

**УКРАЇНСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ЗАЛІЗНИЧНОГО ТРАНСПОРТУ**

**ФАКУЛЬТЕТ ІНФОРМАЦІЙНО-КЕРУЮЧИХ СИСТЕМ
ТА ТЕХНОЛОГІЙ**

Кафедра обчислювальної техніки та систем управління

**МЕТОДИЧНІ ВКАЗІВКИ
до лабораторних робіт**

**з дисципліни
«ЧИСЛОВІ МЕТОДИ І МОДЕЛЮВАННЯ НА ЕОМ»**

Харків – 2020

Методичні вказівки розглянуто і рекомендовано до друку на засіданні кафедри обчислювальної техніки та систем управління 12 лютого 2020 р., протокол № 7.

Призначено для студентів факультету ІКСТ університету денної та заочної форм навчання.

Укладачі:

доценти О. В. Головка,
В. М. Бутенко

Рецензент

доц. С. І. Доценко

ЗМІСТ

Вступ.....	4
Лабораторна робота 1. Побудова найкоротшої зв'язуючої мережі.....	5
Лабораторна робота 2. Моделювання та дослідження поведінки динамічних детермінованих систем.....	10
Лабораторна робота 3. Визначення шляху найменшої вартості методом динамічного програмування.....	17
Лабораторна робота 4. Дослідження моделей, що описуються системами лінійних рівнянь.....	26
Лабораторна робота 5. Обробка даних експерименту методом регресійного аналізу.....	31
Лабораторна робота 6. Моделювання процесів витрати палива.....	37
Лабораторна робота 7. Моделі оптимального керування запасами.....	44
Лабораторна робота 8. Метод Монте-Карло у задачах параметричної оптимізації.....	51
Список літератури.....	60

ВСТУП

Сучасні інформаційні технології – дуже стрімка галузь як науки, так і промислово-інтелектуального виробництва. Крім того, в сучасних умовах це найбільш динамічний сектор ринку праці, який потребує значно менших інвестицій для виходу на міжнародні ринки. Дефіцит фахівців у цій галузі змушує роботодавців знижувати формальні, вхідні критерії приймання на роботу молодих фахівців, а також брати на роботу студентів. Для найшвидшого оволодіння сучасним рівнем математичного та числового моделювання, що є компонентами створення інформаційних технологій, необхідно допомагати студентам отримати компетентності сучасного ринку праці.

У цих методичних вказівках запропоновано лабораторний практикум з восьми лабораторних робіт, які якнайкраще підходять для сучасного розуміння базових моделей та числових методів вирішення майже типових алгоритмів. Можливість реалізації наведених алгоритмів будь-якою мовою програмування дає змогу студенту та будь-якому досліднику самостійно спроектувати програмне забезпечення за наведеними моделями та числовими методами їх рішення й змодельовати процеси. Це не тільки наочно демонструє моделі та методи, але й дає змогу дослідникам самостійно долучитися до процесу створення результатів кінцевого дослідження.

ЛАБОРАТОРНА РОБОТА 1. Побудова найкоротшої зв'язуючої мережі

Мета роботи: набування навичок у рішенні за допомогою ЕОМ задач оптимізації.

Контрольні питання

- 1 Які види структури мережі ви знаєте?
- 2 Який вид структури мережі є максимально надійним?
- 3 Який вид структури мережі має максимальну вартість лінії зв'язку?
- 4 Який вид структури мережі має головну станцію, що зв'язана з кожною станцією мережі?
- 5 Який вид структури мережі утворює кільце із станцій мережі?
- 6 У чому полягає задача побудови найкоротшої зв'язуючої мережі?
- 7 Як визначається відстань l_{ij} між точками (x_i, y_i) і (x_j, y_j) ?

Навчальний матеріал

Одними з поширених технічних систем є різні мережі: електрозв'язку, електропостачання, сигналізації й ін. Лінії зв'язку в таких мережах, що являють собою проводи або кабелі, як правило, мають велику довжину і високу вартість. При цьому основна частка вартості всієї мережі припадає на лінії зв'язку. Тому при проектуванні таких мереж для зменшення їх вартості необхідно вирішувати задачу визначення такого варіанта структури мережі, що мав би мінімальну довжину ліній зв'язку і, отже, мінімальну вартість, тому що вартість лінії зв'язку пропорційна її довжині. Структура визначає характер або тип зв'язків у мережі. Існують різні види структур мережі, наприклад: повнозв'язна, кільцева, деревоподібна, централізована й ін. [1].

Як правило при побудові мережі намагаються мінімізувати її вартість і досягти якомога більшої надійності.

Однією з найбільш дешевих мереж є кільцева (рисунок 1.1). Але її надійність невисока. При втраті працездатності однієї

ділянки важко визначити, яка саме ділянка вийшла з ладу, а коли випадають дві ділянки, мережа розпадається на дві ізольовані частини.

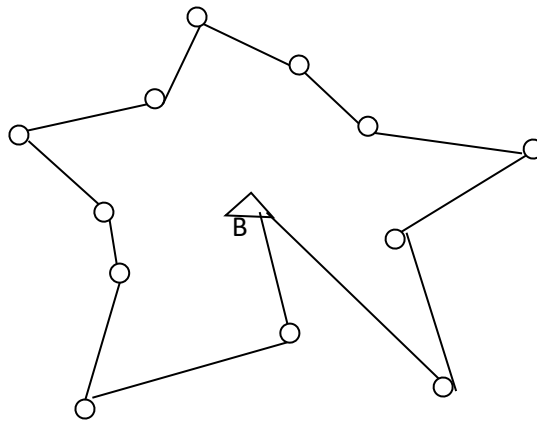


Рисунок 1.1 – Кільцева мережа

Найбільш надійною є повнозв'язна мережа (рисунок 1.2). У такій мережі кожен пункт зв'язаний ділянкою з кожним іншим пунктом. При виході з ладу однієї або навіть декількох ділянок вона не втрачає працездатності. Але її вартість невіправдано висока, особливо при кількості вузлів більше чотирьох.

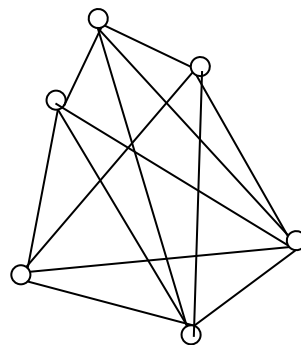


Рисунок 1.2 – Повнозв'язна мережа

Найбільш оптимальною є централізована мережа (рисунок 1.3). З множини пунктів вибирається центральний, з яким і з'єднуються ділянками всі інші пункти. При виході з ладу якоїсь ділянки відразу видно, який пункт втратив зв'язок, де потрібен ремонт. При цьому втрачається зв'язок лише з одним пунктом, а не з фрагментом мережі.

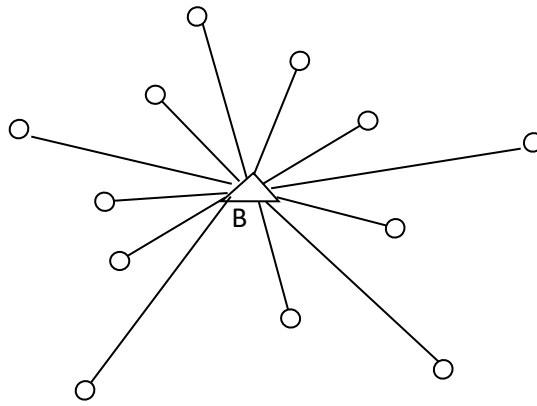


Рисунок 1.3 – Централізована мережа

Постановка задачі

Розглянемо задачу визначення найкоротшої зв'язуючої мережі, що має централізовану структуру (рисунок 1.3).

У такій мережі є один пункт, назвемо його вузол, з яким лініями зв'язку зв'язані всі інші пункти мереж.

Ця задача являє собою задачу оптимізації структури мережі, у якій необхідно визначити оптимальний варіант мережі, тобто варіант, що має мінімальну загальну довжину ліній зв'язку.

Для вирішення задачі і побудови математичної моделі приймемо такі умови:

- будемо вважати, що лінії зв'язку є прямими лініями;
- вузол можна розміщати в кожній із точок, що входять у мережу.

Пункти, які необхідно з'єднати в мережу, пронумеровані від 1 до n і визначаються парою координат. Зобразимо їх у вигляді точок на площині в прямокутній системі координат (рисунок 1.4).

Очевидно, що різні варіанти мережі можна одержувати, поміщаючи вузол у різні точки заданої множини точок. Тоді задача визначення оптимального варіанта мережі буде полягати в тому, щоб знайти таку точку розміщення вузла, для якої загальна довжина ліній зв'язку буде мінімальною.

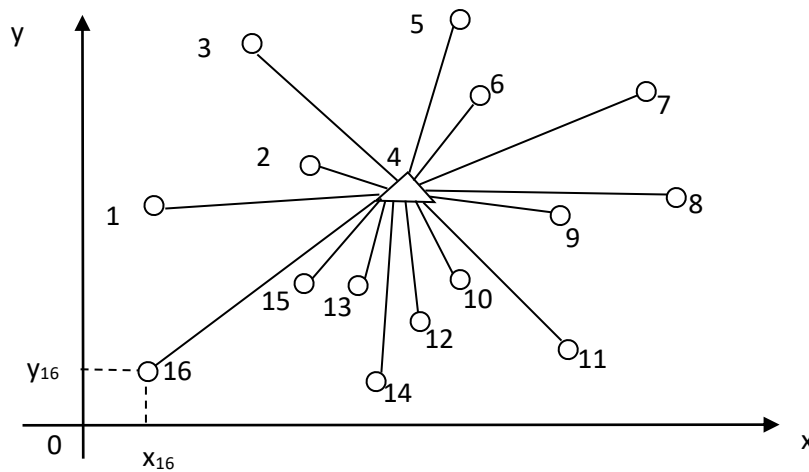


Рисунок 1.4 – Централізована мережа в системі координат

Довжина лінії зв'язку між двома точками i -ю та j -ю визначається за відомою формулою

$$l_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2},$$

де i, j – номери точок, з'єднаних лінією зв'язку;
 x_i, x_j, y_i, y_j – координати точок.

Будемо вважати, що в точці з номером i розміщений вузол. Тоді загальну довжину ліній зв'язку можна визначити за виразом

$$S_i = \sum_{j=1}^n l_{ij},$$

де n – кількість точок.

Мету задачі – визначення мінімуму S_i – можна виразити таким чином:

$$S_i^{\min} = \min_i \sum_{j=1}^n l_{ij}; i = 1, \dots, n;$$

Очевидно, що найкращим методом рішення цієї задачі є повний перебір усіх варіантів, що дає змогу точно визначити оптимальне рішення.

При рішенні задачі на комп'ютері як результат вивести на екран номер точки, у якій розміщений вузол в оптимальному варіанті мережі, і відповідну йому загальну довжину ліній зв'язку.

Завдання та порядок виконання

- 1 Усвідомити постановку задачі та метод її рішення. Вивчити теоретичний матеріал [1, 2].
- 2 Скласти схему алгоритму рішення задачі для довільної кількості точок [2].
- 3 Скласти програму мовою C++ для цього алгоритму [3].
- 4 Набрати програму на комп'ютері та відладити її.
- 5 Задати довільно координати 12 точок у вигляді цілих двозначних чисел, увести їх у програму й вирішити задачу побудови найкоротшої зв'язуючої мережі.
- 6 Як результат вивести на екран довжину ліній зв'язку й номер точки розміщення вузла для оптимального варіанта мережі.
- 7 Оформити звіт з роботи.
Зміст звіту:
 - вхідні дані (координати точок) та рисунок, що містить точки із заданими координатами, розташовані на координатній площині;
 - схема алгоритму;
 - текст програми;
 - результати рішення задачі за допомогою програми; рисунок, на якому позначено точку розміщення вузла та побудовано відповідну мережу;
 - відповіді на контрольні питання.

ЛАБОРАТОРНА РОБОТА 2. Моделювання та дослідження поведінки динамічних детермінованих систем

Мета роботи: вивчення методів моделювання й визначення характеристик перехідних процесів у динамічних системах.

Контрольні питання

- 1 Яка система називається динамічною?
- 2 Яка система називається детермінованою?
- 3 Яка математична модель описує перехідні процеси в динамічних системах?
- 4 Яку математичну задачу необхідно вирішити при дослідженні поведінки динамічної системи?
- 5 У яких випадках доцільно використати числові методи рішення диференціальних рівнянь?
- 6 Які математичні дії лежать в основі числових методів рішення диференціальних рівнянь?
- 7 За рахунок чого рішення числовим методом виходить наближеним?
- 8 Пояснити метод Ейлера.
- 9 Пояснити метод Рунге-Кутта.
- 10 Як впливає величина кроку інтегрування на точність рішення?

Навчальний матеріал

Загальні поняття про динамічні системи

Багато процесів, що відбуваються в системах різної природи, зокрема в технічних системах, являють собою перехідні процеси, що можуть бути досліджені за допомогою математичних моделей.

Перехідний процес – це зміна стану системи в часі, яку ще заведено називати рухом системи. Стан системи описується набором змінних – фазових координат, що залежно від природи системи можуть відображати різні фізичні величини, а саме:

швидкість, прискорення, струм, напруга, температура, тиск тощо. Ці змінні перетворюються при перехідному процесі [1, 4, 5].

Дослідження поведінки динамічної системи за допомогою математичних моделей являє собою побудову функцій, що описують її перехідний процес. Динамічні властивості різних природних і технічних систем такі, що перехідні процеси в них відбуваються за тими самими законами і, отже, описуються тією самою математичною моделлю. Такою моделлю є звичайні диференціальні рівняння (ЗДУ).

Система, у якій можуть відбуватися перехідні процеси, називається *динамічною*.

Система, на поведінку якої не можуть впливати випадкові зовнішні фактори, називається *детермінованою* [1, 4, 5].

При дослідженні систем визначаються функції, що виражають залежність змінних, які описують стан системи, від часу. Для цього необхідно вирішити диференціальне рівняння або систему диференціальних рівнянь.

Постановка задачі

У найбільш загальному випадку система диференціальних рівнянь має такий вигляд:

$$\begin{cases} F_1(x, y_1, y_1', y_1'', \dots, y_2, y_2', y_2'', \dots, y_n, y_n', y_n'', \dots) = 0; \\ F_2(x, y_1, y_1', y_1'', \dots, y_2, y_2', y_2'', \dots, y_n, y_n', y_n'', \dots) = 0; \\ \vdots \\ F_n(x, y_1, y_1', y_1'', \dots, y_2, y_2', y_2'', \dots, y_n, y_n', y_n'', \dots) = 0, \end{cases}$$

де y_1, y_2, \dots, y_n – змінні системи;

x – аргумент – величина, від якої залежать змінні системи в перехідних процесах; найчастіше такою величиною є час.

Однак найбільш зручним для рішення на ЕОМ є подання системи диференціальних рівнянь у нормальній формі Коші. Рівняння вищих порядків на практиці трапляються рідко, тому розглянемо систему ЗДУ першого порядку:

$$\begin{cases} \frac{dy_1}{dx} = f_1(x, y_1, y_2, \dots, y_n); \\ \frac{dy_2}{dx} = f_2(x, y_1, y_2, \dots, y_n); \\ \vdots \\ \frac{dy_n}{dx} = f_n(x, y_1, y_2, \dots, y_n). \end{cases}$$

Для простоти розглянемо випадок, коли $n = 1$, тобто в системі є одна змінна, а перехідний процес описується одним диференціальним рівнянням:

$$\frac{dy}{dx} = f(x, y), \quad \text{или} \quad y' = f(x, y).$$

Для того, щоб уявити поведінку системи, необхідно знайти функцію $y(x)$, тобто вирішити це диференціальне рівняння.

