

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ДЕРЖАВНИЙ ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД  
«УКРАЇНСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ НАУКИ І ТЕХНОЛОГІЙ»**

**МЕТОДИЧНІ ВКАЗІВКИ  
ДО ПРАКТИЧНИХ ЗАНЯТЬ З ДИСЦИПЛІНИ  
“СУЧАСНІ ПРОБЛЕМИ АВТОМАТИЗОВАНОГО УПРАВЛІННЯ”  
ДЛЯ СТУДЕНТІВ СПЕЦІАЛЬНОСТІ 174 – АВТОМАТИЗАЦІЯ,  
КОМП'ЮТЕРНО-ІНТЕГРОВАНІ ТЕХНОЛОГІЇ ТА РОБОТОТЕХНІКА**

**Дніпро УДХТУ 2024**

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ДЕРЖАВНИЙ ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД  
«УКРАЇНСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ НАУКИ І ТЕХНОЛОГІЙ»

МЕТОДИЧНІ ВКАЗІВКИ  
ДО ПРАКТИЧНИХ ЗАНЯТЬ З ДИСЦИПЛІНИ  
“ СУЧАСНІ ПРОБЛЕМИ АВТОМАТИЗОВАНОГО УПРАВЛІННЯ ”  
ДЛЯ СТУДЕНТІВ СПЕЦІАЛЬНОСТІ 174 – АВТОМАТИЗАЦІЯ,  
КОМП’ЮТЕРНО-ІНТЕГРОВАНІ ТЕХНОЛОГІЇ ТА РОБОТОТЕХНІКА

Затверджено на засіданні кафедри  
комп’ютерно-інтегрованих  
технологій і метрології.  
Протокол № від грудня 2023 р.

Дніпро УДХТУ 2024

Методичні вказівки до практичних занять з дисципліни «Сучасні проблеми автоматизованого управління» для студентів спеціальності 174 – Автоматизація, комп'ютерно-інтегровані технології та робототехніка / Укл.: Г. І.Манко. – Дніпро: УДХТУ, 2024. – 72 с.

Укладач Г.І. Манко, канд. техн. наук,

Відповідальний за випуск О. П. Мисов, канд. техн. наук

Навчальне видання

Методичні вказівки  
до практичних занять  
з дисципліни «Сучасні проблеми автоматизованого управління»  
для студентів спеціальності 174 – Автоматизація, комп'ютерно-інтегровані  
технології та робототехніка

Укладач: МАНКО Геннадій Іванович,

Редактор Л.М. Тонкошкур  
Коректор Л.Я. Гоцуцова

Підп. до друку \_\_\_\_\_. Формат 60х84 1/16. Папір ксерокс. Друк різнограф.  
Умовн.-друк. арк. \_\_\_\_\_ Облік. – вид. арк. \_\_\_\_\_ Тираж \_\_\_\_\_ прим. Зам. № \_\_\_\_\_  
Свідотство ДК №303 від 27.12.2000.

---

УДХТУ, 49005, Дніпропетровськ-5, пр-т Гагаріна, 8

---

Видавничо-поліграфічний комплекс ІнКомЦентру

## **ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ**

LQR – лінійно-квадратичний регулятор

LQG – лінійно-квадратичний гаусівський регулятор

АЧХ – амплітудно-частотна характеристика;

АФЧХ – амплітудно-фазова частотна характеристика;

ЛКМ – ліва клавіша миші;

ЛАЧХ – логарифмічна амплітудно-частотна характеристика;

ЛКГ – лінійно-квадратичний гаусівський регулятор

ЛФЧХ – логарифмічна фазо-частотна характеристика;

ПКМ – права клавіша миші;

ПФ – передатна функція;

САУ – система автоматичного управління;

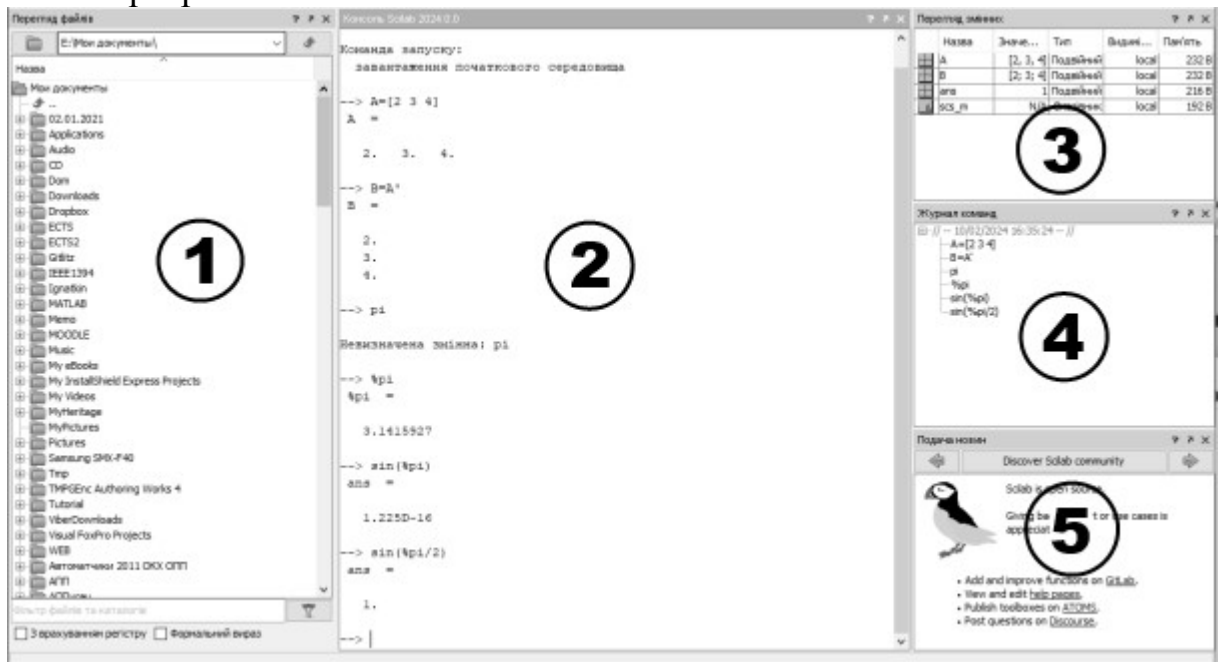
# 1 КОРОТКИЙ ОПИС ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

## 1.1 Теоретичні відомості

### 1.1.1 Пакет Scilab

Пакет Scilab призначений для виконання інженерних і наукових обчислень та математичного моделювання систем.

Вікно програми містить п'ять областей.



1 – Перегляд файлів. Область для вибору файлів на зразок провідника.

2 – Консоль. Основна область, призначена для введення команд та одержання результатів.

3 – Перегляд змінних. Відображення усіх змінних, що використовуються у поточній сесії.

4 – Журнал команд. Список команд, що були введені у поточній сесії.

5 – Подача новин сайту розробника.

Робота Scilab можлива в двох режимах: командному режимі і режимі виконання сценаріїв.

### 1.1.2 Робота в командному режимі

Робота системи в командному режимі нагадує роботу калькулятора. У вікні «Консоль» можна відразу після маркера --> вводити початкові дані і команди за допомогою вбудованого текстового редактора. Введення завершується натисненням клавіші ENTER.

Результати обчислень виводяться з нового рядка (без маркера -->).

Для блокування виведення результатів обчислень деякого вираження його потрібно закінчити знаком «;» (крапка з комою). Це буває зручним для приховання результатів проміжних обчислень.

Для очищення вікна команд слід виконати команду `clc`, натиснути функціональну клавішу <F2> або вибрати опцію «Спорожніти консоль» меню «Змінні».

Як знак привласнення використовується знак рівності «=». Наприклад, введення рядка  $x = 1$  означає оголошення і ініціалізацію змінної  $x$ .

Правила використання імен:

- а) вони можуть складатися з будь-яких латинських і грецьких літер, цифр;
- б) великі і малі літери розрізняються;
- в) починаються тільки з літери;
- г). не можуть містити проміжків;
- д) не можуть збігатися з іменами вбудованих чи визначених користувачем функцій.

Система Scilab має *системні змінні*, які використовуються самою системою і входять до складу зарезервованих:

| Змінна | Призначення                                                                |
|--------|----------------------------------------------------------------------------|
| %i     | Уявна одиниця (корень з $-1$ )                                             |
| %pi    | Число $\pi$                                                                |
| %e     | Число $e$ (основа натурального логарифма, 2,7182818)                       |
| %eps   | Похибка операцій над числами з плаваючою крапкою ( $2,22 \cdot 10^{-16}$ ) |
| %inf   | Системна нескінченність                                                    |
| %t     | Логічне "істинне" (true)                                                   |
| %f     | Логічне "хибне" (false)                                                    |
| %nan   | Невизначений результат ( $0/0$ , $\infty/\infty$ тощо)                     |

Ці змінні можна використати в математичних виразах.

Вбудовані функції записуються малими літерами, а їх аргументи наводяться в круглих дужках, наприклад,  $\sin(x)$ .

У одному рядку команди можна визначити декілька змінних і/або обчислити декілька виразів, наприклад:

|                        |                                  |
|------------------------|----------------------------------|
| Вводиться з клавіатури | --> a=1, b=5, c=a+b              |
| Відповідь системи      | a=<br>1.<br>b=<br>5.<br>c=<br>6. |

Для виведення значення конкретної змінної досить ввести її ім'я, наприклад:

|                        |          |
|------------------------|----------|
| Вводиться з клавіатури | -->a     |
| Відповідь системи      | a=<br>1. |

Якщо виведення результату обчислення тільки що введенного вираження не заблоковане і не вказана змінна, значення якої необхідно вивести, то система

SCILAB сама призначає таку змінну з ім'ям ans, привласнює їй значення останнього вираження і виводить її як результат обчислень, наприклад:

|                        |               |
|------------------------|---------------|
| Вводиться з клавіатури | -->sin(%pi/2) |
| Відповідь системи      | ans=<br>1.    |

### 1.1.3 Типи даних

Типи даних кодуються числами, як показано нижче.

| Код типу | Опис типу                                                 |
|----------|-----------------------------------------------------------|
| 1        | матриця дійсних чи комплексних значень подвоєної точності |
| 2        | матриця поліномів                                         |
| 4        | матриця логічних значень                                  |
| 5        | розріджена матриця                                        |
| 6        | розріджена матриця логічних значень                       |
| 7        | розріджена матриця Matlab'у                               |
| 8        | матриця цілих значень                                     |
| 9        | матриця графічних дескрипторів                            |
| 10       | матриця символьних рядків                                 |
| 13       | компільована функція (Scilab code)                        |
| 14       | бібліотека функцій                                        |
| 15       | список (list)                                             |

Код типу даних  $x$  можна отримати за допомогою функції:  $i = type(x)$ .

### 1.1.4 Числа, матриці і списки

Система Scilab оптимізована для проведення обчислень з векторами, матрицями і списками. Більше того, в системі за умовчанням передбачається, що кожна змінна – це вектор або матриця. Наприклад, якщо задане  $X = 1$ , то система сприймає це як завдання вектору з єдиним елементом, значення якого дорівнює 1. У свою чергу, вектор – це матриця, число рядків або стовпців якої дорівнює 1.

Комплексні числа записуються в алгебраїчній формі з уявною одиницею, яка позначається %i у вигляді  $a+b*\%i$ , де  $a$  і  $b$  – відповідно дійсна й уявна частини числа, наприклад:  $(3+5*\%i) + 2*\%i$ .

Для завдання вектору з великим числом елементів, їх значення потрібно перерахувати в прямокутних дужках, використовуючи як роздільник пропуск або кому. Завдання матриць здійснюється аналогічно, для розділення рядків використовується знак «;» (крапка з комою) Наприклад:

|                                |                       |
|--------------------------------|-----------------------|
| Завдання вектора з 4 елементів | -->X =[1 2 3 4]       |
| Відповідь системи              | X =<br>1. 2. 3. 4.    |
| Завдання матриці 2×2           | -->Y =[1 2; 4 5]      |
| Відповідь системи              | Y =<br>1. 2.<br>4. 5. |

У Scilab передбачена велика кількість операцій, які можна здійснювати з матрицями і векторами. Серед них:

- складання, віднімання (+,-);
- множення (\*);
- обернення (**inv**); оберненою по відношенню до матриці **M** називається така матриця, яка при її множенні на матрицю **M** дає одиничну матрицю;
- ділення зліва направо (/) і справа наліво (\);
- піднесення до степеня (^);
- транспонування (');
- створення нижньої трикутної матриці **A**: **tril(A)**;
- створення верхньої трикутної матриці **A**: **triu(A)**;
- формування матриці одиниць заданого розміру  $n \times m$ : **ones(n, m)**;
- формування матриці нулів заданого розміру  $n \times m$ : **zeros(n, m)**;
- формування матриці з одиницями на діагоналі і нулями поза діагоналлю: **eye(m,n)**;
- формування матриці з випадковими елементами, рівномірно розподіленими на відрізку [0,1]: **rand(m,n)**;
- створення діагональної матриці, елементами головної діагоналі якої є значення вектора **A**; всі інші її елементи дорівнюють "0": **daig(A)**;
- перетворення матриці **A** в матрицю з іншим розміром: **matrix(A, m, n)**;
- визначення максимального елемента матриці **A**: **max(A)**;
- визначення мінімального елемента матриці **A**: **min(A)**;
- власні числа матриці: **eigs(A)**;
- визначник матриці: **det(A)**.

Символ \$, якщо використовується як індекс масиву, позначає його останній елемент.

**Списки** містять елементи, які є довільними об'єктами системи. Для створення списку використовується команда **list**, яка створює список з елементами  $a_1, \dots, a_n$ :

$$list(a1, \dots, an).$$

Операції зі списками:

- а) вставка елемента  $e$  на місце індексу  $i$ :  $L(i) = e$ ;
- б) додавання елемента  $e$  в хвіст:  $L(\$+1) = e$ ;
- в) додавання елемента  $e$  на початок:  $L(0) = e$ ;
- г) видалення  $i$ -того елемента:  $L(i) = null()$ .

### 1.1.5 Математичні функції

У Scilab передбачена наявність великого числа вбудованих математичних функцій. Серед них:

- а) елементарні функції:

| Математичні функції     |                              |
|-------------------------|------------------------------|
| <b>abs(x)</b>           | модуль x                     |
| <b>sqrt(x)</b>          | корінь квадратний з x        |
| Тригонометричні функції |                              |
| <b>sin(x)</b>           | синус                        |
| <b>cos(x)</b>           | косинус                      |
| <b>tan(x)</b>           | тангенс                      |
| <b>ctg(x)</b>           | котангенс                    |
| <b>asin(x)</b>          | арксинус                     |
| <b>acos(x)</b>          | арккосинус                   |
| Експоненціальні функції |                              |
| <b>exp(x)</b>           | експонента числа x           |
| <b>Log(x)</b>           | натуральний логарифм числа x |

#### б) функції комплексного аргументу

Практично усі перелічені вище елементарні математичні функції обчислюються при комплексних значеннях аргументу і набувають в результаті цього комплексних значень результату. Наприклад, функція **sqrt** обчислює, на відміну від інших мов програмування, квадратний корінь з негативного аргументу, а функція **abs** при комплексному значенні аргументу обчислює модуль комплексного числа.

У Scilab є декілька функцій, розрахованих на комплексний аргумент :

**real(Z)** - виділяє дійсну частину комплексного аргументу Z;

**imag(Z)** - виділяє уявну частину комплексного аргументу;

**angle(Z)** - обчислює значення аргументу комплексного числа Z (у радіанах в діапазоні від  $-\pi$  до  $+\pi$ );

**conj(Z)** - видає число, комплексно зв'язане відносно Z.

### 1.1.6 Оператори «:» (двокрапка) і «.» (крапка)

Оператор «:» (двокрапка) використовується для створення впорядкованої послідовності чисел з рівновіддаленими значеннями. Формат запису цього оператора:

[Початкове\_значення]:[Крок]:[Кінцеве\_значення]

Якщо крок не заданий, то він приймається рівним 1 або  $-1$  залежно від того, більше кінцеве значення початкового значення або менше.

При його використанні створюється вектор:

|                                             |                         |
|---------------------------------------------|-------------------------|
| Задається послідовність <b>k</b> з кроком 2 | -->k = 2:2:10           |
| Відповідь системи                           | k =<br>2. 4. 6. 8. 10.  |
| Задається послідовність <b>m</b> з кроком 1 | -->m = 0:2              |
| Відповідь системи                           | m =<br>0. 1. 2.         |
| Використання послідовності <b>m</b>         | -->s=sin(m)             |
| Відповідь системи                           | s =<br>0. 0.8415 0.9093 |

При необхідності здійснення почленного множення (чи ділення) елементів одного масиву на елементи іншого масиву (тієї же розмірності) використовуються відповідно оператори  $(.*)$  і  $(./)$ . Наприклад:

|                                         |                           |
|-----------------------------------------|---------------------------|
| Складання векторів <b>m, s</b>          | -->m+s                    |
| Відповідь системи                       | ans =<br>0. 1.8415 2.9093 |
| Почленне множення елементів <b>m, s</b> | -->m.*s                   |
| Відповідь системи                       | j =<br>0. 0.8415 1.8186   |

Для здійснення операції піднесення елементів масиву до степеня використовується операція «.^».

### 1.1.7 Побудова графіків

Графіки в Scilab завжди виводяться в окреме графічне вікно, котре називають фігурою (figure). Перше вікно формується автоматично і має номер 0. Для створення додаткових вікон використовується команда **figure(n)**, де n – номер вікна.

У одному графічному вікні, але на окремих графічних полях можна побудувати декілька графіків, використовуючи процедуру

*subplot(m,n,p).*

Тут m вказує, на скільки частин розділяється графічне вікно по вертикалі, n – по горизонталі, а p – номер підвікна, в якому будуватиметься графік. При цьому підвікна нумеруються зліва направо зверху вниз.

Для побудови графіків функцій може використовуватися функція **plot**.

Ця функція має різні форми запису залежно від набору вхідних аргументів. Якщо **y** – вектор, функція **plot(y)** робить побудову шматково-лінійного графіку елементів **y** залежно від індексу елементів **y**. Якщо функція **plot(x,y)** має два векторні аргументи однакової розмірності, то буде побудований графік **y** залежно від **x**.

Функція **plot** може мати декілька пар аргументів **x–y**. В цьому випадку вона зображує декілька функцій **y(x)** кривими різного кольору, використовуючи зумовлений (але змінюваний користувачем) список кольорів. Наприклад, **plot(x1, y1, x2, y2, x3, y3)**– вибудовуються три графіки функції : **y1(x1)**, **y2(x2)**, **y3(x3)**.

Також є можливість визначати колір, стиль лінії і маркери за допомогою ще одного аргументу функції **plot**:

`plot(x, y[, 'color_style_marker'] ),`

де **color\_style\_marker**– це символічний рядок (обмежений поодинокими лапками), що характеризує колір, стиль лінії і тип маркера:

-символи визначення кольору: 'c', 'm', 'y', 'r', 'g', 'b', 'w', 'k'. Вони відповідають кольорам cyan, magenta, yellow, red, green, blue, white, black;

- позначення стилю лінії: суцільна '-', пунктирна '--', штрихова ':', штрихпунктирна '-.';

- типи маркера: '+', 'o', '\*', 'x'.

Наприклад, вираження: `plot(x, y, 'r--o')` креслить червону пунктирну криву і вставляє маркери у вигляді символу «о» в кожній точці даних.

Будь-яка з трьох перерахованих складових `color-style-marker` може бути пропущена. Якщо визначається тип маркера, а стиль лінії опущений, система SCILAB будує тільки маркери для цієї кривої.

На графік додатково можна нанести інформацію про криву (заголовок графіку), а також позначити координатні осі.

Заголовок графіку виводиться за допомогою процедури **title**. Якщо після звернення до процедури **plot** викликати **title** таким чином:

```
-->title('<текст>')
```

то над графіком з'явиться текст, записаний між апострофами в дужках. При цьому слід пам'ятати, що текст завжди повинен поміщатися в апострофи.

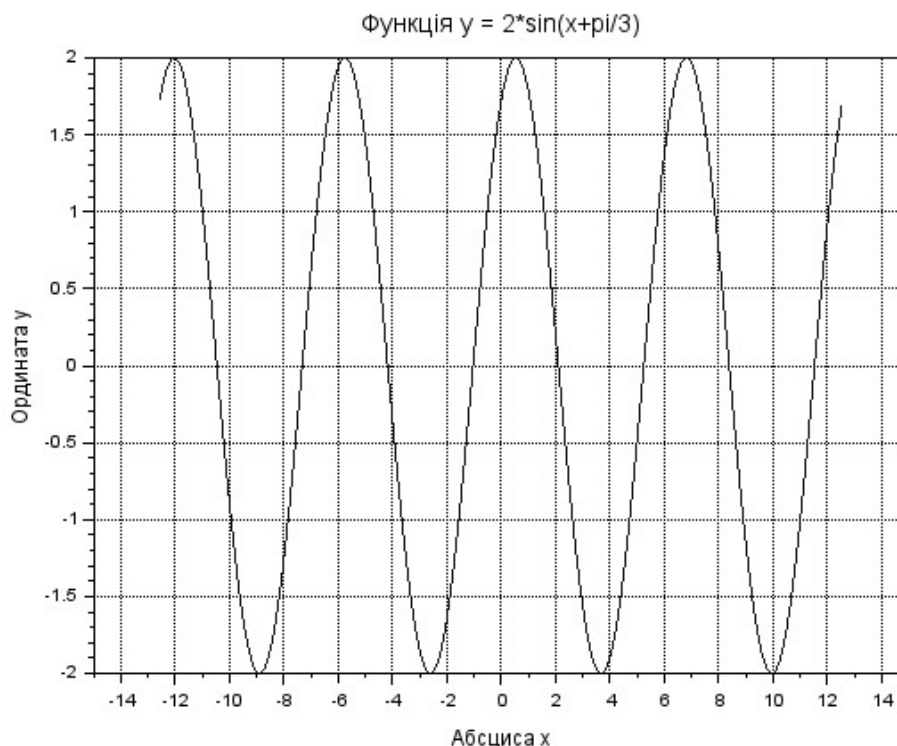
Аналогічно можна вивести пояснення до графіку, які розміщуються уздовж горизонтальної осі (функція **xlabel**) і уздовж вертикальної осі (функція **ylabel**).

Оператор **xgrid** додає сітку на поточний графік для кожної осі.

Наприклад, сукупність операторів

```
-->x = - 4*%pi: %pi/100 : 4*%pi;  
-->y = 2*sin(x+%pi/3);  
-->plot(x, y); xgrid  
-->title('Функція  $y = 2\sin(x+\pi/3)$  ');  
-->xlabel('Абсциса x'); ylabel('Ордината y');
```

приведе до оформлення поля фігури у вигляді:



Команда **legend** служить для виведення легенди графіка. Легенда потрібна, якщо на графіку є декілька ліній і треба показати, що означає кожна з них. Параметрами команди **legend** є символічні рядки, їх повинно бути стільки, скільки побудовано ліній. Рядки описують лінії графіку.

Scilab має також і функцію **clf** для повного очищення графічного вікна:

$$clf([<номер\_графічного\_вікна>])$$

### 1.1.8 Рішення системи лінійних рівнянь алгебри

Розглянемо систему лінійних рівнянь алгебри :

$$\begin{cases} a_{11}x_1 + a_{12}x_2 = b_1; \\ a_{21}x_1 + a_{22}x_2 = b_2. \end{cases}$$

У матричній формі ця система може бути записана у вигляді  $\mathbf{AX} = \mathbf{B}$ , де матриці  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{X}$  мають вигляд:

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}; \quad \mathbf{B} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}; \quad \mathbf{X} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}.$$

Рішення системи рівнянь в матричній формі запису представляється вираженням:

$$\mathbf{X} = \mathbf{A}^{-1}\mathbf{B},$$

де  $\mathbf{A}^{-1}$  – матриця, зворотна матриці  $\mathbf{A}$ .

У Scilab це рішення може бути знайдене при виконанні команд:

-->X=inv(A)\*B або -->X=A\B.

Scilab має вбудовану функцію **linsolve**, яка призначена для розв'язання рівнянь, представлених у вигляді  $\mathbf{A}*\mathbf{x}+\mathbf{b}=0$ . Синтаксис використання цієї функції:

-->x=linsolve(A,b)

### 1.1.9 Програмування в середовищі Scilab

Програми мовою Scilab мають два різновиди – так звані Script-файли (файли-сценарії, або програми, що управляють) і файли-функції (процедури). Обидва різновиди повинні мати розширення імені файлу «.sci» або «.sce». За допомогою сценаріїв оформлюють основні програми, які керують від початку до кінця організацією усього обчислювального процесу, і окремі частини основних програм (вони можуть бути записані у вигляді окремих файлів). Як файл-функції оформлюються окремі процедури і функції (тобто такі частини програми, які розраховані на неодноразове використання Script-файлами або іншими процедурами при змінених значеннях початкових параметрів і не можуть бути виконані без попереднього завдання значень змінних, які називають вхідними).

Для створення сценаріїв система має спеціальний засіб – редактор сценаріїв. Доступ до редактора здійснюється за допомогою опції «SciNotes»

меню «Програми» або натисканням на панелі інструментів кнопки .

Після цього з'являється вікно редактора, в якому і створюється сценарій.

Основні особливості запису тексту сценарію:

а) зазвичай кожен оператор записується в окремому рядку тексту програми. Закінчення введення рядка здійснюється натисненням клавіші <Enter>;

б) щоб запобігти виводу на екран результатів дії оператора програми, запис цього оператора в тексті програми повинен закінчуватися крапкою з комою «;»;

в) можна розміщувати декілька операторів в одному рядку, які відділяються один від одного комою або символом «;». Можна довгий оператор записувати в декілька рядків. При цьому попередній рядок оператора повинен закінчуватися трьома крапками«...»;

г) рядок програми, що розпочинається з символів «//», не виконується, цей рядок сприймається системою як коментар;

д) у програмах Scilab відсутній символ закінчення тексту програми;

е) у мові Scilab змінні не описуються і не оголошуються. Будь-яке нове ім'я, що з'являється в тексті програми при її виконанні, сприймається системою як ім'я матриці. Розмір цієї матриці встановлюється при попередньому введенні значень її елементів або визначається діями зі встановлення значень її елементів, описаними в попередніх операторах або процедурі;

ж) у мові Scilab неможливе використання матриці або змінної, в якій заздалегідь не введені або не вичислені значення її елементів (при виконанні програми з'явиться повідомлення про помилку – "Не визначена змінна:").

