

Математичне програмування

Конспект лекцій для студентів економічних спеціальностей денної та
заочної форм навчання

Кафедра вищої математики.

Укладач О.В. Островська

Зміст

ЕЛЕМЕНТИ ЛІНІЙНОЇ АЛГЕБРИ.....	3
Лекція №1. Визначники. Матриці.....	3
Лекція №2. Системи лінійних алгебраїчних рівнянь.....	12
Лекція № 3. Лінійні простори.....	27
МАТЕМАТИЧНЕ ПРОГРАМУВАННЯ.....	36
Лекція № 4. Постановка задач математичного програмування.....	36
Лекція № 5. Симплекс-метод. Метод штучного базису.....	51
Лекція № 6. Двоїстість у лінійному програмуванні.....	65
Лекція № 7. Транспортна задача.....	75
Лекція № 8. Дискретне програмування.....	88
Лекція № 9. Нелінійне програмування.....	104
Лекція № 10. Динамічне програмування.....	120
Література.....	128

ЕЛЕМЕНТИ ЛІНІЙНОЇ АЛГЕБРИ

Лекція №1. Визначники. Матриці.

Визначники та їхні властивості. Розклад Лапласа.

Матриці та дії над ними.

Обернена матриця.

Ранг матриці.

1.1. Визначники

Вирази $\Delta = \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} = a_{11}a_{22} - a_{12}a_{21}$ (1.1)

і $\Delta = \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} = a_{11}a_{22}a_{33} + a_{12}a_{23}a_{31} + a_{13}a_{21}a_{32} - a_{13}a_{22}a_{31} -$
 $- a_{11}a_{23}a_{32} - a_{12}a_{21}a_{33}$ (1.2)

називаються *визначниками* другого і третього порядків, відповідно.

Символами a_{ij} позначаються елементи визначника, причому індекс i показує номер рядка, а індекс j – номер стовпця, на перетині яких стоїть даний елемент. Елементами визначника можуть бути числа, функції тощо.

Схеми обчислення визначників (1.1) і (1.2) безпосередньо впливають з їх означення. Для обчислення визначника третього порядку зручно користуватися правилом Сарюса: дописавши за визначником його перші два стовпці, дістаємо додатні і від'ємні члени виразу (1.2), відповідно, із схем:

$$\begin{array}{ccccc} a_{11} & a_{12} & a_{13} & a_{11} & a_{12} \\ a_{21} & a_{22} & a_{23} & a_{21} & a_{22} \\ a_{31} & a_{32} & a_{33} & a_{31} & a_{32} \end{array}$$

$$\begin{array}{ccccc} a_{11} & a_{12} & a_{13} & a_{11} & a_{12} \\ a_{21} & a_{22} & a_{23} & a_{21} & a_{22} \\ a_{31} & a_{32} & a_{33} & a_{31} & a_{32} \end{array}$$

та

Приклад 1.1. $\begin{vmatrix} 3 & -5 & 3 \\ 2 & -4 & 2 \\ -1 & 2 & -2 \end{vmatrix} = 3 \cdot (-4) \cdot (-2) + (-5) \cdot 2 \cdot (-1) + 3 \cdot 2 \cdot 2 -$
 $- 3 \cdot (-4) \cdot (-1) - 3 \cdot 2 \cdot 2 - (-5) \cdot 2 \cdot (-2) = 2 \bullet$

Мінором M_{ij} елемента a_{ij} визначника Δ називається визначник, який утворюється з даного шляхом викреслення i -го рядка та j -го стовпця.

Алгебраїчним доповненням A_{ij} елемента a_{ij} називається добуток: $A_{ij} = (-1)^{i+j} M_{ij}$.

Приклад 1.2. Для елемента $a_{21} = -1$ визначника $\begin{vmatrix} 5 & 1 & -2 \\ -1 & 2 & 3 \\ 0 & 4 & -3 \end{vmatrix}$ маємо:

$$M_{21} = \begin{vmatrix} 1 & -2 \\ 4 & -3 \end{vmatrix} = 5; \quad A_{21} = (-1)^{2+1} \begin{vmatrix} 1 & -2 \\ 4 & -3 \end{vmatrix} = -5 \bullet$$

Безпосередньо доводиться формула

$$\Delta = \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} = a_{11}A_{11} + a_{12}A_{12} + a_{13}A_{13} = \\ = a_{1j}A_{1j} + a_{2j}A_{2j} + a_{3j}A_{3j} \quad (i, j = \overline{1,3}).$$

Цей вираз, відомий як розклад Лапласа, використовується для обчислення визначника.

Приклад 1.3.

$$\begin{vmatrix} 1 & -2 & 0 \\ 3 & -8 & 2 \\ 5 & -1 & 4 \end{vmatrix} = 1 \cdot (-1)^{1+1} \begin{vmatrix} -8 & 2 \\ -1 & 4 \end{vmatrix} - 2 \cdot (-1)^{1+2} \begin{vmatrix} 3 & 2 \\ 5 & 4 \end{vmatrix} + 0 \cdot (-1)^{1+3} \begin{vmatrix} 3 & -8 \\ 5 & -1 \end{vmatrix} = -26$$

Тут розклали визначник за елементами першого рядка.

Той факт, що визначник дорівнює сумі добутків елементів будь-якого рядка або стовпця на їхні алгебраїчні доповнення, дає змогу індуктивно ввести означення визначника довільного порядку.

Вважаючи, що встановлено поняття визначника $(n - 1)$ -го порядку, вираз

$$\Delta = \begin{vmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{vmatrix} = \sum_{k=1}^n a_{ik} A_{ik} = \sum_{k=1}^n a_{kj} A_{kj}, \quad (i, j = \overline{1, n})$$

називається визначником n -го порядку.

Визначники мають такі властивості

1. Визначник не зміниться, якщо його рядки замінити відповідними стовпцями..
2. Якщо переставити місцями будь-які два рядки, то визначник змінює знак.
3. Якщо один із рядків складається тільки з нулів, то визначник дорівнює нулю.
4. Якщо визначник має два однакових рядки, то він дорівнює нулю.
5. Спільний множник усіх елементів одного рядка можна винести за знак визначника.
6. Якщо елементи двох рядків пропорційні, то визначник дорівнює нулю.
7. Якщо кожний елемент i -того рядка визначника є сумою двох доданків, то визначник дорівнює сумі двох визначників, у першого з яких i -й рядок складається з перших доданків, а у другого – з других, інші ж елементи всіх трьох визначників однакові.

8. Визначник не зміниться, якщо до елементів одного рядка додати відповідні елементи іншого рядка, помножені на одне і те саме число.

9. Сума добутків елементів будь-якого рядка визначника на алгебраїчні доповнення відповідних елементів іншого рядка дорівнює нулю.

Зазначимо, що всі властивості визначників доводяться безпосередньою перевіркою.

Приклад 1.4.

$$\begin{aligned}\Delta &= \begin{vmatrix} 1 & 0 & -1 & -2 \\ 2 & 1 & -1 & -2 \\ -1 & 1 & 3 & 1 \\ 1 & 2 & -1 & 0 \end{vmatrix} = \begin{vmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 1 & 2 \\ -1 & 1 & 2 & -1 \\ 1 & 2 & 0 & 2 \end{vmatrix} = 1 \cdot (-1)^{1+1} \begin{vmatrix} 1 & 1 & 2 \\ 1 & 2 & -1 \\ 2 & 0 & 2 \end{vmatrix} = \\ &= 1 \cdot (-1)^{1+2} \begin{vmatrix} 1 & -1 \\ 2 & 2 \end{vmatrix} + 2 \cdot (-1)^{2+2} \begin{vmatrix} 1 & 2 \\ 2 & 2 \end{vmatrix} = -(1 \cdot 2 - 2 \cdot (-1)) + \\ &+ 2 \cdot (1 \cdot 2 - 2 \cdot 2) = -8,\end{aligned}$$

де у першому рядку перетворили всі елементи, крім першого, на нулі, додаючи перший стовпець до третього, а до четвертого – перший, помножений на 2 та розклали визначник четвертого порядку за елементами першого рядка, а потім визначник третього порядку – за елементами другого стовпця. ●

1.2. Матриці, дії над ними

Поняття матриці, як і поняття визначника, виникло у зв'язку з вивченням систем лінійних рівнянь.

Числовою *матрицею* розміру $m \times n$ називається прямокутна таблиця чисел, складена з m рядків та n стовпців і записана у вигляді:

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{pmatrix},$$

або коротко $A_{m \times n} = (a_{ij}) \quad (i = \overline{1, m}, j = \overline{1, n})$, де a_{ij} – елементи матриці, причому індекс i означає номер рядка, а j – номер стовпця, на перетині яких стоїть даний елемент.

Відмітимо деякі спеціальні типи матриць, а саме, якщо $m = n$, то матриця A називається *квадратною* порядку n , якщо ж $m \neq n$, то матриця A називається *прямокутною* розміру $m \times n$.

Зауважимо, що у квадратних матрицях як і у визначниках виділяють головну діагональ $(a_{11} a_{22} \dots a_{nn})$.

Матриця-рядок – прямокутна матриця розміру $1 \times n$ $A=(a_1 \ a_2...a_n)$,

матриця-стовпець – прямокутна матриця розміру $m \times 1$ $A=\begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \end{pmatrix}$. Нульова

матриця – матриця, у якій всі елементи дорівнюють нулю, вона позначається 0. Квадратна матриця називається діагональною, якщо всі її елементи, крім тих, що знаходяться на головній діагоналі, дорівнюють нулю. Діагональна матриця називається одиничною і позначається E , якщо кожний елемент її головної діагоналі дорівнює одиниці.

Однією з важливих характеристик квадратної матриці є її визначник, який позначається $\det A$ або Δ і утворюється з тих самих елементів (і у тому

ж порядку), що і матриця A . Отже, якщо $A=\begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix}$, то

$\det A = \begin{vmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{vmatrix}$. Очевидно, $\det E = 1$. Якщо $\det A \neq 0$, то матриця A

називається *невиродженою*, у противному разі вона називається *виродженою*.

Дві матриці $A=(a_{ij})$ і $B=(b_{ij})$ називаються рівними і записують $A=B$, якщо вони однакові за розмірами і мають рівні відповідні елементи, тобто $a_{ij}=b_{ij}$ при всіх i, j . Очевидно, якщо $A=B$, то $B=A$ і, якщо $A=B$ і $B=C$, то $A=C$.

Сумою $C=A+B$ двох матриць однакових розмірів $A=(a_{ij})$ і $B=(b_{ij})$ називається матриця тих же розмірів $C=(c_{ij})$, елементи якої визначаються рівностями $c_{ij}=a_{ij}+b_{ij}$.

Приклад 1.5. $A=\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix}$, $B=\begin{pmatrix} -1 & 0 & 2 \\ 1 & 0 & -2 \end{pmatrix}$, $C=A+B=\begin{pmatrix} 0 & 2 & 5 \\ 5 & 5 & 4 \end{pmatrix}$ ●

Добутком матриці $A=(a_{ij})$ на число α називається матриця $B=(b_{ij})$, елементи якої визначаються рівностями $b_{ij}=\alpha a_{ij}$, і записують $B=\alpha A$ або $B=A\alpha$.

Приклад 1.6. $3 \cdot \begin{pmatrix} -1 & 0 & 2 \\ 1 & 0 & -2 \end{pmatrix} = \begin{pmatrix} -1 & 0 & 2 \\ 1 & 0 & -2 \end{pmatrix} \cdot 3 = \begin{pmatrix} -3 & 0 & 6 \\ 3 & 0 & -6 \end{pmatrix}$ ●

Різниця матриць A і B визначається так $A-B=A+(-1) \cdot B$.

Матрицю $A=(a_{ij})$ можна помножити на матрицю $B=(b_{ij})$ лише за умови, що число стовпців матриці A дорівнює числу рядків матриці B . Такі матриці називаються *узгодженими*.

Добутком $C = A \cdot B$ матриці $A = A_{m \times n} = (a_{ij})$ на матрицю $B = B_{n \times k} = (b_{ij})$ називається матриця $C = C_{m \times k} = (c_{ij})$, у якій елемент c_{ij} дорівнює сумі добутків елементів i -того рядка матриці A на відповідні елементи j -того стовпця матриці B :

$$c_{ij} = a_{i1}b_{1j} + a_{i2}b_{2j} + \dots + a_{in}b_{nj} \quad (i = \overline{1, m}, j = \overline{1, k}).$$

Приклад 1.7. $A = \begin{pmatrix} 2 & 3 & 1 \\ -1 & 0 & 1 \end{pmatrix}, B = \begin{pmatrix} 2 & 1 & -1 \\ 1 & 3 & -2 \\ 0 & 2 & 1 \end{pmatrix},$

$$AB = \begin{pmatrix} 2 \cdot 2 + 3 \cdot 1 + 1 \cdot 0 & 2 \cdot 1 + 3 \cdot 3 + 1 \cdot 2 & 2 \cdot (-1) + 3 \cdot (-2) + 1 \cdot 1 \\ (-1) \cdot 2 + 0 \cdot 1 + 1 \cdot 0 & (-1) \cdot 1 + 0 \cdot 3 + 1 \cdot 2 & (-1) \cdot (-1) + 0 \cdot (-2) + 1 \cdot 1 \end{pmatrix} =$$

$$= \begin{pmatrix} 7 & 13 & -7 \\ -2 & 1 & 2 \end{pmatrix} \bullet$$

Слід зауважити, що операція множення матриць не комутативна, тобто у загальному випадку $AB \neq BA$. Не важко переконатися, наприклад, що $\begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} \neq \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$.

У разі коли $AB = BA$ матриці A і B називаються переставними.

Матриця $A^T = (a_{ij}^*)$ називається *транспонованою* до матриці $A = (a_{ij})$, якщо $a_{ij}^* = a_{ji}$, тобто транспонована матриця A^T утворюється із матриці A заміною її рядків стовпцями із збереженням їх номерів.

Приклад 1.8. Якщо $A = \begin{pmatrix} 1 & 2 \\ 3 & 1 \\ 2 & 4 \end{pmatrix}$, то $A^T = \begin{pmatrix} 1 & 3 & 2 \\ 2 & 1 & 4 \end{pmatrix} \bullet$

Квадратна матриця A називається *симетричною*, якщо $A = A^T$.

За умови, що введені операції мають зміст, виконуються такі їх властивості:

- 1) $A + B = B + A$; 2) $A + (B + C) = (A + B) + C$; 3) $A + 0 = A$;
- 4) $\alpha(\beta A) = \alpha\beta \cdot A$; 5) $\alpha(A + B) = \alpha A + \alpha B$;
- 6) $(\alpha + \beta)A = \alpha A + \beta A$; 7) $A(BC) = (AB)C$;
- 8) $(\alpha A)B = A(\alpha B) = \alpha(AB)$;
- 9) $(A + B)C = AC + BC, C(A + B) = CA + CB$;
- 10) $A \cdot 0 = 0 \cdot A = 0$; 11) $AE = EA = A$;
- 12) $\det(AB) = \det A \cdot \det B$; 13) $(A + B)^T = A^T + B^T$;
- 14) $(\alpha A)^T = \alpha \cdot A^T$; 15) $(AB)^T = B^T A^T$.

1.3. Обернена матриця.

Нехай $A_{n \times n}$ — квадратна матриця. Матриця A^{-1} називається *оберненою* до матриці A , якщо $A A^{-1} = A^{-1} A = E$.

Теорема 1.1. Для існування оберненої матриці A^{-1} необхідно і достатньо, щоб матриця A була невинродженою.

Доведення. Необхідність безпосередньо випливає із співвідношень $\det(AA^{-1}) = \det E$ або $\det A \cdot \det A^{-1} = 1$.

Доведемо достатність умов теореми. Нехай матриця

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix}$$

невироджена, тобто $\det A \neq 0$, тоді існує обернена матриця

$$A^{-1} = \frac{1}{\det A} \begin{pmatrix} A_{11} & A_{21} & \dots & A_{n1} \\ A_{12} & A_{22} & \dots & A_{n2} \\ \dots & \dots & \dots & \dots \\ A_{1n} & A_{2n} & \dots & A_{nn} \end{pmatrix}, \quad (1.3)$$

де A_{ij} – алгебраїчні доповнення елементів a_{ij} визначника матриці A .

Дійсно, добутки $A A^{-1}$ і $A^{-1} A$ дорівнюють матриці, у якій елементи головної діагоналі дорівнюють одиниці, а всі недіагональні елементи – нулю відповідно до властивості 9 визначників і формули Лапласа, тобто $AA^{-1} = A^{-1}A = E$. Неважко також показати, що матриця A має єдину обернену матрицю. Дійсно, якщо A_1^{-1} , A_2^{-1} деякі обернені матриці до матриці A , то

$$A_1^{-1} = A_1^{-1}E = A_1^{-1}(A \cdot A_2^{-1}) = (A_1^{-1}A)A_2^{-1} = E \cdot A_2^{-1} = A_2^{-1} \bullet$$

Доведення теореми дає конструктивний спосіб побудови оберненої матриці. Для цього досить скористатися формулою (1.4).

Приклад 1.9. Знайти обернену матрицю до матриці $A = \begin{pmatrix} 2 & -1 \\ 3 & 1 \end{pmatrix}$.

Оскільки $\det A = \begin{vmatrix} 2 & -1 \\ 3 & 1 \end{vmatrix} = 5 \neq 0$, $A_{11} = (-1)^{1+1} \cdot 1 = 1$, $A_{12} = (-1)^{1+2} \cdot 3 = -3$;
 $A_{21} = (-1)^{2+1} \cdot (-1) = 1$, $A_{22} = (-1)^{2+2} \cdot 2 = 2$, то

$$A^{-1} = \frac{1}{5} \begin{pmatrix} 1 & 1 \\ -3 & 2 \end{pmatrix} = \begin{pmatrix} \frac{1}{5} & \frac{1}{5} \\ -\frac{3}{5} & \frac{2}{5} \end{pmatrix} \bullet$$

З означення оберненої матриці випливає, що $(A^{-1})^{-1} = A$, причому $\det A^{-1} = \frac{1}{\det A}$, тобто A і A^{-1} – взаємно обернені матриці. Слід зауважити, що $(AB)^{-1} = B^{-1}A^{-1}$, тобто $B^{-1}A^{-1}$ – обернена до матриці AB .

1.4. Ранг матриці.

Нехай задано матрицю $A_{m \times n}$, виділимо в ній довільних k рядків і k стовпців ($k \leq \min(m, n)$). Визначник складений з елементів, що стоять на перетині виділених рядків і стовпців, називається *мінором* k -го порядку матриці $A_{m \times n}$.

Рангом $r = r(A_{m \times n})$ матриці $A_{m \times n}$ називається найбільший з порядків її мінорів, відмінних від нуля. Сам мінор при цьому називається базисним. Очевидно, що ранг існує для будь-якої матриці $A_{m \times n}$, причому $0 \leq r \leq \min(m, n)$.

Знаходження рангу матриці безпосередньо за означенням пов'язано з певними незручностями, тому що потрібно обчислити значну кількість визначників. Значно простіше цього можна досягти, виконавши над матрицею елементарні перетворення, що не змінюють її ранг:

- а) переставити місцями два рядки (стовпці);
 - б) помножити всі елементи рядка (стовпця) на один і той самий ненульовий множник;
 - в) додати до елементів рядка (стовпця) відповідні елементи іншого рядка (стовпця).
 - г) відкинути рядок (стовпець), який містить тільки нульові елементи.
- Такі перетворення зводять матрицю A до матриці ступінчатого виду:

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & \cdots & \cdots & a_{1n} \\ 0 & \cdots & 0 & a_{2j_2} & \cdots & a_{2n} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & 0 & a_{rj_r} & \cdots & a_r \\ 0 & \cdots & 0 & \cdots & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & 0 & \cdots & \cdots & 0 \end{pmatrix}_n, \text{ де } a_{11}, a_{2j_2}, \dots, a_{rj_r} \neq 0,$$

яка містить тільки r ненульових рядків. Тоді всі її мінори порядку $k > r$ мають хоча б один нульовий рядок і дорівнюють нулю, а її мінор порядку r :

$$\begin{vmatrix} a_{11} & a_{1j_2} & \cdots & a_{1j_r} \\ 0 & a_{2j_2} & \cdots & a_{2j_r} \\ \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & 0 & a_{rj_r} \end{vmatrix} = a_{11} \cdot a_{2j_2} \cdots a_{rj_r} \neq 0.$$

Звідси ранг ступінчатої, а отже, і самої матриці A дорівнює r .

Приклад 1.10. Знайти ранг матриці $A = \begin{pmatrix} 1 & 3 & 0 \\ 2 & 1 & -1 \\ 3 & 4 & -1 \end{pmatrix}$.

Нехай знак $\sim$ означає, що матрицю здобуто з попередньої за допомогою елементарних перетворень. Тоді до другого і третього рядків додамо перший, помножений, відповідно, на -2 і -3 , а далі від третього рядка віднімемо другий і матимемо:

$$\begin{pmatrix} 1 & 3 & 0 \\ 2 & 1 & -1 \\ 3 & 4 & -1 \end{pmatrix} \sim \begin{pmatrix} 1 & 3 & 0 \\ 0 & -5 & -1 \\ 0 & -5 & -1 \end{pmatrix} \sim \begin{pmatrix} 1 & 3 & 0 \\ 0 & -5 & -1 \\ 0 & 0 & 0 \end{pmatrix}, \quad \begin{vmatrix} 1 & 3 \\ 0 & -5 \end{vmatrix} = -5 \neq 0.$$

Отже, $r(A) = 2$ ●

Завдання для самоконтролю

1. Що називається визначником другого порядку, третього порядку?
2. Що називається мінором, алгебраїчним доповненням?
3. Сформулюйте основні властивості визначників.
4. Як обчислюються визначники вищих порядків?

5. Що називається матрицею?
 6. Як визначається сума і різниця двох матриць, добуток матриці на число та добуток двох матриць?

7. Що називається оберненою матрицею? Як її знайти?
 8. Що називається рангом матриці? Як його знайти?
 9. Які перетворення над матрицею називаються елементарними?

10. Обчислити визначник $\begin{vmatrix} 3 & 4 & -5 \\ 8 & 7 & -2 \\ 2 & -1 & 8 \end{vmatrix}$. (Відповідь: 0).

11. Обчислити визначник $\begin{vmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{vmatrix}$. (Відповідь: -3).

12. Знайти матрицю $A = B^T - B$, якщо $B = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 2 & 1 \\ -1 & 0 & 3 \end{pmatrix}$.

(Відповідь: $A = \begin{pmatrix} 0 & 0 & -1 \\ 0 & 0 & -1 \\ 1 & 1 & 0 \end{pmatrix}$).

13. Знайти добуток $\begin{pmatrix} 1 & 0 & -1 \\ 3 & 2 & 1 \\ 0 & 1 & 2 \\ 2 & 1 & 3 \\ 1 & 2 & 0 \end{pmatrix} \begin{pmatrix} 1 & -1 \\ 2 & -2 \\ 3 & -3 \end{pmatrix}$. (Відповідь: $\begin{pmatrix} -2 & 2 \\ 10 & -10 \\ 8 & -8 \\ 13 & -13 \\ 5 & -5 \end{pmatrix}$).

14. Піднести до степеня $\begin{pmatrix} 1 & 1 & -1 \\ 3 & -1 & 2 \\ 2 & -1 & 0 \end{pmatrix}^3$. (Відповідь: $\begin{pmatrix} 7 & 0 & 0 \\ 0 & 7 & 0 \\ 0 & 0 & 7 \end{pmatrix}$).

15. Знайти обернену матрицю до матриці $A = \begin{pmatrix} 1 & -1 & 3 \\ 0 & 1 & -1 \\ 2 & 2 & -2 \end{pmatrix}$.

(Відповідь: $A^{-1} = \begin{pmatrix} 0 & -1 & \frac{1}{2} \\ \frac{1}{2} & 2 & -\frac{1}{4} \\ \frac{1}{2} & 1 & -\frac{1}{4} \end{pmatrix}$).

16. Знайти ранг матриці $A = \begin{pmatrix} 1 & 2 & -1 & 3 \\ 3 & 6 & -3 & -1 \\ 3 & 6 & -3 & 10 \end{pmatrix}$. (Відповідь: $r(A) = 2$).

Лекція №2. Системи лінійних алгебраїчних рівнянь

Системи лінійних алгебраїчних рівнянь. Формули Крамера.

Матричний спосіб розв'язання системи лінійних рівнянь.

Теорема Кронекера-Капеллі.

Метод послідовного виключення невідомих.

Метод Жордана-Гаусса.

Базисний, фундаментальний та опорний розв'язки системи лінійних рівнянь.

2.1 Формули Крамера.

Однією з основних задач лінійної алгебри є розв'язування систем лінійних алгебраїчних рівнянь.

Системою m лінійних рівнянь з n невідомими $x_1, x_2, \dots, x_n$ називається сукупність рівнянь

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2 \\ \text{-----} \\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n = b_m \end{cases} \quad (2.1)$$

Числа a_{ij} називаються *коефіцієнтами*, а числа b_i – *вільними членами* системи (2.1), де $(i = \overline{1, m}, j = \overline{1, n})$.

Упорядкований набір n чисел $(\alpha_1, \alpha_2, \dots, \alpha_n)$ називається *розв'язком системи* (2.1), якщо при підстановці цих чисел замість невідомих $x_1, x_2, \dots, x_n$ кожне рівняння системи перетворюється в тотожність.

Система рівнянь (2.1) називається *однорідною*, якщо $b_1 = b_2 = \dots = b_m = 0$, і *неоднорідною* у протилежному разі. Зауважимо, що однорідна система завжди має розв'язок, а саме $x_1 = x_2 = \dots = x_n = 0$, який називається *тривіальним*.

Система (2.1) називається *сумісною*, якщо вона має хоча б один розв'язок, і *несумісною*, якщо вона не має жодного розв'язку. Сумісна система називається *визначеною*, якщо вона має єдиний розв'язок, і *невизначеною*, якщо вона має більше, ніж один розв'язок. Дві системи лінійних рівнянь, що мають однакові розв'язки, називаються *еквівалентними*.

Розглянемо випадок, коли число невідомих дорівнює числу рівнянь

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2 \\ \text{-----} \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = b_n \end{cases} \quad (2.2)$$

Утворимо визначник системи (2.2) складений із коефіцієнтів при невідомих:

$$\Delta = \begin{vmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ - & - & - & - \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{vmatrix}$$

Помноживши перше рівняння системи на алгебраїчне доповнення A_{11} , друге - на A_{21} і т.д., останнє на A_{n1} і додавши їх усі, дістанемо $x_1(a_{11}A_{11}+a_{21}A_{21}+\dots+a_{n1}A_{n1})+x_2(a_{12}A_{11}+a_{22}A_{21}+\dots+a_{n2}A_{n1})+\dots+x_n(a_{1n}A_{11}+a_{2n}A_{21}+\dots+a_{nn}A_{n1})=b_1A_{11}+b_2A_{21}+\dots+b_nA_{n1}$, або $x_1 \cdot \Delta = b_1A_{11}+b_2A_{21}+\dots+b_nA_{n1}$, оскільки взяті у дужки коефіцієнти при невідомих $x_2, x_3, \dots, x_n$ дорівнюють нулю за властивістю 9 визначників, а коефіцієнт при x_1 дорівнює Δ за формулою Лапласа. Позначивши $b_1A_{11}+b_2A_{21}+\dots+b_nA_{n1}=\Delta_1$, де Δ_1 – визначник, який утворюється з визначника Δ заміною першого його стовпця, тобто стовпця коефіцієнтів при невідомому x_1 на стовпець вільних членів, матимемо

$$x_1 \cdot \Delta = \Delta_1 \quad (2.3)$$

Аналогічно отримаємо рівності:

$$x_2 \cdot \Delta = \Delta_2, \dots, x_n \cdot \Delta = \Delta_n, \quad (2.4)$$

де Δ_i ($i = \overline{2, n}$) є визначник, що утворюється з Δ заміною його i -го стовпця стовпцем вільних членів.

