МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

МЕТОДИЧНІ ВКАЗІВКИ

до лабораторних робіт з дисципліни
«**Оптимальні системи керування**»
для здобувачів другого (магістерського) рівня вищої освіти
спеціальності 123 Комп’ютерна інженерія
(освітня програма «Спеціалізовані комп’ютерні системи»)
усіх форм навчання

2024

Методичні вказівки до лабораторних робіт з дисципліни «Оптимальні системи керування» для здобувачів другого (магістерського) рівня вищої освіти спеціальності 123 Комп’ютерна інженерія (освітня програма «Спеціалізовані комп’ютерні системи») усіх форм навчання / Укл.: М. В. Єфименко – Запоріжжя: НУ «Запорізька Політехніка», 2024. – 71 с.

Укладачі: М. В. Єфименко, доцент, д.т.н.

Рецензент: Р. К. Кудерметов, доцент, к.т.н.

Відповідальний
за випуск: Т.В. Голуб, доцент, к.т.н.

Затверджено:
на засіданні кафедри
«Комп’ютерні системи та мережі»
Протокол №1 від 06 серпня 2024 р.

Рекомендовано до видання:
НМК факультету КНТ
Протокол №1 від 04 вересня 2024 р.

ЗМІСТ

с.
Вступ ... 4
Лабораторна робота № 1 ... 6
1.1 Теоретичні відомості ... 6
1.2 Порядок виконання роботи ... 14
1.3 Зміст звіту ... 15
1.4 Контрольні питання ... 15
Лабораторна робота № 2 ... 19
2.1 Послідовні коригуючі пристрої (ПІД регулятори) ... 19
2.2 Визначення коефіцієнтів ПІД регулятора ... 20
2.3 Порядок виконання роботи ... 23
2.4 Зміст звіту ... 25
2.5 Контрольні питання ... 25
Лабораторна робота № 3 ... 26
3.1 Теоретичні відомості ... 26
3.2 Порядок виконання роботи ... 33
3.3 Зміст звіту ... 34
3.4 Контрольні питання ... 34
Лабораторна робота № 4 ... 35
4.1 Теоретичні відомості ... 35
4.2 Аналіз керованості та спостережливості системи ... 39
4.3 Порядок виконання роботи ... 45
4.4 Зміст звіту ... 46
4.5 Контрольні питання ... 46
Лабораторна робота №5 ... 47
5.1 Теоретичні відомості ... 47
5.2 Порядок виконання роботи ... 51
5.3 Зміст звіту ... 52
5.4 Контрольні питання ... 52
Лабораторна робота № 6 ... 54
6.1 Теоретичні відомості ... 54
6.2 Порядок виконання роботи ... 57
6.3 Зміст звіту ... 57
6.4 Контрольні питання ... 57
Лабораторна робота № 7 ... 59
7.1 Теоретичні відомості ... 59
7.2 Порядок виконання роботи ... 69
7.3 Зміст звіту ... 69
7.4 Контрольні питання ... 69
Перелік джерел посилань ... 71

ВСТУП

Теорія автоматичного керування (ТАК) – наука про методи проектування законів управління будь-якими об’єктами, що допускають реалізацію за допомогою технічних засобів автоматики.

Система автоматичного керування (САК) – це сукупність взаємозв’язаних об’єктів, в якій протягом тривалого часу потрібним чином змінюються (або підтримуються постійними) задані фізичні параметри в визначеному фізичному процесі без втручання людини.

Основні задачі ТАК наступні:

– аналіз керованості системи: аналізується можливість побудови керування, яке забезпечує бажану поведінку системи;

– аналіз спостережливості системи: аналізується можливість визначення за вимірюваними вихідними сигналами значень відсутніх вихідних сигналів;

– синтез законів керування: виконується розробка законів керування, проектування пристроїв керування, обирається функціональна і структурна схеми пристрою керування, розраховуються його параметри;

– аналіз якісних показників процесів керування: перевіряється виконання вимог, що пред’являються до САК, аналізується поведінка системи, оцінюються її точностні і динамічні характеристики.

Всі системи автоматичного керування поділяються на розімкнуті і замкнуті САК [1, 2, 3].

У розімкнутих системах процес роботи системи не залежить безпосередньо від результату її впливу на керований об’єкт.

У замкнутих системах вихід системи замикається з входом таким чином, щоб вимірювані контрольовані величини, які характеризують певний процес, були б джерелом впливу на систему, причому величина цього впливу залежала б від того, наскільки спостерігаєма величина відрізняється від заданої.

У теорії автоматичного керування розглядаються тільки замкнуті автоматичні системи.

Основні компоненти САК:

– об’єкт керування (ОК), яким система повинна керувати;

– датчики, які забезпечують отримання інформації про об’єкт;

– пристрій керування – головний компонент системи керування, який порівнює задані і вимірювані на об’єкті дані та формує вхідні змінні, що поступають на об’єкт.

Задані дані подаються в пристрій керування. Вихідні дані з об’єкту керування вимірюються за допомогою датчиків і вимірювані дані поступають в пристрій керування. Пристрій керування формує вхідні дані, які поступають на вхід виконуючих органів ОК.

САК вирішують різні задачі керування, які можна умовно розділити на такі:

– задачі слідкування, і як окремий випадок задачі стабілізації;

– задачі термінального керування.

Система стабілізації – це САК, в якій значення заданих даних є постійними протягом тривалого часу, а на виході об’єкта ці значення підтримуються з заданою точністю. Приклад: система опалювання будинку, яка всередині будинку підтримує постійну задану температуру при значних коливаннях температури навколишнього середовища [1, 2, 3].

Система слідкування – це САК, в якій значення заданих даних змінюються в часі, а на виході об’єкта відслідковується зміни цих значень з заданою точністю. Приклад: радарна установка, яка відслідкує переміщення об’єктів, що рухаються.

Термінальна система – це САК, в якій вихідні змінні повинні досягнути значення заданої змінної за визначений час. Приклад: система керування супутником, яка розвертає супутник з поточного кутового положення в момент часу t_0 до потрібного кутового положення в момент часу t_1 [4, 5, 6].

Для дослідження динаміки САК уся система розбивається на окремі ланки.

Ланка системи – це технічний пристрій любої фізичної природи, конструкції та призначення, що описується визначеним диференційним рівнянням.

Рівняння ланки повинно бути складено таким чином, щоб воно відображало залежність (в динамічному процесі) між теми величинами, які в схемі системи вказані на виході та вході даної ланки. Для аналізу систем диференційних рівнянь часто використовуються передатні функції.

Передатна функція ланки – це відношення зображень Лапласу вихідної та вхідної величин при нульових початкових умовах.

ЛАБОРАТОРНА РОБОТА № 1
Вивчення основ роботи з пакетом MATLAB для розрахунків та моделювання САК

Мета роботи – ознайомитися з системою для математичних та інженерних розрахунків MATLAB та вивчити основні принципи роботи з цією системою.

1.1 Теоретичні відомості

В теорії автоматичного керування важливе місце приділяється аналізу та синтезу систем автоматичного керування. Одним з програмних продуктів, призначених для моделювання цих систем, є система MATLAB, яка реалізує широкий спектр математичних методів, засобів візуалізації та допоміжних засобів.

1.1.1 Робочий простір системи MATLAB і її командне вікно

Після запуску програми MATLAB на екрані з'являється командне вікно системи MATLAB, що містить меню, лінійку з кнопками і клієнтську частину із знаком запрошення >>.

Після знаку >> можна вводити з клавіатури числа, імена змінних і знаки операцій, що складають деякий вираз. Після натиснення клавіші Enter вираз обчислюється і результат виводиться на екран (рис. 1.1).

Після обчислення виразу знизу вікна з'являється вільний рядок для введення нових даних і знак >>.

Всі значення змінних, обчислені протягом поточного сеансу роботи, зберігаються в спеціальній області пам'яті, робочому просторі системи MATLAB (Workspace).

Вся видима інформація у вікні системи MATLAB розташовується в двох зонах: перегляду і редагування.

В зоні перегляду можна переглядати будь-яку інформацію, виділяти її та копіювати, але не можна виправляти.

Зона редагування займає один рядок командного вікна з знаком >> та називається рядком введення, який може займати декілька фізичних рядків. Але редагувати можна тільки на останній рядок.

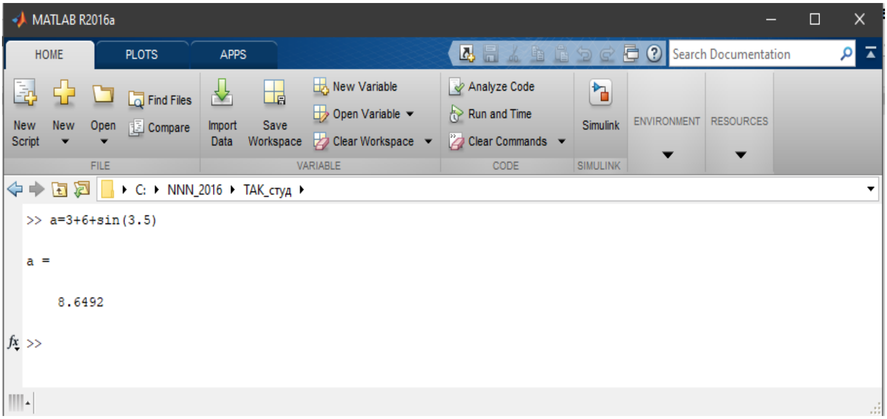


Рисунок 1.1 – Командне вікно системи MATLAB

Команда `clear ім'я1 ім'я2` видаляє задані змінні, а команда `clear` видаляє відразу всі змінні.

Команда `clc` очищує видимий зміст командного вікна, але значення всіх обчислених змінних при цьому зберігаються. Їх можна продивитися, якщо задати ім'я змінної і натиснути Enter.

Командою `who` можна перевірити, які змінні залишилися в робочому просторі.

Для збереження змісту робочого простору потрібно виконати команду меню Save Workspace, задавши каталог на диску та ім'я файлу з розширенням `.mat`, або набрати в командному вікні команду: `save шлях_до_файлу\ім'я_MAT-файлу`. Такі файли називаються MAT-файлами.

Поточний каталог можна визначити за допомогою команди `cd`, а змінити його можна командою `cd шлях_до_нового_каталогу`.

На будь-яку команду системи MATLAB можна отримати довідку, виконавши команду: `help ім'я_команди`.

Система MATLAB підтримує ще пакетний режим роботи, в якому можна розробляти програми, що складаються з послідовності команд користувача та зберігаються на диску у вигляді окремого файлу з розширенням `.m`.

Файли, які містять команди мови MATLAB (М-мови), називаються М-файлами.

Створювати М-файл можна використовуючи спеціальний редактор М-файлів, що входить до складу MATLAB. При цьому всі файли проходять синтаксичний контроль.

Є два типи М-файлів: файли-сценарії і файли-функції.

1.1.2 Створення файл-сценарію

Файл-сценарій, або Script-файл, є найпростішою програмою з записом серії команд без параметрів. Файл-сценарій не має вхідних і вихідних аргументів, використовує тільки глобальні змінні з робочої області, в процесі виконання не компілюється.

Файл-сценарій має наступну структуру: основний коментар, додатковий коментар, тіло файлу з будь-якими виразами. Основний коментар – це перший рядок текстових коментарів, що починається зі знаку `%` в першій позиції, а додатковий – подальші рядки зі знаком `%`. При виконанні команди `help ім'я_файлу` основний коментар виводиться в командному вікні.

Для створення М-файлу потрібно виконати команду New|Script головної панелі інструментів MATLAB (рис. 1.2). З'явиться нове вікно – текстовий редактор з готовим для редагування порожнім документом.

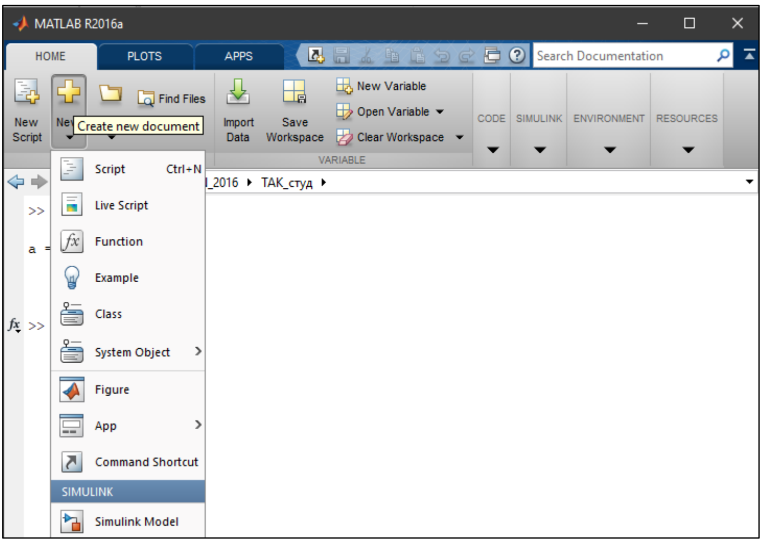


Рисунок 1.2 – Створення М-файлу

Після набору тексту файл-сценарію в текстовому редакторі і його відлагодження необхідно записати М-файл, для чого вибрати пункт меню Save|Save As ім'я_М-файлу. Для виконання М-файла-сценарію можна скористатися командою меню Run, або викликати М-файл в командному вікні, вказавши тільки ім'я цього файлу без розширення.

Примітка: ім'я М-файлу не повинно починатися з цифри.

Розглянемо наступний файл-сценарій для дослідження динамічних характеристик інтегруючої ланки, яка описується рівнянням $x_2' = 3x_1$, та має передатну функцією $W1(s) = 3/s$. В текстовому редакторі набираємо текст М-файлу (ліст. 1.1). Перший рядок – це основний коментар, інші – тіло файлу та коментарі.

Лістинг 1.1 – Текст М-файлу

```
%Дослідження динамічних характеристик ланки
clc;      % Очистка змісту командного вікна
f1=[0 3]; %Коеф-ти чисельника передат. ф-ції
f2=[1 0]; %Коеф-ти знаменника передат. ф-ції
W1=tf(f1,f2) %Формування передат. ф-ції ланки
%Побудова перехідної характеристики ланки
subplot(2,1,1); step(W1); grid on;
%Побудова імпульсної характеристики ланки
subplot(2,1,2);impulse(W1); grid on;
```

Функція `tf(f1,f2)` формує передатну функцію ланки, де `f1, f2` – коефіцієнти чисельника і знаменника передатної функції ланки.

Функція `subplot(m,n,p)` виводить в одному вікні відразу кілька графіків, тобто розбиває вікно на `m x n` підвікон, де `m` – кількість підвікон по вертикалі, `n` – кількість підвікон по горизонталі, `p` – номер підвікна (підвікна підраховуються послідовно за рядками).

Команда `grid on` використовується для виведення сітки на графіку. Примітка: якщо після виразу немає коми, то значення виразу буде виведено на екран.

Після набору тексту файл записуємо (наприклад, `lab1_1.m`) та запукаємо на виконання, скориставшись командою меню Run (рис.1.3).

В результаті отримуємо такі графіки (рис.1.4).

Для запису графіків обирається пункт File|Save As, де задається ім'я файлу з потрібним розширенням (наприклад, `.jpg`).