Виконання сценарію може проводитись у вікні редактора SciNotes використанням меню «Виконати». Тут є наступні опції:

«...файл без виведення»;

«...файл з виведенням»;

«...до курсора з виведенням».

Остання опція використовується при відлагодженні програм.

Файл-функція (процедура) повинна розпочинатися з рядка заголовка :

*function* [*<OutVar>*] = *<ім'я\_процедури>*(*<InVar>*),

де *<OutVar>*, *<InVar>* - відповідно, переліки вихідних і вхідних величин файл-функції. Якщо перелік вихідних величин (*OutVar*) отримає тільки один об'єкт (у загальному випадку – матрицю), то файл-функція є звичайною функцією (однієї або декількох змінних). Перший рядок в цьому випадку має вигляд:

*function* *<ім'я змінної>* = *<ім'я\_процедури>*(*<InVar>*).

Якщо ж в результаті виконання файл-функції мають бути визначені (вичислені) декілька об'єктів (матриць), така файл-функція є вже складніший об'єкт. Загальний вигляд першого рядка в цьому випадку стає таким:

*function* [*y1, y2, ..., yN*] = *<ім'я\_процедури>*(*<InVar>*),

тобто перелік вихідних величин має бути представлений як вектор-рядок з елементами *y1, y2, ..., yN* (усі вони можуть бути матрицями).

У простому випадку функції однієї змінної заголовок набере вигляду:

*function* *y* = *func*(*x*),

де *func* ім'я функції.

Після заголовка слідує тіло функції, в кінці обов'язково має бути оператор:

*endfunction*

Звернення до функції здійснюється так само, як і до будь-якої вбудованої функції системи, тобто з командного рядка. Проте функції, що зберігаються в окремих файлах, повинні бути заздалегідь завантажені в систему, наприклад, за допомогою команди

--> *exes* (*<им'я\_файла>*)

**Приклад:** скласти файл для обчислення функції

$$y = \ln(x + y) + \frac{x}{x + y}.$$

Процес створення файлу передбачає наступне.

1. Викликати меню «Програми» командного вікна «Консоль Scilab» і вибрати в ньому опцію «SciNotes».

2. У вікні текстового редактора, що з'явилося, набрати текст:

```
function y = func(x, z)
y=log(x+z)+x./(x+z);
endfunction
```

3. Зберегти цей текст у файлі під ім'ям func.sci.

Тепер можна користуватися цією функцією при розрахунках. Наприклад:

```
--> exec('func.sci')
--> y = func(1, 1)
y =
1.1931
--> x1=0:1:5;
--> x2=1:1:6;
--> y=func(x1, x2)
y =
0. 1.4319 2.0094 2.3745 2.6417 2.8524
```

При створенні файл-функцій необхідно пам'ятати деякі особливості:

а) з точки зору зони видимості змінні у файл-функції є *локальними*, тому імена змінних, використовувані у файл-функції, можуть не співпадати з іменами відповідних змінних при зверненні до цієї файл-функції;

б) з метою використання у файл-функції глобальної змінної (деякої змінної робочого простору Workspace) необхідно в діалоговому режимі (чи в Script-файлі) і файл-функції оголосити цю змінну з атрибутом **global**. Якщо в одному рядку оголошуються декілька змінних як глобальні, вони повинні відділятися одна від одної пропусками;

в) можливість використання файл-функції як для окремих чисел, так і для векторів і матриць обумовлена застосуванням в записі відповідного файлу замість звичайних знаків арифметичних дій їх аналогів з попередньою крапкою («.\*», «./», «.^» та ін.).

При створенні сценаріїв враховуються наступні особливості:

а) файли сценаріїв є незалежно (самостійно) виконуваними блоками операторів і команд;

б) усі використовувані змінні утворюють так званий робочий простір, який є загальним для усіх виконуваних Script-файлів; з цього виходить, що при виконанні декількох Script-файлів імена змінних в них мають бути погоджені, оскільки одне ім'я означає в кожному з них один і той же об'єкт обчислень;

в) у них відсутній заголовок, тобто перший рядок певного виду і призначення (на відміну від файл-функцій, які розпочинаються із заголовка «function»);

г) звернення до них не вимагає вказівки ніяких імен змінних: усі змінні формуються в результаті виконання програми або сформовані раніше і існують в робочому просторі.

### 1.1.10 Введення і виведення інформації в діалоговому режимі

Для забезпечення взаємодії з користувачем в процесі виконання файлу «\*.sci» використовується ряд спеціальних команд.

Команда **disp** здійснює виведення значень вказаної змінної або вказаного тексту в командне вікно. Звернення до неї має вигляд:

*disp(<змінна або текст в апострофах>)*

Наприклад:

`disp(x1)` – в командне вікно виводиться значення змінної `x1`.

`disp('value')` – в командне вікно виводиться текст `value`.

Щоб вивести значення декількох змінних в один рядок (наприклад, при створенні таблиць даних), треба створити єдиний об'єкт, який містив би усі ці значення. Це можна зробити, об'єднавши відповідні змінні у вектор, користуючись операцією створення вектора-рядка:

`x=[x1 x2 ... x]`.

Тоді виведення значень декількох змінних в один рядок будемати вигляд:

`disp([x1 x2 ... x])`.

Наприклад:

--> `x1=-3.14; x2=-2.5; x3=5.6; x4=-9.33;`

-->`disp([x1 x2 x3 x4])`

`-3.1400 -2.5000 5.6000 -9.3300`

Для одночасного виведення символічної і цифрової інформації в командне вікно зручно використати функцію **sprintf**, наприклад:

-->`disp(sprintf('Параметр 1= %g Параметр2= %g', x1, x2))`

`"Параметр 1 = - 3.14 параметр 2= - 2.5"`

Введення інформації з клавіатури в діалоговому режимі можна здійснити за допомогою функції **input**:

`x = input('<пояснюючий_текст>')`.

При виконанні цієї функції програма чекає введення інформації з клавіатури. Після закінчення введення, яке визначається натисненням клавіші <Enter>, введена інформація запам'ятовується в програмі під ім'ям «x», і виконання програми триває.

### 1.1.11 Галуження і цикли

У мові Scilab для організації галуження використовуються команди **if** та **switch**.

Загальна форма запису оператора **if** має вигляд

*if<умова 1>*

*<група операторів>;*

*[else if<умова 2>]*

*[<група операторів>; ]*

*else*

```
<група операторів>;  
end
```

**Приклад.** Створити файл-функцію, що здійснює визначення знаку числа:

$$\operatorname{sgn}(x) = \begin{cases} 1, & x > 0; \\ 0, & x = 0; \\ -1, & x < 0. \end{cases}$$

Функція розташовуватиметься в файлі *signum.sci*.

```
function y = signum(x)  
if x > 0  
y = 1;  
else if x == 0  
y = 0;  
else  
y = - 1;  
end  
endfunction
```

Команда **select** дозволяє здійснювати галуження за декількома умовами, причому умови розглядаються на рівність. Нижче представлений простий приклад з розділенням на три умови формування параметра введення.

```
function z = count(x)  
select x  
case 1 then  
z = 'one';  
case 2 then  
z = 'two';  
else  
z = 'many';  
end  
endfunction
```

Тут вираження **select** обчислює *x*, а потім виконання програми у файлі переходить туди, де вираження **case** має те ж саме значення, що й *x*. Таким чином, якщо параметр введення *x* має значення, рівне 1, тоді параметр виводу *z* визначається у вигляді рядка 'one', якщо *x* дорівнює 2, тоді *z* визначається у вигляді рядка 'two'.

Після виконання усіх команд, що йдуть за вибраною умовою, програма Scilab зустрічає або інше вираження **case**, або вираження **else**, що призводить до переходу виконання програми до оператора **end**.

Для організації циклів використовуються команди **for** і **while**.

Організація циклу за допомогою команди **for** виглядає таким чином:

```
for n=n_0: [step:] n_k  
[тіло циклу]
```

*end*

Цикл розпочинається з вираження **for** і закінчується вираженням **end**.

Величина  $n_0$  – початкове значення змінної циклу,  $step$  – крок,  $n_k$  – кінцеве значення змінної циклу.

Фрагмент коду для визначення факторіалу  $10!$  представлений нижче. Він може бути виконаний безпосередньо в командному режимі:

```
f=1;
for n=2:1:10
f=f*n;
end
f
```

Відповідь системи:

```
f=
3628800
```

Організація циклу за допомогою команди **while** виглядає таким чином:

```
while<умова>
[тіло циклу]
end
```

Цикл **while** триває до тих пір, поки  $\langle \text{умова} \rangle$  істинно.

Приклад: вчислити ряд  $\sum \frac{1}{n^2}$  з точністю  $10^{-9}$ .

```
n = 1;
oldsum = - 1;
newsum = 0;
while (newsum - oldsum)>1e-9
oldsum = newsum;
newsum = newsum + 1/(n^2);
n = n +1;
end
newsum
```

Відповідь системи:

```
newsum =
1.6449
```

Іноді буває необхідно, щоб програма Scilab передчасно вийшла з циклу, наприклад, при виникненні певної умови. Для цієї мети використовується команда **break**.

## 1.2 Задачі для самостійного розв'язання

**Задача 1.1.** Задайте матрицю **A**:

$$\begin{bmatrix} 3.15 & -2.08 & 6.11 \\ 10.10 & 0.25 & -4.78 \\ 7.03 & -45.2 & 8.46 \end{bmatrix}.$$

Згенеруйте масив **B** розміром  $3 \times 3$  з випадковими елементами, рівномірно розподіленими на інтервалі від 0 до 1. Виконайте дії:

- $A \cdot n \cdot B$ ;  $A \cdot B^T$ , де  $n$  – номер варіанта;
- почленне множення **A** на **B**;

- в) визначити максимальний і мінімальний елементи матриці **B**;  
г) обчислити визначник матриці **B**.

**Задача 1.2.** Задайте вектор-стовпчик **C**, трьома елементами якого є числа 0, n, – n, де n – номер варіанта. Вирішити систему лінійних алгебраїчних рівнянь:  
$$\mathbf{A} \cdot \mathbf{X} = \mathbf{C}.$$

Виконайте перевірку правильності рішення підстановкою **X** у рівняння.

**Задача 1.3.** Побудуйте два графіки у рамках одних осей координат:

$$y = e^{-x^2};$$

$$z = \arctg(x^{1/2});$$

$$x \in [0, 4\pi].$$

Зробіть написи на осях, заголовок для графіку, легенду. Задайте самостійно тип ліній і колір.

**Задача 1.4.** Побудуйте графіки функцій  $y(x)$  і  $z(x)$  задачі 1.3 в різних підобластях одного графічного вікна. Інтервали зміни аргументу  $x$  задайте так, щоб графіки були найбільш інформативні.

**Задача 1.5.** Створіть файл сценарію для розв'язання задач 1.1, 1.2, 1.3, 1.4. Для виведення результатів використовуйте команди **disp**, **sprintf**. У зошиті з практичних робіт наведіть текст сценарію, графіки і скріншот вікна команд Scilab з результатами обчислень.

## 2 ДОСЛІДЖЕННЯ ЛІНІЙНОЇ СТАЦІОНАРНОЇ ДИНАМІЧНОЇ СИСТЕМИ В СЕРЕДОВИЩІ SCILAB

### 2.1 Короткі теоретичні положення

#### 2.1.1 Моделі лінійних систем

Для опису лінійних стаціонарних систем можуть застосовуватися декілька способів:

- а) диференціальні рівняння;
- б) моделі в просторі станів;
- в) передатні функції (ПФ);
- г) моделі виду «нулі-полюси-коефіцієнти передачі»;

Перші два способи описують поведінку системи в часовій області і відображають внутрішні зв'язки між сигналами. Передатні функції і моделі виду «нулі-полюси» відносяться до частотних способів опису, оскільки безпосередньо пов'язані з частотними характеристиками системи і відображають тільки вхід-вихідні властивості.

Лінеаризована модель об'єкта може бути описана диференціальним рівнянням виду:

$$a_n \frac{d^n y}{dt^n} + a_{n-1} \frac{d^{n-1} y}{dt^{n-1}} + \dots + a_1 \frac{dy}{dt} + a_0 y = b_m \frac{d^m u}{dt^m} + b_{m-1} \frac{d^{m-1} u}{dt^{m-1}} + \dots + b_1 \frac{du}{dt} + b_0 u$$

за нульових початкових умов. Причому  $m \leq n$ .

Модель в просторі станів пов'язана із записом диференціальних рівнянь у формі Коши (у вигляді системи рівнянь першого порядку):

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}; \\ \mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u}, \end{cases} \quad (2.1)$$

де  $\mathbf{x}$  - вектор змінних стану розміру  $n$ ;  $\mathbf{u}$  - вектор вхідних сигналів (вектор управління) розміру  $m$  і  $\mathbf{y}$  - вектор вихідних сигналів розміру  $p$ . Крім того,  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{C}$  і  $\mathbf{D}$  - постійні матриці.

Згідно з правилами матричних обчислень, матриця  $\mathbf{A}$  має бути квадратною розміру  $n \times n$ , матриця  $\mathbf{B}$  має розмір  $n \times m$ , матриця  $\mathbf{C}$  -  $p \times n$  і матриця  $\mathbf{D}$  -  $p \times m$ . Для систем з одним входом і одним виходом матриця  $\mathbf{D}$  – скалярна величина.

Передатна функція  $W(s)$  лінійної стаціонарної системи як функція від комплексної змінної  $s$  визначається відношенням перетворення Лапласа виходу до перетворення Лапласа входу за нульових початкових умов:

$$W(s) = \frac{Y(s)}{U(s)}.$$

Для об'єкта, що описується представленим вище диференціальним рівнянням, передатна функція має вигляд:

$$W(s) = \frac{b_m s^m + b_{m-1} s^{m-1} + \dots + b_1 s + b_0}{a_n s^n + a_{n-1} s^{n-1} + \dots + a_1 s + a_0}.$$

З цієї передатної функції можна побудувати модель у формі «нулі-полюси». Нулями називаються корені чисельника, полюсами – корені знаменника ПФ.

Передатну функцію будь-якої лінійної системи можна записати як добуток передавальних функцій елементарних ланок першого та другого порядків.

Елементарна ланка першого порядку називається аперіодичною ланкою:

$$F(s) = \frac{1}{Ts + 1},$$

де  $T$  – стала часу.

Елементарна ланка другого порядку зветься коливальною ланкою:

$$F(s) = \frac{1}{T^2 s^2 + 2\xi Ts + 1},$$

де  $\xi$  – параметр згасання або коефіцієнт демпфірування. Передатна функція має для комплексно-спряжені полюси. При  $\xi = 0$  коливання стають незатухаючими, полюси – уявними числами. При  $\xi = 1$  полюси є парою дійсних чисел, і коливальна ланка перетворюється на аперіодичну ланку другого порядку.

Частота  $\omega_0 = 1/T$  називається власною частотою ланки.

### 2.1.2 Динамічні характеристики лінійних систем

Динамічні властивості систем характеризують реакції на вхідні дії спеціального виду. Динамічні характеристики систем підрозділяються на часові і частотні.

**Часові характеристики.** Імпульсною характеристикою (ваговою функцією)  $w(t)$  називається реакція системи на одиничний нескінченний імпульс (дельта-функція або функція Дірака) за нульових початкових умов. Дельта-функція  $\delta(t)$  визначається наступним чином:

$$\delta(t) = \begin{cases} 0, & t \neq 0, \\ \infty, & t = 0, \end{cases} \quad \int_{-\infty}^{\infty} \delta(t) dt = 1.$$

Це узагальнена функція – математичний об'єкт, що є ідеальним сигналом, який можна розглядати як межу прямокутного імпульсу одиничної площі з центром в точці  $t = 0$  при прагненні ширини імпульсу до нуля.

Перехідною характеристикою (перехідною функцією)  $h(t)$  називається реакція системи (за нульових початкових умов) на одиничний ступінчастий сигнал (одиничний стрибок) :

$$1(t) = \begin{cases} 0, & t < 0; \\ 1, & t \geq 0. \end{cases}$$

Імпульсна  $w(t)$  і перехідна  $h(t)$  функції пов'язані виразами

$$w(t) = \frac{dh(t)}{dt}; \quad h(t) = \int_0^t w(\tau) d\tau.$$

Для систем без інтеграторів перехідна характеристика прагне до постійного значення. Перехідна характеристика системи з диференційною ланкою (чисельник передатної функції має нуль в точці  $s = 0$ ) прагне до нуля. Якщо система містить інтегруючі ланки, перехідна характеристика асимптотично прагне до прямої, параболи і так далі, залежно від кількості інтеграторів.

За перехідною характеристикою можна знайти найважливіші показники якості системи – перерегулювання (*overshoot*) і час перехідного процесу (*settling time*).

Перерегулювання визначається як

$$\sigma = \frac{h_{\max} - h_{\infty}}{h_{\infty}} \times 100\%,$$

де  $h_{\max}$  – максимальне значення функції  $h(t)$ , а  $h_{\infty}$  – значення виходу, що встановилося.

Час перехідного процесу – це час, після якого сигнал виходу відрізняється від значення, що встановилося, не більше, ніж на задану малу величину (у середовищі SCILAB за умовчанням використовується точність 2%).

**Частотні характеристики.** При дослідженні стійкості динамічних систем і проектуванні регуляторів широке застосування отримали частотні характеристики.

При поданні на вхід лінійної системи гармонійного (синусоїдального) сигналу  $u(t) = \sin(\omega t)$  з частотою  $\omega$  (вона вимірюється в радіанах за секунду), на виході буде також гармонійний сигнал тієї ж частоти, але іншої амплітуди і фази  $y(t) = A \sin(\omega t + \varphi)$ , де  $A$  – амплітуда і  $\varphi$  – зсув фази.

Для побудови частотної характеристики потрібно використати підстановку  $s = j\omega$  у ПФ  $W(s)$ . Вираження  $W(j\omega)$  називається частотною передатною

функцією або амплітудно-фазовою частотною характеристикою системи (АФЧХ).

Залежність модуля величини  $W(j\omega)$  від частоти називається амплітудною частотною характеристикою (АЧХ), а залежність аргументу комплексної функції  $W(j\omega)$  (фази) від частоти – фазовою частотною характеристикою (ФЧХ):

$$A(\omega) = |W(j\omega)|, \quad \varphi(\omega) = \arg W(j\omega) = \arctg \frac{\operatorname{Im} W(j\omega)}{\operatorname{Re} W(j\omega)}.$$

АЧХ показує, наскільки посилюється амплітуда сигналів різних частот після проходження через систему, а ФЧХ характеризує зсув фаз сигналів різних частот. Відношення максимального значення АЧХ системи до значення цієї характеристики при  $\omega = 0$  називається показником коливання:

$$M = \max A(\omega) / A(0).$$

Вважається, що значення  $M = 1,5 \dots 1,6$  є оптимальним.

На практиці широке застосування отримала діаграма Боде (логарифмічна амплітудна характеристика – ЛАХ), яка визначається як  $L(\omega) = 20 \lg A(\omega)$ . Вимірюється ЛАХ в децибелах і будується як функція від  $\lg(\omega)$ .

### 2.1.3 Використання пакету Scilab

До складу системи Scilab входить ряд функцій для моделювання систем управління.

Для представлення передатної функції треба створити поліноми чисельника і знаменника. При цьому можна використати функцію `poly`:

$$\text{poly}(\text{vec}, "x", "c").$$

Ця функція формує поліном, чий коефіцієнти у *порядку збільшення степеня* є компонентами вектора *vec*. Символом "x" задається аргумент полінома. Символ "c" вказує, що поліном формується з коефіцієнтів (у випадку формування з коренів використовується символ "r"). Наприклад:

```
--> d=poly([3 5 4 3], 's', 'c')
d =
3 +5s +4s^2 +3s^3
--> e=poly([3 5 4 3], 's', 'r')
e =
180 -201s +83s^2 -15s^3 +s^4
```

Для побудови ПФ треба записати поліноми чисельника і знаменника, розділені символом «/». Наприклад:

```
--> W=poly([2 1], 's', 'c')/poly([3 5 4 3], 's', 'c')
W =
2 +s
-----
3 +5s +4s^2 +3s^3
```

Функція **syslin** визначає лінійну систему у вигляді списку:

$$\text{sys}=\text{syslin}(\text{dom}, H).$$

Перший аргумент *dom* визначає тип системи: неперервна ('c') або дискретна ('d'). Другим параметром *H* задається раніше створена передатна функція.

Другий варіант використання цієї функції:

$$\text{sys}=\text{syslin}(\text{dom}, \text{num}, \text{den}).$$

Тут *num* і *den* – поліноми чисельника і знаменника ПФ.

Третій варіант передбачає використання системної змінної *%s*. Тоді чисельник *num* і знаменник *den* можна записати так, як вони виглядають у передатній функції, наприклад:

```
--> W = syslin("c", %s+2, 3*%s^3+4*%s^2+5*%s+3)
```

Щоб кожного разу не набирати символ «%», можна ввести таку змінну:

```
--> s=%s;
```

Тепер для створення тієї ж системи використаємо:

```
--> W = syslin('c', s+2, 3*s^3+4*s^2+5*s+3)
```

З передатної функції можна створити модель у формі «нулі-полюси-коефіцієнт передачі» («zero-pole-gain»):

```
--> W_zpk = zpk(W)
```

```
W_zpk =  
                                (z+2)  
0.333333 -----  
          (z+0.805482) (z^2+0.527851z+1.24149)
```

Нулями називаються корені чисельника, полюсами – корені знаменника ПФ. Ця функція має один нуль і три полюси. Парі комплексних полюсів відповідає квадратний тричлен  $s^2 - 0,527851s + 1.24149$ .

Для перетворення ПФ у модель у просторі станів використовується функція **tf2ss(sys)**. Створити лінійну систему у просторі станів можна також функцією

$$[sys]=syslin(dom,A,B,C [,D [,x0] ]).$$

В аргументах функції можна, крім матриць **A**, **B**, **C** і **D** задати вектор *x0* (початкові значення змінних стану; значення за замовчуванням є рівними 0).

Функція **damp** дозволяє знайти полюси системи (*p*), а також відповідні їм власні частоти  $\omega_0$  (*w0*) і коефіцієнти демпфування  $\xi$  (*ksi*), наприклад:

```
--> W = syslin("c", 4*%s+5, %s^2+2*%s+3);
```

```
--> [w0,ksi,p] = damp (W)
```

```
w0 =  
    1.7320508  
    1.7320508  
ksi =  
    0.5773503  
    0.5773503  
p =  
    -1. + 1.4142136i  
    -1. - 1.4142136i
```

На практиці досить часто реальні системи автоматичного управління (САУ) представляються у вигляді сполучених між собою окремих динамічних ланок. У Scilab передбачена можливість програмно будувати схему САУ шляхом попереднього введення моделей ланок і подальшого з'єднання цих ланок в єдину структуру.

Задаємо дві системи W1 і W2:

```
-->K=1;
```

```
-->T=3;
```

```
--> W1=syslin('c', K/%s)
S1  =
    1
    -
    s
--> W2=syslin('c', K/poly([2 1], 's', 'c'))
S2  =
    1
    ----
    2 +s
```

До процедур, що здійснюють розрахунок характеристик з'єднань окремих ланок, відносяться:

а) операція «+» («-») – виконує паралельне з'єднання вказаних при зверненні ланок, тобто визначає характеристики моделі системи, що складається з паралельно сполучених ланок, наприклад  $W_1$  і  $W_2$  :

```
--> sys=W1+W2
sys  =
    2 +2s
    -----
    2s +s^2
```

б) операція «\*» – здійснює послідовне з'єднання ланок, імена яких вказані (застосовується для одновимірних систем) :

```
--> sys=W1*W2
sys  =
    1
    -----
    2s +s^2
```

в) операція «/ .» (пріміжок перед точкою обов'язковий!) – з'єднання із зворотним зв'язком, тобто таке з'єднання двох ланок, коли друга вказана ланка складає коло негативного зворотного зв'язку для першої ланки:

```
--> sys=S1 /. S2
sys  =
    2 +s
    -----
    1 +2s +s^2
```

Пакет Scilab надає можливості побудови часових і частотних характеристик систем.

Часові характеристики розраховуються за допомогою функції *csim(u, t, sl)*. Аргумент *u* може приймати значення:

- а) "step" для перехідної характеристики;
- б) "impuls" для імпульсної перехідної характеристики.

Аргумент *t* – вектор часу, аргумент *sl* – дескриптор системи. Приклад команди побудови графіка перехідної характеристики:

```
--> plot(csim("step", 0:0.1:20, sys))
```

Функції побудови частотних характеристик наведені в табл. 2.1.

Таблиця 2.1 – Функції побудови частотних характеристик систем

| Синтаксис                | Опис                                                        |
|--------------------------|-------------------------------------------------------------|
| gainplot(<дескриптор >)  | Побудова амплітудної логарифмічної частотної характеристики |
| phaseplot (<дескриптор>) | Побудова фазочастотної характеристики                       |
| bode(<дескриптор >)      | Побудова діаграми Боде                                      |
| nyquist(<дескриптор >)   | Побудова частотного годографа Найквіста                     |

## 2.2 Приклади і задачі для самостійного розв'язання

**Задача 2.1.** Необхідно сформулювати передатну функцію

$$F(s) = \frac{2s + 4}{s^3 + 0,8s^2 + 2,06s + 2,26}.$$

### Розв'язування

В даному випадку вектори коефіцієнтів чисельника і знаменника передатної функції мають вигляд:  $B=[4 \ 2]$ ,  $A=[2.26 \ 2.06 \ 0.8 \ 1]$ .

