

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Інститут інноваційних технологій і змісту освіти
Національний аерокосмічний університет ім. М. Є. Жуковського
“Харківський авіаційний інститут”

М. Д. КОШОВИЙ, А. Г. МИХАЙЛОВ

**ТЕОРІЯ І ПРАКТИКА МОДЕЛЮВАННЯ
ІНФОРМАЦІЙНО-ВИМІРЮВАЛЬНИХ СИСТЕМ**

Затверджено Міністерством освіти і науки України
як підручник для студентів напрямку підготовки
"Метрологія та інформаційно-вимірвальні технології"
вищих навчальних закладів

Харків “ХАІ” 2014

УДК 681.5.08 : 621.391 (075.8)
ББК 32.816 : 34.9 я 73
К76

Рецензенти: д-р техн. наук, проф. П. О. Качанов,
д-р техн. наук, проф. О. М. Крюков,
д-р техн. наук, проф. Ю. В. Куц

Гриф надано Міністерством освіти і науки України
(лист № 1.11-1723 від 5.02.2014)

Кошовий, М. Д.

К76 Теорія і практика моделювання інформаційно-вимірювальних систем
[Текст] : підручник / М. Д. Кошовий, А. Г. Михайлов. – Х. : Нац.
аерокосм. ун-т ім. М. Є. Жуковського “Харк. авіац. ін-т”, 2014. – 124 с.

ISBN 978-966-662-363-1

Розглянуто особливості моделювання технічних об'єктів за допомогою ЕОМ. Описано алгоритми моделювання об'єктів статичних і динамічних систем. Викладено принципи побудови математичних моделей інформаційно-вимірювальних систем.

Наведено приклади дослідження систем з використанням стандартних пакетів STRATUM-2000, STATGRAPHICS, MATLAB, а також розробок на алгоритмічних мовах програмування СРР, Делфі, Паскаль.

Для студентів денної й заочної форм навчання вищих навчальних закладів технічних спеціальностей, що вивчають інформаційно-вимірювальні системи.

Іл. 57. Табл. 5. Бібліогр.: 20 назв

УДК 681.5.08 : 621.391 (075.8)
ББК 32.816 : 34.9 я 73

© Кошовий М. Д., Михайлов А. Г., 2014
© Національний аерокосмічний
університет ім. М. Є. Жуковського
“Харківський авіаційний інститут”, 2014

ISBN 978-966-662-363-1

ВСТУП

Різні етапи дослідження, синтезу і аналізу систем нерозривно пов'язані з процесом моделювання математичних систем з метою забезпечення кращої наочності й з'ясування механізму явищ, що відбуваються в об'єктах інформаційно-вимірювальних систем.

Моделювання – галузь науки, що застосовується для вирішення широкого кола задач, в числі яких знаходяться такі важливі, як оптимізація процесів і побудови різноманітних систем.

Основу теорії моделювання становить математична статистика (пасивні експерименти повністю базуються на ній), яка, в свою чергу, оснований на теорії ймовірностей. Остання пов'язана з моделюванням через значення вихідних параметрів експериментальних установок. Експериментальні дослідження постійно супроводжують будь-який дослідницький процес, у тому числі й в комп'ютерній галузі. Найбільше практичне значення має оптимізація процесів, або планування екстремальних ситуацій. При цьому завжди задається набір вхідних і керівних характеристик, які контролюються певним набором статистичних значень, що отримуються при експерименті.

У цьому підручнику основну увагу приділено особливостям побудови алгоритмів моделювання інформаційно-вимірювальних систем, необхідних для ідентифікації процесів технічних об'єктів на ЕОМ.

Наведено приклади дослідження систем. Реалізація розглянутих алгоритмів передбачена в інтегрованих середовищах стандартних пакетів, таких, як STRATUM-2000, MATLAB, STATGRAPHICS, а також у вигляді прикладів користувальницьких програм, розроблених на алгоритмічній мові програмування C++, PASCAL, що робить опрацьоване програмне забезпечення незалежним від конфігурації використовуваної ЕОМ.

1. МОДЕЛЮВАННЯ І АНАЛІЗ СИСТЕМ

При моделюванні систем вважається, що знання можуть бути отримані шляхом емпіричного дослідження.

Емпіричний - це такий рівень знання, зміст якого отримано дослідним шляхом (спостереженням або експериментуванням). На цьому рівні фіксуються якості й властивості об'єкта, що вивчається. Дані спостережень і експериментів утворюють емпіричну основу теоретичного дослідження.

Теоретичний рівень надбудовується над емпіричним і має чітку спрямованість - пояснення об'єктивної реальності. Однак теорія будується таким чином, що вона безпосередньо описує не навколишню дійсність, а ідеальні об'єкти. Тому евристичний, або теоретичний, рівень знання спирається на абстрактне мислення, для якого початковим пунктом дослідження є результати, отримані в ході почуттєвого сприйняття.

Необхідність в емпіричних знаннях часом є причиною поділу наук на експериментальні й теоретичні, хоча на практиці неможливо із експериментальних дисциплін повністю виключити теорію, а із теоретичних – будь-яку згадку про експеримент. Для емпіричних знань характерне те, що вони є результатом безпосереднього контакту з живою реальністю в спостереженні або експерименті. Реальні об'єкти, в зв'язку зі складністю всебічного дослідження механізму їх функціонування, як правило, не підлягають суворому математичному опису, що приводить до необхідності проведення натурних, або імітаційних, експериментів. Саме експеримент часто є критерієм істини.

Існує два способи емпіричного дослідження складних систем: спостереження, або пасивні дослідження, і активний експеримент.

Спостереження – цілеспрямоване сприйняття об'єкта без активного втручання в його поведінку. Дослідник повинен пасивно чекати природного прояву необхідних ефектів у поведінці об'єкта, що значно збільшує час збору необхідної інформації.

Щоби спостереження було плідним, воно повинно дбати про досягнення поставленої мети: бути цілеспрямованим, таким, що фіксує тільки характеристики об'єкта, які цікавлять спостерігача, і вести активний пошук цих

характеристик; планомірним і систематичним, тобто враховувати завдання дослідження й можливості сприйняття об'єкта в різних умовах його функціонування.

Будь-якому моделюванню і запланованому спостереженню властиві обмеження, зумовлені пасивністю спостерігача відносно об'єкта. При проведенні пасивного експерименту зазвичай порушуються основні передумови, що забезпечують отримання незалежних оцінок параметрів моделей і гарантують високу ефективність статистичних критеріїв, які використовуються при аналізі.

Отримані в ході пасивного експерименту дані містять недостатньо корисної інформації, погано інтерпретуються, а самі спостереження дуже часто не можуть бути реалізовані через значну тривалість часу збору необхідної інформації й випадковий характер устанавлення набору вхідних величин, що впливають на об'єкт, які цікавлять дослідника.

Експеримент - це вид діяльності, що використовується з метою наукового пізнання, відкриття об'єктивних закономірностей і характеристик об'єктів, який полягає в планомірному впливі на об'єкт, що вивчається, або процес за допомогою спеціальних методів, інструментів і приладів.

Експеримент дозволяє: ізолювати явище, що вивчається, від подібних, неістотних і таких, що затемнюють його суть, явищ, вивчати його в чистому вигляді, багаторазово відтворювати хід процесу в суворо фіксованих, що піддаються контролю, умовах; планомірно змінювати, варіювати, комбінувати різні умови з метою отримання шуканого результату [1, 2].

Існує й інше визначення експерименту: експеримент – це сувора послідовність заздалегідь визначених дій, спрямованих на досягнення мети, отримання числових значень деякої множини величин, що являють собою результат експерименту.

Експеримент повинен мати властивість відтворюваності, тобто повторюваності результатів. Часто експеримент є єдиним ефективним інструментом наукових досліджень.

Будь-який експеримент, якою б складною не була уява про нього, завершується результатами, формулюванням висновків, прийняттям рішень і,

можливо, наданням рекомендацій іншим дослідникам. Ця інформація може бути подана у вигляді таблиць, графіків, математичних формул або словесних описів, тому дослідник повинен володіти математичним апаратом, уміти викласти результати досліджень, робити правильні висновки і приймати обґрунтовані рішення.

Для проведення експерименту необхідна модель об'єкта, а для уточнення параметрів моделі потрібен експеримент. Відносини між експериментом і моделлю двоякі. З одного боку, експеримент дозволяє перевірити і за необхідності уточнити модель, тобто є джерелом інформації для моделювання. З іншого боку, модель диктує, який саме експеримент потрібно провести, тобто модель служить джерелом інформації для організації експерименту.

Для отримання інформації про об'єкт, що цікавить дослідника, необхідно виконати вимірювання, тобто деяку алгоритмічну операцію, яка певному спостережувальному стану об'єкта, процесу або явищу ставить у відповідність певне позначення - число, символ, номер. Потрібна інформація одержується з результатів вимірювання шляхом їх перетворень, або, як ще кажуть, за допомогою оброблення експериментальних даних.

З постановкою і проведенням експериментів фактично пов'язана історія розвитку людства загалом. Однак у більшості випадків ці роботи проводилися хаотично, на рівні інтуїції й попереднього досвіду, тому коефіцієнт корисної дії їх виявлявся досить низьким. Якщо врахувати, що вартість одного експерименту, як правило, вельми висока, то неважко уявити собі, скільки коштувало людству проведення науково не спланованих експериментів. Особливо це важливо при дослідженні складних систем, поведінка яких залежить від великої кількості факторів.

Дуже часто характеристики об'єкта дослідження, які необхідно визначити в ході експерименту, виявляються недоступними для безпосереднього вимірювання. Проведення експерименту вважається "мистецтвом, якому можна навчитися, але якому не можна навчити" [2]. Деякі експериментатори, маючи вельми скудні й недостовірні дані, здатні відкрити нові важливі принципи, тоді як інші навіть при вичерпному дослідженні лишають поза увагою очевидні факти. І, хоча мистецтву робити відкриття не можна навчитися,

навчити мистецтву підготовки до відкриттів все ж можливо. Можна навчити виключати або враховувати випадкові впливи навколишнього середовища, планувати експерименти і визначати порядок зміни умов проведення експерименту, оцінювати помилки, перевіряти отримані дані й подавати їх у впорядкованому і наочному вигляді.

Одним з основних завдань планування процесу моделювання є оптимальне розташування експериментальних точок у відведеній для експерименту області факторного простору. Найважливішою умовою наукового експерименту є мінімізація загальні кількості дослідів, витрат і часових ресурсів, причому зменшення кількості дослідів не повинно істотно відбиватися на якості отриманої інформації. Для побудови плану експерименту останнім часом широко використовується математичне планування – могутній інструмент статистичного дослідження й оптимізації складних систем. Виключаючи сліпий пошук, наукові методи планування значно скорочують кількість дослідів, а отже, й терміни проведення експерименту і витрати на нього, дають можливість дістати кількісні оцінки параметрів моделі.

Для планування процесів моделювання використовується теорія планування експериментів, що являє собою самостійну дисципліну, яку розвивають багато вчених. Початок теорії планування експериментів – 1918 рік, коли з'явилася перша робота Р. Фішера, присвячена розробленню методу планування дисперсійного аналізу. Відтоді ця теорія перетворилася в широко розгалужену і самостійну наукову дисципліну, що має виключно широку сферу практичного застосування. Методи планування експерименту універсальні, тобто придатні для більшості галузей дослідження - хімії, хімічної технології, матеріалознавства, промисловості, сільського господарства, медицини, біології, соціології, економіки, електроніки, радіотехніки, автоматики, обчислювальної техніки та ін.

Предметом теорії планування експерименту є вирішення завдання планування експериментів при натурному моделюванні, що передбачає включення в практику дослідника способів, що дозволяють збільшити ефективність спостережень, надійність оцінок статистичних висновків, наочність інтерпретації результатів.

Завдання планування експерименту в тому чи іншому обсязі вирішуються при проведенні будь-якого експериментального дослідження. При плануванні експерименту необхідно спочатку визначити мету експерименту і показники його якості, вказати характеристики плану і побудувати модель експерименту, вибрати критерії оптимальності плану і встановити обмеження на показники якості дослідження. Для побудови оптимального плану найчастіше не потрібне проведення розрахунків, план можна підібрати із каталогів.

Організація і проведення процесу моделювання потребує застосування особливих методів планування експерименту, тобто процедури вибору кількості дослідів, необхідних і достатніх для вирішення задачі з необхідною точністю або статистичною надійністю, умов постановки дослідів, методів математичного оброблення результатів і методів теорії прийняття рішень.

Таким чином, після того, як вибрано модель об'єкта і сформульовано мету дослідження, необхідно спланувати експеримент: вибрати методи вимірювань величин, за допомогою чого безпосередньо вимірюються і можливі типи засобів вимірювань; дати апріорне оцінювання похибок вимірювань; сформулювати вимоги до метрологічних характеристик засобів вимірювань і умов вимірювання, вибрати засоби вимірювань відповідно до вказаних вимог, а також параметри вимірювальної процедури – координати точок області експерименту, кількості спостережень для кожної точки плану, моментів часу вимірювання, послідовності проходження точок плану; підготувати засоби вимірювання до проведення експерименту; забезпечити умови проведення експерименту. Експеримент необхідно реалізувати таким чином, щоби при мінімальній кількості дослідів, варіюючи значення незалежних змінних за спеціально сформульованими правилами, побудувати математичну модель і знайти значення факторів, що забезпечують оптимальне функціонування системи. Залежно від поставленого перед дослідником завдання можуть застосовуватися різні моделі. У задачах ідентифікації об'єкта використовують моделі регресійного аналізу, причому залежність функції мети від факторів може бути подана у вигляді будь-якої монотонної функції або полінома відповідного порядку. При цьому можливі декілька ситуацій, від яких залежить мета експерименту: структура або тип моделі відомі й результати експерименту

використовуються для знаходження оцінок коефіцієнтів – параметрів моделі; для опису об'єкта може бути використано декілька моделей, і потрібно вибрати кращу з них і оцінити її параметри; модель об'єкта, що вивчається, заздалегідь невідома, і в результаті експерименту потрібно найкращим чином підібрати опис, гіпотезу про структуру якого можна висунути в ході експерименту.

Перші дві ситуації є досить визначеними і можуть бути вирішені застосуванням відповідних стратегій організації експерименту. Розв'язання задачі в умовах невизначеності типу моделі неоднозначне й ускладнене, оскільки висунення гіпотези про вигляд моделі за результатами експерименту може призвести до прийняття неправильного розв'язку.

Відповідно до теорії систем будь-яку систему можна подати у вигляді моделі, на яку впливають вхідні фактори X , або регресори. Спостереження за такою системою здійснюється за оціненими значеннями \tilde{Y} , що включають як вектор відклику Y , так і помилку $V(t)$.

На рис. 1.1 подано систему з адитивною помилкою.

Припустимо, що перетворення X у Y здійснює оператор $A[\cdot]$. Визначення параметрів оператора $A[\cdot]$ є задачею ідентифікації.

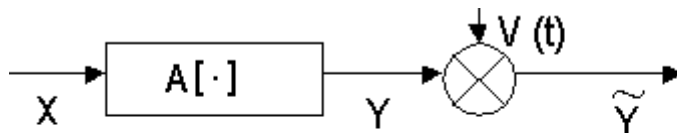


Рис. 1.1. Система з адитивною помилкою

За експериментальними даними X , Y дослідник може побудувати ламану лінію, яка називається лінією регресії (рис. 1.2).

Залежно від вибраної гіпотези H структурної ідентифікації дослідник може вибрати відповідний вигляд моделі (статичний або динамічний) в різних математичних поданнях (поліноміальному, експоненціальному та ін.) [1 - 3].

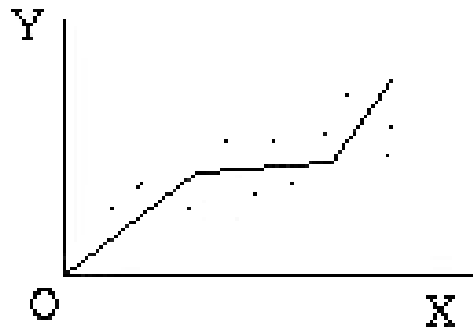


Рис. 1.2. Лінія регресії

1.1. Поліноміальна модель

Найбільш поширеним математичним поданням є поліноміальна модель. Так, наприклад, лінійна модель системи другого порядку має вигляд

$$y = b_0 + b_1x_1 + b_2x_2.$$

Багатовідкликова лінійна модель:

$$\begin{bmatrix} y_1 \\ y_2 \\ \dots \\ y_n \end{bmatrix} = \begin{bmatrix} b_{00}, b_{01}, \dots, b_{0n} \\ b_{10}, b_{11}, \dots, b_{1n} \\ \dots \\ b_{n0}, b_{n1}, \dots, b_{nn} \end{bmatrix} \times \begin{bmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{bmatrix}; \quad (1.1)$$

$$X = \begin{bmatrix} x_1(t) \\ x_2(t) \\ \dots \\ x_n(t) \end{bmatrix}; \quad Y = \begin{bmatrix} y_1(t) \\ y_2(t) \\ \dots \\ y_n(t) \end{bmatrix}.$$

Для одновимірної поліноміальної моделі 3-го порядку вектор

$$X = \begin{bmatrix} 1 \\ x \\ x^2 \\ x^3 \end{bmatrix}.$$

Матриця В називається матрицею параметрів лінійного оператора A[•].
Тоді для перетворення векторів загального вигляду

$$Y = A [B, X] \quad (1.2)$$

матрична лінійна модель така:

$$Y = B \times X. \quad (1.3)$$

Оскільки нашим завданням є пошук матриці В параметрів моделі, то ідентифікація виконується за два етапи:

- 1) вибір гіпотези Н про тип досліджуваної системи і вигляду моделі;
- 2) побудова математичної моделі за результатами спостережень за вхідним X і вихідним Y впливами (визначення параметрів В).

З урахуванням впливу перешкод V множини X і Y можуть бути подані випадковими величинами. Рівняння регресії в цьому випадку визначиться як умовне математичне сподівання Y відносно X, а величина $M[Y/X] = \int y \cdot f(y/x) dx$

буде являти собою усереднену характеристику зв'язку X і Y і дозволить отримати найпростіший вигляд лінійної регресійної моделі:

$$y = a_0 + a_1x.$$

1.2. Оцінювання результатів моделювання

При моделюванні систем найважливішою частиною є процес оцінювання результатів дослідження. Оскільки реальні величини не підпорядковуються одній найпростішій функції або їхній сукупності, то при оцінюванні результатів експерименту вигідно використати апроксимуючі поліноми, за допомогою яких можна описати цю функцію. Ці поліноми отримали назву "регресійні поліноми", а їх знаходження і дослідження - "регресійний аналіз". Як базова функція може використовуватися ряд Тейлора, що має скінченне число членів:

$$F(x) = F(a) + \frac{x-a}{1!} F'(a) + \frac{(x-a)^2}{2!} F''(a) + \dots + \frac{(x-a)^n}{n!} F^{(n)}(a).$$

При експериментальних дослідженнях відомий тільки ряд значень функції, але не відомі її частинні похідні, тому замість апроксимуючого полінома Тейлора в його початковому вигляді використовується аналогічне поліному Тейлора рівняння регресії

$$\hat{Y} = b_0 + \sum_{i=1}^k b_i x_i + \sum_{i,j=1}^k b_{i,j} x_i x_j + \sum_{i=1}^k b_{i,i} x_i^2 + \sum_{i,j,n=1}^k b_{i,j,k} x_i x_j x_n + \dots + \sum_{i=1}^k b_{i,i} x_i^k,$$

де k - число стовпців у матриці планування.

Для експериментального отримання лінійної регресійної моделі використовується факторний експеримент першого порядку:

$$\hat{Y} = b_0 + \sum_{i=1}^k b_i x_i.$$

У цьому випадку – $k+1$ невідомих коефіцієнтів b , а отже, буде достатньо $k+1$ дослідів. При визначенні коефіцієнтів регресії повинні виконуватися необхідні й достатні умови:

1. Результати вимірювань вихідної величини Y в N точках факторного простору є реалізацією нормально розподіленої величини.
2. Дисперсії реалізації y всіх точках факторного простору мають бути однакові, тобто дисперсія не залежить від абсолютного значення величини і від напряму обходу факторного простору.
3. Вхідні змінні (фактори) – це незалежні величини, які вимірюються з нескінченно малою помилкою відносно помилки вихідної величини.

Для визначення того, наскільки адекватна ця модель об'єкту, будемо використовувати критерії адекватності, про які піде мова в наступних підрозділах. При невиконанні умов необхідно розширювати кількість точок експерименту доти, доки умови не будуть виконані.

Одним із найпоширеніших методів визначення параметрів моделі за статистичними експериментальними даними є метод найменших квадратів.

1.3. Метод найменших квадратів при дослідженні лінійних парних моделей

Нехай є N пар спостережень (x_i, y_i) , причому x_i - фіксоване значення вхідної величини. Цьому набору відповідає деяке поле кореляції. Необхідно добрати лінію регресії вигляду $\hat{y} = \hat{a}_0 + \hat{a}_1 x$, яка б найкращим чином описувала поведінку об'єкта. Як уже зазначалося, через наявність похибки вимірювання спостережувані значення вихідної величини \check{Y}_i також будуть випадковими.

Для оцінювання коефіцієнтів регресії a_0 і a_1 використовують метод найменших квадратів (МНК), який дозволяє мінімізувати суму квадратів різниці відхилень експериментальних даних \check{Y}_i і розрахункових значень, отриманих на основі рівняння регресії $\hat{y} = \hat{a}_0 + \hat{a}_1 x$.

Ідея МНК полягає в мінімізації функції

$$Q = \sum_{i=1}^N (\tilde{y}_i - y_i) \rightarrow \min. \quad (1.4)$$

Для лінійної парної залежності маємо

$$Q = \sum_{i=1}^N [\tilde{y}_i - (a_0 + a_1 x_{i1})] \rightarrow \min. \quad (1.5)$$

При знаходженні коефіцієнта необхідно взяти частинні похідні $\frac{\partial Q}{\partial a_i}$ і

прирівняти їх до нуля.

Отримаємо систему рівнянь, що називається системою нормальних рівнянь. Число цих рівнянь відповідає числу невідомих. Введемо фіктивну змінну $x_0 = 1$ і помножимо на неї a_0 . Одержимо таку систему:

$$\frac{\partial Q}{\partial b_0} = -2 \sum_{i=1}^N (\tilde{y}_i - b_0 x_{i0} - b_1 x_{i1}) x_{i0} = 0, \quad (1.6)$$

$$\frac{\partial Q}{\partial b_1} = -2 \sum_{i=1}^N (\tilde{y}_i - b_0 x_{i0} - b_1 x_{i1}) x_{i1} = 0. \quad (1.7)$$

Оцінки коефіцієнтів регресії, що задовольняють МНК, можна знайти із співвідношень

$$b_0 = A_0 / A_i; \quad b_1 = A_1 / A_i,$$

де A – головний визначник системи (за умови, що x_0 - фіктивна змінна, що тотожно дорівнює одиниці),

$$A = \begin{vmatrix} N & \sum_{i=1}^N x_{i1} \\ \sum_{i=1}^N x_{i1} & \sum_{i=1}^N x_{i1}^2 \end{vmatrix} = N \sum_{i=1}^N x_{i1}^2 - \left(\sum_{i=1}^N x_{i1} \right)^2; \quad (1.8)$$

A_i – доповнення;

$$A_0 = \begin{vmatrix} \sum_{i=1}^N \tilde{y}_i & \sum_{i=1}^N x_{i1} \\ \sum_{i=1}^N \tilde{y}_i x_{i1} & \sum_{i=1}^N x_{i1}^2 \end{vmatrix} = \sum_{i=1}^N \tilde{y}_i \sum_{i=1}^N x_{i1}^2 - \sum_{i=1}^N x_{i1} \tilde{y}_i \sum_{i=1}^N x_{i1};$$

$$A_1 = \begin{vmatrix} N & \sum_{i=1}^N \tilde{y}_i \\ \sum_{i=1}^N x_{i1} & \sum_{i=1}^N \tilde{y}_i x_{i1} \end{vmatrix} = N \sum_{i=1}^N \tilde{y}_i x_{i1}^2 - \sum_{i=1}^N x_{i1} \sum_{i=1}^N \tilde{y}_i.$$

Отже,

$$b_0 = \frac{\sum_{i=1}^N \tilde{y}_i \sum_{i=1}^N x_{i1}^2 - \sum_{i=1}^N x_{i1} \tilde{y}_i \sum_{i=1}^N x_{i1}}{N \sum_{i=1}^N x_{i1}^2 - \left(\sum_{i=1}^N x_{i1} \right)^2},$$

$$b_1 = \frac{N \sum_{i=1}^N \tilde{y}_i x_{i1} - \sum_{i=1}^N x_{i1} \sum_{i=1}^N \tilde{y}_i}{N \sum_{i=1}^N x_{i1}^2 - \left(\sum_{i=1}^N x_{i1} \right)^2}.$$

1.4. Метод найменших квадратів при дослідженні лінійних моделей з множинною залежністю

У реальних об'єктах дослідження вихідна величина залежить від декількох вхідних факторів; в цьому випадку говорять про регресійну поверхню. Тоді вираз (1.5) можна записати так:

$$\sum_{i=1}^n \left\{ \begin{array}{c} y_{i1} \\ y_{i2} \\ \dots \\ y_{in} \end{array} \right\} - \begin{array}{c} b_{11}, b_{12}, \dots, b_{1n} \\ b_{21}, b_{22}, \dots, b_{2n} \\ \dots \\ b_{n1}, b_{n2}, \dots, b_{nn} \end{array} \times \begin{array}{c} 1 \\ x_{i1} \\ \dots \\ x_{in} \end{array} \left\} \times \begin{array}{c} 1 \leftarrow x_0 \\ x_{i1} \\ \dots \\ x_{in} \end{array} \right\} = 0. \quad (1.9)$$

Системи (1.4) і (1.5) зручно подати у вигляді N-вимірного добутку векторів (1.10). Використовуючи позначення лінійного оператора (1.3), останній вираз можна записати в скороченій формі:

$$\sum_{i=1}^N \{Y - B \times X\} \times X = 0. \quad (1.10)$$

Виходячи з властивостей системи нормальних рівнянь, на основі якої визначаються оцінки коефіцієнтів рівнянь регресії, для n факторів експерименту можна записати:

$$b_0 \sum_{i=1}^N x_{i0}^2 + b_1 \sum_{i=1}^N x_{i0} x_{i1} + \dots + b_1 \sum_{i=1}^N x_{i0} x_{ij} + \dots + b_n \sum_{i=1}^N x_{i0} x_{in} = \sum_{i=1}^N \tilde{y}_i x_{i0};$$

$$b_0 \sum_{i=1}^N x_{i0} x_{i1} + b_1 \sum_{i=1}^N x_{i1}^2 + \dots + b_j \sum_{i=1}^N x_{i1} x_{ij} + \dots + b_n \sum_{i=1}^N x_{i1} x_{in} = \sum_{i=1}^N \tilde{y}_i x_{i1};$$

$$b_0 \sum_{i=1}^N x_{i0} x_{in} + b_1 \sum_{i=1}^N x_{i1} x_{in} + \dots + b_j \sum_{i=1}^N x_{in} x_{ij} + \dots + b_n \sum_{i=1}^N x_{in}^2 = \sum_{i=1}^N \tilde{y}_i x_{in}.$$

Розв'язання цієї системи рівнянь дозволяє отримати значення оцінок коефіцієнтів b_j методом найменших квадратів.

Скористаємося матричною формою запису:

$$X = \begin{pmatrix} x_{10}, x_{11}, \dots, x_{1j}, \dots, x_{1n} \\ x_{20}, x_{21}, \dots, x_{2j}, \dots, x_{2n} \\ \dots \dots \dots \\ x_{n0}, x_{n1}, \dots, x_{nj}, \dots, x_{nn} \end{pmatrix}.$$

Інформаційна матриця $C = X \cdot X^T$ в цьому випадку буде квадратною з розмірністю $(n+1) \cdot (n+1)$:

$$C = \begin{pmatrix} \sum_{i=1}^N x_{i0}^2, \sum_{i=1}^N x_{i0} x_{i1}, \dots, \sum_{i=1}^N x_{i0} x_{ij}, \dots, \sum_{i=1}^N x_{i0} x_{in} \\ \sum_{i=1}^N x_{i0} x_{i1}, \sum_{i=1}^N x_{i1}^2, \dots, \sum_{i=1}^N x_{i1} x_{ij}, \dots, \sum_{i=1}^N x_{i1} x_{in} \\ \dots \dots \dots \\ \sum_{i=1}^N x_{i0} x_{in}, \sum_{i=1}^N x_{i1} x_{in}, \dots, \sum_{i=1}^N x_{ij} x_{in}, \dots, \sum_{i=1}^N x_{in}^2 \end{pmatrix}. \quad (1.11)$$

Для визначення матриці В оцінок МНК необхідно вираз (1.3) помножити на транспонований вектор X^T і обернену матрицю C^{-1} :

$$B \times C \times C^{-1} = \tilde{Y} \times X^t \times C^{-1},$$

$$B = Y \times C^{-1} \times X^t.$$

1.5. Однофакторні регресійні моделі в системі Statgraphics

Серед найпоширенішого програмного забезпечення ТПЕ можна виділити інтегроване середовище SAS, яке дозволяє працювати в різних операційних системах Unix, Linux, Windows, а також прикладні програми, призначені для роботи під Windows Statistica, Statgraphics.

У цій системі регресійний аналіз виконується з використанням прикладної програми Statgraphics під операційною системою Windows. Statgraphics містить спадне меню, а також набір блоків і робоче поле меню. Бібліотеки організовано у вигляді дочірніх вікон. Основне керування виконується функціональними клавішами або "мишею".

Порядок роботи в системі Statgraphics

1. Активізувати кнопку «Пуск» і вибрати із меню послідовність програми Statgraphics. Із підменю вибрати рядок sgwin - мітку, відповідну exe-файлу. При цьому відкривається вихідне вікно системи Statgraphics (рис. 1.3) і з'являється повідомлення: «Welcome to StatWizard. The StatWizard can help you select the appropriate Statgraphics Plus analysis for collecting and analyzing your data».