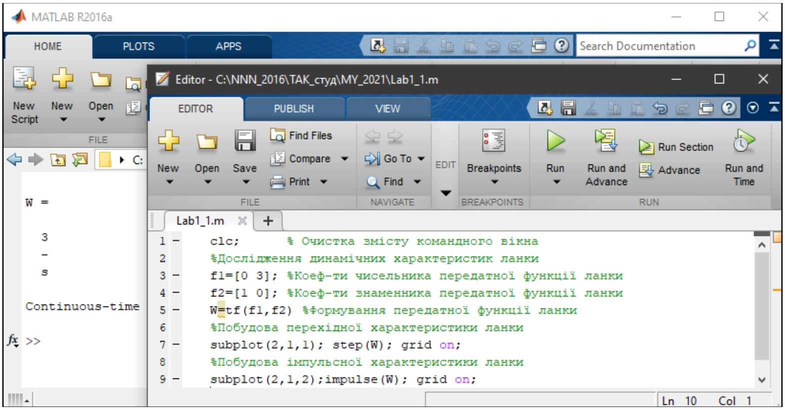


Рисунок 1.3 – М-файл з програмою

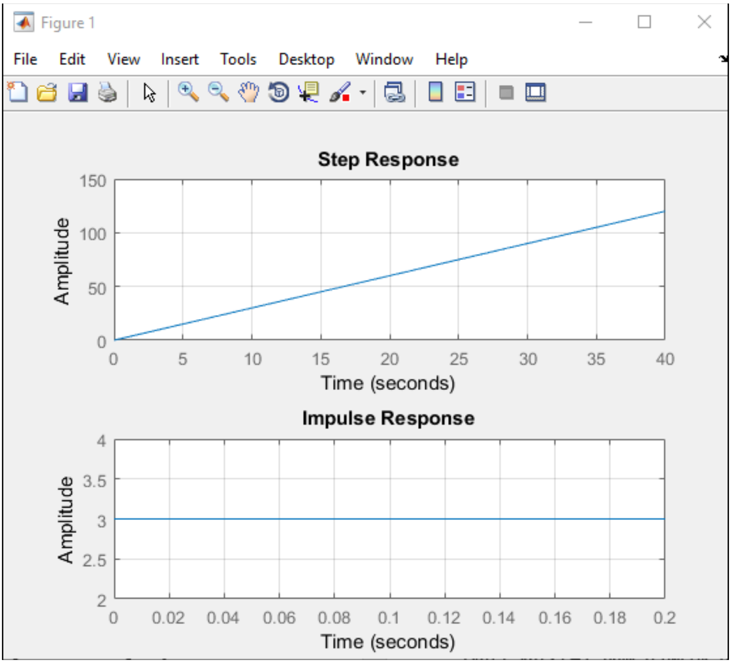


Рисунок 1.4 – Результати роботи

1.1.3 Створення М-функцій

М-файл-функція є самостійним програмним модулем, який спілкується з іншими модулями через свої вхідні і вихідні параметри. Визначення М-файл-функції починається зі слова `function`. Файл-функція може використовувати локальні змінні, які ніяк не впливають на значення змінних за межами функції. При виявленні в процесі виконання програми файл-функція завжди компілюється, а тільки потім виконується.

Файл-функція з декількома вихідними параметрами має наступний запис (літ. 1.2), де `список_параметрів` – імена вхідних параметрів, записаних через кому в круглих дужках, `f_name` – ім'я функції, `var1, var2,…` – імена вихідних змінних, записаних через кому в квадратних дужках, `Основний коментар` – перший рядок коментарів, що виводиться на екран, якщо задати в командному вікні `help f_name`.

Лістинг 1.2 – Файл-функція з декількома вихідними параметрами

```
function[var1,var2,…]=f_name(список_параметрів)
%Основний коментар
%Допоміжний коментар
Тіло файлу з будь-якими виразами
var1=вираз
var2=вираз
```

За допомогою запису `var1=вираз, var2=вираз` файл-функція повертає результати обчислень. Якщо файл-функція має один вихідний параметр, то його ім'я не береться в квадратні дужки. Така функція може використовуватися в математичних виразах в записі `f_name(список_параметрів)`.

Файл-функцію з декількома вихідними параметрами не можна використовувати безпосередньо в математичних виразах. Така функція спочатку записується як окремий елемент програми у вигляді: `[var1,var2]=f_name(список_параметрів)`, а потім вихідні змінні `var1,var2` можна використовувати в подальших виразах.

Файл-функцію створюють аналогічно файлу-сценарію. Набирають текст М-функції, відлагоджують її, створюють М-файл за допомогою пункту меню Save|Save As. Примітка: ім'я М-файлу з М-функцією повинно співпадати з ім'ям самої функції.

Викликати М-файл-функцію можна з командного вікна, або з М-файлу-сценарію, вказавши ім’я цього файлу з файл-функцією.

Розглянемо приклад М-функції для дослідження динамічних характеристик ланки (перехідної та імпульсної характеристик) з заданими параметрами (ліст. 1.3): `St_im` – ім’я М-функції, `n1, n2` – вектори-рядки коефіцієнтів чисельника та знаменника передатної функції ланки, `Tk` – кінцевий час моделювання, `h` – крок моделювання. Вихідні параметри: `y1, y2` – вектори-рядки обчислених значень перехідної та імпульсної функцій ланки, `t` – масив точок часу.

Лістинг 1.3 – Дослідження динамічних характеристик ланки

```
function [y1,y2,t]=St_im(n1,n2,Tk,h);
%Дослідження динамічних характеристик ланки
W=tf(n1,n2)  %Формування передатної ф-ції ланки
i=1:1:fix(Tk/h)+1;  %Завдання кіл-ті точок
T(i,1)=(i-1)*h;     %Формування масиву часу
[s1,T]=step(W,T); %Обчислення перехідної ф-ції
[s2,T]=impulse(W,T);%Обчислення імпульсної ф-ції
y1=s1; y2=s2; t=T;  %Вихідні параметри
```

Записуємо М-файл-функцію з ім’ям `St_im.m`, яке співпадає з ім’ям самої функції `St_im` (рис.1.5).

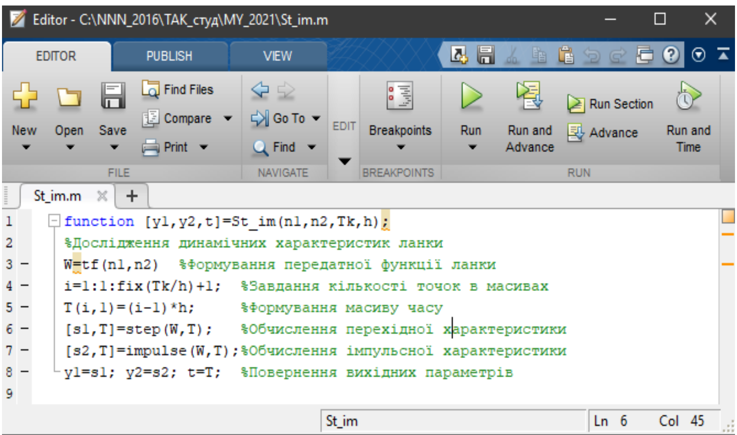


Рисунок 1.5 – М-файл з М-файл-функцією `St_im`

Далі розробляється файл-сценарій для дослідження динамічних характеристик інтегруючої ланки з наступними вхідними параметрами, в якому викликається М-функція (літ. 1.4).

Лістинг 1.4 – М-функція для дослідження інтегруючої ланки

```
clc;
Tk=25; h_int=0.01; % Tk=a0/a1*5 -час моделювання
f1=[0 3];   f2=[1 0];
[s1,s2,T]=St_im(f1,f2,Tk,h_int);
subplot(3,1,1);
set(plot(T,s1,'-b',T,s2,'-.r'),'LineWidth',1.5);
grid on; legend('h1','g1');
f1=[0 20];   f2=[5 1];
[s1,s2,T]=St_im(f1,f2,Tk,h_int);
subplot(3,1,2);
set(plot(T,s1,'-b',T,s2,'-.r'),'LineWidth',1.5);
grid on; legend('h1','g1');
f1=[20 0];   f2=[5 1];
[s1,s2,T]=St_im(f1,f2,Tk,h_int);
subplot(3,1,3);
set(plot(T,s1,'-b',T,s2,'-.r'),'LineWidth',1.5);
grid on; legend('h1','g1');
```

Функція `plot(...)` виводить на екран графіки функцій `s1(T)` та `s2(T)` з завданням вигляду та кольору лінії (рис.1.6). Лінія може бути: безперервна `-` , штрих-пунктирна `-.` , подвійна пунктирна `:`, штрихова `--`, а колір лінії: `k` – чорний, `g` – зелений та ін. Функція `set(...)` задає товщину ліній 1.5 пт.

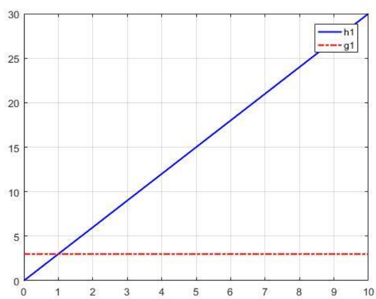


Рисунок 1.6 – Результати виконання М-файлу

1.2 Порядок виконання роботи

1. Ознайомтесь з командним вікном системи MATLAB. Обчисліть наступні математичні вирази та збережіть у файлі з розширенням `.mat`: 2+3, x=pi/4, z=2.301*sin(x), w=4+exp(3)/5, y=8+z, f=sqrt(y)/2.

2. Очистіть робочий простір, а потім зітріть видимий зміст робочого простору, та завантажте зміст робочого простору з файлу.

3. Розрахуйте параметри для передаточних функцій заданих рівнянь, коефіцієнти для яких виберіть з таблиці 1.1.

Рівняння з передаточними функціями приведені нижче:

$$a_0 x_2' + a_1 x_2 = b_0 x_1, \quad W2(s) = \frac{X_2}{X_1} = \frac{b_0}{a_0 s + a_1} = \frac{b_0/a_1}{a_0/a_1 s + 1};$$
$$a_0 x_2' + a_1 x_2 = b_0 x_1', \quad W3(s) = \frac{X_2}{X_1} = \frac{b_0 s}{a_0 s + a_1} = \frac{b_0/a_1 s}{a_0/a_1 s + 1}.$$

Створіть файл-сценарій для дослідження динамічних характеристик ланок з отриманими передатними функціями та виконайте програму. Виведіть графіки перехідних та імпульсних характеристик. Запишіть файл-сценарій на диск і запустіть його на виконання.

4. Створіть М-файл-функцію для дослідження динамічних характеристик ланки з заданою передатною функцією.

5. Створіть файл-сценарій, в якому зробіть три виклики М-файл-функції для дослідження динамічних характеристик ланок з усіма передатними функціями W1, W2, W3. Виведіть графіки перехідних та імпульсних функцій в три підвікна. Запустіть файл на виконання.

Таблиця 1.1 – Коефіцієнти для передаточних функцій

№ в.	1	2	3	4	5	6	7	8	9	10
a_0	15	20	25	50	60	70	80	60	90	100
a_1	3	4	5	10	3	7	10	12	15	20
b_0	60	40	50	80	60	70	30	48	60	50

1.3 Зміст звіту

1. Титульний лист.
2. Мета роботи.
3. Передатні функції для заданих рівнянь за п.3 підрозділу 1.3.
4. Графіки динамічних характеристик ланок за п.4, п.5, підрозділу 1.3.
5. Структурні схеми системи з ПІД та без ПІД-регулятора.
6. Розрахунок коефіцієнтів ПІД-регулятора для заданої системи: k_p , k_i , k_d.
7. Передатна функція системи з ПІД-регулятором.
8. Результати моделювання системи з ПІД та без ПІД-регулятора.
9. Стислі відповіді на контрольні питання.

1.4 Контрольні питання

1. Що таке робочий простір, зона перегляду та зона редагування системи MATLAB? Як записати у файл, стерти та видалити зміст робочого простору?
2. Що таке файл-сценарій, файл-функція? В чому їх різниця?
3. Що таке САК? Дайте визначення та наведіть приклади систем стабілізації, слідкування, термінальної.
4. Що таке ланка САК?
5. Дайте визначення передатної функції ланки. Як задати передатну функцію в системі MATLAB?
6. Для розрахунку коефіцієнтів та побудови перехідних характеристик можна скористатися файлом з лістингу 2.2.

Лістинг 1.5 – Побудова перехідних характеристик

```
clc;
fprintf('Визначення коефіцієнтів ПІД-регуля-
тора\n');
% Дані з таблиці 2_1 варіант №10
Kok=15;  T1=2.5;T2=0.45; T3=0.2;tau_k=0.06; time=15;
a0=T1*T2*T3;
a1=T1*T2+T2*T3+T1*T3;
a2=T1+T2+T3;
a3=Kok+1;
```

Кінець лістингу 1.5

```
fprintf('табл.2_1 варіант №\nKok=%.4f a0=%.4f a1=%.4f a4=%.4f a3=%.4f\n',Kok,a0,a1,a2,a3);
ku=(a1*a2/a0-1)/Kok;
P=2*pi*sqrt(a1/(Kok*ku+1));
fprintf('Межа стійкості ku=%.4f\n Період автоколивань P=%.4f\n',ku,P);
kp=0.6*ku;
ki=0.6*ku/P;
kd=0.15*ku*P;
fprintf('Коефіцієнти ПІД-регулятора kp=%.4f ki=%.4f kd=%.4f\n', kp,ki,kd);
W1=tf([0 Kok],[T1 1]); % 1-а передатна функція;
W2=tf([0 1],[T2 1]); % 2-а передатна функція;
W3=tf([0 1],[T3 1]); % 3-а передатна функція;
Wo=W1*W2*W3;   %передатна функція ОК;
Wi=tf([0 ki],[1 0]);% перед. ф-ція інтегр.складової;
Wd=tf([kd 0],[tau_k 1]);% перед. ф-ція диф. складової;
Wpid=kp+Wi+Wd;        % перед. ф-ція ПІД-регулятора;
Wopid=Wo*Wpid;%перед. ф-ція розімк. сист. з ПІД-регул
fprintf('Передатна функція замкненої системи з ПІД-регул ');
Wspid=feedback(Wopid,1) % передатна функція замкненої системи з ПІД-регул.;
figure;
step(Wo,'b', time); %побудова перех. хар.
figure;
step(Wspid,'r', time); %побудова перех. хар.
grid on;
```

Додатковий матеріал до п.3 підрозділу 1.2.