Процедура утворення передатної функції має вигляд:

```
-->B=[2 4];
-->A=[1 0.8 2.06 2.26];
-->W=poly(B, 's', 'c')/poly(A, 's', 'c')
```

У пам'яті створюється об'єкт W, що описує передатну функцію, і на екрані з'явиться передатна функція у вигляді:

$$\frac{2 + 4s}{1 + 0.8s + 2.06s^2 + 2.26s^3}$$

**Задача 2.2.** Розробити модель замкнутої системи автоматичного управління, в якій об'єкт управління представляється у вигляді з'єднання трьох аперіодичних ланок першого порядку з передатними функціями

$$W_1 = \frac{2}{3s+1}, \quad W_2 = \frac{1}{2s+1}, \quad W_3 = \frac{1}{4s+1},$$

а в зворотний зв'язок включений регулятор з передатною функцією

$$R = 0,5 + \frac{0,2}{s}.$$

Побудувати графік перехідної характеристики системи.

### Розв'язування

```
-->w1=poly([2], 's', 'c')/poly([1 3], 's', 'c'); //формування
ланки w1
-->w2=poly([1], 's', 'c')/poly([1 2], 's', 'c'); //формування
ланки w2
-->w3= poly([1], 's', 'c')/poly([1 4], 's', 'c'); //формування
ланки w3
-->w=w1*w2*w3; //формування об'єкта
```

```
-->R=0.5+poly([0.2], 's', 'c')/poly([0 1], 's', 'c');//ПФ
регулятора
-->W_zs=w/.R; //формування замкнутої системи
-->clf();//очищення вікна графіка
-->plot(csim("step", 0:0.1:20, W_zs))//побудова відгуку
системи
```

**Задача 2.3.** Виконати аналіз системи автоматичного управління, передатна функція якої наведена у таблиці. 2.2.

Таблиця 2.2 – Вихідні дані до задачі 2.3

| Варіант | Передатна функція                                                                | $b_2$ | $b_1$ | $b_0$ | $a_4$ | $a_3$ | $a_2$ | $a_1$ | $a_0$ |
|---------|----------------------------------------------------------------------------------|-------|-------|-------|-------|-------|-------|-------|-------|
| 1       | $W(s) = \frac{b_2 s^2 + b_1 s + b_0}{a_4 s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0}$ | 1     | 2     | 2     | 1     | 3     | 4     | 5     | 3     |
| 2       |                                                                                  | 1     | 3     | 1     | 2     | 2     | 4     | 2     | 1     |
| 3       |                                                                                  | 1     | 2     | 1     | 1     | 3     | 5     | 4     | 7     |
| 4       |                                                                                  | 1     | 0     | 1     | 1     | 2     | 5     | 4     | 3     |
| 5       |                                                                                  | 1     | 2     | 3     | 1     | 3     | 4     | 5     | 2     |
| 6       |                                                                                  | 1     | 1     | 2     | 2     | 3     | 3     | 4     | 4     |
| 7       | $W(s) = \frac{b_2 s^2 + b_1 s + b_0}{a_3 s^3 + a_2 s^2 + a_1 s + a_0}$           | 1     | 2     | 1     | -     | 2     | 5     | 4     | 3     |
| 8       |                                                                                  | 2     | 1     | 1     | -     | 2     | 4     | 5     | 1     |
| 9       |                                                                                  | 1     | 3     | 2     | -     | 1     | 5     | 2     | 3     |
| 10      |                                                                                  | 2     | 1     | 2     | -     | 1     | 5     | 5     | 2     |
| 11      |                                                                                  | 1     | 5     | 1     | -     | 1     | 5     | 4     | 2     |
| 12      |                                                                                  | 2     | 3     | 1     | -     | 2     | 3     | 4     | 1     |
| 13      | $W(s) = \frac{b_2 s^2 + b_1 s}{a_3 s^3 + a_2 s^2 + a_1 s + a_0}$                 | 1     | 1     | -     | -     | 2     | 5     | 4     | 3     |
| 14      |                                                                                  | 1     | 3     | -     | -     | 2     | 4     | 5     | 1     |
| 15      |                                                                                  | 3     | 1     | -     | -     | 1     | 5     | 2     | 3     |
| 16      |                                                                                  | 1     | 4     | -     | -     | 1     | 5     | 5     | 2     |
| 17      |                                                                                  | 1     | 2     | -     | -     | 1     | 5     | 4     | 9     |
| 18      |                                                                                  | 3     | 2     | -     | -     | 2     | 4     | 3     | 2     |
| 19      | $W(s) = \frac{b_2 s^2 + b_1 s + b_0}{a_3 s^3 + a_2 s^2 + a_1 s}$                 | 1     | 1     | 1     | -     | 1     | 1     | 5     | -     |
| 20      |                                                                                  | 2     | 0     | 1     | -     | 2     | 3     | 2     | -     |
| 21      |                                                                                  | 0     | 2     | 1     | -     | 1     | 3     | 3     | -     |
| 22      |                                                                                  | 0     | 3     | 2     | -     | 2     | 2     | 1     | -     |
| 23      |                                                                                  | 1     | 0     | 1     | -     | 2     | 1     | 2     | -     |
| 24      |                                                                                  | 1     | 2     | 1     | -     | 1     | 2     | 2     | -     |
| 25      |                                                                                  | 2     | 3     | 0     | -     | 2     | 0     | 1     | -     |

Порядок проведення аналізу наступний.

1. Запустіть середовище Scilab і в командному режимі очистіть робочий простір і вікно:

```
-->clear;clc
```

2. Введіть передатну функцію як об'єкт Scilab за зразком:

```
--> num = poly([b0 b1 b2], 's', 'c')
--> den = poly([a0 a1 a2 a3 a4], 's', 'c')
--> f = syslin('c', num, den)
```

3. Знайдіть нулі, полюси передатної функції і коефіцієнт посилення:

```
--> z = roots(num) //нулі
```

```

--> p = roots(den) //плюси
--> k = horner(f,0) //коефіцієнт посилення
    Показати нулі і полюси на комплексній площині:
--> plzr(f)
    4. Побудуйте модель початкової системи у формі «нулі-полюси»:
--> f_zpk = zpk(f)
    5. Побудуйте модель системи в просторі стану :
--> f_ss = tf2ss(f)
    Напишіть у зошиті відповідне матричне рівняння.
    6. У командному режимі побудуйте перехідну характеристику:
--> t=[0:0.05:30] //вектор часу
--> plot(csim("step",t,f)) //графік
    7. У командному режимі побудуйте імпульсну характеристику:
--> scf(); //нове графічне вікно
--> plot(csim("impulse",t,f)) //графік
    8. Побудуйте логарифмічні частотні характеристики (діаграми Боде) :
--> scf(); bode(f)
    9. Побудуйте амплітудно-фазову частотну характеристику (частотний
    годограф Найквіста) :
--> scf(); nyquist(f); xgrid
    10. Побудуйте АЧХ системи:
--> w=logspace(-2,2,100); //вектор частот
--> fResp=repfreq(f,w/(2*pi)); //розрахунок АФЧХ
--> Aw=abs(fResp); //розрахунок АЧХ
--> scf(); plot("ln",w,Aw) // графік АЧХ
    11. Обчисліть показник коливності системи:
--> M=max(Aw)/Aw(1)
    12. У зошиті для розв'язання задач наведіть зміст командного вікна і всі
    отримані дані і графіки.

```

## 3 ПОБУДОВА МАТЕМАТИЧНОГО ОПИСУ ОБ'ЄКТА УПРАВЛІННЯ ЕКСПЕРИМЕНТАЛЬНИМИ МЕТОДАМИ

### **3.1 Короткі теоретичні положення**

#### **3.1.1 Статичні і динамічні характеристики**

Математичні моделі об'єктів управління, процедура їх побудови залежать від їх цільового призначення, властивостей об'єкта, а також функцій і завдань системи управління.

При автоматизації виробничих процесів визначають статичні і динамічні характеристики регульованих об'єктів.

Статичні характеристики системи або окремих її ланок є функціональною залежністю вихідних величин від вхідних при статичному (стаціонарному, такому, що встановився) режимі роботи, тобто при значеннях вхідних і вихідних величин, що не змінюються в часі. Статичні характеристики дозволяють визначити значення регульованих параметрів, положення регулюючих органів,

витрати речовин або енергії і інші дані для будь-якого стану досліджуваного об'єкта, що встановився.

Динамічні характеристики є залежностями між змінами вхідних і вихідних величин в динамічному режимі (у часі). Динамічні характеристики дають інформацію про інерційні властивості регульованих об'єктів (систем, елементів систем) і таким чином є початковими даними при синтезі автоматичних систем регулювання. Вони дозволяють виконати цю роботу в повному об'ємі і завершити її розрахунком параметрів налаштування на цьому регульованому об'єкті з метою отримання бажаних форм графіків процесів регулювання.

Статичні і динамічні характеристики можуть бути отримані аналітичним, експериментальним або експериментально-аналітичним методом.

При побудові систем автоматичної стабілізації окремих технологічних параметрів досить мати в розпорядженні динамічні і статичні характеристики об'єкта управління у вузькому діапазоні зміни вхідних і вихідних координат. Для отримання цих характеристик застосовуються численні експериментальні методи. Ці методи базуються на припущенні про лінійність і зосередженість параметрів об'єкта, незмінність в часі його статичних і динамічних характеристик. Прийняття цих допущень дозволяє досить просто описати зміну спостережуваних вихідних координат об'єкта лінійними диференціальними рівняннями з постійними коефіцієнтами.

При побудові експериментальних математичних моделей використовує пасивний або активний експерименти.

При активних методах на вході досліджуваного об'єкта створюються штучні випробувальні дії (ступінчасті, імпульсні, гармонійні та ін.). Пасивними методами називають такі, при яких не допускаються штучні обурення на вході досліджуваного об'єкта, а як вхідні випробувальні дії використовуються природні дії, що виникають в процесі нормальної експлуатації.

### 3.1.2 Апроксимація статичних характеристик

При експериментальному дослідженні статичних характеристик деякого одноканального об'єкта управління отримано  $m$  значень вхідної  $x_i$  ( $i = 1, \dots, m$ ) і вихідної  $y_i$  координат. Треба знайти функціональну залежність  $\hat{y} = f(x, a)$ , що якнайкраще описує експериментальні дані, де  $a$  - вектор параметрів функції. Для цієї мети досить часто використовується метод найменших квадратів. Згідно з цим методом вводиться функція, що характеризує міру близькості експериментальних і розрахункових даних:

$$S = \sum_i (y_i - \hat{y}_i)^2.$$

При використанні методу найменших квадратів задаються видом функції і вектором параметрів цієї функції. Часто як апроксимуючу функцію  $\hat{y} = f(x, a)$  використовують поліном виду:

$$\hat{y} = a_0 + a_1x + a_2x^2 + a_3x^3 + \dots$$

Завдання пошуку апроксимуючої функції (вектору параметрів) ставиться і вирішується як завдання безумовної багатовимірної оптимізації: вимагається знайти вектор параметрів  $a$ , що забезпечує мінімум функції  $S$ .

$$S = \sum_i (y_i - \hat{y}_i)^2 \rightarrow \min_a. \quad (3.1)$$

У пакеті Scilab для апроксимації експериментальних даних, які сформовані як деякий вектор  $Y$ , при деяких значеннях аргументу, які утворюють вектор  $X$  такої ж довжини, що і вектор  $Y$ , використовується функція:

$$[f, a] = \text{leastsq}(\text{list}(\text{fun}, X, Y), \text{init}).$$

Тут  $\text{fun}$  – ім'я файл-функції для розрахунку функції (3.1);

$\text{init}$  – вектор початкових наближень для шуканих коефіцієнтів;

$f$  – результуюче значення (3.1);

$a$  – результуючі значення коефіцієнтів полінома.

### 3.1.3 Апроксимація динамічних характеристик

У інженерній практиці при налаштуванні систем управління математичний опис об'єкта досить часто представляється у вигляді передатних функцій.

Передатна функція об'єкта регулювання може бути визначена в результаті апроксимації експериментальної динамічної характеристики, отриманої в результаті проведення активного експерименту. Як випробувальна дія найчастіше використовується ступінчаста дія.

Алгоритм визначення апроксимуючої передатної функції об'єкта регулювання має вигляд:

а) при поданні на вхід об'єкта ступінчастої дії на виході знімається перехідна характеристика  $h(t)$ ;

б) по виду кривої визначається структура передатної функції. Для великого числа реальних об'єктів регулювання модель об'єкта може бути представлена у вигляді:

$$W(s) = \frac{ke^{-s\tau}}{(T_1s + 1)(T_2s + 1)}. \quad (3.2)$$

### 3.1.4 Апроксимація перехідної характеристики об'єкта передатною функцією

Апроксимація полягає в підборі таких параметрів передатної функції, які забезпечують максимальну близькість експериментальної кривої розгону  $h(t)$  і розрахованій кривій  $h_p(t)$ , що відповідає цій передатній функції.

Параметри налаштування  $k$ ,  $T_1$ ,  $T_2$  та  $\tau$  передатної функції (3.2) можуть бути отримані при мінімізації функції

$$I = \int (h(t) - h_p(t))^2 dt \rightarrow \min_{k, T_1, T_2, \tau}. \quad (3.3)$$

Пакет Scilab містить функцію **datafit** ідентифікації параметрів на основі вимірюваних даних. **datafit** використовується для підгонки цих даних до моделі. Для заданої функції  $G(p, z)$  **datafit** знаходить найкращий вектор параметрів  $p$  для набору векторів вимірювання  $z$ . Вектор  $p$  шукається згідно (3.3). Синтаксис:

$$[p, \text{err}] = \text{datafit}(G, Z, [\text{contr}], p0).$$

Тут  $G$  – ім'я файл-функції для розрахунку функції, що мінімізується;

$Z$  – вектор вимірюваних даних;

$p0$  – початкові наближення для шуканих параметрів;

$\text{contr}$  (необов'язковий аргумент) – обмеження для  $p0$ ;

$err$  – помилка найменшого квадрата.

### 3.2 Приклади і задачі для самостійного розв'язання

**Задача 3.1.** Виконати апроксимацію статичної характеристики одноканального об'єкта управління поліномом  $y=a_1+a_2x+a_3x^2$  за експериментальними даними, наведеними у таблиці. 3.1.

Таблиця 3.1 – Вихідні дані до задачі 3.1

| Варіант | Експериментальні дані |      |      |      |      |      |      |      |      |      |      |
|---------|-----------------------|------|------|------|------|------|------|------|------|------|------|
| 0       | $x_i$                 | 0    | 1    | 2    | 3    | 4    | 5    | 6    | 7    | 8    | 9    |
|         | $y_i$                 | 3,0  | 5,1  | 7,4  | 9,9  | 12,6 | 15,5 | 18,6 | 21,9 | 25,4 | 29,1 |
| 1       | $x_i$                 | 0    | 1    | 2    | 3    | 4    | 5    | 6    | 7    | 8    | 9    |
|         | $y_i$                 | 6,0  | 7,9  | 9,6  | 11,1 | 12,4 | 13,5 | 14,4 | 15,1 | 15,6 | 15,9 |
| 2       | $x_i$                 | 0    | 1    | 2    | 3    | 4    | 5    | 6    | 7    | 8    | 9    |
|         | $y_i$                 | 4,0  | 6,8  | 9,2  | 11,2 | 12,8 | 14,0 | 14,8 | 15,2 | 15,2 | 14,8 |
| 3       | $x_i$                 | 0    | 1    | 2    | 3    | 4    | 5    | 6    | 7    | 8    | 9    |
|         | $y_i$                 | 5,0  | 10,1 | 14,4 | 17,9 | 20,6 | 22,5 | 23,6 | 23,9 | 23,4 | 22,1 |
| 4       | $x_i$                 | 0    | 1    | 2    | 3    | 4    | 5    | 6    | 7    | 8    | 9    |
|         | $y_i$                 | 35,0 | 34,3 | 33,0 | 31,1 | 28,6 | 25,5 | 21,8 | 17,5 | 12,6 | 7,1  |
| 5       | $x_i$                 | 0    | 2    | 4    | 6    | 8    | 10   | 12   | 14   | 16   | 18   |
|         | $y_i$                 | 3,0  | 5,1  | 7,4  | 9,9  | 12,6 | 15,5 | 18,6 | 21,9 | 25,4 | 29,1 |
| 6       | $x_i$                 | 0    | 2    | 4    | 6    | 8    | 10   | 12   | 14   | 16   | 18   |
|         | $y_i$                 | 6,0  | 7,9  | 9,6  | 11,1 | 12,4 | 13,5 | 14,4 | 15,1 | 15,6 | 15,9 |
| 7       | $x_i$                 | 0    | 2    | 4    | 6    | 8    | 10   | 12   | 14   | 16   | 18   |
|         | $y_i$                 | 4,0  | 6,8  | 9,2  | 11,2 | 12,8 | 14,0 | 14,8 | 15,2 | 15,2 | 14,8 |
| 8       | $x_i$                 | 0    | 2    | 4    | 6    | 8    | 10   | 12   | 14   | 16   | 18   |
|         | $y_i$                 | 5,0  | 10,1 | 14,4 | 17,9 | 20,6 | 22,5 | 23,6 | 23,9 | 23,4 | 22,1 |
| 9       | $x_i$                 | 0    | 2    | 4    | 6    | 8    | 10   | 12   | 14   | 16   | 18   |
|         | $y_i$                 | 35,0 | 34,3 | 33,0 | 31,1 | 28,6 | 25,5 | 21,8 | 17,5 | 12,6 | 7,1  |

Порядок проведення аналізу наступний.

1. Запустити середовище Scilab і в командному режимі очистити робочий простір і вікно:

```
-->clear; clc
```

2. Сформуєте масиви експериментальних точок – вектори  $X$  і  $Y$  (відповідно до варіанту, див. табл. 3.1):

```
--> X = [x1 x2 x3 x4 x5 x6 x7 x8 x9 x10];
```

```
--> Y = [y1 y2 y3 y4 y5 y6 y7 y8 y9 y10];
```

3. Задайте вектор початкових наближень для значень шуканих коефіцієнтів:

```
--> init = [2 1 0.1]
```

4. Створюємо функцію fun розрахунку функції (3.1):

```
--> function e=fun(coef, xm, ym)
```

```
> yy =coef(1)+coef(2)*xm+coef(3)*xm.^2
```

```
> e=sum((yy - ym).^2)
```

```
> endfunction
```

5. Виконуємо функцію реалізації метода найменших квадратів:

```
--> [f,a] = leastsq(list(fun, X, Y), init);
```

6. Виводимо значення шуканих коефіцієнтів:

```
--> disp(sprintf('y = %g + %gx + %gx^2',a(1),a(2),a(3)))
      "y = 3 + 2x + 0.1x^2"
```

7. Виконуємо обчислення за поліномом  $y=a_1+a_2x+a_3x^2$ :

```
--> YY =a(1)+a(2)*X+a(3)*X.^2;
```

8. В одних координатних осях побудуйте графіки експериментальних точок, а також апроксимуючого полінома:

```
--> plot(X,Y,'o',X,YY)
```

```
-->
```

```
hl=legend(['експерим.';'розрахунок'],'in_lower_right');
```

**Задача 3.2.** Визначити передатну функцію динамічного об'єкта за кривою розгону, яка представлена даними табл. 3.2.

Таблиця 3.2 – Вихідні дані до задачі 3.2

| T | 0 | 1 | 2 | 3    | 4    | 5    | 6    | 7    | 8    | 9    | 10   | 11   | 12   | 13   | 14   | 15   |
|---|---|---|---|------|------|------|------|------|------|------|------|------|------|------|------|------|
| h | 0 | 0 | 0 | 0.07 | 0.23 | 0.43 | 0.64 | 0.84 | 1.03 | 1.19 | 1.33 | 1.45 | 1.55 | 1.64 | 1.71 | 1.77 |

### Розв'язування

1. Створюємо файл сценарію для вирішення задачі. Для цього викликаємо редактор SciNotes кнопкою .

2. У вікні редактора набираємо текст:

```
// hp - вектор значень розрахованої перехідної
характеристики
// p - вектор параметрів передатної функції об'єкта
// p(1)=k, p(2)=t1, p(3)=t2, p(4)=tau
global t h hp
t=0:1:15;
h=[0 0 0 0.07 0.23 0.43 0.64 0.84 1.03 1.19 1.33 1.45 1.55
1.64 1.71 1.77];
z=[t,h];
// Початкові наближення
p0=[1 1 2 1];
exec('func.sci');
[p, err] =datafit(func,z,p0)
plot(t,h,'o',t,hp,'-')
disp(sprintf('K= %g t1= %g t2= %g tau= %g', p(1), p(2),
p(3), p(4)))
hl=legend(['експеримент';'розрахунок'],'in_lower_right');
```

3. Зберігаємо цей текст у файлі під ім'ям *din\_app.sci*.

4. Створюємо файл-функцію, яка вираховує функцію, що мінімізується.

Текст файл-функції має вигляд:

```
function [f]=func(p,z)
global t h hp
// p(1)=k, p(2)=t1, p(3)=t2, p(4)=tau
s=%s;
```

```

W1=syslin('c',p(1)/( p(2)*s+1));
W2=syslin('c',1/( p(3)*s+1));
sysTd=syslin('c',[1-p(4)*s/2]/[1+p(4)*s/2]);%Delay System
by Pade
W=W1*W2*sysTd;
hp=csim("step",t, W);
f=sum((h - hp).^2);
endfunction

```

У цій функції формується ПФ другого порядку із запізненням  $\tau$ , яке моделюється розкладанням у ряд Паде:

$$e^{-\tau s} \approx \frac{-0.5\tau s + 1}{0.5\tau s + 1}.$$

Рядок `hp=csim("step",t, W)` розраховує відгук на ступінчасту дію. Наступний рядок визначає суму квадратів нев'язок між експериментальною і розрахованою перехідними характеристиками.

5. Зберігаємо введений текст у файлі під ім'ям *func.sci*.

6. У командному режимі запускаємо на виконання Script-файл:

```
--> exec('din_app.sci');
```

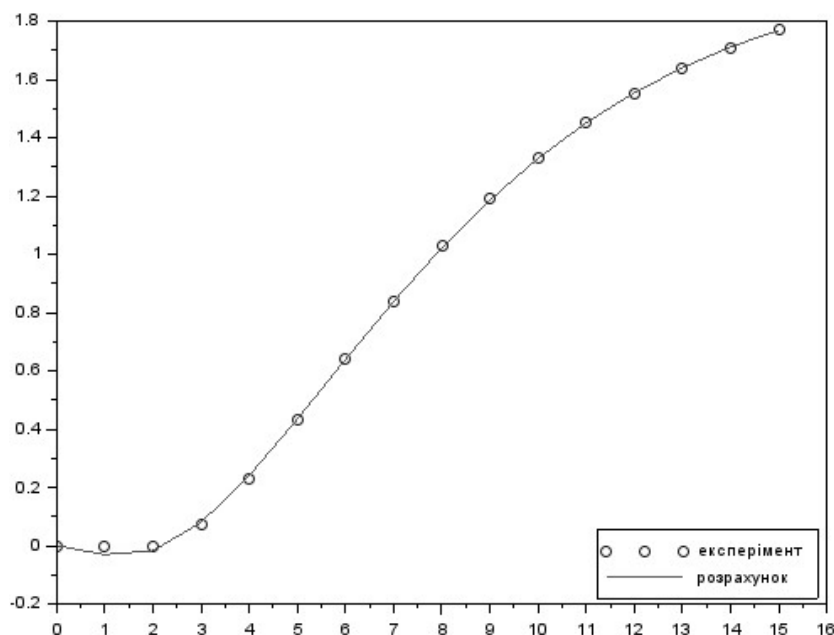
При завершенні виконання цього Script-файлу в командному вікні Scilab відображається результат рішення задачі:

“K=2.0382, t1=2.4265, t2=4.8308, tau=2.0181”

Тобто ПФ має вигляд:

$$W(s) = \frac{2.0382}{(2.4265s + 1)(4.8308s + 1)} \exp(-2.0181s).$$

На графік виводяться експериментально зняті точки перехідної характеристики і розрахована апроксимуюча перехідна характеристика.



**Задача 3.3.** Виконати апроксимацію динамічної характеристики об'єкта управління передатною функцією за кривою розгону, яка представлена даними табл. 3.3.

