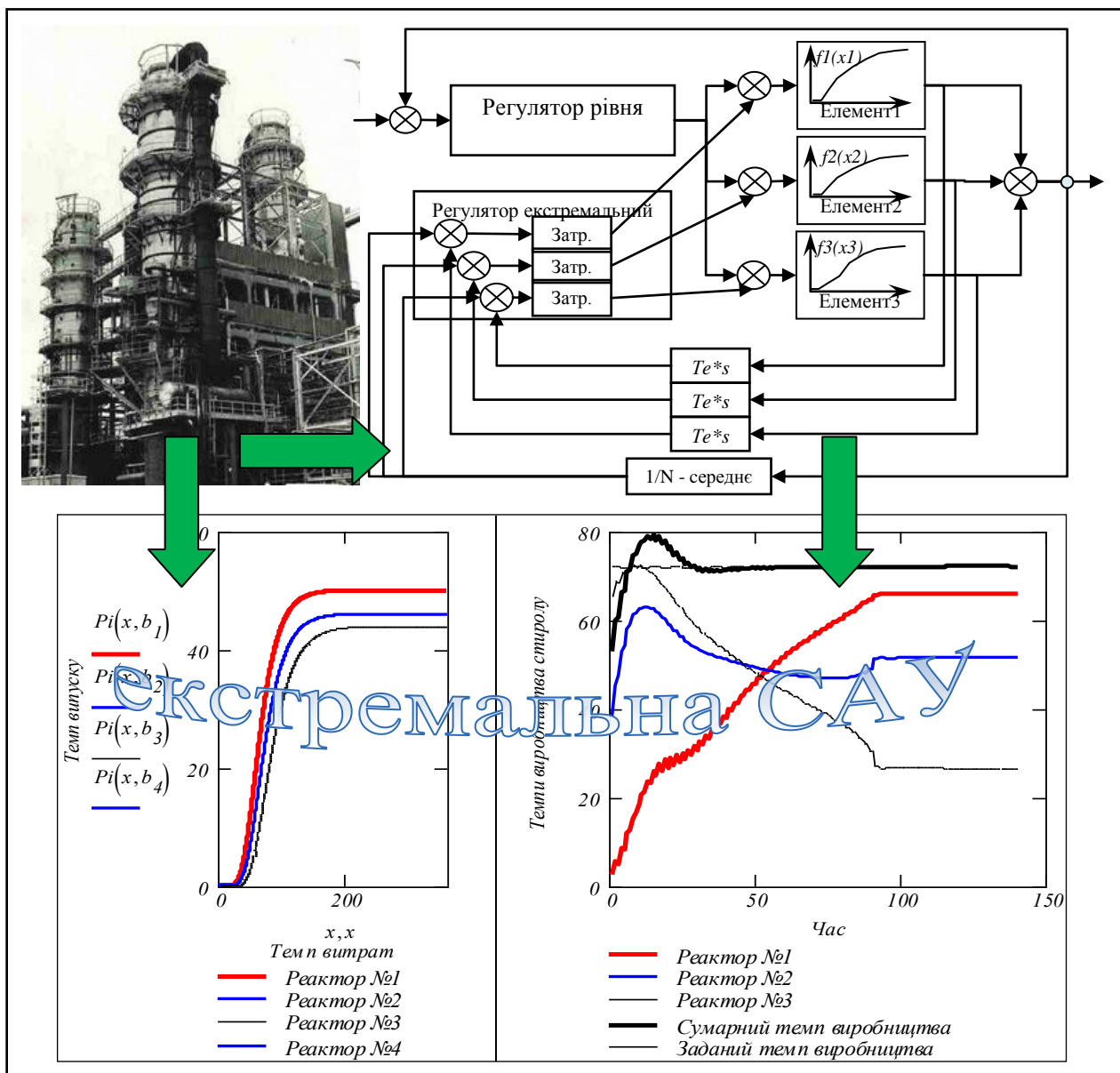


Т. М. Боровська, А. С. Васюра, В. А. Северілов

# МОДЕЛЮВАННЯ ТА ОПТИМІЗАЦІЯ СИСТЕМ АВТОМАТИЧНОГО УПРАВЛІННЯ



Міністерство освіти і науки України  
Вінницький національний технічний університет

**Т. М. Боровська, А. С. Васюра, В. А. Северілов**

**МОДЕЛЮВАННЯ ТА ОПТИМІЗАЦІЯ СИСТЕМ  
АВТОМАТИЧНОГО УПРАВЛІННЯ**

Рекомендовано Міністерством освіти і науки  
України як навчальний посібник для студентів напрямку  
“Системна інженерія” вищих навчальних закладів

Вінниця  
ВНТУ  
2009

УДК 681.51  
Б 83

Рекомендовано до видання Міністерством освіти і науки України.  
Лист № 14/18-Г-1034 від 26.06.07

*Рецензенти:*

**В. П. Квасников**, доктор технічних наук, професор

**В. М. Лисогор**, доктор технічних наук, професор

**А. П. Ладанюк**, доктор технічних наук, професор

**Боровська Т. М.**

Б 83      Моделювання та оптимізація систем автоматичного управління:  
навчальний посібник / Т. М. Боровська, А.С. Васюра, В. А. Северілов. –  
Вінниця: ВНТУ, 2009. – 132 с.

ISBN 978-966-641-319-5

В посібнику розглянуто ряд актуальних задач управління – від постановки задачі до побудови робочої моделі і проведення досліджень на цій моделі. Розглянуто „швидкі” адаптивні системи, що настроюються протягом перехідного процесу в основному контурі, стійкі до відмов вимірювачів і приводів системи, екстремальні системи, а також регулятори на базі нечіткої логіки і штучних нейронних мереж. Об’єкти управління – від роботів, систем хімічних реакторів, літаків з вертикальним зльотом і посадкою до банківської системи. Посібник має дві форми – електронну книгу, що дозволяє вести наукові дослідження, і звичайний „паперовий” посібник. Посібник забезпечує вивчення відповідних дисциплін, в тому числі і магістерських, а також виконання курсових, дипломних проектів і магістерських й інших дисертацій.

УДК 681.51

**ISBN 978-966-641-319-5**

© Т. М. Боровська, А. С. Васюра, В. А. Северілов, 2009

## Зміст

<b>Передмова</b> .....	4
<b>1 Синтез і дослідження САУ з цифровим регулятором</b> .....	5
1.1 Синтез і дослідження САУ з "швидким" імпульсним регулятором.....	5
1.2 Дослідження впливу нелінійностей "обмеження" і "зона нечутливості" .....	9
1.3 Дослідження впливу запізнення .....	14
1.4 Розробка і дослідження адаптивної САУ з настроюванням власної частоти.....	18
Висновки .....	25
<b>2 Синтез і дослідження САУ зі спостерігачем</b> .....	26
2.1 Синтез та дослідження імпульсної САУ зі спостерігачем стану.....	26
2.2 Синтез та дослідження стійкої до відмов вимірювачів імпульсної САУ зі спостерігачами.....	36
Висновки .....	52
<b>3 Синтез і дослідження термінальної САУ</b> .....	53
3.1 Синтез термінальної САУ для типових задач управління кінцевим станом .....	53
3.1.1 Синтез термінальної САУ для задачі розгону.....	54
3.1.2 Синтез термінальної САУ для задачі приведення.....	55
3.1.3 Синтез термінальної САУ для задачі зближення.....	57
3.2 Дослідження термінальної САУ для задачі м'якого зближення.....	60
3.2.1 Розробка модуля термінального управління.....	61
3.2.2 Розробка математичної моделі і програми моделювання САУ... ..	62
3.2.3 Двовимірна задача зближення.....	63
3.3 Розробка і дослідження термінальної САУ стійкої до відмов виконавчих елементів .....	70
Контрольні запитання .....	80
Завдання для самостійної роботи .....	80
<b>4 Приклади побудови адаптивних САУ</b> .....	81
4.1 Система оптимального управління синтезом стиролу .....	81
4.2 Система оптимального управління банком .....	95
Контрольні запитання .....	103
Завдання для самостійної роботи .....	103
<b>5 Дебютні ідеї для побудови адаптивних САУ</b> .....	104
5.1 Управління об'єктом класу „нестійкий маятник” .....	104
5.2 Адаптивний регулятор на базі нечіткої логіки. ....	112
5.3 Адаптивна система на базі штучної нейронної мережі .....	116
5.4 "Швидка" адаптивна САУ .....	121
Висновки.....	129
ПІДСУМКИ.....	129
<b>Література</b> .....	130

## Передмова

В даному посібнику розглянуто спеціальні розділи теорії автоматичного управління - оптимальні та адаптивні САУ. В існуючих посібниках і стандартних програмах ці теми розташовувались десь у кінці, біля висновків і викладались „узагальнено-оглядово”.

Сьогодні практично немає обмежень на складність алгоритмів управління, зокрема фільтрації, ідентифікації, оптимізації. Не проблема і онлайнний (під час функціонування САУ) синтез параметрів регулятора. Вартість апаратних засобів для реалізації системи автоматичного управління – сенсорів, виконавчих елементів і мікроконтролерів - зменшилась в десятки разів. Не зникли, а може стали більш важкими проблеми, пов’язані з нелінійністю, нестационарністю та незворотністю змін в адаптивних САУ. Для таких систем відсутні загальні методи синтезу. Найбільша неприємність може бути подана таким практичним правилом: чим більше ми вкладаємо інтелекту в САУ, тим більш різноманітними, непрогнозовними і дорогими будуть її відмови. Саме тому на практиці часто віддають перевагу САУ з постійними параметрами регуляторів, підібраними так, щоб САУ залишалась задовільною в усьому діапазоні зміни параметрів об’єкта та зовнішніх збурень.

Посібник охоплює дуже незначну частину тематики "оптимальні і адаптивні САУ". Не розглядаються класичні задачі і розв’язання задач ідентифікації та оптимального управління на базі узагальненого квадратичного критерію, класичні задачі оптимального управління при наявності шумів. Перша причина – ці задачі досить вичерпно подані у відомій літературі. Друга (причина) – обсяг посібника заплановано в 130 сторінок. Тому відібрано задачі і системи управління, що:

- мало відображені в масових сучасних посібниках;
- задовільно працюють як мінімум на рівні моделювання;
- мають надійний („твердий”) теоретичний фундамент;
- є придатними для розвитку і модифікації;
- нечутливі до розмірності задачі управління.

Суть посібника – приклади **конструювання робочих моделей** для актуальних задач. В посібнику подано декілька прикладів розробки і випробування адаптивних САУ (на імітаційних моделях), але це працездатні і досить цікаві за своїми властивостями системи. На базі цих прикладів можна ставити нові дослідження, будувати свої власні розробки.

**Посібник повністю виконаний в середовищі математичного пакета** і має активну електронну версію. Велика увага приділяється не тільки задачам, але й технологіям розробки робочих математичних моделей (програмних модулів) і технологіям досліджень на базі обчислювальних експериментів. В посібнику є потрібна рутинна – подано розгорнуті приклади синтезу регуляторів вектора стану, систем зі спостерігачами стану, систем термінального управління, розглянуто питання моделювання нелінійностей, нестационарності, випадкових збурень.

В останньому розділі розглянуто на рівні робочих моделей певні "ескізи", "заготовки" для дещо екзотичних систем – регуляторів на базі: нечіткої логіки, нейронних мереж і просто здорового глузду. Прикладом останнього класу є так звана "швидка адаптивна САУ", настроювання в якій виконуються протягом перехідного процесу в основному контурі.

Задачі останнього розділу зроблені саме як предметні завдання для самостійної розробки на різних рівнях – курсового проекту, магістерської або кандидатської дисертації.

В посібнику дана одна незвична для ТАУ задача – оптимальне управління банківською системою. Для наступного посібника залишені:

- задачі оптимального управління розподіленими і децентралізованими системами, такими як системи паралельно працюючих технологічних агрегатів чи підприємств галузі;
- задачі оптимального управління процесами розвитку виробничих систем (стратегії розвитку, цінові стратегії, кредитні стратегії), які помилково відносять до економіки. Однак це задачі такого класу, як і задача оптимального перельоту до Марсу.

# 1 СИНТЕЗ І ДОСЛІДЖЕННЯ САУ З ЦИФРОВИМ РЕГУЛЯТОРОМ

Існують або співіснували такі дисципліни: теорія автоматичного управління (ТАУ, що іноді називається ТАК), проектування систем автоматичного управління, комп'ютерні системи управління, „машинне проектування”, оптимальне проектування, були навіть „складні системи управління”, існують навіть „інтелектуальні системи”. Дуже неважко зрозуміти, що легко розмістити в безмежних просторах теорії управління регулятори на базі нечіткої логіки чи штучних нейронних мереж, ніж висмикувати щось з теорії управління і втискувати в „інтелектуальні системи”. Щодо "оптимального" чи "машинного" проектування, то невідомо, кому сьогодні потрібне "неоптимальне" і "безмашинне" проектування.

В цьому розділі подано три задачі аналізу і синтезу систем управління:

- Синтез та дослідження САУ з "швидким" імпульсним регулятором;
- Дослідження впливу нелінійностей "обмеження" і "зона нечутливості”;
- Дослідження впливу запізнення.

Ці розділи - фундамент для розгляду "реліктової" адаптивної системи:

- Розробка і дослідження адаптивної САУ з настроюванням власної частоти.

Головна увага приділена технологіям конструювання робочих моделей, придатних для проведення обчислювальних експериментів.

## 1.1 Синтез та дослідження САУ з "швидким" імпульсним регулятором

Розглянемо цю задачу на суто утилітарному, навіть рецептурному рівні.

**Задача.** Задано об'єкт управління  $X^{(k+1)} = F(T) \cdot X^{(k)} + G(T) \cdot B \cdot u_k$ ,

де  $T = 0.8$  – крок квантування,  $k$  – номер кроку квантування.

$$F(T) = \begin{pmatrix} 1 & T & 0 \\ 0 & 1 & T \\ -1 \cdot T1 & -3 \cdot T1 & 1 - 3 \cdot T \end{pmatrix}; \quad G(T) = \begin{pmatrix} T & 0 & 0 \\ 0 & T & 0 \\ 0 & 0 & T \end{pmatrix}; \quad X = \begin{pmatrix} X_1 \\ X_2 \\ X_3 \end{pmatrix}; \quad B = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}.$$

Рівняння регулятора:  $u_k = X_{1,k} \cdot K_1 + X_{2,k} \cdot K_2 + X_{3,k} \cdot K_3$

1. **Знайти** параметри ( $K_1$   $K_2$   $K_3$ ) імпульсного закону управління ("швидкого регулятора"), що забезпечують перехідний процес для довільного початкового стану тривалістю **точно в три такти квантування**.

2. **Розробити** програму моделювання динаміки керованої системи.

3. **Побудувати** графіки перехідного процесу.

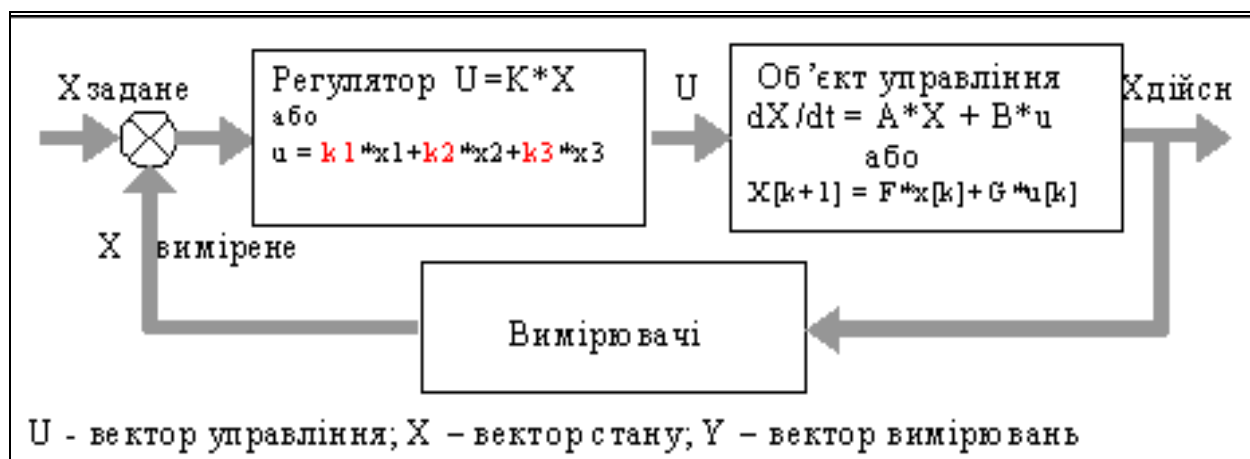


Рис. 1.1. Блок-схема САУ

## Знаходження параметрів $(K_1 \ K_2 \ K_3)$ "швидкого регулятора"

Швидкий регулятор відноситься до класу регуляторів **вектору стану** (vector state controller), а метод його синтезу є окремим випадком **методу стандартних коефіцієнтів**.

Виконаємо синтез (= розрахунок параметрів) регулятора у загальному вигляді.

**Порядок розрахунку параметрів регулятора є таким:**

- 1) отримати характеристичне рівняння для замкненої системи у загальному вигляді (задані параметри об'єкта та невідомі параметри регулятора входять як символічні змінні);
- 2) прирівняти оці загальні вирази при коефіцієнтах характеристичного рівняння їх потрібним значенням. Ці потрібні значення, згідно з теорією УКЧУ, **повинні бути усі =0**;
- 3) розв'язати якимось методом систему рівнянь (вона може бути і нелінійною) - отримуємо шукані значення параметрів регулятора.

**Випишемо вхідні дані задачі (згідно з варіантом завдання):**

Зберемо параметри об'єкта управління у "вектор коефіцієнтів" A:

ORIGIN := 1.  $A_3 := -3$ ;  $A_2 := -3$ ;  $A_1 := -1$ ; крок квантування  $T := 0.8$ .

Дивлячись на те, що подано далі, самостійно і **уважно** набираємо (в своєму документі) **тільки робочі рівняння**

**Записуємо матриці F, G, B – відповідно до завдання.**

$$\text{Об'єкт управління } X^{(k+1)} = F(T) \cdot X^{(k)} + G(T) \cdot B \cdot u_k, \quad (1)$$

де  $T = 0.8$  – крок квантування,  $k$  – номер кроку квантування.

$$F(A, T) := \begin{pmatrix} 1 & T & 0 \\ 0 & 1 & T \\ A_1 \cdot T & A_2 \cdot T & 1 + A_3 \cdot T \end{pmatrix}; \quad G(T) := \begin{pmatrix} T & 0 & 0 \\ 0 & T & 0 \\ 0 & 0 & T \end{pmatrix}; \quad X = \begin{pmatrix} X_1 \\ X_2 \\ X_3 \end{pmatrix}; \quad B := \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}.$$

$$\text{Рівняння регулятора: } u_k = X_{1,k} \cdot K_1 + X_{2,k} \cdot K_2 + X_{3,k} \cdot K_3. \quad (2)$$

Введемо позначення:  $K$  і виводимо для контролю:

$$K := \begin{pmatrix} K_1 \\ K_2 \\ K_3 \end{pmatrix}; \quad I := \text{identity}(3); \quad K^T \rightarrow (K_1 \ K_2 \ K_3); \quad I = \begin{array}{|c|c|c|c|} \hline & 1 & 2 & 3 \\ \hline 1 & 1 & 0 & 0 \\ \hline 2 & 0 & 1 & 0 \\ \hline 3 & 0 & 0 & 1 \\ \hline \end{array}$$

Записуємо характеристичне рівняння (для його отримання підставляємо (2) в (1), виконуємо Z-перетворення і алгебраїчні перетворення). Читається цей вираз так: "детермінант матриці .. повинен дорівнювати нулю".

$$|I \cdot z - (F(A, T) + G(T) \cdot B \cdot K^T)| = 0. \quad (3)$$

Застосовуємо символічний процесор і виводимо результат для контролю

$$I \cdot z - (F(A, T) + G(T) \cdot B \cdot K^T) \rightarrow \begin{pmatrix} z - 1 & -.8 & 0 \\ 0 & z - 1 & -.8 \\ .8 - .8 \cdot K_1 & 2.4 - .8 \cdot K_2 & z + 1.4 - .8 \cdot K_3 \end{pmatrix}.$$

Розкриваємо і спрощуємо визначник (3) за допомогою символічного процесора

$$|I \cdot z - (F(A, T) + G(T) \cdot B \cdot K^T)| \begin{array}{l} \text{collect, z} \\ \text{float, 2} \end{array} \rightarrow z^3 + (-.8 \cdot K_3 - .6) \cdot z^2 + (1.6 \cdot K_3 - .64 \cdot K_2 + .12) \cdot z + .64 \cdot K_2 - .51 \cdot K_1 - 8 \cdot 10^{-3} - .8 \cdot K_3$$

Копіюємо праву частину, прирівнюємо до нуля - отримуємо Z-характеристичне рівняння лінійної імпульсної системи:

$$z^3 + (-0.8 \cdot K_3 - 0.6) \cdot z^2 + (1.6 \cdot K_3 - 0.64 \cdot K_2 + 0.12) \cdot z + (0.64 \cdot K_2 - 0.51 \cdot K_1 - 8 \cdot 10^{-3} - 0.8 \cdot K_3) = 0.$$

Прирівнюємо його коефіцієнти до нуля – отримуємо систему з трьох рівнянь для трьох невідомих – параметрів "швидкого" регулятора  $K_1, K_2, K_3$ . Розв'язуємо цю систему за допомогою вбудованого методу (підпрограми) "Given - Find"

1. Задаємо початкові значення змінних:  $K_1 := 1$ ;  $K_2 := -2$ ;  $K_3 := -3$ .

2. Після ключового слова **Given** записуємо систему рівнянь:

$$-0.8 \cdot K_3 - 0.6 = 0; \quad 1.6 \cdot K_3 - 0.64 K_2 + 0.12 = 0; \quad 0.64 K_2 - 0.51 K_1 - 8.0 \cdot 10^{-3} - 0.8 \cdot K_3 = 0.$$

3. Пишемо, що ми власне шукаємо (вектор коефіцієнтів  $K$ ) і виводимо результат

$$ksi := \text{Find}(K); \quad \text{Знайшли параметри регулятора: } ksi^T = \begin{array}{|c|c|c|c|} \hline & 1 & 2 & 3 \\ \hline 1 & -0.957 & -1.688 & -0.75 \\ \hline \end{array}$$

### Альтернативний метод визначення параметрів регулятора

В загальному випадку вирази для коефіцієнтів є занадто довгими і складними. Пропонуємо **альтернативний метод** визначення коефіцієнтів – без отримання явних виразів для коефіцієнтів. Запишемо вираз для характеристичного рівняння (4) у більш компактному вигляді:

$$|(I \cdot z - M_p)| = 0; \quad \text{де } M_p(K, T) := F(A, T) + G(T) \cdot B \cdot K^T. \quad (4)$$

Вибираємо орієнтований на використання сучасних пакетів метод [8]: "неявні вирази для коефіцієнтів характеристичного рівняння", при якому *не треба розкривати детермінант*. Визначимо такі функції користувача:

$$I := \text{identity}(3); \quad \text{determinant}(K, T, z) := |I \cdot z - M_p(K, T)|; \\ \text{polinom}(V2, V1, V0, z) := (z^3 + V2 \cdot z^2 + V1 \cdot z + V0). \quad (5)$$

Тепер задаємо **потрібні** значення коефіцієнтів  $V2, V1, V0$  характеристичного рівняння. Відповідно до завдання потрібно синтезувати "швидкий" регулятор, для якого перехідний процес закінчується за  $N$  кроків (в даному випадку  $N = 3$ ). Для забезпечення такого процесу усі корені характеристичного рівняння повинні бути нульовими:  $Z_i = 0; i = 1..N$ . Тому всі коефіцієнти характеристичного рівняння, крім першого, повинні бути нульовими

$$V_i = 0; \quad i = 0..N-1; \quad \text{ORIGIN} := 1.$$

Задаємо **потрібні значення** коефіцієнтів  $V$ :  $V0 := 0; V1 := 0; V2 := 0$ ,

**початкові значення** шуканих параметрів  $K_1 := 2; K_2 := 2; K_3 := -9$ ,

**довільні різні значення**  $Z$ -коренів:  $z1 := 1; z2 := 0; z3 := -5$ .

Записуємо (в означеннях (5)) систему рівнянь для коефіцієнтів і розв'язуємо її вбудованим методом "Given - Find".

$$\left. \begin{array}{l} \text{Given} \quad (= \text{дано}) \quad \text{polinom}(V2, V1, V0, z1) = \text{determinant}(K, T, z1); \\ \text{polinom}(V2, V1, V0, z2) = \text{determinant}(K, T, z2); \\ \text{polinom}(V2, V1, V0, z3) = \text{determinant}(K, T, z3). \end{array} \right\} \quad (6)$$

Пишемо, що ми власне шукаємо:  $\text{KoReg}(T) := \text{Find}(K)$ .

Для тих, хто вшановує математику. З точки зору математики тут написано "функція змінної  $T$  дорівнює функції іншої змінної  $K$ " - це можливо, коли ці функції є константами. З точки зору програмування це: "визначимо функцію користувача через функцію Find блока розв'язання: 'Given - Find' ". Транслятор знайде вирази (6), функції (5), побачить, як там входить  $T$  і сформує програму, що буде знаходити розв'язання системи рівнянь (6) для заданого  $T$ .

Виводимо і порівнюємо результати синтезу, отримані альтернативними методами:

$$T = 0.8; \quad \text{KoReg}(T) = \begin{array}{|c|c|} \hline & 1 \\ \hline 1 & -0.953 \\ \hline 2 & -1.687 \\ \hline 3 & -0.75 \\ \hline \end{array}; \quad ksi = \begin{array}{|c|c|} \hline & 1 \\ \hline 1 & -0.957 \\ \hline 2 & -1.688 \\ \hline 3 & -0.75 \\ \hline \end{array} \quad - \text{якщо не збіглося, десь помилки.}$$

### Розробка програми моделювання динаміки керованої системи

В середовищі пакета дуже просто побудувати програму моделювання - вирази для різницевих рівнянь є робочими в середовищі, де виконуються. Нам потрібно тільки задати кількість кроків моделювання та інші параметри. Перепишемо рівняння (1), підставляємо в нього рівняння регулятора (2), а в рівняння регулятора підставляємо знайдені параметри ( $ks$  - коефіцієнт синтезований). Задаємо кількість кроків моделювання  $N_q := 9; k := 1..N_q$ ;



початковий стан  $\psi\chi := (-1 \ 0 \ 0)$ ;  $X^{(1)} := \psi\chi^T$  / Записуємо рівняння процесу (скоротіть запис):

$$X^{(k+1)} := F(A, T) \cdot X^{(k)} + G(T) \cdot B \cdot (X_{1,k} \cdot ksi_1 + X_{2,k} \cdot ksi_2 + X_{3,k} \cdot ksi_3).$$

Виводимо результат:

	1	2	3	4	5	6	7	8	9	10
X =	1	-1	-1	-1	0	0	0	-0	-0	-0
	2	0	0	1.25	0	0	-0	0	0	0
	3	0	1.57	-1.57	0	-0	0	0	0	-0

**Факультатив. АЛЬТЕРНАТИВА.** Можна ж зробити це і у вигляді звичної програми – як на Паскалі:

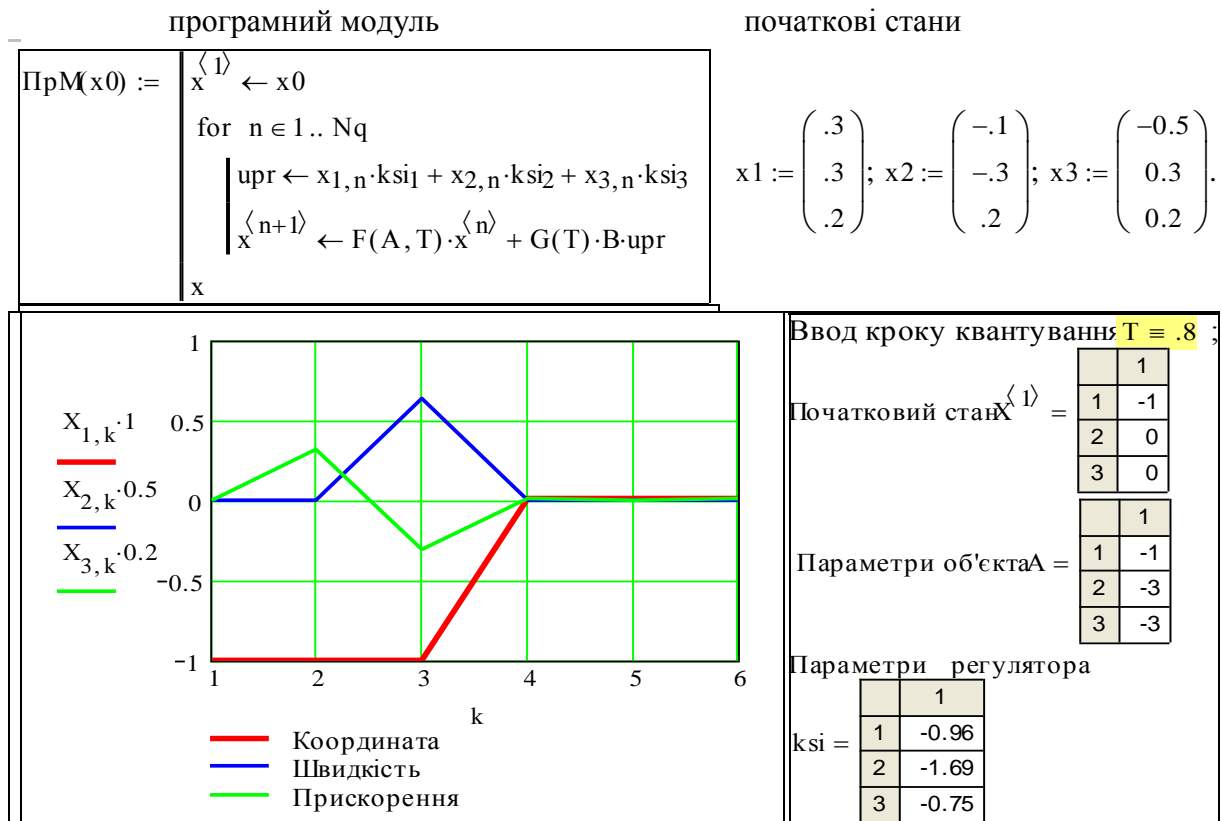


Рис. 1.2. Програма і результати моделювання імпульсної САУ

### Висновки

Визначення параметрів регулятора – математична задача з досить складними алгебраїчними перетвореннями і розв'язанням складної, в загальному випадку, системи алгебраїчних рівнянь. Однак це рутинні процедури з гарантованими результатами. Запропоновано метод синтезу (визначення) параметрів регулятора, де непотрібно знаходити вирази для коефіцієнтів характеристичного рівняння. Якщо параметри об'єкта можуть бути досить точно ідентифіковані, то можна параметри регулятора розраховувати в процесі функціонування системи. Це дає змогу побудувати аналітичну адаптивну систему автоматичного управління (САУ).

### Контрольні запитання

1. Вхідні і вихідні дані задачі синтезу. Які вимоги звичайно ставляться до САУ?
2. Як розшифровуються УКВУ, УКЧУ, що таке „швидкий регулятор“?
3. Визначення вектора стану, вектора управління, і "регулятора вектора стану".

### Завдання для самостійної роботи

1. Побудуйте графіки перехідних процесів для випадків, коли всі параметри регулятора а) збільшені на 5%, б) зменшені на 5%.

## 1.2 Дослідження впливу нелінійностей "обмеження" і "зона нечутливості"

### Постановка задачі

Ціль даного розділу - моделювання та дослідження системи автоматичного управління (САУ) при наявності нелінійностей. Джерелами нелінійностей можуть бути виконавчі елементи (приводи) і вимірювачі. Не завжди нелінійності є результатом недосконалості цих елементів: їх вводять, щоб система просто відключалась біля стану рівноваги (нечутливість), або щоб система була безпечною при великих відхиленнях (обмеження).

В попередніх роботах ми аналітично розраховували параметри регулятора. Однак це можливо тільки, якщо система лінійна. В даному випадку ми визначаємо параметри регулятора без урахування нелінійності, а потім моделюємо систему і уточнюємо параметри регулятора. В даній роботі досліджується вплив так званих **статичних** нелінійностей.

Статичними називають нелінійності, у яких поточний вихід залежить тільки від поточного входу  $z[k] = f(y[k])$ . На рис.1.3 подана схема САУ.

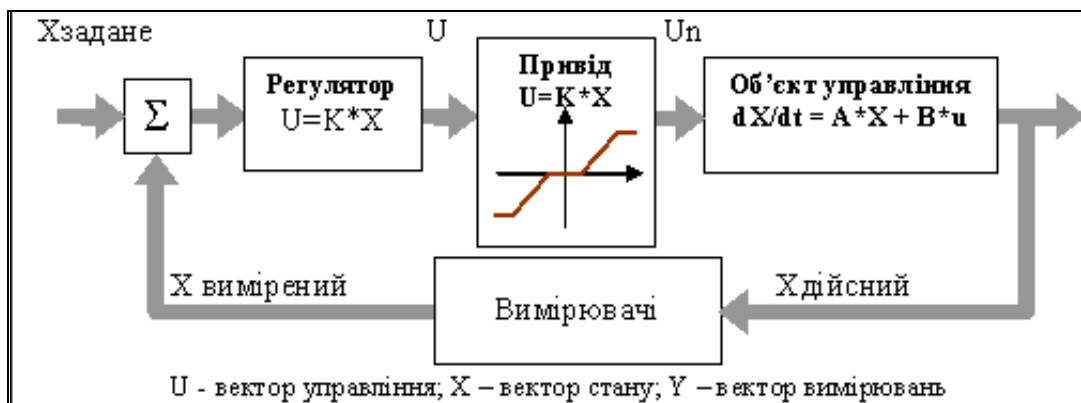


Рис.1.3. Блок-схема системи

### Завдання

1. Розробити робочі математичні моделі нелінійностей 8-20 балів.
2. **Розробити програму** для моделювання САУ з нелінійностями 12 балів.
3. Дослідити вплив: а) нечутливості, б) обмеження, в) нечутливості + обмеження на перехідний процес при варіації величини початкової помилки 12, 13, 15 балів.
4. Знайти граничні значення параметрів нелінійностей, за яких САУ: 12, 13 балів.  
а) ще задовольняє вимоги до якості ПП, б) втрачає стійкість.
5. Дослідити САУ з зоною нечутливості при наявності шумів у вимірювачах 50 балів.

### Розробка математичних моделей нелінійностей

Можна придумати багато версій програми моделювання нелінійностей "зона нечутливості" і "обмеження рівня". Далі подано дві версії такої програми:

а) "перше, що прийшло в голову":

$$NO(x, ob, ne, k) := \begin{cases} 0 & \text{if } 0 \leq |x| < ne \\ k \cdot (|x| - ne) \cdot \text{sign}(x) & \text{if } 0 < |x| - ne \leq 1 \div k \\ ob \cdot \text{sign}(x) & \text{otherwise} \end{cases}$$

б) "якщо задуматись".

$$NO(x, ob, ne, k) := \begin{cases} 0 & \text{if } 0 \leq |x| < ne \\ \min[ob, k \cdot (|x| - ne)] \cdot \text{sign}(x) & \text{otherwise} \end{cases}$$

Тестуємо програму, виводимо (рис.1.4) графіки нелінійностей

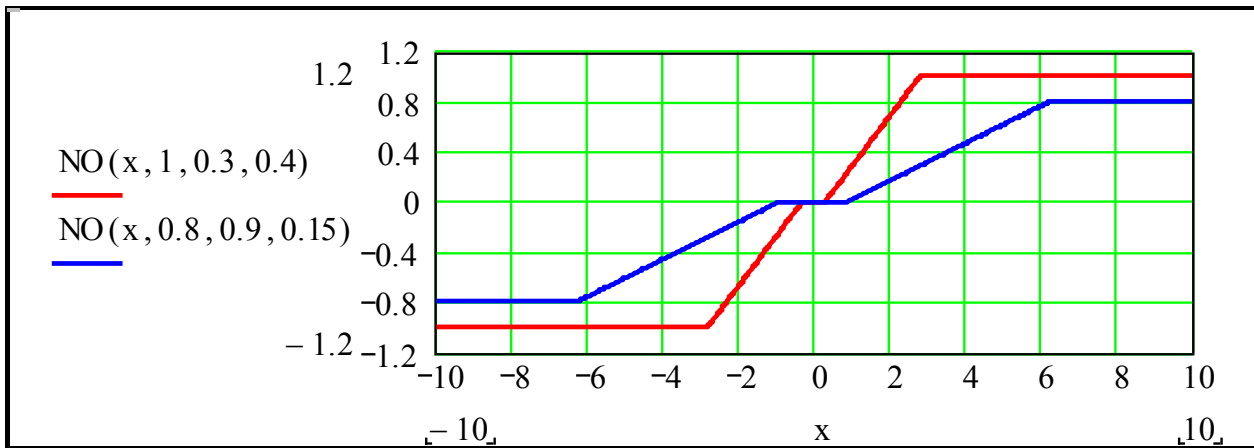


Рис. 1.4. Нелінійності класу "зона нечутливості + обмеження рівня виходу"

### Розробка програми моделювання САУ з нелінійностями

Беремо дані з першої частини посібника "Теорія автоматичного управління. Частина I. Аналіз САУ" [14].

ORIGIN := 1

параметри об'єкта  $a := \begin{pmatrix} -1 \\ 2 \\ -2 \end{pmatrix}$ ; параметри регулятора  $kr := \begin{pmatrix} -1.659 \\ 0.486 \\ -2.87 \end{pmatrix}$ ; крок квантування

(крок моделювання)  $T1 := 0.1$ ; вимоги до перехідного процесу: тривалість  $T_{pz} := 10$  с.; викид (перерегулювання)  $hmz := 0.15$ ; число кроків моделювання  $N := 150$ ;  $n := 1..N$ .

Маємо: матрицю неперервного об'єкта, одиничну матрицю, вектор початкового стану

$A := \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ kr_1 - a_1 & kr_2 - a_2 & kr_3 - a_3 \end{pmatrix}$ ;  $I := \text{identity}(3)$ ;  $I = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$ ;  $x_0 := \begin{pmatrix} -1 \\ 0 \\ 0 \end{pmatrix}$ .

Задаємо параметри об'єкта:  $B := \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$ ;  $A_0 := \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_1 & -a_2 & -a_3 \end{pmatrix}$  і регулятора:  $K := (kr_1 \ kr_2 \ kr_3)$ ,

коефіцієнт підсилення нелінійності беремо  $k = 1$ . Тепер власне складаємо програму. Виводимо для контролю і осмислення те, що закладено в програмі:  $K \cdot x_0 = 1.659$ ;  $NO(K \cdot x_0, 5, 0.2, 1) = 1.459$

```

PN(ob, ne) :=
  IAT ← (I + A_0 · T1)
  BT ← B · T1
  x<sup>1</sup> ← x_0
  for i ∈ 1..N
    u ← K · x<sup>i</sup>
    un ← NO(u, ob, ne, 1 + 0.6 · ne)
    x<sup>i+1</sup> ← IAT · x<sup>i</sup> + BT · un
    Vy x<sup>i+1</sup> ← stack(x<sup>i+1</sup>, u, un)
  Vy x
  
```

**Завдання.** Напишіть коментарі до програми. Наберіть свою версію, або перенаберіть цю. Що є виходом програми?

PN(20, 0) =

	1	2	3	4
1	0	-1	-1	-0.999
2	0	0	0.007	0.019
3	0	0.066	0.126	0.18
4	0	1.659	1.47	1.3
5	0	1.659	1.47	1.3

Обчислюємо три перехідних процеси – номінальний  $qq := PN(20, 0.0)$ , при наявності

зони нечутливості  $qqn := PN(20, neh)$ ; при наявності обмеження  $qqo := PN(obm, 0)$ .

### Аналіз результатів моделювання

Розроблена програма моделювання дозволяє вести інтенсивні дослідження впливу нелінійностей на якість САУ. В дослідженнях можна виділити два рівні:

- рівень аналізу перехідних процесів;
- рівень аналізу функцій впливу.

На рис. 1.5, 1.6 подано приклади аналізу перехідних процесів. На графіках подано пару процесів - еталонний і при наявності нелінійності.

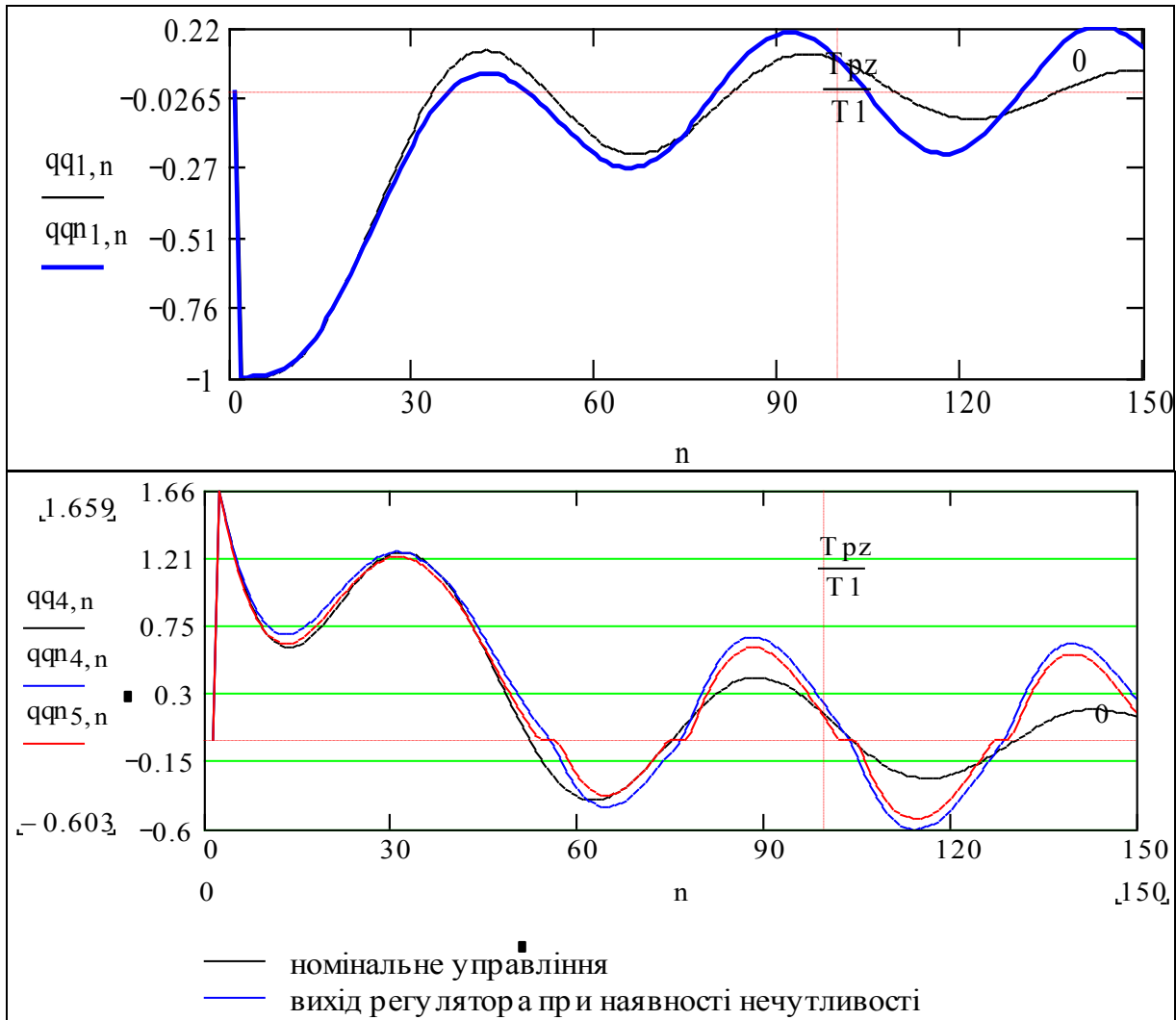


Рис. 1.5. Вплив зони нечутливості на перехідний процес

Подивимось (професійним поглядом) на рис. 1.5 і побачимо: відносно маленька зона нечутливості суттєво погіршує перехідний процес (збільшує коливальність). При дослідженні слід обов'язково пам'ятати, що нелінійності додали системі нову властивість: тепер **перехідні процеси залежать від величини початкового відхилення**. При малих відхиленнях вплив зони нечутливості буде суттєвим, при великих система „не помітить” маленької зони нечутливості. Зону нечутливості іноді вводять навмисно – щоб система не реагувала на малі, припустимі відхилення від заданого режиму (температура, тиск, висота, ..).

Подивимось на рис. 1.6. Бачимо що при вибраному рівні обмеження перехідний процес суттєво погіршується. На нижньому графіку бачимо причину: потрібне управління „зрізається”. Вплив нелінійності і в цьому випадку залежить від величини початкового відхилення, але „навпаки”: при малих відхиленнях потрібні управління не будуть перевищувати обмеження, при великих – обмеження може зробити систему нестійкою.

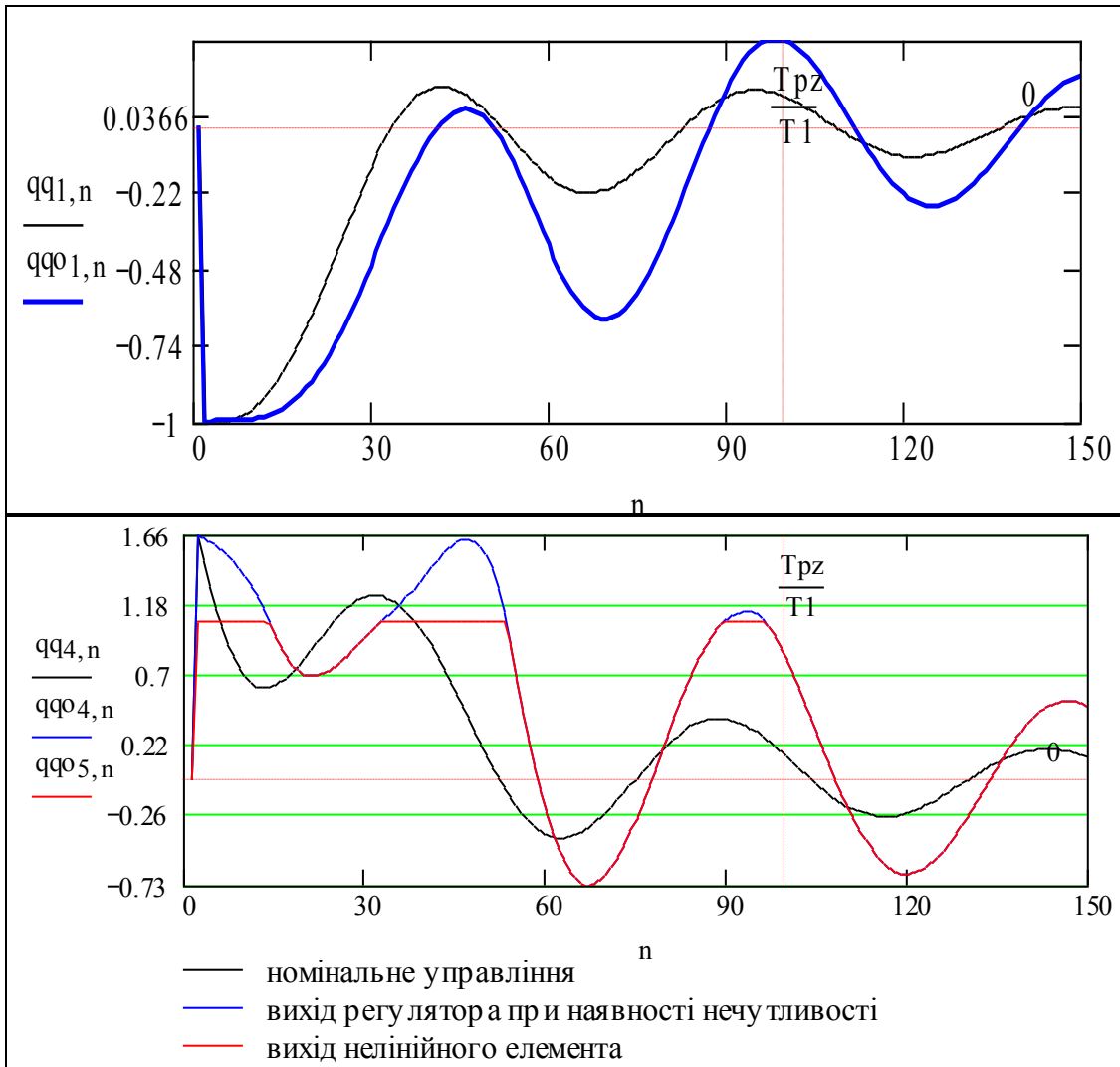


Рис. 1.6. Вплив обмеження на перехідний процес

### Висновки

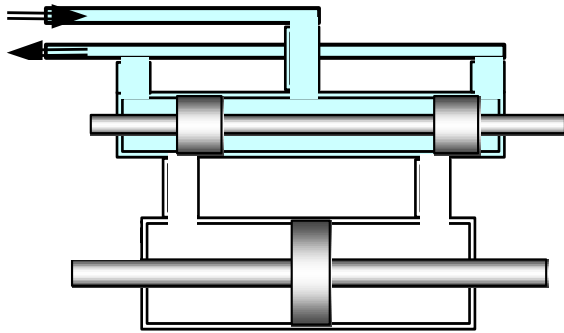
Ми зробили „робочу модель” – програмний модуль для моделювання САУ з нелінійностями, вибрали коливальну САУ з малим запасом стійкості і провели дослідження. Саме для САУ з нестійким об’єктом і малим запасом стійкості вплив нелінійностей є суттєвим і навіть „фатальним”. Наявність зони нечутливості і обмеження у виконавчих елементах та вимірювачах за певних умов приводять САУ до нестійкості.

### Контрольні запитання

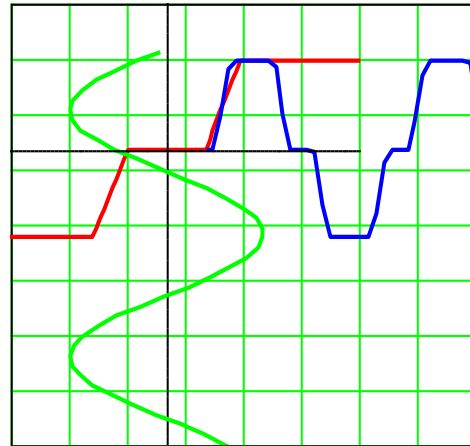
(Оцінки за відповіді в балах рейтингу дисципліни)

1. Запишіть рівняння для нелінійностей "зона нечутливості" та "обмеження".....2 б.
2. Як модифікувати документ для дослідження впливу цих нелінійностей у вимірювачах - координати, швидкості, прискорення?.....10 б.
3. Які методи і критерії для перевірки нелінійних САУ на стійкість ви знаєте?.....4 б.
4. Наведіть приклади елементів САУ із а) зоною нечутливості, б) обмеженням, в) зоною нечутливості та обмеженням.....1 – 9 б.
5. Чи є вплив обмеження і зони нечутливості а) негативним, б) позитивним, а) різним, залежно від обставин?.....3 б.
6. Чи бувають випадки, коли обмеження і зону нечутливості вводять навмисно?.....2 б.
7. Нижче подано схему пристрою. Що це за пристрій? Що є входом і що є виходом? Чи властиві цьому пристрою обмеження і (або) зона нечутливості?.....5 б.

Що це за пристрій?



Що це таке?



8. За рахунок чого у поданому вище пристрої виникають нелінійності а) обмеження, б) зона нечутливості? Чи можна усунути зону нечутливості? Якщо можна, то як? Чи буде це корисним?.....6 б.
9. Нарисуйте пристрій у станах "зона нечутливості", "обмеження" ..... 4 б.
10. В яку сторону буде переміщуватись вихідний шток пристрою при зміщенні вхідного вправо?.....1 б.
11. Що подано на графіку (вище)?.....2 б.
12. Напишіть усі потрібні рівняння для побудови такого графіка?.....1-11б.
13. Що треба написати на осях (функції, змінні) графіка, щоб отримати те, що зображено вище?.....2 б.
14. Нарисуйте графіки функцій, рівняння яких подано нижче?.....3 б.  
 $\omega := 2.5$  ;  $t := -5, -4.9..5$  ;  $x(t) := 2.5 \cdot \sin(\omega \cdot t)$  ;  $og := .9$  ;  $zo := 1$  ;  
 $u(t) := \text{ОбмНеч}(og, zo, t)$  ;  $y(t) := \text{ОбмНеч}(og, zo, x(t))$  ;  
 Максимум 50 балів за всі контрольні запитання.

### Завдання для самостійної роботи

1. Побудуйте графіки для фазових траєкторій – перехідних процесів на фазовій площині.
2. Зробіть програмний модуль для побудови функції впливу **зони нечутливості** на якість перехідного процесу (тривалість, викид). Побудуйте функції впливу для таких номінальних перехідних процесів: аперіодичний, коливальний, на грані стійкості.
3. Зробіть програмний модуль для побудови функції впливу **обмеження** на якість перехідного процесу (тривалість, викид). Побудуйте функції впливу для таких номінальних перехідних процесів: аперіодичний, коливальний, на грані стійкості.
4. Знайдіть граничні значення нелінійностей та проаналізуйте результати.

**Приклад аналізу:** для даного об'єкта управління, для кожного рівня початкової помилки існує критичний рівень обмеження (в даному випадку  $обм = 1.05$ ), при якому система стає нестійкою. Для деяких об'єктів ця границя стійкості дуже різка – ... Можна провести дослідження для розробки регуляторів, стійких до впливу обмежень.

### 1.3 Дослідження впливу запізнення

Ціль даної роботи – отримання навичок використання документованих і недокументованих можливостей сучасних математичних пакетів для моделювання систем автоматичного управління (САУ) при наявності запізнення.

Причинами запізнення управління можуть:

- ∞ запізнення спрацювання виконавчих елементів;
- транспортне запізнення, пов'язане з кінцевими швидкостями передачі сигналів, або матеріальних компонентів процесу – рідини, газу.... ;
- запізнення у самому об'єкті управління;
- ∞ запізнення у самому цифровому регуляторі (воно може бути малим, але завжди присутнє у ЦОМ).

Урахувати запізнення на ціле число кроків управління  $d$  (або моделювання) дуже просто: на вхід об'єкта управління потрібно подавати не  $u[k \cdot T]$ , а  $u[(k-d) \cdot T]$ . Виникає лише незначна технічна проблема, пов'язана з середовищем і методом моделювання, – необхідність визначати перші  $d$  значень вектора стану до початку циклу обчислень.

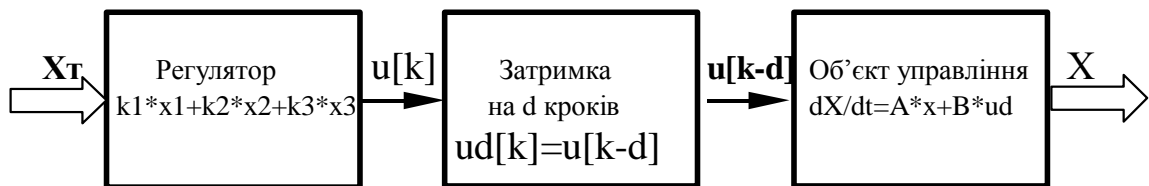


Рис.1.7. Схема САУ з запізненням

#### Завдання

1. Розробити математичну модель запізнення та реалізувати програмними та інструментальними засобами пакета.....10–25 б.
2. Побудувати "стенд" для досліджень впливу запізнення.....15 б.
3. Дослідити вплив запізнення на перехідні процеси.....15 б.
4. Знайти граничне значення запізнення, за яких САУ:
  - а) ще є задовільною, б) втрачає стійкість.....12+13 б.

#### Зразок виконання

Розробка математичної моделі САУ з запізненням

Беремо дані з попереднього розділу:

ORIGIN := 1

$$\text{параметри об'єкта } a := \begin{pmatrix} -1 \\ 2 \\ -2 \end{pmatrix}; \text{ параметри регулятора } kr := \begin{pmatrix} -1.659 \\ 0.486 \\ -2.87 \end{pmatrix};$$

крок квантування (крок моделювання)  $T := 0.2$ ; вимоги до перехідного процесу: тривалість  $T_{pz} := 10$  с.; обмеження викиду  $hmz := 0.15$ ; число кроків моделювання  $N := 90$ ;  $n := 1..N$ .

**Зуваження.** Дані з одного документа в інший можна передати і більш ефективними способами: а) через файлові операції:

записати в файл: `WRITEPRN("t21.prn") := (k1 k2 k3 a2 a1 a0 T Tpz hmz)■`;

прочитати з файлу `Дані := READPRN("t21.prn")■`;

б) через посилання (тоді в цьому документі будуть доступними дані того документа);

в) через Інтернет та інші мережі. Головна проблема, – щоб потрібні документи (файли) були на вказаних місцях. Ми відмовляємось від цих можливостей, щоб забезпечити повну автономність документа.

Копіюємо з розділу 1.1 еквівалентну дискретну математичну модель САУ другого порядку без запізнення, що буде:

- 1) еталоном для оцінювання коректності математичної моделі із запізненням;

2) еталоном для оцінювання впливу запізнення.

Маємо: матрицю неперервного об'єкта, одиничну матрицю, вектор початкового стану

$$A := \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ kr_1 - a_1 & kr_2 - a_2 & kr_3 - a_3 \end{pmatrix}; I := \text{identity}(3); I = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}; x_0^{(1)} := \begin{pmatrix} -1 \\ 0 \\ 0 \end{pmatrix}.$$

Записуємо наближення 2-го порядку перехідної матриці стану  $M_p := I + A \cdot T + \frac{A^2 \cdot T^2}{2!}$ ,

і, нарешті, еквівалентну дискретну модель керованого об'єкта:  $x_0^{(n+1)} := M_p x_0^{(n)}$ .

### Розробка математичної моделі САУ з урахуванням запізнення

Вибираємо з можливих альтернатив побудови математичної моделі оптимальну за критерієм витрат інтелекту. На перших  $d$  кроках моделювання САУ управління через запізнення дорівнює нулю. Припускаємо, що помилки вектора стану на цих кроках нульові.

Для врахування запізнення використовуємо еквівалентну дискретну модель першого порядку (подаємо дві форми запису: а) стару книжну, б) нову – стандарту пакета):

$$x[n+1] = (I + A_0 \cdot T) \cdot x[n] + B \cdot T \cdot u[n-d] = (I + A_0 \cdot T) \cdot x[n] + B \cdot T \cdot K \cdot x[n-d];$$

$$x_{n+1} = (I + A_0 \cdot T) \cdot x_n + B \cdot T \cdot u_{n-d} = (I + A_0 \cdot T) \cdot x_n + B \cdot T \cdot K \cdot x_{n-d}.$$

Бачимо, що тепер не можна в правій частині винести  $x_n$  – власний рух залежить від  $x_n$ , а рух, обумовлений управлінням, – від  $x_{n-d}$ .

Задаємо параметри: регулятора, "нульовий" і початковий стан, параметри моделі об'єкта

$$K := (kr_1 \ kr_2 \ kr_3); x_0 := \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}; x_{nat} := \begin{pmatrix} -1 \\ 0 \\ 0 \end{pmatrix}; A_0 := \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_1 & -a_2 & -a_3 \end{pmatrix}; B := \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}.$$

Для моделювання САУ з запізненням використовуємо еквівалентну дискретну модель першого порядку, а для забезпечення точності як у моделі другого порядку, зменшуємо крок обчислень у стільки = 4 рази, тобто, беремо крок обчислень  $T_1 := T \div \text{стільки}$ ;

$M := N \cdot \text{стільки}$ ;  $q := 2..M$ ;  $x^{(2)} := x_0$ . Виводимо для контролю:  $N = 90$ ;  $M = 360$ ;  $T = 0.2$ ;  $T_1 = 0.05$ . Для контролю моделі задаємо запізнення  $d := 0$ .  $x^{(d+1)} := x_{nat}$ ;  $q := (1 + d)..360$ . **Записуємо робоче рівняння:**

$$x^{(q+1)} := (I + A_0 \cdot T_1) \cdot x^{(q)} + B \cdot K \cdot T_1 \cdot x^{(q-d)}.$$

Будуємо графіки перехідних процесів для базової моделі і отриманої нами моделі. Бачимо, що модель коректна і при вибраному кроці обчислень має задовільну точність.

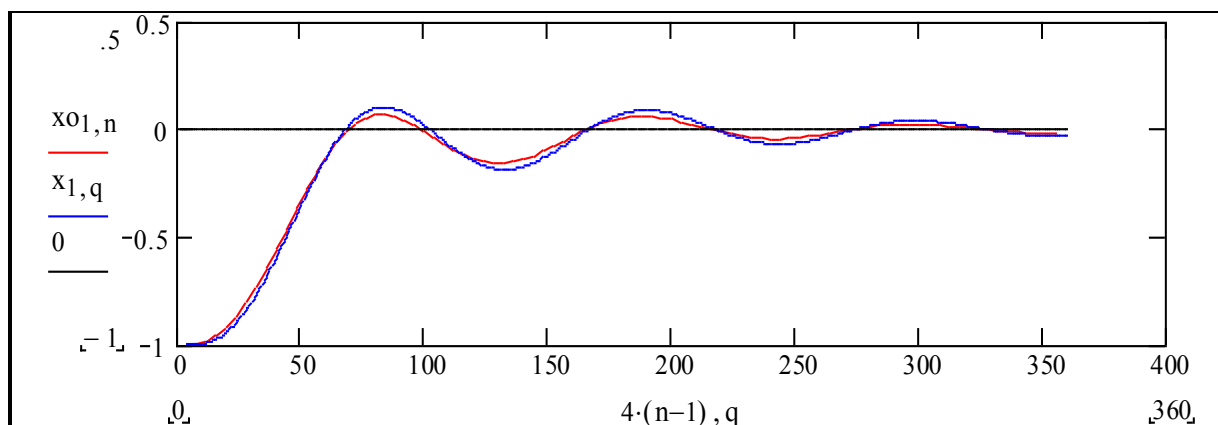


Рис.1.8. Порівняння двох робочих моделей – базової і для урахування запізнення



### Дослідження впливу запізнення. Визначення граничного запізнення

Копіюємо попередній блок формул, незначно доповнюємо його. Вводимо запізнення:

$d := 4$ , отримуємо відповідні процеси. Задаємо: ранжовані змінні  $j := -1, -9..1$ ;  $qq := 1..d$ ;

тривалість\_пш :=  $\frac{Tpz}{T1}$ ;  $x^{(qq)} := x0$ ;  $x^{(d+1)} := xnat$ ;  $q := (1 + d)..360$

$$x^{(q+1)} := (I + A_0 \cdot T1) \cdot x^{(q)} + B \cdot K \cdot T1 \cdot x^{(q-d)}$$

Аналізуємо, робимо висновки.

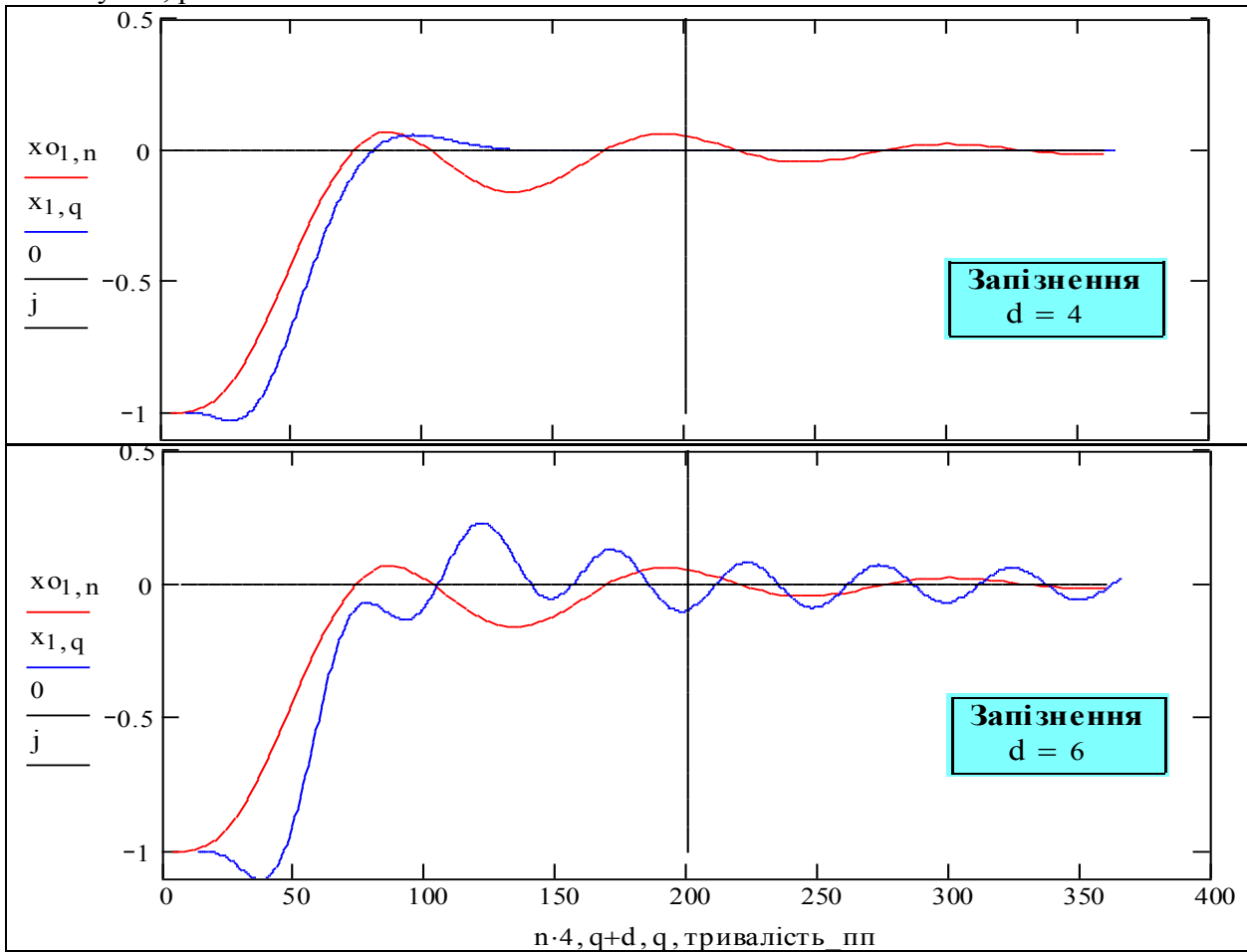


Рис.1.9. Перехідні процеси в САУ при варіації запізнення

Шукаємо граничне запізнення. Для даної системи воно виявляється рівним  $d = 4$ . При запізненні у  $d + 1 = 5$  кроків квантування система стає нестійкою. Це дорівнює  $T1 \cdot d = 0.2$  секунди. На рисунку 1.9 подано графік граничного перехідного процесу.

### Розробка програми моделювання системи з запізненням

Зробимо альтернативну робочу модель – програмний модуль. В середовищі математичного пакета програмні модулі є підпрограмами відносно документа, де вони визначені. "Визначимо функцію користувача через програму" – наша функція  $PP(z)$  бере величину запізнення, а повертає перехідний процес – послідовність векторів стану. Можна розширити список вхідних параметрів – отримуємо більш універсальний модуль.

Багато переваг можемо отримати, якщо модель динаміки системи оформимо як програму. Крім того ми на один рівень стаємо ближче до "істинного програмування" – на мові C++, на якій написано математичний пакет.

Дивимось на програму, бачимо, що там закладені ті ж самі рівняння, що ми писали "врозсип". Програму ми оформили як функцію (визначену користувачем) від запізнення. Продемонструємо одну з переваг – побудуємо серію процесів у вигляді тривимірного графіка.

$PP(z) :=$ $IAT \leftarrow (I + A_0 \cdot T1)$ $BKT \leftarrow B \cdot K \cdot T1$ for $j \in 0..z$ $x^{(j+1)} \leftarrow xnat$ $x^{(z+1)} \leftarrow xnat$ for $i \in (1+z)..200$ $x^{(i+1)} \leftarrow IAT \cdot x^{(i)} + BKT \cdot x^{(i-z+1)}$ $Vyход \leftarrow (x^T)^{(1)}$	$PP(1) =$ <table border="1" style="display: inline-table;"> <tr><td></td><td>1</td></tr> <tr><td>1</td><td>-1</td></tr> <tr><td>2</td><td>-1</td></tr> <tr><td>3</td><td>-1</td></tr> <tr><td>4</td><td>-1</td></tr> <tr><td>5</td><td>-1</td></tr> <tr><td>6</td><td>-1</td></tr> <tr><td>7</td><td>-1</td></tr> <tr><td>8</td><td>-1</td></tr> <tr><td>9</td><td>-1</td></tr> </table> $(I + A_0 \cdot T1) =$ <table border="1" style="display: inline-table;"> <tr><td>1</td><td>0.05</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0.05</td></tr> <tr><td>0.05</td><td>-0.1</td><td>1.1</td></tr> </table> $xnat =$ <table border="1" style="display: inline-table;"> <tr><td>-1</td></tr> <tr><td>0</td></tr> <tr><td>0</td></tr> </table> $B \cdot K \cdot T1 =$ <table border="1" style="display: inline-table;"> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>-0.08</td><td>0.02</td><td>-0.14</td></tr> </table>		1	1	-1	2	-1	3	-1	4	-1	5	-1	6	-1	7	-1	8	-1	9	-1	1	0.05	0	0	1	0.05	0.05	-0.1	1.1	-1	0	0	0	0	0	0	0	0	-0.08	0.02	-0.14
	1																																									
1	-1																																									
2	-1																																									
3	-1																																									
4	-1																																									
5	-1																																									
6	-1																																									
7	-1																																									
8	-1																																									
9	-1																																									
1	0.05	0																																								
0	1	0.05																																								
0.05	-0.1	1.1																																								
-1																																										
0																																										
0																																										
0	0	0																																								
0	0	0																																								
-0.08	0.02	-0.14																																								

Для побудови серії графіків задаємо ранжовану змінну:  $zap := 1..8$  і записуємо таке рівняння  $Mpp^{(9-zap)} := PP(zap)$ . Так ми отримуємо масив, де кожен стовпець – перехідний процес при відповідному запізненні.

