизації 256 Гц, використовуючи підстановку у формулу (1), побудувати графіки АЧХ та ФЧХ.

9. Обчислити комплексну частотну характеристику системи для частоти дискретизації 256 Гц, використовуючи перетворення Фур'є імпульсної характеристики (функція `fft` ()), побудувати графіки АЧХ та ФЧХ. Порівняти результати з отриманими у п. 8.

Зробити висновки щодо характеру зміни модуля коефіцієнта передачі системи. Зробити висновки щодо величини коефіцієнта передачі системи на різних частотах. Визначити проміжки частот, на яких система підсилює/ослаблює сигнал. Візуально та програмно пересвідчитись у правильності визначення коефіцієнту передачі і фазового зміщення, які розраховані в п. 3.

10. Розрахувати реакцію ЛДС на послідовність прямокутних імпульсів з наповненістю а) 30 %, б) 50 % (функція `squawave`). Побудувати графіки вхідного та вихідного сигналів, зробити висновки щодо спотворення вихідного сигналу відносно вхідного.

ЗМІСТ ЗВІТУ

1. Завдання (варіант).
2. Результати виконання 10 пунктів "порядку виконання" з графіками та висновками.
3. Загальні висновки до роботи.
4. Текст написаної програми.

ЗАВДАННЯ

Для всіх варіантів $a_1=1$!

| = | b_0 | b_1 | b_2 | b_3 | b_4 | b_5 | a_1 | a_2 | a_3 | a_4 | a_5 |
|----|-------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| 1 | 0,372 | 0,365 | -0,310 | 0 | 0,003 | -0,209 | 0,467 | -0,298 | 0,195 | -0,372 | 0,470 |
| 2 | 0,425 | 0,420 | -0,589 | 0,006 | -0,035 | -0,539 | 0,188 | 0,485 | 0,188 | 0,002 | 0,026 |
| 3 | 0,393 | 0,338 | 0,018 | -0,115 | -0,225 | -0,133 | 0,982 | 0,596 | -0,431 | 0 | 0,218 |
| 4 | 0,885 | -0,409 | 0,477 | -0,434 | -0,085 | 0,045 | 0,910 | 0,295 | -0,089 | 0,343 | 0,200 |
| 5 | 0,389 | 0,713 | -0,202 | -0,282 | 0,170 | -0,046 | 0,571 | 0,210 | -0,217 | -0,039 | -0,004 |
| 6 | 0,337 | 0,728 | 0,181 | -0,139 | 0 | 0,134 | 0,927 | -0,152 | -0,133 | 0,092 | -0,078 |
| 7 | 0,120 | -0,118 | -0,064 | 0,173 | 0,280 | 0,214 | 0,924 | 0,335 | -0,064 | 0,032 | -0,142 |
| 8 | 0,974 | -0,614 | -0,109 | 0,462 | -0,679 | 0,079 | 0,679 | 0,173 | -0,089 | -0,262 | 0,008 |
| 9 | 0,913 | 0,170 | -0,092 | 0,074 | 0 | 0,061 | 0,651 | 0,073 | -0,642 | 0,193 | -0,013 |
| 10 | 0,914 | 0,276 | -0,103 | -0,100 | 0,050 | -0,203 | 0,602 | 0,082 | 0,508 | 0,036 | 0,064 |
| 11 | 0,740 | -0,966 | 0,037 | 0,193 | 0,228 | 0,265 | 0,226 | 0,520 | -0,083 | -0,004 | -0,013 |
| 12 | 0,328 | -0,263 | 0,344 | 0,375 | -0,419 | 0,276 | 0,561 | -0,573 | -0,691 | 0,038 | 0 |
| 13 | 0,588 | -0,567 | 0,400 | -0,131 | 0,098 | 0,022 | 0,505 | -0,672 | 0,422 | -0,472 | 0,083 |
| 14 | 0,598 | 0,181 | 0,018 | 0 | 0,298 | -0,161 | 0,670 | 0,378 | -0,185 | -0,112 | 0,050 |
| 15 | 0,486 | 0,163 | -0,098 | -0,291 | -0,064 | 0,173 | 0,376 | -0,200 | -0,227 | -0,055 | -0,059 |
| 16 | 0,674 | 0,415 | 0,563 | -0,103 | 0,123 | 0,265 | 0,122 | 0,083 | -0,131 | 0,282 | -0,064 |
| 17 | 0,819 | -0,639 | -0,191 | -0,059 | -0,538 | 0,290 | 0,366 | 0 | 0,382 | -0,098 | 0,170 |
| 18 | 0,785 | -0,197 | 0,226 | -0,129 | -0,239 | 0,217 | 0,844 | 0,068 | 0,049 | -0,203 | 0,125 |
| 19 | 0,304 | 0,788 | 0 | 0,379 | -0,071 | 0,034 | 0,161 | -0,655 | -0,049 | -0,108 | -0,015 |
| 20 | 0,346 | -0,651 | 0,263 | -0,215 | 0,074 | 0,175 | 0,845 | -0,270 | -0,172 | 0,067 | -0,085 |
| 21 | 0,286 | 0,431 | -0,112 | -0,659 | -0,017 | 0,017 | 0,361 | 0,828 | 0,576 | -0,069 | 0,136 |
| 22 | 0,469 | 0,833 | -0,138 | 0,174 | -0,177 | 0,049 | 0,205 | 0,899 | 0,124 | 0,083 | 0,038 |
| 23 | 0,761 | -0,198 | -0,005 | 0 | -0,209 | -0,252 | 0,008 | 0,818 | 0,829 | 0,515 | 0,010 |
| 24 | 0,707 | 0,412 | 0,419 | -0,141 | -0,034 | -0,087 | -0,724 | 0,245 | 0,096 | -0,227 | -0,084 |
| 25 | 0,446 | -0,416 | -0,140 | 0,096 | -0,046 | 0,073 | -0,669 | 0,407 | 0,290 | -0,157 | -0,114 |
| 26 | 0,114 | -0,551 | -0,245 | 0 | 0,275 | -0,021 | 0,661 | 0 | -0,267 | 0,137 | 0,064 |
| 27 | 0,331 | -0,473 | -0,282 | 0,072 | -0,327 | -0,221 | 0,348 | -0,855 | 0,256 | 0,297 | -0,049 |
| 28 | 0,910 | 0,225 | 0 | -0,588 | -0,008 | 0,586 | 0,258 | 0,659 | 0,290 | 0,274 | -0,097 |
| 29 | 0,286 | 0,368 | -0,127 | -0,056 | -0,101 | -0,078 | -1,382 | 0,826 | -0,719 | -0,001 | 0,015 |
| 30 | 0,772 | -0,030 | -0,206 | 0,158 | 0 | -0,129 | 1,684 | -0,307 | -0,380 | -0,049 | 0,133 |

ПРИКЛАД ВИКОНАННЯ

1. Завдання

ЛДС задана, згідно з варіантом, своїми внутрішніми параметрами, тобто коефіцієнтами різницевого рівняння:

| № | b_0 | b_1 | b_2 | b_3 | b_4 | b_5 | a_1 | a_2 | a_3 | a_4 | a_5 |
|----|-------|-------|--------|-------|-------|-------|-------|--------|-------|-------|-------|
| 31 | 0,978 | 0,197 | -0,006 | 0,063 | 0,100 | 0,091 | 0,360 | -0,566 | 0,182 | 0 | 0,144 |

Отже, різницеве рівняння має вигляд:

$$\begin{aligned}
 y_n = & 0,978x^n + 0,197x^{n-1} - 0,006x^{n-2} + \\
 & + 0,063x^{n-3} + 0,1x^{n-4} + 0,091x^{n-5} - 0,36y^{n-1} + \\
 & + 0,566y^{n-2} - 0,182y^{n-3} - 0,144y^{(n-5)}
 \end{aligned}$$

Воно описує рекурсивну ЛДС 5 порядку.