Як відомо, диференціальне рівняння має незліченну множину рішень $y(x)$ і для того, щоб знайти яке-небудь одне рішення, потрібно задати початкові умови: x_0 і y_0 та інтервал зміни x : x_0, x_k . Тоді $x_0 \leq x \leq x_k$. Така задача називається *задачею Коші*.

Рішення цієї задачі аналітичним шляхом буває складним і не завжди можливим (для цього потрібно проінтегрувати функцію $f(x, y)$, тобто знайти $y(x) = \int_{x_0}^{x_k} f(x, y) dx$, при $y = y_0$, а такий інтеграл не завжди можна взяти), тому на практиці найчастіше використовують числові методи, що дають змогу досить просто одержати наближене рішення [1, 5].

Числові методи рішення задачі Коші

При використанні числових методів визначається не сама функція $y(x)$ (первісна) в аналітичному вигляді, а приблизно обчислюються її окремі значення в заданих точках на заданому інтервалі $[x_0, x_k]$ [1, 4].

В основу числових методів покладена така процедура:

1) задаються початкові умови x_0, y_0 та інтервал зміни x - $[x_0, x_k]$;

2) точки, у яких обчислюються значення $y(x)$, задаються таким чином: інтервал $[x_0, x_k]$ розбивається на кінцеву кількість відрізків рівної довжини (рисунок 2.1).

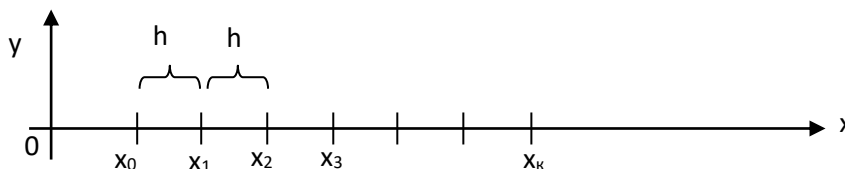


Рисунок 2.1

Значення $x_0, x_1, x_2, \dots, x_k$ і є заданими точками. Відстань між сусідніми точками h (довжина відрізка) називається кроком інтегрування.

3) значення $y(x)$ у точках $x_0, x_1, x_2, \dots, x_k$ обчислюються за допомогою формули розкладання функції в ряд Тейлора:

$$y_{i+1} = y_i + \underbrace{h \cdot y'_i + \frac{h^2}{2!} y''_i + \frac{h^3}{3!} y'''_i + \dots}_{\text{Приріст } y},$$

де y_i, y'_i, y''_i, \dots – значення y і її похідних у попередній (i -й) точці ;
 y_{i+1} – значення y у наступній ($i+1$ -й) точці .

Таким чином, значення $y(x)$ у кожній наступній точці обчислюється зважаючи на значення $y(x)$ і її похідні у попередній точці .

Є кілька методів обчислення $y(x)$. Вони відрізняються один від одного способом обчислення приросту y .

Метод Ейлера

У формулі розкладання функції в ряд Тейлора всі члени ряду, що містять похідні другого і вищих порядків, відкидаються через їхню малість (за рахунок цього обчислення виходить наближеним) і формула набуває виду [1, 4].

$$y_{i+1} = y_i + h \cdot y'_i,$$

або $y_{i+1} = y_i + h \cdot f(x_i, y_i).$

Це можна відобразити на графіку (рисунок 2.2).

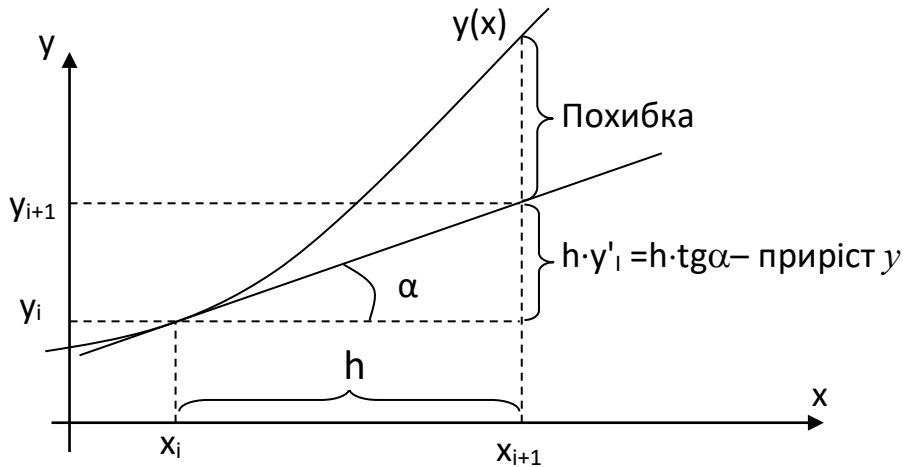


Рисунок 2.2

За рахунок спрощення формули розкладання в ряд Тейлора шукана функція $y(x)$ на кожному відрізку інтервалу інтегрування замінюється рівнянням прямої лінії, дотичної до кривої $y(x)$ на початку відрізка, тобто будується кусочно-лінійна апроксимація.

Похибка цього методу має порядок h^2 . Для підвищення точності потрібно зменшити h , але при цьому збільшується обсяг обчислень.

Метод Рунге-Кутта

Цей метод є більш точним і поширеним. Він дає можливість при меншому обсязі обчислень більш точно обчислити приріст y на кожному кроці. Уточнення досягається за рахунок спеціального підбору координат чотирьох сусідніх точок, у яких обчислюється перша похідна y'_i , а приріст функції $y(x)$ визначається, як усереднене значення приростів y у цих точках [1, 4]:

$$y_{i+1} = y_i + h \cdot (k_1 + 2k_2 + 2k_3 + k_4) / 6,$$

де

$$\begin{aligned}
k_1 &= f(x_i, y_i); \\
k_2 &= f(x_i + 0,5h, y_i + 0,5k_1); \\
k_3 &= f(x_i + 0,5h, y_i + 0,5k_2); \\
k_4 &= f(x_i + h, y_i + k_3).
\end{aligned}$$

Похибка цього методу має порядок h^5 .

При обчисленні значень $y(x)$ числовими методами вхідні дані можуть задаватися у двох варіантах:

- 1) задані $f(x, y)$, x_0 , y_0 , h і N – кількість кроків;
- 2) задані $f(x, y)$, x_0 , y_0 , x_k і N , у цьому випадку необхідно обчислити крок h за формулою

$$h = \frac{x_k - x_0}{N}.$$

Завдання та порядок виконання

1 Вивчити теоретичний матеріал, зокрема числові методи рішення диференціальних рівнянь [1, 4].

2 Вивчити алгоритми, що реалізують методи Ейлера й Рунге-Кутта [1, 4].

3 Скласти програму мовою C++ за алгоритмами рішення задачі методами Ейлера й Рунге-Кутта [2, 3] за варіантами, поданими у таблиці 2.1.

4 Набрати програму на комп'ютері та відладити її.

5 Вибрати варіанти вхідних даних з таблиці 2.1, при цьому номер варіанта відповідає номеру за списком студента в журналі викладача.

6 Виконати програму на комп'ютері.

7 Оформити звіт з роботи.

Зміст звіту:

- вхідні дані;
- тексти програми;
- результати рішення задачі;
- відповіді на контрольні питання.

Алгоритми показані на рисунках 2.3, 2.4.

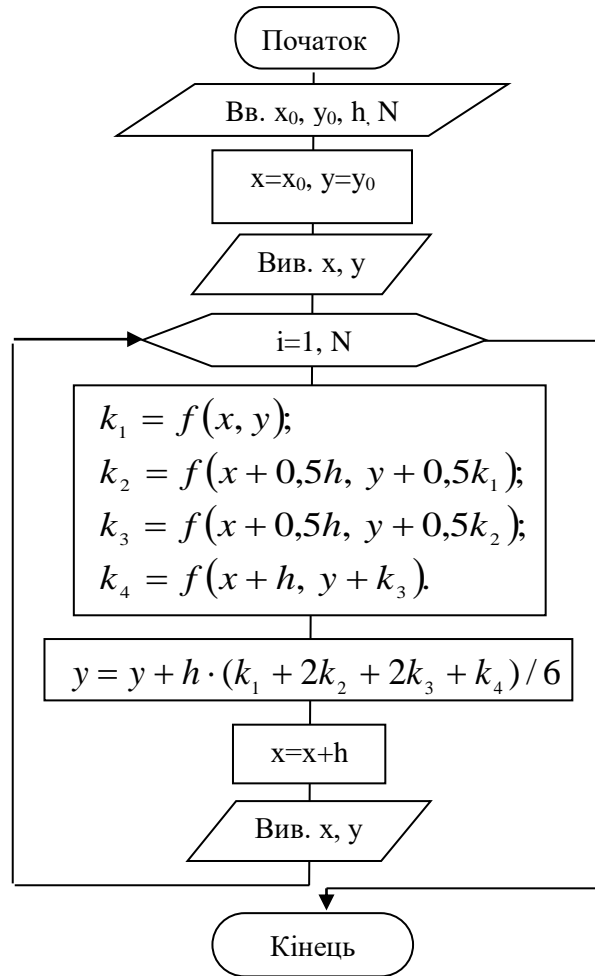
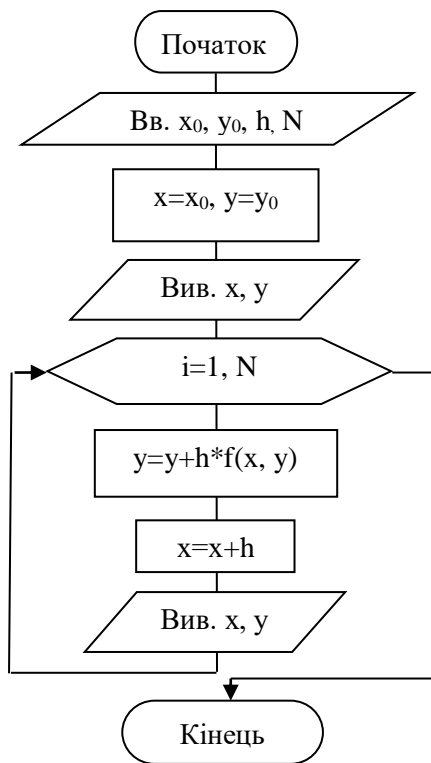


Рисунок 2.3 – Метод Ейлера

Рисунок 2.4 – Метод Рунге-Кутта

Таблиця 2.1 – Варіанти вхідних даних

Варіант	$f(x,y)$	x_0	y_0	h	N
1	$y+x$	0	1	0,25	10
2	$2y+x$	0	0	0,5	15
3	$y-x$	0	1	0,2	20
4	$3y+4x$	0	0	0,25	10
5	$x-y$	0	1	0,5	15
6	$4y+3x$	0	0	0,2	20
7	$y+2x$	0	1	0,25	10
8	$y+3x$	0	0	0,5	15
9	$3y-2x$	0	1	0,2	20
10	$y-2x$	0	0	0,25	10
11	$4y+2x$	0	1	0,5	15
12	$y-3x$	0	0	0,2	20
13	$2y-x$	0	1	0,25	10
14	$3y-x$	0	0	0,5	15
15	$2x-3y$	0	1	0,2	20

ЛАБОРАТОРНА РОБОТА 3. Визначення шляху найменшої вартості методом динамічного програмування

Мета роботи: вивчення методів рішення задач оптимізації на прикладі методу динамічного програмування.

Контрольні питання

1 Який принцип лежить в основі методу динамічного програмування? У чому він полягає?

2 Для рішення яких задач застосовується метод динамічного програмування?

3 Які етапи містить у собі оптимізація методом динамічного програмування? Яким чином вони здійснюються?

4 Чому оптимізацію кроків варто здійснювати, рухаючись від кінця до початку?

5 Пояснити, як з оптимальних кроків будується оптимальний шлях.

Навчальний матеріал

Основні поняття і математична модель оптимізації

Багато процесів, що відбуваються в системах різної природи, зокрема в технічних системах, являють собою перехідні процеси, що можуть бути досліджені за допомогою математичних моделей.

У практиці проектування технічних систем, дослідження і моделювання систем, керування технологічними й економічними процесами часто трапляються задачі, коли з множини варіантів необхідно вибрати найкращий за яким-небудь показником варіант. Процес знаходження такого варіанта називається оптимізацією.

До таких задач, наприклад, належать:

- визначення найкоротшого шляху між двома пунктами;
- розподіл ресурсів між підприємствами;
- вибір маршруту в транспортній мережі;

- вибір параметрів технічної системи;
- розміщення елементів електронної схеми на платі й ін.

Оптимізувати можна як систему, так і процес.

Показник, за яким відбувається оптимізація, називається критерієм оптимізації. Як критерій вибирається, як правило, найбільш важлива в розглянутій задачі характеристика системи або процесу, що оптимізуються. Величина критерію виражається числовим значенням і може бути обчислена або обміряна. При цьому кожному варіантові, аналізованому в процесі оптимізації, відповідає визначене значення критерію оптимізації.

Варіант, при якому критерій досягає мінімального або максимального значення, називається оптимальним.

Математична модель оптимізації – це задача оптимізації, сформульована в математичному вигляді.

Критерій виражається у вигляді функції, що залежить від визначених параметрів або характеристик, значення яких можуть змінюватися в процесі оптимізації. Така функція називається функцією мети, або цільовою функцією, а параметри, від яких вона залежить, – змінними оптимізації [2].

Математична модель оптимізації має такий загальний вигляд.

Цільова функція W прагне до мінімуму або максимуму:

$$W(x_1, x_2, \dots, x_n) \rightarrow \min (\max),$$

де x_1, x_2, \dots, x_n - змінні оптимізації;

при виконанні обмежень:

$$f_1(x_1, x_2, \dots, x_n) = 0;$$

$$f_2(x_1, x_2, \dots, x_n) = 0;$$

...

$$f_m(x_1, x_2, \dots, x_n) = 0.$$

При цьому $m < n$.

Зміст обмежень полягає в тому, що змінні оптимізації можуть набувати не будь-яких значень, а тільки таких, для яких виконується система обмежень.

Сукупність значень x_1, x_2, \dots, x_n , що задовольняє обмеження, називається припустимим рішенням.

Сукупність припустимих рішень називається областю, або множиною припустимих рішень.

Припустиме рішення, для якого W досягає екстремуму, називається оптимальним рішенням.

Цільова функція може мати один або багато екстремумів. У першому випадку вона називається однокстремальною, а у другому – багатокстремальною. Екстремуми багатокстремальної функції називаються локальними екстремумами, а той з них, що відповідає оптимальному рішенням, – глобальним екстремумом [3].

Залежно від специфіки задачі, виду цільової функції й обмежень існують різні методи рішення таких задач, об'єднані під загальною назвою - методи математичного програмування. Одним з них є метод динамічного програмування.

Метод динамічного програмування

Метод динамічного програмування призначений для пошуку оптимальних рішень у задачах оптимізації процесів, що складаються з визначеної кількості етапів (кроків), які послідовно впливають один за іншим. У таких задачах у загальному випадку необхідно знайти оптимальний процес здійснення ряду визначених дій, спрямованих на досягнення якоїсь мети [2, 4].

Прикладами таких задач можуть бути:

- планування діяльності групи промислових підприємств протягом ряду років;
- переміщення транспортного засобу з одного пункту в інший через ряд проміжних пунктів;
- здійснення ряду заходів щодо підвищення надійності проектованої технічної системи тощо.