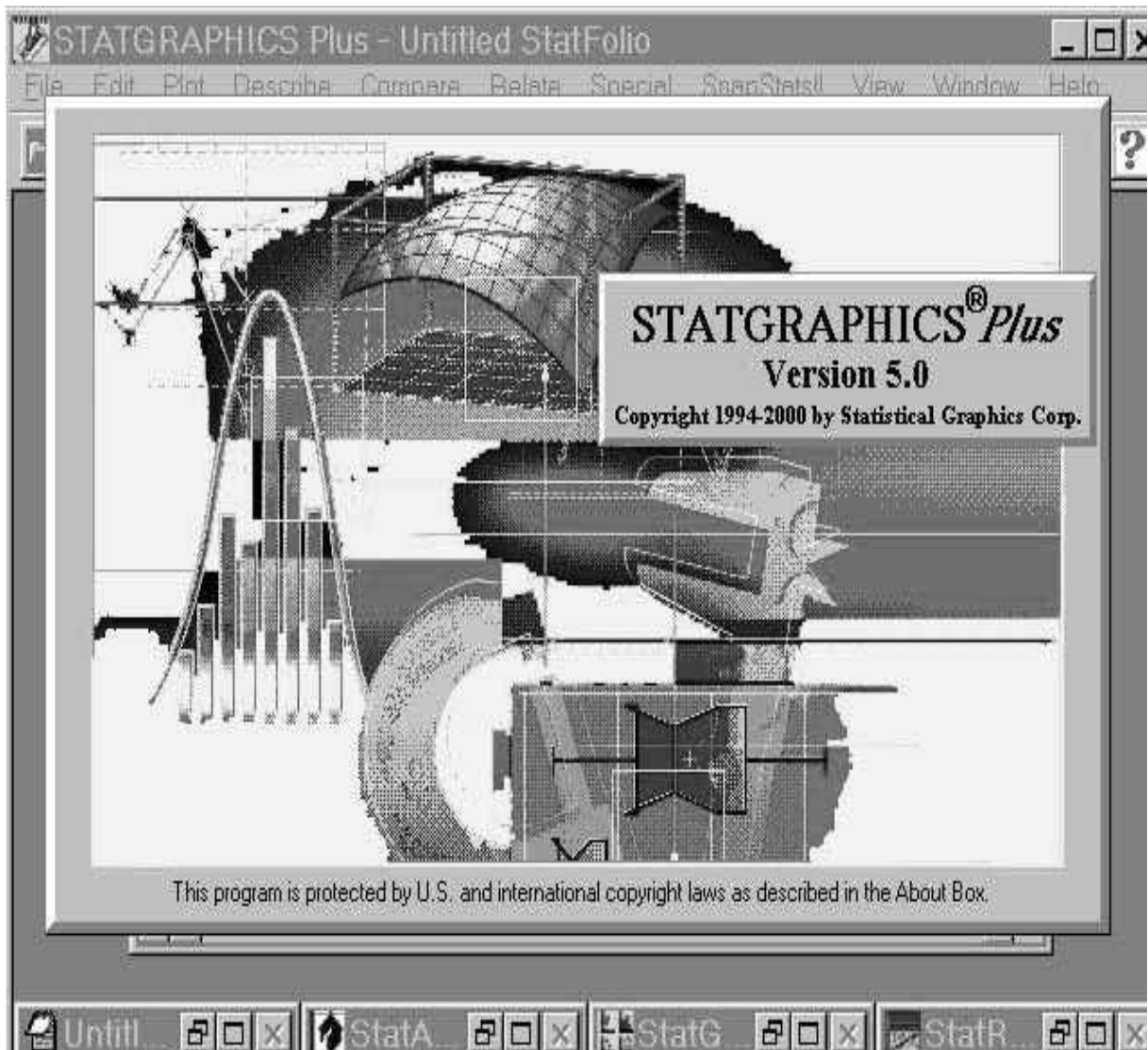


Рис. 1.3. Початкове вікно Statgraphics

Далі з'являється командне вікно (рис. 1.4), що дозволяє вибрати один із таких режимів:

- 1) аналіз існуючих даних;
- 2) побудова нового експерименту;
- 3) проведення аналізу без урахування даних.

Для розв'язання наших завдань необхідно вибрати опцію аналізу існуючих даних, що приводить до відкриття вікна введення даних (рис. 1.5).



Рис. 1.4. Вікно вибору режимів Statgraphics

2. Активізувати режим досліджень «Relate-Simpleregression» і отримати параметри моделі (рис. 1.6).
3. Одержати загальний вигляд моделі і діаграми до описаних об'єктів.
4. Вийти із оболонки Statgraphics з використанням команди Exit у запрошенні командного вікна.

Контрольні запитання

1. Подайте систему електроосвітлення у вигляді «чорного ящика».
2. Перелічіть основні математичні подання моделей.
3. Перелічіть основні типи моделей за характером процесів, що в них відбуваються.
4. Запишіть приклад поліноміального подання лінійної моделі розмірності 3-го порядку.

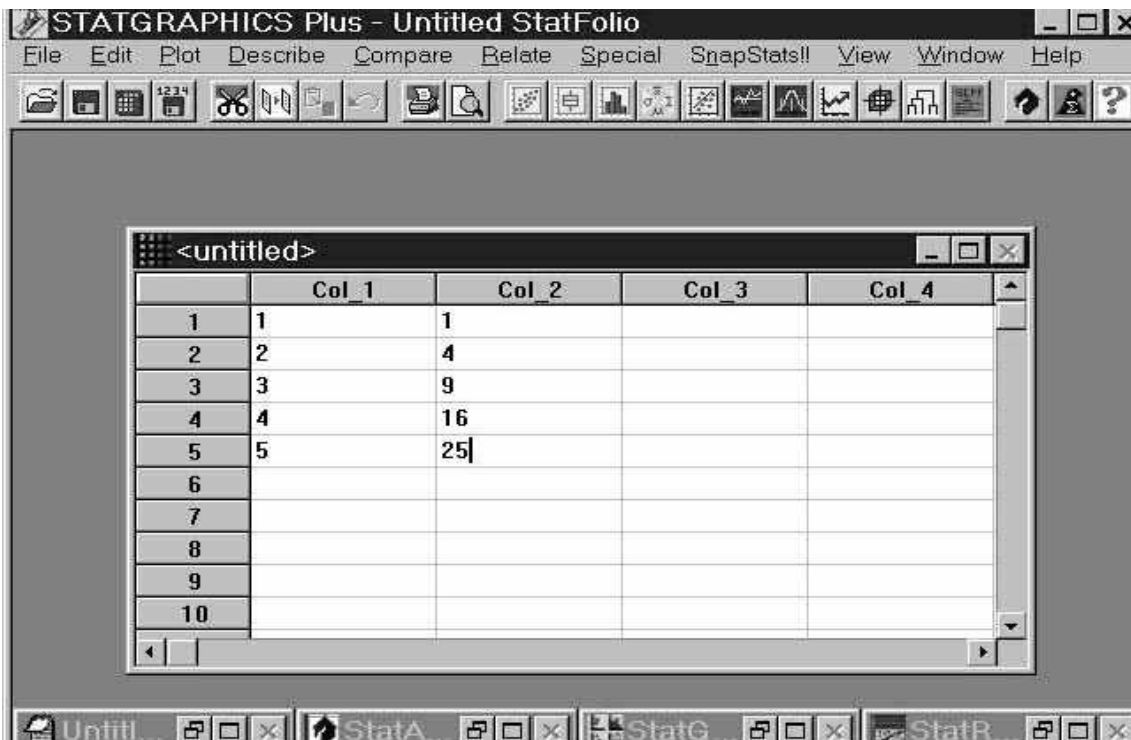


Рис. 1.5. Вікно введення даних Statgraphics

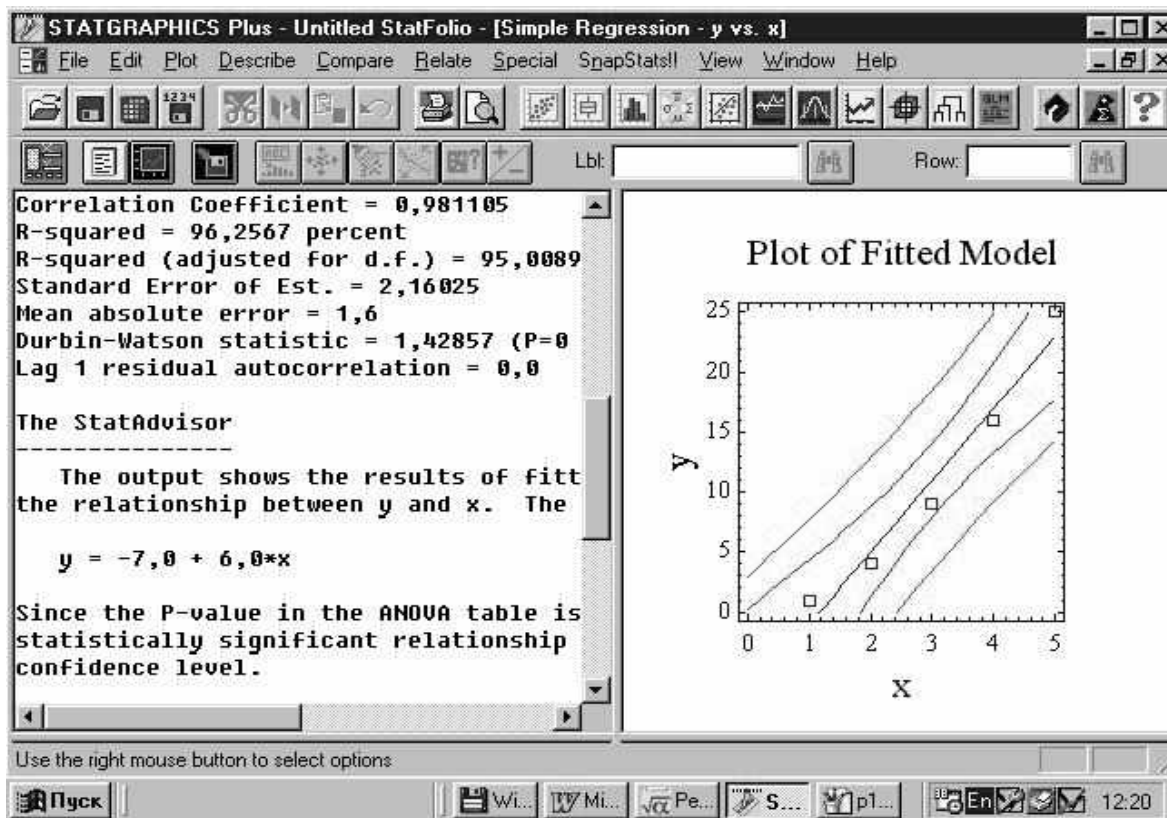


Рис. 1.6. Результат простого регресійного аналізу

5. Запишіть приклад поліноміального подання моделі 2-го порядку з урахуванням квадратичних факторів.
6. Векторно-операторне подання моделей загального вигляду.
7. Запишіть приклад векторно-матричного подання лінійних моделей.
8. Запишіть приклад базисної функції динамічної моделі 3-го порядку у вигляді ряду Тейлора.
9. Сформулюйте необхідні умови формування вхідних впливів для визначення параметрів моделі.
10. Перелічіть основні методи визначення параметрів моделі за вхідними і вихідними даними.
11. Визначте коефіцієнти регресії за відомим головним визначником і доповненнями.
12. Сформууйте матрицю вхідних впливів для системи 3-го порядку.
13. Запишіть інформаційну матрицю для системи 3-го порядку.
14. Отримайте значення параметрів поліноміальної моделі за відомими значеннями інформаційної матриці й векторів вхідних і вихідних впливів.
15. Перелічіть відоме програмне забезпечення математичної статистики і планування експериментів.
16. Сформууйте вектор вхідних впливів і матрицю для системи 4-го порядку.
17. Запишіть матрицю керування для системи 5-го порядку.
18. Запишіть приклад поліноміального подання моделі 4-го порядку з урахуванням квадратичних факторів.

2. МОДЕЛЮВАННЯ ЛІНІЙНИХ СИСТЕМ 2-ГО І ВИЩИХ ПОРЯДКІВ

2.1. Диференціальна і скінченно-різницева моделі динамічних систем

Розглянемо лінійну систему 2-го порядку, яка має такий загальний вигляд:

$$\begin{aligned}\dot{x}_1(t) &= a_{11} \cdot x_1(t) + b_1 U(t) + a_{12} \cdot x_2(t); \\ \dot{x}_2(t) &= a_{21} x_2(t) + b_2 U(t) + a_{22} x_1(t).\end{aligned}\tag{2.1}$$

За допомогою системи зв'язаних рівнянь відображуємо рівняння 2-го порядку:

$$\ddot{y} + k_1 \dot{y} + k_2 y = U(t) + R_1 \dot{U}(t) + R_2 \ddot{U}(t).\tag{2.2}$$

Необхідно вибрати підстановку у вигляді рівняння спостереження

$$y(t) = c_1 x_1(t) + c_2 x_2(t) + dU(t).\tag{2.3}$$

Тоді рівняння (2.2) можна записати як систему зв'язаних рівнянь.

Під час запису диференціального рівняння вигляду (2.2) вибирають початкові підстановки. Далі виконують запис диференціального рівняння з використанням цих підстановок так, щоби x_1 або x_2 були не вище за перший степінь. Там, де це не вдається, необхідно використовувати 2-гу підстановку, щоби з рівняння пішло не менш ніж два доданки. Рівняння переписують так, щоби перша похідна була зліва, а нульова – справа. При цьому перше рівняння виходить з диференціального, а друге відображує підстановку. Рівняння спостереження записують, виходячи з необхідного контролю системи і вибраної підстановки.

У випадку, якщо система – динамічна, вона може бути записана у вигляді n рівнянь 1-го порядку у формі Коші:

$$\begin{aligned}\dot{x}_1 &= a_{11} \cdot x_1 + a_{12} \cdot x_2 + \dots + b_1 \cdot U; \\ \dot{x}_2 &= a_{21} \cdot x_1 + a_{22} \cdot x_2 + \dots + b_2 \cdot U; \\ &\dots \dots \dots ; \\ \dot{x}_n &= a_{n1} \cdot x_1 + a_{n2} \cdot x_2 + \dots + b_n \cdot U.\end{aligned}\tag{2.4}$$

На ЕОМ інтеграція диференціальних рівнянь може бути виконана за допомогою різних числових методів. Якщо значення змінних стану x_1, x_2 і т. д. поставити у відповідність тимчасовим інтервалам з номерами $n, n+1$ і т. д., то після перетворення матриці A у B можна отримати:

$$x_1[n + 1] = b_{11} \cdot x_1[n] + b_{12} \cdot x_2[n] + b_{13} \cdot U[n];$$

$$x_2[n + 1] = b_{21} \cdot x_1[n] + b_{22} \cdot x_2[n] + b_{23} \cdot U[n].$$

Таким чином, система диференціальних рівнянь може бути описана у вигляді системи скінченно-різницевих рівнянь, які легко реалізуються на комп'ютері.

2.2. Побудова структурних схем моделей

Моделювання динамічних систем з використанням аналогових обчислювальних машин (АВМ) не потребує обов'язкового переведення в дискретну форму, оскільки містить динамічні елементи у своєму наборі.

З урахуванням цієї особливості будують більшість програмних пакетів об'єктного моделювання на ЕОМ. Візуальна вистава відповідає аналоговим компонентам. Реалізація кожного елемента-об'єкта – цифрова.

Інтегруюча ланка з використанням конденсатора C може бути подана так, як показано на рис. 2.1, а підсилювальна – на рис. 2.2.

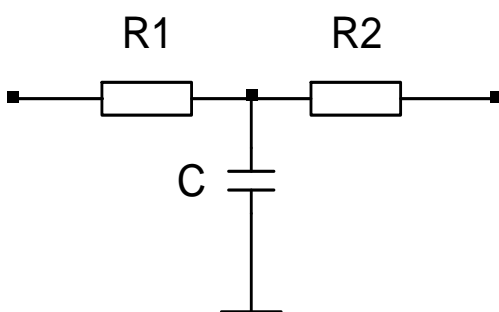


Рис. 2.1. Інтегруюча ланка

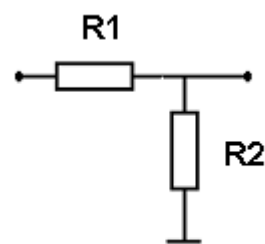


Рис. 2.2. Підсилювальна ланка

Для отримання ідеального інтегратора необхідно, щоб опір $R_2 \rightarrow \infty$. З цією метою використовується операційний підсилювач з коефіцієнтом підсилювання $k \rightarrow \infty$ з від'ємним оберненим зв'язком, в який включено конденсатор C (рис. 2.3).

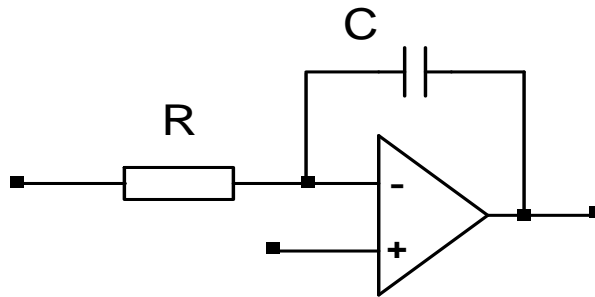


Рис. 2.3. Інтегруюча ланка з операційним підсилювачем

Для додання коефіцієнту посилення знака, а також множників $\times 10$ і $\times 100$ також використовується операційний підсилювач з оберненим зв'язком.

Для зміни знака (здобуття інвертора) застосовують параметри $k_y = -R_2 \times (1/R_1)$, $R_1 = R_2$, $k_y = -1$.

Множник $\times 10$ отримаємо при $R_2 = 10 \cdot R_1$.

Розглянемо відображення елементів АВМ на схемах (рис. 2.4). Для того, щоби прослідкувати за реакцією системи, необхідно вибрати точку спостереження.

Формула	АВМ	ТАУ
$y = 0.5 \cdot x$		
$y = \int x(t) dt$		
$y = -x$		

Рис. 2.4. Подання елементів

Для спостереження за перебігом процесу в точці спостереження підключають об'єкти «вольтметр» і «осцилограф». Для моделювання механічної системи (наприклад, коливанням 2-го порядку) складемо рівняння

$$m\ddot{y} + k_{\partial}\dot{y} + k_{yup} = 0.$$

Зведення системи до вигляду Коші дає систему двох зв'язаних систем 1-го порядку, аналогічних (2.2), але коефіцієнти матимуть інше значення:

$$K_1 = -\frac{k_{yup}}{m};$$

$$K_2 = -\frac{k_{\partial}}{m}.$$

Таким чином, механічна коливальна система буде еквівалентна, а значить, реалізовується на електричній системі або АВМ. Іншими словами, отримано заміщення механічної системи електричною.

З урахуванням сказаного структурну схему для перших двох рівнянь системи (2.4) показано на рис. 2.5.

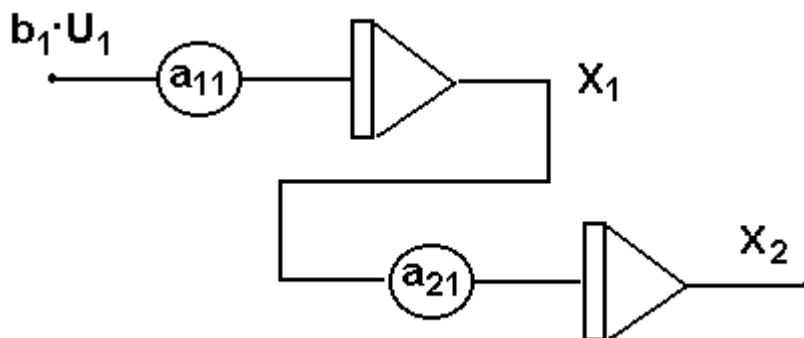


Рис. 2.5. Фрагмент структурної схеми багатовимірної системи

На рис. 2.6 подано схему моделювання цієї системи в програмному забезпеченні SIAM, а також результати моделювання для сигналів X_1, X_2 .

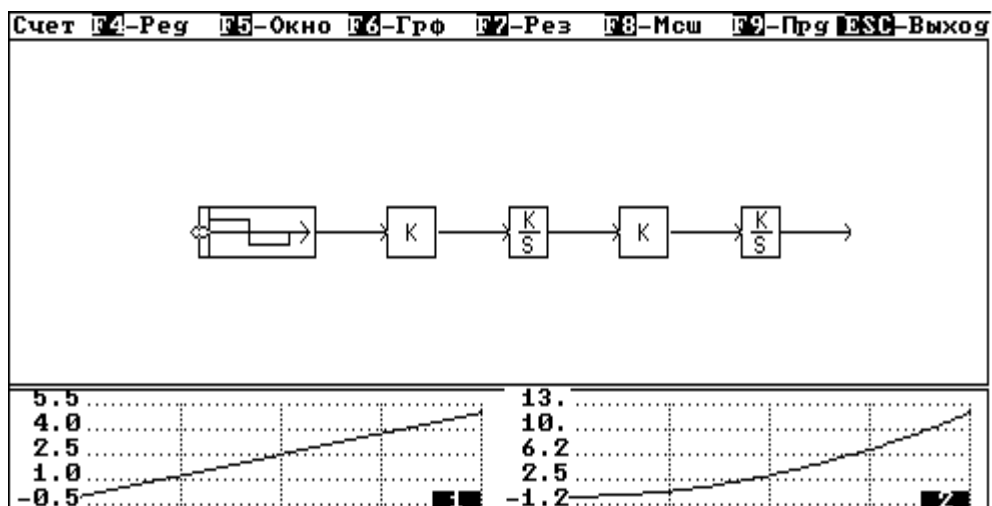


Рис. 2.6. Схема і результати моделювання для сигналів X_1 , X_2

2.3. Розроблення практичного алгоритму моделювання вимірювача висоти

Вимірювач висоти є складним пристроєм, моделі якого спрощено можна подати за допомогою системи другого порядку. Суть розробленого алгоритму полягає в такому (рис. 2.7):

1. Триває налаштування графічних областей, виведення зображень індикаторів і стрілок у нульовому положенні.

2. Необхідно ввести значення швидкопідйомності DVN .

3. Виконують лінійну зміну значення швидкопідйомності від попереднього значення DVN до заданого значення $DVN3$. Якщо швидкопідйомність збільшується, то поточне її значення DV визначають за типовою формулою.

Якщо швидкопідйомність зменшується, то поточне її значення DV визначають за відповідною формулою.

4. Виконують перетворення поточного значення швидкопідйомності DV у кут повороту стрілки варіометра $angle3$ з урахуванням нелінійної залежності кута повороту стрілки від значення швидкопідйомності.

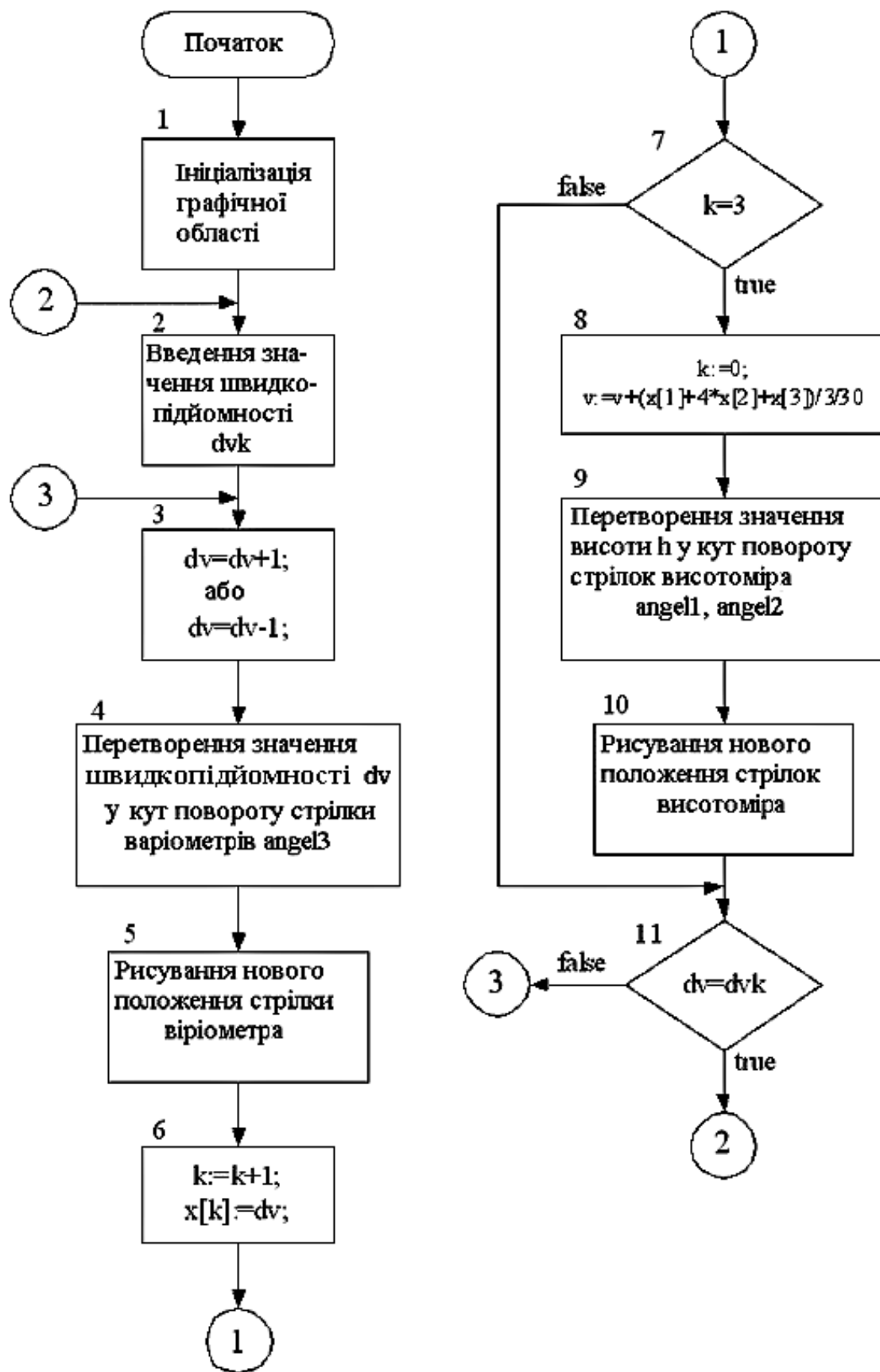


Рис. 2.7. Алгоритм моделювання

5. Здійснюють обчислення поточних координат точок стрілки варіометра. Далі видаляють колишнє зображення стрілки і на підставі певних координат рисують нове зображення стрілки.

6. Виконують формування масиву x , яке містить три послідовних значення швидкопідйомності .

7. Якщо отримано 3 послідовних значення швидкопідйомності, то виконують п. 8 , в іншому випадку – п. 1.

8. Стоп.

Обчислення візуальних компонентів здійснюють у такій послідовності:

1. Виконують обчислення поточного значення висоти V . Висота зв'язана зі швидкопідйомністю пропорційною залежністю.

Виконання інтегрування проводять з використанням числових методів. Для цього використовують формулу Сімпсона, яка дозволяє визначити числове значення інтеграла за трьома точками. Тоді поточне значення висоти визначають за відповідною формулою.

2. Здійснюють перетворення поточного значення висоти V у кут повороту стрілок висотоміра $angel1$, $angel2$.

3. Виконують обчислення поточних координат точок стрілок висотоміра. Далі видаляють старе зображення стрілок і на підставі певних координат рисують нове зображення стрілок.

4. Якщо досягнуто задане значення швидкопідйомності DVK , то виконують п. 2 , в іншому випадку - п. 3 .

Розроблений алгоритм дозволяє моделювати спільну роботу варіометра і барометричного висотоміра в діапазоні висот від 0 до 20000 метрів, діапазон швидкопідйомності від -400 м/с до 400 м/с, як показано на рис. 2.8. Закон зміни швидкопідйомності - лінійний.



Рис. 2.8. Результат моделювання

Реалізацію алгоритму на мові Делфі наведено нижче.

```

unit Unit1;
interface
uses
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, StdCtrls, ExtCtrls;
type
matr=array [1..5] of TPoint;
TForm1 = class(TForm)
Image1: TImage;
Label1: TLabel;
Button1: TButton;
    Edit1: TEdit;
    Label2: TLabel;
Timer1: TTimer;

```

```
procedure FormActivate(Sender: TObject);
procedure Button1Click(Sender: TObject);
procedure Timer1Timer(Sender: TObject);
procedure FormClose(Sender: TObject; var Action: TCloseAction);
```

```
private
```

```
    Private declarations
```

```
public
```

```
    Public declarations
```

```
end;
```

```
var
```

```
    Form1: TForm1; w,h:integer; ind:TBitmap;
    vn,vk,v,d,sch:integer;    Angle1,angle2:real;
    kordst:matr;
```

```
Const GtoR=57.29577951;
```

```
implementation
```

```
$R *.dfm
```

```
Procedure DrawSt1(angle:real); var
```

```
koord:matr;
```

```
begin          koord[1].X:=Round(260+17*sin((angle-28)/GtoR));
```

```
koord[2].X:=Round(260+190*sin((angle-2.4)/GtoR));
```

```
koord[3].X:=Round(260+220*sin((angle)/GtoR));
```

```
koord[4].X:=Round(260+190*sin((angle+2.4)/GtoR));
```

```
koord[5].X:=Round(260+17*sin((angle+28)/GtoR));
```

```
koord[1].Y:=Round(260-17*cos((angle-28)/GtoR));
```

```
koord[2].Y:=Round(260-190*cos((angle-2.4)/GtoR));
```

```
koord[3].Y:=Round(260-220*cos((angle)/GtoR));
```

```
koord[4].Y:=Round(260-190*cos((angle+2.4)/GtoR));
```

```
koord[5].Y:=Round(260-17*cos((angle+28)/GtoR));
```

```
Form1.Image1.Canvas.Polygon(koord);
```

```
end;
```

```
Procedure DrawSt2(angle:real); var
```

```
koord:matr;
```

```
begin
```

```

koord[1].X:=Round(260+17*sin((angle-28)/GToR));
koord[2].X:=Round(260+110*sin((angle-4.15)/GToR));
koord[3].X:=Round(260+140*sin((angle)/GToR));
koord[4].X:=Round(260+110*sin((angle+4.15)/GToR));
koord[5].X:=Round(260+17*sin((angle+28)/GToR));
koord[1].Y:=Round(260-17*cos((angle-28)/GToR));
koord[2].Y:=Round(260-110*cos((angle-4.15)/GToR));
koord[3].Y:=Round(260-140*cos((angle)/GToR));
koord[4].Y:=Round(260-110*cos((angle+4.15)/GToR));
koord[5].Y:=Round(260-17*cos((angle+28)/GToR));
Form1.Image1.Canvas.Polygon(koord);
end;

```

```

Procedure      Draw;      begin
Form1.Image1.Picture.Bitmap:=ind;
Form1.Image1.Canvas.Pen.Color:=167772
15;
Form1.Image1.Canvas.Brush.Color:=1677
7215; DrawSt1(angle1); DrawSt2(angle2);
end;

```

```

procedure TForm1.FormActivate(Sender: TObject); begin
kordst[1,1]:=15;
kordst[1,2]:=190;
kordst[1,3]:=220;
kordst[1,4]:=190;
kordst[1,5]:=15;
kordst[2,1]:=15;
kordst[2,2]:=110;
kordst[2,3]:=140;
kordst[2,4]:=110;
kordst[2,5]:=15;
ind:=Tbitmap.Create;
w:=Image1.Width;
h:=Image1.Height;
Ind.Width:=w;
Ind.Height:=h;      ind.LoadFromFile('ind.bmp');
vn:=0;
angle1:=0;

```



```

angle2:=0;
draw;
end;

procedure TForm1.Button1Click(Sender: TObject); begin
vk:=StrToInt(Edit1.Text);
d:=vk-vn;
v:=vn;
vn:=vk;
sch:=1;
Button1.Enabled:=false;
Timer1.Enabled:=true; end;

procedure TForm1.Timer1Timer(Sender: TObject); begin
if (sch<=(abs(d))) then begin
sch:=sch+1;
if d>=0 then begin
v:=v+1;
angle2:=v*360/20000; angle1:=(Trunc(angle2*1000) mod
18000)*20/1000; draw;
end
else begin
v:=v-1;
angle2:=v*360/20000;
angle1:=(Trunc(angle2*1000) mod 18000)*20/1000; draw;
end;
end
else begin
Timer1.Enabled:=false; Button1.Enabled:=true;
end;
end;

end;

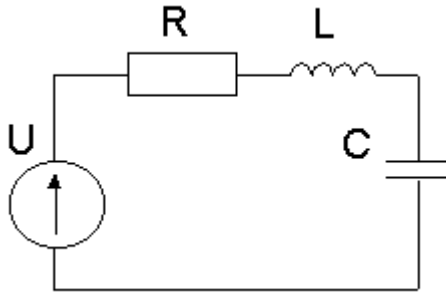
procedure TForm1.FormClose(Sender: TObject; var Action: TCloseAction);
begin
ind.Free;
end;

end.