В **z-області** це рівняння відповідатиме такій передавальній функції:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{n=0}^{N-1} b_n z^{-n}}{\sum_{m=1}^{M-1} a_m z^{-m}} =$$

$$= \frac{0,978 + 0,197z^{-1} - 0,006z^{-2} + 0,063z^{-3} + 0,1z^{-4} + 0,091z^{-5}}{1 + 0,36z^{-1} - 0,566z^{-2} + 0,182z^{-3} + 0,144z^{-5}}$$

В системі Scilab задання векторів коефіцієнтів матиме вигляд:

```
// коефіцієнти різницевої схеми
b=[0.978 0.197 -0.006 0.063 0.100 0.091];
a=[1 0.360 -0.566 0.182 0 0.144].
```

2. Визначення відгуку системи на гармонійний сигнал

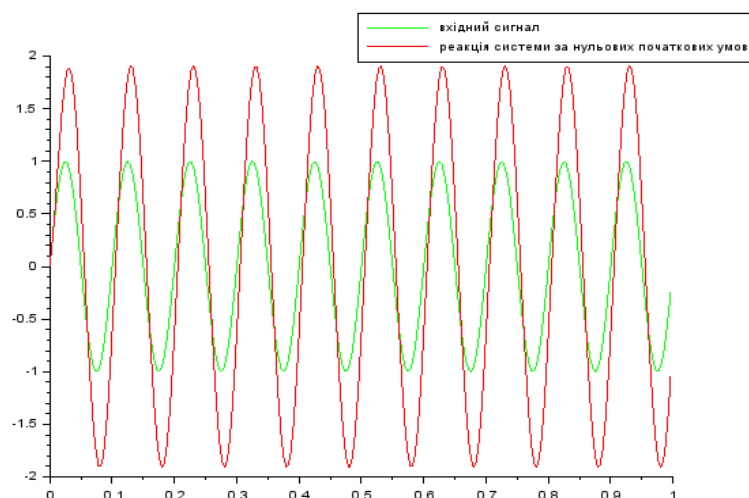
Формування синусоїдального сигналу частоти 10 Гц, тривалістю 1 с, амплітуди 1 В, дискретизованого з частотою 256 Гц в системі Scilab можна описати так:

```
A=1; // Амплітуда
f=10; // Частота сигналу, Гц
fd=256; // Частота дискретизації, Гц
T=1; // Тривалість сигналу, с
dt=1/fd; // Крок дискретизації, с
// x -відліки синусоїдального сигналу частоти 10 Гц тривалістю 1 сек. амплітуди 1 В,
// дискретизованого з частотою 256 Гц
t=0:dt:T-dt;
x=A*sin(2*%pi*f*t).
```

2.1. Для отримання реакції ЛДС на заданий сигнал за нульових початкових умов використовуємо вбудовану функцію `filter`. Вона передбачає такі параметри: `b`, `a`, – коефіцієнти різницевої схеми, `x` – вхідний сигнал. Тому реалізація буде така:
`y0 = filter(b, a, x);`

```
figure(1,'BackgroundColor',[1,1,1]);
plot2d(t,x,3),plot2d(t,y0,5)
legend('вхідний сигнал', 'реакція системи за нульових початкових умов', 'position',[T/2-
dt,max(x,y0)+0.5]).
```

Це приведе до виводу графіків вхідного і вихідного сигналів на спільній часовій осі:

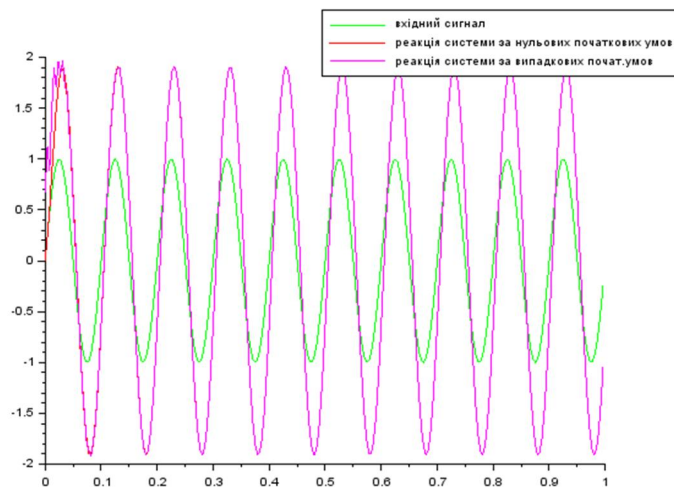


2.2. Для отримання реакції ЛДС на заданий сигнал за випадкових початкових умов використовуємо вбудовану функцію `filter` з четвертим параметром, який визначає початкові умови, та задаємо вектор випадкових значень, довжина якого відповідає кількості елементів у довшому з векторів коефіцієнтів:

```
l1=max (length (a),length (b))-1;
z0=rand (l1,1);
y1 =filter (b, a, x,z0); //y1 -реакція системи на вхідний сигнал за випадкових поч.умов
```

```
figure (2,'BackgroundColor',[1,1,1]);
plot2d (t,x,3),plot2d (t,y0,5), plot2d (t,y1,2);
legend ('вхідний сигнал', 'реакція системи за нульових початкових умов',...
'реакція системи за випадкових почат.умов', 'position',[T/2-dt,max (max (x,y0,y1))+0.5]).
```

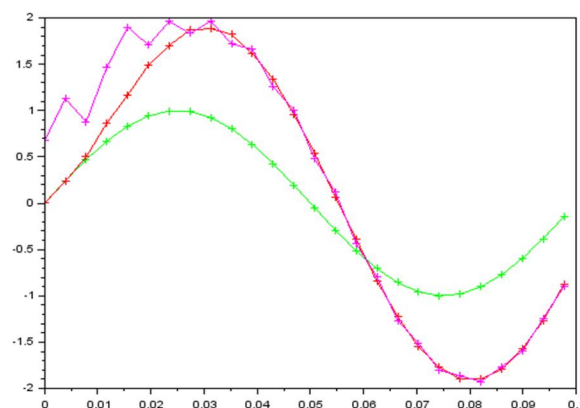
Це приведе до виводу графіків вхідного і вихідного сигналів на спільній часовій осі:



Для відображення перших 100 мс сигналів обмежимо часовий проміжок для виводу та визначимо довжину відповідних векторів сигналів:

```
T1=100*0.001;
t1=0:dt:T1-dt;
x1=x (1:length (t1));
y11=y1 (1:length (t1));
y01=y0 (1:length (t1));
figure (3,'BackgroundColor',[1,1,1]);
plot (t1,x1,'g+-'),plot (t1,y01,'r+-'),plot (t1,y11,'m+-').
```

Отримаємо:

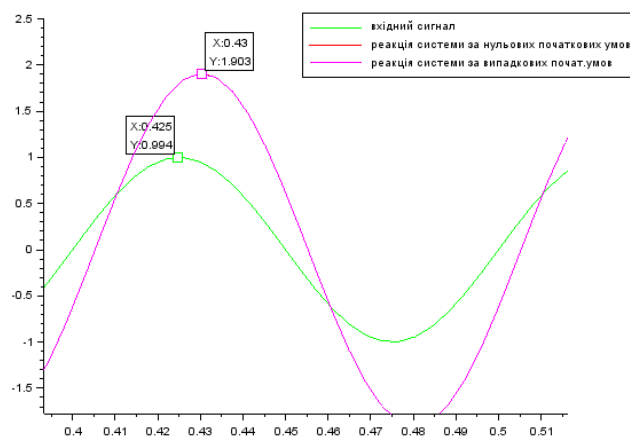


Отже, вхідний та вихідний сигнал – синусоїдальні, мають однаковий період, проте вихідний сигнал за ненульових початкових умов має спотворення на початку. Сигнали (з запізненням та без) різняться приблизно у перші 100 мс, далі вони співпадають. Хоча при кожному запуску програми це змінюється, оскільки генеруються випадкові початкові умови.