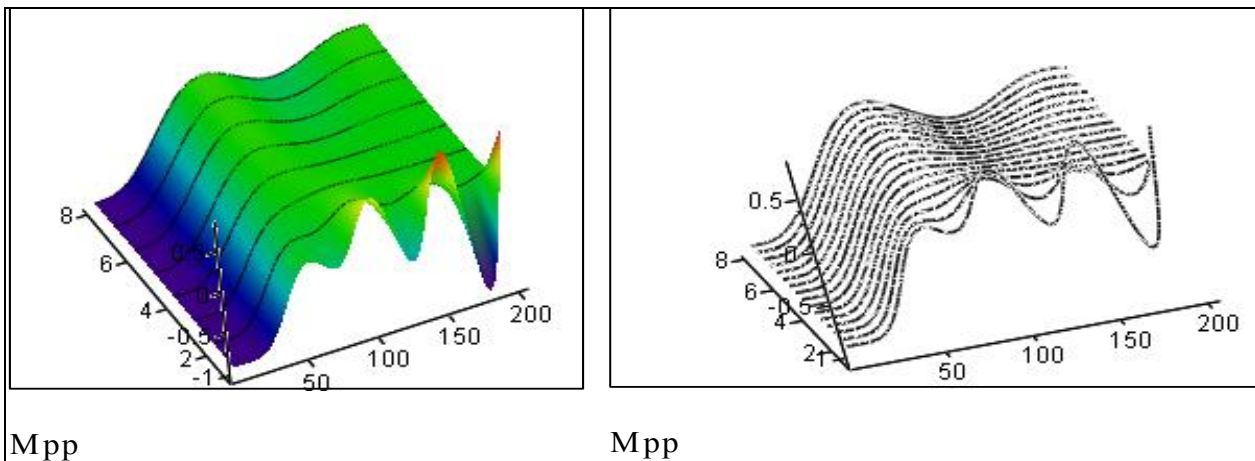


Рис.1.10. Серія перехідних процесів в САУ при варіації запізнення

### Висновки

Дивимось на вихід програми – серію графіків перехідних процесів - і бачимо дещо антиінтуїтивну інформацію для роздумів: при збільшенні запізнення з 0 (графік №8) до 8 кроків квантування (графік №1) якість перехідних процесів спочатку покращується, а вже потім –погіршується. Тобто: запізнення не завжди шкідливе.

### Контрольні запитання

1. Джерела запізнення. Способи „боротьби” з запізненням.
2. Існує критичне запізнення, при якому САУ стає нестійкою. Чи так це?
3. Способи визначення критичного запізнення.

### Завдання для самостійної роботи

Побудуйте графіки перехідних процесів з запізненням для системи неколивальним перехідним процесом.

Вважається, що лікувати людей має право спеціаліст з відповідним дипломом, і, головне, – відповідним рівнем знань і вмінь. Для конструювання "інтелектуальних регуляторів" на базі нейронечіткої логіки та генетично-мурашиних алгоритмів немовби не треба знати професійні основи ТАУ – теорії автоматичного управління. В розділі 4 подано розробку адаптивного управління для банківської системи. Банкір ніколи не придумає таку систему, бо не отримує відповідних знань з теорії управління, зокрема, про екстремальні регулятори.

## 1.4 Розробка і дослідження адаптивної САУ з настроюванням власної частоти

### Проблеми розробки адаптивних САУ для масових об'єктів

Згадаємо історію. Десять тисяч років тому варварські орди зруйнували Римську імперію. Вважається, що однією з причин були зміни в технології виробництва зброї. В ту епоху зброя – мечі, щити, місьюрки – була бронзовою і досягла межі досконалості. Одночасно вичерпалися доступні джерела міді, цинку, срібла, олова. Повний комплект зброї могли дозволити тільки дуже багаті. Армії були маленькими (3000–5000), професійними. Варвари навчилися виплавляти залізо з доступної болотної руди. Їхні ковалі могли виробляти зброю у великих кількостях. До Італії вторгалися орди чисельністю в сотні тисяч, усі при залізній зброї... Ідеї адаптивних САУ виповнюється майже сто років, практично значимі розробки з'явилися 50 років тому і довго залишалися експериментальними, поки не з'явилася "масова зброя" – потужні і дешеві мікроконтролери.

Досить взяти 3–4 випадкових факти сучасності, наприклад:

☞ сьогодні в Англії слово «тракторист» вже не є синонімом «дурника», бо трактор обладнано багатьма мікропроцесорними системами та Інтернетом;

☞ основою дитячого конструктора Mindstorms є "дорослий" мікропроцесорний блок управління роботами, доповнений "дорослими" сенсорами, приводами і засобами зв'язку і всього за \$200;

☞ відомий комп'ютерний щотижневик пише в одному номері "Эпитафия экспертным системам", а в другому – "Экспертные системы ещё живы", висновки приходять самі: Світ вже вступив в епоху інтелектуалізації і рашморізації масових об'єктів управління, і

♣ **технічна реалізація** систем управління будь-якої інтелектуальної складності – **сьогодні не проблема**, але

♠ існує дефіцит придатних для практичного втілення концепцій і методів "інтелектуальних" систем управління (з самонастроюванням, самоорганізацією, навчанням) – це актуальна проблема сьогодення [13].

### Термінологічні проблеми

**Штучний інтелект** – це, в першу чергу, комплексне історико-соціо-психологічне явище, яке О. Вентцель вважала необхідним **визначати** за допомогою "серії розмов по поведінці". От, наприклад, таке **означення** подане авторитетними спеціалістами і масовим тиражем: "В 1986 и 1987 гг. это был ИИ (Искусственный интеллект). Не важно, что машины того времени едва могли посчитать в реальном времени скорости катящихся шаров. Каждый раз при появлении усовершенствованного продукта продавец начинал вопить: "Теперь это с ИИ!". Никто даже толком не понимал, что в действительности означает "ИИ", и припёртым в споре в угол апологетам ИИ не оставалось ничего другого, как признать, что это было просто надуманное словосочетание, если не полнейший обман с самого начала" [1].

**Адаптивна САУ** – у навчальному посібнику **визначено**: "Для повышения качества управления объектами при неполной информации (в условиях неопределённости) применяются **принцип адаптации**. Термин "адаптация" заимствован из биологии, где им обозначают свойство приспособления организма к изменениям внешней среды" [41].

І цей науковий напрямок теж має ту ж саму соціо-психологічну специфіку: "Термины "адаптация", "самообучение", "обучение" модны в современной теории автоматического управления. К сожалению, как правило, эти термины не имеют однозначного толкования, а зачастую не имеют просто никакого толкования. Это создаёт благоприятную почву для безудержных фантастических рассуждений" [42]. Зараз закінчується друга хвиля такого процесу, породжена новими технічними можливостями і пошуком нових "ніш" на перенасиченому ринку програмно-технічних засобів і, головне, високою потребою на дійсно інтелектуальні системи управління.

Сорок років тому при створенні напрямку була відзначена його математична специфіка: "Удивительно, что уже сама конструкция соответствующего математического аппарата для размещения этих проблем и выбор соответствующего математического аппарата для их решения требуют значительной осторожности и определённой изобретательности. ...достаточно ясно, что никакая единая формулировка и никакой единый метод не будут достаточно мощными и исчерпывающими для решения разнообразных вопросов, которые могут возникнуть, или получения разнообразных ответов, которые могут потребоваться. **Комбинация искусно сочетаемых различных идей и методов – вот, что нам понадобится**" (виділено нами) [6].

Варто навести і епіграф до розділу "Процеси з адаптацією", звідки взято попередню цитату: *"Не верь ничему, где бы ты это ни прочёл и кто бы ни сказал это, не верь, даже если это сказал я, – если это не согласуется с твоим рассудком и чувством здравого смысла"* [Будда (623–544 до н.е.)].

**Масовий об'єкт управління.** Об'єкт управління будемо вважати масовим, якщо він випускається у мільйонних кількостях, є відносно дешевим, використовується або експлуатується неспеціалістом з проектування і ремонту, або діє взагалі майже самостійно. Під цю дефініцію підпадають автомашини і сільгоспмашини, підйомно-транспортні механізми – крани, екскаватори, роботи-маніпулятори, теплообмінники, кондиціонери, сушарки та інші термодинамічні системи і не підпадають літаки-винищувачі чи космічні кораблі. Трактор з мікропроцесорною адаптивною системою управління обприскування стає не тільки більш ефективнішим, але ще й на додаток більш інтелектуалізованим робочим місцем.

#### **Проблеми створення адаптивних САУ для масових об'єктів**

Як відомо, адаптивні САУ є нелінійними і нестационарними, навіть якщо об'єкт управління – лінійний. В запровадженні адаптивних САУ у масову інженерну практику можна виділити три класи проблем:

– проблеми реалізації складних, нелінійних алгоритмів, що "остаточно" вирішилися на базі сучасних апаратно-програмних платформ, але виявився брак життєздатних схем і концепцій для адаптивних САУ;

– проблеми міжгалузевих бар'єрів. Традиційно, в першу чергу, адаптивні САУ розроблялися для об'єктів авіаційної та ракетно-космічної техніки, в інші галузі (у нас) цей досвід з відомих причин не переносився;

– проблеми небезпечності адаптивних САУ. Стисло це можна сформулювати як емпіричний закон (з винятком): **чим більше інтелекту ми вкладаємо в систему, тим більш дорозжчими і різноманітнішими будуть її відмови.**

На сьогодні проблема безпечності, стійкості до широких класів збурень є головною для систем управління масових об'єктів. Перший крок у розв'язанні проблеми – перенесення на нові об'єкти та апаратні платформи (= рімейк) тих зразків адаптивних САУ 50–80-х років, що пройшли жорсткий натуральний відбір і вижили. Далі розглядається спрощений приклад синтезу і аналізу одної з достиглих для масового запровадження адаптивних САУ.

#### **Автоматизація розробки адаптивних САУ. Принципи і приклад**

Сьогодні використання САПР стало необхідною умовою будь-яких технічних систем. Непомітною в силу очевидності концептуальною основою сьогоднішніх САПР є те, що середньостатистичний коефіцієнт новизни оточуючих нас технічних систем і виробів становить 2–5%. Принципово нові вироби і технології є рідкими, епохальними подіями.

Якщо виділити і параметризувати структуру якогось класу технічних систем, то проектування при малих значеннях коефіцієнта новизни можна звести до заповнення розробником специфікації (фрейма) об'єкта даного класу [13, 37]. Якість результату буде обумовлена трьома факторами

- якістю фрейма,
- повнотою бази знань (попередніх розробок) і

– кваліфікацією розробника.

"Напрямний" фрейм-сценарій (НФС) заповнення специфікації об'єкта є фактично спеціалізованою експертною надбудовою над іншими – оптимізаційними, розрахунковими і графічними – модулями САПР. В цій "екологічній ніші" експертних систем існують зразки настільки ефективні, що сам факт їх існування буває "фірмовим секретом" розробника, а фрейм описується в термінах "це схоже на гру у сім питань". Між іншим така методологія ("мова виконуваних специфікацій") є стандартом Міноборони США для розробки програмного забезпечення.

Вибір структур і алгоритмів адаптивної САУ теж може бути автоматизовано використанням відповідного НФС розробки. НФС є лінгвістичною моделлю об'єкта проектування, такою, що коректні відповіді спеціаліста на питання НФС породжують специфікацію системи, реалізація якої може бути повністю автоматизована. Для програмних систем це означає, що специфікація автоматично транслюється у робочу програму [5, 7]. Розробка ефективного НФС вимагає певного концептуального бачення проблеми [8, 9]. Розглянемо приклад роботи з можливим НФС розробки адаптивної САУ – питання і вибрані відповіді.

"1. *Основний контур системи?*" – Об'єкт управління, (для прикладу) – 3-ого порядку з такою інтерпретацією компонентів вектора стану:  $x$  – координата;  $v$  – швидкість,  $w$  – прискорення. Закон управління – лінійний регулятор вектора стану. Вектор управління – одновимірний:  $u = k_1 * x + k_2 * v + k_3 * w$ .

"2. *Об'єкт настроювання?*" – Параметри  $k_1$ ,  $k_2$ ,  $k_3$  регулятора.

"3. *Метод настроювання?*" – З можливих альтернатив настроювання параметрів регулятора вибираємо ітеративний метод, тобто контур самонастроювання зі зворотним зв'язком, що підтримує задане значення показника якості регулювання.

"4. *Показник якості перехідного процесу?*" (= змінна управління контуру самонастроювання) – З можливих альтернатив – від узагальненого інтегрального квадратичного критерію до викиду – вибираємо показник-індикатор, а саме: **частоту коливань у перехідному процесі, або власну демпфовану частоту  $\omega_d$** . Індикатор – вимірювана або спостережувана змінна, що пов'язана монотонною залежністю з основним критерієм якості, інші назви – підциль, псевдоціль, робочий критерій.

"5. *Засоби дії на показники якості управління в основному контурі?*" – Вибираємо безпосередню зміну параметрів регулятора  $k_1$ ,  $k_2$ ,  $k_3$  – це будуть **змінні управління в контурі самонастроювання**. Реалізація такого управління у сучасних цифрових регуляторах потребує тільки зміни параметрів програми управління.

"6. *Спосіб вимірювання власної частоти  $\omega_d$ ?*" – З можливих альтернатив: перетворення Фур'є, виділення періодичного сигналу з нормалізацією і підрахунком переходів через нуль..., вибираємо вимірювання різниці частот ( $\omega_z - \omega_d$ ) за допомогою двох фільтрів аналогових або дискретних фільтрів з протилежними частотними характеристиками.  $\omega_z$  – потрібне значення частоти, що обумовлене потрібною швидкістю основного контуру.

"7. *Спосіб забезпечення вхідної інформації?*" – Щоб вимірювати показники ПП, треба стимулювати перехідні процеси (ПП). Можливі альтернативи стимуляції ПП – а) періодично подавати імпульси на вхід основного контуру, б) подавати шум, г) нічого не подавати, якщо в основному контурі досить шумів та сигналів управління.

Для професіонала відповіді на питання 1–7 є майже вичерпними для побудови адаптивної САУ, блок-схему якої подано на рис. 1.11.

В цьому прикладі вибрано варіант, коли потрібна якість управління  $Q_z$  задається непрямо – еталонною власною демпфованою частотою  $\omega_z$ . Дійсна частота  $\omega_d$  при коректному виборі параметрів регулятора  $k_2$  і  $k_3$  є монотонною функцією параметра регулятора  $k_1$  – "штрафу за помилку", а тривалість перехідного процесу  $T_r$  приблизно пропорційна  $\omega_d$ . Таким чином,  $\omega_d$  – параметр досить просто вимірюваний і досить просто пов'язаний з іншими параметрами САУ:  $k_1$  монотонно змінює  $\omega_d$ , що монотонно пов'язана з  $Q_d$ .

Згідно з "лінійною логікою" слід вибрати метод та пристрій для оптимального вимі-

рювання "чистої" частоти  $\omega_d$ . Розглянемо дві альтернативи реалізації вимірювача частоти – аналогову на базі двох лінійних фільтрів і випрямлювачів (рис. 1.12) і програмну реалізацію на базі еквівалентної дискретної моделі цієї аналогової системи.

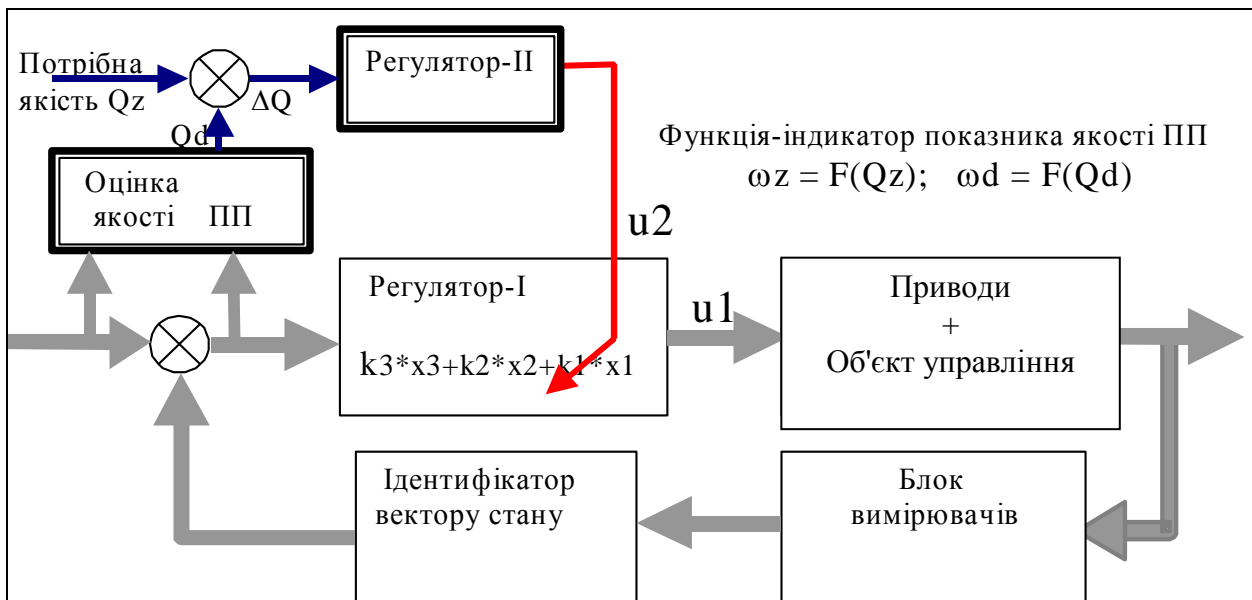


Рис. 1.11. Блок-схема САУ з контуром самонастроювання регулятора

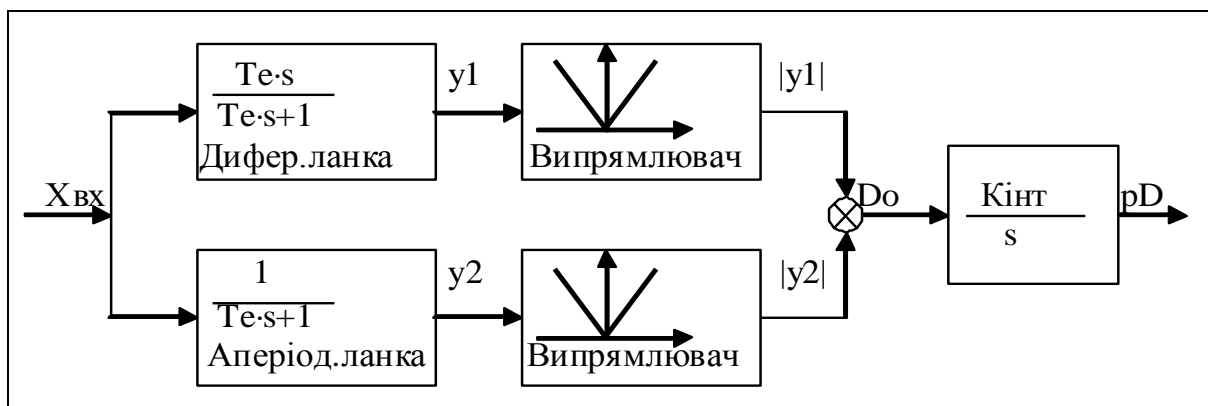


Рис. 1.12. Блок-схема частотного дискримінатора

Записуємо передаточні функції фільтрів, поряд з ними подаємо схеми цих фільтрів. Нижче записуємо вирази для амплітудних частотних характеристик фільтрів та вираз для різниці  $Ad(\omega) = A1(\omega) - A2(\omega)$ .

$$W1(T, s) := \frac{T \cdot s}{T \cdot s + 1} \quad \begin{array}{c} \text{---} \text{---} \text{---} \\ | \\ \text{---} \end{array} \quad T = R \cdot C$$

$$W2(T, s) := \frac{1}{T \cdot s + 1} \quad \begin{array}{c} \text{---} \text{---} \text{---} \\ | \\ \text{---} \end{array} \quad T = R \cdot C$$

$$A1(\omega) := \frac{T \cdot \omega}{\sqrt{1 + T^2 \cdot \omega^2}}; \quad A2(\omega) := \frac{T \cdot \omega}{\sqrt{1 + T^2 \cdot \omega^2}}; \quad Ad(T, \omega) := \frac{T \cdot \omega - 1}{\sqrt{1 + T^2 \cdot \omega^2}}.$$

Бачимо, що характеристики такого вимірювача частоти (рис. 1.12) нелінійні. Ще один недолік схеми – вихід пропорційний не тільки частоті, але й амплітуді вхідного сигналу. Але, як побачимо далі, цей недолік є дуже великою перевагою. Єдиним реальним недоліком схеми є неможливість використання лінійних методів. Потрібні методи і технології, орієнтовані на імітаційне моделювання.

Сьогодні обробка сигналів виконується переважно в цифрових пристроях. Побудуємо еквівалентну дискретну математичну модель частотного дискримінатора, де окремі рівняння відповідають елементам аналогової схеми.

Фільтри: суматор і випрямлювач: інтегратор:

$$\begin{pmatrix} y1_{k+1} \\ y2_{k+1} \end{pmatrix} := \begin{pmatrix} y1_k \\ y2_k \end{pmatrix} + \begin{bmatrix} \omega e \cdot (X(k \cdot T) - y1_k) \\ dX(k \cdot T) - \omega e \cdot y2_k \end{bmatrix} \cdot T; \quad pD_1 := 0; \quad Do_k := (|y1_k| - |y2_k|); \quad pD_{k+1} := \left[ \frac{pD_k \cdot (k-1) + Do_k}{k} \right].$$

У цій моделі імена змінних аналогічні позначенням на рис. 1.12. Зокрема,  $y1, y2$  – виходи фільтрів;  $X(k \cdot T), dX(k \cdot T)$  – вхідний сигнал і його похідна;  $Pd$  – еквівалент інтеграла,  $Pd_1$  – його початкове значення. Записані рівняння виконуються в середовищі математичного пакета і водночас – це майже готова програма реалізації у мікропроцесорній системі. Як приклад досліджень цифрової реалізації на рис. 1.13 подано графіки процесів у модулі "дискримінатор" для випадку загасаючого гармонічного сигналу – імітації імпульсної перехідної характеристики (ІПХ). І для таких нестационарних сигналів усталений вихід монотонно залежить від помилки підтримки заданої частоти.

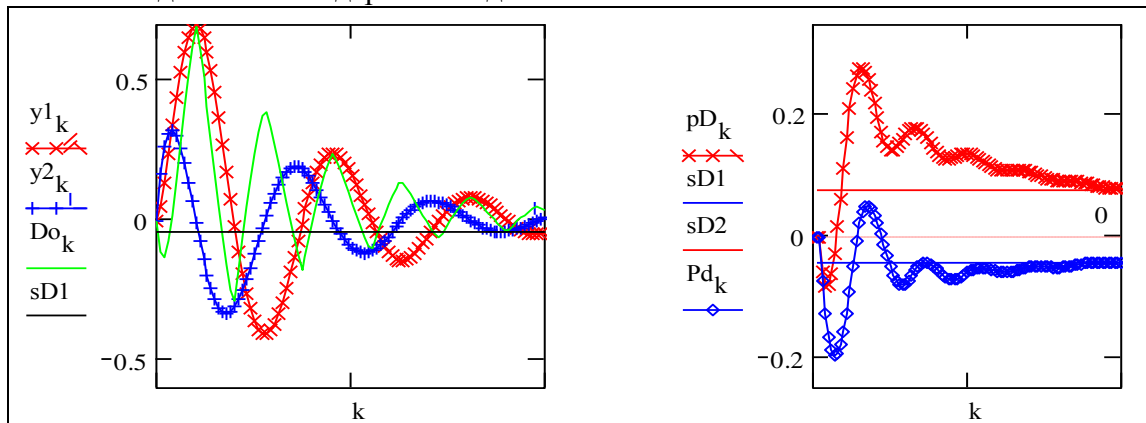


Рис. 1.13. Перехідні процеси у частотному дискримінаторі

Аналітичне дослідження статичних і динамічних характеристик частотного дискримінатора пов'язане із значними математичними ускладненнями, але можливе, наприклад, на базі методу гармонічної лінеаризації. Можна показати працездатність пристрою на більш складних за спектром вхідних сигналах. Але це зроблено вже на моделі повної адаптивної системи. На рис. 1.13  $pD$  і  $sD1$  – вихід дискримінатора і усталене значення для різниці частот  $\omega d - \omega e = 0.1$ ,  $Pd$  і  $sD2$  – те ж саме для  $\omega d - \omega e = -0.1$ .

### Розробка математичної моделі САУ з системою самонастроювання

Опорний параметр для системи самонастроювання – власна демпфована частота, визначається в процесі синтезу основного контуру, що є тривіальним для управління вектором стану лінійним об'єктом і тому не наводиться. Подана далі система рівнянь разом з розташованими поряд графіками і зонами вхідних даних є "стендом" для проведення обчислювальних експериментів. Головні компоненти "стенду" зібрані в межах однієї екранної сторінки, що дозволяє змінювати параметри, рівняння і оперативно спостерігати перехідні процеси.

Для розробника-професіонала саме алгоритмів управління і професіонала у предметній області об'єкта управління найзручніша альтернатива – повністю прозора і відкрита для змін онлайн-діюча (live) математична модель. Проблеми виникають тільки з розташуванням в межах одного екрана моделі високого порядку.

На рис. 1.14 подано приклад спрощеної „стартової” версії такого стенду з налаштуванням тільки одного параметра регулятора –  $K1$  і без елементів управління експериментом. Для налаштування періодично подаються імпульси (функція  $Imp(n, N)$ , де  $N$  – кількість тактів квантування між імпульсами). Фактично це система нелінійних різницевих рівнянь 7-го порядку. Перші три рівняння описують динаміку об'єкта управління. Рівняння

4–6 описують частотний дискримінатор і останнє рівняння – вихід контуру самонастроювання. Користувач може вводити параметри як у зонах вхідних даних – кнопками, спіннерами, слайдерами, так і безпосередньо у самих рівняннях. Якість контуру самонастроювання визначають два параметри: **Kadapt** – "штраф за помилку" власної частоти;  $\alpha$  – параметр згладжування вихідного сигналу дискримінатора. Ці параметри можуть бути вибрані аналітично (для лінеаризованої моделі контуру самонастроювання) і пошуковими методами – а) "вручну" в середовищі цього документа, за 10–25 хвилин; б) за допомогою пошукової системи, що фактично буде контуром самонастроювання 3-го рівня.

$$\begin{pmatrix} x_{n+1} \\ v_{n+1} \\ w_{n+1} \\ y1_{n+1} \\ y2_{n+1} \\ pD_{n+1} \\ K1_{n+1} \end{pmatrix} := \begin{pmatrix} x_n \\ v_n \\ w_n \\ y1_n \\ y2_n \\ pD_n \cdot \alpha \\ K1_n \end{pmatrix} + \begin{pmatrix} v_n \\ w_n \\ (K1_n - A0) \cdot x_n + (k2 - a1) \cdot v_n + (k3 - a2) \cdot w_n + K1_n \cdot \text{Imp}(n, N) \\ \omega Z \cdot (x_n - y1_n) \\ v_n - \omega Z \cdot y2_n \\ \left[ \frac{(|y1_n| - |y2_n|)}{2} \right] \cdot \omega \text{kvant} \cdot (1 - \alpha) \\ (-pD)_n \cdot \text{Kadapt} \end{pmatrix} \cdot T$$

$\uparrow \omega \cdot -1$     $K \equiv 2.8$     $\omega Z \equiv .87$     $A0 \equiv 1$     $a0 = 1$     $\alpha \equiv .2$     $\text{Kadapt} \equiv 6$     $\omega 0 := .87$   
 $\downarrow \omega \cdot -2$     $xplu_{\text{ш}} := x_n$     $Kplu_{\text{ш}} := K1_n$     $K1_1 := -8$

Рис. 1.14. Приклад стенда для дослідження адаптивної САУ

На рис. 1.15 подано приклади перехідних процесів у адаптивній САУ для випадків початково нестійкого основного контуру. Вибрано аперіодично нестійкий без управління об'єкт класу "нестійкий маятник", тому основний контур має дві границі стійкості – залежності від параметра K1 – верхню (велике K1 і коливальна нестійкість), і нижню (мале K1 і аперіодична нестійкість).

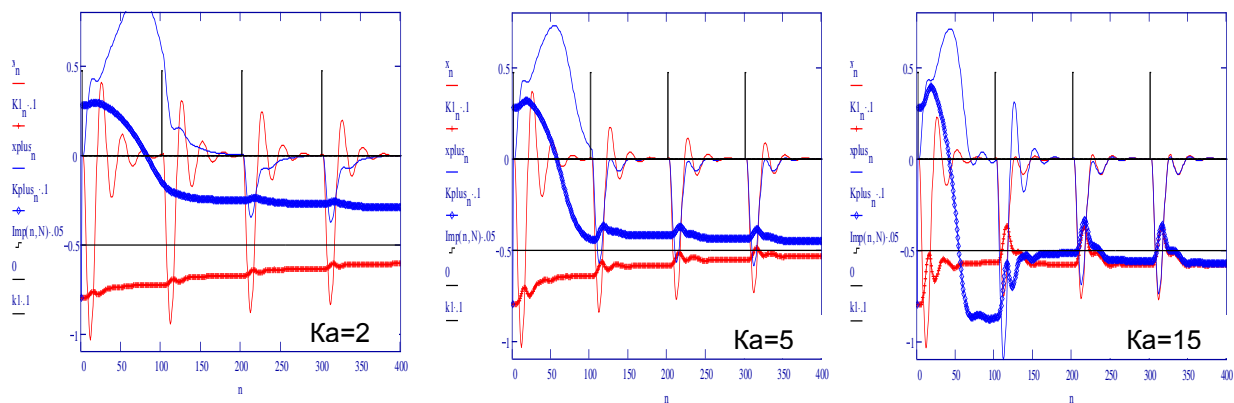


Рис. 1.15. Перехідні процеси в адаптивній САУ при варіації параметра Kadapt

На графіках рис. 1.15 тонкі лінії – перехідні процеси в основному контурі – реакції на тестові імпульси, що подаються з певним періодом, жирні лінії – перехідні процеси в контурі адаптації (самонастроювання), тобто зміна параметра K1 основного контуру (штрафу за помилку координати). Горизонтальні лінії – потрібні виходи основного контуру (нульова помилка координати) і контуру адаптації – розрахункове значення K1, що забезпечує вибраний (заданий) перехідний процес в основному контурі. Для прикладу на етапі синтезу основного контуру вибрано перехідний процес з підвищеною коливальністю. Неважко на даній системі моделювання поставити й інші обчислювальні експерименти.



Розглянемо приклад дослідження впливу шумів в основному контурі. Ці шуми, з одного боку – завади, обурення, з іншого – стимулятор реакцій основного контуру, тобто джерело потрібної для самонастроювання інформації. Для моделювання системи при наявності шумів достатньо сконструювати функцію Шум(.) і замінити нею функцію генерації тестових імпульсних сигналів Imp(.) у «стенді». На рис. 1.16 подано графіки перехідних процесів в адаптивній САУ при дії шумів різної амплітуди ( $A_s=0.1$ ,  $A_s=0.3$ ). Дослідження дозволяють зробити висновок, що в певному діапазоні значень амплітуди  $A_s$  шуму САУ буде стійкою і задовільною, але процес адаптації стає ймовірнішим і в середньому повільнішим, ніж при дії імпульсних тест-сигналів.

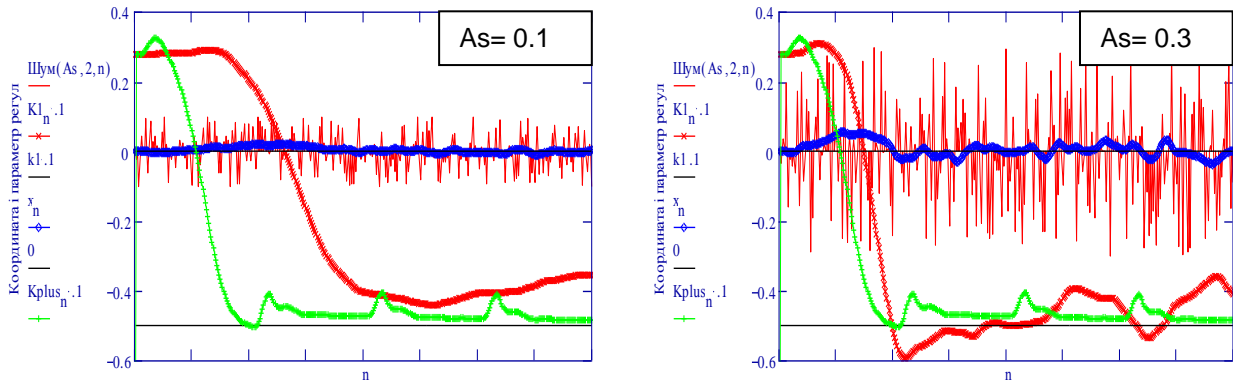


Рис. 1.16. Перехідні процеси при наявності шумів в основному контурі

На графіку для порівняння подано процес зміни  $K_1$  при дії імпульсних тестових сигналів. При занадто малій амплітуді шуму настроювання стає повільним, а при великій – контур адаптації працює нестабільно.

### Розробка сценарію імітаційного моделювання адаптивної САУ

Поведінка адаптивних САУ може залежати від рівнів вхідних сигналів та їх взаємодії, особливостей включення, рівня і характеру шумів та збурень, і все це бажано виявити і вилікувати до апаратної реалізації. Характерна особливість розробки адаптивних САУ – **інтеграція етапів розробки і випробувань**. В нашому випадку цілі експериментів: а) пошук оптимальних параметрів контуру самонастроювання, б) визначення границь стійкості і задовільності адаптивної САУ в цілому, г) виявлення усіх ганджів адаптивної САУ, особливо концептуальних. Розглянута адаптивна САУ одна з небагатьох, що використовувалась, але і для неї існують досить прості варіанти вхідних сигналів в основному контурі, за яких контур адаптації виводить систему з ладу, «заганяючи» параметр  $K_1$  на верхню або нижню границю. Ця характерна для адаптивних САУ проблема може бути вирішена.

### Особливості САУ з настроюванням власної частоти

Розглянута САУ належить до класу “швидких” адаптивних систем, в яких швидкість перехідних процесів в контурі адаптації такого ж порядку, як і в основному контурі. Таким чином відсутній основний недолік “класичних” адаптивних САУ – запізнення в контурі адаптації і мала швидкість настроювання. Досягнуто це за рахунок ланцюга „недосконалостей”:

- ☞ замість оцінки якості перехідного процесу за узагальненим квадратичним критерієм (а на ньому побудовано 90% робіт, присвячених дослідженню оптимальних і адаптивних САУ) вибрано індикатор – частоту коливальності у перехідних процесах;
- ☞ замість цієї частоти використовується показник, що є нелінійною функцією не тільки частоти, а також і амплітуди вхідного сигналу;
- ☞ замість інтегрування сигналу помилки використовується цифрове згладжування.

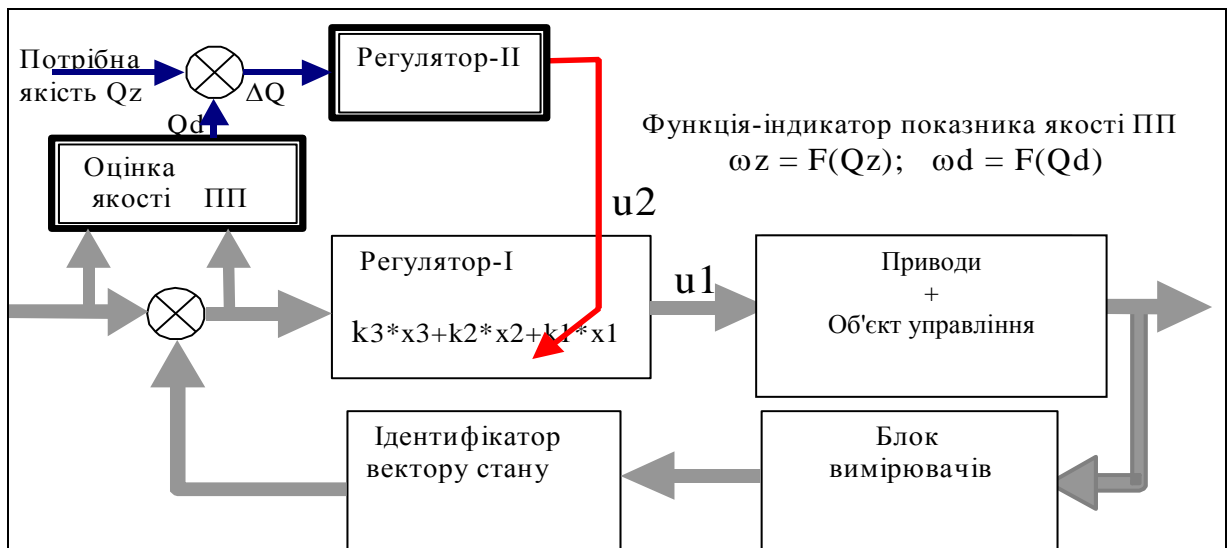
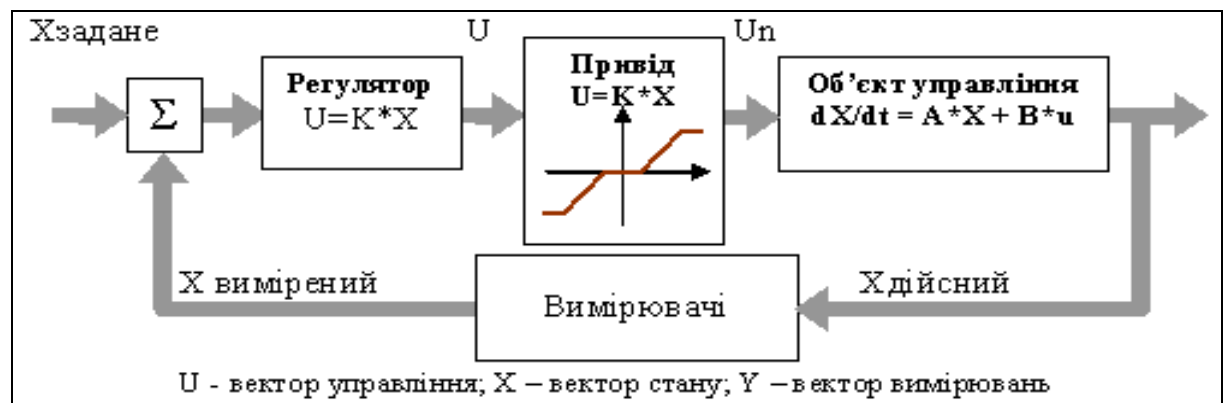
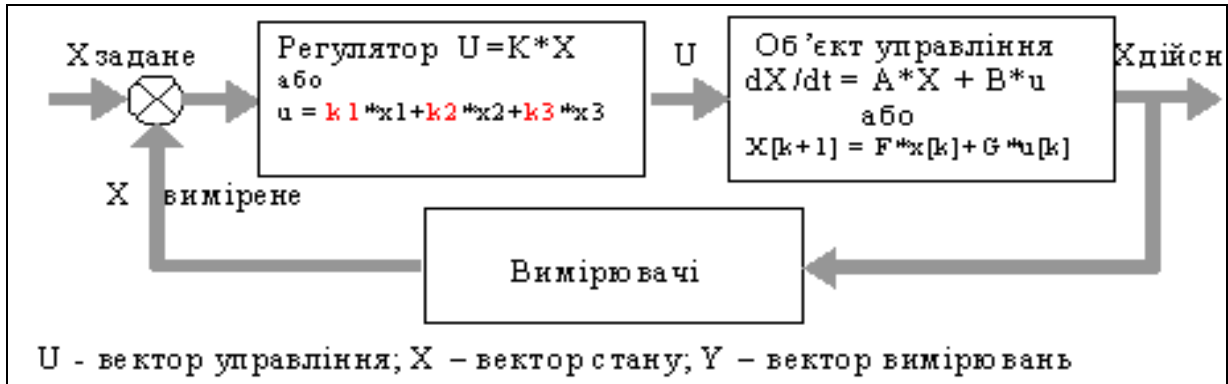
Плата за позитивні практичні властивості системи полягає в тому, що система є суттєво нелінійною, занадто інтегрованою, такою, що не допускає декомпозиції на функціональні підсистеми, швидкі і повільні складові процесів і в цілому дуже незручна для традиційних методів аналізу і синтезу систем управління. Як резюме можна повторити пода-

ну на початку цитату: "Комбинация искусно сочетаемых различных идей и методов – вот, что нам понадобится".

### Висновки

Отримаємо дешево, ефективну і небезпечну, як місток без перил, "інтелектуальну" систему. Для того, щоб ставити її на масові об'єкти – сушарки, молотарки..., треба провести великий обсяг випробувань і досліджень системи при усіх можливих і "неможливих" умовах функціонування і вбудувати в неї надійні і природні "запобіжники на всі ситуації".

Отримали моделі і програми синтезу та моделювання таких задач:



## 2 СИНТЕЗ І ДОСЛІДЖЕННЯ САУ ЗІ СПОСТЕРІГАЧЕМ

В цьому розділі подано класичну задачу:

### Синтез і дослідження імпульсної САУ з швидким регулятором і спостерігачем.

Головна увага приділена технологіям конструювання робочих моделей, придатних для проведення обчислювальних експериментів та методів синтезу, придатних для використання в мікроконтролерах.

### 2.1 Синтез та дослідження імпульсної САУ зі спостерігачем стану

Починаємо з освоєння професійних інструментів – методів синтезу типових систем управління – зі спостерігачем вектора стану і орієнтованих на реалізацію мікропроцесорними засобами. Дослідження різних типів імпульсного управління. Розглядається об'єкт 3-го порядку і управління синтезується саме як імпульсне.

#### Теоретичні відомості

Спостерігач – це не маленький зелений чоловічок. Імпульсні системи – це не сурогатні замітники неперервних систем. З одного боку – при зменшенні кроків квантування вони наближуються до неперервних, з іншого – у них є свої, тільки їм притаманні властивості. В цій роботі розглядається так званий аперіодичний регулятор (інші назви "швидкий регулятор", "регулятор з кінцевим часом усталення", але прогресивне людство так і не винайшло влучної назви). До неперервних лінійних регуляторів "швидкий" регулятор відноситься так само, як швидке перетворення Фур'є відноситься до звичайного перетворення Фур'є.

Згадаємо, що наближення якоїсь залежності рядом Фур'є теоретично вимагає безкінечної кількості складових; дискретну залежність з  $N$  значень можна наблизити  $N$  складовими ряду Фур'є. Аналогічно при лінійному неперервному управлінні початкова помилка стає нульовою у безкінечності. Швидкий регулятор може забезпечити нульову помилку точно на  $N$ -ому кроці для динамічної системи  $N$ -го порядку. Виникає природне питання – якою ціною? Чи є у такого управління дійсні переваги? Відповіді на ці та інші питання – в цій роботі. Далі розглядається загальна задача синтезу САУ зі спостерігачем на прикладі синтезу швидкого регулятора і швидкого спостерігача.

#### Синтез імпульсної САУ зі спостерігачем

Задано  $N$ -вимірну систему з регулятором вектора стану. Об'єкт управління – неперервний. Записуємо канонічні рівняння САУ зі спостерігачем вектору стану і скалярним управлінням.

Об'єкт:                      Вимірювачі:                      Спостерігач + модель вимірювачів:                      Регулятор:

$$\frac{d}{dt}x_0 = A \cdot x_0 + B \cdot u; \quad y_0 = C \cdot x_0; \quad \frac{d}{dt}x_n = A \cdot x_n + B \cdot u + L \cdot (y_n - y_0); \quad y_n = C \cdot x_n; \quad u = K \cdot x_n.$$

Підставляємо управління в рівняння об'єкта. Записуємо еквівалентне різницеве рівняння:

$$\frac{d}{dt}x_0 = A \cdot x_0 + B \cdot u; \quad u = k^T \cdot x_0; \quad \frac{d}{dt}x = (A + B \cdot k^T) \cdot x; \quad x^{(k+1)} = F_p \cdot x^{(k)},$$

$$\text{де } F_p = I + M \cdot \frac{dT}{1!} + M^2 \cdot \frac{dT^2}{2!} + M^3 \cdot \frac{dT^3}{3!} + \dots, \text{ де } M = A + B \cdot k^T.$$

Для задачі синтезу обмежимося першим наближенням перехідної функції стану  $F$ :

$$F_p = I + (A + B \cdot k^T) \cdot dT.$$

Підставляємо  $F$  в еквівалентне різницеве рівняння, виконуємо  $Z$ -перетворення:

$$x^{(k+1)} = [I + (A + B \cdot k^T) \cdot dT] \cdot x^{(k)}; \Rightarrow \square z \cdot X(z) = [I + (A + B \cdot k^T) \cdot dT] \cdot X(z).$$

Виконуємо алгебраїчні перетворення і записуємо характеристичне рівняння:

$$[I \cdot z - [I + (A + B \cdot k^T) \cdot dT]] \cdot X(z) = 0 \Rightarrow \square [I \cdot z - [I + (A + B \cdot k^T) \cdot dT]] = 0.$$

Аналогічно отримуємо Z-характеристичне рівняння для спостерігача. А саме – запишемо рівняння вільного руху спостерігача ( $u, x_0$  – зовнішні входи):

$$\frac{d}{dt} x_n = A \cdot x_n + B \cdot u + L \cdot C \cdot (x_n - x_0) \quad \Leftrightarrow \quad \frac{d}{dt} x_n = A \cdot x_n + L \cdot C \cdot x_n \quad \Leftrightarrow$$

$$\frac{d}{dt} x_n = (A + L \cdot C) \cdot x_n \quad \square \square.$$

Записуємо еквівалентне різницеве рівняння:

$$x_n^{(k+1)} = F_n \cdot x_n^{(k)}, \text{ де } F_n = I + M \cdot \frac{dT}{1!} + M^2 \cdot \frac{dT^2}{2!} + M^3 \cdot \frac{dT^3}{3!} + \dots, \text{ де } M = (A + L \cdot C).$$

Для задачі синтезу обмежимося, як і при синтезі регулятора, першим наближенням буде вираз:  $\square \square \square \square \square F_n = I + (A + L \cdot C) \cdot dT$ .

Підставляємо F у різницеве рівняння, виконуємо Z-перетворення:

$$x_n^{(k+1)} = [I + (A + L \cdot C) \cdot dT] \cdot x_n^{(k)} \Rightarrow z \cdot X_n(z) = [I + (A + L \cdot C) \cdot dT] \cdot X_n(z).$$

Виконуємо алгебраїчні перетворення і записуємо характеристичне рівняння спостерігача:

$$[I \cdot z - [I + (A + L \cdot C) \cdot dT]] \cdot X_n(z) = 0 \Rightarrow \square [I \cdot z - [I + (A + L \cdot C) \cdot dT]] = 0.$$

**Зауваження.** Детермінанти записані в означеннях математичного пакета: "|A|" замість "det(A)".

Якщо розкрити ці детермінанти, то отримаємо алгебраїчні рівняння N-го ступеня відносно комплексної змінної Z. Будемо вважати, що вибрано еталонний перехідний процес і крок квантування dT, тобто відомі коефіцієнти еталонного характеристичного рівняння.

Прирівнюємо коефіцієнти характеристичного рівняння (1) їх потрібним значенням – так отримуємо N рівнянь. У випадку скалярного управління маємо N невідомих параметрів регулятора (або спостерігача). В загальному випадку невідомих параметрів N×M, де M – розмірність вектора управління. Додаткові рівняння отримуємо із розв'язання оптимізаційних задач і обмежень. Можна бачити, що схема синтезу лінійного імпульсного регулятора вектора стану ідентична схемі синтезу неперервного лінійного регулятора – порівняйте, вкажіть різницю...

### Управління з кінцевим часом усталення

Як було сказано вище, при управлінні з кінцевим часом усталення, незалежно від початкових умов, помилка на N- ому кроці перехідного процесу (ПП) стає нульовою. Зауважимо, що можна зробити так, щоб тривалість ПП була більше N. Поведінка лінійної імпульсної динамічної системи повністю визначається її коренями. Визначимо, якими вони повинні бути для "швидкого регулятора". Запишемо різницеве рівняння процесу для перших двох кроків, потім послідовно підставляємо значення  $x^{(1)}$  із першого рівняння в друге:

загальна залежність:  $x^{(k+1)} = F \cdot x^{(k)}$ ; перший крок  $x^{(1)} = F \cdot x^{(0)}$ ; другий крок  $x^{(2)} = F \cdot x^{(1)}$ .

Результат підстановки:  $x^{(2)} = F^2 \cdot x^{(0)}$ . Узагальнюємо отриману залежність між станами і

записуємо для N-го кроку:  $x^{(N)} = F^N \cdot x^{(0)}$ . Згідно з дефініцією швидкого регулятора повинно бути  $x^{(N)} = 0$  при будь-яких початкових умовах  $x^{(0)}$ . Це може бути тільки за умови:

$F^N = 0$ . З математики відомо, що для матриці F порядку N для виконання цієї умови необхідно, щоб усі власні числа - корені характеристичного рівняння - були нульовими. А це можливо, коли всі коефіцієнти характеристичного рівняння дорівнюють нулю, тобто хара-

ктеристичне рівняння буде:  $Z^N = 0$ . Подаємо приклад перехідної матриці стану для системи ( $N = 3$ ) з швидким регулятором і спостерігачем. Бачимо: на третьому кроці процесу компоненти 4, 5, 6 вектора стану (це оцінки) стануть нульовими. а на 6-му кроці – нульовим стане увесь вектор стану.

$$M1 := \begin{pmatrix} 1 & 0.5 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0.5 & 0 & 0 & 0 \\ -4 & -6 & -2 & -3.5 & -7 & -2.5 \\ 0 & 0 & 0 & -1.5 & 0.5 & 0 \\ 0 & 0 & 0 & -4.5 & 1 & 0.5 \\ 0 & 0 & 0 & -4.5 & 1 & 0.5 \end{pmatrix} \quad M1^3 = \begin{pmatrix} 0 & 0 & 0 & -0.88 & -1.75 & -0.63 \\ 0 & 0 & 0 & 25.75 & -2.13 & -1.13 \\ 0 & 0 & 0 & -99.5 & 23.5 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad M1^6 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

### Постановка задачі розробки програми синтезу імпульсного регулятора

Розглядається імпульсна система управління по вектора стану. Для деякого класу об'єктів розробляється САПР імпульсних систем управління по вектора стану з кінцевим часом усталення і з спостерігачем, що теж повинен мати кінцевий час усталення. **Необхідно розробити програму моделювання САУ, орієнтовану на розрахунки і дослідження впливу відхилень параметрів об'єкта від номінальних.** Будемо розглядати задачу синтезу на типовому прикладі об'єкта 3-го порядку з одним управлінням. Компоненти вектора стану об'єкта мають таку інтерпретацію:  $x_1$  – координата,  $x_2$  – швидкість,  $x_3$  – прискорення. Дано: рівняння об'єкта і спостерігача у канонічній матрично-векторній формі ( $u$  – управління,  $v$  – збурення):

$$X_o^{(k+1)} = F(T) \cdot X_o^{(k)} + G(T) \cdot B \cdot u^{(k)} + G(T) \cdot B \cdot v^{(k)}; \quad (2.1)$$

$$\begin{aligned} y_n &= C \cdot x_n; \quad y_o = C \cdot x_o; \quad \Delta y = y_n - y_o; \\ X_n^{(k+1)} &= F(T) \cdot X_n^{(k)} + G(T) \cdot B \cdot u^{(k)} + G(T) \cdot L \cdot C \cdot \Delta y^{(k)}. \end{aligned} \quad (2.2)$$

Задано неперервний об'єкт:

$$I := \text{identity}(3); \quad A := \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_0 & -a_1 & -a_2 \end{pmatrix}; \quad B := \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}; \quad C := \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}. \quad (2.3)$$

Записуємо перше і друге наближення для відповідних перехідних функцій  $F(\cdot)$  і  $G(\cdot)$

$$F1(A, T) := I + \frac{A \cdot T}{1!}; \quad F2(A, T) := I + \frac{A \cdot T}{1!} + \frac{A^2 \cdot T^2}{2!}. \quad (2.4)$$

Контроль:  $G1(A, T) := I \cdot T$ ;  $G2(A, T) := I \cdot T + \frac{A \cdot T^2}{2!}$ ;

$$F2(A, T) = \begin{array}{c|ccc} & 0 & 1 & 2 \\ \hline 0 & 1 & 0.8 & 0.32 \\ 1 & -0.96 & 0.36 & 0.48 \\ 2 & -1.44 & -1.92 & -0.12 \end{array}.$$

Контроль (символьним процесором) – на відповідність формул:

$$F1(A, T1) \rightarrow \begin{pmatrix} 1 & T1 & 0 \\ 0 & 1 & T1 \\ -a_0 \cdot T1 & -a_1 \cdot T1 & 1 - a_2 \cdot T1 \end{pmatrix} \quad G1(A, T1) \rightarrow \begin{pmatrix} T1 & 0 & 0 \\ 0 & T1 & 0 \\ 0 & 0 & T1 \end{pmatrix}$$

На рис. 2.1 подана схема САУ, що відповідає рівнянням (2.1) – (2.2).

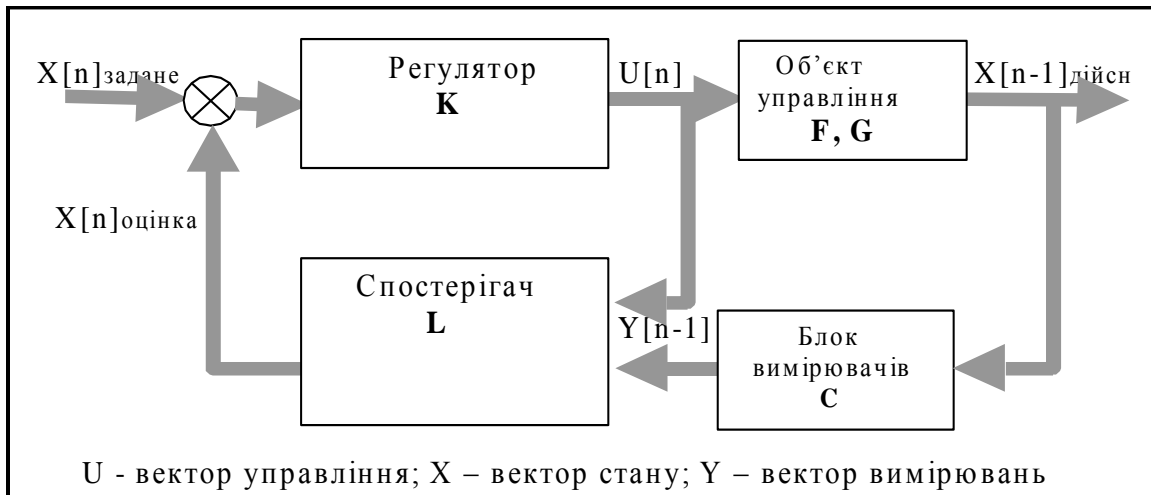


Рис. 2.1. Блок-схема імпульсної системи із спостерігачем стану

### Завдання

1. Побудувати структурну схему САУ зі спостерігачем.....5 б.
2. Розробити програму для розрахунку параметрів "швидкого" регулятора альтернативними методами. Визначити параметри регулятора.....15 б.
3. Розробити програму для розрахунку параметрів "швидкого" спостерігача. Визначити параметри спостерігача.....15 б.
4. Розробити програму моделювання САУ для випадку однакових кроків квантування у контурах регулятора і спостерігача.....15 б.
5. Оцінити чутливість швидких регуляторів до впливу відхилень параметрів об'єкта від розрахункових.....15 б.
6. Знайти параметри регулятора близького до швидкого, але з меншими витратами на управління.....10 б.
7. Зробити аналіз переваг і недоліків управління з кінцевим часом встановлення.....10б.

### Розробка модуля для вибору бажаного розподілення Z-коренів

Нагадаємо, що еталонний процес – це перехідний процес (ПП) для динамічної системи, оптимальний відносно певного критерію – інтегральної квадратичної помилки, витрат енергії на управління, сумарних навантажень, тривалості ПП та ін.

Ми розглянули один з можливих еталонних перехідних процесів - з кінцевим часом усталення. Коефіцієнти характеристичного рівняння такого процесу (крім 1-го) дорівнюють нулю. Можна визначити коефіцієнти для інших, у чомусь оптимальних розподілень.

Узагальнимо метод стандартних коефіцієнтів - зробимо стенд, що дозволяв би знаходити еталонні процеси ad hoc (відповідно до конкретного випадку) і обчислювати для них коефіцієнти характеристичного рівняння.

Для вибору бажаного перехідного процесу необхідно бачити одночасно і перехідний процес, відповідні йому корені Z-характеристичного рівняння і показники якості перехідного процесу. Подаємо корені та перехідний процес на одному графіку. Увага! Необхідні для побудови комплексного графіка залежності подано в електронній версії посібника – вони є сервісними. На рис. 2.2 подано версію стенда, де разом подано графік перехідного процесу для лінійної імпульсної системи третього порядку і розташування коренів Z-характеристичного рівняння цієї системи. Відомо [21, 24,25], що необхідною і достатньою умовою стійкості лінійної імпульсної системи є розташування коренів Z-характеристичного рівняння в середині одиничного кола. Головне призначення стенда (в електронній книзі) – розуміння зв'язку між розташуванням коренів і перехідним процесом. Стенд дозволяє експериментувати – змінювати корені і бачити наслідки.



Рис.2.2. Стенд для вибору еталонного перехідного процесу

Виводимо значення коефіцієнтів еталонного характеристичного рівняння:

$$E0(im, re, z3) = 0.16 ; E1(im, re, z3) = 0.09 ; E2(im, re, z3) = -0.4 .$$

Тепер, коли вибрано значення коренів, які забезпечують потрібну якість ПП, вводимо нові позначення для коефіцієнтів Z-характеристичного рівняння (це тепер константи).

$$Vtr1 := E0(im, re, z3) ; Vtr2 := E1(im, re, z3) ; Vtr3 := E2(im, re, z3).$$

Приклад контролю в процесі розробки програмних модулів. Ніколи невідомо, де і як виникне помилка в програмі розрахунку САУ, тому перевіряємо:

- характеристичне рівняння дійсно = 0 при підстановці кореня;
- корені отриманого еталонного характеристичного рівняння дорівнюють заданим.

$$1 \cdot z^3 + E2(im, re, z3) \cdot z^2 + E1(im, re, z3) \cdot z + E0(im, re, z3) = 0 ;$$

$$\text{poly roots} \left( \begin{pmatrix} E0(im, re, z3) \\ E1(im, re, z3) \\ E2(im, re, z3) \\ 1 \end{pmatrix} \right) = \begin{matrix} & & 1 \\ 1 & & -0.4 \\ 2 & & 0.4-0.5i \\ 3 & & 0.4+0.5i \end{matrix} ; \begin{matrix} (z3) \\ z2 \\ z1 \end{matrix} = \begin{matrix} & & 1 \\ 1 & & -0.4 \\ 2 & & 0.4+0.5i \\ 3 & & 0.4-0.5i \end{matrix} .$$

### Розробка програми розрахунку параметрів "швидкого" регулятора

**Визначення (дефініція) параметрів регулятора.** Очевидний шлях розрахунку параметрів регулятора такий:

- отримати характеристичне рівняння для замкненої системи у загальному вигляді (задані параметри об'єкта та невідомі параметри регулятора входять як символічні змінні);
- прирівняти оці загальні вирази при коефіцієнтах характеристичного рівняння їх потрібним значенням. Ці потрібні значення, згідно з теорією УКЧУ, повинні бути усі =0;
- розв'язати якимось методом систему рівнянь (вона може бути і нелінійною) - отримуємо шукані значення параметрів регулятора.

**Випишемо вхідні дані варіанта завдання:**

**Варіант 11.**  $a2 := 2$ ;  $a1 := 2$ ;  $a0 := 1$ ;  $Тр := 2$ ;  $hm := 0.1$ ;  $Т := 0.8$  .  
Записуємо матриці А, В, С – згідно з (3) і F G - згідно з (4), беремо перші наближення.

$$A := \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_0 & -a_1 & -a_2 \end{pmatrix}; B := \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}; C := \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}; k := \begin{pmatrix} k_1 \\ k_2 \\ k_3 \end{pmatrix}.$$

Контроль:

$$F1(A, T) = \begin{array}{|c|c|c|c|} \hline & 1 & 2 & 3 \\ \hline 1 & 1 & 0.8 & 0 \\ \hline 2 & 0 & 1 & 0.8 \\ \hline 3 & -0.8 & -1.6 & -0.6 \\ \hline \end{array}$$

$$G1(A, T) = \begin{array}{|c|c|c|c|} \hline & 1 & 2 & 3 \\ \hline 1 & 0.8 & 0 & 0 \\ \hline 2 & 0 & 0.8 & 0 \\ \hline 3 & 0 & 0 & 0.8 \\ \hline \end{array}$$

Записуємо характеристичне рівняння у двох альтернативних формах (перша форма коректна тільки для першого наближення!)

$$\left| I \cdot z - \left[ I + (A + B \cdot k^T) \cdot T \right] \right| = 0; \quad \left| I \cdot z - \left( F1(A, T) + G1(A, T) \cdot B \cdot k^T \right) \right| = 0. \quad (2.5)$$

Контроль (там, за полями сторінки задано умовні позначення A, B, k, z)

$$\left[ I \cdot z - \left[ I + (A + B \cdot k^T) \cdot T \right] \right] - \left[ I \cdot z - \left( F1(A, T) + G1(A, T) \cdot B \cdot k^T \right) \right] = \begin{array}{|c|c|c|c|} \hline & 1 & 2 & 3 \\ \hline 1 & 0 & 0 & 0 \\ \hline 2 & 0 & 0 & 0 \\ \hline 3 & 0 & 0 & 0 \\ \hline \end{array}.$$

Можна записати вираз для характеристичного рівняння (2.5) у ще більш компактно-му вигляді:

$$\left| (I \cdot z - M_p) \right| = 0,$$

$$\text{де } M_p(k, T) := F1(A, T) + G1(A, T) \cdot B \cdot k^T. \quad (2.6)$$

Для об'єкта високого порядку розкрити в символьному вигляді детермінант (2.5) – важка задача. Символьний процесор видасть формули неосяжних розмірів. Можливі альтернативні шляхи розв'язання задачі синтезу, що не потребують розкриття детермінанта.

Вибираємо орієнтований на використання сучасних пакетів метод [12, 14, 15] "неявні вирази для коефіцієнтів характеристичного рівняння", при якому *не треба розкривати детермінант*. Визначимо такі функції користувача:

$$I := \text{identit}(3) \quad \text{determinan}(k, T, z) := \left| I \cdot z - M_p(k, T) \right| \\ \text{polinon}(V2, V1, V0, z) := (z^3 + V2 \cdot z^2 + V1 \cdot z + V0). \quad (2.7)$$

Тепер задаємо **потрібні** значення коефіцієнтів V2, V1, V0 характеристичного рівняння. Наприклад, ми вибрали "швидкий" регулятор. **Нагадаємо**: перехідний процес закінчується за N кроків (в даному випадку N=3), якщо усі корені характеристичного рівняння  $Z_i = 0$ ;  $i = 1.. N$ . Це означає, що всі коефіцієнти характеристичного рівняння, крім першого, повинні бути нульовими  $V_i = 0$ ;  $i = 0.. N-1$ .

$$\text{Задаємо потрібні значення коефіцієнтів } V: \quad V_0 := 0; \quad V_1 := 0; \quad V_2 := 0,$$

$$\text{початкові значення шуканих параметрів } K_1 := 2; K_2 := 2; K_3 := -99,$$

деякі значення  $z_1 := 1; z_2 := 0; z_3 := -4$ . Записуємо (в позначеннях (2.7)) систему рівнянь для коефіцієнтів і розв'язуємо її вбудованими числовими методами.

Підказка за полями електронної книги =>

Given (= дано)

$$\text{polinon}(V2, V1, V0, z1) = \text{determinan}(K, T, z1)$$

$$\text{polinon}(V2, V1, V0, z2) = \text{determinan}(K, T, z2)$$

$$\text{polinon}(V2, V1, V0, z3) = \text{determinan}(K, T, z3)$$

(2.8)

$$(\text{=що знайти}) \quad K_{si}(T) := \text{Find}(K).$$

Виводимо результат синтезу:



$$\begin{pmatrix} ks1 \\ ks2 \\ ks3 \end{pmatrix} := Ksi(T); T = 0.8; \begin{pmatrix} ks1 \\ ks2 \\ ks3 \end{pmatrix} = \begin{array}{|c|c|} \hline & 1 \\ \hline 1 & -0.95 \\ \hline 2 & -2.69 \\ \hline 3 & -1.75 \\ \hline \end{array}; Ksi(0.5) = \begin{array}{|c|c|} \hline & 1 \\ \hline 1 & -7 \\ \hline 2 & -10 \\ \hline 3 & -4 \\ \hline \end{array}.$$

**Тест №1.** Напишіть словесну формулу рівнянь (2.8). Покажіть, що ці рівняння еквівалентні умові "усі коефіцієнти характеристичного рівняння системи з регулятором дорівнюють коефіцієнтам еталонного характеристичного рівняння". Поясніть, чому можна у (2.8) підставляти досить довільні значення а) K1, K2, K3, б) z1, z2, z3.

**Тест №2.** Дайте інтерпретацію виразу (2.8) в термінах програмування.

**Тест №3.** Вирази (2.7) і (2.8) – "чорні ящики". Як за мінімальний час (на комп'ютері) визначити залежність параметрів регулятора k1, k2, k3 від кроку квантування T?

### Розробка програми для розрахунку параметрів "швидкого" спостерігача

**Визначення параметрів спостерігача.** Порядок синтезу спостерігача такий же, як і для синтезу регулятора. Скопіюємо і відповідні блоки попереднього розділу, скоротимо коментарі, виконаємо необхідні модифікації.

Запишемо задану матрицю вимірювання C та матрицю L шуканих параметрів спостерігача. Увага! За межами сторінки (електронної книги) приховано те, що дозволяє формулам (2.9–2.11) знати структуру L – матриці 3×3 з трьома ненульовими елементами. Пам'ятаємо, що під усіма об'єктами приховано модулі, написані на C++.

$$C := \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}; L := \begin{pmatrix} ls1 & 0 & 0 \\ ls2 & 0 & 0 \\ ls3 & 0 & 0 \end{pmatrix}; L \cdot C \rightarrow \begin{pmatrix} ls1 & 0 & 0 \\ ls2 & 0 & 0 \\ ls3 & 0 & 0 \end{pmatrix}$$

$$\boxed{|I \cdot z - [I + (A + L \cdot C) \cdot dT]| = 0} \quad (2.9)$$

Тут закрито допомогу: порядок отримання характеристичного рівняння спостерігача. Записуємо характеристичне рівняння спостерігача вибраної структури L

$$\boxed{|(I \cdot z - Nn(l1, l2, l3, T))| = 0} \quad (2.10)$$

$$\text{де } Nn(l1, l2, l3, T) := F1(A, T) + G1(A, T) \cdot \begin{pmatrix} 11 & 0 & 0 \\ 12 & 0 & 0 \\ 13 & 0 & 0 \end{pmatrix} \cdot C, \quad I := \text{identity}(3).$$

Визначимо такі функції користувача:

$$\text{determ}(l1, l2, l3, T, z) := |I \cdot z - Nn(l1, l2, l3, T)|$$

$$\text{polind}(V2, V1, V0, z) := (z^3 + V2 \cdot z^2 + V1 \cdot z + V0). \quad (2.11)$$

Вибираємо, для прикладу "швидкий" спостерігач. Задаємо **потрібні** значення коефіцієнтів V2, V1, V0 характеристичного рівняння. **Нагадаємо:** перехідний процес закінчується за N кроків (в даному випадку N = 3), якщо всі корені характеристичного рівняння Zi = 0; i = 1.. N. Це означає, що всі коефіцієнти характеристичного рівняння, крім першого, повинні бути нульовими Vi = 0; i = 0.. N-1.

Задаємо **потрібні значення** коефіцієнтів V: **V0 := 0; V1 := 0; V2 := 0,**

**початкові значення** шуканих параметрів l1 := 2; l2 := 2; l3 := -101.11, (<=чому?)

**деякі значення** z1 := -3; z2 := 3; z3 := 0. Записуємо (в позначеннях (2.7)) систему рівнянь для коефіцієнтів і розв'язуємо її вбудованими числовими методами.

$$\text{Given (= дано)} \quad \begin{array}{l} \text{polind}(V2, V1, V0, z1) = \text{determ}(l1, l2, l3, T, z1) \\ \text{polind}(V2, V1, V0, z2) = \text{determ}(l1, l2, l3, T, z2) \\ \text{polind}(V2, V1, V0, z3) = \text{determ}(l1, l2, l3, T, z3) \end{array} \quad (2.12)$$

$Lsi(T) := \text{Find}(11, 12, 13)$  (=що знайти). Виводимо результат синтезу:

$$\begin{pmatrix} ls1 \\ ls2 \\ ls3 \end{pmatrix} := Lsi(T); T = 0.8 ; \begin{pmatrix} ls1 \\ ls2 \\ ls3 \end{pmatrix} = \begin{pmatrix} & 1 \\ 1 & -1.75 \\ 2 & 0.81 \\ 3 & 0.92 \end{pmatrix} ; Lsi(0.77) = \begin{pmatrix} & 1 \\ 1 & -1.9 \\ 2 & 0.73 \\ 3 & 1.14 \end{pmatrix} .$$

А тепер подивимось, як залежать параметри швидкого регулятора і швидкого спостерігача від заданого кроку квантування:  $T_v := 0.1, 0.175.. 2.1$ .

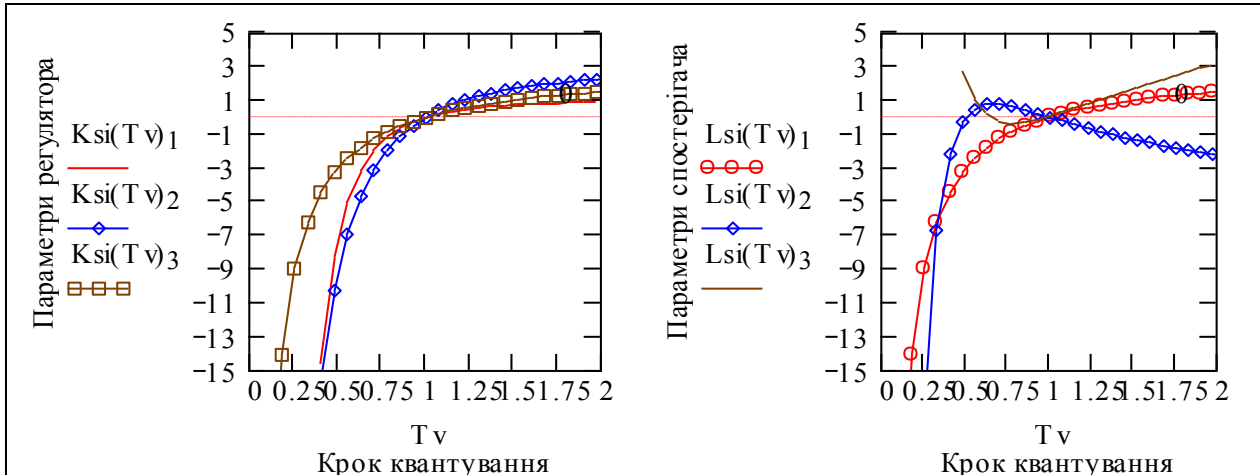


Рис. 2.3. Залежність параметрів "швидкого" регулятора і спостерігача від заданого кроку квантування

Прокоментуємо графіки на рис. 2.3. Для кроків квантування  $T < 0.4$  с значно збільшуються модулі параметрів регулятора і спостерігача. Для  $T > 1$  с значення параметрів стабілізуються. Це природно – чим менший крок – тим менший час перехідного процесу, тим більші зусилля необхідні для приведення об'єкта управління в потрібний стан. Це, поряд з іншими вимогами, повинно враховуватись *при виборі кроку квантування*.