Таблиця 3.3– Варіанти завдань до задачі 3.3

| 0        |          | 1        |          | 2        |          | 3        |          | 4        |          | 5        |          | 6        |          | 7        |          | 8        |          | 9        |          |
|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| <i>T</i> | <i>h</i> | <i>T</i> | <i>h</i> | <i>T</i> | <i>h</i> | <i>T</i> | <i>h</i> | <i>T</i> | <i>h</i> | <i>T</i> | <i>h</i> | <i>T</i> | <i>h</i> | <i>T</i> | <i>h</i> | <i>T</i> | <i>h</i> | <i>T</i> | <i>h</i> |
| 0        | 0        | 0        | 0        | 0        | 0        | 0        | 0        | 0        | 0        | 0        | 0        | 0        | 0        | 0        | 0        | 0        | 0        | 0        | 0        |
| 1        | 0        | 1        | 0        | 1        | 0        | 1        | 0        | 1        | 0        | 2        | 0        | 2        | 0        | 2        | 0        | 2        | 0        | 2        | 0        |
| 2        | 0        | 2        | 0        | 2        | 0        | 2        | 0,14     | 2        | 0,098    | 4        | 0        | 4        | 0        | 4        | 0        | 4        | 0,14     | 4        | 0,1      |
| 3        | 0,07     | 3        | 0,09     | 3        | 0,13     | 3        | 0,46     | 3        | 0,31     | 6        | 0,07     | 6        | 0,09     | 6        | 0,13     | 6        | 0,46     | 6        | 0,32     |
| 4        | 0,23     | 4        | 0,29     | 4        | 0,43     | 4        | 0,84     | 4        | 0,56     | 8        | 0,23     | 8        | 0,29     | 8        | 0,43     | 8        | 0,84     | 8        | 0,57     |
| 5        | 0,43     | 5        | 0,53     | 5        | 0,79     | 5        | 1,19     | 5        | 0,8      | 10       | 0,43     | 10       | 0,53     | 10       | 0,79     | 10       | 1,19     | 10       | 0,88     |
| 6        | 0,64     | 6        | 0,77     | 6        | 1,15     | 6        | 1,52     | 6        | 1,02     | 12       | 0,64     | 12       | 0,77     | 12       | 1,15     | 12       | 1,52     | 12       | 1,1      |
| 7        | 0,84     | 7        | 0,99     | 7        | 1,49     | 7        | 1,81     | 7        | 1,21     | 14       | 0,84     | 14       | 0,99     | 14       | 1,49     | 14       | 1,81     | 14       | 1,23     |
| 8        | 1,03     | 8        | 1,19     | 8        | 1,78     | 8        | 2,05     | 8        | 1,36     | 16       | 1,03     | 16       | 1,19     | 16       | 1,78     | 16       | 2,05     | 16       | 1,4      |
| 9        | 1,19     | 9        | 1,35     | 9        | 2,03     | 9        | 2,24     | 9        | 1,49     | 18       | 1,19     | 18       | 1,35     | 18       | 2,03     | 18       | 2,24     | 18       | 1,5      |
| 10       | 1,33     | 10       | 1,49     | 10       | 2,23     | 10       | 2,4      | 10       | 1,6      | 20       | 1,33     | 20       | 1,49     | 20       | 2,23     | 20       | 2,4      | 20       | 1,62     |
| 11       | 1,45     | 11       | 1,61     | 11       | 2,4      | 11       | 2,52     | 11       | 1,69     | 22       | 1,45     | 22       | 1,61     | 22       | 2,4      | 22       | 2,52     | 22       | 1,71     |
| 12       | 1,55     | 12       | 1,69     | 12       | 2,54     | 12       | 2,63     | 12       | 1,75     | 24       | 1,55     | 24       | 1,69     | 24       | 2,54     | 24       | 2,63     | 24       | 1,78     |
| 13       | 1,64     | 13       | 1,76     | 13       | 2,64     | 13       | 2,71     | 13       | 1,81     | 26       | 1,64     | 26       | 1,76     | 26       | 2,64     | 26       | 2,71     | 26       | 1,83     |
| 14       | 1,71     | 14       | 1,82     | 14       | 2,72     | 14       | 2,79     | 14       | 1,85     | 28       | 1,71     | 28       | 1,82     | 28       | 2,72     | 28       | 2,79     | 28       | 1,88     |
| 15       | 1,77     | 15       | 1,86     | 15       | 2,79     | 15       | 2,82     | 15       | 1,88     | 30       | 1,77     | 30       | 1,86     | 30       | 2,79     | 30       | 2,82     | 30       | 1,92     |

## 4 АНАЛІЗ СТІЙКОСТІ ЛІНІЙНОЇ СИСТЕМИ

### 4.1 Короткі теоретичні положення

#### 4.1.1 Поняття стійкості системи

Стійкою називається система автоматики, яка після припинення дії збурюючих чинників прагне до початкового або нового стійкого стану, тобто перехідні процеси в ній є такими, що затухають.

Про стійкість системи можна судити, спостерігаючи її перехідні процеси. Система стійка, якщо її вільні рухи затухають з часом і в системі встановлюється вимушений процес (рис. 4.1).

Оцінити вигляд перехідного процесу можна шляхом розв'язання диференціальних рівнянь системи. Проте є і простіші методи.

Передатна функція лінійної системи має вигляд:

$$W(s) = \frac{b_m s^m + \dots + b_1 s + b_0}{a_n s^n + \dots + a_1 s + a_0}.$$

Чисельник ПФ визначає вимушений рух системи під дією вхідного чинника, знаменник – власний рух. Саме від знаменника і залежить стійкість системи. Якщо прирівняти знаменник ПФ нулю, отримаємо характеристичне рівняння системи:

$$a_n s^n + \dots + a_1 s + a_0 = 0.$$

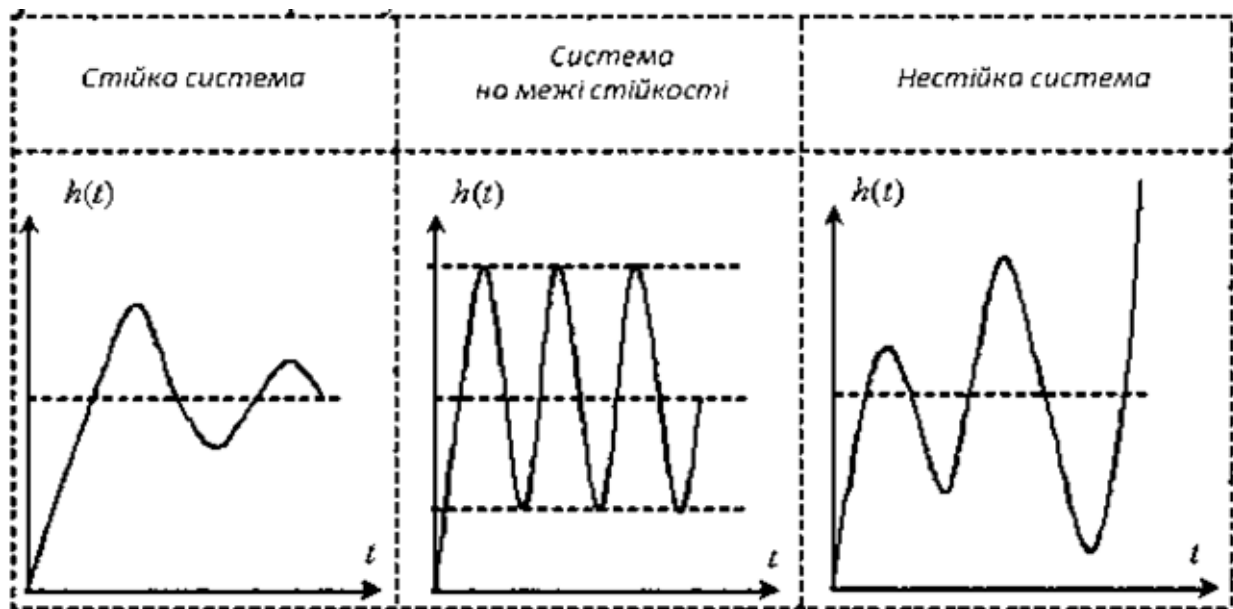


Рис. 4.1

Якщо всі корені характеристичного рівняння (що звуться полюсами системи) мають негативні дійсні частини, система є стійкою, якщо хоч один корінь має позитивну дійсну частину, система нестійка. У випадку нульових значень система знаходиться на межі стійкості.

Для багатьох реальних систем визначення полюсів є складною задачею. Тому вироблений ряд методів, які дозволяють за непрямыми ознаками оцінити стійкість. Вони називаються критеріями стійкості.

#### 4.1.2 Критерій Рауса-Гурвіца

Критерій Рауса-Гурвіца є алгебраїчним критерієм і застосовується до коефіцієнтів характеристичного рівняння замкнутої системи.

З коефіцієнтів характеристичного рівняння складають матрицю за правилом:

1. По діагоналі записуються коефіцієнти від  $a_{n-1}$  до  $a_0$ .
2. Кожен рядок доповнюється коефіцієнтами із зростаючими індексами зліва направо так, щоб чергувалися рядки з непарними і парними індексами.
3. У разі відсутності індексу, а також, якщо він менше 0 або більше  $n$ , на його місце пишеться 0.

Таким чином, матриця Гурвіца набуває наступного вигляду:

$$\begin{bmatrix} a_{n-1} & a_{n-3} & a_{n-5} & \dots & 0 & 0 \\ a_n & a_{n-2} & a_{n-4} & \dots & 0 & 0 \\ 0 & a_{n-1} & a_{n-3} & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & a_1 & 0 \\ 0 & 0 & 0 & \dots & a_2 & a_0 \end{bmatrix}.$$

Критерій стійкості формулюється так:

Щоб система була стійкою, необхідно і достатньо, щоб при  $a_n > 0$  були позитивними усе  $n$  діагональних визначників, що отримуються з матриці Гурвіца.

Перші три визначники матриці Гурвіца мають наступний вигляд:

$$\Delta_1 = a_{n-1}; \Delta_2 = \begin{vmatrix} a_{n-1} & a_{n-3} \\ a_n & a_{n-2} \end{vmatrix}; \Delta_3 = \begin{vmatrix} a_{n-1} & a_{n-3} & a_{n-5} \\ a_n & a_{n-2} & a_{n-4} \\ 0 & a_{n-1} & a_{n-3} \end{vmatrix}.$$

Критерій Гурвіца дозволяє судити про абсолютну стійкість, але він не дає можливості оцінювати відносну стійкість по коренях характеристичного рівняння.

#### 4.1.1 Критерій Найквіста

Частотний критерій стійкості Найквіста аналізує АФЧХ розімкненої системи.

Нехай є ПФ розімкненої системи  $W(j\omega)$ .

Для знаходження дійсної і уявної частини частотної ПФ треба звільнитися від уявності в знаменнику шляхом множення чисельника і знаменника на величину, комплексноспряжену знаменнику, а потім виконати розділення на дійсну і уявну частини. Передатна функція набуває вигляду:

$$W(j\omega) = P(\omega) + jQ(\omega).$$

Основні властивості АФЧХ розімкненої системи наступні.

1. Якщо розімкнена система не має інтегруючих ланок, то при  $\omega = 0$  її АФЧХ починається на дійсній осі в точці  $P(\omega) = K$  (де  $K$  – коефіцієнт посилення розімкненої системи). Закінчується АФЧХ на початку координат при  $\omega \rightarrow \infty$  (рис. 4.1, а).

2. Якщо розімкнена система має одну інтегруючу ланку, то її АФЧХ починається при  $\omega = 0$  в нескінченності на негативній уявній півосі, а закінчується на початку координат при  $\omega \rightarrow \infty$  (рис. 4.1, б).

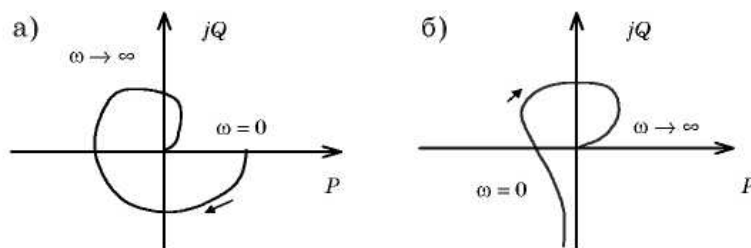


Рис. 4.1 – АФЧХ розімкненої системи

Критерій стійкості Найквіста формулюється так:

а) якщо розімкнена система стійка або знаходиться на межі стійкості, то для того, щоб замкнута система була стійка, необхідно і достатньо, щоб АФЧХ розімкненої системи при зміні частоти  $\omega$  від 0 до  $\infty$  не охоплювала точку з координатами  $(-1, j0)$ ;

б) якщо розімкнена система нестійка, а її передатна функція має  $m$  полюсів праворуч від уявної осі на комплексній площині, то для стійкості замкнутої системи необхідно і достатньо, щоб АФЧХ розімкненої системи при зміні частоти  $\omega$  від  $-\infty$  до  $+\infty$  охоплювала  $m$  разів точку з координатами  $(-1, j0)$ .

При використанні цього критерію треба враховувати дві особливості:

а) якщо розімкнена система знаходиться на межі стійкості, то її АФЧХ йде в нескінченність. Для перевірки критерію Найквіста треба подумки з'єднати кінець АФЧХ дугою нескінченно великого радіусу з позитивною дійсною піввіссю;

б) на практиці АФЧХ може будуватися тільки для позитивних частот ( $0 < \omega < +\infty$ ). При застосуванні критерію Найквіста вважається, що гілка АФЧХ для негативних частот симетрична відносно дійсної осі.

Фізичний сенс критерію стійкості Найквіста полягає в тому, що система буде нестійка, якщо фаза вихідного сигналу протилежна до фази вхідного сигналу, а коефіцієнт посилення більше 1.

#### 4.1.3 Точність в режимі, що встановився

Нехай передатну функцію розімкненої системи можна представити у виді

$$W(s) = KW_1(s),$$

де передатна функція  $W_1(s)$  має властивість  $\lim_{s \rightarrow 0} W_1(s) = 1$ . Тоді передатна функція замкнутої системи за помилкою

$$\Phi_\varepsilon(s) = \frac{1}{1 + KW_1(s)}.$$

Значення помилки, що встановилося, при постійному вхідному сигналі  $x(t) = x_0$ , що має зображення за Лапласом  $X(s) = \frac{x_0}{s}$ , може бути обчислене за теоремою про кінцеве значення:

$$\varepsilon_\infty = \lim_{t \rightarrow \infty} \varepsilon(t) = \lim_{s \rightarrow 0} s \Phi_\varepsilon(s) X(s) = \frac{x_0}{1 + K}.$$

Таким чином, при збільшенні коефіцієнта посилення  $K$  помилка зменшується (проте запас стійкості також зменшується і система може стати нестійкою). Величина  $K$  називається *добротністю* системи. При будь-якому кінцевому  $K$  в такій системі помилка буде кінцевою. Для лінійно зростаючого сигналу помилка лінійно зростатиме.

Тепер нехай

$$W(s) = \frac{K}{s^\nu} W_1(s),$$

де  $\nu$  - ціле число і  $\lim_{s \rightarrow 0} W_1(s) = 1$ . Тоді для усіх вхідних сигналів виду

$$x(t) = x_0 + x_1 t + x_2 t^2 + \dots + x_{\nu-1} t^{\nu-1}$$

система забезпечуватиме нульову помилку, що встановилася, при будь-яких значеннях коефіцієнтів  $x_0, \dots, x_{\nu-1}$ . Таким чином, при  $\nu > 0$  система відстежує постійний сигнал без помилки, що встановилася. Такі системи називають *астатичними*.

Число  $\nu$  називається *порядком астатизму*. Для сигналу

$$x(t) = x_\nu t^\nu, \quad X(s) = \frac{\nu! x_\nu}{s^{\nu+1}}$$

помилка, що встановилася, є рівною

$$\varepsilon_\infty = \frac{\nu! x_\nu}{K}.$$

Вище розглянутим є випадок астатизму по відношенню до задаючої дії. Аналогічно може йтися про астатизм по відношенню до збурюючої дії.

#### 4.2 Приклади і задачі для самотійного розв'язання

**Задача 4.1.** За допомогою критерію Гурвіца визначити стійкість системи, що має передатну функцію

$$W(s) = \frac{Y(s)}{U(s)} = \frac{5 \cdot s + 4}{s^5 + 16 \cdot s^4 + 95 \cdot s^3 + 260 \cdot s^2 + 324 \cdot s + 144}$$

#### Розв'язування

Для перевірки стійкості САУ по Гурвіцу можна використати наступну функцію:

```
function out=RHfunc(lm)
N= length(lm); // визначення розмірності
g=zeros(N, N); // створення матриці з нульовими значеннями
// заповнення непарних рядків матриці Гурвіца
s=0;
for i=1:2:N
    j=1; j=j+s; r=0;
    for r=2:2:N
        g(i, j)=lm(r); j=j+1;
    end
    s=s+1;
end
// заповнення парних рядків матриці Гурвіца
s=0;
for i=2:2:N
    j=1; j=j+s; r=0;
    for r=1:2:N
        g(i, j)=lm(r); j=j+1;
    end
    s=s+1;
end
g=g(1:N-1, 1:N-1)
disp('Матриця Рауса-Гурвіца')
disp(g)
// Перевірка стійкості
minor=1; // ознака додатності мінорів
for i=1:N-1
    dd(i) = det(g(1:i, 1:i)); //обчислення мінорів
    if dd(i)<0
        minor=0; //ознака від'ємності мінорів
    end
end
disp(' Мінори: ')
disp(dd')
```

```
// виведення результатів
if minor==0
out='Нестійка система';
else
out='Стійка система';
end
endfunction
```

Цю функцію треба набрати у редакторі SciNotes і зберегти під ім'ям RHfunc.sci.

Для використання функції її треба загрузити у Scilab введенням з консолі команди

```
--> exec('RHfunc.sci', -1)
```

Створюємо вектор з коефіцієнтів знаменника передатної функції:

```
--> C=[1 16 95 260 324 144]
```

```
C =
```

```
1.    16.    95.    260.    324.    144.
```

Вводимо команду виводу результату роботи функції RHfunc.sci:

```
--> disp(RHfunc(C))
```

"Матриця Рауса-Гурвіца"

```
16.    260.    144.    0.    0.
1.     95.    324.    0.    0.
0.     16.    260.    144.    0.
0.     1.     95.    324.    0.
0.     0.     16.    260.    144.
```

" Мінори: "

```
16.    1260.    246960.    63504000.    9.145D+09
```

"Стійка система"

Система є стійкою.

**Задача 4.2.** За допомогою критерію Гурвіца визначити стійкість системи, що має передатну функцію, наведену у табл. 4.1.

**Задача 4.3.** Дана передатна функція розімкнутої системи

$$W = \frac{2s + 1}{2s^4 + 3s^3 + 2s^2 + 3s + 1}.$$

Визначити, чи буде стійкою замкнута система.

**Розв'язування**

Для перевірки стійкості САУ за Найквістом спочатку треба з'ясувати, чи є стійкою розімкнена система.

Таблиця 4.1 – Вихідні дані до задачі 4.2

| Вари-<br>ант | Передатна функція розімкненої системи                   |
|--------------|---------------------------------------------------------|
| 0            | $W = \frac{2}{s^4 + 5s^3 + 5s^2 + 3s + 1}$              |
| 1            | $W = \frac{1}{0,05s^4 + 0,1s^3 + s^2 + s + 1}$          |
| 2            | $W = \frac{1}{0,1s^3 + 0,1s^2 + s + 1}$                 |
| 3            | $W = \frac{100}{5s^4 + 0,1s^3 + 2s^2 + 2s + 1}$         |
| 4            | $W = \frac{1}{8s^3 + 4s^2 + 2s + 1}$                    |
| 5            | $W = \frac{10}{s^5 + 3s^4 + 2s^3 + 2s^2 + s + 1}$       |
| 6            | $W = \frac{3}{0,1s^3 + 0,01s^2 + 0,1s + 1}$             |
| 7            | $W = \frac{10}{2s^3 + 2s^2 + s + 1}$                    |
| 8            | $W = \frac{1}{s^3 + 0,1s^2 + 0,1s + 1}$                 |
| 9            | $W = \frac{10}{2s^5 + 3s^4 + 3s^3 + 0,5s^2 + 0,5s + 1}$ |

Сформуємо передатну функцію:

```
--> W=poly([1 2], 's', 'c')/poly([1 3 2 3 2], 's', 'c')
```

W =

$$1 + 2s$$

$$\frac{1 + 2s}{1 + 3s + 2s^2 + 3s^3 + 2s^4}$$

Задамо безперервну САУ щодо її передатної функції:

```
--> sys=syslin('c', W)
```

sys =

$$1 + 2s$$

$$\frac{1 + 2s}{1 + 3s + 2s^2 + 3s^3 + 2s^4}$$

Розглянемо реакцію на стрибок:

```
--> plot(csim("step", 0:0.1:20, sys))
```

Будується графік перехідного процесу, показаний на рис. 4.2.

Як видно з графіка, розімкнена система нестійка, і, згідно з критерієм Найквіста, потрібно, щоб АФЧХ розімкнутої системи охоплювала точку  $(-1, j0)$  стільки разів, скільки полюсів є праворуч від уявної осі.

Для побудови АФЧХ треба викликати команду **nyquist**:

```
--> nyquist(sys)
```

```
--> xgrid
```

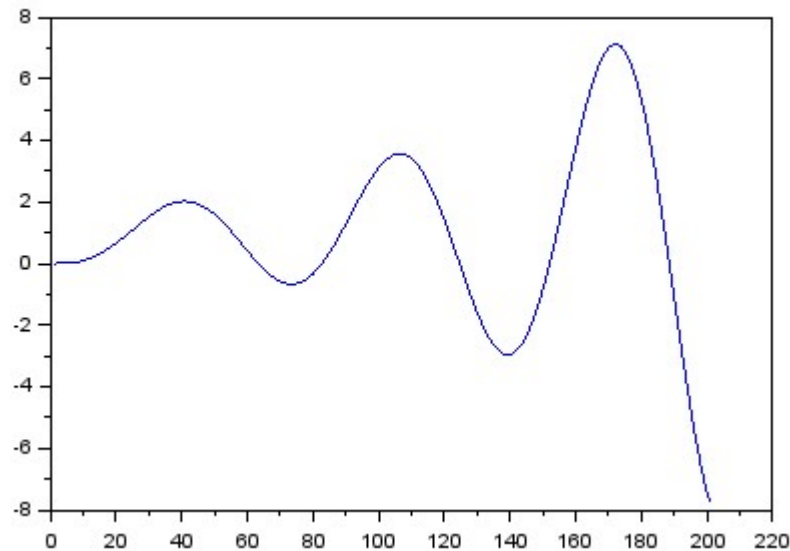


Рис. 4.2 – Перехідна реакція

Діаграма Найквіста показана на рис. 4.3. Як бачимо, АФЧХ жодного разу не охоплює точку  $(-1, j0)$ , тому і замкнута система буде нестійкою.

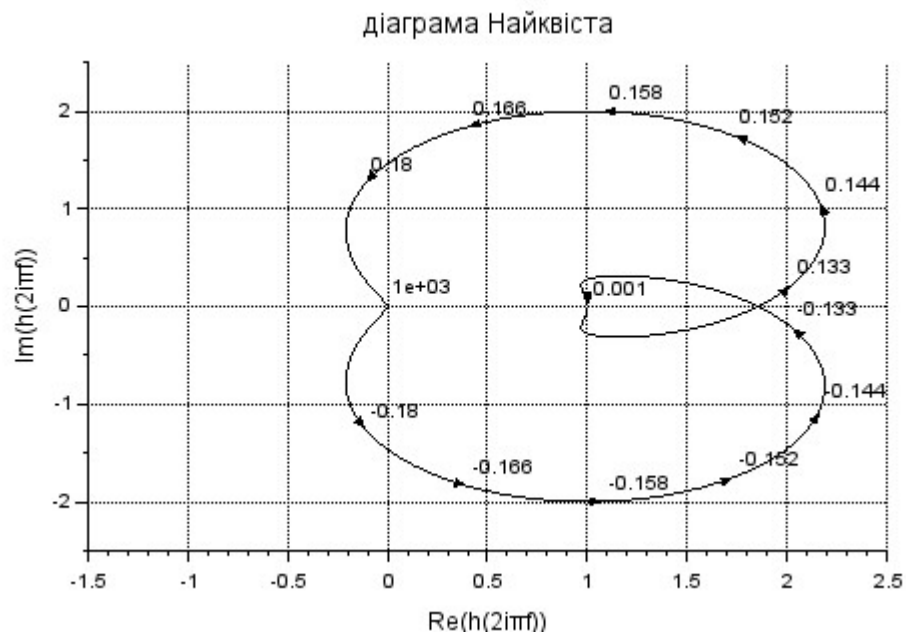


Рис. 4.3 – Діаграма Найквіста для задачі 4.3

**Задача 4.4.** САУ має перехідну характеристику

$$W = \frac{10}{2s^3 + 2s^2 + 3s + 1}.$$

Визначити запаси стійкості системи по амплітуді і по фазі.

**Розв'язування**

Використовуючи ЛАЧХ і ЛФЧХ, можна оцінити запаси стійкості системи по амплітуді і по фазі.

Надаємо змінній  $s$  значення системної змінної  $s$ , що використовується для створення поліномів:

```
--> s=s;
```

Створимо САУ, задану в умові задачі:

```
--> sys=syslin('c',10/(2*s^3+2*s^2+3*s+1))
```

```
sys =
```

```
10
```

```
-----
```

```
1 +3s +2s^2 +2s^3
```

Визначаємо запас стійкості системи по амплітуді:

```
--> [gm,fr]=g_margin(sys)
```

```
gm =
```

```
-13.979400
```

```
fr =
```

```
0.1949242
```

Тобто маємо запас по амплітуді  $\approx 14$  дБ на частоті  $\approx 0.195$  Гц.

Запас по фазі:

```
--> [pm,frp]=p_margin(sys)
```

```
pm =
```

```
-51.961006
```

```
frp =
```

```
0.3011768
```

Маємо запас  $\approx 52^\circ$  на частоті 0.3 Гц.

Покажемо на графіку:

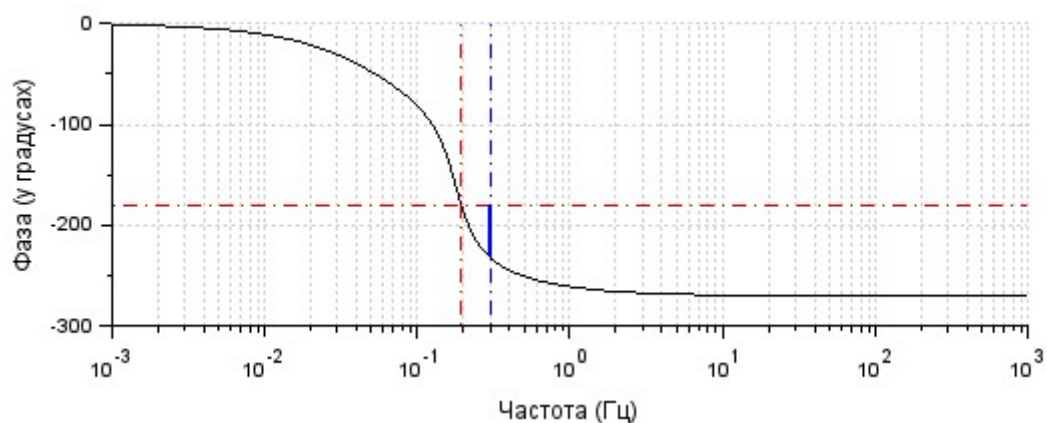
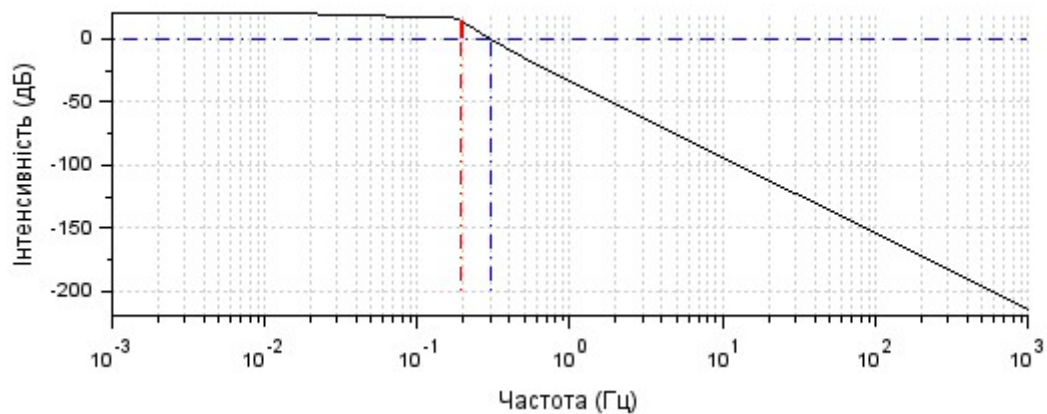
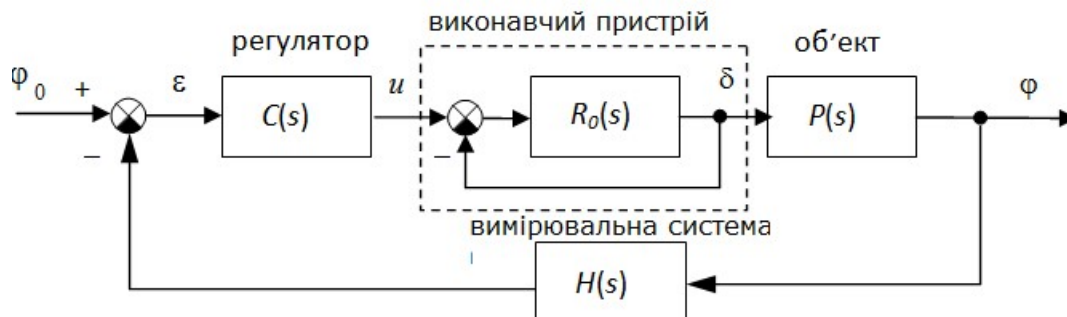


Рис. 4.4 – Визначення запасів стійкості по амплітуді і по фазі

**Задача 4.5.** Виконати дослідження стійкості замкнутої САУ методом Найквіста по заданій передатній функції розімкненої системи. Оцінити запаси стійкості по амплітуді і по фазі. Варіанти завдань приведені в табл. 4.1.