Вхідний сигнал посилюється. Різниця фаз майже непомітна (найімовірніше, вона становить лише декілька градусів).

3. Визначення параметрів ЛДС за графіками сигналів

Для визначення основних характеристик ЛДС, тобто коефіцієнта підсилення та різниці фаз можна скористатися редактором графіків. За його допомогою визначаємо проміжок, який містить піки вхідного і вихідного сигналів після перехідних процесів. Далі збільшуємо вибрану ділянку та застосуємо інструмент накладання даних. Отримуємо:



Отже, коефіцієнт підсилення на частоті 10 Гц становить:

$$K_{\omega}|_{2\pi*10} \approx \frac{1,903}{0,994} \approx 1,9$$

Різниця фаз : $\varphi_{\omega}|_{2\pi*10} \approx 0,43 - 0,425 \approx 0,005 \text{ с} \approx 5 \text{ мс}$

4. Дослідження властивостей ЛДС

Щоб дослідити властивість адитивності заданої системи, формуємо два гармонійні сигнали з частотами 7 Гц та 70 Гц:

`f1=7;`

`f2=70;`

`s1=0.1*A*sin(2*pi*f1*t);`

`s2=0.5*A*sin(2*pi*f2*t).`

Визначаємо реакцію системи на кожен з сигналів окремо та на суму цих сигналів:

`ys1=filter(b,a,s1);`*//реакція на s1*

`ys2=filter(b,a,s2);`*//реакція на s2*

`ys3=filter(b,a,(s1+s2));`*//реакція на (s1+s2).*

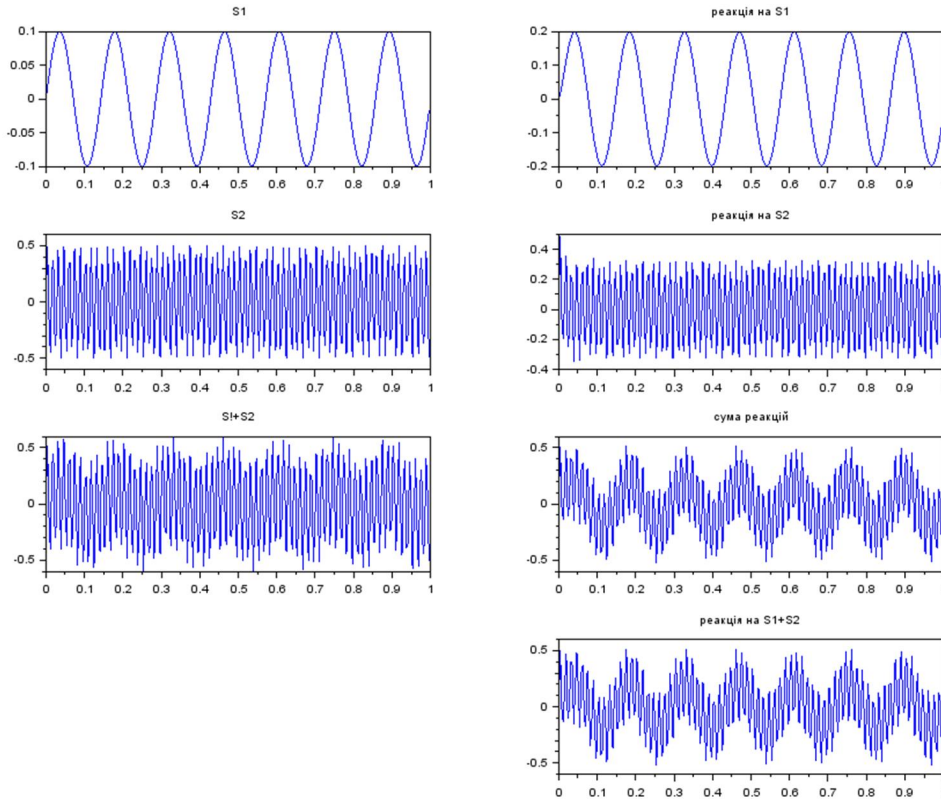
Для відображення і порівняння результатів використовуємо спільне графічне вікно:

`figure(4,'BackgroundColor',[1,1,1]);`

`subplot(4,2,1),plot(t,s1),title('S1');`

```
subplot (4,2,3),plot (t,s2),title ('S2');
subplot (4,2,5),plot (t, (s1+s2)),title ('S1+S2');
subplot (4,2,2),plot (t,ys1),title ('реакція на S1');
subplot (4,2,4),plot (t,ys2),title ('реакція на S2');
subplot (4,2,6),plot (t, (ys1+ys2)),title ('сума реакцій ');
subplot (4,2,8),plot (t,ys3),title ('реакція на S1+S2').
```

Це призводить до виводу таких графіків:



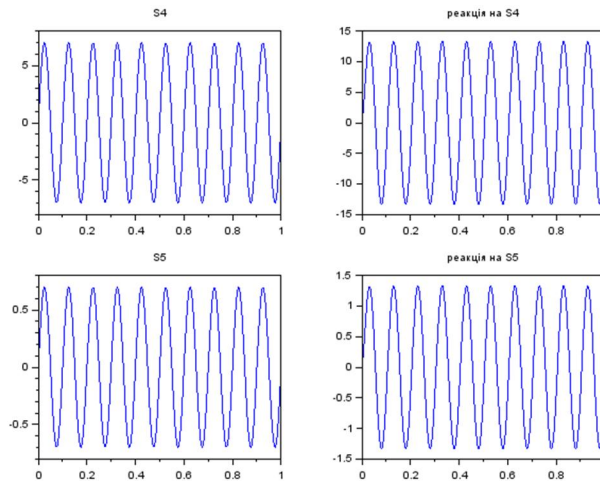
Вони дають змогу переконалися, що реакція системи на суму вхідних сигналів є такою ж як сума реакцій на кожен з цих сигналів окремо. Це доводить, що система володіє властивістю адитивності.

Для підтвердження властивості однорідності (гомогенності) ЛДС, створимо сигнали, що будуть послаблені та посилені відносно вхідного у 7 разів та обрахуємо реакцію на кожен з них:

```
s4=7*x;
s5=0.7*x;
ys4 =filter (b, a, s4);//реакція на s4
ys5 =filter (b, a, s5);//реакція на s5.
```

Відображення графіків:

```
figure (5,'BackgroundColor',[1,1,1])
subplot (2,2,1),plot (t,s4),title ('S4');
subplot (2,2,3),plot (t,s5),title ('S5');
subplot (2,2,2),plot (t,ys4),title ('реакція на S4');
subplot (2,2,4),plot (t,ys5),title ('реакція на S5').
```



З отриманих графіків бачимо, що посилення/послаблення вхідного сигналу приводить до посилення/послаблення відгуку системи у таку ж кількість разів. Отже властивість гомогенності підтверджено.