**Завдання** (перевірте свій інтелект в ТАУ):

- знайдіть такі параметри об'єкта (1331), при яких, для деякого кроку квантування, параметри швидкого регулятора  $k_1, k_2, k_3$  (спостерігача) будуть нульовими.....5 б.;
- визначіть – добре це, чи ні, дайте інтерпретацію такому управлінню.....25 б.

### Розробка програми моделювання САУ зі спостерігачем

Використовуємо результати попередніх пунктів завдання, формуємо з ММ основного контуру (ОК) та контуру спостерігача (КС) ММ повної системи. Для цього дивимось на структурну схему, читаємо коментар до неї і записуємо рівняння ОК (основного контуру) та КС (контуру спостерігача):  $\text{ORIGIN} := 1$

$$X^{(n+1)} = F \cdot X^{(n)} + G \cdot K \cdot x^{(n)} ; \quad x^{(n+1)} = F \cdot x^{(n)} + G \cdot K \cdot x^{(n)} + L \cdot C \cdot (x^{(n)} - X^{(n)}) .$$

Замість оцінки координати  $x$  вводимо нову змінну  $\Delta x = x - X$  – помилку оцінювання, підставляємо у  $K$  та  $L$  результати синтезу.

$$X^{(n+1)} = F \cdot X^{(n)} + G \cdot K \cdot X^{(n)} + G \cdot K \cdot x^{(n)} ; \quad \Delta x^{(n+1)} = F \cdot \Delta x^{(n)} + L \cdot C \cdot \Delta x^{(n)} . \quad (2.13)$$

Вводимо **розширений вектор стану X** (подано нижче) та перехідну матрицю повної системи  $Mps$  (=Матрицю повної системи). Для формування блочної матриці повної системи використовуємо вбудовані функції пакета `augment ()` та `stack ()`:

$$oo := \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} ; L := \begin{pmatrix} ls1 & 0 & 0 \\ ls2 & 0 & 0 \\ ls3 & 0 & 0 \end{pmatrix} ; K := (ks1 \quad ks2 \quad ks3) ; Mps = \begin{pmatrix} F + G \cdot K & G \cdot B \cdot K \\ oo & F + L \cdot C \end{pmatrix} ;$$

$$\begin{aligned}
\text{Верх} &:= \text{augment}[(F1(A,T) + G1(A,T) \cdot B \cdot K), G1(A,T) \cdot B \cdot K]; \\
\text{Низ} &:= \text{augment}[00, (F1(A,T) + G1(A,T) \cdot L \cdot C)]; \\
T &= 0.8; \quad \text{Msp} := \text{stack}(\text{Верх}, \text{Низ}).
\end{aligned}
\tag{2.14}$$

Виводимо перехідну матрицю стану для повної системи:

основний контур + контур спостерігача

$$\text{Msp} = \begin{array}{c|cccccc}
 & 1 & 2 & 3 & 4 & 5 & 6 \\
\hline
1 & 1 & 0.8 & 0 & 0 & 0 & 0 \\
2 & 0 & 1 & 0.8 & 0 & 0 & 0 \\
3 & -1.56 & -3.75 & -2 & -0.76 & -2.15 & -1.4 \\
4 & 0 & 0 & 0 & -0.4 & 0.8 & 0 \\
5 & 0 & 0 & 0 & 0.65 & 1 & 0.8 \\
6 & 0 & 0 & 0 & -0.06 & -1.6 & -0.6
\end{array}$$

Задаємо вектор початкових умов, кількість кроків, крок

$$X = \begin{pmatrix} X1 \\ X2 \\ X3 \\ \Delta x1 \\ \Delta x2 \\ \Delta x3 \end{pmatrix} \begin{pmatrix} \text{"-вектор"} \\ \text{"стану"} \\ \text{"системи"} \\ \text{"-помилки"} \\ \text{"оцінювання"} \\ \text{"вектора стану"} \end{pmatrix}; \quad X^{(1)} := \begin{pmatrix} -1 \\ 0 \\ 0 \\ 0.4 \\ 0.0 \\ 0.0 \end{pmatrix}; \quad n := 1..8; \quad T = 0.8.$$

Нарешті запишемо різницеве рівняння 6-го порядку, за яким і обчислюється ПП.

$$X^{(n+1)} := \text{Msp} X^{(n)}. \tag{2.15}$$

Знову повертаємось від помилок оцінювання вектора стану до оцінок:

$$x_{1,n} := X_{1,n} + X_{4,n}; \quad x_{2,n} := X_{2,n} + X_{5,n}; \quad x_{3,n} := X_{3,n} + X_{6,n}$$

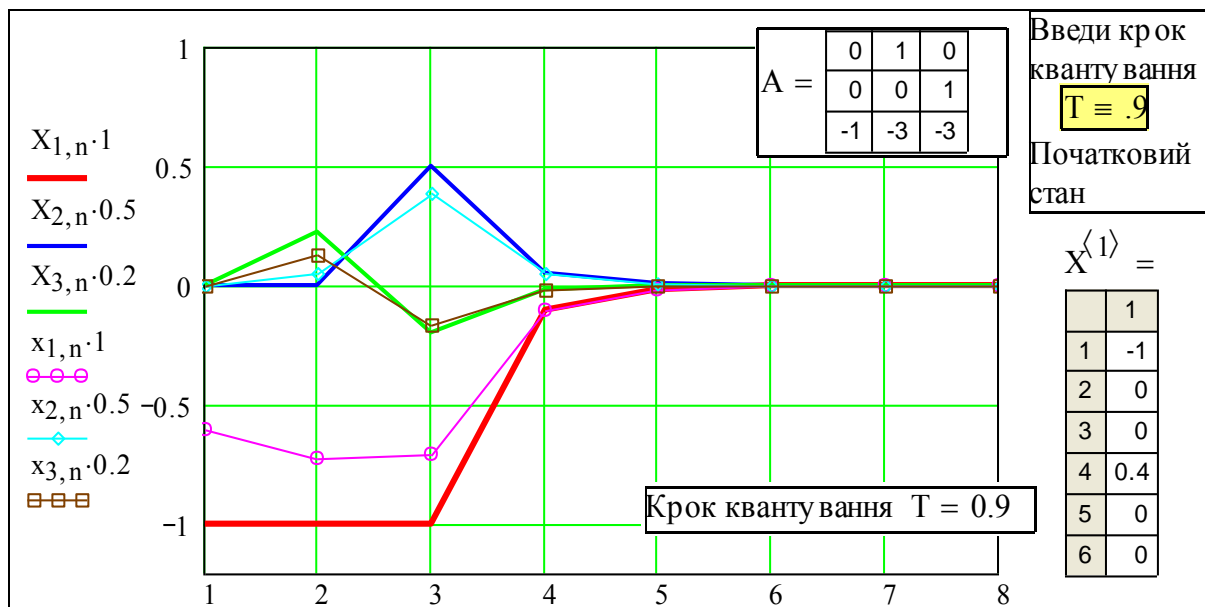


Рис. 2.4. Стенд. Перехідний процес в імпульсній САУ з спостерігачем

Далі подано декілька копій екрана з різними перехідними процесами. Це перший крок в дослідженні системи. Наступний крок – побудова пошукових програм, ядром яких будуть рівняння (2.13)–(2.15). Ці програми будуть шукати найбільш точні, економні, завадостійкі регулятори та ін. Розробіть таку програму.

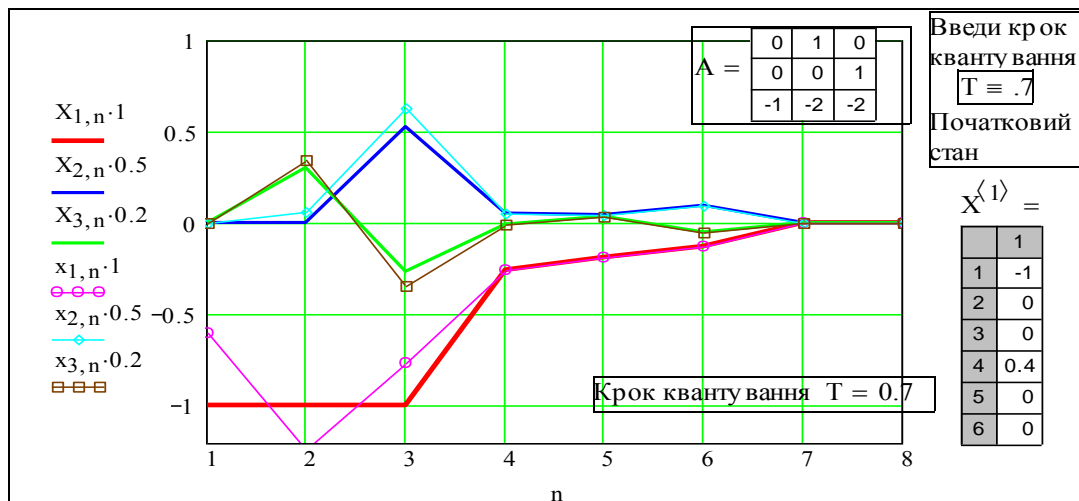


Рис. 2.5. Стенд. Перехідний процес в імпульсній САУ з спостерігачем

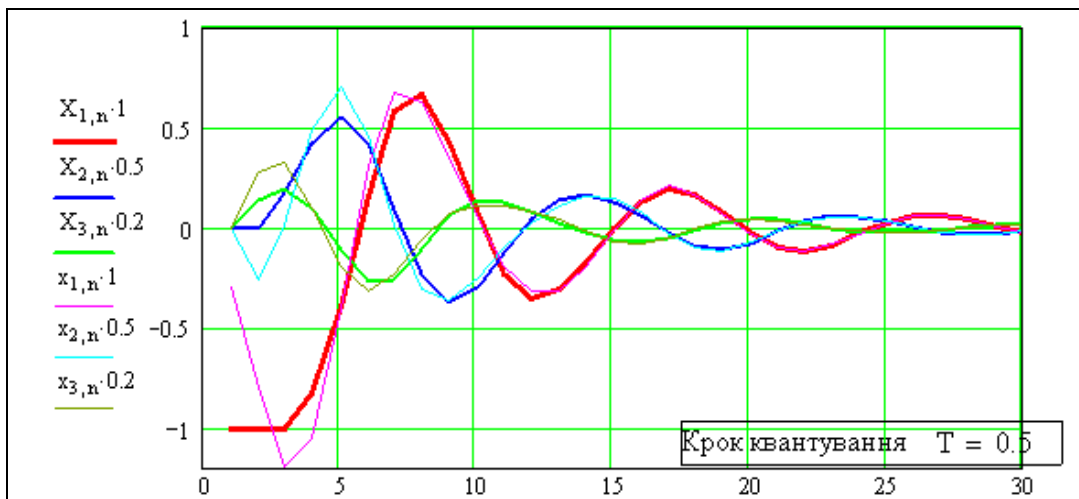


Рис. 2.6. Тестовий приклад: коливальний основний контур + коливальний спостерігач

### Висновки

Цікаво, що система, параметри якої розраховувались за довгим ланцюгом досить неочевидних залежностей, дійсно працює – все закінчується через 6 кроків, при будь-яких початкових умовах – тобто: розрахунки коректні – перехідні процеси задовільні. Спостерігач, згідно з теорією, на третьому кроці дає вже безпомилкові оцінки. Для такого дива необхідно лише дві умови: а) об'єкт дійсно лінійний, б) параметри об'єкта визначені точно.

#### Переваги і недоліки управління з кінцевим часом усталення:

- + легко розв'язується задача синтезу параметрів, тому легко будувати САУ з самонастроюванням на базі саме такого регулятора;
- + незалежно від помилок початкового стану рівно через  $N$  кроків помилки всіх компонентів вектора стану стають нульовими – це зручно для певних дискретних систем;
- система чутлива до відхилень помилок оцінювання і вимірювання координат та параметрів об'єкта від розрахункових – її бажано зробити з настроюванням;
- система задовільно працює тільки для стійких об'єктів управління.

#### Контрольні запитання

1. Запишіть канонічну систему рівнянь неперервної САУ з неперервним спостерігачем.
2. Запишіть канонічну систему рівнянь імпульсної САУ з імпульсним спостерігачем.
3. Схема синтезу імпульсної САУ з спостерігачем стану.
4. Умови стійкості лінійних імпульсних САУ.
5. Необхідні умови для коректної роботи спостерігача.
6. Що таке "швидкий" регулятор?
7. Властивості швидкого регулятора.

## 2.2 Синтез та дослідження стійкої до відмов вимірювачів імпульсної САУ зі спостерігачами

### Розробка програмних модулів для синтезу та моделювання стійкої до відмов вимірювачів САУ

Відомо, що існує імовірнісна "надійність", засмічена схоластичними "науковими" роботами і надійність реальна, яку, щоб відрізнити, називають гарантоспроможність, живучість, відмовостійкість, відмовонечутливість, "плавна деградація".

Особливість концепцій живучості в тому, що не розглядаються ймовірності подій (відмов), а увага зосереджується на пошуку конструкторських рішень, що забезпечують функціонування системи і при відмовах елементів. Термін "плавна деградація" означає, що якість функціонування системи при відмові певного елемента повинна зменшуватись пропорційно відносній вартості елемента.

В цьому документі дається приклад розробки програми для дослідження проектного рішення, що забезпечує відмовонечутливість і відмовостійкість САУ при відмовах вимірювачів. Проведемо порівняльний аналіз двох альтернатив побудови САУ. На рис. 2.7 подана схема САУ з регулятором вектора стану та резервованими вимірювачами компонентів вектора стану  $X_1$ ,  $X_2$ ,  $X_3$ . Для прикладу взято типове трикратне ідентичне резервування вимірювачів. Система повністю виходить з ладу при одночасній відмові трьох вимірювачів якогось з компонентів вектора стану. В реальних технічних системах одночасний вихід ідентичних резервованих елементів – досить ймовірна подія:

а) елементи можуть бути з одної бракованої партії;

б) елементи можуть бути компактно розташовані, щоб при "згорянні" одного могли б згоріти інші;

г) на елементи може одночасно подіяти одне й те ж збурення – радіація, перевантаження, зміна температури.

Таким чином, при виході тільки трьох з дев'яти вимірювачів система стає непридатною. Чи можна зробити деградацію системи більш плавною? При розумних витратах матеріальних ресурсів і штучного інтелекту (природний є необмеженим ресурсом)? – Можна:

На рис. 2.8 подана схема САУ з регулятором вектора стану (vector state controller) і системою спостерігачів стану (vector state observer). Кожен з цих спостерігачів розрахований для певної ситуації вимірювання (що задається матрицею вимірювань  $C$ ). Кожен спостерігач видає оцінку всього вектора стану. Таким чином, система буде працездатною, якщо деякі компоненти вектора стану взагалі не вимірюються (звичайно, за умови, що САУ в цій конфігурації є спостережуваною [21, 24,25, 39,40]).

Призначення блоків "Логіка", "Аналіз, логіка" в схемах на рис. 2.7, 2.8 – ідентифікація та ізоляція некоректно працюючого елемента. Реалізація САУ за альтернативою 2 не вимагає великих додаткових витрат – спостерігачі реалізуються на дешевих мікроконтролерах.

### Завдання

1. Скласти специфікацію програмних модулів (входи, виходи).
2. Скласти схему викликів підпрограм.
3. Виконати дослідження впливу відмов (згідно з індивідуальним завданням).
4. Дослідити вплив повних відмов вимірювачів.

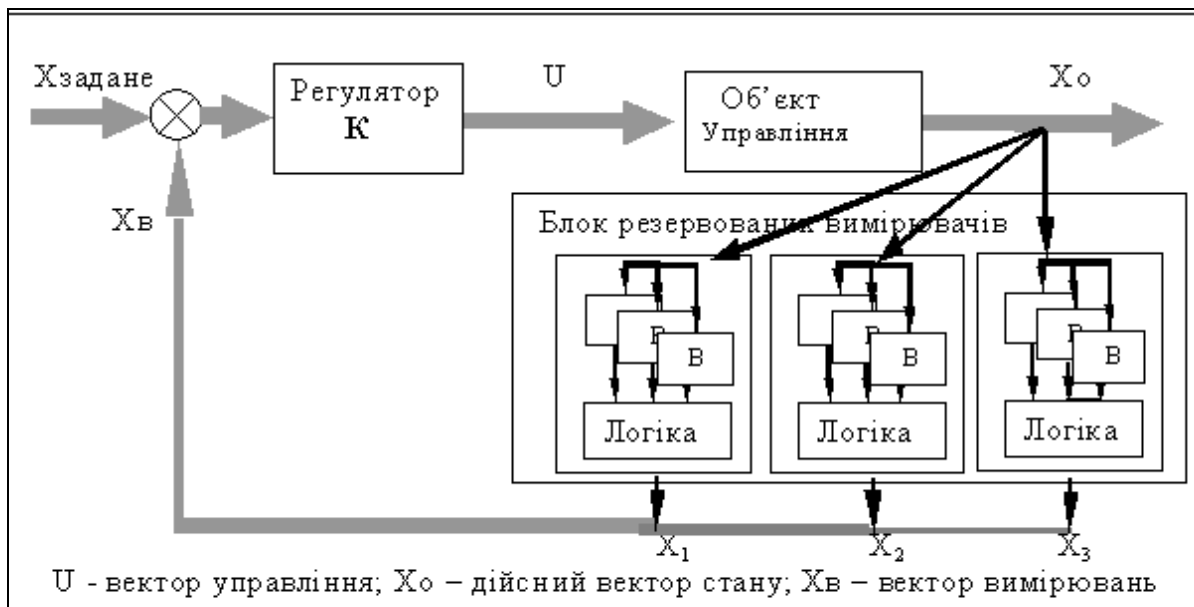


Рис. 2.7. Альтернатива 1 побудови відмовостійкої САУ. Резервовані вимірювачі

Дивимось одразу на обидві схеми (розвиває це стереобачення проблем). Показником набуття його (стереобачення) буде спроможність винайти ще хоч одну альтернативу побудови відмовостійкої системи.

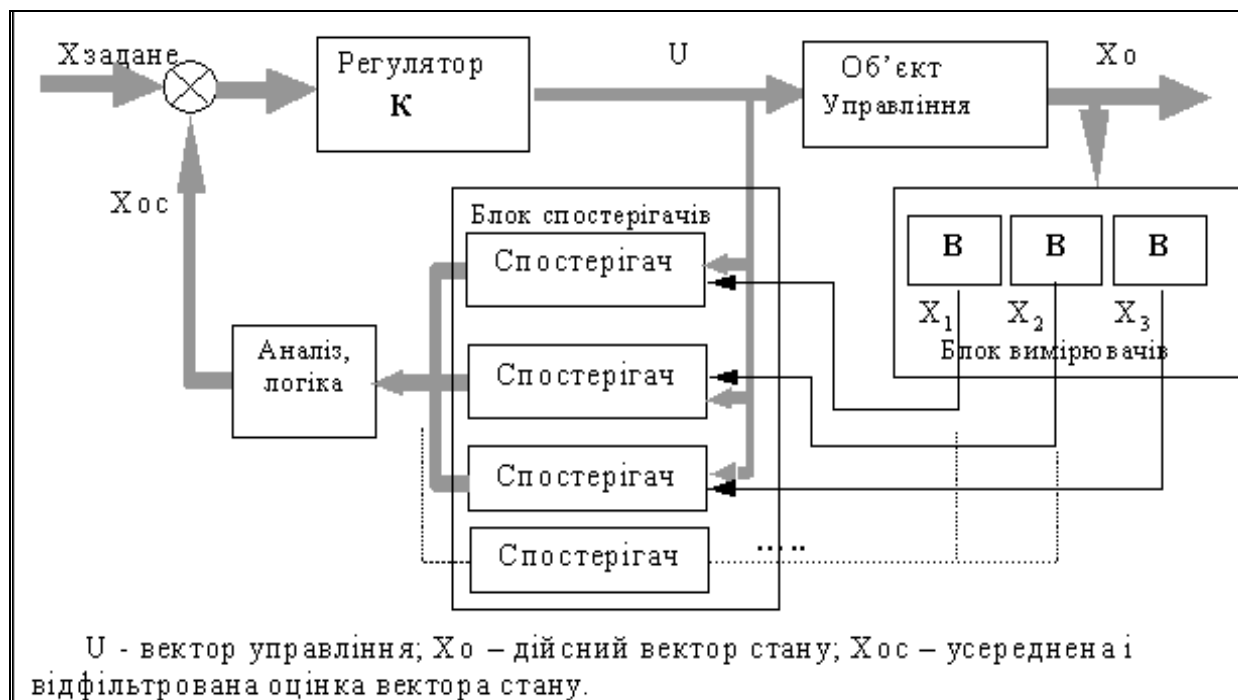


Рис. 2.8. Альтернатива 2 побудови відмовостійкої САУ. Нерезервовані вимірювачі та система спостерігачів

Деталізуємо опис системи, поданої на рис. 2.8. Розглядається така альтернатива побудови системи спостерігачів (не краща!): на кожен з трьох спостерігачів подається тільки один сигнал X<sub>1</sub>, X<sub>2</sub>, X<sub>3</sub>. Цьому відповідають такі матриці вимірювань C:

$$C_1 := \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}; \quad C_2 := \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}; \quad C_3 := \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Кожен із спостерігачів видає оцінку всього вектора стану, тобто значення  $X_1, X_2, X_3$ . При вибраній структурі матриці об'єкта система у всіх випадках є спостережувана. Всі ці варіанти дають коректні оцінки вектора стану. Для управління можна використовувати який завгодно з цих виходів, або середнє.

При відмові одного з вимірювачів його вихід буде значно відрізнятись від інших. Блок "Логіка, аналіз" повинен ідентифікувати некоректний канал та ігнорувати його вихід. Дещо складніша ситуація при відмові двох вимірювачів. Але ідентифікація некоректних каналів принципово можлива. Для виконання завдання потрібно конкретизувати і формалізувати відмови вимірювачів і умови випробувань системи.

**Часткова відмова** – помилка вимірювання з номінальних 0–2% (наприклад) збільшується до 10–30%. Конкретно відмова може проявлятися як: адитивний шум, нечутливість, обмеження вихідного сигналу, збільшення величини квантування рівня і часу.

**Повна відмова** – вихідний сигнал стає нульовим. Цей випадок відповідає стандарту відмовобезпечності – якщо технічна підсистема виходить з ладу, то вона не повинна стати джерелом відмов інших – не викликати пожежі, не створювати коротких замикань, не видавати помилкові сигнали.

**Тестові сигнали** – розглядаємо роботу системи в режимі відпрацювання командних сигналів  $X_{\text{задане}}$  (рис. 2.9) виду: а) послідовність імпульсів, б) гармонічний сигнал, в) шум.

Подаємо приклади тестових сигналів і графіки "вхід-вихід" для імпульсних сигналів:  
 $u_1(q, P) := \text{if}(\text{mod}(q, P) = 0, 1, 0)$ ;  $u_2(q, \omega) := \sin(\omega \cdot q)$ ;  $u_3(q) := (\text{rnd}(1) - 0.5)$ .

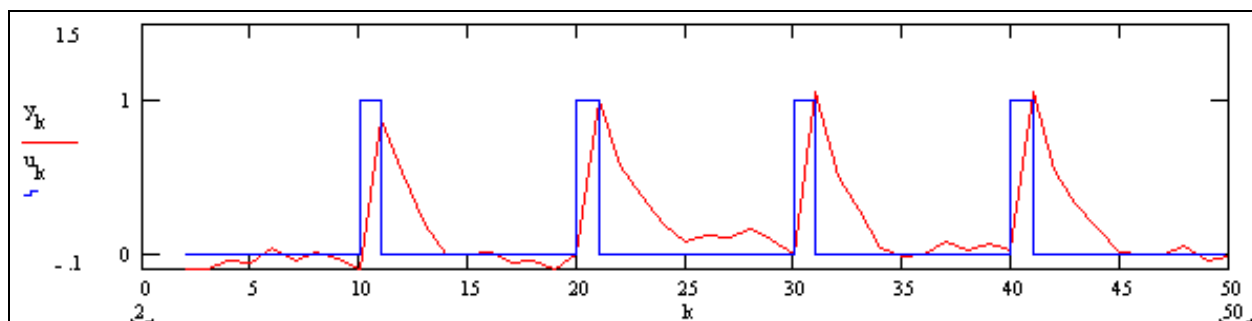


Рис. 2.9. Тестові сигнали для САУ

### Завдання для самостійного виконання

1. Розробка на базі поданих програмних модулів програми моделювання САУ при відмовах вимірювачів
2. Виконання дослідження функціонування САУ при відмовах (згідно із заданим варіантом):
  - 2.1 дослідження САУ при частковій відмові вимірювача  $X_1$ ;
  - 2.2 дослідження САУ при частковій відмові вимірювача  $X_2$ ;
  - 2.3 дослідження САУ при частковій відмові вимірювача  $X_3$ ;
  - 2.4 дослідження САУ при послідовній відмові вимірювачів  $X_1, X_3$ ;
  - 2.5 дослідження САУ при послідовній відмові вимірювачів  $X_1, X_2$ ;
  - 2.6 дослідження САУ при послідовній відмові вимірювачів  $X_2, X_3$ ;
  - 2.7 дослідження САУ при одночасній відмові вимірювачів  $X_1, X_3$ ;
  - 2.8 дослідження САУ при одночасній відмові вимірювачів  $X_1, X_2$ ;
  - 2.9 дослідження САУ при одночасній відмові вимірювачів  $X_2, X_3$ .

Базові програми зроблені для об'єкта 3-го порядку з одним входом, одним виходом. Потенційно ці програми можуть бути масштабовані – модифіковані для довільних об'єктів вищих порядків. Іноді це легко зробити, іноді – неможливо.

### Модуль синтезу швидкого регулятора

Увага!! Цей модуль розраховано тільки для об'єктів 3-го порядку з одним входом, одним виходом, тільки з трипараметричною матрицею. Майже неважко узагальнити його. Записуємо вхідні дані для синтезу:

$$A := \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_0 & -a_1 & -a_2 \end{pmatrix}; \quad B := \begin{pmatrix} 0 \\ 0 \\ b \end{pmatrix}; \quad k := \begin{pmatrix} k_1 \\ k_2 \\ k_3 \end{pmatrix}; \quad B \cdot k^T \rightarrow \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ b \cdot k_1 & b \cdot k_2 & b \cdot k_3 \end{pmatrix}; \quad b := 1.$$

Знаходимо перехідну матрицю основного контуру  $I := \text{identity}(3)$      $\text{ORIGIN} := 1$

$$I \cdot z - [I + (A + B \cdot k^T) \cdot dT] \rightarrow \begin{bmatrix} z-1 & -dT & 0 \\ 0 & z-1 & -dT \\ -(-a_0 + b \cdot k_1) \cdot dT & -(-a_1 + b \cdot k_2) \cdot dT & z-1 - (-a_2 + b \cdot k_3) \cdot dT \end{bmatrix}.$$

Визначаємо детермінант цієї матриці (тут  $T$  – крок квантування) – це характеристичне рівняння:

$$\text{Charpavn} = z^3 - 3 \cdot z^2 + z^2 \cdot a_2 \cdot T - z^2 \cdot T \cdot k_3 + 3 \cdot z - 2 \cdot z \cdot a_2 \cdot T + 2 \cdot z \cdot T \cdot k_3 \dots \\ + z \cdot T^2 \cdot a_1 - z \cdot T^2 \cdot k_2 - 1 + a_2 \cdot T - T \cdot k_3 - T^2 \cdot a_1 + T^2 \cdot k_2 + T^3 \cdot a_0 - k_1 \cdot T^3.$$

Збираємо коефіцієнти при однакових степенях  $z$ . Згідно з теорією швидких регуляторів коефіцієнти характеристичного рівняння повинні бути нульовими [12, 16, 18, 20-27, 29, 32, 33, 39-40]. Прирівнюємо вирази для коефіцієнтів характеристичного рівняння до нуля. Отримуємо систему трьох рівнянь, розв'язуємо її у символьному вигляді.

$$\left. \begin{aligned} \text{Give}(aa_0 - kk_1) \cdot Tt^3 + (-aa_1 + kk_2) \cdot Tt^2 + (aa_2 - kk_3) \cdot Tt - 1 &= 0; \\ (aa_1 - kk_2) \cdot Tt^2 + (-2 \cdot aa_2 + 2 \cdot kk_3) \cdot Tt + 3 &= 0; \\ (-kk_3 + aa_2) \cdot Tt - 3 &= 0. \end{aligned} \right\}$$

Отримаємо розв'язання в символьній формі. Нагадаємо тим, хто знає тільки "книжну" математику: в математичному пакеті одну і ту ж змінну доводиться визначати по-різному: для звичайних обчислень – з присвоєним числовим значенням, для символьних обчислень – без значення. Між іншим, тут подано два об'єкта, це й – текстовий, і математичний вираз →

$$\text{Find}(kk_1, kk_2, kk_3) \rightarrow \begin{array}{l} \frac{(Tt^3 \cdot aa_0 - 1)}{Tt^3} \\ \frac{(Tt^2 \cdot aa_1 - 3)}{Tt^2} \\ \frac{(Tt \cdot aa_2 - 3)}{Tt} \end{array}$$

$$\text{Kr}(Av, T) := \begin{array}{l} K_1 \leftarrow Av_1 - \frac{1}{T^3} \\ K_2 \leftarrow Av_2 - \frac{3}{T^2} \\ K_3 \leftarrow Av_3 - \frac{3}{T} \\ K \end{array}$$

Визначаємо функцію користувача "синтез регулятора", що бере вектор параметрів об'єкта  $Av$  і крок квантування  $T$  та повертає вектор параметрів регулятора  $\text{Kr}(\cdot)$ . Виконаємо тестування, виведемо два розв'язки.

$$\text{Kr} \left[ \begin{pmatrix} -1 \\ -2 \\ -3 \end{pmatrix}, .4 \right] = \begin{array}{|c|c|} \hline & 1 \\ \hline 1 & -16.62 \\ \hline 2 & -20.75 \\ \hline 3 & -10.5 \\ \hline \end{array}$$

$$\text{Kr} \left[ \begin{pmatrix} -1 \\ -2 \\ -3 \end{pmatrix}, 4 \right] = \begin{array}{|c|c|} \hline & 1 \\ \hline 1 & -1.02 \\ \hline 2 & -2.19 \\ \hline 3 & -3.75 \\ \hline \end{array}$$



### Модуль "регулятор"

В першому наближенні це дуже проста програма, потім в розділах 3 і 4 будуть подані більш узагальнені, складні, нелінійні закони управління.

$U(K, X) := (K_1 \ K_2 \ K_3) \cdot \begin{pmatrix} X_1 \\ X_2 \\ X_3 \end{pmatrix}$	Тестуємо цей модуль автономно і разом з модулем "синтез регулятора":
$A1 := \begin{pmatrix} 1 \\ -2 \\ -3 \end{pmatrix} \quad X1 := \begin{pmatrix} -3 \\ -2 \\ -3 \end{pmatrix} \quad T := .4$	$k1 := \text{Kr}(A1, T) \quad k1 := \begin{pmatrix} 3 \\ 2 \\ 1 \end{pmatrix}$
$U(k1, X1) = 116.87$	
$U(\text{Kr}(A1, T), X1) = 116.87$	

### Модуль синтезу швидкого спостерігача

Увага!! Модуль розраховано тільки на об'єкт 3-го порядку з одним входом, одним виходом тільки з трипараметричною матрицею. Цей модуль отримано копіюванням і модифікацією модуля синтезу регулятора.

**Випадок "вимірюється тільки X1".** Записуємо вхідні дані:

параметри об'єкта      матриця вимірювань      матриця параметрів спостерігача

$$A := \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a0 & -a1 & -a2 \end{pmatrix}; \quad C1 := \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}; \quad L1 := \begin{pmatrix} l1 & 0 & 0 \\ l2 & 0 & 0 \\ l3 & 0 & 0 \end{pmatrix}; \quad L1 \cdot C1 \rightarrow \begin{pmatrix} l1 & 0 & 0 \\ l2 & 0 & 0 \\ l3 & 0 & 0 \end{pmatrix}$$

$$I := \text{identity}(3); \quad I \cdot z - [I + (A + L1 \cdot C1) \cdot T] \rightarrow \begin{pmatrix} z - 1 - .4 \cdot l1 & -.4 & 0 \\ -.4 \cdot l2 & z - 1 & -.4 \\ .4 \cdot a0 - .4 \cdot l3 & .4 \cdot a1 & z - 1 + .4 \cdot a2 \end{pmatrix}$$

Розкриваємо детермінант, збираємо складові при однакових степенях z:

$$z^3 + (-l1 \cdot T - 3 + a2 \cdot T) \cdot z^2 + (-l2 \cdot T^2 - 2 \cdot a2 \cdot T + 2 \cdot l1 \cdot T + T^2 \cdot a1 - l1 \cdot T^2 \cdot a2 + 3) \cdot z + l1 \cdot T^2 \cdot a2 - l1 \cdot T^3 \cdot a1 - 1 + l2 \cdot T^2 - l2 \cdot T^3 \cdot a2 + T^3 \cdot a0 + a2 \cdot T - T^2 \cdot a1 - T^3 \cdot l3 - l1 \cdot T = 0$$

Прирівнюємо вирази для коефіцієнтів їх потрібним значенням, тобто - нулям.

$$\text{Given } (-l1 + a2) \cdot Tt - 3 = 0; \quad (-l2 + a1 - l1 \cdot a2) \cdot Tt^2 + (2 \cdot l1 - 2 \cdot a2) \cdot Tt + 3 = 0;$$

$$(-l2 \cdot a2 - l1 \cdot a1 - l3 + a0) \cdot Tt^3 + (l1 \cdot a2 + l2 - a1) \cdot Tt^2 + (-l1 + a2) \cdot Tt - 1 = 0$$

$$\text{Find}(l1, l2, l3) \rightarrow \left( \begin{array}{c} \frac{Tt \cdot a2 - 3}{Tt} \\ \frac{Tt^2 \cdot a1 - Tt^2 \cdot a2^2 + 3 \cdot Tt \cdot a2 - 3}{Tt^2} \\ \frac{-2 \cdot Tt^3 \cdot a1 \cdot a2 + Tt^3 \cdot a2^3 - 3 \cdot Tt^2 \cdot a2^2 + 3 \cdot Tt \cdot a2 + 3 \cdot Tt^2 \cdot a1 + Tt^3 \cdot a0 - 1}{Tt^3} \end{array} \right)$$

Визначаємо функцію користувача "розрахунок (синтез) спостерігача"

L01(Av, T) :=	$K_{n1} \leftarrow Av_3 - \frac{3}{T}$ $K_{n2} \leftarrow \left[ Av_2 - (Av_3)^2 + 3 \cdot \frac{Av_3}{T} - \frac{3}{T^2} \right]$ $K_{n3} \leftarrow \left[ \left[ -2 \cdot Av_2 \cdot Av_3 + Av_1 + (Av_3)^3 \right] + \left[ -3 \cdot (Av_3)^2 + 3 \cdot Av_2 \right] \cdot \frac{1}{T} + 3 \cdot Av_3 \cdot \frac{1}{T^2} - \frac{1}{T^3} \right]$ $K_n$
---------------	--

Тестуємо отриманий модуль синтезу спостерігача:

$$AA := \begin{pmatrix} 11 \\ 1 \\ -3 \end{pmatrix}; L01(AA, 0.2) = \begin{pmatrix} -15.3 \\ -78.59 \\ -122.277 \end{pmatrix}; L01(AA, 1.0) = \begin{pmatrix} -3.3 \\ -2.99 \\ 12.403 \end{pmatrix}; Aa := \begin{pmatrix} -10 \\ 10 \\ -3 \end{pmatrix}.$$

**Випадок "вимірюється тільки X2"**. Копіюємо попередній блок і дещо (?) змінюємо:

$$A := \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_0 & -a_1 & -a_2 \end{pmatrix}; L2 := \begin{pmatrix} 0 & l_2 & 0 \\ 0 & l_3 & 0 \\ 0 & l_1 & 0 \end{pmatrix}; C2 := \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}; L2 \cdot C2 \rightarrow \begin{pmatrix} 0 & l_2 & 0 \\ 0 & l_3 & 0 \\ 0 & l_1 & 0 \end{pmatrix};$$

$$I = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}; I \cdot z - [I + (A + L2 \cdot C2) \cdot dT] \rightarrow \begin{bmatrix} z-1 & -(1+l_2) \cdot dT & 0 \\ 0 & z-1-l_3 \cdot dT & -dT \\ a_0 \cdot dT & -(-a_1+l_1) \cdot dT & z-1+a_2 \cdot dT \end{bmatrix}.$$

Розкриваємо визначник - отримуємо характеристичне рівняння системи:

$$|I \cdot z - [I + (A + L \cdot C) \cdot T]| \rightarrow \begin{aligned} & z^3 + (-l_2 \cdot T - 3 + a_2 \cdot T) \cdot z^2 + (-l_3 \cdot T^2 - 2 \cdot a_2 \cdot T + 2 \cdot l_2 \cdot T + T^2 \cdot a_1 - l_2 \cdot T^2 \cdot a_2 + 3) \cdot z + \\ & + l_2 \cdot T^2 \cdot a_2 - T^2 \cdot a_1 + l_3 \cdot T^2 + T^3 \cdot a_0 + l_1 \cdot T^3 \cdot a_0 - 1 + a_2 \cdot T - l_2 \cdot T = 0. \end{aligned}$$

Прирівнюємо вирази для коефіцієнтів до нулів.

$$\text{Given } -l_2 \cdot T - 3 + a_2 \cdot T = 0; \quad -T^2 \cdot l_3 - 2 \cdot a_2 \cdot T + 2 \cdot l_2 \cdot T - l_2 \cdot T^2 \cdot a_2 + T^2 \cdot a_1 + 3 = 0; \\ l_2 \cdot T^2 \cdot a_2 - T^2 \cdot a_1 + T^2 \cdot l_3 + a_0 \cdot T^3 + a_0 \cdot T^3 \cdot l_1 - 1 + a_2 \cdot T - l_2 \cdot T = 0$$

$$\text{Find}(l_1, l_2, l_3) \text{ float, 1} \rightarrow \begin{pmatrix} -1 \cdot \frac{-1 \cdot 10^2 + 8 \cdot a_0}{a_0} \\ -8. + a_2 \\ -2 \cdot 10^1 + 8 \cdot a_2 - 1 \cdot a_2^2 + a_1 \end{pmatrix}.$$

На основі отриманого виразу записуємо модуль синтезу спостерігача стану:

$$L02(Av, T) := \begin{array}{l} Kn_1 \leftarrow \frac{-(Av_1 \cdot T^3 - 1)}{(Av_1 \cdot T^3)} \\ Kn_2 \leftarrow \left[ \frac{1}{T} \cdot (-3 + Av_3 \cdot T) \right] \\ Kn_3 \leftarrow \left[ \frac{-1}{T^2} \cdot \left[ (Av_3)^2 \cdot T^2 - 3 \cdot Av_3 \cdot T - T^2 \cdot Av_2 + 3 \right] \right] \\ Kn \end{array}$$

Тестуємо отриманий модуль синтезу спостерігача:

$$AA := \begin{pmatrix} 11 \\ 1 \\ -3 \end{pmatrix}; L02(AA, 0.2) = \begin{pmatrix} 10.364 \\ -15.3 \\ -78.59 \end{pmatrix}; L02(AA, 1.0) = \begin{pmatrix} -0.909 \\ -3.3 \\ -2.99 \end{pmatrix}; Aa := \begin{pmatrix} -10 \\ 10 \\ -3 \end{pmatrix}.$$

**Випадок "вимірюється тільки X3"**. Копіюємо попередній блок і дещо (?) змінюємо:

$$A := \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a0 & -a1 & -a2 \end{pmatrix}; L3 := \begin{pmatrix} 0 & 0 & l1 \\ 0 & 0 & l2 \\ 0 & 0 & l3 \end{pmatrix}; C3 := \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}; L3 \cdot C3 \rightarrow \begin{pmatrix} 0 & 0 & l1 \\ 0 & 0 & l2 \\ 0 & 0 & l3 \end{pmatrix};$$

$$I \cdot z - [I + (A + L3 \cdot C3) \cdot dT] \rightarrow \begin{bmatrix} z-1 & -dT & -l1 \cdot dT \\ 0 & z-1 & -(1+l2) \cdot dT \\ a0 \cdot dT & a1 \cdot dT & z-1 - (-a2+l3) \cdot dT \end{bmatrix}.$$

Розкриваємо визначник – отримуємо характеристичне рівняння системи:

$$|I \cdot z - [I + (A + L3 \cdot C3) \cdot T]| \rightarrow \blacksquare \\ z^3 + (-l3 \cdot T - 3 + a2 \cdot T) \cdot z^2 + (l1 \cdot T^2 \cdot a0 - 2 \cdot a2 \cdot T + 2 \cdot l3 \cdot T + T^2 \cdot a1 + l2 \cdot T^2 \cdot a1 + 3) \cdot z + (-l2 \cdot T^2 \cdot a1 + T^3 \cdot a0 + l2 \cdot T^3 \cdot a0 - l3 \cdot T - a0 \cdot T^2 \cdot l1 - 1 + a2 \cdot T - T^2 \cdot a1) = 0.$$

Прирівнюємо вирази для коефіцієнтів їх потрібним значенням, тобто – нулям.

$$\text{Given } -3 + a2 \cdot T - T \cdot l3 = 0; \quad a0 \cdot T^2 \cdot l1 - 2 \cdot a2 \cdot T + 2 \cdot T \cdot l3 + T^2 \cdot a1 + T^2 \cdot a1 \cdot l2 + 3 = 0; \\ -T^2 \cdot a1 \cdot l2 + a0 \cdot T^3 + a0 \cdot T^3 \cdot l2 - T \cdot l3 - a0 \cdot T^2 \cdot l1 - 1 + a2 \cdot T - T^2 \cdot a1 = 0$$

$$\text{Find}(l1, l2, l3) \text{ float, 2} \rightarrow \begin{pmatrix} -3.1 \cdot \frac{-6 \cdot a0 + 5 \cdot a1}{a0^2} \\ -1.13 \cdot \frac{-1.3 \cdot 10^2 + 8 \cdot a0}{a0} \\ -7.5 + a2 \end{pmatrix}.$$

На основі отриманого виразу записуємо модуль синтезу спостерігача стану:

$$L03(Av, T) := \begin{cases} Kn_1 \leftarrow \frac{-(Av_2 - 3 \cdot Av_1 \cdot T)}{[(Av_1)^2 \cdot T^3]} \\ Kn_2 \leftarrow \frac{-(Av_1 \cdot T^3 - 1)}{(Av_1 \cdot T^3)} \\ Kn_3 \leftarrow \frac{1}{T} \cdot (-3 + Av_3 \cdot T) \\ Kn \end{cases}$$

Тестуємо отриманий модуль синтезу спостерігача:

$$AA := \begin{pmatrix} 11 \\ 1 \\ -3 \end{pmatrix}; L03(AA, 0.2) = \begin{pmatrix} 5.785 \\ 10.364 \\ -15.3 \end{pmatrix}; Aa := \begin{pmatrix} -10 \\ 10 \\ -3 \end{pmatrix}; L03(Aa, 1.0) = \begin{pmatrix} -0.4 \\ -1.1 \\ -6 \end{pmatrix}.$$

### Модуль "модель САУ з спостерігачем"

Тепер розробимо модуль, що дозволяє моделювати поведінку автоматичної системи управління. Особливість цього модуля в тому, що він використовує як підпрограми попередні модулі – "синтез", "регулятор", "спостерігач". Дамо стандартизований опис модуля.

Параметри класу (структури)

Порядок дифференціалювання об'єкта, структура матриці параметрів об'єкта (форми Коші)

$$N := 3; AA := \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_0 & -a_1 & -a_2 \end{pmatrix}.$$

Функції обчислення перехідних процесів... Входи

1) параметри – вектори параметрів об'єкта –  $Av$  та  $B$ , крок квантування –  $T$ :

$$Av = \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix}; T := 0.5; B = \begin{pmatrix} 0 \\ 0 \\ b \end{pmatrix};$$

2) глобальні входи – число кроків моделювання  $Tm := 10$  та початкове значення век-

тора стану об'єкта  $x_0 := \begin{pmatrix} -1 \\ .5 \\ -0.5 \end{pmatrix}$ .

Вихід – перехідний процес – матриця складена з послідовних значень вектора стану.

$$x_{1s}(AV, \tau_1) = \begin{array}{c|cccccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\ \hline 1 & -1 & -0.5 & -0.5 & 0.1 & -0.1 & -0.1 & 0 & 0 & 0 & 0 \\ 2 & 0.5 & 0 & 0.6 & -0.2 & 0 & 0.1 & 0 & 0 & 0 & 0 \\ 3 & -0.5 & 0.6 & -0.8 & 0.2 & 0.1 & -0.1 & 0 & 0 & 0 & 0 \\ 4 & -1.1 & -0.7 & -0.6 & 0.1 & -0.1 & -0.1 & 0 & 0 & 0 & 0 \\ 5 & 0.4 & 0.1 & 0.7 & -0.2 & 0 & 0.1 & 0 & 0 & 0 & 0 \\ 6 & -0.4 & 0.4 & -1 & 0.2 & 0.1 & -0.1 & 0 & 0 & 0 & 0 \end{array}$$

Записуємо перше і друге наближення для відповідних перехідних функцій

$$F1(A, T) := I + \frac{A \cdot T}{1!}; \quad F2(A, T) := I + \frac{A \cdot T}{1!} + \frac{A^2 \cdot T^2}{2!};$$

$$F1(AA, 0.2) \rightarrow \begin{pmatrix} 1 & .2 & 0 \\ 0 & 1 & .2 \\ -.2 \cdot a0 & -.2 \cdot a1 & 1 - .2 \cdot a2 \end{pmatrix};$$

$$G1(A, T) := T; \quad T = 0.5;$$

$$G2(A, T) := T + \frac{A \cdot T^2}{2!}.$$

Задаємо початкові значення вектора стану і його оцінки спостерігачем

$$x_0 := \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}; \quad x_o := \begin{pmatrix} -1 \\ 0.5 \\ -0.5 \end{pmatrix}; \quad x_n := x_o - \begin{pmatrix} 0.1 \\ 0.1 \\ -0.1 \end{pmatrix}.$$

Виводимо матриці вимірювання для ситуацій "працює тільки один вимірювач".  
Вимірюється: тільки координата:      тільки швидкість:      тільки прискорення:

$$C1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}; \quad C2 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}; \quad C3 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Запишемо три програми моделювання САУ зі спостерігачем для трьох випадків вимірювання. Програми отримуємо відповідною модифікацією (наприклад, замінюємо  $L \rightarrow L1, L \rightarrow L2, L \rightarrow L3, C \rightarrow C1, C \rightarrow C2, C \rightarrow C3$ ) базової програми з розділу 2.1.

Подаємо на рис. 2.10 програму моделювання САУ зі спостерігачем для випадку вимірювання тільки швидкості. Програма майже не потребує коментарів. Нагадаємо тільки позначення:  $A$  – матриця стану об'єкта управління;  $B$  – матриця управління;  $x_0$  – вектор стану об'єкта;  $x_n$  – вектор стану спостерігача;  $K$  – матриця параметрів регулятора;  $l_1$  – параметри спостерігача;  $L$  – матриця параметрів спостерігача;  $u_k$  – управління;  $y_0^{(k)}$  – вектор вимірювань;  $y_n^{(k)}$  – оцінка вектора вимірювань;  $x_0^{(k+1)}$  – поточний стан об'єкта;  $\Delta y^{(k)}$  – помилка оцінювання;  $v_{\text{ход}}$  – зовнішній вхід спостерігача;  $x_n^{(k+1)}$  – поточний стан спостерігача;  $XX^{(k)}$  – вектор стану повної системи.

Для випадку вимірювання тільки швидкості матриця стану об'єкта управління  $A$  та матриця параметрів спостерігача  $L$  мають вигляд:

$$A \leftarrow \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -Av_1 & -Av_2 & -Av_3 \end{pmatrix}; \quad L \leftarrow \begin{pmatrix} 0 & l_1 & 0 \\ 0 & l_2 & 0 \\ 0 & l_3 & 0 \end{pmatrix}.$$

$x2s(Av, T) :=$	$I \leftarrow \text{identity}(3)$ $A \leftarrow \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -Av_1 & -Av_2 & -Av_3 \end{pmatrix}$ $B \leftarrow B_0$ $X_0^{(1)} \leftarrow x_0$ $X_n^{(1)} \leftarrow x_n$ $K \leftarrow Kr(Av, T)$ $l \leftarrow L02(Av, T)$ $L \leftarrow \begin{pmatrix} 0 & l_1 & 0 \\ 0 & l_2 & 0 \\ 0 & l_3 & 0 \end{pmatrix}$ for $k \in 1..T_m$ $u_k \leftarrow K^T \cdot X_n^{(k)}$ $Y_0^{(k)} \leftarrow C2 \cdot X_0^{(k)}$ $Y_n^{(k)} \leftarrow C2 \cdot X_n^{(k)}$ $X_0^{(k+1)} \leftarrow F1(A, T) \cdot X_0^{(k)} + G1(A, T) \cdot B \cdot u_k$ $\Delta y^{(k)} \leftarrow Y_n^{(k)} - Y_0^{(k)}$ $v_{xod} \leftarrow G1(A, T) \cdot B \cdot u_k + G1(A, T) \cdot L \cdot C2 \cdot \Delta y^{(k)}$ $X_n^{(k+1)} \leftarrow F1(A, T) \cdot X_n^{(k)} + v_{xod}$ $XX^{(k)} \leftarrow \text{stack}(X_0^{(k)}, X_n^{(k)})$ XX	Програма майже не потребує коментарів. A – матриця стану об'єкта управління; B – матриця управління; $X_0$ – вектор стану об'єкта; $X_n$ – вектор стану спостерігача; K – матриця параметрів регулятора; l – параметри спостерігача; L – матриця параметрів спостерігача. – управління; – вектор вимірювань; – оцінка вектора вимірювань; – об'єкт; – помилка оцінювання; – зовнішній вхід спостерігача; – спостерігач; – формуємо вектор стану повної системи.
-----------------	--	--

Рис. 2.10. Програма моделювання САУ зі спостерігачем для випадку вимірювання тільки швидкості

**Неважко написати** один модуль для усіх трьох випадків – слід додати тільки параметр варіанта вхідних даних спостерігача.

Побудуємо серію перехідних процесів для певного об'єкта при різних ситуаціях вимірювання (тобто для  $C = C1, C2, C3$ ).

Задаємо кількість кроків моделювання  $T_m$  та відповідну ранжовану змінну - номер такту квантування. Будуємо серію перехідних процесів.

Сучасні інформаційні технології в проектуванні – це, перш за все, перевірка кожного кроку експериментально. Для цього робимо стенд для дослідження впливу  $T_{квант}$  та параметрів об'єкта  $Av$  – змінюйте відповідні параметри, спостерігайте процеси, робіть попередні висновки. Тут  $\tau_1, \tau_2$  – два різних кроки квантування.

$$AV := \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}; \quad \tau_1 := 1; \quad \tau_2 := .5; \quad T_m := 10; \quad k := 1..T_n.$$

$$x1s(AV, \tau_2) =$$

	1	2	3	4	5	6	7	8	9	10
1	-1	-0.75	-0.625	0.175	0	0	0	0	0	0
2	0.5	0.25	1.6	-0.35	0	0	0	0	0	0
3	-0.5	2.7	-3.9	0.7	0	0	0	0	0	0
4	-1.1	-0.75	-0.625	0.175	0	0	0	0	0	0
5	0.4	0.25	1.6	-0.35	0	0	0	0	0	0
6	-0.4	2.7	-3.9	0.7	0	0	0	0	0	0

На рис. 2.11 подано шість процесів – з двома значеннями кроку квантування та трьома варіантами спостережів – на базі вимірювання  $x_1$ ,  $x_2$ , та  $x_3$  відповідно.

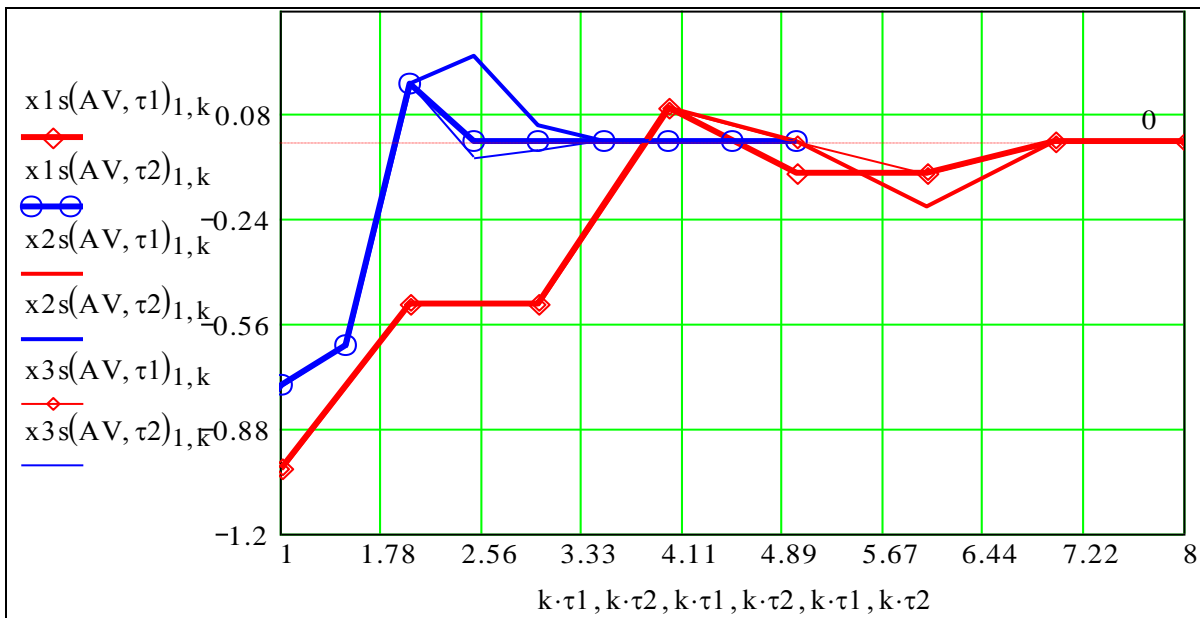


Рис. 2.11. Перехідні процеси в САУ. Координати

На рис. 2.12 подано два процеси в деталях: координати, швидкості, прискорення.

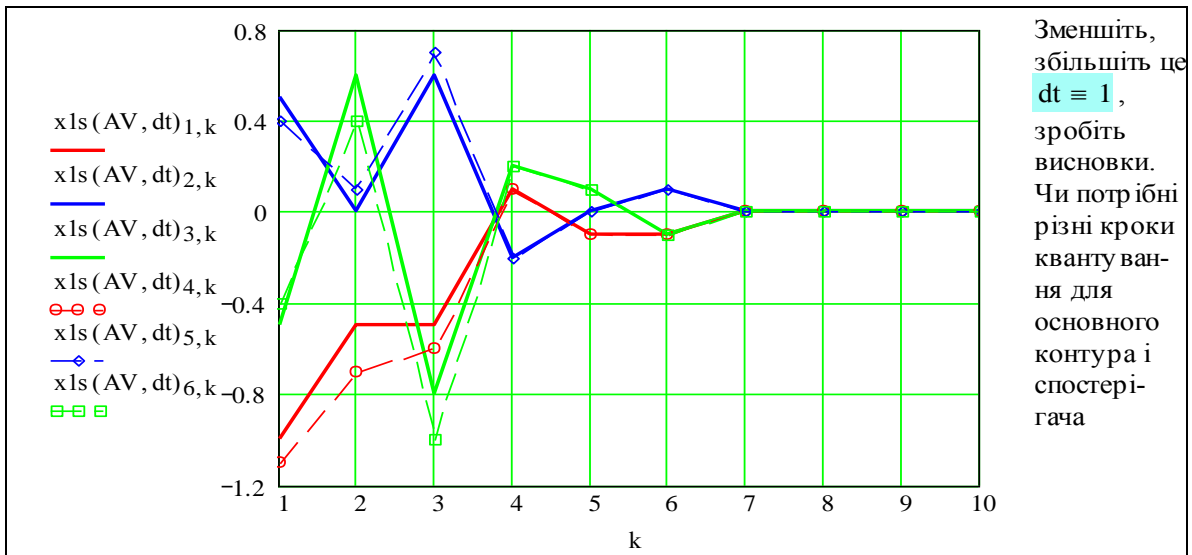


Рис. 2.12. Перехідні процеси в САУ. Складові вектора стану – координата, швидкість, прискорення

## Модуль "моделювання відмовостійкої САУ"

Тепер розробимо модуль, що дозволяє моделювати поведінку системи управління. Особливість цього модуля в тому, що він використовує як підпрограми попередні модулі – "синтез", "регулятор", "спостерігач". Дамо стандартизований опис модуля.

Параметри класу (структури):

порядок диференціального об'єкта = 3, структура матриці параметрів об'єкта (форми Коші),

$$N := 3; \quad AA := \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_0 & -a_1 & -a_2 \end{pmatrix}; \quad \text{Входи: } Av = \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix}; \quad B = \begin{pmatrix} 0 \\ 0 \\ b \end{pmatrix};$$

1) параметри – вектори параметрів об'єкта Av та B, крок квантування T;

2) глобальні входи – число кроків моделювання Tmd = 45, початкове значення вектора

стану об'єкта  $x_0 := \begin{pmatrix} -1 \\ .5 \\ -0.5 \end{pmatrix}$ , функція відмов вимірювачів, T = 0.5 ;

3) вихід – перехідний процес – матриця складена з послідовних значень вектора стану.

	1	2	3	4	5	6	7	8
1	-0.8	-0.8	-0.8	0.16	0.49	0.984	0.875	0.796
2	0	0	1.921	0.659	0.988	-0.219	-0.158	-0.409
3	0	3.842	-2.523	0.657	-2.414	0.122	-0.502	-0.373
4	-0.6	0.3	-1.2	0.06	0.49	0.984	0.875	0.796
5	0	0.3	1.321	0.559	0.988	-0.219	-0.158	-0.409
6	0	3.142	-1.923	0.757	-2.414	0.122	-0.502	-0.373
7	-0.6	-0.6	-0.6	0.16	0.49	0.984	0.875	0.796
8	0	0	1.871	0.659	0.988	-0.219	-0.158	-0.409
9	0	3.742	-2.573	0.657	-2.414	0.122	-0.502	-0.373
10	-0.6	-0.6	-0.4	0.16	0.49	0.984	0.875	0.796
11	0	0	1.521	0.659	0.988	-0.219	-0.158	-0.409
12	0	3.742	-2.423	0.657	-2.414	0.122	-0.502	-0.373

### Сценарій моделювання

Задаємо таким сценарієм моделювання: САУ відпрацьовує заданий зовнішній вхід (гармонічний сигнал, послідовність імпульсів, або шум); на деякому кроці квантування відмовляє один із вимірювачів, потім - ще один. На основі аналізу перехідних процесів оцінюються альтернативи побудови відмовостійкої САУ.

Для реалізації відмовостійкої системи і моделювання її згідно зі сценарієм необхідно розробити ряд модулів. Далі подано приклади відповідних модулів.

#### Альтернативні варіанти функції відмов вимірювачів

Функція Ot1(.) – обнуляє відповідний компонент виходу, якщо номер кроку квантування  $k > kot_1$  – моменту відмови; Ot2(.) – починаючи з кроку  $kot_1$  – додає до вхідного сигналу шум заданої амплітуди.

$$Ot1(X, kot, k) := \begin{bmatrix} X_1 \cdot (1 - \Phi(k - kot_1)) \\ X_2 \cdot (1 - \Phi(k - kot_2)) \\ X_3 \cdot (1 - \Phi(k - kot_3)) \end{bmatrix}; \quad Ot2(X, kot, k) := \begin{bmatrix} X_1 \cdot (1 + .3 \cdot \Phi(k - kot_1) \cdot \text{rnd}(1)) \\ X_2 \cdot (1 + .3 \cdot \Phi(k - kot_2) \cdot \text{rnd}(1)) \\ X_3 \cdot (1 + .3 \cdot \Phi(k - kot_3) \cdot \text{rnd}(1)) \end{bmatrix}$$

**Варіанти функції входу** – задані значення вектора стану об'єкта:

випадковий сигнал гармонічний сигнал імпульси з періодом P  
 $u_3(q) := (\text{rnd}(1) - 0.5); \quad u_2(q, \omega) := \sin(\omega \cdot q); \quad u_1(q, P) := \text{if}(\text{mod}(q, P) = 0, 1, 0).$



Задаємо початкові стани об'єкта і спостерігача, матриці вимірювань

$$x_0 := \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}; x_n := x_0 - \begin{pmatrix} 0.1 \\ 0.0 \\ -0.0 \end{pmatrix}; C1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}; C2 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}; C3 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

**Функції, що задають структури матриць параметрів спостерігача.** Це є одне з можливих рішень у конструюванні досить універсальної програми синтезу спостерігача. Записуємо функції L01(Av,T), L02(Av,T), L03(Av,T), що подають структуру. Вимірюємо:

$$\begin{array}{l} \text{координату:} \\ L1(qq) := \begin{pmatrix} qq_1 & 0 & 0 \\ qq_2 & 0 & 0 \\ qq_3 & 0 & 0 \end{pmatrix}; \end{array} \quad \begin{array}{l} \text{швидкість:} \\ L2(qq) := \begin{pmatrix} 0 & qq_1 & 0 \\ 0 & qq_2 & 0 \\ 0 & qq_3 & 0 \end{pmatrix}; \end{array} \quad \begin{array}{l} \text{прискорення} \\ L3(qq) := \begin{pmatrix} 0 & 0 & qq_1 \\ 0 & 0 & qq_2 \\ 0 & 0 & qq_3 \end{pmatrix}. \end{array}$$

**Функція логіки формування вихідної оцінки вектора стану об'єкта – центр проектування в даній роботі.** Вимоги до цієї функції: детермінована "ізоляція" сигналів від елемента, що відмовив; простота, відсутність механічних та інших переключень. Різні логічні пристрої, які виявляють та відключають некоректно працюючі елементи, самі є джерелами відмов. Вкрай нерозумно спроектувати систему під ідеальні елементи та умови, а потім ліпити на неї резервування, діагностику, контроль. В тексті програми багато "чорних ящиків" – матрично-векторних функцій. Замість коментарів подаємо те, що було необхідним для контролю і відлагодження програми. Це "інформаційна технологія".

$Xf(Mv) := \begin{array}{l} M_{\text{razn}}^{\langle 1 \rangle} \leftarrow  Mv^{\langle 2 \rangle} - Mv^{\langle 3 \rangle}  \\ M_{\text{razn}}^{\langle 2 \rangle} \leftarrow  Mv^{\langle 1 \rangle} - Mv^{\langle 3 \rangle}  \\ M_{\text{razn}}^{\langle 3 \rangle} \leftarrow  Mv^{\langle 1 \rangle} - Mv^{\langle 2 \rangle}  \\ M_{\text{rt}} \leftarrow M_{\text{razn}}^T \\ \text{for } j \in 1..3 \\ \quad \left  \begin{array}{l} \text{sum}_j \leftarrow \sum M_{\text{rt}}^{\langle j \rangle} \\ \text{sto} \leftarrow 100 \end{array} \right. \\ \text{sumO} \leftarrow \sum \text{sum} \\ \text{for } j \in 1..3 \\ \quad \left  \begin{array}{l} \text{ves}_j \leftarrow \begin{cases} 0.33 & \text{if } M_{\text{rt}} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \\ \frac{\sum M_{\text{razn}}^{\langle j \rangle}}{\text{sumO}} & \text{otherwise} \end{cases} \end{array} \right. \\ \text{Xoc} \leftarrow \text{ves}_1 \cdot Mv^{\langle 1 \rangle} + \text{ves}_2 \cdot Mv^{\langle 2 \rangle} + \text{ves}_3 \cdot Mv^{\langle 3 \rangle} \end{array}$	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="border: 1px solid black; padding: 5px;"><math>x_1 := \begin{pmatrix} 1.2 \\ 2.2 \\ 10. \end{pmatrix}</math></td> <td style="border: 1px solid black; padding: 5px;"><math>x_2 := \begin{pmatrix} 1.2 \\ 2.2 \\ 10 \end{pmatrix}</math></td> <td style="border: 1px solid black; padding: 5px;"><math>x_3 := \begin{pmatrix} 1.2 \\ 22 \\ 13. \end{pmatrix}</math></td> </tr> <tr> <td style="border: 1px solid black; padding: 5px;"><math>M_{\text{r}}^{\langle 1 \rangle} :=  x_2 - x_3 </math></td> <td style="border: 1px solid black; padding: 5px;"><math>M_{\text{r}}^{\langle 2 \rangle} :=  x_1 - x_3 </math></td> <td></td> </tr> <tr> <td style="border: 1px solid black; padding: 5px;"><math>M_{\text{r}}^{\langle 3 \rangle} :=  x_1 - x_2 </math></td> <td style="border: 1px solid black; padding: 5px;"><math>M_{\text{t}} := M_{\text{r}}^T</math></td> <td></td> </tr> <tr> <td colspan="3" style="border: 1px solid black; padding: 5px; text-align: center;"><math>j := 1..3</math></td> </tr> <tr> <td colspan="3" style="border: 1px solid black; padding: 5px;"><math>M_{\text{r}} = \begin{pmatrix} 0 &amp; 0 &amp; 0 \\ 19.8 &amp; 19.8 &amp; 0 \\ 3 &amp; 3 &amp; 0 \end{pmatrix}</math></td> </tr> <tr> <td colspan="3" style="border: 1px solid black; padding: 5px;"><math>M_{\text{t}} = \begin{pmatrix} 0 &amp; 19.8 &amp; 3 \\ 0 &amp; 19.8 &amp; 3 \\ 0 &amp; 0 &amp; 0 \end{pmatrix}</math></td> </tr> <tr> <td style="border: 1px solid black; padding: 5px;"><math>\text{su}_j := \sum M_{\text{t}}^{\langle j \rangle}</math></td> <td style="border: 1px solid black; padding: 5px;"><math>\text{suO} := \sum \text{su}_j</math></td> <td></td> </tr> <tr> <td colspan="3" style="border: 1px solid black; padding: 5px; text-align: center;"><math>\text{suO} = 45.6</math></td> </tr> <tr> <td style="border: 1px solid black; padding: 5px;"><math>\text{su} = \begin{pmatrix} 0 \\ 39.6 \\ 6 \end{pmatrix}</math></td> <td colspan="2" style="border: 1px solid black; padding: 5px;"><math>\text{vs}_j := \frac{\sum M_{\text{r}}^{\langle j \rangle}}{\text{suO}}</math></td> </tr> </table>	$x_1 := \begin{pmatrix} 1.2 \\ 2.2 \\ 10. \end{pmatrix}$	$x_2 := \begin{pmatrix} 1.2 \\ 2.2 \\ 10 \end{pmatrix}$	$x_3 := \begin{pmatrix} 1.2 \\ 22 \\ 13. \end{pmatrix}$	$M_{\text{r}}^{\langle 1 \rangle} :=  x_2 - x_3 $	$M_{\text{r}}^{\langle 2 \rangle} :=  x_1 - x_3 $		$M_{\text{r}}^{\langle 3 \rangle} :=  x_1 - x_2 $	$M_{\text{t}} := M_{\text{r}}^T$		$j := 1..3$			$M_{\text{r}} = \begin{pmatrix} 0 & 0 & 0 \\ 19.8 & 19.8 & 0 \\ 3 & 3 & 0 \end{pmatrix}$			$M_{\text{t}} = \begin{pmatrix} 0 & 19.8 & 3 \\ 0 & 19.8 & 3 \\ 0 & 0 & 0 \end{pmatrix}$			$\text{su}_j := \sum M_{\text{t}}^{\langle j \rangle}$	$\text{suO} := \sum \text{su}_j$		$\text{suO} = 45.6$			$\text{su} = \begin{pmatrix} 0 \\ 39.6 \\ 6 \end{pmatrix}$	$\text{vs}_j := \frac{\sum M_{\text{r}}^{\langle j \rangle}}{\text{suO}}$	
$x_1 := \begin{pmatrix} 1.2 \\ 2.2 \\ 10. \end{pmatrix}$	$x_2 := \begin{pmatrix} 1.2 \\ 2.2 \\ 10 \end{pmatrix}$	$x_3 := \begin{pmatrix} 1.2 \\ 22 \\ 13. \end{pmatrix}$																										
$M_{\text{r}}^{\langle 1 \rangle} :=  x_2 - x_3 $	$M_{\text{r}}^{\langle 2 \rangle} :=  x_1 - x_3 $																											
$M_{\text{r}}^{\langle 3 \rangle} :=  x_1 - x_2 $	$M_{\text{t}} := M_{\text{r}}^T$																											
$j := 1..3$																												
$M_{\text{r}} = \begin{pmatrix} 0 & 0 & 0 \\ 19.8 & 19.8 & 0 \\ 3 & 3 & 0 \end{pmatrix}$																												
$M_{\text{t}} = \begin{pmatrix} 0 & 19.8 & 3 \\ 0 & 19.8 & 3 \\ 0 & 0 & 0 \end{pmatrix}$																												
$\text{su}_j := \sum M_{\text{t}}^{\langle j \rangle}$	$\text{suO} := \sum \text{su}_j$																											
$\text{suO} = 45.6$																												
$\text{su} = \begin{pmatrix} 0 \\ 39.6 \\ 6 \end{pmatrix}$	$\text{vs}_j := \frac{\sum M_{\text{r}}^{\langle j \rangle}}{\text{suO}}$																											

Тестуємо модуль – порівнюємо результати  $\leftarrow$  цих  $\rightarrow$  і цих  $\rightarrow$  обчислень:

$$x_1 = \begin{pmatrix} 1.2 \\ 2.2 \\ 10 \end{pmatrix}; \quad \text{vs}_1 \cdot x_1 + \text{vs}_2 \cdot x_2 + \text{vs}_3 \cdot x_3 = \begin{pmatrix} 1.2 \\ 2.2 \\ 10 \end{pmatrix}; \quad \text{vs} = \begin{pmatrix} 0.5 \\ 0.5 \\ 0 \end{pmatrix}.$$

Система моделювання відмовостійкої САУ – кінцевий результат. Подамо програму моделювання відмовостійкої САУ на рис. 2.13.