Формули для рівняння 1 (ліст. 1.6):

$$a_0 x_2' + a_1 x_2 = b_0 x_1,$$

$$a_0 s X_2 + a_1 X_2 = b_0 X_1,$$

$$(a_0 s + a_1) X_2 = b_0 X_1,$$

$$W(s) = \frac{X_2}{X_1} = \frac{b_0}{a_0 s + a_1} = \frac{b_0/a_1}{a_0/a_1 s + 1}.$$

Лістинг 1.6 – Для рівняння 1

```
  f1=[0   b0/a1]; %чисельник перед. ф-ції
  f2=[ a0/a1 1]; %знаменник перед. ф-ції
W=tf(f1,f2)
Приклад для варіанту_1
  f1=[0   60/3]; %чисельник перед. ф-ції
  f2=[ 15/3 1]; %знаменник перед. ф-ції
```

Формули для рівняння 2 (ліст. 1.7):

$$a_0 x_2' + a_1 x_2 = b_0 x_1',$$

$$a_0 \mathrm{s} \mathrm{X}_2 + a_1 X_2 = b_0 \mathrm{s} \mathrm{X}_1,$$

$$(a_0 s + a_1) X_2 = b_0 \mathrm{s} \mathrm{X}_1,$$

$$W(s) = \frac{X_2}{X_1} = \frac{b_0 s}{a_0 s + a_1} = \frac{b_0/a_1 s}{a_0/a_1 s + 1}.$$

Лістинг 1.7 – Для рівняння 2

```
  f1=[b0/a1  0]; %чисельник перед. ф-ції
  f2=[ a0/a1 1]; %знаменник перед. ф-ції
W=tf(f1,f2)
Приклад для варіанту_1
  f1=[60/3 0 ]; %чисельник перед. ф-ції
  f2=[ 15/3 1]; %знаменник перед. ф-ції
```

Приклад отримання передатної функції (x_1 – вхідна, x_2 – вихідна величини):

$$a_0 x_2'''' + a_1 x_2''' + a_2 x_2'' + a_3 x_2' + a_4 x_2 = \mathrm{b} \mathrm{x}_1.$$

Перетворення за Лапласом:

$$a_0 \cdot s^4 X_2 + a_1 \cdot s^3 X_2 + a_2 \cdot s^2 X_2 + a_3 \cdot \mathrm{s} \mathrm{X}_2 + a_4 X_2 = \mathrm{b} \mathrm{X}_1,$$

$$(a_0 \cdot s^4 + a_1 \cdot s^3 + a_2 \cdot s^2 + a_3 \cdot s + a_4) X_2 = \mathrm{b} \mathrm{X}_1.$$

Передатна функція:

$$W(s) = \frac{X_2}{X_1} = \frac{b}{a_0 s^4 + a_1 s^3 + a_2 s^2 + a_3 s + a_4},$$

$$W(s) = \frac{X_2}{X_1} = \frac{b/a_4}{a_0/a_4 s^4 + a_1/a_4 s^3 + a_2/a_4 s^2 + a_3/a_4 s + 1},$$

де $k = b/a_4$ – коефіцієнт передачі.

ЛАБОРАТОРНА РОБОТА № 2
Дослідження САК з ПІД регулятором

Мета роботи – вивчити вплив складових ПІД-регулятора на якісні показники САК.

2.1 Послідовні коригуючі пристрої (ПІД регулятори)

Структурна схема САК з ПІД-регулятором показана на рис. 2.1, де g – вхідний сигнал, y– вихідний сигнал, $y = g - y$ – помилка системи, u – керуючий сигнал, $W_{\text{ПІД}}$ – передатна функція ПІД-регулятора, $W_{\text{о}}(s)$ – передатна функція об'єкта керування.

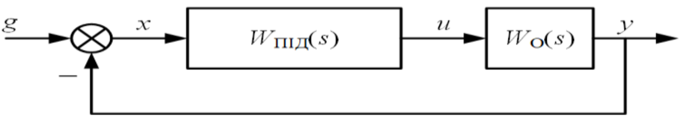


Рисунок 2.1 – Структурна схема САК з ПІД-регулятором

Передатна функція ПІД-регулятора має вигляд [1, 2]:

$$W_{\text{ПІД}}(s) = k_p + \frac{k_i}{s} + k_d s. \tag{2.1}$$

При цьому перша складова являє собою пропорційну ланку (П), друга складова – інтегруючу ланку (І), третя складова – диференціюючу ланку (Д). На підставі цього регулятор має назву ПІД-регулятор. Існують також П, ПІ, та ПД-регулятори, структура яких залежить від того, які складові входять до регулятора.

Структурна схема ПІД-регулятора зображена на рис. 2.2.

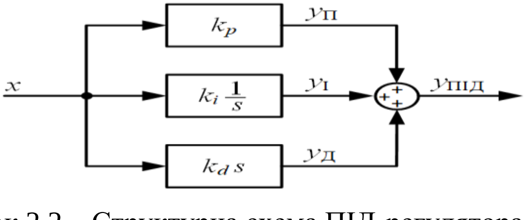


Рисунок 2.2 – Структурна схема ПІД-регулятора

Для покращення завадостійкості на практиці третю складову y (1.1) замінюють на диференціюючи ланку з уповільненням. Тоді передатна функція ПІД-регулятора буде мати вигляд

$$W_{\text{ПІД}}(s) = k_p + \frac{k_i}{s} + \frac{k_d s}{\tau_k s+1}. \qquad (2.2)$$

Структурна схема ідеального ПІД-регулятора зображена на рис. 2.3.

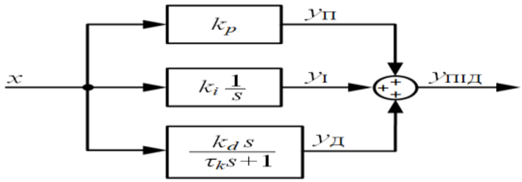


Рисунок 2.3 – Структурна схема ідеального ПІД-регулятора

2.2 Визначення коефіцієнтів ПІД регулятора

Для визначення коефіцієнтів регулятора найчастіше використовується метод Циглера – Нікольса. При використанні цього методу коефіцієнти k_p, k_i, k_d можна знайти двома способами: експериментально або аналітично. При експериментальному визначенню коефіцієнти k_i, k_d прирівнюються нулю, а коефіцієнт k_p збільшується до значення $k_{кр}$, поки система стане нестійкою. Значення $k_{кр}$ – це межа коливальної стійкості для коефіцієнта k_p. При k_p=$k_{кр}$ обчислюється період автоколивань – $P_{кр}$. (обирається кілька коливань, обчислюється час цих коливань та ділиться на кількість) Потім коефіцієнти обчислюються за формулами: $k_p = 0.6k_u$, $k_i = 0.6k_u/P_u$, $k_d = 0.15k_uP_u$. Перевагою цього методу є те, що він не потребує знання математичної моделі ОК.

Аналітичний метод використовується, коли відома математична модель ОК. При аналітичному розрахунку коефіцієнтів спочатку знаходять коливальну межу стійкості і частоту коливань системи на межі стійкості а потім по частоті коливань системи на межі стійкості знаходять період автоколивань $P_{кр}$.

Приклад. Нехай передаточна функція об’єкту керування має вигляд:

$$W_{OK}(s) = \frac{K_{ok}}{(T_1 s+1)(T_2 s+1)}.$$

Для покращення якості перехідного процесу використовуємо ПІД-регулятор з передатною функцією, що має вигляд:

$$W_{\text{рег}}(s) = k_p + \frac{k_i}{s} + \frac{k_d s}{\tau_k s+1},$$

де k_p , k_i , k_d – коефіцієнти передачі пропорціональної, інтегруючої та диференційної складових;

τ_k – постійна часу диференційної складової ПІД-регулятора (може дорівнювати 0.1…0.3 k_d /k_p).

Для визначення коефіцієнтів регулятора використаємо метод Циглера-Нікольса.

Для розрахунку коефіцієнтів ПІД регулятора використаємо аналітичний метод. Для цього знайдемо передатну функцію ОК із Р-законом регулювання (рис. 2.4).

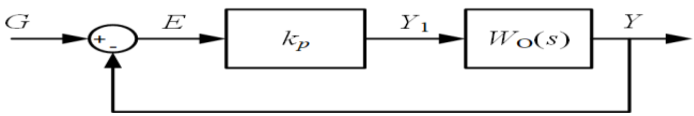


Рисунок 2.4 – Передатна функція ОК із Р-законом регулювання

Маємо

$$W_p(s) = \frac{k_p W_{ok}(s)}{1+k_p W_{ok}(s)} = \frac{k_p K_{ok}}{(T_1 s+1)(T_2 s+1)+k_p K_{ok}}. \qquad (2.3)$$

Характеристичне рівняння системи має вигляд

$$(T_1 s + 1)(T_2 s + 1) + k_p K_{ok} = 0. \qquad (2.4)$$

Якщо для такої системи знайти коливальну границю стійкості, то отримаємо

$$\omega_{\text{кол}} = \sqrt{\frac{1+k_p K_{ok}}{T_1 T_2}}. \qquad (2.5)$$

Враховуючи що $\omega_{\text{кол}} = 2\pi/p$, знаходимо період незатухаючих коливань P

$$P = 2\pi\sqrt{\frac{T_1 T_2}{1+k_p K_{ok}}}. \qquad (2.6)$$

Для подальших розрахунків для визначеності покладемо $k_{ok} = 10$, $k_p = 2$, $T_1 = 0.6$, $T_2 = 0.5$. Тоді отримаємо

$$k_u = k_p = 2,$$
$$P = 2\pi\sqrt{\frac{T_1 T_2}{1+k_p K_{ok}}} = 2\pi\sqrt{\frac{0.6\cdot 0.5}{1+2\cdot 10}} = 2\pi\sqrt{\frac{0.3}{21}} = 0.25,$$
$$k_p = 0.6, k_u = 1.2,$$
$$k_i = 0.6\frac{k_u}{P} = 0.6\frac{2}{0.25} = 1.6,$$
$$k_d = 0.15 k_u P.$$

М-файл для моделювання роботи ОК з ПІД регулятором має наступний вигляд (ліст. 2.1).

Після запуску М-файл-сценарію отримаємо передатну функцію системи з ПІД-регулятором

$$W_{PID} = \frac{2.433s^2+12.24s+15.98}{0.0045s^4+0.3165s^3+3.548s^2+13.24s+15.98}.$$

Результати моделювання системи керування без ПІД-регулятора та з ПІД-регулятором (k_p =1.2, k_i = 1.5929, k_d = 0.2253) наведені на рис.2.5.

Лістинг 2.1 – Моделювання роботи ОК з ПІД регулятором

```
clc
% параметри ОКа
Kok=10;T1=0.6;T2=0.5;
%розрахунок коефіцієнтів ПІД регулятора
ku=2;
P=2*pi*sqrt(T1*T2/(1+ku*Kok));
kp=0.6*ku; ki=kp/P; kd=0.15*ku*P; tau_k=0.015;
%знаходження передатної функції замкнутої системи
```

Кінець лістингу 2.1

```
W1=tf([0 Kok],[T1 1]); % 1-а передатна функція;
W2=tf([01],[T2 1]); % 2-а передатна функція;
Wo=W1*W2; %передатна функція розімкнутої системи;
Wi=tf([0 ki],[1 0]); % перед. ф-ція інтегр.складової;

Wd=tf([kd 0],[tau_k 1]); % перед. ф-ція диф.складової;

Wpid=kp+Wi+Wd;%передатна функція ПІД-регулятора;
W=Wo*Wpid %перед. ф-ція розімк. системи з ПІД рег.;
% перед. ф-ція замкн. сис. з ПІД-регул.;
Wos=feedback(W,1);
t_mod=3 %час моделювання
step(Wo,'b',Wos,'r', t_mod); %побудова перех. хар.
grid on;
```

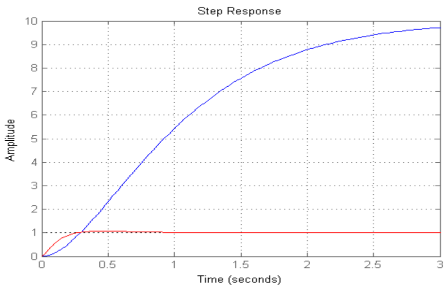


Рисунок 2.5 – Результати моделювання системи без ПІД та з ПІД-регулятором

2.3 Порядок виконання роботи

1. Розробіть структурні схеми системи керування для ОК з передатною функцією:

$$W_{\text{ок}}(s) = \frac{K_{\text{ок}}}{(T_1 s+1)(T_2 s+1)(T_3 s+1)}$$

без введення та з введенням ПІД-регулятора.

2. Створіть файл-сценарій для моделювання замкнутої системи без ПІД-регулятора та з ПІД-регулятором.

Примітка. Приведений ОК можна моделювати трьома послідовно з'єднаними інерційними ланками,

$$W_{\text{ок}}(s) = \frac{K_{\text{ок}}}{(T_1 s+1)} \cdot \frac{1}{(T_2 s+1)} \cdot \frac{1}{(T_3 s+1)},$$

а коефіцієнти ПІД-регулятора розрахувати за наступними виразами:

$$k_u = \frac{\frac{a_1 a_2}{a_0} - 1}{K_{ok}}$$

$$P = 2\pi \sqrt{\frac{T_1 T_2}{1+k_p K_{ok}}},$$

$$k_p = 0.6k_u, \quad k_i = 0.6k_u/P, \quad k_d = 0.15k_u P.$$

де

$$a_0 = T_1 T_2 T_3,$$

$$a_1 = T_1 T_2 + T_1 T_3 + T_2 T_3,$$

$$a_2 = T_1 + T_2 + T_3,$$

$$a_3 = 1 + k_p K_{ok}.$$

Коефіцієнти оберіть відповідно варіанту з таблиці 2.1.

Таблиця 2.1 – Коефіцієнти ПІД-регулятора

№	$K_{\text{ок}}$	T_1, с	T_2, с	T_3, с	$\tau_{\text{к}}$, с	$T_{\text{кін}}$,с
1	10	0.5	0.15	0.1	0.01	8
2	5	1.0	0.5	0.3	0.05	10
3	40	5.0	0.5	0.1	0.04	12
4	15	8.0	1.0	0.5	0.1	30
5	10	6.0	2.0	0.8	0.25	50
6	20	10.0	4.0	0.6	0.3	80
2	12	4.0	1.5	0.3	0.15	25
8	8	2.0	1.2	0.25	0.12	15
9	10	3.0	0.9	0.4	0.1	20
10	15	2.5	0.45	0.2	0.06	15

3. Запустіть М-файл і отримайте графіки перехідних функцій для системи без введення та з введенням ПІД-регулятора.

2.4 Зміст звіту

1. Титульний лист.
2. Мета роботи.
3. Передатні функції для заданих рівнянь за п.3 підрозділу 1.3.
4. Графіки динамічних характеристик ланок за п.4, п.5, підрозділу 1.3.
5. Структурні схеми системи з ПІД та без ПІД-регулятора.
6. Розрахунок коефіцієнтів ПІД-регулятора для заданої системи: k_p , k_i , k_d.
7. Передатна функція системи з ПІД-регулятором.
8. Результати моделювання системи з ПІД та без ПІД-регулятора.
9. Стислі відповіді на контрольні питання.

2.5 Контрольні питання

1. Які показники якості перехідного процесу Ви знаєте?
2. Наведіть визначення різних зворотних зв’язків (гнучкі, жорсткі, додатні, від’ємні, місцеві, головні, одиничні).
3. Як аналітичним способом обчислюється помилка, що склалася?
4. Які послідовні корегуючі пристрої Ви знаєте?
5. Які системи є статичними, а які астатичними?
6. Яка роль пропорційної, диференційної та інтегруючої складових ПІД-регулятора? Наведіть передатні функції кожної складової.
7. Якими методами можна зменшити помилку, що встановилася?

ЛАБОРАТОРНА РОБОТА № 3
Дослідження дискретних САК

Мета роботи: навчитися аналізувати дискретні САК, вивчити вплив періоду дискретизації на показники дискретної САК.

3.1 Теоретичні відомості

3.1.1 Дискретні САК

Дискретними системами називають системи автоматичного керування, в яких використовується дискретизація вхідних або вихідних сигналів. При дискретизації сигналів «безперервні» значення сигналів замінюються дискретними. Дискретизація сигналів може бути виконана за часом, за рівнем, або за часом та рівнем одночасно. До таких систем відносяться імпульсні та цифрові системи.