```

3. МОДЕЛЮВАННЯ КОЛІВАЛЬНИХ СИСТЕМ У СИСТЕМІ MATLAB-SIMULINK

Розглянемо приклад електричної коливальної динамічної системи 2-го порядку (рис. 3.1):



$$U = i(t) \cdot r + L \frac{di(t)}{dt} + \frac{1}{c} \int i(t) dt;$$

$$L i'' + r i' + \frac{1}{c} i = U';$$

$$L i'' + \frac{r}{L} i' + \frac{1}{cL} i = 0.$$

Рис. 3.1. Система другого порядку

Виконавши підстановки

$$i = x_1, \quad i' = \dot{x}_1, \quad x_2 = \dot{x}_1, \quad \dot{x}_2 = \ddot{x}_1 = i'',$$

$$i = x_1, \quad i' = \dot{x}_1,$$

одержимо

$$\begin{cases} \dot{x}_2 + \frac{r}{L} x_2 + \frac{1}{cL} x_1 = 0, \\ \dot{x}_1 = x_2. \end{cases}$$

Запишемо останню систему в формі Коші:

$$\begin{cases} \dot{x}_1 = x_2, \\ \dot{x}_2 = -\frac{1}{cL} x_1 - \frac{r}{L} x_2. \end{cases} \quad (3.1)$$

При цьому

$$a_{11} = 0; \quad a_{12} = 1;$$

$$a_{21} = -\frac{1}{cL}; \quad a_{22} = -\frac{r}{L}.$$

На підставі рівняння (3.1), отриманого за описом електричного кола диференціального рівняння 2-го порядку, зробимо висновок про можливість побудови останнього на електричних елементах. Для його реалізації достатньо елемента R, який реалізує передавальний коефіцієнт, і двох елементів реактивних C, L, кожен з яких дає одну похідну або інтегрування.

Тоді структурна схема рівняння (3.1) з використанням умовних позначень АВМ буде мати такий вигляд, як показано на рис. 3.2.

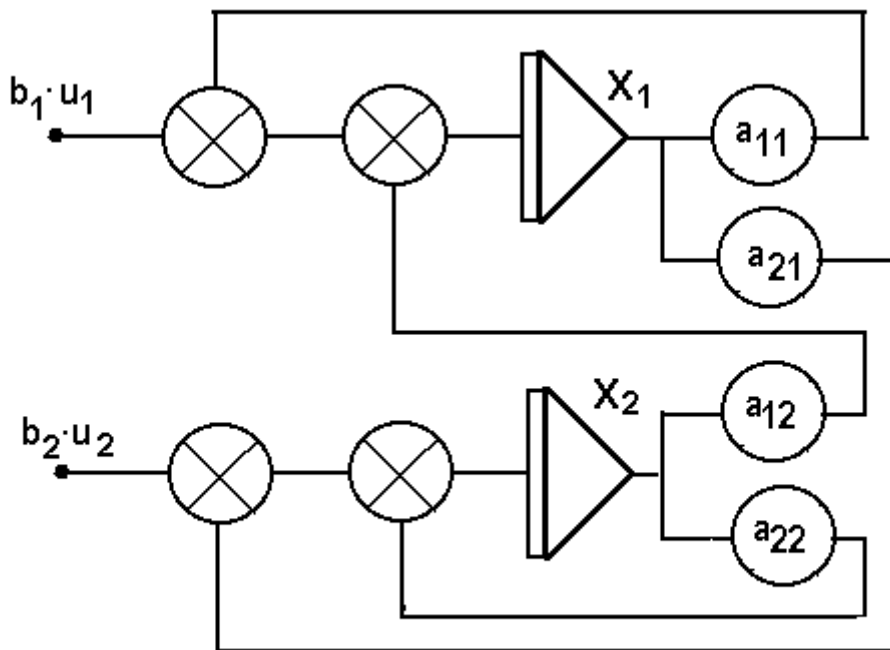


Рис. 3.2. Структурна схема системи (3.1)

Вираз (3.1) з урахуванням зовнішнього впливу можна записати так:

$$\begin{cases} \dot{x}_1 = x_2; \\ \dot{x}_2 = -\frac{1}{cL}x_1 - \frac{r}{L}x_2 + bU. \end{cases} \quad (3.2)$$

Для того щоби простежити за реакцією системи, необхідно вибрати точку спостереження.

Для спостереження за перебігом процесу в точці спостереження підключають об'єкти «вольтметр» і «осцилограф».

Програмні пакети об'єктного моделювання використовують аналогічне заміщення розробленими об'єктами.

Під об'єктно-орієнтованим моделюванням будемо розуміти підготовку окремих модулів, необхідну для конструювання моделі із скінченних об'єктів, що мають вхід / вихід і певну функцію.

При компонуванні модулів створюється база даних, яка називається робочою. У робочій області записують послідовність заголовків, кожен з вихідних параметрів якого є входом для наступного. Присвоєння змінних відбувається шляхом призначення зв'язків візуальних систем ООМ. Цей зв'язок - графічний. Координати входу дорівнюють координатам виходу. Цей зв'язок здійснюється при накладенні області виходу на область входу.

3.1. Моделювання коливальних систем у системі Matlab-simulink

Для моделювання коливальної системи з використанням модуля Matlab-simulink необхідно ввести команду simulink, що приводить до відкриття двох вікон untitled і Library: simulink.

З вікна Library: simulink слід активізувати елемент Sources, що приводить до відкриття вікна Library: simulink / Sources, з якого, в свою чергу, вибирають елемент, що є початковим у структурній схемі, наприклад Cojstant.

Вибраний блок необхідно перетягнути за допомогою "миші", утримуючи її праву кнопку, в набірне вікно.

Потім підключають послідовно дві інтегруючі ланки з лінійного елемента бібліотеки «осцилограф» (Scope), із замикальних елементів бібліотеки Sinks, а також множники Gain і суматори Sum, як показано на рис. 3.3.

Результати моделювання коливальної системи для різних параметрів показано на рис. 3.4 - 3.6, а для системи без оберненого зв'язку за швидкістю (рис. 3.7) - на рис. 3.8.

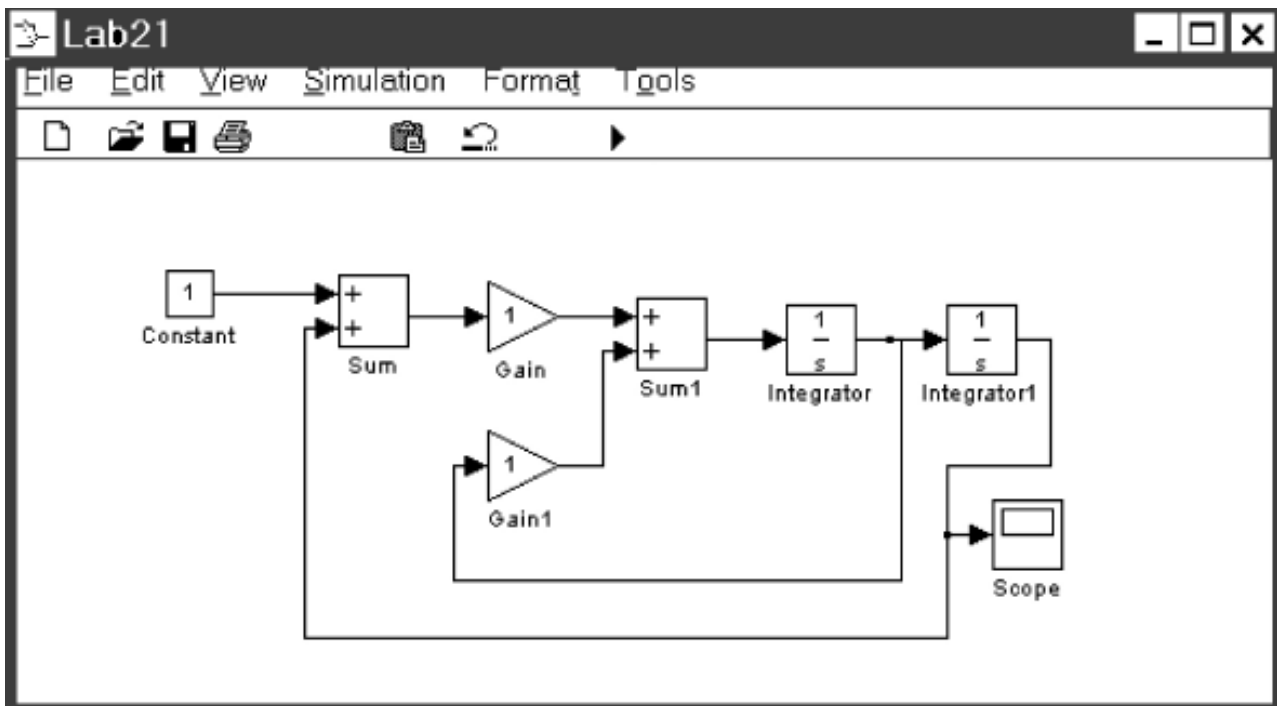


Рис. 3.3. Структурна схема коливальної системи

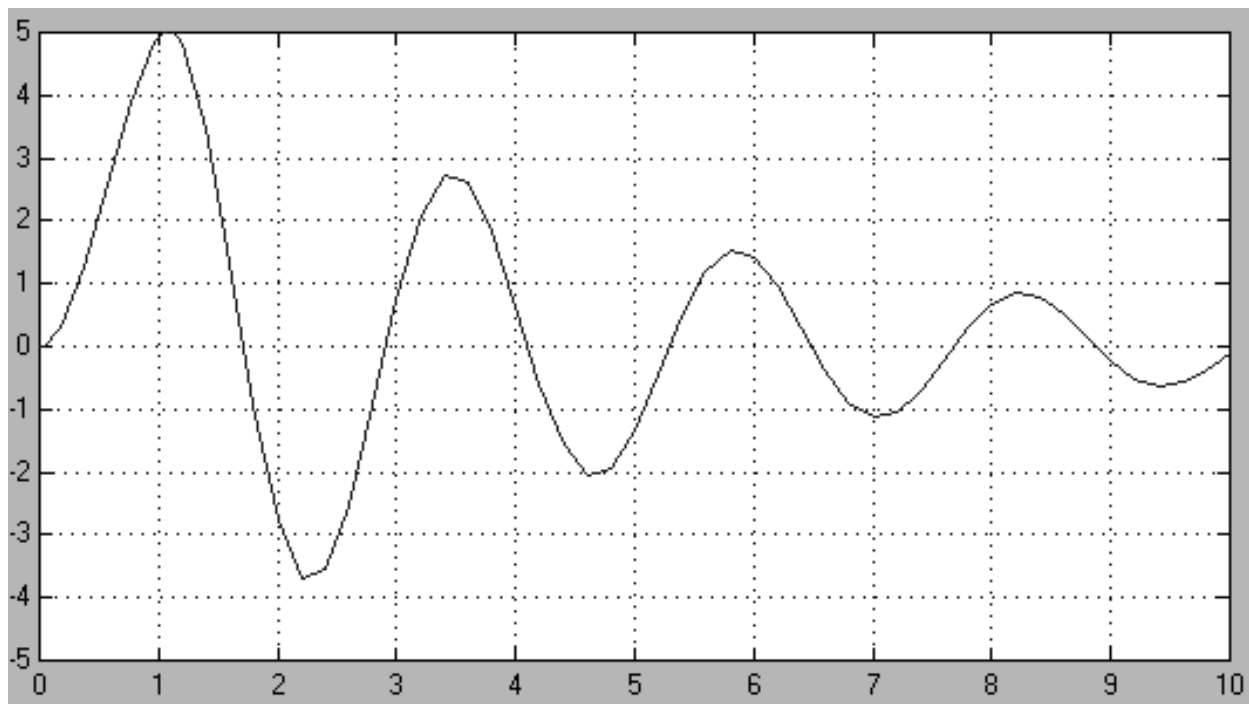


Рис. 3.4. Результати моделювання коливальної системи

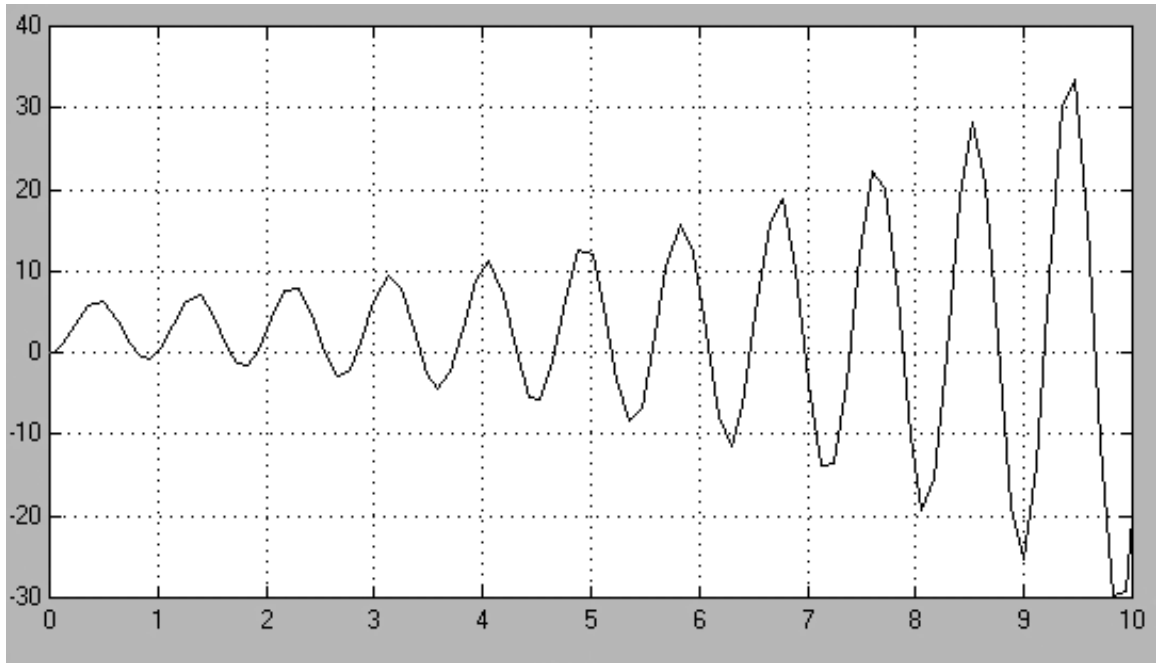


Рис. 3.5. Результати моделювання нестійкої коливальної системи

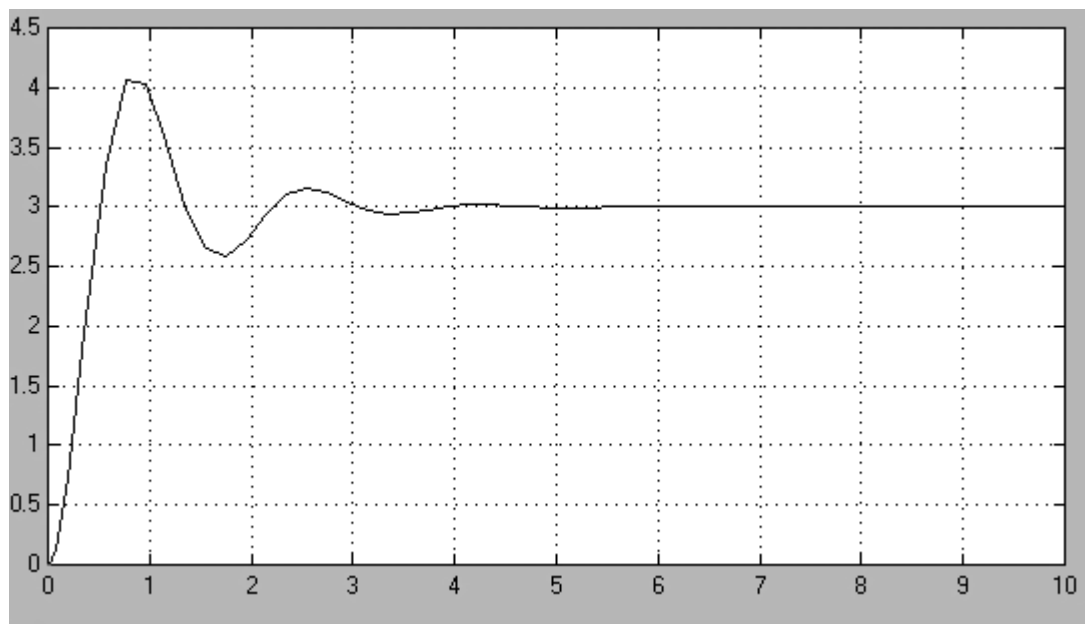


Рис. 3.6. Результати моделювання коливальної системи при $a_{12}=0.7$

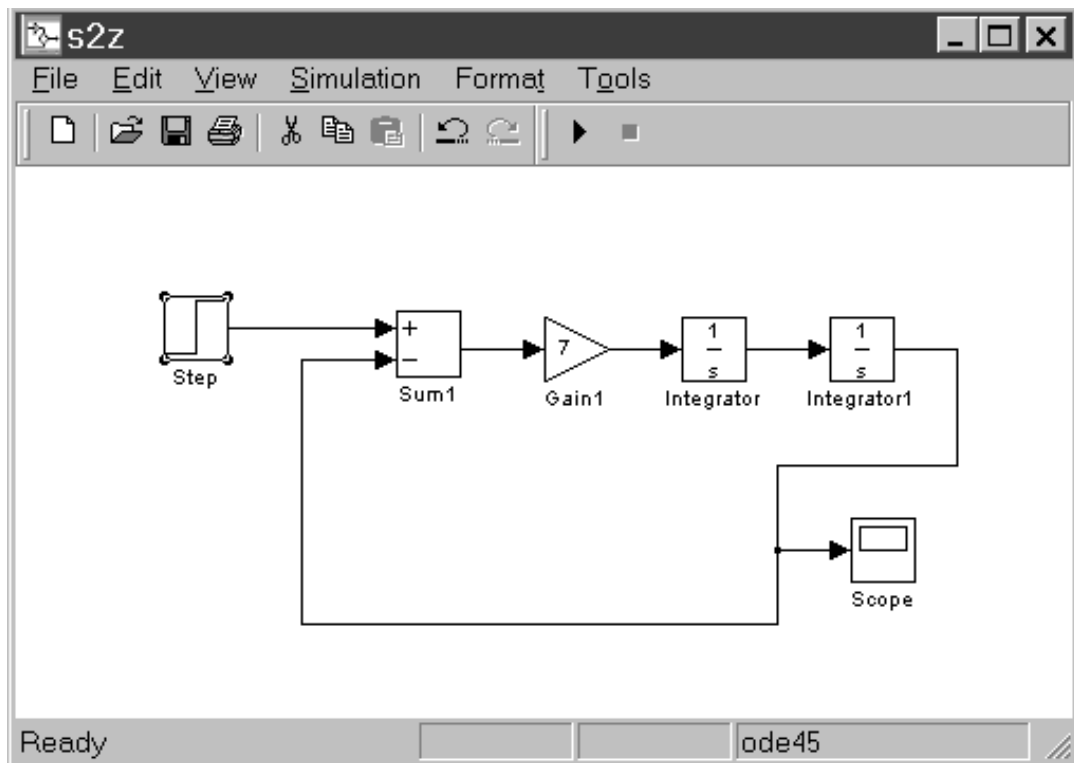


Рис. 3.7. Структурна схема коливальної системи без оберненого зв'язку за першою похідною

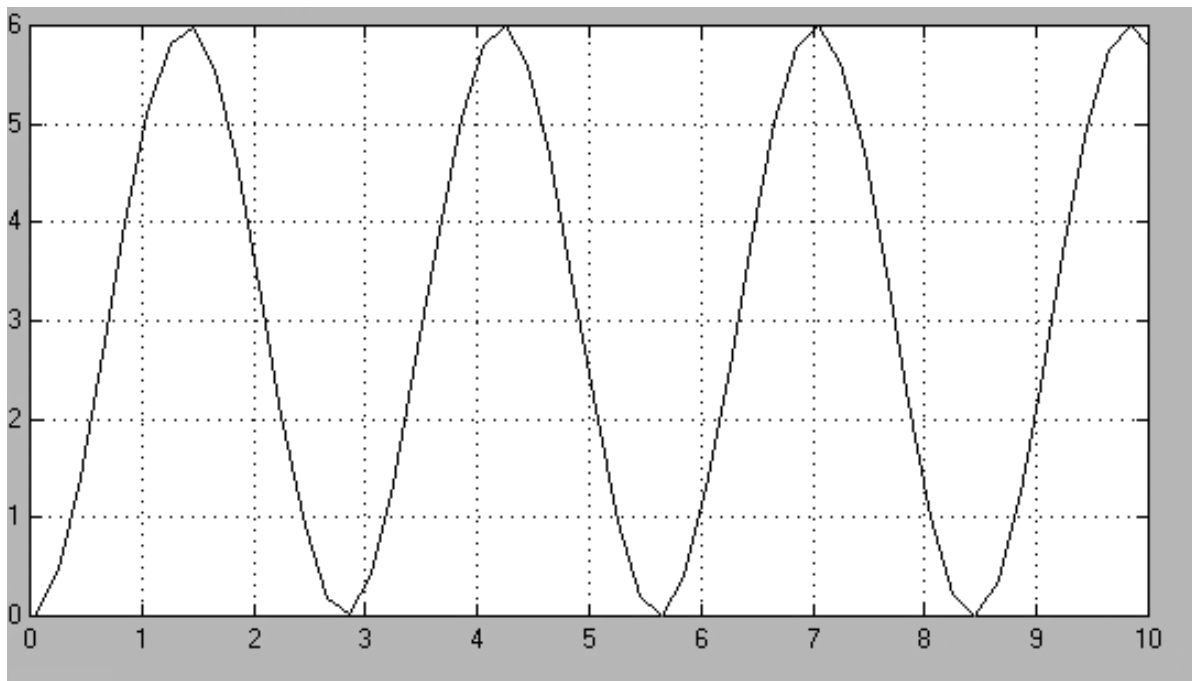


Рис. 3.8. Результати моделювання коливальної системи при $a_{12}=0$

Завдання до самопідготовки

Задача 3.1

Скласти структурну схему для системи $3x''' + 5x'' + 4x + 7 = u$. Спостереження проводити за відкликом $y = x + 3$.

Задача 3.2

Побудувати структурну схему в системі **Matlab-simulink** для рівняння $5x''' + 7x'' + 4 = 3u$.

Задача 3.3

Скласти структурну схему для системи $3x''' + 8x = u$. Спостереження проводити за відкликом $y = 4x + 7$.

Задача 3.4

Подати в канонічному вигляді систему $7x''' + x'' + 5x = u$.

Задача 3.5

Скласти структурну схему для системи $3x''' + 7x'' + 4x + 2 = u$. Спостереження проводити за відкликом $y = x + 1$.

Задача 3.7

Побудувати структурну схему в системі **Matlab-simulink** для рівняння $x''' + 2x'' + 11 = 3u$.

Задача 3.7

Подати в канонічному вигляді систему $9x''' + 4x = u + 1$.

Найпростішим прикладом (4.3) є рівняння

$$\begin{bmatrix} a_1 \\ a_2 \\ \dots \\ a_m \end{bmatrix} \cdot [x_1 \quad x_2 \quad \dots \quad x_n] = \begin{bmatrix} a_1 x_1 & a_1 x_2 & \dots & a_1 x_n \\ a_2 x_1 & a_2 x_2 & \dots & a_2 x_n \\ \dots & \dots & \dots & \dots \\ a_m x_1 & a_m x_2 & \dots & a_m x_n \end{bmatrix},$$

причому

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}^T,$$
$$\begin{bmatrix} a_1 \\ a_2 \end{bmatrix}^T = [a_1 \quad a_2].$$

4.1. Обернення матриць

Для виконання операції обернення скористаємося одиничною матрицею

$$A \cdot A^{-1} = I, \tag{4.4}$$

$$I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

$$\det A = a_{11} a_{22} - a_{12} a_{21},$$

або

$$A^{-1} = \frac{1}{\det A} \tilde{A}.$$

Доведемо, що

$$\begin{aligned} y_1(t) &= c_{11}x_1(t) + c_{12}x_2(t) + \dots + c_{1n}x_n(t) + d_1U(t); \\ y_2(t) &= c_{21}x_1(t) + c_{22}x_2(t) + \dots + c_{2n}x_n(t) + d_2U(t). \end{aligned} \quad (4.6)$$

Запишемо рівняння (4.6) у векторній формі (рівняння стану):

$$\dot{x} = Ax + BU.$$

Рівняння спостереження:

$$Y = Cx + DU.$$

При цьому

$$Y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}, \quad \text{тоді матриця } C = \begin{bmatrix} c_{11} & c_{12} & \dots & c_{1n} \\ c_{21} & c_{22} & \dots & c_{2n} \end{bmatrix},$$

$$D = \begin{bmatrix} d_{11} & d_{12} \\ d_{21} & d_{22} \end{bmatrix}, \quad U(t) = \begin{bmatrix} U_1(t) \\ U_2(t) \end{bmatrix}. \quad (4.7)$$

4.3. Моделювання структурних схем систем високого порядку

Структурну схему можна виконати спрощено, як показано на рис. 4.1, в цьому випадку вектор замінюється подвійною стрілкою.

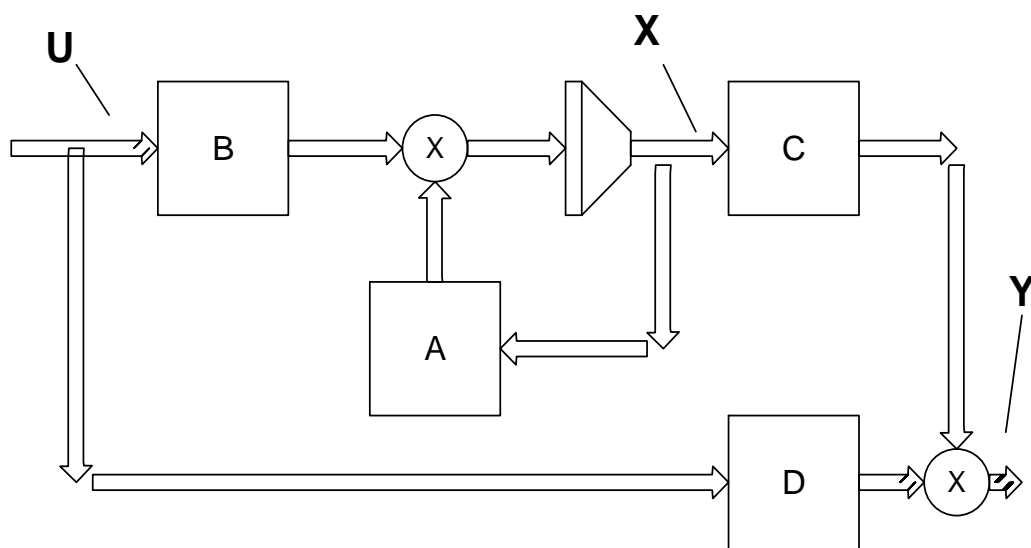


Рис. 4.1. Спрощений вигляд структурної схеми

Приклад 4.2

Розглянемо електричний ланцюг 3-го порядку (рис. 4.2).

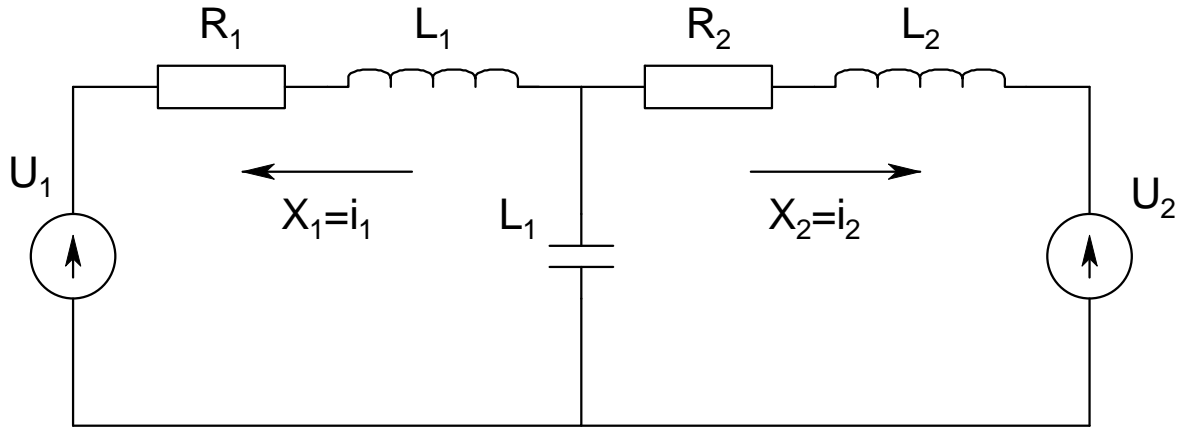


Рис. 4.2. Система 3-го порядку

Запишемо рівняння за законом Кірхгофа:

$$x_1 + x_2 + x_C = 0;$$

$$\dot{x}_3 = -x_1 - x_2;$$

$$\frac{1}{c} \int x_C dt + L \frac{dx_1}{dt} + R_1 x_1 = U_1;$$

$$L \frac{dx_2}{dt} + R_2 x_2 + \frac{1}{c} \int x_C dt = U_2.$$

Запишемо рівняння у формі Коші:

$$\begin{cases} \dot{x}_1 = -\frac{1}{LC}x_C - \frac{R}{L}x_1 + \frac{U_1}{L} & (x_C = \dot{x}_3); \\ \dot{x}_2 = \frac{U_2}{L} - \frac{R_2}{L}x_2 - \frac{1}{LC}x_C & (x_C = \dot{x}_3); \\ \dot{x}_3 = -x_1 - x_2 & (x_C = x_1 + x_2). \end{cases}$$

Виразуємо $\frac{1}{c} \int x_C dt = x_3$, рівняння одержимо в іншій формі:

$$\begin{cases} \dot{x}_1 = -\frac{1}{L_1}x_C - \frac{R_1}{L_1}x_1 + \frac{U_1}{L_1}; \\ \dot{x}_2 = \frac{U_2}{L_2} - \frac{R_2}{L_2}x_2 + \frac{1}{L_2}x_C; \\ \dot{x}_3 = -\frac{1}{C}x_1 - \frac{1}{C}x_2. \end{cases}$$

Матриці системи можна записати в такому вигляді:

$$\dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix}; \quad A = \begin{bmatrix} -(R_1+1)/L_1 & -1/L_1 & 0 \\ -1/L_2 & -(R_2+1)/L_2 & 0 \\ -1/c & -1/c & 0 \end{bmatrix};$$

$$B = \begin{bmatrix} 1/L & 0 \\ 0 & 1/L \\ 0 & 0 \end{bmatrix}; \quad U = \begin{bmatrix} U_1 \\ U_2 \\ 0 \end{bmatrix}.$$

Рівняння спостереження:

$$y = Cx + DU,$$

де

$$C = \begin{bmatrix} R_1 & 0 & 0 \\ 0 & R_2 & 0 \\ 1 & 1 & 0 \end{bmatrix}; \quad x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix};$$

$$y = \begin{bmatrix} x_1 R_1 \\ x_2 R_2 \\ x_1 + x_2 \end{bmatrix}.$$

Завдання 4.1

Зобразити структурну схему за допомогою таких рівнянь:

$$\begin{cases} \dot{x}_1 = -\frac{R_1}{L_1} x_1 - \frac{1}{L_1} x_3 + \frac{U_1}{L_1}; \\ \dot{x}_2 = -\frac{R_2}{L_2} x_2 - \frac{1}{L_2} x_3 + \frac{U_2}{L_2}; \\ \dot{x}_3 = -\frac{1}{c} x_1 - \frac{1}{c} x_2; \\ y_1 = x_1 R_1; \\ y_2 = x_2 R_2. \end{cases}$$

4.4. Системи високого порядку з одним входом і одним виходом

Існують складні системи, для яких багатовимірність вектора стану обумовлюється не кількістю вхідних сигналів, а їхніми степенями або похідними.