<pre> vs(Av, T) := I ← identity(3) A ← <math>\begin{pmatrix} 0 &amp; 1 &amp; 0 \\ 0 &amp; 0 &amp; 1 \\ -Av_1 &amp; -Av_2 &amp; -Av_3 \end{pmatrix}</math> B ← B0 Xo<sup>&lt;1&gt;</sup> ← xo X1n<sup>&lt;1&gt;</sup> ← xn X2n<sup>&lt;1&gt;</sup> ← xn X3n<sup>&lt;1&gt;</sup> ← xn K ← Kr(Av, T) l1 ← L01(Av, T) l2 ← L02(Av, T) l3 ← L03(Av, T) for k ∈ 1..Tmd   Xoc<sup>&lt;k&gt;</sup> ← Xf(augment(X1n<sup>&lt;k&gt;</sup>, X2n<sup>&lt;k&gt;</sup>, X3n<sup>&lt;k&gt;</sup>))   цк ← K<sup>T</sup> · <math>\begin{bmatrix} Xoc<sup>&lt;k&gt;</sup> - (1 \cdot \sin(0.4 \cdot k)) \\ 0 \\ 0 \end{bmatrix}</math>   Xo<sup>&lt;k+1&gt;</sup> ← F1(A, T) · Xo<sup>&lt;k&gt;</sup> + G1(A, T) · B · цк   OO ← Ot1(Xo<sup>&lt;k+1&gt;</sup>, kot, k)   QQ ← Ot2(Xo<sup>&lt;k+1&gt;</sup>, kot, k)   X1<sup>&lt;k+1&gt;</sup> ← OO · vo + QQ · (¬vo)   Δy1<sup>&lt;k&gt;</sup> ← C1 · (X1n<sup>&lt;k&gt;</sup> - X1<sup>&lt;k&gt;</sup>)   Δy2<sup>&lt;k&gt;</sup> ← C2 · (X2n<sup>&lt;k&gt;</sup> - X1<sup>&lt;k&gt;</sup>)   Δy3<sup>&lt;k&gt;</sup> ← C3 · (X3n<sup>&lt;k&gt;</sup> - X1<sup>&lt;k&gt;</sup>)   vx1 ← G1(A, T) · B · цк + G1(A, T) · l1(l1) · C1 · Δy1<sup>&lt;k&gt;</sup>   vx2 ← G1(A, T) · B · цк + G1(A, T) · l2(l2) · C2 · Δy2<sup>&lt;k&gt;</sup>   vx3 ← G1(A, T) · B · цк + G1(A, T) · l3(l3) · C3 · Δy3<sup>&lt;k&gt;</sup>   X1n<sup>&lt;k+1&gt;</sup> ← F1(A, T) · X1n<sup>&lt;k&gt;</sup> + vx1   X2n<sup>&lt;k+1&gt;</sup> ← F1(A, T) · X2n<sup>&lt;k&gt;</sup> + vx2   X3n<sup>&lt;k+1&gt;</sup> ← F1(A, T) · X3n<sup>&lt;k&gt;</sup> + vx3   XX<sup>&lt;k&gt;</sup> ← stack(Xo<sup>&lt;k&gt;</sup>, X1n<sup>&lt;k&gt;</sup>, X2n<sup>&lt;k&gt;</sup>, X3n<sup>&lt;k&gt;</sup>) XX </pre>	<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: auto;"> <p><b>Завдання (обов'язкове)</b> Зробіть специфікацію складових програми: Будовані функції пакета: ..... Функції користувача (підпрограми): ..... Вхідні дані Вихідні дані (скаляри, вектори, матриці)</p> </div>	<p>Недолік програми - мало місця для коментарів. Нагадуємо тільки позначення: Xo - вектор стану об'єкта; X1n..X3n - вектори стану спостерігачів, (тобто оцінки вектора стану об'єкта Xo)  - Розрахунок параметрів регулятора і спостерігачів.  - цикл моделювання -  -остаточна оцінка вектора стану  - управління  - модель об'єкта - модель відмов  - помилки оцінювання вимірюваних координат різними спостерігачами.      - зовнішні входи спостерігачів      - спостерігачі    - формування виходу.</p>
--	--	--

Рис. 2.13. Програма моделювання відмовостійкої САУ

Параметри експерименту: відмова (повна – 1, часткова – 0)  $v_0 \equiv 0$ ; крок квантування  $T \equiv 0.5$ ;  $T_{md} \equiv 45$ ;  $k := 1..T_{md}$ ;  $хах := -5, -4.8..5$ .

$$AV \equiv \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}; \quad x_0 \equiv \begin{pmatrix} -0.8 \\ 0 \\ 0 \end{pmatrix}; \quad x_n \equiv \begin{pmatrix} -0.6 \\ 0 \\ 0 \end{pmatrix}; \quad kot \equiv \begin{pmatrix} 33 \\ 15 \\ 25 \end{pmatrix}; \quad Myy := vs(AV, 0.5).$$

На рис. 2.14 подано приклад процесів в САУ в двох формах "все разом" і "в деталях".

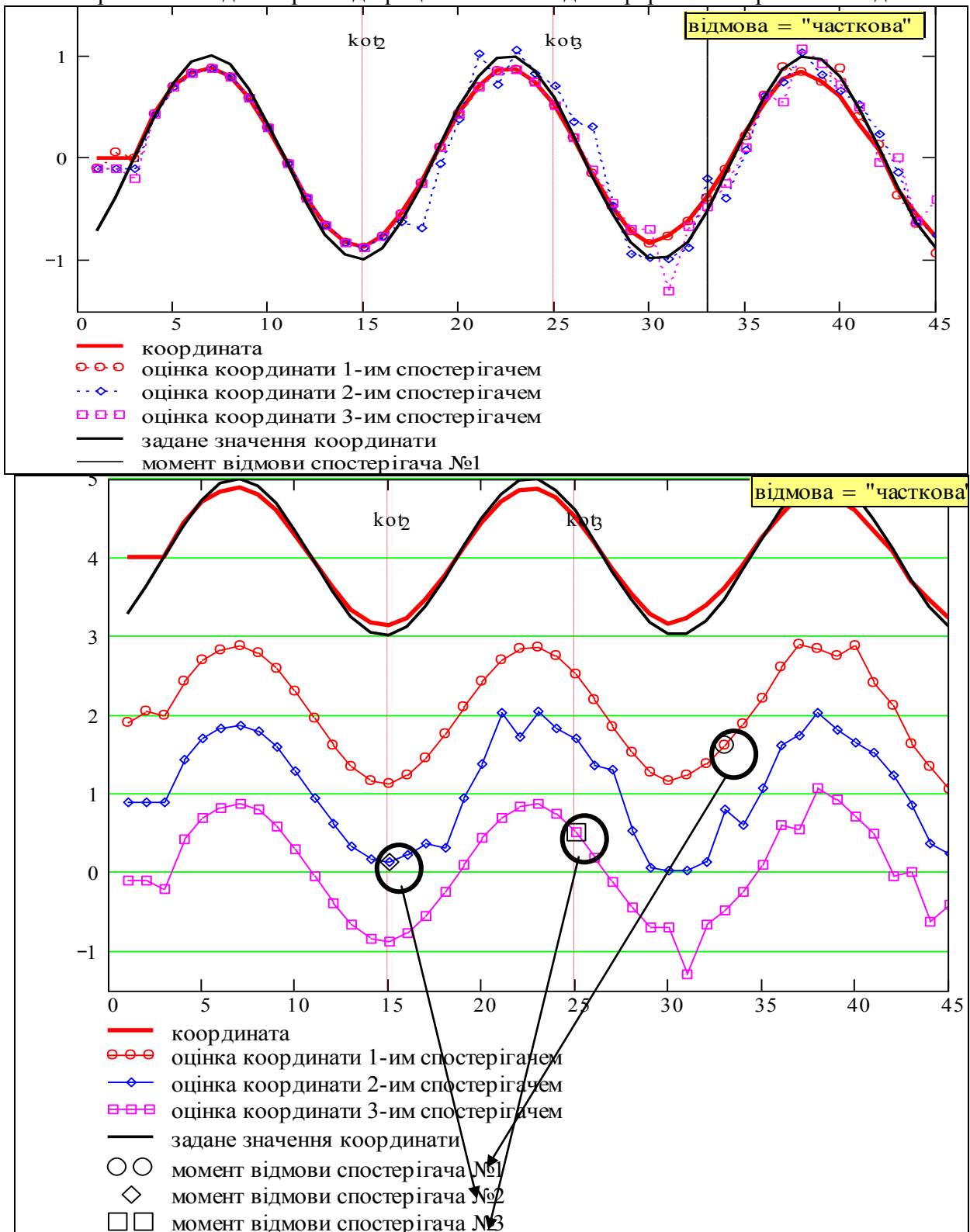


Рис. 2.14. Результати моделювання відмовостійкої САУ. Приклад 1

На рис. 2.15 подано результат моделювання для повної відмови вимірювачів: X1, X2, X3 на відповідних кроках.

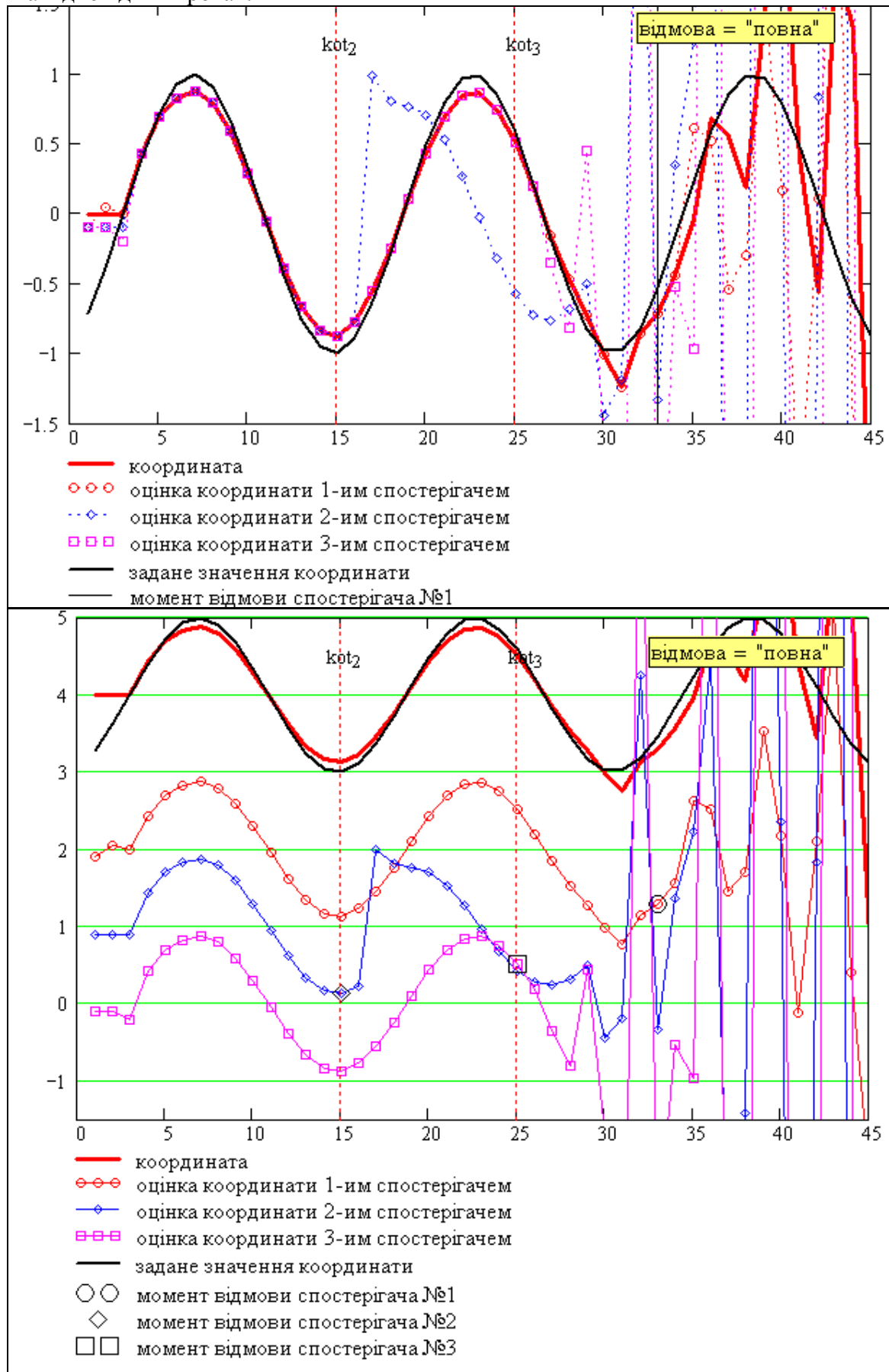
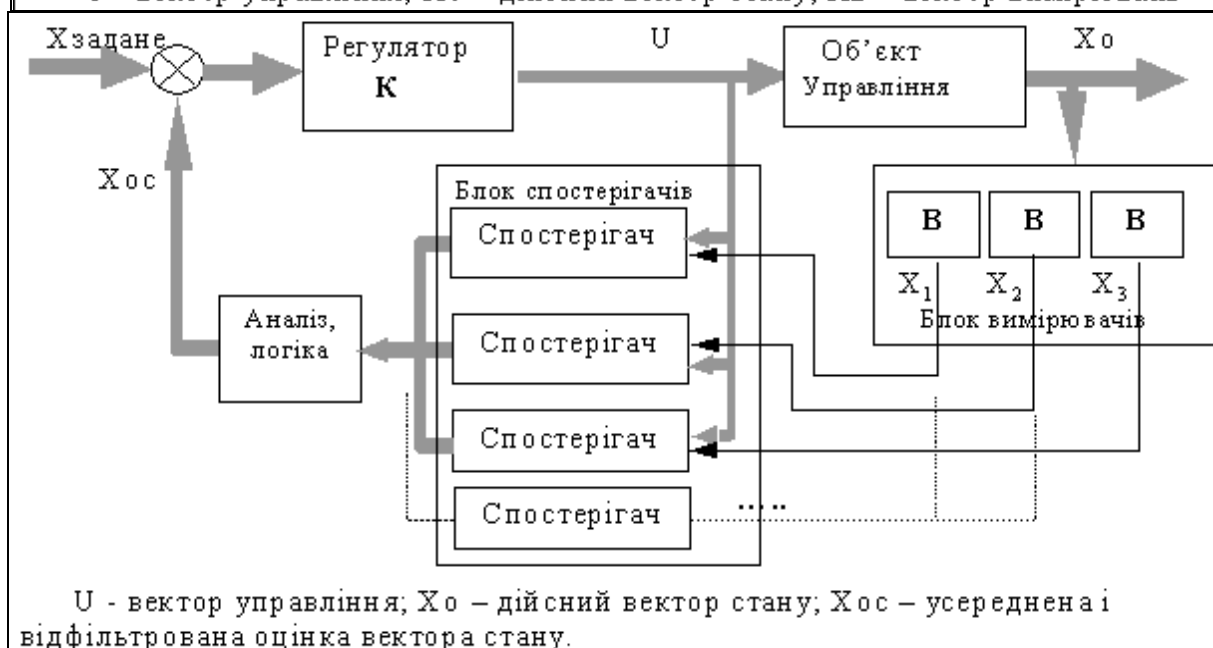
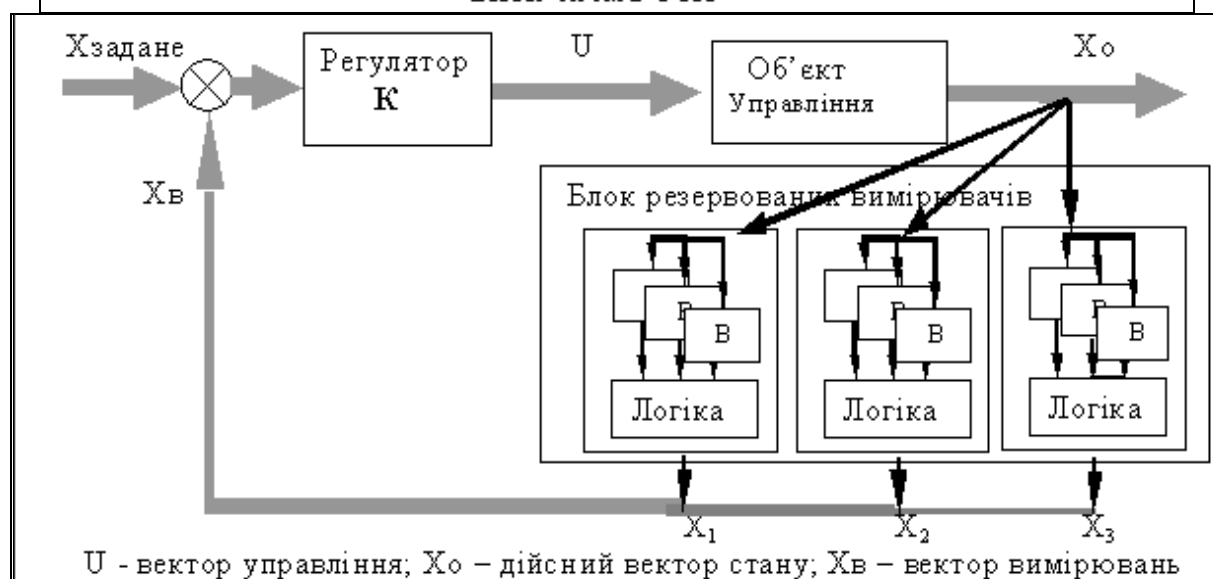
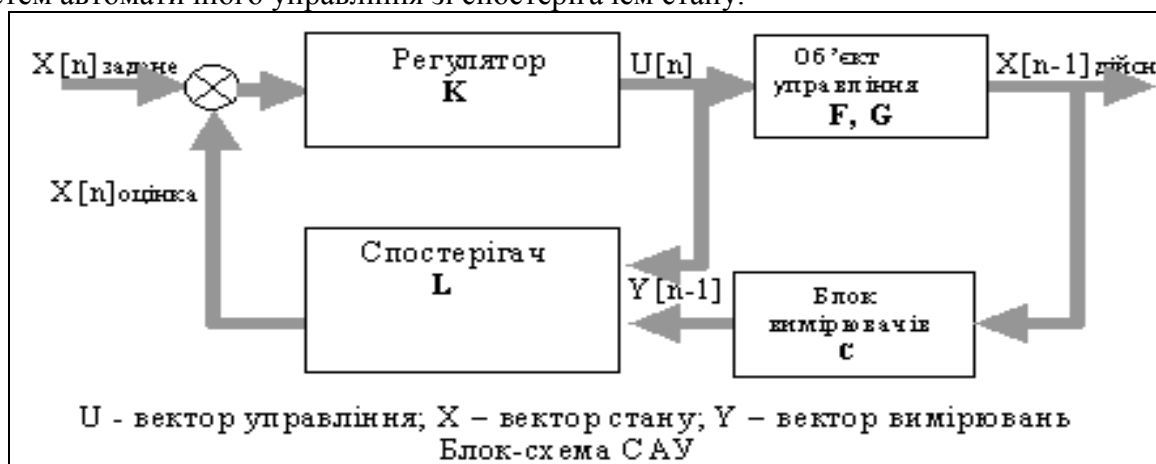


Рис. 2.15. Результати моделювання відмовостійкої САУ. Приклад 2

## Висновки

Програма моделювання працездатна, розроблена відмовостійка САУ – задовільна, тому що підпрограма "ізоляції" вимірювачів, які відмовили, незадовільно працює при відмові двох вимірювачів. Розробили робочі моделі для синтезу і моделювання імпульсних систем автоматичного управління зі спостерігачем стану.



### 3 СИНТЕЗ І ДОСЛІДЖЕННЯ ТЕРМІНАЛЬНОЇ САУ

В цьому розділі будемо робити моделі для синтезу управління кінцевим станом і моделювання динаміки об'єкта управління. Сьогодні існує мода приклеювати етикетки „модель” і „моделювання” на все сутнє. Чи є сенс у словосполученні „модель синтезу”? Так ми називаємо програму, що імітує введення вхідних даних і розрахунок параметрів закону управління і введення цих параметрів у регулятор поточного стану. Задачі управління кінцевим станом зустрічаються нам на кожному кроці, але вони мало розглядаються в літературі з управління.

Головна увага приділена технологіям конструювання робочих моделей, придатних для проведення обчислювальних експериментів та методів синтезу, що можуть використовуватися в мікроконтролерах.

#### 3.1 Синтез термінальної САУ для типових задач управління кінцевим станом

**Постановка і загальне розв'язання задачі.** Розглянемо задачу управління рухом деякого керованого об'єкта по одній осі декартової системи координат. Будемо вважати об'єкт матеріальною точкою і запишемо для нього звичайні ньютонівські рівняння динаміки. Потім, коли ми відпрацюємо таку одновимірну задачу для об'єкта другого порядку, ми розглянемо системи вищих порядків.

$$\frac{d}{dt}V(t) = \frac{1}{m} \cdot \left( \sum_{i=1}^N F_i - \sum_{j=1}^K F_j \right); \quad \frac{d}{dt}S(t) = V(t), \quad (3.1)$$

де  $\sum_{i=1}^N F_i$ ;  $\sum_{j=1}^K F_j$  – суми контрольованих ("управлінь") і неконтрольованих ("збурень")

узагальнених сил. Ця задача відноситься до класу варіаційних задач, де шукається не одна точка, а функція. Запишемо задачу в стандартній формі.

*Об'єкт управління:* координати  $S$  – положення,  $V$  – швидкість,  $ud$  – прискорення дійсне – в нашій задачі воно є управлінням,  $ut$  – прискорення потрібне.

*Обмеження диференціальні:*

$$\frac{d}{dt}V(t) = \frac{1}{m} \cdot \left( \sum_{i=1}^N F_i - \sum_{j=1}^K F_j \right); \quad \frac{d}{dt}S(t) = V(t).$$

*Обмеження алгебраїчні:* на управління і координати – вводяться далі.

*Граничні умови:*  $S(0) = S_0$ ;  $V(0) = V_0$ ;  $S(T) = S_k$ ;  $V(T) = V_k$ , що означає: задано початковий стан, до моменту часу  $T$  треба перейти в заданий кінцевий стан.

*Критерій:* 
$$J = \int_0^T u^2 dt \quad (3.2)$$

– характеризує витрати енергії на управління процесом переходу.

*Ціль оптимізації*  $\min_{ut}(J)$  – знайти таке управління, що дає мінімум критерію, тобто мінімізує витрати енергії на управління. Зауважимо, що критерій є функціоналом, і нагадаємо, що: *функція* – це відображення точки в точку;

*функціонал* – відображення функції в точку;

*оператор* – відображення функції в функцію.

Канонічний шлях розв'язання варіаційної задачі – записати звичайно нелінійні диференціальні рівняння Ейлера-Лагранжа, або застосувати метод динамічного програмування Р. Беллмана чи метод принципу максимуму Л. Понтрягіна. Є ще один шлях дуже важкий, але прос-

тий – "вгадати" вид розв'язання. Для даної задачі структура оптимального управління відома – оптимальне управління шукаємо в класі поліномів:

$$u = \sum_{i=0}^{m-1} C_i \cdot t^i, \quad (3.3)$$

де  $m$  – число граничних умов.

Розв'язання задачі тепер зводиться до знаходження невідомих коефіцієнтів  $C_i$ . Число граничних умов може бути і більше, і менше порядку системи. Будемо поки розглядати випадки, коли число граничних умов не більше порядку системи. Задачі термінального управління в залежності від числа кінцевих умов класифікуються [1] на задачі:

- 1) розгону  $(S_0, V_0) \Rightarrow (S_k, V_k)$ , де треба забезпечити потрібну кінцеву швидкість;
- 2) приведення  $(S_0, V_0) \Rightarrow (S_k, V_k)$ , де треба забезпечити потрібну кінцеву координату;
- 3) зближення  $(S_0, V_0) \Rightarrow (S_k, V_k)$ , де треба забезпечити потрібні швидкість і координату.

Розглянемо тепер послідовно розв'язання цих задач [1].

### 3.1.1 Синтез термінальної САУ для задачі розгону

Потрібно розігнати (загальмувати) об'єкт від швидкості  $V_0$  в момент  $t = 0$  до швидкості  $V_k$  в момент  $t=T$ . Вибираємо метод, згідно з яким управлінням є вища похідна в рівнянні динаміки (в нашому випадку – прискорення) [5].

Приймаємо головне припущення про те, що система управління має досконалі виконавчі елементи (приводи), досить потужні, щоб компенсувати збурення і досить точні, щоб забезпечити умову:  $u_d = u_t$  – дійсне прискорення дорівнює потрібному.

$$u = \sum_{i=0}^{m-1} C_i \cdot t^i \text{ для } m=1 \text{ маємо } u = C_0, \text{ інтегруємо } V(T) = \int_0^T C_0 dt, \text{ звідки маємо:}$$

$$V_k = V_0 + C_0 \cdot T; \quad \boxed{C_0 = u = (V_k - V_0) \div T}.$$

Швидкість в процесі розгону буде змінюватись згідно з рівнянням:  $V(t) = V_0 + \frac{V_k - V_0}{T} \cdot t$ .

Отриманий закон управління має недолік: якщо в середині процесу будуть діяти збурення, або виявляться помилки, то вони не будуть виправлені, бо закон розраховується на початку процесу переходу.

Введемо зворотний зв'язок (3.3): будемо вважати поточний стан об'єкта початковим і для кожного поточного стану будемо розраховувати програму переходу в кінцевий стан за час  $(T - t)$ , що залишається до кінця процесу.

Записуємо вираз для потрібного управління з 3.3 (не плутайте з 3.3 – тридцять три):

$$u(t) = \frac{V_k - V(t)}{T - t}. \quad (3.4)$$

Перевіримо, чи дійсно таке управління приводить об'єкт у потрібний кінцевий стан. Підставляємо (3.4) у диференціальне рівняння (3.1):

$$(T - t) \cdot \frac{d}{dt} V(t) + V(t) = V_k, \quad (3.5)$$

замінімо змінну:  $T - t = \tau$  і виконаємо підстановку Ейлера  $\tau = e^{-z}$ . В результаті отримуємо диференціальне рівняння з постійними коефіцієнтами  $\frac{d}{dz} V + V = V_k$ , що має аналітичне

розв'язання  $V(t) = V_0 \cdot e^{-z} + V_k$ . З початкових умов визначаємо постійну інтегрування:

$$V_0 = \frac{(V_0 - V_k)}{T},$$

робимо обернену підстановку і отримуємо розв'язання нестационарного рівняння (3.5):

$$V(t) = V_0 + \frac{V_k - V_0}{T} \cdot t.$$

Це розв'язання повністю збігається з розв'язанням для управління без ЗЗ. Згідно з першим законом проблемології, розв'язання однієї проблеми породжує, як мінімум, одну нову проблему. Управління з ЗЗ має неприємну "особливість" (математичний термін): в момент  $T$  управління стає нескінченним:

$$u(t) = (V_k - V(T)) \div (T - T).$$

Можливі чотири альтернативи усунення цієї особливості [1].

1. Відключення системи управління. Якщо вимоги до точності невисокі, можна в певний момент перед кінцем процесу відключити систему управління.

2. Відключення зворотного зв'язку. В певний момент перед кінцем процесу перейти на програмне управління.

3. Винесення кінцевої фазової точки за межі інтервалу  $(0, T)$ . Кінцевий стан у момент  $T + \Delta T$  задається так, щоб у момент  $T$  досягався заданий кінцевий стан об'єкта.

4. Метод "ведучої" фазової точки. В кожний момент часу для об'єкта задається нова кінцева точка. Ця "ведуча точка" рухається (в комп'ютерній програмі) по номінальній траєкторії переходу з певним часовим випередженням на  $\Delta T$  одиниць часу. Можна сказати, що об'єкт знаходиться в стані пошуку за "ведучою" фазовою точкою. Згідно з легендарними джерелами цей метод використовувався для заохочення запряженого віслюка, перед яким підвішувалась "ведуча точка" – сіно чи морква. Тому вибираємо останній метод, як перевірений часом і зрозумілий.

Отримаємо робочі формули для цього методу. Визначаємо номінальне управління:  $u = (V_k - V_0)/T$  і записуємо рівняння для швидкості руху ведучої точки на інтервалі часу  $(\Delta T, T + \Delta T)$ :

$$V_c(t) = V_0 + \frac{V_k - V_0}{T} \cdot (t + \Delta T). \quad (3.6)$$

Закон управління буде  $u = (V_c(t) - V(t)) \div \Delta T$ . Підставляємо в цей закон управління вираз для  $V_c$  і отримуємо:

$$u = k_0 + k_1 \cdot t + k_v \cdot V, \quad (3.7)$$

де  $k_0 = \frac{V_k - V_0}{T} + \frac{V_0}{\Delta T}$ ;  $k_1 = \frac{V_k - V_0}{T \cdot \Delta T}$ ;  $k_v = \frac{-1}{\Delta T}$ .

Величина  $\Delta T$  характеризує тривалість перехідних процесів стабілізації відхилень від програмної траєкторії, або довжину "поводка", яким об'єкт "прив'язаний" до ведучої точки. Назва цього параметра – "жорсткість" управління.

### 3.1.2 Синтез термінальної САУ для задачі приведення

Потрібно привести об'єкт в точку з координатою  $S_k$  в момент  $t = T$ . Маємо одну граничну умову, тому **потрібне управління** шукається у вигляді:  $u = C_0$ , двічі інтегруємо:

$$V(T) = \int_0^T C_0 dt, \quad V_k = V_0 + C_0 \cdot T; \quad V(T) = \int_0^T (V_0 + C_0 \cdot t) dt;$$

$$S_k = S_0 + V_0 \cdot T + \frac{C_0 \cdot T^2}{2} \text{ solve, } C_0 \rightarrow -2 \cdot \frac{-S_k + S_0 + V_0 \cdot T}{T^2} - \text{розв'язуємо відносно } C_0$$

і отримуємо вираз для термінального управління:



$$u = 2 \cdot \frac{(S_k - S_0) - V_0 \cdot T}{T^2}.$$

Швидкість в процесі розгону буде змінюватись згідно з рівнянням:

$$V(t) = -V_0 + \frac{V_k - V_0}{T} \cdot t.$$

**Введемо зворотний зв'язок:** як і в задачі розгону вважати поточний стан об'єкта початковим і для кожного поточного стану будемо розраховувати програму переходу в кінцевий стан за час  $(T - t)$ , що залишається до кінця процесу. Записуємо вираз для потрібного управління з ЗЗ:

$$u(t) = 2 \cdot \frac{(S_k - S(t))}{(T - t)^2} - \frac{V(t)}{(T - t)}. \quad (3.8)$$

Перевіримо, як і для задачі розгону, чи дійсно таке управління приводить об'єкт у потрібний кінцевий стан. Підставляємо управління (3.8) у диференціальне рівняння (3.1).

$$(T - t)^2 \cdot \frac{d^2 S}{dt^2} + 2 \cdot (T - t) \cdot \frac{d S(t)}{dt} + 2 \cdot S(t) = 2S_k,$$

замінімо змінну:  $T - t = \tau$  і виконаємо підстановку Ейлера  $\tau = e^{-z}$ . В результаті отримуємо диференціальне рівняння з постійними коефіцієнтами:

$$\frac{d^2 S}{dz^2} - 3 \cdot \frac{d S(t)}{dz} + 2 \cdot S = 2S_k,$$

що має аналітичний розв'язок. Якщо виконати досить об'ємні рутинні процедури (загальне розв'язання, визначення постійних інтегрування, повернення від змінної  $z$  до змінної  $t$ ), то отримаємо вираз, що при  $t = T$  дає потрібне значення координати

$$S(t) = S_0 + V_0 \cdot t + \left[ 2 \cdot \frac{(S_k - S_0) - V_0 \cdot T}{T^2} \right] \cdot \frac{t^2}{2}.$$

**Усунення особливості.** Використаємо метод "ведучої" фазової точки. Замінюємо у виразі для управління (3.8) кінцеві координати на координати "ведучої" фазової точки (а часу нам буде залишатись до кінця процесу не  $(T - t)$ , а завжди  $\Delta T$ ) і отримуємо закон управління без особливості:

$$u(t) = 2 \cdot \frac{(S_c - S(t))}{\Delta T^2} - \frac{V(t)}{\Delta T}. \quad (3.9)$$

Знаходимо вирази для координат "ведучої" точки, підставляємо в (3.9) вирази для  $V_c(t)$ ,  $S_c(t)$  і отримуємо:

$$u(t) = k_0 + k_1 \cdot t + k_2 \cdot t^2 + k_s \cdot S(t) + k_v \cdot V(t), \quad (3.10)$$

$$\text{де } k_0 = 2 \cdot \frac{S_k}{T^2} + \frac{(T - \Delta T)}{\Delta T \cdot T} \cdot \left[ 2 \cdot S_0 \cdot \frac{(T - \Delta T)}{\Delta T \cdot T} + 2 \cdot V_0 \right]; \quad k_1 = 2 \cdot \frac{V_0}{\Delta T} \cdot \left( \frac{1}{\Delta T} - \frac{2}{T} \right) + \frac{4 \cdot (S_k - S_0)}{\Delta T \cdot T^2};$$

$$k_2 = \frac{2 \cdot (S_k - S_0)}{\Delta T^2 \cdot T^2}; \quad k_s = \frac{-2}{\Delta T^2}; \quad k_v = \frac{-2}{\Delta T}.$$

### 3.1.3 Синтез термінальної САУ для задачі зближення

Розглядаємо замкнену в часі задачу зближення, коли треба отримати заданий кінцевий стан саме в заданий момент часу. Наприклад, якщо літак треба посадити на смугу раніше або пізніше заданого моменту часу, то там буде інший літак (в сучасних аеропортах інтервали між літаками – хвилини). Потрібно привести об'єкт в точку з координатами  $(S_k, V_k)$  саме в момент часу  $t = T$ . Маємо дві граничних умови, тому **потрібне управління** шукається у вигляді:

$$u = C_0 + C_1 \cdot t. \quad (3.11)$$

Неважко здогадатись, що формули в цій задачі будуть значно більшими, ніж в перших двох задачах (пункти 3.1.1, 3.1.2), тобто можливостей для помилок буде порівняно більше. Тому застосуємо "нову інформаційну технологію" (звичайно це означає використання Office) – формули буде виводити символічний процесор. Сучасні стандарти забороняють виконувати алгебраїчні перетворення вручну: помилки в програмах дорого коштують.

Двічі інтегруємо вираз для потрібного прискорення (3.11):

$$\int_0^T (C_0 + C_1 \cdot t) dt \rightarrow C_0 \cdot T + \frac{1}{2} \cdot C_1 \cdot T^2; \quad V(t) = V_0 + C_0 \cdot t + \frac{1}{2} \cdot C_1 \cdot t^2;$$

$$\int_0^T V(t) dt \rightarrow \int_0^T V(t) dt; \quad S(t) = S_0 + V_0 \cdot t + \frac{1}{2} \cdot C_0 \cdot t^2 + \frac{1}{6} \cdot C_1 \cdot t^3.$$

Підставляємо кінцеві умови і отримуємо систему двох рівнянь для визначення постійних  $C_0$  та  $C_1$ . Розв'язуємо систему у символічному вигляді.

$$\text{Given } C_0 \cdot T + \frac{1}{2} \cdot C_1 \cdot T^2 = V_k - V_0; \quad V_0 \cdot T + \frac{1}{2} \cdot C_0 \cdot T^2 + \frac{1}{6} \cdot C_1 \cdot T^3 = S_k - S_0.$$

$$\text{Find}(C_0, C_1) \rightarrow \begin{pmatrix} -2 \cdot \frac{2 \cdot V_0 \cdot T + T \cdot V_k - 3 \cdot S_k + 3 \cdot S_0}{T^2} \\ 6 \cdot \frac{V_0 \cdot T + T \cdot V_k - 2 \cdot S_k + 2 \cdot S_0}{T^3} \end{pmatrix}.$$

Записуємо вирази для  $C_0$ ,  $C_1$ , знаходимо помилки в першоджерелі (не було тоді символічного процесора) [1].

$$C_0 = \frac{6 \cdot (S_k - S_0)}{T^2} - \frac{(2 \cdot V_k + 4 \cdot V_0)}{T}; \quad C_1 = \frac{12 \cdot (S_0 - S_k)}{T^3} - \frac{6 \cdot (V_0 + V_k)}{T^2}. \quad (3.12)$$

**Введемо зворотний зв'язок:** як і в задачі розгону вважати поточний стан об'єкта початковим і для кожного поточного стану будемо розраховувати програму переходу в кінцевий стан за час  $(T - t)$ , що залишається до кінця процесу. Робимо в (3.11) такі заміни: підставляємо  $C_0, C_1$  з (3.11), замінюємо  $T \Rightarrow (T - t)$ ,  $V_0 \Rightarrow V(t)$ ,  $S_0 \Rightarrow S(t)$ .

Робимо це за допомогою символічного процесора (substitute – підставити).

$$C_0 + C_1 \cdot t \left| \begin{array}{l} \text{substitute } C_0 = \frac{6 \cdot (S_k - S_0)}{T^2} - \frac{(2 \cdot V_k + 4 \cdot V_0)}{T} \\ \text{substitute } C_1 = \frac{12 \cdot (S_0 - S_k)}{T^3} - \frac{6 \cdot (V_0 + V_k)}{T^2} \end{array} \right.$$

$$u(t) := \left[ \frac{6 \cdot S_k - 6 \cdot S_0}{T^2} - \frac{2 \cdot V_k + 4 \cdot V_0}{T} + \left( \frac{12 \cdot S_0 - 12 \cdot S_k}{T^3} - \frac{6 \cdot V_0 + 6 \cdot V_k}{T^2} \right) \cdot t \right]. \quad (3.13)$$

$$u(t) \text{ substitute } T = T - t \rightarrow \frac{6 \cdot S_k - 6 \cdot S_o}{(T - t)^2} - \frac{4 \cdot V_o + 2 \cdot V_k}{T - t} + \left[ \frac{12 \cdot S_o - 12 \cdot S_k}{(T - t)^3} - \frac{6 \cdot V_o + 6 \cdot V_k}{(T - t)^2} \right] \cdot t.$$

Звідки маємо вираз:

$$u(t) := \frac{6 \cdot (S_k - S(t))}{(T - t)^2} - \frac{2 \cdot (V_k + 2 \cdot V(t))}{T - t} \quad (3.14)$$

і фундаментальне запитання: куди поділася друга складова?

В даному випадку  $t$  – час від початку процесу, а у нас процес починається в кожен момент часу, тобто **у нас завжди  $t = 0$**  (як у відомому чаюванні).

**Усунення особливості.** Використаємо метод "ведучої" фазової точки. Замінюємо у виразі для управління (3.14) кінцеві координати на координати "ведучої" фазової точки (а часу нам буде залишатись до кінці процесу не  $(T - t)$ , а завжди  $\Delta T$ ) і отримуємо закон управління без особливості.

$$u(t) := \left[ \frac{6 \cdot (S_c - S(t))}{\Delta T^2} - \frac{2 \cdot (V_c + 2 \cdot V(t))}{\Delta T} \right]. \quad (3.15)$$

Запишемо вирази для координат "ведучої" фазової точки, що рухається по номінальній траєкторії з випередженням на  $\Delta T$

$$V_c(t) = V_o + C_o \cdot (t + \Delta T) + \frac{1}{2} \cdot C_1 \cdot (t + \Delta T)^2;$$

$$S_c(t) = S_o + V_o \cdot (t + \Delta T) + \frac{1}{2} \cdot C_o \cdot (t + \Delta T)^2 + \frac{1}{6} \cdot C_1 \cdot (t + \Delta T)^3.$$

Підставляємо це в (3.15) і після алгебраїчних перетворень отримуємо:

$$u(t) = k_0 + k_1 \cdot t + k_2 \cdot t^2 + k_3 \cdot t^3 + k_s \cdot S(t) + k_v \cdot V(t). \quad (3.16)$$

Між іншим, користування символьним процесором вимагає певних навичок, алгебраїчного мислення і просто терпіння. Спробуйте переставити порядок алгебраїчних операцій ("підставити", "спростити", "згрупувати") у виразі, що поданий нижче. Це фактично програма символьних перетворень. При користуванні символьним процесором виникають проблеми великих формул. Далі наведено програму символьних перетворень.

u(t)	$\text{substitute } V_c = V_o + C_o \cdot (t + \Delta T) + \frac{1}{2} \cdot C_1 \cdot (t + \Delta T)^2$ $\text{substitute } S_c = S_o + V_o \cdot (t + \Delta T) + \frac{1}{2} \cdot C_o \cdot (t + \Delta T)^2 + \frac{1}{6} \cdot C_1 \cdot (t + \Delta T)^3$ $\text{simplify}$ $\text{collect, } V(t)$ $\text{collect, } S(t)$ $\text{collect, } t$
------	---

Копіюємо вирази для коефіцієнтів закону управління, які отримані в результаті виконання програми символьних перетворень.

$$k_0 = \frac{C_o \cdot \Delta T^2 + 6 \cdot S_o + 4 \cdot V_o \cdot \Delta T}{\Delta T^2}; \quad k_1 = \frac{6 \cdot V_o + 4 \cdot C_o \cdot \Delta T + C_1 \cdot \Delta T^2}{\Delta T^2};$$

$$k_2 = \frac{3 \cdot C_o + 3 \cdot C_1 \cdot \Delta T}{\Delta T^2} - \frac{C_1}{\Delta T}; \quad k_3 = \frac{C_1}{\Delta T^2}; \quad k_v = \frac{-4}{\Delta T}; \quad k_s = \frac{-6}{\Delta T^2}.$$

## Висновки

Таким чином ми розв'язали задачу синтезу термінального управління для задачі зближення – маємо закон управління і вирази для параметрів закону управління як функцій граничних параметрів. Закон управління має програмну складову і зворотні зв'язки. Крім того, закон управління не має звичайної для простих методів термінального управління особливості біля кінцевого стану. Отриманий закон управління забезпечує максимум інтегрального критерію витрат енергії на управління. Крім цього, певним чином мінімізуються і зусилля, що діють на об'єкт в процесі переходу. Це важливо, коли об'єкти не допускають великих перевантажень.

Продемонстровано принципову можливість побудови і виробництва складних систем роботів маніпуляторів, за принципами на яких побудована сучасна мікроелектроніка. Можливе створення аналогічних за структурою і організацією монтажу, налагодженням та експлуатацією електронно-механічних систем класу роботів-маніпуляторів. Необхідна умова – використання модульних, децентралізованих інформаційно-обчислювальних систем, що візьмуть на себе дійсно важкі за старими мірками проблеми управління і адаптації до зовнішніх, внутрішніх (відмови елементів) збурень складних суттєво нелінійних систем.

## Узагальнення

Природно після висновків рухатись далі. Узагальнимо отримане. Ми послідовно розв'язали три задачі термінального управління і тепер можемо порівняти їх і зробити певні висновки та узагальнення відносно структури термінального управління. Випишемо (копіюємо) ці закони управління:

$$u(t) = k_0 + k_1 \cdot t + k_2 \cdot t^2 + k_v \cdot V(t) \text{ – розгін, одна гранична умова;}$$

$$u(t) = k_0 + k_1 \cdot t + k_2 \cdot t^2 + k_s \cdot S(t) + k_v \cdot V(t) \text{ – приведення, одна гранична умова;}$$

$$u(t) = k_0 + k_1 \cdot t + k_2 \cdot t^2 + k_3 \cdot t^3 + k_s \cdot S(t) + k_v \cdot V(t) \text{ – зближення, дві граничних умови.}$$

Дивлячись на ці закони неважко записати закон управління для наступної задачі:

$$u(t) = k_0 + k_1 \cdot t + k_2 \cdot t^2 + k_3 \cdot t^3 + k_4 \cdot t^3 + k_s \cdot S(t) + k_v \cdot V(t) \text{ – м'яке зближення, три граничних умови: } S(T) = S_k, V(T) = V_k, u(T) = u_k = 0.$$

Наступний розділ присвячений розробці програмного комплексу для цієї задачі.

## Контрольні запитання

1. Постановка задачі термінального управління.
2. Суть розглянутого методу термінального управління.
3. В якій формі шукається управління? Чи є воно оптимальним, за яким критерієм?
4. Структура закону управління кінцевим станом.
5. Назвіть 4–5 задач управління кінцевим станом (термінального управління).

## Завдання для самостійної роботи

1. Побудуйте графіки перехідних процесів для випадків, коли всі параметри регулятора: а) збільшені на 5%, б) зменшені на 5%.
2. Модифікуйте модель для врахування випадкових шумів у вимірювачах координати і швидкості.
3. Модифікуйте модель для врахування випадкових збурень (прискорень), що діють на об'єкт.
4. Модифікуйте модель для задачі м'якого зближення.

## 3.2 Дослідження термінальної САУ для задачі м'якого зближення

**Вступ. Постановка задачі.** В розділі 2 ми поставили і розв'язали задачу оптимального управління кінцевим станом певного динамічного об'єкта. В 95% підручників і монографій 95% задач, що розглядаються, це задачі управління поточним станом – регулювання температури, рівня, напруги, висоти польоту. В 50% масової літератури все обмежується методом ЛАЧХ і критерієм Гурвіца та ПІД-регулятором.

В нашому реальному оточенні більше половини задач – це задачі управління кінцевим станом: банальна задача стрільби, екзотична задача заправки паливом у повітрі, масова задача зупинки транспортного засобу (велосипеда, автомобіля, супертанкера) в потрібній точці, задача стикування космічних кораблів, задачі, що їх виконують підйомні крани, роботи-маніпулятори та ін.

Особливість таких задач, на відміну від задач управління поточним станом, в тому, що треба забезпечити досягнення саме потрібного кінцевого стану. В кожен момент часу управління повинно обчислюватись із умови досягнення саме кінцевого стану. Траєкторії руху до кінцевого стану можуть бути різними. Природно вибрати серед них оптимальну, відносно деякого критерію якості управління. Існувало приблизно чотири причини неуваги масової літератури з теорії і практики управління до задач термінального управління:

а) задачі термінального управління звичайно відносяться до класу варіаційних задач і вимагають більш глибоких знань з математики;

б) задачі термінального управління вимагають досить складних обчислень і не могли бути реалізовані в епоху ПІД-регуляторів;

в) в попередні 40 років системи термінального управління, з названих вище причин будувались переважно для дорогих і секретних об'єктів – ракет, космічних апаратів та ін.;

г) задачі термінального управління мають математичну, інформаційну і психологічну особливість – чим ближче до кінця процесу, тим важче виправити помилки: спочатку все виправити дуже легко, але помилки відомі дуже приблизно, потім помилки відомі точно, але вже пізно.

Неважко побачити, що пункт г) – це фактично постановка задачі прийняття рішень не тільки в техніці, але й у бізнесі. Тобто, класична задача термінального управління – здорова основа для розв'язання задач підтримки рішень там, де поки домінують (і не використовуються) "інтелектуальні системи підтримки рішень".

Звичайно, кінцеві помилки дорого коштують. Це "навали" (на причал – для кораблів), "наїзди" (для наземного транспорту), "козли" (в авіації і науці) з відповідними наслідками. Тобто існує широкий клас задач, де головним результатом є кінцевий стан. Тому система термінального управління повинна бути безпечною і відмовостійкою.

Саме з цих позицій будуємо програмні засоби:

– в даному розділі будуємо систему м'якого зближення, в якій кінцеві помилки координат є "нелетальними";

– в розділі 4 будуємо відмовостійку систему, в якій відмови паралельно працюючих виконавчих елементів є "нелетальними" (в тому числі і в прямому значенні).

Ставиться задача зробити програмну систему для синтезу термінального управління і моделювання термінальної системи.

Треба розробити програму для системи термінального управління, яка бере задані граничні умови переходу і повертає параметри закону термінального управління. Система термінального управління і виконує цей перехід.

Головна задача – розробити програму для моделювання двовимірного і тривимірних процесів, коли треба контролювати координати об'єкта в просторі. На рис. 3.1 подано схему одного каналу управління. Параметри об'єкта та вимоги до якості управління подано далі. Слід також подивитись на приклади реальних об'єктів – перейти в кінець розділу на сторінку 71.

### 3.2.1 Розробка модуля термінального управління

Згідно з аксіомами загальної теорії систем:

а) "системи синтезуються і конструюються", тобто, потенційно можна зробити цілеорієнтовану систему для будь-якої задачі (навіть якщо ви свідомо не конструюєте системи, це буде робитись стихійно, з гарантованими небажаними наслідками);

б) "системи синтезуються частинами" – це немовби очевидна банальність (системний аналітик розбиває складну задачу або на функціональні кубики (цеглини), з яких потім складає систему, або в послідовність редукційних кубиків, які поступово ускладнюються).

В цьому розділі розглядається задача управління спочатку для однієї координати (одного каналу управління системи), без урахування обмежень. Потім неважко буде розробити двоканальну систему управління, систему з урахуванням обмежень на траєкторію. Послідовність пунктів розв'язання задачі відповідає типовій послідовності розробки реальних систем управління. З доступних альтернатив побудови САУ вибираємо *термінальне управління, задачу зближення*, для дуже відповідальних випадків – *задачу м'якого зближення*.

Одна з проблем реалізації задачі – точне вимірювання відстаней, швидкостей. Для систем причалювання, приземлення потрібно контролювати дві координати – X та Y, швидкості та прискорення за цими координатами. Для роботів-маніпуляторів потрібно контролювати три координати. Ці величини одночасно повинні стати = 0 в заданій точці. Канали управління цими координатами ідентичні, далі розглядаємо тільки один канал. Структурна схема системи термінального управління подана на рис. 3.1.

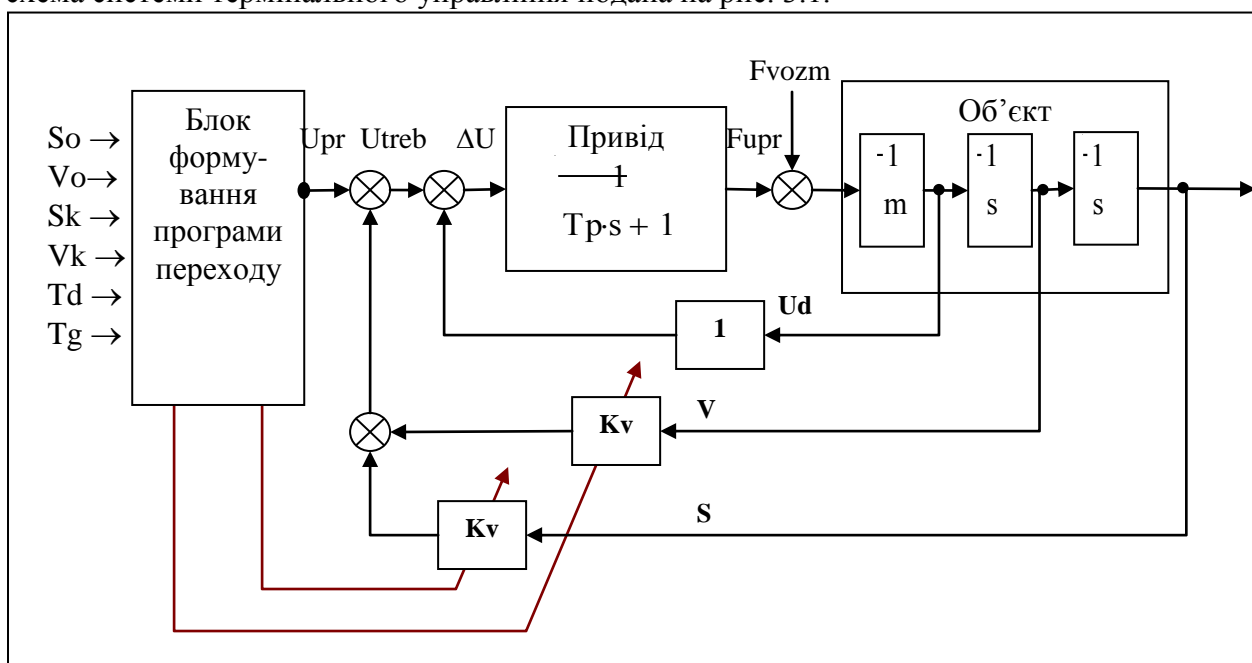


Рис. 3.1. Схема одного каналу системи термінального управління

В розділі 3.1 визначили структуру і параметри таких задач термінального управління: – зближення, дві граничних умови  $S(T) = S_k$ ,  $V(T) = V_k$ ,

$$u(t) = k_0 + k_1 \cdot t + k_2 \cdot t^2 + k_3 \cdot t^3 + k_s \cdot S(t) + k_v \cdot V(t);$$

– м'яке зближення, три граничних умови:  $S(T) = S_k$ ,  $V(T) = V_k$ ,  $u(T) = u_k = 0$ ,

$$u(t) = k_0 + k_1 \cdot t + k_2 \cdot t^2 + k_3 \cdot t^3 + k_4 \cdot t^3 + k_s \cdot S(t) + k_v \cdot V(t).$$

Беремо з розділу 2 розрахункові формули для нашої задачі (порядок об'єкта  $n=2$ , число граничних умов  $m=2$  (задача зближення). Вводимо (там далі на стенді, а тут тільки для послідовності викладення) параметри задачі.

Постійна часу приводу  $T_{p1} = 0.2$ . Задана тривалість процесу  $T_d = 15$ . Жорсткість управління  $T_{gl} = 4$ . Крок моделювання  $T = 1$ . ORIGIN:= 1. Задані граничні умови: початкові  $so1 = 4$ ;  $vo1 = 0.2$ ; кінцеві  $sk1 = 0.1$ ;  $vk1 = 15$ .

Робимо модуль для визначення жорсткості управління. Стартове значення шуканої змінної  $T_{gg} := 1$ .

**Give1**

$0.01 \cdot T_{dd} \leq T_{gg} \leq .3T_{dd}$  – жорсткість повинна бути менше третини тривалості процесу;

$2 \cdot T_{pp} \leq T_{gg} \leq 5T_{pp}$ ;  $T_{gg} > 0$  – жорсткість повинна бути в 2-3 рази більшою тривалості перехідного процесу;

$T_{g}(T_{dd}, T_{pp}) := \text{Find}(T_{gg})$  – визначаємо в термінах розв'язувального блока функцію користувача, яка бере тривалість процесу і постійну часу приводу.

Перевіряємо цю функцію:  $T_{g}(10, 0.1) = 0.2$ .

Записуємо функції користувача – вирази для параметрів закону термінального управління (див. підрозділ 3.1)

$U_k := 0$ ;  $V_k := 0$ ;  $S_k := 0$

$$C0(So, Vo, Td) := \frac{-12 \cdot So}{T_d^2} + \frac{12 \cdot S_k}{T_d^2} - \frac{6 \cdot Vo}{T_d} - \frac{6 \cdot V_k}{T_d} + U_k;$$

$$C1(So, Vo, Td) := \frac{48 \cdot So}{T_d^3} - \frac{48 \cdot S_k}{T_d^3} + \frac{18 \cdot Vo}{T_d^2} + \frac{30 \cdot V_k}{T_d^2} - \frac{6 \cdot U_k}{T_d};$$

$$C2(So, Vo, Td) := \frac{-36 \cdot So}{T_d^4} + \frac{36 \cdot S_k}{T_d^4} - \frac{12 \cdot Vo}{T_d^3} - \frac{24 \cdot V_k}{T_d^3} + \frac{6 \cdot U_k}{T_d^2};$$

$$K0(So, Vo, Tg, Td) := \frac{12So}{T_g^2} + \frac{6Vo}{T_g} + C0(So, Vo, Td);$$

$$K1(So, Vo, Tg, Td) := \frac{12Vo}{T_g^2} + \frac{6C0(So, Vo, Td)}{T_g} + C1(So, Vo, Td);$$

$$K2(So, Vo, Tg, Td) := \frac{6 \cdot C0(So, Vo, Td)}{T_g^2} + \frac{3C1(So, Vo, Td)}{T_g} + C2(So, Vo, Td);$$

$$K3(So, Vo, Tg, Td) := \frac{2C1(So, Vo, Td)}{T_g^2} + \frac{2 \cdot C2(So, Vo, Td)}{T_g}$$

$$K4(So, Vo, Tg, Td) := \frac{C2(So, Vo, Td)}{T_g^2}; \quad K_v(Tg) := \frac{-6}{T_g}; \quad K_s(Tg) := \frac{-1}{T_g}.$$

### 3.2.2 Розробка математичної моделі і програми моделювання САУ

Запишемо спочатку систему диференціальних рівнянь (неперервну математичну модель) термінальної САУ. Для базової моделі маємо:

$$a_2 \cdot \frac{d^2}{dt^2} Y(t) + a_1 \cdot \frac{d}{dt} Y(t) + a_0 \cdot Y(t) = ud(.) + zb(t).$$

Для базової моделі:  $a_2 = 10$ ;  $a_1 = 0$ ;  $a_0 = 0$ , тому маємо:

$$\frac{d^2}{dt^2} Y(t) = \frac{1}{a_2} \cdot (ud(.) + zb(t)).$$

Врахуємо інерційність приводу, замінюємо:  $Y = s$ ;  $dY/dt = v$  – для узгодження з літературою [5] і отримуємо:

$$\frac{d}{dt} ud(t) = \frac{1}{T_p} \cdot (utr(t) - ud(t)); \quad \frac{d}{dt} v(t) = \frac{1}{a_2} \cdot (ud(.) + zb(t)); \quad \frac{d}{dt} s(t) = v(t);$$

$$utr(t) = upr(t) + k_2 \cdot v(t) + k_1 \cdot s(t).$$

В цих рівняннях  $utr(t)$ ,  $ud(t)$  – потрібне і дійсне управління,  $upr(t)$  – програмна складова потрібного управління.

Для аналізу і синтезу САУ ми використовуємо ЦОМ – дискретний пристрій, ЦОМ використовується також і для реалізації САУ. Тому переходимо до еквівалентної дискретної математичної моделі. В пакеті більш ніж достатньо вбудованих функцій числового інтегрування диференціальних рівнянь. Це зручні "чорні ящики", а нам потрібне щось прозоре і безвідмовне. Подамо задачу у вигляді канонічної матрично-векторної системи диф-рівнянь. Введемо вектор стану  $x$  з компонентами:  $x_1 = s$ ;  $x_2 = v$ ;  $x_3 = ud$ .

Підставимо у дифрівняння приводу вираз для управління у вигляді:

$$utr(t) = upr(t) + k_2 \cdot x_2(t) + k_1 \cdot x_1(t).$$

Запишемо систему рівнянь САУ у формі Коші:

$$\left. \begin{aligned} dx_1/dt &= 0 \cdot x_1 + 1 \cdot x_2 + 0 \cdot x_3; \\ dx_2/dt &= 0 \cdot x_1 + 0 \cdot x_2 + 1 \cdot x_3; \\ dx_3/dt &= K_s/T_p \cdot x_1 + K_v/T_p \cdot x_2 - 1/T_p \cdot x_3 + 1/T_p \cdot u(t). \end{aligned} \right\} \quad (3.17)$$

Запишемо рівняння системи у матрично-векторному вигляді: введемо матриці:

$$I := \text{identity}(3); \quad I = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}; \quad A(T_p, T_g) := \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ \frac{K_s(T_g)}{T_p} & \frac{K_v(T_g)}{T_p} & \frac{-1}{T_p} \end{pmatrix}; \quad B(T_p) := \begin{pmatrix} 0 \\ 0 \\ \frac{1}{T_p} \end{pmatrix}.$$

Виділяємо програмну (залежну від часу) складову в термінальному управлінні

$$u(s_0, v_0, t, T_g) := K_0(s_0, v_0, T_g) + K_1(s_0, v_0, T_g) \cdot t + K_2(s_0, v_0, T_g) \cdot t^2 + K_3(s_0, v_0, T_g) \cdot t^3.$$

Тепер отримуємо еквівалентну дискретну математичну модель (ЕДМ) термінальної САУ. Запишемо перехідні матриці  $F(\cdot)$  і  $G(\cdot)$  для ЕДМ

$$F(T, T_p, T_g) := I + A(T_p, T_g) \cdot T + \frac{(A(T_p, T_g) \cdot T)^2}{2}; \quad G(T, T_p, T_g) := I \cdot T + \frac{A(T_p, T_g) \cdot T^2}{2}. \quad (3.18)$$

### 3.2.3 Двовимірна задача зближення

Ми розробили документи і виконали дослідження для випадку управління однією координатою. Тепер зробимо програму та інтерфейс для моделювання. Початковий і стартовий стани. Параметри системи (подаємо розшифровку)

$$X_{sf} = \begin{pmatrix} -7 & 7 \\ 0 & 0 \\ 15 & 10 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} vo1 & vo2 \\ vk1 & vk2 \\ so1 & so2 \\ sk1 & sk2 \end{pmatrix}; \quad P_t = \begin{pmatrix} 0.2 & 0.2 \\ 15 & 15 \\ 4 & 4 \\ 0.1 & 0.1 \end{pmatrix} \begin{pmatrix} T_{p1} & T_{p2} - \text{"пост. часу приводів"} \\ T_d & T_d - \text{"тривалість процесу"} \\ T_{g1} & T_{g2} - \text{"жорсткості у управлінні"} \\ T & T - \text{"крок обчислень"} \end{pmatrix}.$$

За прототипи беремо програмні модулі для синтезу термінального управління однією координатою. Модуль синтезу залишаємо без принципів змін, зробимо лише декілька методично-косметичних змін. Для економії (слів, інтелекту) подаємо поряд прототип. Порівняйте присвоєння, поясніть навіщо робити вихід рядком, а потім транспонувати у стовпець?

Модуль моделювання робимо на базі прототипу (подано праворуч для порівняння) за принципом "те ж саме, але вдвічі більше". Робимо певні косметичні зміни (можна це назвати "підвищуємо ефективність програми"). Оптимізуємо програму за двома критеріями "працевдатність", "читабельність". Саме для читабельності ми подаємо рівняння програми



в "природних позначеннях" – як у літературі [5] ( $T_p$ ,  $T_d$ ,  $K_1$ ,  $K_s$ ,  $K_v$ , ...).

Відповідні присвоєння складають половину тексту. Ще третина тексту – формування відповідних матриць для неперервної і еквівалентної дискретної моделей динаміки. Останні 15% тексту – цикл, власне моделювання динаміки. Теоретична основа програми:

$$\text{неперервна модель: } \frac{d}{dt}x(t) = A \cdot x(t) + B \cdot (U_{pt}(t) + u_{os}(x(t)))$$

$$\text{та еквівалентна дискретна модель: } x^{(k+1)} = F \cdot x^{(k)} + G \cdot B \cdot U_{pt}.$$

Означення і отримання цих рівнянь розглянуті в попередніх і в даному підрозділах – вище. Звернемо увагу на те, куди зникла в рівняннях складова зворотного зв'язку:

$$u_{os}(x(t)) = K_s \cdot x(t) + K_v \cdot \frac{d}{dt}x(t) \text{ – ми включили її до матриці стану } A.$$

На рис 3.2 подано модуль синтезу термінального управління.

Syn(Par, Xo) :=	$v_0 \leftarrow X_{o1}$ $v_k \leftarrow X_{o2}$ $s_0 \leftarrow X_{o3}$ $s_k \leftarrow X_{o4}$ $T_p \leftarrow Par_1$ $T_d \leftarrow Par_2$ $T_g \leftarrow Par_3$ $T \leftarrow Par_4$ $c_0 \leftarrow \frac{-12 \cdot s_0}{T_d^2} + \frac{12 \cdot s_k}{T_d^2} - \frac{6 \cdot v_0}{T_d} - \frac{6 \cdot v_k}{T_d} + 0$ $c_1 \leftarrow \frac{48 \cdot s_0}{T_d^3} - \frac{48 \cdot s_k}{T_d^3} + \frac{18 \cdot v_0}{T_d^2} + \frac{30 \cdot v_k}{T_d^2} - \frac{6 \cdot 0}{T_d}$ $c_2 \leftarrow \frac{-36 \cdot s_0}{T_d^4} + \frac{36 \cdot s_k}{T_d^4} - \frac{12 \cdot v_0}{T_d^3} - \frac{24 \cdot v_k}{T_d^3} + \frac{6 \cdot 0}{T_d^2}$ $K_0 \leftarrow \frac{12 \cdot s_0}{T_g^2} + \frac{6 \cdot v_0}{T_g} + c_0$ $K_1 \leftarrow \frac{12 \cdot v_0}{T_g^2} + \frac{6 \cdot c_0}{T_g} + c_1$ $K_2 \leftarrow \frac{6 \cdot c_0}{T_g^2} + \frac{3 \cdot c_1}{T_g} + c_2$ $K_3 \leftarrow \frac{2c_1}{T_g^2} + \frac{2 \cdot c_2}{T_g}$ $K_4 \leftarrow c_2 \div T_g^2$ $K_v \leftarrow -6 \div T_g$ $K_s \leftarrow -12 \div T_g^2$ $qq \leftarrow (K_0 \ K_1 \ K_2 \ K_3 \ K_4 \ K_v \ K_s)$ $(qq)^T$
-----------------	---

Рис. 3.2. Текст модуля синтезу параметрів термінального управління

На рис. 3.3 подано текст програми. Бачимо, що програма бере два масиви – масив параметрів Pr та масив граничних умов Xs. Програма викликає двічі модуль синтезу параметрів законів управління – для розрахунку параметрів законів управління, для кожного каналу управління. Між іншим, в новій версії пакета довгі стовпці присвоювання можна замінити одним векторним присвоєнням. Пам'ятаємо що тексти "програм" в математичному пакеті – це тільки інтерфейси для управління відповідними модулями на мові C++.

$\text{Pr}(\text{Pt}, \text{Xs}) :=$ $\text{by} \leftarrow \text{Syn}(\text{Pt}^{\langle 1 \rangle}, \text{Xsf}^{\langle 1 \rangle})$ $\text{K01} \leftarrow \text{by}_1$ $\text{K11} \leftarrow \text{by}_2$ $\text{K21} \leftarrow \text{by}_3$ $\text{K31} \leftarrow \text{by}_4$ $\text{K41} \leftarrow \text{by}_5$ $\text{Kv1} \leftarrow \text{by}_6$ $\text{Ks1} \leftarrow \text{by}_7$ $\text{my} \leftarrow \text{Syn}(\text{Pt}^{\langle 2 \rangle}, \text{Xsf}^{\langle 2 \rangle})$ $\text{K02} \leftarrow \text{my}_1$ $\text{K12} \leftarrow \text{my}_2$ $\text{K22} \leftarrow \text{my}_3$ $\text{K32} \leftarrow \text{my}_4$ $\text{K42} \leftarrow \text{my}_5$ $\text{Kv2} \leftarrow \text{my}_6$ $\text{Ks2} \leftarrow \text{my}_7$ $\text{Td} \leftarrow \text{Pt}_{2,1}$ $\text{T} \leftarrow \text{Pt}_{4,1}$ $\text{Nkr} \leftarrow \text{Td} \div \text{T}$ $\text{Tp1} \leftarrow \text{Pt}_{1,1}$ $\text{Tp2} \leftarrow \text{Pt}_{1,2}$ $\text{xx1} \leftarrow (\text{Xs}_{3,1} \quad \text{Xs}_{1,1} \quad 0 \quad \text{Xs}_{3,2})$ $\text{x}^{\langle 1 \rangle} \leftarrow \text{xx1}^T$	$\text{B} \leftarrow \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 \div \text{Tp1} & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \div \text{Tp2} \end{pmatrix}$ $\text{A} \leftarrow \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ \frac{\text{Ks1}}{\text{Tp1}} & \frac{\text{Kv1}}{\text{Tp1}} & \frac{-1}{\text{Tp1}} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & \frac{\text{Ks2}}{\text{Tp2}} & \frac{\text{Kv2}}{\text{Tp2}} & \frac{-1}{\text{Tp2}} \end{pmatrix}$ $\text{F} \leftarrow \text{identity}(6) + \text{A} \cdot \text{T} + \frac{\text{A}^2 \cdot \text{T}^2}{2}$ $\text{G} \leftarrow \text{identity}(6) \cdot \text{T} + \frac{\text{A} \cdot \text{T}^2}{2}$ $\text{for } k \in 1 \dots \text{Nkr}$ $\quad \text{t} \leftarrow k \cdot \text{T}$ $\quad \text{Uptr} \leftarrow \begin{pmatrix} \text{K01} + \text{K11} \cdot \text{t} + \text{K21} \cdot \text{t}^2 + \text{K31} \cdot \text{t}^3 + \text{K41} \cdot \text{t}^4 \\ \text{K02} + \text{K12} \cdot \text{t} + \text{K22} \cdot \text{t}^2 + \text{K32} \cdot \text{t}^3 + \text{K42} \cdot \text{t}^4 \end{pmatrix}$ $\quad \text{ud} \leftarrow 0$ $\quad \text{x}^{\langle k+1 \rangle} \leftarrow \text{F} \cdot \text{x}^{\langle k \rangle} + \text{G} \cdot \text{B} \cdot \text{Uptr}$ $\text{x}$
--	--

Рис. 3.3. Програма моделювання термінальної САУ

На рис. 3.3 текст програми розрізано на дві частини: перша – синтез параметрів законів управління, друга – власне моделювання. За незрозумілою традицією в текстах публікацій і студентських роботах подають математичні вирази, блок-схеми "алгоритмів програм", а власне тексти програм виносять в додатки. Можемо бачити (рис. 3.3), що текст програми містить фактично "природні" формули, що складають математичну модель. Текст програми економно структурований лініями, і тому є фактично блок-схемою програми. Відзначимо також, що програма просто примітивна за структурою – абсолютно "не інтелектуальна". За рахунок низки тривіальних присвоєнь (в першій частині підрозділу ми отримали "читабельні" математичні вирази, в другій частині – програми). Розпаковуємо вихід програми  $\text{Dup} := \text{Pr}(\text{Pt}, \text{Xsf})^T$ , вводимо зрозумілі для людини позначення

$s1_k := Dup_{k,1}$ ;  $s2_k := Dup_{k,4}$ ;  $v1_k := Dup_{k,2}$ ;  $v2_k := Dup_{k,5}$ ;  $u1_k := Dup_{k,3}$ ;  $u2_k := Dup_{k,6}$ .  
 Будуємо графіки перехідних процесів (рис. 3.4, 3.5).