Тут можливі такі випадки:

1) $\Delta \neq 0$, тоді система (2.2) визначена і має єдиний розв'язок за формулами Крамера:

$$x_1 = \frac{\Delta_1}{\Delta}, x_2 = \frac{\Delta_2}{\Delta}, \dots, x_n = \frac{\Delta_n}{\Delta}; \quad (2.5)$$

2) $\Delta = \Delta_1 = \Delta_2 = \dots = \Delta_n = 0$, тоді система (3.2) або невизначена, або несумісна;

3) $\Delta = 0$ і хоча б один з визначників Δ_i відмінний від нуля, тоді система (2.2) несумісна.

Зауваження. Щодо випадків 2) і 3), то при $\Delta = 0$ і $b_1 = b_2 = \dots = b_n = 0$ система (2.2) є невизначеною, а при $n=2$ вона також є невизначеною, але за умови $\Delta = \Delta_1 = \Delta_2 = 0$.

Слід зазначити, що формули Крамера не є ефективними при $n \geq 4$, оскільки обчислення визначників є досить кропіткою роботою.

Приклад 2.1 Розв'язати систему
$$\begin{cases} x_1 + 2x_2 = 3 \\ -x_1 + x_2 + 2x_3 = 5 \\ 3x_1 + x_3 = -2 \end{cases}$$

Оскільки $\Delta = \begin{vmatrix} 1 & 2 & 0 \\ -1 & 1 & 2 \\ 3 & 0 & 1 \end{vmatrix} = \begin{vmatrix} 1 & 2 \\ 0 & 1 \end{vmatrix} - 2 \begin{vmatrix} -1 & 2 \\ 3 & 1 \end{vmatrix} = 1 + 14 = 15 \neq 0$, то задана система визначена. Знайдемо

$$\Delta_1 = \begin{vmatrix} 3 & 2 & 0 \\ 5 & 1 & 2 \\ -2 & 0 & 1 \end{vmatrix} = 3 \begin{vmatrix} 1 & 2 \\ 0 & 1 \end{vmatrix} - 2 \begin{vmatrix} 5 & 2 \\ -2 & 1 \end{vmatrix} = 3 - 18 = -15,$$

$$\Delta_2 = \begin{vmatrix} 1 & 3 & 0 \\ -1 & 5 & 2 \\ 3 & -2 & 1 \end{vmatrix} = \begin{vmatrix} 5 & 2 \\ -2 & 1 \end{vmatrix} - 3 \begin{vmatrix} -1 & 2 \\ 3 & 1 \end{vmatrix} = 9 + 21 = 30,$$

$$\Delta_3 = \begin{vmatrix} 1 & 2 & 3 \\ -1 & 1 & 5 \\ 3 & 0 & -2 \end{vmatrix} = 3 \begin{vmatrix} 2 & 3 \\ 1 & 5 \end{vmatrix} - 2 \begin{vmatrix} 1 & 2 \\ -1 & 1 \end{vmatrix} = 21 - 6 = 15,$$

і за формулами Крамера (3.5) матимемо розв'язок системи $x_1 = -\frac{15}{15} = -1$,
 $x_2 = \frac{30}{15} = 2$, $x_3 = \frac{15}{15} = 1$ ●

2.2. Матричний спосіб

Нехай задано систему (2.2). Матрицю $A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix}$, складену з

коефіцієнтів системи (2.2), називають *матрицею системи*.

Якщо $X = \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{pmatrix}$ і $B = \begin{pmatrix} b_1 \\ b_2 \\ \dots \\ b_n \end{pmatrix}$, то задану систему можна записати у вигляді

матричного рівняння з невідомою матрицею X :

$$AX=B.$$

(2.6)

У разі коли матриця A системи (2.2) має обернену матрицю A^{-1} , маємо $A^{-1}AX=A^{-1}B$ або $X=A^{-1}B$ (2.7)

оскільки $A^{-1}A=E$ і $EX=X$. Формула (2.7) – матричний запис розв'язку системи (2.2), або розв'язок матричного рівняння (2.6).

Приклад 2.2 Розв'язати систему $\begin{cases} x_1 - x_3 = 2 \\ x_1 + x_2 + x_3 = 6 \\ x_2 + 3x_3 = 5 \end{cases}$

Матриця системи $A = \begin{pmatrix} 1 & 0 & -1 \\ 1 & 1 & 1 \\ 0 & 1 & 3 \end{pmatrix}$ не вироджена, оскільки

$$\det A = \begin{vmatrix} 1 & 0 & -1 \\ 1 & 1 & 1 \\ 0 & 1 & 3 \end{vmatrix} = 1 \begin{vmatrix} 1 & 1 \\ 1 & 3 \end{vmatrix} - 1 \begin{vmatrix} 1 & 1 \\ 0 & 1 \end{vmatrix} = 2 - 1 = 1 \neq 0. \text{ Тоді, знайшовши}$$

$$A_{11} = (-1)^{1+1} \begin{vmatrix} 1 & 1 \\ 1 & 3 \end{vmatrix} = 2; \quad A_{12} = (-1)^{1+2} \begin{vmatrix} 1 & 1 \\ 0 & 3 \end{vmatrix} = -3; \quad A_{13} = (-1)^{1+3} \begin{vmatrix} 1 & 1 \\ 0 & 1 \end{vmatrix} = 1;$$

$$A_{21} = (-1)^{2+1} \begin{vmatrix} 0 & -1 \\ 1 & 3 \end{vmatrix} = -1; A_{22} = (-1)^{2+2} \begin{vmatrix} 1 & -1 \\ 0 & 3 \end{vmatrix} = 3; A_{23} = (-1)^{2+3} \begin{vmatrix} 1 & 0 \\ 0 & 1 \end{vmatrix} = -1;$$

$$A_{31} = (-1)^{3+1} \begin{vmatrix} 0 & -1 \\ 1 & 1 \end{vmatrix} = 1; A_{32} = (-1)^{3+2} \begin{vmatrix} 1 & -1 \\ 1 & 1 \end{vmatrix} = -2; A_{33} = (-1)^{3+3} \begin{vmatrix} 1 & 0 \\ 1 & 1 \end{vmatrix} = 1,$$

дістанемо обернену матрицю $A^{-1} = \frac{1}{1} \begin{pmatrix} 2 & -1 & 1 \\ -3 & 3 & -2 \\ 1 & -1 & 1 \end{pmatrix} = \begin{pmatrix} 2 & -1 & 1 \\ -3 & 3 & -2 \\ 1 & -1 & 1 \end{pmatrix}$. Отже,

розв'язком системи за формулою (2.7) буде $X = \begin{pmatrix} 2 & -1 & 1 \\ -3 & 3 & -2 \\ 1 & -1 & 1 \end{pmatrix} \cdot \begin{pmatrix} 2 \\ 6 \\ 5 \end{pmatrix} = \begin{pmatrix} 3 \\ 2 \\ 1 \end{pmatrix}$, де

$$X = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}, B = \begin{pmatrix} 2 \\ 6 \\ 5 \end{pmatrix}, \text{ або } x_1 = 3, x_2 = 2, x_3 = 1$$

2.3. Критерій сумісності системи лінійних рівнянь

Нехай задано систему (2.1). Складемо матрицю $A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{pmatrix}$ і

розширену матрицю $\tilde{A} = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} & b_1 \\ a_{21} & a_{22} & \dots & a_{2n} & b_2 \\ \dots & \dots & \dots & \dots & \dots \\ a_{m1} & a_{m2} & \dots & a_{mn} & b_m \end{pmatrix}$ заданої системи.

При дослідженні таких систем важливе значення має наступна **теорема Кронекера - Капеллі**.

Теорема 2.1. Для того, щоб система лінійних рівнянь була сумісною, необхідно і достатньо, щоб ранг матриці системи дорівнював рангу розширеної матриці цієї системи $r(A) = r(\tilde{A}) = r$; r називають *рангом системи*, причому якщо ранг системи дорівнює числу невідомих $r = n$, то система визначена, якщо ж буде менший від числа невідомих $r < n$, то система – невизначена.

Приклад 2.3. Дослідити систему

$$\begin{cases} x_1 + 2x_2 + 3x_3 - x_4 = 0 \\ x_1 - x_2 + x_3 + 2x_4 = 4 \\ x_1 + 5x_2 + 5x_3 - 4x_4 = -4 \\ x_1 + 8x_2 + 7x_3 - 7x_4 = 6 \end{cases}$$

Розв'язання. Віднімаючи перший рядок матриці A від усіх інших її рядків і після цього додаючи другий рядок спочатку до третього, а потім до четвертого рядка, помноженого на 0,5, дістанемо:

$$A = \begin{pmatrix} 1 & 2 & 3 & -1 \\ 1 & -1 & 1 & 2 \\ 1 & 5 & 5 & -4 \\ 1 & 8 & 7 & -7 \end{pmatrix} \sim \begin{pmatrix} 1 & 2 & 3 & -1 \\ 0 & -3 & -2 & 3 \\ 0 & 3 & 2 & -3 \\ 0 & 6 & 4 & -6 \end{pmatrix} \sim \begin{pmatrix} 1 & 2 & 3 & -1 \\ 0 & -3 & -2 & 3 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix},$$

$$\begin{vmatrix} 1 & 2 \\ 0 & -3 \end{vmatrix} = -3 \neq 0, \quad r(A) = 2.$$

Аналогічно, віднімаючи перший рядок матриці $\tilde{A}$ від усіх її інших рядків, а потім додаючи другий рядок до третього рядка і до четвертого помноженого на 0,5 та після цього переставивши два останні рядки, матимемо:

$$\tilde{A} = \begin{pmatrix} 1 & 2 & 3 & -1 & 0 \\ 1 & -1 & 1 & 2 & 4 \\ 1 & 5 & 5 & -4 & -4 \\ 1 & 8 & 7 & -7 & 6 \end{pmatrix} \sim \begin{pmatrix} 1 & 2 & 3 & -1 & 0 \\ 0 & -3 & -2 & 3 & 4 \\ 0 & 3 & 2 & -3 & -4 \\ 0 & 6 & 4 & -6 & 6 \end{pmatrix} \sim$$

$$\sim \begin{pmatrix} 1 & 2 & 3 & -1 & 0 \\ 0 & -3 & -2 & 3 & 4 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 7 \end{pmatrix} \sim \begin{pmatrix} 1 & 2 & 3 & -1 & 0 \\ 0 & -3 & -2 & 3 & 4 \\ 0 & 0 & 0 & 0 & 7 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix},$$

$$\begin{vmatrix} 3 & -1 & 0 \\ -2 & 3 & 4 \\ 0 & 0 & 7 \end{vmatrix} = 7 \begin{vmatrix} 3 & -1 \\ -2 & 3 \end{vmatrix} = 7(9 - 2) = 49 \neq 0, \quad r(B) = 3.$$

Таким чином, задана система несумісна, оскільки $r(A) \neq r(B)$ ●

2.4 Метод послідовного виключення

На практиці при розв'язанні систем лінійних рівнянь використовують значно раціональнішу обчислювальну схему, ніж формули Крамера або матричний спосіб, а саме - метод послідовного виключення невідомих (метод Гауса), який ґрунтується на еквівалентних перетвореннях рівнянь системи, тобто на тих самих елементарних перетвореннях, які виконувалися над рядками і стовпцями матриці при знаходженні її рангу.

Отже, застосовуючи елементарні перетворення до системи (2.1), побудуємо еквівалентну їй систему спеціального вигляду.

Змінимо порядок рівнянь в системі, якщо це потрібно, і виберемо першим те рівняння, в якому коефіцієнт при x_1 не дорівнює нулю. Тоді не порушуючи загальності, вважатимемо, що $a_{11} \neq 0$. Помножимо перше рівняння на $-\frac{a_{i1}}{a_{11}}$, ($i, = \overline{1, n}$) і додамо його почленно до рівнянь з номерами ($i, = \overline{2, n}$).

Після такої операції у рівняннях з номерами $i > 1$ буде виключено невідоме x_1 . При цьому може статись так, що разом з x_1 будуть виключені невідомі $x_2, x_3, \dots, x_{k-1}$, але знайдеться хоча б одне рівняння, до якого входить невідоме x_k . Одне з таких рівнянь візьмемо тепер за друге рівняння системи.

Далі так само скористаємося цим другим рівнянням системи

$$a'_{2k}x_k + a'_{2k+1}x_{k+1} + \dots + a'_{2n}x_n = b'_2$$

для виключення невідомого x_k з усіх рівнянь, що залишились, крім першого. Продовжуючи цей процес, після r кроків дістанемо ступінчасту систему еквівалентну системі (2.1):

$$(2.8) \quad \begin{cases} a_{11}^*x_1 + \dots + a_{1n}^*x_n = b_1^* \\ a_{2k}^*x_k + \dots + a_{2n}^*x_n = b_2^* \\ a_{3p}^*x_p + \dots + a_{3n}^*x_n = b_3^* \\ \dots \\ a_{rs}^*x_s + \dots + a_{rn}^*x_n = b_r^* \\ 0 = b_{r+1}^* \\ \dots \\ 0 = b_m^* \end{cases}$$

де $a_{11}^* = a_{11} \neq 0$, $a_{2k}^* \neq 0$, $a_{3p}^* \neq 0$, ..., $a_{rs}^* \neq 0$; $r \leq m$.

Підводячи підсумок, зазначимо:

1) якщо система (2.8) містить хоча б одне рівняння виду $0 = b_i^*$, де $b_i^* \neq 0$ ($i = \overline{r+1, n}$), то вона, а отже, і еквівалентна їй система (2.1) несумісній;

2) якщо ж система (2.8) не має рівнянь виду $0 = b_i^*$ ($b_i^* \neq 0, i = \overline{r+1, n}$) і $r < n$, то взявши невідомі $x_1, x_k, x_p, \dots, x_s$ за основні, а решту – за вільні та знайшовши з r -го рівняння x_s і підставивши його в перші $r - 1$ рівнянь, "піднімаючись по системі вгору", знайдемо всі інші основні невідомі, які, очевидно, виражаються через вільні невідомі, при цьому еквівалентні системи (2.1) і (2.8) мають безліч розв'язків;

3) у разі, коли $r = n$, система (3.1) зводиться до *трикутного* виду:

$$(2.9) \quad \begin{cases} a_{11}^*x_1 + \dots + a_{1n}^*x_n = b_1^* \\ a_{22}^*x_2 + \dots + a_{2n}^*x_n = b_2^* \\ \dots \\ a_{nn}^*x_n = b_n^* \end{cases}$$

в якій всі невідомі - основні і знайшовши x_n з останнього рівняння та знову "піднімаючись по системі вгору", знайдемо всі інші невідомі, причому в цьому випадку система (2.1) має єдиний розв'язок.

Слід зауважити, що застосовуючи метод Гауса, зручніше приводити до ступінчатого вигляду не систему рівнянь, а лише її розширену матрицю.

Приклад 2.4 Розв'язати систему методом Гауса

$$\begin{cases} x - y + 2z = -1 \\ -x + 2y - 3z = 3 \\ 2x - y + 3z = 2 \end{cases}$$

Розв'язання. Виконавши елементарні перетворення над рядками розширеної матриці заданої системи і позначивши символом $\rightarrow$ перехід при цьому до відповідної матриці, дістанемо:

$$\left(\begin{array}{ccc|c} 1 & -1 & 2 & -1 \\ -1 & 2 & -3 & 3 \\ 2 & -1 & 3 & 2 \end{array} \right) \rightarrow \left(\begin{array}{ccc|c} 1 & -1 & 2 & -1 \\ 0 & 1 & -1 & 2 \\ 0 & 1 & -1 & 4 \end{array} \right) \rightarrow \left(\begin{array}{ccc|c} 1 & -1 & 2 & -1 \\ 0 & 1 & -1 & 2 \\ 0 & 0 & 0 & 2 \end{array} \right).$$

Таким чином, отримали еквівалентну систему:

$$\begin{cases} x - y + 2z = -1 \\ 0 \cdot x + y - z = 2 \\ 0 \cdot x + 0 \cdot y + 0 \cdot z = 2 \end{cases},$$

яка, очевидно, разом із заданою є несумісною ●

2.5. Метод Жордана-Гауса

Модифікований метод Жордана-Гауса, побудований на сукупності зручних обчислювальних алгоритмів, є більш досконалим, ніж метод Гауса і дозволяє безпосередньо відшукувати значення невідомих при розв'язуванні систем лінійних рівнянь, знаходити ранг матриці та обернену матрицю, обчислювати визначники і, більше того, він з певними доповненнями лежить в основі симплексного метода розв'язування задач лінійного програмування.

При розв'язуванні системи рівнянь (2.1) роблять певну кількість ітерацій *перетворення Жордана – Гауса*, переходячи від однієї розрахункової таблиці до іншої (за допомогою елементарних перетворень кожний раз система зводиться до еквівалентної). Наведемо алгоритм однієї ітерації перетворення Жордана - Гауса:

1. В розширеній матриці B системи, записаній у вигляді таблиці, виберемо відмінний від нуля елемент a_{qp} і назвемо його розв'язувальним.
2. Всі елементи розв'язувального q -го рядка ділимо на a_{qp} .
3. У розв'язувальному p -му стовпці замість a_{qp} пишемо одиницю, а замість інших його елементів - нулі.
4. Усі інші елементи (коефіцієнти та вільні члени) визначають за формулами:

$$a'_{ij} = a_{ij} - \frac{a_{ip} a_{qj}}{a_{qp}}, \quad b'_i = b_i - \frac{a_{ip} \cdot b_q}{a_{qp}}, \text{ якщо } i \neq q;$$

причому обчислення доцільно виконувати, користуючись схемою прямокутника, а саме для знаходження елемента a'_{ij} слід із елемента a_{ij} , що перетворюється, відняти добуток елементів a_{ip} і a_{qj} , розташованих у протилежних вершинах прямокутника, поділений на розв'язувальний

$$\text{елемент } a_{qp} : \begin{array}{ccccc} a_{ij} & \cdots & \cdots & \cdots & a_{ip} \\ \vdots & \cdots & \cdots & \cdots & \vdots \\ a_{qj} & \cdots & \cdots & \cdots & a_{qp} \end{array}$$

Дамо також рекомендації щодо спрощення розрахунків.

1. За розв'язувальний елемент доцільно обирати одиницю.
2. Якщо у розв'язувальному стовпці є нулі, то рядки, що відповідають їм, переписуються без змін.
3. Якщо у розв'язувальному рядку є нулі, то відповідні їм стовпці переписуються без змін.

4. Якщо є два пропорційних рядки, то один з них можна закреслити.

5. Наступна ітерація виконується так само, при цьому розв'язувальний елемент обирають з інших рядків і стовпців.

Після скінченного числа r ітерацій, де r – ранг системи, можливі такі випадки:

- 1) $r = n$ і система має єдиний розв'язок;
- 2) $r \leq m < n$ і система має безліч розв'язків та зводиться до системи у базисному вигляді.

$$\begin{cases} x_1 + \dots + b_{1r+1}x_{r+1} + \dots + b_{1n}x_n = c_1 \\ x_2 + \dots + b_{2r+1}x_{r+1} + \dots + b_{2n}x_n = c_2 \\ \dots \\ x_r + \dots + b_{rr+1}x_{r+1} + \dots + b_{rn}x_n = c_r \end{cases}, \quad (2.10)$$

де невідомі $x_1, \dots, x_r$ називаються *базисними*, а невідомі $x_{r+1}, x_{r+2}, \dots, x_n$ – *вільними*. Якщо у (2.10) покласти $x_{r+1} = x_{r+2} = \dots = x_n = 0$, то одержимо *базисний розв'язок* $x_1 = c_1, x_2 = c_2, \dots, x_r = c_r$.

Якщо ж одну вільну невідому прирівняти одиниці, а інші – нулю, то дістанемо, так званий, *фундаментальний розв'язок*. Невід'ємний базисний розв'язок називається *опорним*.

- 3) При перетворенні одержали рівняння, усі коефіцієнти якого – нулі, а права частина не дорівнює нулю, тоді система несумісна.

Приклад 2.5. Розв'язати методом Жордана-Гауса систему

$$\begin{cases} x_1 + x_2 + x_3 + x_4 + x_5 = 7 \\ 3x_1 + 2x_2 + x_3 + x_4 - 3x_5 = -2 \\ x_2 + 2x_3 + 2x_4 + 6x_5 = 23 \\ 5x_1 + 4x_2 + 3x_3 + 3x_4 - x_5 = 12 \end{cases}$$

Розв'язання. Запишемо коефіцієнти і вільні члени в таблицю

x_1	x_2	x_3	x_4	x_5	b_i
1	1	1	1	1	7
3	2	1	1	-3	-2
0	1	2	2	6	23
5	4	3	3	-1	12

За розв'язувальний елемент візьмемо коефіцієнт при x_1 в першому рівнянні. Перший рядок перепишемо без змін (він містить розв'язувальний елемент), а всі елементи першого стовпця, крім розв'язувального, замінимо нулями. До всіх інших елементів застосуємо схему прямокутника. Обчислимо, наприклад, елемент a'_{43} :

$$\begin{aligned} a_{11} &= 1, & a_{13} &= 1, \\ a_{41} &= 5, & a_{43} &= 3, \end{aligned}$$

$$a'_{43} = a_{43} - \frac{a_{13} \cdot a_{41}}{a_{11}} = 3 - \frac{1 \cdot 5}{1} = -2.$$

Зауважимо, що третій рядок переписується без змін (йому відповідає нуль в першому стовпці). Отже, після першої ітерації, дістанемо еквівалентну систему, коефіцієнти і вільні члени якої записуються у таблицю

x_1	x_2	x_3	x_4	x_5	b_i
1	1	1	1	1	7
0	-1	-2	-2	-6	-23
0	1	2	2	6	23
0	-1	-2	-2	-6	-23

Викресливши другий і четвертий рядки (вони разом з третім рядком є пропорційні між собою) та застосувавши перетворення Жордана-Гауса до рядків, що залишились, (за розв'язувальний елемент візьмемо коефіцієнт при x_2), дістанемо

x_1	x_2	x_3	x_4	x_5	b_i
1	0	-1	-1	-5	-16
0	1	2	2	6	23

Таким чином, задану систему зведено до *одиночного базису* і вона є сумісною та невизначеною; базисні невідомі – x_1, x_2 , а вільні невідомі – x_3, x_4, x_5 .

Загальним розв'язком буде
$$\begin{cases} x_1 = x_3 + x_4 + 5x_5 - 16 \\ x_2 = -2x_3 - 2x_4 - 6x_5 + 23 \end{cases}$$
 Базисний

розв'язок $x_1 = -16, x_2 = 23, x_3 = x_4 = x_5 = 0$ не є опорним. Фундаментальних розв'язків маємо три:

$$x'_1 = -15, x'_2 = 21, x'_3 = 1, x'_4 = x'_5 = 0,$$

$$x''_1 = -15, x''_2 = 21, x''_3 = 0, x''_4 = 1, x''_5 = 0,$$

$$x'''_1 = -1, x'''_2 = 17, x'''_3 = x'''_4 = 0, x'''_5 = 1 \bullet$$

Приклад 2.6. Економічна система складається з двох взаємозв'язаних галузей виробництва. Продукція кожної з них частково йде на зовнішні потреби (кінцевий продукт) і частково використовується у цих же галузях (галузеві потреби). Міжгалузеві виробничі зв'язки системи задано таблицею:

галузеві потреби					
			Кінцевий	Валовий	Кінцевий про-

галузеві пропозиції	1	2	продукт	випуск	дукт нового асортименту
1	100	160	240	500	480
2	275	40	85	400	170

- Визначити: 1. коефіцієнти прямих витрат;
 2. коефіцієнти повних витрат;
 3. коефіцієнти посередницьких витрат;
 4. валовий випуск продукції нового асортименту.

Розв'язання. Кожна з двох галузей виступає і як виробник продукції і як її споживач. Нехай x_i – валовий випуск продукції i -тої галузі. y_i – її кінцевий продукт, тоді $x_i - y_i$ є частиною продукції, що передбачається для внутрішніх потреб ($i = 1, 2$); x_{ik} – частина продукції i -тої галузі, що забезпечує випуск продукції k -тої галузі у розмірі x_k ($k = 1, 2$). Знайдемо коефіцієнти прямих витрат, тобто елементи структурної (технологічної) матриці:

$$a_{11} = \frac{x_{11}}{x_1} = \frac{100}{500} = 0,2; a_{12} = \frac{x_{12}}{x_2} = \frac{160}{400} = 0,4;$$

$$a_{21} = \frac{x_{21}}{x_1} = \frac{275}{500} = 0,55; a_{22} = \frac{x_{22}}{x_2} = \frac{40}{400} = 0,1; \quad A = \begin{pmatrix} 0,2 & 0,4 \\ 0,55 & 0,1 \end{pmatrix}.$$

Міжгалузеві виробничі взаємні зв'язки задовольняють умовам:

$$\begin{cases} x_1 - 0,2x_1 - 0,4x_2 = y_1 \\ x_2 - 0,55x_1 - 0,1x_2 = y_2 \end{cases},$$

або у матричному вигляді: $X - AX = Y$, де $X = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$, $Y = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}$. Звідси маємо

$EX - AX = Y$ і $(E - A)X = Y$, де E – одинична матриця. Позначимо $E - A = B$, тоді дістанемо $BX = Y$, причому

$$B = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} - \begin{pmatrix} 0,2 & 0,4 \\ 0,55 & 0,1 \end{pmatrix} = \begin{pmatrix} 0,8 & -0,4 \\ -0,55 & 0,9 \end{pmatrix},$$

де

$$\det B = \begin{vmatrix} 0,8 & -0,4 \\ -0,55 & 0,9 \end{vmatrix} = 0,5 \neq 0.$$

Оскільки $B_{11} = (-1)^{1+1} \cdot 0,9 = 0,9$; $B_{12} = (-1)^{1+2} \cdot (-0,55) = 0,55$;

$$B_{21} = (-1)^{2+1} \cdot (-0,4) = 0,4; \quad B_{22} = (-1)^{2+2} \cdot 0,8 = 0,8,$$

то $B^{-1} = \frac{1}{0,5} \begin{pmatrix} 0,9 & 0,4 \\ 0,55 & 0,8 \end{pmatrix} = \begin{pmatrix} 1,8 & 0,8 \\ 1,1 & 1,6 \end{pmatrix}$. Елементи матриці B^{-1} – це коефіцієнти повних витрат.

Коефіцієнти посередницьких (непрямих) витрат – елементи матриці $C = B^{-1} - A = \begin{pmatrix} 1,8 & 0,8 \\ 1,1 & 1,6 \end{pmatrix} - \begin{pmatrix} 0,2 & 0,4 \\ 0,55 & 0,1 \end{pmatrix} = \begin{pmatrix} 1,6 & 0,4 \\ 0,55 & 1,5 \end{pmatrix}$.

Розв'язавши матричне рівняння $BX = Y$, дістанемо валовий випуск продукції нового асортименту:

$$X = B^{-1}Y = \begin{pmatrix} 1,8 & 0,8 \\ 1,1 & 1,6 \end{pmatrix} \cdot \begin{pmatrix} 480 \\ 170 \end{pmatrix} = \begin{pmatrix} 1000 \\ 800 \end{pmatrix},$$

тобто для першої галузі $x_1 = 1000$, а для другої $x_2 = 800$ ●

Завдання для самоконтролю

1. Що називається системою лінійних алгебраїчних рівнянь ?
2. Яка система лінійних рівнянь називається сумісною, несумісною, визначеною, невизначеною?
3. Записати формули Крамера. В якому випадку вони застосовуються?
4. В чому полягає матричний спосіб розв'язання систем лінійних рівнянь?