5. 6. Знаходження імпульсної характеристики ЛДС

Відомо, що імпульсна характеристика – це реакція системи на одиничний імпульс. Тому формуємо вхідну послідовність, що складається з одиниці та 29 нулів:

```
u0=[1,zeros(1,29)];
```

та знаходимо відгук системи:

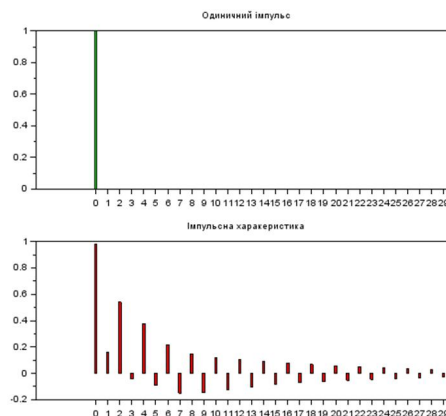
```
h= filter(b, a, u0);
```

відображаємо графіки, з використанням функції bar:

```
figure(6,'BackgroundColor',[1,1,1])
```

```
subplot(2,1,1),bar(0:29,u0,0.2,'green'),title('Одиничний імпульс');
```

```
subplot(2,1,2),bar(0:29,h,0.2,'red'),title('Імпульсна характеристика');
```



Як видно з рисунку, імпульсна характеристика має затухаючий характер, але 30 відліків недостатньо для її повного спадання. Тому знаходимо повну ІХ та відображаємо її графік.

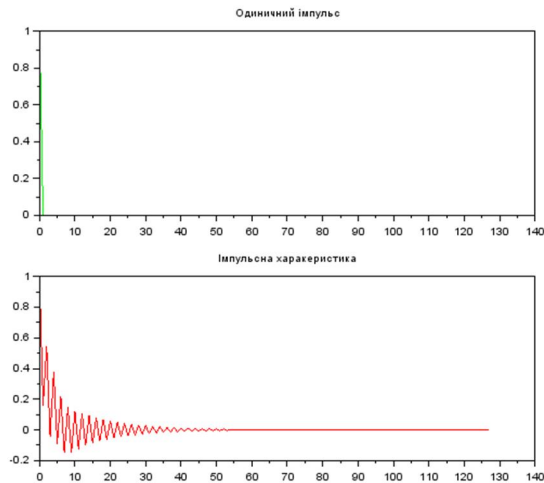
```
u0=[1,zeros(1,127)];
```

```
h= filter(b, a, u0);
```

```
figure(7,'BackgroundColor',[1,1,1])
```

```
subplot(2,1,1),plot(0:127,u0,'green'),title('Одиничний імпульс');
```

```
subplot(2,1,2),plot(0:127,h,'red'),title('Імпульсна характеристика');
```

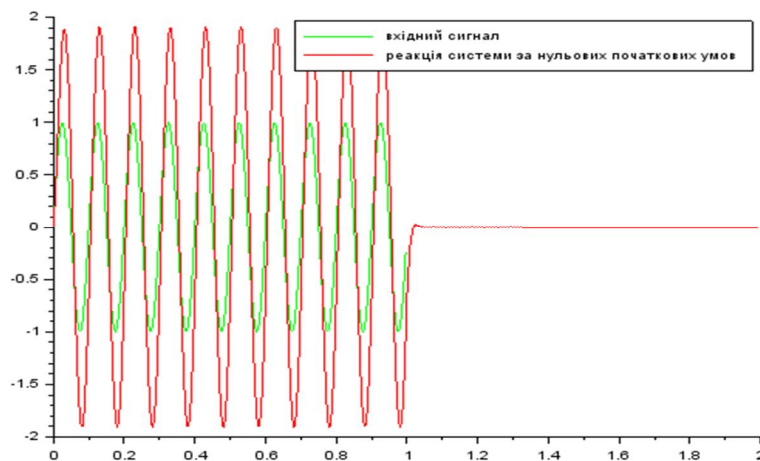


З отриманих результатів стає зрозумілим, що імпульсна характеристика затухає до 55 відліку і далі є нульовою. Це дозволяє стверджувати, що задана ЛДС є стійкою.

7. Моделювання роботи ЛДС з використанням рівняння згортки

Реакцію системи на довільний сигнал можна отримати шляхом згортки цього сигналу та імпульсної характеристики системи. Оскільки в попередніх пунктах IX характеристика вже знайдена, підставимо її значення у функцію conv, обрахувавши повну послідовність:

```
u0=[1,zeros(1,255)];
h=filter(b,a,u0);
ys1_1=conv(x,h);
t1=0:dt:dt*(length(ys1_1)-1);
figure(8,'BackgroundColor',[1,1,1])
plot2d(t,x,3),plot2d(t1,ys1_1,5);//вказати межі осей!!!
legend('вхідний сигнал','реакція системи за нульових початкових умов');
```



Результати виконання пунктів 2 і 7 співпадають, але в п. 7 ми бачимо, що програма продовжувала рахувати реакцію системи навіть тоді, коли сигнал уже закінчився. Щоб не було цього неіснуючого “хвоста”, треба доповнити вхідний сигнал нульовими відліками.

8. Дослідження роботи ЛДС в частотній області

Для знаходження комплексної частотної характеристики (КЧХ) системи вико ристовуємо формулу:

$$H(\hat{\omega}) = \frac{\sum_{i=0}^{N-1} b_i e^{-ji\hat{\omega}}}{1 + \sum_{k=1}^{M-1} a_k e^{-jk\hat{\omega}}}$$

тут $\hat{\omega}$ задає нормовану кругову частоту, тобто: $\hat{\omega} = \frac{\omega}{\omega_d}$.

Оскільки задана частота дискретизації становить 256 Гц, і є лінійною, то при написанні програми зводимо все до кругових частот.

Для структурованості загальної (моделюючої) програми реалізуємо обчислення КЧХ (за відомими коефіцієнтами різницевої схеми) окремою функцією, що буде міститися у файлі <freqz.sci>, і яку потрібно завантажувати у середовище до виконання основної програми.

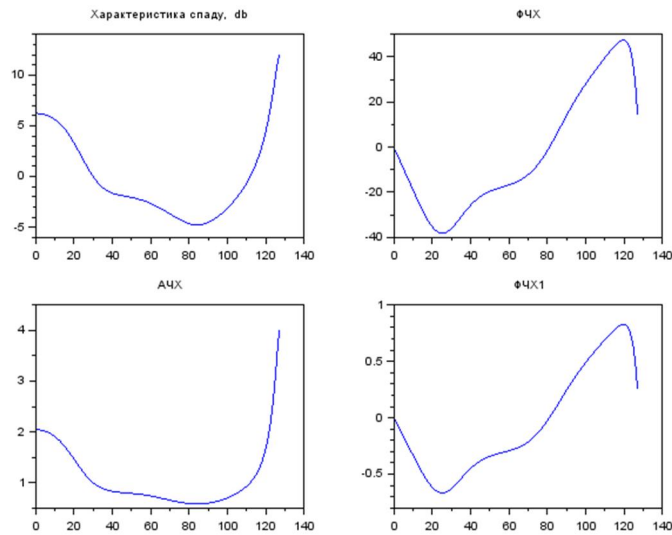
```
function [H, f]=freqz (b, a, Fs)
N=length (b);
M=length (a);
l=max (N,M);
N_M=N-M;
if N_M<0 then b=[b,zeros (1,abs (N_M))];
else
    a=[a,zeros (1,N_M)];
end

for f=0:1:Fs/2-1
    w_norm=2*%pi* (f/Fs);
    k=0:1:l-1;
    s1=sum (a.*exp (-k.*%i*w_norm));
    s2=sum (b.*exp (-k.*%i*w_norm));
    H (1,f+1)=s2./s1;
end
f=0:1:Fs/2-1;
endfunction.
```

Після її виклику в програмі застосуємо вбудовану функцію `dbphi (H)` для переходу до амплітудо частотної (АЧХ) та фазо-частотної характеристик (ФЧХ) системи. Далі – виведемо відповідні графіки.