В імпульсних системах проводиться квантування сигналів за часом, або за рівнем, а в цифрових – за часом та рівнем одночасно. В імпульсних системах є одна або кілька ланок з імпульсним елементом.

Цифрова система складається з об’єкту керування (ОК), з аналого-цифрових (АЦП) і цифро-аналогових (ЦАП) перетворювачів, таймеру часу та пристрою керування, що реалізований на комп’ютері (спецобчислювачі, мікропроцесори).

Для аналізу дискретних систем можна використовувати різницеві рівняння, в яких стан системи визначено тільки в точках квантування [4]. Для аналізу різницевих рівнянь зручно використовувати Z-перетворення, що є аналогом перетворення Лапласа. Якщо в перетворенні Лапласа ввести комплексну зміну $z = e^{sT_s}$ з періодом дискретизації T_s, то отримаємо:

$$X(z) = Z\{x_i\} = \sum_{i=0}^{\infty} x_i\, e^{-\mathrm{i}T_s} = \sum_{i=0}^{\infty} x_i\, z^{-i}.$$

Основні властивості Z-перетворення:

– лінійність: $x_i = a_1 x_{1i} + a_2 x_{2i}$; $X(z) = a_1 X_1(z) + a_2 X_2(z)$;

– зсув за часом: $Z\{x_{i-m}\} = z^{-m}X(z)$; $Z\{x_{i+m}\} = z^{m}(X(z) - F_0(z))$, $F_0(z) = \sum_{j=0}^{m-1} x_j\, z^{-j}$;

– згортання: $Z(x \cdot g) = Z\sum_{j=0}^{\infty} x_j\, g(k-j) = Z(x)\cdot Z(g)$.

Дискретні системи поділяються на лінійні та нелінійні. В загальному випадку всі системи є нелінійними.

Лінійна дискретна система (ЛДС) – це система, яка може бути реалізована за допомогою трьох базових операцій: складення; множення на постійний коефіцієнт; затримки.

Передатна функція ЛДС – це відношення Z-перетворення вихідної послідовності сигналів до Z-перетворень вхідної послідовності сигналів за нульовими початковими умовами.

Для стійкості дискретної системи необхідно та достатньо, щоб всі полюси передатної функції (корні характеристичного рівняння) знаходилися на Z-площині всередині одиничного кола $|z| < 1$.

Частотні характеристики ЛДС – це функції аргументу $e^{j\omega Ts}$**,** вони є періодичними за частотою з періодом, що відповідає частоті дискретизації $\omega s = 2\pi / Ts$.

Для дискретних систем одним з важливих параметрів є період дискретизації.

Цей період можна визначити, якщо скористатися теоремою Котельникова: сигнал $x(t)$, перетворення Фур'є якого не має складових з частотами більш ніж ωs, повністю визначається своїми миттєвими значеннями, що відраховуються в дискретні моменти часу $T_S \le 2\pi/\omega s$.

Частоту $\omega_N = \omega s/2$ називають частотою Найквіста. Тому теорему Котельникова можна сформулювати інакше: сигнал $x(t)$, перетворення Фур'є якого прямує до нуля при частотах більших, ніж частота Найквіста ω_N, може бути повністю відновленим.

При роботі з дискретними системами важливо обрати частоту дискретизації такою, щоб вона була достатньо велика порівняно з частотною складовою сигналу. Крім того, перед квантуванням бажано профільтрувати безперервний сигнал, щоб частотні складові, що перебільшують частоту Найквіста $\omega_N = \omega s/2$, не спотворювали низькочастотні компоненти основного сигналу.

Критерієм вибору періоду дискретизації може бути також час для перетворення, відновлення вхідних сигналів та постійні часу замкненої системи.

Для перетворення безперервної замкнутої системи керування в дискретну період дискретизації можна обрати $T_S \approx 0{,}15 \ldots 0{,}5/\omega_c$, де ω_c – частота зрізу безперервної розімкнутої системи керування в рад/сек. При такому періоді дискретизації частота Найквіста буде в 5–20 разів більша частоти зрізу.

Для перетворення безперервної системи в дискретну використовують різні екстраполятори або дрібно-раціональну апроксимацію.

Екстраполятор нульового порядку отримує дискретний сигнал $u[k]$, генерує безперервний сигнал $u(t)$, що екстраполює кожне дискретне значення постійним значенням підчас одного періоду дискретизації, який подається на вхід безперервної ланки: $u(t)=u[k]$ при $kT_s \le t < (k+1)T_s$.

В екстраполяторі першого порядку використовується лінійна екстраполяція: $u(t)=u[k]+(t–kT_s)(u[k+1]-u[k])/T_s$ при $kT_s \le t < (k+1)T_s$.

Дрібно-раціональна апроксимація Тастіна використовує приблизне відношення для представлення експоненти:

$$z = \exp(sT_s) = \frac{1+sT_s/2}{1-sT_s/2}.$$

В системі MATLAB є багато функції для дослідження дискретних систем.

Функція SysD=tf(num,den,Ts) формує передатну функцію дискретної системи з періодом дискретизації Ts.

Функція SysD=c2d(SysC, Ts, ’метод’) реалізує перетворення безперервної системи SysC в дискретну SysD з періодом дискретизації Ts з використанням одного з методів: екстраполяторів нульового (`zoh`), першого (`foh`) порядків, дрібно-раціональну апроксимацію Тастіна (Tustin).

Функція Sys=d2d(Sys, Ts) формує дискретну систему Sys з новим періодом дискретизації Ts. При цьому дискретній системі повинен відповідати екстраполятор нульового порядку та відповідати послідовність перетворень Sys1=c2d(d2d(Sys), Ts).

Функція SysC=d2c(SysD,Ts) реалізує перетворення в безперервну систему SysC дискретної системи SysD з періодом дискретизації Ts, якщо використовувався екстраполятор першого порядку, а функція SysC=d2c(SysD) – якщо використовувався екстраполятор нульового порядку.

3.1.2 Дослідження впливу дискретного ПІД-регулятора

Проаналізуємо вплив дискретного регулятора на характеристики всієї системи. Нехай передатна функція безперервного об'єкту керування має вигляд:

$$W_{\text{ок}}(s) = \frac{k_{\text{ок}}}{(T_1 s+1)(T_2 s+1)}.$$

де $k_{ок}$=20 – коефіцієнт передачі;

T_1=0.6, T_2=0.5 – постійні часу.

Передатна функція безперервного ПІД-регулятора має вигляд:

$$W_{\text{рег}}(s) = k_p + \frac{k_i}{s} + \frac{k_d s}{\tau_k s+1},$$

де k_p, k_i, k_d – коефіцієнти передачі пропорціональної, інтегруючої та диференційної складових, що дорівнюють k_p= 1.2, k_i = 2.18, k_d = 0.165;

τ_k – постійна часу диференційної складової ПІД-регулятора (τ_k=0.015).

Перетворимо безперервні об'єкт та регулятор в дискретні з різними періодами дискретизації, скориставшись екстраполятором нульового порядку.

Якщо визначити частоту зрізу розімкнутої системи з ПІД-регулятором, то можна визначити період дискретизації за формулою $T_S \approx 0{,}15\ldots0{,}5/\omega_C$. Для нашої системи ω_C =12.3375, T_S=0.012…0.04.

При перетворенні аналогового регулятора в цифровий можна обирати період дискретизації з урахуванням коефіцієнтів передачі пропорційної та диференційної складової регулятора T_S=0.1…0.4k_d/k_p. Для нашої системи T_S=0.0143…0.053.

Обираємо два періоди дискретизації T_S1 = 0.025, T_S2 = 0.0125.

За допомогою М-файл-сценарію (ліст. 3.1) отримаємо передатні функції регулятора та всієї системи, а потім промоделюємо систему з різними періодами дискретизації з використанням методів: екстраполяторів нульового (`zoh`) та першого (`foh`) порядків.

Лістинг 3.1 – М-файл-сценарію моделювання системи

```
%Отримання загальної передатної функції системи;
clc;
K1=20;   T1=0.6;
K2=1;    T2=0.5; tk=0.015;
kp=1.2;    ki=2.23; kd=0.161;
W1=tf([0 K1],[T1 1]); % 1-а передатна функція;
W2=tf([0 K2],[T2 1]); % 2-а передатна функції;
W12=W1*W2; %перед.функція розімкнутої системи;
%перед. ф-ція ПІД-регулятора;
Wpid=kp+tf([0 ki],[1 0])+tf([kd 0],[tk 1]);
%передатна Ф-я розімкн. системи з ПІД-регулятором;
W12pid=W12*Wpid;
%обчислення частоти зрізу
[Am,Pm,Wca,Wcp]=margin(W12pid);
Wcp %Виведення на екран частоти зрізу
%передатна функція замкн. системи з ПІД-регулятором;
Wo=feedback(W12pid,1);
%Перетворення безперервної системи в дискретну
%за допомогою екстрополятору нульового порядку 'zoh'
Ts1=0.025; %завдання періоду дискретизації
%дискретна розімкн. система
W12_d1=c2d(W12,Ts1,'zoh');
%дискретний ПІД-регулятор
Wpid_d1=c2d(Wpid,Ts1,'zoh')
%дискретна передатна функція
%розімкнутої системи з ПІД-регулятором;
W12pid_d1=W12_d1*Wpid_d1;
%дискретна передатна ф-ція замкненої системи
% з ПІД-регулятором;
Wo_d1=feedback(W12pid_d1,1)
Ts2=0.0125; %завдання нового періоду дискретизації
W12_d2=c2d(W12,Ts2,'zoh');
Wpid_d2=c2d(Wpid,Ts2,'zoh')
W12pid_d2=W12_d2*Wpid_d2;
Wo_d2=feedback(W12pid_d2,1)
%Виведення графиків з часом моделювання 0.8 с.
step(Wo,'-.k',Wo_d1,'b',Wo_d2,'r',0.8); grid on;
```

Після моделювання отримаємо такі передатні функції дискретних регуляторів:

$$W_{\mathrm{pid1}}(z) = \frac{11.93\ \mathrm{z}^2 - 22.84\ \mathrm{z} + 10.95}{\mathrm{z}^2 - 1.189\ \mathrm{z} + 0.189}, \tag{3.1}$$

$$W_{\mathrm{pid2}}(z) = \frac{11.93\ \mathrm{z}^2 - 23.16\ \mathrm{z} + 11.24}{\mathrm{z}^2 - 1.435\ \mathrm{z} + 0.4346}, \tag{3.2}$$

та такі передатні функції систем:

$$W_{o_{d1}}(z) = \frac{0.2411\ \mathrm{z}^3 - 0.2276 z^2 - 0.2263\ \mathrm{z} + 0.2146}{z^4\ \text{-}2.853 z^3 + 3.145 z^2 - 1.672\ \mathrm{z} + 0.3869}, \tag{3.3}$$

$$W_{o_{d2}}(z) = \frac{0.06121\ \mathrm{z}^3 - 0.05852 z^2 - 0.05933\ \mathrm{z} + 0.05679}{\mathrm{z}^4 - 3.328 z^3 + 4.135 z^2 - 2.279\ \mathrm{z} + 0.4719}. \tag{3.4}$$

З результатів моделювання безперервної системи та дискретних систем з періодом дискредитації `0.025 с` і `0.0125 с` за допомогою екстрополятору нульового порядку `'zoh'` (рис. 3.1) бачимо, що при зменшенні періоду дискретизації перехідна характеристика дискретної САК наближається до перехідної характеристики безперервної САК.

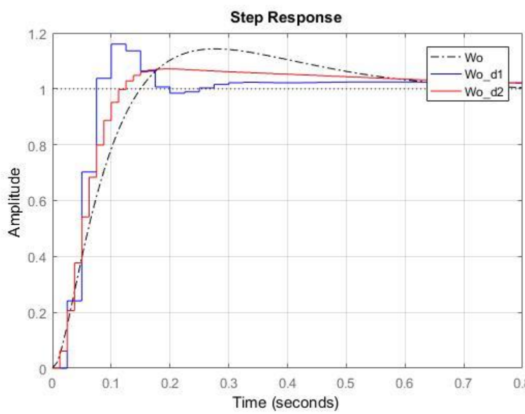


Рисунок 3.1 – Результати моделювання (за допомогою `'zoh'`)

При перетворенні безперервної системи в дискретну за допомогою екстрополятору першого порядку `'foh'` отримаємо такі передатні функції дискретних регуляторів:

$$W_{\text{pid1}}(z) = \frac{6.93452\ \mathrm{z}^2 - 11.85\ \mathrm{z} + 5.445}{\mathrm{z}^2 - 1.189\ \mathrm{z} + 0.189}, \quad (3.5)$$

$$W_{\text{pid2}}(z) = \frac{8.496\ \mathrm{z}^2 - 16.28\ \mathrm{z} + 7.798}{\mathrm{z}^2 - 1.435\ \mathrm{z} + 0.4346}, \quad (3.6)$$

та такі передатні функції систем:

$$W_{o_{d1}}(z) = \frac{0.0438\ \mathrm{z}^4 - 0.09076\ \mathrm{z}^3 + 0.2357 z^2 - 0.06765\ \mathrm{z} + 0.0353}{1.044\ z^4 - 3.009\ z^3 + 3.137 z^2 - 1.378\ \mathrm{z} + 0.2076}, \quad (3.7)$$

$$W_{o_{d2}}(z) = \frac{0.01436\ \mathrm{z}^4 - 0.0297\ \mathrm{z}^3 - 0.08285\ z^2 - 0.02562\ \mathrm{z} + 0.013}{1.015 z^4 - 3.36 z^3 + 4.111 z^2 - 2.194 \mathrm{z} + 0.4282}. \quad (3.8)$$

Результати моделювання безперервної системи та дискретних систем з періодом дискредитації `0.025 с` і `0.0125 с` за допомогою екстрополятору першого порядку `'foh'` на рис. 3.2.

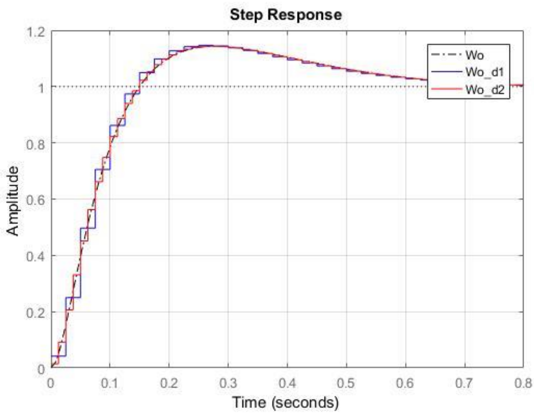


Рисунок 3.2 – Результати моделювання (за допомогою `'foh'`)

3.2 Порядок виконання роботи

1. Промоделюйте системи керування об'єктом з ПІД-регулятором. Передатна функція об'єкту:

$$W_{\text{ок}}(s) = \frac{k_{\text{ок}}}{(T_1 s+1)(T_2 s+1)(T_3 s+1)},$$

коефіцієнти для якої оберіть відповідно до табл. 1.1 із лабораторної роботи №1. Передатна функція ПІД-регулятора:

$$W_{\text{рег}}(s) = k_p + \frac{k_i}{s} + \frac{k_d s}{\tau_k s+1},$$

де τ_k – постійна часу ПІД-регулятора, значення якої задайте відповідно до табл. 1.1;

k_p , k_i , k_d – коефіцієнти ПІД-регулятора, значення яких дорівнюють тим, що обчислені в лабораторній роботі №1.