5. Сформулювати теорему Кронекера-Капеллі.

6. Як дослідити і розв'язати систему лінійних рівнянь методом Гауса?

7. Розв'язати систему за формулами Крамера
$$\begin{cases} 2x_1 - 4x_2 + 3x_3 = 1 \\ x_1 - 2x_2 + 4x_3 = 3 \\ 3x_1 - x_2 + 5x_3 = 2 \end{cases}$$

(Відповідь: $x_1 = -1$, $x_2 = 0$, $x_3 = 1$).

8. Розв'язати систему за формулами Крамера
$$\begin{cases} x_1 + 2x_2 + 3x_3 = -1 \\ x_1 - 3x_2 - 2x_3 = 3 \\ 2x_1 - x_2 + x_3 = -2 \end{cases}$$

(Відповідь: несумісна).

9. Використовуючи формули Крамера, розв'язати систему

$$\begin{cases} x_1 + x_2 - 2x_3 = 3 \\ x_1 + 2x_2 + x_3 = 4 \\ 2x_1 + 3x_2 - x_3 = 7 \end{cases} \quad (\text{Відповідь: } x_1 = 5x_3 + 2, \quad x_2 = 1 - 3x_3).$$

Матричним способом розв'язати системи

10.
$$\begin{cases} x_1 - x_3 = 2 \\ x_1 + x_2 + x_3 = 6 \\ x_2 + 3x_3 = 5 \end{cases} \quad (\text{Відповідь: } x_1 = 3, \quad x_2 = 2, \quad x_3 = 1).$$

11.
$$\begin{cases} 2x_1 + 3x_2 + x_3 = 1 \\ 2x_1 + x_2 + 3x_3 = 11 \\ 3x_1 + 2x_2 + x_3 = 5 \end{cases} \quad (\text{Відповідь: } x_1 = 2, \quad x_2 = -2, \quad x_3 = 3).$$

Дослідити на сумісність системи

12.
$$\begin{cases} x_1 + x_2 - 3x_3 + x_4 = 3 \\ 3x_1 + x_2 - 5x_3 + 5x_4 = 5 \\ x_1 - x_2 + x_3 + 3x_4 = -1 \end{cases} \quad (\text{Відповідь: Невизначена}).$$

$$13. \begin{cases} 3x_1 - 5x_2 + 2x_3 + 4x_4 = 2 \\ 7x_1 - 4x_2 + x_3 + 3x_4 = 5 \\ 5x_1 + 7x_2 - 4x_3 - 6x_4 = 3 \end{cases} \quad (\text{Відповідь: Несумісна}).$$

Дослідити системи методом Гауса і у випадку сумісності розв'язати

$$14. \begin{cases} 2x_1 - x_2 + 3x_3 + 2x_4 = 1 \\ 3x_1 + 3x_2 + 3x_3 + 2x_4 = 1 \\ 3x_1 - x_2 - x_3 + 2x_4 = -1 \\ 3x_1 - x_2 + 3x_3 - x_4 = -1 \end{cases}$$

$$(\text{Відповідь: } x_1 = -\frac{1}{2}, x_2 = \frac{1}{8}, x_3 = \frac{3}{8}, x_4 = \frac{1}{2}).$$

$$15. \begin{cases} 3x_1 + 4x_2 + 5x_3 + 7x_4 = 1 \\ 2x_1 + 6x_2 - 3x_3 + 4x_4 = 2 \\ 4x_1 + 2x_2 + 13x_3 + 10x_4 = 0 \\ 5x_1 + 21x_3 + 13x_4 = 3 \end{cases}$$

(Відповідь: Несумісна).

Розв'язати методом Жордана-Гауса системи

$$16. \begin{cases} x_1 - x_2 + x_3 + x_4 = 2 \\ x_1 + x_2 - 3x_3 + 2x_4 = 1 \\ x_1 + 2x_2 - x_3 + x_4 = 3 \\ 2x_1 - x_2 + x_3 - x_4 = 1 \end{cases}$$

(Відповідь: $x_1 = x_2 = x_3 = x_4 = 1$).

Знайти загальний та базисний розв'язки системи

$$17. \begin{cases} x_1 + 2x_2 + x_3 + x_4 = 5 \\ x_2 + x_3 + x_4 = 3 \\ x_1 + x_2 = 2 \end{cases}$$

$$(\text{Відповідь: } X = \begin{pmatrix} -1 + c_1 + c_2 \\ 3 - c_1 - c_2 \\ c_1 \\ c_2 \end{pmatrix}, X_B = \begin{pmatrix} -1 \\ 3 \\ 0 \\ 0 \end{pmatrix}).$$

$$18. \begin{cases} x_1 + x_2 = 1 \\ x_1 + x_2 + x_3 = 4 \\ x_2 + x_3 + x_4 = -3 \\ x_3 + x_4 + x_5 = 2 \\ x_4 + x_5 = -1 \end{cases}$$

$$\text{(Відповідь: } X = \begin{pmatrix} 6-c \\ 5-c \\ 3 \\ -1-c \\ c \end{pmatrix}, X_B = \begin{pmatrix} 6 \\ 5 \\ 3 \\ -1 \\ 0 \end{pmatrix}).$$

$$19. \begin{cases} 4x_1 + 2x_2 - 3x_3 + 2x_4 = 3 \\ 2x_1 + 3x_2 - 2x_3 + 3x_4 = 2 \\ 3x_1 + 2x_2 - 3x_3 + 4x_4 = 1 \end{cases}$$

$$\text{(Відповідь: } X = \begin{pmatrix} -10c+10 \\ c \\ -16c+15 \\ 4-5c \end{pmatrix}, X_B = \begin{pmatrix} 10 \\ 0 \\ 15 \\ 4 \end{pmatrix}).$$

Лекція № 3. Лінійні простори.

Багатовимірні вектори. Лінійна залежність і незалежність системи векторів. Теорема Штейниця

Ранг і базис системи векторів.

n-вимірний лінійний простір. Базис та вимірність простору.

Підпростір лінійного простору.

Перехід до нового базису.

3.1. Багатовимірні вектори. Лінійна залежність і незалежність системи векторів

Кожному одновимірному вектору (на прямій) співставляється дійсне число, кожному двовимірному вектору (на площині) – упорядкована пара дійсних чисел, а кожному тривимірному вектору (у просторі) – упорядкована трійка таких чисел. Узагальнюючи, скажемо, що упорядкована множина n дійсних чисел $x_1, x_2, \dots, x_n$ називається *n-вимірним вектором* і позначається $\mathbf{x} = (x_1,$

$x_2, \dots, x_n)$ або $\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$. Числа $x_1, x_2, \dots, x_n$ називаються компонентами вектора

$\mathbf{x}$. Вектори $\mathbf{x} = (x_1, x_2, \dots, x_n)$ і $\mathbf{y} = (y_1, y_2, \dots, y_n)$ вважаються рівними, якщо $x_i = y_i$ ($i = \overline{1, n}$). Сумою векторів $\mathbf{x}$ і $\mathbf{y}$ називається вектор $\mathbf{x} + \mathbf{y} = (x_1 + y_1, \dots, x_n + y_n)$; добутком вектора $\mathbf{x}$ на число α називається вектор $\alpha\mathbf{x} = (\alpha x_1, \dots, \alpha x_n)$; вектор $\mathbf{0} = (0, \dots, 0)$ називається нульовим; вектор $-\mathbf{x} = (-x_1, -x_2, \dots, -x_n)$ називається протилежним вектору $\mathbf{x}$.

Розглянемо систему n -вимірних векторів:

$$\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k. \quad (3.1)$$

Вектор $\mathbf{x}$ є *лінійною комбінацією* векторів (3.1), якщо існують такі числа $\alpha_1, \alpha_2, \dots, \alpha_k$, що $\mathbf{x} = \alpha_1\mathbf{x}_1 + \alpha_2\mathbf{x}_2 + \dots + \alpha_k\mathbf{x}_k$. Вектори (3.1) називаються *лінійно залежними*, якщо рівність:

$$\alpha_1\mathbf{x}_1 + \alpha_2\mathbf{x}_2 + \dots + \alpha_k\mathbf{x}_k = \mathbf{0} \quad (3.2)$$

можлива за умови, що хоча б одне з чисел $\alpha_i \neq 0$ ($i = \overline{1, k}$). Якщо ж рівність (3.2) можлива лише за умови $\alpha_1 = \alpha_2 = \dots = \alpha_k = 0$, то система векторів (3.1) називається *лінійно незалежною*.

Наприклад, система векторів $\mathbf{x}_1 = (1; 2; -1)$, $\mathbf{x}_2 = (2; 0; 3)$, $\mathbf{x}_3 = (-1; 6; 9)$ лінійно залежна, адже $3\mathbf{x}_1 - 2\mathbf{x}_2 - \mathbf{x}_3 = \mathbf{0}$. Вектори $\mathbf{x}_1 = (1; 0)$, $\mathbf{x}_2 = (1; 1)$ лінійно незалежні, бо рівність $\alpha_1\mathbf{x}_1 + \alpha_2\mathbf{x}_2 = \mathbf{0}$ виконується лише при $\alpha_1 = \alpha_2 = 0$. Дійсно, остання рівність еквівалентна системі рівнянь
$$\begin{cases} \alpha_1 + \alpha_2 = 0 \\ \alpha_2 = 0 \end{cases}$$
, розв'язком якої є $\alpha_1 = \alpha_2 = 0$.

Справедливі такі властивості:

1. Якщо деяка підсистема $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_s$ ($s \leq k$) системи векторів (3.1) лінійно залежна, то вся система векторів лінійно залежна.

2. Система векторів лінійно залежна, якщо вона містить нульовий вектор або рівні чи пропорційні вектори.

3. Якщо система векторів лінійно незалежна, то і будь-яка її підсистема лінійно незалежна. Така властивість є очевидною.

4. Вектори (3.1) лінійно залежні тоді і тільки тоді, коли один з них є лінійною комбінацією інших.

Теорема 3.1. (Штейниця). Якщо кожний із векторів $\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_m$ є лінійною комбінацією векторів $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n$ і $m > n$, то вектори $\mathbf{y}_i$ ($i = \overline{1, m}$) – лінійно залежні.

Наслідок. В будь-якій системі n -вимірних векторів не може бути більше, ніж n лінійно незалежних.

Нехай маємо вектори (3.1), для яких виконується (3.2), або

$$\begin{cases} x_{11}\alpha_1 + x_{21}\alpha_2 + \dots + x_{k1}\alpha_k = 0; \\ x_{12}\alpha_1 + x_{22}\alpha_2 + \dots + x_{k2}\alpha_k = 0; \\ \text{-----} \\ x_{1n}\alpha_1 + x_{2n}\alpha_2 + \dots + x_{kn}\alpha_k = 0, \end{cases} \quad (3.3)$$

де $M = \begin{pmatrix} x_{11} & x_{21} & \dots & x_{k1} \\ x_{12} & x_{22} & \dots & x_{k2} \\ \dots & \dots & \dots & \dots \\ x_{1n} & x_{2n} & \dots & x_{kn} \end{pmatrix}$ – матриця системи, стовпці якої складені з

компонент векторів $\mathbf{x}_i$ ($i = \overline{1, k}$). Оскільки рівність (3.2) і система (3.3) еквівалентні, то вектори (3.1) лінійно залежні тоді і тільки тоді, коли система рівнянь (3.3) має ненульовий розв'язок $\alpha_1, \alpha_2, \dots, \alpha_k$, або вектори (3.1) лінійно незалежні тоді і тільки тоді, коли ранг матриці M дорівнює k , тобто числу векторів системи (3.3).

У випадку коли $k = n$, досить розглянути лише складений із

компонент векторів (3.1) визначник $\Delta = \begin{vmatrix} x_{11} & x_{21} & \dots & x_{n1} \\ x_{12} & x_{22} & \dots & x_{n2} \\ \dots & \dots & \dots & \dots \\ x_{1n} & x_{2n} & \dots & x_{nn} \end{vmatrix}$. Якщо $\Delta \neq 0$, то

система векторів (3.1) лінійно незалежна, якщо ж $\Delta = 0$, то така система лінійно залежна.

Приклад 3.1. З'ясувати, чи будуть вектори $\mathbf{x}_1 = (2; 1; 0; 1)$, $\mathbf{x}_2 = (1; -1; 2; -1)$, $\mathbf{x}_3 = (1; 2; 1; 0)$ лінійно незалежними?

Розв'язання. $M = \begin{pmatrix} 2 & 1 & 1 \\ 1 & -1 & 2 \\ 0 & 2 & 1 \\ 1 & -1 & 0 \end{pmatrix} \sim \begin{pmatrix} 1 & -1 & 0 \\ 2 & 1 & 1 \\ 1 & -1 & 2 \\ 0 & 2 & 1 \end{pmatrix} \sim \begin{pmatrix} 1 & -1 & 0 \\ 0 & 3 & 1 \\ 0 & 0 & 2 \\ 0 & 2 & 1 \end{pmatrix} \sim$

$$\sim \begin{pmatrix} 1 & -1 & 0 \\ 0 & 3 & 1 \\ 0 & 2 & 1 \\ 0 & 0 & 2 \end{pmatrix} \sim \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \sim \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}; \quad r(M) = 3. \quad \text{Отже, задані вектори}$$

лінійно незалежні●

3.2. Ранг і базис системи векторів.

Рангом r заданої системи векторів називається максимальне число лінійно незалежних векторів цієї системи.

Тут справджуються наступні теореми.

Теорема 3.2. Ранг системи векторів дорівнює рангу матриці, складеної з компонент цих векторів.

Доведення. Дійсно, якщо ранг системи векторів дорівнює r , то вона має максимально r лінійно незалежних векторів, компоненти яких складають матрицю M з рангом r . Матриця M є частиною матриці M' , яка складається з компонент усіх векторів системи. Матриця M' має ранг не менший r , але і не більший r , інакше в системі знайшлося б більше ніж r лінійно незалежних векторів. Отже $r(M') = r$ ●

Наслідок. Ранг матриці дорівнює максимальному числу її лінійно незалежних стовпців (або рядків).

Теорема 3.3. Будь-який вектор системи векторів можна єдиним чином зобразити у вигляді лінійної комбінації векторів базису цієї системи.

Теорема 3.4. Якщо до системи векторів (3.1) додати або з неї вилучити вектор, який є лінійною комбінацією векторів цієї системи, то ранг такої системи не зміниться.

Доведення. Розглянемо системи векторів, а саме

$$\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k; \quad (3.1)$$

$$\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k; \quad (3.4)$$

$$\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_{k-1}, \quad (3.5)$$

причому $\mathbf{x}_0 = \alpha_1 \mathbf{x}_1 + \alpha_2 \mathbf{x}_2 + \dots + \alpha_k \mathbf{x}_k$ і $\mathbf{x}_k = \alpha'_1 \mathbf{x}_1 + \dots + \alpha'_{k-1} \mathbf{x}_{k-1}$.

Якщо вказана система (3.1) має ранг r , то будь-який вектор цієї системи є лінійною комбінацією r векторів її базису. Вектор $\mathbf{x}_0$, який лінійно виражається через вектори (3.1), буде також лінійно виражатися через ці ж r лінійно незалежні вектори. Отже, ті ж самі r векторів утворюють базис системи (3.4) і тоді її ранг теж дорівнює r .

Розглянемо тепер систему (3.5). Якщо до неї додати вектор $\mathbf{x}_k$, то за попереднім її ранг не зміниться, але тоді вона співпадає з системою (3.1), яка має ранг r . Таким чином, ранг системи (3.5) теж дорівнює r ●

Теорема 3.5. У будь-якій системі векторів кожен її множину лінійно незалежних векторів можна доповнити до базису цієї системи.

3.3. n -вимірний лінійний простір.

Дійсні числа можна додавати і множити за відомими правилами арифметики, отримуючи при цьому такі ж самі числа, тобто дійсні числа

утворюють поле. (Поняття, на якому ґрунтується означення лінійного простору, що надалі, як правило, використовується неявно).

Приклади лінійних просторів відомі ще з аналітичної геометрії. Так, вектори на прямій (на площині або у просторі) утворюють одновимірний (двовимірний або тривимірний) простір R^1 (R^2 або R^3) над полем дійсних чисел.

Для будь-яких n -вимірних векторів $\mathbf{x}$, $\mathbf{y}$, $\mathbf{z}$ і довільних дійсних чисел α і β виконуються властивості:

- 1) $\mathbf{x} + \mathbf{y}$ – однозначно визначений n -вимірний вектор;
- 2) $\mathbf{x} + \mathbf{y} = \mathbf{y} + \mathbf{x}$;
- 3) $(\mathbf{x} + \mathbf{y}) + \mathbf{z} = \mathbf{x} + (\mathbf{y} + \mathbf{z})$;
- 4) існує вектор $\mathbf{0}$: $\mathbf{x} + \mathbf{0} = \mathbf{x}$;
- 5) для будь-якого вектора $\mathbf{x}$ існує протилежний вектор $-\mathbf{x}$, так що $\mathbf{x} + (-\mathbf{x}) = \mathbf{0}$;
- 6) $\alpha\mathbf{x}$ – n -вимірний вектор;
- 7) $\alpha(\beta\mathbf{x}) = (\alpha\beta)\mathbf{x}$;
- 8) $\alpha(\mathbf{x} + \mathbf{y}) = \alpha\mathbf{x} + \alpha\mathbf{y}$;
- 9) $(\alpha + \beta)\mathbf{x} = \alpha\mathbf{x} + \beta\mathbf{x}$;
- 10) $1 \cdot \mathbf{x} = \mathbf{x}$.

Сукупність усіх n -вимірних векторів із введеними операціями додавання і множення на дійсне число, які задовольняють вище вказаним властивостям, називається n -вимірним лінійним (або векторним) простором і позначається R^n .

Слід зауважити, що будь-яка множина елементів (які інтуїтивно відповідають векторам і внаслідок цього вони називаються векторами), що задовольняє властивостям 1) – 10), називається *лінійним простором* над полем дійсних чисел. Всі можливі лінійні простори можна поділити на два класи. Простори, в яких є лінійно незалежна система, що складається з n векторів, але немає лінійно незалежних систем з більшою кількістю векторів, називаються *скінченновимірними* або просторами вимірності n (n -вимірними просторами). Простори, в яких є лінійно незалежні системи, що складаються з як завгодно великої кількості векторів, називаються *нескінченновимірними*.

В лінійній алгебрі вивчаються лише скінченновимірні (n -вимірні) простори. Зазначимо, що для таких просторів залишаються вірними всі поняття, означення та твердження, розглянуті для систем n -вимірних векторів.

Як і у просторах R^2 та R^3 кожній парі векторів $\mathbf{x} = (x_1, \dots, x_n)$ і $\mathbf{y} = (y_1, \dots, y_n)$ простору R^n можна поставити у відповідність дійсне число $\mathbf{x}\mathbf{y} = x_1y_1 + \dots + x_ny_n$, яке називається *скалярним добутком* цих векторів.

Використовуючи це поняття визначають довжину (модуль) вектора $\mathbf{x} = (x_1, \dots, x_n)$ як число $|\mathbf{x}| = \sqrt{\mathbf{x}\mathbf{x}} = \sqrt{\sum_{i=1}^n x_i^2}$.

Лінійний простір з означеною у ньому операцією скалярного добутку називається *евклідовим* і позначається так само R^n .

Вимірність простору – це максимальне число лінійно незалежних його векторів. Так вимірність простору плоских векторів дорівнює 2.

Базисом n -вимірному лінійному простору називається будь-яка система з n лінійно незалежних векторів цього простору.

Розглянемо в n -вимірному лінійному просторі систему одиничних векторів:

$$\mathbf{e}_1 = (1; 0; \dots; 0); \mathbf{e}_2 = (0; 1; \dots; 0); \dots; \mathbf{e}_n = (0; 0; \dots; 1).$$

Оскільки рівність $\alpha_1 \mathbf{e}_1 + \dots + \alpha_n \mathbf{e}_n = \mathbf{0}$ можлива лише при $\alpha_1 = \alpha_2 = \dots = \alpha_n = 0$, то система одиничних векторів є лінійно незалежною і, таким чином, утворює базис n -вимірному лінійному простору.

Будь-який вектор $\mathbf{x} = (x_1, x_2, \dots, x_n)$ цього простору розкладається за векторами такого базису, а саме $\mathbf{x} = x_1 \mathbf{e}_1 + \dots + x_n \mathbf{e}_n$.

Приклад 3.5. Знайти розклад вектора $\mathbf{x} = (3; 6; 5)$ за базисом $\mathbf{x}_1 = (1; 2; 3)$; $\mathbf{x}_2 = (3; 2; 1)$; $\mathbf{x}_3 = (1; 0; 1)$.

Розв'язання. Запишемо рівняння для знаходження коефіцієнтів розкладу вектора $\mathbf{x}$ за базисом: $\alpha_1 \mathbf{x}_1 + \alpha_2 \mathbf{x}_2 + \alpha_3 \mathbf{x}_3 = \mathbf{x}$, яке еквівалентне системі лінійних рівнянь:

$$\begin{cases} \alpha_1 + 3\alpha_2 + \alpha_3 = 3 \\ 2\alpha_1 + 2\alpha_2 = 6 \\ 3\alpha_1 + \alpha_2 + \alpha_3 = 5 \end{cases}.$$

Розв'язуючи її, отримуємо $\alpha_1 = 2$; $\alpha_2 = 1$; $\alpha_3 = -2$. Отже, $\mathbf{x} = 2\mathbf{x}_1 + \mathbf{x}_2 - 2\mathbf{x}_3$ ●

Підпростір лінійного простору R^n – це множина R_1 його елементів, яка сама є лінійним простором відносно введених в R^n операцій додавання і множення на число.

Очевидно, що єдиний нульовий вектор $\mathbf{0} \in R^n$ і сам простір R^n є відповідно найменшим та найбільшим своїми підпросторами. У просторі R^3 підпросторами будуть R^2 і R^1 (площина і пряма, що проходять через початок координат).

Нескладно показати, що підмножина R_1 елементів простору R^n буде його підпростором тоді і тільки тоді, коли виконуються умови:

- 1) $\mathbf{x} + \mathbf{y} \in R_1, \quad \mathbf{x}, \mathbf{y} \in R_1$;
- 2) $\alpha \mathbf{x} \in R_1, \quad \mathbf{x} \in R_1, \alpha$ – дійсне число;

або інакше $\alpha \mathbf{x} + \beta \mathbf{y} \in R_1, \mathbf{x}, \mathbf{y} \in R_1, \alpha, \beta$ – дійсні числа.

Ясно, що вимірність підпростору лінійного простору не перебільшує вимірності самого простору. Неважко переконатись, що перетин $R_1 \cap R_2$ підпросторів R_1 і R_2 лінійного простору R , тобто сукупність всіх векторів, що належать одночасно і R_1 і R_2 , також є підпростором цього простору; сума

$R_1 + R_2$ цих підпросторів, тобто множина всіх векторів виду $x + y$, де $x \in R_1$, $y \in R_2$, теж є підпросторами лінійного простору.

Якщо $x, y, \dots, z$ – деякі вектори лінійного простору R , то очевидно, всі можливі лінійні комбінації цих векторів $\alpha x + \beta y + \dots + \gamma z$ є підпростором в R . Цей підпростір називається *лінійною оболонкою* векторів $x, y, \dots, z$, і кажуть, що такий підпростір породжується векторами $x, y, \dots, z$ або напнутий на ці вектори. Неважко зрозуміти, що лінійна оболонка векторів $x, y, \dots, z$ співпадає з перетином всіх підпросторів, які містять ці вектори.

3.4. Перехід до нового базису.

Нехай $e_1, \dots, e_n$ – старий базис простору R^n , а $e'_1, \dots, e'_n$ – його новий базис. Елементи нового базису виражаються через вектори старого базису, а саме:

$$e'_1 = a_{11}e_1 + a_{21}e_2 + \dots + a_{n1}e_n;$$

$$e'_2 = a_{12}e_1 + a_{22}e_2 + \dots + a_{n2}e_n;$$

$$-----$$

$$e'_n = a_{1n}e_1 + a_{2n}e_2 + \dots + a_{nn}e_n. \quad (3.8)$$

Матриця $A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix}$ називається матрицею переходу від базису

$e_1, \dots, e_n$ до базису $e'_1, \dots, e'_n$ і є завжди невинродженою.

І навпаки, всяка квадратна матриця порядку n з відмінним від нуля визначником може бути матрицею переходу від одного базису n -вимірного простору до іншого.

Нехай тепер вектор $x = x_1e_1 + \dots + x_ne_n$ – в старому базисі і в той же час $x = x'_1e'_1 + \dots + x'_ne'_n$ – в новому. Враховуючи (6.1), маємо:

$$\begin{aligned} x &= x'_1(a_{11}e_1 + a_{21}e_2 + \dots + a_{n1}e_n) + x'_2(a_{12}e_1 + a_{22}e_2 + \dots + a_{n2}e_n) + \\ &+ \dots + x'_n(a_{1n}e_1 + a_{2n}e_2 + \dots + a_{nn}e_n) = (a_{11}x'_1 + a_{12}x'_2 + \dots + a_{1n}x'_n)e_1 + \\ &+ (a_{21}x'_1 + a_{22}x'_2 + \dots + a_{2n}x'_n)e_2 + \dots + (a_{n1}x'_1 + a_{n2}x'_2 + \dots + a_{nn}x'_n)e_n. \end{aligned}$$

З єдиності розкладу вектора x за базисом $e_1, \dots, e_n$ випливає:

$$x_1 = a_{11}x'_1 + a_{12}x'_2 + \dots + a_{1n}x'_n;$$

$$x_2 = a_{21}x'_1 + a_{22}x'_2 + \dots + a_{2n}x'_n;$$

$$-----$$

$$x_n = a_{n1}x'_1 + a_{n2}x'_2 + \dots + a_{nn}x'_n.$$

Таким чином, старі координати вектора x знаходять через нові його координати за допомогою матриці переходу A .

Приклад 3.6. Довести, що кожна система векторів утворює базис простору та знайти зв'язок між координатами довільного вектора в цих азисах: $e_1 = (1; 1; 1)$, $e_2 = (1; 2; 1)$, $e_3 = (1; 1; 3)$; $e'_1 = (1; 2; 1)$, $e'_2 = (2; 1; 3)$, $e'_3 = (3; 2; 1)$.