Розглянемо системи 3-го порядку:

$$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}; \quad x = \begin{bmatrix} x_1 \\ x_1^2 \\ x_1^3 \end{bmatrix}; \quad x = \begin{bmatrix} x_1 \\ \dot{x}_1 \\ \ddot{x}_1 \end{bmatrix}.$$

Розв'язок таких диференціальних рівнянь може бути отримано в аналітичному або дискретному вигляді, при цьому необхідно привести диференціальне рівняння до стандартного вигляду.

Для рівнянь (4.5) матриця буде містити такі елементи:

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Розглянемо приклад вихідного рівняння $y'' = U$.

Виконаємо такі підстановки:

$$\begin{aligned} |x_1 = y|, & \quad |x_2 = y'|; \\ x_3 = y''; \\ \dot{x}_1 = x_2; \\ \dot{x}_2 = x_3; \\ \dot{x}_3 = U. \end{aligned}$$

На підставі цього отримаємо

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}.$$

Нарисуємо структурну схему (рис. 4.3).

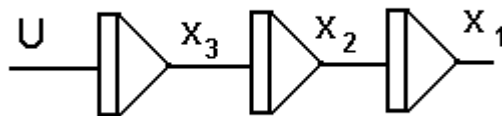


Рис. 4.3. Структурна схема системи високого порядку з одним входом і одним виходом

5. ГЕОМЕТРИЧНЕ МОДЕЛЮВАННЯ СТАНУ СИСТЕМ НА ФАЗОВІЙ ПОВЕРХНІ

Розглянемо систему другого порядку (2.1), для якої вектор стану має вигляд

$$x(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}. \quad (5.1)$$

Кожна точка векторного простору може бути описана за допомогою двох координат $x_1(t)$, $x_2(t)$. Ці координати можна розглядати як декартові координати двовимірного метричного простору. Багатовимірний вектор стану x можна подати у вигляді координат багатовимірного простору, який називається простором стану. Двовимірний простір стану називають *фазовою площиною*.

Зобразимо фазову площину (рис. 5.1).

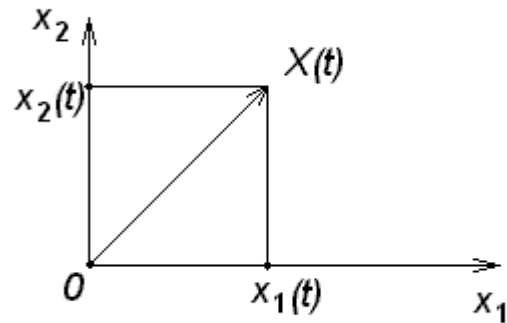


Рис. 5.1. Фазова площина

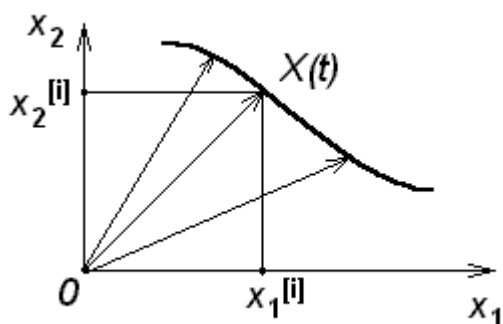
5.1. Моделювання траєкторії руху динамічної системи

При зміні значень вектора стану за його скінченними координатами можна побудувати таблицю значень, а сукупність точок, побудованих за цією таблицею, називається *траєкторією руху динамічної системи* (рис. 5.2).

Напрямок руху траєкторії можна визначити, обчисливши кутовий коефіцієнт дотичної в точці i , яка нас цікавить:

$$k^{[i]} = \frac{\Delta x_2^{[i]}}{\Delta x_1^{[i]}} = \frac{x_2^{[i]} - x_2^{[i-1]}}{x_1^{[i]} - x_1^{[i-1]}}.$$

5.2. Моделювання траєкторій руху нелінійних систем



Розглянемо систему, яка описує керування об'єктом за допомогою релейної функції U :

$$\begin{aligned}\dot{x} &= Ax + BU; \\ y &= Cx.\end{aligned}$$

Рис. 5.2. Траєкторія системи

Опишемо цю систему у вигляді такого набору коефіцієнтів:

$$A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}; \quad B = [0 \quad 1]^T; \quad c = [1 \quad 0].$$

Розпишемо цю систему за допомогою заданих коефіцієнтів:

$$\begin{cases} \dot{x}_1(t) = x_2(t); \\ \dot{x}_2(t) = U(t); \end{cases} \quad (5.2)$$

Структурну схему такої системи подано на рис. 5.3.

Розглянемо два варіанти, коли керування додатне або від'ємне, і порівняємо їх.

Для початку одержимо аналітичний розв'язок рівняння траєкторії.

Подамо $x_1 = f(x_2)$.

Проінтегруємо систему (5.2):

$$\begin{cases} x_1 = x_2 t; \\ x_1 = U \frac{t^2}{2}. \end{cases} \quad (5.3)$$

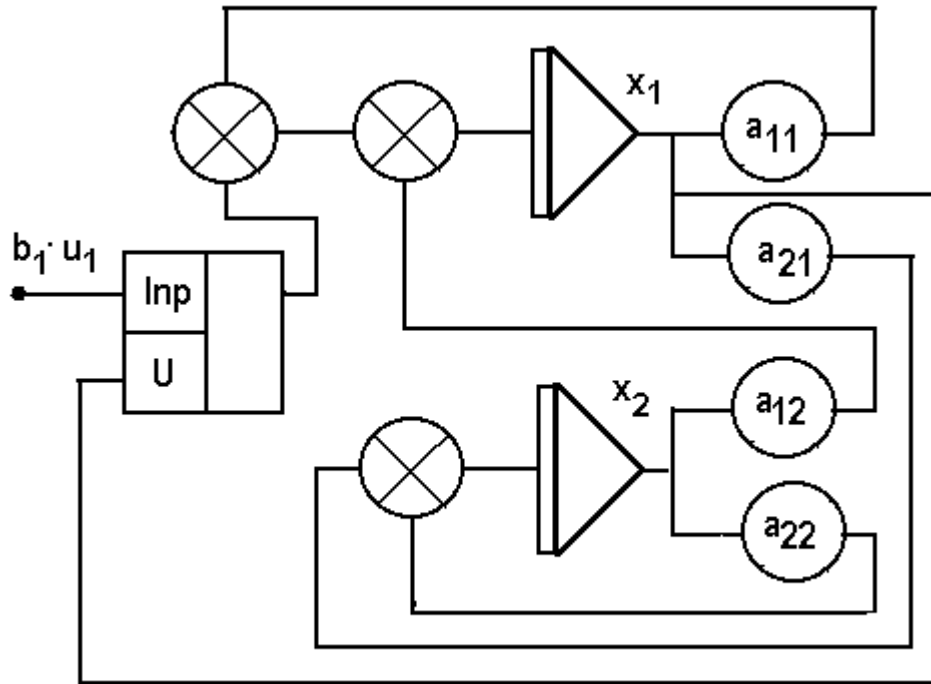


Рис. 5.3. Структурна схема нелінійної системи

Ця система може бути інтерпретована у вигляді механічного об'єкта.

Задача 5.1

Виконати дослідження системи лінійного руху: X_1 – координата; X_2 – швидкість; U – прискорення. При цьому відстань $D = \frac{at^2}{2}$.

Система кутових координат: X_1 – кут; X_2 – кутова швидкість.

Перше прискорення – $\varphi = \omega \cdot t$, друге – $\varphi = \left(\frac{M}{I}\right) \frac{t^2}{2}$.

Отримаємо систему

$$\begin{cases} \phi = \omega \cdot t; \\ \phi = \frac{M}{I} \frac{t^2}{2}. \end{cases} \quad (5.4)$$

Виразимо x_1 через x_2 :

$$x_1 = U \cdot x_1^2 / 2x_2^2;$$

$$x_1 = \frac{2x_2^2}{U}.$$

Таким чином, для описаної системи траєкторія буде являти собою параболу.

При різних початкових умовах з'являється сукупність кривих

$$x_1 = f(x_2, G_0^{[i]}),$$

які відрізняються на $const G_0^{[i]}$:

$$G_0^{[1]} = \begin{bmatrix} x_{01} \\ x_{02} \end{bmatrix}.$$

Для підтримки стійкості необхідно ввести рівняння керування.

Зведена система:

$$U = \begin{cases} -U_0 & \text{при } \phi > 0, \\ +U_0 & \text{при } \phi < 0. \end{cases}$$

5.2.1. Лінійний закон керування

Розглянемо лінійну функцію керування $U = -kt$.

Станом рівноваги на фазовій площині називається такий стан x_e , при

якому $x_e = \begin{bmatrix} x_{1e} \\ x_{2e} \end{bmatrix}$, і для будь-яких зовнішніх впливів $U=U_0$ у вигляді будь-

якого часу t ($0 \leq t < \infty$) система зберігає нульові значення координат, тобто

$$\dot{x}_1 = 0;$$

$$\dot{x}_2 = 0.$$

Стан рівноваги є нульовою точкою і належить до сім'ї особливих, або критичних, точок.

Стан рівноваги x_e може бути траєкторією, але при цьому траєкторія системи ніколи не перетнеться з траєкторією рівноваги, а може тільки підходити до неї як завгодно близько.

5.3. Моделювання збіжності системи на фазовій площині

Для забезпечення компенсації похибок у реальних системах необхідно використовувати складні закони керування, які враховують як сигнал відхилення, так і похідні. Дослідження таких законів зручно проводити на фазовій площині. Для законів керування, що містять похідні

$$U(t) = -k_1\varphi + k_2\dot{\varphi},$$

можна побудувати лінію перемикавання, кут якої визначається співвідношенням коефіцієнтів k_1, k_2 .

Якщо в закон керування подати похідну з протилежним знаком, то запізнювання збільшується.

У цьому випадку така траєкторія називається фокусом. Фокус може бути збіжним і розбіжним. Сходження системи здійснюється по траєкторії типу «вузол».

Для дискретних законів керування, наприклад $U(t) = -x$ при $x > x_0$, можна побудувати лінію перемикавання, зміщення якої визначається співвідношенням величин x і x_0 .

5.3.1. Точки траєкторії системи

Усі точки, що знаходяться на траєкторії, називаються регулярними. Особливі точки системи можуть бути ізольованими (нерегулярними).

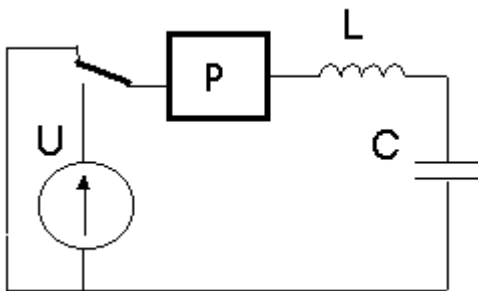
При входженні траєкторії в траєкторію рівноваги існує кут входження, який для евклідового простору відповідно до (5.2) і (5.3) може визначатися як напрямний косинус.

Приклад 5.1. Моделювання нелінійних систем у *matlab-simulink*

Розв'язання нелінійних систем, приклад яких наведено на рис. 5.4, найчастіше здійснюється шляхом лінеаризації або умовного перемикання лінійних ділянок:

$$i_D = I_0 [e^{\frac{qU_0}{kT}} - 1], \quad \dot{x} = \frac{1}{c} f(x(t) + U(t)).$$

Розглянемо приклад релейної динамічної системи 2-го порядку (див. рис. 5.4):



$$U = i(t) \cdot r + L \frac{di(t)}{dt} + \frac{1}{c} \int i(t) dt;$$

$$Li'' + ri' + \frac{1}{c}i = U';$$

$$i'' + \frac{r}{L}i' + \frac{1}{cL}i = \frac{U'}{L};$$

$$U = \begin{cases} 0 & \text{при } i > 0; \\ 0 & \text{при } i < 0. \end{cases}$$

Рис. 5.4. Нелінійна система 2-го порядку

виконавши підстановки

$$i = x_1; \quad \dot{i} = \dot{x}_1; \quad x_2 = \dot{x}_1; \quad \ddot{x}_2 = \ddot{x}_1 = i'',$$

отримаємо

$$\begin{cases} \dot{x}_2 + \frac{r}{L}x_2 + \frac{1}{cL}x_1 = 0, \\ \dot{x}_1 = x_2. \end{cases}$$

Запишемо останню систему у формі Коші:

$$\begin{cases} \dot{x}_1 = x_2, \\ \dot{x}_2 = -\frac{1}{cL}x_1 - \frac{r}{L}x_2. \end{cases} \quad (5.5)$$

При цьому

$$a_{11} = 0, \quad a_{12} = 1, \\ a_{21} = -\frac{1}{cL}, \quad a_{22} = -\frac{r}{L}.$$

Тоді структурна схема рівняння (5.5) з використанням умовних позначок буде мати такий вигляд, як показано на рис. 5.5.

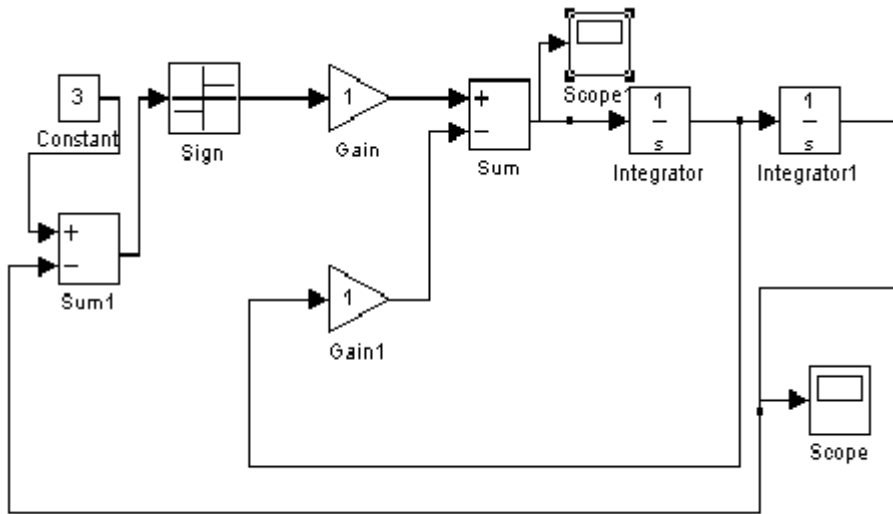


Рис. 5.5. Структурна схема релейної системи

Якщо в (5.5) додати сторонній вплив, то

$$\begin{cases} \dot{x}_1 = x_2, \\ \dot{x}_2 = -\frac{1}{cL}x_1 - \frac{r}{L}x_2 + SIGN(I). \end{cases}$$

Для того щоби простежити за реакцією системи, необхідно вибрати точку спостереження.

Для спостереження за перебігом процесу в точці спостереження підключаються об'єкти «вольтметр» і «осцилограф».

Отримані діаграми подано на рис. 5.6 і 5.7.

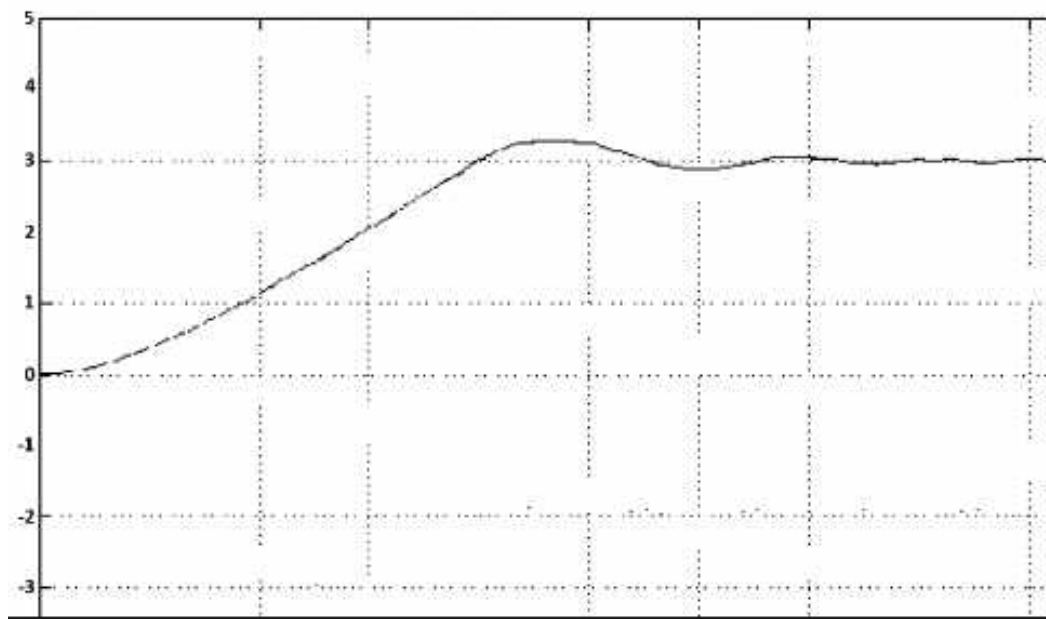


Рис. 5.6. Діаграма: вигляд графіка на осцилографі (Score)

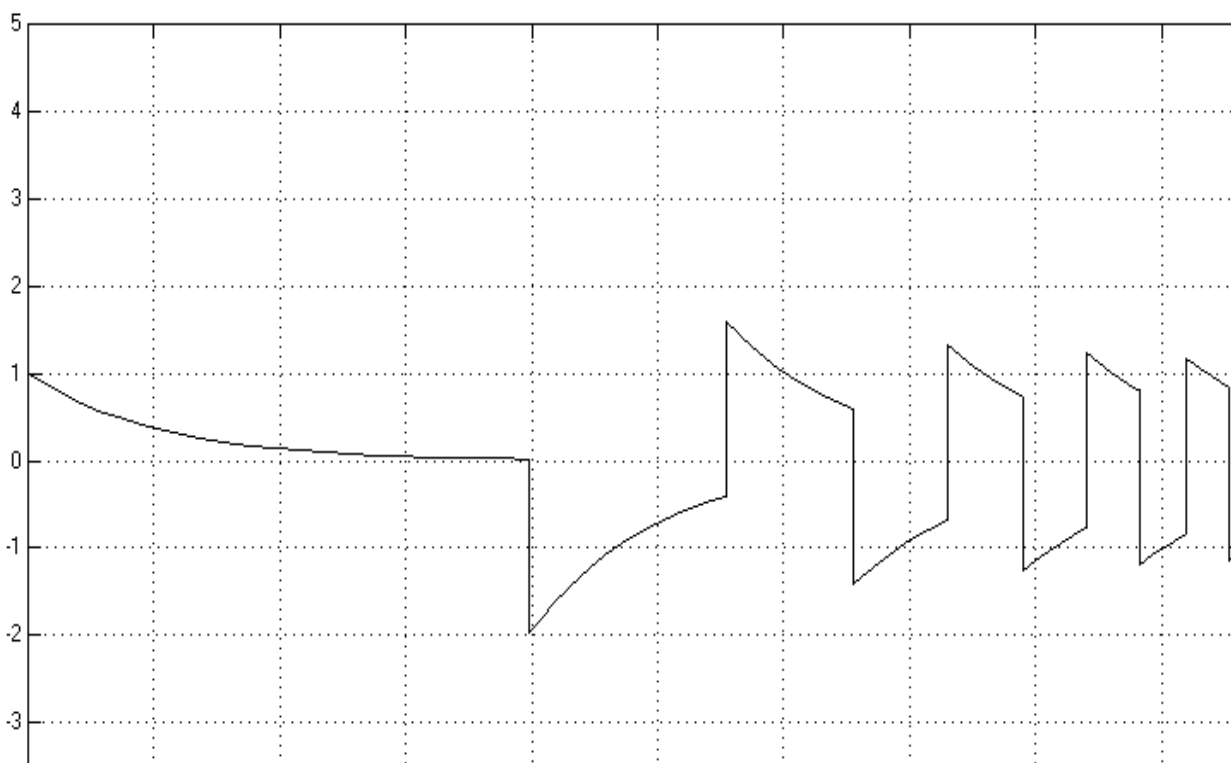


Рис. 5.7. Діаграма: вигляд графіка на осцилографі (Score 1)

Ці графіки мають вигляд автоколивального режиму при зменшенні частоти імпульсів. Траєкторію системи у фазовому просторі показано на рис. 5.8 і 5.9.

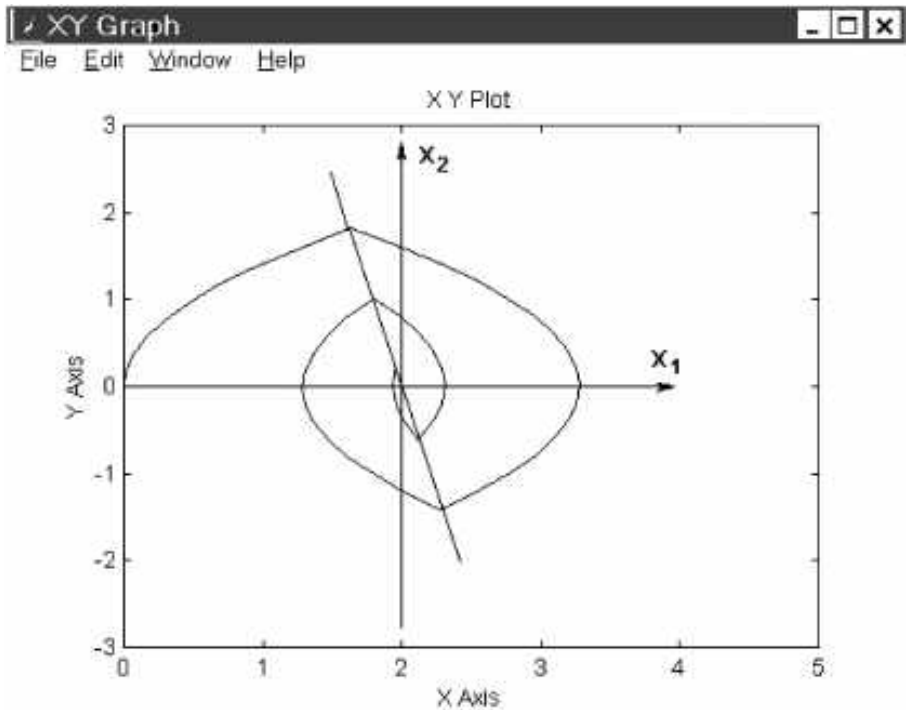


Рис. 5.8. Траєкторія системи у фазовому просторі

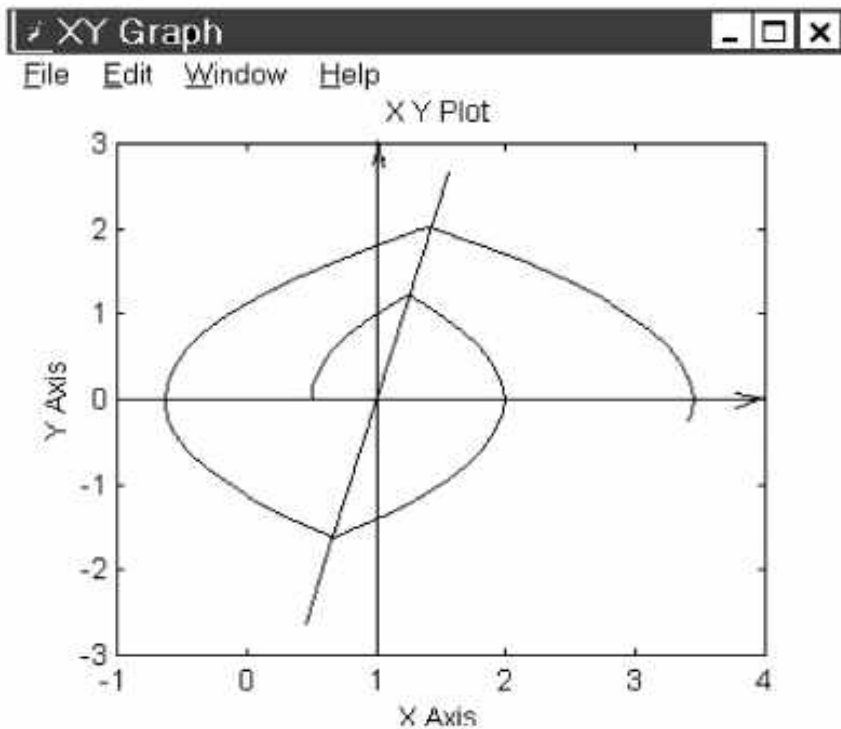


Рис. 5.9. Траєкторія нестійкої системи у фазовому просторі

Задача 5.1

Побудувати структурну схему і знайти розв'язок при $t=0.1$ для такої системи: а) при $c=10e-8$, $U=\text{tg}(3t/2)$; б) при $c=3 \cdot 10e-8$, $U=\sin(5t)$:

$$U = i(t) \cdot r + L \frac{di(t)}{dt} + \frac{1}{c} \int i(t) dt;$$

$$4i'' + 5i' + \frac{1}{c}i = U';$$

$$U = \begin{cases} 0 & \text{при } i > 4; \\ 1 & \text{при } i < -4. \end{cases}$$

Задача 5.2

Побудувати структурну схему і знайти розв'язок при $t=0.2$ для такої системи: а) при $L=10e-6$, $U=\text{tg}(2t)$; б) при $L=8 \cdot 10e-6$, $U=\sin(5t)$:

$$U = i(t) \cdot r + L \frac{di(t)}{dt} + \frac{1}{c} \int i(t) dt;$$

$$4i'' + 5i' + \frac{1}{c}i = U';$$

$$U = \begin{cases} 0 & \text{при } i > 4; \\ 1 & \text{при } i < -4. \end{cases}$$

6 . МОДЕЛЮВАННЯ СИСТЕМ З ДИСКРЕТНИМ ЧАСОМ. МОДЕЛІ ЦИФРОВИХ СИСТЕМ

Системи з дискретним часом належать до класу систем, в яких вхідні дії й вихідні сигнали визначено тільки для деяких регулярних моментів часу на заданому інтервалі, і їх неможливо описувати диференціальними рівняннями. Прикладами таких систем є ЕОМ і мікропроцесорні пристрої.

Вхідні й вихідні змінні зображуються у вигляді ряду послідовностей або можуть бути записані як $U[i]$, $X[i]$, $Y[i]$, при цьому час t для будь-якого індексу i визначається як $Y[i] = AX[n] + BU[n]$.

Водночас до класу цифрових систем можуть належати такі системи, в яких індекс змінних є порядковим номером стану і не залежить від часу.

6.1. Лінійні різницеві моделі з дискретним часом

Лінійні стаціонарні системи, що функціонують в дискретному часі, можуть бути описані лінійними різницевиими рівняннями 1-го порядку $X[n+1] = AX[n] + BU[n]$.

Розглянемо приклад розв'язання системи 2-го порядку в дискретному вигляді (рис. 6.1).

Виконавши підстановки

$$i = x_1; i' = \dot{x}_1; x_2 = \dot{x}_1; \dot{x}_2 = \dot{x}_1 = i'',$$

отримаємо

$$\begin{cases} \dot{x}_2 + \frac{r}{L} x_2 + \frac{1}{cL} x_1 = 0, \\ \dot{x}_1 = x_2. \end{cases}$$

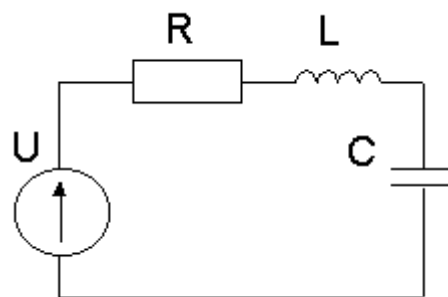


Рис. 6.1. Система 2-го порядку

Запишемо систему у вигляді Коші:

$$\begin{cases} \dot{x}_1 = x_2, \\ \dot{x}_2 = -\frac{1}{cL}x_1 - \frac{r}{L}x_2. \end{cases} \quad (6.2)$$

На підставі рівняння (6.2), отриманого за описом електричного кола диференціального рівняння 2-го порядку, зробимо висновок про можливість побудови останнього на електричних елементах. Для його реалізації достатньо елемента R, який реалізує передавальний коефіцієнт, і двох елементів реактивних C, L, кожен з яких дає одну похідну або інтегрування. Тоді структурна схема рівняння (6.2) з використанням умовних позначень АВМ буде мати такий вигляд, як показано на рис. 6.2. Вираз (6.2) з урахуванням збурень:

$$\begin{cases} \dot{x}_1 = x_2, \\ \dot{x}_2 = -\frac{1}{cL}x_1 - \frac{r}{L}x_2 + bU. \end{cases} \quad (6.3)$$

Для того щоби простежити за реакцією системи, необхідно вибрати точку спостереження.

Для спостереження за перебігом процесу в системі matlab до точки спостереження підключаються об'єкти «вольтметр» або «осцилограф», як показано на рис. 6.3.