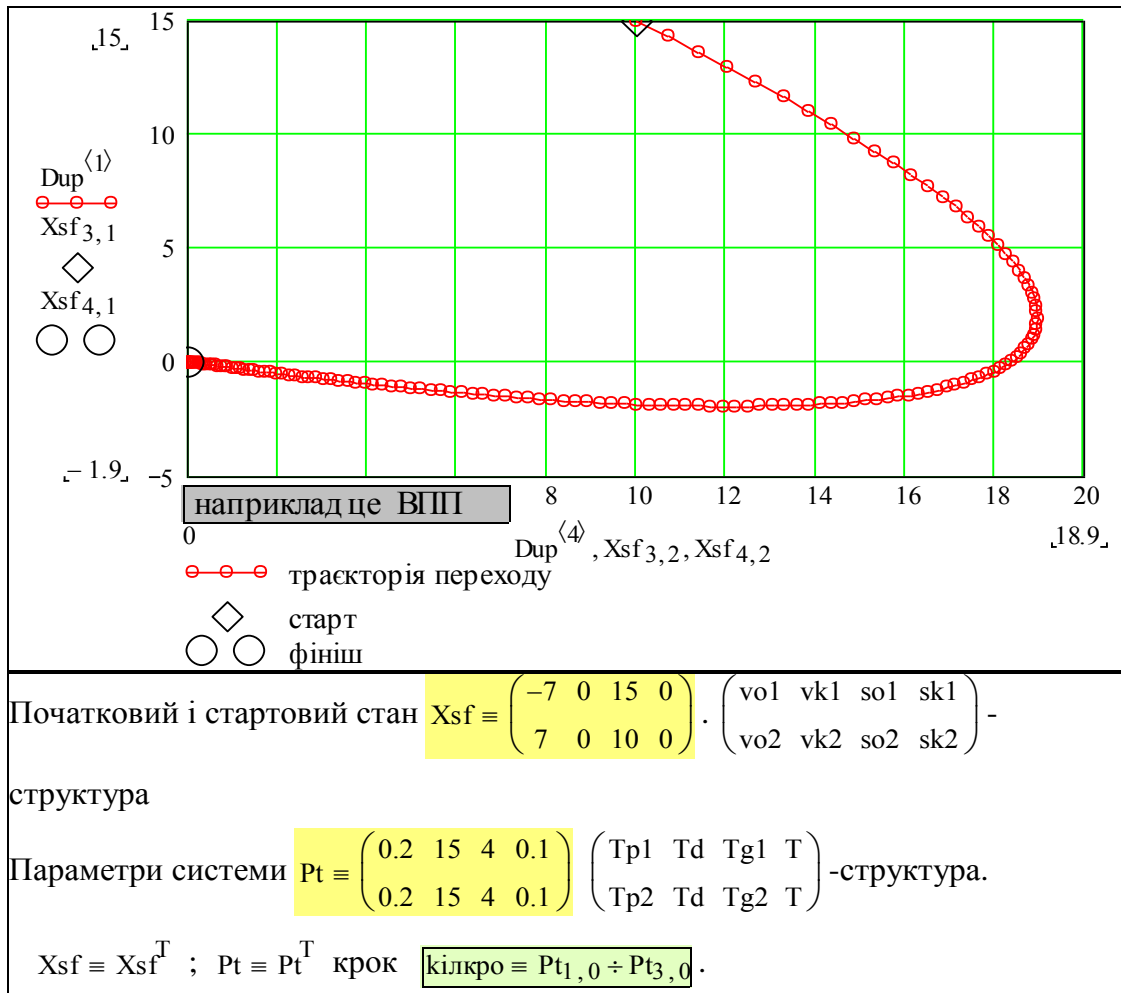


Рис. 3.4. Стенд для аналізу процесів термінального управління. Частина 1

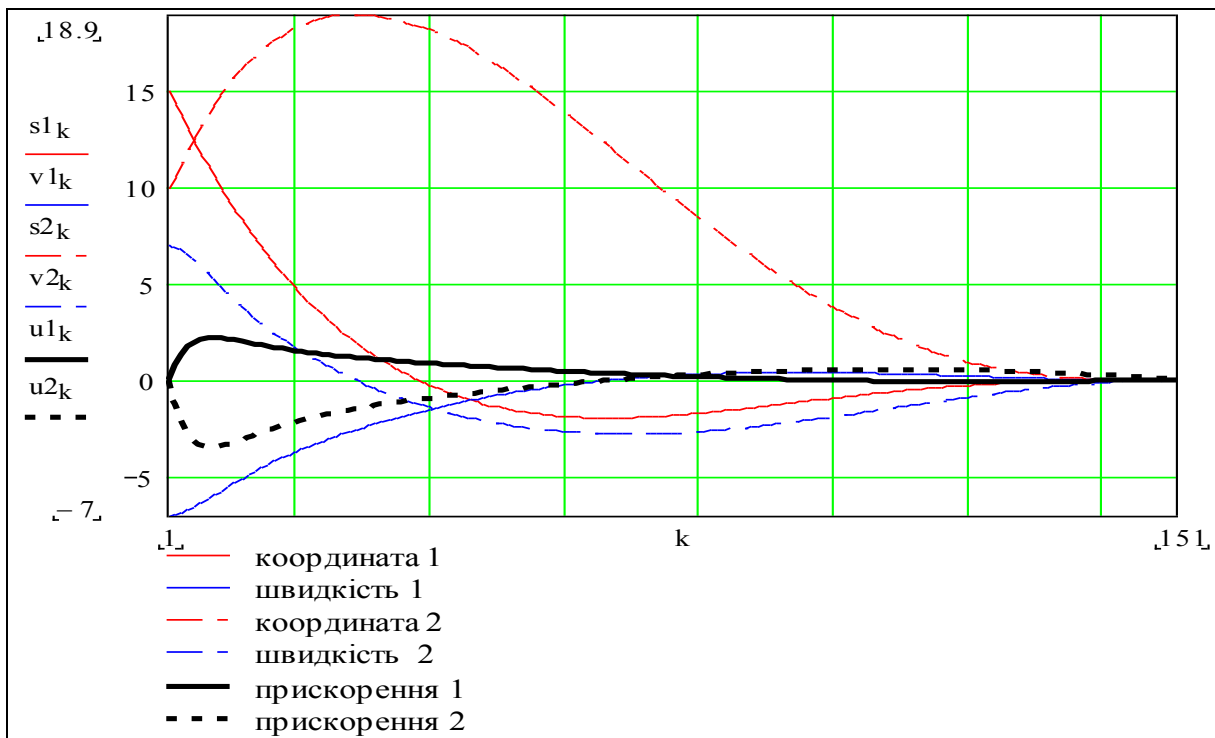


Рис. 3.5. Стенд для аналізу процесів термінального управління. Частина 2

Дивимось на рис.3.4, 3.5, бачимо:

а) програма коректна і працездатна, бо приводить об'єкт з будь-якого початкового стану в заданий кінцевий стан і заданий момент часу;

б) програма не враховує обмежень на поточні координати конкретного об'єкта.

Якщо це траєкторія заходу на посадку, то літак значну частину її проходить під землею. Такі обмеження неважко врахувати. Але, згідно з основним постулатом теорії систем, "не можна отримати велику перемогу одразу, все повинно розвиватись поступово".

**Завдання для самостійного виконання.** Введіть в моделі синтезу термінального управління і моделювання обмеження на припустиму траєкторію. Це немовби легко зробити. Але чи буде процес оптимальним? Підказка "системи синтезуються частинами".

На рис. 3.5 подано два процеси з різними початковими умовами. Бачимо особливість системи м'якого зближення: прискорення поступово спадає до нуля, тобто об'єкт при невеликих невизначеностях і помилках управління "не вдариться об термінальний стан". Обчислимо ще один процес для побудови трьох траєкторій з різними початковими станами.

$$Pt2 := \begin{pmatrix} 0.2 & 0.2 \\ 15 & 15 \\ 3 & 3 \\ 0.1 & 0.1 \end{pmatrix}; \quad Xs2 := \begin{pmatrix} -8 & -5 \\ 0 & 0 \\ 10 & 5 \\ 0 & 0 \end{pmatrix}; \quad Pt3 := \begin{pmatrix} 0.2 & 0.2 \\ 15 & 15 \\ 5 & 5 \\ 0.1 & 0.1 \end{pmatrix}; \quad Xs3 := \begin{pmatrix} -8 & 0 \\ 0 & 0 \\ 6 & 15 \\ 0 & 0 \end{pmatrix};$$

$$Du2 := Pr(Pt2, Xs2)^T; \quad S1_k := Du2_k, 1; \quad S2_k := Du2_k, 4; \quad V1_k := Du2_k, 2; \quad V2_k := Du2_k, 5;$$

$$U1_k := Du2_k, 3; \quad U2_k := Du2_k, 6; \quad Du3 := Pr(Pt3, Xs3)^T; \quad ss1_k := Du3_k, 1; \quad ss2_k := Du3_k, 4;$$

$$vv1_k := Du3_k, 2; \quad vv2_k := Du3_k, 5; \quad uu1_k := Du3_k, 3; \quad uu2_k := Du3_k, 6$$

Рис. 3.6. Стенд для аналізу процесів термінального управління. Частина 3

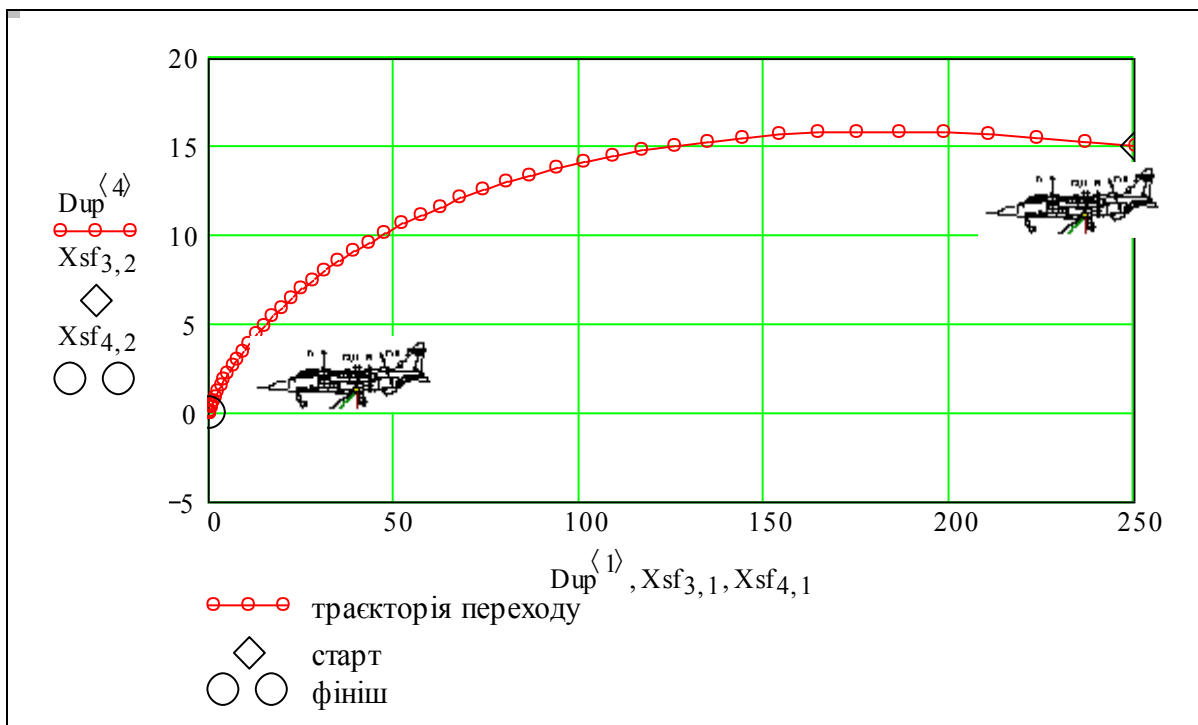


Рис. 3.7. Фазова траєкторія оптимального зближення. Приклад

На рис. 3.8 подано дві траєкторії перехідних процесів в координатах: час, відстань від кінцевої точки, висота над кінцевою точкою. Траєкторії відрізняються тільки початковими станами. Бачимо, що з будь-якого початкового стану система термінального

управління приводить об'єкт, в потрібний момент часу, до заданого кінцевого стану. Бачимо також, що кінцеві ділянки траєкторій збігаються, і в кінці траєкторія стає специфічною: прискорення асимптотично зменшується до нульового і об'єкт "підкрадається" (професійна термінологія) до кінцевого стану.

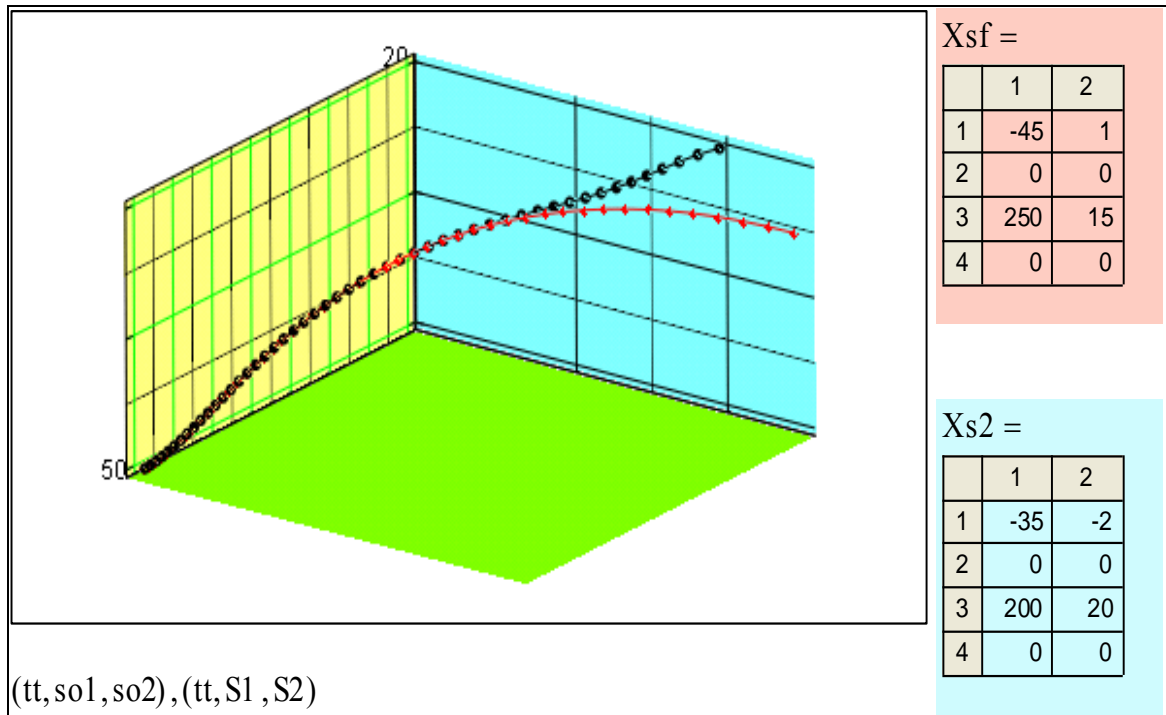
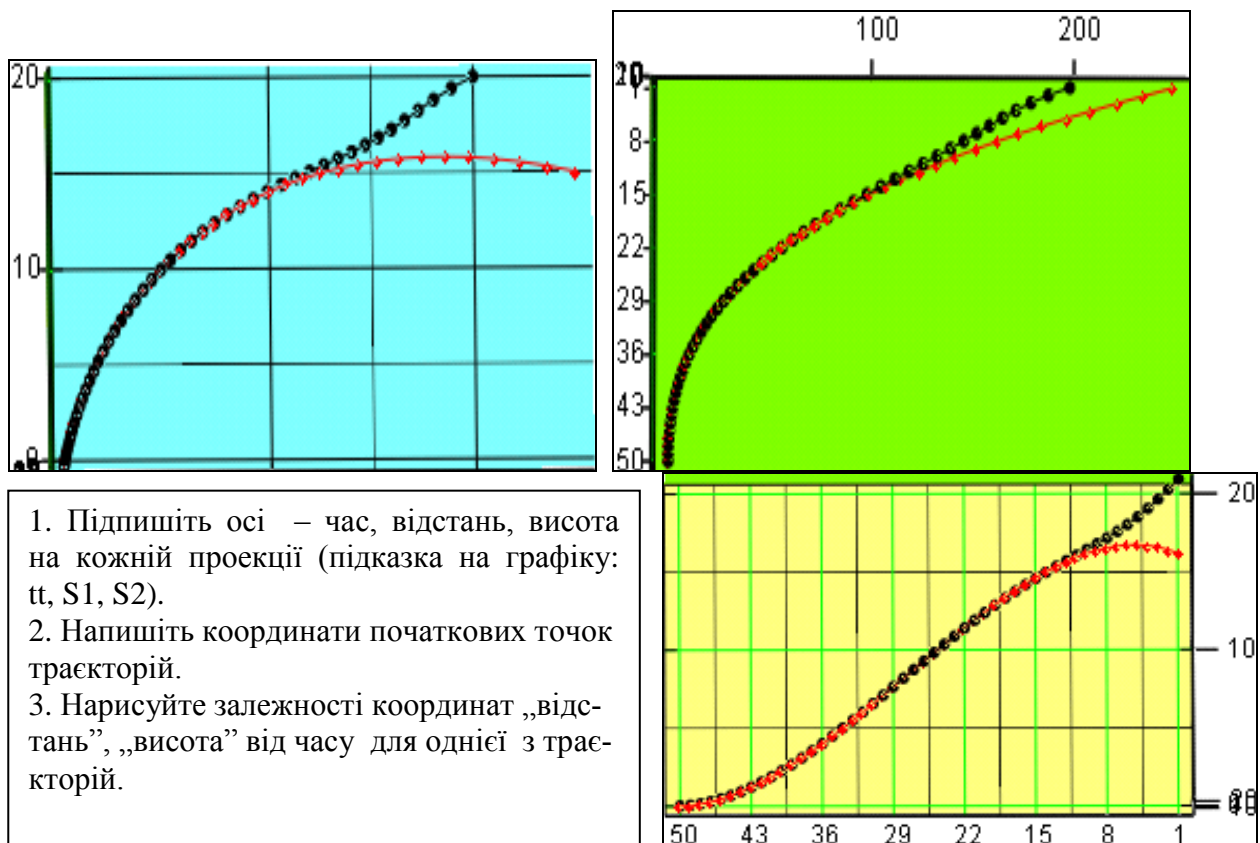


Рис. 3.8. Фазова траєкторія оптимального зближення. Приклад

**Завдання для самостійного виконання №1.** Нижче подано три проекції тривимірного графіка (рис. 3.8). Гортаючи попередні сторінки, знайдіть відповіді на ряд питань.



### Контрольні запитання

1. Назвіть 4-5 задач термінального управління.
2. Наведіть приклади об'єктів термінального управління.
3. Наведіть приклади двовимірних, тривимірних задач термінального управління.
4. Що є управлінням в задачі м'якого зближення?
5. В чому суть методу "ведучої" фазової точки?

### Завдання для самостійного виконання

1. Побудуйте графіки перехідних процесів для випадків, коли всі параметри регулятора: а) збільшені на 5%, б) зменшені на 5%.
2. Зробіть програму для моделювання тривимірного процесу (координати – відстань, висота, бокове відхилення).
3. Зробіть програму для процесу з довільною розмірністю.

### Висновки

На прикладі конкретної задачі подано технології розробки робочої моделі. Може здатись дивним, що довга низка формул, на базі яких створена програма, виявилась безпомилковою – система термінального управління дійсно приводить об'єкт до заданого кінцевого стану.

Створені програмні модулі придатні (після реалізації в мікроконтролері) для вбудовування в системи управління причалуванням, ліфти для президентів, системи посадки на палубу авіаносця в негоду, системи управління підйомними кранами, що розвантажують і завантажують важливі для народного господарства вантажі: яйця дієтичні, посуд скляний та ін. Майже неважко розробити документ для синтезу і моделювання термінального управління в тривимірному просторі, або з урахуванням обмежень на траєкторію переходу.

Список об'єктів термінального управління не вичерпується задачею – літаком вертикального зльоту і посадки чи мостовим краном. Саме управління кінцевим станом потрібне для таких об'єктів і процесів:

- роботи-маніпулятори;
- інвестиційні проекти розвитку виробничих систем;
- енергоекономічне управління рухом поїздів;
- "автопілоти" для автомобілів – дотримання безпечної дистанції та ін.
- причалування кораблів;
- вирощування сільгоспкультур за інтенсивними технологіями.

Список задач управління кінцевим станом не закінчується задачею м'якого зближення. Розглянута технологія і розроблені програмні модулі можуть бути використані для розв'язання таких актуальних задач [5]:

- задача зближення з іншим рухомих об'єктом;
- задача зустрічі двох і більше активних об'єктів;
- задача узгодженого руху декількох об'єктів;
- задача оптимального управління об'єктом з довільною динамікою;
- задача узгодженого управління "колективом" паралельно працюючих приводів в системі термінального управління.

Саме така задача розглядається в наступному підрозділі.

### 3.3 Розробка і дослідження термінальної САУ стійкої до відмов виконавчих елементів

Особливість реальних термінальних САУ – великі втрати при відмовах в системі управління. Дійсно – поїзд чи корабель не зупинився в потрібному місці, кран не тільки розбив контейнер, але ще й опустив його на інший контейнер з чимось легкозаймистим. Імовірнісна (совкова) теорія надійності може дати тільки ритуальні розрахунки "середнього часу безвідмовної роботи" та ін. На практиці систему роблять відмовостійкою за рахунок неідентичного резервування критичних елементів – сенсорів, приводів, а ці критичні елементи роблять відмовобезпечними.

Відмовобезпечність означає, що при відмові електропривод, наприклад, не вибухає, не горить, не заважає роботі інших приводів. Щодо звичайної надійності, яку оцінювали ймовірностями відмови, часом до першої відмови, то високі технології сучасного виробництва зробили ритуальними розрахунки традиційних показників, дійсно, розрахунок показує ймовірність відмови приводу 0.9999999999999999? Реальна відмова може бути спричинена порушенням правил експлуатації, терактом, стихійним лихом і прихованими "до вирішального моменту" помилками проектування і виготовлення.

Теорія автоматичного управління дозволяє робити так звані "ультранадійні" системи за рахунок вибору алгоритмів і методів управління з певними властивостями. До такого класу належить патентований метод термінального управління, де об'єктом регулювання є вища похідна в диференціальному рівнянні об'єкта управління.

#### Постановка задачі

Розглянемо приклад побудови та випробування відмовостійкої системи термінального управління певним об'єктом чи процесом. Це може бути будь-який процес, де в заданий кінцевий момент часу треба мати потрібний стан. Наприклад: вирощування сільгосппродукції, виконання операції роботом-маніпулятором, причалювання танкера, посадка ЛВЗП і просто перенесення вантажу краном. Неважко побачити і зрозуміти, чому і для чого в цих процесах задіяні паралельні виконавчі елементи.

Згідно з завданням в одному з дипломних проектів розроблено рекламний дайджест – одна сторінка з характеристиками термінальної системи – "на продаж", для спеціалістів.

#### Обґрунтування вибору методу термінального управління

Відомі такі альтернативні методи термінального управління:

- 1) модифікації управління поточним станом;
- 2) пряме розв'язання варіаційної задачі з локальним критерієм;
- 3) метод спряжених функцій для варіаційної задачі Лагранжа;
- 4) використання вищої похідної як змінної управління.

Вибираємо альтернативу 4, переваги якої:

- простота реалізації (в ЦОМ),
- нечутливість до різних збурень (робастність);
- можливість легко забезпечити паралельну роботу виконавчих елементів.

Додатковий фактор – метод запатентований (Smith A., Metcalf E. Патент США №3182933) і реально використовується (ЛВЗП Харрієр). Цей метод забезпечує оптимальні за інтегральним критерієм витрати енергії управління. Саме це і потрібно в більшості задач, починаючи з економічних. Вибираємо цей критерій, тому що таке управління мінімізує і перевантаження – це важливо, щоб не пошкоджувати об'єкти.

Задача має занадто багато інтерпретацій, виберемо з них найбільш вражаючу, загальнозрозумілу і очевидну (в прямому значенні) інтерпретацію. Дійсно, вражає, коли *єдиний в світі надзвуковий літак ЯК-141 з масою 16 тонн* м'яко сідає на палубу в негоду при злагодженій роботі трьох двигунів з сумарною тягою 23 тонни.

На рис. 3.9 показано об'єкт такого класу – найбільш масовий літак вертикального зльоту і посадки (ЛВЗП) Харрієр (маса середня 10 тонн, двигун – один, тяга 10 тонн). На

рис. 3.10 подана специфікація термінальної системи з паралельно працюючими приводами. Між іншим, сьогодні такими ж методами можливо управляти і більш "масивними" об'єктами – металургійним комбінатом, супермаркетом, системою банків, національною економікою.

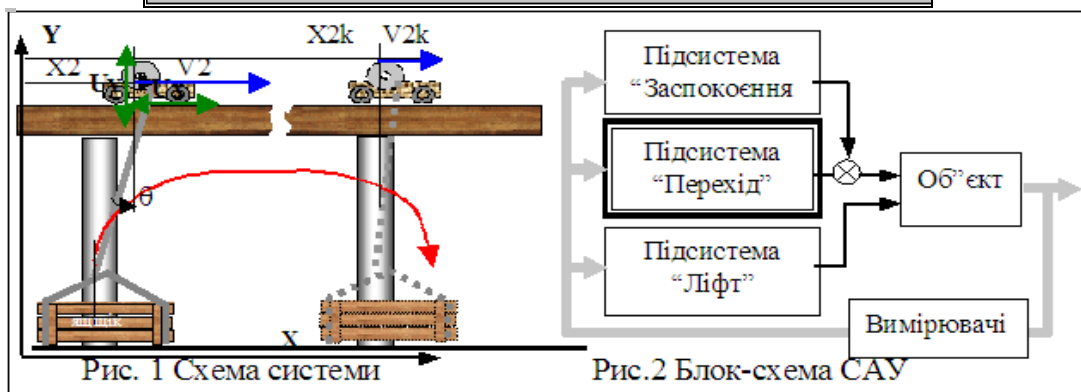
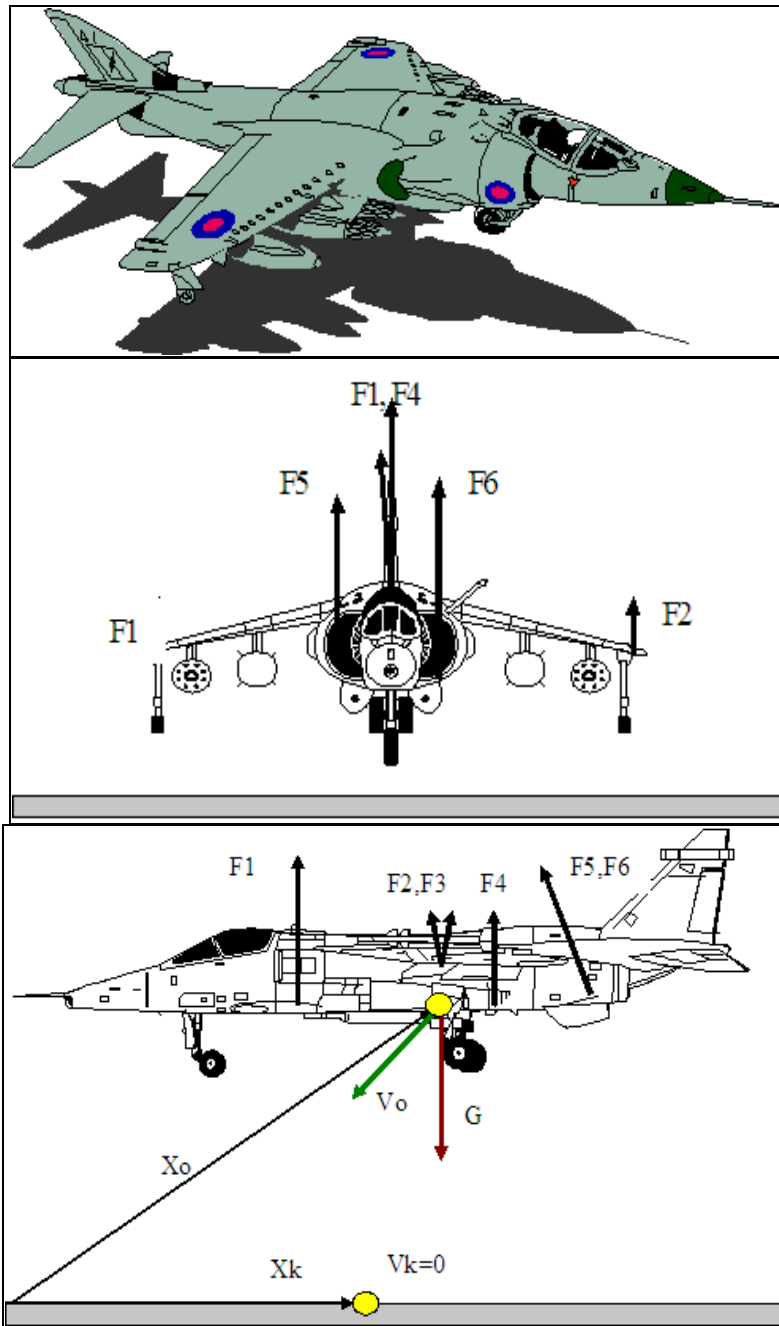
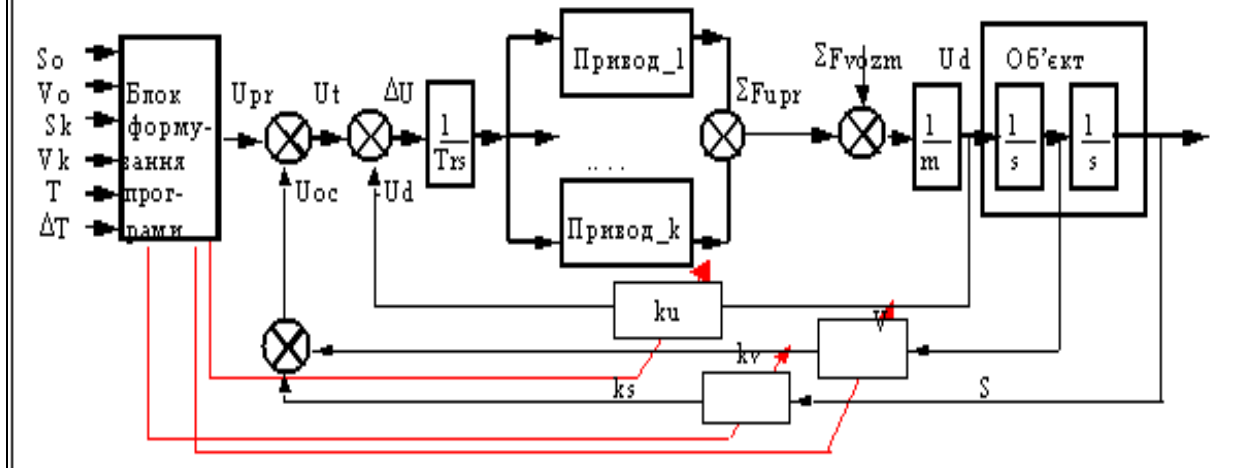


Рис. 3.9. Типові об'єкти управління кінцевим станом



### Структурна схема термінальної САУ с паралельними приводами



Список змінних і параметрів (кзз – коефіцієнт зворотного зв'язку)

$U_{pr}$  – програмне управління;  $k_u$  – кзз "штраф за прискорення";  
 $U_{oc}$  – сигнал зворотного зв'язку;  $k_v$  – кзз "штраф за швидкість";  
 $U_t$  – потрібне управління;  $k_s$  – кзз "штраф за координату";  
 $U_d$  – дійсне управління;  $Tr_s$  – постійна часу контуру приводів;  
 $T$  – заданий час переходу;

$k_{0t}, k_{1t}, k_{2t}, k_{3t}, k_{4t}, \dots$  – розраховуються за рівняннями методу;

$k_u, k_v, k_s, \dots$  – розраховуються за рівняннями методу на початку процесу переходу;

$\Delta T$  – "жорсткість управління" вибирається проектувальником.

Критерії оптимізації управління

J1 – "інтеграл від витрат енергії на управління";

J2 – "помилка вектора стану в кінцевий момент часу".

Критерії суперечні – зменшення витрат збільшує помилку і навпаки, тому використовується зважена сума, або один з критеріїв виноситься в обмеження.

Методи оптимізації:

наближене розв'язання варіаційної задачі, пошукові методи, самонастроювання.

Рис. 3.10. Специфікація термінальної системи з паралельними приводами

Записуємо рівняння для параметрів закону термінального управління. Це результат розв'язання задачі "знайти параметри закону управління, при яких траєкторія руху об'єкта при заданому початковому стані у заданий момент часу проходить через заданий стан".

$$\left. \begin{aligned}
 c_0(s_0, v_0, T_d) &= 6 \cdot \frac{s_0 - s_k}{T_d^2} - \frac{4 \cdot v_0 + 2 \cdot v_k}{T_d}; & c_1(s_0, v_0, T_d) &= 12 \cdot \frac{s_0 - s_k}{T_d^3} + 6 \cdot \frac{v_0 + v_k}{T_d^2}; \\
 K_0(s_0, v_0, T_g) &= 6 \cdot \frac{s_0}{T_g^2} + 4 \cdot \frac{v_0}{T_g} + c_0(s_0, v_0, T_d); & K_v(T_g) &= \frac{-4}{T_g}; & K_s(T_g) &:= \frac{-6}{T_g^2}; \\
 K_1(s_0, v_0, T_g) &= 6 \cdot \frac{v_0}{T_g^2} + 4 \cdot \frac{c_0(s_0, v_0, T_d)}{T_g} + c_1(s_0, v_0, T_d); \\
 K_2(s_0, v_0, T_g) &= 3 \cdot \frac{c_0(s_0, v_0, T_d)}{T_g^2} + 2 \cdot \frac{c_1(s_0, v_0, T_d)}{T_g}; & K_3(s_0, v_0, T_g) &= \frac{c_1(s_0, v_0, T_d)}{T_g^2}
 \end{aligned} \right\} (3.19)$$

Збираємо рівняння (3.19) у програмний модуль – функцію від параметрів системи і початкового та кінцевого станів. Граничні стани та параметри системи збираємо у два вектори. На рис. 3.11 подано модуль синтезу (розрахунку) параметрів термінального управління. Програму робимо орієнтованою на людину (для ЦОМ тут десяток зайвих операцій). Порівнюємо записані вище рівняння з оцим, що нижче. Бачимо, що все записане в природному для людини вигляді, все відкрите і контрольоване.

Syn(Pp, Xo) :=	$v0 \leftarrow X_{01}$
	$vk \leftarrow X_{02}$
	$s0 \leftarrow X_{03}$
	$sk \leftarrow X_{04}$
	$Tp \leftarrow Pp_1$
	$Td \leftarrow Pp_2$
	$Tg \leftarrow Pp_3$
	$T \leftarrow Pp_4$
	$c0 \leftarrow 6 \cdot \frac{sk - s0}{Td^2} - \frac{4 \cdot v0 + 2 \cdot vk}{Td}$
	$c1 \leftarrow 12 \cdot \frac{s0 - sk}{Td^3} + 6 \cdot \frac{v0 + vk}{Td^2}$
	$K0 \leftarrow 6 \cdot \frac{s0}{Tg^2} + 4 \cdot \frac{v0}{Tg} + c0$
	$K1 \leftarrow 6 \cdot \frac{v0}{Tg^2} + 4 \cdot \frac{c0}{Tg} + c1$
	$K2 \leftarrow 3 \cdot \frac{c0}{Tg^2} + 2 \cdot \frac{c1}{Tg}$
	$K3 \leftarrow \frac{c1}{Tg^2}$
	$Kv \leftarrow \frac{-4}{Tg}$
	$Ks \leftarrow \frac{-6}{Tg^2}$
	$v_{yx} \leftarrow (K0 \ K1 \ K2 \ K3 \ Kv \ Ks)^T$
	$v_{yx}$

Рис. 3.11. Модуль синтезу (розрахунку) параметрів термінального управління

Це один з прикладів сучасних інформаційних технологій: якщо поки не можна робити програми без участі людини, то програма повинна бути написана "по-людськи", максимально прозорою, щоб сам автор і хто завгодно міг прочитати і перевірити її.

Повертає програма вектор з шості коефіцієнтів закону управління. Незавжди цю програму переробити для інших задач термінального управління – наприклад, м'якого зближення, для об'єктів вищих порядків.

### Розробка математичної моделі і програми для моделювання динаміки термінальної САУ з паралельно працюючими приводами

Запишемо спочатку систему диференціальних рівнянь термінальної САУ (неперервну ММ). Відповідно з блок-схемою системи запишемо:

$$\text{Об'єкт: } \frac{d}{dt} S = V; \quad \frac{d}{dt} V = U d(U_{\text{усім}}); \quad \text{Приводи: } \frac{d}{dt} U_{\text{усім}} = \frac{1}{T_{\text{пр}}} \cdot (U_{\text{потрібне}} - U_{\text{дійсне}}).$$

Знов таки, відповідно з блок-схемою управління визначається у такій послідовності:  
1) як функція координат і часу формується потрібне управління  $U_{\text{потрібне}}$ :

$$U_{\text{потрібне}} = K_0 + K_1 \cdot t + K_2 \cdot t^2 + K_3 \cdot t^3 + K_v \cdot V + K_s \cdot S;$$

2) потім воно порівнюється з діючим на об'єкт дійсним управлінням –  $U_{\text{дійсне}}$ ;

3) потім інтеграл від різниці ( $U_{\text{потрібне}} - U_{\text{дійсне}}$ ) подається на усі приводи як  $U_{\text{усім}}$ .  
Управління  $U_{\text{дійсне}}$  є сумою виходів усіх приводів  $\sum U_{\text{пр}}$  і зовнішніх збурень  $X_{\text{взм}}$ .

**Розробка бібліотеки модулів виконавчих елементів.** Для дослідження різних характеристик і конфігурацій сумісно працюючих приводів (виконавчих елементів) потрібен досить автономний модуль для моделювання приводів. Нижче наведено статичні нелінійні характеристики приводів – їх можна замінити на які завгодно.

Параметри характеристик приводів

$$p_1 \equiv .5; \quad q_1 \equiv 3; \quad r_1 \equiv 1; \quad p_2 \equiv .3; \quad q_2 \equiv .9; \quad r_1 \equiv 1; \quad p_3 \equiv 1.7; \quad q_3 \equiv .06.$$

Допоміжні функції  $u := -5, -4.6..5$ ;  $zez(y) \equiv \text{if}(y > 0, .5, 1.5)$ .

Робимо бібліотеку можливих нелінійностей приводів та сегментів робота. Цю бібліотеку користувач нашої системи може розширювати в усіх вимірах до безмежних границь, при бажанні і наявності відповідної потенції:

$$1. \quad P_1(u, p_1, q_1, r_1) \equiv \left( \frac{\exp(p_1 + q_1 \cdot u) \cdot r_1}{1 + \exp(p_1 + q_1 \cdot u)} - .5 \right) \cdot zez(u) - 0.1 \cdot (u < 0);$$

$$2. \quad P_2(u, p_2, q_2) \equiv q_2 \cdot (1 - \exp(-p_2 \cdot |u|)) \cdot \left( \frac{u}{|u|} \right);$$

$$3. \quad P_3(u, p_3, q_3) \equiv q_3 \cdot (|u|)^{p_3} \cdot \frac{u}{|u|}.$$

Будуємо графіки цих нелінійностей: випуклої (вгору), ввігнутої (вниз), ввігнуто-випуклої несиметричної (рис. 3.12). Це головні класи гладких статичних нелінійностей в системах управління.

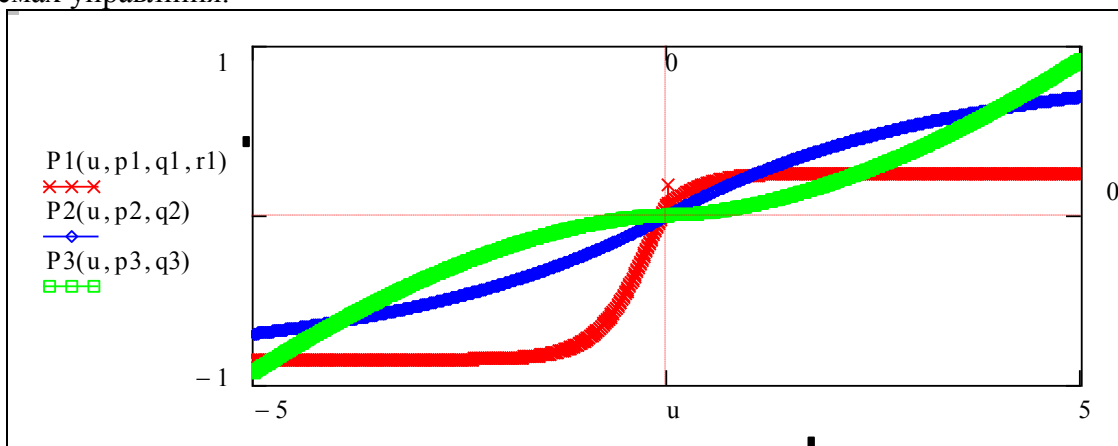


Рис. 3.12. Типові "гладкі" нелінійні характеристики приводів

Для прикладу відібрано приводи з "випуклою", "вгнутою" і несиметричною характеристиками. Неважко в даному програмному середовищі аналогічно промодельовати нечутливість, гістерезис, люфт, нелінійну динаміку та ін. Подаємо ще один модуль характеристик з негладкими нелінійностями.

Наступний модуль (рис. 3.13) імітує зону нечутливості у вимірювачах чи приводах. Використовуємо альтернативні версії функції нечутливості – власну функцію  $\text{sgn}()$  (функцію користувача) і вбудовану функцію пакета  $\Phi(x)$  – функцію Хевісайда, що дорівнює:

$$\text{sgn}(z) \equiv \frac{z}{|z|}; \quad \Phi(0) = 1; \quad \Phi(-1) = 0; \quad \Phi(1) = 1;$$

$$\text{sign}(33) = 1; \quad \text{sgn}(-33) = -1; \quad i := -10, -9..10.$$

Записуємо вираз для функцій "зона нечутливості" та "обмеження", тестуємо їх:

$$\text{neth}(a, z) \equiv \frac{z}{|z|} \cdot \Phi(|z| - a) \cdot (|z| - a);$$

$$\text{obmeg}(a, z) \equiv -\frac{z}{|z|} \cdot \Phi(|z| - a) \cdot (|z| - a) + z.$$

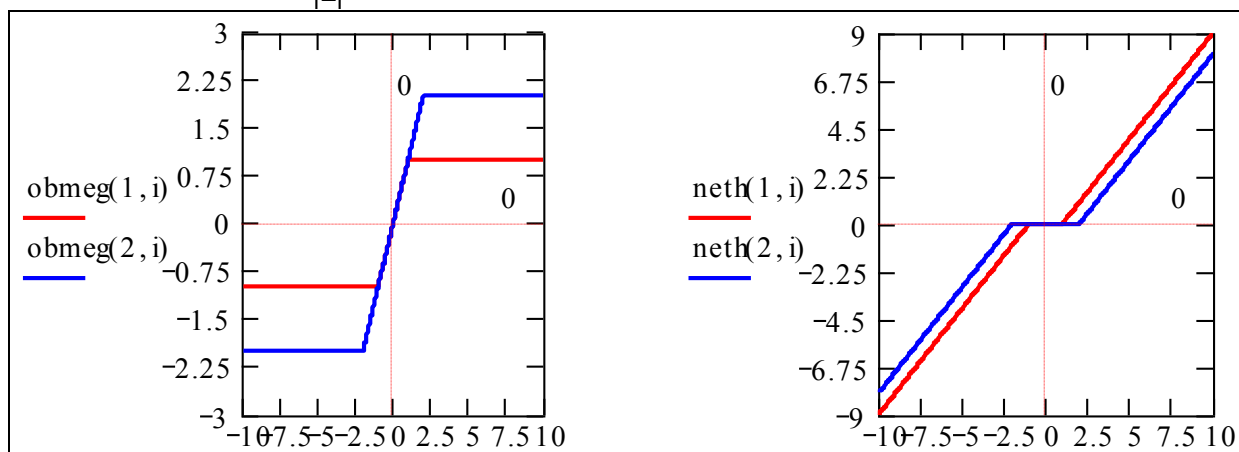


Рис. 3.13. Характеристики виконавчих елементів класів "зона нечутливості" та "обмеження рівня"

Будуємо графіки цих нелінійностей для значення зони  $a = 1, 2$ . Бачимо, що функції коректно описують нечутливість та обмеження. Тепер ми можемо вбудовувати цю функцію у відповідні програми. Неважко зробити аналогічні модулі "зона нечутливості + обмеження" (рис. 3.14), "гістерезис", "запізнення", наприклад:

$$\text{ОбмНеч}(\text{рівень}, \text{зона}, i) := \text{obmeg}(\text{рівень}, \text{neth}(\text{зона}, i))$$

$$x := -20, -19..20; \quad \text{Pvv}(A, v, x) := A \cdot (|x|)^v \cdot \text{sign}(x)$$

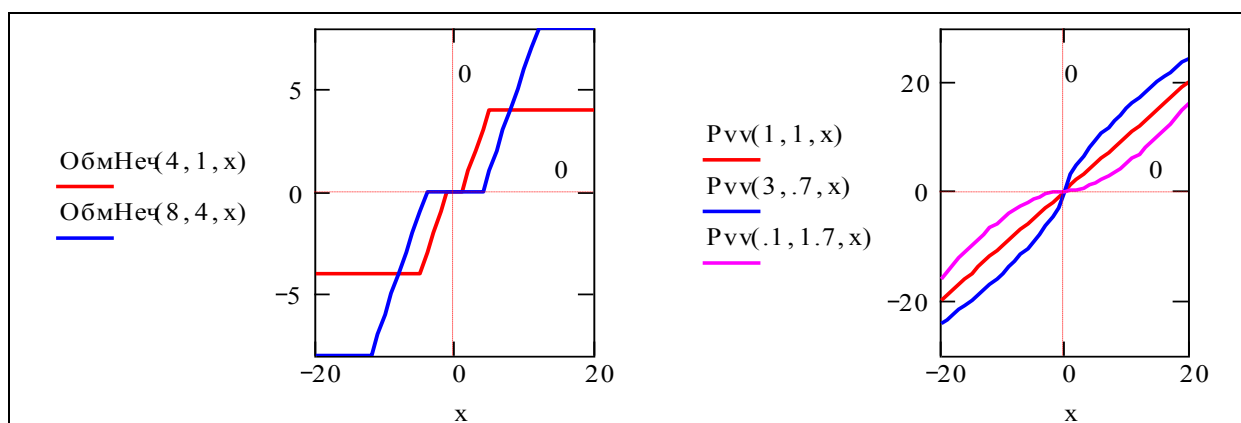


Рис. 3.14. Характеристики виконавчих елементів класів "обмеження + нечутливість" та "гладка нелінійність"

**Розробка програми моделювання.** Деталізуємо систему рівнянь динаміки САУ, підставимо у рівняння характеристики приводів. Користувач може вибрати інший модуль з бібліотеки приводів.

$$\left. \begin{aligned} u &= U_{\text{сiм\_однаковс}}(t); \\ U_{\text{дiйсн}}(u) &= P1(u, p1, q1, r1) + P2(u, p2, q2) + P3(u, p3, q3); \\ \frac{d}{dt} U_{\text{сiм\_однаковс}}(t) &= \frac{1}{T_r} \cdot U_{\text{потр}}(t, V, S, ss0, v0) - U_{\text{дiйсн}}(U_{\text{сiм\_однаковс}}(t)); \\ \frac{d}{dt} V(t) &= U_{\text{дiйсн}}(U_{\text{сiм\_однаковс}}(t)); \quad \frac{d}{dt} S(t) = V(t). \end{aligned} \right\} \quad (3.20)$$

Конструюємо функцію "відмова приводу з певного моменту часу".

$$\text{ORIGIN} := 1; \quad \text{От}(t, T_0) \equiv (1 - \Phi(t - T_0)).$$

Моменти відмов вводяться далі на стенді, в абсолютному присвоєнні:  $T_0 = \begin{pmatrix} 40 \\ 100 \end{pmatrix}$ .

Згідно з постановкою задачі зусилля окремих виконавчих елементів підсумовуються. Запишемо рівняння для управління

$$U_{\text{д2}}(u, t, T_0) := (P1(u, p1, q1, r1) \cdot \text{От}(t, T_{01}) + P2(u, p2, q2) \cdot 1 + 10P3(u, p3, q3) \cdot \text{От}(t, T_{02})). \quad (3.21)$$

Формуємо зону вхідних даних (далі, біля графіків) даних для програми моделювання:

$$X_0 := \begin{pmatrix} v_0 \\ v_k \\ s_0 \\ s_k \end{pmatrix}; \quad \begin{pmatrix} T_p \\ T_d \\ T_g \\ T \end{pmatrix} = \begin{pmatrix} 0.6 \\ 40 \\ 12 \\ 0.1 \end{pmatrix}; \quad \begin{pmatrix} \text{"Постійна приводу:"} \\ \text{"Тривалість процесу:"} \\ \text{"Жорсткість у правління:"} \\ \text{"Крок квантування:"} \end{pmatrix}; \quad P_p := \begin{pmatrix} T_p \\ T_d \\ T_g \\ T \end{pmatrix}.$$

$$N := \frac{T_d}{T}; \quad k := 1..N; \quad N = 400; \quad T_0 = \begin{pmatrix} 40 \\ 100 \end{pmatrix}.$$

На рис.3.15 подано видача програми синтезу. Виводимо для контролю результати синтезу – розрахунку параметрів управління для певної конкретної задачі переходу, присвоємо цим параметрам зрозумілі імена.

$$\text{Syn}(P_p, X_0) = \begin{pmatrix} 17.5 \\ -0.304 \\ -0.033 \\ 0.001 \\ -0.333 \\ -0.042 \end{pmatrix} \quad \begin{pmatrix} K_0 \\ K_1 \\ K_2 \\ K_3 \\ K_v \\ K_s \end{pmatrix} := \begin{pmatrix} \text{Syn}(P_p, X_0)_1 \\ \text{Syn}(P_p, X_0)_2 \\ \text{Syn}(P_p, X_0)_3 \\ \text{Syn}(P_p, X_0)_4 \\ \text{Syn}(P_p, X_0)_5 \\ \text{Syn}(P_p, X_0)_6 \end{pmatrix}$$

Рис. 3.15. Видача програми синтезу

Записуємо програму моделювання (рис. 3.16). За структурою це просто цикл "по кроках моделювання", в якому обчислюється система рівнянь динаміки (3.20).

```

Pr(X0) := "ORIG"
          Uyq ← 0
          v1 ← X01
          ss1 ← X03
          Kpr ←  $\frac{T}{Tp}$ 
          for k ∈ 1..N
            t ← k·T
            Uprg ← K0 + K1·t + K2·t2 + K3·t3
            Uoc ← Kv·vk + Ks·ssk
            Uptr ← Uprg + Uoc
            ssk+1 ← ssk + vk·T
            vk+1 ← vk + Uyq·T
            Udijs ← Ud2(Uyq, k, To)
            Uyq+1 ← Uyq + Kpr·(Uptr – Udijs)
            Msost⟨k⟩ ←  $\begin{pmatrix} ssk \\ vk \\ Udijs \\ Uptr \end{pmatrix}$ 
          Msost

```

Рис. 3.16. Текст програми моделювання відмовостійкої САУ

Програма використовує підпрограму розрахунку (синтезу) параметрів закону управління (3.21) і функцію користувача "групи приводів", яка, в свою чергу, складається з функцій користувача – характеристик приводів.

Можемо досліджувати різні системи з паралельно працюючими приводами. Для цього достатньо лише замінити функцію Ud2(Uyq, k, To). Більш радикальний спосіб – зробити програму функцією від цієї функції.

Тестуємо програму, виводимо вихід для різних початкових умов.

Pr(X0) =	1	2	3	4	5	6	7	8	9	10	11	12	13
1	400	401	402	403	404	405	406	407	408	409	410	411	411
2	10	10	10	10	10	10	10	9	9	9	9	9	9
3	0	-1	-1	-2	-2	-2	-2	-2	-3	-3	-3	-3	-3
4	-3	-3	-3	-3	-3	-3	-3	-3	-3	-3	-3	-3	-3

$$\text{Pr} \begin{pmatrix} 10 \\ 0 \\ 300 \\ 0 \end{pmatrix} = \begin{array}{c|cccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ \hline 1 & 300 & 301 & 302 & 303.002 & 304.009 & 305.023 & 306.044 \\ 2 & 10 & 10 & 10.024 & 10.07 & 10.133 & 10.209 & 10.293 \\ 3 & 0.184 & 0.253 & 0.46 & 0.637 & 0.776 & 0.876 & 0.939 \\ 4 & 1.636 & 1.563 & 1.481 & 1.391 & 1.295 & 1.193 & 1.088 \end{array}$$

Перепишемо вихід програми в масив:  $Vy := \text{Pr}(Xo)$  – потім все буде швидко виконуватись. Обчислюємо дійсне управління

$$\text{Удійсн}(u) \equiv (P1(u, p1, q1, r1) + P2(u, p2, q2) + P3(u, p3, q3))$$

і управління з урахуванням відмов 2-х приводів з трьох:

$$u2d_k := \text{Уд2}(Vy_{3,k}, k \cdot T, T_0).$$

Задаємо моменти відмов:  $T_0 \equiv \begin{pmatrix} 40 \\ 100 \end{pmatrix}$ .

Задаємо початковий стан  $Xo^T = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 10 & 0 & 400 & 0 \end{array}$ .

Розробка виконується з перспективами використання в різних напрямках. І навіть коли розробка виконується "для себе", то після місячної перерви автор не буде знати що, де, чому, коли вводиться і виводиться в його програмі. Тому завжди у всіх сенсах і для всіх випадків корисно розробити наведений нижче документ.

### Інструкція користувачу

На рис. 3.17 наведено "стенд", де зібрані входи і виходи задачі. **Змінюйте** (обережно, бо програма і модель системи в стартовій версії не захищені від неприпустимих вхідних даних) **усі вхідні параметри**:

- початкові і кінцеві умови,
- кількість кроків, параметри закону управління – жорсткість та постійну часу приводу.

Виконайте ці експерименти так, щоб можна було зробити висновки відносно властивості системи приходити у заданий термінальний стан при зміні:

- початкового стану,
- часу переходу,
- параметрів управління.

### Вибір параметрів та дослідження термінальної САУ при паралельній роботі приводів

Ми розробили комплекс програмних модулів для розрахунку параметрів термінального регулятора, моделювання виконавчих елементів, моделювання відмов і моделювання системи в цілому. Тепер можемо виконати дослідження системи в різних режимах, оцінити чутливість, виконати пошукову оптимізацію параметрів системи управління.

Подаємо приклади проведення пошукових досліджень у вигляді копій екранів стенду. На самому стенді (див. рис. 3.17) подано ситуацію, коли в процесі гальмування швидкість спочатку підвищується. Причина цього в фіксованому часі переходу – при малій швидкості система спочатку розганяє об'єкт управління.

На рис. 3.18. подано номінальний процес в системі з нелінійними приводами.

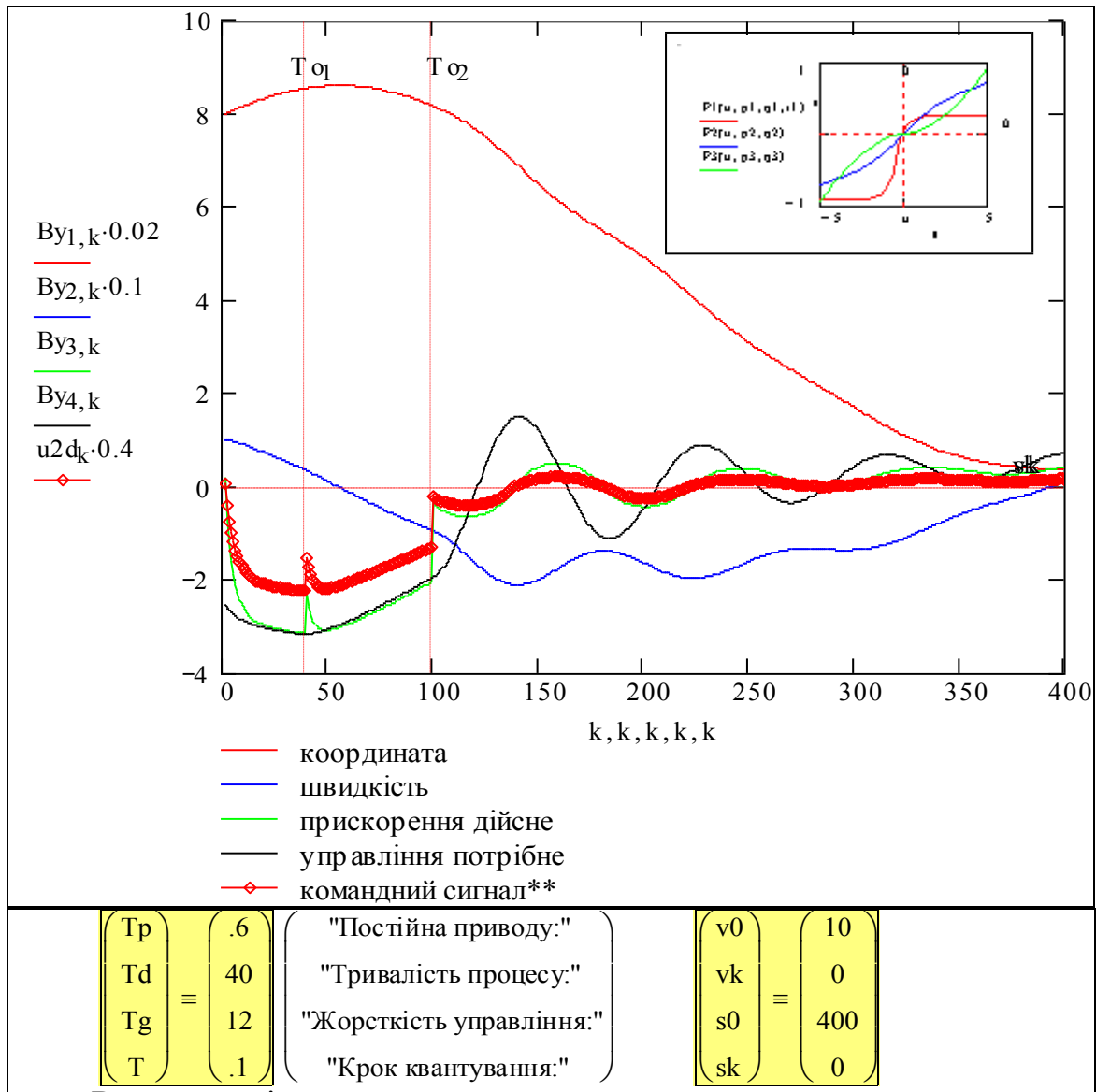


Рис. 3.17. Стенд "дослідження відмовостійкої САУ"

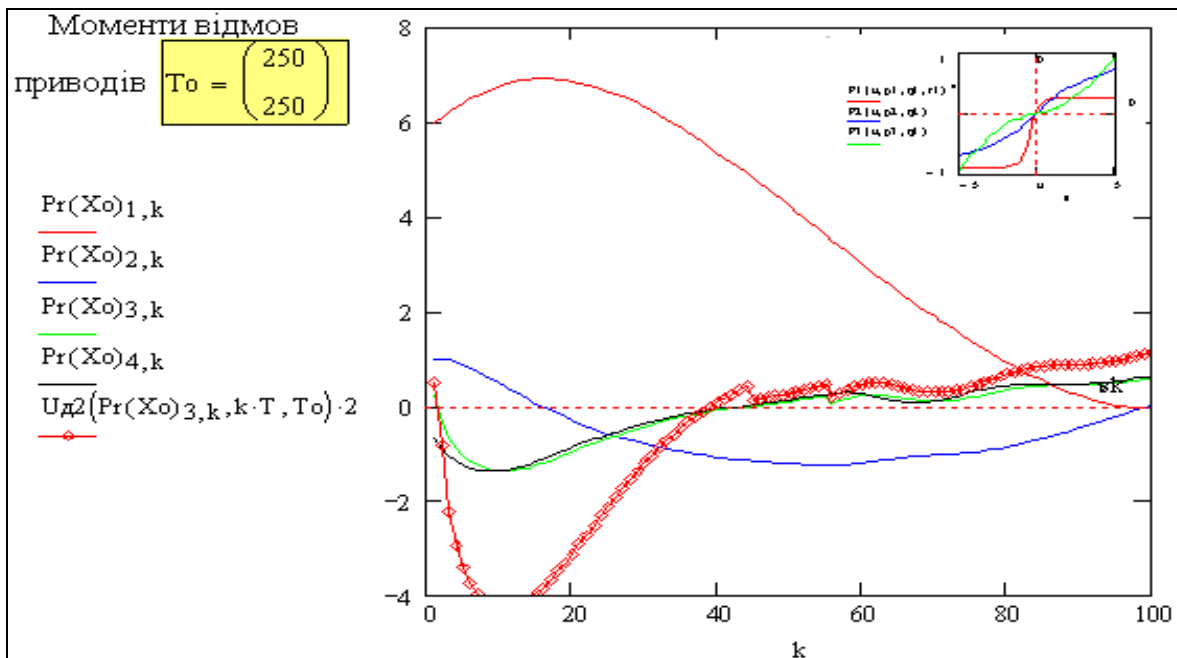


Рис. 3.18. Номінальний процес в системі з нелінійними приводами



На рис. 3.17 подано приклад процесу для найбільш важкого випадку: в момент найбільшого навантаження, на 40-й та 100-й секундах (дивись маркери) відмовляють два приводи з трьох. Система залишається задовільною. На рис. 3.18 відмови приводів мають місце на завершальному етапі переходу, коли не потрібно великих значень управління.

Подаємо на рис. 3.19 перехідні процеси для специфічної задачі: "зупинитись в поточній точці через  $T_d$  одиниць часу". В цьому випадку швидкість об'єкта зменшується (за модулем), змінює знак, об'єкт повертається в задану точку. Подано два таких процеси для різних напрямків руху. Вони несиметричні, тому що один привід має несиметричну характеристику.

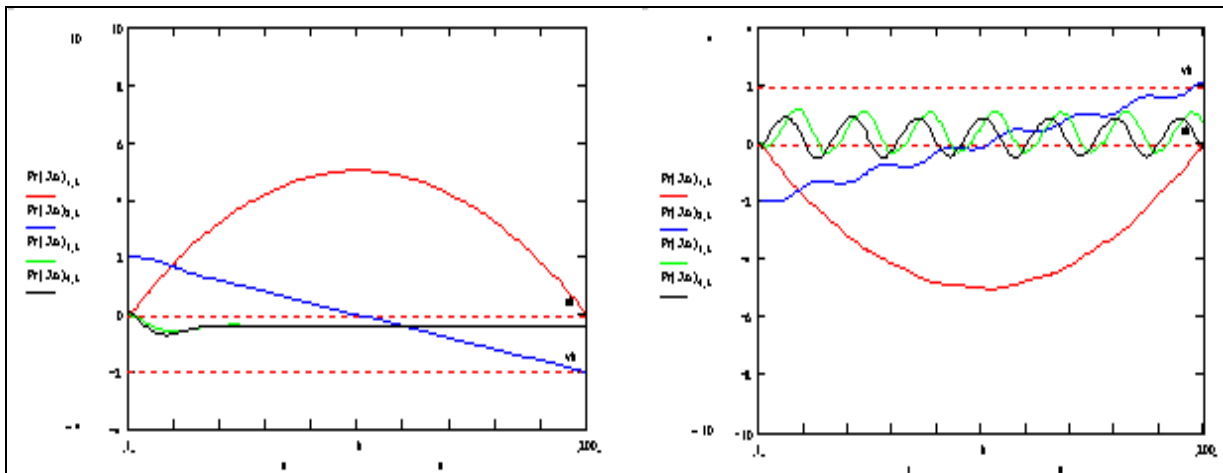


Рис. 3.19. Тестування системи при несиметричних характеристиках приводу

### Висновки

Розроблено працездатний багаточільовий комплекс програм для синтезу та дослідження термінальних САУ, де змінна управління – вища похідна в моделі динаміки об'єкта. Отримано розв'язання для багатовимірної задачі м'якого зближення і розроблено відповідні інтерфейси. Підтверджена функціональна придатність термінальної САУ з управлінням вищою похідною для підключення паралельно працюючих приводів без їх попереднього узгодження при значних нелінійностях характеристик. В більш широкому контексті ця властивість є базою для забезпечення відмовонечутливості і відмовостійкості САУ без реконфігурації.

### Контрольні запитання

1. Дайте означення понять: надійність, відмовонечутливість, відмовостійкість, відмовобезпечність.
2. Як розшифровуються скорочення УКВУ, УКЧУ, що таке "швидкий регулятор"?
3. Визначення вектора стану, вектора управління і "управління вектором стану".
4. До чого власне адаптується система термінального управління з паралельно працюючими приводами?
5. Логіка управління відмовостійкою системою.
6. Як об'єднати вимоги ефективності і живучості?

### Завдання для самостійного виконання

Проведіть такі дослідження відмовостійкості і якості термінальної САУ

1. Вплив відхилень параметрів об'єкта (приводів) і параметрів закону управління.
2. Вплив випадкових шумів у вимірювачах.
3. Вплив нелінійностей елементів САУ.
4. Визначення границь стійкості системи в просторі відхилень параметрів.
5. Дослідити поведінку системи при дрейфі і тимчасових відмовах параметрів.

## 4 ПРИКЛАДИ ПОБУДОВИ АДАПТИВНИХ САУ

В цьому розділі подано дві абсолютно несхожі в змістовному плані задачі для системи хімічних реакторів та для банківської системи: – синтез і дослідження адаптивної САУ для оптимального розподілу навантаження між реакторами; – синтез і дослідження адаптивної САУ для встановлення оптимальних ставок кредитів і депозитів. Спільне в цих задачах – принцип побудови системи управління, що є "симбіозом" двох регуляторів – звичайного і екстремального. У кожного регулятора своя задача, але спільні виконавчі елементи. Головна увага приділена **технологіям конструювання робочих моделей**, придатних для проведення обчислювальних експериментів та методів синтезу, придатних для реалізації в мікроконтролерних системах.

### 4.1 Система оптимального управління синтезом стиролу

**Опис об'єкта управління. Аналіз альтернатив управління.** Розглядається хімічний реактор для синтезу стиролу. Стирол – сировина для великотоннажної пластмаси – полістиролу. Виробництво стиролу звичайно є масовим. Реактори для синтезу стиролу працюють разом, паралельно. Причини цього – велика потреба в продукті, старіння реакторів, періодична зупинка їх для заміни каталізатора при умові підтримки заданого рівня випуску продукції. Паралельна робота реакторів, великі вимоги до надійності функціонування виробництва в цілому обумовлюють і методи управління – групового, одразу всіма реакторами. Перша ціль управління – підтримка потрібного рівня сумарного виробництва незалежно від стану окремих реакторів. Специфіка масового виробництва продуктів у реакторах в тому, що навіть малі частки відсотка зміни експлуатаційних витрат приводять до великих сумарних економічних ефектів. З цієї причини режими роботи реакторів – тиск, температуру, концентрації – ретельно оптимізують. Характеристики реакторів з часом погіршуються – проходить процес "отруєння каталізатору". Звичайно в батареї реакторів паралельно працюють реактори з різними ефективностями (одиниць продукції на одиницю експлуатаційних витрат). Зрозуміло, що треба кращі реактори завантажувати більше. Друга ціль управління – таке розподілення навантаження по реакторах, що дає мінімум експлуатаційних витрат при заданому рівні випуску продукції. Розглянемо стисло технологічний процес синтезу стиролу (рис. 4.1).

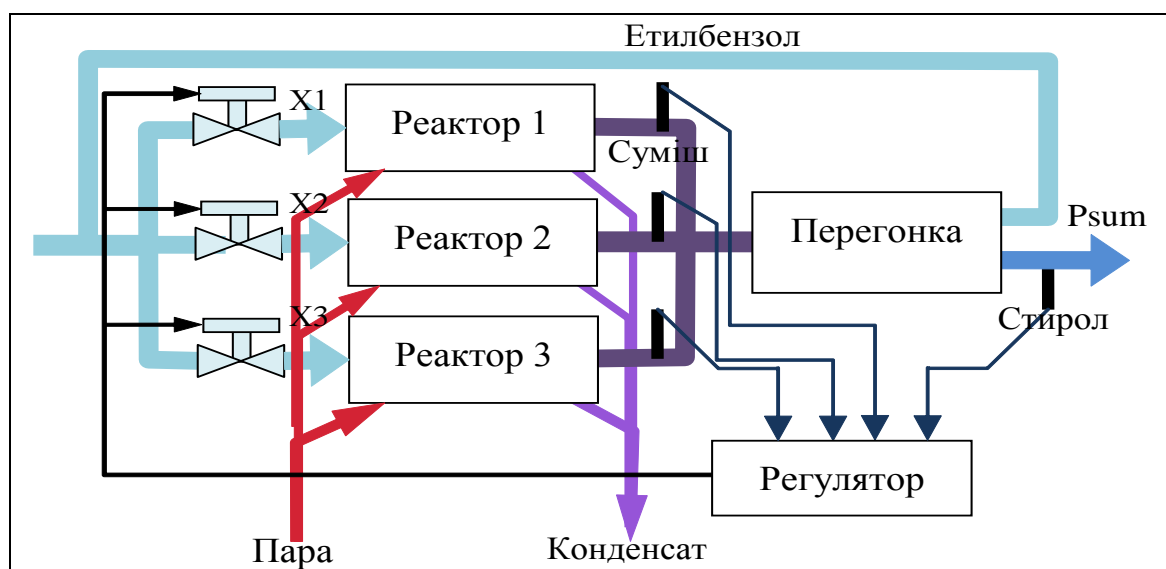


Рис. 4.1. Схема системи виробництва стиролу

В реакторі нагрівається етилбензол в присутності каталізатора, утворюється суміш стиролу, етилбензолу, бензолу, толуолу, що переганяється, і виділений етилбензол знову йде у реактор. Реактор підігрівається парою. Особливості системи – нелінійні і нестационарні характеристики реактора, а також паралельна робота декількох реакторів на потрібний

сумарний вихід стиролу. Необхідно розробити систему оптимального управління розподілом етилбензолу по окремих реакторах з урахуванням обмеження на сумарну продуктивність. Критерій – сумарні виробничі витрати на заданий випуск продукту. На рисунку 4.1 наведено спрощену схему системи та модель процесу синтезу стиролу.

Реактор для синтезу стиролу – не ракета, – він значно складніший. Щоб побудувати його аналітичну модель, треба враховувати закони каталітичної хімії, термодинаміки, газодинаміки і багато іншого. Для цілей практики іноді буває досить зняти експериментально характеристики реактора, а потім виконати апроксимацію цих характеристик, наприклад, поліномами. На рис. 4.2 подано апроксимації статичних характеристик реактора.

**Розробка системи оптимального управління синтезом стиролу.** Математична модель реактора для аспекту "залежність виробничих витрат від виходу стиролу" – апроксимація реальної характеристики поліномом. Запишемо похідну від цієї залежності (рис. 4.3).

$$C(a1, a2, a3, P) := 20 + a1 \cdot P + .5 \cdot a2 \cdot P^2 + a3 \cdot P^5,$$

$$dC(a1, a2, a3, P) := a1 + a2 \cdot P + 0.01 \cdot a3 \cdot P^4. \quad (4.1)$$

Математична модель реактора для аспекту "залежність виходу стиролу від рівня подачі етилбензолу" (рис. 4.4). Ця характеристика є оптимізованою за іншими вхідними змінними – витратою пари, температурою, тиском та іншими. Використовуємо як перше наближення модель з одним параметром  $b1$  [28]. Реактори мають характеристики одного класу і відрізняються параметром  $b1$ . Модель нереалістична для малих значень  $x$ .

$$P_i(b1, x) := b1 \cdot x^{0.5}. \quad (4.2)$$

Задаємо ранжовані змінні  $P$  та  $X$  і будуюмо графіки характеристик

ORIGIN := 1; X := 0, 0.25..40; P := 0, 0.5..25.