В усіх цих задачах існує показник, за яким оцінюється ефективність усього процесу, або критерій оптимізації. Позначимо його через W .

Зміст критерію W полягає в тому, що звичайно це або вигравш (ефект, прибуток) і в цьому випадку його необхідно

максимізувати: $W \rightarrow \max$, або програш (витрати, штраф) і тоді його потрібно мінімізувати: $W \rightarrow \min$.

Процес, що оптимізується, складається з окремих кроків, отже, загальний виграш (програш) може бути представлений, як сума виграшів (програшів) на окремих кроках:

$$W = \sum_{i=1}^m w_i,$$

де w_i – виграш (програш) на i -му кроці;
 m – кількість кроків.

Така властивість критерію W називається адитивністю, а сам критерій – адитивним.

Оптимізація багатокрокового процесу є керованою операцією, у процесі якої ми можемо вибирати якісь параметри або шляхи, що впливають на хід і результат процесу. При цьому на кожному кроці приймається визначене рішення, від якого залежить як значення w_i , так і значення W . Процедуру вибору рішення на кожному кроці назвемо керуванням. Математично крокове керування може бути відображено в загальному випадку деякою величиною x_i , $i=1, 2, \dots, m$, що залежно від специфіки задачі може мати вигляд змінної, функції, вектора тощо. Тоді сукупність крокових керувань являє собою керування всією операцією: $x = (x_1, x_2, \dots, x_m)$ [1, 4].

Оскільки W залежить від x , то завдання полягає в тому, щоб знайти таке керування x , при якому W досягає максимуму:

$$W^* = \max_x \{W(x)\}, \text{ або мінімуму: } W^* = \min_x \{W(x)\}.$$

Керування x , при якому досягається екстремум критерію W , називається оптимальним. Позначимо його x^* .

Оскільки керування x складається з m крокових керувань x_i , то кожне x_i також повинно бути оптимальним.

Ця задача вирішується методом динамічного програмування, в основу якого покладений принцип оптимальності Беллмана, [1 4], який стосовно цієї задачі полягає в тому, що якщо керування на кожному кроці оптимальне, то і керування в цілому також оптимальне, тобто:

$$x^* = (x^*_1, x^*_2, \dots, x^*_m);$$

$$W^* = W(x^*) = \sum_{i=1}^m w_i(x^*_i).$$

Таким чином, необхідно знайти оптимальні крокові керування $x^*_1, x^*_2, \dots, x^*_m$ і тоді їхня сукупність дасть загальний оптимум.

Основна складність задачі полягає в тому, що кроки розглянутого процесу не існують окремо самі по собі, а взаємно зв'язані один з одним і кожний попередній крок впливає на наступні. Тому для визначення оптимального крокового керування x^*_i необхідно врахувати не тільки виграш (програш) на цьому i -му кроці, але й наслідки впливу, цього керування на всі наступні кроки [4].

Рішення задачі методом динамічного програмування містить у собі два основних етапи:

1) оптимізація кроків, при цьому визначаються умовно-оптимальні керування для кожного кроку;

2) оптимізація процесу в цілому, при цьому з оптимальних кроків складається оптимальний процес шляхом вибору оптимальних керувань з умовно-оптимальних [1, 4].

Процес, що оптимізується, має початок і кінець. Очевидно, що якщо покрокову оптимізацію починати з початку, тобто з першого кроку, то практично не можливо визначити наслідки впливу цього кроку на всі наступні кроки. Виявилось, що набагато простіше зробити навпаки: оптимізувати кроки у зворотному порядку, рухаючись від кінця до початку. Діючи так, можна легко визначити наслідки кожного кроку і знайти умовно-оптимальні керування. І далі, на другому етапі, рухаючись уже від початку до кінця, з умовно-оптимальних керувань вибираються оптимальні, тобто визначається $x^* = (x^*_1, x^*_2, \dots, x^*_m)$. У цьому і полягає ідея методу динамічного програмування.

Визначення шляху мінімальної вартості методом динамічного програмування

Розглянемо метод динамічного програмування на прикладі рішення задачі визначення шляху найменшої вартості між двома

точками. При цьому шлях від початкової точки до кінцевої проходить через ряд проміжних точок. Наприклад, задано 20 точок, що в абстрагованому вигляді можна розмістити як показано на рисунку 3.1 [1, 4].

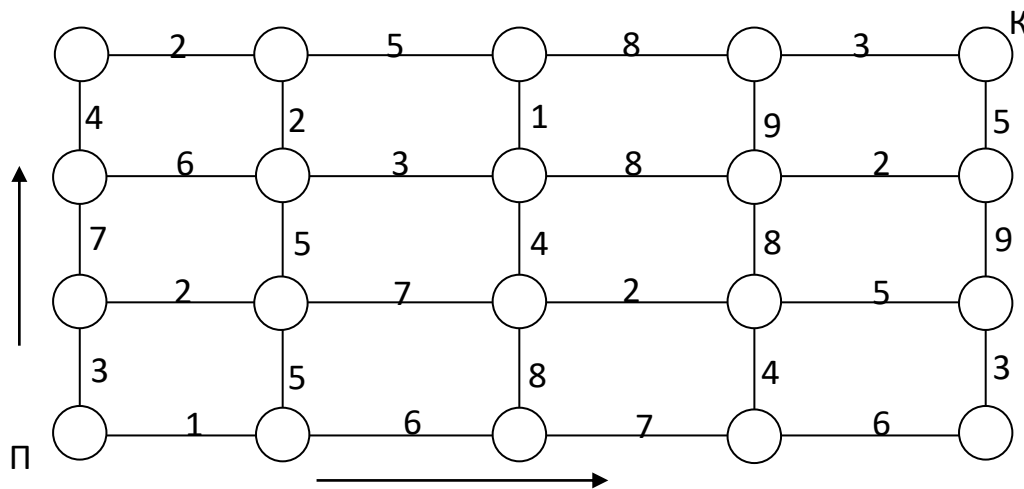


Рисунок 3.1

Відома вартість шляху між сусідніми точками, яка показана у вигляді чисел на лініях, що з'єднують точки. Ліва нижня точка є початковою, а права верхня – кінцевою. Шлях складається з кроків – відрізків ліній, що з'єднують суміжні точки.

Рухатися по шляху можна тільки нагору і вправо. Таким чином, увесь шлях буде складатися із семи кроків: три вертикальних і чотири горизонтальних. Необхідно знайти таку послідовність кроків, вартість якої буде мінімальною.

Оптимізацію кроків починаємо з кінця. У кружечках, що позначають точки, будемо записувати вартість оптимального шляху з цієї точки в кінцеву. Оскільки точки розташовані у вигляді матриці, то зручно позначити їх номерами рядка і стовпця, у яких вони розміщуються (рисунок 3.2).

У кінцевій точці (3,4) записуємо нуль, тому що вартість шляху з цієї точки в саму себе дорівнює нулеві. Далі переходимо до точки (3,3). З цієї точки в кінцеву є єдиний шлях – крок вправо, його вартість дорівнює 3, цей крок і буде оптимальним. Записуємо 3 у точці (3,3) і позначаємо оптимальний крок стрілкою. Наступна точка (3,2), з неї шлях у кінцеву точку також єдиний – крок вправо до точки (3,3) плюс крок від точки (3,3) до точки (3,4).

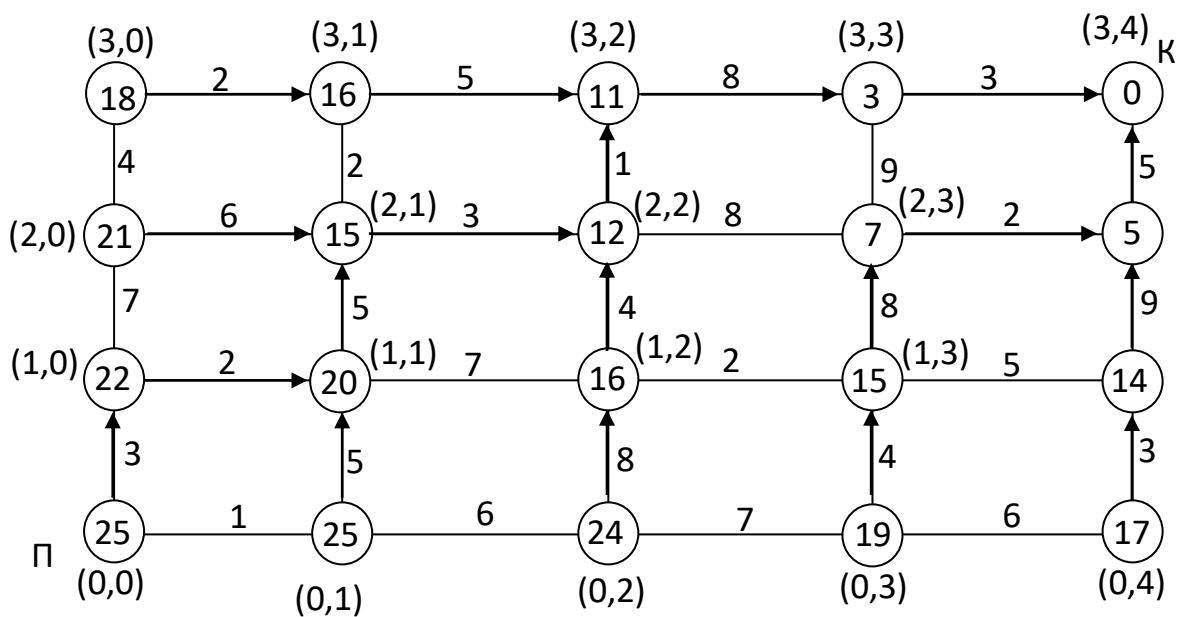


Рисунок 3.2

Вартість оптимального шляху з точки $(3,3)$ у кінцеву точку відома і дорівнює 3, отже, вартість оптимального шляху від точки $(3,2)$ до кінцевої точки дорівнює вартості шляху від точки $(3,3)$ до точки $(3,4)$ плюс вартість кроку з точки $(3,2)$ у точку $(3,3)$, тобто $3+8=11$. Записуємо в точці $(3,2)$ 11 і позначаємо стрілкою оптимальний крок. Аналогічно визначаються оптимальні кроки і вартості шляху для всіх точок, що розміщуються у верхньому рядку. Для точок, розташованих у крайньому правому стовпці дії аналогічні, тому що для них існує єдиний шлях у кінцеву точку, що складається з вертикальних кроків нагору.

Для інших точок шлях у кінцеву точку не є єдиним, тому що з них можна робити два кроки - нагору або вправо. Очевидно, що оптимальним буде крок, при якому вартість шляху до кінцевої точки буде меншою. Наприклад, із точки $(2,3)$ можна рухатися через точку $(2,4)$ і через точку $(3,3)$. У першому випадку вартість шляхи становить $2+5=7$, а в другому - $9+3=12$, отже, оптимальним є крок вправо. Позначаємо цей крок стрілкою, а в точці $(2,3)$ записуємо 7.

Аналогічно визначаються оптимальні кроки і вартості шляху для всіх інших точок. На рисунку 3.2 показаний результат оптимізації всіх кроків.

Далі з оптимальних кроків складаємо оптимальний шлях, рухаючись по стрілках з початкової точки в кінцеву (рисунок 3.3).

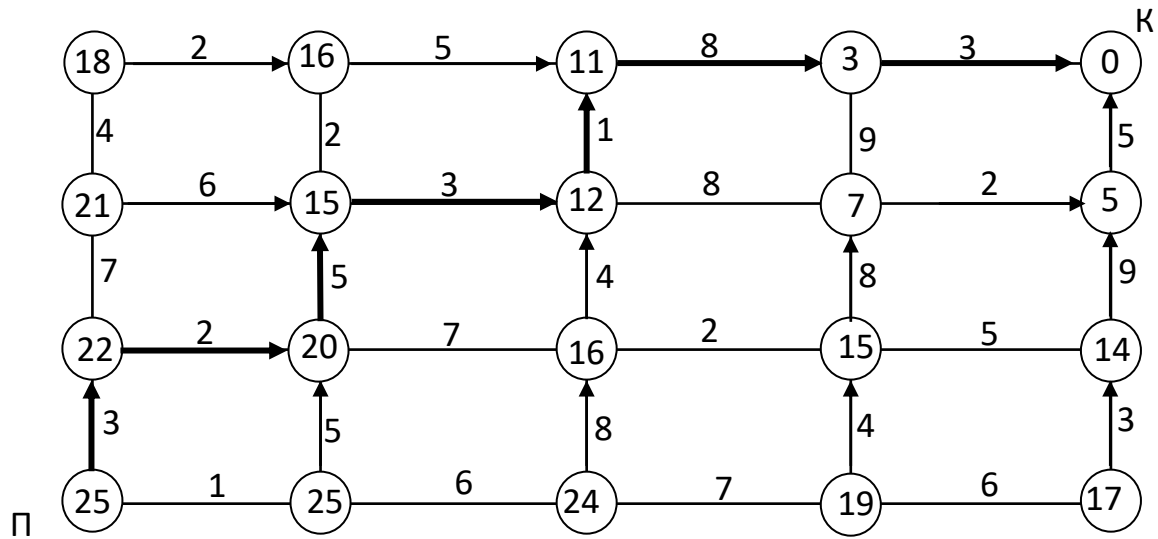


Рисунок 3.3

Для запису вищевикладених дій у вигляді алгоритму введемо позначення:

- $a(i, j)$ – вартість горизонтального кроку;
- $b(i, j)$ – вартість вертикального кроку;
- $c(i, j)$ – вартість оптимального шляху з точки (i, j) до кінцевої точки.

Ці величини зручно подати у вигляді двовимірних масивів: a - розміром 4 на 4, b - розміром 3 на 5 і c - розміром 4 на 5.

Вихідними даними є масиви a і b . Нумерацію рядків і стовпців у цих масивах починаємо з нуля, рухаючись від початкової точки до кінцевої.

Алгоритм показаний на рисунках 3.4, 3.5.