**Задача 4.6.** САУ має наступну структурну схему:



Передатна функція об'єкта:

$$P(s) = \frac{K}{s(T_s s + 1)}, \text{ где } K = 0.0694 \text{ рад/с}, T_s = 18.2 \text{ с}.$$

Виконавчий механізм являє собою інтегруючу ланку:

$$R_0(s) = \frac{1}{T_R s}, \quad T_R = 2 \text{ c},$$

охоплену одиничним негативним зворотним зв'язком. Модель вимірювального пристрою є аперіодичною ланкою з передатною функцією:

$$H(s) = \frac{1}{T_{oc}s + 1}, \quad T_{oc} = 6 \text{ c.}$$

Дослідити описану САУ.

## Розв'язування

1. Передатна функція об'єкта:

```
--> K = 0.0694;Ts=18.2;
--> P = syslin('c',K, s*(Ts*s+1))
P =
    0.0694
```

$$s + 18.2s^2$$

2. Передатна функція інтегруючої ланки:

```
--> TR=2; R0 = syslin('c',1, TR*s)
R0 =
    1
    --
    2s
```

### 3. Передатна функція виконавчого пристрою:

$$R = R_0 / (1 + \frac{R_0}{R_{\infty}})$$

$$\frac{1}{1 + 2s}$$

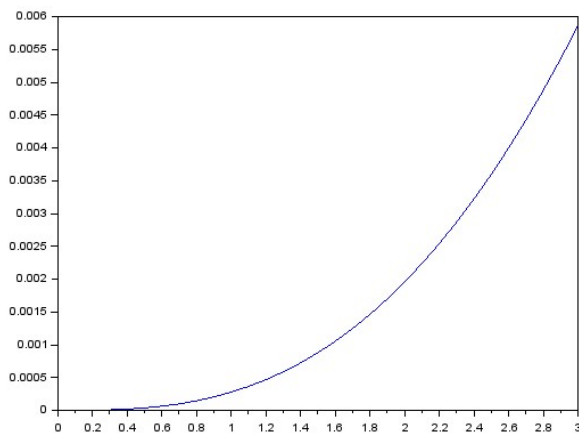
4. Передатна функція послідовного з'єднання об'єкта з виконавчим пристроєм:

```
--> G=P*R
```

$$G = \frac{0.0694}{s^2 + 20.2s + 36.4}$$

4. Перехідна характеристика для отриманої моделі:

```
--> plot(0:0.1:3, csim("step", 0:0.1:3, G))
```



5. Передатна функція вимірювального пристрою (датчика):

```
--> Tос=6; H = syslin('c', 1, Tос*s+1)
```

$$H = \frac{1}{1 + 6s}$$

6. Передатна функція розімкнутого контуру:

```
--> L = G*H
```

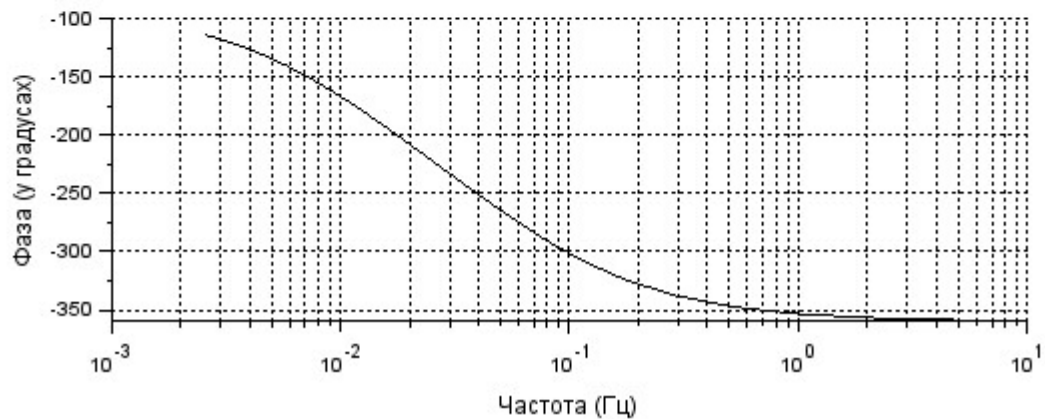
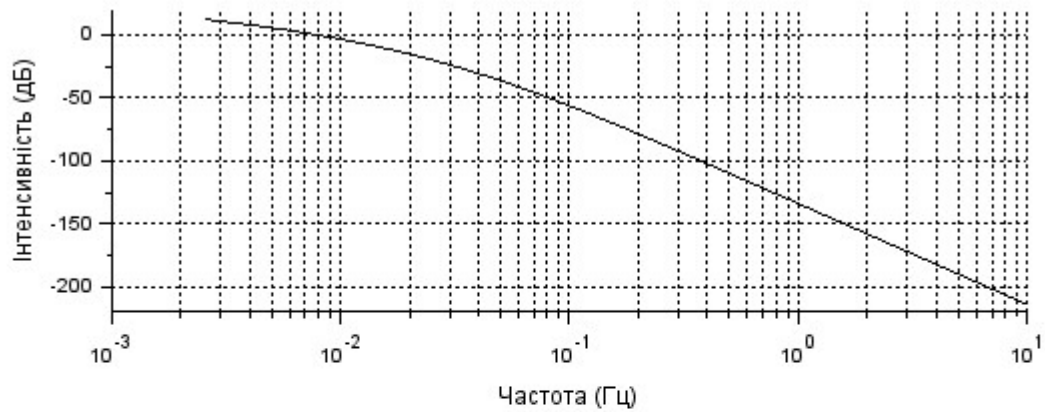
$$L = \frac{0.0694}{s^2 + 26.2s + 157.6s + 218.4}$$

7. Побудуємо діаграму Бодє в новому вікні:

```
--> figure(2);
```

```
--> bode(L, 0.001, 10)
```

```
--> xgrid; subplot(2,1,2); xgrid //нанесення сіток
```



На частоті зрізу (у точці перетину першого графіку з нульовим рівнем по осі ординат), згідно другого графіку, фаза не досягає рівня  $-180^\circ$ . Це означає, що система стійка.

8. Визначаємо запас стійкості по амплітуді:

```
--> [gm, fr] = g_margin(L)
gm =
    7.1156965
fr =
    0.0126777
```

Маємо запас стійкості 7,1 дБ на частоті 0,013 Гц, що є достатнім.

9. Визначаємо запас стійкості по фазі:

```
--> [phm, fr] = p_margin(L)
phm =
    25.993998
fr =
    0.0078457
```

Маємо запас стійкості  $26^\circ$  на частоті 0,008 Гц.

**Задача 4.7.** Дослідити САУ із задачі 4.6, згідно варіантів, наведених у табл. 4.2. Збережіть змінні робочого середовища для подальшого використання. Для цього використайте команду *save<ім'я файла>*.

Таблиця 4.2 – Вихідні дані для задачі 4.7

| Варіант | $T_s$ , с | $K$ , рад/с | $T_R$ , с | $T_{oc}$ , с |
|---------|-----------|-------------|-----------|--------------|
| 1       | 16.0      | 0.06        | 1         | 1            |
| 2       | 16.2      | 0.07        | 2         | 2            |
| 3       | 16.4      | 0.08        | 1         | 3            |
| 4       | 16.6      | 0.07        | 2         | 4            |
| 5       | 16.8      | 0.06        | 1         | 5            |
| 6       | 17.0      | 0.07        | 2         | 6            |
| 7       | 17.2      | 0.08        | 1         | 1            |
| 8       | 17.4      | 0.07        | 2         | 2            |
| 9       | 17.6      | 0.06        | 1         | 3            |
| 10      | 17.8      | 0.07        | 2         | 4            |
| 11      | 18.0      | 0.08        | 1         | 5            |
| 12      | 18.2      | 0.09        | 2         | 6            |
| 13      | 18.4      | 0.10        | 1         | 1            |
| 14      | 18.6      | 0.09        | 2         | 2            |
| 15      | 18.8      | 0.08        | 1         | 3            |
| 16      | 16.0      | 0.07        | 2         | 4            |
| 17      | 17.2      | 0.08        | 2         | 5            |
| 18      | 18.4      | 0.09        | 2         | 6            |
| 19      | 19.6      | 0.10        | 1         | 1            |
| 20      | 18.2      | 0.07        | 2         | 6            |

## 5 СИНТЕЗ РЕГУЛЯТОРІВ ЗАСОБАМИ SCILAB

### 5.1 Прості типи регуляторів

**П-регулятор.** Простий пропорційний регулятор (П-регулятор) є звичайним підсилювачем з передатною функцією

$$C(s) = K_c.$$

**ПІ-регулятор.** Якщо остаточна помилка регулювання є небажаною, в регулятор вводять інтегруючу складову:

$$C(s) = K_c \left( 1 + \frac{1}{T_I s} \right),$$

де  $T_I$  – стала часу інтегруючої ланки.

**ПД-регулятор.** Для поліпшення якості регулювання і підвищення швидкодії в закон управління вводять похідну від сигналу помилки:

$$C(s) = K_c (1 + T_D s),$$

де  $T_D$  – стала часу диференційної ланки. На практиці реалізувати ідеальне диференціювання неможливо, оскільки частотна характеристика ланки нескінченно збільшується на високих частотах. Тому диференційну ланку використовують з додатковим фільтром:

$$C(s) = K_c \left( 1 + \frac{T_D s}{T_f s + 1} \right).$$

Такий регулятор має назву ПДФ-регулятора (pdf-controller).

Тут постійна часу фільтру  $T_f$  зазвичай в 3–10 разів менше, ніж  $T_D$ . Надмірне збільшення  $T_D$  може привести до нестійкості системи, зменшення цієї величини затягує перехідний процес.

**ПІД-регулятор.** На відміну від ПД-регулятора, він містить інтегратор і система стає астатичною як по задаючій дії, так і по збуренню (тобто, постійне збурення повністю компенсується). Його передатна функція має вигляд

$$C(s) = K_c \left( 1 + T_D s + \frac{1}{T_I s} \right).$$

При збільшенні  $T_I$  перехідний процес затягується, при зменшенні – зменшується запас стійкості, перехідний процес набуває вираженого коливального характеру, при подальшому зменшенні  $T_I$  втрачається стійкість.

У зашумлених системах диференціальну ланку використовують з додатковим фільтром:

$$C(s) = K_c \left( 1 + \frac{T_D s}{T_f s + 1} + \frac{1}{T_I s} \right).$$

Такий регулятор має назву ПІДФ-регулятора (pidf-controller).

За допомогою правильно налагодженого ПІД- або ПІДФ-регулятора у більшості випадків вдається забезпечити виконання усіх вимог до системи. В силу своєї простоти, вони отримали найширше поширення. За статистикою більше 90% усіх промислових регуляторів є саме ПІД-регуляторами.

## 5.2 Кореневий годограф

Багато важливих властивостей системи (наприклад, швидкодія, перерегулювання) визначаються розташуванням коренів характеристичного рівняння на комплексній площині. Простий спосіб корекції системи – застосувати П-регулятор (підсилювач з коефіцієнтом  $K$ ), який змінює коефіцієнт посилення розімкненої системи і розташування цих коренів. На рис 5.1 показана система, що складається з об'єкта управління  $W_o(s)$  і регулятора  $K$ .

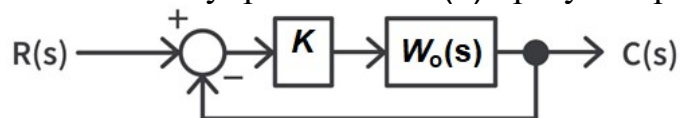


Рис. 5.1 – Система з П-регулятором

Передатна функція такої системи:

$$\frac{K W_o(s)}{1 + K W_o(s)}.$$

Характеристичне рівняння:

$$1 + K W_o(s) = 0 \text{ або } K = -1/W_o(s).$$

При зміні  $K$  від 0 до  $\infty$  корені рівняння описують на  $s$ -площині криві, які називаються *кореневим годографом*.

У пакеті Scilab кореневий годограф будується функцією `evans(sys [, kmax])`. Тут `sys` – дескриптор передатної функції об'єкта управління, `kmax`

(необов'язковий аргумент) використовується, коли треба обмежити найбільше значення для  $K$ . Початок годографа (при  $K = 0$ ) для кожного кореня позначений хрестиком.

На годографі курсором миші обирається точка, яку дослідник вважає найкращою для розміщення полюсів. Координати цієї точки відносно осей графіка [абсциса  $x$ ; ордината  $y$ ] визначає функція `locate` (1). Потім нам потрібно помножити  $[x; y]$  координати, що повертаються цією функцією, на  $[1, \%i]$ , щоб отримати положення точки в комплексній площині як  $x + jy$ .

Значення  $W_o(s)$  у вибраній точці можна отримати за допомогою функції `horner()`. Отже, для розрахунку  $K = -1/W_o(s)$  треба ввести у консолі Scilab команду:

```
--> K = -1/real(horner(sys, [ 1 %i ]*locate(1)))
```

Після цього відкриється вікно раніше створеного кореневого годографа, на якому мишею треба вказати бажану точку. І Scilab видасть квітанцію зі значенням коефіцієнта регулятора  $K$ .

### 5.3 Синтез методом частотних характеристик

У вітчизняній літературі класичним став метод синтезу пристроїв, що коригують, за допомогою логарифмічних амплітудно-фазових частотних характеристик (ЛАФЧХ) розімкненої системи (діаграм Боде за зарубіжною термінологією).

Нехай розімкнена система має передатну функцію  $W(s)$ . ЛАФЧХ включає дві криві – амплітудну частотну характеристику (ЛАЧХ)

$$L_m(\omega) = 20 \lg |W(j\omega)|$$

і фазову (ЛФЧХ)

$$\varphi(\omega) = \arg W(j\omega).$$

Низькочастотна частина ЛАЧХ визначає точність системи, середньочастотна (поблизу частоти зрізу  $\omega_c$ ) – стійкість і якість перехідного процесу, високочастотна – чутливість до завад. Якщо система містить інтегратор, низькочастотна частина має ненульовий нахил (20 дБ на декаду для одного інтегратора), постійний сигнал відстежується без помилки, що встановилася. Для системи з двома інтеграторами ЛАФЧХ має в області низьких частот нахил 40 дБ на декаду, без помилки, що встановилася, відстежується не лише постійний, але і лінійно зростаючий сигнал. Складніші вимоги до точності призводять до того, що ЛАЧХ не повинна заходити в деякі заборонені області.

*Запас стійкості за амплітудою  $g_m$*  (у дБ) – це відстань від ЛАЧХ до горизонтальної прямої  $L_m=0$  дБ на частоті, на якій фазова характеристика перетинає пряму  $\varphi=-180^\circ$ . На цій частоті система повинна мати коефіцієнт посилення менше 1 (чи  $L_m(\omega)<0$ ).

*Запас стійкості за фазою  $\varphi_m$*  (у градусах) – це відстань від частотної характеристики до горизонтальної прямої  $\varphi=-180^\circ$  на частоті зрізу  $\omega_c$ . На цій частоті фазова характеристика повинна мати значення *більше*, ніж  $-180^\circ$ .

Допустимим вважається запас за амплітудою не менше 6 дБ і запас за фазою не менше 30 градусів, інакше не гарантується стійкість системи.

Зазвичай вимагається, щоб система мала високу точність (великий коефіцієнт посилення по контуру) для низьких частот і пригнічувала високочастотні завади (мала низьке посилення в області високих частот). Частота зрізу вибирається виходячи з вимог до швидкодії.

Для забезпечення хорошої якості перехідних процесів рекомендується, щоб ЛАЧХ перетинала вісь  $L=0$  з нахилом 20 дБ/дек. Це пояснюється тим, що нахил 20 дБ/дек, що відповідає аперіодичній ланці, призводить до найменшої коливальності перехідного процесу. Точки переходу (зламу асимптотичної ЛАЧХ) від низькочастотної частини до середньочастотної і далі до високочастотної повинні знаходитися від осі  $L=0$  на 12...16 дБ.

У загальному випадку будується бажана ЛАЧХ  $L_d(j\omega)$ , що задовольняє вимогам до системи, потім ЛАЧХ послідовного коригуючого пристрою (регулятора) визначається як різниця між  $L_d(j\omega)$  і ЛАЧХ існуючої розімкненої системи. Вигляд бажаної ЛАЧХ показаний на рис. 5.2

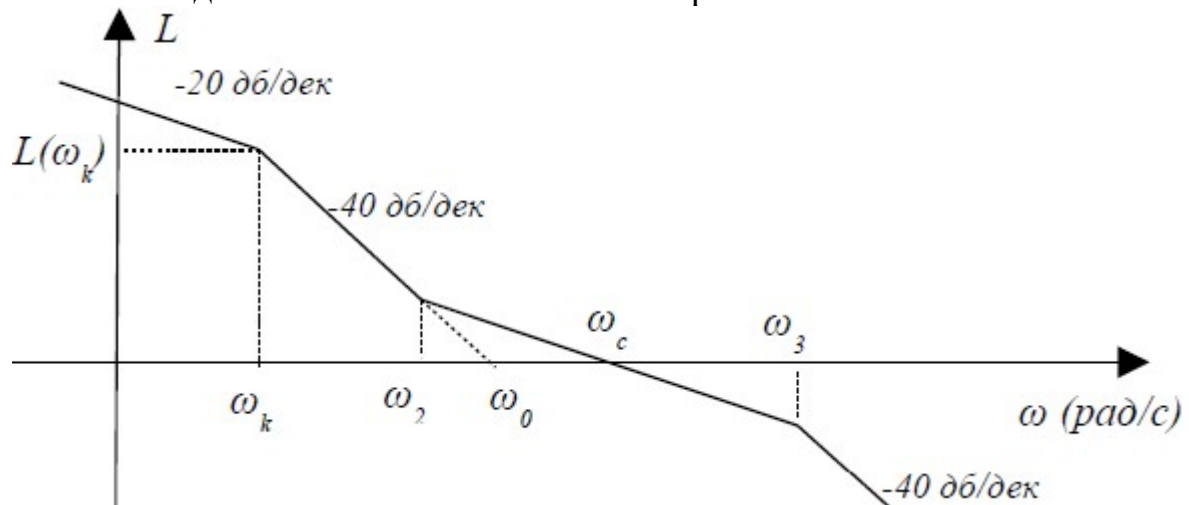


Рис. 5.2 – Вигляд бажаної ЛАЧХ

Такий ЛАЧХ відповідає наступна ПФ:

$$G(s) = \frac{k(\tau_2 s + 1)}{s(T_k s + 1)(T_3 s + 1)},$$

де  $k$  – коефіцієнт посилення; а постійні часу  $\tau_2$  і  $T$  відповідають сполучним частотам:

$$\tau_2 = \frac{1}{\omega_2}, \quad T_k = \frac{1}{\omega_k}, \quad T_3 = \frac{1}{\omega_3}.$$

Вихідні дані для вирішення задачі синтезу: передатна час перехідного процесу  $T_{пп}$ , перерегулювання  $\sigma$ , показник колювання  $M$ ; допустима динамічна помилка  $\epsilon_d$ , максимальна швидкість динамічного режиму  $\Omega_{max}$ , максимальне прискорення в динамічному режимі  $\epsilon_{max}$ .

На ЛАЧХ виділяється контрольна точка  $A_k$ . Цій точці відповідають частота і амплітуда:

$$\omega_k = \frac{\epsilon_{max}}{\Omega_{max}}; \quad \mathcal{E}_{max} = \frac{\Omega_{max}^2}{\epsilon_{max}}.$$

Для визначення меж середньочастотної ділянки вводиться поняття базової частоти:

$$\omega_0 = \sqrt{\frac{\varepsilon_{\max}}{x_{\max}}}.$$

За базовою частотою обчислюється частота зрізу:

$$\omega_c = \sqrt{\frac{M}{M-1}} \omega_0.$$

За частотою зрізу визначаються частоти  $\omega_2, \omega_3$ , відповідні початку та кінцю середньочастотної ділянки:

$$\omega_2 \leq \frac{M-1}{M} \omega_c; \quad \omega_3 \geq \frac{M+1}{M} \omega_c \cdot 2.$$

Необхідна кількість частот, що сполучають ділянки, відповідає порядку вихідної передатної функції. Значення цих частот у високочастотній ділянці вибирають виходячи з наступних умов:

$$\omega_4 \geq \omega_3 \cdot 2; \quad \omega_5 \geq \omega_4 \cdot 2 \dots$$

Коефіцієнт передачі бажаної системи:

$$k = 10^{\frac{L_G(\omega=1)}{20}}.$$

У загальному випадку структура ПФ, що відповідає бажаній ЛАХ, матиме вигляд:

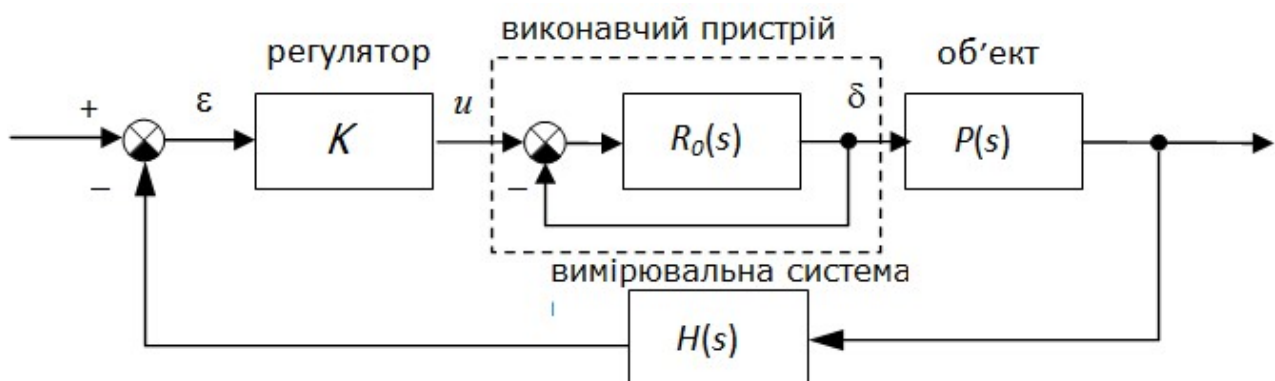
$$G(s) = \frac{k \cdot \left( \frac{1}{\omega_2} s + 1 \right)}{s \left( \frac{1}{\omega_k} s + 1 \right) \left( \frac{1}{\omega_3} s + 1 \right) \left( \frac{1}{\omega_4} s + 1 \right) \left( \frac{1}{\omega_5} s + 1 \right) \dots}.$$

Якщо вихідна (наявна) динамічна система має передатну функцію  $W(s)$ , а ПФ регулятора позначити  $K(s)$ , то ПФ бажаної системи  $G(s) = W(s)K(s)$ . Звідси шукана ПФ регулятора:

$$K(s) = \frac{G(s)}{W(s)}.$$

### 5.3 Приклади і задачі для самостійного розв'язання

**Задача 5.1.** САУ має наступну структурну схему:



Передатна функція об'єкта

$$P(s) = \frac{K_o}{s(T_s s + 1)}, \text{ де } K_o = 0,0694 \text{ рад/с, } T_s = 18.2 \text{ с.}$$

Виконавчий механізм являє собою інтегруючу ланку

$$R_0(s) = \frac{1}{T_R s}, \quad T_R = 2 \text{ с,}$$

охоплену одиничним негативним зворотним зв'язком. Модель вимірювального пристрою є аперіодичною ланкою з передатною функцією

$$H(s) = \frac{1}{T_{oc} s + 1}, \quad T_{oc} = 6 \text{ с.}$$

Спроекувати П-регулятор методом кореневого годографа.