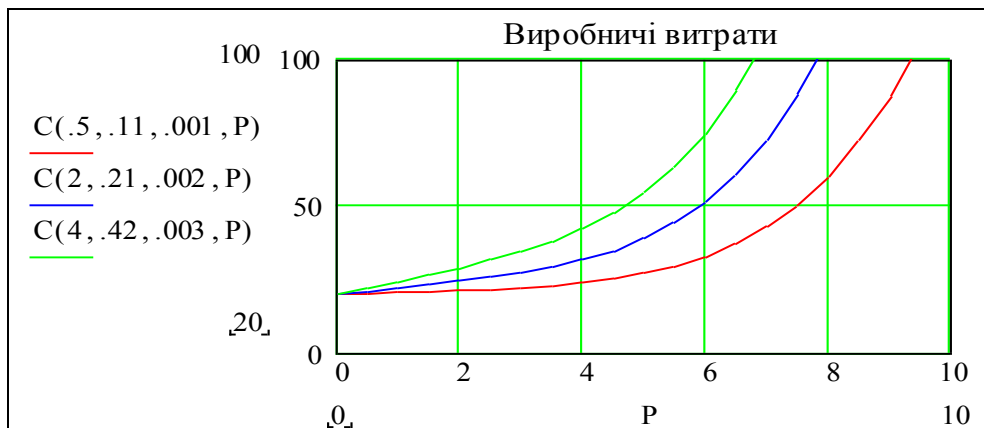


Рис. 4.2. Залежності виробничих витрат від виходу стиролу

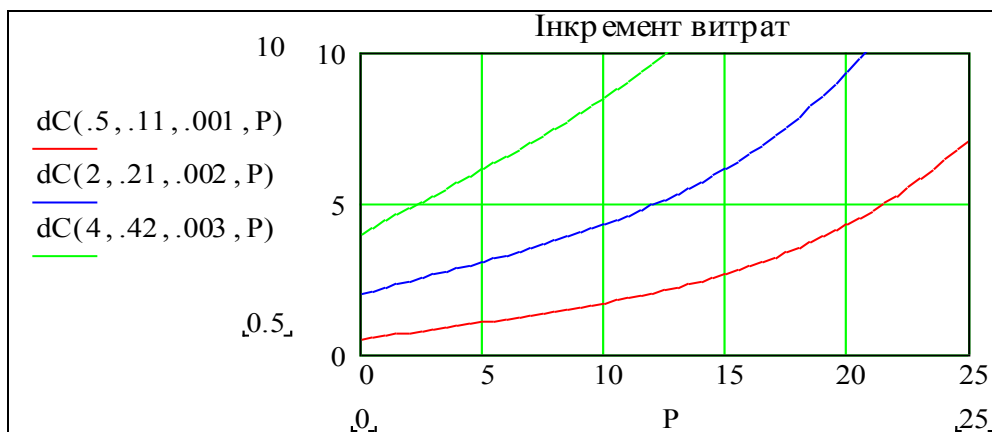


Рис. 4.3. Залежності приросту виробничих витрат від виходу стиролу

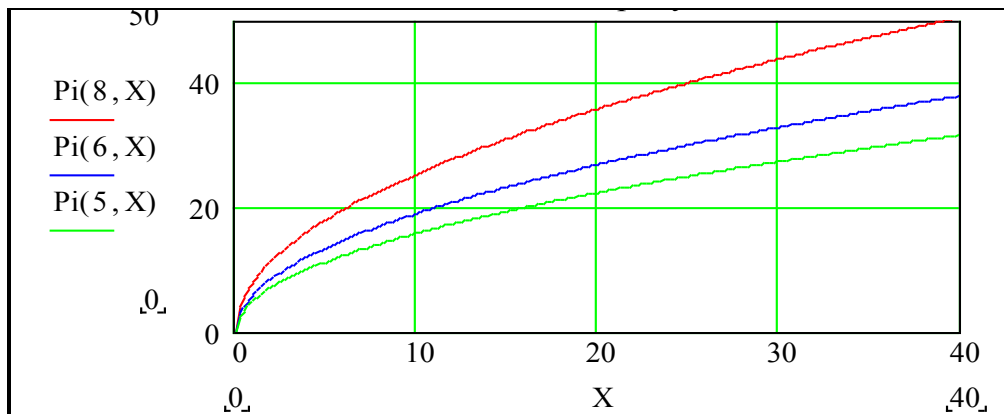


Рис. 4.4. Залежності виходу стиролу (P) від подачі етилбензолу (X)

### Формулювання оптимізаційної задачі

Задана (вибрана) цільова функція – сумарні виробничі витрати. Ціль оптимізації – мінімум цієї функції.

$$C_{sum} = F(P_1, P_2, P_3) = C_1(P_1) + C_2(P_2) + C_3(P_3).$$

Знайти:  $\min\{F(P_1, P_2, P_3)\}.$  (4.3)

Задано обмеження на сумарний вихід стиролу, тобто:

$$P_1 + P_2 + P_3 = P_{sum}; \quad P_{sum} = P_{tr}; \quad G(P_1, P_2, P_3) := (P_{tr} - P_1 + P_2 + P_3) = 0. \quad (4.4)$$

**Задано:**

- 1) такт квантування  $T = 5$  с ;
- 2) вимоги до перехідного процесу: – тривалість 5–15 тактів квантування;  
– перерегулювання <10%.;  
– точність регулювання – 5%.

**Ставимо задачі першого рівня:**

1. Побудувати структурну схему управління реакторами. Обґрунтувати вибір методу синтезу і критерію оптимальності САУ.
2. Розробити програму для знаходження оптимального розподілу навантаження реакторів методом невизначених множників Лагранжа. Визначити оптимальний розподіл і параметри регулятора.
3. Розробити математичну модель і програму моделювання АСУТП.
4. Виконати моделювання САУ. Дослідити поведінку САУ при варіації характеристик реакторів.
5. Запропонувати рішення щодо модифікації програми моделювання для дослідження системи при від-/підключеннях реакторів, при врахуванні динаміки реакторів.

**Задачі другого рівня:**

1. Урахування в програмі того, що реактори є інерційними об'єктами, тобто вихід реактора визначається не алгебраїчним:  $P(b_1, x) = b_1 \cdot x^{0.5}$ , а диференціальним рівнянням:

$$Tr \frac{d}{dt} P_d(t) + P_d(t) = P_i(b_1, x).$$

2. Урахування в програмі того, що управління  $x \geq 0$  (подача сировини не може бути від'ємною).
3. Модифікація програми для випадку 4-х реакторів, довільного числа реакторів.
4. Урахування в програмі повільного дрейфу характеристик реакторів (лінійні функції часу, інші функції...).
5. Урахування в програмі підключення і відключення реактора в певні моменти часу.
6. Оптимальний регулятор не працює, коли немає перехідних процесів, тому в програму слід ввести тестові імпульси.

7. Модифікація програми – замінити простіші ПІ-регулятор та ПІД-регулятор на регулятори вектора стану.

Спочатку розглянемо альтернативні шляхи знаходження оптимального розподілення навантаження між реакторами.

### Аналітичне розв'язання задачі

Очевидна альтернатива – управління процесом можна побудувати на безпосередньому обчисленні оптимальних значень виходу стиролу для кожного реактора. Потім ці значення встановлюються за допомогою відповідних локальних систем регулювання, що повинні їх підтримувати, змінюючи подачу етилбензолу.

Для знаходження оптимального розв'язку використаємо метод невизначених множників Лагранжа. Згідно з цим методом записуємо функцію Лагранжа:

$$L(P1, P2, P3, \lambda) = F(P1, P2, P3) + G(P1, P2, P3) \cdot \lambda. \quad (4.5)$$

Записуємо необхідні умови екстремуму цієї функції:

$$\frac{d}{dP1} L(P1, P2, P3, \lambda) = 0 ; \quad \frac{d}{dP2} L(P1, P2, P3, \lambda) = 0 ; \quad \frac{d}{dP3} L(P1, P2, P3, \lambda) = 0 .$$

Додавши сюди рівняння обмеження отримаємо чотири рівняння для чотирьох невідомих змінних – P1, P2, P3,  $\lambda$ .

Підставимо конкретні значення функцій (4.3) і (4.4) в (4.5).

$$F(P1, P2, P3) := a11 + a21 \cdot P1 + a31 \cdot P1^4 + a12 + a22 \cdot P2 + a32 \cdot P2^4 + a13 + a23 \cdot P3 + a33 \cdot P3^4 .$$

$$G(P1, P2, P3) := Ptr - P1 - P2 - P3 ;$$

$$L(P1, P2, P3, \lambda) := F(P1, P2, P3) + G(P1, P2, P3) \cdot \lambda .$$

Для знаходження аналітичних виразів похідних використаємо апарат символьних обчислень:

$$\frac{\partial}{\partial P1} L(P1, P2, P3, \lambda) \rightarrow ; \quad \frac{\partial}{\partial P2} L(P1, P2, P3, \lambda) \rightarrow ; \quad \frac{\partial}{\partial P3} L(P1, P2, P3, \lambda) \rightarrow a22 + 4 \cdot a32 \cdot P2^3 - \lambda .$$

Вводимо значення параметрів функцій виробничих витрат для усіх реакторів:

$$a11 := .5 ; a21 := 0.25 ; a31 := 0.001 ; a12 := 1 ; a22 := 0.3 ;$$

$$a32 := 0.0017 ; a13 := 2 ; a23 := 0.3 ; a33 := 0.005 .$$

Вводимо значення обмеження та початкові значення шуканих змінних:

$$Ptr := 80 ; P1 := 10 ; P2 := 10 ; P3 := 10 ; \lambda := 10 .$$

Записуємо і розв'язуємо числовим методом систему рівнянь:

$$\left. \begin{array}{l} \text{Given } a21 + 4 \cdot a31 \cdot P1^3 - \lambda = 0 \\ a23 + 4 \cdot a33 \cdot P3^3 - \lambda = 0 \\ a22 + 4 \cdot a32 \cdot P2^3 - \lambda = 0 \\ Ptr - P1 - P2 - P3 = 0 \end{array} \right\}$$

$$\begin{pmatrix} X1_{\text{опт}} \\ X2_{\text{опт}} \\ X3_{\text{опт}} \\ \text{лямОпт} \end{pmatrix} := \text{Find}(P1, P2, P3, \lambda) ; \quad \begin{pmatrix} X1_{\text{опт}} \\ X2_{\text{опт}} \\ X3_{\text{опт}} \\ \text{лямОпт} \end{pmatrix} = \begin{pmatrix} 33.023 \\ 27.667 \\ 19.31 \\ 144.304 \end{pmatrix} .$$

Контроль виконання обмежень:  $X1_{\text{опт}} + X2_{\text{опт}} + X3_{\text{опт}} = 80$  ; збіглося:  $Ptr = 80$  .

Можна узагальнити цю задачу на випадок N реакторів, зробити також розв'язання функцією значення обмеження:  $X_{\text{опт}}(Ptr)$ .

### Розробка системи управління реакторами

При розробці структурної схеми виходимо із визначених нами (розділи 1–2) функцій системи управління техпроцесом:

- підтримка заданого рівня виходу стиrolу;
- оптимізація розподілу навантаження між реакторами.

Для виконання першої задачі поставимо ПІ (Пропорційно-Інтегральний) регулятор. Виконання другої задачі ускладнено проблемами оперативного вимірювання характеристики  $S(P_1, P_2, P_3)$ . Замість цього використовуємо *індикатор* – величину, що пов'язана з потрібною величиною монотонною залежністю. Такою величиною може бути похідна  $dP_i/dX_i$  – вихід стиrolу на одиницю подачі етилбензолу.

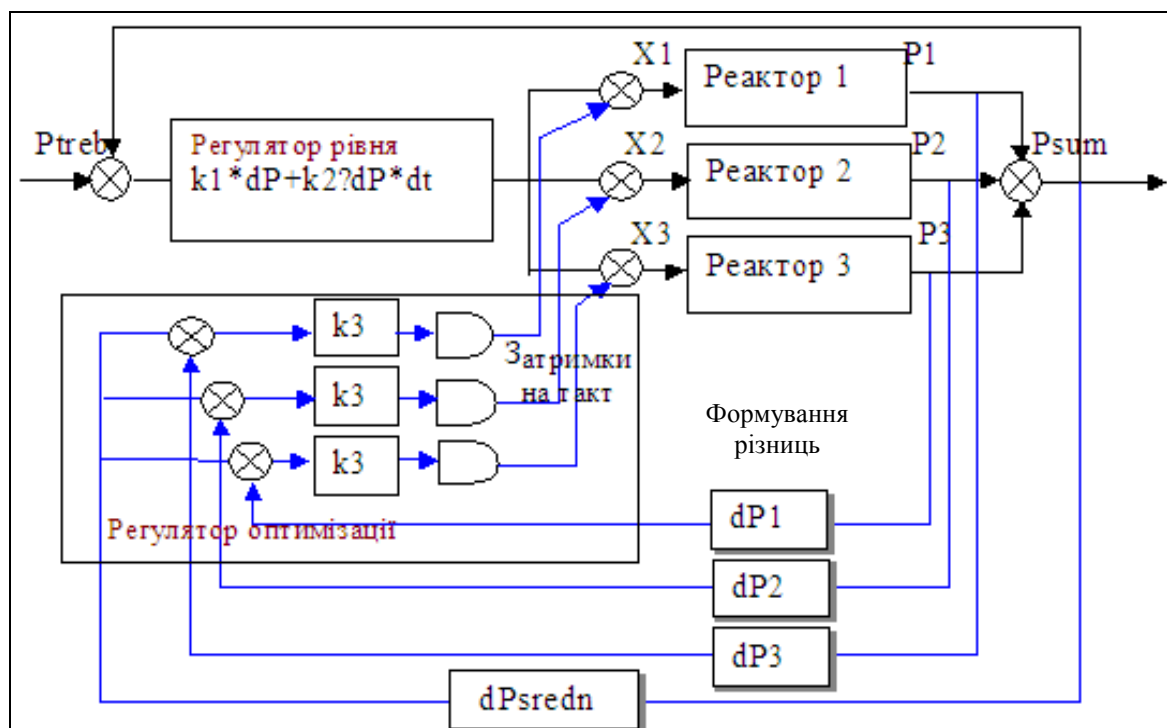


Рис. 4.5. Схема системи оптимального управління хімічними реакторами

Словесна модель регулятора оптимізації розподілу така (рис. 4.5):

- регулятор рівня подає однакові управління в усі реактори,
- вимірюються "відгуки"  $dP_i$  кожного з реакторів на це управління,
- формуються відхилення від середнього, і з затримкою на такт,
- на реактори подаються "вирівнюючі" управління. В результаті збільшується навантаження реакторів, у яких  $dP_i$  вище середнього, зменшується для тих, де  $dP_i$  – нижче середнього. Такий принцип перерозподілу дуже поширений. Для випадку монотонних випуклих функцій усталеним станом системи буде точка екстремуму.

### Розробка базової програми моделювання

На підставі вибраних альтернатив побудови системи і дивлячись на розроблену структурну схему записуємо мовою і за технологією програмування Mathcad програму моделювання системи (рис. 4.6), яка одночасно є власним алгоритмом і блок-схемою. Чому алгоритмом? – сьогодні FOR – NEXT, WHILE, IF стали загальнозрозумілими термінами і неважко прочитати текст програми. Чому блок-схемою? – текст структуровано. Чому це програма? – а тому, що вона безпосередньо виконується в середовищі пакета. Поданий далі текст програми майже не потребує коментарів, але коментарі – стандарт розробки програм.

<pre> PP(kk) := X1,1 ← 5 X2,1 ← 10 X3,1 ← 25 Ip ← 0 P&lt;1&gt; ← Pi(b, X&lt;1&gt;) for k ∈ 1..N   q ← 2·k - 1   Psum ← Pi(b1, X1, q) + Pi(b2, X2, q) + Pi(b3, X3, q)   dPsum ← Ppotribne - Psum   Ip ← Ip + dPsum   dXogr ← kk1·dPsum + kk2·Ip   X&lt;q+1&gt; ← X&lt;q&gt; + (dXogr)   sto ← 100   P&lt;q+1&gt; ← Pi(b, X&lt;q+1&gt;)   dP ← P&lt;q+1&gt; - P&lt;q&gt;   dPm ← mean(dP) + 10 - √100   dXopt ← (kk3 / dPm)   X&lt;q+2&gt; ← X&lt;q+1&gt; + [dXopt(dP1 - dPm)                     dXopt(dP2 - dPm)                     dXopt(dP3 - dPm)]   P&lt;q+2&gt; ← (Pi(b1, X1, q+2)            Pi(b2, X2, q+2)            Pi(b3, X3, q+2)) P </pre>	$X^{(1)} = \begin{pmatrix} 10 \\ 5 \\ 25 \end{pmatrix}$ $X^{(1)} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$ $kk = \begin{pmatrix} 0.5 \\ 0.05 \\ 5 \end{pmatrix}$ $b = \begin{pmatrix} 8 \\ 6 \\ 5 \end{pmatrix}$
--	---

Рис. 4.6. Текст програми моделювання системи оптимального управління хімічними реакторами

В програмі використовуємо векторизацію. Нижче подано для порівняння дві альтернативи подання одного й того ж виразу:

$$\begin{pmatrix} X_{1,q+1} \\ X_{2,q+1} \\ X_{3,q+1} \end{pmatrix} \leftarrow \begin{pmatrix} \text{if}(X_{x1,q+1} < 0, 0, X_{x1,q+1}) \\ \text{if}(X_{x2,q+1} < 0, 0, X_{x2,q+1}) \\ \text{if}(X_{x3,q+1} < 0, 0, X_{x3,q+1}) \end{pmatrix}; X^{(q+1)} \leftarrow 0 \text{ if } X^{(q+1)} < 0.$$

*Векторизація обчислень не тільки скорочує і спрощує запис програм обробки багатомірних даних. Головний виграв векторизації – у швидкості обчислень. Сучасні процесори виконують векторизовані вирази паралельно.*

Програма повертає масив векторів стану системи – розподілень навантажень окремих реакторів в функції часу. Виводимо цей масив:

	1	2	3	4	5	6	7	8	9	10	11	12
PP(kk) =	17.9	31	33.6	29.7	31.7	31.4	33.2	32.7	34.1	33.8	35	34.7
	19	26.8	26.6	23.8	23.7	23.5	23.5	23.2	23.3	23	23.2	22.9
	25	29.6	28.6	26.8	25.9	25.8	24.9	24.7	23.8	23.6	22.8	22.7

### Дослідження системи управління реакторами

Згідно з завданням виконуємо перевірку системи на стійкість та чутливість. Система в цілому нелінійна, тому аналітично можна знайти тільки певні оцінки стійкості та якості регулювання. Реакція лінійної системи не залежить від амплітуди вхідного сигналу. В нашому випадку характеристики реактора – нелінійні, і реакція буде залежати від амплітуди вхідного сигналу. Точні оцінки чутливості і стійкості можна отримати тільки моделюванням. Для дослідження системи та вибору параметрів регулятора  $k_1, k_2, k_3$  створюємо стенд (рис. 4.7). Центр стенда – графіки перехідних процесів. Біля цих графіків розташовуємо блоки вхідних і вихідних даних. Далі подано приклад такого стенда. Вхідні параметри: продуктивність реакторів, параметри закону управління, задана видача:

$$N \equiv 25; q := 1..N \cdot 2; b \equiv \begin{pmatrix} 8 \\ 6 \\ 5 \end{pmatrix}; kk \equiv \begin{pmatrix} 0.5 \\ 0.05 \\ 5 \end{pmatrix}; P_{potribne} = 80; Sum_q := \text{mean}(PP(kk)^{\langle q \rangle}).$$

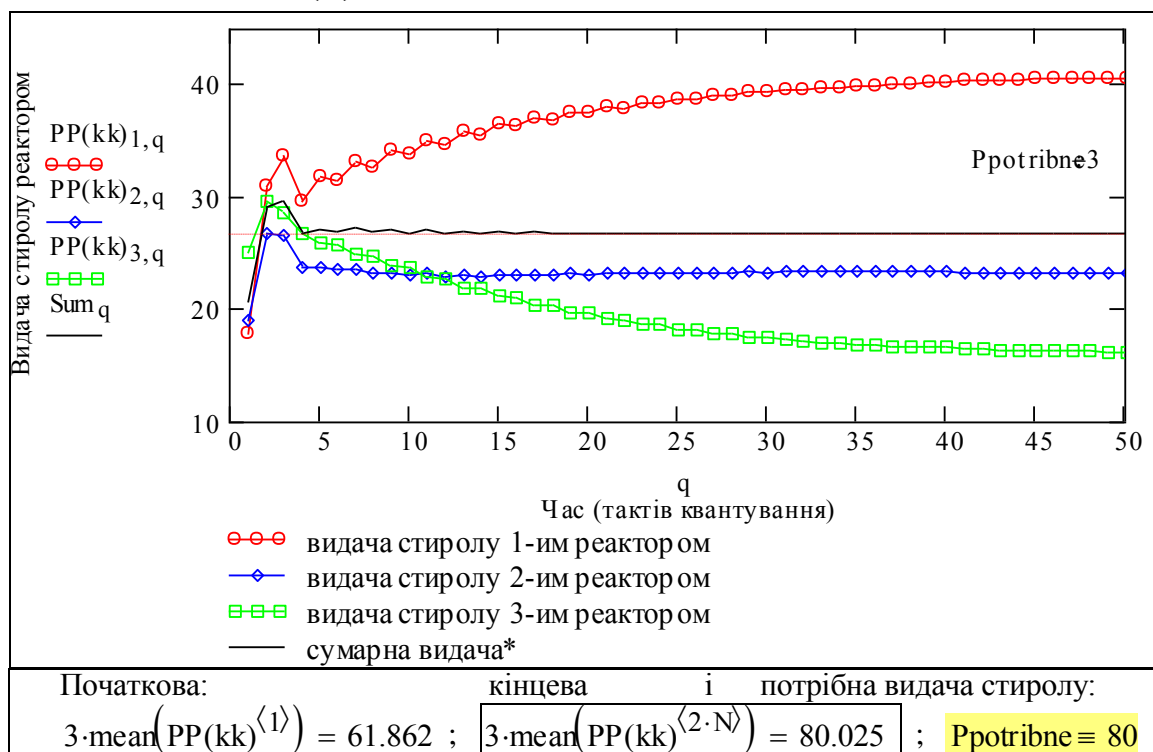


Рис. 4.7. Перехідні процеси в системі реакторів

Початковий і кінцевий (оптимальний) розподіли навантаження реакторів

$$PP(kk)^{\langle 1 \rangle} = \begin{pmatrix} 17.889 \\ 18.974 \\ 25 \end{pmatrix}; \quad PP(kk)^{\langle 2 \cdot N \rangle} = \begin{pmatrix} 40.594 \\ 23.219 \\ 16.213 \end{pmatrix}.$$

З графіка, поданого вище, можна визначити параметри регулятора, що забезпечують стійкість і оптимальність (мінімум експлуатаційних витрат) усталеного стану та малу помилку стабілізації потрібного сумарного виходу. Дослідимо вплив початкових умов на стійкість і усталені значення виходу.



## Дослідження системи управління при варіації початкового розподілу навантаження

Порівняємо перехідні процеси при незмінному загальному навантаженні і незмінних параметрах закону управління, а початкові навантаження окремих реакторів будемо змінювати. На рис. 4.8 подано приклад – два процеси при різних початкових умовах.

Висновки. Стан рівноваги не залежить від початкових умов.

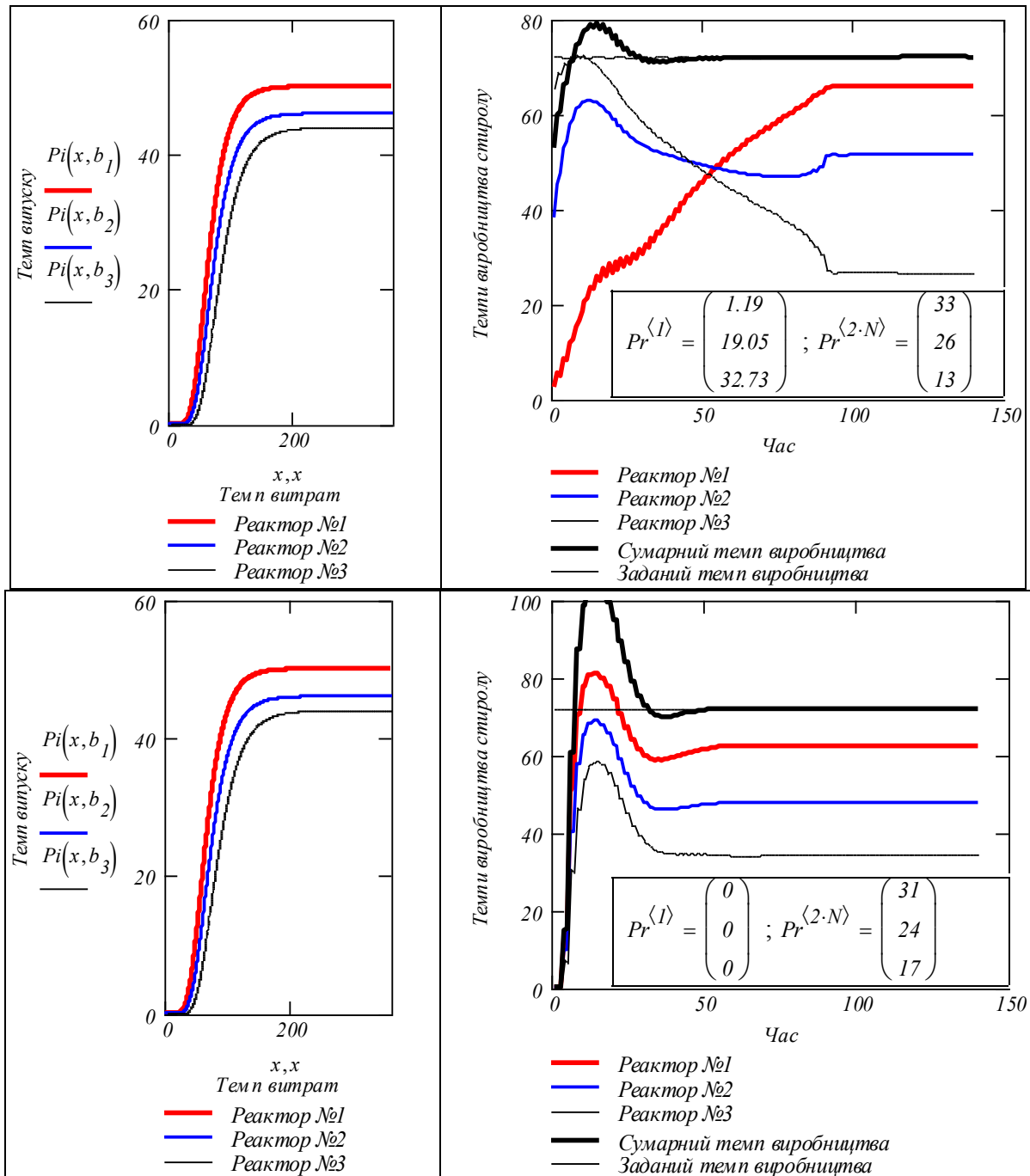


Рис. 4.8. Перехідні процеси при різних початкових умовах

### Завдання для самостійного виконання

Проведіть дослідження впливу загального завантаження системи на стійкість і установлені значення виходу. Модифікуйте програму для побудови функцій впливу певного параметра на показники перехідного процесу – тривалість, установлену похибку, викид. Проведіть комплексне дослідження впливу вибраного параметра: побудуйте функцію впливу і приклади перехідних процесів для характерних точок функції впливу.

### Дослідження системи управління при варіації рівня навантаження

Порівняємо перехідні процеси при незмінних: початковому навантаженні окремих реакторів і параметрах закону управління. Змінюємо задане навантаження (уставку).

На рис. 4.9 подано приклад – два перехідних процеси при різних навантаженнях: 67 та 72 одиниці вимірювання. Наша система є нелінійною і показники перехідних процесів залежать від рівня навантаження.

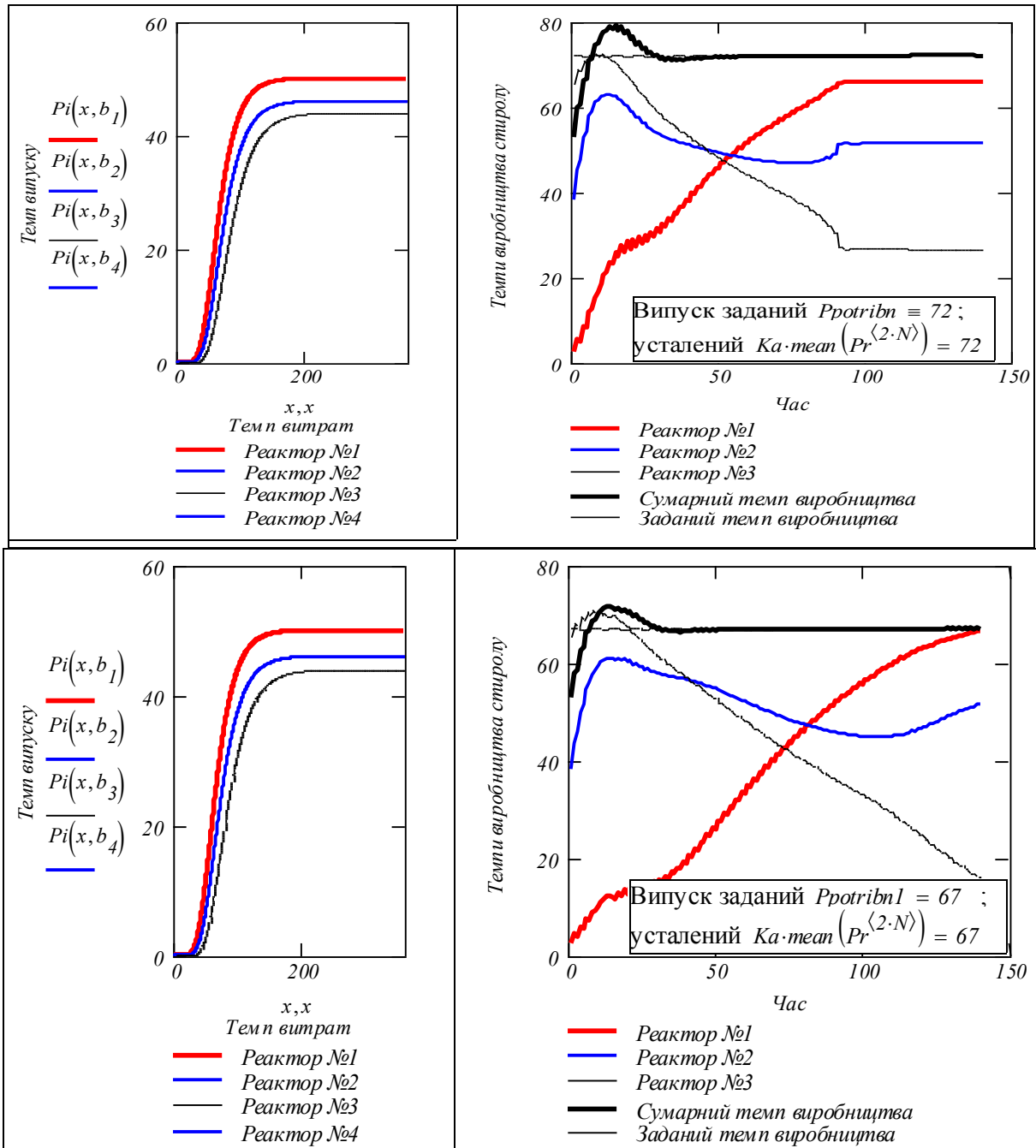


Рис. 4.9. Перехідні процеси для навантажень 72, 67 одиниць

### Розробка математичної моделі і програми моделювання для дослідження впливу інерційності реакторів

Парадоксальною ознакою вдалої математичної моделі є те, що вона "сама себе вбиває", але по-різному: 1) модель виявляється зручною для модифікацій і зникає серед множини більш точних, спеціалізованих моделей; 2) модель виявляється генератором безлічі красивих і правдоподібних результатів, людство "застрягає" на цій можливо ситуативній моделі.

Виконаємо модифікацію моделі. В першому наближенні реактори вважалися статичними нелінійними елементами. В дійсності, при подачі сигналу управління для зміни темпу подачі етилбензолу  $X_i$  не одразу встановлюється потрібний темп, не одразу встановлюються параметри реакції – температура, тиск. І, взагалі, бажано мати більш універсальну модель, придатну і для процесів з великими постійними часу. Суть модифікації в переході від статичної моделі до динамічної:

$$P(b_1, x) = b_1 \cdot x^{0.5} \implies \text{Tr} \frac{d}{dt} P_d(t) + P_d(t) = P_i(b_1, x);$$

$$\frac{d}{dt} P(b_1, x, t) = \frac{1}{\text{Tr}} \cdot (P_i(b_1, x) - P(b_1, x, t)) \implies P_{d_{k+1}} = P_{d_k} + (P_i(b_1, x) - P_{d_k}) \cdot \frac{T}{\text{Tr}}.$$

Задаємо параметри, початкові умови і пишемо *дві версії програми* моделювання реактора  $N_h := 20$ ;  $k := 1..N_h$ ;  $b_1 = 8$ ;  $\text{Tr} := 2.5$ ;  $T := .5$ ;  $P_{d_1} := 0$ ;  $x := 10$ .

$$P_{d_{k+1}} := P_{d_k} + (P_i(b_1, x) - P_{d_k}) \cdot \frac{T}{\text{Tr}}.$$

Як альтернативу різницево-рівнянню робимо програмний модуль.

```

pD(b, x) :=
  pd1 ← 0
  for k ∈ 1..Nh
    pd_{k+1} ← pd_k + (Pi(b, x) - pd_k) · T / Tr
  pd
  
```

Будуємо графіки, порівнюємо (рис. 4.10).

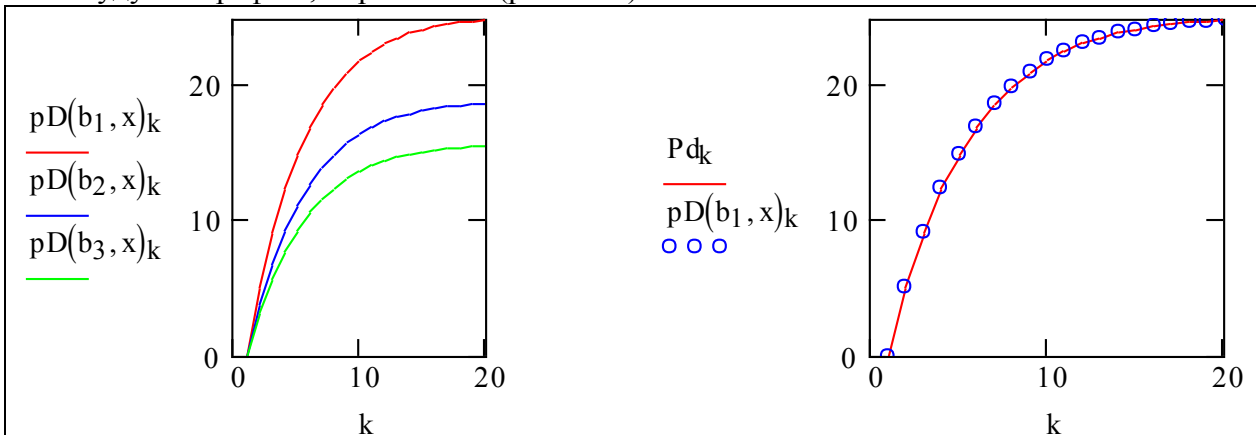


Рис. 4.10. Тестування моделей інерційності реактора

Модифікуємо попередню програму – додаємо відповідні рядки, подібні до тих, що подані вище, але не скалярні, а матрично-векторні (рис. 4.11).

Використовуємо векторизацію обчислень (стрілки над виразами означають: "виконати операцію над усіма компонентами вектора чи матриці"). В багатопроцесорних системах саме ці елементи програми можуть виконуватись паралельно.

В закон управління додана складова – аналог похідної від помилки. Програма з певних причин не виводить систему до екстремального розподілення навантажень. Одна з цих причин в тому, що для перерозподілу навантаження потрібно додавати до сигналу управління періодичні тестові імпульси – саме на базі реакцій на збурення ідентифікуються характеристики реакторів. Без тестових сигналів система "зависає" біля оптимального стану. Розглянута в розділі 1 адаптивна система теж вимагала для настроювання тестових сигналів.

P2(kq) :=	$X^{(1)} \leftarrow \begin{pmatrix} 16 \\ 25 \\ 15 \end{pmatrix}$ $dPsum \leftarrow 0$ $Ip \leftarrow 0$ $P^{(1)} \leftarrow \overrightarrow{Pi(b, X^{(1)})}$ for k ∈ 1..N <div style="border-left: 1px solid black; padding-left: 10px; margin-left: 20px;"> <math display="block">q \leftarrow 2 \cdot k - 1</math> <math display="block">Psum \leftarrow Pi(b_1, X_{1,q}) + Pi(b_2, X_{2,q}) + Pi(b_3, X_{3,q})</math> <math display="block">dPp \leftarrow dPsum</math> <math display="block">dPsum \leftarrow Ppo - Psum</math> <math display="block">Dp \leftarrow dPsum - dPp</math> <math display="block">Ip \leftarrow Ip + dPsum</math> <math display="block">dXogr \leftarrow kq_1 \cdot dPsum + kq_2 \cdot Ip + kq_4 \cdot Dp</math> <math display="block">Друга \leftarrow "половина"</math> <math display="block">X^{(q+1)} \leftarrow \overrightarrow{(X^{(q)} + dXogr)}</math> <math display="block">P^{(q+1)} \leftarrow P^{(q)} + \left( \overrightarrow{Pi(b, X^{(q+1)})} - P^{(q)} \right) \cdot \frac{T}{Tr}</math> <math display="block">dP \leftarrow P^{(q+1)} - P^{(q)}</math> <math display="block">dPm \leftarrow \text{mean}(dP)</math> <math display="block">dXopt \leftarrow \frac{kq_3}{dPm}</math> <math display="block">X^{(q+2)} \leftarrow X^{(q+1)} + \overrightarrow{dXopt \cdot (dP_2 - dPm)}</math> <math display="block">P^{(q+2)} \leftarrow P^{(q+1)} + \left( \overrightarrow{Pi(b, X^{(q+2)})} - P^{(q+1)} \right) \cdot \frac{T}{Tr}</math> </div>
	P

Рис. 4.11. Програма моделювання системи реакторів з урахуванням інерційності

В цій програмі: P – вихід стиролу, X – подача етилбензолу. Задаємо параметри ПД-регулятора, для порівняння наводимо параметри ПІ-регулятора. Задаємо рівень сумарного виходу стиролу:

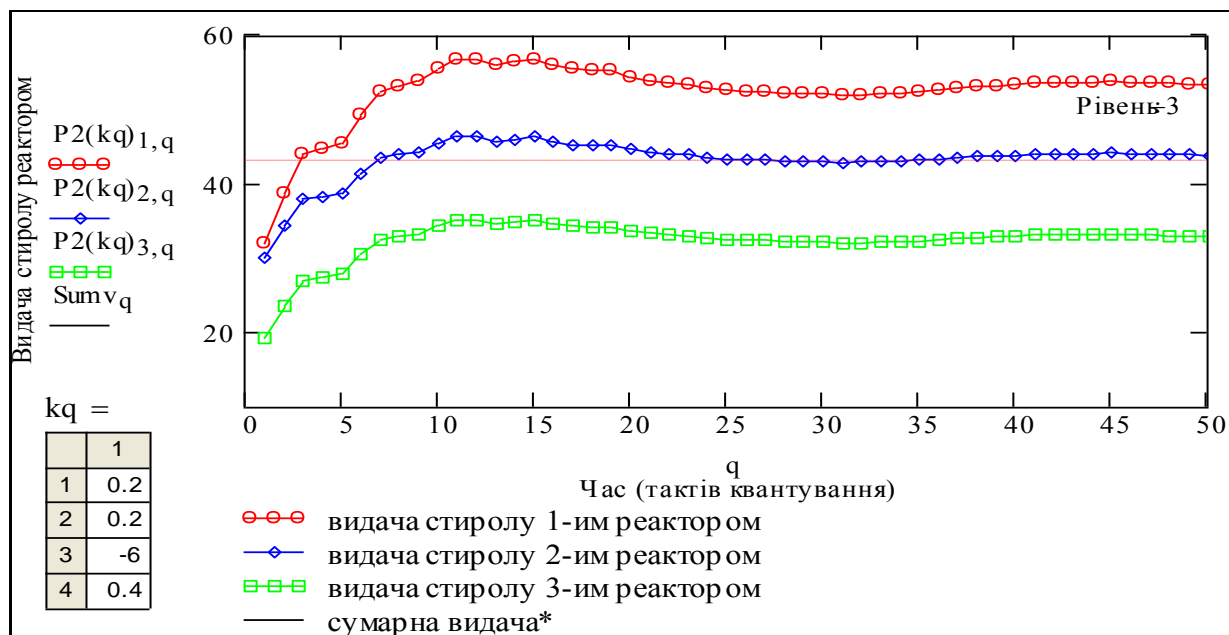
$$kk_1 = 0.5 ; \quad kk_2 = 0.05 ; \quad kk_3 = 5 ; \quad \boxed{Ppo \equiv 130} ;$$

$$\boxed{kq_1 := .2} ; \quad \boxed{kq_2 := 0.2} ; \quad \boxed{kq_3 := -6} ; \quad \boxed{kq_4 := .4} .$$

Задаємо параметри ефективності реакторів:

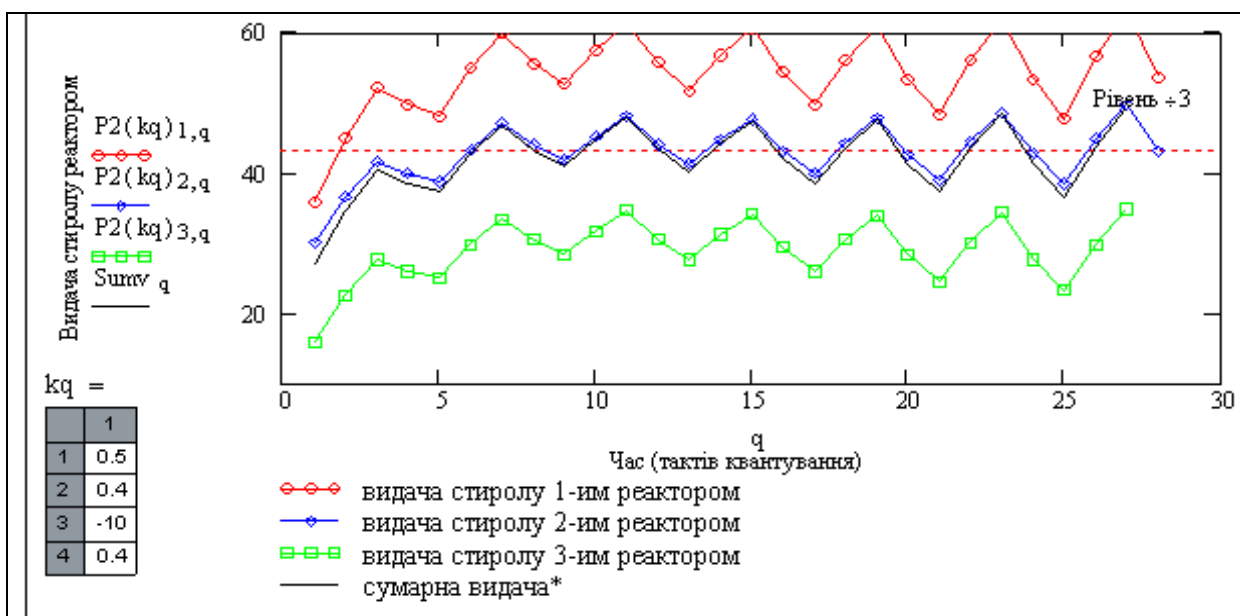
$$\text{Рівень} := P_{rc}; \quad P2(kq)^{\langle 2 \cdot N \rangle} = \begin{pmatrix} 53.337 \\ 43.866 \\ 32.959 \end{pmatrix}; \quad b = \begin{pmatrix} 8 \\ 6 \\ 5 \end{pmatrix}.$$

Будуємо графіки перехідних процесів (рис. 4.12).



Початкова:  $3 \cdot \text{mean}(P2(kq)^{\langle 1 \rangle}) = 81.365$  | кінцева  $3 \cdot \text{mean}(P2(kq)^{\langle 2 \cdot N \rangle}) = 130.16$  | і потрібна видача стиролу: Рівень = 130

Рис. 4.12. Перехідні процеси в системі реакторів при врахуванні інерційності



Початкова:  $3 \cdot \text{mean}(P2(kq)^{\langle 1 \rangle}) = 81.588$  | кінцева  $3 \cdot \text{mean}(P2(kq)^{\langle 2 \cdot N \rangle}) = 134.824 + 11.586i$  | і потрібна видача стиролу: Рівень = 130

Рис.4.13. Перехідні процеси в системі реакторів. Границя стійкості

Проаналізуємо перехідні процеси, подані на рис. 4.13. Система є нелінійною і має режим пошуку оптимального розподілення. Ці фактори є причиною особливостей поведінки системи, нехарактерних для лінійних систем.

Система втрачає стійкість при збільшенні параметрів регулятора відносно номінальних. При цьому виникає специфічний режим з коливаннями в 5 кроків квантування. Це пов'язано з режимом настроювання на оптимальний розподіл навантаження. Для усунення цього недоліку потрібні більш досконалі алгоритми настроювання.

### Розробка альтернативної форми подання результатів моделювання

Наша система нелінійна. Аналітично можна знайти тільки певні оцінки стійкості та якості регулювання. Основним „інструментом” синтезу (вибору параметрів) регулятора є моделювання. І тут важливим є зручність і природність інтерфейсу.

Розглянемо альтернативні версії інтерфейсу (стенда) для вибору параметрів регулятора  $k_1, k_2, k_3$ . Центр стенда – графіки перехідних процесів. Біля цих графіків розташовуємо зони вхідних параметрів і показників перехідних процесів. Далі подано "заготовку" такого стенда. Відмінність подання перехідних процесів від графіків на рис. 4.7–4.9 – використання графіків "в приростах" (стандартний графік в Excel). Дійсно, глобальна ціль системи – підтримка заданого сумарного випуску продукту. На рис. 4.14, 4.15 подано динаміку перерозподілу заданого навантаження між окремими реакторами.

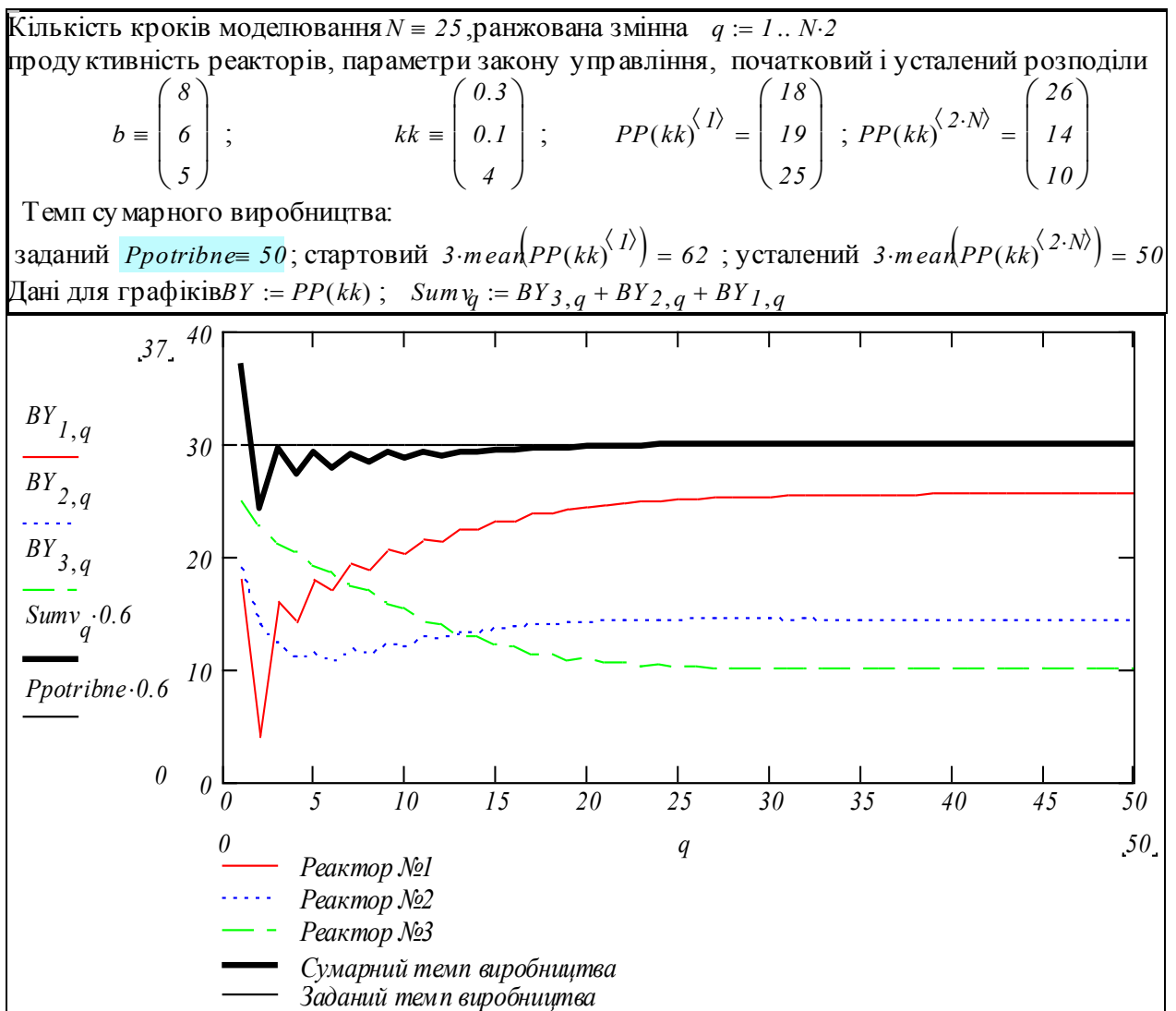


Рис. 4.14. Перехідні процеси в системі реакторів. Варіант подання

Можемо бачити наочно результати роботи двох регуляторів – сумарної видачі стиролу і оптимального розподілу навантаження. Поряд подаються графіки характеристик реакторів – узагальнені виробничі функції. Головний висновок – система управління підтримує заданий сумарний темп випуску продукту і оптимальний розподіл цього темпу між реакторами.

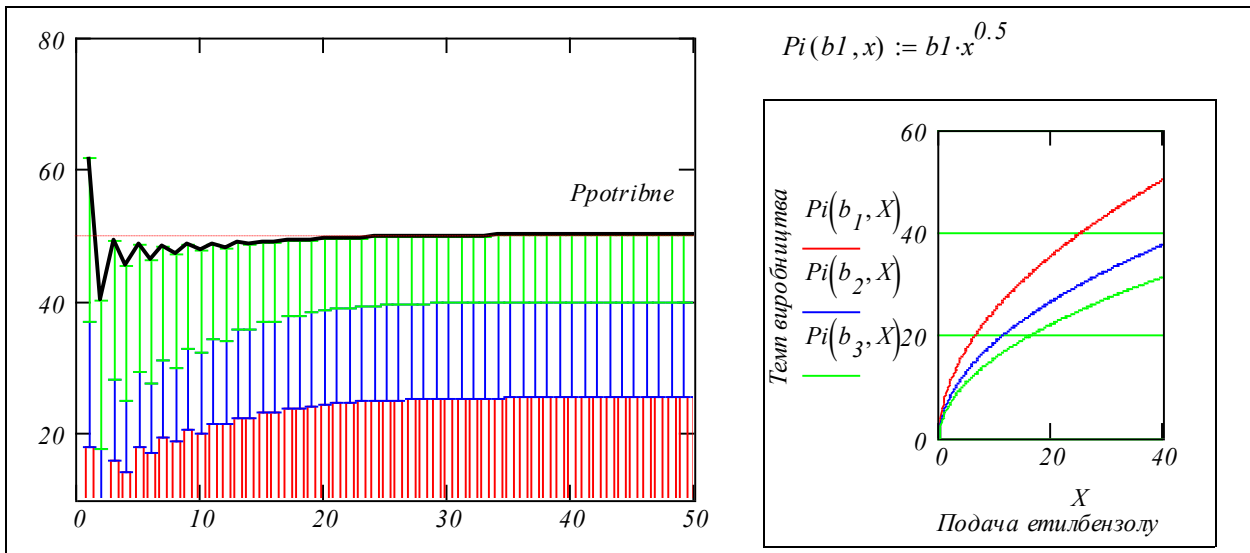


Рис. 4.15. Перехідні процеси в системі реакторів. Варіант подання

### Висновки

Розроблено математичну модель системи паралельно працюючих технологічних агрегатів. Виробництво стиролу – тільки приклад інтерпретації моделі. Результати моделювання підтвердили достатню адекватність моделі і придатність її для подальших модифікацій.

Розроблено систему управління, в якій працює "колектив" з двох регуляторів – регулятора стабілізації заданого рівня видачі стиролу і регулятора оптимізації розподілу навантаження між окремими реакторами. Регулятори працюють послідовно, так що регулятор рівня забезпечує інформацією регулятор оптимального розподілу.

Розроблена система моделювання дозволяє досліджувати такий незручний для автоматизації і, тим більше, небезпечний для натурних експериментів, нелінійний об'єкт – реактор. Система моделювання і алгоритм адаптивного управління мають великий потенціал подальшого розвитку і вдосконалення.

### Контрольні запитання

1. Які задачі повинна виконувати САУ, які вимоги висуваються до САУ? Вхідні і вихідні дані задачі синтезу.
2. Як розшифровується: "ПІД-регулятор", який закон регулювання використано для підтримки заданого рівня випуску продукту?
3. Опишіть (словесно, алгоритмічно) логіку роботи регулятора оптимального розподілу навантаження між реакторами.
4. Причини, що обумовлюють встановлення різних навантажень для різних реакторів.
5. Що таке "екстремальний регулятор"?
6. Суть "методу невизначених множників Лагранжа" і порядок його застосування в оптимізаційних задачах.
7. До чого власне адаптується система?
8. Чи можна подібну систему застосувати для управління банком? Якщо можна, то як?

### Завдання для самостійної роботи

1. Побудуйте графіки перехідних процесів для випадків, коли всі параметри регулятора:
  - а) збільшені на 5%, б) зменшені на 5%.
2. Промодельте роботу системи при аварійному відключенні одного з реакторів.
3. Промодельте систему при плановому відключенні одного з реакторів.
4. Промодельте систему при плановому введенні нового реактора.
5. Зробіть "універсальну" програму для системи з довільним і змінним числом реакторів.

## 4.2 Система оптимального управління банком

**Постановка проблеми.** Ефективна система банків – одна з необхідних умов ефективності національної економіки і добробуту кожного громадянина. Потенційно банки оптимізують розподіл суспільних ресурсів між споживанням, накопиченням і розвитком. Клієнтами банків сьогодні є більша частина населення розвинених країн.

Банк – специфічна система масового обслуговування. Специфіка банківської діяльності у високому рівні конкуренції та "примхливості" клієнтів. Ці два фактори разом породжують високий рівень невизначеності і ризиків.

Загально визнано, що рівень банку визначається інтенсивністю використання і якістю математичних моделей – засобів прогнозування і управління. Життя змушує банки розробляти необхідні робочі моделі. Однак корисні моделі не виходять за межі організації. В програші залишається в першу чергу освіта – майбутніх спеціалістів вчать, в основному, на словесних та елементарних загальних моделях. Знання основ банківської діяльності потрібні сьогодні практично всім. Парадоксальний аспект проблеми моделювання банків в Україні в тому, що це "нічийна територія". Для спеціалістів з фінансів це "математика", для спеціалістів з технічних наук – це "економіка". Хто повинен розробляти моделі? Фінансист, програміст, математик? Це питання не розв'язується дискусіями. В даній роботі подано процес побудови робочої моделі банківської системи на базі теорії управління і системного аналізу.

**Аналіз останніх досліджень і публікацій.** Обсяг публікацій з банківської справи зростає експоненційно. В Росії і Україні щорічно видаються десятки посібників і підручників – своїх і перекладів з англійської. Однак публікацій з новими і конструктивними результатами небагато. Причини: системні – концентрація і розсіювання знань по джерелах публікацій (закони Бредфорда, Зіпфа); комерційні – ефективні моделі звичайно існують як "ноу-хау"; класифікаційні – помилкова класифікація задачі.

Переглянута література була поділена на три класи (що можуть перетинатись): а) фундаментальні роботи з моделювання і управління; б) навчальні посібники і підручники, в) дисертації. Особливістю літератури класів б) і в) є вторинність – це поглиблення якоїсь колії, наприклад, моделі на базі міжгалузевого балансу, нечіткої логіки та нейронних мереж (особливо популярних в фінансових задачах) та ін. Виділено певні роботи – представники класів: типовий підручник для фінансистів [6, 7], підручник з елементами розробки моделей [8], підручник фундаментального рівня [9], класичний фундаментальний підручник-першоджерело [5]. В дисертаціях з банківської справи (захищених як за економічними, так і за технічними спеціальностями) виявлено багато цінних висновків і спостережень. Сьогодні з'являється все більше літератури безпосередньо з моделювання банківських систем. Однак все це не складається в єдину систему моделей. Тому вирішено не узгоджувати знайдене в літературі, а зробити робочу модель (таку, що виконується в середовищі математичного пакета) з самого початку – від твердо встановлених емпіричних законів банківської справи до робочих моделей функціонування банку. Робимо це з використанням методології конструювання математичних моделей [4]. Робоча модель є інструментом для пошуку оптимальних, адаптивних і просто безпечних управлінь, і тут раціонально йти від першоджерел [1–3] з теорії управління.

**Невирішені частини проблеми.** Сьогодні потрібні не стільки нові моделі для нових систем і процесів, скільки раціональні технології розробки математичних моделей, здатних відобразити об'єкт, що еволюціонує. Потрібен інший підхід до моделювання, зокрема, робота з моделювання, де формули, набрані в текстовому редакторі, є прикладом нераціональної технології конструювання моделей.

**Постановка задачі.** Виділимо з численних тактичних і стратегічних задач банку базову задачу: "дешево брати в оренду гроші" і "дорого здавати". Виправданням для такої грубої моделі є те, що її потім можна необмежено надбудовувати і розбудовувати. Подано на рис. 4.16 схему банківської системи, що відповідає базовій задачі банку.



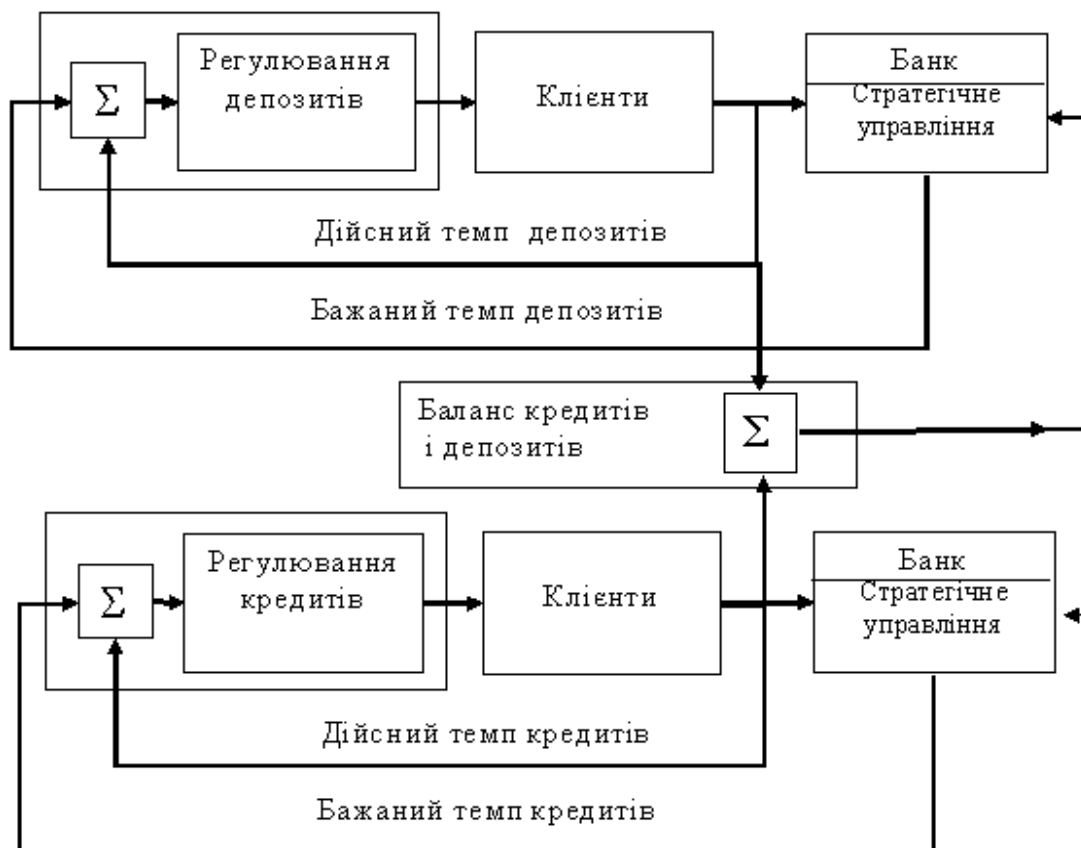


Рис. 4.16. Схема банківської системи

Ця схема подібна схемі двоканальної системи автоматичного регулювання. Одразу можна бачити одну з головних причин складності – *непрогнозований і метаморфозний об'єкт управління – клієнти*. Однак велика кількість і різноманітність клієнтів є скоріше фактором спрощення – в умовах стабільної національної економіки можна усереднювати та агрегувати характеристики клієнтів.

На жаль, в певних умовах поведінка клієнтів буває уніфікованою і детермінованою – це панічне вилучення коштів з банку. Банки, на відміну від виробничих і торговельних організацій, певним чином зростаються з клієнтами. Очевидно, що треба розглядати банк і клієнтів як єдину *банківську систему*: банк повинен створювати процвітаючих і надійних клієнтів, клієнти повинні дбати про ефективність і надійність свого банку.

### Розробка функціональних моделей

Потоки (темпи) депозитів і кредитів мають певну структуру – розподіли за тривалістю, процентними ставками, надійністю та ін. На першому етапі розглядаємо агреговані усереднені характеристики цих потоків. Працездатну агреговану модель потім неважко модифікувати для врахування структури потоків. Підсистема „депозити” (рис. 4.16) перетворює вхідний темп депозитів у вихідний згідно з правилами і умовами вкладання коштів на депозит і зняття їх. Вхідний темп депозитів залежить від багатьох чинників: ставки депозиту, зростання економіки, темпу інфляції, конкуренції, настроїв і сподівань клієнтів та ін. На першому етапі розглядаємо тільки залежність темпу депозитів від середньої ставки депозиту.

### Модель вхідного темпу депозитів

Будемо вважати стан економіки і стан банку (частку фінансового ринку) стабільними. Припускаємо, що кожному рівню ставки депозитів відповідає певний усталений темп депозитів, тобто існує монотонно зростаюча функція ставки депозиту  $tdp(rd, A)$ , де  $rd$  – ставка депозиту,  $A$  – потенційний (максимально можливий для даного регіону і періоду часу) темп депозитів.

Реальний темп депозитів тільки в ідеальних умовах досягатиме усталеного рівня. Цей процес залежить від темпу зростання інформованості клієнтів, наявності прикладу і порад інших клієнтів, а також від ірраціональних та випадкових факторів. Сукупну дію названих факторів на потенційних клієнтів при зміні ставки депозиту задаємо різницеvim рівнянням першого порядку.

$$tdvx_{k+1} = tdvx_k + Kd \cdot (tdp(rd, A) - tdvx_k) \cdot dT,$$

де  $tdvx_k$  – поточний темп депозитів;  $dT$  – крок обчислень;  $Kd$  – коефіцієнт реакції темпу депозитів на зміну ставки.

Враховуємо також сезонність та випадкову складову і формуємо відповідний програмний модуль (рис. 4.17).

$Dpo(rd, dp) =$	<pre> for k ∈ 1..Km     kolyv ← 1 + αd · sin(ом · π · k) – "тренд пропозиції депозитів"     tdvx0 ← tdp(rd, Adp) · kolyv – "максимальний темп депозитів"     tdvx_{k+1} ← tdvx_k + <math>\frac{1}{Td_p}</math> · (tdvx0 – tdvx_k) · dT – "поточний темп"     tdvx_{k+1} ← rnorm(1, tdvx_{k+1}, σm)_1 – "випадкова реалізація темпу"         </pre>
-----------------	--

Рис. 4.17. Модель вхідного темпу депозитів

#### Модель вихідного темпу депозитів

Модель вихідного темпу депозитів формуємо у виді програмного модуля, текст якого подано на рис. 4.18.

$dprvux(rd, xd, Td) =$	<pre> kzd ← round(<math>\frac{Td}{Tp} \cdot Np</math>) – "кроків повернення депозитів" for k ∈ 1..Np     vupad ← rnd(nakor) + (1 – nakor) – "випадковість"     xdd_k ← <math>\begin{cases} xd(k) &amp; \text{if } (k &lt; kzd) \text{ – "повний вхідний темп"} \\ xd(k) + zald_{k-1} &amp; \text{otherwise} \end{cases}</math>     pas1 ← <math>\begin{cases} 0 &amp; \text{if } k &lt; kzd \text{ – "вихідний темп без процентів"} \\ xdd_{k-kzd} &amp; \text{otherwise} \end{cases}</math>     pas2 ← pas1 · (1 + rd · Td) – "вихідний темп з процентами"     pas3 ← pas2 · vupadko – "дійсний вихідний темп"     zald_k ← pas2 – pas3 – "залишок депозитів"     vux^{(k)} ← (stack(xdd_k, pas3, zald_k) – вектор) стану         </pre>
------------------------	---

Рис. 4.18. Модель вихідного темпу депозитів

Входи програмного модуля – ставка депозитів  $rd$ , середній термін депозиту і вхідний темп депозитів (фактично це програмний модуль вхідного потоку депозитів), виходи – вихідний темп депозитів без урахування залишків, залишки депозитів.

### Модель вхідного темпу кредитів

Як і для темпів депозитів, припускаємо, що кожному рівню ставки кредитів відповідає ustalений темп кредитів, тобто існує монотонно спадна функція ставки кредиту  $tkr(rk, A)$ , де  $rk$  – ставка кредиту,  $A$  – потенційний (максимально можливий для даного регіону і періоду часу) темп кредитів.

Реальний темп кредитів тільки в ідеальних умовах досягатиме потенційного рівня – зі зростанням інформованості клієнтів, дії прикладу інших клієнтів, дії ірраціональних та випадкових факторів. Процес реакції потенційних клієнтів на зміну ставки кредиту вважаємо і задаємо різницеvim рівнянням першого порядку

$$tkvx_{k+1} = tkvx_k + Kk \cdot (tkr(rk, A) - tkvx_k) \cdot dT,$$

де  $tkvx_k$  – поточний темп кредитів,  $dT$  – крок обчислень,  $Kk$  – коефіцієнт реакції темпу кредитів на зміну ставки. Враховуємо також сезонність та випадкову складову.

### Модель вихідного темпу кредитів

Модель вихідного темпу кредитів формуємо на базі вбудованої фінансової функції пакета "виплата боргу рівними частками протягом терміну кредиту":

$$pmt \left( \frac{\text{процент}}{\text{числ\_пер\_рік}}, \text{число\_періодів}, \text{борг\_стартовий}, \text{борг\_кінцевий} \right).$$

Вихідний потік буде визначатись виразом, подібним інтегралу згортки:

$$brgs(k, rk, xk, Tk) := - \sum_{m=1}^k pmt[rk \cdot \Delta t, (Tk \div \Delta t), xk(m), 0] \cdot [(k-m) < (Tk \div \Delta t)],$$

де  $\Delta t$  – крок моделювання,  $Tk$  – середня тривалість кредиту,  $rk$  – ставка кредиту,  $xk(m)$  – значення вхідного темпу кредитів в дискретний момент часу  $m$ .

### Підсистема управління

Природно використати математичну модель банківської системи для пошуку ефективних або хоч безпечних управлінь. На першому етапі поставимо за мету – сформуувати закони управління банківською системою на базі застосування методології і методів теорії управління. Цілі управління для першого етапу – максимізація доходу банку та балансування темпів кредитів і депозитів. Потім на базі працездатної моделі можна розглянути більш тонкі управління – згладжування вхідних темпів кредитів і депозитів, диференціювання ставок, урахування прогнозів валютних курсів, економічних показників та ін. Розглядаємо як змінні управління ставки кредитів і депозитів.

### Логіка управління

Виділимо "логічні управління": управління балансом і управління для оптимізації доходу. Управління для збалансування темпів кредитів і депозитів повинно одночасно змінювати ставки кредитів і депозитів так, щоб темпи "йшли назустріч". Для першого наближення беремо простіший, пропорційний закон управління балансом:

$$\text{зміна ставки кредиту } \boxed{\Delta rk = K1k \cdot \delta ba}; \text{ зміна ставки депозиту } \boxed{\Delta rd = -K1d \cdot \delta ba},$$

де  $K1k$ ,  $K1d$  – "коефіцієнти підсилення",  $\delta ba$  – різниця темпів депозитів і кредитів (помилка регулювання). Різні знаки обумовлені тим, що для збільшення темпу кредитів треба зменшувати ставку, а для збільшення темпу депозитів – збільшувати її.

Управління доходом банку відноситься до іншого класу – екстремального: нам потрібно не стабілізувати значення доходу на якомусь заданому рівні, а максимізувати за рахунок вибору значень ставки кредиту –  $rk$ . Ставка депозиту прив'язується до ставки кредиту за рахунок умови збалансування темпів кредитів і депозитів.

Логіка екстремального управління така: на поточному кроці визначаємо приріст доходу за попередній період, якщо дохід зростає – продовжуємо змінювати управління в тому ж напрямку, що і на попередньому кроці; якщо дохід зменшується – змінюємо управ-

лінійя в протилежному напрямку. Це досить примітивна і наївна логіка "індикаторної поведінки". Можна її вдосконалювати введенням фільтрації (не реагувати на випадкові і короткочасні збурення) і прогнозування (реагувати не тільки на зміну доходу, але і на тенденції зміни доходу). Перекладаємо ці міркування на мову рівнянь та алгоритмів. Глянемо також на банківську систему, як на систему "попит-пропозиція":

- процентна ставка – це ціна грошей;
- залежність "ставка кредиту – темп кредитів" – це крива попиту на гроші;
- залежність "ставка депозиту – темп депозитів" – це крива пропозиції грошей.

На рисунку 4.19 подано постановку і розв'язання задачі про оптимальну різницю ставок – якщо існують монотонно спадна функція попиту на кредити і монотонно зростаюча функція пропозиції депозитів, то існує оптимальна різниця ставок, що дає максимум процентного доходу. Бачимо, що задача пошуку оптимальної різниці ставок еквівалентна геометричній задачі про вписаний прямокутник з максимальною площею. Можемо побачити (рис.4.19) також, що найбільший рівноважний стан досягається при рівних ставках кредиту і депозиту.