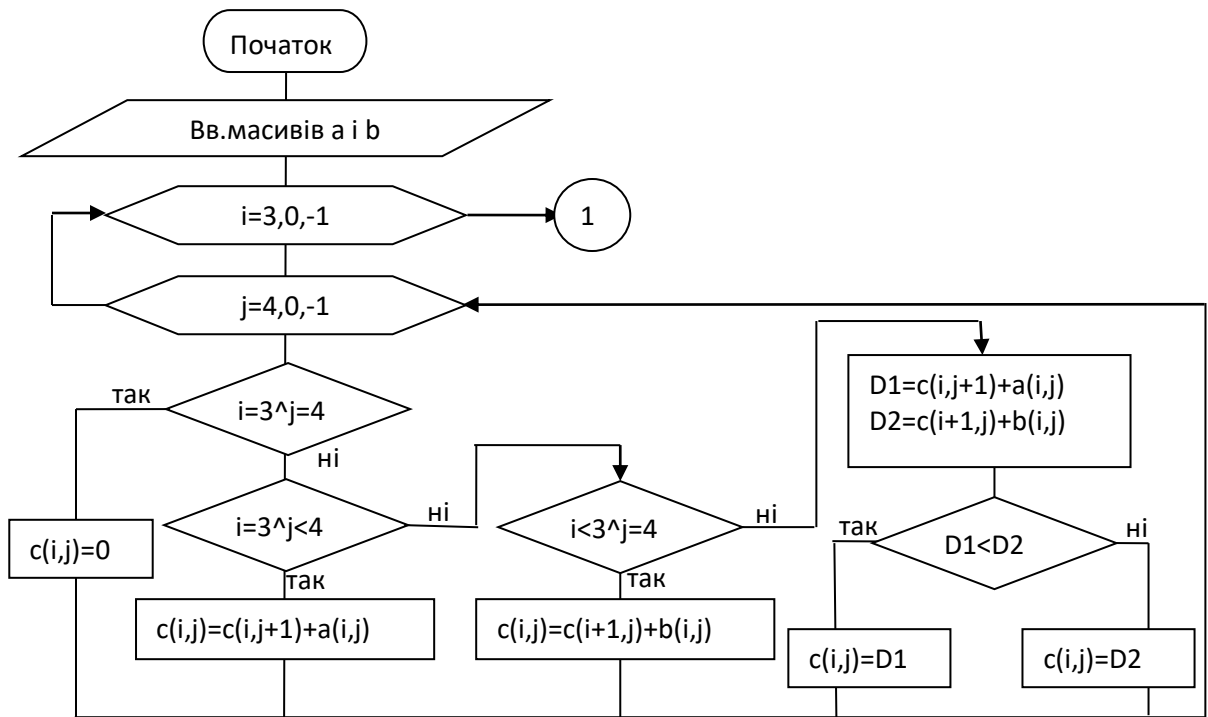


Рисунок 3.4 – Алгоритм оптимізації кроків та обчислення $c(i,j)$

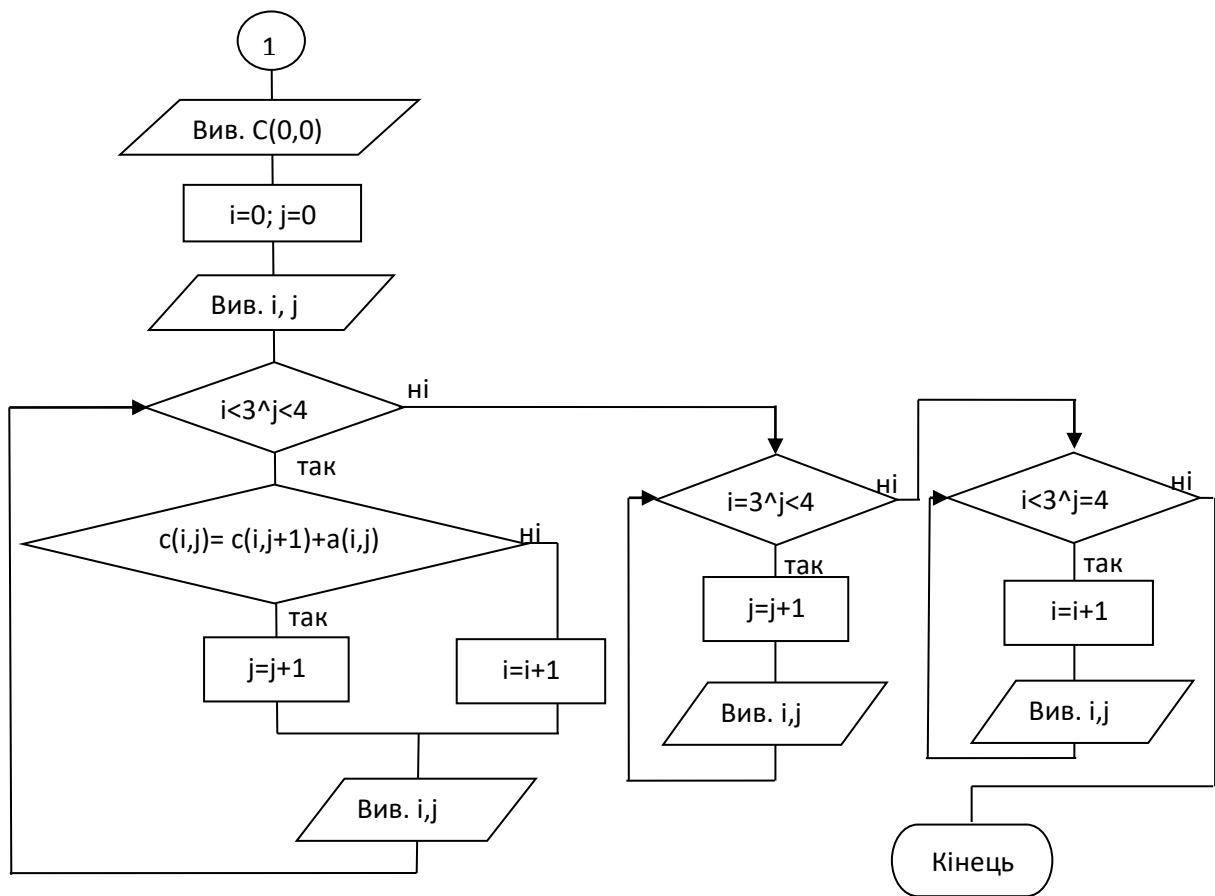


Рисунок 3.5 – Алгоритм визначення оптимального шляху

Завдання та порядок виконання

- 1 Вивчити теоретичний матеріал [1, 2, 4].
- 2 За алгоритмом, що реалізує метод динамічного програмування для задачі визначення шляху найменшої вартості між двома точками, скласти програму мовою C++ [3].
- 3 Набрати програму на комп'ютері та відладити її.
- 4 Задати в програмі довільно масиви горизонтальних – $a(i, j)$ і вертикальних – $b(i, j)$ кроків у вигляді цілих двозначних чисел. Розміри масивів: $a[4][4]$, $b[3][5]$.
- 5 Виконати програму.
- 6 Як результат вивести на екран вартість оптимального шляху із точки (0, 0) у точку (3, 4) і оптимальний шлях.

ЛАБОРАТОРНА РОБОТА 4. Дослідження моделей, що описуються системами лінійних рівнянь

Мета роботи: вивчення й реалізація на ЕОМ методів рішення систем лінійних рівнянь.

Контрольні питання

- 1 Яка система називається динамічною?
- 2 Які методи рішення систем лінійних рівнянь ви знаєте?
- 3 Які етапи містить у собі метод Гаусса?
- 4 У чому полягає сутність прямого ходу методу Гаусса?
- 5 У чому полягає сутність зворотного ходу методу Гаусса?
- 6 За якої умови можливе застосування методу Гаусса?

Навчальний матеріал

Розглянемо методи розв'язання систем лінійних алгебраїчних рівнянь (СЛАР). У загальному випадку СЛАР має такий вигляд [1, 6]:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1; \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2; \\ \dots \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = b_n, \end{cases}$$

де x_k — невідомі величини; a_{ij} — задані коефіцієнти рівнянь, b_i — вільні члени рівнянь СЛАР [1, 4].

Рішенням системи є значення невідомих x_1, x_2, \dots, x_n , що перетворюють кожне рівняння в тотожність.

Метод називається точним (прямим), якщо за кінцеве число арифметичних операцій з точними числами можна одержати точне рішення СЛАР. Інакше – метод наближений (ітераційний). Відзначимо, що точні методи — це не настільки ідеалізовані алгоритми, що на ЕОМ їх не можна реалізувати для СЛАР із цілочисловими коефіцієнтами. Використовуючи на ЕОМ арифметику цілих чисел, за допомогою точних методів можна знаходити точні рішення [1, 4, 6].

Метод Гаусса

Метод Гаусса належить до точних методів. Рішення системи рівнянь складається з двох етапів.

1 Спочатку система рівнянь шляхом послідовного виключення невідомих x_2, x_3, \dots, x_n з рівнянь від 2-го по n -не приводиться до трикутного вигляду. Цей етап ще називають прямим ходом. Приведення системи до трикутного вигляду здійснюється таким чином:

а) кожне з рівнянь від 2-го по n -не множиться на постійний коефіцієнт, який визначається з умови:

$$a_{11} = m_2 \cdot a_{22} = m_3 \cdot a_{31} = \dots = m_n \cdot a_{n1}$$

звідки випливає, що

$$m_2 = a_{11} / a_{22}, m_3 = a_{11} / a_{31}, \dots, m_n = a_{11} / a_{n1};$$

б) з рівнянь від 2-го по n -не віднімається 1-ше рівняння. При цьому система рівнянь набуде вигляду:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1; \\ a'_{22}x_2 + \dots + a'_{2n}x_n = b'_2; \\ \dots \\ a'_{n2}x_2 + \dots + a'_{nn}x_n = b'_n, \end{cases}$$

де $a'_{22} = m_2 \cdot a_{22} - a_{12}$; $a'_{23} = m_2 \cdot a_{23} - a_{13}$; ... $a'_{2n} = m_2 \cdot a_{2n} - a_{1n}$,
 $b'_2 = m_2 \cdot b_2 - b_1$ і аналогічно для інших рівнянь.

Таким чином з рівнянь від 2-го по n -не виключається невідома x_1 ;

в) аналогічно з рівнянь від 3-го по n -не виключається невідома x_2 , з рівнянь від 4-го по n -не – x_3 і так далі доти, поки система не набуде трикутного вигляду:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1; \\ a'_{22}x_2 + \dots + a'_{2n}x_n = b'_2; \\ \dots \\ a^{n-1}_{nn}x_n = b^{n-1}_n. \end{cases}$$

2 Далі на другому етапі, що називається зворотним ходом, визначаються корені системи рівнянь. Спочатку з n -го рівняння визначається

$$x_n = b^{n-1}_n / a^{n-1}_{nn}.$$

Після цього x_n підставляється в $(n-1)$ -ше рівняння, з якого визначається x_{n-1} , потім x_n і x_{n-1} підставляються в $(n-2)$ -ге рівняння, з якого визначається x_{n-2} , і так доти, поки не будуть визначені всі корені.

Метод Гаусса можливо використовувати тільки за умови, що при приведенні системи до трикутного вигляду на всіх етапах виключення невідомі коефіцієнти a_{ij} , що розташовані на головній діагоналі матриці коефіцієнтів, відмінні від нуля.

Алгоритм вирішення системи методом Гаусса наведений на рисунку 4.1.

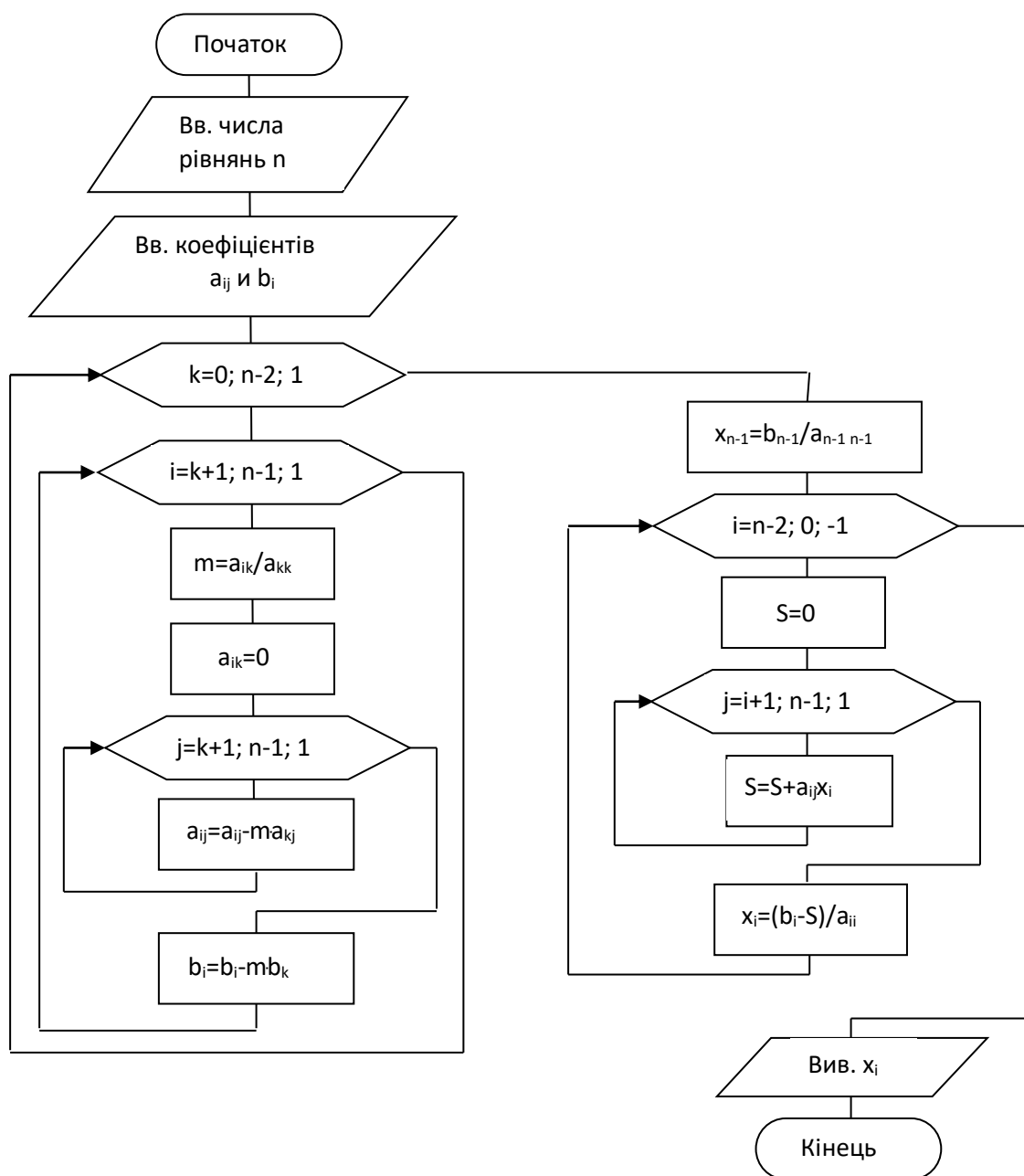


Рисунок 4.1 – Алгоритм вирішення системи методом Гаусса

Завдання та порядок виконання

- 1 Вивчити теоретичний матеріал [2, 4].
- 2 За алгоритмом, що реалізує метод Гаусса, скласти програму мовою C++.

З таблиці 4.1 вибрати варіант вхідних даних, номер варіанта відповідає номеру за списком студента в журналі викладача.

4 Набрати програму на комп'ютері та відладити її.

5 Вирішити систему із трьох лінійних рівнянь:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 = b_1; \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 = b_2; \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 = b_3. \end{cases}$$

Таблиця 4.1 – Варіанти вхідних даних

Варіант	Коефіцієнти рівнянь											
	a ₁₁	a ₁₂	a ₁₃	a ₂₁	a ₂₂	a ₂₃	a ₃₁	a ₃₂	a ₃₃	b ₁	b ₂	b ₃
1	2	-1	3	-4	5	7	2	1	-1	7	2	5
2	-4	1	2	-2	3	6	5	-4	8	2	1	-3
3	3	1	-4	3	1	5	1	2	3	7	3	1
4	-5	2	2	4	-1	7	-1	4	5	5	-3	-1
5	6	6	-3	2	-2	3	1	3	4	4	5	2
6	1	-2	2	7	4	8	-1	5	1	6	-5	-5
7	-2	4	1	-2	-8	9	2	1	6	8	2	3
8	10	-3	-1	-7	6	2	-2	-4	8	3	-6	7
9	8	-5	6	3	5	4	-3	7	9	5	3	6
10	-7	8	8	9	9	1	-4	-4	4	2	-4	-5
11	5	3	-4	3	4	3	-3	-3	5	4	8	-9
12	3	-2	9	-7	-3	3	5	4	2	3	9	4
13	-3	8	-2	2	-5	3	7	9	4	8	-7	3
14	4	4	-3	8	-2	6	2	-8	3	6	8	6
15	8	-5	1	4	-6	-7	-3	-5	6	1	3	4

ЛАБОРАТОРНА РОБОТА 5. Обробка даних експерименту методом регресійного аналізу

Мета роботи: вивчення методики побудови регресійної моделі за даними експерименту.