### Розв'язування

1. Передатна функція об'єкта

```
--> s=%s; P=syslin('c', 0.0694/(18.2*s^2+s))
P =
    0.0694
```

```
-----
s +18.2s^2
```

2. Передатна функція інтегруючої ланки

```
--> R0 = syslin('c', 1/(2*s))
R0 =
```

```
1
--
2s
```

3. Передатна функція виконавчого пристрою з урахуванням зворотного зв'язку:

```
--> R=R0/. 1
```

```
R =
    1
-----
1 +2s
```

4. Передатна функція вимірювального пристрою (датчика)

```
--> H = syslin('c', 1/(6*s+1))
```

```
H =
    1
-----
1 +6s
```

5. Передатна функція розімкнутого контуру (послідовне з'єднання виконавчого пристрою, об'єкта і датчика)

```
--> Wo = R*P*H
```

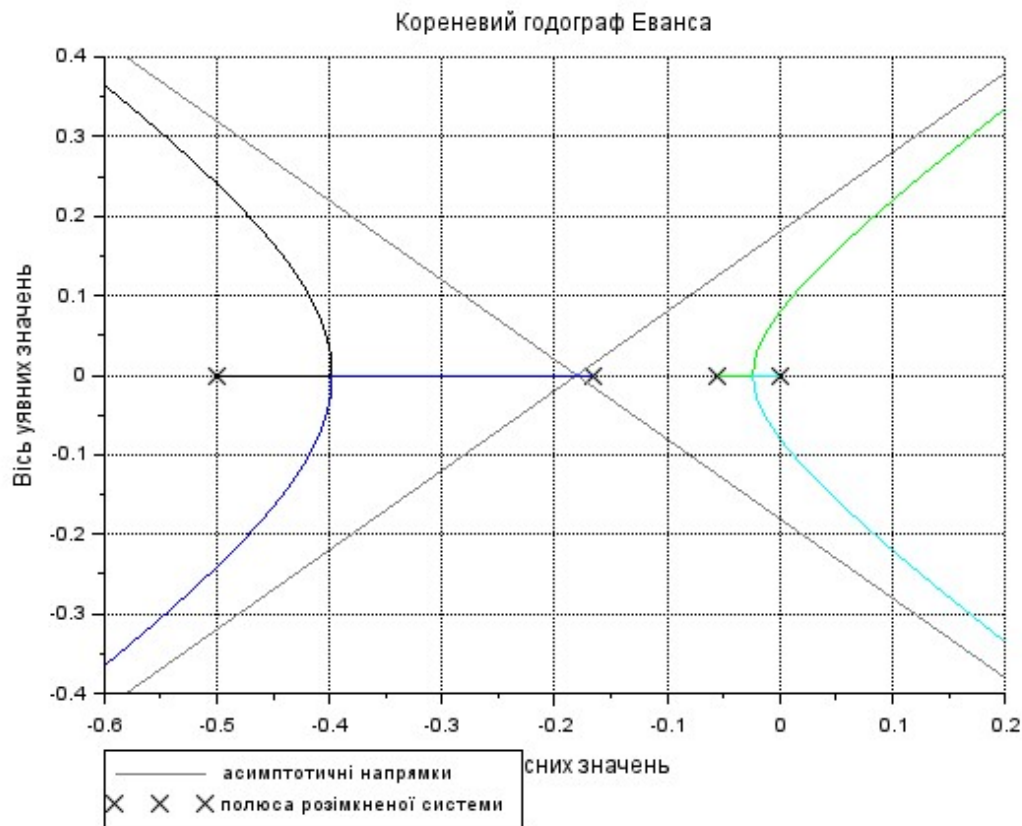
```
Wo =
    0.0694
-----
s +26.2s^2 +157.6s^3 +218.4s^4
```

6. Будуємо кореневий годограф:

```
--> evans(Wo); xgrid
```

7. Виділимо на графіку значущу область для зручності подальшого опрацювання, для чого вкажемо граничні значення для осей абсцис і ординат  $[X_{min}, X_{max}, Y_{min}, Y_{max}]$ :

```
--> mtlb_axis([-0.6 0.2 -0.4 0.4])
```



8. На графіку хрестиками показані початкові положення полюсів системи. Позначимо їх як  $p_1, p_2, p_3$  і  $p_4$ , рахуючи з лівого боку до правого. При збільшенні коефіцієнта передачі полюси рухаються вздовж кольорових траєкторій. Як бачимо, критичними для стійкості системи є полюси  $p_3$  і  $p_4$ , їх траєкторії уходять на праву півплощину, що веде до втрати стійкості. За робочу приймемо точку, де траєкторії цих полюсів сходяться.

Вводимо команду:

```
--> K=-1/real(horner(Wo,[1 %i]*locate(1)))
```

9. Клацаємо мишею у точці на графіку, де сходяться годографи  $p_3$  і  $p_4$ :

```
K =  
0.1584742
```

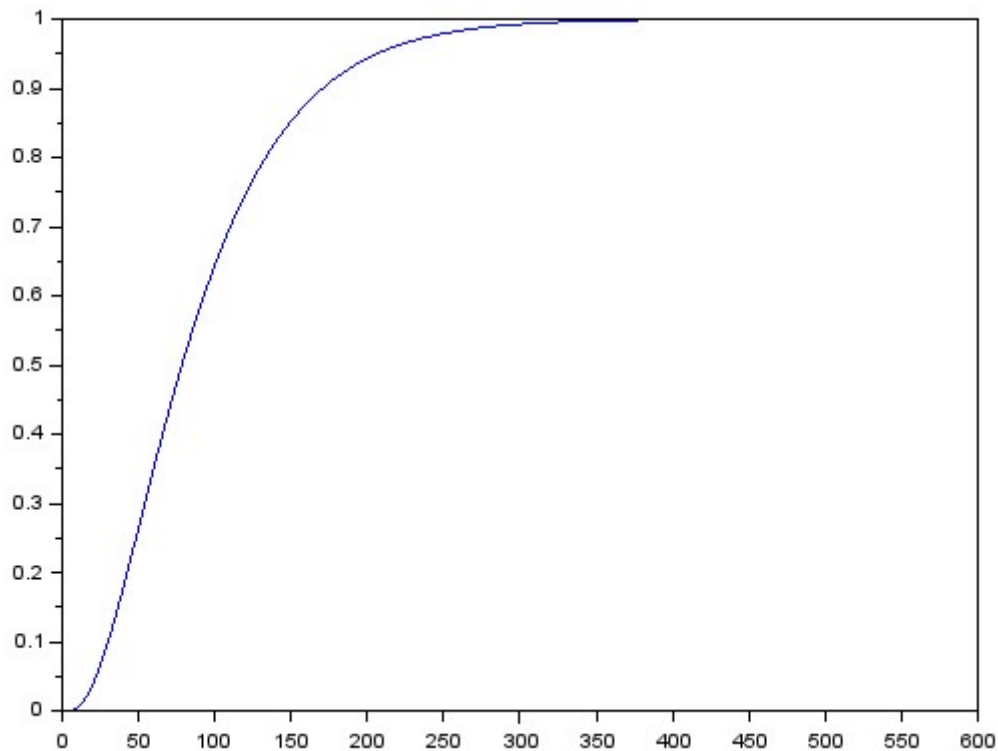
10. Передатна функція з'єднання  $K$  і  $W_o$ , охоплених негативним зворотним зв'язком:

```
--> sys=(K*Wo) /. 1  
sys =
```

```
0.0109981
```

```
-----  
0.0109981 +s +26.2s^2 +157.6s^3 +218.4s^4
```

11. Будуємо графік перехідного процесу:



**Задача 5.2.** За зразком задачі 5.1 виконати синтез регулятора згідно варіантів, наведених у табл. 5.1.

Таблиця 5.1 – Вихідні дані для задачі 5.2

| Варіант | $T_s$ , с | $K_o$ , рад/с | $T_R$ , с | $T_{oc}$ , с | Варіант | $T_s$ , с | $K_o$ ,<br>рад/с | $T_R$ , с | $T_{oc}$ , с |
|---------|-----------|---------------|-----------|--------------|---------|-----------|------------------|-----------|--------------|
| 1       | 16.0      | 0.06          | 1         | 1            | 14      | 18.6      | 0.09             | 2         | 2            |
| 2       | 16.2      | 0.07          | 2         | 2            | 15      | 18.8      | 0.08             | 1         | 3            |
| 3       | 16.4      | 0.08          | 1         | 3            | 16      | 16.0      | 0.07             | 2         | 4            |
| 4       | 16.6      | 0.07          | 2         | 4            | 17      | 17.2      | 0.08             | 2         | 5            |
| 5       | 16.8      | 0.06          | 1         | 5            | 18      | 18.4      | 0.09             | 2         | 6            |
| 6       | 17.0      | 0.07          | 2         | 6            | 19      | 19.6      | 0.10             | 1         | 1            |
| 7       | 17.2      | 0.08          | 1         | 1            | 20      | 18.2      | 0.07             | 2         | 6            |
| 8       | 17.4      | 0.07          | 2         | 2            | 21      | 18.5      | 0.12             | 1         | 4            |
| 9       | 17.6      | 0.06          | 1         | 3            | 22      | 19.1      | 0.08             | 2         | 3            |
| 10      | 17.8      | 0.07          | 2         | 4            | 23      | 19.4      | 0.09             | 2         | 6            |
| 11      | 18.0      | 0.08          | 1         | 5            | 24      | 17.6      | 0.10             | 1         | 2            |
| 12      | 18.2      | 0.09          | 2         | 6            | 25      | 18.0      | 0.0694           | 3         | 3            |
| 13      | 18.4      | 0.10          | 1         | 1            |         |           |                  |           |              |

**Задача 5.3.** Для об'єкта управління з передатною функцією

$$W(s) = \frac{20}{s(0.1s + 1)(0.05s + 1)}$$

методом частотних характеристик синтезувати регулятор, що забезпечує виконання наступних умов:

- максимальна швидкість системи  $\Omega_{\max} = 30$  град/с;
- максимальне прискорення  $\epsilon_{\max} = 0,5$  град/с<sup>2</sup>;
- максимальна помилка регулювання  $x_{\max} = 0,033$  град;

- показник коливальності  $M = 1,5$ ;
- допустиме перерегулювання  $\sigma \leq 30 \%$ ;
- час перехідного процесу  $T_{\text{пн}} \leq 1$  с.

### Розв'язування

Задаємо умови задачі:

```
--> Xmax=0.033; Ωmax=30; εmax=30; M=1.5;
```

Параметри ПФ об'єкта:

```
ko=20; To1=0.1; To2=0.05;
```

Частота та амплітуда гармонійного впливу:

```
--> ωk=εmax/Ωmax; gmax=Ωmax^2/εmax;
```

Базова частота:

```
--> ω0=sqrt(εmax/Xmax);
```

Частота зрізу:

```
--> ωс=sqrt(M/(M-1))*ω0;
```

Межі середньочастотної області:

```
--> ω2=(M-1)/M*ωс, ω3=(M+1)/M*ωс*2
```

```
ω2 =
17.407766
```

```
ω3 =
174.07766
```

Вибираємо такі значення:

```
--> ω2=10; ω3=200; ω4=2*ω3;
```

Бажаний коефіцієнт передачі:

```
--> k= gmax/Xmax
```

```
k =
909.09091
```

Сталі часу бажаної системи:

```
--> τ2=1/ω2; Tk=1/ωk; T3=1/ω3; T4=1/ω4
```

ПФ бажаної системи:

```
--> s=poly(0,'s');
```

```
--> numG=k*(τ2*s+1)
```

```
--> denG=s*(Tk*s+1)*(T3*s+1)*(T4*s+1)
```

```
-> G=syslin('c',numG/denG)
```

```
G =
          909.09091 +90.909091s
-----
s +1.0075s^2 +0.0075125s^3 +0.0000125s^4
```

ПФ об'єкта управління:

```
--> W=syslin('c', ko/(s*(To1*s+1)*(To2*s+1)))
```

```
W =
          20
-----
s +0.15s^2 +0.005s^3
```

ПФ регулятора

--> K=G/W

K =

$$\frac{909.09091 + 227.27273s + 18.181818s^2 + 0.4545455s^3}{20 + 20.15s + 0.15025s^2 + 0.00025s^3}$$

Перевіряємо роботу синтезованого регулятора K з об'єктом W. ПФ системи, охопленої негативним зворотним зв'язком:

--> sys=(K\*W) /. 1

sys =

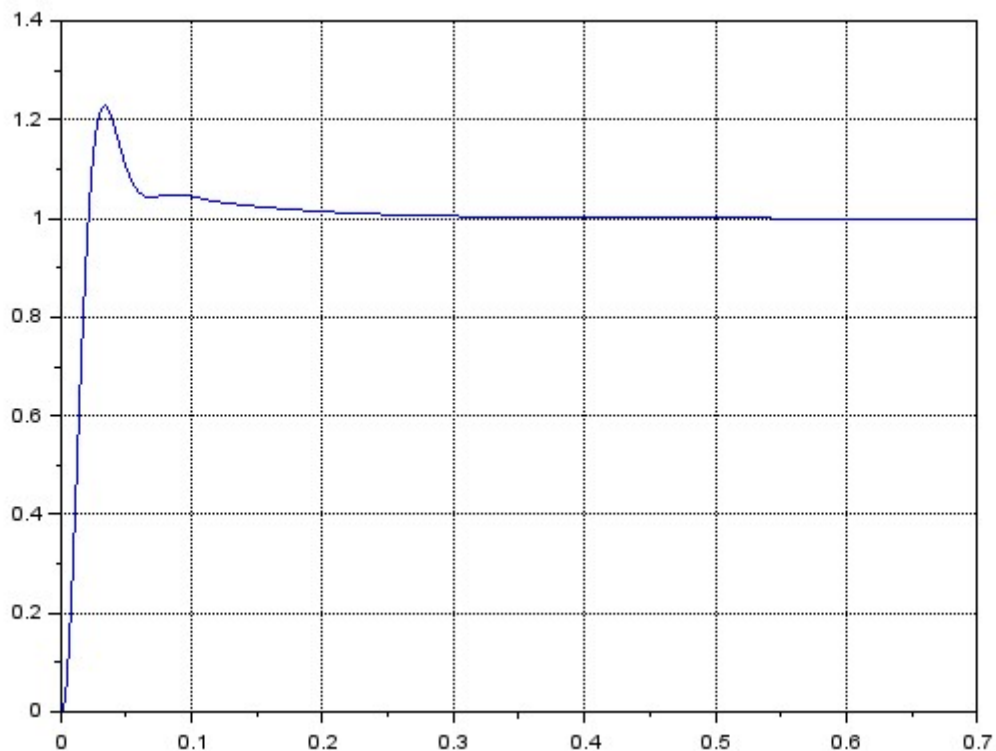
$$\frac{72727273 + 7272727.3s}{72727273 + 7352727.3s + 80600s^2 + 601s^3 + s^4}$$

$$\frac{72727273 + 7272727.3s}{72727273 + 7352727.3s + 80600s^2 + 601s^3 + s^4}$$

Побудуємо графік перехідної характеристики:

--> t=0:0.001:1;

--> plot(t,csim("step",t,sys)); xgrid



З графіка отримуємо параметри перехідного процесу:

а) перерегулювання

$$\sigma = \frac{h_{\max} - h_{уст}}{h_{уст}} 100 = \frac{1.23 - 1}{1} 100 = 23 \%;$$

б) час перехідного процесу  $T_{\text{шт}}=0.227$  с при входженні в один відсотковий коридор.

Отримані результати задовольняють умовам задачі.

**Задача 5.4.** За зразком задачі 5.3 виконати синтез регулятора згідно варіантів, наведених у табл. 5.2.

Таблиця 5.2 – Вихідні дані для задачі 5.4

| Варі-<br>ант | Передатна функція                 | $T_{пп}, c$ | $M$  | $\sigma, \%$ | $X_{max},$<br>град | $\Omega_{max},$<br>град/с | $\varepsilon_{max},$<br>град/с <sup>2</sup> |
|--------------|-----------------------------------|-------------|------|--------------|--------------------|---------------------------|---------------------------------------------|
| 1            | $\frac{250}{s(0.1s+1)(0.02s+1)}$  | 0.8         | 1.35 | 30           | 0.2                | 25                        | 20                                          |
| 2            | $\frac{100}{s(0.04s+1)(0.01s+1)}$ | 0.6         | 1.19 | 20           | 0.01               | 0.2                       | 0.6                                         |
| 3            | $\frac{100}{s(0.05s+1)(0.02s+1)}$ | 0.45        | 1.35 | 30           | 0.1                | 2                         | 8                                           |
| 4            | $\frac{72}{s(0.05s+1)(0.2s+1)}$   | 0.8         | 1.30 | 25           | 0.2                | 25                        | 20                                          |
| 5            | $\frac{100}{s(0.05s+1)(0.1s+1)}$  | 0.5         | 1.19 | 20           | 0.01               | 0.2                       | 0.5                                         |
| 6            | $\frac{160}{s(0.1s+1)(0.02s+1)}$  | 0.4         | 1.3  | 25           | 0.1                | 2                         | 6                                           |
| 7            | $\frac{50}{s(0.04s^2+0.2s+1)}$    | 0.7         | 1.25 | 20           | 0.2                | 15                        | 10                                          |
| 8            | $\frac{100}{s(0.25s^2+0.5s+1)}$   | 0.6         | 1.3  | 25           | 0.1                | 2                         | 0.8                                         |
| 9            | $\frac{32}{s(0.05s+1)(0.1s+1)}$   | 0.8         | 1.35 | 30           | 0.2                | 25                        | 12                                          |
| 10           | $\frac{200}{s(0.1s+1)(0.02s+1)}$  | 0.6         | 1.19 | 20           | 0.01               | 0.2                       | 0.6                                         |
| 11           | $\frac{100}{s(0.08s+1)(0.03s+1)}$ | 0.45        | 1.35 | 30           | 0.1                | 2                         | 8                                           |
| 12           | $\frac{160}{s(0.1s+1)(0.02s+1)}$  | 0.8         | 1.30 | 25           | 0.2                | 25                        | 2                                           |
| 13           | $\frac{100}{s(0.08s+1)(0.03s+1)}$ | 0.5         | 1.19 | 20           | 0.01               | 0.2                       | 0.5                                         |
| 14           | $\frac{160}{s(0.1s+1)(0.02s+1)}$  | 0.4         | 1.3  | 25           | 0.1                | 2                         | 6                                           |
| 15           | $\frac{160}{s(0.1s+1)(0.02s+1)}$  | 0.7         | 1.25 | 20           | 0.2                | 15                        | 32                                          |
| 16           | $\frac{50}{s(0.04s^2+0.2s+1)}$    | 0.6         | 1.3  | 25           | 0.1                | 2                         | 0.8                                         |
| 17           | $\frac{50}{s(0.04s^2+0.2s+1)}$    | 0.1         | 1.3  | 30           | 0.1                | 0.2                       | 0.8                                         |
| 18           | $\frac{100}{s(0.25s^2+0.5s+1)}$   | 0.2         | 1.3  | 30           | 0.2                | 0.4                       | 0.8                                         |
| 19           | $\frac{100}{s(0.25s^2+0.5s+1)}$   | 0.1         | 1.35 | 30           | 0.01               | 0.2                       | 0.8                                         |
| 20           | $\frac{100}{s(0.25s^2+0.5s+1)}$   | 0.2         | 1.35 | 30           | 0.02               | 0.4                       | 0.8                                         |

## 6 МОДАЛЬНЕ УПРАВЛІННЯ

### 6.1 Короткі теоретичні положення

Розглянемо одновимірну динамічну систему (SISO – Single Input Single Output), модель якої у просторі станів має вигляд:

$$\begin{cases} \frac{dX(t)}{dt} = AX(t) + Bu(t), \\ y(t) = CX(t) + Du(t), \end{cases} \quad (6.1)$$

де  $X(t)$  – вектор-стовпчик стану розмірності  $[n \times 1]$ ;  $A$  – матриця коефіцієнтів об'єкта  $[n \times n]$ ;  $B$  – матриця входів  $[n \times 1]$ ;  $u(t)$  – сигнал управління;  $Y$

– вектор виходів  $[k \times 1]$ ;  $\mathbf{C}$  – матриця виходів  $[1 \times n]$ ;  $\mathbf{D}$  – матриця обходу, тобто впливу входу безпосередньо на вихід системи  $[n \times 1]$ . Звичайно  $\mathbf{D} = 0$ .

Система, описувана матрицями  $\mathbf{A}$  і  $\mathbf{B}$ , є керованою, якщо вона може бути переведена з довільного початкового стану  $x_0$ , у момент часу  $t_0$ , у довільний кінцевий стан  $x_1$ , у момент часу  $t_1$ , за обмежений проміжок часу  $[t_0, t_1]$ , за допомогою обмеженого управління  $u(t)$ , визначеного на інтервалі  $t_0 < t < t_1$ .

Для SISO-системи з одним входом і одним виходом вводиться поняття матриці керованості (розміром  $n \times n$ ):

$$\mathbf{Q} = [\mathbf{B} \quad \mathbf{AB} \quad \mathbf{A}^2\mathbf{B} \quad \dots \quad \mathbf{A}^{n-1}\mathbf{B}] \quad (6.2)$$

Система (6.1) повністю керована тоді і тільки тоді, коли ранг матриці керованості (6.2) дорівнює порядку системи:  $\text{rank}(\mathbf{Q}) = n$ .

Нагадаємо, що під рангом матриці мається на увазі найвищий з порядків відмінного від нуля мінору цієї матриці. Ранг матриці дорівнює найбільшому числу лінійно незалежних рядків.

Модальне управління припускає формування таких зворотних зв'язків, при яких забезпечується задане розташування полюсів замкнутої системи. Модою називається та складова рішення диференціального рівняння, що є відповідною одному конкретному полюсу.

Розташування полюсів в основному визначає характер перехідного процесу в системі. Звичайно розглядаються такі кореневі оцінки якості перехідного процесу, як час перехідного процесу, ступінь стійкості, коливальність і перерегулювання. Для оцінки швидкодії системи використовується поняття ступеня стійкості  $\eta$ , під яким розуміється абсолютне значення дійсної частини найближчого до уявної осі кореня (тому що корені, що мають якнайменшу по модулю дійсну частину, дають в перехідному процесі поволі самозатухаючу складову). Час перехідного процесу  $t_n$  можна приблизно оцінити по формулі

$$t_n \approx \frac{3}{\eta}. \quad (6.3)$$

Запас стійкості системи оцінюється коливальністю. Система має схильність до коливань, якщо характеристичне рівняння містить комплексні корені  $\eta_{1,2} = -\alpha \pm j\beta$ . Коливальність оцінюється формулою:

$$\mu = \frac{\beta}{\alpha}. \quad (6.4)$$

По значенню коливальності можна оцінити перерегулювання:

$$\delta \leq e^{-\frac{\pi}{\mu}} \cdot 100\%. \quad (6.5)$$

Охопимо систему (6.1) негативним зворотним зв'язком (рис. 6.1). У цій системі  $g(t)$  – задаючий вплив, управління по стану описується виразом

$$u(t) = -\mathbf{K}\mathbf{X}(t), \quad (6.6)$$

де  $\mathbf{K}$  – вектор коефіцієнтів зворотного зв'язку.

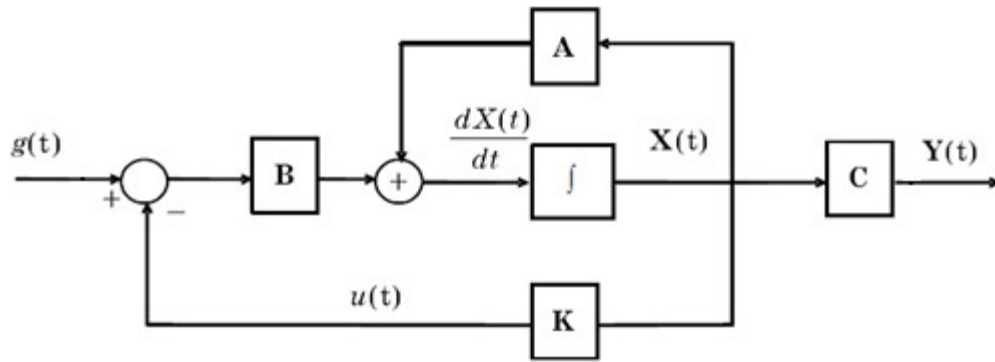


Рис. 6.1 – Структура системи зі зворотним зв'язком

Таким чином, система, замкнута регулятором, приводиться до наступного вигляду:

$$\frac{dX(t)}{dt} = (A - BK)X(t). \quad (6.7)$$

Основна теорема модального управління свідчить, що якщо лінійна динамічна система (6.1) є керованою, то лінійний зворотний зв'язок може бути вибраний таким чином, що матриця  $(A - BK)$  забезпечить бажане розташування коренів.

Аккерманом була запропонована формула, що дозволяє з допомогою перетворення подібності перевести модель довільної структури в канонічну форму керованості, визначити шукані коефіцієнти  $K$ , а потім перерахувати отримане рішення стосовно початкової структури. Формула Аккермана має вигляд:

$$K = [0 \ 0 \ \dots \ 0 \ 1] \cdot [B \ AB \ A^2B \ \dots \ A^{n-1}B]^{-1} \times \\ \times [A^n \ \beta_{n-1}A^{n-1} \ \dots \ \beta_1A \ \beta_0I], \quad (6.8)$$

де  $\beta_i$  – коефіцієнти характеристичного полінома матриці  $(A - BK)$ . Таким чином, задача модального синтезу зводиться до вибору бажаних коренів характеристичного полінома замкнутої системи, при яких забезпечуються задані параметри перехідного процесу, після чого відповідно до стандартного алгоритму розраховуються коефіцієнти зворотних зв'язків по стану.

Матриця керованості може бути побудована за допомогою функції **cont\_mat**, яка викликається однією з команд:

```
--> Q = cont_mat(A, B)
--> Q = cont_mat(sys)
--> Q = cont_mat(sys.A, sys.B)
```

В пакеті Scilab є функція **stabil**, за допомогою якої можна забезпечити бажане розташування полюсів лінійної системи (відповідно до формули Аккермана):

```
--> k = stabil(A, B, P),
```

де  $A$  та  $B$  – матриці системи;  $P$  – вектор, що задає бажане розміщення полюсів системи.

Для генерації матриці  $K$  використовується вбудована Scilab-функція **rpol**( $A, B, P$ ), яка за заданими матрицями  $A$  та  $B$  та вектором полюсів підбирає елементи матриці  $K$  так, щоб власні значення матриці  $(A - BK)$  були полюсами.

Слід зауважити, що метод модального управління не гарантує рівність сталої помилки нулю. Для забезпечення рівності задаючої дії і вихідного сигналу системи в сталому режимі вводиться коефіцієнт масштабування  $k_0$ . На цей коефіцієнт повинна множитися вхідна дія.

## 6.2 Приклади і задачі для самостійного розв'язання

**Задача 6.1.** Система описується матрицями

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -8 & -9 \end{bmatrix}; \mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}; \mathbf{C} = [8 \quad 0 \quad 0]; \mathbf{D} = 0.$$

Обґрунтувати можливість модального управління за допомогою критерію керованості.