Розв'язання. Обчислимо визначники, складені відповідно з координат систем векторів у деякому базисі тривимірного простору, а саме:

$$\begin{vmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 3 \end{vmatrix} = 2 \neq 0; \quad \begin{vmatrix} 1 & 2 & 3 \\ 2 & 1 & 2 \\ 1 & 3 & 1 \end{vmatrix} = 10 \neq 0.$$

Отже, системи векторів $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$ та $\mathbf{e}'_1, \mathbf{e}'_2, \mathbf{e}'_3$ є лінійно незалежними і утворюють базиси тривимірного простору. Нехай $\mathbf{x}$ – довільний вектор, тоді:
 $\mathbf{x} = x_1\mathbf{e}_1 + x_2\mathbf{e}_2 + x_3\mathbf{e}_3 = x'_1\mathbf{e}'_1 + x'_2\mathbf{e}'_2 + x'_3\mathbf{e}'_3$. Звідси маємо:

$$\begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 3 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 1 & 2 \\ 1 & 3 & 1 \end{pmatrix} \cdot \begin{pmatrix} x'_1 \\ x'_2 \\ x'_3 \end{pmatrix}. \quad \text{Знайдемо обернену матрицю}$$

$$D^{-1} = \begin{pmatrix} \frac{5}{2} & -1 & -\frac{1}{2} \\ -1 & 1 & 0 \\ -\frac{1}{2} & 0 & \frac{1}{2} \end{pmatrix} \quad \text{до матриці} \quad D = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 3 \end{pmatrix} \quad \text{і помножимо останню}$$

рівність на матрицю D^{-1} зліва, тобто:

$$\begin{pmatrix} \frac{5}{2} & -1 & -\frac{1}{2} \\ -1 & 1 & 0 \\ -\frac{1}{2} & 0 & \frac{1}{2} \end{pmatrix} \cdot \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 3 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} \frac{5}{2} & -1 & -\frac{1}{2} \\ -1 & 1 & 0 \\ -\frac{1}{2} & 0 & \frac{1}{2} \end{pmatrix} \cdot \begin{pmatrix} 1 & 2 & 3 \\ 2 & 1 & 2 \\ 1 & 3 & 1 \end{pmatrix} \cdot \begin{pmatrix} x'_1 \\ x'_2 \\ x'_3 \end{pmatrix},$$

або $\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 0 & \frac{5}{2} & 5 \\ 1 & -1 & -1 \\ 0 & -\frac{1}{2} & -1 \end{pmatrix} \cdot \begin{pmatrix} x'_1 \\ x'_2 \\ x'_3 \end{pmatrix}$. Таким чином, $\begin{matrix} x_1 = \frac{5}{2}x'_2 + 5x'_3; \\ x_2 = x'_1 - x'_2 - x'_3; \\ x_3 = -\frac{1}{2}x'_2 - x'_3, \end{matrix}$ де

$$A = \begin{pmatrix} 0 & \frac{5}{2} & 5 \\ 1 & -1 & -1 \\ 0 & -\frac{1}{2} & -1 \end{pmatrix} \quad \text{– матриця переходу від базиса } \mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3 \text{ до нового базису } \mathbf{e}'_1, \mathbf{e}'_2, \mathbf{e}'_3.$$

Завдання для самоконтролю

1. З'ясувати поняття лінійної залежності векторів та навести його властивості.
2. Сформулювати теорему Штейниця.
3. Дати ознаку лінійної залежності і незалежності векторів.
4. Що називається рангом та базисом системи векторів?
5. Дати поняття n -вимірного вектора, n -вимірного лінійного простору, підпростору.
6. Що таке вимірність простору?
7. Який зв'язок між координатами вектора в старому та новому базисах?

З'ясувати, які системи векторів лінійно залежні, а які незалежні

8. $\mathbf{a}_1 = (1; -2; 1; -3)$, $\mathbf{a}_2 = (1; 4; 1; 2)$, $\mathbf{a}_3 = (1; -3; 1; 4)$.

(Відповідь: лінійно незалежні).

9. $\mathbf{a}_1 = (3; -2; 4; 1)$, $\mathbf{a}_2 = (-1; 2; -1; 2)$, $\mathbf{a}_3 = (1; 2; 2; 5)$.

(Відповідь: лінійно залежні).

Знайти ранг системи векторів .

10. $\mathbf{a}_1 = (2; 3; 4; -1)$, $\mathbf{a}_2 = (3; 1; 2; 0)$, $\mathbf{a}_3 = (1; 1; 3; 5)$.(Відповідь: $r = 3$).

11. $\mathbf{a}_1 = (4; 8; 4; 4; 8)$, $\mathbf{a}_2 = (3; 6; 3; 3; 6)$, $\mathbf{a}_3 = (-5; -7; -8; -1; -1)$, $\mathbf{a}_4 = (2; 4; 2; 2; 4)$, $\mathbf{a}_5 = (3; 2; 7; -5; -6)$, $\mathbf{a}_6 = (1; 0; 2; 3; 1)$.(Відповідь: $r = 4$).

12. $\mathbf{a}_1 = (1; 2; 1)$, $\mathbf{a}_2 = (3; 7; 4)$, $\mathbf{a}_3 = (5; 5; 0)$, $\mathbf{a}_4 = (4; 1; -3)$.

(Відповідь: $r = 2$).

Довести, що вектори $\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_n$ утворюють базис та знайти координати вектора $\mathbf{x}$ у цьому базисі:

13. $\mathbf{e}_1 = (2; 3; 4)$, $\mathbf{e}_2 = (3; -2; 1)$, $\mathbf{e}_3 = (-1; 2; 1)$, $\mathbf{x} = (4; 3; 6)$.

(Відповідь: $(1; 1; 1)$).

14. $\mathbf{e}_1 = (1; -1; 1)$, $\mathbf{e}_2 = (2; 1; 1)$, $\mathbf{e}_3 = (1; 3; 1)$, $\mathbf{x} = (3; -4; 2)$.

(Відповідь: $(2; 1; -1)$).

15. Показати, що кожна система векторів: $\mathbf{e}_1 = (1; 2; 1)$, $\mathbf{e}_2 = (2; 3; 3)$, $\mathbf{e}_3 = (3; 7; 1)$ і $\mathbf{e}'_1 = (3; 1; 4)$, $\mathbf{e}'_2 = (5; 2; 1)$, $\mathbf{e}'_3 = (1; 1; -6)$ утворює базис та знайти зв'язок між координатами довільного вектора у цих базисах.

(Відповідь:

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 27 \\ 9 \\ 4 \end{pmatrix} x'_1 + \begin{pmatrix} 7 \\ 20 \\ 12 \end{pmatrix} x'_2 + \begin{pmatrix} 1 \\ 9 \\ 8 \end{pmatrix} x'_3 = \begin{pmatrix} 27x'_1 + 7x'_2 + x'_3 \\ 9x'_1 + 20x'_2 + 9x'_3 \\ 4x'_1 + 12x'_2 + 8x'_3 \end{pmatrix}).$$

Знайти координати вектора $\mathbf{x}$ у базисі $\mathbf{e}'_1, \mathbf{e}'_2, \mathbf{e}'_3$, якщо він заданий у базисі $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$:

$$16. \begin{cases} \mathbf{e}'_1 = \mathbf{e}_1 + \mathbf{e}_2 + 3\mathbf{e}_3; \\ \mathbf{e}'_2 = \frac{3}{2}\mathbf{e}_1 - \mathbf{e}_2; \\ \mathbf{e}'_3 = -\mathbf{e}_1 + \mathbf{e}_2 + \mathbf{e}_3, \end{cases} \quad \mathbf{x} = (1; 2; 4). \quad (\text{Відповідь: } (2; -2; -2)).$$

$$17. \begin{cases} \mathbf{e}'_1 = \mathbf{e}_1 + \mathbf{e}_2 + 4\mathbf{e}_3; \\ \mathbf{e}'_2 = \frac{4}{3}\mathbf{e}_1 - \mathbf{e}_2; \\ \mathbf{e}'_3 = -\mathbf{e}_1 + \mathbf{e}_2 + \mathbf{e}_3, \end{cases} \quad \mathbf{x} = (1; 3; 6). \quad (\text{Відповідь: } (3; -6; -6)).$$

18. Знайти зв'язок між координатами довільного вектора в базисах: $\mathbf{e}_1 = (1; 2; 3)$, $\mathbf{e}_2 = (2; 1; 2)$, $\mathbf{e}_3 = (0; 1; 1)$ і $\mathbf{e}'_1 = (0; 1; 1)$, $\mathbf{e}'_2 = (1; 0; 1)$, $\mathbf{e}'_3 = (1; 1; 0)$.

$$\text{(Відповідь: } \begin{cases} x_1 = x'_2 - 3x'_3; \\ x_2 = 2x'_3; \\ x_3 = x'_1 - 2x'_2 + 5x'_3. \end{cases} \text{)}$$

19. Показати, що системи векторів: $\mathbf{e}_1 = (1; 1; 1)$, $\mathbf{e}_2 = (2; 1; 1)$, $\mathbf{e}_3 = (1; 1; 3)$ і $\mathbf{e}'_1 = (\mathbf{O}; \mathbf{1}; \mathbf{1})$, $\mathbf{e}'_2 = (1; 0; 1)$, $\mathbf{e}'_3 = (1; 0; 2)$ утворюють базис. Знайти зв'язок між координатами довільного вектора в цих базисах.

$$\text{(Відповідь: } \begin{cases} x_1 = 2x'_1 - \frac{3}{2}x'_2 - 2x'_3; \\ x_2 = -x'_1 + x'_2 + x'_3; \\ x_3 = \frac{1}{2}x'_2 + x'_3. \end{cases} \text{)}.$$

МАТЕМАТИЧНЕ ПРОГРАМУВАННЯ

Лекція № 4. Постановка задач математичного програмування

Класифікація задач математичного програмування.

Моделі задач лінійного програмування.

Різні форми запису задач лінійного програмування.

Основні властивості розв'язків задач лінійного програмування.

Геометрична інтерпретація і графічний метод розв'язання задачі лінійного програмування.

4.1. Класифікація задач математичного програмування

Математичне програмування – математична дисципліна, яка використовує математичні моделі і методи для розв'язування оптимізаційних задач, до яких належать задачі раціонального розподілу обмежених ресурсів для ефективного досягнення поставленої мети. Такі проблеми виникають у багатьох сферах діяльності людини, насамперед економічній: розподіл сировини, обладнання, праці, коштів при плануванні виробництва, розподілі та збуті продукції.

Розв'язування екстремальних економічних задач методами математичного програмування включає у себе два етапи: а) побудову математичної моделі; б) знаходження оптимального розв'язку. Як відомо, математика досліджує не реальні об'єкти і явища, а їхні абстракції. Однією із таких абстракцій є математична модель – спрощений опис реального явища засобами математичного апарату. В ній відображається суть ситуації, яка досліджується, і враховуються найбільш характерні риси, які можуть вплинути на результат. Модель задачі математичного програмування містить шукані невідомі $\mathbf{x} = (x_1, x_2, \dots, x_n)$, які належать деякій множині $X \subset \mathbb{R}^n$, цільову функцію $F = (x_1, x_2, \dots, x_n)$ кількісної оцінки ефективності розв'язку:

$$F(\mathbf{x}) \rightarrow \max(\min),$$

умови-обмеження на ресурси

$$\varphi_i(\mathbf{x}) \begin{cases} \leq \\ \geq \end{cases} b_i, \quad i = 1, 2, \dots, m.$$

Тут кожне із m обмежень має лише один знак відношення $\leq$, $=$ або $\geq$; функції F , φ_i та числа b_i ($i = \overline{1, m}$), вважаються заданими. Елементи множини X , які задовольняють умовам обмежень, називаються *допустимими розв'язками* або *планами*. Сукупність таких планів утворює *область D допустимих розв'язків* задачі. Допустимий план, при якому цільова функція $F(\mathbf{x})$ досягає максимуму (мінімуму), називається *оптимальним розв'язком* або *оптимальним планом* задачі математичного програмування.

В залежності від особливостей функцій $F(\mathbf{x})$ і $\varphi_i(\mathbf{x})$ ($i = \overline{1, m}$), задачі математичного програмування поділяються на ряд класів.

Якщо функції $F(\mathbf{x})$ та $\varphi_i(\mathbf{x})$ – лінійні, тобто $F(\mathbf{x}) = \sum_{j=1}^n c_j x_j$, $\varphi_i(\mathbf{x}) = \sum_{j=1}^n a_{ij} x_j$ ($i = \overline{1, m}$), то маємо задачу *лінійного програмування*. Тут a_{ij} , c_j – задані числа.

До задач *нелінійного програмування* належать такі задачі, математична модель яких містить хоча б одну нелінійну функцію $F(\mathbf{x})$ або $\varphi_i(\mathbf{x})$. Якщо при цьому відомо, що D – опукла область, то маємо задачу *опуклого програмування*. Частинним випадком задач опуклого програмування є задача *квадратичного програмування*. У ній область D задається лінійними обмеженнями (рівняннями, нерівностями), а $F(\mathbf{x})$ – квадратична функція.

Задачі математичного програмування, в яких на всі або частину змінних накладено умову цілочисельності, називаються задачами *цілочислового програмування*. Вони, в свою чергу, є частинним випадком більш широкого класу задач *дискретного програмування*.

Задача, процес пошуку розв'язку якої займає багато послідовних етапів (кроків), належить до задач *динамічного програмування*.

Задачі, математичні моделі яких враховують випадкові фактори і невизначеності, називаються задачами *стохастичного програмування*.

4.2. Моделі задач лінійного програмування

Важливе місце в математичному програмуванні посідають задачі лінійного програмування. Загальна задача лінійного програмування полягає в знаходженні екстремуму (максимуму або мінімуму) функції

$$F = \sum_{j=1}^n c_j x_j \rightarrow \max(\min) \quad (4.1)$$

за умов

$$\sum_{j=1}^n a_{ij} x_j \begin{cases} \leq \\ = \\ \geq \end{cases} b_i, \quad i = \overline{1, m}; \quad (4.2)$$

$$x_j \geq 0, \quad j = \overline{1, k}, \quad (k \leq n). \quad (4.3)$$

Такі задачі досить часто зустрічаються на практиці, наприклад, при вирішенні проблем, пов'язаних з розподілом ресурсів, плануванням виробництва, організацією роботи транспорту, тощо.

Побудова математичної моделі конкретної задачі передбачає виконання такої послідовності дій:

- 1) введення змінних, значення яких потрібно знайти;
- 2) формулювання критерію оптимальності, запис цільової функції;
- 3) визначення обмежень на ресурси і вираження цих умов через змінні.

Розглянемо деякі моделі задач лінійного програмування.

Задача 4.1. (про використання ресурсів). Для виробництва n видів продукції $\Pi_1, \Pi_2, \dots, \Pi_n$ використовується m видів ресурсів $R_1, R_2, \dots, R_m$ (сировина, знаряддя, праця, тощо).

Відомо: b_i – запаси сировини R_i ($i = \overline{1, m}$);

a_{ij} – кількість одиниць ресурсу R_i , що використовується для виробництва одиниці продукції Π_j ($i = \overline{1, m}, j = \overline{1, n}$);

c_j – прибуток від реалізації одиниці продукції Π_j ($j = \overline{1, n}$).

Потрібно так спланувати виробництво продукції, щоб загальний прибуток від її реалізації був найбільшим.

Складемо математичну модель задачі. Кількість одиниць продукції Π_j , запланованої до випуску, позначимо через $x_j \geq 0$. Дані задачі представимо у вигляді таблиці 4.1.

Таблиця 4.1

Види ресурсів	Види продукції						Запаси сировини
	Π_1	Π_2	$\dots$	Π_j	$\dots$	Π_n	
R_1	a_{11}	a_{12}	$\dots$	a_{1j}	$\dots$	a_{1n}	b_1
R_2	a_{21}	a_{22}	$\dots$	a_{2j}	$\dots$	a_{2n}	b_2
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
R_i	a_{i1}	a_{i2}	$\dots$	a_{ij}	$\dots$	a_{in}	b_i
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
R_m	a_{m1}	a_{m2}	$\dots$	a_{mj}	$\dots$	a_{mn}	b_m
Прибуток від од. продукції	c_1	c_2	$\dots$	c_j	$\dots$	c_n	
План випуску	x_1	x_2	$\dots$	x_j	$\dots$	x_n	

Витрати ресурсу R_i на виробництво x_j одиниць продукції Π_j становлять $a_{ij}x_j$ одиниць, а план $\mathbf{x} = (x_1, x_2, \dots, x_n)$ випуску продукції повинен бути таким, щоб загальні витрати цього ресурсу не перевищували його запасу:

$$\sum_{j=1}^n a_{ij}x_j \leq b_i \quad (i = \overline{1, m}).$$

Прибуток, одержаний від реалізації вироблених x_j одиниць продукції Π_j , становить c_jx_j , а загальний прибуток виробництва становитиме:

$$F(\mathbf{x}) = \sum_{j=1}^n c_jx_j. \quad (4.4)$$

Таким чином, необхідно визначити при яких значеннях невідомих $\mathbf{x} = (x_1, x_2, \dots, x_n)$ функція (4.4) набуває максимуму за умов:

$$\sum_{j=1}^n a_{ij}x_j \leq b_i, \quad i = \overline{1, m};$$

$$x_j \geq 0, \quad j = \overline{1, n} \bullet \quad (4.5)$$

Задача 4.2. (про перевезення). У пунктах постачання $A_1, A_2, \dots, A_m$ міститься однорідна продукція, яку треба перевезти у пункти споживання $B_1, B_2, \dots, B_n$. Відомі запаси a_i продукції у кожному пункті відправлення A_i ($i = \overline{1, m}$), попит b_j кожного споживача B_j ($j = \overline{1, n}$) і вартість c_{ij} перевезення продукції із пункту A_i в пункт B_j . Потрібно скласти такий план перевезення продукції з пунктів постачання в пункти споживання, при якому буде перевезена вся продукція, задоволені потреби кожного споживача і загальна вартість усіх перевезень буде мінімальною.

Нехай x_{ij} – кількість одиниць продукції, що планується перевезти з пункту постачання A_i в пункт споживання B_j . Невідомі x_{ij} будемо розглядати як компоненти матриці перевезень $X_{m \times n} = (x_{ij})$. Вартість перевезення x_{ij} одиниць продукції становить $c_{ij} x_{ij}$. Тоді загальна вартість усіх перевезень обчислюється за формулою:

$$F = \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} \bullet \quad (4.6)$$

Згідно умови задачі:

а) вся продукція повинна бути вивезеною, тобто:

$$\sum_{j=1}^n x_{ij} = a_i, \quad i = \overline{1, m}; \quad (4.7)$$

б) забезпечений весь попит, тобто:

$$\sum_{i=1}^m x_{ij} = b_j, \quad j = \overline{1, n}, \quad (4.8)$$

причому

$$x_{ij} \geq 0. \quad (4.9)$$

Отже, маємо математичну модель *транспортної задачі*: знайти мінімум цільової функції (4.6) за умов (4.7) – (4.9) •

4.3. Різні форми запису задач лінійного програмування

Розглянуті приклади показують, що різні моделі задач лінійного програмування можуть мати відмінності: в одних задачах шукається максимум цільової функції, в інших – мінімум; в одних випадках шукані змінні залежать від одного індексу, в інших — від двох; обмеження на змінні можуть задаватись рівняннями або нерівностями. Крім того, в деяких задачах вимога невід'ємності поширюється не на всі змінні. Наведені відмінності мають формальний характер. Зокрема, немає необхідності виділяти задачі максимізації і мінімізації цільової функції, оскільки $\min F(\mathbf{x}) = -\max (-F(\mathbf{x}))$.

В залежності від вигляду системи обмежень всі задачі лінійного програмування (ЗЛП) мають одну із трьох основних форм: стандартну, канонічну і загальну.

Стандартна (симетрична) задача має вигляд:

$$F = \sum_{j=1}^n c_j x_j \rightarrow \max ; \quad (4.10)$$

$$\sum_{j=1}^n a_{ij} x_j \leq b_i, \quad i = \overline{1, m}; \quad (4.11)$$

$$x_j \geq 0, \quad j = \overline{1, n}. \quad (4.12)$$

Це задача з однотипною системою обмежень-нерівностей і невід'ємними змінними.

Введемо позначення $\mathbf{c} = (c_1, c_2, \dots, c_n)$, $A_{m \times n} = (a_{ij})$, $\mathbf{x}^T = (x_1, x_2, \dots, x_n)$, $\mathbf{b}^T = (b_1, b_2, \dots, b_m)$ і запишемо задачу у матричній формі:

$$F = \mathbf{c} \cdot \mathbf{x} \rightarrow \max ; \quad (4.10')$$

$$A\mathbf{x} \leq \mathbf{b}; \quad (4.11')$$

$$\mathbf{x} \geq \mathbf{0}. \quad (4.12')$$

Позначивши через A_j j -тий стовпчик матриці A , приходимо до векторного запису цієї задачі:

$$F = \mathbf{c} \cdot \mathbf{x} \rightarrow \max ; \quad (4.10'')$$

$$A_1 x_1 + A_2 x_2 + \dots + A_n x_n \leq \mathbf{b}; \quad (4.11'')$$

$$\mathbf{x} \geq \mathbf{0}. \quad (4.12'')$$

У стандартній задачі співвідношення між числом невідомих n і числом обмежень m може бути довільним: $n > m$, $n = m$, $n < m$.

Канонічною (основною) ЗЛП називається задача, в якій всі змінні невід'ємні і обмеження мають форму рівнянь:

$$F = \mathbf{c} \cdot \mathbf{x} \rightarrow \max ; A\mathbf{x} \leq \mathbf{b}; \mathbf{x} \geq \mathbf{0} \quad (4.13)$$

У канонічній задачі число невідомих завжди більше числа рівнянь: $n > m$.

Загальною ЗЛП (4.1) – (4.3) називається задача, система обмежень якої містить як рівняння, так і нерівності. Крім того, вимога невід'ємності може не поширюватись на всі змінні.

Вектор $\mathbf{x} = (x_1, x_2, \dots, x_n)$, компоненти якого задовольняють обмеженням загальної ЗЛП, називається *допустимим розв'язком* (або *планом*).

План $\mathbf{x}^* = (x_1^*, x_2^*, \dots, x_n^*)$, при якому цільова функція приймає своє максимальне (мінімальне) значення, називається *оптимальним* або *розв'язком* ЗЛП.

Всі три форми ЗЛП еквівалентні між собою: загальна задача легко зводиться до стандартної або канонічної, а останні можуть бути перетворені одна в другу.

Нехай канонічна задача має m обмежень-рівнянь, серед яких може бути r лінійно незалежних ($r \leq m < n$). Значить, систему r рівнянь з n змінними $x_1, x_2, \dots, x_n$ можна розв'язати відносно r базисних змінних, виразивши їх через решту $n - r$ вільних змінних. Оскільки всі змінні, в тому числі і базисні, невід'ємні, то, "опускаючи" базисні змінні, переходимо від рівнянь до нерівностей. При цьому базисні змінні виводяться також із цільової функції. При такому способі переходу від канонічної до стандартної задачі кількість змінних зменшується до $n - r$.

Нехай система обмежень стандартної задачі має m нерівностей з n змінними. Для переходу до канонічної задачі вводяться m додаткових невід'ємних (балансових) змінних x_{n+i} , $i = \overline{1, m}$. При цьому кожна нерівність $\sum_{j=1}^n a_{ij}x_j \leq b_i$ замінюється рівнянням $\sum_{j=1}^n a_{ij}x_j + x_{n+i} = b_i$, $i = \overline{1, m}$. У результаті одержимо розширену задачу із $n + m$ змінними.

Якщо на змінну, наприклад x_k , не накладається умова невід'ємності, то для зведення задачі до канонічної форми потрібно зробити заміну

$$x_k = x'_k - x''_k,$$

де змінні $x'_k \geq 0$, $x''_k \geq 0$.

Отже, будь-яку ЗЛП можна звести до канонічної форми.

Приклад.4.1 Звести до канонічної форми задачу лінійного програмування:

$$F = -2x_1 + x_2 \rightarrow \min ;$$

$$\begin{cases} 2x_1 + 3x_2 \geq 6; \\ x_1 - x_2 \leq 5; x_1 \geq 0. \end{cases}$$

Розв'язання. В даній задачі обмеження мають вигляд нерівностей, шукається мінімум цільової функції і відсутня вимога невід'ємності змінної x_2 . Для зведення до канонічної форми (7.15) замінимо функцію F на $F_1 = -F$, введемо в обмеженнях-нерівностях балансові змінні $x_3 \geq 0$, $x_4 \geq 0$ і зробимо заміну $x_2 = x'_2 - x''_2$, $x'_2 \geq 0$, $x''_2 \geq 0$; одержимо задачу

$$F_1 = 2x_1 - x'_2 + x''_2 \rightarrow \max ;$$

$$\begin{cases} 2x_1 + 3x'_2 - 3x''_2 - x_3 = 6; \\ x_1 - x'_2 + x''_2 + x_4 = 5; \end{cases}$$

$$x_1 \geq 0, x'_2 \geq 0, x''_2 \geq 0, x_3 \geq 0, x_4 \geq 0.$$

4.4. Основні властивості розв'язків задач лінійного програмування.

Задачі лінійного програмування належать до задач на умовний екстремум.

Проте для дослідження лінійної функції $F(\mathbf{x}) = \sum_{j=1}^n c_j x_j$ багатьох змінних на умовний екстремум не можна застосувати добре відомі методи математичного аналізу, оскільки частинні похідні $\frac{\partial F}{\partial x_j} = c_j \neq 0$ ($j = \overline{1, n}$). Це означає, що всередині області допустимих розв'язків екстремальних точок не існує. Вони можуть належати лише межі допустимої області D , утвореної системою лінійних обмежень. Для розв'язування таких задач розроблені спеціальні методи.

Властивості розв'язків задач лінійного програмування тісно пов'язані з властивостями опуклих множин. Розглянемо вектори $\mathbf{x}^1, \mathbf{x}^2, \dots, \mathbf{x}^s \in R^n$. Точку $\mathbf{x} \in R^n$ називають *опуклою лінійною комбінацією* цих точок, якщо її можна представити у вигляді:

$$\mathbf{x} = \alpha_1 \mathbf{x}^1 + \alpha_2 \mathbf{x}^2 + \dots + \alpha_s \mathbf{x}^s \quad (4.14)$$

де $\alpha_i \geq 0$ ($i = \overline{1, s}$), $\alpha_1 + \alpha_2 + \dots + \alpha_s = 1$.

Множина X називається *опуклою*, якщо вона разом із довільними своїми точками $\mathbf{x}^1 \in X$, $\mathbf{x}^2 \in X$ містить і опуклу лінійну комбінацію цих точок, тобто

$$\mathbf{x} = \alpha \mathbf{x}^1 + (1 - \alpha) \mathbf{x}^2, \quad (4.15)$$

де $0 \leq \alpha \leq 1$.

З'ясуємо геометричний зміст цього означення. Якщо $\mathbf{x}$, $\mathbf{x}^1$, $\mathbf{x}^2 \in \mathbb{R}^n$, то рівняння (4.15) визначає пряму, якій належить відрізок з кінцями $\mathbf{x}^1$ та $\mathbf{x}^2$. Множина X називається опуклою, якщо вона разом із своїми довільними точками $\mathbf{x}^1$, $\mathbf{x}^2 \in X$ містить у собі і відрізок, який їх з'єднує.

Прикладами опуклих множин є відрізок, пряма, півплощина, круг, куля, куб та ін.

Точка опуклої множини X називається *крайньою* (або *кутовою*), якщо її неможливо представити у вигляді опуклої лінійної комбінації будь-яких двох різних точок цієї множини. Наприклад, крайні точки трикутника – це його вершини, крайні точки круга – це точки кола, яке його обмежує.

Опукла замкнена обмежена множина на площині із скінченим числом кутових точок називається *опуклим многокутником*.

Розглянемо задачу лінійного програмування, записану у векторному вигляді:

$$F = \mathbf{c} \cdot \mathbf{x} \rightarrow \max; \quad (4.16)$$

$$A_1 x_1 + A_2 x_2 + \dots + A_n x_n = \mathbf{b}; \quad (4.17)$$

$$\mathbf{x} \geq 0. \quad (4.18)$$

Із цього запису видно, що вектор $\mathbf{x} \in \mathbb{R}^n$ буде планом задачі (4.16) – (4.18) тільки тоді, коли вектор $\mathbf{b} \in \mathbb{R}^m$ буде невід'ємною лінійною комбінацією векторів $A_j \in \mathbb{R}^m$ з коефіцієнтами $x_j \geq 0$.