//Знаходження частотної характеристики за коефіцієнтами різницевої схеми

```
[H,f]=freqz (b,a,fd);
[db, phi] = dbphi (H);
ASH=abs (H);
FSH=atan (imag (H)./real (H))
//вивід графіка АЧХ та ФЧХ в шкалі db та у абсолютних значення
figure (9)
subplot (2,2,1),plot (f,db),title (' Характеристика спаду, db');
subplot (2,2,3),plot (f,ASH),title (' АЧХ');
subplot (2,2,2),plot (f,phi),title (' ФЧХ');
subplot (2,2,4),plot (f,FSH),title (' ФЧХ1').
```

9. Дослідження роботи ЛДС в частотній області з використанням ІХ

Отримати КЧХ ЛДС можна, обравши перетворення Фур'є від її ІХ. Оскільки вона вже знайдена, то просто застосовуємо функцію Scilab'а – `fft (H)` та виводимо отримані результати графічно.

//Знаходження частотної характеристики як ШПФ від ІХ

```
u0=[1,zeros (1,255)];
```

```
h= filter (b, a, u0);
```

```
H1=fft (h);
```

```
[db1, phi1] = dbphi (H1);
```

```
ASH1=abs (H1);
```

```
FSH1=atan (imag (H1),real (H1));
```

```
//FSH1=atan (imag (H1)./real (H1))
```

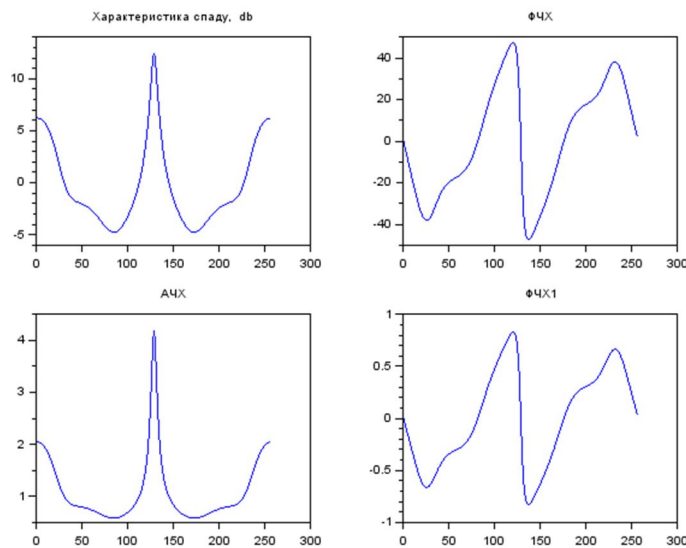
```
figure (10,'BackgroundColor',[1,1,1])
```

```
subplot (2,2,1),plot (db1),title (' Характеристика спаду, db');
```

```
subplot (2,2,3),plot (ASH1),title (' АЧХ');
```

```
subplot (2,2,2),plot (phi1),title (' ФЧХ');
```

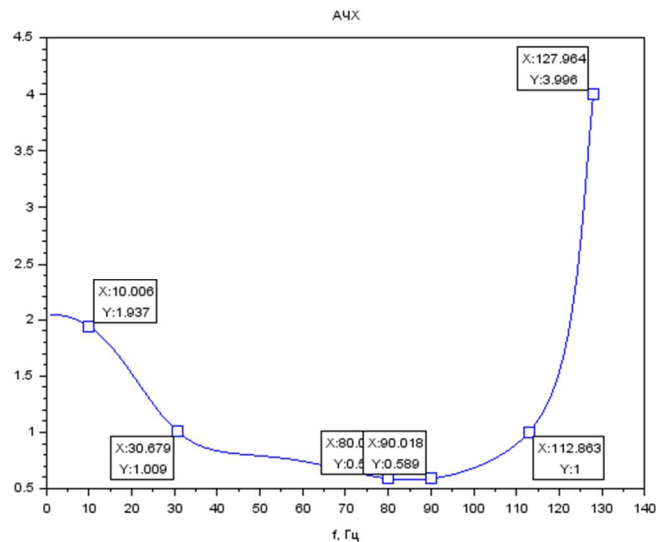
```
subplot (2,2,4),plot (FSH1),title (' ФЧХ1');
```



Порівнюючи результати з отриманими в п. 8, бачимо, що перші 128 відліків АЧХ та ФЧХ повністю співпадають. Що ж стосується другої половини, то вони формуються для т. зв. уявних частот за рахунок симетричності ДПФ, тому до розгляду не беруться.

Аналіз передавальних характеристик системи

Для наочного представлення, виведемо окремо графік АЧХ та позначимо на ньому дані в ключових точках (за допомогою графічного редактора):



Бачимо, що коефіцієнт передачі системи має нерівномірний характер: до частоти 80 Гц він зменшується, а від частоти 90 Гц – зростає. При цьому система підсилює вхідні сигнали в двох частотних діапазонах: від 0 Гц до 30 Гц та від 112 Гц до 128 Гц, і послаблює в діапазоні від 30 Гц до 112 Гц.

Коефіцієнт підсилення на частоті 10 Гц становить 1,9, що співпадає з розрахованим у п. 3. Для підтвердження застосуємо команди:

```
disp ('АЧХ на 10 Гц становить:', ASH (11));
```

```
disp ('ФЧХ на 10 Гц становить:', FSH (11));
```

в консолі побачимо:

```
"АЧХ на 10 Гц становить:" 1.9103312
```

```
"ФЧХ на 10 Гц становить:" -0.3319267 (рад).
```

10. Дослідження реакції ЛДС на послідовність прямокутних імпульсів

Для визначення відгуку системи створимо послідовність прямокутних імпульсів за допомогою функції `squarewave`, взявши за основу існуючий вектор часу та частоту імпульсів 3 Гц, і встановимо коефіцієнт наповненості 30 % (за допомогою другого параметру функції):

```
squarewave (2*%pi*t*3,30);
```

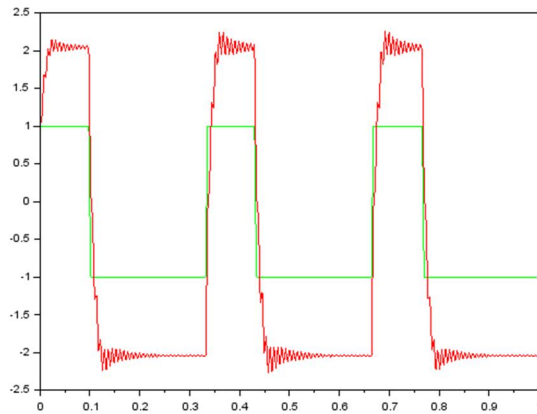
Знайдемо результат роботи системи та виведемо графіки на спільному рисунку:

```
s6=squarewave (2*%pi*3*t,30);
```

```
ys6=filter (b, a, s6);
```