2. Створіть М-файл-сценарій, в якому визначить передатну функцію для системи з безперервними об'єктом та ПІД-регулятором. Розрахуйте період дискретизації T_s за різними методиками. Оберіть період дискретизації T_s (близький до мінімального).

3. Створіть М-файл-сценарій, в якому перетворіть безперервні об'єкт та ПІД-регулятор в дискретні з періодами дискретизації T_s та $T_s/2$.

4. Проведіть моделювання систем з безперервним та дискретними ПІД-регуляторами: `Wo`, `Wo_d1`, `Wo_d2`. З різними методами перетворення безперервних регуляторів в дискретні (екстраполятори нульового, першого порядків). Час моделювання оберіть з табл. 4.1 згідно варіанту. Графіки перехідних характеристик цих систем збережіть.

5. Запишіть передатні функції дискретних ПІД-регуляторів та передатні функції дискретних систем.

Таблиця 3.1 – Час моделювання

№ вар.	1	2	3	4	5	6	7	8	9	10
$T_{кін}$, *с*	1.5	6.0	3.0	10.0	20.0	20.0	8.0	7.0	6.0	4.0

3.3 Зміст звіту

1. Титульний лист.
2. Мета роботи.
3. Результати моделювання систем з безперервним та дискретними ПІД-регуляторами.
4. Передатні функції дискретних ПІД-регуляторів та передатні функції дискретних всієї системи.
5. Стислі відповіді на контрольні питання.

3.4 Контрольні питання

1. Які системи є дискретними?
2. З яких обов'язкових складових складається цифрова САК?
3. Які засоби для аналізу дискретних систем Ви знаєте?
4. Які властивості у *Z*-перетворення?
5. За допомогою яких операцій може бути реалізована лінійна дискретна система?
6. В якому діапазоні розглядаються частотні характеристики для дискретних систем?
7. Як визначити період дискретизації?
8. Як перетворити безперервну систему в дискретну?

ЛАБОРАТОРНА РОБОТА № 4
Створення та дослідження математичних моделей САК у просторі станів

Мета роботи: ознайомитися з методами створення та дослідження математичних моделей лінійної стаціонарної системи в середовищі програми MATLAB у просторі стану за допомогою пакету Control System Toolbox.

4.1 Теоретичні відомості

У загальному випадку кількість змінних, які повністю характеризують стан об'єкта дорівнює порядку системи. Ці змінні приймають за координати n-мірного вектору, який називається вектором стану. n-вимірний вектор $x \in R^n$, координатами якого є n змінних, які повністю характеризують стан об'єкта керування, називається вектором стану. Безліч можливих положень вектору станів називається простором станів.

Загалом вектор стану описується системою нелінійних диференціальних рівнянь виду (4.1)

$$\dot{x} = f(x, u), x \in R^n\ , u \in R^m, \tag{4.1}$$

де x – вектор станів;

u – вектор вхідних змінних чи вектор керування.

На будь-якому об'єкті керування завжди є датчики первинної інформації. Рівняння, що зв'язує координати вектору стану з вихідними сигналами датчиків, прийнято називати рівнянням вимірювань. Загалом це рівняння також є нелінійним [3]:

$$y = g(x, u),\ \mathrm{y} \in R^r. \tag{4.2}$$

Рівняння (4.1), (4.2) є математичною моделлю об'єкта керування у просторі стану. Далі будемо розглядати лише лінійні системи з постійними параметрами.

Вектор станів лінійної системи описується лінійною системою диференціальних рівнянь виду (4.3), (4.4):

$$\dot{x}(t) = Ax(t) + Bu(t), \qquad (4.3)$$

$$y(t) = Cx(t) + Du(t) \qquad (4.4)$$

де

$$x(t) \in R^n, u(t) \in R^m, \quad y(t) \in R^r$$

$$x(t) = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}, \mathrm{u(t)} = \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_m \end{pmatrix}, \mathrm{y(t)} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_r \end{pmatrix}$$

$$A(t) = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix}, \qquad B(t) = \begin{pmatrix} b_{11} & \cdots & b_{1m} \\ \vdots & \ddots & \vdots \\ b_{n1} & \cdots & b_{nm} \end{pmatrix},$$

$$C(t) = \begin{pmatrix} c_{11} & \cdots & c_{1n} \\ \vdots & \ddots & \vdots \\ c_{r1} & \cdots & c_{rn} \end{pmatrix}, \; D(t) = \begin{pmatrix} d_{11} & \cdots & d_{1n} \\ \vdots & \ddots & \vdots \\ d_{r1} & \cdots & d_{rn} \end{pmatrix}.$$

Для опису лінійних систем у просторі стану в пакеті Control System Toolbox застосовуються моделі підкласу `ss`, які ґрунтуються на лінійних диференціальних чи різницевих рівняннях.

Для формування моделі в підкласі `ss` використовується функція `sys=ss(A,B,C,D)`. В результаті отримаємо опис `ss` об'єкту у вигляді наступної четвірки матриць $[A \quad B \quad C \quad D]$, які повинні мати узгоджені розміри. Для моделей з нульовою матрицею можна використовувати присвоєння $D = 0$, як скорочену форму запису нульової матриці.

Приклад. Розглянемо модель електричного двигуна. У двигуні протікають два процеси – механічний (характеризується кутовою швидкістю обертання) та електричний (струм в обмотках).

Для механічного процесу маємо:

$$J\Omega = K_m i. \qquad (4.5)$$

Для електричного процесу:

$$L\frac{di}{dt} + Ri + K_e\Omega = u. \qquad (4.6)$$

В рівняннях (4.5), (4.6) використано наступні позначення: J, L, R_{Ω}, K_e, K_m – конструкційні параметри двигуна, u – вхідна напруга.

Знайшовши похідні з рівнянь (4.5) та (4.6), отримаємо систему (2.7):

$$\begin{cases} \frac{d\Omega}{dt} = \frac{K_m}{J} i, \\ \frac{di}{dt} = -\frac{R}{L} i - \frac{K_e}{L}\Omega + \frac{1}{L}u \end{cases}. \tag{4.7}$$

Ввівши до розгляду вектор стану $x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} \Omega \\ i \end{pmatrix}$, систему (4.7) можна записати так

$$\begin{cases} \frac{dx_1}{dt} = \frac{K_m}{J} x_2 \\ \frac{dx_2}{dt} = \frac{-R}{L} x_2 - \frac{K_e}{L} x_1 + \frac{1}{L} u. \end{cases} \tag{4.8}$$

або

$$\dot{x} = Ax + Bu, \tag{4.9}$$

$$A = \begin{pmatrix} 0 & \frac{K_m}{J} \\ \frac{-K_e}{L} & \frac{-R}{L} \end{pmatrix}, B = \begin{pmatrix} 0 \\ \frac{1}{L} \end{pmatrix}. \tag{4.10}$$

Рівняння (4.8) і є потрібним рівнянням для вектора стану електродвигуна. Припустимо, що ми вимірюємо кутову швидкість обертання електродвигуна. Тоді рівняння виходу: $y = x_1$. Таким чином, маємо наступну модель електродвигуна в просторі станів, яка в пакеті Control System Toolbox може бути сформована як у ліст. 4.1.

$$\dot{x} = Ax + Bu,$$

$$y = Cx,$$

$$A = \begin{pmatrix} 0 & \frac{K_m}{J} \\ \frac{-K_e}{L} & \frac{-R}{L} \end{pmatrix}, B = \begin{pmatrix} 0 \\ \frac{1}{L} \end{pmatrix}, C = (1 \quad 0), D = 0$$

Лістинг 4.1 – Модель в пакеті Control System Toolbox

```
%Формування чисельних параметрів моделі
Km=0.015;
Ke=0.2;
J=0.02;
R=2;
L=0.5;
%Розрахунок матриць моделі
A=[0 Km/J; -Ke/L -R/L];
B=[0;1/L];
C=[1 0];
D=0;
% Створення моделі в підкласі ss
sys=ss(A,B,C,D)
```

В результаті на екрані отримаємо такі дані:

```
sys =

A =
           x1      x2
   x1       0     0.1
   x2      -3   -2000
B =
        u1
   x1    0
   x2   10
C =
        x1  x2
   y1    1   0
D =
        u1
   y1    0
```

Крім форми `sys=ss(A,B,C,D)`, яка дозволяє створити в середовищі програми MATLAB модель ОК в просторі стану за матрицями `A,B,C,D`, є форма, яка дозволяє створити модель ОК в просторі стану за передатною функцією.

Нехай модель ОК задана у вигляді передатної функції

$$W(s) = \frac{s+1}{s^3+3s^2+3s+2}.$$

Модель ОК в просторі стану може бути сформована як у ліст. 4.2.

Лістинг 4.2 – Модель ОК в просторі стану

```
W=tf([1 1],[1 3 3 2])
sys1=ss(W)
```

На екрані отримаємо:

```
sys1 =

A =
           x1    x2    x3
   x1      -3  -1.5    -1
   x2       2     0     0
   x3       0     1     0
B =
         u1
   x1    1
   x2    0
   x3    0
C =
   x1   x2   x3
   y1 0  0.5  0.5
D =
         u1
   y1   0
```

4.2 Аналіз керованості та спостережливості системи

Поняття керованості і спостережливості є одними з основних понять теорії керування.

На змістовному рівні керованість означає можливість приведення системи у будь-який заданий стан, спостережливість – можливість визначення стану системи за результатами вимірів. Ці властивості дуже суттєві для побудови працездатних систем автоматичного керування. Розглянемо способи аналізу цих властивостей у пакеті Control System Toolbox.

4.2.1 Аналіз керованості

Керованість – властивість об’єкту керування бути переведеним з будь-якого початкового стану в будь-який кінцевий стан за обмежений проміжок часу за умови відсутності обмежень на керування.

Дана властивість визначає, чи можливо взагалі керувати ОК так, як потрібно для досягнення певного вихідного значення.

Для визначення керованості системи потрібно знайти матрицю керованості P системи, яка формується з матриць моделі простору станів A та B:

$$P = (B \;\; AB \;\; A^2B \;\; \dots A^nB),$$

а потім знайти ранг цієї матриці за допомогою функції *rank*(P).

Для систем з постійними параметрами існує така теорема: n-вимірна лінійна система з постійними параметрами є керованою, якщо ранг матриці керованості дорівнює n.

В пакеті Control System Toolbox існує функція *ctrb*(*A, B*) яка дозволяє знайти матрицю керованості системи.

Приклад. Знайдемо матрицю керованості для моделі електродвигуна (ліст. 4.3).

Лістинг 4.3 – Аналіз керованості

```
%Формування чисельних параметрів моделі
Km=0.015;
Ke=0.2;
J=0.02;
R=2;
L=0.5;
%Розрахунок матриць моделі
A=[0 Km/J; -Ke/L -R/L];
B=[0;1/L];
C=[1 0];
D=0;
% Створення моделі в підкласі ss
sys=ss(A,B,C,D)
P= ctrb(A, B)% Знаходження матриці керованості
rankP=rank(P) % Визначення рангу матриці керованості
```

На екрані отримаємо:

```
rankP =
     2
```

Ранг матриці керованості *P* дорівнює 2, отже система керована.

4.2.2 Аналіз спостережливості

Спостережливістю системи називається така її властивість, коли шляхом спостереження (вимірювання) на кінцевому інтервалі часу її вихідних величин $y(t)$ можна однозначно визначити всі координати вектору $x(t)$ стану системи. У цьому випадку кажуть, що система є повністю спостережувана.

Система називається не в повному обсязі спостережливою, якщо через вимір y(t) можна визначити лише частину координат вектору стану.

З точки зору побудови регуляторів, це означає, що якщо система не спостережувана, то для неї неможливо буде відновити вектор стану і відповідно нею не можна буде керувати на практиці.

У реальних ОК дуже рідко відомий весь вектор стану і для систем такого типу не можна побудувати регулятор. Тому дослідження спостережливості при проектуванні системи керування дуже важливе. В пакеті Control System Toolbox існує функція obsv(*A, C*), яка дозволяє знайти матрицю спостережливості системи.

Приклад. Аналіз спостережливості моделі електродвигуна.

Лістинг 4.4 – Аналіз спостережливості

```
%Анализ спостережуваності моделі електродвигуна
clc
%Формування чисельних параметрів моделі
Km=0.015;
Ke=0.2;
J=0.02;
R=2;
L=0.5;
%Розрахунок матриць моделі
A=[0 Km/J; -Ke/L -R/L];
B=[0;1/L];
C=[1 0];
D=0;
```

```
% Створення моделі в підкласі ss
sys=ss(A,B,C,D)
Q=obsv(A,C)%Знаходження матриці спостережливості
rankQ=rank(Q) % Визначення рангу матриці спостереж-
ливості
```

На екрані отримаємо:

```
rankQ =
     2
```

Ранг матриці керованості Q дорівнює 2, отже система спостережувана.

4.2.3 Аналіз стійкості моделі в просторі стану

Стійкість є однією із основних вимог до САК.

Під стійкістю лінійної системи розуміють властивість загасання перехідного процесу з часом, інакше кажучи, – наступна властивість власного руху системи $x_{вл}(t) \to 0$ при $t \to \infty$.

Для стійкості існує наступна теорема: для того щоб лінійна система керування була стійка, необхідно і достатньо, щоб усі корені її характеристичного рівняння мали від'ємну дійсну частину.

Із цієї теореми слідує, що аналіз стійкості лінійної системи полягає в знаходженні коренів характеристичного рівняння системи. В пакеті Control System Toolbox для цього існує функція `pole`. Звернення до неї наступне `Korni=pole(sys)`.

Приклад. Аналіз стійкості моделі електродвигуна (ліст. 2.5).

Лістинг 4.5 – Аналіз стійкості

```
%Формування чисельних параметрів моделі
Km=0.015;
Ke=0.2;
J=0.02;
R=2;
L=0.5;
%Розрахунок матриць моделі
A=[0 Km/J; -Ke/L -R/L];
B=[0;1/L];
C=[1 0];
D=0;
```

Кінець лістингу 4.5

```
% Створення моделі в підкласі ss
sys=ss(A,B,C,D)
%Знаходження коренів характеристичного рівняння
Korni=pole(sys)
```

На екрані отримаємо:

```
Korni =
   -0.0765
   -3.9235
```

4.2.4 Отримання перехідної та імпульсної характеристик системи

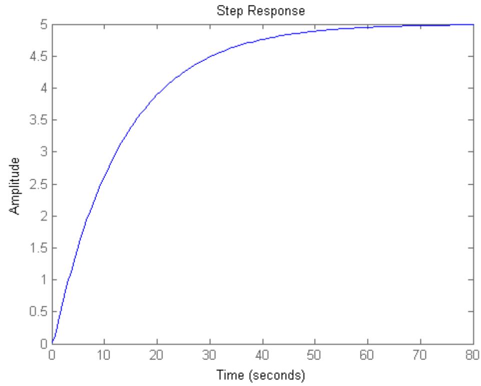
Перехідну та імпульсну характеристику системи можна отримати за допомогою команд `step(sys)`,`impulse(sys)`.

При цьому перелік координат вектору стану, для яких розраховуються часові характеристики, задається матрицею С.

Побудова часових характеристик по координаті x_1 (ліст. 4.6), по координаті x_2 (ліст. 4.7). Результати побудови характеристик можна бачити на графіках (рис. 4.1, 4.2).