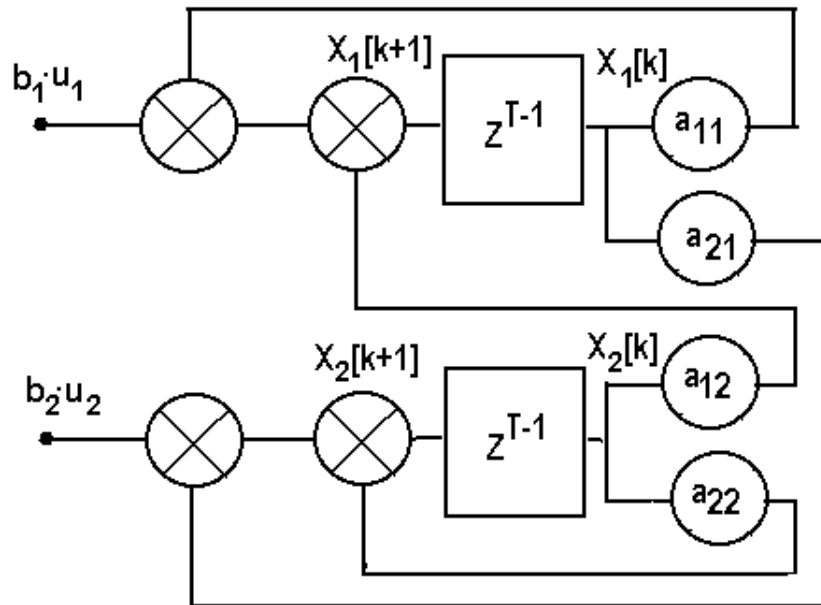


Рис. 6.2. Структурна схема системи (6.1)

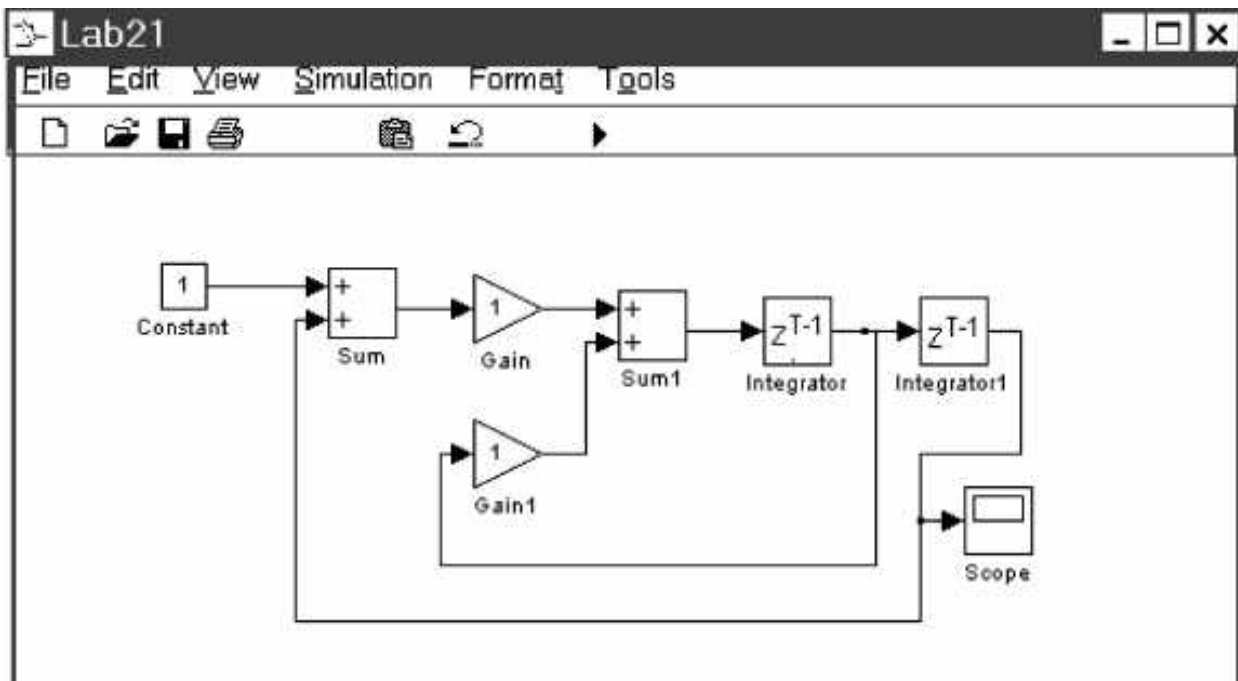


Рис. 6.3. Структурна схема лінійної системи

6.2. Моделювання цифрової системи

Як найпростішу цифрову систему розглянемо систему, що самозамикається, в яка в початковому стані $X(k) = 0$ буде залишатися як завгодно довго до моменту появи зовнішнього сигналу $F(k)$:

$$X(k+1) = X(k) \vee F(k).$$

Структурну схему такої системи показано на рис. 6.4, а її реалізацію в системі matlab – на рис. 6.5.

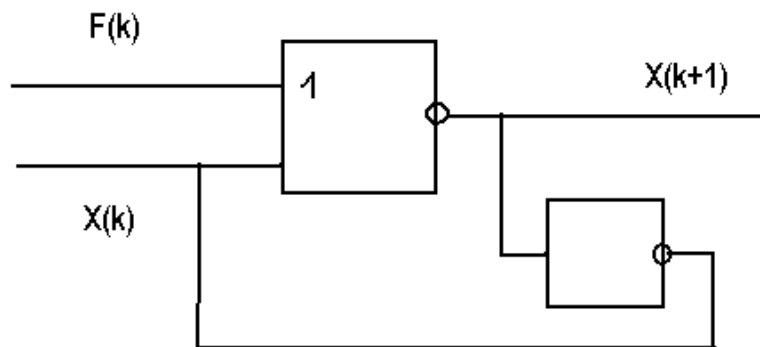


Рис. 6.4. Самозамкнена система

Для ефективного дослідження такої системи пропонується як вхідний вплив і вихідне значення подавати прямокутні сигнали зі значеннями $[0,1]$ з різною частотою проходження.

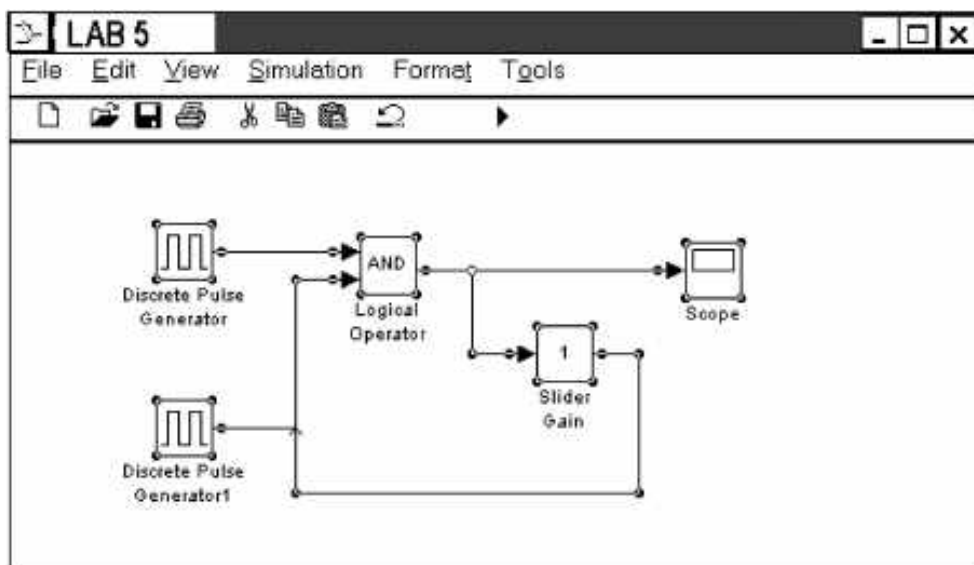


Рис. 6.5. Структурна схема системи, що самозамикається, в пакеті MATLAB

6.3. Приклади моделювання цифрових автоматних систем

Розглянемо приклад послідовних рівнянь автоматної системи 2-го порядку (рис. 6.6).

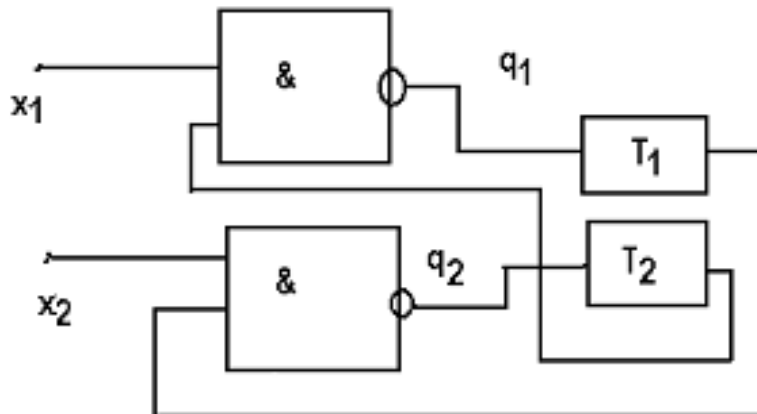


Рис. 6.6. Послідовна система

Така система може бути описана таким чином:

$$\begin{aligned} X_1(k+1) &= X_2(k) \vee q_1(k), \\ X_2(k+1) &= X_1(k) \vee q_2(k). \end{aligned}$$

У системі matlab цьому рівнянню відповідає структурна схема, наведена на рис. 6.7.

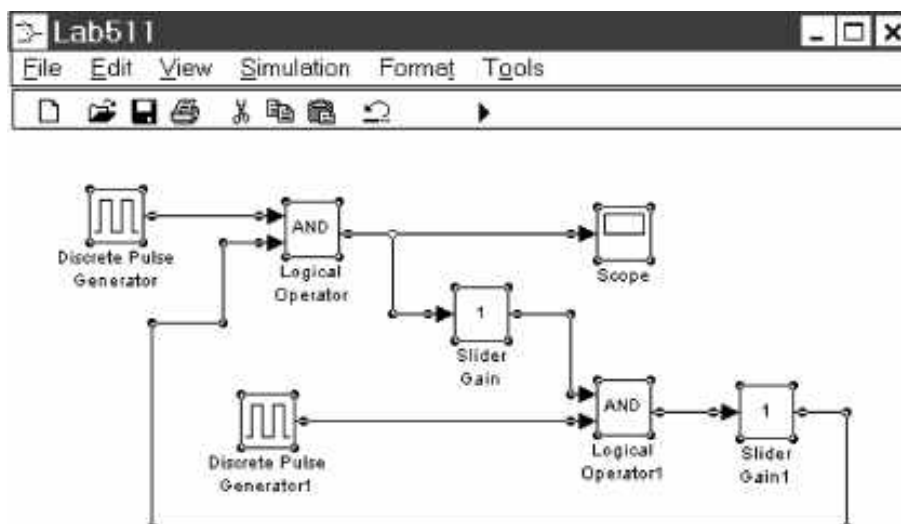


Рис. 6.7. Структурна схема послідовної системи

Для ефективного дослідження такої системи пропонується як вхідний вплив подавати прямокутні послідовності імпульсів зі значеннями $[0,1]$ з різною частотою проходження.

Для підтримки стійкості такої системі необхідно додати рівняння керування.

Для дослідження також пропонується система з додаванням сигналів, де як вхідний вплив слід подавати константи або прямокутні сигнали зі значеннями $[0,1]$ (рис. 6.8).

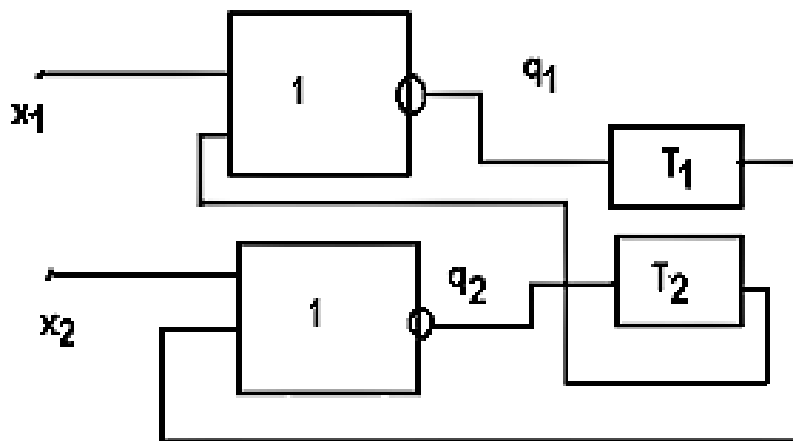


Рис. 6.8. Послідовна система №2

У системі matlab цьому відповідає структурна схема, наведена на рис. 6.9.

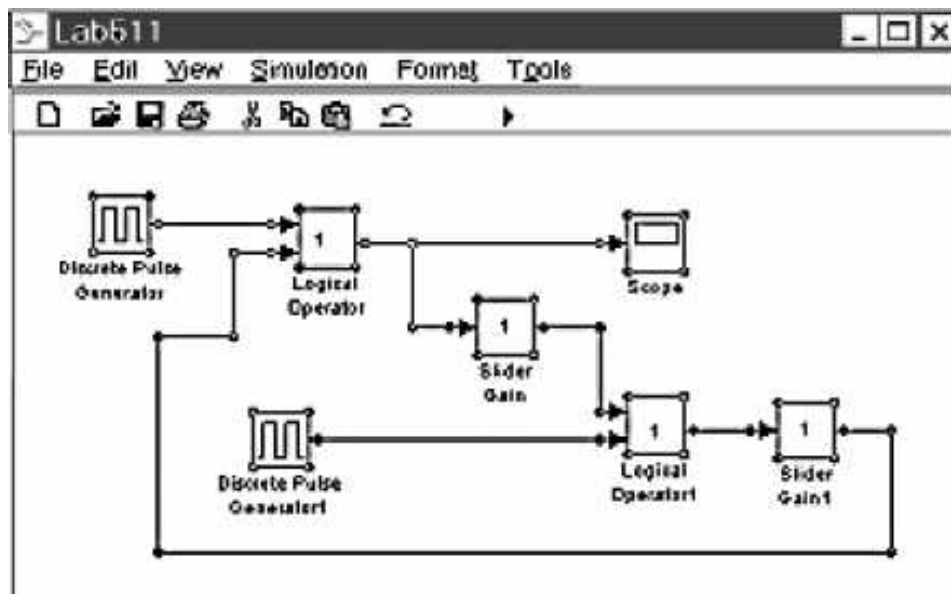


Рис. 6.9. Структурна схема послідовної системи №2

Завдання до самопідготовки

Побудувати структурну схему системи і провести дослідження зазначеної дискретної системи відповідно до даних табл. 6.1.

Таблиця 6.1

Умови завдань

Но- мер варі- анта	Модель дискретної системи	Значення параметрів
1	$x_1=1 \cdot x_2 \vee 1 \cdot q_1;$ $x_2=1 \cdot x_1 \vee 0 \cdot q_2$	$x_1 = 0; x_2 = 1,$ $q_1 = 0, q_2 = 1$
2	$x_1=1 \cdot x_2 \vee 0 \cdot q_2;$ $x_2=1 \cdot x_1 \vee 0 \cdot q_1$	$x_1=0; x_2=0, q_1=0, q_2=1$
3	$x_1=1 \cdot x_1 \cdot 1 \cdot q_1;$ $x_2=1 * x_2 \vee 0 * q_2$	$x_1=0; x_2=1, q_1=0, q_2=1$
4	$x_1= x_1 \cdot q_1 \vee 1 \cdot q_2;$ $x_2=1 \cdot x_2 \vee 0 \cdot q_2$	$x_1=1; x_2=1, q_1=0, q_2=0$
5	$x_1=1 \cdot x_2 \vee 1 \cdot q_1;$ $x_2=1 \cdot x_1 \vee 1 \cdot q_2$	$x_1=1; x_2=0, q_1=0, q_2=1$
6	$x_1=1 \cdot x_2 \vee 1 \cdot q_1 \vee x_2 \vee 1 \cdot q_2;$ $x_2=1 \cdot x_2 \vee 0 \cdot q_2$	$x_1=0; x_2=1, q_1=0, q_2=1$
7	$x_1=1 \cdot x_2 \vee 1 \cdot q_1 \vee x_2 \vee 1 \cdot q_2;$ $x_2=1 \cdot x_2 \vee 1 \cdot q_2$	$x_1=0; x_2=1, q_1=0, q_2=1$
8	$x_1=1 \cdot x_1 \cdot 1 \cdot q_1;$ $x_2=1 * x_2 \vee 0 * q_2$	$x_1=1; x_2=1, q_1=0, q_2=1$
9	$x_1= x_1 \cdot q_1 \vee 1 \cdot q_2;$ $x_2=1 \cdot x_2 \vee 0 \cdot q_2$	$x_1=1; x_2=1, q_1=1, q_2=0$
10	$x_1=1 \cdot x_2 \vee 1 \cdot q_1;$ $x_2=1 \cdot x_1 \vee 1 \cdot q_2$	$x_1=1; x_2=0, q_1=1, q_2=1$

Закінчення табл. 6.1

Но- мер варі- анта	Модель дискретної системи	Значення параметрів
11	$x_1=1 \cdot x_2 \vee 1 \cdot q_1 \vee x_2 \vee 1 \cdot q_2;$ $x_2=1 \cdot x_2 \vee 0 \cdot q_2$	$x_1=0; x_2=1, q_1=1, q_2=1$
12	$x_1=1 \cdot x_2 \vee 1 \cdot q_1 \vee x_2 \vee 1 \cdot q_2;$ $x_2=1 \cdot x_2 \vee 1 \cdot q_2$	$x_1=1; x_2=1, q_1=1, q_2=1$
13	$x_1=1 \cdot x_2 \vee 0 \cdot q_1 \vee x_2 \vee 1 \cdot q_2;$ $x_2=1 \cdot x_2 \vee 1 \cdot q_2$	$x_1=0; x_2=1, q_1=0, q_2=1$
14	$x_1=1 \cdot x_1 \cdot 0 \cdot q_1;$ $x_2=1 * x_2 \vee 0 * q_2$	$x_1=1; x_2=0, q_1=0, q_2=1$
15	$x_1= x_1 \cdot q_1 \vee 1 \cdot q_2;$ $x_2=1 \cdot x_2 \vee 1 \cdot q_2$	$x_1=0; x_2=1, q_1=1, q_2=0$

7. ОРГАНІЗАЦІЯ НАПІВНАТУРНОГО МОДЕЛЮВАННЯ

7.1. Активний експеримент

Для збільшення результативності напівнатурного моделювання експеримент проводять в певних точках, що відповідають найбільш характерним ділянкам вхідних впливів.

Виконання ідентифікації при цьому передбачає:

- 1) вибір раціональних моментів спостереження;
- 2) задання певних початкових умов;
- 3) подачу на вхід спеціального керівного сигналу.

Таким чином, проведення активного експерименту на основі деякого заздалегідь сформованого плану забезпечує активний вплив оператора на хід досліджень.

При активному експерименті для визначення параметра $A[\cdot]$ проводять експеримент $X^T = [X_{10}, X_{20}, X_{30}, \dots, X_{N0}]$, при цьому $N \geq M$, де M – очікуваний порядок моделі.

План експерименту подається у вигляді матриці плану

$$E_{0N} = \left\{ \begin{array}{cccc} X_{10}, & X_{20}, & X_{30}, & \dots, & X_{N0} \\ n_{10}, & n_{20}, & n_{30}, & \dots, & n_{N0} \end{array} \right\},$$

де n_{io} - початкові значення частки експерименту в кожному досліді. Ці значення можуть вибиратися у вигляді відносної величини

$$P_{io} = \frac{n_{io}}{N_0},$$

де N - загальна кількість експериментів.

Наприклад, для $N=3$ $P_{io} = \frac{1}{3}$,

$$E(X, N) = \begin{Bmatrix} -1, & 0, & 1 \\ 1/3, & 1/3, & 1/3 \end{Bmatrix}.$$

Зазвичай існують визначувана область X , допустима для точок експерименту $X_{i0} \in X$, а також область, в якій спостерігаються найкращі результати. Завданням активної ідентифікації є відшукування цієї області.

Основними критеріями при побудові плану експерименту є критерії ортогональності й рототабельності.

7.1.1. Критерій ортогональності

При плануванні потрібний вибір таких точок плану $E(X, N)$, при яких сума почленних добутоків будь-яких двох векторів-стовпців дорівнює нулю:

$$\sum X_i[j] \times X_i[l] = 0; j \neq l; j, l = 1 \dots k.$$

Використання цього критерію дозволяє отримати незалежні оцінки, а також спростити деякі обчислення.

7.1.2. Критерій рототабельності

В області планування точок експерименту необхідно здійснити їх вибір так, щоби точність прогнозу була однаковою в усіх напрямках на рівних відстанях від центру експерименту. Виконання цього критерію відповідає вимогам рівнозначності.

Ці критерії виконуються при повному факторному експерименті, який складено із множини точок у M -вимірному просторі (табл. 7.1).

Таблиця 7.1

Формування вектора X при повному факторному експерименті

№ п/п	X_1	X_2	X_3
1	+1	+1	+1
2	-1	+1	+1
3	+1	-1	+1
4	-1	-1	+1
5	+1	+1	-1
6	-1	+1	-1
7	+1	-1	-1
8	-1	-1	-1

Якщо в багатофакторному експерименті використано всі можливі комбінації рівнів факторів, то такий експеримент називається повним факторним експериментом (ПФЕ). Залежність кількості експериментів від кількості факторів для лінійного рівняння регресії наведено в табл. 7.2.

Таблиця 7.2

Залежність кількості експериментів від факторів

Кількість факторів K	Кількість невідомих параметрів	Кількість дослідів ПФЕ	Достатня кількість дослідів для визначення параметрів
3	4	8	2^{3-1}
4	5	16	2^{4-1}
5	6	32	2^{5-2}
6	7	64	2^{6-3}

Таким чином, ПФЕ включає 2^k дослідів, які при знаходженні коефіцієнтів для лінійної моделі можуть повністю не використовуватися. У принципі ПФЕ дозволяє знайти 2^k коефіцієнтів регресії при 2^k базисних функцій (для планів більш високого порядку). Перші $k+1$ –базисні функції очевидні - вони становлять лінійну модель ($f_0=1, f_1=X_1, f_2=X_2, f_3=X_3$). Для подальших базисних функцій використовуємо нижченаведені значення функцій. Для квадратичної моделі трифакторного експерименту ПФЕ буде являти собою матрицю базисних функцій (табл. 7.3).

Таблиця 7.3

ПФЕ для квадратичної моделі трифакторного експерименту

№ п/п	Параметри							
	X_0	X_1	X_2	X_3	X_1X_2	X_1X_3	X_2X_3	$X_1X_2X_3$
1	+1	-1	-1	-1	+1	+1	+1	-1
2	+1	+1	-1	-1	-1	-1	+1	+1
3	+1	-1	+1	-1	-1	+1	-1	+1
4	+1	+1	+1	-1	+1	-1	-1	-1
5	+1	-1	-1	+1	+1	-1	-1	+1
6	+1	+1	-1	+1	-1	+1	-1	-1
7	+1	-1	+1	+1	-1	-1	+1	-1
8	+1	+1	+1	+1	+1	+1	+1	+1

7.2. Аналіз методів розроблення планів багатфакторного експерименту

Пошук оптимальних умов є однією з найпоширеніших науково-технічних завдань. Ці умови виникають у той момент, коли встановлено можливість проведення процесу і необхідно знайти найкращі (оптимальні в деякому розумінні) умови його реалізації.

Метод планування експерименту - один з найефективніших методів, що дозволяє отримувати статистичні математичні моделі процесів, використовуючи факторне планування, регресійний аналіз і рух по градієнту [4 - 7].

Застосування планування експерименту робить поведінку експериментатора цілеспрямованою і організованою, істотно сприяє підвищенню продуктивності його праці і надійності отриманих результатів. Важливими достоїнствами методу є його універсальність, придатність у більшості галузей дослідження, що цікавлять сучасну людину.

Фактором називається вимірювана змінна величина, що набуває в деякий момент часу певного значення. Фактори відповідають способам впливу на об'єкт дослідження.

При плануванні експерименту зазвичай одночасно змінюються декілька факторів. Тому дуже важливо сформулювати вимоги, які ставляться до сукупності факторів. Передусім висувається вимога сумісності. Сумісність факторів означає, що всі їхні комбінації здійсненні й безпечні.

При плануванні експерименту важлива незалежність факторів, тобто можливість установа фактора на будь-якому рівні незалежно від рівнів інших факторів. Якщо ця умова нездійсненна, то неможливо планувати експеримент. Отже, обов'язкова умова - відсутність кореляції між факторами. Вимога некорельованості не означає, що між значеннями факторів немає ніякого зв'язку. Достатньо, щоби зв'язок не був лінійним.

Крім того, передбачають таке:

- множину визначальних факторів задано;
- кожен з факторів – керований;
- результати дослідів відтворюються;
- досліди рівноцінні, тобто відмінністю у вартості можна знехтувати;
- розв'язується задача пошуку оптимальних умов (або в деяких випадках - інтерполяції);
- математична модель процесу заздалегідь не відома.

Питанням розвитку методу планування експерименту приділяється досить велика увага у вітчизняній і зарубіжній науковій літературі. Розглянемо основні напрями цього розвитку.

Питанню автоматизації вибору планів дробового факторного експерименту присвячена робота, в якій розроблено ряд тверджень, покладених в основу алгоритму вибору плану дробового факторного експерименту. Створено алгоритми і програми, що реалізують процес автоматичного вибору на основі апріорної інформації плану дробового

експерименту, який влаштовує дослідника, і синтезу його на ЕОМ. Застосування розроблених програм дозволяє не залучати для проведення експериментів операторів з високим ступенем кваліфікації [4].

Наступним важливим питанням є оцінювання адекватності лінійних регресійних моделей за допомогою виділення псевдопаралельних спостережень у наявній вибірці. Алгоритм повинен виконувати перевірку значущості й стохастичної незалежності залишків, а також значущості множинного коефіцієнта кореляції, дисперсії відтворюваності й дисперсії залишків.

Дослідженню алгоритму аналізу адекватності моделей з виділенням псевдопаралельних спостережень присвячено роботу [8].

7.2.1. Властивості повного факторного експерименту

Матриця планування ПФЕ має такі основні властивості:

1. *Симетричність плану* відносно центру експерименту, тобто сума значень рівнів будь-якого фактора дорівнює нулю.

2. *Нормування плану* виражається як сума квадратів значень рівнів будь-якого фактора, що дорівнює кількості N рядків матриці планування ПФЕ:

$$\sum_{j=1}^N x_j = 0; \quad \sum_{j=1}^N x_j^2 = 0.$$

3. *Ортогональність плану* визначається як сума попарних добутків значень рівнів будь-яких двох факторів (крім $j=i$), що дорівнює нулю:

$$\sum_{\substack{j=1, j=1 \\ j \neq i}}^N x_i x_j = 0.$$

4. *Рототабельність плану* визначає той факт, що точність прогнозу значень функції відклику однакова на рівній відстані від центра і не залежить від напрямку обходу.

Зазначимо, що властивості ортогональності й рототабельності взаємовиключні.

Визначення коефіцієнтів рівняння регресії

Після проведення дослідів в усіх точках факторного простору необхідно знайти коефіцієнти рівняння регресії. Для цього скористаємося методом найменших квадратів, тобто сума квадратичних різниць теоретичних і експериментальних даних має бути мінімальною:

$$\Phi = \sum_{i=1}^n (\hat{Y}_i - Y_i)^2 \rightarrow \min;$$

$$\hat{Y}_i = \varphi(X_1, \dots, X_k, b_0, \dots, b_k);$$

$$\Phi = \sum_{i=1}^n (\varphi(X_1, \dots, X_k, b_0, \dots, b_k) - Y_i)^2.$$

Після диференціювання отримаємо

$$\left\{ \begin{array}{l} \frac{\partial \Phi}{\partial b_0} = 2 \sum_{i=1}^n (\varphi(X_1, \dots, X_k, b_0, \dots, b_k) - Y_i) \frac{\partial \varphi}{\partial b_0} = 0; \\ \dots \\ \frac{\partial \Phi}{\partial b_k} = 2 \sum_{i=1}^n (\varphi(X_1, \dots, X_k, b_0, \dots, b_k) - Y_i) \frac{\partial \varphi}{\partial b_k} = 0. \end{array} \right.$$

Для лінійної регресії при $k=2$ запишемо рівняння в повній формі:

$$\begin{cases} b_0 + \left(\frac{1}{n} \sum_{i=1}^n X_{1i}\right)b_1 + \left(\frac{1}{n} \sum_{i=1}^n X_{2i}\right)b_2 = \frac{1}{n} \sum_{i=1}^n Y_i; \\ \left(\frac{1}{n} \sum_{i=1}^n X_{1i}\right)b_0 + \left(\frac{1}{n} \sum_{i=1}^n X_{1i}^2\right)b_1 + \left(\frac{1}{n} \sum_{i=1}^n X_{2i}X_{1i}\right)b_2 = \frac{1}{n} \sum_{i=1}^n X_{1i} Y_i; \\ \left(\frac{1}{n} \sum_{i=1}^n X_{2i}\right)b_0 + \left(\frac{1}{n} \sum_{i=1}^n X_{1i}X_{2i}\right)b_1 + \left(\frac{1}{n} \sum_{i=1}^n X_{2i}^2\right)b_2 = \frac{1}{n} \sum_{i=1}^n X_{2i} Y_i. \end{cases}$$

Таким чином, отримано систему лінійних рівнянь з коефіцієнтами регресії як невідомими. Для її розв'язання можна скористатися методом Крамера.

Введемо підстановку для суми вхідних впливів:

$$\begin{cases} b_0 + m_{x1}b_1 + m_{x2}b_2 = m_y; \\ m_{x1}b_0 + a_1b_1 + a_2b_2 = a_{11}; \\ m_{x2}b_0 + a_2b_1 + a_3b_2 = a_{22}. \end{cases}$$

Отримаємо шукані параметри:

$$b_0 = \frac{\begin{vmatrix} m_y & m_{x1} & m_{x2} \\ a_{11} & a_1 & a_2 \\ a_{22} & a_2 & a_3 \end{vmatrix}}{\begin{vmatrix} 1 & m_{x1} & m_{x2} \\ m_{x1} & a_1 & a_2 \\ m_{x2} & a_2 & a_3 \end{vmatrix}}; \quad b_1 = \frac{\begin{vmatrix} 1 & m_y & m_{x2} \\ m_{x1} & a_{11} & a_2 \\ m_{x2} & a_{22} & a_3 \end{vmatrix}}{\begin{vmatrix} 1 & m_{x1} & m_{x2} \\ m_{x1} & a_1 & a_2 \\ m_{x2} & a_2 & a_3 \end{vmatrix}}; \quad b_2 = \frac{\begin{vmatrix} 1 & m_{x1} & m_y \\ m_{x1} & a_1 & a_{11} \\ m_{x2} & a_2 & a_{22} \end{vmatrix}}{\begin{vmatrix} 1 & m_{x1} & m_{x2} \\ m_{x1} & a_1 & a_2 \\ m_{x2} & a_2 & a_3 \end{vmatrix}}.$$

7.3. Практичне використання планів ПФЕ в системі Statgraphics

У системі Statgraphics повний трифакторний експеримент проводиться в такому порядку.

1. Запустити прикладну програму Statgraphics і вибрати опцію "Планування нового експерименту".

2. Скласти матрицю планування для повного трифакторного експерименту з використанням додаткового нульового фактора ($X_0=1$) і

заповнити таблицю кодованими значеннями X_1 , X_2 і X_3 .

3. Перевірити властивості повного факторного експерименту: симетричність, нормування, ортогональність і рототабельність.

4. Провести імітаційний експеримент у всіх точках ПФЕ. Визначити значення функції відклику.

5. Активізувати режим досліджень, використовуючи функції Menu - Relate - Multiply Regressions, і дістати параметри моделі (рис. 7.1).

6. Отримати коефіцієнти b_0 , b_1 , b_2 рівняння регресії у вигляді $Y = b_0 X_0 + b_1 X_2 + b_2 X_2 + b_3 X_3$ (рис. 7.2).

7. Записати загальний вигляд моделі і зарисувати діаграми з коментарями до описаних об'єктів.