### Розв'язування

Маємо систему третього порядку:  $n=3$ . Визначаємо матриці системи у робочому просторі SCILAB за допомогою команд

```
-->A=[0 1 0; 0 0 1; 0 -8 -9]; B=[0; 0; 1];
```

Будуємо матрицю керованості:

```
--> Cc = cont_mat(A, B)
```

```
Cc =
```

```
0.    0.    1.
0.    1.   -9.
1.   -9.   73.
```

Визначаємо ранг матриці керованості:

```
-->rank(Cc)
```

```
ans =
```

```
3
```

Оскільки  $\text{rank}(C_c)=n$ , система є керованою, і модальне управління можливе.

**Задача 6.2.** Розрахувати значення коефіцієнтів зворотних зв'язків для об'єкта управління з задачі 6.1. Бажані полюси задати вектором  $\mathbf{P}=[-1, -2, -3]$ .

### Розв'язування

Розрахувати значення коефіцієнтів зворотних зв'язків можна за допомогою команд

```
-->A=[0 1 0; 0 0 1; 0 -8 -9]; B=[0; 0; 1];
```

```
-->P=[-1, -2, -3];
```

```
-->K= stabil(A, B, P)
```

```
K =
```

```
-6.   -3.    3.
```

Таким чином, рівняння управління має бути сформоване у вигляді:

$$u(t) = -[K][X(t)] = \begin{bmatrix} 6 & 3 & -3 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} = 6x_1(t) + 3x_2(t) - 3x_3(t).$$

**Задача 6.3.** Для об'єкта управління з задачі 6.1 з урахуванням результатів задачі 6.2 визначити передатну функцію системи модального управління і коефіцієнт масштабування та побудувати графік перехідного процесу.

### Розв'язування

Після розв'язання задачі 6.2 вводимо матрицю C:

```
-->C=[8 0 0]; D=0;
```

Матриця A коефіцієнтів замкненої системи:

```
--> As=A+B*K;
```

Модель замкненої системи:

```
-->sys = syslin('c', As, B, C, D);
```

Передатна функція:

```
--> W=ss2tf(sys)
W =
      8 -1.705D-13s -4.263D-14s^2
-----
      6 +11s +6s^2 +s^3
```

Розрахунок перехідного процесу

```
--> t=0:0.1:10;
--> out=csim("step", t, sys);
```

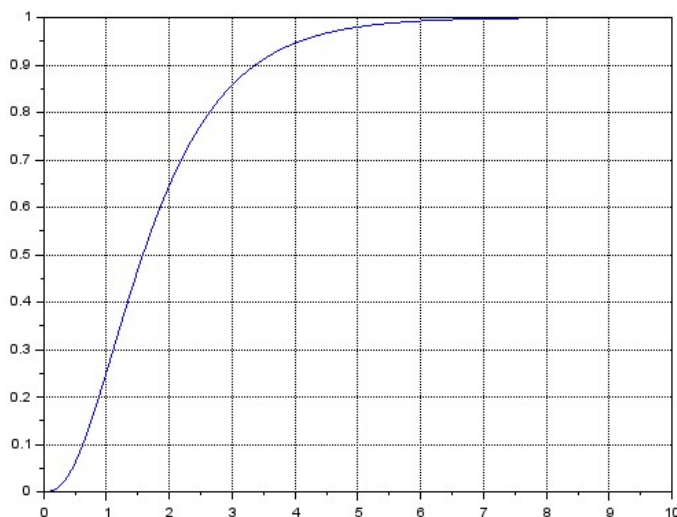
Для обчислення коефіцієнта масштабування  $k_0$

```
--> k0=1/out($)
```

k0 =  
0.7501022

Будуємо графік перехідного процесу:

```
-->plot(t, k0*out);xgrid
```



**Задача 6.4.** Для об'єкта управління, що описується матрицями **A**, **B**, **C** (див. Додаток А) визначити можливість модального управління і, у випадку позитивного висновку, синтезувати систему модального управління:

- а) розрахувати значення коефіцієнтів зворотних зв'язків системи;
- б) визначити передатну функцію системи модального управління;
- в) розрахувати значення коефіцієнта масштабування, на який повинна множитися вхідна дія для забезпечення рівності задаючої дії і вихідного сигналу системи в сталому режимі;
- г) побудувати графік перехідного процесу.

## 7 СПОСТЕРІГАЧІ СТАНУ

### 7.1 Короткі теоретичні положення

Розглянемо динамічну систему, представлену у просторі станів:

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}; \\ \mathbf{y} = \mathbf{C}\mathbf{x}. \end{cases}$$

Метод модального управління припускає, що усі компоненти вектора змінних стану  $\mathbf{x}$  можуть бути виміряні. Проте на практиці деякі компоненти можуть бути невідомі по одній з причин:

- вимірювальних приладів може бути недостатньо;
- деякі компоненти вектора  $\mathbf{x}$  можуть не мати фізичного змісту.

Проте для деяких систем усі компоненти вектора  $\mathbf{x}$  можуть бути відновлені за спостереженнями векторів входу  $\mathbf{u}$  і виходу  $\mathbf{y}$ .

Система називають спостережуваною тоді і тільки тоді, коли існує кінцевий час  $T$  такий, що початковий стан  $\mathbf{x}(0)$  може бути визначений в результаті спостереження вихідної змінної  $\mathbf{y}(t)$  при заданому управлінні  $\mathbf{u}(t)$ .

Спостережувана система дозволяє користувачам робити висновки про внутрішній стан складної системи за її вхідними і вихідними даними.

Спостережуваність системи описується умовою  $\text{rank}(\mathbf{N}) = n$ , де  $\mathbf{N}$  – матриця спостережуваності розміром  $n \times n$ :

$$\mathbf{N} = [\mathbf{C}^T \quad \mathbf{A}^T \mathbf{C}^T \quad (\mathbf{A}^T)^2 \mathbf{C}^T \quad \dots \quad (\mathbf{A}^T)^{n-1} \mathbf{C}^T]. \quad (7.1)$$

Для системи з одним входом і одним виходом система спостережувана, якщо детермінант цієї матриці відмінний від нуля.

Для того, щоб визначити усі компоненти вектора стану об'єкта, можна використовувати його модель

$$\frac{d\hat{\mathbf{x}}(t)}{dt} = \mathbf{A}\hat{\mathbf{x}}(t) + \mathbf{B}\mathbf{u}(t), \quad (7.2)$$

де  $\hat{\mathbf{x}}(t)$  – оцінка стану об'єкта.

Якщо початковий стан об'єкта і моделі співпадають і модель адекватна об'єктові, то можна вважати у будь-який момент часу, що  $\hat{\mathbf{X}}(t) = \mathbf{X}(t)$ .

Проте практично добитися повної адекватності об'єкта і моделі неможливо, неможливою може бути і повна рівність початкових умов. Тому на практиці можна розраховувати лише на виконання умови

$$\lim_{t \rightarrow \infty} \hat{\mathbf{x}}(t) = \mathbf{x}(t). \quad (7.3)$$

Подібну властивість мають так звані асимптотичні спостерігаючі пристрої. Асимптотичний спостерігаючий пристрій використовує для відновлення вектора стану зворотний зв'язок за помилкою спостереження. Загальний вигляд системи управління із спостерігачем показаний на рис. 7.1.

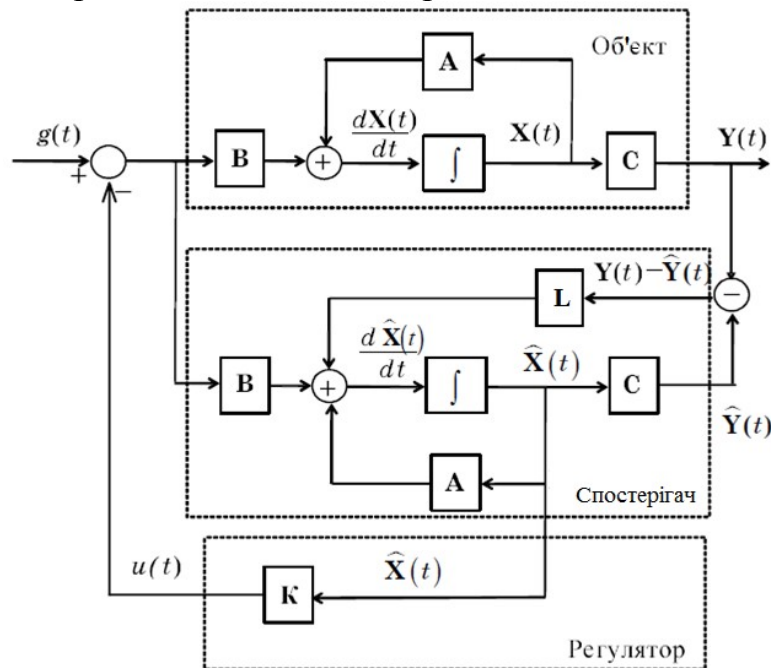


Рис. 7.1 – Структура системи управління зі спостерігачем

Згідно рис. 7.1, робота спостерігаючого пристрою описується рівнянням

$$\begin{aligned} \frac{d\hat{\mathbf{x}}}{dt} &= \mathbf{A}\hat{\mathbf{x}}(t) + \mathbf{B}u(t) + \mathbf{L}(\mathbf{y}(t) - \mathbf{C}\hat{\mathbf{x}}(t)); \\ \frac{d\hat{\mathbf{x}}}{dt} &= \mathbf{A}\hat{\mathbf{x}}(t) + \mathbf{B}u(t) + \mathbf{L}\mathbf{y}(t) - \mathbf{L}\mathbf{C}\hat{\mathbf{x}}(t); \\ \frac{d\hat{\mathbf{x}}}{dt} &= (\mathbf{A} - \mathbf{L}\mathbf{C})\hat{\mathbf{x}}(t) + \mathbf{B}u(t) + \mathbf{L}\mathbf{y}(t), \end{aligned} \quad (7.4)$$

де  $\mathbf{L}$  – матриця параметрів спостерігаючого пристрою.

Власний рух системи, яка описується рівнянням (7.4), визначається матрицею коефіцієнтів  $\mathbf{H} = \mathbf{A} - \mathbf{L}\mathbf{C}$ , для її стійкості всі власні значення матриці  $\mathbf{H}$  повинні мати негативні дійсні частини.

Параметри спостерігача і параметри регулятора можуть розраховуватися незалежно.

Зрозуміло, що процеси в спостерігачі повинні протікати швидше, ніж перехідний процес в системі. Емпірично встановлено, що спостерігач повинен мати швидкодію, що в 2–4 рази перевищує швидкодію системи.

Матриця спостережуванності (7.1) може бути побудована за допомогою функції **obsv**, яка також може викликатися в одному з варіантів :

```
--> N = obsv_mat(A, C)
--> N = obsv_mat(sys)
--> N = obsv_mat(sys.A, sys.C)
```

У Scilab є функція **ppol(A, B, P)**, яка по заданим матрицям **A**, **B** і вектору полюсів **P** підбирає елементи матриці **K** так, щоб власні значення матриці **A–BK** були полюсами системи. Проте для спостерігача має виконуватися така ж умова, але для матриці **A–CK**, тому аргументи треба дещо змінити: транспонувати матрицю **A** і замінити **B** на **C<sup>T</sup>**:

```
--> L = ppol(AT, CT, Pe),
```

де **Pe** – вектор бажаних полюсів спостерігача.

Спостерігач можна розглядати як динамічну систему, яка представляється моделлю у просторі станів:

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{A}_e \mathbf{x} + \mathbf{B}_e \mathbf{u}; \\ \mathbf{y} = \mathbf{C}_e \mathbf{x} - \mathbf{D}_e \mathbf{u}. \end{cases} \quad (7.5)$$

У системі на рис. 7.1 здійснюється управління за змінними стану, тож будемо вважати виходом системи саме вектор змінних стану. У другому рівнянні моделі (7.5) матриця **C<sub>e</sub>** буде одиничною, а матриця **D<sub>e</sub>** нульовою.

Таким чином, матриці моделі (7.5) слід формувати у наступному вигляді:

$$\begin{aligned} \mathbf{A}_e &= \mathbf{A} - \mathbf{L}\mathbf{C}; \\ \mathbf{B}_e &= [\mathbf{B}, \mathbf{L}]; \\ \mathbf{C}_e &= \mathbf{eye}(\mathbf{A}); \\ \mathbf{D}_e &= \mathbf{zeros}([\mathbf{B}, \mathbf{L}]). \end{aligned}$$

При цьому динамічну систему спостерігача створюємо командою:

```
--> est = syslin('c', A-L*C, [B,L], eye(A), zeros([B,L]));
```

В Scilab існують спеціальні функції для формування спостерігача. Функція **observer** формує спостерігаючий пристрій у вигляді ss-об'єкта для оцінювання вектора змінних стану моделі об'єкта управління **sys** і для заданої матриці коефіцієнтів зворотних зв'язків спостерігача **L**:

```
--> est = observer(sys, L)
```

Функція **obscont** формує регулятор для заданої в просторі станів моделі об'єкта управління **sys**, матриці коефіцієнтів зворотних зв'язків по змінних стану **K** і матриці коефіцієнтів зворотних зв'язків спостерігача **L**:

```
--> rsys = obscont(sys, K, L)
```

Треба мати на увазі, що при відомій точній моделі об'єкта і якщо початковий стан спостерігача співпадає з початковим станом об'єкта управління, то спостерігач не впливає на динаміку системи. Якщо ж початкові умови об'єкта і спостерігача різні, то перехідний процес буде на початковому етапі відрізнятися від процесу в системі зі зворотним зв'язком за істинним векторним станом (поки не закінчується перехідний процес у спостерігачі).

## 7.2 Приклади і задачі для самостійного розв'язання

**Задача 7.1.** Визначити спостережуванність об'єкта управління, що описується матрицями задачі 6.1

### Розв'язування

Виконуємо наступні команди.

```
-->A=[0 1 0; 0 0 1; 0 -8 -9]; B=[0; 0; 1];  
-->C=[8 0 0]; D=0;  
--> N = obsv_mat(A, C);  
--> rank(N)  
ans =  
3.
```

Оскільки ранг матриці спостережуванності є рівним порядку, то система спостережувана.

**Задача 7.2.** Сформувати модель спостерігача для об'єкта задачі 5.1.

### Розв'язування

Виконуємо наступні команди:

```
-->A=[0 1 0; 0 0 1; 0 -8 -9]; B=[0; 0; 1];  
-->C=[8 0 0]; D=0;
```

Задаємо початковий стан  $x_0$ :

```
-->x0=[1;1;1];
```

Створюємо динамічну систему об'єкта:

```
-->Sys=syslin('c',A,B,C,0,x0);
```

Створюємо динамічну систему спостерігача:

```
-->Obs=observer(Sys);
```

Початковий стан спостерігача:

```
-->xch0=[1.5;1.5;1.5];
```

```
-->Obs('X0')=xch0;
```

Створюємо систему «вхід  $u(t)$  – вихід  $x(t)$ »:

```
-->Sys2=syslin('c',A,B,eye(3,3),[0;0;0],x0);
```

Формуємо вхідний сигнал  $100\sin(t)$ :

```
-->deff('ut=u(t)', 'ut=100*sin(t)');
```

Розраховуємо перехідну характеристику

```
-->T=0:0.1:20;
```

```
-->x=csim(u,T, Sys2);
```

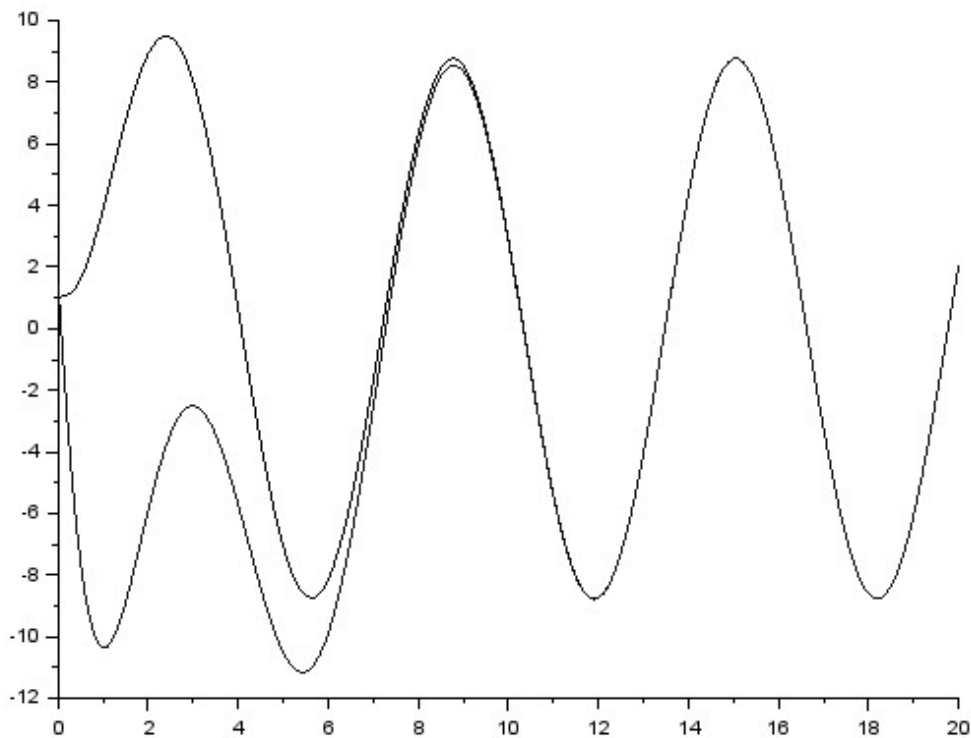
Створимо систему «спостерігач-об'єкт»:

```
-->SObs=Obs*[1; Sys];
```

Перехідний процес системи «спостерігач-об'єкт»:

```
-->xhat=csim(u,T, SObs);
```

```
-->plot(T', [x(2,:); xhat(2,:) ]')
```



Оскільки початкові умови об'єкта і спостерігача різні, то перехідний процес спостерігача на початковому етапі відрізняється від процесу в системі зі зворотним зв'язком. Починаючи з 10-ої секунди, спостерігач точно відстежує стан об'єкта.

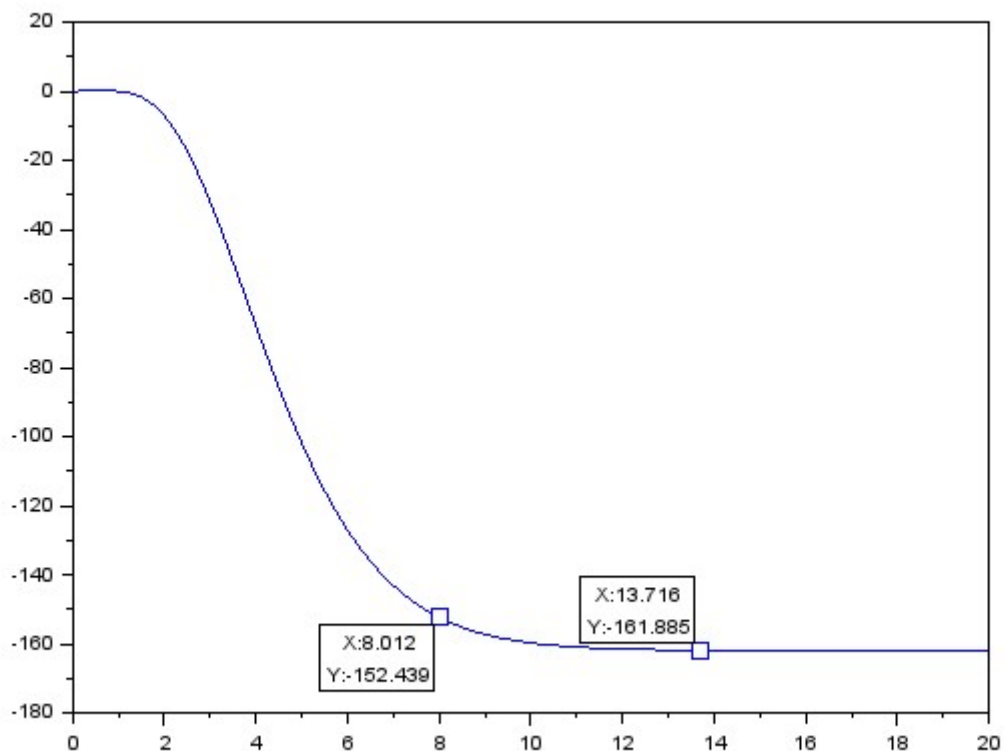
**Задача 7.3.** Синтезувати систему регулювання для об'єкта задачі 6.1 з використанням контролера на основі спостерігача.

### Розв'язування

Створюємо сценарій:

```
A=[0 1 0; 0 0 1; 0 -8 -9]; B=[0; 0; 1];
C=[8 0 0]; D=0;
//Створюємо динамічну систему об'єкта:
P=syslin('c',A,B,C,0,x0);
//Матриця посилення контролера
Kc=-prol(A,B,[-1,-1,-1]);
//Матриця зворотних зв'язків спостерігача
Kf=-prol(A',C',[-2,-2,-2]);Kf=Kf';
//ПФ системи з негативним зворотним зв'язком
cl=P/(-obscont(P,Kc,Kf));
// Побудова перехідного процесу
T=0:0.1:20;
clf
plot(T,csim("step",T,cl))
```

В результаті виконання сценарію отримуємо графік перехідного процесу:



Параметри перехідного процесу: перерегулювання відсутнє, час встановлення у 5-відсотковий коридор 8 с.

**Задача 7.4.** Для об'єкта управління, що описується матрицями **A**, **B**, **C** (див. табл. 7.1) визначити спостережуваність об'єкта і, у випадку позитивного висновку, розрахувати значення коефіцієнтів зворотних зв'язків спостерігача, сформулювати модель спостерігача, синтезувати систему регулювання і побудувати графіки перехідних процесів.

Таблиця 7.1

| Варі-<br>ант | <b>A</b>                      | <b>B</b>   | <b>C</b>  |
|--------------|-------------------------------|------------|-----------|
| 1            | [-1 -1 0; 1 -1 1; -10 -9 -12] | [1; 0; 1]  | [1 0 0]   |
| 2            | [-1 1 0; 1 -1 1; -5 -6 -7]    | [0; 1; 2]  | [12 0 0]  |
| 3            | [-1 1 0; 1 -2 1; -5 -12 -7]   | [1; 1; 2]  | [-11 0 0] |
| 4            | [-1 -1 0; 0 -1 1; -2 -11 -10] | [0; 0; 2]  | [5 0 0]   |
| 5            | [-1 -2 0; 0 -1 1; -1 -1 -12]  | [2; 0; 2]  | [-4 0 0]  |
| 6            | [-3 -1 0; -1 -1 1; -9 -10 -9] | [2; 2; 2]  | [-34 0 0] |
| 7            | [-2 -2 0; 2 -2 -1; -6 -3 -8]  | [1; 2; 3]  | [32 0 0]  |
| 8            | [-2 -4 0; 1 -2 3; -4 -7 -3]   | [1; -2; 3] | [11 0 0]  |

## Продовження таблиці 7.1

| Варі-<br>ант | A                              | B            | C         |
|--------------|--------------------------------|--------------|-----------|
| 9            | [-2 1 0; 3 -2 1; -5 -21 -4]    | [2; -2; 2]   | [-10 0 0] |
| 10           | [-3 -3 0; 5 -3 1; -15 -2 -8]   | [1; -2; 5]   | [-1 0 0]  |
| 11           | [-3 -2 0; 2 -1 2; -4 -25 -5]   | [1; -2; 1]   | [-21 0 0] |
| 12           | [-2 2 0; 2 -1 3; -7 -20 -5]    | [1; -3; 1]   | [14 0 0]  |
| 13           | [-1 -2 0; 2 -1 1; -5 -2 -9]    | [1; -1; 3]   | [-55 0 0] |
| 14           | [-3 -2 0; 1 -1 2; -11 -1 -12]  | [-1; -2; -3] | [15 0 0]  |
| 15           | [-3 -2 0; 1 -1 2; -22 -11 -12] | [1; -2; 3]   | [-5 0 0]  |
| 16           | [-1 -1 0; 2 -1 2; -2 -1 -12]   | [4; -2; 4]   | [-10 0 0] |
| 17           | [-1 -3 0; 6 -1 6; -10 -10 -10] | [5; -5; 5]   | [20 0 0]  |
| 18           | [-5 -5 0; 5 -5 5; -5 -5 -5]    | [5; -5; 5]   | [5 0 0]   |
| 19           | [-2 -1 0; 3 -2 1; -9 -12 -18]  | [3; -2; 2]   | [10 0 0]  |
| 20           | [-1 -1 0; 1 -1 1; -1 -1 -1]    | [1; -1; 1]   | [1 0 0]   |

## 8 ЛІНІЙНО-КВАДРАТИЧНИЙ ГАУСІВСЬКИЙ (ЛКГ) РЕГУЛЯТОР

### 8.1 Короткі теоретичні відомості

Лінійно-квадратичний гаусівський регулятор (ЛКГ, англ. LQG) - набір методів і математичного апарату теорії управління для синтезу систем управління з негативним зворотним зв'язком для лінійних систем, які піддаються адитивному гаусівському шуму. Так називають випадкові збурення, розподілені за нормальним законом.

Об'єкт управління представляють наступною матричною моделлю:

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{Ax} + \mathbf{Bu} + \mathbf{w}; \\ \mathbf{y} = \mathbf{Cx} + \mathbf{Du} + \mathbf{v}, \end{cases} \quad (8.1)$$

де  $\mathbf{w}$  – збурення, що діють на об'єкт управління;

$\mathbf{v}$  – шуми вимірювання.

Подавлення випадкових збурень можна забезпечити використанням фільтра Калмана. LQG-регулятор є поєднанням фільтра Калмана і лінійно-квадратичного регулятора (LQR). Структура такого регулятора показана на рис. 8.1.