Лістинг 4.6 – Побудова часових характеристик по координаті x_1

```
%Розрахунок матриць моделі
A=[0 Km/J; -Ke/L -R/L];
B=[0;1/L];
C=[1 0];%визначення координати, для якої будуть
% будуватись часові характеристики
% (в даному випадку це x1)
D=0;
% Створення моделі в підкласі ss
sys=ss(A,B,C,D)
%Побудова часових характеристик для координати x1
figure;
step(sys)
figure;
impulse(sys)
```



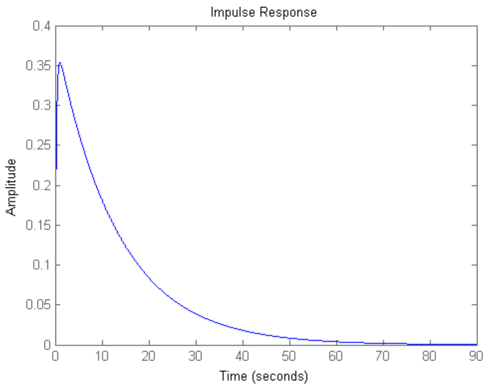
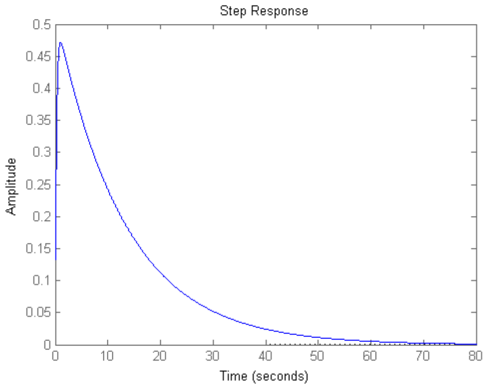


Рисунок 4.1 – Результати побудови часових характеристик для x_1

Лістинг 4.7 – Побудова часових характеристик по координаті x_2

```
C=[0 1]; % координати x2 для побудови часових характеристик
sys=ss(A,B,C,D)
%Побудова часових характеристик для координати x2
figure;
step(sys)
figure;
impulse(sys)
```



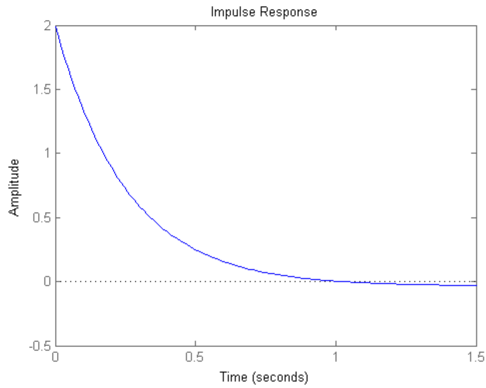


Рисунок 4.2 – Результати побудови часових характеристик для x_2

4.3 Порядок виконання роботи

1. По передатній функції ланки $W(s) = \frac{\text{К}}{T^2 s^2 + 2\text{dTs} + 1}$ і даними таблиці 2.1 побудувати модель ланки у просторі стану.
2. Дослідити керованість отриманої моделі.
3. Дослідити спостережливість отриманої моделі.
4. Дослідити стійкість отриманої моделі.
5. Побудувати часові характеристики моделі для координати x_1.

Таблиця 4.1 – Вихідні дані для побудови коливальної ланки у просторі стану

№	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
К	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
T, c	0.5	0.5	0.5	0.5	0.5	1.0	1.0	1.0	1.0	1.0	2.0	2.0	2.0	2.0	2.0
d	0	0.1	0.2	0.8	0.3	0.5	0.7	0.5	0.6	0.3	0.2	0	0.4	0.7	0.1

4.4 Зміст звіту

1. Титульний лист.
2. Мета роботи.
3. Тексти створених М-файлів для проведення досліджень і результати їх роботи.
4. Стислі відповіді на контрольні питання.

4.5 Контрольні питання

1. Визначення вектору стану
2. Визначення простору стану
3. Теорема про керованість лінійної системи з постійними параметрами
4. Теорема про спостережливість лінійної системи з постійними параметрами
5. Як створити модель системи керування у просторі стану?
6. За допомогою яких функцій можна побудувати часові характеристики системи?
7. Як знайти корені системи керування у просторі стану?

ЛАБОРАТОРНА РОБОТА №5
Синтез модального управління для лінійних САК

Мета роботи: ознайомитися з методами побудови та дослідження модальних регуляторів для лінійної стаціонарної системи в середовищі програми MATLAB за допомогою пакету Control System Toolbox.

5.1 Теоретичні відомості

Характер перехідних процесів у системі визначається розташуванням коренів λ_i її характеристичного рівняння. Дійсно, рішення однорідного диференціального рівняння n-го порядку $x(t)$ має вигляд:

$$x(t) = \sum_{i=1}^{n} C_i(t) x_i(t) \,,$$

де постійні C_i визначаються початковими умовами [4]. Тому забезпечення «хороших» перехідних процесів у системі можливо досягнуто якщо характеристичний многочлен має задані корні. Це призводить до задачі отримання коефіцієнтів характеристичного рівняння замкнутої системи, які забезпечують заданий розподіл коренів. Корні характеристичного рівняння однозначно залежать від його коефіцієнтів, тому модальне управління – це цілеспрямоване зміна коефіцієнтів характеристичного рівняння об'єкта з допомогою зворотного зв'язку за станом. Регулятори, побудовані виходячи з зазначеної вимоги, називаються модальними регуляторами.

Постановка задач синтезу модального регулятора наступна: для повністю керованого об'єкта з одним входом:

$$\dot{x} = Ax + bu, \tag{5.1}$$

де A – матриця розміру $n \times n$;

b – n-мірний вектор-стовпець;

$u = -k_1 x_1 - k_2 x_2 \cdots - k_n x_n$ – скалярне управління.

Потрібно знайти коефіцієнти k_i, i=1,2,…n, закону управління, які забезпечують необхідний розподіл полюсів характеристичного рівняння

$$\lambda^n + a_{n-1}\lambda^{n-1} + \cdots a_0 = 0.$$

Розв'язання задачі засноване на наступній теоремі: лінійна керована система з постійними параметрами та скалярною вхідною змінною u

$$\dot{x} = Ax + bu$$

завжди може бути перетворена до канонічної форми виду (5.2):

$$\dot{x}' = \tilde{A}x' + \tilde{b}u, \tag{5.2}$$

де

$$\tilde{A} = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & 1 \\ -a_0 & -a_1 & \cdots & \cdots & -a_{n-1} \end{pmatrix}, \tilde{b} = \begin{pmatrix} 0 \\ \vdots \\ \vdots \\ 1 \end{pmatrix}.$$

При цьому матриця перетворення в канонічну форму має вигляд

$$T = \tilde{P}P^{-1},$$

де $P = (b \quad Ab \quad A^2b \quad ... \quad A^nb)$ – матриця керованості системи;

$\tilde{P} = (\tilde{b} \quad A\tilde{b} \quad A^2\tilde{b} \quad ... \quad A^n\tilde{b})$ – матриця керованості системи в канонічній формі.

Виберемо u у вигляді

$$u = -\tilde{k}_1x_1' - \tilde{k}_2x_2' \cdots - \tilde{k}_nx_n' = -\tilde{k}x', \tilde{k} = (\tilde{k}_1\tilde{k}_2 \cdots \tilde{k}_n),$$

$$x' = \begin{pmatrix} x_1' \\ x_2' \\ \vdots \\ x_n' \end{pmatrix}.$$

Підставивши в рівняння (5.2) управління u, отримаємо рівняння замкнутої системи у канонічній формі

$$\dot{x}' = \tilde{A}_з x',$$

де

$$\tilde{A}_з = \begin{pmatrix} 0 & 1 & & 0 & \cdots & 0 \\ 0 & 0 & & 1 & \cdots & 0 \\ \cdots & \cdots & & \cdots & \cdots & \cdots \\ 0 & 0 & & 0 & \cdots & 1 \\ -(a_0 + \tilde{k}_1) & -(a_1 + \tilde{k}_2) & \cdots & \cdots & & -(a_{n-1} + \tilde{k}_n) \end{pmatrix}.$$

Введемо позначення

$$g_{i-1} = a_{i-1} + \tilde{k}_i$$

Тоді матрицю $\tilde{A}_з$ можна представити у вигляді

$$\tilde{A}_з = \begin{pmatrix} 0 & 1 & & 0 & \cdots & 0 \\ 0 & 0 & & 1 & \cdots & 0 \\ \cdots & \cdots & & \cdots & \cdots & \cdots \\ 0 & 0 & & 0 & \cdots & 1 \\ -g_0 & -g_1 & \cdots & \cdots & & -g_{n-1} \end{pmatrix}.$$

Характеристичне рівняння для цієї матриці має вигляд

$$\lambda^n + g_{n-1}\lambda^{n-1} + \cdots g_0 = 0 \,.$$

Це рівняння виражається через своє корні α_i наступним чином:

$$\lambda^n + g_{n-1}\lambda^{n-1} + \cdots g_0 = (\lambda - \alpha_1)(\lambda - \alpha_2) \cdots (\lambda - \alpha_n).$$

При цьому згідно з теоремою Вієта коефіцієнти цього рівняння визначаються виразом (5.3):

$$g_{n-1} = -(\alpha_1 + \alpha_2 + \cdots \alpha_n),$$
$$g_{n-2} = \alpha_1\alpha_2 + \alpha_1\alpha_3 + \cdots \alpha_1\alpha_n + \alpha_2\alpha_3 + \cdots + \alpha_{n-1}\alpha_n,$$
$$g_{n-3} = -(\alpha_1\alpha_2\alpha_3 + \alpha_1\alpha_2\alpha_4 + \cdots \alpha_{n-2}\alpha_{n-1}\alpha_n), \quad (5.3)$$
$$g_1 = (-1)^{n-1}(\alpha_1\alpha_2\alpha_3 \cdots \alpha_{n-1} + \alpha_1\alpha_2\alpha_4 \cdots \alpha_n + \cdots \alpha_2\alpha_3 \cdots \alpha_n),$$
$$g_0 = (-1)^n(\alpha_1\alpha_2\alpha_3 \cdots \alpha_{n-1}).$$

Задавшись бажаними коріннями α_i можна спочатку знайти коефіцієнти g_i, а потім коефіцієнти $\tilde{k}_i = g_{i-1} - a_{i-1}$ закону управління $u = -\tilde{k}_1 x_1' - \tilde{k}_2 x_2' \cdots - \tilde{k}_n x_n'$.

На практиці немає необхідності шукати коефіцієнти g_i за формулами (3.3). Для реалізації зазначеного алгоритму можна скористатися вбудованими функціями пакета Control System Toolbox системи MATLAB. У цьому пакеті є дві команди: `acker(A,B,P)` та `place(A,B,P)`, які дозволяють для заданої в просторі стану моделі системи `(A,B)` по вектору бажаних коренів замкнутої системи P розрахувати коефіцієнти k_i закону управління

$$u = -k_1 x_1 - k_2 x_2 \cdots - k_n x_n.$$

Приклад. Для моделі `A`, `B` та вектору `P` (ліст. 5.1) бажаного розташування коренів λ_i характеристичного рівняння замкнутої системи знайти коефіцієнти k_i закону управління u:

$$A = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, B = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, p = \begin{pmatrix} -0.1 \\ -0.2 \end{pmatrix}.$$

Лістинг 5.1 – Реалізація прикладу

```
%Розрахунок матриць моделі
A=[0 1; 0 0];
B=[0;1];
C=[0 0];
D=0;
P=[-0.1;-0.2]
sys=ss(A,B,C,D) % Створення моделі в підкласі ss
%Знаходження коефіцієнтів ki закону управління u
K=acker(sys.A,sys.B,P)
```

На екрані отримаємо:

```
sys =

A =
        x1  x2
   x1    0   1
   x2    0   0
B =
        u1
   x1    0
   x2    1
C =
        x1  x2
   y1    0   0
D =
        u1
   y1    0

Continuous-time state-space model.

P =
   -0.1000
   -0.2000

ans =
K =
    0.0200    0.3000
```

5.2 Порядок виконання роботи

1. По передатній функції ланки $W(s) = \frac{K}{a_3 s^3 + a_2 s^2 + a_1 s + a_0}$ і даними таблиці 3.1 побудувати модель ланки у просторі стану.

2. Дослідити керованість отриманої моделі.

3. Дослідити стійкість отриманої моделі.

4. По вектору бажаних коренів замкнутої системи p розрахувати коефіцієнти k_i закону управління $u = -k_1 x_1 - k_2 x_2 \cdots - k_n x_n$. Бажані корені взяти із таблиці 3.2.

5. Побудувати часові характеристики замкнутої моделі для координати x_1.

Таблиця 5.1 – Дані для побудови моделі ланки у просторі стану

Варіант	a_3	a_2	a_1	a_0	K
1	1	-1	-0.07	0.534	1
2	1	-0.9	-0.01	0.105	1
3	1	1	-0.07	-0.534	1
4	1	-0.4	0.33	0.09	1
5	1	0.8	0	-1.024	1
6	1	0.1	0.2	-0.2	1
7	1	0.6	-0.11	-0.366	1
8	1	-1.5	0.71	-0.105	1
9	-1	1.5	0.71	0.105	1
10	1	0.9	-0.01	-0.105	1

Таблиця 5.2 – Корені замкнутої системи

Варіант	λ_1	λ_2	λ_3	Варіант	λ_1	λ_2	λ_3
1	-0.1	-0.3	-0.07	6	-1	-0.1	-0.2
2	-0.01	-0.9	-0.01	7	-0.33	-0.6	-0.11
3	-0.5	-0.5	-0.07	8	-1	-1.5	-0.71
4	-0.1	-0.4	-0.33	9	-0.1	-0.5	-0.71
5	-0.6	-0.08	-0.2	10	-0.21	-0.9	-0.01

5.3 Зміст звіту

1. Титульний лист.
2. Мета роботи.
3. Тексти створених М-файлів для проведення досліджень і результати їх роботи.
4. Стислі відповіді на контрольні питання.

5.4 Контрольні питання

1. Визначення біномного розподілу коренів.
2. Визначення розподілу коренів за Баттервортом.
3. Постановка задачі модального керування.

4. Канонічна форма лінійної системи з постійними параметрами та скалярною вхідною змінною u.

5. Матриця перетворення до канонічної форми.

Примітка: для розрахунку коефіцієнтів закону керування та побудови перехідних характеристик можна скористатися лістингом 5.2.

Лістинг 5.2 – Побудова часових характеристик

```
clear
clc
den=[1 -1 -0.07 0.534];%знаменник передатної функції розімкнутої системи
num=[0 0 0 1];%чисельник передатної функції розімкнутої системи
p=[-0.1-0.3  -0.07];%бажані корені замкн. системи
% Створення моделі у вигляді передатної функції
W=tf(num,den)
sys=ss(W) % Створення моделі в підкласі ss
Korni=pole(sys) %Знах. коренів розімкнутої системи
%Аналіз керованості системи
P= ctrb(sys.A, sys.B);% Знах. матриці керованості
rankP=rank(P) % Визначення рангу матриці керованості
if rankP==3
    fprintf('система керована')
else
    fprintf('система не керована')
end
%Знаходження коефіцієнтів k_i закону управління u
K=acker(sys.A,sys.B,p)
% Створення моделі замкнутої системи в підкласі ss
Az=sys.A-sys.B*K;
C=[1 0 0];%визначення координати для побудови
% часових характ.(в даному випадку це x_1)
D=0;
sys_z=ss(Az,sys.B,C,D)
Korni_z=pole(sys_z) %Знах. коренів замкнутої системи
%Побудова часових характеристик для координати x_1
figure;
step(sys_z)
figure;
impulse(sys_z)
```

ЛАБОРАТОРНА РОБОТА № 6
Синтез оптимального управління для лінійних систем у середовищі MATLAB

Мета роботи: ознайомитися з методами побудови та дослідження оптимальних регуляторів для лінійної стаціонарної системи в середовищі програми MATLAB за допомогою пакету Control System Toolbox.