Контрольні питання

- 1 Дати визначення регресійної моделі.
- 2 За якої умови регресійна модель найбільш точно відповідає даним експерименту?
- 3 У чому полягає суть методу найменших квадратів?
- 4 Як визначаються коефіцієнти регресійної моделі?

Навчальний матеріал

Побудова моделі залежності за результатами експерименту. Регресійний аналіз

При дослідженні і моделюванні різних фізичних, технологічних, економічних та інших процесів часто виникає необхідність визначити характер зміни якої-небудь змінної величини залежно від зміни іншої величини, аналітичний вираз цієї залежності, значення, якого може набувати ця величина за різних умов функціонування системи. Такі дані дають змогу прогнозувати поведінку системи і дають можливість приймати обґрунтовані рішення з питань проектування систем, керування системою тощо. Для одержання таких даних проводяться експерименти з моделлю системи або процесу. Результати експериментів являють собою сукупності значень досліджуваних величин. Вплив випадкових факторів під час експерименту на систему або її модель приводить до того, що результати експерименту набувають характеру випадкових величин. Тоді для виявлення закономірностей і побудови математичної моделі процесу використовуються методи статистичної обробки даних [1, 4, 5].

У багатьох випадках задача полягає в тому, щоб знайти аналітичний вираз функції однієї змінної шляхом визначення

рівняння лінії, що проходить через задані точки (експериментальні значення досліджуваної величини). Рішення цієї задачі за допомогою інтерполяційного полінома Лагранжа недоцільне з таких причин:

1) при великій кількості точок обчислення полінома досить громіздке;

2) дані експерименту завжди мають розкид через вплив випадкових факторів, тому рівняння кривої може вийти дуже складним і його важко буде використовувати практично.

Тому найчастіше застосовується метод статистичної обробки даних, який називають регресійним аналізом.

Регресійний аналіз дає можливість побудувати математичну модель залежності, що найкраще відповідає експериментальним даним. При цьому крива апроксимуючої функції проходить не через точки, отримані в результаті експерименту, а поблизу них, даючи усереднену характеристику процесу. Така процедура називається згладжуванням експериментальної залежності (рисунок 5.1).

Апроксимуючу, або функцію, що згладжує, називають ще регресійною моделлю [1, 4]. При побудові регресійної моделі необхідно, щоб вона якомога точніше відображала характер залежності, що моделюється, і при цьому були б можливо повніше виключені випадкові відхилення. Для рішення цієї задачі використовується метод найменших квадратів, суть якого полягає в тому, що критерієм відповідності функції даним експерименту є функція помилки, обумовлена як сума квадратів відхилень експериментальних точок від кривої, що згладжує:

$$S = \sum_{i=1}^N \delta_i^2,$$

де i – номер експериментальної точки;

N – кількість експериментальних точок;

δ_i – відхилення в i -й точці.

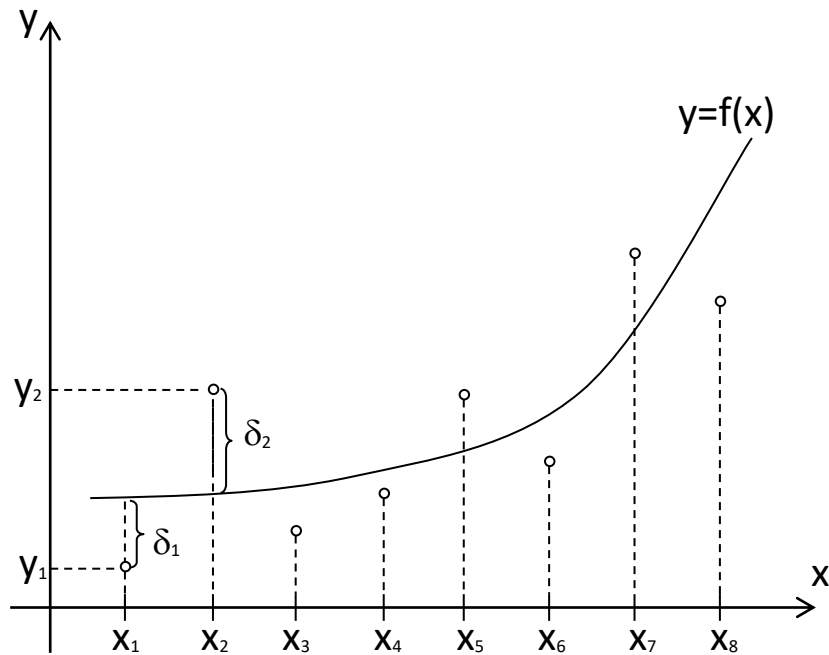


Рисунок 5.1

Необхідно знайти таку функцію $y=f(x)$, для якої функція помилки S мінімальна.

Як функція, що згладжує, частіше всього використовується степеневий поліном $y=f(A, B, C, \dots, x)$, де A, B, C, \dots – коефіцієнти рівняння. Порядок полінома вибирається з огляду на аналіз природи досліджуваної залежності і досвід фахівців. Функція помилки буде мати вигляд [4, 6]

$$S = \sum_{i=1}^N [y_i - f(A, B, C, \dots, x_i)]^2.$$

Задача зводиться до того, щоб знайти такі значення коефіцієнтів A, B, C, \dots , при яких функція S досягає мінімуму. У цьому випадку A, B, C, \dots – це змінні, від яких залежить S , і очевидно, що мінімум S можна знайти, вирішивши систему диференціальних рівнянь:

$$\begin{cases} \frac{dS}{dA} = 0; \\ \frac{dS}{dB} = 0; \\ \frac{dS}{dC} = 0; \\ \dots \end{cases}$$

Якщо апроксимуюча функція лінійна (поліном 1-го порядку), то вона має вигляд

$$y = A + Bx.$$

Функція помилки при цьому набуде вигляду

$$S = \sum_{i=1}^N [y_i - (A + Bx_i)]^2.$$

Після піднесення до квадрата одержимо вираз

$$S = \sum_{i=1}^N (y_i^2 - 2Ay_i - 2Bx_i y_i + A^2 + 2ABx_i + B^2 x_i^2).$$

Для визначення мінімуму S треба вирішити систему рівнянь:

$$\begin{cases} \frac{dS}{dA} = 0; \\ \frac{dS}{dB} = 0. \end{cases}$$

Обчисливши похідні від S по A і по B , одержимо систему рівнянь у вигляді

$$\begin{cases} \sum_{i=1}^N (-2y_i + 2A + 2Bx_i) = 0; \\ \sum_{i=1}^N (-2x_i y_i + 2Ax_i + 2Bx_i^2) = 0. \end{cases}$$

Після виконання простих перетворень система рівнянь набуде вигляду

$$\begin{cases} AN + B \sum_{i=1}^N x_i = \sum_{i=1}^N y_i; \\ A \sum_{i=1}^N x_i + B \sum_{i=1}^N x_i^2 = \sum_{i=1}^N x_i y_i. \end{cases}$$

Таким чином, отримано систему лінійних алгебраїчних рівнянь, рішення якої дасть шукані значення коефіцієнтів А і В.

Якщо як регресійна модель вибирається квадратична функція, що має вигляд полінома 2-го порядку

$$y = A + Bx + Cx^2,$$

то функція помилки набуде вигляду

$$S = \sum_{i=1}^N [y_i - (A + Bx_i + Cx_i^2)]^2.$$

Виконавши аналогічні перетворення, одержимо таку систему рівнянь:

$$\begin{cases} AN + B \sum_{i=1}^N x_i + C \sum_{i=1}^N x_i^2 = \sum_{i=1}^N y_i; \\ A \sum_{i=1}^N x_i + B \sum_{i=1}^N x_i^2 + C \sum_{i=1}^N x_i^3 = \sum_{i=1}^N x_i y_i; \\ A \sum_{i=1}^N x_i^2 + B \sum_{i=1}^N x_i^3 + C \sum_{i=1}^N x_i^4 = \sum_{i=1}^N x_i^2 y_i. \end{cases}$$

Аналогічно можуть бути отримані системи рівнянь для поліномів будь-якого порядку.

Для рішення системи лінійних алгебраїчних рівнянь можуть бути використані такі відомі методи, як метод Гаусса або метод Зейделя.

Завдання та порядок виконання

- 1 Вивчити теоретичний матеріал .
- 2 Побудувати схему алгоритму обчислення коефіцієнтів системи лінійних рівнянь:

$$\begin{cases} AN + B \sum_{i=1}^N x_i + C \sum_{i=1}^N x_i^2 = \sum_{i=1}^N y_i; \\ A \sum_{i=1}^N x_i + B \sum_{i=1}^N x_i^2 + C \sum_{i=1}^N x_i^3 = \sum_{i=1}^N x_i y_i; \\ A \sum_{i=1}^N x_i^2 + B \sum_{i=1}^N x_i^3 + C \sum_{i=1}^N x_i^4 = \sum_{i=1}^N x_i^2 y_i. \end{cases}$$

У цій системі рівнянь постійними коефіцієнтами є N та всі суми ($\sum_{i=1}^N x_i$, $\sum_{i=1}^N x_i^2$ та ін.), а змінними – величини A, B і C – коефіцієнти регресійної моделі $y = A + Bx + Cx^2$.

- 3 Скласти програму мовою C++.
- 4 Для свого варіанта вхідних даних (вибрати з таблиці відповідно до номера за списком у журналі викладача) виконати програму та визначити числові значення коефіцієнтів системи лінійних рівнянь.
- 5 Записати систему лінійних рівнянь з числовими значеннями коефіцієнтів.
- 6 Вирішити отриману систему лінійних рівнянь методом Гаусса, використовуючи програму, розроблену у лабораторній роботі 4.
- 7 Записати формулу регресійної моделі з числовими значеннями коефіцієнтів A, B і C .

ЛАБОРАТОРНА РОБОТА 6. Моделювання процесів витрати палива

Мета роботи: вивчення й реалізація на ЕОМ числових методів обчислення визначених інтегралів.

Контрольні питання

1 На чому базуються числові методи обчислення визначених інтегралів?

2 Які числові методи обчислення визначених інтегралів ви знаєте? Чим вони відрізняються один від одного?

3 Пояснити метод прямокутників.

4 Пояснити метод трапецій.

5 Пояснити метод Сімпсона.

6 Від чого і як залежить точність обчислення інтеграла числовими методами?

Навчальний матеріал

Числові методи обчислення визначених інтегралів

Ці методи використовуються, коли точне обчислення визначеного інтеграла неможливе або досить складне. Це буває в тих випадках, коли підінтегральна функція не виражається через елементарні функції, виражається, але дуже складно, або задана в табличному вигляді, тобто взагалі не має аналітичного виразу.

Як відомо, визначений інтеграл від якої-небудь функції геометрично являє собою площу фігури, обмеженої графіком цієї функції, границями інтервалу інтегрування і віссю абсцис. При використанні числових методів площа цієї фігури обчислюється не точно, а приблизно за рахунок того, що складна фігура замінюється на близьку до неї більш просту [1, 4, 6].

Числові методи інтегрування базуються на таких діях:

1) інтервал інтегрування розбивається на відрізки однакової довжини;

2) на кожному такому відрізку функція, що інтегрується, апроксимується більш простою, елементарною функцією;

3) на кожному відрізку будується елементарна фігура, обмежена графіком апроксимуючої функції, границями відрізка і віссю абсцис;

4) обчислюється площа всіх елементарних фігур;

5) обчислюється інтеграл, як сума площ елементарних фігур.

Залежно від виду апроксимуючої функції елементарні фігури будуть мати різну конфігурацію. Відповідно до цього існують різні методи інтегрування.

Метод прямокутників

Функція, що інтегрується, $y=f(x)$ апроксимується функціями нульового порядку. Елементарні фігури являють собою прямокутники (рисунок 6.1).

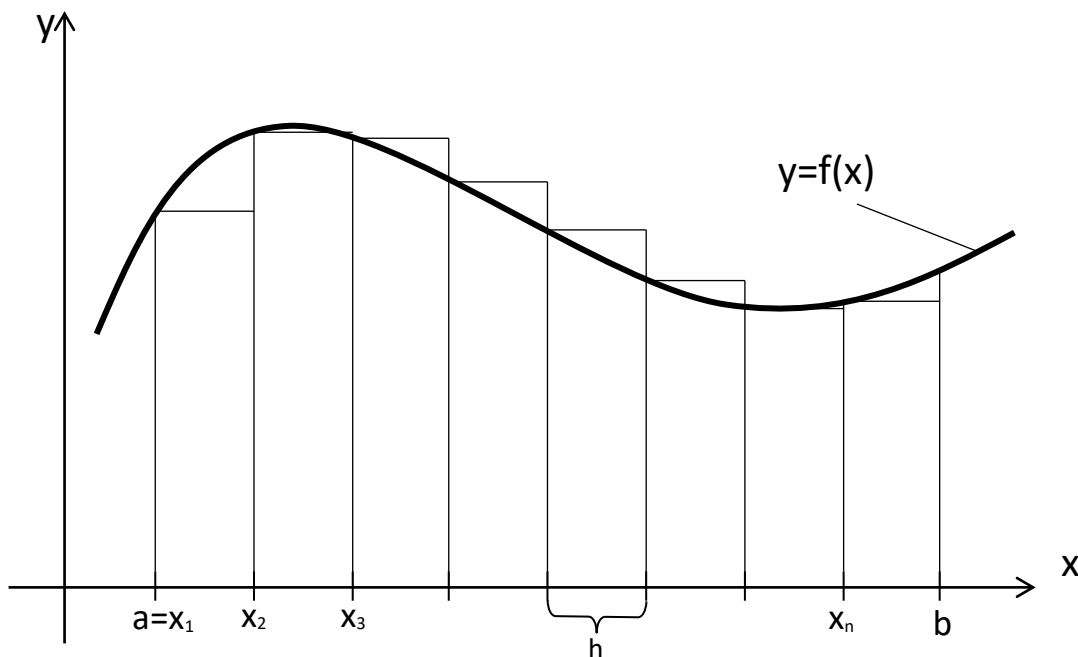


Рисунок 6.1 – Метод прямокутників

Наближене значення інтеграла обчислюється, як сума площ усіх прямокутників:

$$\int_a^b f(x)dx \approx hy_1 + hy_2 + \dots + hy_n = h(y_1 + y_2 + \dots + y_n),$$

або

$$D=h \cdot S,$$

де

$$S = \sum_{i=1}^n y_i;$$

n - кількість відрізків;
 $h=(b-a)/n$ - крок інтегрування.

Алгоритм обчислення інтеграла показаний на рисунку 6.2.