План $\mathbf{x} = (x_1, x_2, \dots, x_n)$, задачі (4.16) – (4.18) називається *опорним*, якщо вектори A_j , які відповідають додатним компонентам x_j , утворюють лінійно незалежну систему.

Із означення опорного плану випливає, що число його додатних компонент не може перевищувати m – числа обмежень задачі. Дійсно, вектори $A_j \in \mathbb{R}^m$, а в m -вимірному просторі максимальне число лінійно незалежних векторів рівне m .

Опорний план називається *невиродженим*, якщо він містить рівно m додатних компонент, і *виродженим* – якщо він містить менше за m додатних компонент.

Приклад 4.2. Задача лінійного програмування має вигляд:

$$F = x_1 - 2x_2 + 4x_3 \rightarrow \max;$$

$$\begin{cases} 3x_1 + x_2 + 2x_3 = 2; \\ x_1 - 2x_3 = 0, \end{cases}$$

$$x_1, x_2, x_3 \geq 0.$$

Встановити, чи є план $\mathbf{x} = (0,5; 0; 0,25)$ опорним.

Розв'язання. Вектор $\mathbf{x} = (0,5; 0; 0,25)$ – план задачі, оскільки його компоненти задовольняють систему обмежень задачі. Додатним компонентам $x_1 = 0,5$, $x_3 = 0,25$ відповідають вектори-стовпці $A_1 = \begin{pmatrix} 3 \\ 1 \end{pmatrix}$, $A_3 = \begin{pmatrix} 2 \\ -2 \end{pmatrix}$. Обчислимо визначник, складений із компонент цих векторів:

$$\Delta = \begin{vmatrix} 3 & 2 \\ 1 & -2 \end{vmatrix} = -6 - 2 = -8 \neq 0.$$

Оскільки $\Delta \neq 0$, то вектори A_1 , A_3 лінійно незалежні і план $\mathbf{x} = (0,5; 0; 0,25)$ є опорним. Число його додатних компонент становить $m = 2$ і співпадає з числом рівнянь системи обмежень задачі, тому маємо не вироджений опорний план.

Перепишемо задачу (4.16) – (4.18) у матричному вигляді:

$$F = \mathbf{c} \cdot \mathbf{x} \rightarrow \max; \quad (4.19)$$

$$A\mathbf{x} = \mathbf{b}; \quad (4.20)$$

$$\mathbf{x} \geq 0. \quad (4.21)$$

Основні властивості розв'язків цієї задачі сформулюємо у наступних теоремах.

Теорема 4.1. Множина усіх планів задачі (4.19) – (4.21) – опукла.

Доведення. Нехай $\mathbf{x}^1$ і $\mathbf{x}^2$ – плани задачі (4.19) – (4.21), тобто справедливі рівняння

$$A\mathbf{x}^1 = \mathbf{b}, A\mathbf{x}^2 = \mathbf{b},$$

де $\mathbf{x}^1 \geq \mathbf{0}$, $\mathbf{x}^2 \geq \mathbf{0}$. Візьмемо два числа $\alpha_1 \geq 0$, $\alpha_2 \geq 0$ таких, що $\alpha_1 + \alpha_2 = 1$.

Помножимо перше рівняння на α_1 , друге на α_2 і додамо; одержимо рівність

$$A(\alpha_1 \mathbf{x}^1 + \alpha_2 \mathbf{x}^2) = \mathbf{b},$$

тобто опукла лінійна комбінація $\mathbf{x} = \alpha_1 \mathbf{x}^1 + \alpha_2 \mathbf{x}^2$ задовольняє систему (4.20).

Оскільки $\mathbf{x}^1 \geq \mathbf{0}$, $\mathbf{x}^2 \geq \mathbf{0}$, $\alpha_1 \geq 0$, $\alpha_2 \geq 0$, то $\mathbf{x} \geq \mathbf{0}$, тобто виконується і умова (4.21). Отже, опукла лінійна комбінація $\mathbf{x}$ – план задачі лінійного програмування.

Опуклу множину планів задачі лінійного програмування позначимо через D .

Теорема 4.2. Цільова функція $F = \mathbf{c} \cdot \mathbf{x}$ задачі лінійного програмування (4.19) – (4.21) досягає максимального значення у крайній точці опуклої множини D планів задачі. Якщо цільова функція має максимум більше ніж в одній крайній точці, то вона набуває цього ж значення у будь-якій точці, яка є опуклою лінійною комбінацією цих точок.

Теорема 4.3. Вектор $\mathbf{x}$ є опорним планом задачі (4.16) – (4.18) тоді і тільки тоді, коли $\mathbf{x}$ – вершина многогранника D допустимих планів.

Із цієї теореми випливає, що множина опорних планів задачі лінійного програмування співпадає з множиною крайніх точок або вершин многогранника D , який визначається обмеженнями задачі.

Наслідок 4.1. Кожна вершина многогранника розв'язків має не більше ніж m додатних координат.

Наслідок 4.2. Кожній вершині многогранника розв'язків D відповідає $k \leq m$ лінійно незалежних векторів із системи $A_1, A_2, \dots, A_n$.

Отже, якщо цільова функція задачі лінійного програмування обмежена на опуклій множині планів, то існує крайня точка (опорний план) цієї множини, в якій цільова функція набуває максимуму. Тому для розв'язання задачі лінійного програмування необхідно дослідити тільки вершини многогранника розв'язків – опорні плани задачі.

4.5. Геометрична інтерпретація і графічний метод розв'язування задачі лінійного програмування.

Розглянемо задачу лінійного програмування, записану у стандартній формі (4.10) – (4.12). Для наочності обмежимося випадком $n = 2$, тобто задача містить дві невідомі x_1 і x_2 . Маємо таку задачу лінійного програмування:

$$F = c_1x_1 + c_2x_2 \rightarrow \max(\min); \quad (4.22)$$

$$a_{i1}x_1 + a_{i2}x_2 \leq b_i, \quad i = \overline{1, m}; \quad (4.23)$$

$$x_1 \geq 0, \quad x_2 \geq 0. \quad (4.24)$$

Кожна нерівність (4.23), (4.24) системи обмежень на площині Ox_1x_2 геометрично визначає півплощину із граничними прямими відповідно $a_{i1}x_1 + a_{i2}x_2 = b_i$ ($i = \overline{1, m}$) та $x_j = 0$ ($j = \overline{1, 2}$). Перетин вказаних півплощин утворює опуклу область D – многокутник розв'язків.

Таким чином, геометрично задача лінійного програмування полягає в знаходженні такої вершини многокутника розв'язків, в якій функція $F(x_1, x_2)$ набуває найбільшого (найменшого) значення. Така вершина існує тоді, коли многокутник розв'язків не порожній і на ньому цільова функція обмежена зверху.

Перейдемо до геометричної інтерпретації цільової функції (4.22). Рівняння $F = c_1x_1 + c_2x_2$ при фіксованому значенні $F = h$ (h – довільна стала) визначає на площині пряму лінію l : $c_1x_1 + c_2x_2 = h$ – лінію рівня. Вектор $\mathbf{n} = (c_1, c_2)$ ($-\mathbf{n}$) перпендикулярний до лінії рівня l і вказує напрямом зростання (спадання) цільової функції. На геометричній інтерпретації задач лінійного програмування ґрунтується *графічний метод* відшукування оптимальних значень цільової функції.

1. За обмеженнями (4.23), (4.24) будуються граничні прямі.
2. Для кожного із обмежень (4.23), (4.24) визначають півплощину і знаходять многокутник розв'язків D .
3. Будують вектор $\mathbf{n} = (c_1, c_2)$ і лінію рівня l : $c_1x_1 + c_2x_2 = h$, яка перетинає область D .
4. Зміщують лінію рівня паралельно самій собі у напрямку вектора $\mathbf{n}$ ($-\mathbf{n}$) до тих пір, поки пряма l не досягне крайньої точки (вершини) області D , в якій цільова функція F набуває максимального (мінімального) значення.

5. Визначають координати вершини екстремуму, як координати точки перетину відповідних прямих і за формулою (4.22) обчислюють $F_{\max}$ ($F_{\min}$).

При розв'язуванні задачі (4.22) – (4.24) графічним методом можуть зустрітись випадки:

а) цільова функція набуває максимального значення в єдиній точці B і $F_{\max} = F(B)$ (рис. 4.1,а);

б) функція (4.22) набуває максимального значення в кожній точці відрізка AB (Рис. 4.1,б);

в) функція необмежена зверху в області D , тобто $F_{\max} = \infty$ (рис. 4.1,в);

г) задача не має розв'язку (система обмежень (4.23), (4.24) несумісна) (рис. 4.1,г).

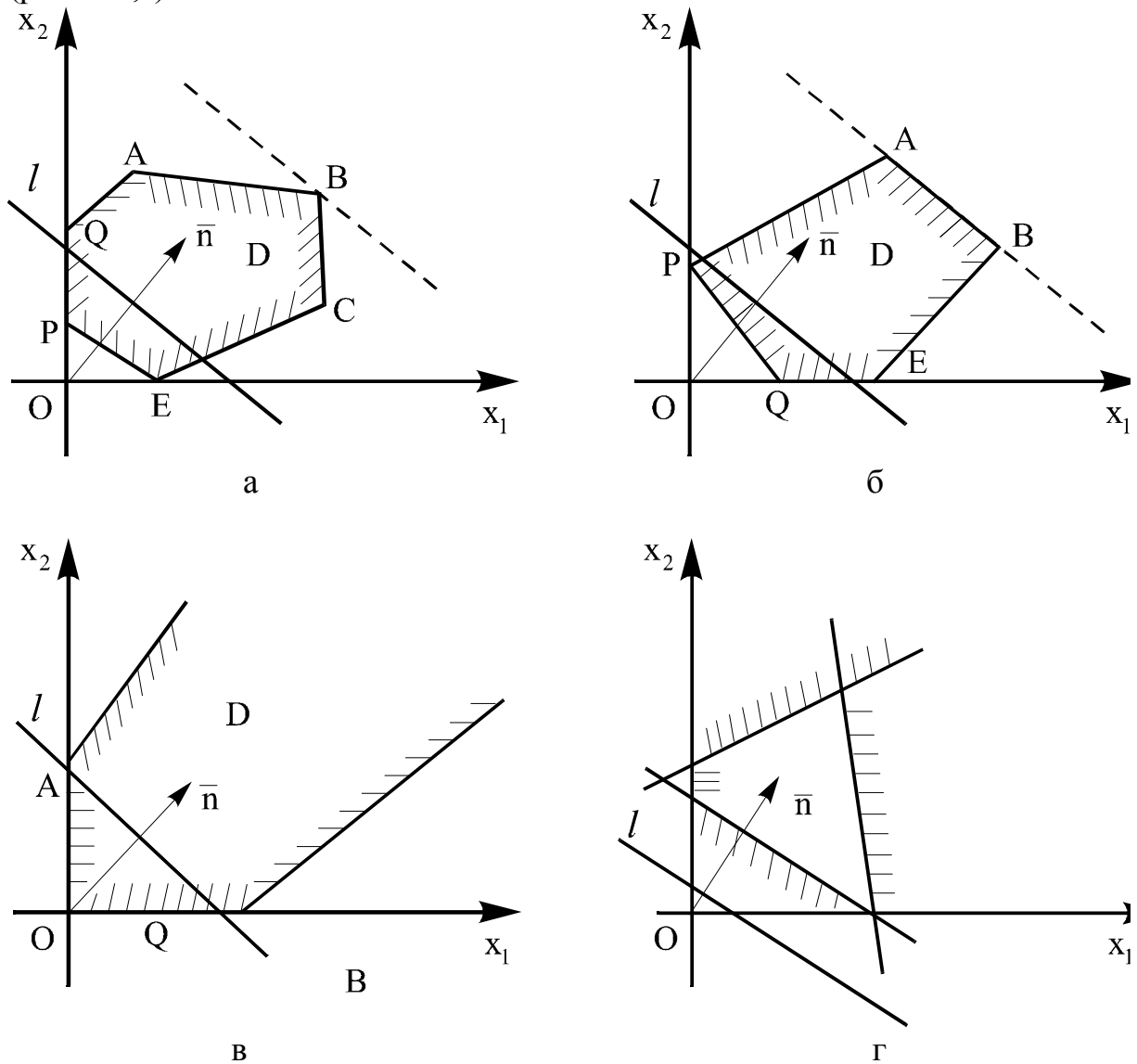


Рис. 4.1

Приклад 4.3 Розв'язати графічним методом ЗЛП:

$$F = 2x_1 + 3x_2 \rightarrow \max(\min);$$

$$\begin{cases} x_1 + x_2 \leq 6; \\ x_1 + 4x_2 \geq 4; \\ 2x_1 - x_2 \geq 0; \end{cases}$$

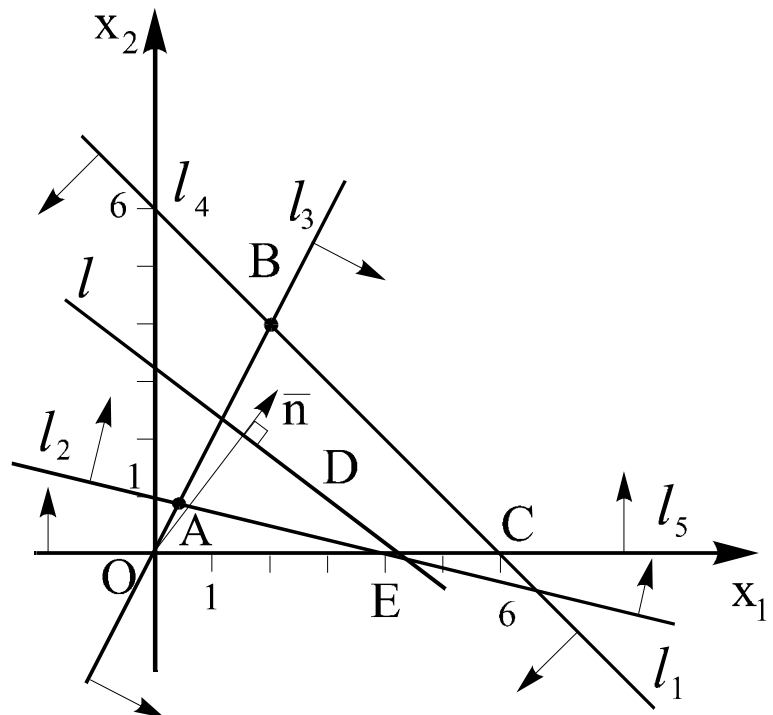


Рис. 4.2

$$x_1 \geq 0; x_2 \geq 0.$$

Розв'язання. На площині Ox_1x_2 будемо прями

$$\begin{aligned} l_1 : x_1 + x_2 &= 6; \\ l_2 : x_1 + 4x_2 &= 4; \\ l_3 : 2x_1 - x_2 &= 0; \\ l_4 : x_1 &= 0; \\ l_5 : x_2 &= 0. \end{aligned}$$

Знаходимо півплощини, в яких виконуються нерівності системи обмежень задачі. Для цього береться довільна пробна точка, яка не належить прямій, і її координати підставляються у відповідну нерівність. Якщо вибрана точка задовольняє нерівність, то нерівність буде виконуватись у півплощині, яка містить пробну точку. В іншому випадку береться півплощина, яка не містить пробної точки. В перетині усіх півплощин (рис. 4.2) є область допустимих розв'язків D – чотирикутник $ABCE$.

Будемо вектор $\mathbf{n} = (2; 3)$. Перетнемо область D прямою $l: 2x_1 + 3x_2 = h$, яка перпендикулярна до вектора $\mathbf{n}$. Цей вектор вказує напрям зростання цільової функції. Паралельним зміщенням лінії рівня l визначаємо крайні точки A, B многокутника, в яких функція F досягає відповідно мінімуму та максимуму. Оскільки точка A лежить на перетині прямих l_2 і l_3 , то її координати визначаються із системи рівнянь:

$$\begin{cases} x_1 + 4x_2 = 4; \\ 2x_1 - x_2 = 0 \end{cases} \Rightarrow A\left(\frac{4}{9}; \frac{8}{9}\right).$$

Координати точки B знаходимо із системи:

$$\begin{cases} x_1 + x_2 = 6; \\ 2x_1 - x_2 = 0 \end{cases} \Rightarrow B(2; 4).$$

Підставивши координати цих точок у цільову функцію, маємо:

$$F_{\max} = F(B) = 2 \cdot 2 + 3 \cdot 4 = 16,$$

$$F_{\min} = F(A) = 2 \cdot \frac{4}{9} + 3 \cdot \frac{8}{9} = \frac{32}{9}.$$

Зауваження. Графічним методом можна розв'язати задачу лінійного програмування, записану в канонічній формі, якщо виконується умова $n - r = 2$, де n – число невідомих, а r – ранг матриці, складеної із коефіцієнтів системи обмежень задачі.

Завдання для самоконтролю

1. Сформулювати означення задач математичного програмування, лінійного програмування.
2. Що називається математичною моделлю економічної задачі і як будується така модель?
3. Назвіть еквівалентні форми задач лінійного програмування.
4. Вкажіть перетворення, які необхідно виконати для переходу від однієї форми ЗЛП до іншої.
5. Яка множина називається опуклою?
6. Сформулюйте означення плану, виродженого і неvirодженого опорного плану, оптимального плану.
7. Що називається областю допустимих розв'язків ЗЛП?
8. В якій точці області цільова функція ЗЛП досягає свого оптимального значення?
9. На чому ґрунтується графічний метод розв'язування ЗЛП?
10. Ферма проводить відгодівлю худоби з комерційною метою. Нехай у наявності два види корму: K_1 і K_2 , вартістю відповідно 0,15 та 0,25 грн. за 1 кг. Кожна тварина щоденно повинна одержати не менше 9 од. вуглеводів, не менше 8 од. жирів і не менше 12 од. білків. Для кормів K_1 , K_2 відомий вміст білків, жирів і вуглеводів (в одиницях на 1 кг корму) і наведено в таблиці 4.2.

Таблиця 4.2

Поживні речовини	Кількість од. поживних речовин в 1 кг корму	
	K_1	K_2
вуглеводи	2	1,5
жири	0,5	1
білок	1	0,8

Потрібно скласти добовий раціон, що забезпечує одержання необхідної кількості поживних речовин при мінімальній вартості.

(Відповідь. $F = 0,15x_1 + 0,25x_2 \rightarrow \min$,

$$\begin{cases} 2x_1 + 1,5x_2 \geq 9; \\ 0,5x_1 + x_2 \geq 8; \\ x_1 + 0,8x_2 \geq 12. \end{cases} \quad x_1 \geq 0, x_2 \geq 0.)$$

11. Виконати замовлення по виробництву 32 виробів B_1 та 4 виробів B_2 узялися бригади B_1 і B_2 . Продуктивність бригади B_1 по виробництву виробів B_1 і B_2 складає відповідно 4 і 2 виробу за одиницю часу, фонд робочого часу цієї бригади 9,5 од. Продуктивність бригади B_2 – 1 і 3, а фонд її робочого часу – 4 од. Витрати пов'язані з виробництвом одиниці виробу для бригади B_1 дорівнюють відповідно 9 і 20 тис. грн, для бригади B_2 – 15 і 30 грн.

Скласти математичну модель задачі по показнику витрат на виконання замовлення, якщо фонд робочого часу для бригади B_2 має бути повністю використаним.

$$(Відповідь. F = 9x_{11} + 20x_{12} + 15x_{21} + 30x_{22}; \begin{cases} x_{11} + x_{21} = 32; \\ x_{12} + x_{22} = 4; \\ \frac{x_{11}}{4} + \frac{x_{12}}{2} \leq 9,5; \\ \frac{x_{21}}{1} + \frac{x_{22}}{3} = 4; \end{cases}$$

$$x_{ij} \geq 0 \quad (i = 1, 2; j = 1, 2) .)$$

12. Звести до канонічної форми ЗЛП.

$$F = x_1 - x_2 - 2x_3 \rightarrow \min ;$$

$$\begin{cases} x_1 - 2x_2 + x_3 \geq 4, \\ -3x_1 + x_2 + x_3 \leq 9, \\ 2x_1 + 3x_2 + 3x_3 = 10, \end{cases}$$

$$x_1 \geq 0, \quad x_3 \geq 0 .$$

$$(Відповідь. F = x_1 - x'_2 + x''_2 - 2x_3 \rightarrow \min ;$$

$$\begin{cases} x_1 - 2x'_2 + 2x''_2 + x_3 - x_4 = 4, \\ -3x_1 + x'_2 - x''_2 + x_3 + x_5 = 9, \\ 2x_1 + 3x'_2 - 3x''_2 + 3x_3 = 10, \end{cases}$$

$$x_1 \geq 0, \quad x'_2 \geq 0, \quad x''_2 \geq 0, \quad x_3 \geq 0, \quad x_4 \geq 0, \quad x_5 \geq 0 .)$$

13. Звести до стандартної форми ЗЛП.

$$F = 2x_1 + 3x_2 - x_4 + 2x_5 + 6 \rightarrow \max ; \begin{cases} 3x_1 + 9x_2 + 5x_3 - 2x_4 - 2x_5 = 7, \\ 2x_1 - 2x_2 - 2x_3 + x_4 - x_5 = 1, \\ 9x_1 - 8x_2 - 7x_3 + 3x_4 - 4x_5 = 3, \end{cases}$$

$$x_j \geq 0 (j = \overline{1, 5}) .$$

$$(Відповідь. F = 4x_1 + \frac{34}{5}x_2 - \frac{2}{5} \rightarrow \max ;$$

$$\begin{cases} -x_1 + \frac{13}{5}x_2 \leq \frac{9}{5}, \\ -2x_1 + \frac{13}{5}x_2 \leq \frac{14}{5}, \\ -2x_1 - \frac{3}{5}x_2 \leq -\frac{9}{5}, \end{cases} \quad x_1 \geq 0, \quad x_2 \geq 0 .)$$

14. Для збереження нормальної життєдіяльності людина щодоби повинна споживати білків не менше 120 умовних одиниць, жирів – не менше 70 і вітамінів – не менше 10 умовних одиниць. Вміст їх у продуктах Π_1 і Π_2 дорівнює відповідно $(0,2; 0,075; 0)$ і $(0,1; 0,1; 0,1)$. Вартість 1 од. продукту Π_1 – 2 грош. од., Π_2 – 3 грош. од. Потрібно так організувати харчування щоб його вартість була мінімальною, а організм отирав необхідну кількість поживних речовин.

(Відповідь. $x^* = (800; 100)$, $F_{opt} = 2 \cdot 800 + 3 \cdot 100 = 1900$.)

Лекція № 5. Симплекс-метод. Метод штучного базису.

Симплексний метод розв'язування задач лінійного програмування.

Алгоритм симплекс-методу.

Вибір початкового допустимого базисного розв'язку.

M-метод.

5.1. Симплексний метод розв'язування задач лінійного програмування.

Основним алгебраїчним методом розв'язування задач лінійного програмування є *симплексний метод*, запропонований у 1949 році американським математиком Д. Данцігом. Цей метод застосовується тоді, коли задача лінійного програмування записана у канонічній формі (4.13). Відомо, що оптимальний розв'язок задачі лінійного програмування знаходиться в одній із крайніх точок многогранника розв'язків. В основі симплекс-методу лежить впорядкований перехід від однієї вершини (опорного плану) до іншої, в якій значення цільової функції зростає. Це досягається шляхом зміни базису. Виходячи із будь-якого початкового опорного плану, за скінченне число кроків (ітерацій) симплекс-метод приводить до оптимального розв'язку або встановлює відсутність такого розв'язку.

Розглянемо канонічну задачу лінійного програмування:

$$F = \mathbf{c} \cdot \mathbf{x} \rightarrow \max; A\mathbf{x} = \mathbf{b}; \mathbf{x} \geq 0, \quad (5.1)$$

де $A_{m \times n} = (a_{ij})$, $\mathbf{c}, \mathbf{x} \in R^n$, $\mathbf{b} \in R^m$ і вважаємо $\mathbf{b} \geq 0$.

Припустимо, що система рівнянь $A\mathbf{x} = \mathbf{b}$ зведена до еквівалентної базисної форми:

$$\sum_{j=1}^n x_j \mathbf{A}_j = \mathbf{b}, \quad (5.2)$$

$$\text{де } \mathbf{A}_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \mathbf{A}_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \dots, \mathbf{A}_m = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix}, \mathbf{A}_{m+1} = \begin{pmatrix} a_{1m+1} \\ a_{2m+1} \\ \vdots \\ a_{mm+1} \end{pmatrix}, \dots$$

Вектори $\mathbf{A}_1, \mathbf{A}_2, \dots, \mathbf{A}_m$ утворюють базис m -вимірного простору. Тому в розкладі (5.2) невідомі $x_1, x_2, \dots, x_m$ – базисні, а $x_{m+1}, \dots, x_n$ – вільні. Загальний розв'язок системи (9.2) має вигляд:

$$x_i = b_i - \sum_{j=m+1}^n a_{ij} x_j, i = \overline{1, m}. \quad (5.3)$$

Звідси, прирівнявши вільні змінні до нуля і враховуючи, що $\mathbf{b} \geq 0$, приходимо до початкового опорного плану:

$$\mathbf{x}^0 = (b_1, b_2, \dots, b_m, 0, \dots, 0). \quad (5.4)$$

Враховуючи рівняння (9.3), вилучимо із цільової функції $F = \mathbf{c} \cdot \mathbf{x}$ базисні змінні; маємо

$$F = \sum_{i=1}^n c_i x_i = \sum_{i=1}^m c_i \left(b_i - \sum_{j=m+1}^n a_{ij} x_j \right) + \sum_{j=m+1}^n c_j x_j.$$

Позначимо:

$$F_0 = \sum_{i=1}^m c_i b_i, \quad (5.5)$$

$$z_j = \sum_{i=1}^m c_i a_{ij}, j = 1, \dots, n; \quad (5.6)$$

запишемо цільову функцію у вигляді

$$F = F_0 - \sum_{j=m+1}^n (z_j - c_j) x_j$$

або

$$F = F_0 - \sum_{j=m+1}^n \Delta_j x_j, \quad (5.7)$$

де

$$\Delta_j = z_j - c_j, j = \overline{1, n}. \quad (5.8)$$

Коефіцієнти Δ_j називаються *оцінками* відповідних змінних x_j . Зауважимо, що для базисних змінних із співвідношень (5.6) маємо: $z_j = c_j$, звідки $\Delta_j = 0, j = \overline{1, m}$.

Дослідимо зв'язок між вільними змінними і величиною цільової функції. Можливі три випадки.

1. Всі коефіцієнти Δ_j при вільних змінних x_j функції (5.7) невід'ємні: $\Delta_j \geq 0, j = m+1, \dots, n$. Тоді будь-яке збільшення вільних змінних $x_{m+1}, \dots, x_n$, які в базисному розв'язку рівні нулю, веде до зменшення функції (5.7). Отже, опорний план (5.4) є оптимальним і $F_{\max} = F(\mathbf{x}^0) = F_0$.

2. Серед оцінок вільних змінних є хоча б одна від'ємна, наприклад $\Delta_k < 0$, і при цьому серед коефіцієнтів відповідного їй стовпця $\mathbf{A}_k$ базисної системи немає додатних. В цьому випадку функція $F(\mathbf{x})$ необмежена в області допустимих розв'язків.