6.1 Теоретичні відомості

Постановка задач синтезу оптимального регулятора наступна [2, 3, 4]: для повністю керованого об'єкта

$$\dot{x} = Ax + Bu,$$

де A – матриця розміру $n \times n$;
B – матриця розміру $n \times m$;
$u = -Kx$ – закон управління;
потрібно знайти матрицю коефіцієнтів K закону управління, яка мінімізує функціонал

$$J(u) = \int_0^{\infty} (x^T Q x + u^T R u + x^T N u) dt.$$

Позитивно визначені симетричні матриці Q, N, R визначають співвідношення між якістю регулювання (тривалістю перехідного процесу) та витратами на управління і призначаються користувачем. Матриця зворотних зв'язків K обчислюється за формулою

$$K = R^{-1}(B^T S + N^T),$$

де S – рішення алгебраїчного рівняння Ріккаті:

$$A^T S + SA - (SB + N)R^{-1}(B^T S + N^T) + Q = 0.$$

Цей закон управління називається лінійним квадратичним оптимальним управлінням (ЛК).

Для реалізації зазначеного алгоритму в пакеті Control System Toolbox системи MATLAB є функция lqr. Синтаксис функції наступний (якщо аргумент N опущений, то за умовчанням N = 0):

[К, S, e] = lqr(A, B, Q, R)
[К, S, e] = lqr(A, B, Q, R, N)

Опис: функція [К, S, e] = lqr(A, B, Q, R, N) розраховує матрицю коефіцієнтів зворотних зв’язків *K*, таку, що закон управління

$$u = -Kx$$

мінімізує квадратичний критерій якості

$$J(u) = \int_0^\infty (x^T Q x + u^T R u + x^T N u) dt$$

для безперервної моделі, заданої у просторі станів. На додаток до матриці коефіцієнтів зворотних зв’язків *K* функція lqr повертає рішення S рівняння Ріккаті та власні значення e матриці замкнутої системи. Нижче наведено М-файл синтезу оптимального регулятора (ліст. 5.1) і результати моделювання (рис. 5.1) для прикладу: для моделі A, B знайти матрицю коефіцієнтів зворотних зв’язків *K* закону управління u:

$$A = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

Лістинг 6.1 – М-файл синтезу оптимального регулятора

```
%Параметри системи
A=[0 1; 0 0]; B=[0;1]; C=[0 0];
D=0;
%Параметри критерію якості управління
Q=[1 0;0 1];
R=1;
sys=ss(A,B,C,D) % Створення моделі в підкласі ss
[K,S,e]=lqr(sys.A,sys.B,Q,R)  %Знаходження  матриці
коефіцієнтів K закону управління u
```

На екрані отримаємо

```
sys =
A =
        x1  x2
   x1   0   1
   x2   0   0
B =
        u1
   x1   0
   x2   1
C =
        x1  x2
   y1   0   0
D =
        u1
   y1   0
Continuous-time state-space model.
K =
    1.0000    1.7321
S =
    1.7321    1.0000
    1.0000    1.7321
e =
  -0.8660 + 0.5000i
  -0.8660 - 0.5000i
```

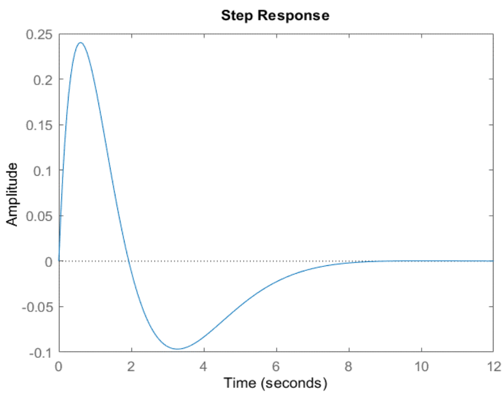


Рисунок 6.1 – Результати моделювання

6.2 Порядок виконання роботи

1. Задати параметри критерію якості управління для свого варіанта згідно таблиці 6.1.

2. Розрахувати оптимальну матрицю коефіцієнтів зворотних зв’язків *K*.

3. Побудувати часові характеристики замкнутої моделі для заданої координати згідно таблиці 6.1.

Таблиця 6.1 – Задана координата

Варіант	Матриця Q	Матриця R	Задана координата
1	[1 0 0; 0 1 0; 0 0 1]	1	x1
2	[½ 0 0; 0 1 0; 0 0 2]	1	x2
3	[2 0 0; 0 1 0; 0 0 2]	½	x3
4	[3 0 0 ;0 ½ 0; 0 0 2]	3	x1
5	[¼ 0 0; 0 ⅓ 0; 0 0 2/2]	1	x2
6	[½ 0 0; 0 1 0; 0 0 5]	1	x2
7	[⅙ 0 0; 0 ⅓ 0; 0 0 2]	1	x3
8	[½ 0 0; 0 1 0; 0 0 2]	1	x1
9	[⅛ 0 0; 0 1 0; 0 0 ½]	1	x2
10	[½ 0 0; 0 1 0; 0 0 2]	1	x3

6.3 Зміст звіту

1. Титульний лист.
2. Мета роботи.
3. Тексти створених М-файлів для проведення досліджень і результати їх роботи.
4. Часові характеристики замкнутої моделі для заданої координати.
5. Стислі відповіді на контрольні питання.

6.4 Контрольні питання

1. За якою формулою обчислюється матриця зворотних зв’язків *K*.
2. Привести алгебраїчне рівняння Ріккаті.
3. Постановка задачі оптимального керування.

4. Для чого призначена функция lqr, її синтаксис.

5. Для чого призначені матриці Q, N, R.

Примітка: для розрахунку коефіцієнтів закону керування та побудови перехідних характеристик можна скористатися лістингом 6.2.

Лістинг 6.2 – Побудова перехідних характеристик

```
clear
clc
den=[1 -1 -0.07 0.534];%знаменник передатної функції розімкнутої %системи
num=[0 0 0 1];%чисельник передатної функції розімкнутої системи
% Створення моделі у вигляді передатної функції
W=tf(num,den)
%Параметри критерію якості управління
Q=[1 0 0;0 1 0;0 0 1];
R=1;
sys=ss(W) % Створення моделі в підкласі ss
Korni=pole(sys) %Дослідж. стійкості розімкн. системи
%Аналіз керованості системи
P= ctrb(sys.A, sys.B);% Знах. матриці керованості
rankP=rank(P) % Визначення рангу матриці керованості
if rankP==3
    fprintf('система керована')
else
    fprintf('система не керована')
end
%Знаходження коефіцієнтів k_iзакону управління u
[K,S,e]=lqr(sys.A,sys.B,Q,R)
% Створення моделі замкнутої системи в підкласі ss
Az=sys.A-sys.B*K;
C=[1 0 0];%визнач. коорд.(в даному випадку це x_1)
D=0;
sys_z=ss(Az,sys.B,C,D)
Korni_z=pole(sys_z) %Знах. коренів замкнутої системи
%Побудова часових характеристик для координати x_1
figure;
step(sys_z)
figure;
impulse(sys_z)
```

ЛАБОРАТОРНА РОБОТА № 7
Знаходження екстремалі функціоналу засобами пакету MATLAB

Мета роботи: ознайомитися з методами знаходження та дослідження екстремалі функціоналу в середовищі програми MATLAB за допомогою пакету символьних розрахунків Symbolic Math Toolbox.

7.1 Теоретичні відомості

Визначення 1. Функцією називається будь-яке правило, за яким заданому числу x з деякої їх множини ставиться у відповідність число y. Зазвичай ми позначаємо функцію так:

$$y = f(x). \tag{7.1}$$

Множина чисел x, для яких визначена функція y, називається областю визначення функції. Зазвичай область визначення функції – це інтервал або система інтервалів (відкритих, напіввідкритих, закритих), або вся числова вісь. У випадку, якщо функції ставиться у відповідність число, то це буде функціонал [7].

Визначення 2. Функціоналом називається будь-яке правило, за яким заданій функції $y(x)$ з деякої їх множини ставиться у відповідність число J

$$J = J\big(y(x)\big). \tag{7.2}$$

Множину функцій $y(x)$, на яких визначено функціонал $J(y(x))$, назвемо класом функцій. При цьому кожну функцію $y(x)$, що належить заданому класу, розглядають як точку деякого простору. Простір, елементами якого є функції, називається функціональним простором. Функціональний простір R називається нормованим, якщо кожній функції $y(x)$ з R ставиться у відповідність деяке позитивне число, прийняте називати нормою функції $y(x)$ і позначати не одинарними вертикальними рисками, як модуль, а подвійними $\|\cdot\|$. Для однозначного тлумачення назва класу проставляється у вигляді індексу внизу після визначення норми. Залежно від правила обчислення норми розрізняють класи: C_1 (7.3), L_2 (7.4), C_0 (7.5).

$$\|f(x)\|_{C_1} = \max\left(\max_{\forall x \in [x_1, x_2]} |f(x)|, \max_{\forall x \in [x_1, x_2]} |\dot{f}(x)| \right), \tag{7.3}$$

$$\|f(x)\|_{L_2} = \sqrt{\int_{x_1}^{x_2} f^2(x)dx}, \tag{7.4}$$

$$\|f(x)\|_{C_0} = \max_{\forall x \in [x_1, x_2]} |f(x)|. \tag{7.5}$$

Визначення 3. Варіацією функції *y(x)* на функції $y_0(x)$ називається величина

$$\delta y(x) = y(x) - y_0(x).$$

Визначення 4. Збільшенням функціоналу J на функції $y_0(x)$ відповідним варіації функції δy, називається величина

$$\Delta J(y_0) = J(y_0 + \delta y) - J(y_0).$$

Визначення 5. Основною задачею варіаційного обчислення є дослідження на екстремум функціоналів, тобто знаходження таких функцій (з даного класу), які надають функціоналу найбільше чи найменше значення. Такі функції називають екстремалями.

Основна теорема варіаційного обчислення. Якщо функціонал $J(y(x))$ досягає в точці y мінімуму (максимуму), то диференціал функціоналу, якщо він існує, у цій точці дорівнює нулю

$$dJ = J'_y dy = 0,$$

$$J(y) = \int_{x_1}^{x_2} F(x, y, y')dx \to extr,$$

де J'_y – частинна похідна функціоналу J по y.

Диференційне рівняння Ейлера.

Нехай заданий функціонал (7.6), що залежить від функції однієї змінної *y(x)* та її похідної *y'(x)*:

$$J(y) = \int_{x_1}^{x_2} F(x, y, y')dx \tag{7.6}$$

з заданими граничними умовами (7.7)

$$\begin{cases} y(x_1) = y_1 \\ y(x_2) = y_2 \end{cases} \qquad (7.7)$$

де *F(x, y, y')* – функція, що диференціюється по *y* і y'.

Необхідною умовою існування екстремуму функціоналу (7.6) є виконання умов:

$$F_y - \frac{dF_{y'}}{dx} = 0, \qquad (7.8)$$

$$\frac{\partial F}{\partial y'}\delta y = 0, \text{ для } x_1, x_2 \qquad (7.9)$$

Рівняння (7.8) називається диференціальним рівнянням Ейлера-Лагранжа, а рівняння (7.9) – умовою трансверсальності. При цьому необхідною умовою існування мінімуму (максимуму) функціоналу є те, що на оптимальній траєкторії друга варіація функціонала має бути $\frac{\partial^2 F}{\partial y^2} \geq 0$ ($\leq$0).

Приклад 1. Знайти екстремаль функціоналу

$$J = \int_0^2 \left(\frac{1}{2}\dot{y}^2 + y\dot{y} + \dot{y} + y\right)dx \to extr \qquad \begin{cases} y(0) = 1 \\ y(2) = -2 \end{cases}.$$

Маємо

$$F = \frac{1}{2}\dot{y}^2 + y\dot{y} + \dot{y} + y,$$
$$F_y = 1 + \dot{y},$$
$$F_{\dot{y}} = 1 + \dot{y} + y.$$

Рівняння Ейлера для цього функціоналу матиме вигляд

$$F_y - \frac{dF_{\dot{y}}}{dx} = 1 + \dot{y} - \ddot{y} - \dot{y} = 1 - \ddot{y} = 0. \qquad (7.10)$$

Розв’язавши диференціальне рівняння (7.10), отримаємо

$$y(x) = C_1 + C_2 x + \frac{x^2}{2},$$
$$\dot{y}(x) = C_2 + x.$$

Для визначення C_1 і C_2 скористаємося граничними умовами:

$$y(0) = C_1 = 1,$$

$$y(2) = C_1 + 2C_2 + 2 = -2.$$

Звідки отримуємо

$$C_1 = 1, C_2 = \frac{-5}{2} \ .$$

Приклад 2. Розглянемо цю задачу але для випадку коли граничні умови не визначені: знайти екстремаль функціоналу

$$J = \int_0^2 \left(\frac{1}{2}\dot{y}^2 + y\dot{y} + \dot{y} + y\right) dx \to extr$$

Тепер для визначення C_1 та C_2 скористаємося умовами трансверсальності. Маємо для x=0

$$F_{\dot{y}} = \dot{y} + y + 1 = C_2 + x + C_1 + C_2 x + \frac{x^2}{2} = C_2 + C_1 + 1 = 0,$$

для $x = 2$

$$F_{\dot{y}} = \dot{y} + y + 1 = C_2 + x + C_1 + C_2 x + \frac{x^2}{2} =$$
$$= C_2 + 2 + C_1 + 2C_2 + 2 + 1 = 0.$$

Тепер із системи спільних рівнянь

$$C_2 + C_1 =–1, \qquad 3C_2 + C_1 = -5$$

можна знайти C_1 и C_2.

Приклад 3. Дослідження екстремалі функціоналу в середовищі програми MATLAB за допомогою пакету символьних розрахунків Symbolic Math Toolbox.

Завдання: знайти екстремаль функціоналу і побудувати графік для:

$$J = \int_{-1}^{1} \left(\frac{1}{2}\dot{y}^2 + y\dot{y} + \dot{y} + y\right) dx \rightarrow extr, \qquad \begin{matrix} y(-1) = 1 \\ y(1) = -2 \end{matrix}.$$

У цьому прикладі підінтегральна функція *F(x, y, y')* є функцією загального виду, тому складемо рівняння Ейлера-Лагранжа і вирішимо його засобами MATLAB. Потім збудуємо графік рішення.

Очистимо пам'ять. Надрукуємо заголовок розв'язуваної задачі. Опишемо символьні змінні. Для вирішення рівняння Ейлера використовуємо прийняті в MATLAB позначення похідних: *Dy* для y' і *D2y* для y''. Аргумент позначимо *x*, а функцію – *y* (ліст. 6.1).