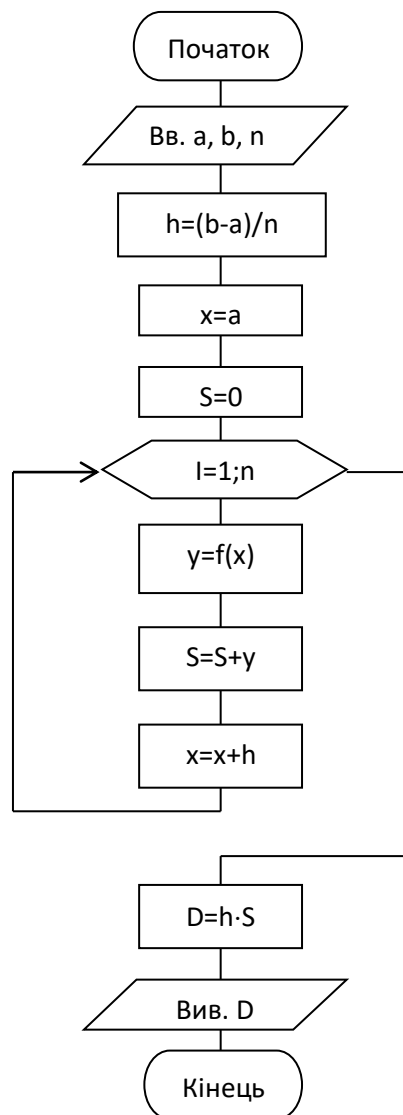


Рисунок 6.2 – Алгоритм обчислення інтеграла методом прямокутників

Метод трапецій

Функція, що інтегрується, $y=f(x)$ апроксимується функціями першого порядку (кусочно-лінійна апроксимація). Елементарні фігури являють собою трапеції (рисунок 6.3).

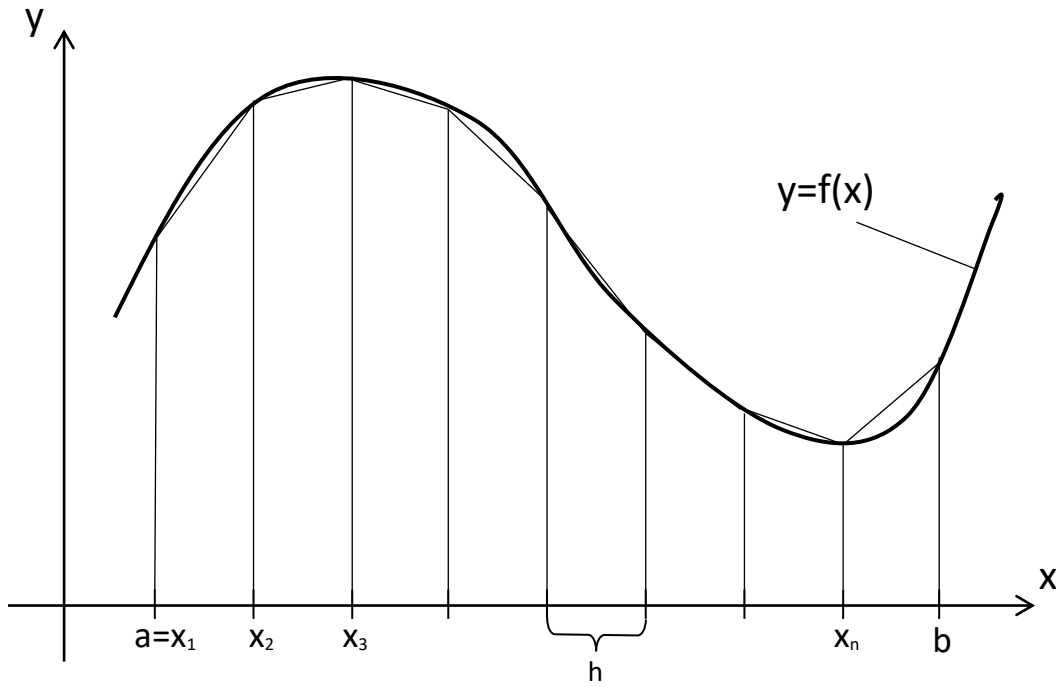


Рисунок 6.3 – Метод трапецій

Наближене значення інтеграла D обчислюється, як сума площ усіх трапецій:

$$\int_a^b f(x)dx \approx h \cdot \frac{y_1 + y_2}{2} + h \cdot \frac{y_2 + y_3}{2} + \dots + h \cdot \frac{y_n + y_{n+1}}{2},$$

або враховуючи, що $y_1=f(a)$ і $y_{n+1}=f(b)$,

$$D = h \cdot [f(a)/2 + y_2 + y_3 + \dots + y_n + f(b)/2].$$

Позначивши $S = \sum_{i=2}^n y_i$, одержимо

$$D = h \cdot [f(a)/2 + S + f(b)/2].$$

Метод трапецій є більш точним, ніж метод прямокутників, тому що апроксимуюча функція проходить ближче до функції, що інтегрується.

Алгоритм обчислення інтеграла показаний на рисунку 6.4.

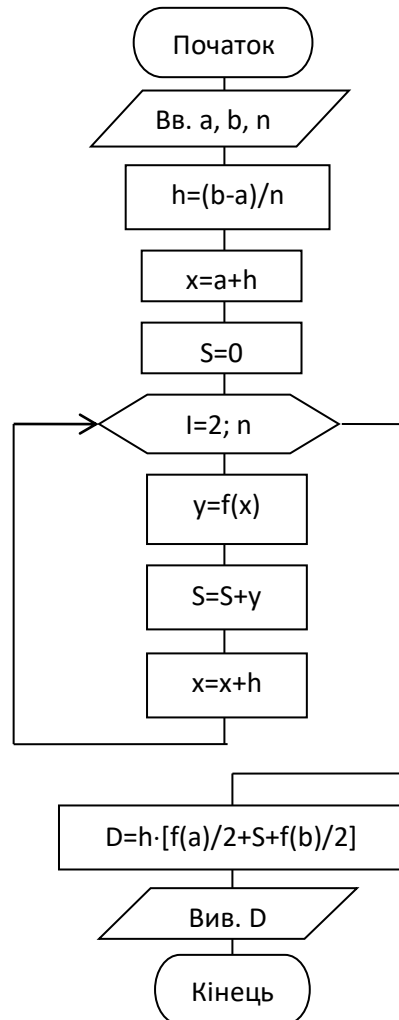


Рисунок 6.4 - Алгоритм обчислення інтеграла методом трапеції

Метод Сімпсона

Функція, що інтегрується, $y=f(x)$ апроксимується функціями другого порядку. Елементарні фігури являють собою криволінійні трапеції (рисунок 6.5). Кількість відрізків n повинна бути непарною [1, 4, 5].

Інтеграл приблизно обчислюється за формулою

$$\int_a^b f(x)dx \approx \frac{h}{3}(y_1 + 4y_2 + 2y_3 + \dots + 4y_{n-1} + 2y_n + y_{n+1}).$$

З урахуванням того, що $y_1=f(a)$ і $y_{n+1}=f(b)$, а також

$$S_1=y_2+y_4+\dots+y_{n-1};$$

$$S_2=y_3+y_5+\dots+y_n,$$

одержимо такий вираз для обчислення інтеграла:

$$D = \frac{h}{3} \cdot [f(a) + 4S_1 + 2S_2 + f(b)].$$

Алгоритм показаний на рисунку 6.6.

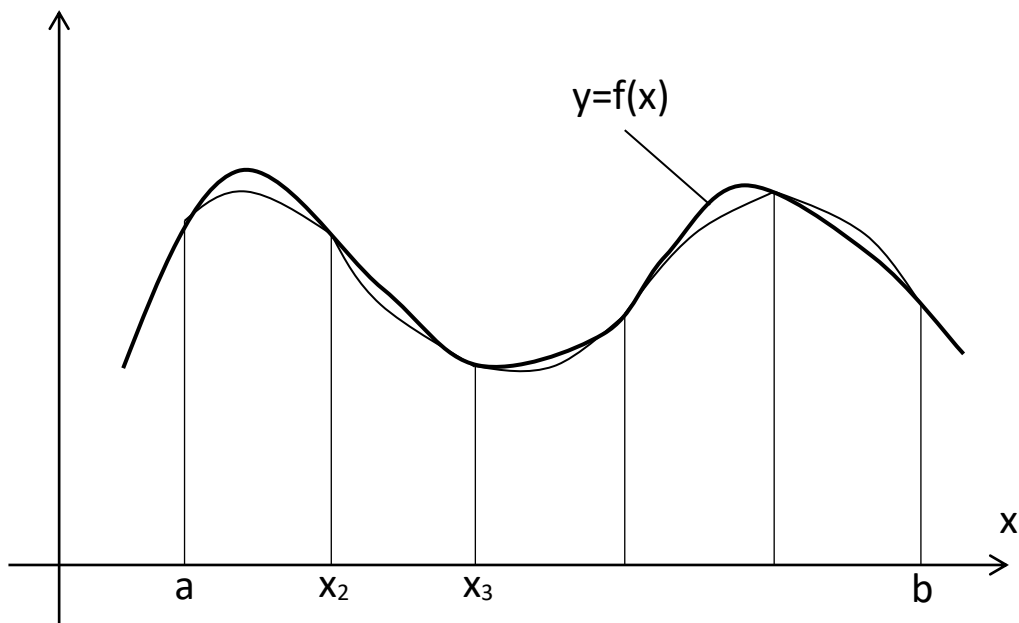


Рисунок 6.5

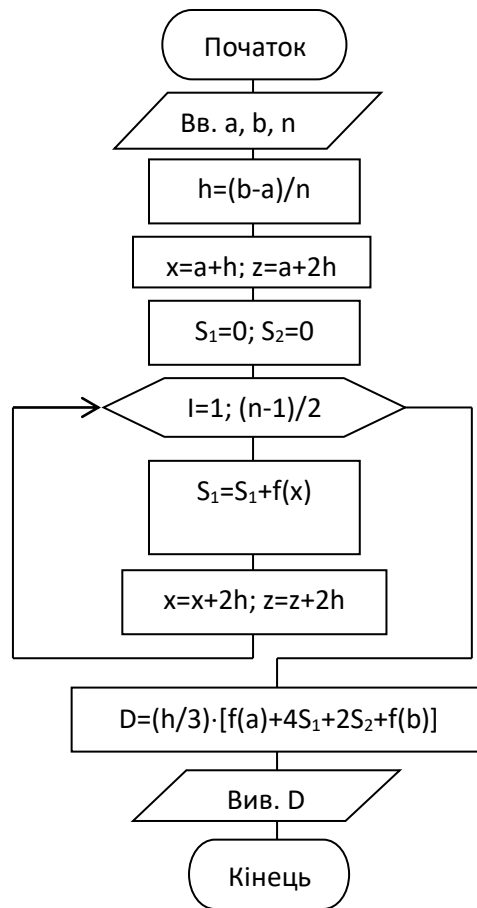


Рисунок 6.6 – Алгоритм обчислення інтеграла методом Сімпсона

Завдання та порядок виконання

- 1 Вивчити теоретичний матеріал [1, 4].
- 2 Для методу обчислення визначеного інтеграла, що відповідає своєму варіанту вхідних даних, скласти програму мовою C++. Варіант вибрати з таблиці вхідних даних відповідно до номера за списком у журналі викладача.
- 3 Набрати та відладити програму.
- 4 Ввести вхідні дані та виконати програму.

ЛАБОРАТОРНА РОБОТА 7. Моделі оптимального керування запасами

Мета роботи: вивчення моделей оптимального керування запасами ресурсів на підприємстві в системі матеріально-технічного забезпечення.

Контрольні питання

1 Чим характеризується система матеріально-технічного забезпечення?

2 Чому недоцільний як занадто великий, так і занадто малий запас ресурсів на складі?

3 Який критерій використовується в задачі оптимального керування запасами, які величини при цьому є змінними оптимізації?

4 У чому полягає математичний метод, використовуваний для визначення мінімуму функції витрат?

5 Що таке «стратегія керування»?

Навчальний матеріал

Методи і моделі оптимального керування запасами

У будь-якому виробничому процесі використовуються ресурси, постачання якими здійснює система матеріально-технічного забезпечення (МТЗ). Для нормальної безперебійної роботи виробництва на підприємстві необхідний запас ресурсів. Однією з основних задач в організації МТЗ є визначення оптимального розміру запасів. Необхідність вирішення такої задачі обумовлена тим, що як занадто малий, так і занадто великий запаси недоцільні [4].

З одного боку, запас треба збільшувати, тому що існують такі фактори:

- різна інтенсивність постачань і споживання;
- випадкові зміни попиту на продукцію й обсягів постачань ресурсів;

- непередбачені збої у виробничому процесі через причини, не пов'язані з постачанням;
- збої в системі постачання;
- зміна кон'юнктури ринку;
- дискретність постачань і безперервність споживання.

Через це недостатній запас може стати причиною вичерпання ресурсів і зриву виробничого процесу, що спричинить великі втрати.

З другого боку, надмірно великий запас також призведе до втрат тому, що:

- збільшаться витрати на збереження ресурсу;
- з'являться втрати за рахунок фізичного старіння ресурсу, тому що він буде довго зберігатися;
- при тривалому зберіганні може статися моральне старіння ресурсу;
- кошти, витрачені на придбання ресурсу, будуть виведені з обороту.

Задача оптимального керування запасами полягає у визначенні оптимальних моментів часу (періодичності) і обсягів постачань, що забезпечують поповнення запасів [4].

Як правило, системи МТЗ характеризуються:

- великою кількістю постачальників;
- великою номенклатурою споживаних ресурсів;
- складністю зв'язків між споживачами і постачальниками;
- наявністю змін у структурі МТЗ і номенклатурі ресурсів.

За таких умов моделі оптимального керування запасами складні і не завжди можуть бути формалізовані. Тому на практиці найчастіше використовуються спрощені моделі, що дають змогу одержати практично корисні наближені рішення ціною не занадто великих витрат. При цьому можуть бути використані математичні методи із застосуванням ЕОМ [4].

Розглянемо спрощений варіант задачі керування запасами, коли в системі МТЗ є один постачальник і один споживач ресурсу. При цьому використовується один вид ресурсу. Для побудови математичної моделі введемо такі позначення:

L – інтенсивність постачання ресурсу (кількість ресурсу, що поставляється в одиницю часу);

М – інтенсивність споживання ресурсу (кількість ресурсу, що витрачається в одиницю часу);

Y – максимальний запас ресурсу на складі;

n – вартість зберігання одиниці ресурсу в одиницю часу;

K – максимально припустимий дефіцит ресурсу;

P – вартість збитків унаслідок дефіциту одиниці ресурсу в одиницю часу;

S – вартість організації постачання;

T – період постачання;

C – витрати на керування запасами протягом періоду T.

Критерієм оптимізації є витрати C, що містять у собі витрати на організацію постачання і зберігання ресурсу, а також втрати внаслідок дефіциту ресурсу на складі. Очевидно, що оптимальне керування запасом – це таке керування, при якому C досягає мінімуму.

Якщо виразити критерій у вигляді цільової функції, то в загальному випадку вона буде мати вигляд:

$$C=f(L, M, Y, n, K, p, S, T).$$

У цьому випадку вважаємо, що величини L, M, n, K, p і S постійні. Тоді функція витрат C буде залежати тільки від Y і T. Отже, Y і T є змінними оптимізації і задача оптимізації буде зводитися до того, щоб визначити значення Y і T, для яких C досягне мінімуму [4].

Для спрощення математичної моделі подамо процес постачань і споживання ресурсу як безперервний і лінійний (реально він дискретний і найчастіше нелінійний). Тоді динаміку зміни запасу ресурсу на складі протягом періоду T можна відобразити на графіку (рисунок 7.1).