Дійсно, якщо $\Delta_k < 0$ і всі $a_{ik} \leq 0, i = \overline{1, m}$, то надаючи у рівняннях (5.3) і (5.7) всім вільним змінним, крім x_k , нульових значень, маємо:

$$x_i = b_i - a_{ik} x_k, i = \overline{1, m}; \quad (5.9)$$

$$F = F_0 - \Delta_k x_k. \quad (5.10)$$

Оскільки $\Delta_k < 0$, то при збільшенні x_k функція (5.10) зростатиме. При цьому, оскільки $a_{ik} \leq 0$, із рівнянь (5.9) видно, що всі змінні x_i лишатимуться невід'ємними при будь-яких як завгодно великих значеннях x_k . Це означає, що функція $F(\mathbf{x})$ максимуму не має, оптимального плану не існує, хоча допустимих планів нескінченно багато.

3. Існує вільна змінна x_k з оцінкою $\Delta_k < 0$ і є коефіцієнт $a_{is} > 0$. Тоді можна знайти новий опорний план, якому відповідає більше значення цільової функції.

Дійсно, як видно із (5.10) при $\Delta_k < 0$ збільшення x_k веде до зростання функції $F(x)$. Із (9.9) видно, що збільшувати x_k можна доти, поки

$$b_i - a_{ik}x_k \geq 0 \text{ або } x_k \leq \frac{b_i}{a_{ik}}, \quad i = \overline{1, m}.$$

Найбільше можливе значення $x_k = \theta_k$, яке задовольняє наведеним нерівностям, визначається із умови:

$$\theta_k = \min_{a_{ik} > 0} \frac{b_i}{a_{ik}} = \frac{b_r}{a_{rk}}, \quad (5.11)$$

тобто $x_k = \frac{b_r}{a_{rk}}$ і вільна змінна x_k вводиться до числа базисних замість базисної змінної x_r . Для цього за методом Жордана-Гауса з розв'язувальним елементом a_{rk} вилучимо змінну x_k із усіх рівнянь системи, крім r -го:

$$a'_{ij} = \begin{cases} a_{ij} - \frac{a_{rj}a_{ik}}{a_{rk}}, & i \neq r; \\ \frac{a_{rj}}{a_{rk}}, & i = r, \end{cases} \quad (5.12)$$

$$b'_i = \begin{cases} b_i - \frac{a_{ik}b_r}{a_{rk}}, & i \neq r; \\ \frac{b_r}{a_{rk}}, & i = r, \end{cases} \quad (5.13)$$

Зауважимо, якщо співвідношення (5.11) досягає мінімуму у кількох рядках $i = r, s, \dots$, то нульових значень набувають змінні $x_k, x_s, \dots$, а нова базисна змінна x_k вводиться замість однієї з них. Отже, одержаний в цьому випадку план буде виродженим.

Розглянемо, як зміниться цільова функція при переході до нового опорного плану. Згідно (5.10) і (5.11) її новим значенням буде:

$$F' = F - \Delta_k \frac{b_r}{a_{rk}}. \quad (5.14)$$

Оскільки $\Delta_k < 0$, $\frac{b_r}{a_{rk}} > 0$, то функція F зростає на величину $\Delta_k \frac{b_r}{a_{rk}}$.

Видно, що формула (5.14) повторює формулу (5.13) жорданових виключень із заміною коефіцієнта a_{ik} i -того рядка матриці A на коефіцієнт Δ_k приведеної цільової функції. Таким чином, рівняння (5.7) можна записати у вигляді

$$F + \sum_{j=m+1}^n \Delta_j x_j = F_0 \quad (5.15)$$

і розглянути як $(m+1)$ -ше рівняння системи з базисною змінною F і виконувати над елементами Δ_j, F_0 перетворення Жордана-Гауса.

Отже, в результаті жорданових перетворень над системою $Ax = b$, доповненою рівнянням (5.15), приходимо до нового опорного плану, який відповідає більшому значенню функції F .

Із формули (5.14) випливає, що до базисних змінних доцільно вводити ту змінну, якій відповідає *найбільше за абсолютною величиною від'ємне значення* Δ_j .

Досліджені три випадки можна сформулювати у вигляді теорем.

Теорема 5.1. (*критерій оптимальності*). Якщо для деякого базисного розв'язку

$$x^* = (x_1^*, x_2^*, \dots, x_m^*, 0, \dots, 0)$$

задачі лінійного програмування (9.1) виконуються умови

$$\Delta_j = z_j - c_j \geq 0, \quad j = \overline{1, n},$$

то x^* – оптимальний розв'язок.

Теорема 5.2. Якщо для деякого базисного розв'язку існує хоча б одна вільна змінна x_k , для якої $\Delta_k = z_k - c_k < 0$ і $a_{ik} \leq 0$ ($i = \overline{1, m}$), то цільова функція задачі лінійного програмування необмежена на множині її планів.

Теорема 5.3. Якщо опорний план x задачі лінійного програмування не вироджений і існує $\Delta_k < 0$, а серед чисел a_{ik} є додатні, то знайдеться опорний план x^* такий, що $F(x) < F(x^*)$.

Наведені теореми дають можливість перевірити на оптимальність знайдений опорний план і виявити доцільність переходу до нового базисного розв'язку.

5.2. Алгоритм симплекс-методу.

Дослідження опорного плану на оптимальність і результати обчислень доцільно представляти у вигляді *симплекс-таблиці* (таблиця 5.1).

Таблиця 5.1.

C_6	x_6	b	c_1	c_2	$\dots$	c_n
			A_1	A_2	$\dots$	A_n
c_1	x_1	b_1	1	0	$\dots$	a_{1n}
c_2	x_2	b_2	0	1	$\dots$	a_{2n}
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
c_m	x_m	b_m	0	0	$\dots$	a_{mn}
Δ		F_0	Δ_1	Δ_2	$\dots$	Δ_n

В стовпчику C_6 записані коефіцієнти цільової функції, які відповідають базисним змінним із стовпця x_6 .

Алгоритм симплекс-методу включає такі етапи.

1. Записуємо симплекс-таблицю.

2. Виконуємо обчислення за формулами (5.5), (5.6), (5.8):

$$F_0 = c_6 \cdot b, \quad z_j = c_6 \cdot A_j, \quad \Delta_j = z_j - c_j, \quad j = \overline{1, n}$$

і заповнюємо $(m + 1)$ -й рядок таблиці.

3. Проглядаємо знаки всіх коефіцієнтів Δ_j . Якщо всі $\Delta_j \geq 0$, то задача розв'язана: маємо оптимальний базисний розв'язок і $F_{\max} = F_0$. Якщо серед оцінок Δ_j є від'ємні, то переходимо до наступного пункту.

4. Серед значень $\Delta_j < 0$ знаходимо найбільше за абсолютною величиною і відмічаємо відповідний стовпчик. Нехай це буде k -тий стовпчик. Якщо в цьому стовпчику всі елементи $a_{ik} \leq 0$, то цільова функція необмежена: $F_{\max} = \infty$. Розв'язок закінчено. Якщо серед a_{ik} є додатні, то переходимо до наступного пункту.

5. Для кожного $a_{ik} > 0$ вибраного стовпця обчислюємо відношення $\frac{b_i}{a_{ik}}$; знаходимо найменше і відмічаємо відповідний рядок. Нехай це буде рядок за номером r . Елемент a_{rk} , який знаходиться на перетині відмічених рядка і стовпця буде розв'язувальним.

6. За формулами (5.12) – (5.14) виконуємо жорданові перетворення над $(m + 1)$ рядками таблиці з розв'язувальним елементом a_{rk} . Переходимо до пункту 3.

Приклад. 5.1 Розв'язати симплексним методом канонічну ЗЛП.

$$F = 4x_1 + x_2 + x_3 - 3x_4 \rightarrow \max;$$

$$\begin{cases} 2x_1 - x_2 + x_3 = 6; \\ x_1 - x_2 + x_4 = 8; \\ x_1 + x_2 + x_5 = 18, \end{cases} \quad x_j \geq 0, \quad j = \overline{1, 5}.$$

Розв'язання. Система рівнянь записана в базисній формі; тут змінні x_3, x_4, x_5 – базисні, x_1, x_2 – вільні:

$$\begin{cases} x_3 = 6 - 2x_1 + x_2; \\ x_4 = 8 - x_1 + x_2; \\ x_5 = 18 - x_1 - x_2. \end{cases}$$

Прирівнюючи вільні змінні до нуля, одержимо базисний розв'язок $\mathbf{x} = (0; 0; 6; 8; 18)$, який можна взяти за початковий опорний план.

Для перевірки цього плану складаємо симплексну таблицю 5.2. Обчислюємо

$$F_0 = \mathbf{c}_6 \cdot \mathbf{b} = 1 \cdot 6 - 3 \cdot 8 + 0 \cdot 18 = -18;$$

$$z_1 = \sum_{i=1}^3 c_i a_{i1} = \mathbf{c}_6 \cdot \mathbf{A}_1 = 1 \cdot 2 - 3 \cdot 1 + 0 \cdot 1 = -1; \quad \Delta_1 = z_1 - c_1 = -1 - 4 = -5;$$

$$z_2 = \sum_{i=1}^3 c_i a_{i2} = \mathbf{c}_6 \cdot \mathbf{A}_2 = 1 \cdot (-1) - 3 \cdot (-1) + 0 \cdot 1 = 2; \quad \Delta_2 = z_2 - c_2 = 2 - 1 = 1;$$

$$z_3 = \sum_{i=1}^3 c_i a_{i3} = \mathbf{c}_6 \cdot \mathbf{A}_3 = 1 \cdot 1 - 3 \cdot 0 + 0 \cdot 0 = 1; \quad \Delta_3 = z_3 - c_3 = 1 - 1 = 0$$

і заповнюємо перший рядок оцінок таблиці. Видно, що серед оцінок вільних змінних є від'ємна: $\Delta_1 = -5 < 0$. Це означає, що початковий опорний план можна покращити. Визначимо змінну, яку потрібно вилучити із базисних. Для цього знаходимо:

$$\theta = \min_{a_{i1} > 0} \left\{ \frac{b_i}{a_{i1}} \right\} = \min \left\{ \frac{6}{2}, \frac{8}{1}, \frac{18}{1} \right\} = \frac{6}{2} = 3,$$

тобто з базису вилучимо змінну x_3 і елемент a_{11} — розв'язувальний. Цей елемент у таблиці позначаємо *квадратом*.

Таблиця 5.2

c_6	x_6	b	4	1	1	-3	0
			A_1	A_2	A_3	A_4	A_5
1	x_3	6	2	-1	1	0	0
-3	x_4	8	1	-1	0	1	0
0	x_5	18	1	1	0	0	1
Δ		-18	-5	1	0	0	0
4	x_1	3	1	$-\frac{1}{2}$	$\frac{1}{2}$	0	0
-3	x_4	5	0	$-\frac{1}{2}$	$-\frac{1}{2}$	1	0
0	x_5	15	0	$\frac{3}{2}$	$-\frac{1}{2}$	0	1
Δ		-3	0	$-\frac{3}{2}$	$\frac{5}{2}$	0	0
4	x_1	8	1	0	$\frac{1}{3}$	0	$\frac{1}{3}$
-3	x_4	10	0	0	$-\frac{2}{3}$	1	$\frac{1}{3}$
1	x_2	10	0	1	$-\frac{1}{3}$	0	$\frac{2}{3}$
Δ		12	0	0	2	0	1

Проводимо виключення за методом Жордана-Гауса і переходимо до другої ітерації.

Знову проглядаємо одержаний рядок оцінок. Оскільки $\Delta_2 = -\frac{3}{2} < 0$, то одержаний план не оптимальний і його можна покращити, ввівши змінну x_2 до базисних замість x_5 (в третьому рядку вектора A_2 знаходиться єдиний елемент $a_{32} > 0$). Виконуємо виключення Жордана з розв'язувальним елементом $a_{32} = \frac{3}{2}$. Видно, що після другої ітерації всі коефіцієнти рядка оцінок невід'ємні: $\Delta_j \geq 0$, $j = \overline{1,5}$. Отже, одержали оптимальний план $x = (8; 10; 0; 10; 0)$, при якому $F_{\max} = 12$.

5.3. Вибір початкового допустимого базисного розв'язку.

Симплексний метод можна застосовувати до тих задач, для яких відомий початковий опорний план. Такий план вважається відомим, якщо в системі обмежень задачі лінійного програмування $Ax = b$, $b \geq 0$ матриця $A_{m \times n}$ містить одиничну підматрицю $E_{m \times m}$ або її можна виділити (наприклад, шляхом перетворень Жордана-Гауса), не порушуючи умову $b \geq 0$.

Приклад 5.2. Розв'язати симплексним методом ЗЛП

$$F = -6x_1 - x_2 + x_4 - 2x_5 \rightarrow \min$$

$$\begin{cases} 4x_1 + 2x_2 + x_3 + x_4 + x_5 = 8, \\ 2x_1 + x_2 - x_3 = 2, \\ x_1 + x_3 + x_5 = 2; \end{cases}$$

$$x_j \geq 0, j = \overline{1, 5}.$$

Розв'язання. Видно, що матриця системи рівнянь має лише один одиничний вектор A_4 . Випишемо розширену матрицю і методом жорданових виключень виділимо ще два одиничні вектори:

$$\tilde{A} = \left(\begin{array}{ccccc|c} 4 & 2 & 1 & 1 & 1 & 8 \\ 2 & 1 & -1 & 0 & 0 & 2 \\ 1 & 0 & 1 & 0 & 1 & 2 \end{array} \right) \sim \left(\begin{array}{ccccc|c} 0 & 0 & 3 & 1 & 1 & 4 \\ 2 & 1 & -1 & 0 & 0 & 2 \\ 1 & 0 & 1 & 0 & 1 & 2 \end{array} \right) \sim \left(\begin{array}{ccccc|c} -1 & 0 & 2 & 1 & 0 & 2 \\ 2 & 1 & -1 & 0 & 0 & 2 \\ 1 & 0 & 1 & 0 & 1 & 2 \end{array} \right)$$

Запишемо дану задачу в базисній формі:

$$Z = -F = 6x_1 + x_2 - x_4 + 2x_5 \rightarrow \max;$$

$$\begin{cases} -x_1 + 2x_2 + x_3 = 2, \\ 2x_1 + x_2 - x_3 = 2, \\ x_1 + x_2 + x_5 = 2; \end{cases}$$

$$x_j \geq 0, j = \overline{1, 5}.$$

Оскільки x_2, x_4, x_5 – базисні змінні і вільні члени системи рівнянь невід'ємні, то маємо початковий опорний план

$$x = (0; 2; 0; 2; 2).$$

Складаємо симплекс-таблицю (табл.10.1) і перевіряємо на оптимальність даний опорний план:

Таблиця 5.3

$\mathbf{c}_6$ $\mathbf{x}_6$ $\mathbf{b}$			6	1	0	-1	2
			A_1	A_2	A_3	A_4	A_5
-1	x_4	2	-1	0	2	1	0
1	x_2	2	2	1	-1	0	0
2	x_5	2	1	0	1	0	1
Δ		4	-1	0	-1	0	0
-1	x_4	3	0	1/2	3/2	1	0
6	x_1	1	1	1/2	-1/2	0	0
2	x_5	1	0	-1/2	3/2	0	1
Δ		5	0	1/2	-3/2	0	0
-1	x_4	2	0	1	0	1	-1
6	x_1	4	1	-1	0	0	3
0	x_3	2/3	0	-1/3	1	0	2/3
Δ		6	0	0	0	0	1

$$Z_0 = \mathbf{c}_6 \cdot \mathbf{b} = -2 + 2 + 4 = 4;$$

$$\Delta_1 = Z_1 - c_1 = \mathbf{c}_6 \cdot \mathbf{A}_1 - c_1 = 1 + 2 + 2 - 6 = -1;$$

$$\Delta_2 = Z_2 - c_2 = \mathbf{c}_6 \cdot \mathbf{A}_2 - c_2 = 0;$$

$$\Delta_3 = Z_3 - c_3 = \mathbf{c}_6 \cdot \mathbf{A}_3 - c_3 = -2 - 1 + 2 - 0 = -1;$$

$$\Delta_4 = \Delta_5 = 0.$$

Після двох ітерацій з розв'язувальними елементами $a_{21}=2$ та $a'_{33} = \frac{3}{2}$, виділеними в табл.10.1 рамками, одержуємо план $\mathbf{x} = (4; 0; \frac{2}{3}; 2; 0)$. Оскільки всі $\Delta_j \geq 0, j = \overline{1, 5}$, то цей план оптимальний і $Z_{\max} = 6$. Отже, $F_{\min} = -6$.

5.4. Метод штучного базису

Існують канонічні задачі лінійного програмування, які не приводяться до базисного вигляду або це вимагає значних обчислень. При розв'язуванні таких задач застосовують *метод штучного базису* (М – метод).

Розглянемо канонічну задачу лінійного програмування

$$F = \sum_{j=1}^n c_j x_j \rightarrow \max; \quad (5.16)$$

$$\sum_{j=1}^n a_{ij} x_j = b_i, i = \overline{1, m}; \quad (5.17)$$

$$x_j \geq 0, j = \overline{1, n}, \quad (5.18)$$

де $b_i \geq 0$ і система (5.17) не містить одиничної матриці. В кожне з рівнянь системи (5.17) введемо по одній змінній $x_{n+i} \geq 0, i = \overline{1, m}$. Ці змінні

включаються у цільову функцію з однаковими від'ємними коефіцієнтами – M , де M – досить велике додатне число. Приходимо до розширеної задачі з $n + m$ змінними (M -задачі):

$$F = \sum_{j=1}^n c_j x_j - M \sum_{i=1}^m x_{n+i}; \quad (5.19)$$

$$\sum_{j=1}^n a_{ij} x_j + x_{n+i} = b_i, \quad i = \overline{1, m}; \quad (5.20)$$

$$x_j \geq 0, \quad j = \overline{1, m+n}, \quad (5.21)$$

Одиничні вектори-стовпці $A_{n+1}, A_{n+2}, \dots, A_{n+m}$ матриці системи рівнянь (5.20) які відповідають штучним змінним, утворюють штучний одиничний базис. Йому відповідає початковий опорний план

$$x = (\underbrace{0, \dots, 0}_n, b_1, b_2, \dots, b_m)$$

і значення цільової функції $F = -M \sum_{i=1}^m x_{n+i}$.

Маючи початковий опорний план, можна застосувати симплекс-метод для знаходження оптимального плану розширеної задачі (5.19) – (5.21). Числа M значно перевищують решту коефіцієнтів цільової функції. За рахунок цього з базису виводяться штучні змінні і вводяться до базису змінні початкової задачі. Для знаходження оптимального плану задачі (5.16) – (5.18) користуються теоремою.

Теорема 5.4. Якщо в оптимальному плані $x^0 = (x_1, x_2, \dots, x_n, \underbrace{0, \dots, 0}_m)$ M – задачі штучні змінні $x_{n+i} = 0, \quad i = \overline{1, m}$, то план $x = (x_1, \dots, x_n)$ – є оптимальним для початкової задачі (5.16) – (5.18).

Отже, якщо симплекс-методом одержано оптимальний план M – задачі, в якому всі штучні змінні рівні нулю, то його перші n компонент утворюють оптимальний план початкової задачі.

Справедливі твердження.

1. Якщо в оптимальному плані хоча б одна із штучних змінних додатна, то початкова задача не має початкових планів – її умови несумісні.
2. Якщо M – задача не має розв'язку, то його не має і початкова задача.

Таким чином, при розв'язуванні M – задачі симплексним методом або приходимо до оптимального розв'язку початкової задачі або встановлюємо її нерозв'язність.

Зауваження.

1. Якщо матриця коефіцієнтів системи рівнянь (5.17) містить $k < m$ різних одиничних векторів, то при складанні M – задачі вводяться $m-k$ штучні змінні.
2. Після виведення штучної змінної з базису, вона не зможе більше туди потрапити. Тому, якщо не вимагається знайти обернену

матрицю, то в подальших симплекс-таблицях стовпчик виведеної з базису штучної змінної можна не розглядати.

Приклад 5.3. Розв'язати симплексним методом ЗЛП:

$$F = 5x_1 + 3x_2 + 4x_3 - x_4 \rightarrow \max;$$

$$\begin{cases} x_1 + 3x_2 + 2x_3 + 2x_4 = 3, \\ 2x_1 + 2x_2 + x_3 + x_4 = 3; \end{cases} \quad x_j \geq 0, j = \overline{1,4}.$$

Розв'язання. Введемо дві штучні змінні x_5, x_6 ; маємо розширену задачу:

$$F = 5x_1 + 3x_2 + 4x_3 - x_4 - Mx_5 - Mx_6 \rightarrow \max;$$

$$\begin{cases} x_1 + 3x_2 + 4x_3 - x_4 + x_5 = 3, \\ 2x_1 + 2x_2 + x_3 + x_4 + x_6 = 3; \end{cases} \quad x_j \geq 0, j = \overline{1,6}.$$

Видно, що x_5, x_6 – базисні, а x_1, x_2, x_3, x_4 – вільні змінні. Базисний розв'язок $x = (0; 0; 0; 0; 3; 3)$ буде початковим опорним планом M – задачі. Запишемо симплексну таблицю (табл. 10.2) і скористаємось симплекс-методом для розв'язування цієї задачі:

Таблиця 5.4

c_6	x_6	b	5	3	4	-1	-M	-M
			A_1	A_2	A_3	A_4	A_5	A_6
-M	x_5	3	1	3	2	2	1	0
-M	x_6	3	2	2	1	1	0	1
Δ		-6M	-3M-5	-5M-3	-3M-4	-3M+1	0	0
3	x_2	1	1/3	1	2/3	2/3	1/3	0
-M	x_6	1	$\frac{4}{3} \frac{3}{4}$	0	-1/3	-1/3	-2/3	1
Δ		3-M	-4M/3-4	0	M/3-2	M/3+3	5M/3+1	0
3	x_2	3/4	0	1	$\frac{4}{3} \frac{3}{4}$	3/4	1/2	-1/4
5	x_1	3/4	1	0	-1/4	-1/4	-1/2	3/4
Δ		6	0	0	-3	2	M-1	M+3
4	x_3	1	0	4/3	1	1	2/3	-1/3
5	x_1	1	1	1/3	0	0	-1/3	2/3
Δ		9	0	4	0	5	M+11/9	M+2

Отже, знайдено оптимальний план вихідної задачі $x = (1; 0; 1; 0)$, при якому $F_{\max} = F(x) = 9$.

Зауважимо, що стовпці A_5 , A_6 утворюють матрицю, з якої можна одержати обернену, помінявши місцями перший і другий рядки.

Завдання для самоконтролю

1. Як одержати початковий опорний план ЗЛП?
2. Як обчислити оцінки компонент опорного плану?
3. Сформулювати критерій оптимальності опорного плану ЗЛП на максимум.
4. Який вигляд має симплексна таблиця?
5. Які перетворення потрібно виконати, щоб перейти до нового базису?
6. Коли цільова функція необмежена в області допустимих розв'язків?
7. Коли вводяться штучні змінні і які коефіцієнти їм відповідають у цільовій функції ЗЛП на максимум?
8. Коли оптимальний план розширеної задачі буде оптимальним і для вихідної задачі?
9. Знайти початковий опорний план.
 $F = x_1 - 3x_2 + x_3 \rightarrow \min;$

$$\begin{cases} 3x_1 - x_2 + 2x_3 \leq 7, \\ -2x_1 + 4x_2 \leq 12, \\ -4x_1 + 3x_2 + 8x_3 \leq 10, \\ x_j \geq 0 (j = \overline{1,3}). \end{cases}$$

(Відповідь. $x_0 = (0; 0; 0; 7; 12; 10)$, $F(x_0) = 0$.)

10. Розв'язати симплексним методом ЗЛП.

$$F = -2x_1 + 3x_2 - 6x_3 - x_4 \rightarrow \min;$$

$$\begin{cases} 2x_1 + x_2 - 2x_3 + x_4 = 24, \\ x_1 + 2x_2 + 4x_3 \leq 22, \\ x_1 - x_2 + 2x_3 \geq 10, \end{cases} \quad x_j \geq 0 (j = \overline{1,4}).$$

(Відповідь. $\vec{x}_{opt}^* = \left(0; 0; \frac{11}{2}; 35; 0; 1\right)$, $F_{\min} = -68$.)

11. Розв'язати симплексним методом ЗЛП.

$$F = -x_1 - 2x_2 - 3x_3 + x_4 \rightarrow \min;$$

$$\begin{cases} x_1 + 2x_2 + 3x_3 = 15, \\ 2x_1 + x_2 + 5x_3 = 20, \\ x_1 + 2x_2 + x_3 + x_4 = 10, \end{cases} \quad x_j \geq 0 (j = \overline{1,4}).$$

(Відповідь. $x_{opt}^* = (2,5; 2,5; 2,5; 0)$ $F_{\min} = -15$.)

12. Розв'язати симплексним методом ЗЛП.

$$F = 2x_1 - x_2 + 3x_3 - 2x_4 + x_5 \rightarrow \max;$$

$$\begin{cases} -x_1 + x_2 + x_3 = 1, \\ x_1 - x_2 + x_4 = 1, \\ x_1 + x_2 + x_5 = 2, \end{cases} \quad x_j \geq 0 (j = \overline{1,5}).$$

(Відповідь. $x_{opt}^* = (1; 0; 2; 0; 0)$, $F_{\max} = 8$.)

13. Розв'язати симплексним методом ЗЛП.

$$F = 2x_1 + x_2 - x_3 - x_4 \rightarrow \min;$$

$$\begin{cases} 2x_1 + x_2 - 3x_3 + x_4 = 6, \\ x_1 + x_2 + 2x_3 - x_4 = 2, \\ x_1 + x_2 + x_3 + x_4 = 7, \end{cases} \quad x_j \geq 0 (j = \overline{1,4}).$$

(Відповідь. $x_{opt}^* = (3; 0; 1; 3)$, $F_{\min} = 2$.)

або в матричній формі:

$$F = \mathbf{c} \cdot \mathbf{x} \rightarrow \max; A\mathbf{x} \leq \mathbf{b}; \mathbf{x} \geq 0. \quad (6.7)$$

Оскільки її можна записати у канонічній формі, то в силу попереднього задача, двоїста до стандартної має вигляд:

$$Q = \mathbf{y} \cdot \mathbf{b} \rightarrow \min; \mathbf{y}A \geq \mathbf{c}, \mathbf{y} \geq 0. \quad (6.8)$$

Неважко переконатись, що двоїста задача до (6.4) співпадає з (6.1), а двоїста задача до (6.8) співпадає з (6.7). Тому пряму і двоїсту задачі називають *взаємно двоїстими* або *взаємно спряженими*. При цьому задачі (6.7) і (6.8) називаються *симетричними*, оскільки системи обмежень в обох випадках мають вигляд нерівностей і всі змінні невід'ємні. Пару взаємно двоїстих задач (6.1) і (6.4) називають *несиметричними*.

Порівнюючи взаємно спряжені задачі, сформулюємо загальні правила переходу від прямої задачі до двоїстої.

1. Змінних y_i у двоїстій задачі стільки, скільки рядків має матриця A прямої задачі.

2. Матриця коефіцієнтів системи обмежень двоїстої задачі є транспонованою до матриці A прямої задачі. Число умов у системі обмежень двоїстої задачі рівне числу змінних прямої задачі.

3. Задача максимізації переходить у задачу мінімізації.

4. Вектор $\mathbf{c}$ коефіцієнтів цільової функції прямої задачі стає вектором правих частин обмежень двоїстої задачі, а вектор $\mathbf{b}$ обмежень прямої задачі стає вектором коефіцієнтів цільової функції двоїстої задачі.

5. Якщо $x_j \geq 0$, то $\sum_{i=1}^m a_{ij} y_i \geq c_j$. Якщо ж x_j може набувати як додатних,

так і від'ємних значень, то $\sum_{i=1}^m a_{ij} y_i = c_j$. Аналогічно, якщо $\sum_{i=1}^n a_{ij} x_j \leq b_i$, то

$y_i \geq 0$; якщо $\sum_{i=1}^n a_{ij} x_j = b_i$, то y_i може приймати значення будь-якого знаку.