Лістинг 7.1 – Постановка задачі

```
     clear all % очищення пам’яті
     clc % очищення командного вікна
     disp(‘Розв’язання задачі ‘) % відображення назви за-
вдання
     syms x y  Dy D2y % опис символьних змінних
     % введення підінтегральної функції та граничних умов
     F=y*Dy+Dy+y+Dy^2/2; % тут має бути ваша підінтегра-
льна функція
     % граничні умови зліва
     x1=-1;
     y1=1;
     % граничні умови праворуч
     x2=1;
     y2=2;
     % Друкуємо їх.
     disp (‘ Вихідні дані: ‘)
     fprintf(‘Підінтегральна  функція  F(x,y,y’’)=%s\n’,
char(F))
     fprintf(‘%s\n’,’Граничні умови зліва:’ )
     fprintf(‘%s\n %e\n %s\n %e\n’, ‘x1=‘,x1,  ‘y1=‘,y1)
     x1=
```

Кінець лістингу 7.1

```
 -1.000000e+00
 y1=
 1.000000e+00
fprintf('%s\n','Граничні умови праворуч:' )
fprintf('%s\n %e\n %s\n %e\n', 'x2=',x2,  'y2=',y2)
x2=
 1.000000e+00
 y2=
 2.000000e+00
```

Виведемо диференціальне рівняння Ейлера-Лагранжа. Знайдемо часткові похідні Fy і Fy'. Надрукуємо їх (ліст. 6.2).

Лістинг 7.2 – Виведення диференціального рівняння

```
dFdy=diff(F,y); % обчисл. похідної функції F по y
dFdy1=diff(F,Dy); % обчисл. похідної функції F по y'
fprintf('Fy=%s\n',char(dFdy))
fprintf('Fy''=%s\n',char(dFdy1))

Fy=Dy + 1
Fy'=Dy + y + 1
```

До рівняння Ейлера входить повна похідна d/dx. Обчислимо її за звичайною формулою диференціювання складної функції:

$$\frac{dF_{y'}}{dx} = \frac{\partial F_{y'}}{\partial x} + \frac{\partial F_{y'}}{\partial y} y' + \frac{\partial F_{y'}}{\partial y'} y''.$$

Надрукуємо її (ліст. 7.3).

Лістинг 7.3 – Виведення складної функції

```
d_dFdy1_dx=diff(dFdy1,x); %Fy'/dx
d_dFdy1_dy=diff(dFdy1,y); % Fy'/dy
d_dFdy1_dy1=diff(dFdy1,Dy); % Fy'y'
dFy1dx=d_dFdy1_dx+d_dFdy1_dy*Dy+d_dFdy1_dy1*D2y;
fprintf('dFy''/dx=%s\n',char(dFy1dx))
dFy'/dx=D2y + Dy
```

Складемо ліву частину диференціального рівняння Ейлера та спростимо її. Перетворимо символьну змінну `Euler` на рядок (ліст. 7.4). Для розв'язання рівняння Ейлера скористаємось командою `dsolve`, що дозволяє шукати рішення диференціальних рівнянь (ліст. 7.5). Сформуємо рівняння для граничних умов. Підставимо у знайдене аналітичне рішення `Sol` граничні точки `x1` та `x2` та прирівняємо їх відповідно `y1` та `y2` (ліст. 7.6).

Лістинг 7.4 – Перетворення символьної змінної на рядок

```
Euler=dFdy-dFy1dx; % рівняння Ейлера
deqEuler=[char(Euler) '=0']; % в рядок додали =0
fprintf('Рівняння Ейлера:\n%s\n',deqEuler)
Рівняння Ейлера:
1 - D2y=0
```

Лістинг 7.5 – Пошук рішення диференціальних рівнянь

```
% Вирішуємо рівняння Ейлера
Sol=dsolve(deqEuler,'x');
% немає рішень або більше одного
if length(Sol)~=1
   error('Немає рішень або більше одного рішення!');
else
   disp(' Загальне рішення рівняння Ейлера:');
fprintf('y(x)=%s\n',char(Sol))
end
Загальне рішення рівняння Ейлера:
y(x)=C2 + (x*(C1 + x))/2
```

Лістинг 7.6 – Рівняння для граничних умов

```
SolLeft=subs(Sol,x,x1) ;% підставили x1
SolRight=subs(Sol,x,x2) ;% підставили x2
EqLeft=[char(SolLeft) '=' char(sym(y1))] ;% =y1
EqRight=[char(SolRight) '=' char(sym(y2))] ;% =y2
disp('Рівняння для граничних умов:')
fprintf('%s\n',EqLeft,EqRight)
Рівняння для граничних умов:
C2 - C1/2 + 1/2=1
C1/2 + C2 + 1/2=2
```

Розв’язуємо отриману систему кінцевих рівнянь – знаходимо значення довільних постійних C1 і C2. Привласнюємо знайдені рішення символьним константам, отриманим під час вирішення диференціального рівняння та обчислюємо аналітичне рішення sol21 (ліст. 7.7). Таке обчислення зводиться до того що, що у нього будуть підставлені знайдені значення констант C1 = 1, C2 = 1. Друкуємо знайдене рівняння екстремалі (рис. 7.1):

$$y(x) = 0.5 * x * (x + 1.0) + 1.0.$$

Лістинг 7.7 – Побудова графіку

```
% побудова графіку
xp1=linspace(x1,x2); % задаємо масив абсцис
y21=subs(Sol21,x,xp1); % обчислюємо ординати
figure;
   set(gcf,’color’,[1 1 1]);
   lns1=line(xp1, y21);
   set(lns1,’LineWidth’,[2.5],’color’,[0    0    0],
’LineStyle’,’-’);
   set(gca,’GridLineStyle’,’-’,’LineWidth’,[1.5],
’FontAngle’,’normal’,’FontName’,’Times New Roman Cyr’,...
   ‘FontSize’,14);
   xlabel(‘x’);
   ylabel(‘y’);
   grid on;
```

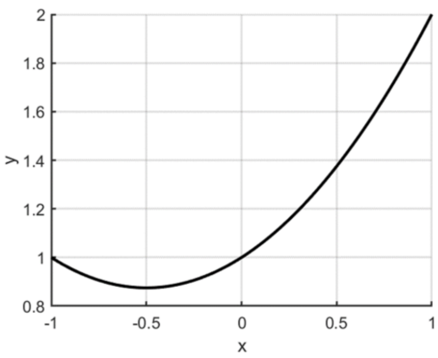


Рисунок 7.1 – Графік екстремалі

Для дослідження екстремалі функціоналу в середовищі програми MATLAB за допомогою пакету символьних розрахунків Symbolic Math Toolbox наводимо повний лістинг програми (ліст. 7.8).

Лістинг 7.8 – Дослідження екстремалі функціоналу

```
clear all % очищення пам'яті
clc % очищення командного вікна
disp('Розв'язання задачі ') % відображення назви за-
вдання
syms x y  Dy D2y % опис символьних змінних
% введення субінтегральної функції та граничних умов
F=Dy^2+4*y^2-8*x*y+2*x^2; % тут має бути ваша піді-
нтегральна функція
% граничні умови зліва
x1=-1;
y1=1;
% граничні умови праворуч
x2=1;
y2=2;
% Друкуємо їх.
disp (' Вихідні дані: ')
fprintf('Підінтегр. ф-ція F(x,y,y'')=%s\n',char(F))
fprintf('%s\n', Граничні умови зліва:' )
fprintf('%s\n %e\n %s\n %e\n', 'x1=',x1,  'y1=',y1)
fprintf('%s\n','Граничні умови праворуч:' )
fprintf('%s\n %e\n %s\n %e\n', 'x2=',x2,  'y2=',y2)
% Починаємо виведення диференціал. рівняння Ейлера.
% Знайдемо частин. похідні Fy, Fy' та надрукуємо їх.
dFdy=diff(F,y); % обчисл. похідної функції F по y
dFdy1=diff(F,Dy); % обчисл. похідної функції F по y'
fprintf('Fy=%s\n',char(dFdy))
fprintf('Fy''=%s\n',char(dFdy1))
% До рівняння Ейлера входить повна похідна dFy'/dx.
% Обчислимо її
d_dFdy1_dx=diff(dFdy1,x); %Fy'/dx
d_dFdy1_dy=diff(dFdy1,y); % Fy'/dy
d_dFdy1_dy1=diff(dFdy1,Dy); % Fy'y'
dFy1dx=d_dFdy1_dx+d_dFdy1_dy*Dy+d_dFdy1_dy1*D2y;
fprintf('dFy''/dx=%s\n',char(dFy1dx))
Euler=dFdy-dFy1dx; % рівняння Ейлера
deqEuler=[char(Euler) '=0']; % в рядок додали =0
```

Продовження лістингу 7.8

```
fprintf('Рівняння Ейлера:\n%s\n',deqEuler)
% Вирішуємо рівняння Ейлера
Sol=dsolve(deqEuler,'x');
if length(Sol)~=1 % немає рішень або більше одного
    error('Немає рішень або більше одного рі-
шення!');
else
disp(' Загальне рішення рівняння Ейлера:') ;
fprintf('y(x)=%s\n',char(Sol))
end
% Сформуємо тепер рівняння для граничних умов.
% Підставимо у знайдене аналітичне рішення sol
% граничні точки x1 и x2, та прирівняємо їх відпо-
відно %y1 та y2.
SolLeft=subs(Sol,x,x1) ;% підставили x1
SolRight=subs(Sol,x,x2) ;% підставили x2
EqLeft=[char(SolLeft) '=' char(sym(y1))] ;% =y1
EqRight=[char(SolRight) '=' char(sym(y2))] ;% =y2
disp('Рівняння для граничних умов ')
fprintf('%s\n',EqLeft,EqRight)
% Розв'язуємо отриману систему скінченних рівнянь і
знаходимо значення довільних констант C1 і C2.
% Призначаємо знайдені розв'язки символьним конста-
нтам, що отримані при розв'язані диференціального рів-
няння. Тепер обчислимо аналітичний розв'язок sol21.
% Такий розрахунок зводиться до того, що в нього
будуть підставлятися знайдені значення констант C1 і C2.
% Друкуємо знайдене рівняння екстремалі.
 [C1,C2] = solve(SolLeft == char(sym(y1)),SolRight
==char(sym(y2)))
Sol21=vpa(eval(Sol),14); % підставили C1,C2
disp('Рівняння екстремалі:')
fprintf('y (x)=%s\n',char(Sol21))
% побудова графику
xp1=linspace(x1,x2); % задаємо масив абсцис
y21=subs(Sol21,x,xp1); % обчислюємо ординати
figure;
   set(gcf,'color',[1 1 1]);
   lns1=line(xp1, y21);
   set(lns1,'LineWidth',[2.5],'color',[0   0   0],
'LineStyle','-');
```

Кінець лістингу 7.8

```
        set(gca,'GridLineStyle','-','LineWidth',[1.5],
'FontAngle','normal','FontName','Times    New    Roman
Cyr',...
        'FontSize',14);
        xlabel('x');
        ylabel('y');
        grid on
```

7.2 Порядок виконання роботи

1. По даним таблиці 6.1 задати підінтегральну функцію *F(x, y, y')* функціоналу, що досліджується, і граничні умови.
2. Отримати диференційне рівняння Ейлера.
3. Знайти рішення рівняння Ейлера.
4. Сформувати рівняння для граничних умов і знайти значення констант C1 і C2.
5. Побудувати графік екстремалі.

7.3 Зміст звіту

1. Титульний лист.
2. Мета роботи.
3. Тексти створених М-файлів для проведення досліджень і результати їх роботи.
4. Стислі відповіді на контрольні питання.

7.4 Контрольні питання

1. Що таке функціонал? Чим він відрізняється від функції?
2. У чому полягає основна задача варіаційного обчислення?
3. Які класичні задачі варіаційного обчислення ви знаєте?
4. Що називається класом функцій? Які класи ви знаєте?
5. Що називається варіацією функції?
6. Сформулюйте основну теорему варіаційного обчислення.
7. Приведіть диференційне рівняння Ейлера.

Таблиця 7.1 – Варіанти для індивідуального завдання

Варіант	*F(x, y, y')*	Граничні умови
1	$J = \int_{-1}^{1} (\dot{y}^2 - 4y^2 + 2xy - x^2)dx$	$y(-1) = -1$ $y(1) = 2$
2	$J = \int_{-1}^{1} (\dot{y}^2 + 4y^2 - 8xy + 2x^2)dx$	$y(-1) = 2$ $y(1) = 4$
3	$J = \int_{-1}^{1} (\dot{y}^2 + 4y^2 + 4x^2y + xcosx)dx$	$y(-1) = 2$ $y(1) = 0.5$
4	$J = \int_{0}^{2} (\dot{y}^2 - 4\dot{y}cos2x + 4x^2y + 5sin3x)dx$	$y(0) = 2$ $y(2) = -3$
5	$J = \int_{0}^{2} (\dot{y}^2 + 9y^2 + 2xy + xsinx)dx$	$y(0) = 1$ $y(2) = 2$
6	$J = \int_{0}^{2} (\dot{y}^2 + 9y^2 + 2xy - xsinx)dx$	$y(0) = 1$ $y(2) = 2$
7	$J = \int_{1}^{3} \left(\dot{y}^2 - \frac{4\dot{y}}{x} + xsinx\right) dx$	$y(1) = 1$ $y(3) = -2$
8	$J = \int_{-1}^{1} (\dot{y}^2 - 2\dot{y}e^x + cosx)dx$	$y(-1) = 2$ $y(1) = 3$
9	$J = \int_{0}^{1} (\dot{y}^2 - 9y^2 + 2ysinx - x^2e^x)dx$	$y(0) = 1$ $y(1) = -1$
10	$J = \int_{-1}^{1} (\dot{y}^2 + y^2 + 4ye^x - xsinx)dx$	$y(-1) = 1$ $y(1) = 3$

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Теорія автоматичного керування : навч. посібник / П. В. Леонтьєв та ін. ; за заг. ред. П. В. Леонтьєва. – Суми : Сумський державний університет, 2024. – 296 с. URL: https://essuir.sumdu.edu.ua/handle/123456789/97189

2. Хісматулін В. Ш. Теорія оптимальних систем автоматичного керування : навч. посібник / В. Ш. Хісматулін, О. О. Сосунов, В. О. Сотник – Харків : УкрДУЗТ, 2022. – 229 с. URL: http://lib.kart.edu.ua/handle/123456789/13082

3. Системи автоматичного керування технологічними комплексами : навч. посібник / А. М. Сільвестров, М. Я. Островерхов, О. В. Шефер, Н. А. Ладік, Д. К. Зіменков. – Київ : КПІ ім. Ігоря Сікорського, 2023. – 466 с. URL: https://ela.kpi.ua/handle/123456789/73320

5. Yefymenko N., Kudermetov R. Dynamic model of vector motion and its application in spacecraft uniaxial orientation problems. Space Science and Technology, 30(4):24–33, 2024. DOI: 10.15407/knit2024.04.024

6. Yefymenko M. V., Kudermetov R. K. Terminal control of quadcopter spatial motion. Radio Electronics, Computer Science, Control, 2(73):232–243, 2025. DOI: 10.15588/1607-3274-2025-2-20

7. Яровий А. Т., Страхов Є. М. Методи оптимізації та варіаційне числення : навч.-метод. посібник / А. Т. Яровий, Є. М. Страхов. – Одеса : ФОП «Сухачов», Київ : Видавництво «Освіта України», 2017 – 136 с. URL: https://dspace.onu.edu.ua/handle/123456789/10394