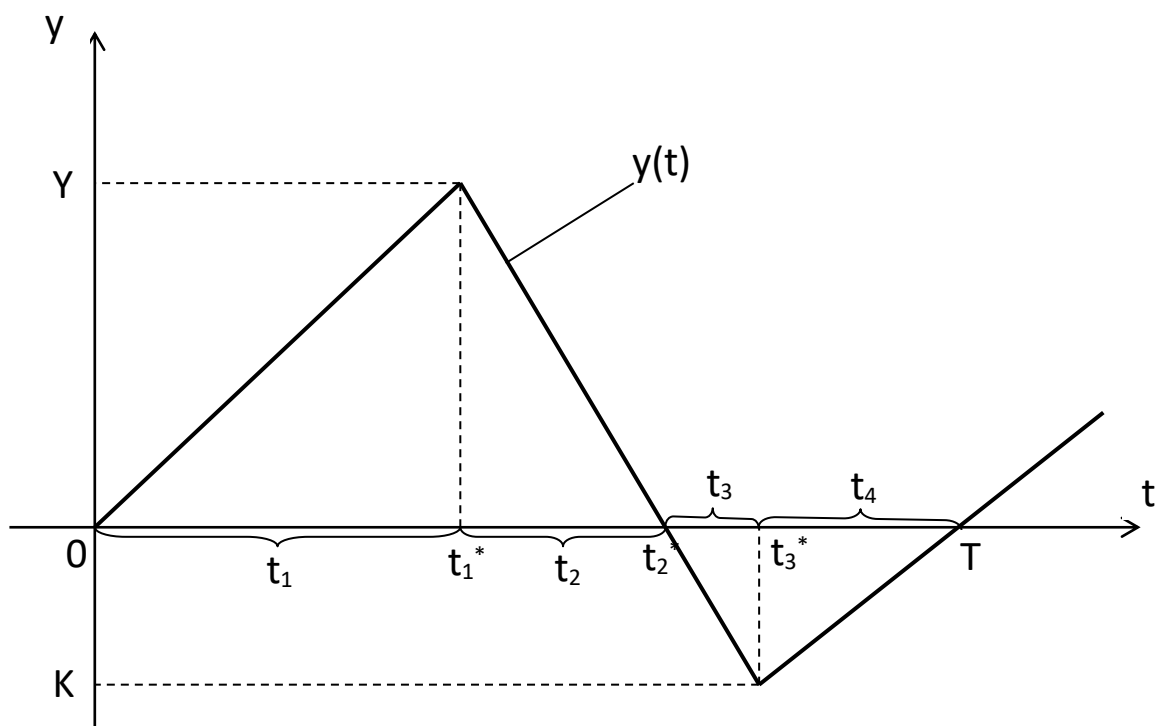


Рисунок 7.1

Функція $y(t)$ показує залежність запасу (y) ресурсу на складі від часу.

На інтервалі t_1 ідуть постачання і споживання ресурсу, при цьому $L > M$.

На інтервалах t_2 і t_3 постачань немає, а відбувається тільки споживання ресурсу, при цьому на інтервалі t_3 запасу на складі вже нема, росте дефіцит ресурсу і виробництво працює за рахунок запасу ресурсу, що є у виробничому приміщенні.

На інтервалі t_4 відновлюються постачання, але ресурс, що надходить, іде не на склад, а прямо у виробництво, за рахунок чого зменшується дефіцит.

Функція витрат C на періоді часу від 0 до T виражається такою формулою:

$$C = S + \int_0^T c(t) dt,$$

де $c(t)$ — частина витрат, що залежить від наявності запасу на складі.

З огляду на те, що $T = t_1 + t_2 + t_3 + t_4$, а також те, що витрати на інтервалах t_1 і t_2 містять у собі вартість зберігання запасу ресурсу

на складі, а на інтервалах t_3 і t_4 – втрати від дефіциту, одержимо такий вираз:

$$C = S + n \cdot \int_0^{t_1^*} y(t)dt + n \cdot \int_{t_1^*}^{t_2^*} y(t)dt + p \cdot \int_{t_2^*}^{t_3^*} y(t)dt + p \cdot \int_{t_3^*}^T y(t)dt. \quad (7.1)$$

Використовуючи геометричну інтерпретацію інтегралів як площ фігур (у цьому випадку трикутників), вираз 7.1 набуде вигляду

$$C = S + n \cdot 0,5 \cdot Y \cdot t_1 + n \cdot 0,5 \cdot Y \cdot t_2 + p \cdot 0,5 \cdot K \cdot t_3 + p \cdot 0,5 \cdot K \cdot t_4. \quad (7.2)$$

Інтервали t_1 , t_2 , t_3 і t_4 виражаються через L , M , Y і K таким чином:

$$t_1 = \frac{Y}{L-M}; t_2 = \frac{Y}{M}; t_3 = \frac{K}{M}; t_4 = \frac{K}{L-M}. \quad (7.3)$$

Зважаючи на це:

$$\begin{aligned} t_1 + t_2 &= \frac{LY}{M(L-M)}; \\ t_3 + t_4 &= \frac{LK}{M(L-M)}. \end{aligned} \quad (7.4)$$

Позначивши $B = \frac{L}{M(L-M)}$, одержимо

$$\begin{aligned} t_1 + t_2 &= B \cdot Y; \\ t_3 + t_4 &= B \cdot K; \\ K &= \frac{T}{B} - Y. \end{aligned} \quad (7.5)$$

З урахуванням виразів (7.3) – (7.5) формула (7.2) після простих перетворень набуде вигляду

$$C = \frac{S}{T} + \frac{Y^2 B(n+p)}{2T} - pY + \frac{pT}{2B}. \quad (7.6)$$

Мінімум C можна знайти, вирішивши систему рівнянь

$$\begin{cases} \frac{dC}{dT} = 0; \\ \frac{dC}{dY} = 0. \end{cases}$$

У результаті рішення цієї системи диференціальних рівнянь одержимо формули оптимальних значень T і Y :

$$\begin{aligned} T &= \sqrt{\frac{2S(1+n/p)}{Mn(1-M/L)}}; \\ Y &= \sqrt{\frac{2SM(1-M/L)}{n(1+n/p)}}. \end{aligned} \quad (7.7)$$

Підставивши (7.7) у (7.6), одержимо формулу мінімального значення C

$$C = \sqrt{\frac{2SMn(1-M/L)}{1+n/p}}. \quad (7.8)$$

Ця задача може також бути вирішена з урахуванням стратегії керування. Стратегія керування це сукупність правил і умов, що визначають специфіку керування запасами у виробничому процесі.

Розглянутий вище варіант задачі назвемо стратегією А. У цьому випадку в системі допускається дефіцит і постачання йдуть безупинно.

Інші варіанти стратегії керування наведено нижче.

Стратегія В

У системі допускається дефіцит, тобто $K > 0$. Постачання здійснюються дискретно партіями розміром $Y+K$ на початку періоду постачання, отже, $L = \infty$. З урахуванням цих умов вирази (7.7), (7.8) будуть мати вигляд:

$$T = \sqrt{\frac{2S(1+n/p)}{Mn}}; Y = \sqrt{\frac{2SM}{n(1+n/p)}}; C = \sqrt{\frac{2SMn}{1+n/p}}.$$

Стратегія С

Неприпустимий дефіцит: $K=0$. Постачання йдуть безупинно. Зрив постачань неприпустимий: $p=\infty$. Формули (7.7), (7.8) набудуть такого вигляду:

$$T = \sqrt{\frac{2S}{Mn(1 - M/L)}}; Y = \sqrt{\frac{2SM(1 - M/L)}{n}}; C = \sqrt{2SMn(1 - M/L)}.$$

Стратегія Д

Неприпустимий зрив постачань: $p=\infty$. Неприпустимий дефіцит: $K=0$. Постачання здійснюються дискретно партіями розміром $Y+K$ на початку періоду постачання: $L=\infty$. У цьому випадку:

$$T = \sqrt{\frac{2S}{Mn}}; Y = \sqrt{\frac{2SM}{n}}; C = \sqrt{2SMn}.$$

Завдання та порядок виконання

- 1 Вивчити теоретичний матеріал [3].
 - 2 Скласти схему алгоритму обчислення Y , T і C для чотирьох стратегій керування.
 - 3 Скласти за алгоритмом програму мовою C++ згідно з варіантом таблиці 7.1.
 - 4 Вибрати варіант вхідних даних (таблиця 7.1).
 - 5 Відладити та виконати програму на комп'ютері.
- Варіант вхідних даних вибирається за таблицею 7.1:

Таблиця 7.1 – Вхідні дані

Параметр	S	L	M	n	p
Значення	2	5	2-4	0,2*N	2

N – номер варіанта задачі (номер за списком у журналі викладача)

ЛАБОРАТОРНА РОБОТА 8. Метод Монте-Карло у задачах параметричної оптимізації

Мета роботи: вивчення методів рішення задач оптимізації методом випадкового пошуку.

Контрольні питання

1 У яких випадках при моделюванні систем використовуються випадкові числа?

2 Які існують методи генерації випадкових чисел?

3 Які функції генерації випадкових чисел є в мовах програмування?

4 У чому полягає метод Монте-Карло (випадкового пошуку)?

5 У яких випадках доцільно застосовувати випадковий пошук?

6 Що передбачає метод простого перебору визначення екстремуму функції?

7 Що передбачає метод золотого перерізу визначення екстремуму функції ?

Навчальний матеріал

Під статистичним моделюванням заведено розуміти побудову і дослідження математичної моделі, у якій поведінка системи відображається з урахуванням впливу на систему випадкових факторів зовнішнього середовища. Модель реалізується у вигляді алгоритму, що дає змогу на ЕОМ одержати статистичні дані про процеси, що відбуваються в системі [1, 4].

Статистичне моделювання застосовується в таких галузях:

1 Дослідження стохастичних систем

Стохастичними називаються системи, на поведінку яких впливають випадкові фактори зовнішнього середовища, у результаті чого вхідні і вихідні величини, а також внутрішні характеристики або параметри системи змінюються випадковим чином.

2 Рішення детермінованих задач

Суть використання статистичного моделювання для рішення детермінованих задач полягає в тому, що в детерміновану задачу навмисно вводиться елемент випадковості, або детермінована система замінюється еквівалентною їй стохастичною системою. До таких задач належать різні інженерні, проектні і наукові задачі, наприклад оптимізаційні задачі, рішення яких детермінованим (аналітичним) шляхом або досить складне, або неможливе. Використання статистичної моделі детермінованої задачі дає змогу одержати наближене рішення, цілком достатнє для практичних цілей.

У процесі статистичного моделювання проводиться серія іспитів з моделлю, у результаті яких визначаються множини значень досліджуваних величин [1, 4]. При досить великій кількості іспитів отримані результати набувають статистичної стійкості, що дає змогу виявити закономірності функціонування системи й одержати досить точні оцінки досліджуваних величин. При цьому як характеристики приймаються не конкретні значення величин, а їхні математичні очікування:

$$m_x = \lim\left[\left(\sum_{i=1}^n x_i\right) / n\right] \text{ при } n \rightarrow \infty,$$

де x_i – значення випадкової величини при i -му іспиті;
 n – кількість іспитів.

Моделювання випадкових величин

Для статистичного моделювання систем на ЕОМ необхідне формування значень випадкових величин. Це робиться за допомогою генераторів випадкових чисел.

Існують три методи генерації випадкових чисел: апаратний, табличний і алгоритмічний [4].

Апаратний метод полягає в тому, що випадкові числа виробляються спеціальним електронним пристроєм, приєднаним до ЕОМ, як зовнішній пристрій. Джерелом випадкових чисел є який-небудь природний фізичний процес, найчастіше електромагнітної природи, що має випадковий характер (рисунок 8.1).



Рисунок 8.1 – Апаратний метод генерації випадкових чисел

Як фізичний процес використовуються шуми в електронних приладах, явища розпаду радіоактивних елементів та ін.

Переваги цього методу:

- запас чисел необмежений;
- не вимагає витрати часу на обчислення і пам'яті ЕОМ.

Недоліки

- нестабільність фізичного процесу;
- неможливість відтворення послідовності випадкових чисел;
- необхідний спеціальний пристрій.

Табличний метод полягає в тому, що випадкові числа, отримані яким-небудь способом, записуються в пам'ять ЕОМ у вигляді масиву і потім шляхом звертання до масиву використовуються при формуванні випадкової величини.

Перевагою є можливість багаторазового відтворення послідовності чисел.

До недоліків варто віднести обмеженість запасу чисел, а також великий обсяг пам'яті для їх зберігання.

Однак найчастіше використовується алгоритмічний метод, коли випадкові числа генеруються в ЕОМ за допомогою спеціальних алгоритмів. Стандартні програми (функції), що генерують випадкові числа, є практично у всіх мовах програмування. У мові BASIC – це функція RND, у мові C ++ - функції rand і random.

Переваги алгоритмічного методу:

- стабільність генерації випадкових чисел;
- не вимагає багато пам'яті ЕОМ;
- не використовує спеціальні зовнішні пристрої.

До недоліків належать помітні витрати машинного часу, а також те, що числа, які генеруються, є не суто випадковими, а псевдовипадковими, тому що існує алгоритм їх генерації. Для генерації випадкових чисел алгоритму в принципі бути не може.

Використання методу статистичних іспитів для рішення задач оптимізації

Для рішення задач оптимізації набули поширення алгоритми, основані на методі статистичних іспитів, який називається ще методом випадкового пошуку, або методом Монте-Карло. Суть цього методу полягає в тому, що в процес пошуку оптимального рішення включається елемент випадковості або гри. При цьому аналізовані варіанти генеруються не за якимось визначеним алгоритмом, а випадково, так само, як у грі черговий хід визначається числом, що випадає при киданні кісток. При рішенні таких задач на ЕОМ випадковий пошук реалізується за допомогою генераторів випадкових чисел [4].

У найбільш загальному випадку задача оптимізації полягає у визначенні значень змінних x_1, x_2, \dots, x_n , для яких цільова функція $W=F(x_1, x_2, \dots, x_n)$ досягає максимуму або мінімуму при виконанні відповідних обмежень. Як приклад такої задачі можна навести задачу оптимізації проектованої технічної системи, де x_1, x_2, \dots, x_n – це параметри системи. При рішенні цієї задачі методом Монте-Карло значення x_1, x_2, \dots, x_n задаються випадково. Доцільність використання випадкового пошуку обумовлена тим, що при кількості змінних $n > 3$ кількість варіантів величезна, а складність генерації варіантів зростає зі збільшенням n і тому вирішити таку задачу перебором усіх варіантів практично не можливо. Практика показала, що метод Монте-Карло в таких випадках дає рішення, досить близьке до оптимального.

Розглянемо аналогічну задачу з однією змінною x . Необхідно знайти значення x , при якому цільова функція $y=f(x)$ досягає максимуму:

$$y=f(x) \rightarrow \max \text{ при } x \in [x_n, x_k],$$

де x_H - початкове значення x ;
 x_K - кінцеве значення x .

Рішення задачі методом Монте-Карло полягає в такому:

1) у заданому інтервалі $[x_H, x_K]$ генеруються випадкові значення змінної x . Кількість таких значень N задається заздалегідь;

2) для кожного значення x обчислюється $f(x)$.

3) кожне наступне значення $f(x)$ порівнюється з попереднім і запам'ятовується максимальне з них, а також значення x , що відповідає максимальному $f(x)$.

Алгоритм рішення цієї задачі наведено на рисунку 8.2.