Приклад 6.1 Скласти двоїсту задачу до задачі лінійного програмування:

$$F = 3x_1 + x_2 - 4x_3 \rightarrow \max;$$

$$\begin{cases} 2x_1 - x_2 + 4x_3 \leq 12; \\ x_1 + 3x_2 - 2x_3 = 8; \\ 2x_1 + 5x_2 - 6x_3 \leq 7; \end{cases}$$

$$x_1 \geq 0, x_2 \geq 0, x_3 \geq 0.$$

Розв'язання. Для даної задачі маємо:

$$A = \begin{pmatrix} 2 & -1 & 4 \\ 1 & 3 & -2 \\ 2 & 5 & -6 \end{pmatrix}, \mathbf{b} = \begin{pmatrix} 12 \\ 8 \\ 7 \end{pmatrix}, \mathbf{c} = \begin{pmatrix} 3 \\ 1 \\ -4 \end{pmatrix}.$$

Згідно правил побудови двоїстої задачі запишемо:

$$Q = 12y_1 + 8y_2 + 7y_3 \rightarrow \min;$$

$$\begin{cases} 2y_1 + y_2 + 2y_3 \geq 3; \\ -y_1 + 3y_2 + 5y_3 \geq 1; \\ 4y_1 - 2y_2 - 6y_3 \geq -4; \\ y_1 \geq 0, y_3 \geq 0. \end{cases}$$

Зауважимо, що задача максимізації з обмеженнями-нерівностями виду " $\leq$ " переходить у двоїсту задачу мінімізації з обмеженнями виду " $\geq$ " і навпаки.

6.2. Економічне трактування двоїстих задач.

Розглянемо задачу про використання ресурсів. Її математична модель наводилась в лекції 4 і має вигляд (6.6), де a_{ij} – кількість одиниць ресурсу R_i , що витрачається на виробництво одиниці продукції P_j ; b_i – запас ресурсу R_i ; c_j – доход від реалізації одиниці продукції P_j . Задача полягає в складанні такого плану $x = (x_1, x_2, \dots, x_n)$ випуску продукції, щоб загальний доход від виробництва був максимальним.

Оцінимо наявні на підприємстві ресурси. Будемо вважати, що вони можуть використовуватись або для виробництва продукції, або продаватись іншому підприємству. Позначимо через $y_i \geq 0$ вартість одиниці ресурсу R_i , $i = \overline{1, m}$. Тоді затрати на виробництво одиниці продукції P_j становитимуть

$$\sum_{i=1}^m a_{ij} y_i, \quad j = \overline{1, n}. \text{ Порядок цих ресурсів доцільний за умови, що } \sum_{i=1}^m a_{ij} y_i \geq c_j,$$

$$j = \overline{1, n}. \text{ Вартість усіх ресурсів становить } \sum_{i=1}^m b_i y_i \text{ і підприємство-покупець}$$

зацікавлене, щоб ця величина була якнайменшою. В результаті приходимо до двоїстої задачі (6.8): знайти таку ціну одиниці кожного ресурсу, щоб затрати на випуск одиниці кожної продукції були не менші, ніж доход від реалізації цієї продукції, і загальна вартість усіх ресурсів була б мінімальною.

Змінні y_i називаються *оцінками* або *неявними цінами*.

6.3. Теорема двоїстості.

З'ясуємо зв'язок між оптимальними розв'язками взаємно двоїстих задач. Розглянемо пару двоїстих задач (6.7) і (6.8). Позначимо області допустимих розв'язків цих задач через D та G :

$$D = \{ x \mid Ax \leq b, x \geq 0 \}, \quad G = \{ y \mid yA \geq c, y \geq 0 \}.$$

Лема 1. Для будь-яких допустимих планів $\bar{x}$ та $\bar{y}$ відповідно задач (6.7) і (6.8) справедлива нерівність:

$$c \cdot x \leq y \cdot b. \quad (6.9)$$

Доведення. Нехай x – розв'язок задачі (6.7), а y – розв'язок задачі (6.8). Оскільки $Ax \geq b, x \geq 0$, то маємо нерівність $yAx \leq y \cdot b$. Аналогічно із того, що $yA \geq c, x \geq 0$, одержимо $yAx \geq c \cdot x$. Отже, маємо $c \cdot x \leq yAx \leq y \cdot b$, що й доводить лему.

Нерівність (6.9) називають основною нерівністю теорії двоїстості. Її економічний зміст полягає у тому, що при будь-якому допустимому плані виробництва x загальна вартість всієї продукції не більша, ніж сумарна оцінка ресурсів при будь-якому допустимому векторі оцінок y .

Лема 2. Якщо для деяких допустимих планів x^* , y^* пари двоїстих задач (6.7) і (6.8) виконується рівність:

$$c \cdot x^* \leq y^* \cdot b, \quad (6.10)$$

то вектори x^* , y^* – оптимальні плани відповідних прямої і двоїстої задач.

Доведення. Вектор x^* буде оптимальним розв'язком задачі (6.7) тоді і тільки тоді, коли для будь-якого допустимого вектора x виконуватиметься нерівність $F = c \cdot x \leq c \cdot x^*$. Нехай x – допустимий план прямої задачі. Тоді, згідно з лемою 1 і умовою (6.10), маємо $c \cdot x \leq y^* \cdot b = c \cdot x^*$. Оптимальність вектора y^* доводиться аналогічно.

Теорема 6.1 (*перша теорема двоїстості*). Якщо одна із двоїстих задач (6.7) або (6.8) має оптимальний розв'язок, то існує оптимальний розв'язок і другої задачі, причому значення цільових функцій співпадають: $F_{\max} = Q_{\min}$.

Якщо цільова функція однієї із задач не обмежена (для прямої – зверху, для двоїстої – знизу), то друга задача не має допустимих планів.

Економічний зміст теореми 6.1: якщо задача визначення оптимального плану випуску продукції має розв'язок, то існує також розв'язок і задачі визначення оцінок ресурсів; причому доход, одержаний від реалізації оптимального плану, співпадає із загальною оцінкою ресурсів. Рівність значень цільових функцій при відповідних розв'язках пари двоїстих задач достатня для того, щоб ці розв'язки були оптимальними. Це означає, що план виробництва і вектор оцінок ресурсів оптимальні тоді і тільки тоді, коли вартість виробленої продукції і сумарна оцінка ресурсів співпадають. Оцінки виступають як інструмент балансування витрат і результатів. Двоїсті оцінки мають ту властивість, що вони гарантують рентабельність оптимального плану, тобто рівність загальної вартості продукції і ресурсів обумовлює збитковість будь-якого іншого плану, відмінного від оптимального.

Теорема 6.2 (*друга теорема двоїстості*). Для того, щоб допустимі розв'язки x^* та y^* відповідних двоїстих задач (6.7) і (6.8) були оптимальними, необхідно і достатньо, щоб виконувались умови:

$$1) (y^* A - c) x^* = 0; \quad 2) y^* (b - Ax^*) = 0.$$

Таким чином: якщо j -та компонента оптимального вектора x^* додатна, то j -те обмеження двоїстої задачі перетворюється в рівність; якщо ж j -те обмеження двоїстої задачі не перетворюється у строгу рівність оптимальним планом цієї задачі, то j -та компонента оптимального плану x^* дорівнює нулю, тобто:

$$\text{якщо } x_j^* > 0, \text{ то } \sum_{i=1}^m a_{ij} y_i^* = c_j;$$

якщо $\sum_{i=1}^m a_{ij}y_i^* > c_j$, то $x_j^* = 0$ ($j = \overline{1, n}$).

Аналогічно, якщо $y_i^* > 0$, то $\sum_{j=1}^m a_{ij}x_j^* = b_i$;

якщо $\sum_{j=1}^m a_{ij}x_j^* < b_i$, то $y_i^* = 0$ ($i = \overline{1, m}$).

З'ясуємо економічний зміст умов теореми 6.2. Згідно умови 1), якщо деякий продукт P_j входить до оптимального плану виробництва ($x_j^* > 0$), то в оптимальній системі цін двоїстої задачі витрати ресурсів на виготовлення цього продукту співпадають з його вартістю. Якщо ж витрати ресурсів на випуск продукції P_j перевищують його вартість, то ця продукція не виробляється ($x_j^* = 0$).

За умовою 2), якщо в оптимальній системі цін ресурс R_i одержує відмінну від нуля ціну $y_i^* > 0$, то у відповідності з оптимальним планом виробництва прямої задачі цей ресурс буде повністю використаний. Якщо ж ресурс R_i витрачається не повністю, то його ціна $y_i^* = 0$.

Отже, двоїсті оцінки можуть бути мірою дефіцитності ресурсів. Дефіцитний ресурс (той, що повністю витрачається за оптимальним планом виробництва) має оцінку $y_i^* > 0$. Ресурс, який не використовується повністю при оптимальному плані виробництва, одержує нульову оцінку.

Наведені теореми двоїстості залишаються справедливими і для несиметричної пари спряжених задач.

Нехай симплексним методом знайдено оптимальний план $\mathbf{x}^*$ задачі (6.1). Йому відповідає базис, утворений векторами, наприклад $\mathbf{A}'_1, \mathbf{A}'_2, \dots, \mathbf{A}'_m$ і $\mathbf{c}_0 = (c_1, c_2, \dots, c_m)$. Позначимо через B матрицю, складену із компонент векторів $\mathbf{A}_1, \mathbf{A}_2, \dots, \mathbf{A}_m$ базису, а B^{-1} – обернену до неї.

Теорема 6.3. Якщо задача лінійного програмування (6.1) має оптимальний план $\mathbf{x}^*$, то вектор:

$$\mathbf{y}^* = \mathbf{c}_0 B^{-1} \quad (6.12)$$

буде оптимальним розв'язком двоїстої задачі (6.4).

Отже, щоб знайти оптимальний план двоїстої задачі, потрібно знати матрицю B^{-1} . Виявляється, матрицю B^{-1} можна визначити із останньої симплекс-таблиці: вона стоїть на місці одиничної матриці, утвореної компонентами базисних векторів початкового опорного плану.

Приклад 6.2. До задачі лінійного програмування:

$$F = 5x_1 + 3x_2 + 4x_3 - x_4 \rightarrow \max;$$

$$\begin{cases} x_1 + 3x_2 + 2x_3 + 2x_4 = 3; \\ 2x_1 + 2x_2 + x_3 + x_4 = 3; \end{cases}$$

$$x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, x_4 \geq 0$$

скласти двоїсту задачу і знайти її розв'язок.

Розв'язання. Видно, що

$$A = \begin{pmatrix} 1 & 3 & 2 & 2 \\ 2 & 2 & 1 & 1 \end{pmatrix}, \mathbf{b} = \begin{pmatrix} 3 \\ 3 \end{pmatrix}, \mathbf{c}^T = (5; 3; 4; -1).$$

Вводимо невідомі $\mathbf{y} = (y_1, y_2)$ і записуємо двоїсту задачу:

$$Q = 3y_1 + 3y_2 \rightarrow \min; \begin{cases} y_1 + 2y_2 \geq 5; \\ 3y_1 + 2y_2 \geq 3; \\ 2y_1 + y_2 \geq 4; \\ 2y_1 + y_2 \geq -1. \end{cases}$$

Розв'яжемо спочатку пряму задачу. Розв'язок цієї задачі наведено в табл. 5.2 лекції 5. Із останньої симплекс-таблиці видно, що компоненти векторів $\mathbf{A}_5, \mathbf{A}_6$ утворюють матрицю, з якої можна одержати обернену, помінявши місцями перший і другий рядки:

$$B^{-1} = \begin{pmatrix} -1/3 & 2/3 \\ 2/3 & -1/3 \end{pmatrix}, \text{ а } \mathbf{c}_6 = (c_1, c_3) = (5; 4).$$

Оптимальним розв'язком двоїстої задачі буде вектор:

$$\mathbf{y}^* = \mathbf{c}_6 \cdot B^{-1} = \begin{pmatrix} 2 \\ 1 \end{pmatrix}, \text{ тобто } y_1^* = 2, y_2^* = 1.$$

Йому відповідає мінімальне значення цільової функції двоїстої задачі: $Q_{\min} = 3 \cdot 2 + 3 = 9$, що співпадає із $F_{\max}$.

6.4. Двоїстий симплекс-метод.

Тісний зв'язок між взаємно двоїстими задачами полягає в тому, що при розв'язуванні однієї із них одночасно розв'язується і друга. Дійсно, якщо для задачі (6.1) заповнити симплексну таблицю, то можна помітити, що вона включає і умови двоїстої задачі (6.4): стовпці прямої стають рядками двоїстої і навпаки. Елементи стовпця $\mathbf{b}$ є коефіцієнтами вектора обмежень прямої задачі і одночасно оцінками плану двоїстої задачі. Елементи Δ -рядка – оцінки плану прямої задачі і одночасно коефіцієнти вектора обмежень двоїстої задачі. Тому немає потреби будувати спеціальну таблицю при розв'язуванні двоїстої задачі, а можна користуватись симплекс-таблицею прямої задачі. Компоненти оптимального плану двоїстої задачі знаходяться в тих клітинках Δ -рядка, які відповідають початковому одиничному базису прямої задачі.

Симплекс-метод застосовується для розв'язування задач з невід'ємними коефіцієнтами $\mathbf{b}_i$ вектора $\mathbf{b}$ і коефіцієнтами Δ_j довільних знаків. Іноді буває легше знайти базис, який задовольняє критерію оптимальності (всі $\Delta_j \geq 0$), але не задовольняє умові допустимості (не всі $b_i \geq 0$). Варіант симплекс-метода, який використовують для розв'язування таких задач називається *двоїстим симплекс-методом*. Цим методом розв'язуються задачі лінійного програмування:

$$F = \mathbf{c} \cdot \mathbf{x} \rightarrow \max; A\mathbf{x} = \mathbf{b}; \mathbf{x} \geq 0, \quad (6.13)$$

де матриця A містить одиничний базис і всі коефіцієнти $\Delta_j \geq 0$, $j = \overline{1, n}$. При цьому не вимагається виконання умови $\mathbf{b} \geq 0$. Таку задачу будемо називати задачею у *двоїстій базисній формі*.

Звичайний симплекс-метод, який застосовується до задач у базисній формі, приводить до послідовності еквівалентних задач із зростаючими значеннями цільової функції і невід'ємними значеннями b_i . Двоїстий симплекс-метод, який застосовується до задач у двоїстій базисній формі, приводить до послідовності задач із спадними значеннями цільової функції, невід'ємними коефіцієнтами Δ_j , $(j = \overline{1, n})$ і значеннями b_i будь-якого знаку. Перетворення задач виконуються до тих пір, доки не буде встановлено, що початкова задача не має допустимого розв'язку або буде одержана задача із допустимим базисним розв'язком (усі $b_i \geq 0$), який одночасно і оптимальний.

Ці особливості алгоритмів знайшли відображення у назвах. Звичайний симплекс-метод називають методом *послідовного покращення планів*, а двоїстий – методом *послідовного уточнення оцінок*.

Розглянемо задачу (6.13) у двоїстій базисній формі. Для неї можливий один із трьох випадків:

1. Всі координати вектора обмежень невід'ємні: $b_i \geq 0$, $i = \overline{1, m}$. Тоді вони дають не тільки допустимий, але й оптимальний план задачі, оскільки за припущенням всі оцінки $\Delta_j \geq 0$.

2. Існує рядок i ($i \leq m$) такий, що $b_i < 0$ і всі $a_{ij} \geq 0$ ($j = \overline{1, n}$). Тоді задача не має розв'язків через несумісність обмежень.

Дійсно, для будь-якого невід'ємного вектора $\mathbf{x} = (x_1, x_2, \dots, x_n)$ при всіх $a_{ij} \geq 0$ маємо $\sum_{j=1}^n a_{ij} x_j \geq 0$. Тому ні при якому із векторів $\bar{\mathbf{x}} \geq 0$ не виконується i -те обмеження задачі (6.13):

$$\sum_{j=1}^n a_{ij} x_j = b_i, \text{ де } b_i < 0.$$

3. Існує рядок r ($r \leq m$) такий, що $b_r < 0$ і $a_{rj} < 0$ хоча б для одного $j \in \{1, 2, \dots, n\}$. Нехай елемент $a_{rk} < 0$ такий, що

$$\frac{\Delta_k}{a_{rk}} = \max_{a_{rj} < 0} \left\{ \frac{\Delta_j}{a_{rj}} \right\}. \quad (6.14)$$

Тоді введення в базис вектора A_k дає новий план $\mathbf{x}'$, для якого оцінки $\Delta_j \geq 0$ ($j = \overline{1, n}$) і значення цільової функції $F(\mathbf{x}') \leq F(\mathbf{x})$.

Приведемо алгоритм двоїстого симплексного методу розв'язування задачі лінійного програмування, записаної у двоїстій базисній формі.

1. За даними задачі (6.13) складаємо симплекс-таблицю.

2. Перевіряємо знаки компонент вектора $\mathbf{b}$. Якщо всі $b_i \geq 0$ ($i = \overline{1, m}$), то маємо випадок 1: базисний розв'язок і значення цільової функції, записані у

стовпці $\mathbf{b}$ таблиці, дають оптимальний план задачі (6.13). Якщо не всі $b_i \geq 0$, то переходимо до пункту 3.

3. Серед від'ємних коефіцієнтів b_i вибираємо найбільше за абсолютною величиною значення b_r ; рядок r обираємо розв'язальним.

4. Перевіряємо знаки усіх елементів a_{rj} ($j = \overline{1, n}$) розв'язального рядка. Якщо усі $a_{rj} \geq 0$, то маємо випадок 2. Процес розв'язування закінчено, задача не має планів. Якщо знайдеться хоча б один елемент $a_{rj} < 0$, то маємо випадок 3. Переходимо до пункту 5.

5. Серед елементів $a_{rj} < 0$ розв'язального рядка вибираємо провідний елемент a_{rk} , для якого виконується умова (6.14).

6. Виконуємо жорданові перетворення симплексної таблиці з розв'язувальним елементом a_{rk} і переходимо до пункту 2.

Зауважимо, що двоїстий симплекс-метод відрізняється від звичайного правилом вибору вектора, який потрібно вивести з базису, і вектора, який слід ввести в базис.

Двоїстий симплекс-метод доцільно використовувати для розв'язування задач, які мають одиничний базис, але не належать до задач в базисній або двоїстій базисній формі, оскільки мають одночасно від'ємні коефіцієнти серед елементів вектора $\mathbf{b}$ і Δ -рядка. Розв'язування такої задачі складається із двох етапів: спочатку за допомогою процедури двоїстого симплекс-методу вилучаються усі $x_j < 0$, а потім звичайним симплекс-методом знаходиться оптимальний план. При цьому на першому етапі потрібно пункт 5 алгоритму замінити на пункт 5'.

5'. Серед елементів $a_{rj} < 0$ розв'язального рядка вибрати елемент a_{rk} , для якого:

$$\frac{b_r}{a_{rk}} = \max_{a_{rj} < 0} \left\{ \frac{b_r}{a_{rj}} \right\}. \quad (6.15)$$

Завдання для самоконтролю.

1. Як формулюється двоїста задача до задачі у стандартній формі, у канонічній формі?

2. Які задачі ЛП називаються симетричними, несиметричними?

3. Сформулюйте теореми двоїстості, дайте їх економічне трактування?

4. Як за розв'язком прямої задачі знайти розв'язок двоїстої задачі?

5. Для наведеної задачі записати двоїсту. Розв'язати одну з них симплексним методом і за знайденим розв'язком одержати розв'язок іншої:

$$F = 5x_1 + 4x_2 + 6x_3 \rightarrow \max;$$

$$\begin{cases} x_1 + x_2 + x_3 \leq 6, \\ 2x_1 - x_2 + 3x_3 \geq 9, \\ 3x_1 + x_2 + 2x_3 \geq 11, \\ x_j \geq 0 (j = \overline{1, 3}). \end{cases}$$

$$(\text{Відповідь. } x_{opt}^* = (0; 0; 6 | 0; 9; 1), \quad F_{\max} = 36).$$

Розв'язок двоїстої задачі $y_{onm}^* = (6; 0; 0 | 1; 2; 0)$, $Q_{\min} = 36$.)

6. Продукція в цеху може виготовлятися трьома технологічними способами $T_j (j = \overline{1,3})$. Обсяг ресурсів $b_i (i = \overline{1,3})$ та їх витрати за одиницю часу для кожної технології, а також продуктивності технологій (у грошових одиницях за одиницю часу роботи по даній технології) представлено в табл. 8.1.

Таблиця 8.1

Ресурси	Технологічні способи			Обсяг	$\vec{Y}$
	T_1	T_2	T_3	ресурсу	
Робоча сила, люд.-год.	15	20	25	1200	y_1
Сировина, т	2	3	2,5	150	y_2
Електроенергія, кВт-год	35	60	60	300	y_3
Продуктивність технологічного способу	300	250	450		
План $\vec{X}$	x_1	x_2	x_3		

Потрібно визначити оптимальний план використання кожного технологічного способу $x_j (j = \overline{1,3})$, тобто час використання кожного способу. Записати розв'язок двоїстої задачі та перевірити умови другої теореми двоїстості.

(Відповідь. $x_{opt}^* = (60; 0; 12 | 0; 0; 180)$, $F_{\max} = 23400$,
 $y_{opt}^* = (12; 60; 0 | 0; 170; 0)$, $Q_{\min} = 23400$.)

7. Розв'язати двоїстим симплексним методом ЗЛП.

$$F = x_1 + x_2 + 2x_3 \rightarrow \max;$$

$$\begin{cases} x_1 + x_2 + x_3 = 8, \\ x_1 - x_2 \geq 4, \\ x_1 + 2x_2 \geq 6, \end{cases}$$

$$x_1 \geq 0, x_2 \geq 0, x_3 \geq 0.$$

(Відповідь. $x^* = (14/3; 2/3; 8/3; 0; 0)$ і $F_{\max} = 32/3$.)

8. Знайти розв'язок двоїстої задачі до даної

$$F = -3x_1 + 5x_2 + x_3 - x_4 \rightarrow \max$$

$$3x_1 + 8x_2 + x_3 + x_4 = 50$$

$$5x_1 - 4x_2 - x_3 + x_4 = 14$$

$$x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, x_4 \geq 0.$$

(Відповідь. $y^* = (0,25; -0,75)$, $Q_{\min} = 2$.)

9. Розв'язати двоїстим симплексним методом ЗЛП.

$$F = 7x_1 + 7x_2 + 2x_3 + x_4 + 6x_5 \rightarrow \max$$

$$3x_1 - x_2 + x_3 + 2x_4 = -3$$

$$-x_1 + 3x_2 - 3x_4 + x_6 = 12,$$

$$2x_1 + x_2 + x_4 + x_5 = 4,$$

$$x_j \geq 0 (j = \overline{1,2,\dots,6})$$

(Відповідь. $x_{opt} = (17, 0, 0, 12, 3, 0)$, $F_{\max} = 30$.)

Лекція № 7. Транспортна задача.

Властивості транспортної задачі.

Побудова початкового опорного плану.

Метод потенціалів.

Відкрита модель транспортної задачі.

7.1. Властивості транспортної задачі.

Важливе місце в лінійному програмуванні посідає транспортна задача, яка полягає в мінімізації загальної вартості перевезень деякої кількості продукції із m пунктів відправлення A_i у n пунктів споживання B_j . Задачі такого типу виникають при вирішенні проблем раціонального розподілу ресурсів, які у відповідності із виробничими міркуваннями перевозяться із пунктів виробництва до пунктів споживання. Математична модель транспортної задачі розглядалася у лекції 4: знайти значення змінних x_{ij} , при яких

$$F = \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} \rightarrow \min ; \quad (7.1)$$

за умов

$$\sum_{j=1}^n x_{ij} = a_i, \quad (i = \overline{1, m}); \quad (7.2)$$

$$\sum_{i=1}^m x_{ij} = b_j, \quad (j = \overline{1, n}). \quad (7.3)$$

$$x_{ij} \geq 0, \quad (i = \overline{1, m}, j = \overline{1, n}). \quad (7.4)$$

Тут x_{ij} – обсяг продукції, планується перевезти з пункту відправлення A_i у пункт споживання B_j ; a_i – запас продукції у пункті відправлення A_i ; b_j – обсяг попиту пункту споживання B_j ; c_{ij} – вартість перевезення одиниці продукції із пункту постачання A_i у пункт споживання B_j .

Якщо загальний запас продукції у всіх пунктах відправлення дорівнює сумарній потребі усіх пунктів споживання:

$$\sum_{i=1}^m a_i = \sum_{j=1}^n b_j, \quad (7.5)$$

то таку задачу називають закритою. Якщо ж умова балансу (7.5) не виконується, то транспортну задачу називають відкритою.

Розглядатимемо закриту транспортну задачу. Введемо вектори:

$$\bar{x} = (x_{11}, x_{12}, \dots, x_{1n}, x_{21}, x_{22}, \dots, x_{2n}, \dots, x_{m1}, x_{m2}, \dots, x_{mn})^T \in R^{mn},$$

$$\bar{b} = (a_1, a_2, \dots, a_m, b_1, b_2, \dots, b_n)^T \in R^{m+n},$$

$$\bar{c} = (c_{11}, c_{12}, \dots, c_{1n}, c_{21}, c_{22}, \dots, c_{2n}, \dots, c_{m1}, c_{m2}, \dots, c_{mn})^T \in R^{mn}$$

$$\bar{A}_{ij} = (0, \dots, 0, 1, 0, \dots, 0, 1, 0, \dots, 0)^T \in R^{m+n},$$

де $\bar{A}_{ij}$ – стовпці елементів матриці $A_{(m+n) \times (mn)}$ при невідомих x_{ij} системи рівнянь (7.2), (7.3). Це дозволяє записати транспортну задачу (7.1) – (7.4) у матричній формі:

$$F = \bar{c} \cdot \bar{x} \rightarrow \min; A\bar{x} = \bar{b}; \bar{x} \geq 0. \quad (7.6)$$

Отже, маємо задачу лінійного програмування із mn змінними і $m + n$ обмеженнями рівняннями.

Будь-який невід'ємний розв'язок системи рівнянь (7.2), (7.3), який визначається матрицею перевезень $X_{m \times n} = (x_{ij})$, називається планом транспортної задачі.

Вигляд рівнянь системи (7.2), (7.3) і відповідної їм матриці A , елементами якої є лише нулі і одиниці, надає транспортній задачі специфічних властивостей.

1. Ранг матриці A системи обмежень (7.2), (7.3) транспортної задачі дорівнює $m + n - 1$.

Для доведення цієї властивості кожне з рівнянь (7.2) помножимо на $c_i = 1$, ($i = \overline{1, m}$), а кожне з рівнянь (7.3) – на $c_{m+j} = -1$, ($j = \overline{1, n}$) і результат додамо. Враховуючи умову балансу (7.5), маємо:

$$\sum_{i=1}^m c_i \sum_{j=1}^n x_{ij} + \sum_{j=1}^n c_{m+j} \sum_{i=1}^m x_{ij} = \sum_{i=1}^m a_i - \sum_{j=1}^n b_j = 0.$$

Це означає, що система рівнянь (7.2), (7.3) лінійно залежна і $r(A) \leq m + n - 1$.