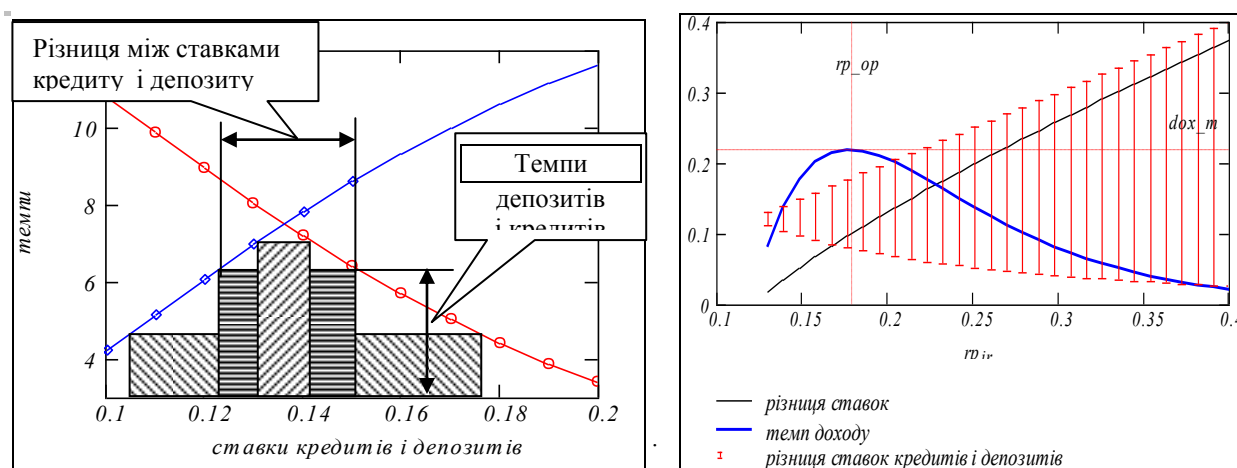


Рис. 4.19. До визначення оптимальної різниці ставок кредитів і депозитів

Зміна ставок кредитів і визначається дією розглянутих двох регуляторів:

$$rK_j = rK_{j-1} + \Delta 1rk + \Delta 2rk; \quad rD_j = rD_{j-1} + \Delta 1rd + \Delta 2ra,$$

де  $\Delta 1rk$ ,  $\Delta 1rd$  – управління від регулятора балансу,  $\Delta 2rk$ ,  $\Delta 2ra$  – управління від екстремального регулятора.

### Програма моделювання

На рис. 4.20 подано схему програми моделювання. Інакше це називається "програма в псевдокодах". Програма складається з функціональних модулів, розглянутих вище, – вхідних і вихідних темпах кредитів і депозитів, трендах і невизначеностях, алгоритмах управління. За рахунок настроювання і заміни модулів програма дозволяє відобразити будь-яку специфіку функціонування банківської системи. Програма написана в *середовищі мови програмування математичного пакета*. Ця мова максимально наближена до природної мови математики, тому текст програми практично не відрізняється від запису системи рівнянь математичної моделі. Саме тому використовуємо термін "робоча модель".

Входи програми зібрані у матрицю з вісімнадцяти параметрів, вихід програми – десять дискретних функцій часу: вхідні і вихідні темпи депозитів, баланси "вхід-вихід" кредитів і депозитів, ставки кредитів і депозитів, загальний баланс банку та баланс вхідних темпів кредитів і депозитів.

**Інтерфейс програми.** Технологія конструювання моделей ставить специфічні вимоги і до тексту програми і до інтерфейсів. Це можливість доступу до всіх елементів програми, орієнтованість інтерфейсу на обчислювальний експеримент. Вхідні дані для програми (рис. 4.21) вводяться з коментарями, у "змістовному вигляді", а потім збираються у

матрицю параметрів  $mp$ . В першому рядку матриці зібрані параметри вхідних і вихідних потоків кредитів, в другому – те ж саме для депозитів.

$MBU(mp) =$	$dtk \leftarrow$ "крок обчислень" $ome \leftarrow$ "частота сезонних коливань" $kpk \leftarrow$ "кількість кроків повернення кредиту" $kpd \leftarrow$ "кількість кроків повернення депозиту" $rK_1 \leftarrow$ "стартова ставка кредиту" $rD_1 \leftarrow$ "стартова ставка депозиту" "стартові значення темпів кредитів і депозитів:" "стартове значення балансу:" for $j \in Z..Np1$ – "цикл по кроках моделювання" "максимальний рівень вхідного темпу кредитів" "поточне значення вхідного темпу кредитів" "випадкова реалізація вхідного темпу кредитів" "темп повернення поточного кредиту" "вихідний темп кредитів" "баланс кредитів" "максимальний рівень вхідного темпу депозитів" "поточне значення вхідного темпу депозитів" "випадкова реалізація темпу" "алгоритм повернення депозитів" "вихідний темп депозитів" "залишок депозитів" "баланс депозитів" "баланс банку" "алгоритм управління балансом" "алгоритм управління максимізації доходу" "матриця вихідних масивів"
-------------	--

Рис. 4.20. Схема програми моделювання банківської системи

**Результати тестування програми.** Програма моделювання не мала прототипів, тому виконано комплексне тестування для перевірки:

"синтаксичної" коректності (виявлення банальних помилок – втрачених складових і переплутаних знаків);

"семантичної" коректності (перевірка результатів моделювання при нульових та рівних ставках, при однакових термінах кредитів і депозитів);

"прагматичної" коректності (відповідність статистичним даним, можливість налаштування і модифікації програми).

Спочатку була випробувана модель з фіксованими ставками кредитів і депозитів, потім в модель був введений модуль управління з найпростішими законами управління –

пропорційними. На рис. 4.22 подано приклади двох реалізацій перехідних процесів на відповідних фазових площинах з урахуванням випадкових збурень.

Резерв банку  $резерв := -3$  Залишкова частка депозиту  $vnkr := 0.0$ .  
**Ставки:** кредиту  $rko := 0.08$ , депозиту  $rdo := 0.06$ , середні терміни:  
 кредиту  $Tko := 5$ ; депозиту  $Tdo := 3$ . **Клієнти:**  
**Кредити:** інерційність  $Tkr1 := 1.1$ ; максимальний рівень  $po ko := 21$ ; сезонність  $ask := 2$  частота  $om := 1$ , розкид  $\sigma k := 0.1$ . Параметри попиту  
**Депозити:** інерційність  $Tdp1 := 1.1$ ; максимальний рівень  $pod o := 21$ ; сезонність  $asd := 2$  частота (=) розкид  $\sigma d := 0.1$ . Параметри пропозиції  
 Період моделювання  $Tp1 := 30$ ; число кроків  $Np1 := 150$ ;  $k := 1..Np1$ ;  
 $\Delta t := Tp1 \div Np1 \quad \Delta t = 0.2$ . **Параметри законів управління:**  
 регулювання балансу  $K1k := 0.003$ ;  $K1d := 0.003 \cdot \text{натисни}$ ;  
 максимізація доходу  $K2k := 0.0003$ ;  $K2d := 0.0003$ ;  $\Delta o := 0.0$ .  
 Матриця параметрів  $mp := \begin{pmatrix} rko & Tko & po ko & ask & \sigma k & om & K1k & K2k & Tkr1 \\ rdo & Tdo & pod o & asd & \sigma d & vnkr & K1d & K2d & Tdp1 \end{pmatrix}$

Рис. 4.21. Модуль вхідних даних банківської системи

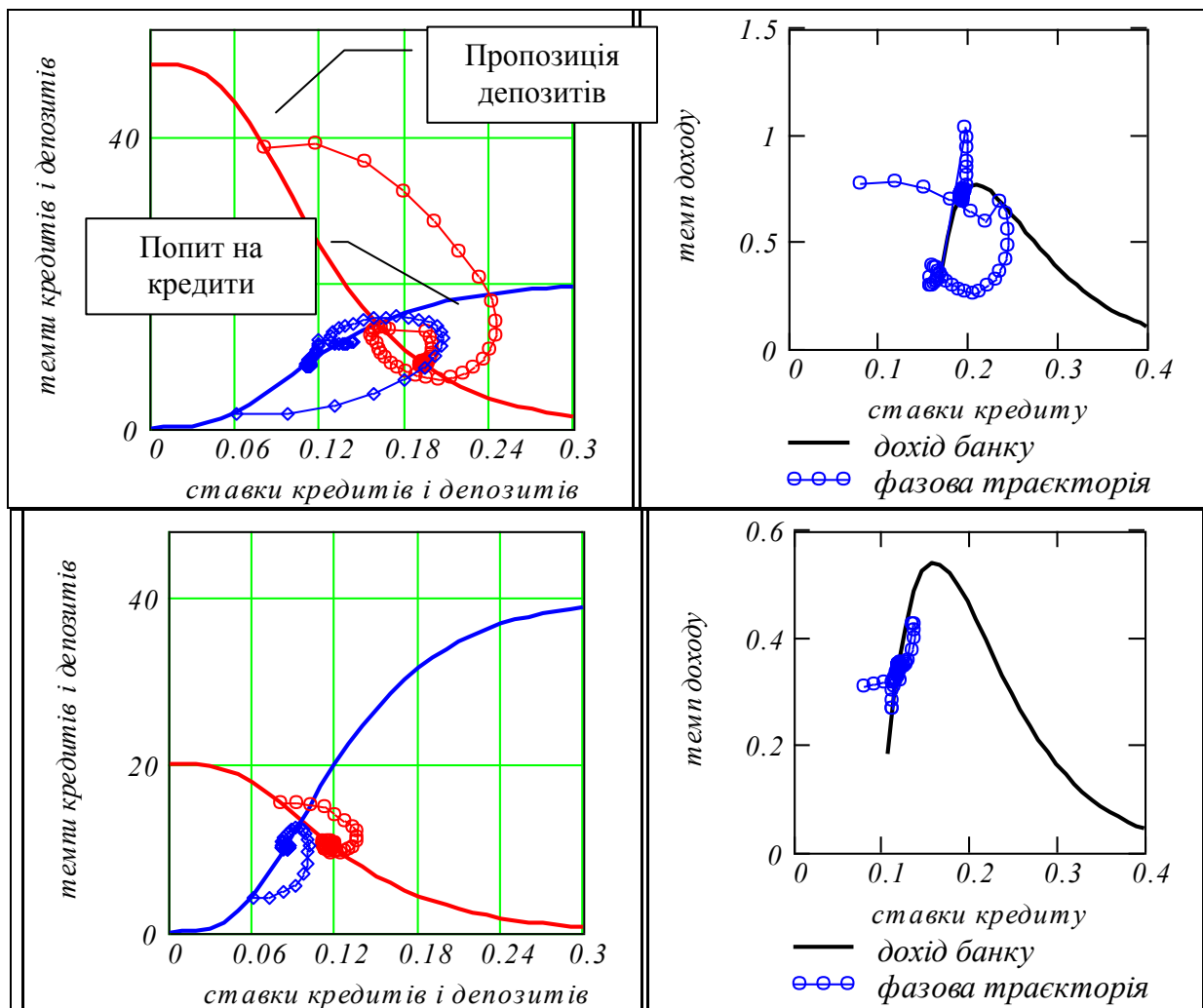


Рис. 4.22. Перехідні процеси на фазовій площині для випадків стартової ставки кредиту нижче і вище оптимальної

Бачимо, що: а) підсистема стабілізації балансу вхідних потоків кредитів і депозитів працює задовільно; б) підсистема максимізації доходу є *незадовільною*: не в усіх випадках приводить систему до максимуму, іноді – взагалі робить систему нестійкою. Причини незадовільності підсистеми максимізації: а) підсистеми управління діють спільно на об'єкти управління, що може приводити до небажаної інтерференції; б) занадто великі збурення і неоптимальні значення коефіцієнтів зворотних зв'язків:  $K1k$ ,  $K1d$ ,  $K2k$ ,  $K2d$ ; в) алгоритм пошуку максимуму має органічну ваду – коли дохід майже незмінний – припиняється оптимізація.

На рис. 4.23 подано перехідні процеси в часі для тих же умов, що і на рис. 4.22. Можемо бачити, що система управління приводить ставки кредиту і депозиту досить точно до оптимального стану (11%, 9%) з початкових станів "малі ставки" та "високі ставки".

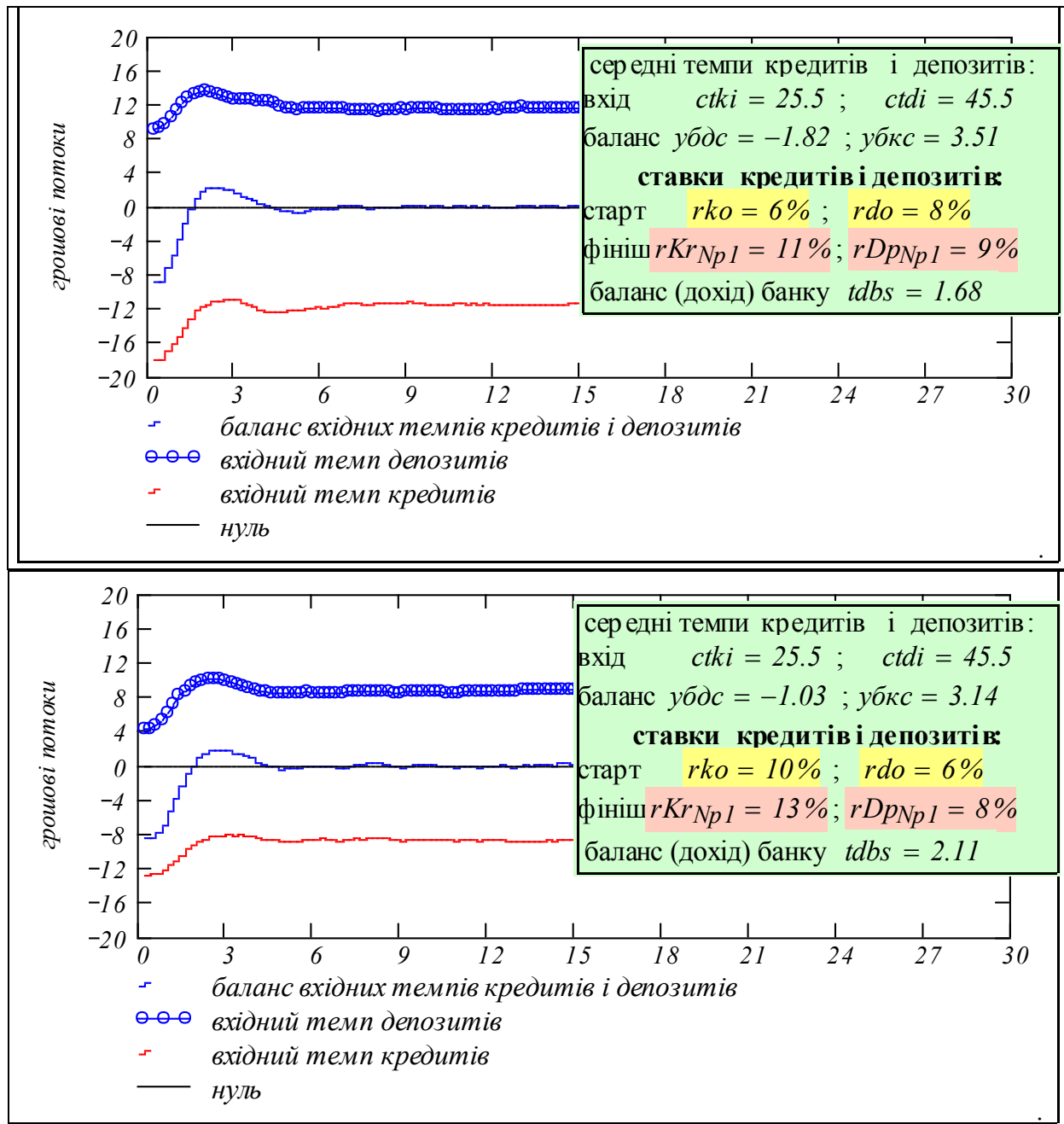


Рис. 4.23. Перехідні процеси для випадків стартової ставки кредиту нижче і вище оптимальної

## Напрямки розвитку моделі банківської системи

Початкова ціль розробки була в створенні максимально спрощеної математичної моделі банку як системи масового обслуговування. Розробка повинна була дати можливість студентам виконувати експерименти над віртуальною реальністю і дати початкове розуміння суті банківської справи: джерела нестійкості і виникнення "безпричинних" банкрутств. Робочу модель довелося будувати "з нуля". Навіть найпростіша з реалістичних моделей виявилася досить об'ємною. Однак ця модель виявилася базовою – вона дозволила відносно легко враховувати і досліджувати різноманітні аспекти банківської діяльності. В даному підрозділі як приклад розширення подано моделі управління банком. Крім цього були розроблені моделі темпів кредитів і депозитів з урахуванням інфляції, конкуренції та панічної поведінки клієнтів. Чисто технічною задачею є модифікація моделі для врахування диференціації ставок кредитів і депозитів а також диференціації клієнтів. Розроблені на базі класичних портфельних задач субмоделі для оптимізації структури активів і пасивів.

### Висновки

Побудована базова модель динаміки банківської системи. Модель придатна для розвитку в двох напрямках – дескриптивному, тобто, по можливості, точному відтворенню поведінки реальних банківських систем, і прескриптивному, тобто пошуку режимів функціонування і методів управління, що не спостерігалися в реальності, але є бажаними і можливими для реалізації. На конкретному прикладі показано виникнення та існування "ізолюваних від науки" (американський термін) областей, нашою мовою – "нічийних територій" – на стиках традиційних наук. Моделювання банківської системи не є класичною задачею фінансів, а задачею системного аналізу та теорії управління.

### Завдання для самостійного виконання

1. Дослідження впливу *ставки кредиту* при незмінних інших параметрах.
2. Дослідження впливу *ставки депозиту* при незмінних інших параметрах.
3. Дослідження впливу *середнього терміну кредиту* при незмінних інших параметрах.
4. Дослідження впливу *середнього терміну депозиту* при незмінних інших параметрах.
5. Дослідження впливу дисбалансу середніх темпів кредитів і депозитів.
6. Дослідження впливу амплітуди сезонних коливань при незмінних інших параметрах.
7. Дослідження впливу амплітуди випадкової складової при незмінних інших параметрах.
8. Дослідження впливу зсуву по фазі між сезонними коливаннями темпів кредитів і депозитів при незмінних інших параметрах.
9. Дослідження впливу одночасної пропорційної зміни ставок кредитів і депозитів.
10. Дослідження впливу дисбалансу середніх темпів кредитів і депозитів.
11. Оцінка частотних розподілів максимального та середнього відхилень темпу доходів банку ( простіше рішення: понатискати змінну  $qq$  і спостерігати реалізації змінних).
12. Розробіть модуль інформаційного управління: "інформуємо потенційних клієнтів – збільшуємо і стабілізуємо потоки кредитів і депозитів".

### Контрольні питання

1. Визначення депозиту, строкового депозиту, рахунку до запиту.
2. Зв'язок ставок депозитів і термінів депозитів.
3. Визначення кредиту, визначення ставки кредиту.
3. Назвіть джерела випадковості в депозитах.
4. Назвіть джерела випадковості в кредитах.
5. Які кредити є кращими для банку: коротко- чи довгострокові?
6. Абстрактна задача: банк бере депозити і надає кредити без процентів. Темпи кредитів і депозитів постійні і однакові. Яким буде усталений баланс банку: нуль, більше або менше нуля?



## 5 ДЕБЮТНІ ІДЕЇ ДЛЯ ПОБУДОВИ АДАПТИВНИХ САУ

### 5.1 Управління об'єктом класу "нестійкий маятник"

В цьому розділі розглядаємо занадто популярну серед прихильників інтелектуальних систем задачу – стабілізацію нестійкого об'єкта. Загальнозрозумілим прикладом такого об'єкта є нестійкий маятник. Замість постановки задачі подамо три картинки з трьох різних джерел. Перша – копія екрана, видача програми, що дозволяє вести імітаційні експерименти (рис. 5.1).

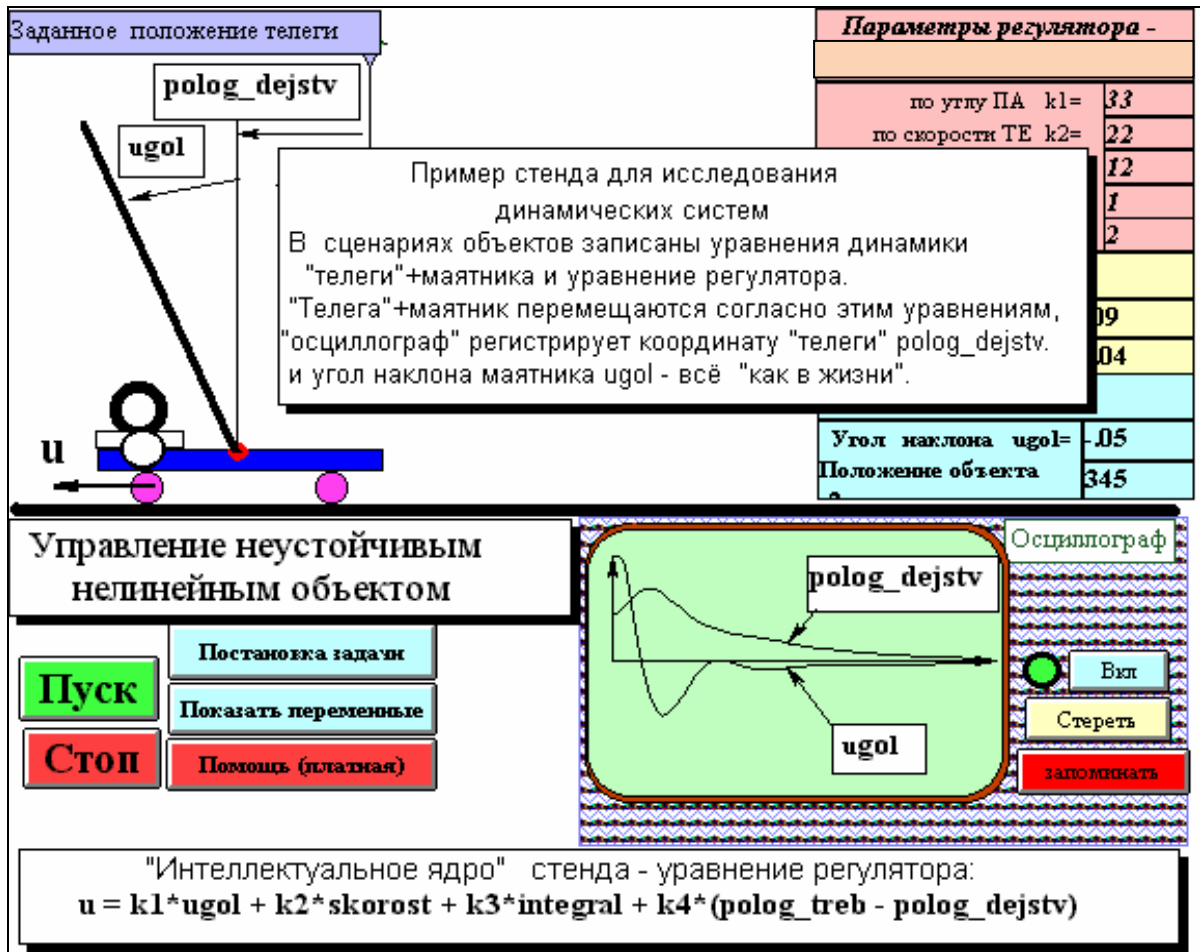


Рис. 5.1. Стенд „моделювання системи стабілізації нестійкого маятника”

Друга картинка – з електронної книги пакета Mathcad. Третя – заставка до матеріалу, що поданий далі (рис. 5.2).

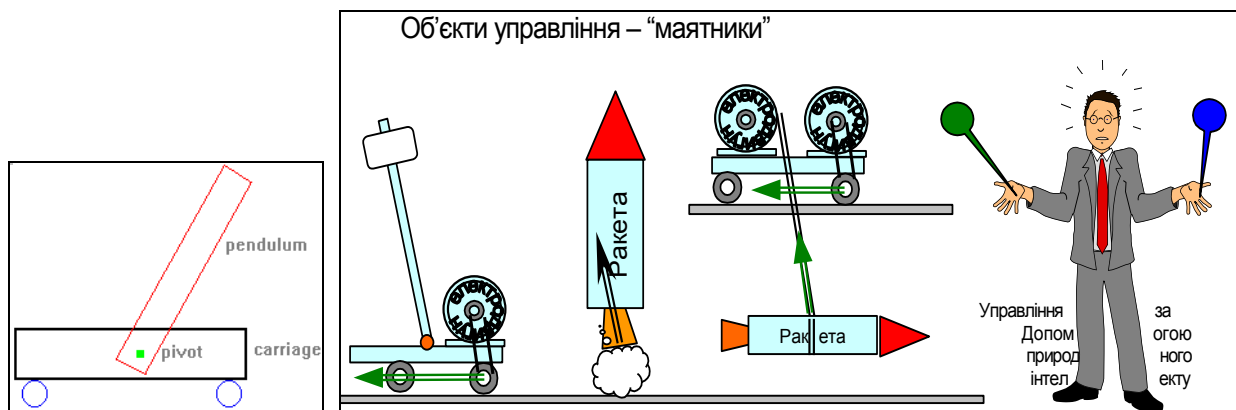


Рис. 5.2. Модель "нестійкий маятник" та її інтерпретації

Очевидно (буквально), що ця задача досить актуальна для теорії і практики.

Ціль даного розділу – на практиці підтвердити експертне правило: "не застосовуй адаптивні та інтелектуальні системи без крайньої необхідності". Між іншим термін "адаптивний" не має формального визначення. Часто вважається, що і просто системи із зворотними зв'язками є адаптивними.

Увага! Такий ж об'єкт – "нестійкий маятник" взято як тестовий в розділі 1.4 для адаптивної САУ з настроюванням власної частоти. В цьому розділі розглядається зовсім проста система.

### Постановка задачі

В останній час в досить стабільній, класичній області – теорії управління динамічними системами – з'являються відносні новації, наприклад, регулятори на базі нечіткої логіки (випускаються навіть відповідні мікросхеми), регулятори на базі генетичних алгоритмів оптимізації. Ці підходи є відносно новими тому, що роботи-першоджерела з'явилися 40–70 років тому. Нова хвиля інтересу до таких напрямів обумовлена двома причинами:

- виросло нове покоління, що не знає дефектів цих напрямків;
- з'явилися нові потужні і дешеві програмно-технічні засоби реалізації найскладніших і найвитратніших методів типу генетичних алгоритмів;
- з'явилися нові важливі для практики наукові результати, методи, узагальнення.

Особливість нових методів – неможливість аналітично довести стійкість, збіжність. Тому, для демонстрації працездатності методу, його застосовують на тестовій задачі – наочній, не дуже складній, але з характерними для певного класу задач особливостями – нестійкістю, нелінійністю, нестационарністю. Для "екзотичних" регуляторів на роль тестового об'єкта вибирається нестійкий маятник. Такий об'єкт – представник цілого класу, до якого відносяться:

- 1) стабілізація коливань вантажу на тросі під час перенесення і підйому-спуску (змінна довжина підвісу, змінні маси вантажів);
- 2) стабілізація ракети (нестійкість, змінна маса, змінна аеродинаміка);
- 3) стабілізація процесу сушіння – змінна маса і термодинамічні характеристики того, що сушать, – жому, молока, цементного клінкеру;
- 4) стабілізація валютно-фінансового ринку та інші подібні задачі сучасної економіки.

Як правило, для таких систем немає аналітичних результатів, єдиний інструмент відлагодження і оптимізації їх – стендове моделювання, обчислювальний експеримент.

Найбільш шокуючим для "нових" методів є той факт, що регулятори для таких об'єктів можна легко синтезувати (розрахувати) класичними методами теорії автоматичного управління. Досить тільки мати професійний погляд на методи аналізу і синтезу, *розділяти математичні моделі на моделі синтезу (розумно спрощені) і аналізу (розумно уточнені)*.

Останнє. Може виникнути чисто совкове, децю непристойне питання – який тип регулятора кращий, перспективніший? Ті, хто серйозно займаються генетичними алгоритмами, знають, що стійка популяція повинна мати досить широкий набір генотипів, невеликі кількості "мутантів" для непередбачуваного майбутнього.

Сучасний регулятор і повинен і може бути певним *колективом регуляторів різних типів* – це забезпечить стійкість та ефективність у непередбачуваних зовнішніх умовах.

У даному розділі ставиться задача розробки для описаного класу об'єктів програмної системи для синтезу регуляторів методом стандартних коефіцієнтів і розробка зручної системи моделювання керованої системи та пошукової оптимізації. Нагадуємо - для синтезу використовуємо лінеаризовану модель, для аналізу – нелінійну модель.

### Обґрунтування вибору методу синтезу і критерію оптимальності САУ

Для синтезу САУ використовуються такі методи:

- метод ЛАЧХ – зручний для безмашинного визначення параметрів регулятора, але приблизний і незручний для багатовимірних САУ;

- метод корневих годографів – теж приблизний, графоаналітичний, зручний для наближених розрахунків;
- група аналітичних методів конструювання регуляторів, до яких належить *метод стандартних коефіцієнтів (СК)*.

Метод СК зручний для формалізації і алгоритмізації, його суттєва перевага в тому, що процедуру синтезу можна повністю віддати ЦОМ, зробити онлайнною – коли параметри регулятора періодично обчислюються в процесі функціонування САУ.

Недоліки методу – чутливість до розмірності об'єкта. Для поставлених цілей перевага має цей метод. Головний крок у методі СК – вибір еталонного ПП, оптимального за якимось типовим критерієм. Для цих критеріїв вже раз і назавжди обчислені стандартні коефіцієнти характеристичних рівнянь.

Ну, так зробимо цей вибір – виберемо критерій "перерегулювання", або "викид" тому, що наша ціль – заспокоєння коливань. Нульове значення цьому критерію дає тільки одне розподілення коренів – кратне (усі корені дійсні і однакові), стандартні коефіцієнти для системи 3-го порядку будуть: 1, 3, 3, 1 (це біноміальні коефіцієнти). Додаткова перевага цього розподілу – *мінімум перенавантажень у ПП*.

### Отримання математичної моделі для синтезу

Беремо задану ММ – "візок з вантажем на зворотному маятнику" і:

- вводимо нову змінну  $dS = s_1 - s_2$ ;
- припускаємо амплітуду коливань малою, так що  $\sin(\theta) \approx \theta = dS/Ld$ ;
- змінну –  $Iugl$  – інтеграл від помилки (кута відхилення маятника) на етапі синтезу ігноруємо – вона потрібна тільки для астатизму. І із заданої дискретної ММ отримуємо спрощену неперервну ММ для синтезу регулятора:

Вводимо нові змінні:

$$\begin{pmatrix} v1_i \\ v2_i \\ s1_i \\ s2_i \\ Iugl_i \end{pmatrix} \Rightarrow \begin{pmatrix} V1_i \\ V2_i \\ dS_i \end{pmatrix};$$

Записуємо дискретну модель:

$$\begin{pmatrix} V1_{i+1} \\ V2_{i+1} \\ dS_{i+1} \end{pmatrix} = \begin{pmatrix} V1_i \\ V2_i \\ dS_i \end{pmatrix} + \begin{pmatrix} \frac{mm \cdot g}{Ld} \cdot dS \\ \frac{1}{Tp} \cdot U \\ V2_i \cdot -V1_i \end{pmatrix} \cdot T.$$

Записуємо також неперервну модель динаміки вантажу на тросі:

$$\begin{pmatrix} \frac{d}{dt} V1 \\ \frac{d}{dt} V2 \\ \frac{d}{dt} \Delta S \end{pmatrix} = \begin{pmatrix} 0 & 0 & \frac{mm \cdot g}{Ld} \\ 0 & 0 & 0 \\ -1 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} V1_1 \\ V2_1 \\ dS_1 \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{1}{Tp} \cdot U \\ 0 \end{pmatrix}.$$

### Розробка програми для розрахунку параметрів регулятора

Порядок синтезу:

- 1) отримуємо характеристичне рівняння замкненої САУ;
  - 2) вибираємо еталонний розподіл коренів і відповідні стандартні коефіцієнти;
  - 3) прирівнюємо потрібні значення коефіцієнтів виразам для коефіцієнтів характеристичного рівняння замкненої САУ – отримуємо систему алгебраїчних рівнянь;
  - 4) розв'язуємо ці рівняння – отримуємо шукані  $k_1, k_2, k_3$ .
- Далі подано програму, де все це робиться крок за кроком.

### Отримання характеристичних рівнянь

$$U = k_1 \cdot V_1 + k_2 \cdot V_2 + k_3 \cdot \Delta S; \quad K := (K_1 \quad K_2 \quad K_3);$$

$$A := \begin{pmatrix} 0 & 0 & a \\ 0 & 0 & 0 \\ -1 & 1 & 0 \end{pmatrix}; \quad B := \begin{pmatrix} 0 \\ b \\ 0 \end{pmatrix}; \quad X := \begin{pmatrix} V_1 \\ V_2 \\ \Delta S \end{pmatrix}; \quad U = K \cdot X; \quad I := \text{identity}(3).$$

$$\frac{d}{dt} X = A \cdot X + B \cdot U = A \cdot X + B \cdot K \cdot X = (A + B \cdot K) \cdot X.$$

$$A + B \cdot K = \begin{pmatrix} 0 & 0 & a \\ 0 & 0 & 0 \\ -1 & 1 & 0 \end{pmatrix} + \begin{pmatrix} 0 \\ b \\ 0 \end{pmatrix} \cdot (K_1 \quad K_2 \quad K_3) \rightarrow \begin{pmatrix} 0 & 0 & a \\ b \cdot K_1 & b \cdot K_2 & b \cdot K_3 \\ -1 & 1 & 0 \end{pmatrix};$$

$$I \cdot s - (A + B \cdot K) = I \cdot s - \begin{pmatrix} 0 & 0 & a \\ b \cdot K_1 & b \cdot K_2 & b \cdot K_3 \\ -1 & 1 & 0 \end{pmatrix} \rightarrow \begin{pmatrix} s & 0 & -a \\ -b \cdot K_1 & s - b \cdot K_2 & -b \cdot K_3 \\ 1 & -1 & s \end{pmatrix};$$

$$\left| \begin{pmatrix} s & 0 & -a \\ -b \cdot K_1 & s - b \cdot K_2 & -b \cdot K_3 \\ 1 & -1 & s \end{pmatrix} \right| \rightarrow s^3 - s^2 \cdot b \cdot K_2 - s \cdot b \cdot K_3 - b \cdot K_1 \cdot a + a \cdot s - a \cdot b \cdot K_2;$$

$$s^3 - s^2 \cdot b \cdot K_2 - s \cdot b \cdot K_3 - b \cdot K_1 \cdot a + a \cdot s - a \cdot b \cdot K_2 \Rightarrow$$

$$s^3 - s^2 \cdot b \cdot K_2 + (a - b \cdot K_3) \cdot s - a \cdot b \cdot (K_2 + K_1)$$

$$\text{Tr} := 3; \quad \text{Ld} := 5; \quad a := \frac{m \cdot g}{L_d}; \quad b := \frac{1}{T_p}.$$

Отримали вирази для коефіцієнтів характеристичного рівняння. Збираємо їх у вектор:

$$v := \begin{bmatrix} a \cdot b \cdot (K_2 + K_1) \\ (a - b \cdot K_3) \\ b \cdot K_2 \\ 1 \end{bmatrix}.$$

**Беремо із довідника** значення безрозмірних коефіцієнтів, безрозмірну тривалість  $\text{Tr}_b$  ПП і викид  $hm$  для цього процесу. Записуємо характеристичні рівняння – еталонне та отримане вище:

$$hm := 0; \quad \text{Tr}_b := 6.5; \quad \begin{pmatrix} a_0 b \\ a_1 b \\ a_2 b \end{pmatrix} := \begin{pmatrix} 3 \\ 3 \\ 1 \end{pmatrix}; \quad s^3 - s^2 \cdot a_2 b \cdot \omega_0 + a_1 b \cdot \omega_0^2 \cdot s - a_0 b \cdot \omega_0^3;$$

$$s^3 - s^2 \cdot b \cdot K_2 + (a - b \cdot K_3) \cdot s - a \cdot b \cdot (K_2 + K_1).$$

Прирівнюємо відповідні коефіцієнти рівнянь, поданих вище:

$$b \cdot K_2 = a_2 b \cdot \omega_0; \quad (a - b \cdot K_3) = a_1 b \cdot \omega_0^2; \quad a \cdot b \cdot (K_2 + K_1) = a_0 b \cdot \omega_0^3.$$

### Визначення параметрів регулятора вектора стану

Подаємо для контролю вхідні дані: тривалість (безрозмірна), викид для еталонної системи, записуємо задані значення цих параметрів, дивимось, чи задовольняє еталонний процес обмеження для викиду (задовольняє), і обчислюємо власну частоту  $\omega_0$ :

$$T_{pb} = 6.5; \quad h_m = 0; \quad T_{pz} := 10 \text{ c}; \quad h_{mz} := 0.0; \quad \omega_0 := \frac{T_{pb}}{T_{pz}}; \quad \omega_0 = 0.65.$$

Переходимо тепер до обчислення коефіцієнтів. Записуємо значення параметрів (довжина троса  $L_d$ , постійна часу приводу  $T_p$  та ін.) і початкові значення шуканих параметрів –  $K_1, K_2, K_3$  для розв'язання системи рівнянь.

$$L_d := 5; \quad T_p := 3; \quad m_m := 1; \quad a := \frac{m_m \cdot 9.81}{L_d}; \quad b := \frac{1}{T_p}; \quad a_0b = 3; \quad a_1b = 3; \quad a_2b = 1;$$

$$K_1 := 1; \quad K_2 := 1; \quad K_3 := 1.$$

Записуємо систему рівнянь, розв'язуємо її, виводимо результати

$$\begin{aligned} \text{Given} \quad & b \cdot K_2 = a_2b \cdot \omega_0; \\ & (a - b \cdot K_3) = a_1b \cdot \omega_0^2; \\ & a \cdot b \cdot (K_2 + K_1) = a_0b \cdot \omega_0^3. \end{aligned}$$

$$K_a := \text{Find}(K_1, K_2, K_3).$$

Вводимо інші позначення для знайдених параметрів регулятора, щоб не плуталися із значеннями, отриманими альтернативними методами.

$$\begin{pmatrix} kk1 \\ kk2 \\ kk3 \end{pmatrix} := K_a; \quad \begin{pmatrix} kk1 \\ kk2 \\ kk3 \end{pmatrix} = \begin{pmatrix} -0.69 \\ 1.95 \\ 2.084 \end{pmatrix}.$$

*Виконано першу частину процедури розробки системи (алгоритму) управління. Отримано результати розрахунку – параметри регулятора  $k_1, k_2, k_3$ ; модуль синтезу – працюєдатний. Цей модуль можна масштабувати – збільшувати порядок об'єкта. Для цього слід тільки переписати вхідні матриці і вектори.*

Усі подальші вирази переобчислюються і перебудовуються автоматично.

### **Розробка програми моделювання САУ з використанням нелінійної моделі динаміки**

Беремо за основу програми задані рівняння об'єкта. Записуємо програму моделювання в об'єктно-орієнтованій мові математичного пакета. Задаємо:

кількість кроків моделювання, період моделювання в сек., визначаємо крок обчислень.

$$\text{KolHag} := 200; \quad i := 1.. \text{KolHag}; \quad \text{PeriodMod} := 25; \quad T := \frac{\text{PeriodMod}}{\text{KolHag}}; \quad \text{Завада} := -.1;$$

$$\text{Stre}b := 0; \quad g := 9.81; \quad T = 0.125; \quad T_p := 3; \quad L_d := 5; \quad m_m := 1.$$

Записуємо поряд для порівняння синтезований аналітичними методами закон управління і розширений закон, параметри якого будуть послідовно уточнюватись:

- розробником на базі результатів моделювання, знань та інтуїції;
- системою самонастроювання ітеративно-пошуковими методами.

$$U_{\text{аналітичне}} = k_1 \cdot V_1 + k_2 \cdot V_2 + k_3 \cdot \Delta S;$$

$$\text{упр}(v_1, v_2, \text{ugl}, \text{Inugl}, s_2) := k_1 \cdot v_1 + k_2 \cdot v_2 + k_3 \cdot \text{ugl} + k_4 \cdot \text{Inugl} + k_5 \cdot (\text{Stre}b - s_2);$$

- вводимо нову змінну  $dS = s_1 - s_2$ ;

- припускаємо амплітуду коливань малою, так що  $\sin(\theta) \approx \theta = dS/L_d$ ;  $\text{ugl} = \frac{dS}{L_d}$ .

Вводимо тестову функцію – зміну довжини підвісу вантажу з часом – нестационарність. В повній системі моделювання ці дані будуть передаватись від модуля "підйом-спуск".

Нестационарність:  $ld_i := ld \cdot (1 - 0.004 \cdot i)$ .

$$ugol \geq \frac{s2 - s1}{Ld}; \quad ugl(S2, S1, LD) := \text{asin}\left(\frac{S2 - S1}{LD}\right); \quad oL := \frac{1}{Ld}$$

Задаємо початковий стан об'єкта

$$\begin{pmatrix} v1_1 \\ v2_1 \\ s1_1 \\ s2_1 \\ Iugl_1 \end{pmatrix} := \begin{pmatrix} 0 \\ 0 \\ -10 \\ -10 \\ 0 \end{pmatrix}; \quad \begin{pmatrix} \text{"v1-швидкість візка"} \\ \text{"v2- швидкість вантажа"} \\ \text{"s1- координата візка"} \\ \text{"s2- координата вантажа;" } \\ \text{"Iugol-нтеграл від кута нахилу"} \end{pmatrix}; \quad T = 0.125; \quad Tr = 3.$$

Еквівалентна дискретна математична модель для аналізу

$$\begin{pmatrix} \text{"швидкість візка---->"} \\ \text{"швидкість вантажа->"} \\ \text{"координата візка---->"} \\ \text{"координата вантажа->"} \\ \text{"нтеграл від кута---->"} \end{pmatrix} \begin{pmatrix} v1_{i+1} \\ v2_{i+1} \\ s1_{i+1} \\ s2_{i+1} \\ Iugl_{i+1} \end{pmatrix} := \begin{pmatrix} v1_i \\ v2_i \\ s1_i \\ s2_i \\ Iugl_i \end{pmatrix} + \begin{bmatrix} [mm \cdot g \cdot (s2_i - s1_i) \cdot oL + \text{Завада}] \cdot T \\ \frac{T}{Tr} \cdot \text{upr}(v1_i, v2_i, ugl(s2_i, s1_i, Ld), Iugl_i, s2_i) \\ v1_i \cdot T \\ v2_i \cdot T \\ ugl(s2_i, s1_i, Ld) \cdot T \end{bmatrix}$$

Задаємо значення  $g=9.81$ ;  $Tr = 4$  – параметр привода;  $ld = 5$  – довжина троса;

$$ugl(S2, S1, LD) := \text{asin}\left(\frac{S2 - S1}{LD}\right); \quad \text{upr}(v1, v2, ugl, Iugol, s2).$$

Контроль даних:  $-kk1 = 0.69$ ;  $-kk2 = -1.95$ ;  $-kk3 = -2.08$ .

Зона вхідних параметрів  $k1 \equiv 3$ ;  $k2 \equiv -7$ ;  $k3 \equiv -10$ ;  $k4 \equiv .1$ ;  $k5 \equiv 1.5$ .

Задаємо значення параметрів системи і будуємо графіки перехідних процесів.

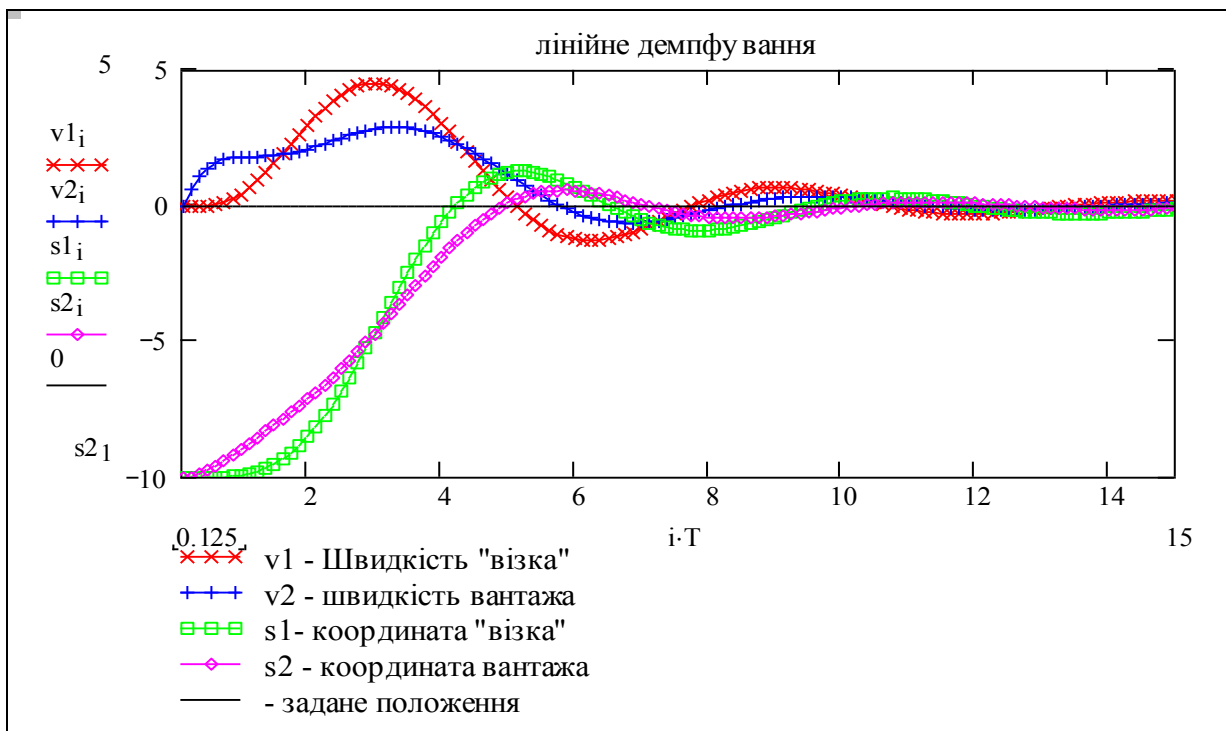


Рис. 5.3. Перехідні процеси в системі стабілізації нестійкого маятника

Робимо блок для візуалізації динаміки (математичний пакет – не найкраще для цього програмне середовище)  $k := 1..45$ ;  $l := 1..100$ ;

$$X_{1+3\cdot(k-1)} := s_{12\cdot k}; \quad Y_{1+3\cdot(k-1)} := l_{d_{1+3\cdot(k-1)}}; \quad X_{3\cdot(k-1)+2} := s_{22\cdot k}; \quad Y_{3\cdot(k-1)+2} := 0;$$

$$X_{3\cdot k} := s_{22\cdot k}; \quad Y_{3\cdot k} := -1.$$

Серія послідовних положень маятника – відображення анімації в паперовий документ.

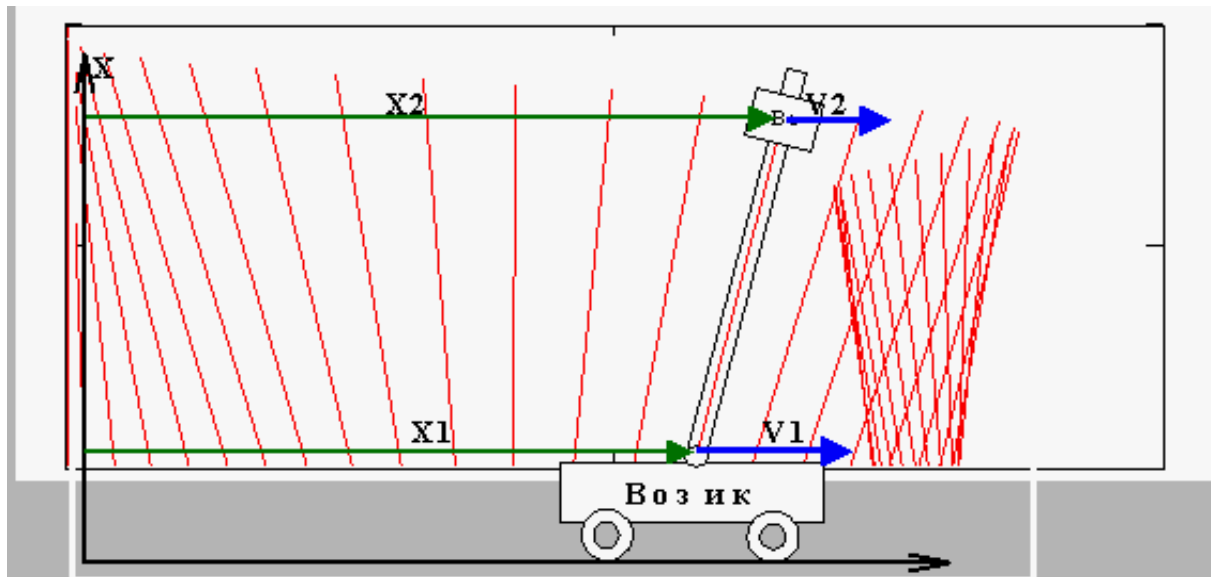


Рис. 5.4. Послідовність положень маятника при переході до заданого

### Застосування нелінійного демпфування

Покажемо приклад використання цього модуля для пошуку проектних рішень. Професіонал, наприклад, змінить лінійний закон демпфування на такий, де величина демпфування залежить і від помилки положення.

$$i := 1..KolНа\grave{c}; \quad k_1 := -29; \quad k_2 := -0.8; \quad k_3 := -1; \quad k_4 := 9; \quad streb := 0.$$

$$uprv(ugl, vvoza, l_{ugl}, s_{2voza}) := k_1 \cdot ugl + \frac{k_2}{(.01 + |ugl|)} \cdot vvoza + k_3 \cdot l_{ugl} + k_4 \cdot (streb - s_{2voza}).$$

$$\text{Нестационарність: } l_{d_i} := L_d \cdot (1 - 0.004 \cdot i);$$

$$\begin{pmatrix} v_{11} \\ v_{21} \\ s_{11} \\ s_{21} \\ l_{ugol1} \end{pmatrix} := \begin{pmatrix} 0 \\ 0 \\ -3 \\ -3 \\ 0 \end{pmatrix}; \quad T = 0.125; \quad u_{gol} \geq \frac{s_2 - s_1}{L_d}; \quad u_{gol}(s_2, s_1, L_d) := \arcsin\left(\frac{s_2 - s_1}{L_d}\right).$$

Запишемо рівняння динаміки системи

$$\begin{pmatrix} v_{1i+1} \\ v_{2i+1} \\ s_{1i+1} \\ s_{2i+1} \\ l_{ugol_{i+1}} \end{pmatrix} := \begin{pmatrix} v_{1i} \\ v_{2i} \\ s_{1i} \\ s_{2i} \\ l_{ugol_i} \end{pmatrix} + \begin{bmatrix} (mm \cdot g \cdot \sin(ugol(s_{2i}, s_{1i}, l_{d_i}))) \cdot T \\ \frac{T}{T_p} \cdot uprv(ugol(s_{2i}, s_{1i}, l_{d_i}), v_{2i}, l_{ugol_i}, s_{2i}) \\ v_{1i} \cdot T \\ v_{2i} \cdot T \\ u_{gol}(s_{2i}, s_{1i}, l_{d_i}) \end{bmatrix}.$$

Вводимо змінну:  $\text{Позиція\_Возу} := 1$ . Будуємо графіки перехідних процесів (рис. 5.5).

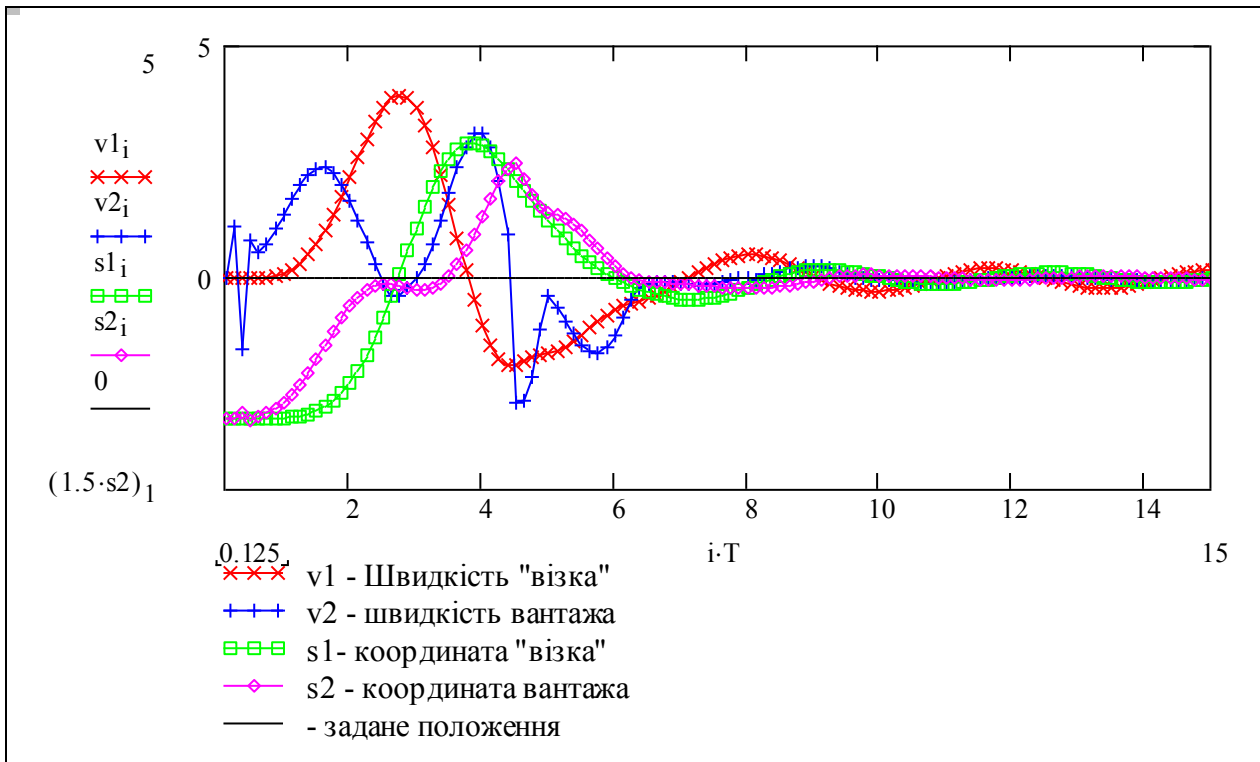


Рис. 5.5. Перехідні процеси в системі стабілізації нестійкого маятника

Бачимо, що таке проектне рішення легко було перевірити і воно виявилось придатним для подальшого випробування.

### Настроювання регулятора при зміні довжини підвісу

Оцінимо можливість заспокоєння коливань альтернативними діями управління. Для даної задачі можна отримати аналітичні вирази для коефіцієнтів регулятора в функції параметрів об'єкта. Розв'язуємо рівняння у символічній формі. Тепер не важко побудувати систему аналітичного самонастроювання – вона повинна змінювати коефіцієнти регулятора за цими виразами

Given

$$be \cdot Ke2 = a2be \cdot \omega0; \quad (ae - be \cdot Ke3) = a1be \cdot \omega0^2; \quad ae \cdot be \cdot (Ke2 + Ke1) = a0be \cdot \omega0^3;$$

$$\text{Find}(Ke1, Ke2, Ke3) \rightarrow \begin{pmatrix} \omega0 \cdot \frac{-ae \cdot a2be + a0be \cdot \omega0^2}{ae \cdot be} \\ a2be \cdot \frac{\omega0}{be} \\ \frac{ae - a1be \cdot \omega0^2}{be} \end{pmatrix}.$$

### Завдання для самостійного виконання

1. Проведіть дослідження граничних припустимих відхилень об'єкта управління від стану рівноваги.
2. Проведіть дослідження для знаходження оптимального лінійного і нелінійного демпфування.



## 5.2 Адаптивний регулятор на базі нечіткої логіки

### Вступ

Що таке регулятор? З фундаментально-філософської точки зору це просто відношення, перетворення, обернене до перетворення об'єкта управління:

$$x = \text{Obj}(u); u = \text{Reg}(x); x = \text{Obj}(x, \text{Reg}(x)); x = \text{Obj}(\text{Obj}(x)^{-1}); \text{ звідки маємо: } x = x.$$

Принцип оберненої моделі можна реалізувати методами нечіткої логіки – побудувати регулятор, що буде поступово настроюватись на динаміку певного невизначеного, нелінійного, нестационарного об'єкта управління.

Справжня нечітка логіка досить незвична для осмислення. Тому почнемо з найелементарнішого прикладу – *регулятора для статичного об'єкта*. Між іншим цей приклад має певне практичне значення для побудови *регулятора усталених станів для динамічного об'єкта*.

Пройдемо ж крок за кроком увесь шлях від концепції до розробки програми реалізації адаптивного регулятора на базі нечіткої логіки. І зупинимось на цьому кроці – щоб щось залишити для самостійної роботи.

### Постановка задачі

Викладемо ідею нечіткого адаптивного регулятора, як вона подана в книзі М. Пешеля [34]. Задано об'єкт управління, що описується в ідеальному випадку чіткою залежністю:  $y = f(x)$ . Однак ця залежність з різних причин нечітка (неповністю визначена, розмита). Поставимо за мету (а не ціль, тому що поки не знаємо, як це зробити) розробку системи, що використовує інформацію про дійсну реакцію об'єкта для настроювання регулятора.

### Конструювання розподілу нечіткості (= функції належності) регулятора

Коли читаєш наукову літературу з нечіткої логіки, то матеріал здається не тільки нечітким, але й незрозумілим і непотрібним.

Для того, щоб отримати чітку уяву про нечітку логіку, зробимо робочу модель нечіткої функції однієї змінної і поекспериментуємо з нею. Задаємо параметри

$$\text{ORIGIN} := 1; A := .4; \alpha := .5 \text{ і чітку функцію: } f(x) := A \cdot x^\alpha.$$

"Розмиваємо" (фаззифікуємо!) цю функцію – задаємо певний двовимірний розподіл нечіткості. Рнчо =  $fz(x, y)$ , де  $y = f(x)$ . Робимо це так: задаємо параметр розмитості (це дисперсія)  $\sigma(x) := 0.08 \cdot (1 + 2 \cdot x)$ ; в цілях економії інтелекту для розподілу нечіткості використовуємо вбудовану функцію нормального розподілу:

$$fz(x, y) := \text{dnorm}(x - y, y, \sigma(x)).$$

Формат цієї функції `dnorm(вхідна_змінна, математичне_очікування, розкид)`. Задаємо розмір сітки:

$$D_i := 40 \text{ та індекси: } i := 1..D \quad j := 1..D \quad h_a := 1 \div D.$$

Робимо програму для розрахунку масиву значень і будуємо тривимірні графіки функції розподілу нечіткості. Дивимось на графіки на рис. 5.6. Бачимо нечітку залежність. На одному з графіків подано перетини, що відповідають розподілам нечіткості для відповідних значень вхідної змінної, на другому графіку – лінії рівня. Ми зробили величину нечіткості (дисперсію) пропорційною величині вхідної змінної. Це іноді відповідає реальності.



### Конструювання алгоритму адаптації

Для реалізації і моделювання нечіткого регулятора потрібно вибрати і реалізувати такі функціональні модулі:

1. **Функція належності одного експерименту (проби):** задано певний випадковий вхідний сигнал, подано на модель об'єкта і отримано певний вихідний сигнал, що є результатом дії певного чіткого механізму і різних видів невизначеності (збурення, помилки вимірювання, квантові та ін. властивості об'єкта). Можливо, що і вхідний сигнал реалізується з певними похибками. Треба задати відповідний двовимірний розподіл імовірностей.

2. **Імітація події:** за допомогою певного генератора відповідно до розподілу нечіткості формується реалізація (хр, ур).

3. **Комбінування функції належності регулятора та проби.** Альтернативи комбінування розглянуто далі.

4. **Конструювання функції належності для об'єкта управління.** Задаємо функцію розподілу нечіткості регулятора:

$$dXv(x, u) := \text{dnorm}\left[x - 2, .1u^2, \left[0.7 + 0.04 \cdot (u - 1)^2\right]\right], \text{ дискретизуємо цю функцію}$$

$$x := 0, .5..20; u := 0, .5..20; X_i := i - 1; U_j := j - 1; mO_{i, j} := dXv(X_i, U_j).$$

5. **Конструювання функції належності для вимірювання виходу об'єкта управління.** Задаємо функцію розподілу нечіткості проби (подали вхід – отримали вихід)  $\sigma_p := 1$ ;

$$dXi(x, u, xi, ui) := \text{dnorm}\left[x - xi, xi, \left[\sigma_p + 0.001 \cdot (x)^2\right]\right] \cdot \text{dnorm}\left[u - ui, ui, \left[\sigma_p + 0.0005 \cdot (u)^2\right]\right];$$

$$Kys := 3; X_i := i - 1; U_j := j - 1.$$

Задаємо функції розподілу для декількох проб:  $ri_{1, j} := Kys \cdot dXi(X_i, U_j, 2, 5)$ ;

$$ri_{2, j} := Kys \cdot dXi(X_i, U_j, 7, 9); \quad ri_{3, j} := Kys \cdot dXi(X_i, U_j, 11, 8);$$

$$ri_{4, j} := Kys \cdot dXi(X_i, U_j, 13, 9); \quad ri_{5, j} := Kys \cdot dXi(X_i, U_j, 16, 12).$$

Записуємо альтернативи згортки розподілів нечіткості:

$$\text{добуток: } QQ := \left[ (ri_1 + 1) \cdot (ri_3 + 1) \cdot (ri_4 + 1) \cdot (ri_5 + 1) \cdot (By + 1) \right];$$

$$\text{сума: } PP := (ri_1 + ri_3 + ri_4 + ri_5 + 0.3By).$$

На рис. 5.8 подано настроювання нечіткого регулятора.

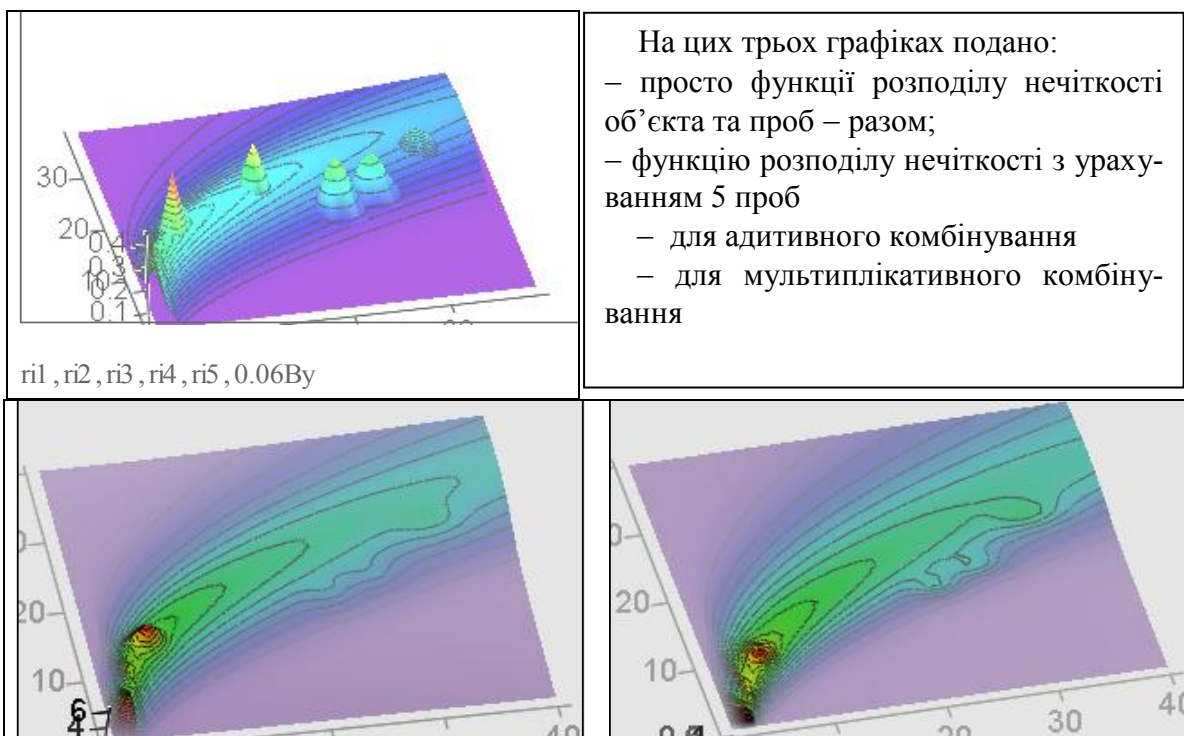


Рис. 5.8. Настроювання нечіткого регулятора

### Конструювання програми функціонування системи з нечітким регулятором

З розглянутих вище компонентів збираємо програму (рис. 5.9). Достатньо нераціональну і недосконалу і помірно працездатну. Це, певним чином, перша проба.

Число точок дискретизації функції розподілу нечіткості:  $M := 25$ ;  $i:=1..M$ ;  $j:=1..M$ .

```

Rr(N) :=
  xt ← runif(N, 1, 14)
  for i ∈ 1..M
    for j ∈ 1..M
      mRi,j ← dnorm(j - 5, i, 3)
  for k ∈ 1..N
    "Регулятор видає у правління:"
    ut ← 0.05 rnd(k) + rnd(1)
    "Реакція об'єкта - випадкова з певним розподілом"
    xv ← norm[ 1, 0.15 · ut2, [ 0.2 + 0.02 · (ut - 1)1 ] ]1
    "В цьому подвійному циклі настроюється розподіл"
    "нечіткості регулятора"
    for i ∈ 1..M
      for j ∈ 1..M
        ri,j ← dXi(i, j, xv, ut)
        mRi,j ← [ S · mRi,j + (1 - S) · ri,j ]
  mR
  
```

Рис. 5.9. Програма моделювання нечіткого регулятора

Головний параметр для настроювання нечіткої характеристики регулятора  $S \equiv 0.96$

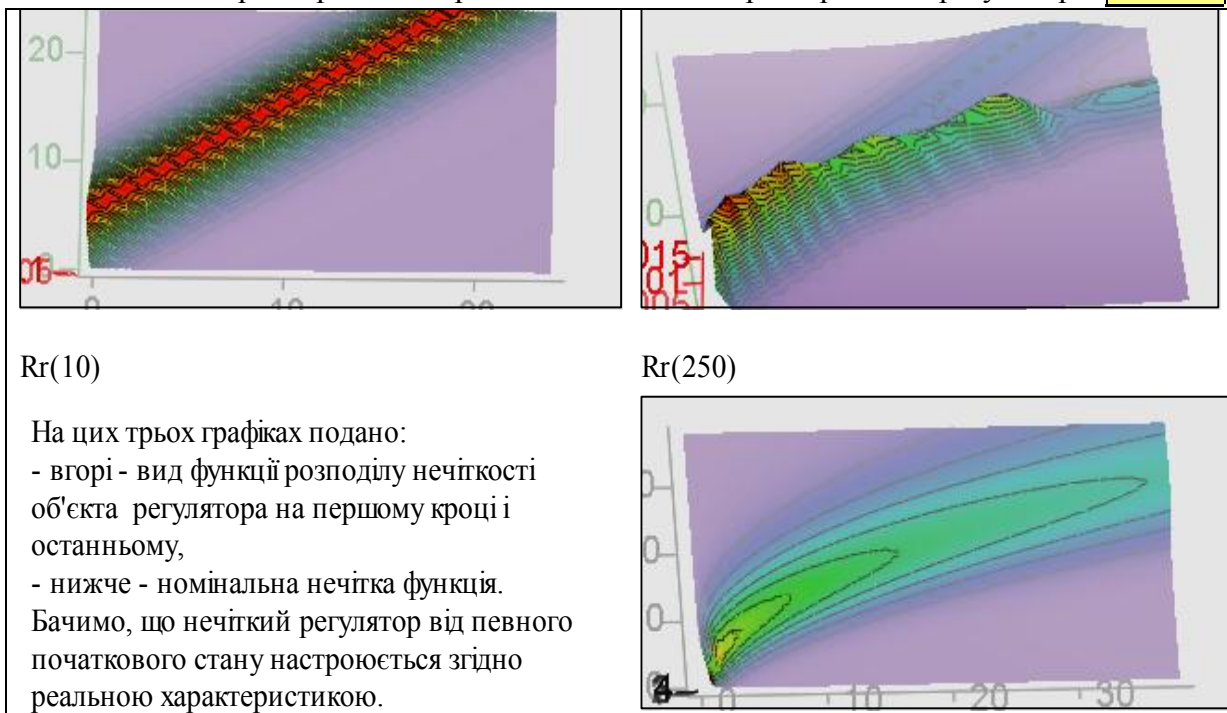


Рис. 5.10. Розподіли нечіткості: стартовий, кінцевий, ідеальний („точний”)

**Висновок.** Природно зробити нечіткий ПІД-регулятор для об'єкта другого порядку і визначити, чи не ускладнюється побудова і настроювання такого регулятора.

Отримано середовище для випробування численних версій нечіткого регулятора.

## 5.3 Адаптивна система на базі штучної нейронної мережі

### Нейронна мережа Кохонена

Мережа Кохонена (Kohonen's Neural Network), інша назва Kohonen's Self Organizing Feature Map (SOFM) ("така, що організується, карта ознак Кохонена").

Ця мережа була запропонована Кохоненом в 1984 році. В цьому напрямку виконано багато робіт з модифікації базової моделі і побудови математичної теорії. Таке широке розповсюдження і велика кількість робіт можуть бути ознакою слабкої практичної корисності, тому що продуктивні напрямки зникають – стають фірмовими секретами.

### Модель мережі Кохонена

В напрямку штучних нейронних мереж існує традиція виведення їх з дійсних чи гіпотетичних механізмів роботи мозку, хоч природно вивести її з моделей кінцевих автоматів і багатовимірних нелінійних імпульсних систем зі зворотними зв'язками – тобто моделей Я. Ципкіна [40].

Алгоритм Кохонена вважається подібним процесам, що відбуваються у мозку при самонавчанні. Мережа Кохонена призначається для розділення векторів з певної множини на підгрупи – кластери. Мережа складається з  $M$  нейронів. При навчанні і роботі мережі на всі  $N$  входів кожного нейрона подається один і той же сигнал –  $N$ -вимірний вектор  $X$ . В процесі навчання змінюються "синаптичні ваги" нейронів. В книзі В. Круглова і В. Борисова вкрай неякісно перекладено матеріали – тести і рисунки з першоджерел – це, певним чином, вторинні недоліки цього напрямку.