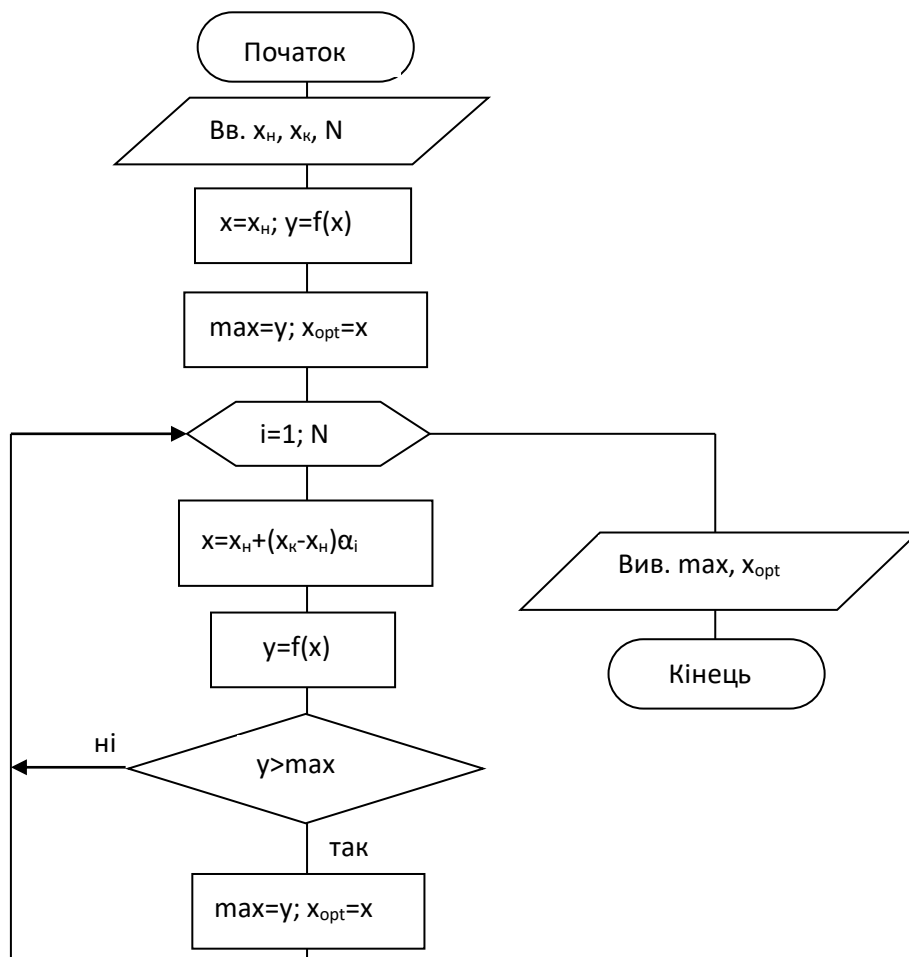


Рисунок 8.2 – Алгоритм методу Монте-Карло

У цьому алгоритмі, крім вищевказаних, використовуються також такі позначення:

\max - максимальне значення y ;

α_i - випадкове число в діапазоні від 0 до 1;

x_{opt} - оптимальне значення x , для якого $y = \max$.

При реалізації алгоритму мовами програмування можуть бути використані стандартні функції генерації випадкових чисел.

Використання методу золотого перерізу для рішення задач оптимізації

При моделюванні систем трапляються випадки, коли математична модель використовується для управління якимось процесом у режимі реального часу й оптимальне рішення необхідно визначити дуже швидко, практично миттєво, особливо якщо процес відбувається з високою швидкістю [4]. У таких випадках основною вимогою до методу є мінімум часу обчислень, тому використання таких методів, як простий перебір або випадковий пошук недоцільно. Для вирішення таких задач набули поширення методи пошуку оптимального рішення, побудовані на звуженні інтервалу невизначеності шляхом симетричного ділення відрізка, що містить екстремум, відносно його середини, сутність яких полягає у такому:

- на інтервалі $[a, b]$ певним способом задаються дві точки x_1 та x_2 , розміщені симетрично відносно середини інтервалу (рисунок 8.3);

- у точках x_1 та x_2 обчислюється значення $f(x)$ і порівнюються один з одним;

- з відрізків $[a, x_1]$ та $[x_2, b]$ відкидається той, який не містить екстремуму;

- для відрізка, який містить екстремум, тобто $[a, x_2]$ або $[x_1, b]$, процедура ділення повторюється доти, поки довжина цього відрізка не досягне заданої точності ϵ , після чого за точку екстремуму приймається середина відрізка.

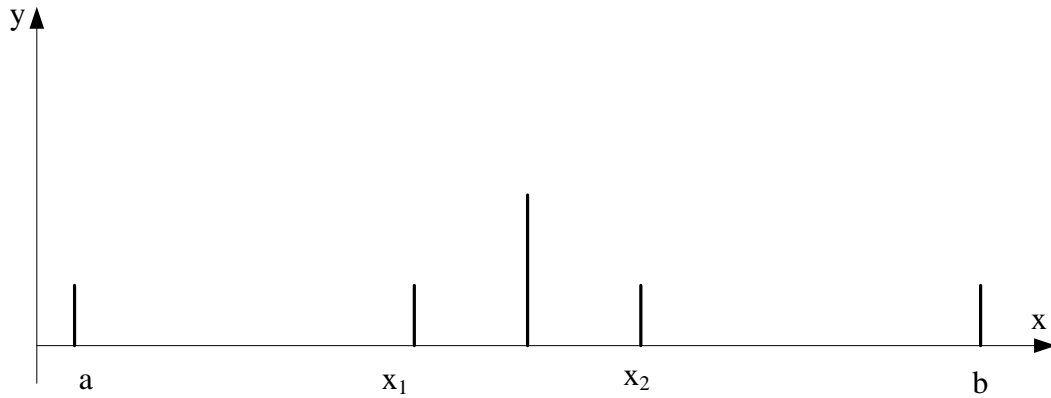


Рисунок 8.3 – Розбиття інтервалу $[a, b]$ симетричними відносно центра точками x_1 та x_2

Методи, що розглядаються, можуть бути використані тільки для унімодальних функцій [1]. Розглянемо умови унімодальності для задачі пошуку мінімуму $f(x)$.

Функція $f(x)$ називається унімодальною на інтервалі $[a, b]$, якщо виконуються такі умови:

- на інтервалі $[a, b]$ є тільки одна точка мінімуму x^* , $f(x^*) = \min(f(x))$, $x \in [a, b]$;

- для будь-яких точок $x_1, x_2 \in [a, b]$ виконується співвідношення:

якщо $x^* \leq x_1 \leq x_2$, то $f(x^*) \leq f(x_1) \leq f(x_2)$ і $\min(f(x))$ міститься на відрізку $[a, x_2]$ (рисунок 8.4)

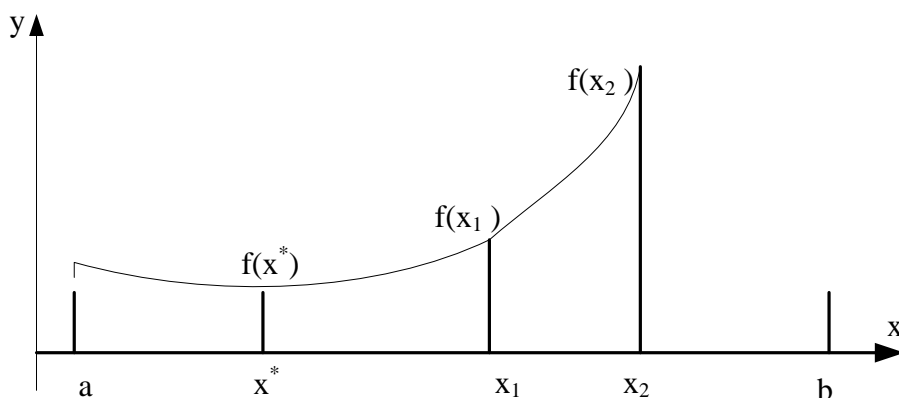


Рисунок 8.4 – Ілюстрація унімодальності функції $f(x)$ якщо $x^* \leq x_1 \leq x_2$

якщо $x^* \geq x_1 \geq x_2$, то $f(x^*) \leq f(x_1) \leq f(x_2)$ і $\min(f(x))$ міститься на відрізку $[x_1, b]$ (рисунок 8.5).

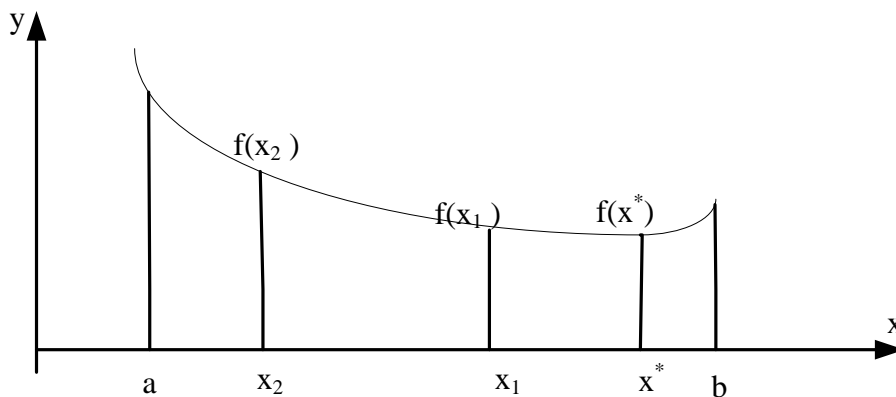


Рисунок 8.5 – Ілюстрація унімодальності функції $f(x)$
якщо $x^* \geq x_1 \geq x_2$

Таким чином, унімодальна функція монотонна з обох сторін від точки мінімуму.

Одним з методів, де використовується симетричне ділення відрізка, є метод золотого перерізу, у якому точки x_1 та x_2 ділять інтервал $[a, b]$ у співвідношенні золотого перерізу.

Золотим перерізом називається таке ділення відрізка на дві частини, при якому увесь відрізок належить до більшої частини так само, як більша частина до меншої.

Якщо візьмемо довжину відрізка за 1, то довжина більшої частини $\tau=0,618034$.

Це число τ є співвідношенням золотого перерізу. Одним із його властивостей є те, що якщо відрізок $[a, b]$ точка x_1 ділить у співвідношенні золотого перерізу, то симетрична їй точка x_2 ділить відрізок $[x_1, b]$ також у співвідношенні золотого перерізу. Аналогічно точка x_1 ділить у співвідношенні золотого перерізу відрізок $[a, x_2]$. Таким чином, на кожному кроці ділення відрізка, крім першого, можна обчислювати значення $f(x)$ тільки в одній точці: x_1 або x_2 , а значення $f(x)$ у другій точці береться з попереднього кроку, де воно вже було обчислено. За рахунок цього зменшується обсяг і час обчислень.

Алгоритм методу золотого перерізу подано на рисунку 8.6.

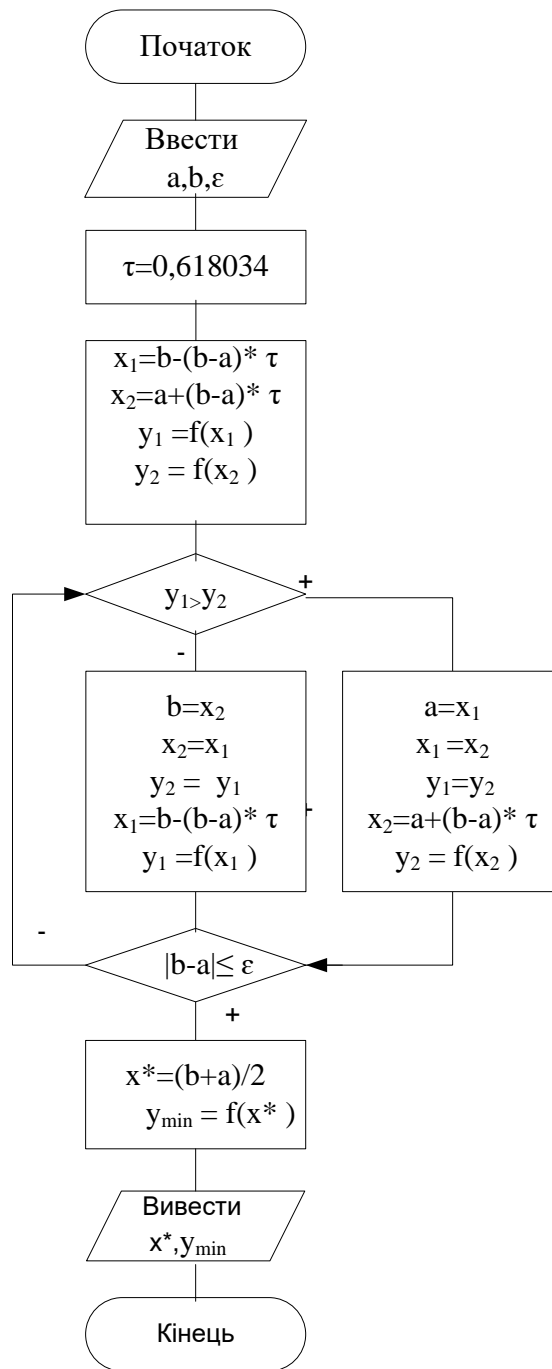


Рисунок 8.6. – Алгоритм методу золотого перерізу

Завдання та порядок виконання

- 1 Вивчити теоретичний матеріал [1, 4].
- 2 За алгоритмом визначення екстремуму цільової функції методом Монте-Карло скласти програму мовою C++.
- 3 Вибрати варіант вхідних даних. Номер варіанта відповідає номеру прізвища студента за списком у журналі викладача.
- 4 Відладити та виконати програму на комп'ютері.

СПИСОК ЛІТЕРАТУРИ

1 Математичні методи та моделі в розрахунках на ЕОМ : навч. посіб. / М. І. Данько, В. С. Меркулов, В. О. Гончаров та ін.; за заг. ред. М. І. Данька. Харків : УкрДАЗТ, 2008. 172 с.

2 Методичні вказівки до лабораторних робіт з дисциплін «Математичне моделювання на ЕОМ», «Математичні методи та моделі в розрахунках на ЕОМ» / В. С. Меркулов, В. Г. Пчолін, О. В. Казанко. Харків : УкрДАЗТ, 2010. 102 с.

3 Основи програмування мовою С++. Інтегроване середовище Visual С++. Ч. 2. Методичні вказівки до лабораторних робіт з дисциплін «Програмування», «Алгоритмізація та програмування» та «Інформатика» / В. М. Бутенко, С. Є. Бантюков, О. В. Головка, О. В. Чаленко, С. О. Бантюкова. Харків : УкрДУЗТ, 2018. 58 с.

4 Болотов О. Б. Числові методи і моделювання на ЕОМ : конспект лекцій. Харків : УкрДАЗТ, 2014. 65 с.

5 Болотов О. Б. Математичні методи й моделі в розрахунках на ЕОМ : конспект лекцій. Харків : УкрДАЗТ, 2006. 46 с.

6 Методичний посібник до лабораторних робіт по 1-му модулю дисципліни «Математичні методи та моделі в розрахунках на ЕОМ». Харків : УкрДАЗТ, 2006. 46 с.

МЕТОДИЧНІ ВКАЗІВКИ
до лабораторних робіт
з дисципліни
«ЧИСЛОВІ МЕТОДИ І МОДЕЛЮВАННЯ НА ЕОМ»

Відповідальний за випуск Головка О. В.

Редактор Еткало О. О.

Підписано до друку 07.07.20 р.

Формат паперу 60x84 1/16. Папір писальний.

Умовн.-друк.арк. 3,0. Тираж 5. Замовлення №

Видавець та виготовлювач Український державний університет
залізничного транспорту,
61050, Харків-50, майдан Фейербаха, 7.
Свідоцтво суб'єкта видавничої справи ДК № 6100 від 21.03.2018 р