8. Зробити висновки з виконаної роботи. Вийти з оболонки Statgraphics з використанням команди Exit у запрошенні командного вікна.

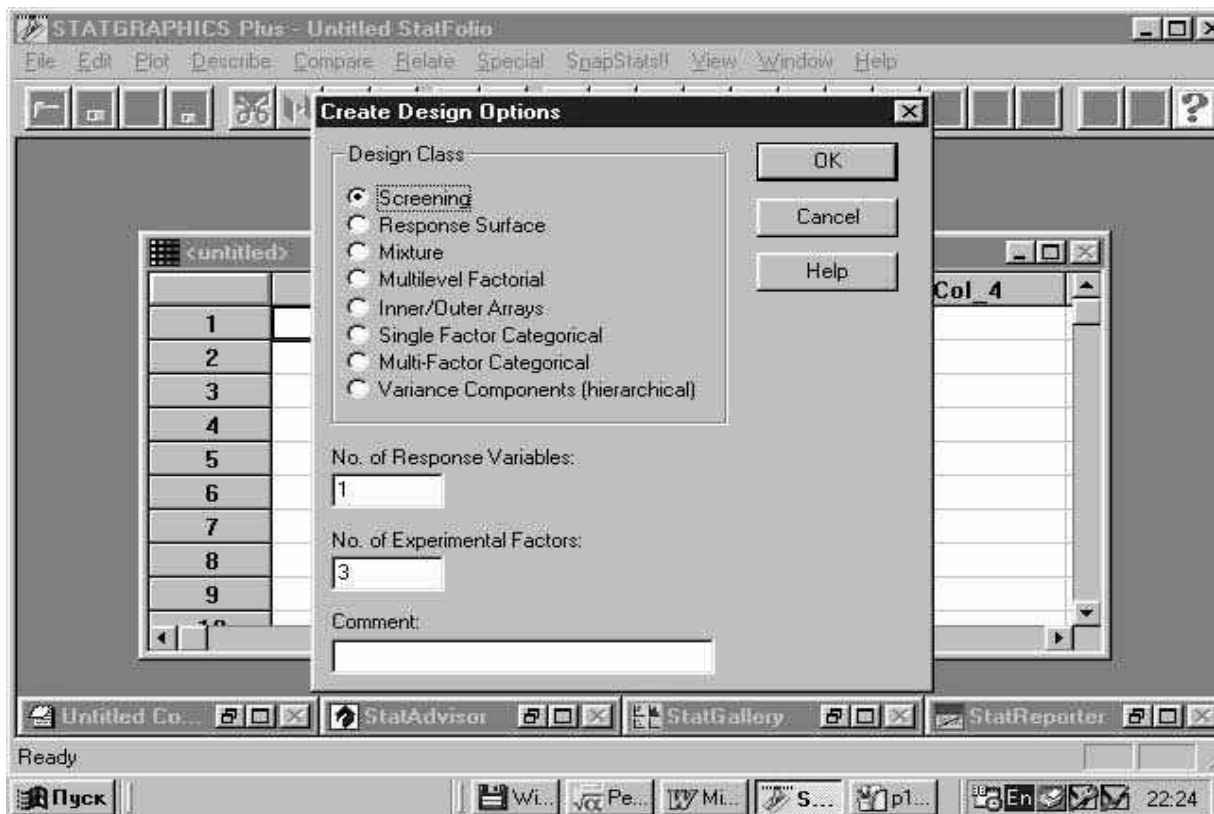


Рис. 7.1. Вибір режиму побудови нового експерименту

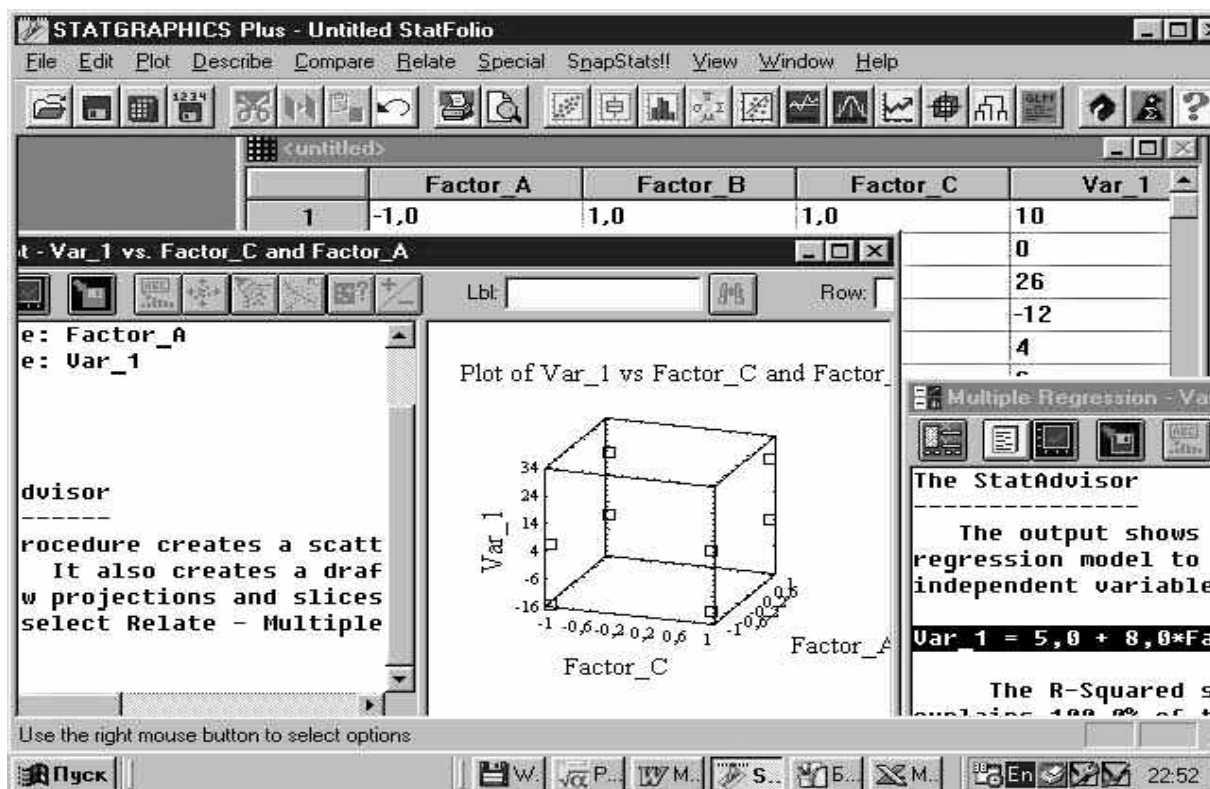


Рис. 7.2. Отримання графічних і числових результатів для ПФЕ

Контрольні запитання

1. Перелічіть умови, необхідні для визначення коефіцієнтів регресії.
2. Дайте визначення повного факторного експерименту.
3. Опишіть відмінність активного експерименту від пасивного.
4. Сформуйте план ПФЕ для двовимірної моделі.
5. Визначіть кількість експериментів ПФЕ для кількості факторів, що дорівнює 5, 7, 8, 9.

8. ОБРОБЛЕННЯ РЕЗУЛЬТАТІВ МОДЕЛЮВАННЯ ПРИ МІНІМІЗАЦІЇ КІЛЬКОСТІ ЕКСПЕРИМЕНТІВ

8.1. Дробово-факторний експеримент

У деяких випадках немає необхідності використовувати повний факторний експеримент. Тоді кількість рядків матриці ПФЕ зменшують до кількості коефіцієнтів регресійної моделі. Це проводиться у випадках лінійної регресійної моделі. Дробовий факторний експеримент (ДФЕ) задовольняє всі властивості повного факторного експерименту.

Для скорочення обсягу експерименту при багатоконпонентних системах запропоновано методику лінійного перетворення істотно корельованих інформаційних планів активного експерименту в ортогональні інформаційні плани узагальнених аргументів меншої розмірності, що дозволяє істотно скоротити обсяг експерименту. Скорочення обсягу експерименту досягається за рахунок запропонованого в методиці способу ідентифікації точок векторного поля узагальнених аргументів відновлюваної функції регресії числовими значеннями точок скалярного поля інгредієнтів багатоконпонентної системи [9].

Очевидно, що кількість дослідів швидко зростає зі збільшенням розмірності факторного простору.

Оскільки $N=2^k$, то при $K=3$ $N=8$ і вже при $K=7$ $N=128$. Для експерименту можна взяти тільки частину точок плану для прийнятої кількості дослідів (див. табл. 7.2). Отримаємо дробово-факторний експеримент з кількістю експериментів $N=2^{k-d}$, де d - показник дробовості ДФЕ.

8.1.1. Приклад побудови ДФЕ

Розглянемо приклад експериментальних досліджень виробу, модель якого $A [B, X]$ згідно з гіпотезою H_1 являє собою лінійний тривимірний поліном $Y=b_0+b_1x_1+b_2x_2+b_3x_3$.

Вхідні впливи $x_1 = U_{вх}$, $x_2 = U_{змш}$, $x_3 = t$ мають такий діапазон:

$$\left\{ \begin{array}{l} U_{вх} = (0..10)B; \\ U_{зміщ} = (-1..+1)B; \\ t = (0..50)^{\circ}C. \end{array} \right.$$

Для побудови плану ДФЕ виконаємо нормалізацію значень шляхом їх масштабування і центрування.

Об'єкт дослідження розглядається як носій деяких невідомих або таких, що підлягають дослідженню властивостей і якостей, своєрідний «чорний ящик». У цієї конфігурації вектор $X_1...X_k$ являє собою групу контрольованих і керованих величин, які можуть змінюватися належним чином у ході експерименту, а $Z_1...Z_k$ - тільки контрольовані характеристики. Першу групу характеристик ($X_1...X_k$) також називають факторами, або керованими впливами. Y – функція відклику (поверхня відклику), або простіше - реакція системи на вплив факторів. Можна виділити і третю, не позначену на ідеальній моделі, систему вхідних сигналів – шуми або перешкоди, які в реальному житті є не чим іншим, як помилками обслуговуючого персоналу, впливом зовнішнього середовища, похибками приладів і т. д. До цієї ж групи можна віднести вплив тих характеристик, які не можуть контролюватися ззовні або через їхню складність, або через незнання їхньої природи і неможливості їх контролювати.

Вихідне значення визначається як $Y=U_{вих}$, а вектор вхідних впливів – параметрами $x_1 - x_3$.

Різні характеристики об'єктів мають різну фізичну природу, а отже, й розмірність, що ускладнює побудову експериментальної моделі. Тому на практиці значення факторів, які мають реальний фізичний зміст, нормують певним чином (зводять до певного раніше заданого набору значень). Для будь-якого фактора (набору значень) X існує нижній X_{min} і верхній X_{max} рівні зміни значень. Наведемо алгоритм нормування фактора.

Вибираємо масштаб і положення осей координат таким чином, щоби X_{min} відповідало -1 , а X_{max} - $+1$.

Обчислюємо значення X_0 для цього фактора таким чином:

$$X_0 = \frac{X_{max} + X_{min}}{2}.$$

Визначаємо інтервал зміни фактора:

$$dx = X_0 - X_{min} = X_{max} - X_0.$$

Знаходимо нормоване значення X_n для кожного фактора:

$$X_n = \frac{X - X_0}{dx}.$$

Оскільки очікувана модель лінійна, то матрицю оцінок параметрів B можна отримати, розв'язавши рівняння $Y = B \times X$.

Для цього необхідно використати метод Гаусса розв'язання лінійних систем або МНК:

$$\sum (Y - B \times X) X = 0.$$

При відновленні реальних параметрів моделі слід виконати обернене масштабування:

$$b_i = mb_i + b_0^{\approx}.$$

Запишемо загальну формулу для обчислення параметрів математичної моделі:

$$b_i = \left(\sum_{j=1}^n x_{ij} x_j \right) / \left(\sum_{j=1}^n x_{ij}^2 \right).$$

Виберемо генеруюче співвідношення $x_3 = x_1 \cdot x_2$ для отримання стовпця x_3 . Тоді оцінки параметрів такі:

$$b_0 = (1 \cdot 8.1 - 1 \cdot 0.1 - 1 \cdot 0.1 - 1 \cdot 1.9) / 4 = 0;$$

$$b_1 = (1 \cdot 8.1 + 0.1 - 0.1 + 1.9) / 4 = 1;$$

$$b_2 = (1 \cdot 8.1 + 1 \cdot (-0.1) + (-1) \cdot (-0.1) + (-1) \cdot (1.9)) / 4 = 1;$$

$$b_3 = (1 \cdot 8.1 + 0.1 + 0.1 - 1.9) / 4 = 0.1.$$

Виконаємо перевірку:

$$Y_1 = 0 + 1 \cdot x_1 + 1 \cdot x_2 + 0.1 \cdot x_3 = 0 + 1 + 1 + 0.1 = 8.1;$$

$$Y_2 = 0 - 1 + 1 + 0.1 = 0.1.$$

Коефіцієнти знайдено правильно.

Отриманий план ДФЕ відрізняється такими властивостями:

1. Ортогональність:

$$\sum_{i=1}^N x_{ij} x_{ik} = 0; j \neq k; \quad j, k = \overline{1, n};$$

$$\Sigma = (+1)(+1) + (-1)(+1) + (+1)(-1) + (-1)(-1) = 0;$$

$$\Sigma = (+1)(+1) + (+1)(-1) + (-1)(-1) + (+1)(-1) = 0;$$

$$\Sigma = (+1)(+1) + (-1)(-1) + (-1)(+1) + (+1)(-1) = 0.$$

Ортогональність показує розташування базисних векторів у просторі (рис. 8.1).

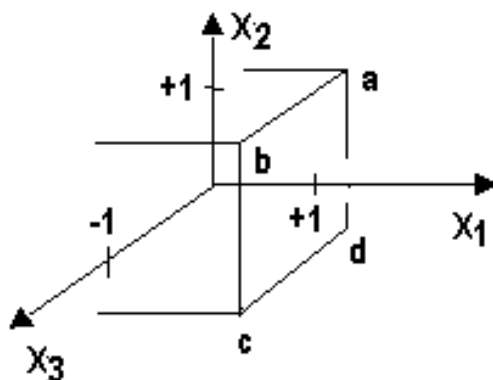


Рис. 8.1. Розташування базисних векторів і точок плану в просторі вхідних впливів

2. Симетричність: $\sum_{i=1}^N x_{ij} = 0; j = \overline{1, n}.$

3. Нормування: $\sum x_{ij}^2 = N = 4.$

Отримані оцінки параметрів моделі описують міру впливу факторів x_1, x_2, x_3 на вектор відклику Y .

8.2. Оброблення результатів експериментів

8.2.1. Знаходження порядкової дисперсії

Припустимо, що в кожній точці факторного простору, якій відповідає один з рядків матриці планування, проводиться серія з m дослідів. Для будь-якої i -ї точки обчислюємо середнє значення вихідної величини

$$y_i = \sum_{u=1}^m \frac{y_{iu}}{m} \quad (8.1)$$

і порядкову дисперсію вихідної величини

$$S^2\{y_i\} = \sum_{u=1}^m \frac{(y_{iu} - y_i)^2}{m-1} \quad (8.2)$$

Розглянемо етапи оброблення результатів експерименту на прикладі (табл. 8.1).

Таблиця 8.1
Результати експерименту по ДФЕ

$\frac{N_0}{n/n}$	X_0	X_1	X_2	X_1X_2	Y_{1i}	Y_{2i}	Y_{3i}	Y_i	$S^2\{y_i\}$
1	+1	-1	-1	+1	21	17	24	20,7	12,3
2	+1	+1	-1	-1	45	43	47	45	4
3	+1	-1	+1	-1	5	8	7	6,7	2,3
4	+1	+1	+1	+1	28	27	29	28	1

Середнє значення вихідної величини Y_i в кожній точці знайдемо за формулою (8.1) при ($m = 3$):

$$Y_1 = (21+17+24)/3 = 20,7;$$

$$Y_2 = (45+43+47)/3 = 45;$$

$$Y_3 = (5+8+8)/3 = 6,7;$$

$$Y_4 = (28+27+29)/3 = 28.$$

Визначимо за формулою (8.2) порядкову дисперсію:

$$S^2\{y_1\} = [(21-20,7)^2 + (17-20,7)^2 + (24-20,7)^2] / 2 = 12,3;$$

$$S^2\{y_2\} = [(45-45)^2 + (43-45)^2 + (47-45)^2] / 2 = 4;$$

$$S^2\{y_3\} = [(5-6,7)^2 + (8-6,7)^2 + (7-6,7)^2] / 2 = 2,3;$$

$$S^2\{y_4\} = [(28-28)^2 + (27-28)^2 + (29-28)^2] / 2 = 1.$$

8.2.2. Перевірка однорідності за критерієм Кохрена

Серед всієї сукупності розрахованих порядкових дисперсій вибираємо максимальну $S^2\{y_{ij\max}\}$ і беремо відношення цієї дисперсії до суми всіх порядкових дисперсій $S^2\{y_{ij}\}$, тобто визначаємо розрахункове значення коефіцієнта Кохрена:

$$G_p = \frac{S^2\{Y_i\}_{\max}}{\sum_{i=1}^N S^2\{Y_i\}}.$$

У разі ідеальної однорідності порядкових дисперсій коефіцієнт G_p прямував би до значення $1/N$.

Розрахункове значення коефіцієнта Кохрена порівнюємо з табличним значенням G -критерію, яке вибираємо з таблиць для прийнятого рівня значущості α і для чисел степеня вільності чисельника f_1 і знаменника f_2 відповідно:

$$f_1 = m - 1; f_2 = N.$$

Для цього значення f_1 знаходимо в горизонтальному заголовку таблиці, а f_2 вибираємо зліва у вертикальному заголовку таблиці і на перетині отримуємо табличне значення G_T коефіцієнта Кохрена. Якщо виконується умова

$$G_p < G_T,$$

то всі порядкові дисперсії визнаються однорідними з вибраним рівнем статистичної значущості α (достовірність $p = 1 - \alpha$). В протилежному випадку гіпотезу відхиляють.

Визначимо розрахункове значення коефіцієнта:

$$G_p = 12,3 / (12,3 + 4 + 2,3 + 1) = 0,43.$$

Відповідно до таблиці коефіцієнтів для $\alpha = 0,05$, $f_1=3 - 1 = 2$, $f_2 = 4$ знаходимо $G_T = 0,77$, $G_T > G_p$, тобто умова виконується.

8.2.3. Перевірка нуль-гіпотези за критерієм Стьюдента

Після перевірки однорідності переходимо до визначення оцінок коефіцієнтів за формулою

$$a_k = \sum_{i=1}^N \frac{y_{ik} x_{ik}}{N},$$

де k - номер вектор-стовпця.

У нашому прикладі маємо:

$$a_0 = (28 + 45 + 7 + 28) / 4 = 25.1;$$

$$a_1 = (-28 + 45 - 7 + 28) / 4 = 11.4;$$

$$a_2 = (-28 - 45 + 7 + 28) / 4 = -7.75;$$

$$a_{12} = (28 - 45 - 7 + 28) / 4 = -0.75.$$

Визначені таким чином коефіцієнти рівняння регресії необхідно оцінити на статистичну значущість. Оцінка проводиться за t -критерієм Стьюдента. Для кожного коефіцієнта a_k обчислюємо коефіцієнт рівняння регресії та перевіряємо відхилення від нуля знайденої оцінки:

$$t_k = \frac{|a_k|}{S\{a_k\}},$$

де $S\{a_k\}$ - оцінка середнього квадратичного відхилення похибки визначення коефіцієнта,

$$S\{a_k\} = \sqrt{S^2\{a_k\}}.$$

Оцінка дисперсії коефіцієнтів, знайдених за експериментальними даними:

$$S^2\{a_k\} = \frac{S^2_B}{N \times m}.$$

Оцінка генеральної дисперсії відтворюваності S^2_B , що характеризує

точність одного вимірювання, є середньою із всіх порядкових дисперсій:

$$S_B^2 = \sum_{i=1}^N \frac{S^2\{y_i\}}{N}.$$

При вибраному рівні статистичної значущості α за таблицями розподілу Стьюдента при числі степенів вільності $f = N(m - 1)$ визначаємо табличне значення коефіцієнта $t_{табл.}$. Це значення порівнюємо з розрахунковим значенням коефіцієнта. Якщо виконується нерівність

$$t_{табл.} > t_k,$$

то приймається нуль-гіпотеза, тобто вважається, що знайдений коефіцієнт a_k є статистично незначним і його потрібно виключити з рівняння регресії.

Для розглянутого прикладу

$$S^2_{\epsilon} = (12,3 + 4 + 2,3 + 1) / 4 = 4,9,$$
$$S^2\{a_k\} = 4,97 / (4 \times 3) = 0,4; \quad S\{a_k\} = 0,64.$$

Визначимо розрахункові значення коефіцієнта Стьюдента:

$$t_0 = 25,1 / 0,64 = 39,2; \quad t_1 = 11,4 / 0,64 = 17,8;$$
$$t_2 = 7,75 / 0,64 = 12,1; \quad t_{12} = 0,75 / 0,64 = 1,17.$$

З таблиць при рівні статистичної значущості $\alpha = 0,05$ і числі степенів вільності $f = 4(3 - 1) = 8$ знайдемо табличне значення коефіцієнта $t_{\tau} = 8,3$. Зіставимо розрахункові значення t_k з табличним t_m . Нерівність виконується для t_{12} . Отже, можна передбачити, що a_{12} статистично незначущий і його можна виключити з рівняння регресії.

Рівняння регресії, що містить статистично значущі коефіцієнти, буде (в кодованій системі)

$$Y' = 25,1 + 11,4 x_1 - 7,75 x_2.$$

8.2.4. Перевірка адекватності за критерієм Фішера

Отримане рівняння регресії необхідно перевірити на адекватність досліджуваному об'єкту. Для цього оцінимо, наскільки відрізняються середні значення y_i вихідної величини, отриманої в точках факторного простору, і значення y'_i , обчислені з рівняння регресії в тих же точках факторного простору. Для цього використовуємо дисперсію адекватності

$$S_{ад}^2 = \frac{m}{N - L} \sum_{i=1}^N (y_i - y'_i)^2,$$

де L - кількість значущих коефіцієнтів.

Адекватність моделі перевіряємо за F -критерієм Фішера:

$$F_p = S_{ад}^2 / S_v^2.$$

Знайдене розрахунковим шляхом F_p порівнюємо з табличним значенням F_m , яке визначається при рівні значущості α і числі степенів вільності $f_{ад} = N - L$ і $f_v = N(m - 1)$. Якщо

$$F_p < F_m,$$

то отримана математична модель з прийнятим рівнем статистичної значущості α адекватна експериментальним даним.

Для розглядуваного прикладу дістанемо:

$$Y'_1 = 25,1 + 11,4 (-1) - 7,75 (-1) = 21,45;$$

$$Y'_2 = 25,1 + 11,4 (+1) - 7,75 (-1) = 44,25;$$

$$Y'_3 = 25,1 + 11,4 (-1) - 7,75 (+1) = 5,95;$$

$$Y'_4 = 25,1 + 11,4 (+1) - 7,75 (+1) = 28,75.$$

Розрахуємо оцінку дисперсії адекватності:

$$S_{ад}^2 = 3[(20,7 - 21,45)^2 + (45 - 44,25)^2 + (6,7 - 5,95)^2 + (28 - 28,75)^2] / (4-3) = 3,38;$$

$$F_p = S_{ад}^2 / S_v^2 = 3,38 / 4,92 = 0,69.$$

Табличне значення коефіцієнта Фішера при рівні статистичної значущості $\alpha = 0,05$ і числі степенів вільності $f_{ad} = 4 - 3 = 1$ і $f_b = 4(3-1)=8$ буде $F_m = 5,38$. Отже, при вибраному рівні статистичної значущості отримана в результаті експерименту модель $Y' = 25,1 + 11,4 x_1 - 7,75 x_2$ адекватна досліджуваному об'єкту.

8.3. Оброблення результатів дробово-факторного моделювання в пакеті Statgraphics

У системі Statgraphics оброблення результатів дробово-факторного експерименту проводиться в такому порядку:

1. Запустити прикладну програму Statgraphics і вибрати опцію "Планування нового експерименту" (рис. 8.2).
2. Скласти матрицю планування для повного трифакторного експерименту з використанням додаткового нульового фактора ($x_0 = 1$) і заповнити таблицю кодованими значеннями x_1, x_2 і x_3 .
3. Перевірити властивості дробово-факторного експерименту: симетричність, нормування, ортогональність і рототабельність.
4. Провести імітаційний експеримент у всіх точках ДФЕ. Визначити значення функції відклику.
5. Активізувати режим досліджень, використовуючи функції Menu Relate Multiply Regressions, і отримати параметри моделі (рис. 8.3).



Рис. 8.2. Вибір режиму побудови нового експерименту

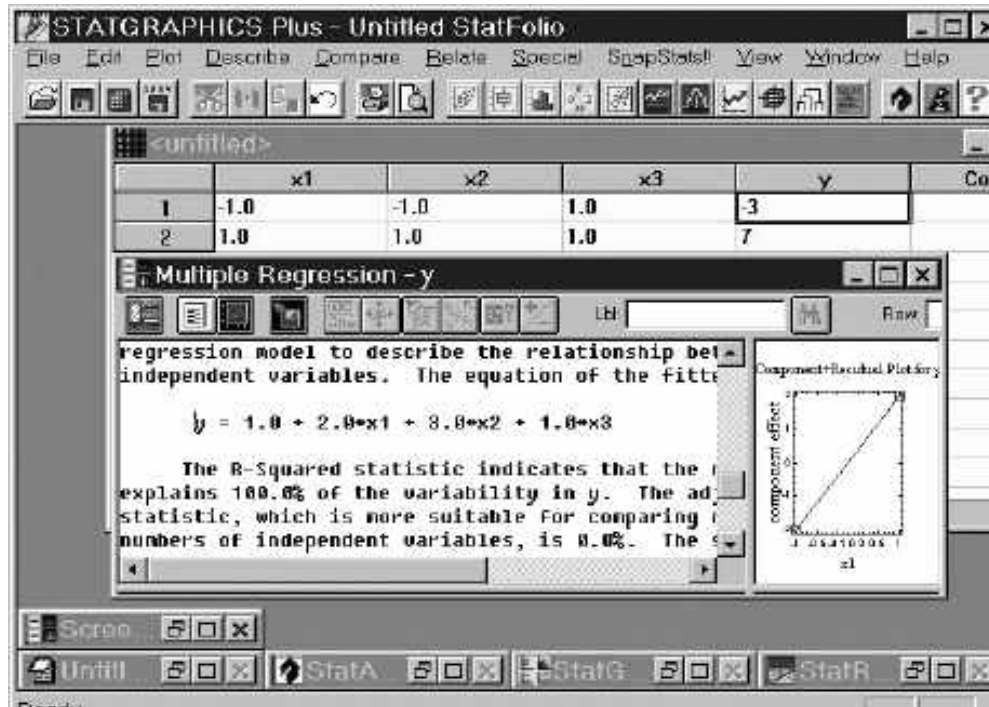


Рис. 8.3. Отримання графічних і числових результатів для ДФЕ

6. Отримати коефіцієнти b_0, b_1, b_2, b_3 рівняння регресії у вигляді $Y = b_0 + b_1 x_1 + b_2 x_2 + b_3 x_3$.

7. Записати загальний вигляд моделі і зарисувати діаграми з коментарями до описаних об'єктів.

8. Виконати статистичне оброблення результатів планування. Обчислити дисперсію отриманих результатів, перевірити однорідність за критерієм Кохрена.

9. Оцінити достовірність отриманих параметрів моделі за критерієм Стюдента.

10. Перевірити адекватність отриманої моделі за критерієм Фішера.

18. Вийти з оболонки Statgraphics з використанням команди Exit у запрошенні командного вікна.

Контрольні запитання

1. Перелічіть умови, необхідні для визначення коефіцієнтів регресії.
2. Дайте визначення повного факторного експерименту.
3. Опишіть відмінність активного експерименту від пасивного.
4. Сформууйте план ДФЕ для двовимірної моделі.
5. Визначіть кількість експериментів ДФЕ для кількості факторів, що дорівнює 5, 7, 8, 9.
6. Запишіть приклад поліноміального подання моделі 2-го порядку з урахуванням квадратичних факторів.
7. Запишіть векторно-матричне подання лінійних моделей.
8. Запишіть приклад базисної функції динамічної моделі 3-го порядку у вигляді ряду Тейлора.
9. Сформулюйте необхідні умови формування вхідних впливів для визначення параметрів моделі.
10. Перелічіть основні методи визначення параметрів моделі за вхідними і вихідними даними.
11. Визначіть коефіцієнти регресії за відомими головними визначником і доповненнями.
12. Запишіть інформаційну матрицю для системи 3-го порядку.

8.4. Моделювання систем при пошуку екстремальної області

8.4.1. Стратегії пошуку області екстремуму

Повний і дробовий факторні експерименти надають досліднику можливість добирати поліноміальні моделі для опису локальних областей поверхні відклику досліджуваної системи. Змінюючи координати базової точки (центру плану), можна вивчити рельєф усієї поверхні відклику в області визначення вхідних змінних.

Цілі такого вивчення можуть бути різними. Так, можливо поставити задачу визначення стану об'єкта при різних комбінаціях значень вхідних змінних. Це також може бути завдання вибору найкращої (за будь-яким критерієм) системи з деякої множини систем, наприклад системи, для якої забезпечується максимальна стійкість до варіації вхідних змінних (вибір схеми LC-генератора гармонічних коливань з мінімальною залежністю частоти від параметрів схеми і зовнішніх факторів).

У більшості випадків мету дослідження тією чи іншою мірою пов'язують з оптимізацією, тобто з пошуком таких значень вхідних змінних, при яких досягається екстремальне значення вихідної змінної показника оптимізації. У цьому випадку потрібно говорити про організацію деякої цілеспрямованої стратегії експерименту, що дозволяє вийти по поверхні відклику в область, розташовану поблизу точки максимуму $Y(X)$ або мінімуму $y(X)$.

Увесь клас таких задач характерний тим, що експериментатора цікавить вже не модель досліджуваного об'єкта, а сама процедура пошуку області простору вхідних змінних, де досягається найкраще значення вихідного параметра.

Для розв'язання задачі оптимізації, тобто для пошуку вектора вхідних змінних, при якому функція досягає екстремуму, можна застосувати два принципово різних способи: а) якщо відома математична модель, що описує функцію відклику тієї частини факторного простору, де розташовано екстремум, то задачу оптимізації розв'язують аналітичним або числовим методом; б) якщо математичний опис відсутній, то здійснюють експериментальний пошук області екстремуму.

У першому випадку знаходять частинні похідні від функції за кожним з факторів і отримують систему рівнянь, розв'язком якої і є шуканий вектор.

У цей час розроблено велику кількість наближених методів оптимізації, причому ефективність кожного з них залежить від вигляду функції, що

оптимізується. Оскільки апріорні відомості про характер функції відклику часто відсутні, то для підвищення надійності результату доцільно використовувати декілька методів.

Якщо функція відклику багатоекстремальна, зі складним рельєфом, то велике значення мають правильний вибір початкової точки, з якої відбувається рух до екстремуму, кроку, з яким виконується рух, і вміле поєднання методів випадкового неспрямованого пошуку з детермінованими методами.