Вхідні сигнали – вектори дійсних чисел – послідовно подаються на входи мережі і виконується самонавчання. Після певного числа прикладів навчання закінчується і синаптичні ваги вже визначають кластери. Крім того, ваги організуються так, що топологічно близькі нейрони східно реагують на східні сигнали (вектори).

Для реалізації алгоритму необхідно визначити *міру близькості нейронів*. Розміри зони близькості повинні зменшуватись...

Задаємо число нейронів  $Ne := 12$  та число ознак  $Pr := 3$ , період моделювання  $Tm := 100$ ;  $t := 1..Tm$ ;  $ORIGIN := 1$ .

### Алгоритм Кохонена

**Крок 1.** Ініціалізація мережі. Коефіцієнтам мережі числом  $Ne \cdot Pr = 36$  присвоюються малі випадкові числа.  $D(0)_j$  – початкова зона сусідства. Задаємо індекси ознак,  $i := 1..Pr$  та індекси нейронів  $j := 1..Ne$ ;  $t := 1..Tm$ . Задаємо початкові значення ваги:

$$w1 := \begin{pmatrix} \text{rnd}(0.1) & \text{rnd}(0.1) & \text{rnd}(0.1) \\ \text{rnd}(0.1) & \text{rnd}(0.1) & \text{rnd}(0.1) \\ \text{rnd}(0.1) & \text{rnd}(0.1) & \text{rnd}(0.1) \\ \text{rnd}(0.1) & \text{rnd}(0.1) & \text{rnd}(0.1) \end{pmatrix};$$

$$ww_{i,j} := \text{rnd}(0.1); w1 := ww; (w1)_{1,1} = 0.001; (w1)_{3,2} = 0.074.$$

**Крок 2.** Подача до мережі нового вхідного сигналу.

**Крок 3.** Обчислення відстані  $d_j$  – від вхідного сигналу до кожного з нейронів за формулою:

$$d_j := \sum_{i=1}^N [x_{t,j} - (w_t)_{i,j}]^2,$$

де  $x_{t,j}$  –  $i$ -ий елемент вхідного сигналу в момент  $t$ ;  $w(t)_{i,j}$  – вага зв'язку  $i$ -го компонента вхідного вектора з  $j$ -м нейроном.

*Зауваження:*  $x(t)_i$  – "функція часу, що є  $i$ -им компонентом вектор-функції часу  $x(t)$ ";  $x_{t,j}$  – формально це: " $i$ -ий рядок матриці  $x$ "; змістовно: " $i$ -ий компонент дискретизованої вектор-функції часу".

Щоб ефективно працювати в середовищі математичних пакетів треба сприймати кожен математичний об'єкт як, єдиний в двох іпостаях – неперервній і дискретній.

**Крок 4.** Вибір  $kk$ -нейрона, для якого відстань  $d_{kk}$  є мінімальною.

**Крок 5.** Настроювання ваг для нейрона  $kk$  та всіх сусідніх нейронів з зони сусідства  $D(t)_j$  згідно з рівнянням:

$$(w_{t+1})_{i,j} = (w_t)_{i,j} + \eta_t \cdot [x_{t,j} - (w_t)_{i,j}],$$

де  $\eta$  – крок навчання ( $0 \leq \eta_t \leq 1$ ), що зменшується до нуля з часом.

Введемо функцію:  $norm(xx, nmi, dis) := \frac{dnorm(xx, nmi, dis)}{dnorm(nmi, nmi, dis)}$ .

Зробимо генератор тестових векторів. Закладемо в нього 4 класи об'єктів. Це тільки макет для створення узагальненої програми (рис. 5.11).

```

xx(t) := kyd ← md(1)
        xto ← | 1 if kyd ≤ 0.25
              | 2 if 0.25 < kyd ≤ 0.5
              | 3 if 0.5 < kyd ≤ 0.75
              | 4 if kyd > 1.0000
        vvz ← Shum
        zv ← 1 - vvz
        ay ← [ 4 · (zv + rnd(vvz))  3 · (zv + rnd(vvz))  2 · (zv + rnd(vvz))  1 · (zv + rnd(vvz)) ]
        qq ← [ 1 · (zv + rnd(vvz))  4 · (zv + rnd(vvz))  3 · (zv + rnd(vvz))  2 · (zv + rnd(vvz)) ]
        zz ← [ 2 · (zv + rnd(vvz))  1 · (zv + rnd(vvz))  4 · (zv + rnd(vvz))  3 · (zv + rnd(vvz)) ]
        fo ← [ 3 · (zv + rnd(vvz))  2 · (zv + rnd(vvz))  1 · (zv + rnd(vvz))  4 · (zv + rnd(vvz)) ]
        qq1 ← ay · (kyd ≤ 0.25) + qq · (0.25 < kyd ≤ 0.50)
        qq2 ← zz · (0.50 < kyd ≤ 0.75) + fo · (kyd > 0.75) · 1
        vex ← qq1 + qq2
        (
          ((vex)T
           xto
        )
    
```

Рис. 5.11. Модуль генерування тестових векторів (образів)

Розпаковуємо вихід програми – генератора тестових послідовностей об'єктів класифікації  $x_t := xx(t)_1$ ;  $MK^{(t)} := xx(t)_1$ ;  $K_{yt} := xx(t)_2$ ;  $X1_t := MK_{1,t}$ ;  $Y1_t := MK_{2,t}$ ;  $Z1_t := MK_{3,t}$ . На рис. 5.12 подано випробування модуля генерування тестових векторів.

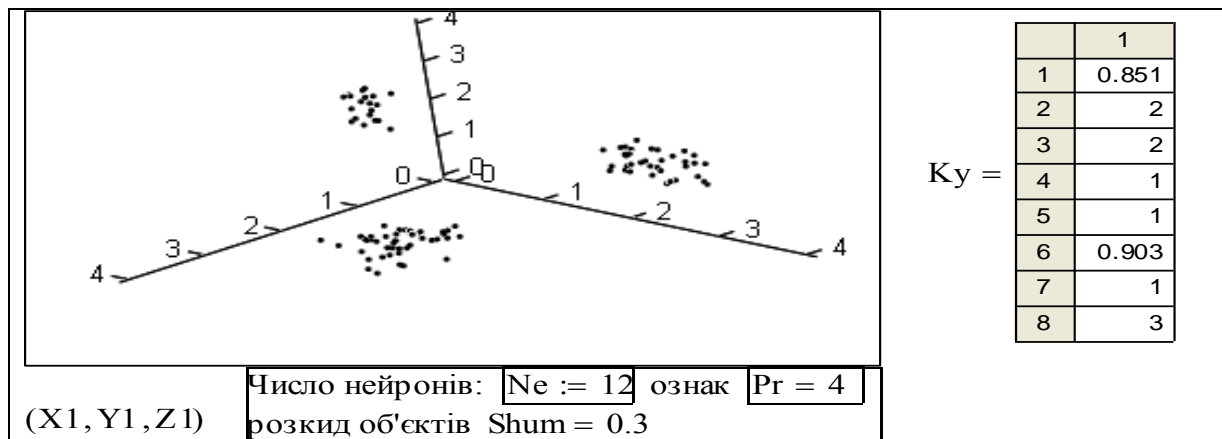


Рис. 5.12. Випробування модуля генерування тестових векторів

**Крок 6.** Повернення до кроку 2. На рис. 5.13 подано модуль "нейронна мережа Кохонена".

```

уНу:= w1 ← ww
      η1 ← startη
      dis ← startdis
      for t ∈ 1..Tm
        xt ← xx(t)1
        qd_t ← xx(t)2
        dmi ← 10000
        for i ∈ 1..NE
          di ← ∑_{j=1}^{Pr} [(xt)_j - (wt)_{j,i}]^2
          if di < dmi
            dmi ← di
            nmi ← i
          "qq"
        for i ∈ 1..NE
          for j ∈ 1..Pr
            QQ_{j,i} ← (wt)_{j,i} + η_t · [(xt)_j - (wt)_{j,i}] · rozm(i, nmi, dis)
            "бачимо, що ваги коректуються згідно з помилкою"
          wt_{t+1} ← QQ
          η_{t+1} ← η_t · dηη
          dis ← dis · dgn + 0.001
          nomi ← nmi
      ( w
        nomi
        qd )
  
```

Рис. 5.13. Модуль "нейронна мережа Кохонена"

Розпаковуємо вихід програми:  $Ves := y_{Hy}$ ;  $QD := y_{Hy}$ ;  $Nmi := y_{Hy}$ ;  $n8N := 1$ ;  $Shum = 0.3$ .

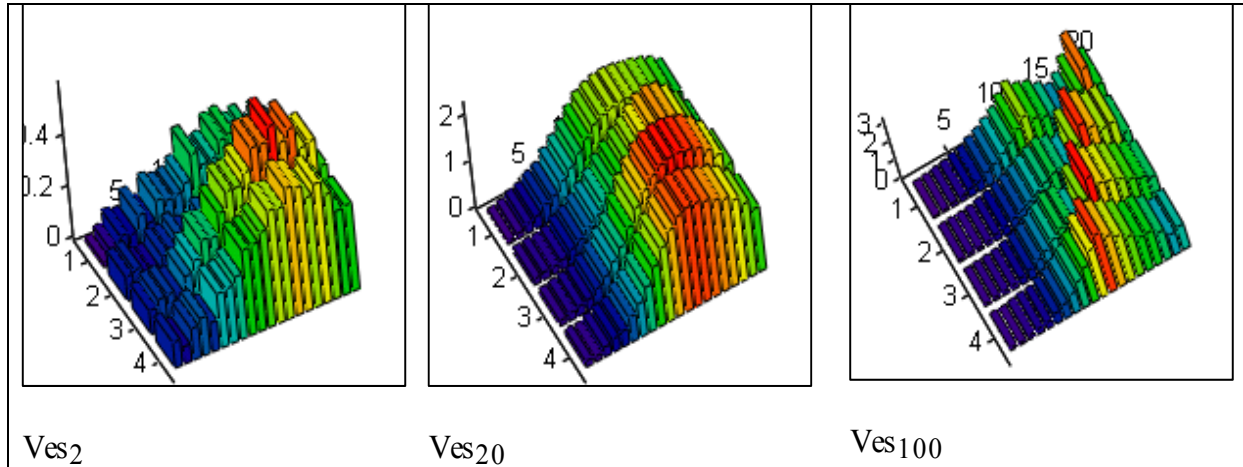


Рис. 5.14. Еволюція матриці Кохонена – три моменти часу

Параметри процесу навчання: розкид елементів кластерів  $Shum \equiv 0.3$ ; стартова границя сусідніх нейронів:  $startdis \equiv 6$ ; звуження границь нейрона  $dgn \equiv 0.997$ ; стартова швидкість навчання:  $start\eta \equiv 0.14$ ; зменшення швидкості навчання:  $d\eta \equiv 0.99$ .

Число нейронів:  $NE \equiv 20$ ; число ознак (входів нейронів)  $Pr = 4$ .

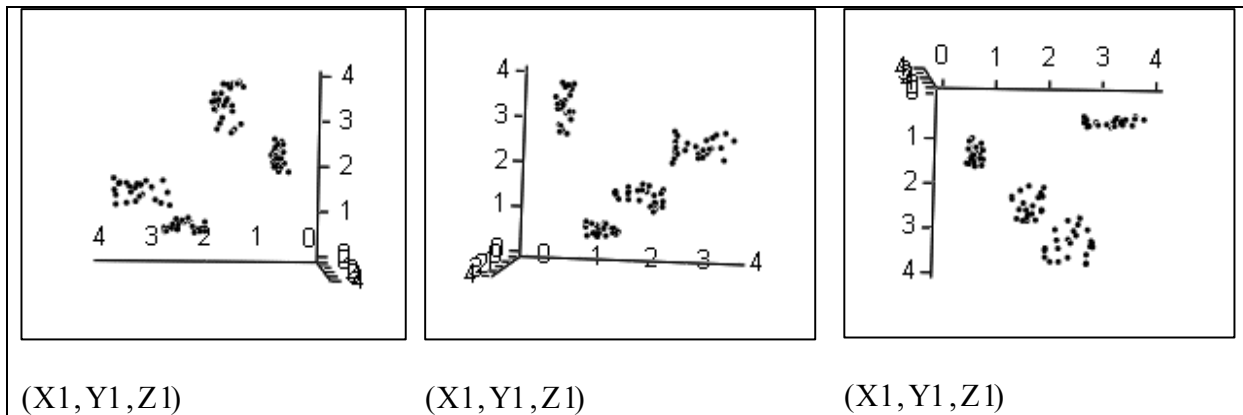


Рис. 5.15. Тестова множина об'єктів. Три проекції в просторі перших трьох ознак

Задаємо номери нейронів:  $pr := 10$ ;  $dr := 11$ ;  $tr := 12$ , для яких дивимось перехідні процеси.

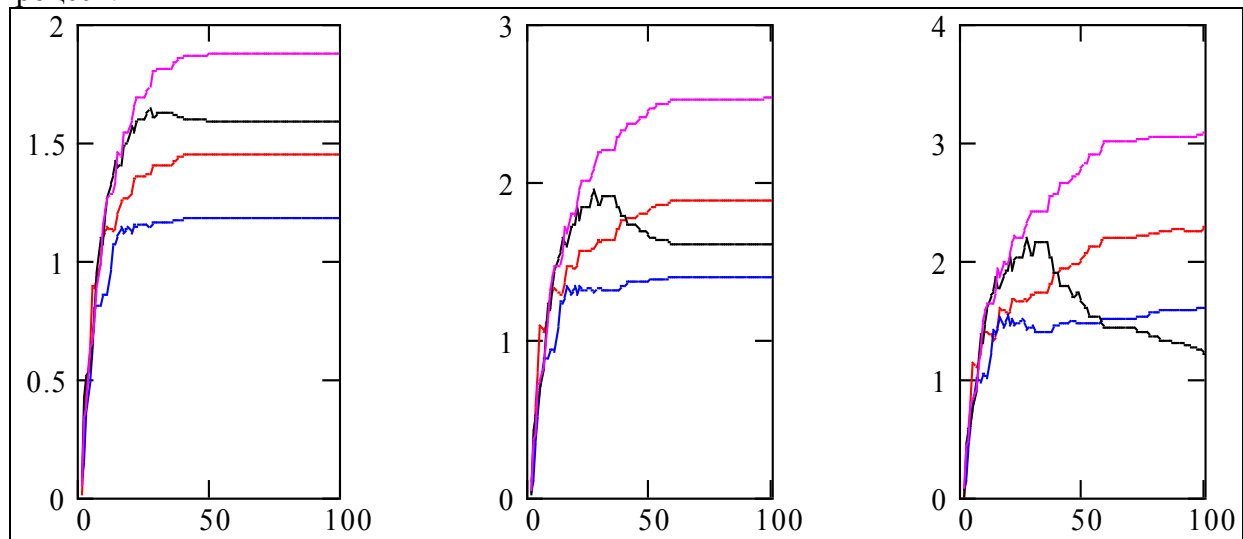


Рис. 5.16. Перехідні процеси у вибраних нейронах



Робимо анімацію. Будемо дивитись еволюцію стану нейронів. Виразимо стан нейронів через системну змінну "фрейм":  $Sost := Ves_{2FRAME+1}$ . Далі все робимо згідно з інструкцією.

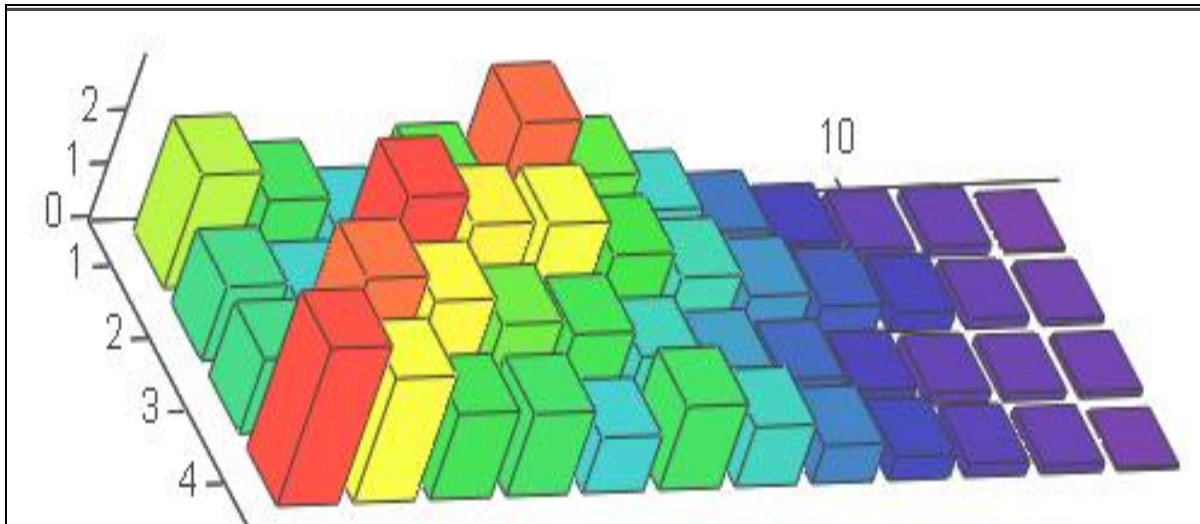


Рис. 5.17. Кінцевий стан "матриці Кохонена" для задачі "4 ознаки, 12 нейронів"

### Висновки

Проаналізуємо, що ми власне отримали і до чого це можна застосувати.

За статистикою і комп'ютерною пресою термін "інтелектуальний", застосований до програмних продуктів, в 90% випадків є просто рекламною прикрасою. З того, що входить в 10%, 9% – це програми на базі нейронних мереж, мурашиних, мишиних і генетичних алгоритмів, що розв'язують задачі, які розв'язуються і без застосування цієї термінології, і 1% новаторських робіт. Однак це природна пропорція.

### Контрольні запитання

1. Алгоритм функціонування штучного нейрона.
2. Алгоритми взаємодії штучних нейронів.
3. Що власне налаштовується в штучній нейронній мережі і для чого?

### Завдання для самостійного виконання

1. Проведіть пошук в Інтернеті – знайдіть документи з відкритими програмами моделювання штучних нейронних мереж, нейронечітких мереж та ін.
2. Проведіть пошук в Інтернеті – знайдіть посібники, монографії з прикладами задач і програм практичного застосування штучних нейронних мереж.
3. Зберіть в Інтернеті літературу, де розглядаються теоретичні і практичні питання напрямку "штучні соціальні мережі".
4. Те, що подано вище, – тільки напівфабрикат. Добудуйте систему так, щоб вона вирішувала певні задачі, наприклад, класифікувала ситуації управління, підручники, художню літературу та ін.

## 5.4 "Швидка" адаптивна САУ

В останньому розділі ми розглядаємо на конкретних прикладах альтернативні і, можливо, перспективні напрямки побудови адаптивних систем управління. На перший погляд ці приклади мають мало спільного. Стандартна реакція в такій ситуації – "вибрати кращу альтернативу".

Спільними рисами альтернатив є розпаралелювання задачі і розумна інтеграція функцій ідентифікації, оптимізації і власне управління. Щодо вибору кращої альтернативи, то можна висунути гіпотезу – кращою альтернативою буде синтез розглянутих альтернатив.

В даному підрозділі розглядаємо останню альтернативу – "швидку" адаптивну САУ. Ми знаємо "швидке перетворення Фур'є", "швидкий регулятор".

Визначимо термін – "швидка". Класичні адаптивні САУ з підручників і монографій виникли як надбудови над звичайними регуляторами. Контур адаптації в такій САУ формував інформацію про якість перехідного процесу і згідно з цим змінював параметри регулятора основного контуру.

Номінально в такій САУ мало місце запізнювання на величину тривалості перехідного процесу. З урахуванням шумів, накопичення даних і фільтрації запізнення могло бути ще більшим. Такі адаптивні САУ могли успішно працювати на об'єктах з дуже повільною зміною параметрів.

Для випадків, коли параметри об'єкта значно змінюються (більше 10% за період перехідного процесу), потрібні інші рішення. Одне з таких рішень подається в даному підрозділі. Суть його в тому, що під певний клас об'єктів і певний клас структур регуляторів (лінійний, релейний) можна знайти, підібрати параметризовані ПРАВИЛА зміни параметрів регулятора, виходячи з прогнозу поведінки об'єкта на базі значень поточного вектора стану. Правило – певне висловлювання, наприклад: "якщо компоненти вектора стану задовольняють певне відношення  $\spadesuit(x_1, x_2, x_3, \dots)$ , то змінити параметри регулятора згідно з алгоритмом:  $\heartsuit(k_1, k_2, k_3, \dots, x_1, x_2, x_3)$ ".

Такі правила фактично задають розбиття простору стану на класи еквівалентності. Знаходження правил – нетривіальна задача, тому існує метаправило: якщо правило працює, не слід робити його загальновідомим.

### Розробка базової програми моделювання "швидкої" адаптивної САУ

Прочитаємо дані з файлів, дані контрольного прикладу з розділу 1:

$K := \text{READPRN}("t21.prn"); K = (0.342 \ 0.486 \ -2.87 \ -2 \ 2 \ 1 \ 0.2 \ 10 \ 0.15);$

$\text{ORIGIN} := 1$ . Розшифровуємо (див. розд. 1.2) дані з файлу:

$kn1 := K_{1,1}; kn2 := K_{1,2}; kn3 := K_{1,3}$  – номінальні параметри регулятора;

$a2 := K_{1,4}; a1 := K_{1,5}; a0 := K_{1,6}$  – номінальні параметри об'єкта;

$T := K_{1,7}; \text{Trz} := K_{1,8}; \text{h mz} := K_{1,9}$  – вимоги до якості регулювання.

Виводимо для контролю дані – номінальні параметри регулятора та параметри моделі об'єкта:  $kn1 = 0.342$ ;  $kn2 = 0.486$ ;  $kn3 = -2.87$ ;  $a1 = 2$ ;  $a1 = 2$ ;  $a0 = 1$ ;  $T = 0.2$ ;  $\text{h mz} = 0.15$ , задаємо кількість кроків моделювання:  $N := 150$ ;  $n := 1..N$ .

Записуємо компоненти математичної моделі об'єкта та початкові умови

$$I := \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}; B := \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}; x^{(1)} := \begin{pmatrix} -1 \\ 0 \\ 0 \end{pmatrix}; k1 := -4; k2 := -4; k3 := -4.$$

**Завдання 1.** На базі поданих слів і формул побудуйте схему адаптивної системи.

Задаємо крок  $T := .1$ , точність обчислень  $\text{TOL} = 0.001$ ;  $t_n := T \cdot n$ ;  $j := 1..3$ . Вводимо розширений вектор стану, задаємо початкове значення вектора параметрів регулятора:

$$\begin{pmatrix} x_1 \\ v_1 \\ w_1 \\ k_{11} \\ k_{21} \\ k_{31} \\ u_1 \end{pmatrix} := \begin{pmatrix} -1 \\ 0 \\ 0 \\ K_{01} \cdot 1 \\ K_{02} \\ K_{03} \\ 0 \end{pmatrix}; \quad K_0 = \begin{pmatrix} 4 \\ 0 \\ -6 \end{pmatrix}; \quad \boxed{\gamma := 0.15} \text{ – параметр настроювання.}$$

Вводимо значення параметра настроювання і записуємо рівняння системи: перші три рівняння – 1, 2, 3 – описують динаміку об'єкта управління певного класу; рівняння 4, 5, 6 – настроювання параметрів об'єкта за певними правилами; рівняння 7 – регулятор.

$$\begin{pmatrix} x_{n+1} \\ v_{n+1} \\ w_{n+1} \\ k_{1n+1} \\ k_{2n+1} \\ k_{3n+1} \\ u_{n+1} \end{pmatrix} := \begin{pmatrix} x_n \\ v_n \\ w_n \\ k_{1n} \\ k_{2n} \\ k_{3n} \\ 0 \end{pmatrix} + \begin{pmatrix} 1 \cdot v_n \\ 1 \cdot w_n \\ -a_0 \cdot x_n - a_1 \cdot v_n - a_2 \cdot w_n + (k_{1n} \cdot x_n + k_{2n} \cdot v_n + k_{3n} \cdot w_n) \\ \text{Наст}[x_n, (k_{1n} + 5)] \cdot 5 \\ -\text{Наст}[v_n, (k_{2n} + 5)] \cdot 16 + \text{Наст}[x_n, (k_{1n} + 1)] \cdot (-7) \\ \text{Наст}[w_n, (k_{3n} + 7)] \cdot 2 + \text{Наст}[x_n, (k_{1n} + 1)] \cdot (-2) \\ k_{1n} \cdot x_n + k_{2n} \cdot v_n + k_{3n} \cdot w_n \end{pmatrix} \cdot T \quad Kk := \begin{pmatrix} k_{1N} \\ k_{2N} \\ k_{3N} \end{pmatrix}.$$

**Завдання 2.** Дивлячись на систему рівнянь угорі, зробіть програму моделювання адаптивної САУ. Визначте через цю програму функцію користувача за таким зразком. Будемо графіки перехідних процесів в системі.

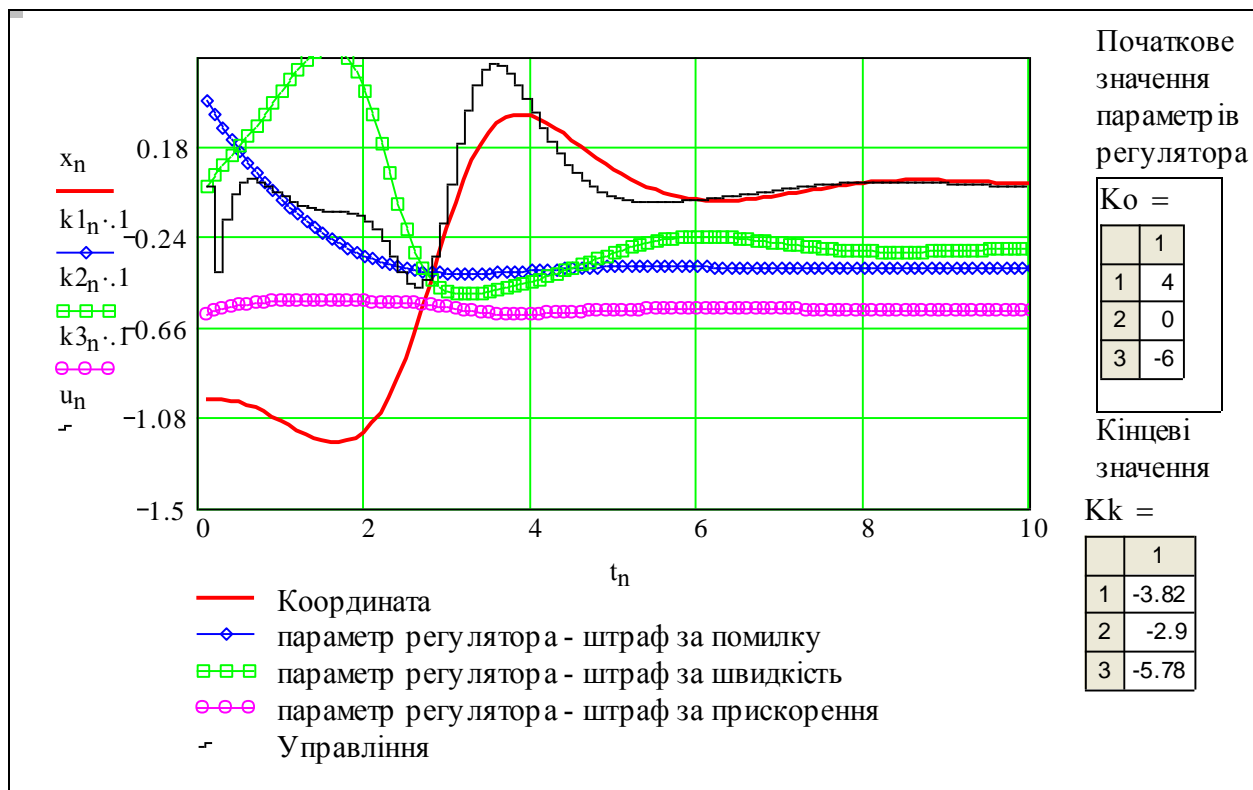


Рис. 5.18. Перехідний процес в адаптивній САУ. Приклад

Першим об'єктом дослідження – залежність кінцевих значень параметра регулятора від початкових – нагадуємо, що ми досліджуємо суттєво нелінійну і нестационарну систему. Вводимо початкове, виводимо кінцеве значення. Вводимо змінні для графіків.

$$K_o \equiv \begin{pmatrix} 4 \\ -0 \\ -6 \end{pmatrix}; K_k = \begin{pmatrix} -3.82 \\ -2.9 \\ -5.78 \end{pmatrix}; xv_n := x_n; uv_n := u_n; kv1_n := k1_n; kv2_n := k2_n; kv3_n := k3_n.$$

Подивимось на графіки (рис. 5.18): там подано перехідні процеси для *нестійкого недемпфованого* об'єкта з *нестійким* на початку регулятором. Бачимо, що процеси стабілізації об'єкта і процеси настроювання закінчуються одночасно. Система в цілому стійка.

Дивимось ще вище на систему рівнянь. Бачимо, що настроювання всіх параметрів регулятора виконується функцією Наст(Координата, Параметр регулятора), в якій власне "упакована" вся складність алгоритму настроювання.

Бачимо *очевидну недосконалість* системи: велику кількість чисел – параметрів "підстроювання" системи самонастроювання. Якщо неможливо раціонально задати ці параметри (аналітично чи через певні процедури), то це означатиме нежиттєздатність даної адаптивної системи. Залишаємо цю проблему для самостійного розв'язання.

Бачимо *очевидну обмеженість* моделі об'єкта управління: це "динаміка матеріальної точки з ненульовою масою". Це вузький для теорії і поширений для практики клас динамічних систем. Згадаємо задачі термінального управління з розділу 3 – там ми теж почали з розгляду руху матеріальної точки.

#### Дослідження "швидкої" адаптивної системи. Вплив початкових умов

Ми отримали не стільки ключ до знань, скільки інструмент – "відмичку". Проведемо серію пошукових експериментів на моделі адаптивної системи. Далі подано серію копій екрана. На графіку (рис. 5.19) подано перехідні процеси для координати, параметрів закону управління, управління та числові значення параметрів управління в початковий і кінцевий моменти процесу.

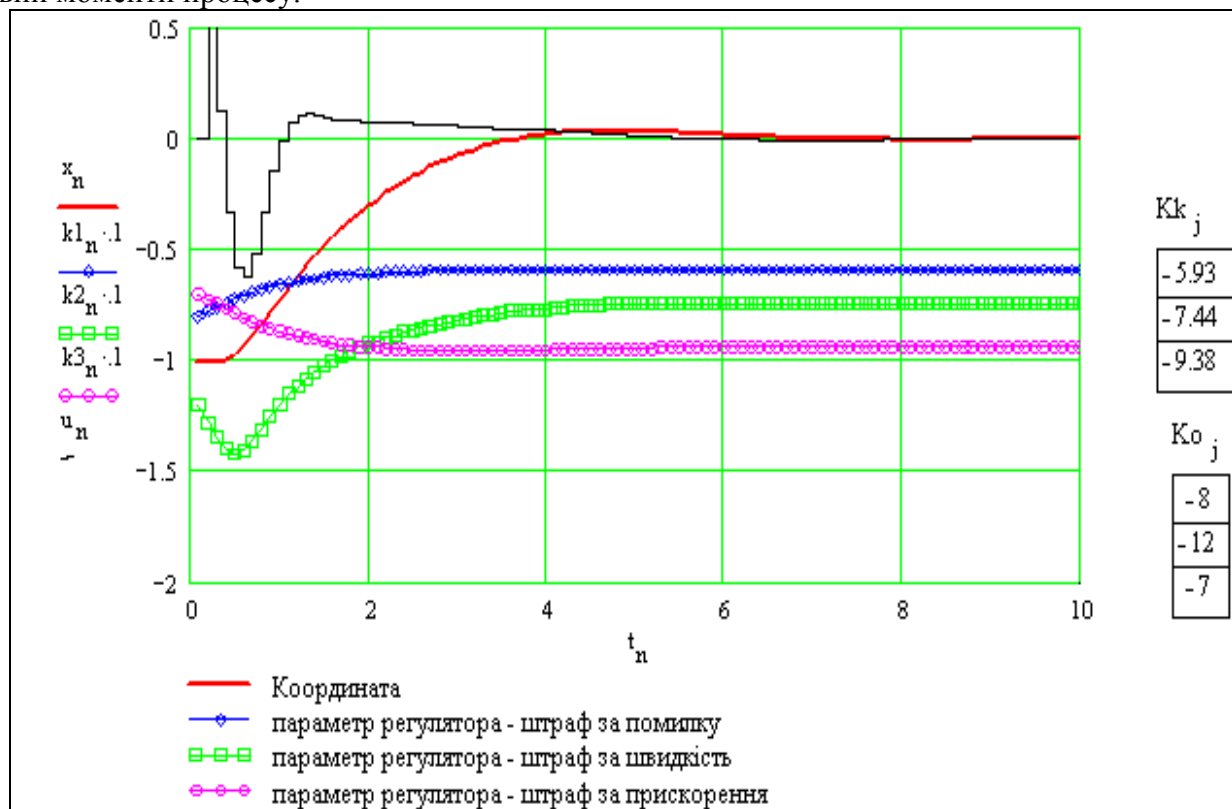


Рис. 5.19. Перехідні процеси для занадто *стійкого*, демпфованого об'єкта

Бачимо, що параметри регулятора змінюються несуттєво – система є задовільною.

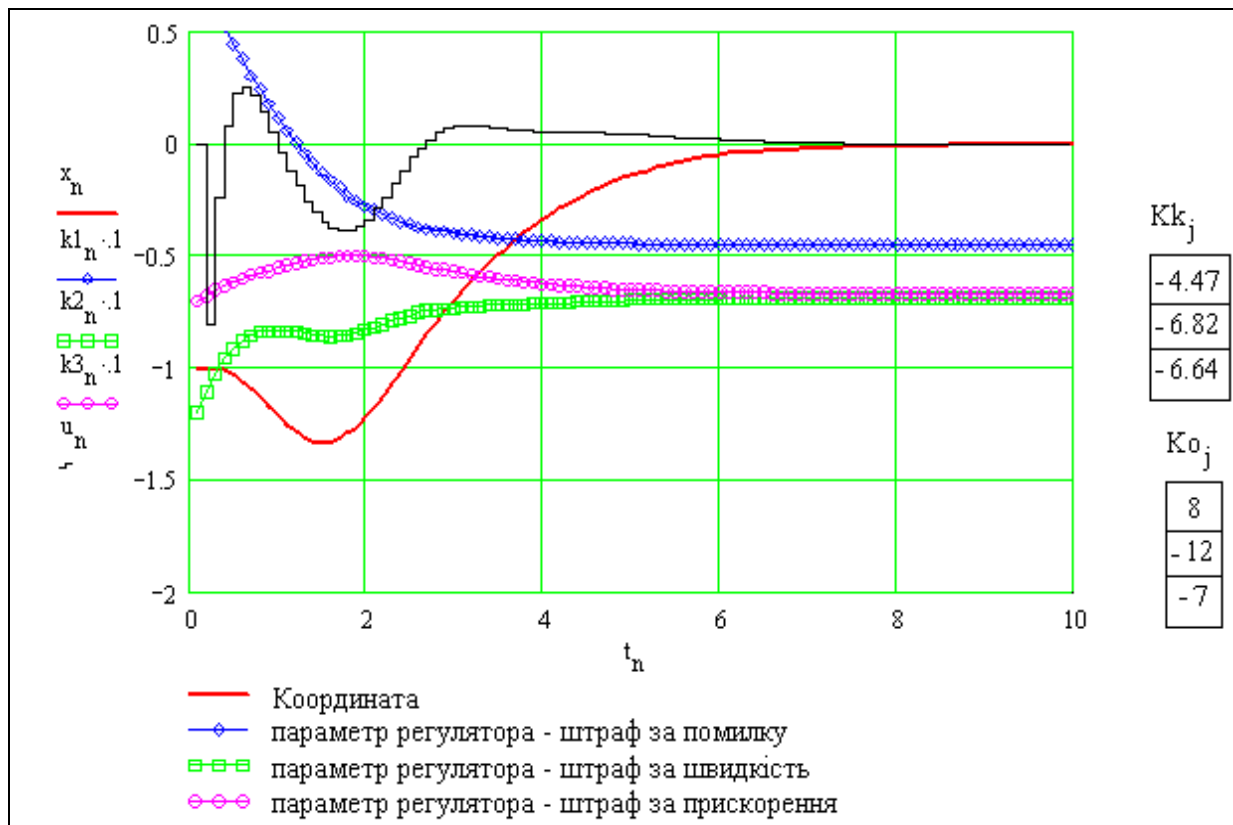


Рис. 5.20. Перехідні процеси для *нестійкого, занадто демпфованого* об'єкта

Бачимо на рис. 5.20, що перехідний процес по координаті збільшується порівняно з попереднім випадком – з 4 до 6. В цілому всі процеси стійкі. Кінцеві параметри регулятора приблизно такі, як і в попередньому випадку.

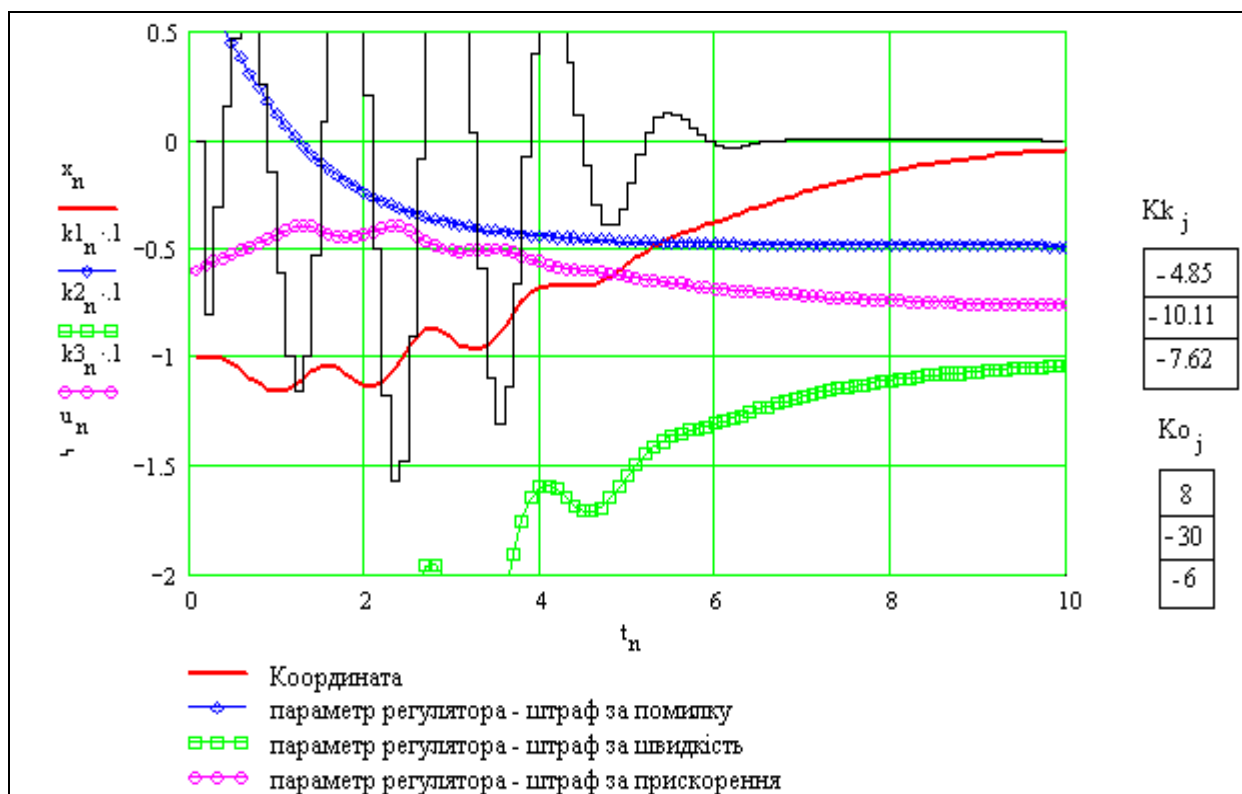


Рис. 5.21. Критична ситуація: *занадто нестійкий, занадто демпфований* об'єкт

Бачимо на рис. 5.21, що перехідний процес по координаті не тільки розтягнутий, але й коливальний. Маємо коливання управління з великою амплітудою. Цей експеримент є цінним: ми знайшли "хворобу" нашої адаптивної системи.

Порівняно зі звичайними САУ наша система має антиінтуїтивну поведінку: при збільшенні початкового демпфування виникають високочастотні коливання управління. В цілому ж процес є стійким.

### Дослідження "швидкої" адаптивної системи. Вплив параметрів настроювання

Запропонована адаптивна система – необґрунтована, невпорядкована, єдиний позитив системи – вона працює. В алгоритмі настроювання маємо десять параметрів. Природно припустити, що між ними є певні залежності, що їх число можна зменшити, що їх величини можна не підбирати, а розраховувати. Перший крок на цьому шляху – провести прості дослідження впливу цих параметрів на якість перехідних процесів в адаптивній системі.

Робимо ще один екземпляр програмного об'єкта "рівняння динаміки адаптивної системи". Змінюємо відповідні параметри і будуємо разом два набори перехідних процесів.

**Тест.** Якщо Ви достатньо орієнтуєтесь в динамічних системах (вивчали розділи 1-4), в формах опису їх динаміки (диференційні, різнищеві рівняння), то Вам неважко визначити, який параметр "приховано" під псевдонімом  $Mu\_Mu$ .

$$\begin{pmatrix} x_{n+1} \\ v_{n+1} \\ w_{n+1} \\ k1_{n+1} \\ k2_{n+1} \\ k3_{n+1} \\ u_{n+1} \end{pmatrix} := \begin{pmatrix} x_n \\ v_n \\ w_n \\ k1_n \\ k2_n \\ k3_n \\ 0 \end{pmatrix} + \begin{bmatrix} 1 \cdot v_n \\ 1 \cdot w_n \\ -a0 \cdot x_n - a1 \cdot v_n - a2 \cdot w_n + (k1_n \cdot x_n + k2_n \cdot v_n + k3_n \cdot w_n) \\ \text{Наст}[x_n, (k1_n + 5)] \cdot 44 \\ -\text{Наст}[v_n, (k2_n + 5)] \cdot 26 + \text{Наст}[x_n, (k1_n + 1)] \cdot (1) \\ -\text{Наст}[w_n, (k3_n + 1)] \cdot 1 + \text{Наст}[x_n, (k1_n + 1)] \cdot (-2) \\ k1_n \cdot x_n + k2_n \cdot v_n + k3_n \cdot w_n \end{bmatrix} \cdot Mu\_Mu.$$

На рис. 5.22 подано разом два перехідних процеси – один для випадку, коли параметри регулятора більше номінальних, другий для випадку, коли параметри регулятора менше номінальних.

Термін "номінальне значення параметрів регулятора" має дві інтерпретації: – оптимальні значення параметрів, знайдені певним методом синтезу; – усталені значення параметрів, що є результатом функціонування адаптивної системи.

Для кожного процесу виведено по шість змінних – три координати вектора стану об'єкта і три параметри регулятора.

Бачимо, що система настроюється для довільних початкових значення параметра регулятора  $K1$ , якщо встановлені достатні значення  $K2$  і  $K3$ . Система може настроїтись з нульового демпфування, якщо  $K1$  є номінальним.

Можемо виділити такі недоліки адаптивної системи:

– усталені помилки в контурі адаптації – при різних параметрах системи настроювання коефіцієнти основного контуру  $K1$ ,  $K2$ ,  $K3$  встановлюються на близьких, але різних значеннях;

– не використано нечітку логіку.

Головна перевага адаптивної системи – процес настроювання виконується протягом перехідного процесу в основному контурі.

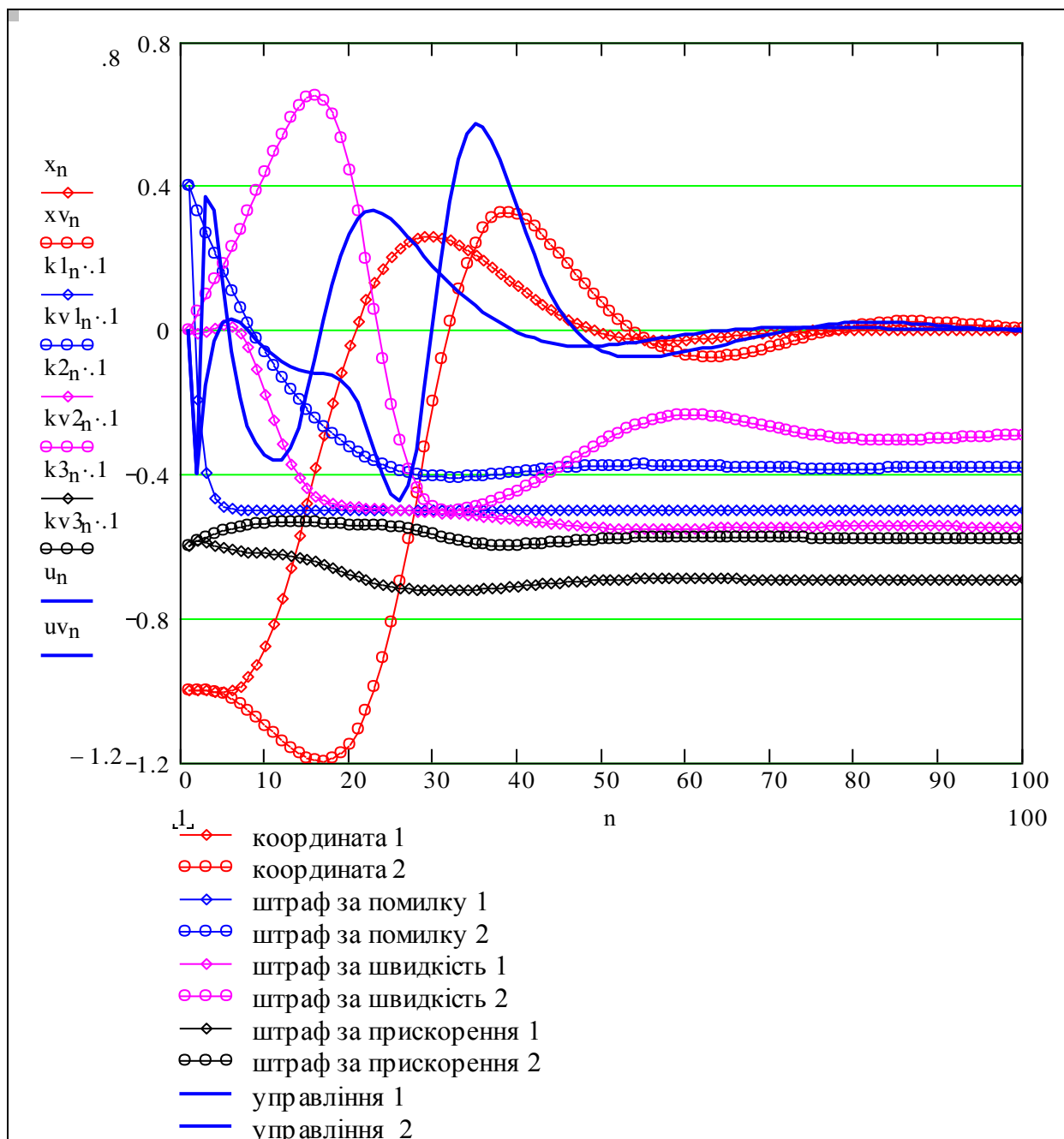


Рис. 5.22. Порівняння процесів при різних параметрах настроювання

### Альтернативні методи отримання інформації для настроювання

За рахунок чого настроювалася наша система? – за рахунок аналізу поведінки системи в перехідному процесі. А якщо помилка в системі нульова, перехідні процеси відсутні? – Ми вже знаємо (підрозділи 1.5, 5.1), що в такому випадку настроювання не буде – вчать-ся саме на помилках. Для процесу навчання (адаптації) слід забезпечити систему перехідними процесами помірною рівня: достатньо інтенсивними для навчання, і недостатньо – для виникнення нестійкості. Забезпечити адаптивну систему потрібною інформацією можна такими способами: – періодична подача тестових імпульсів; – подача гармонічного сигналу; – подача "білого шуму".

Ці сигнали додаються до основного сигналу управління і є малими відносно середньої величини сигналу управління. Коли САУ працює в умовах постійних збурень і шумів (у вимірювачах та виконавчих елементах), то ці спеціальні сигнали для настроювання не потрібні.

Задаємо число кроків моделювання  $N := 400$ ;  $n := 1..N$ , дискретний час  $t_n := T \cdot n$ ,

початковий вектор стану  $\begin{pmatrix} x_1 \\ v_1 \\ w_1 \\ k_{11} \\ k_{21} \\ k_{31} \\ u_1 \end{pmatrix} := \begin{pmatrix} -0.5 \\ 0 \\ 0 \\ K_{01} \cdot 1 \\ K_{02} \\ K_{03} \\ 0 \end{pmatrix}$  та параметри регулятора  $\begin{pmatrix} k_{11} \\ k_{21} \\ k_{31} \end{pmatrix} := \begin{pmatrix} -1 \\ -1 \\ -6 \end{pmatrix}$ .

Формуємо тестові дії

– гармонічні:  $Grm(n, \omega v) := .15 \cdot \sin(\omega v \cdot n \cdot T)$  ;

– випадкові: Шум( $A_s, k_s, n$ ) :=  $A_s \cdot (\text{rnd}(2) - 1) \cdot \text{if}(\text{mod}(n, k_s) = 0, 1, 0)$ ;

– імпульсні:  $Imp(n, A_i, N) := \text{if}\left(\text{mod}\left(n - 1, \frac{N}{4}\right) > 0, 0, 1\right) \cdot A$ .

Задаємо параметри тестових дій:  $\omega v := 0.8$  ;  $A_s := .9$  ;  $k_s := 4$  ;  $A_i := 8$ .

$$\begin{pmatrix} x_{n+1} \\ v_{n+1} \\ w_{n+1} \\ k_{1n+1} \\ k_{2n+1} \\ k_{3n+1} \\ u_{n+1} \end{pmatrix} := \begin{pmatrix} x_n \\ v_n \\ w_n \\ k_{1n} \\ k_{2n} \\ k_{3n} \\ 0 \end{pmatrix} + \begin{bmatrix} 1 \cdot v_n \\ 1 \cdot w_n \\ (k_{1n} - a_0) \cdot x_n + (k_{2n} - a_1) \cdot v_n + (k_{3n} - a_2) \cdot w_n + k_{1n} \cdot Imp(n, A_i, N) \\ \text{Наст}[x_n, (k_{1n} + 5)] \cdot 5 \\ -\text{Наст}[v_n, (k_{2n} + 7)] \cdot 2.2 + \text{Наст}[x_n, (k_{1n} + 2)] \cdot (-3) \\ \text{Наст}[w_n, (k_{3n} + 7)] \cdot 2 + \text{Наст}[x_n, (k_{1n} + 1)] \cdot (-0) \\ k_{1n} \cdot x_n + k_{2n} \cdot v_n + k_{3n} \cdot w_n \end{bmatrix} \cdot T.$$

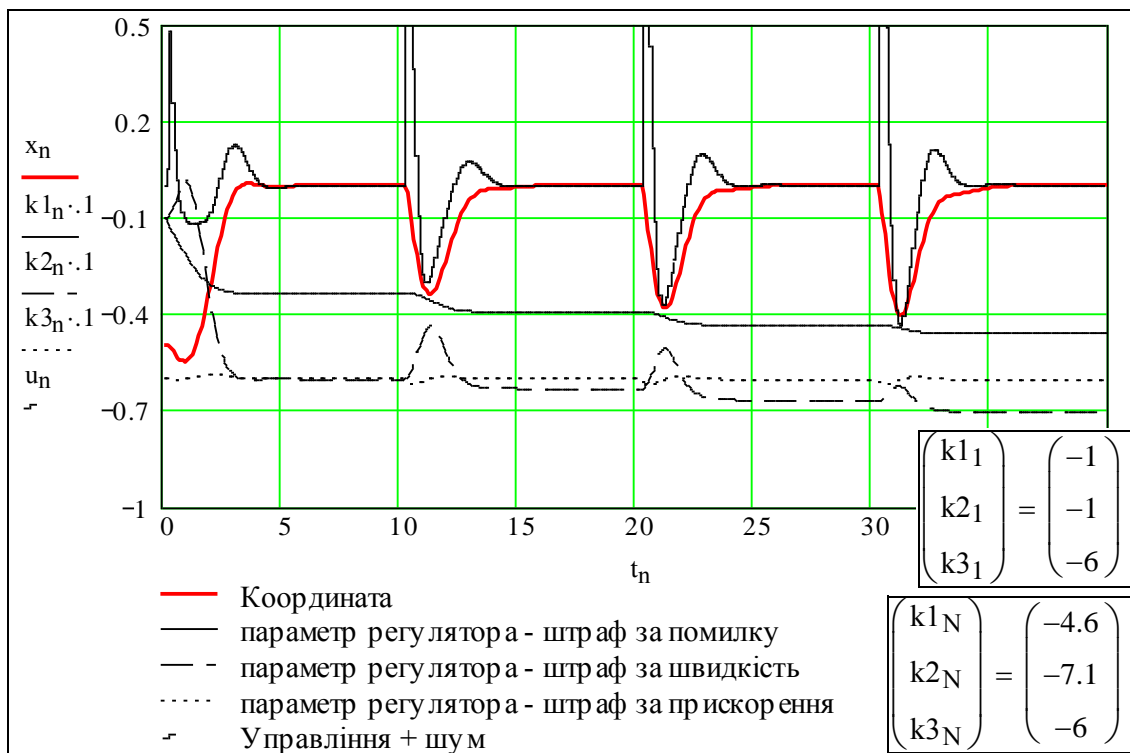


Рис. 5.23. Процес настроювання при використанні імпульсних тестових сигналів



На рис. 5.23 бачимо як спочатку нестійка САУ уже після першого імпульсу стає стійкою. Недолік даних імпульсних дій – ненульове середнє, слід чередувати додатні і від’ємні імпульси.

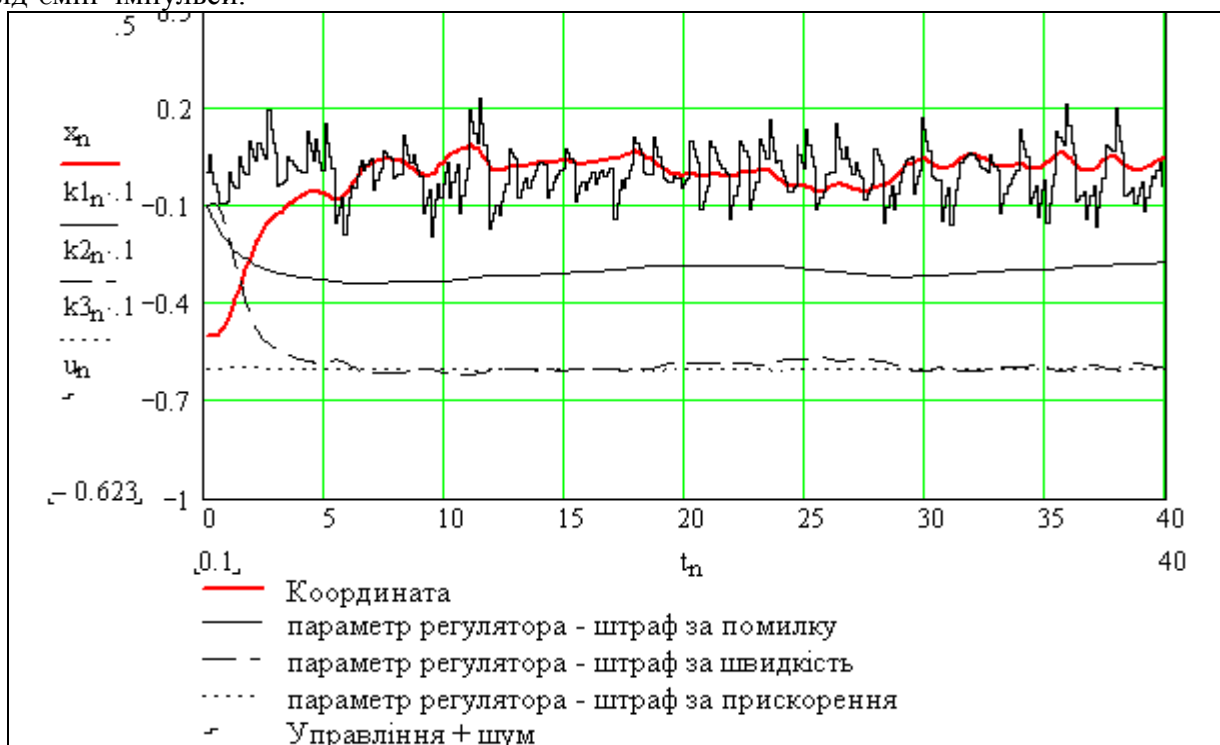


Рис. 5.24. Процес настроювання при використанні випадкових тестових сигналів

Система досить швидко настроюється під дією шумів (рис. 5.24). Чим більше рівень шуму, тим швидке проходить настроювання, однак шуми – це збурення. Звичайно роблять рівень тестових шумів змінним – так, щоб вони зменшувались з часом. Однак це створює нові проблеми – надійності: все що ми робимо змінним, може змінюватись зовсім в іншу сторону...

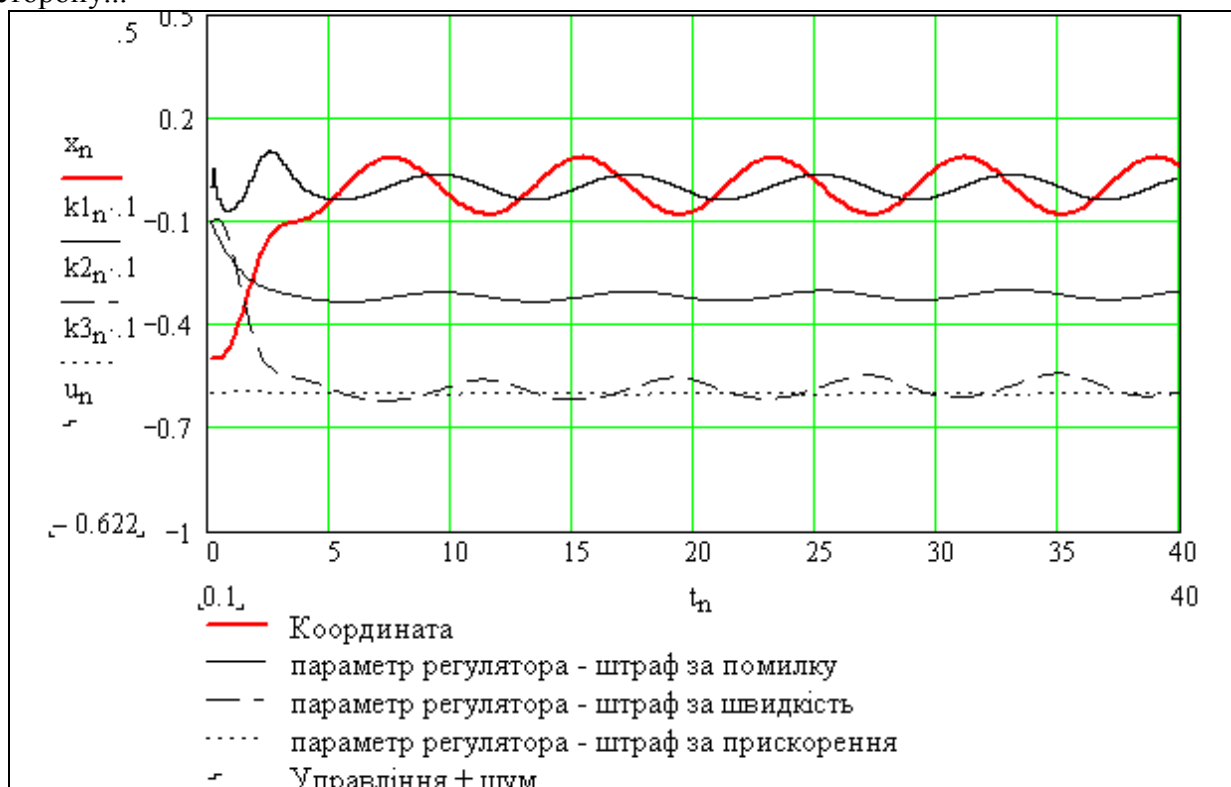


Рис. 5.25. Процес настроювання при використанні гармонічного тестового сигналу

Бачимо (рис. 5.25), що процес настроювання подібний до процесу настроювання при випадкових діях. Побічна проблема – коефіцієнти регулятора мають гармонічну складову. Як і при інших видах тестових дій, слід вибрати оптимальний рівень тестових дій.

### Висновки

Отримали певну робочу гіпотезу, придатну, як мінімум для розбудови магістерської, кандидатської, ... дисертацій. Можна довести цю адаптивну систему до практичного використання.

В цьому розділі отримали, як і в попередніх, підтвердження головному принципу роботи з адаптивними та іншими інтелектуальними системами: завжди слід пам'ятати, що ці системи звичайно створюють більше проблем, ніж розв'язують.

Розглянута система має ряд недоліків. Існують два шляхи усунення недоліків:

- настроювати "сторожів, щоб стерегти сторожів";
- збирати мережеві системи з альтернативних регуляторів, піти на певну диверсифікацію.

При бажанні можна назвати це: "штучні нейронні мережі", "штучні соціальні мережі", "колективи автоматів", "ансамблі динамічних систем", навіть "нейродинамічні системи". На жаль, це не позбавляє від необхідності думати. Однак думати слід за раціональними технологіями і предметно. Цей розділ – приклад предметного думання: гіпотези, висновки, рішення реалізуємо в середовищі математичного пакета, експериментуємо, і так отримуємо предметний матеріал для роздуму.

Це не "нова інформаційна технологія", а скоріше напівзабута інтелектуальна технологія. Відомо, що більшість відомих конструкторів зброї – Лауреатів, Академіків починали слюсарями, і потім кожний елемент гвинтівки чи автомата випробовували тисячі разів своїми руками.

### ПІДСУМКИ

Розглянули ряд задач адаптивного управління і побудували працюючі моделі адаптивних систем, а саме:

- систему з настроюванням власної частоти коливань в перехідних процесах;
- систему стійку до відмов вимірювачів;
- систему термінального управління стійку до відмов виконавчих елементів;
- приклади систем з екстремальними регуляторами:
  - ♥ систему оптимального розподілу навантаження між реакторами для синтезу стиролу;
  - ♥ систему для збалансування і оптимізації потоків кредитів і депозитів;
- розглянуто ряд екзотичних регуляторів.

Всі програми доведені до рівня "генераторів результатів" і базових програм, придатних для подальшого розвитку.

Тепер можна побачити, що ми в більшості задач розглядали одну і ту ж задачу – задачу оптимального **розподілу обмежених узагальнених ресурсів між паралельно працюючими елементами** – реакторами, приводами, вимірювачами, нейронами.

Ми зовсім не розглядали інший клас задач розподілу – в процесах розвитку, коли об'єкт з часом суттєво змінюється, росте. Це тема іншої книги.

## ЛІТЕРАТУРА

1. Бадьора С. П. Оптимальне управління інтегрованою системою «виробництво-постачання». Задача згладжування // Інформаційні технології та комп'ютерна інженерія. – 2006. – №1(5). – С. 58–62.
2. Бадьора С. П., Боровська Т. М. Детермінована модель для прогнозування розвитку розподілених систем // Вісник ВПІ. – 2006. – №1. – С. 45–61.
3. Бадьора С. П., Боровська Т. М. Імовірнісна модель для прогнозування розвитку розподілених систем // Вісник ВПІ. – 2006. – №2. – С. 41–57.
4. Бадьора С. П., Боровська Т. М., Северілов В. А. Моделювання багатопродуктових виробничих систем // Вісник ВПІ. – 2004. – №2. – С. 76.
5. Батенко А. П. Управление конечным состоянием движущихся объектов. – М.: Советское радио, 1977. — 277 с.
6. Беллман Р. Процессы регулирования с адаптацией. — М.: Наука, 1964. — 317 с.
7. Беллман Р., Гликсберг И., Гросс О. Некоторые вопросы математической теории управления. — М.: Издат. Иностран. Литер., 1962. — 233 с.
8. Болтянский В. Г. Математические методы оптимального управления. — М.: Наука, 1966. — 307 с.
9. Боровская Т. Н., Северилов В. А., Колесник И. С. Детская экономика. Моделирование и оптимизация производственных систем // Компьютеры + Программы. — 2002. — №2. — С. 43–47.
10. Боровська Т. М., Бадьора С. П., Северілов В. А. Оптимальне управління розвитком техніко-економічних систем. Цінові стратегії // Вісник ВПІ. — 2003. — № 6. — С. 142–147.
11. Боровська Т. М., Колесник І. С., Северілов В. А. Оптимальне управління розвитком техніко-економічних систем. Кредитні стратегії // Вісник ВПІ. — 2003. — № 6. — С. 137–142.
12. Боровська Т. М., Колесник І. С., Северілов В. А. Основи теорії управління та дослідження операцій. Навчальний посібник. – Вінниця: ВНТУ, 2003. – 242 с.
13. Боровська Т. М., Северілов В. А., Васюра А. С. Проблеми розробки адаптивних САУ для масових об'єктів // Вісник ВПІ. – 1998. – № 6. – С. 12–17.
14. Боровська Т. М., Северілов В. А., Васюра А. С. Теорія автоматичного управління. Частина 1. Аналіз САУ. Навчальний посібник. - Вінниця: ВДТУ, 2002. – 96 с.
15. Боровська Т. М. Декомпозиційні структури для прикладних програм синтезу регуляторів // Вісник ВПІ. — 2000. — № 1. — С. 17–22.
16. Боровська Т. М., Колесник І. С., Северілов В. А. Спеціальні розділи вищої математики. Навчальний посібник. – Вінниця: ВНТУ, 2003. – 182 с.
17. Боровська Т. М., Колесник І. С., Северілов В. А. Оптимізація розподілу обмеженого ресурсу у виробничій системі на базі агрегування виробничих функцій // Інформаційні технології та комп'ютерна інженерія. – 2005. – № 1. – С. 12–18.
18. Боровська Т. М., Северілов В. А. Методичні вказівки до лабораторних робіт з курсу ТАУ. — Вінниця: ВПІ, 1992. — 64 с.
19. Боровська Т. М., Северілов В. А., Колесник І. С., Бадьора С. П. Моделювання і оптимізація у менеджменті. Навчальний посібник. – Вінниця: УНІВЕРСУМ-Вінниця, 2009. – 145 с.
20. Вагапов В. Б. Основы теории радиоэлектронных автоматических систем. Справочник. – Киев: Издательство УСХА, 1992. – 238 с.
21. Воронов А. А. Теория автоматического управления. Ч. 1, 2. — М.: Энергия, 1986. — 503 с.

22. Егоров А. И. Оптимальное управление линейными системами. — Киев: Выща школа, 1988. — 356 с.
23. Живучесть САУ. Методы декомпозиции и реконфигурации: Методические указания к изучению спецразделов курса "Теория автоматического управления" / Сост. В. А. Северилов, Т. Н. Боровская. — Винница: ВПИ, 1989. — 36 с.
24. Изерман Р. Цифровые системы управления / Пер. с англ. — М.: Мир, 1984. — 541 с.
25. Куо Б. Теория и проектирование цифровых систем управления / Пер. с англ. — М.: Машиностроение, 1986. — 448 с.
26. Куропаткин П. В. Оптимальные и адаптивные системы. — М.: Высшая школа, 1980. — 288 с.
27. Ли Т. Г., Адамс Г. Э. Управление процессами с помощью вычислительных машин. Моделирование и оптимизация. — М.: Советское радио, 1972. — 313 с.
28. Мерриэм К. У. Теория оптимизации и расчёт систем с обратной связью. — М.: Мир, 1967. — 550 с.
29. Методичні вказівки до виконання на ПЕОМ лабораторних і практичних занять з курсу ТАУ / Укл. Т. М. Боровська, В. А. Северілов, П. В. Северілов. — Вінниця: ВПІ, 1994. — 53 с.
30. Методичні вказівки до лабораторних робіт з курсу ТАУ / Т. М. Боровська, В. А. Северілов, П. В. Северілов. — Вінниця: ВПІ, 1992. — 60 с.
31. Первозванский А. А. Курс теории автоматического управления. — М.: Наука, 1986. — 615 с.
32. Пузырёв В. А. Управление технологическими процессами производства микроэлектронных приборов. — М.: Радио и связь, 1984. — 160 с.
33. Руководство по проектированию систем автоматического управления. Под ред. проф. В. А. Бесекерского. — М.: Высшая школа, 1983. — 295 с.
34. Пешель М. Моделирование сигналов и систем. — М.: Мир, 1981. — 301 с.
35. Северілов В. А., Боровська Т. М. Принципи побудови модульних мехатронних систем // Вісник ВПІ. — 1999. — № 2. — С. 6–10.
36. Северилов В. А. К выбору проектной концепции робота-манипулятора. / Теория автоматизированного проектирования, вып. 3. Межвуз. тем.сборн.науч.трудов. — Харьков, 1981. — С. 96–101.
37. Северилов В. А., Боровская Т. Н. Живучесть САУ. — Винница: ВПИ, 1989. — 34 с.
38. Теория автоматического управления. Под ред. А. В. Нетушила. Учебник для вузов. — М.: Высшая школа, 1976. — 400 с.
39. Теория автоматического управления: Учеб. для вузов по спец. "Автоматика и телемеханика". — В 2-х ч. — Ч. 1. Теория линейных систем автоматического управления. Под ред. А. Воронова. — М.: Высшая школа, 1986. — 504 с.
40. Цыпкин Я. З. Адаптация и обучение в автоматических системах. — М.: Наука, 1968. — 400 с.

*Навчальне видання*

**Таїса Миколаївна Боровська,  
Анатолій Степанович Васюра,  
Віктор Андрійович Северілов**

**МОДЕЛЮВАННЯ ТА ОПТИМІЗАЦІЯ СИСТЕМ  
АВТОМАТИЧНОГО УПРАВЛІННЯ**

Навчальний посібник

Редактор В. Дружиніна

Оригінал-макет підготовлено Т. Боровською

Підписано до друку  
Формат 29,7×42¼ Папір офсетний  
Гарнітура Times New Roman  
Друк різнографічний Ум. друк. арк.  
Наклад 100 прим. Зам. №

Вінницький національний технічний університет  
Свідоцтво суб'єкта видавничої справи  
серія ДК № 3516 від 01.07.2009 р.  
21021, м. Вінниця, Хмельницьке шосе, 95  
ВНТУ, ГНК, к. 114  
Тел. (0432) 59-85-32

Віддруковано у Вінницькому національному технічному університеті,  
в комп'ютерному інформаційно-видавничому центрі  
Свідоцтво суб'єкта видавничої справи  
серія ДК № 3516 від 01.07.2009 р.  
21021, м. Вінниця, Хмельницьке шосе, 95  
ВНТУ, ГНК, к. 114  
Тел. (0432) 59-81-59