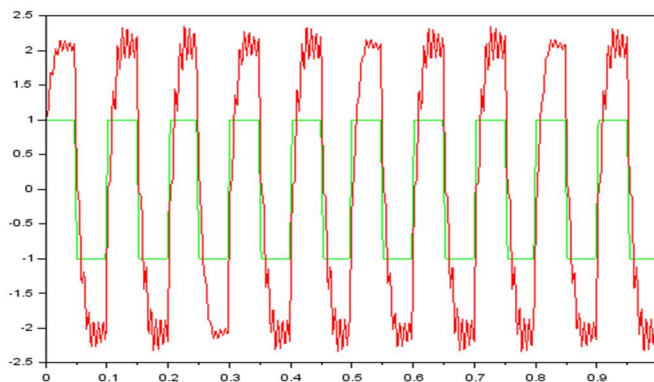
```
figure (11,'BackgroundColor',[1,1,1]);
```

```
plot (t,s6,'g',t,ys6,'r').
```



Для коефіцієнта наповненості 50 % оберемо частоту імпульсів 10Гц, а всі інші дії будуть аналогічні:

```
s7=squarewave (2*%pi*10*t,50);
ys7=filter (b, a, s6);
figure (12,'BackgroundColor',[1,1,1]);
plot (t,s7,'g',t,ys7,'r').
```



Спотворення вихідного сигналу в обох випадках відбувається при перепаді амплітуди. Спотворення стають помітнішими при збільшенні частоти вхідних імпульсів. В обох випадках вхідний сигнал посилюється приблизно у 2 рази.

Висновки

Під час лабораторної роботи була досліджена лінійна дискретна система, задана коефіцієнтами різницевої схеми (п. 1, 2). Знайдено її імпульсну та частотну характеристики (п. 3, 5, 6, 7). Шляхом аналізу було встановлено, що ця система однорідна та адитивна (п. 4). Вона є стійкою (п. 5, 6). Її частотна характеристика має неоднорідний характер (п. 8, 9). Визначено реакцію ЛДС на гармонійні сигнали (п. 1, 4) та на послідовність прямокутних імпульсів (п. 10).

Лабораторна робота № 6

ФІЛЬТРАЦІЯ ЗОБРАЖЕНЬ У ПРОСТОРОВІЙ ОБЛАСТІ

МЕТА РОБОТИ

Ознайомитися з методами та засобами фільтрації сигналів і зображень. Проілюструвати процес фільтрації зображення в просторовій області.

ТЕОРЕТИЧНІ ВІДОМОСТІ

Фільтрація зображень у часовій (просторовій) області. Просторова фільтрація зображень – це різновид обробки зображень у просторовій області, на відміну від класичного поняття “фільтрації”, яка виконується із залученням спектральних характеристик фільтрів та спектрів зображень. При просторовій фільтрації для розрахунку яскравості пікселя результуючого зображення використовуються яскравості пікселів у деякому okolí початкового зображення.

При обробці зображень фільтрація дозволяє виділяти границі об’єктів, розмивати зображення, добиватися ефекту тиснення, вирівнювати освітлення на фотознімках і т. д.

Серед статистичних методів та алгоритмів фільтрації сигналів і зображень найчастіше застосовуються екстремальна та медіанна фільтрація. Зокрема, медіанну фільтрацію доцільно використовувати, коли сигнал чи зображення являє собою адитивну суміш корисного сигналу та імпульсних завад (тобто, на зображенні спостерігаються білі чи чорні цятки, або “сніг”).

В просторовій області процес фільтрації сигналів описується рівнянням одновимірної згортки:

$$y(m) = \sum_{n=0}^{N-1} x(n) \cdot h(m-n),$$

де $y(m)$ – вихідний сигнал; $x(n)$ – вхідний сигнал; $h(m-n)$ – імпульсна характеристика фільтра.

Фільтрація зображень в часовій області зводиться до двовимірної лінійної згортки:

$$g(p, q) = \sum_{i=-M}^M \sum_{j=-M}^M h(i, j) x(p+i, q+j),$$

де $x(p, q)$ – вхідне зображення; $g(p, q)$ – фільтроване зображення; $h(i, j)$ – імпульсна характеристика фільтра (маска, що визначає вид фільтрації); $P \times Q$ – розмір зображення; $p = 0, 1, \dots, P-1$, $q = 0, 1, \dots, Q-1$; $M \times M$ – розмір вікна фільтрації (апертури); $i = -M, -M+1, \dots, M$, $j = -M, -M+1, M$.

Очевидно, що процес фільтрації – це послідовне обчислення згортки обраної маски з “вікном” (частиною) зображення. Елементи вікна розташовані в області точки для якої обчислюється згортка. Отже, маска фільтра, її ще називають *апертурою*, “пробігає” всі елементи зображення, утворюючи вихідне зображення.

Як правило, основну інформацію в зображеннях несуть контури об’єктів. При фільтрації зашумлених зображень степінь згладжування контурів об’єктів залежить від

розмірів апертури фільтра. При малих розмірах апертури краще зберігаються контрастні деталі зображення, але гірше пригашаються шуми. При великих розмірах апертури – навпаки. Таке протиріччя долається шляхом застосування фільтрів з адаптивним розміром апертур. В адаптивних фільтрах великі апертури застосовуються для монотонних ділянок зображення, а малі – поблизу неоднорідних деталей.

Маски для низькочастотної та високочастотної фільтрації визначають тип фільтрації. Низькочастотна фільтрація забезпечує згладжування шуму, тобто усунення високо-частотних складових. Вона досягається за рахунок використання масивів з додатними елементами. Прикладом таких масок можуть бути наступні масиви, що мають розмір 3×3 точки. Зауважимо, що для того, щоб процедура пригашення шуму не приводила до зміщення середньої яскравості зображення, ці масиви є нормованими.

$$h = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}; \quad h = \frac{1}{10} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix}; \quad h = \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}.$$

Для високочастотної фільтрації можна навести такі маски:

$$h = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}; \quad h = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{bmatrix}; \quad h = \begin{bmatrix} 1 & -2 & 1 \\ -2 & 5 & -2 \\ 1 & -2 & 1 \end{bmatrix}.$$

Особливістю таких апертур є те, що алгебраїчно сума елементів кожної з них дорівнює одиниці. Використання високочастотних масок призводить до виділення границь об'єктів, тому може бути використана маска Роберта, різницевий оператор Собеля, Кірша.

Широкого поширення набули методи контрастування (один із випадків високочастотної фільтрації), в яких використовується оператор Лапласа. На практиці він заміняється згорткою зображення з однією з масок:

$$h = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}; \quad h = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}; \quad h = \begin{bmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{bmatrix}.$$

Алгоритми лінійної фільтрації. Виконувати лінійну фільтрацію двовимірних масивів (зображень) можна за алгоритмами, що приводяться нижче. Перший з них реалізовує процес безпосередньо за формулою згортки, а другий – зменшує необхідний об'єм пам'яті, що може суттєво впливати на швидкість обробки при великих розмірах вхідного зображення.

В обох із запропонованих алгоритмів границі зображень обробляються без фільтрації, тобто, у вихідній матриці елементи крайніх рядків та стовпців співпадають з вхідними. Такий вид обробки найбільш прийнятний для тестових прикладів, хоча в реальних системах використовуються методи копіювання сусідів або інші, складніші алгоритми доповнення.

Алгоритм 1. Прямий

1. Перший і останній рядки (верхня і нижня границі зображення)

for (p = 0; p < M; p++)

for (q = 0; q < Q; q++)

{ g[p, q]=x[p, q]; g[P-1-p, q]=x[P-1-p, q]}.