Всі пошукові методи знаходження екстремуму при активному експерименті поділяються на класичні й факторні.

За класичними методами пошукові дослідження для просування до області екстремуму здійснюються шляхом почергового варіювання незалежних змінних. Інші фактори на цей час фіксуються, тобто залишаються незмінними. Ідеї, що знаходяться в основі побудови стратегій пошуку екстремуму, можуть бути при цьому вельми різноманітними, однак існують методи, що претендують на універсальність. Серед таких методів потрібно зазначити методи покоординатного пошуку і градієнтні методи. Всі методи, що входять в ці дві групи, розрізняються лише правилами вибору кроку, з яким здійснюється просування до екстремуму.

Якщо порівняти вираз для градієнта з розкладанням функції відклику в околі точки X_0 у ряд Тейлора і обмежитися лінійними членами ряду, то можна зробити висновок про те, що координати вектор-градієнта відносно базової точки збігаються з коефіцієнтами B лінійної моделі. Суть стратегії всіх градієнтних методів полягає в тому, що на кожному етапі просування до екстремуму навколо вибраної точки проводяться пробні дослідження, за результатами яких оцінюється напрям градієнта у факторному просторі, після чого з вибраним кроком здійснюється просування в нову базову точку, яка буде ближче до екстремуму, ніж вихідна. Таким чином, у градієнтному методі координати нової точки на $(i+1)$ -му кроці знаходять за формулою $X(i+1) = X(i) + \Delta y(X_i)$. Ті чи інші градієнтні методи розрізняються лише правилами вибору кроку X . Вибір дуже малого кроку може призвести до великої похибки оцінок складових градієнта, а отже, й до істотної помилки у виборі напрямку руху по поверхні відклику.

При великому кроці неможливо врахувати особливості рельєфу поверхні. Стратегію пошуку області екстремуму для градієнтних методів подано на рис. 8.4.

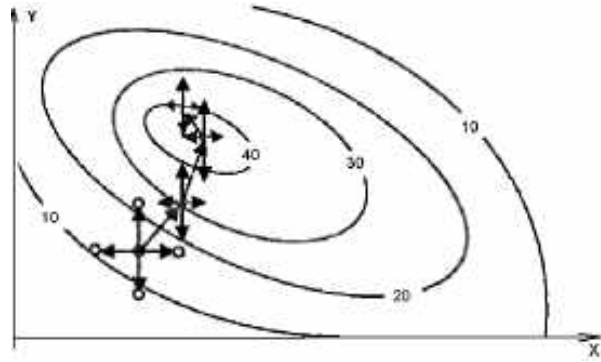


Рис. 8.4. Стратегія пошуку екстремуму градієнтними методами

Пробні дослідження в кожній точці дають значення функції відклику, за якими обчислюються складові градієнта. Напрямок градієнта при вибраному кроці і має бути таким, щоби його проекції на осі координат були пропорційні відповідним коефіцієнтам лінійного рівняння моделі об'єкта. Для цього крок і помножують на відповідне значення коефіцієнта Δu , тобто одержують значення V . Отримані значення використовують для знаходження координат нової точки, в околі якої відбувається нова серія дослідів і визначається новий напрям градієнта. Просування до екстремуму продовжується доти, доки всі значення V не стануть менше заданої величини E . Градієнтний метод швидше збігається, але більше зазнає випадкових перешкод і помилок при визначенні напрямку градієнта.

8.5. Факторні методи пошуку області екстремуму

Одним з найбільш широко використовуваних факторних методів є метод крутого сходження (метод Бокса – Уілсона). При реалізації цього методу просування до області екстремуму, як і в класичних градієнтних методах, відбувається в напрямі градієнта, знайденого на основі пробних дослідів в околі базової точки. Однак якщо в класичних методах проводяться по два пробних дослідження з кожного боку базової точки шляхом почергового варіювання кожної змінної при фіксованих інших, то при реалізації методу крутого сходження (МКС) дослідження ставляться відповідно до матриці плану повного або дробового факторного експерименту. У цьому випадку в оцінюванні коефіцієнта моделі, а тже, і кожної складової градієнта беруть участь всі $N=2^i$ точок експерименту. Оцінки виходять більш точними і завадостійкими. Це дає

можливість у знайденому напрямі градієнта здійснювати не один, як при класичних методах, а декілька пробних кроків.

Урахування коефіцієнтів при взаємодіях факторів дозволяє отримати інформацію про крутість поверхні відклику в районі базової точки, а отже, й більш правильно вибрати крок, з яким здійснюється просування до екстремуму. Крім цього, статистична оцінка результатів факторного експерименту значно підвищує їхню надійність.

Крокова процедура пошуку області екстремуму при МКС виконується за декілька циклів i . З центром у початковій точці $X_1 \{x_{11}, x_{21}, x_{31}, \dots\}$ проводиться повний або дробовий факторний експеримент для визначення складових вектора градієнта у вигляді оцінок коефіцієнтів рівняння лінійної поліноміальної моделі $Y = B_0 + B_1X_1 + B_2X_2 + \dots + B_nX_n$.

Побудова моделі містить в собі всі етапи ПФЕ, тобто перевірку однорідності дисперсій, обчислення, а також перевірку значущості коефіцієнтів і перевірку адекватності лінійного наближення в околі початкової точки. Потрібно врахувати, що рух по градієнту (при пошуку максимуму) або проти градієнта (при пошуку мінімуму) буде більш ефективним, якщо значення коефіцієнтів B приблизно одного порядку.

Якщо коефіцієнти відрізняються на порядок і більше, то вигідніше знову поставити експеримент, змінивши інтервали варіювання, а не починати рух по градієнту.

Траєкторія крутого сходження подається в натуральних значеннях змінних. Координати нових точок визначаються за кожною вхідною змінною додаванням до початкового рівня x_n значення, що дорівнює кроку n . Вибір кроку X доцільно зробити спочатку для деякої базової змінної, що створює максимальний приріст функції в напрямі градієнта, тобто $X_p = F(Bx_p)$, де p - інтервал варіювання, вибраний для базової змінної.

За знайденим значенням і знаком кроку визначаються координати нових точок факторного простору. Якщо x_n - n -та координата точки траєкторії на i -му кроці просування, то

$$x_n = x_0 + B \times X_p \quad (n = 1, 2, \dots).$$

Для деяких точок обчислюються передбачувані значення для функції. Визначаючи ці значення, слід пам'ятати, що в рівняння моделі необхідно

підставляти кодовані значення змінних, а для визначення нових точок траєкторії – натуральні значення змінних.

Координати деяких точок (приблизно через 3-4 кроки) реалізуються на об'єкті для експериментальної перевірки змін відклику у вибраному напрямі. Якщо в деякій точці передбачуваному збільшенню (зменшенню) значення відклику відповідає зменшення (збільшення) значення вихідної змінної на об'єкті, то рух у вибраному напрямі потрібно припинити. Координати точки з найкращим значенням вихідної змінної приймаються за новий нульовий рівень, знову проводиться ПФЕ або ДФЕ і визначається новий напрям. Оскільки крутість функції в міру наближення до екстремуму спадає, то для кожного подальшого наближення крок необхідно зменшувати. Процедура пошуку здійснюється доти, доки значення коефіцієнтів рівняння регресії не стануть порівнювати з помилками їх визначення і лінійна апроксимація не стане неадекватною.

Оскільки поверхня відклику в заданих межах зміни рівнів факторів може мати декілька локальних екстремумів, то за результатами однієї реалізації процедури Бокса – Уілсона не можна говорити про встановлення глобального екстремуму. Тому серію експериментів слід повторити декілька разів, починаючи рух з випадково вибраних і рівномірно розподілених точок факторного простору. Метод крутого сходження особливо ефективний у випадках, коли функція унімодальна, і при великих помилках експерименту. Застосування методу крутого сходження неефективне в околі екстремуму, оскільки в цій області коефіцієнти B , зменшуючись, можуть стати менше від помилок в їх визначенні, і через невизначеність, що виникла, збільшується ймовірність руху в помилковому напрямі. У зв'язку з цим в області, близькій до екстремуму, потрібно застосовувати плани більш високого (зазвичай другого) порядку.

9. МОДЕЛЮВАННЯ В ІНТЕГРОВАНОМУ ПАКЕТІ STRATUM

Серед багатьох комп'ютерних програм, призначених для створення і дослідження візуальних моделей технічних об'єктів і систем, особливої уваги заслуговує пакет STRATUM-2000, який розроблено для навчального процесу. Програму STRATUM також розраховано на науковців та інженерів, але вона досить легко освоюється студентами та школярами і може бути застосована як для моделювання складних динамічних об'єктів і систем, так і для проведення простих віртуальних експериментів.

Інтегрована оболонка пакета STRATUM є додатком з багатовіконним інтерфейсом. Інтерфейс припускає наявність головного вікна програми і довільного числа дочірніх вікон. Назва головного вікна містить найменування пакета і шлях до відкритого проекту. Термін "проект" означає сукупність даних, що належать до однієї моделі. Дані проекту зберігаються в декількох файлах, розташованих у певній папці. Головний файл проекту має розширення *.str.

Наявність декількох дочірніх вікон дозволяє одночасно працювати з декількома об'єктами проекту або з різними аспектами одного і того ж об'єкта. Головне вікно містить меню і панель кнопок або інструментальну панель. З усіх дочірніх вікон лише одне в кожен момент часу має фокус введення, заголовок цього вікна виділено кольором (рис. 9.1). Щоби перенести фокус введення в це вікно, потрібно натиснути ліву клавішу "миші" на його зображенні.

Дочірні вікна інтегрованої оболонки за типом візуалізації належать до двох типів: вікна проекту, класу і редагування системи рівнянь і вікна редакторів схем. У дочірньому вікні можна виділити один або декілька об'єктів. На дереві об'єктів виділений об'єкт позначається кольором, а на схемі - спеціальними квадратами по контуру цього об'єкта (див. рис 9.1).

По натисненні на зображенні об'єкта правої клавіші "миші" з'являється впливаюче меню, яке містить команди цього об'єкта.

Значна частина дій у пакеті виконується методом «drag and drop», коли необхідний об'єкт виділяється натисканням лівої клавіші "миші", а потім переміщається за допомогою "миші" з притиснутою лівою клавішею в потрібну область того ж чи іншого дочірнього вікна після відпускання клавіші графічного

маніпулятора. Під час переміщення курсор графічного маніпулятора індикує області, в які можна перемістити цей об'єкт.

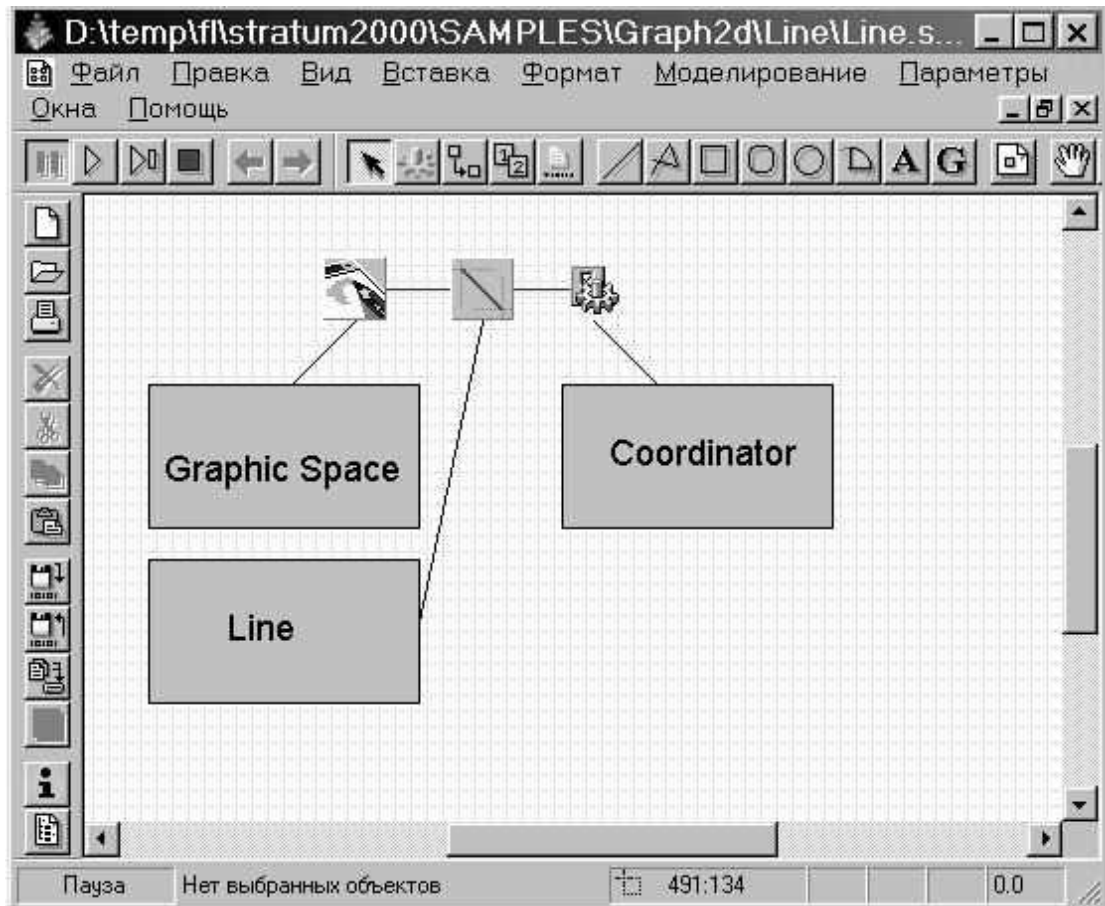


Рис. 9.1. Проект «mex_object»

9.1. Створення проекту

Входимо в інтегровану оболонку пакета і натискаємо кнопку "новий" або виконуємо команду головного меню «Проект_Новий». У діалозі вибираємо шлях до папки проекту (наприклад, "D:\student\"), вводимо ім'я проекту (наприклад, «mex_object») і натискаємо кнопку «Створити». Після цього з'явиться папка " D:\student\mex_object" і в ній – файл бази даних проекту «mex_object.str».

Передбачається, що для нового проекту відкривається нова папка, якщо використовується існуюча, то всі дані в ній будуть знищені. За допомогою

команди «Проект_Зберегти_як» можна зберігати в тій же папці різні модифікації цього проекту (наприклад, `mex_object1.str`).

Зауваження. Надалі в цій папці з'являться ще два файли - «`mex_object.ini`» і «`mex_object_em.ini`», – в яких зберігаються відповідно установки проекту і виконуваної моделі.

Після цього в інтегрованому середовищі з'являться вікна (див. рис. 9.1).

9.1.1. Вікно керування проектом містить дерево основних складових проекту. До складових проекту належать: класи блоків, глобальні константи, глобальні процедури і функції, віртуальний стенд й імпортовані бібліотеки класів. За допомогою впливаючого по натисненні на праву кнопку "миші" меню можна додавати нові компоненти, видаляти і редагувати існуючі. У новому проекті за замовчуванням додано клас безперервного блока «`mex_object`», при цьому ім'я класу збігається з ім'ям проекту.

9.1.2. Вікно віртуального стенда містить структурну схему системи, що моделюється, тобто екземпляри блоків і зв'язки між ними. За замовчуванням у віртуальний стенд поміщено екземпляр класу «`Mex_object`» з ім'ям «`Mex_object_1`», а наступний екземпляр цього класу в цій схемі отримає за замовчуванням ім'я «`Mex_object_2`». Керувати структурними компонентами можна з допомогою впливаючого по натисненні правої клавіші "миші" меню. Щоби додати новий блок, досить активізувати потрібний клас, буксирувати його у вікно структури і кинути в потрібному місці.

Розглядувана модель маятника є ізольованою системою, і загалом немає особливої необхідності в структурній схемі й блоках. Проте ця версія пакета орієнтована на блокове моделювання, і тому ізольована система подається як блок без входів і виходів, а її модель – у вигляді віртуального стенда з примірником такого блока.

9.1.3. Вікно редагування доданого за замовчуванням класу (класу «Маятник») містить дерево складових опису класу. Оскільки цей блок передбачається безперервним, то за замовчуванням в нього додано порожню систему рівнянь з ім'ям «`Система_рівнянь_1`».

У цій версії пакета повністю безперервна система розглядається як випадок гібридного автомата, головна карта поведінки якого містить тільки один вузол, якому і приписано відповідну систему рівнянь. Тому якщо відкрити головну карту поведінки такого класу (з допомогою подвійного клацання "миші" на відповідному вузлі дерева), то можна побачити, що вона складається з єдиної точки Init .

9.1.4. Вікно редагування системи рівнянь, відкрите на порожній блок «Система_рівнянь_1».

Розглянемо модель математичного маятника. Це модель ізольованої системи. Тому для неї як основа підходить клас, що створюється за замовчуванням при відкритті нового проекту. У нього тепер необхідно додати необхідні змінні й рівняння. Цей проект назвемо «Маятник.str». Шлях до нього – «E:\ Mex_object.str».

9.1.5. Введення змінних, параметрів і констант. Переходимо до вікна редагування класу «Object», виділяємо в дереві об'єктів вузол «Параметри», натисканням правої клавіші "миші" викликаємо впливаюче меню і виконуємо команду «Додати». Те ж саме можна зробити за допомогою команди «Редагувати / Додати» головного меню або кнопки "+" на панелі кнопок.

У діалозі вводимо ідентифікатор змінної *Color*, залишаємо заданий за замовчуванням тип *PenColor*, задаємо початкове значення 0 і коментар (рис. 9.2).

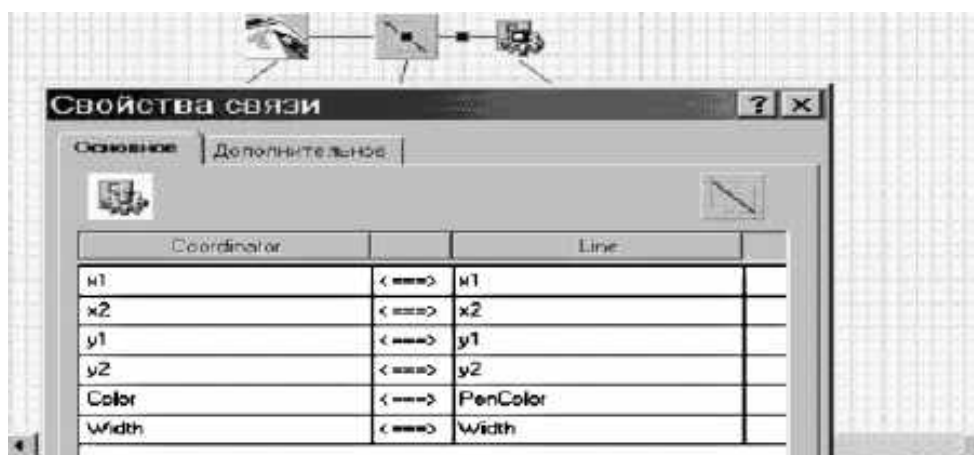


Рис. 9.2. Редагування класу

Аналогічним чином додаємо параметр X1. Далі виділяємо вузол «Внутрішні змінні» і додаємо змінні X2,Y1, Y2 і константу With=50 (рис. 9.3).

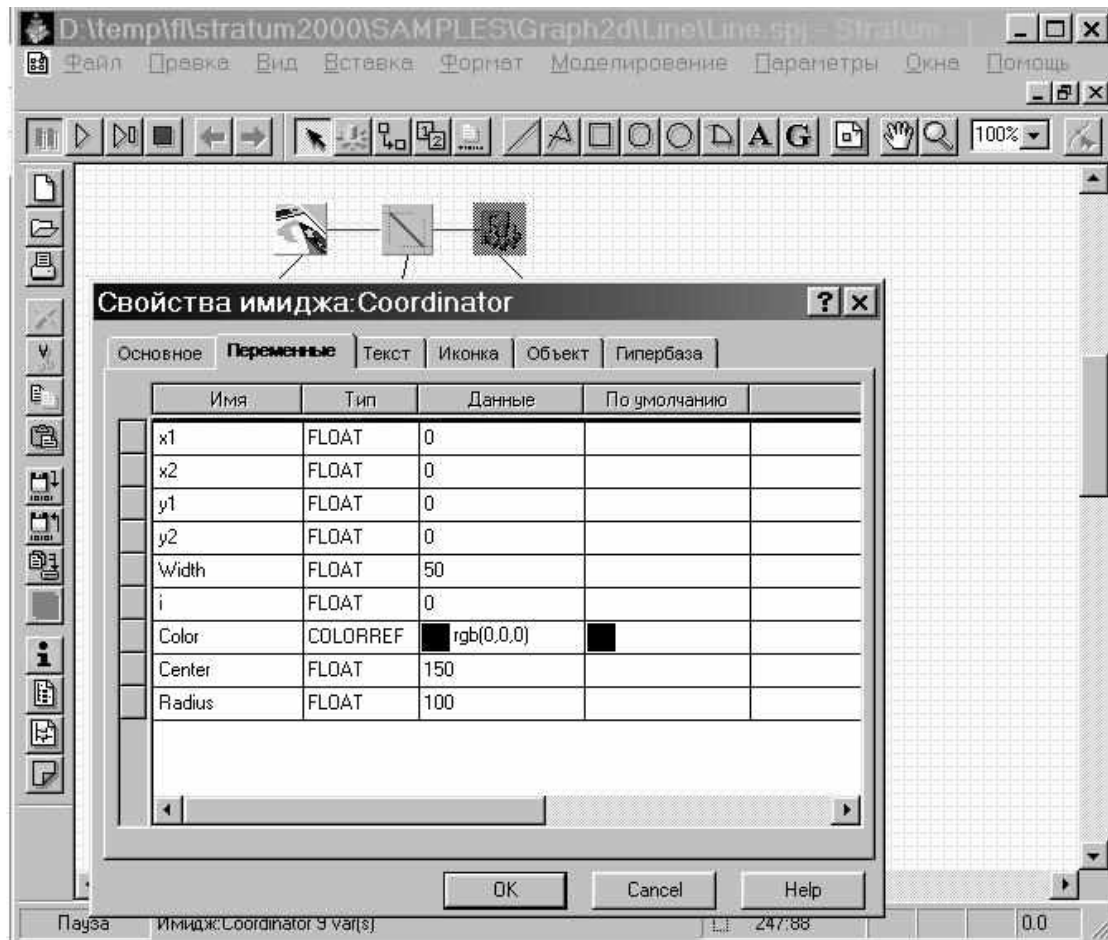


Рис. 9.3. Додавання параметрів

Змінити введені визначення можна, двічі клацнувши на них "мишею", а також за допомогою команди «Редагувати» впливаючого меню, команди «Редагувати / Змінити» головного меню або кнопки на панелі. Можна редагувати визначення як текст за допомогою команди «Редагувати як текст» головного меню. На рис. 9.3 в режимі текстового редагування знаходиться визначення змінної X1. Завершення редагування відбувається за клавішею «Enter», скасування редагування – шляхом натискання клавіші «Esc».

Можна видалити введені визначення за допомогою команди «Видалити» впливаючого меню, команди «Редагувати_Видалити» головного меню або кнопки "Дел" на панелі кнопок.

9.1.6. Введення рівнянь. У вікні редагування рівнянь за допомогою подвійного клацання "миші" на вузлі «Рівняння» або команди «Змінити» впливаючого меню викликаємо спеціальний редактор формул, який дозволяє вводити математичні вирази в близькому до природного вигляді, і вводимо необхідні рівняння (рис. 9.4).

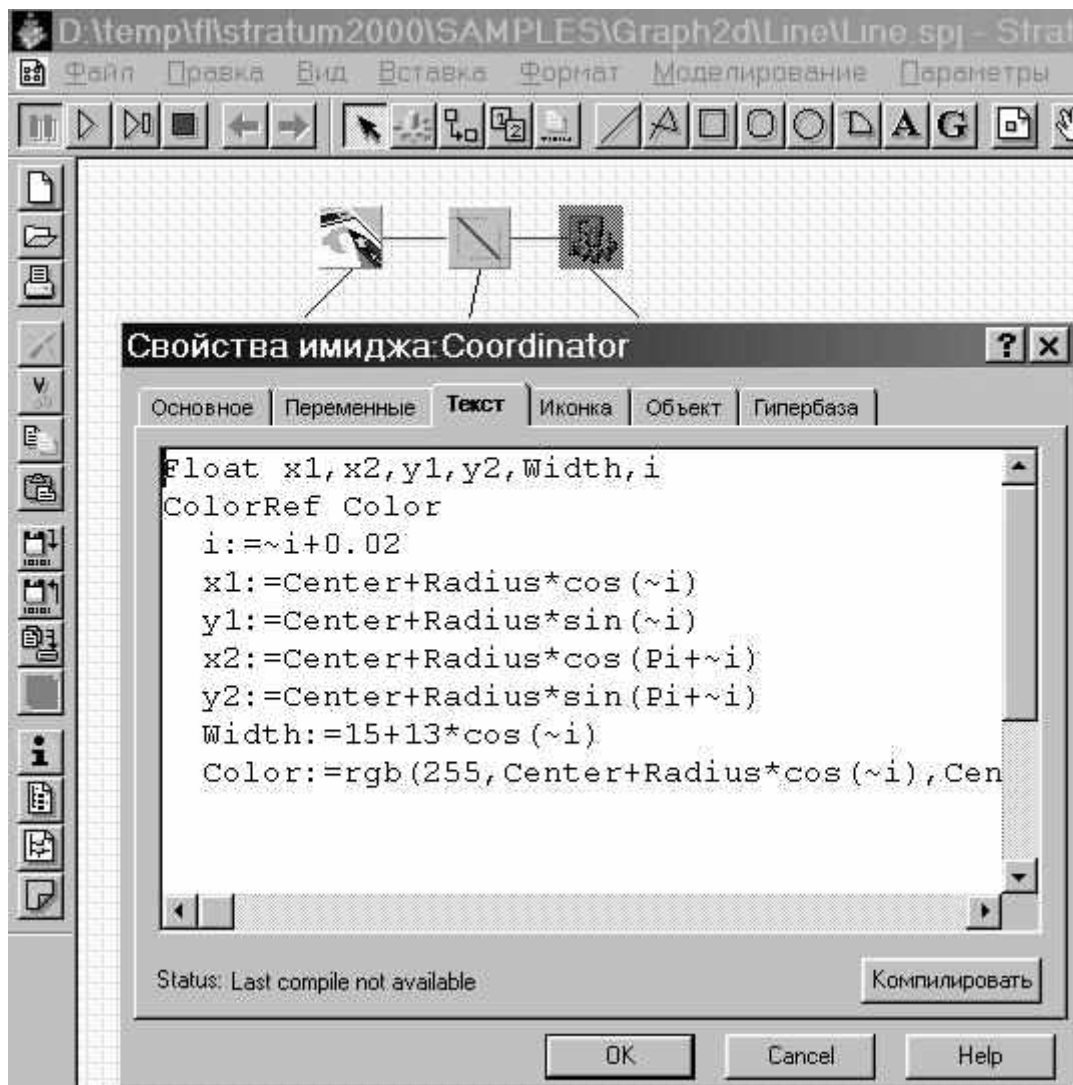


Рис. 9.4. Введення рівнянь

Система рівнянь може включати звичайні диференціальні рівняння першого порядку і алгебраїчні рівняння. Форма введення рівнянь в цій версії пакета досить вільна. Наприклад, останні два рівняння можна подати у вигляді

$$\begin{aligned} \text{Width} &:= 15 + 13 * (\text{Cos}(i+1)); \\ \text{Color} &:= \text{RGB}(225, \text{Center} + \text{Radius} * \text{Cos}(i+1)). \end{aligned}$$

Така форма рівнянь потребує від пакета значно більше зусиль, ніж подана на рис. 9.4. Аналітично дозволені рівняння за відсутності в них циклів зводяться просто до послідовності операторів присвоювання.

Якщо не вказати, які змінні є шуканими (оператор find), то транслятор спробує визначити їх сам, виходячи з контексту завдання: шуканими не можуть бути параметри, константи і входи. Транслятор також прагне зробити шуканими інтегровані змінні, що стоять в лівих частинах формул X і Y . Якщо транслятор не може вибрати шукані змінні, то він викликає діалог і пропонує зробити це вручну.

Крім рівнянь і формул в систему можуть бути включені необхідні коментарі. Коментар починається символом «-» і триває до кінця рядка.

9.1.7. Додаткове введення рівнянь. Розглянемо додаткову модель: одновимірний рух матеріальної точки з масою m під дією допустимої сили F і в'язкого тертя з коефіцієнтом k . Шлях до готового прикладу «Model\Obj1.str»).

Рівняння руху може бути записано в природній «фізичній» формі:

$$m * x'' + k * x' = F.$$

Примусова сила подається на вхід з генератора синусоїдального сигналу (рис. 9.5), тому шуканою змінною є змінна x .

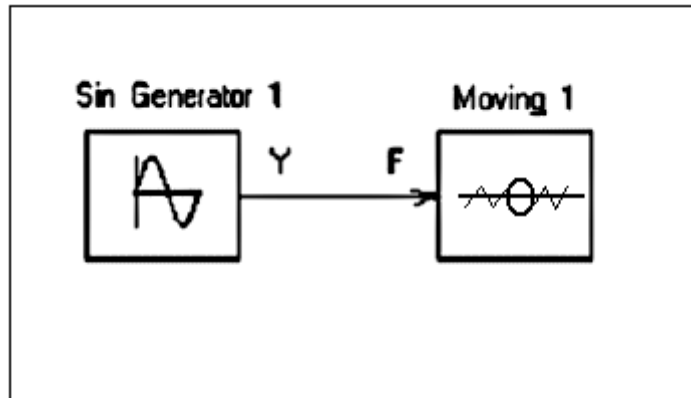


Рис. 9.5. Генератор синусоїдального сигналу

Транслятор сам перетворює це рівняння до «канонічної» форми, необхідної для числового розв'язання. Можна подивитися на це подання за допомогою команди «Показати рівняння в канонічній формі» впливаючого меню. Для цього рівняння буде мати такий вигляд:

$$\begin{aligned} d(x) / dt &= x'; \\ d(x') / dt &= x''; \\ m x'' &= F - m \mu x'. \end{aligned}$$

Перед двокрапкою показано змінну, яка визначається з цього рівняння. У цій версії пакета похідні видимі тільки в межах цієї системи рівнянь (це означає, що не можна, наприклад, використовувати швидкість у діях переходу або в тілі функції).

Не змінюючи рівняння руху, зробимо вхідною змінною x , а не F (рис. 9.6).

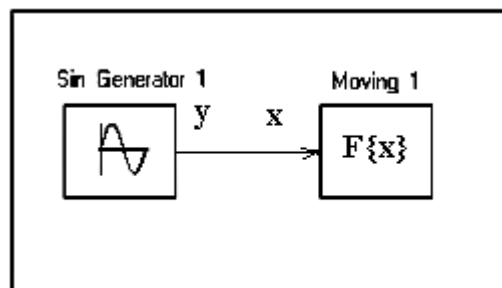


Рис. 9.6. Вхідна змінна x

Тепер шуканою змінною є сила F (тобто ми розв'язуємо обернену задачу: яку силу потрібно прикладати, щоби тіло рухалося). Однак для цього завдання ми будемо мати зовсім іншу «канонічну» систему рівнянь:

$$x' = d(x) / dt;$$

$$x'' = d(x') / dt;$$

$$m x'' = F - \mu x'.$$

Для визначення сили необхідне подвійне диференціювання змінної x . Оскільки у цій версії пакета аналітичні перетворення не підтримуються, зробити це можна тільки за допомогою числового диференціювання. Це дуже погано для більшості числових методів, і тому транслятор попереджає користувача.