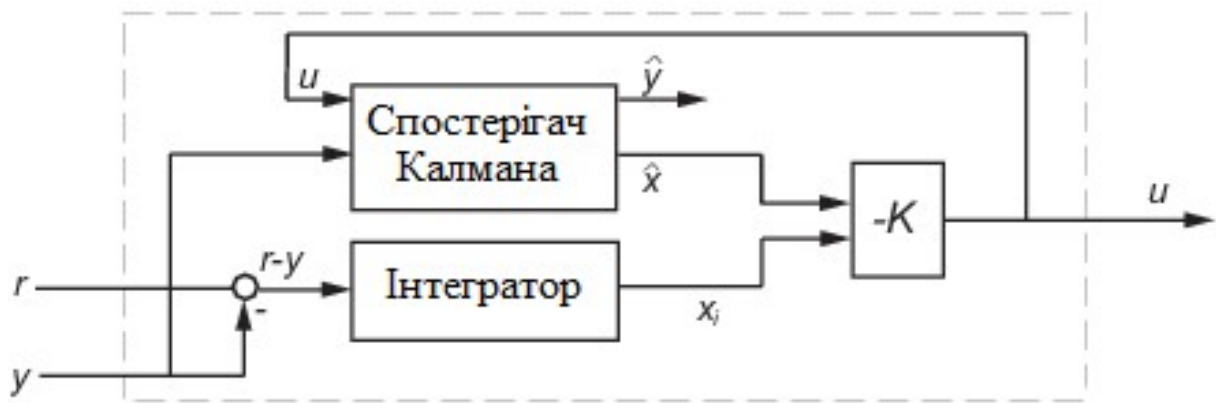


Рисунок 8.1

Розв'язання задачі LQG-оптимального синтезу розпадається на рішення таких задач:

- а) задачі LQR-оптимізаційного синтезу;
- б) задачі синтезу фільтра Калмана;
- в) об'єднання результатів цих розрахунків у схему згідно рис. 8.1.

Завдання LQR-оптимізації полягає у синтезі регулятора  $\mathbf{u} = \mathbf{K}\mathbf{x}$  шляхом мінімізації критерію у вигляді квадратичного функціоналу:

$$J(\mathbf{u}) = \int_0^{\infty} (\mathbf{x}^T(t)\mathbf{Q}\mathbf{x}(t) + \mathbf{u}^T(t)\mathbf{R}\mathbf{u}(t) + 2\mathbf{x}^T(t)\mathbf{N}\mathbf{u}(t))dt,$$

де  $\mathbf{Q}$ ,  $\mathbf{R}$  і  $\mathbf{N}$  – матриці вагових коефіцієнтів.

Закон управління, знайдений за LQR-алгоритмом, має вигляд:

$$\mathbf{u} = -\mathbf{R}^{-1}(\mathbf{B}^T\mathbf{S} + \mathbf{N}^T)\mathbf{x},$$

де  $\mathbf{S}$  – матриця, яка може бути обчислена рішенням рівняння Ріккати:

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

Задача синтезу фільтра Калмана розглядається у розділі 7.

Для автоматизації цих обчислень в пакеті Scilab передбачені функції:

- а) **gcare** – розв'язування рівняння Рікати для неперервної системи;
- б) **lqr** – знаходить LQR-оптимальне підсилення для неперервних систем;
- в) **lqe** – лінійно-квадратичний естиматор (фільтр Калмана);
- г) **obscont** – контролер на основі спостерігача (фільтра Калмана).

Проте задачу LQG-синтезу можна вирішити простіше – використанням двох спеціальних функцій: **lqg2stan** і **lqg**.

Функція **lqg2stan** створює так званий доповнений об'єкт (augmented plant), який враховує збурення і шуми. Синтаксис функції:

$$[P\_aug, r] = lqg2stan(P, Qxu, Qwv),$$

де  $P$  – номінальна модель об'єкта у просторі станів у формі рівнянь (2.1);

$Qxu$ ,  $Qwv$  – вагові матриці;

$P\_aug$  – ss-модель доповненого об'єкта;

$r$  – вектор-рядок з розмірностей векторів виходу і входу  $[ny \ nu]$ .

Вихідними даними для розрахунку доповненого об'єкта виступають адитивна дисперсія шуму збурення  $E(d^2)$ , дисперсія шуму вимірювання  $E(n^2)$ , а також матриці вагових коефіцієнтів  $\mathbf{Q}$  і  $\mathbf{R}$ , які враховують вплив змінних стану  $x_i$  і входів  $u_i$  відповідно. З використанням цих даних розраховують:

а) шум збурення, що додається до входу  $\mathbf{u}$ :

$$\mathbf{Q}_{ww} = \mathbf{B} \cdot E(d^2) \cdot \mathbf{B}^T;$$

б) шум вимірювання, що додається до виходу  $\mathbf{y}$ :

$$\mathbf{R}_{vv} = E(n^2) \cdot \mathbf{I},$$

де  $\mathbf{I}$  – діагональна одинична матриця розмірності  $n_u \times n_u$ ;

в) вагову матрицю  $\mathbf{Q}_{wv}$  як блочну діагональну матрицю, створену з  $\mathbf{Q}_{ww}$  і  $\mathbf{R}_{vv}$  за допомогою функції **blockdiag**;

г) вагову матрицю  $\mathbf{Q}_{xu}$  як блочну діагональну матрицю, створену з матриць  $\mathbf{Q}$  і  $\mathbf{R}$  за допомогою функції **blockdiag**.

Функція **lqg** розраховує матрицю коефіцієнтів LQG-регулятора. Її синтаксис:

$$\mathbf{K} = \text{lqg}(\mathbf{P\_aug}, r),$$

де  $\mathbf{P\_aug}$ ,  $r$  – дані, що є виходами функції **lqg2stan**.

Можна також виконати LQG-розрахунок, використовуючи номінальну модель об'єкта  $\mathbf{P}$ :

$$\mathbf{K} = \text{lqg}(\mathbf{P}, \mathbf{Q}_{xu}, \mathbf{Q}_{wv}).$$

ЛКГ-контролери можуть використовуватися як для неперервних, так і дискретних для систем.

Незважаючи на те, що синтез ЛКГ-контролерів передбачає систематичну процедуру розрахунку для оптимізації якості системи, головним його недоліком є те, що в розгляд не приймається робастність системи. Тому ЛКГ-синтез проводиться тільки для систем, що мають надійну і точну лінійну динамічну модель. Для підвищення робастності системи управління застосовують більш складні алгоритми, такі як мінімаксий ЛКГ-синтез, або комбінований ЛКГ-синтез.

## 8.2 Приклади і задачі для самостійного розв'язання

**Задача 8.1.** Виконати LQG-оптимізацію системи регулювання об'єкта з передатною функцією, заданою вектором  $\text{TF}=[60, 1.5, 1, 110]$ , перший елемент якого є чисельником передатної функції, а три наступних – коефіцієнтами знаменника. На систему діє збурення типу білого шуму  $d$  та наявний деякий шум вимірювання  $n$  з такими шумовими властивостями:  $E(d^2)=1$ ,  $E(n^2)=0.01$ .

### Розв'язування

У редакторі SciNotes створюємо сценарій розв'язування задачі, який зберігаємо з іменем Theme8.sci.

Першою командою сценарію очищуємо консоль від результатів попередніх обчислень:

```
clear; clc
```

Вводимо вихідні дані задачі:

```
Ed2=1; En2=0.01; TF=[60 1.5 1 110]
```

Формуємо вигляд передатної функції:

```
s=poly(0, 's');  
str=string(TF(1))+ '/' + string(TF(2)) + '*s^2+' ...  
+ string(TF(3)) + '*s+' + string(TF(4)) + ')' '
```

Відкриваємо діалогове вікно для інтерактивного багаторядкового введення. Це дає можливість корекції передатної функції:

```
rep=x_dialog('ПФ та, що треба?', str)  
if rep==[] then return, end
```

Створюємо передатну функцію об'єкта:

```
Plant=evstr(rep);  
Ptf=syslin('c', Plant);
```

Переводимо її у форму простору станів:

```
Pss=tf2ss(Ptf);  
[A,B,C,D]=abcd(Pss); // матриці ПФ  
[ny,nu,nx]=size(Pss); // розмірності матриць C, B, A
```

Введення адитивної дисперсії шуму збурення (з можливістю корекції)

```
str=string(Ed2);  
rep=x_dialog('Шум збурення:', str)  
if rep==[] then return, end  
Ed2 =evstr(rep);
```

Введення адитивної дисперсії шуму вимірювання (з можливістю корекції)

```
str=string(En2);  
rep=x_dialog('Шум збурення:', str)  
if rep==[] then return, end  
En2 =evstr(rep);
```

Формування вагової матриці  $Q_{wv}$ :

```
R_vv=En2*eye(1,1); // дисперсія шуму вимірювання  
Q_wv=B*Ed2*B'; // шум збурення, що додається до входу u  
Qwv=blockdiag(Q_wv,R_vv);
```

Для формування вагової матриці  $Q_{xu}$  треба задати вагу впливу на результат кожної зі змінних стану  $x_i$  і кожного з входів  $u_i$ . Згідно умови задачі, об'єкт має один вхід і один вихід. Вважаємо, що вплив усіх цих факторів однаковий, і вага кожного є рівню одиниці :

```
Q_xx=eye(nx,nx); // ваги змінних стану  
R_uu = 1; // ваги входів u  
Qxu=blockdiag(Q_xx,R_uu);
```

Розрахунок LQG-регулятора:

```
J=lqg(Pss, Qxu, Qwv);
```

ПФ замкнутого контуру:

```
Sys=Pss/ . (-J) ;
```

Побудова графіків перехідних процесів:

```
t = 0:0.01:3;
```

```
clf;
```

```
subplot(2,1,1)
```

```
y=csim("step",t, Sys); // реакція на ступінчастий сигнал
```

```
plot(t,y);xgrid
```

```
subplot(2,1,2)
```

```
plot(t,csim("impulse",t, Sys));xgrid
```

Розрахунок перерегулювання:

```
 $\sigma = (\max(y) - y(\infty)) / y(\infty) * 100;$ 
```

```
err = abs(y-y($));
```

```
h_end = find(err >= 0.05*y($));
```

Розрахунок часу установлення перехідного процесу:

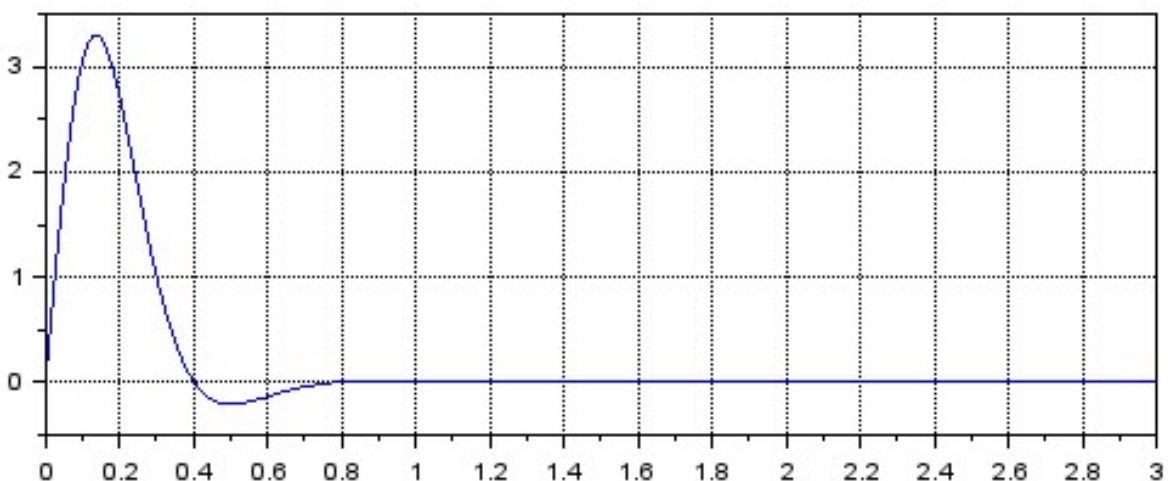
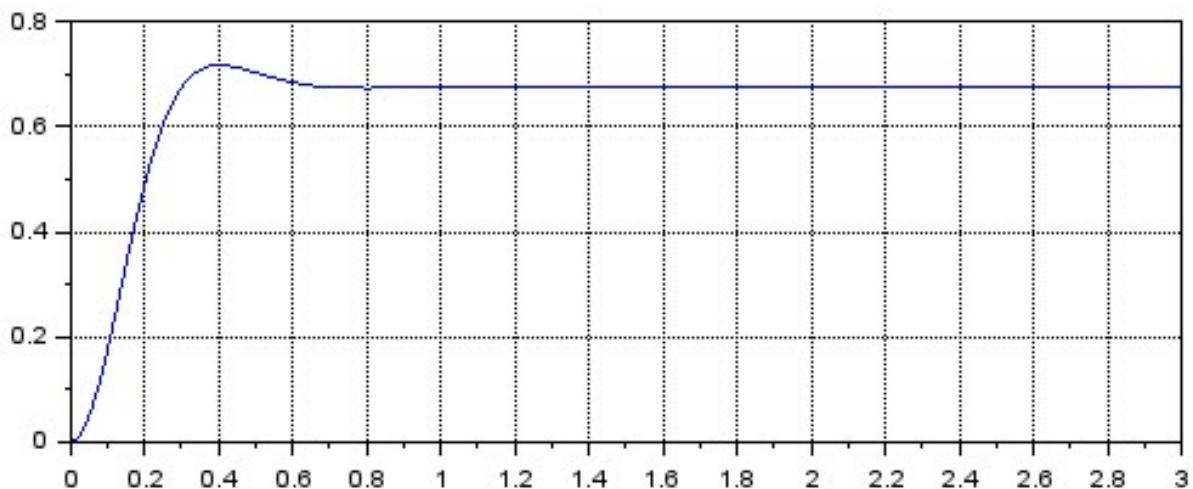
```
set_time = t(max(h_end)+1);
```

Виведення результатів:

```
disp(sprintf('Перерегулювання  $\sigma =$  %g %%,...
```

```
час установлення %g c',  $\sigma$ , set_time))
```

Клавішею F5 зберігаємо сценарій у файлі і запускаємо на виконання. У графічному вікні спостерігаємо графіки перехідної і вагової характеристик:



У вікні консолі читаємо результат розрахунків параметрів:  
 "Перерегулювання  $\sigma = 21.6783 \%$ , час установлення  $0.57 \text{ с}$ "

**Задача 8.2.** За зразком задачі 8.1 виконати LQG-оптимізацію системи регулювання для варіантів з табл. 8.1

Таблиця 8.1

| Варіант | $E(d^2)$ | $E(n^2)$ | TF            | Варіант | $E(d^2)$ | $E(n^2)$ | TF            |
|---------|----------|----------|---------------|---------|----------|----------|---------------|
| 1       | 1        | 0.01     | 100 1 1 100   | 11      | 40       | 0.11     | 60 1.5 1 90   |
| 2       | 50       | 0.03     | 100 1 1 100   | 12      | 70       | 0.02     | [80 1.5 1 80] |
| 3       | 75       | 0.08     | 100 1 1 100   | 13      | 30       | 0.3      | 100 1.5 1 100 |
| 4       | 60       | 0.12     | [100 2 1 90]  | 14      | 80       | 0.1      | 50 1 1 90     |
| 5       | 100      | 0.01     | [75 1 1 100]  | 15      | 10       | 0.03     | 60 1.5 1 110  |
| 6       | 110      | 0.2      | 100 1.5 1 100 | 16      | 20       | 0.02     | [70 1 1 100]  |
| 7       | 40       | 0.11     | [60 1 1 100]  | 17      | 60       | 0.1      | [60 1 1 100]  |
| 8       | 50       | 0.35     | 100 1 1 75    | 18      | 30       | 0.3      | [75 1 1 100]  |
| 9       | 1        | 0.1      | 75 2 1 75     | 19      | 40       | 0,2      | [80 2 1 80]   |
| 10      | 1        | 0.1      | 60 1.5 1 90   | 20      | 50       | 0.3      | [100 2 1 90]  |

## ЛІТЕРАТУРА

1. Фетісов, В.С. Математична система Scilab : навч.-метод. посібн. 2-ге вид., перероб. і доп. Ніжин: НДУ ім. М. Гоголя, 2022. – 82 с.
2. Дубовой, В.М. Імітаційне моделювання в системі Scilab/Xcos : Навчальний посібник [Електронний ресурс] / В. М. Дубовой, М. С. Юхимчук. – Режим доступу : [https://web.posibnyky.vntu.edu.ua/fksa/10dubovyj\\_imitacijne\\_modelyuvannya\\_v\\_systemi\\_Scilab-Xcos/txt/d&yu\\_vstup.html](https://web.posibnyky.vntu.edu.ua/fksa/10dubovyj_imitacijne_modelyuvannya_v_systemi_Scilab-Xcos/txt/d&yu_vstup.html). – Заголовок з екрану.
3. Лабораторная работа № 1. Модальное управление и наблюдающие устройства. Автоматизация синтеза регуляторов и наблюдателей состояния в среде пакета SCILAB [Електронний ресурс]. – Режим доступу : <http://gigabaza.ru/doc/191541.html>. – Заголовок з екрану.
4. Методичні вказівки до лабораторних робіт з курсу “Теорія керування” для студентів базового напрямку 6. 040303 “Системний аналіз” / Укл.: Є.С. Струк – Львів : Видавництво Львівської політехніки, 2011. – 37с.

## ДОДАТОК А

Таблиця А.1 – Вихідні дані для задач

| №<br>варіанта | А                                                                                                  | В                                                        | С                     |
|---------------|----------------------------------------------------------------------------------------------------|----------------------------------------------------------|-----------------------|
| 1             | $\begin{bmatrix} -39 & -71 & -23 \\ 23 & 40 & 13 \\ -6,6 & -10,7 & -3,3 \end{bmatrix}$             | $\begin{bmatrix} 0.228 \\ -0.127 \\ 0.038 \end{bmatrix}$ | [2800 84000 112000]   |
| 2             | $\begin{bmatrix} -44 & -78 & -25 \\ 26 & 43,7 & 14 \\ -7,5 & -11,8 & -3,7 \end{bmatrix}$           | $\begin{bmatrix} 0.228 \\ -0.127 \\ 0.038 \end{bmatrix}$ | [2800 84000 112000]   |
| 3             | $\begin{bmatrix} -54 & -98 & -32 \\ 31 & 55 & 17,8 \\ -9 & -15 & -4,8 \end{bmatrix}$               | $\begin{bmatrix} 0.228 \\ -0.127 \\ 0.038 \end{bmatrix}$ | [35000 105000 140000] |
| 4             | $\begin{bmatrix} -60,8 & -107 & -34,7 \\ 35 & 60 & 19 \\ -10 & -16,7 & -5,3 \end{bmatrix}$         | $\begin{bmatrix} 0.228 \\ -0.127 \\ 0.038 \end{bmatrix}$ | [35000 105000 140000] |
| 5             | $\begin{bmatrix} -72,4 & -130,6 & -42,5 \\ 41,5 & 73 & 23,6 \\ -12 & -20,6 & -6,6 \end{bmatrix}$   | $\begin{bmatrix} 0.228 \\ -0.127 \\ 0.038 \end{bmatrix}$ | [42000 126000 168000] |
| 6             | $\begin{bmatrix} -80,4 & -142,5 & -46 \\ 45,9 & 79,6 & 25,6 \\ -13,6 & -22,6 & -7,2 \end{bmatrix}$ | $\begin{bmatrix} 0.228 \\ -0.127 \\ 0.038 \end{bmatrix}$ | [42000 126000 168000] |
| 7             | $\begin{bmatrix} -100,5 & -174,4 & -56 \\ 57 & 97 & 31 \\ -17 & -28 & -8,9 \end{bmatrix}$          | $\begin{bmatrix} 0.228 \\ -0.127 \\ 0.038 \end{bmatrix}$ | [42000 126000 168000] |
| 8             | $\begin{bmatrix} -13,2 & -16,7 & -4,7 \\ 8,6 & 9,5 & 2,6 \\ -2,4 & -1,6 & -0,3 \end{bmatrix}$      | $\begin{bmatrix} 0.228 \\ -0.127 \\ 0.038 \end{bmatrix}$ | [3500 10500 14000]    |
| 9             | $\begin{bmatrix} -11,4 & -16,7 & -5,13 \\ 7,5 & 9,7 & 2,9 \\ -2 & -1,6 & -0,36 \end{bmatrix}$      | $\begin{bmatrix} 0.228 \\ -0.127 \\ 0.038 \end{bmatrix}$ | [7000 21000 2800]     |
| 10            | $\begin{bmatrix} -16,2 & -26,3 & -8,3 \\ 10,2 & 15 & 4,6 \\ -2,9 & -3,2 & -0,9 \end{bmatrix}$      | $\begin{bmatrix} 0.228 \\ -0.127 \\ 0.038 \end{bmatrix}$ | [14000 42000 56000]   |
| 11            | $\begin{bmatrix} -22,3 & -38,6 & -12,4 \\ 13,6 & 21,9 & 6,9 \\ -3,9 & -5,3 & -1,6 \end{bmatrix}$   | $\begin{bmatrix} 0.228 \\ -0.127 \\ 0.038 \end{bmatrix}$ | [21000 63000 84000]   |
| 12            | $\begin{bmatrix} -34 & -54 & -17 \\ 20 & 30,5 & 9,4 \\ -5,9 & -7,9 & -2,3 \end{bmatrix}$           | $\begin{bmatrix} 0.228 \\ -0.127 \\ 0.038 \end{bmatrix}$ | [21000 63000 84000]   |

Продовження таблиці А.1

| №<br>варіанта | А                                                                                                  | В                                                        | С                     |
|---------------|----------------------------------------------------------------------------------------------------|----------------------------------------------------------|-----------------------|
| 13            | $\begin{bmatrix} -34 & -54 & -17 \\ 20 & 30,5 & 9,4 \\ -5,9 & -7,9 & -2,3 \end{bmatrix}$           | $\begin{bmatrix} 0,228 \\ -0,127 \\ 0,038 \end{bmatrix}$ | [2800 84000 112000]   |
| 14            | $\begin{bmatrix} -39 & -71 & -23 \\ 23 & 40 & 13 \\ -6,6 & -10,7 & -3,3 \end{bmatrix}$             | $\begin{bmatrix} 0,228 \\ -0,127 \\ 0,038 \end{bmatrix}$ | [2800 84000 112000]   |
| 15            | $\begin{bmatrix} -44 & -78 & -25 \\ 26 & 43,7 & 14 \\ -7,5 & -11,8 & -3,7 \end{bmatrix}$           | $\begin{bmatrix} 0,228 \\ -0,127 \\ 0,038 \end{bmatrix}$ | [35000 105000 140000] |
| 16            | $\begin{bmatrix} -54 & -98 & -32 \\ 31 & 55 & 17,8 \\ -9 & -15 & -4,8 \end{bmatrix}$               | $\begin{bmatrix} 0,228 \\ -0,127 \\ 0,038 \end{bmatrix}$ | [35000 105000 140000] |
| 17            | $\begin{bmatrix} -60,8 & -107 & -34,7 \\ 35 & 60 & 19 \\ -10 & -16,7 & -5,3 \end{bmatrix}$         | $\begin{bmatrix} 0,228 \\ -0,127 \\ 0,038 \end{bmatrix}$ | [42000 126000 168000] |
| 18            | $\begin{bmatrix} -72,4 & -130,6 & -42,5 \\ 41,5 & 73 & 23,6 \\ -12 & -20,6 & -6,6 \end{bmatrix}$   | $\begin{bmatrix} 0,228 \\ -0,127 \\ 0,038 \end{bmatrix}$ | [42000 126000 168000] |
| 19            | $\begin{bmatrix} -80,4 & -142,5 & -46 \\ 45,9 & 79,6 & 25,6 \\ -13,6 & -22,6 & -7,2 \end{bmatrix}$ | $\begin{bmatrix} 0,228 \\ -0,127 \\ 0,038 \end{bmatrix}$ | [42000 126000 168000] |
| 20            | $\begin{bmatrix} -100,5 & -174,4 & -56 \\ 57 & 97 & 31 \\ -17 & -28 & -8,9 \end{bmatrix}$          | $\begin{bmatrix} 0,228 \\ -0,127 \\ 0,038 \end{bmatrix}$ | [3500 10500 14000]    |
| 21            | $\begin{bmatrix} -13,2 & -16,7 & -4,7 \\ 8,6 & 9,5 & 2,6 \\ -2,4 & -1,6 & -0,3 \end{bmatrix}$      | $\begin{bmatrix} 0,228 \\ -0,127 \\ 0,038 \end{bmatrix}$ | [7000 21000 2800]     |
| 22            | $\begin{bmatrix} -11,4 & -16,7 & -5,13 \\ 7,5 & 9,7 & 2,9 \\ -2 & -1,6 & -0,36 \end{bmatrix}$      | $\begin{bmatrix} 0,228 \\ -0,127 \\ 0,038 \end{bmatrix}$ | [14000 42000 56000]   |
| 23            | $\begin{bmatrix} -16,2 & -26,3 & -8,3 \\ 10,2 & 15 & 4,6 \\ -2,9 & -3,2 & -0,9 \end{bmatrix}$      | $\begin{bmatrix} 0,228 \\ -0,127 \\ 0,038 \end{bmatrix}$ | [21000 63000 84000]   |
| 24            | $\begin{bmatrix} -22,3 & -38,6 & -12,4 \\ 13,6 & 21,9 & 6,9 \\ -3,9 & -5,3 & -1,6 \end{bmatrix}$   | $\begin{bmatrix} 0,228 \\ -0,127 \\ 0,038 \end{bmatrix}$ | [21000 63000 84000]   |

## ЗМІСТ

|                                                                                    |    |
|------------------------------------------------------------------------------------|----|
| 1 Короткий опис програмного забезпечення.....                                      | 4  |
| 2 Дослідження лінійної стаціонарної динамічної системи в середовищі Scilab .....   | 17 |
| 3 Побудова математичного опису об'єкта управління експериментальними методами..... | 25 |
| 4 Аналіз стійкості лінійної системи .....                                          | 31 |
| 5 Синтез регуляторів засобами Scilab .....                                         | 43 |
| 6 Модальне управління.....                                                         | 53 |
| 7 Спостерігачі стану.....                                                          | 58 |
| 8 Лінійно-квадратичний гаусівський (ЛКГ) регулятор.....                            | 64 |
| Література .....                                                                   | 69 |
| ДОДАТОК А.....                                                                     | 70 |