2. Решта рядків

```
for (p = M; p < P - M; p++)
{
    2.1. Перші і останні M елементів рядка (ліва і права границі зображення)
    for (q=0; q < M; q++)
        { g[p, q]=x[p, q]; g[p, Q-1-q]=x[p, Q-1-q] };
    2.2. Решта елементів (основне перетворення)
    for (q=M; q < Q - M; q++)
        { Sum=0.0;
          for (i = - M; i <= M; i++)
              for (j = -M; j <= M; j++)
                  Sum = Sum +h [i,j] * x[i+p,j+q];
          g[p,q] = Sum;
        }
}
```

Алгоритм 2. Прямий, з мінімізацією необхідної пам'яті

1. Ініціалізація додаткової матриці $xtemp[p, q]$, розміру $(M+1) \times Q$

```
for (p = 0; p <= M; p++)
    for (q = 0; q < Q; q++)
        xtemp[p, q]=x[p, q].
```

2. Фільтрація

```
for (p = M; p < P - M; p++)
    {for (q=M; q < Q - M; q++)
      { Sum=0.0;
        for (j = -M; j <= M; j++)
            {for (i = 1; i <= M; i++)
              Sum = Sum +h [i,j] * x[i+p,j+q];
            for (i = -M; i < 1; i++)
              Sum = Sum +h[i,j] * xtemp[i+M,j+q];
            }
          g[p,q] = Sum;
        }
}
```

Модифікація матриці $xtemp$

```
for (i = 1; i < M; i++)
    for (q = 0; q < Q; q++)
        xtemp[i-1, q]= xtemp[i, q];
for (q = 0; q < Q; q++)
    xtemp[M, q]= x[p+1, q];
}
```

Для того, щоб легко працювати із зображеннями в системі Scilab, існує багато можливостей. Окрім вже вбудованих функцій, широкий вибір пропонує зовнішній модуль IPVC (Image Processing and Computer Vision Toolbox), який можна вільно завантажити за посиланням:

<https://atoms.scilab.org/toolboxes/IPCV>.

КОНТРОЛЬНІ ЗАПИТАННЯ

1. Що таке фільтрація сигналів?
2. Що таке апертура?
3. Для чого виконується фільтрація сигналів, зокрема зображень?
4. Які функції Scilab дозволяють зчитувати та відображати зображення?
5. Які функції Scilab забезпечують виконання фільтрації?

ПОРЯДОК ВИКОНАННЯ РОБОТИ

1. Сформувати вхідне зображення – **фотографія Вашого обличчя** у форматі, що підтримується Scilab (.jpg, .png, .bmp тощо).
2. Написати програму яка буде фільтрувати вхідне зображення згідно з заданим варіантом апертури для таких випадків:
 - 2.1. Формат вхідного зображення – RGB.
 - 2.2. Формат вхідного зображення – Gray.
 - 2.3. Формат вхідного зображення – Binary.
3. Відобразити результат фільтрації в новому вікні.
4. Додати до вхідного зображення шум та провести фільтрацію (для всіх форматів)
5. Відобразити результат фільтрації в новому вікні.
6. Зробити висновки щодо призначення фільтра.

ЗМІСТ ЗВІТУ

1. Завдання.
2. Текст написаної програми та пояснення алгоритму її роботи.
3. Результати роботи – вхідні та вихідні зображення.
4. Висновки.
5. Список використаної літератури.

ВАРІАНТИ ЗАВДАННЯ

| № | Імпульсна характеристика фільтра |
|---|--|
| 1 | 2 |
| 1 | $h=1/9*[1\ 1\ 1;1\ 1\ 1;1\ 1\ 1];$ |
| 2 | $h=1/10*[1\ 1\ 1;1\ 2\ 1;1\ 1\ 1];$ |
| 3 | $h=1/16*[1\ 2\ 1;2\ 4\ 2;1\ 2\ 1];$ |
| 4 | $h=[0\ -1\ 0;-1\ -5\ -1;0\ -1\ 0];$ |
| 5 | $h=[-1\ -1\ -1;-1\ 9\ -1;-1\ -1\ -1];$ |
| 6 | $h=[1\ -2\ 1;-2\ 5\ -2;1\ -2\ 1];$ |
| 7 | $h=[1\ 1\ 1;1\ -2\ 1;-1\ -1\ -1];$ |
| 8 | <p><i>Оператор Кірса:</i> $i=0:1:7;$ $g(n, m) = \max(1, \max(5S(i) - 3T(i))),$ де: $S(i)=A(i)+A(i+1)+A(i+2);$ $T(i)=A(i+3)+A(i+4)+A(i+5)+A(i+6)+A(i+7);$</p> |

| | |
|----|--|
| 1 | 2 |
| | а елементи А (і) є сусідами поточного пікселя зображення і визначаються так: <div style="text-align: center;"> A0 A1 A2 A7 x (n,m) A3 A6 A5 A4 </div> |
| 9 | h=[-1 1 1;-1 -2 1;-1 1 1]; |
| 10 | h=[-1 -1 1;-1 -2 1;1 1 1]; |
| 11 | h=[-1 -1 -1;1 -2 1; 1 1 1]; |
| 12 | h=[1 -1 -1;-1 -2 1; 1 1 1]; |
| 13 | <i>Оператор Юліца:</i> $g(n, m) = \frac{1}{4} \log \left(\frac{ x(n, m) ^4}{A1 * A3 * A5 * A7} \right)$ а елементи А (і) є сусідами поточного пікселя зображення і визначаються так: <div style="text-align: center;"> A0 A1 A2 A7 x (n,m) A3 A6 A5 A4 </div> |
| 14 | h=[1 1 1;1 -2 -1;1 -1 -1]; |
| 15 | h=[0 -1 0;-1 -4 -1;0 -1 0]; |
| 16 | h=[-1 -1 -1;-1 8 -1;-1 -1 -1]; |
| 17 | h=[1 -2 1;-2 4 -2;1 -2 1]; |
| 18 | h=[-1 2 -1;-1 2 -1;-1 2 -1]; |
| 19 | h=[-1 -1 -1;2 2 2;-1 -1 -1]; |
| 20 | h=[-1 -1 2;-1 2 -1;2 -1 -1]; |
| 21 | <i>Оператор Собеля.</i> $g(n, m) = \sqrt{X^2 + Y^2} ;$ де: X= (A2+2*A3+A4)- (A0+2*A7+A6); Y= (A0+2*A1+A2)- (A6+2*A5+A4); а елементи А (і) є сусідами поточного пікселя зображення і визначаються так: <div style="text-align: center;"> A0 A1 A2 A7 x (n,m) A3 A6 A5 A4; </div> |
| 22 | <i>Оператор Робертса.</i> $g(n, m) = \sqrt{(x(n, m) - x(n + 1, m + 1))^2 + (x(n, m + 1) - x(n + 1, m))^2}$ |
| 23 | $g(n, m) = x(n, m) - x(n + 1, m + 1) + x(n, m + 1) - x(n + 1, m) $ |
| 24 | h=[2 -1 -1;-1 2 -1;-1 -1 2] |
| 25 | h=[1 1 1;-1 -2 1; -1 -1 1]; |
| 26 | h=[1 1 -1;1 -2 -1; 1 1 -1]; |
| 27 | h=[-1 -1 -1;0 4 0;-1 -1 -1]; |
| 28 | h=[-1 -1 -1;2 4 2;-1 -1 -1]; |
| 29 | h=[0 0 0; 2 2 2; 0 0 0]; |
| 30 | h=[-1 -1 -1;1 16 1;-1 -1 -1]; |

ПРИКЛАД ВИКОНАННЯ

1. Завдання

Імпульсна характеристика фільтра: $h=[1 \ 1 \ 1; 1 \ 0 \ 1; -1 \ -1 \ -1]$.