9.2. Створення і запуск виконуваної моделі

Створення моделі проводиться за допомогою команди «Модель / Створити» головного меню або кнопки на панелі. Запуск і створення моделі виконується за допомогою команди «Модель / Пуск» головного меню або кнопки на панелі.

Пакет STRATUM належить до категорії компілюючих. Для елементів опису модельованої системи генерується "код" на мові програмування, який потім компілюється в машинний код і зв'язується з Run Time Library (RTL) STRATUM за допомогою штатного компілятора командного рядка цієї мови. У цій версії пакета як проміжна мова використовується Pascal.

Пакет STRATUM може створювати моделі двох видів: візуальну у формі додатка Win32 і «приховану» у формі Win32 DLL. Далі розглядається тільки робота з візуальними моделями.

При створенні моделі спочатку можна побачити вікно генератора коду, а потім – одне за іншим два вікна консольних додатків - компілятора ресурсів (необхідний для приєднання іконки класу) і компілятора Object Pascal Delphi. Вся робота з проміжним кодом проводиться в локальній папці ...\\Tmp (у цьому прикладі – в D:\\Student), в ній формується файл model.exe. Зазвичай це триває кілька секунд. При виході з інтегрованого середовища весь вміст цієї папки

автоматично знищується. Тому, якщо треба залишити проміжний код для незалежного використання (наприклад, щоби використовувати діалоговий налагоджувач Delphi), необхідно скористатися кнопкою або командою «Модель/Створити». Для того, щоби зберегти модель як незалежну від оболонки виконувану програму, треба скористатися командою «Модель/Зберегти_як» головного меню.

Якщо папка проекту недоступна для запису (розташована на CD або закрита на запис для користувача), то всі тимчасові файли і файл виконуваної моделі формуються в робочій папці Windows (наприклад, C:\Temp).

Виконання команди «Пуск» завершується запуском створеної програми model.exe. У будь-який момент можна перервати виконання моделі з інтегрованого середовища (наприклад, у випадку її зависання) за допомогою кнопки або команди «Модель/Стоп».

9.3. Експерименти з візуальною моделлю

На рис. 9.7 показано головне вікно візуальної моделі після першого запуску. Візуальна модель, так само як і інтегрована оболонка, є багатовіконним додатком. У заголовку головного вікна вказується ім'я виконуваної моделі, яке можна змінити у вікні віртуального стенда інтегрованої оболонки (за замовчуванням це ім'я проекту). У лівій частині інструментальної панелі відображається поточне значення модельного часу (початкове значення 0). У нижній частині головного вікна виводиться інформація про поточні типи сукупної системи рівнянь і загальна кількість рівнянь (у випадку системи з перемиканнями вони можуть змінюватися при прогоні моделі).

У лівому верхньому куті розташовано вікно віртуального стенда, яке відображає структуру моделі і є «провідником» по ній. Для всіх блоків, присутніх у моделі (в цьому випадку є один блок Object_1), автоматично відкриваються вікна змінних. У заголовку дочірніх вікон вказується блок або елемент блока, який відображається в цьому вікні. Характерно, що після створення екземпляра цього пристрою його параметри набули вказаних значень і фазові змінні ініціалізовані зазначеними виразами.

9.3.1. Запуск моделі. Запустимо виконання моделі за допомогою кнопки на панелі головного вікна моделі або командою «Моделювання / Пуск» головного меню. Почнуть змінюватися модельний час і значення фазових змінних. Зупинимо виконання за допомогою кнопки або команди «Моделювання / Стоп» і потім закінчимо цей прогін командою «Моделювання / Рестарт». У результаті цих дій цей екземпляр випробуваної системи буде знищено і створено новий, знову з початковими значеннями змінних.

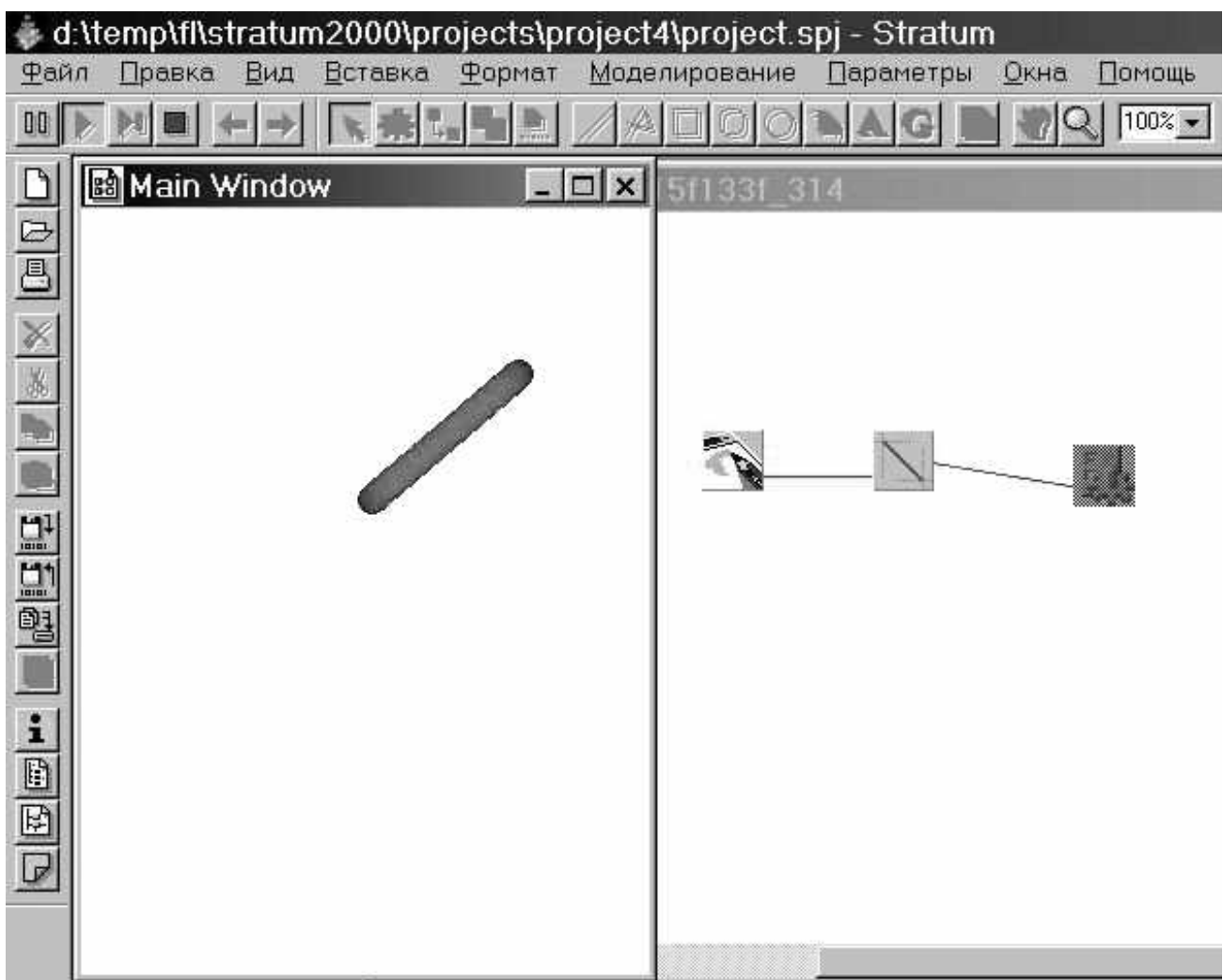


Рис. 9.7. Запуск моделі

Д.1. Моделювання статичного об'єкта мовою Паскаль

```

Program samolet;

{Подключение стандартных модулей}
uses graph,crt;

{Описание типов переменных}
type
  masiv1 = array [0..1,0..9] of word;
  masiv2 = array [0..1,0..5] of word;

{Задание массива точек объекта}
const XY1 : array [0..9,0..1] of word = ((100,100),
                                          (110,100),
                                          (130,110),
                                          (150,110),
                                          (160,120),
                                          (180,130),
                                          (160,150),
                                          (110,150),
                                          (110,130),
                                          (100,100));

{Задание массива точек дополнительного объекта}
const XY2 : array [0..5,0..1] of word = ((110,150),
                                          (110,140),
                                          (130,140),
                                          (135,145),
                                          (130,150),
                                          (110,150));

{Описание переменных}
var
  GrDr,GrMd : integer;
  x11,y11,x12,y12,x21,y21,x22,y22,i,j : integer;

{Тело программы}
BEGIN
  {Очистка экрана}
  clrscr;

```

```

{Определение номера, установленного в системе драйвера}
DetectGraph(GrDr,GrMd);

{Инициализация графического режима}
InitGraph(GrDr,GrMd,'');

{Установка цвета фона окна}
setbkcolor(2);

{Определение начальной координаты объекта}
x11:=XY1[0,0];y11:=XY1[0,1];

{Рисование начальной координаты объекта}
moveto(x11,y11);

{Цикл рисования точек объекта}
for j := 0 to 9 do
begin
  x12:=XY1[j,0];y12:=XY1[j,1];
  lineto(x12,y12);
  delay(1000) {Задержка}
end;

{Определение начальной координаты дополнительного
объекта}
x21:=XY2[0,0];y21:=XY2[0,1];
{Рисование начальной координаты дополнительного
объекта}
moveto(x21,y21);

{Рисование массива точек дополнительного объекта}
for j := 0 to 5 do
begin
  x22:=XY2[j,0];y22:=XY2[j,1];
  lineto(x22,y22);
  delay(1000) {Задержка}
end;

{Ожидание нажатия клавиши}
readln;
{Закрытие графического режима}
CloseGraph;
END.

```

Д.2. Моделирование рухомого об'єкта мовою Паскаль

```
{Моделирование движения объекта }
Program samolet;

{Подключение стандартных модулей}
uses graph,crt;

{Описание типов переменных}
type
masiv1 = array [0..1,0..9] of word;
masiv2 = array [0..1,0..5] of word;

{Задание массива точек объекта}
const XY1 : array [0..9,0..1] of word = ((100,100),
                                         (110,100),
                                         (130,110),
                                         (150,110),
                                         (160,120),
                                         (180,130),
                                         (160,150),
                                         (110,150),
                                         (110,130),
                                         (100,100));

{Задание массива точек дополнительного объекта}
const XY2 : array [0..5,0..1] of word = ((110,150),
                                         (110,140),
                                         (130,140),
                                         (135,145),
                                         (130,150),
                                         (110,150));

{Описание переменных}
var
  GrDr,GrMd : integer;
  x1_1,x1_2,x2_1,x2_2,x11,y11,x12,y12,x21,y21,x22,y22,i,j :
integer;
  label M,N;

{Тело программы}
BEGIN
  clrscr; {Очистка экрана}
```

```

{Определение номера, установленного в системе драйвера}
DetectGraph(GrDr,GrMd);

{Инициализация графического режима}
InitGraph(GrDr,GrMd,'');

{Установка цвета фона окна}
setbkcolor(2);

{Определение начальной координаты объекта}
x11:=XY1[0,0];y11:=XY1[0,1];
{Рисование начальной координаты объекта}
moveto(x11,y11);

{Цикл рисования точек объекта}
for j := 0 to 9 do
  begin
    x12:=XY1[j,0];y12:=XY1[j,1];
    lineto(x12,y12);
    delay(1000) {Задержка}
  end;

{Определение начальной координаты дополнительного
объекта}
x21:=XY2[0,0];y21:=XY2[0,1];

{Рисование начальной координаты дополнительного объекта}
moveto(x21,y21);

{Рисование массива точек дополнительного объекта}
for j := 0 to 5 do
  begin
    x22:=XY2[j,0];y22:=XY2[j,1];
    lineto(x22,y22);
    delay(1000) {Задержка}
  end;

{Моделирование динамики полета}
for i := 0 to 500 do begin
{Изменение связанной системы координат объекта}
x11:=XY1[0,0]+i;y11:=XY1[0,1];
{Рисование первой точки объекта} moveto(x11,y11);
{Цикл рисования массива точек объекта}
for j := 0 to 9 do
  begin

```

```

    x12:=XY1[j,0]+i;y12:=XY1[j,1];
    lineto(x12,y12);
    delay(1) {Задержка}
end;

{Изменение связанной системы координат дополнительного
объекта}
    x21:=XY2[0,0]+i;y21:=XY2[0,1];
    moveto(x21,y21);

{Цикл рисования массива точек дополнительного объекта}
for j := 0 to 5 do
    begin
        x22:=XY2[j,0]+i;y22:=XY2[j,1];
        lineto(x22,y22);
        delay(1) {Задержка}
    end;
    ClearDevice; {Очистка графического экрана}
end;
readln; {Ожидание нажатия клавиши}
CloseGraph; {Закрытие графического режима}
END.

```

Д.3. Моделирование относительного движения объекта на языке Паскаль

```
{Моделирование подвижного объекта с отделением
дополнительного объекта}
Program samolet;

{Подключение стандартных модулей}
uses graph,crt;

{Описание типов переменных}
type
masiv1 = array [0..1,0..9] of word;
masiv2 = array [0..1,0..5] of word;

{Задание массива точек объекта}
const XY1 : array [0..9,0..1] of word = ((100,100),
                                         (110,100),
                                         (130,110),
                                         (150,110),
                                         (160,120),
                                         (180,130),
                                         (160,150),
                                         (110,150),
                                         (110,130),
                                         (100,100));

{Задание массива точек дополнительного объекта}
const XY2 : array [0..5,0..1] of word = ((110,150),
                                         (110,140),
                                         (130,140),
                                         (135,145),
                                         (130,150),
                                         (110,150));

{Описание переменных}
var
  GrDr,GrMd : integer;
  x11,y11,x12,y12,x21,y21,x22,y22,i,j : integer;
  label M,N,O;
{Тело программы}
BEGIN
  {Очистка экрана}
  clrscr;
```

```

{Определение номера, установленного в системе драйвера}
DetectGraph(GrDr,GrMd);

{Инициализация графического режима}
InitGraph(GrDr,GrMd,'');

{Установка цвета фона окна}
setbkcolor(2);

{Определение начальной координаты объекта}
x11:=XY1[0,0];y11:=XY1[0,1];

{Рисование начальной координаты объекта}
moveto(x11,y11);

{Цикл рисования точек объекта}
for j := 0 to 9 do
  begin
    x12:=XY1[j,0];y12:=XY1[j,1];
    lineto(x12,y12);
    delay(1000) {Задержка}
  end;

{Определение начальной координаты дополнительного
объекта}
x21:=XY2[0,0];y21:=XY2[0,1];
{Рисование начальной координаты дополнительного объекта}
moveto(x21,y21);

{Рисование массива точек дополнительного объекта}
for j := 0 to 5 do
  begin
    x22:=XY2[j,0];y22:=XY2[j,1];
    lineto(x22,y22);
    delay(1000)
  end; {Задержка}

{Моделирование динамики полета}
for i := 0 to 500 do
  begin

{Изменение связанной системы координат объекта}
    x11:=XY1[0,0]+i;y11:=XY1[0,1];
    {Рисование первой точки объекта}
    moveto(x11,y11);

```

```

{Цикл рисования массива точек объекта}
  for j := 0 to 9 do
    begin
      x12:=XY1[j,0]+i;y12:=XY1[j,1];
      lineto(x12,y12);
      delay(1) {Задержка}
    end;

{Изменение связанной системы координат дополнительного
объекта}
  x21:=XY2[0,0]+i;y21:=XY2[0,1];
  moveto(x21,y21);
{Цикл рисования массива точек дополнительного объекта}
  for j := 0 to 5 do
    begin
      x22:=XY2[j,0]+i;y22:=XY2[j,1];
      lineto(x22,y22);
      delay(1) {Задержка}
    end;

{Моделирование условия срабатывания отделения
дополнительного объекта по нажатию клавиши}
  if keypressed=true then goto M else goto N;
{Моделирование относительного движения}
M:
{Изменение связанной системы координат дополнительного
объекта ускорено по отношению к координатам объекта}
  x21:=XY2[0,0]+i*2;y21:=XY2[0,1];
  moveto(x21,y21);
  for j := 0 to 5 do
    begin
      x22:=XY2[j,0]+i*2;y22:=XY2[j,1];
      lineto(x22,y22);
      delay(1)
    end;

  N:
  {Изменение связанной системы координат дополнительного
объекта равноускорено по отношению к координатам
объекта}
  x21:=XY2[0,0]+i;y21:=XY2[0,1];
  moveto(x21,y21);
  for j := 0 to 5 do
    begin
      x22:=XY2[j,0]+i;y22:=XY2[j,1];

```



```
        lineto(x22,y22);
        delay(1)
    end;
    {Очистка графического экрана}
    ClearDevice;
end;
readln; {Ожидание нажатия клавиши}
{Закрытие графического режима}
CloseGraph;
END.
```

Д.4. Моделирование условных движений объекта языком Паскаль

```
{Моделирование подвижного объекта и совпадения с целью}  
Program samolet;
```

```
{Подключение стандартных модулей}  
uses graph,crt;
```

```
{Описание типов переменных}  
type
```

```
masiv1 = array [0..1,0..9] of word;  
masiv2 = array [0..1,0..5] of word;  
masiv3 = array [0..1,0..4] of word;
```

```
{Задание массива точек объекта}
```

```
const XY1 : array [0..9,0..1] of word = ((100,100),  
                                           (110,100),  
                                           (130,110),  
                                           (150,110),  
                                           (160,120),  
                                           (180,130),  
                                           (160,150),  
                                           (110,150),  
                                           (110,130),  
                                           (100,100));
```

```
{Задание массива точек дополнительного объекта}
```

```
const XY2 : array [0..5,0..1] of word = ((110,150),  
                                           (110,140),  
                                           (130,140),  
                                           (135,145),  
                                           (130,150),  
                                           (110,150));
```

```
{Задание массива точек цели}
```

```
const XY3 : array [0..4,0..1] of word = ((590,130),  
                                           (600,130),  
                                           (600,160),  
                                           (590,160),  
                                           (590,130));
```

```
{Описание переменных}  
var
```

```
GrDr,GrMd : integer;
```

```

x11,y11,x12,y12,x21,y21,x22,y22,x31,y31,x32,y32,i,j :
integer;
  label M,N,O;

{Тело программы}
BEGIN

  {Очистка экрана}
  clrscr;
  {Определение номера, установленного в системе драйвера}
  DetectGraph(GrDr,GrMd);

  {Инициализация графического режима}
  InitGraph(GrDr,GrMd,' ');

  {Установка цвета фона окна}
  setbkcolor(2);

  {Определение начальной координаты объекта}
  x11:=XY1[0,0];y11:=XY1[0,1];
  {Рисование начальной координаты объекта}
  moveto(x11,y11);
  {Цикл рисования точек объекта}
  for j := 0 to 9 do
    begin
      x12:=XY1[j,0];y12:=XY1[j,1];
      lineto(x12,y12);
      delay(1000) {Задержка}
    end;

  {Определение начальной координаты дополнительного
  объекта}
  x21:=XY2[0,0];y21:=XY2[0,1];
  {Рисование начальной координаты дополнительного
  объекта}
  moveto(x21,y21);
  {Рисование массива точек дополнительного объекта}
  for j := 0 to 5 do
    begin
      x22:=XY2[j,0];y22:=XY2[j,1];
      lineto(x22,y22);
      delay(1000) {Задержка}
    end;

  {Определение начальной координаты цели}
  x31:=XY3[0,0];y31:=XY3[0,1];

```

```

{Рисование массива точек цели}
moveto(x31,y31);
for j := 0 to 4 do
  begin
    x32:=XY3[j,0];y32:=XY3[j,1];
    lineto(x32,y32);
    delay(1) {Задержка}
  end;

{Моделирование динамики полета}
for i := 0 to 400 do
  begin

{Изменение связанной системы координат объекта}
    x11:=XY1[0,0]+i;y11:=XY1[0,1];
    {Рисование первой точки объекта}
    moveto(x11,y11);

{Цикл рисования массива точек объекта}
    for j := 0 to 9 do
      begin
        x12:=XY1[j,0]+i;y12:=XY1[j,1];
        lineto(x12,y12);
        delay(1) {Задержка}
      end;

{Изменение связанной системы координат дополнительного
объекта}
    x21:=XY2[0,0]+i;y21:=XY2[0,1];
    moveto(x21,y21);

{Цикл рисования массива точек дополнительного объекта}
    for j := 0 to 5 do
      begin
        x22:=XY2[j,0]+i;y22:=XY2[j,1];
        lineto(x22,y22);
        delay(1) {Задержка}
      end;

{Рисование начальной точки цели}
    x31:=XY3[0,0];y31:=XY3[0,1];
    moveto(x31,y31);
{Цикл рисования точек цели}
    for j := 0 to 4 do
      begin

```

```

    x32:=XY3[j,0];y32:=XY3[j,1];
    lineto(x32,y32);
    delay(1) {Задержка}
end;

```

```

{Моделирование условия срабатывания отстреливания
дополнительного объекта по нажатию клавиши}
    if keypressed=true then goto M else goto N;
{Моделирование относительного движения}

```

M:

```

{Изменение связанной системы координат дополнительного
объекта ускорено по отношению к координатам объекта}

```

```

    x21:=XY2[0,0]+i*2;y21:=XY2[0,1];
    moveto(x21,y21);
    for j := 0 to 5 do
        begin
            x22:=XY2[j,0]+i*2;y22:=XY2[j,1];
            lineto(x22,y22);
            delay(1)
        end;

```

```

{Моделирование анализа попадания дополнительного
объекта в цель}

```

```

    if x22=590 then goto O else goto N;

```

O: {Дополнительный объект попал в цель}

```

{Реакция на попадание в виде красной заливки
графического экрана}

```

```

    floodfill(595,145,4);
    delay(1); {Задержка}

```

{Выбор цвета заливки окна}

```

    setbkcolor(4);

```

{Выбор цвета линий}

```

    setColor(4);

```

{Гашение мишени путем повторного рисования ее цветом
линий, совпадающим с цветом окна}

```

    x31:=XY3[0,0];y31:=XY3[0,1];
    moveto(x31,y31);
    for j := 0 to 4 do
        begin
            x32:=XY3[j,0];y32:=XY3[j,1];
            lineto(x32,y32);
            delay(1) {Задержка}
        end;

```

N:

```

    {Изменение связанной системы координат дополнительного
    объекта равноускорено по отношению к координатам
    объекта}
    x21:=XY2[0,0]+i;y21:=XY2[0,1];
    moveto(x21,y21);
    for j := 0 to 5 do
        begin
            x22:=XY2[j,0]+i;y22:=XY2[j,1];
            lineto(x22,y22);
            delay(1)
        end;
    ClearDevice; {Очистка графического экрана}
end;
readln; {Ожидание нажатия клавиши}
CloseGraph; {Закрытие графического режима}
END.

```

Д.5. Моделирование руху динамічного об'єкта

```
{Моделирование подвижного объекта, обладающего массой}  
Program Lab2;
```

```
{Подключение стандартных модулей}  
Uses Graph,Crt;
```

```
{Описание констант}  
Const N=15;  
      f=15;  
      c=2;
```

```
{Задание матриц А и В, управляющих траекторией полета  
объекта относительно осей X и Y}  
A : array [1..3,1..3] of real = ((0,      1e-7, -1e-7),  
                                 (-1e-7, 0,      1e-7),  
                                 (1e-7,   -1e-7, 0));  
  
B : array [1..3,1..3] of real = ((-1e-4, 0,      0),  
                                 (0,     -1e-4, 0),  
                                 (0,      0,   -1e-4));
```

```
{Описание типов переменных}  
Type P=Record  
      x,y:Integer end;  
Var k:array[1..n] of P;
```

```
{Описание переменных}  
  ux, gd,gr,x,y,xold,yold:integer;  
  wz,vx,vy, dh:real;
```

```
{Описание меток}  
  Label M,L;
```

```
{Процедура рисования} {Заголовок процедуры}  
Procedure Plane(x,y,c:Integer);
```

```
{Описание переменных}  
Var i:Integer;
```

```
{Тело процедуры}  
Begin  
{Выбор цвета фона графического окна}  
  SetBKColor(2);
```

```

{Выбор цвета выводимых линий}
  SetColor(c);

{Цикл рисования}
  For i:=1 to n-1 do
    {Процедура рисования}
    Line(x+k[i].x,y+k[i].y,x+k[i+1].x,y+k[i+1].y)
  End;

{Тело программы}
Begin
{Задание координат точек объекта}

{Инициализация графического режима}
  InitGraph(gd,gr,'C:\TP55\');

{Задание начальных координат}
  x:=5;
  y:=425;

{Задание начальных скоростей}
  vx:=1;
  vy:=1;
  wz:=a[1,2];

{Операторы приравнивания координат}
  xold:=x;
  yold:=y;

{Задание шага интегрирования}
  dh:=0.005;

{Задание возмущения}
  Ux:=-20000;

{Оператор условия, при котором будет выполняться
рисование}
  While x<=700 do
    Begin
{Вызов процедуры рисования объекта}
      Plane(xold,yold,f);
{Закрытие графического режима}
      CloseGraph;
    End.

```


БІБЛІОГРАФІЧНИЙ СПИСОК

1. Петров, А. П. Моделирование на ЭОМ [Текст]: навч. посібник для студентів інженерних спеціальностей вищих навчальних закладів/ А. П. Петров. – К.: Політехніка, 2007. – 340 с.
2. Давиденко, А. П. Организация и планирование научных исследований, патентоведение [Текст]: учебник. / А. П. Давиденко. – Х.: НТУ “ХПИ”, 2004. – 320 с.
3. Остапчук, М. В. Математичне моделювання на ЕОМ [Текст]: підручник / М. В. Остапчук, Г. М. Станкевич. – К.: Політехніка, 2005. – 392 с.
4. Кошевой, Н. Д. Автоматизация экспериментальных исследований [Текст]: монография / Н. Д. Кошевой, В. А. Гаевой. – Х.: Факт, 2001. – 112 с.
5. Михайлов, А. Г. Моделирование на ЭВМ [Текст] / А. Г. Михайлов. – Х.: Нац. аэрокосм. ун-т "ХАИ", 1999. – 55 с.
7. Шеффе, Г. Дисперсионный анализ [Текст] / Г. Шеффе. – М.: Наука, 1980. – 511 с.
8. Кендалл, М. Дж. Многомерный статистический анализ и временные ряды [Текст] / М. Дж. Кендалл, А. Стюарт. – М.: Наука, 1976. – 736 с.
9. Брандт, З. Статистические методы анализа наблюдений [Текст] / З. Брандт. – М.: Мир, 1975. – 310 с.
10. Пугачев, В. С. Теория вероятностей и математическая статистика [Текст] / В. С. Пугачев. – М.: Наука, 1979. – 496 с.
11. Кострікін, А. І. Введення в алгебру [Текст] / А. І. Кострікін. – М.: Наука, 1977. - 495 с.
12. Кук, Д. Компьютерная математика [Текст]: учебник; пер. с англ. / Д. Кук, Д. Бейз. – М.: Наука, 1990. – 384 с.
13. Компьютерная алгебра: символьные и алгебраические вычисления [Текст]: учебник; пер. с англ.; под ред. Б. Бухбергера и др. - М.: Мир, 1986. – 392 с.
14. Курош, А. Г. Теория групп [Текст] / А. Г. Курош. - М.: Наука, 1967. – 648 с.
15. Літнарівч, Р. М. Ілюстрований самовчитель по MAPLE [Текст]: у 15 ч. / Р. М. Літнарівч. – Рівне: МЕРУ, 2009.
16. Половко, А. М. MATLAB для студента [Текст]: учебник / А. М. Половко, П. Н. Бутусов. – СПб.: БХВ-Петербург, 2005. – 320 с.
17. Макаров, Е. Г. Инженерные расчеты в MathCad [Текст] / Е. Г. Марков. – СПб.: Питер, 2003. – 448 с.

18. Лурье, Б. Я. Классические методы автоматического управления [Текст] / Б. Я. Лурье, П. Дж. Энрайт; под ред. А. А. Ланнэ. – СПб.: БХВ – Петербург, 2004. – 640 с.
19. Волков, Е. А. Численные методы [Текст] / Е.А. Волков. – М.: Наука, 1982. – 255 с.
20. Олссон, Г. Цифровые системы автоматизации и управления [Текст] / Д. Пиани. – СПб.: Невский диалект, 2001. – 557 с.

ЗМІСТ

ВСТУП.....	3
1. МОДЕЛЮВАННЯ І АНАЛІЗ СИСТЕМ.....	4
2. МОДЕЛЮВАННЯ ЛІНІЙНИХ СИСТЕМ 2-ГО І ВИЩИХ ПОРЯДКІВ.....	23
3. МОДЕЛЮВАННЯ КОЛИВАЛЬНИХ СИСТЕМ У СИСТЕМІ MATLAB-SIMULINK.....	34
4. МОДЕЛЮВАННЯ СИСТЕМ ВИСОКОГО ПОРЯДКУ.....	41
5. ГЕОМЕТРИЧНЕ МОДЕЛЮВАННЯ СТАНУ СИСТЕМ НА ФАЗОВІЙ ПОВЕРХНІ	49
6. МОДЕЛЮВАННЯ СИСТЕМ З ДИСКРЕТНИМ ЧАСОМ. МОДЕЛІ ЦИФРОВИХ СИСТЕМ.....	59
7. ОРГАНІЗАЦІЯ НАПІВНАТУРНОГО МОДЕЛЮВАННЯ.....	67
8. ОБРОБЛЕННЯ РЕЗУЛЬТАТІВ МОДЕЛЮВАННЯ ПРИ МІНІМІЗАЦІЇ КІЛЬКОСТІ ЕКСПЕРИМЕНТІВ.....	77
9. МОДЕЛЮВАННЯ В ІНТЕГРОВАНОМУ ПАКЕТІ STRATUM.....	94
ДОДАТОК.....	105
БІБЛІОГРАФІЧНИЙ СПИСОК.....	121

Навчальне видання

КОШОВИЙ Микола Дмитрович
МИХАЙЛОВ Андрій Георгійович

**ТЕОРІЯ І ПРАКТИКА МОДЕЛЮВАННЯ
ІНФОРМАЦІЙНО-ВИМІРЮВАЛЬНИХ СИСТЕМ**

Редактор Л. О. Кузьменко

Зв. план, 2014

Підписано до друку 27.11.2014

Формат 60 x 84 1/16. Папір офс. № 2. Офс. друк

Ум. друк. арк. 6,9. Обл.-вид. арк. 7,75. Наклад 500 пр. Замовлення 332.

Ціна вільна

Видавець і виготовлювач
Національний аерокосмічний університет ім. М. Є. Жуковського
“Харківський авіаційний інститут”
61070, Харків–70, вул. Чкалова, 17
[http: //www.khai.edu](http://www.khai.edu)

Видавничий центр “ХАІ”
61070, Харків–70, вул. Чкалова, 17
izdat@khai.edu

Свідоцтво про внесення суб'єкта видавничої справи
до Державного реєстру видавців, виготовлювачів і розповсюджувачів
видавничої продукції сер. ДК № 391 від 30.03.2001