2. Текст програми та пояснення до неї

| | |
|---|--|
| <code>clear</code> | |
| <code>clc</code> | |
| <code>close; close;</code> | |
| <code>rgb_Img = imread ('shrek2.jpg');</code> | <i>//завантаження вхідного зображення</i> |
| <code>G_Img = rgb2gray (rgb_Img)</code> | <i>//перетворення у сіро шкальний формат</i> |
| <code>B_Img = double (im2bw (G_Img,0.5));</code> | <i>//перетворення у бінарний формат</i> |
| <code>figure (1),</code> | |
| <code>subplot (3,2,1), imshow (rgb_Img), title ('Original Image');</code> | <i>// відображення вхідних зображень</i> |
| <code>subplot (3,2,3), imshow (G_Img), title ('Gray Image');</code> | |
| <code>subplot (3,2,5), imshow (B_Img), title ('Binary Image');</code> | |
| <code>h=[1 1 1;1 0 1;-1 -1 -1];..</code> | <i>//визначення маски фільтра</i> |
| <code>filtered_rgb_Img = imfilter (rgb_Img, h);</code> | <i>//фільтрація зображень</i> |
| <code>filtered_G_Img=imfilter (G_Img, h);</code> | |
| <code>filtered_B_Img=imfilter (B_Img, h);</code> | |
| <code>subplot (3,2,2), imshow (filtered_rgb_Img), title ('Filtered Original Image');</code> | <i>// відображення вихідних зображень</i> |
| <code>subplot (3,2,4), imshow (filtered_G_Img), title ('Filtered Gray Image');</code> | |
| <code>subplot (3,2,6), imshow (filtered_B_Img), title ('Filtered Binary Image');</code> | |
| <code>Noise_rgb_Img=imnoise (rgb_Img, 'gaussian');</code> | <i>//накладання Гаусівського шуму</i> |
| <code>Noise_G_Img=imnoise (G_Img, 'gaussian');</code> | |
| <code>Noise_B_Img=imnoise (B_Img, 'gaussian');</code> | |
| <code>filtered_Noise_rgb_Img = imfilter (Noise_rgb_Img, h);</code> | <i>//фільтрація зашумлених зображень</i> |
| <code>filtered_Noise_G_Img=imfilter (Noise_G_Img, h);</code> | |
| <code>filtered_Noise_B_Img=imfilter (Noise_B_Img, h);</code> | |
| <code>figure (2)</code> | |
| <code>subplot (3,2,1), imshow (Noise_rgb_Img), title ('Original Image with noise');</code> | <i>// відображення вхідних та фільтрованих зображень</i> |
| <code>subplot (3,2,3), imshow (Noise_G_Img), title ('Gray Image with noise');</code> | |
| <code>subplot (3,2,5), imshow (Noise_B_Img), title ('Binary Image with noise');</code> | |
| <code>subplot (3,2,2), imshow (filtered_rgb_Img), title ('Filtered RGB Image');</code> | |
| <code>subplot (3,2,4), imshow (filtered_G_Img), title ('Filtered Gray Image');</code> | |
| <code>subplot (3,2,6), imshow (filtered_B_Img), title ('Filtered Binary Image');</code> | |

Ця програма створена в системі Scolab з використанням модуля IPVC. Зокрема, застосовано такі його функції:

- imread – дозволяє читати зображення (шлях до якого вказаний як параметр) у змінну;
- imshow – дозволяє виводити зображення, в поточному графічному вікні;
- rgb2gray – перетворює зображення з RGB-подання у Gray;
- im2bw – перетворює зображення з Gray у Binary;
- imfilter – проводить обробку зображення фільтром, що задається імпульсною характеристикою;
- imnoise – додає шум (заваду) визначеного типу до вказаного зображення.

3. Результат виконання розробленої програми

Результат обробки різних типів вхідного зображення:



Результат обробки різних типів вхідного зображення з шумом.



Висновки

Під час виконання лабораторної роботи створено програму, що дозволяє проводити фільтрацію зображень, поданих у RGB, Gray та Binary форматах. Застосування заданого варіантом фільтра привело до висвітлення вхідних зображень. При обробці зображень з завадами відбулося згладжування шуму. Це особливо помітно на чорно-білому зображенні.

РЕКОМЕНДОВАНІ ДЖЕРЕЛА

1. Солонина А. И. Основы цифровой обработки сигналов. 2-е изд. : учеб. пособ. / А. И. Солонина, Д. А. Улахович, С. М. Арбузов, Е. Б. Соловьева. – БХВ-Петербург, 2005. – 753 с.
2. Айфичер Э. Цифровая обработка сигналов: практический подход, 2-е изд. / Эммануил Айфичер, Барри Джервис : пер. с англ. – М.: Изд. дом “Вильямс”, 2004. – 992 с.
3. Сергиенко А. Б. Цифровая обработка сигналов. – СПб.: Питер, 2002. – 608 с.
4. Алексеев Е. Р. SCILAB: Решение инженерных и математических задач / Е. Р. Алексеев, О. В. Чеснокова. – 2008. – 260 с.
5. Ваврук Є. Я. Алгоритми та засоби обробки сигналів : навч. посіб. / Є. Я. Ваврук, О. Л. Лашко, Р. Б. Попович. – Львів: Сполом, 2021. – 240 с.
6. Павлова М. И. Руководство по работе с пакетом SCILAB / М. И. Павлова. – 2012. – 209 с.
7. SCILAB documentation – <https://www.scilab.org/>.
8. SCILAB documentation – Основи роботи в SCILAB. Методичний довідник для роботи в системі SciLab в рамках вивчення навчальної дисципліни “Методи та засоби опрацювання сигналів і зображень” – файл Scilab_book.doc.
9. <https://atoms.scilab.org/toolboxes/PCV>.

ЗМІСТ

| | |
|--|-----------|
| Вступ | 3 |
| Лабораторна робота № 1 | |
| Виконання елементарних операцій над сигналами в системі scilab..... | 4 |
| Лабораторна робота № 2 | |
| Дискретизація і квантування сигналів | 12 |
| Лабораторна робота № 3 | |
| Згортка та кореляція сигналів | 18 |
| Лабораторна робота № 4 | |
| Визначення і дослідження спектральних характеристик сигналів..... | 28 |
| Лабораторна робота № 5 | |
| Дослідження лінійних дискретних систем в часовій та частотній області | 38 |
| Лабораторна робота № 6 | |
| Фільтрація зображень в просторовій області..... | 52 |
| Рекомендовані джерела | 59 |

ЕЛЕКТРОННЕ НАВЧАЛЬНЕ ВИДАННЯ

ЛАБОРАТОРНИЙ ПРАКТИКУМ
з дисципліни
ТЕХНОЛОГІЇ ОПРАЦЮВАННЯ СИГНАЛІВ
ТА ЗОБРАЖЕНЬ

Навчальний посібник

Редактор *Голько Софія*
Комп'ютерне верстання *Наталії Максимюк*

Режим доступу: <http://eom.lp.edu.ua/textbooks/np-tosz.pdf>

Видавець і виготівник: Видавництво Львівської політехніки
Свідоцтво суб'єкта видавничої справи и ДК № 4459 від 27.12.2012 р.

вул. Ф. Колесси, 4, Львів, 79013
тел. +380 32 2584103, факс +380 32 2584101
vlp.com.ua, ел. пошта: vmr@vlp.com.ua