Покажемо тепер, що насправді $r(A) = m + n - 1$. Для цього достатньо в матриці A знайти квадратний мінор порядку $m + n - 1$, визначник якого відмінний від нуля. Таким буде мінор, складений з перших $m + n - 1$ компонент векторів $\bar{A}_{1n}, \bar{A}_{mn}, \dots, \bar{A}_{mn}, \bar{A}_{11}, \bar{A}_{12}, \dots, \bar{A}_{1n-1}$:

$$\begin{array}{c} m \left\{ \begin{array}{c} \left| \begin{array}{cccccccc} 1 & 0 & \dots & 0 & 1 & 1 & \dots & 1 \\ 0 & 1 & \dots & 0 & 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 1 & 0 & 0 & \dots & 0 \end{array} \right| \\ \vdots \\ \left| \begin{array}{cccccccc} 0 & 0 & \dots & 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 0 & 0 & 0 & \dots & 1 \end{array} \right| \end{array} \right. = 1 \\ n-1 \left\{ \begin{array}{c} \vdots \\ \vdots \\ \vdots \\ \vdots \\ \vdots \end{array} \right. \end{array}$$

$\square \square \square \square \square \square$
 m

$\square \square \square \square \square \square$
 $n-1$

2. Закрита транспортна задача має оптимальний план.

Для доведення цієї властивості покажемо, що для транспортної задачі завжди існує допустимий план і лінійна функція обмежена на множині планів.

Нехай

$$\sum_{i=1}^m a_i = \sum_{j=1}^n b_j = S > 0.$$

Розглянемо величини

$$x_{ij} = \frac{a_i b_j}{S} \geq 0, \quad (i = \overline{1, m}; j = \overline{1, n}).$$

Ці величини задовольняють систему рівнянь (7.2) – (7.3):

$$\sum_{j=1}^n x_{ij} = \frac{a_i}{S} \sum_{j=1}^n b_j = \frac{a_i}{S} \cdot S = a_i, \quad (i = \overline{1, m}),$$

$$\sum_{i=1}^m x_{ij} = \frac{b_j}{S} \sum_{i=1}^m a_i = \frac{b_j}{S} \cdot S = b_j, \quad (j = \overline{1, n}).$$

Отже, набір значень x_{ij} є планом задачі (7.1) – (7.4).

Виберемо серед значень c_{ij} найбільше $C = \max c_{ij}$ і найменше $c = \min c_{ij}$. Для лінійної функції (7.1) з урахуванням (7.2) маємо оцінки:

$$c \cdot S = c \sum_{i=1}^m a_i = c \sum_{i=1}^m \sum_{j=1}^n x_{ij} \leq \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} \leq C \sum_{i=1}^m \sum_{j=1}^n x_{ij} = C \sum_{i=1}^m a_i = C \cdot S,$$

тобто цільова функція обмежена на множині її планів.

3. Опорний план транспортної задачі містить не більше $m + n - 1$ додатних перевезень.

Опорний план $X_{m \times n} = (x_{ij})$ транспортної задачі називається невикористаним, якщо він містить $m + n - 1$ додатних величин x_{ij} .

Отже, якщо одержано опорний план транспортної задачі, то матриця X має рівно $m + n - 1$ додатних компонент, а решта – нулі.

Транспортну задачу, як задачу лінійного програмування можна розв'язувати симплекс-методом. Однак через простий вигляд системи обмежень (7.2) – (7.3) для її розв'язування розроблено більш простий варіант симплекс-методу – метод потенціалів. Розв'язування транспортної задачі цим методом включає такі етапи:

- 1) знаходження початкового опорного плану;
- 2) дослідження плану на оптимальність;
- 3) перехід до іншого опорного плану шляхом заміни однієї із базисних змінних на вільну.

Перед тим як розглянути метод розв'язування транспортної задачі (7.1) – (7.4), представимо умову задачі у вигляді таблиці перевезень (табл. 7.1), яку ще називають транспортною. Клітинки таблиці, в яких записані ненульові перевезення, називаються зайнятими, а решта – вільними. Клітинку таблиці, яка розміщена на перетині i -го рядка та j -го стовпчика, позначають через (i, j) .

Циклом у таблиці перевезень називають замкнену ламану лінію, вершини якої розміщені в клітинах таблиці і з кожної вершини виходять два відрізки: один по рядку, другий – по стовпчику.

Таблиця 7.1

B_j					Запаси
A_i	B_1	B_2	$\dots$	B_n	a_i
	c_{11}	c_{12}		c_{1n}	
A_1	x_{11}	x_{12}	$\dots$	x_{1n}	a_1
	c_{21}	c_{22}		c_{2n}	
A_2	x_{21}	x_{22}	$\dots$	x_{2n}	a_2
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
	c_{m1}	c_{m2}		c_{mn}	
A_m	x_{m1}	x_{m2}	$\dots$	x_{mn}	a_m
Потреби			$\dots$		
b_j	b_1	b_2		b_n	

Вершини одного і того самого відрізка циклу називають сусідніми. Отже, тільки дві сусідні вершини циклу можуть лежати в одному стовпці або одному рядку таблиці.

7.2. Побудова початкового опорного плану.

Для розв'язування транспортної задачі методом потенціалів потрібно мати початковий опорний план. Існує кілька способів побудови початкового опорного плану: метод північно-західного кута, метод мінімальної вартості, метод Фогеля та інші.

Найпростішим із них є метод північно-західного кута. Згідно з цим методом побудову опорного плану починають з лівої верхньої клітини (1,1) таблиці перевезень. Продукцію першого пункту відправлення розподіляють таким чином, щоб спочатку максимально врахувати потреби першого споживача, потім другого і т. д. до повного вичерпування запасу пункту A_1 . Таким же чином послідовно розподіляється продукція пунктів відправлення $A_2, A_3, \dots, A_m$. При цьому пункти споживання B_j , потреби яких повністю враховані, тимчасово не розглядаються.

Заповнені зазначеним способом клітини утворюють характерну східчасту фігуру, яка починається у клітині (1, 1), а закінчується у клітині (m, n) транспортної таблиці.

Оскільки, при побудові початкового опорного плану вартості перевезень не враховувались, то цей план буде далеким від оптимального.

Приклад.7.1. Методом північно-західного кута побудувати початковий опорний план транспортної задачі, яка задана таблицею 7.2.

Таблиця 7.2

Пункти відправлення	Пункти споживання				Запаси a_i
	B_1	B_2	B_3	B_4	
A_1	4	3	2	6	50
A_2	2	4	5	1	70
A_3	3	6	7	5	100
Потреби b_j	40	60	50	70	

Розв'язання. Будуємо опорний план $X_{3 \times 4} = (x_{ij})$, де кожне значення x_{ij} поміщатиметься у клітину (i, j) транспортної таблиці. Розподіляємо запаси $a_1 = 50$ од. пункту відправлення A_1 . Оскільки потреби пункту споживання B_1 становлять $b_1 = 40$ од., то в клітину $(1, 1)$ таблиці 7.3 записуємо число $x_{11} = \min(a_1, b_1) = \min(50, 40) = 40$. Потреби пункту B_1 повністю враховано, а запаси пункту A_1 зменшились до $a_1 - x_{11} = 50 - 40 = 10$ од. Це число вписуємо в клітину $(1, 2)$ таблиці 7.3: $x_{12} = \min(a_{11} - x_{11}, b_2) = \min(10, 60) = 10$. Внаслідок такого розподілу запаси продукції пункту відправлення A_1 повністю вичерпані. У подальших розрахунках пункти A_1 та B_1 не розглядаються.

Розподіляємо запаси продукції пункту A_2 . Оскільки потреби пункту B_2 уже частково враховані, то у клітину $(2, 2)$ вноситься число $x_{12} = \min(a_2, b_2 - x_{12}) = \min(70, 60 - 10) = 50$. Потреби пункту B_2 повністю враховані, а у пункті відправлення A_2 залишилось $a_2 - x_{22} = 70 - 50 = 20$ од. продукції.

Потреби пункту B_3 можна задовольнити частково рештками запасу пункту A_2 ; у клітину $(2, 3)$ записуємо $x_{23} = \min(a_2 - x_{22}, b_3) = \min(20, 50) = 20$.

Розподіляємо запаси пункту A_3 . Знаходимо $x_{33} = \min(a_3, b_3 - x_{23}) = \min(100, 50 - 20) = 30$. Потреби споживача B_3 враховані повністю, а у пункті A_3 залишилось $a_3 - x_{33} = 100 - 30 = 70$ од. продукції, що становить потребу пункту B_4 ; приймаємо $x_{34} = 70$. Остаточного одержуємо таблицю 7.3.

Таблиця 7.3

$A_i \backslash B_j$	B_1	B_2	B_3	B_4	a_i
A_1	4 40	3 10	2	6	50
A_2	2	4 50	5 20	1	70
A_m	3	6	7 30	5 70	100
b_j	40	60	50	70	

Отже, маємо початковий опорний план:

$$X = \begin{pmatrix} 40 & 10 & 0 & 0 \\ 0 & 50 & 20 & 0 \\ 0 & 0 & 30 & 70 \end{pmatrix}.$$

Цей план є невиродженим, оскільки число додатних компонент рівне $6 = m + n - 1$. Йому відповідає значення цільової функції:

$$F = \sum_{i=1}^3 \sum_{j=1}^4 c_{ij} x_{ij} = 4 \cdot 40 + 3 \cdot 10 + 4 \cdot 50 + 5 \cdot 20 + 7 \cdot 30 + 5 \cdot 70 = 1050.$$

Метод мінімальної вартості враховує тарифи перевезень при виборі пунктів відправлення і споживання. Суть цього методу полягає у тому, що вибирається клітина (i, j) таблиці перевезень із мінімальною вартістю і приймається $x_{ij} = \min(a_i, b_j)$. Якщо запаси продукції A_i пункту відправлення повністю вичерпані або повністю враховані потреби B_j пункту споживання, то відповідний i -й рядок або j -й стовпець транспортної таблиці не враховується при подальшому аналізі. У частині таблиці, що залишилась, знову вибирають найменшу вартість і продовжують розподіл запасів. Процес триває доти, поки не будуть вичерпані запаси усіх пунктів відправлення і повністю задоволений попит усіх пунктів споживання.

Приклад 7.2. Методом мінімальної вартості знайти початковий опорний план транспортної задачі, заданої таблицею 7.2.

Розв'язання. Знаходимо у таблиці найменшу вартість c_{ij} : $c_{24} = 1$. Приймаємо $x_{24} = \min(a_2, b_4) = \min(70, 70) = 70$ і виключаємо з подальшого розгляду другий рядок та четвертий стовпець. У частині таблиці, що залишилась найменшою є вартість $c_{13}=2$. У клітину $(1,3)$ записуємо число $x_{13} = \min(a_1, b_3) = \min(50, 50) = 50$; перший рядок і третій стовпець далі не розглядатимуться. Залишилось розподілити запаси продукції пункту відправлення A_3 . У третьому рядку знаходимо найменшу вартість серед незаповнених клітин: $c_{31} = 3$. Приймаємо $x_{31} = \min(a_3, b_1) = \min(10, 40) =$

40. Решту – частину запасу пункту A_3 розміщуємо у клітину (3, 2): $x_{23} = \min(a_3 - x_{31}, b_2) = \min(100 - 40, 60) = 60$.

В результаті повного розподілу продукції приходимо до таблиці 7.4.

Таблиця 7.4

$A_i \backslash B_j$	B_1	B_2	B_3	B_4	a_i
A_1	4	3	2	6	50
A_2	2	4	5	1	70
A_m	3	6	7	5	100
b_j	40	60	50	70	

Отже, маємо початковий опорний план:

$$X = \begin{pmatrix} 0 & 0 & 50 & 0 \\ 0 & 0 & 0 & 70 \\ 40 & 60 & 0 & 0 \end{pmatrix},$$

для якого: $F = 2 \cdot 50 + 1 \cdot 70 + 3 \cdot 40 + 6 \cdot 60 = 650$ од. Видно, що цей план більш близький до оптимального.

Оскільки число додатних компонент матриці X рівне 4, $m + n - 1 = 6$, то одержаний план є виродженим.

Зауважимо, що у випадку виродженого опорного плану число його додатних компонент доповнюється до кількості $m + n - 1$ за рахунок вільних клітин із нульовим базисним значенням.

7.3. Метод потенціалів.

Для знаходження оптимального плану транспортної задачі розроюлено кілька методів. Найбільш поширеним із них є метод потенціалів.

До транспортної задачі (7.6) запишемо двоїсту:

$$Q = \bar{y} \cdot \bar{b} \rightarrow \max; \quad \bar{y}A \leq \bar{c} \quad (7.7)$$

Для зручності двоїстий вектор $\bar{y} \in R^{m+n}$ запишемо у вигляді $\bar{y} = (u_1, u_2, \dots, u_m, v_1, v_2, \dots, v_n)$, де перші m компонент u_i відповідають рівнянням (7.2), а наступні n компонент – рівнянням (7.3). Числа u_i, v_j називають потенціалами відповідних пунктів відправлення і споживання. Якщо записати вектори $\bar{y}, \bar{b}, \bar{c}$ і матрицю A задачі (7.7) покомпонентно, то приходимо до двоїстої задачі у вигляді:

$$Q = \sum_{i=1}^m a_i u_i + \sum_{j=1}^n b_j v_j \rightarrow \max; \quad (7.8)$$

$$u_i + v_j \leq c_{ij}, \quad i = \overline{1, m}, \quad j = \overline{1, n}. \quad (7.9)$$

Звідси видно, що збільшення потенціалів u_i і v_j веде до зростання цільової функції (7.8), оскільки $a_i > 0$, $b_j > 0$ для всіх i, j . Проте нерівності (7.9) обмежують ріст цільової функції: сума потенціалів пунктів відправлення і споживання не повинна перевищувати вартість перевезення між ними одиниці продукції. Із другої теореми двоїстості випливає, що оптимальний план прямої задачі може містити значення перевезення $x_{ij} \neq 0$ тільки у тому випадку, якщо сума потенціалів пунктів A_i та B_j дорівнює вартості c_{ij} перевезення між ними одиниці продукції. Застосовуючи цей результат до пари двоїстих задач (7.1) – (7.4) і (7.8) – (7.9), приходимо до ознаки оптимальності плану транспортної задачі.

Теорема 7.1. (критерій оптимальності). Для того, щоб план $X = (x_{ij})$ транспортної задачі був оптимальним, необхідно і достатньо, щоб йому відповідала система із $m + n$ чисел u_i і v_j , для якої виконуються умови:

$$u_i + v_j = c_{ij} \quad \text{при} \quad x_{ij} > 0; \quad (7.10)$$

$$u_i + v_j \leq c_{ij} \quad \text{при} \quad x_{ij} = 0. \quad (7.11)$$

Введемо для змінних x_{ij} оцінки:

$$\Delta_{ij} = u_i + v_j - c_{ij}. \quad (7.12)$$

Тоді із (7.10), (7.11) одержимо критерій оптимальності у такому вигляді: для того, щоб план $X = (x_{ij})$ транспортної задачі був оптимальним, необхідно і достатньо, щоб усі його оцінки Δ_{ij} виду (7.12) були недодатніми.

На сформульованому критерії оптимальності ґрунтується метод потенціалів. Розглянемо алгоритм цього методу. Нехай одним із наведених раніше методів знайдено не вироджений опорний план $X = (x_{ij})$ із $m + n - 1$ додатними компонентами x_{ij} .

1. В кожному із пунктів A_i та B_j вводяться відповідні потенціали u_i та v_j . Їхні значення визначаються із системи $m + n - 1$ рівнянь:

$$u_i + v_j = c_{ij}, \quad (7.13)$$

складеної для базисних (заповнених) клітин транспортної таблиці. Оскільки система (7.13) містить $m + n$ невідомих, то одній з невідомих можна надати довільного значення, наприклад $u_1 = 0$. Після цього полідовно визначаються значення решти потенціалів.

2. Для кожної вільної клітини (i, j) за формулою (7.12) обчислюються оцінки Δ_{ij} . Якщо серед чисел Δ_{ij} немає додатних, то знайдений опорний план є оптимальним. Якщо хоча б для однієї вільної клітини $\Delta_{ij} > 0$, то план не є оптимальним; переходять до пункту 3.

3. Серед оцінок $\Delta_{ij} > 0$ вибирають максимальну: $\max_{\Delta_{ij} > 0} \{\Delta_{ij}\} = \Delta_{kl}$; вільна змінна x_{kl} вводиться до базису. Для цього у транспортній таблиці виділяють цикл, одна з вершин якого міститься у вільній клітині (k, l) , а решта – у базисних. При цьому вільна клітина (k, l) позначається знаком "+", а всі інші – поперемінно знаками "-" і "+". Такий означений цикл називається циклом перерахунку. У від'ємних вершинах такого циклу визначається найменше з чисел x_{ij} : $\min x_{ij} = \theta$. Це число θ додається до елементів у додатних клітинах і одночасно віднімається від чисел у від'ємних клітинах.

Всі інші елементи таблиці перевезень лишають без змін. В результаті таких перетворень одержують новий опорний план і перевіряють його на оптимальність (повертаються до пункту 1).

Операцію переходу від одного опорного плану транспортної задачі до іншого опорного плану називають зсувом по циклу перерахунку. Зазначимо, що при зсуві по циклу перерахунку число базисних клітин лишається незмінним. Якщо мінімальне значення θ зустрічається в кількох вільних клітинах, то звільнюють лише одну, а решта вважаються заповненими нульовими перевезеннями.

Процес повторюється доти, поки розв'язок не стане оптимальним. Для оптимального плану обчислюється мінімальне значення цільової функції (7.1).

7.4. Відкрита модель транспортної задачі.

Якщо умова балансу (7.5) не виконується, то маємо відкриту модель транспортної задачі. Для відкритої моделі можливі два випадки:

$$\sum_{i=1}^m a_i < \sum_{j=1}^n b_j ; \quad (7.14)$$

$$\sum_{i=1}^m a_i > \sum_{j=1}^n b_j ; \quad (7.15)$$

Нехай виконується умова (7.14). Тоді математична модель транспортної задачі набуває вигляду:

$$\begin{aligned} F &= \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} \rightarrow \min ; \\ \sum_{j=1}^n x_{ij} &= a_i, i = \overline{1, m} ; \\ \sum_{i=1}^m x_{ij} &\leq b_j, j = \overline{1, n} ; \\ x_{ij} &\geq 0, i = \overline{1, m}, j = \overline{1, n} . \end{aligned} \quad (7.16)$$

Для розв'язування цієї задачі потрібно ввести фіктивний $(m + 1)$ -й пункт відправлення A_{m+1} із запасом:

$$a_{m+1} = \sum_{j=1}^n b_j - \sum_{i=1}^m a_i$$

одиниць продукції і вважати $c_{m+1 j} = 0, j = \overline{1, n}$. В результаті приходимо до розширеної задачі, для якої виконується умова балансу.

Якщо виконується умова (7.15), то маємо транспортну задачу:

$$\begin{aligned} F &= \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} \rightarrow \min ; \\ \sum_{j=1}^n x_{ij} &\leq a_i, i = \overline{1, m} ; \\ \sum_{i=1}^m x_{ij} &= b_j, j = \overline{1, n} ; \end{aligned}$$

$$x_{ij} \geq 0, \quad i = \overline{1, m}, \quad j = \overline{1, n}. \quad (7.17)$$

У цьому випадку вводиться фіктивний $(n + 1)$ -й пункт споживання B_{n+1} , потреба якого становить:

$$b_{n+1} = \sum_{i=1}^m a_i - \sum_{j=1}^n b_j$$

одиниць продукції і приймаються $c_{i, n+1} = 0, \quad i = \overline{1, m}$.

Одержуємо розширену задачу з m пунктами відправлення і $(n + 1)$ -м пунктом споживання, для якої виконується умова балансу.

Зауважимо, що до моделі транспортної задачі зводяться деякі економічні задачі, які не мають нічого спільного із перевезенням продукції. Тому в залежності від конкретної задачі коефіцієнти c_{ij} можуть набувати різного змісту: вартість, відстань, час продуктивність праці тощо.

Завдання для самоконтролю.

1. Записати математичну модель транспортної задачі ЛП.
2. Яка модель називається закритою, яка відкритою? Як перейти від відкритої моделі до закритої?
3. Які існують методи побудови початкового опорного плану?
4. Скільки додатних компонент містить невироджений опорний план?
5. Що називають циклом в таблиці перевезень?
6. Опишіть алгоритм методу потенціалів.
7. Побудувати опорний план перевезень транспортної задачі, умова якої задана табл. 7.5, методом: а) північно-західного кута; б) мінімального елемента; і знайти значення цільової функції для кожного з цих планів.

Таблиця 7.5

Постачальники	Споживачі				Запас вантажу a_i
	B_1	B_2	B_3	B_4	
A_1	4	7	2	3	30
A_2	3	1	0	4	190
A_3	5	6	3	7	250
Потреба у вантажі b_j	70	120	150	130	470

(Відповідь. а) $X_1 = \begin{pmatrix} 30 & 0 & 0 & 0 \\ 40 & 120 & 30 & 0 \\ 0 & 0 & 120 & 130 \end{pmatrix}$.

Цьому плану відповідає значення цільової функції:

$$F_1 = 4 \cdot 30 + 3 \cdot 40 + 1 \cdot 120 + 0 \cdot 30 + 3 \cdot 120 + 7 \cdot 130 = 1630;$$

б) $X_2 = \begin{pmatrix} 0 & 0 & 0 & 30 \\ 0 & 40 & 150 & 0 \\ 70 & 80 & 0 & 100 \end{pmatrix}$.

Цей розв'язок є невиродженим. Йому відповідають транспортні затрати:

$$F_2 = 3 \cdot 30 + 1 \cdot 40 + 0 \cdot 150 + 5 \cdot 70 + 6 \cdot 80 + 7 \cdot 100 = 1660.)$$

8. Знайти оптимальний план перевезень транспортної задачі, заданої в табл. 7.6.

Таблиця 7.6

Постачальники	Споживачі				Запас вантажу a_i
	B_1	B_2	B_3	B_4	
A_1	6	4	2	7	40
A_2	8	10	14	12	36
A_3	16	12	6	13	24
Потреба у вантажі b_j	24	20	30	26	100

$$(Відповідь. \dots X_{opt} = \begin{pmatrix} 0 & 20 & 6 & 14 \\ 24 & 0 & 0 & 12 \\ 0 & 0 & 24 & 0 \end{pmatrix})$$

Відповідне значення цільової функції:

$$F(X_{opt}) = 4 \cdot 20 + 2 \cdot 6 + 7 \cdot 14 + 8 \cdot 24 + 12 \cdot 12 + 6 \cdot 24 = 670).$$

9. Знайти оптимальний план перевезень транспортної задачі, заданої в табл. 7.7.

Таблиця 7.7

Постачальники	Споживачі				Запас вантажу a_i
	B_1	B_2	B_3	B_4	
A_1	4	10	5	3	30
A_2	6	7	2	8	100
A_3	8	9	12	11	70
Потреба у вантажі b_j	50	55	70	45	

$$(Відповідь. \dots X_{opt} = \begin{pmatrix} 15 & 0 & 0 & 45 & 0 \\ 30 & 0 & 70 & 0 & 0 \\ 5 & 55 & 0 & 0 & 10 \end{pmatrix})$$

зі значенням цільової функції $F(X_{opt}) = 1050$ грош. од.).

10. Знайти оптимальний план перевезень транспортної задачі, заданої в табл. 7.8.

Таблиця 7.8

Постачальники	Споживачі				Запас вантажу a_i
	B_1	B_2	B_3		
A_1	2	3	1		45
A_2	4	1	5		130
A_3	2	8	9		90
Потреба у вантажі b_j	70	115	80		

(Відповідь. $X_{opt} = \begin{pmatrix} 0 & 0 & 45 \\ 0 & 115 & 15 \\ 70 & 0 & 20 \end{pmatrix}$)

зі значенням цільової функції $F(X_{opt}) = 555$ грош. од.).

Лекція № 8. Дискретне програмування.

Моделі задач цілочислового лінійного програмування.

Метод відтинання.

Метод гілок і меж.

8.1. Моделі задач цілочислового лінійного програмування.

До задач дискретного програмування належать такі задачі математичного програмування, в яких потрібно знайти оптимальне значення цільової функції $F(\bar{x})$, визначеної на деякій дискретній множині $G = \{\bar{x}^1, \bar{x}^2, \dots, \bar{x}^s\}$. Частинним випадком таких задач є задача лінійного цілочислового програмування, в якій на всі або на частину змінних накладено додаткову вимогу цілочисловості. Це спостерігається у тих задачах, в яких за економічним або фізичним змістом змінні (наприклад, кількість виробленої продукції, машин, станків, людей) є неподільними величинами.

Математична модель задачі лінійного програмування має вигляд:

$$F = \bar{c} \cdot \bar{x} \rightarrow \max(\min); \quad (8.1)$$

$$A\bar{x} = \bar{b}, \bar{x} \geq 0, \bar{x} \in R^n; \quad (8.2)$$

$$x_j - \text{цілі числа}, j = 1, 2, \dots, l; l \leq n. \quad (8.3)$$

Якщо $l = n$, то задача (8.1) – (8.3) називається повністю цілочисловою, якщо $l < n$ – частково цілочисловою.

Розглянемо деякі приклади конкретних моделей лінійного цілочислового програмування.

1. Транспортна задача з неподільною продукцією.

Це задача про перевезення (див. лекцію 4), в якій запаси $a_i, i = \overline{1, m}$, попит $b_j, j = \overline{1, n}$ є цілочисловими величинами. За змістом у цьому випадку змінні x_{ij} також мають бути цілочисловими.

2. Задача про призначення. Нехай для виконання n різних робіт виділено n претендентів, причому з кожною роботою можна закріпити лише одного претендента. Відома ефективність c_{ij} виконання i -тої роботи претендентом j . Потрібно так закріпити претендентів за видами робіт, щоб загальна ефективність була максимальною.

Введемо булівські змінні:

$$x_{ij} = \begin{cases} 1 - \text{робота } i \text{ закріплена за претендентом } j, \\ 0 - \text{у протилежному випадку.} \end{cases}$$

Математична модель задачі має вигляд:

$$F = \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij} \rightarrow \max; \quad (8.4)$$

$$\sum_{j=1}^n x_{ij} = 1, i = \overline{1, n}; \quad \sum_{i=1}^n x_{ij} = 1, j = \overline{1, n}; \quad (8.5)$$

$$x_{ij} \in \{0; 1\}. \quad (8.6)$$

Перше обмеження (8.5) означає, що на кожну роботу i призначено одного претендента, друге – кожен претендент j закріплений за однією роботою.

3. Задача про розподіл капіталовкладень. Існує n незалежних між собою проектів (способів) використання суми S капіталовкладень. Для кожного проекту відомі затрати a_j і прибуток c_j від його реалізації, $j = \overline{1, n}$. Наявної суми капіталовкладень недостатньо для здійснення усіх проектів. Потрібно визначити перелік проектів, які забезпечать максимальний прибуток при їх реалізації.

Введемо булівські змінні $x_j = 1$, якщо вибирається проект j , і $x_j = 0$ – у протилежному випадку.

Тоді математична модель задачі буде такою:

$$F = \sum_{i=1}^n \sum_{j=1}^n c_j x_j \rightarrow \max ; \quad (8.7)$$

$$\sum_{j=1}^n a_j x_j \leq S ; \quad (8.8)$$

$$x_j \in \{0;1\}, \quad j = \overline{1, n}. \quad (8.9)$$

Видно, що наведені задачі цілочислового програмування відрізняються від задачі лінійного програмування лише наявністю додаткової умови (8.3). Тому виникає допущення, що цілочислові розв'язки можна отримати, розв'язуючи відповідні задачі лінійного програмування. Якщо оптимальний план задачі лінійного програмування виражається у цілих числах, то він буде також розв'язком цілочислової задачі. Таку властивість мають

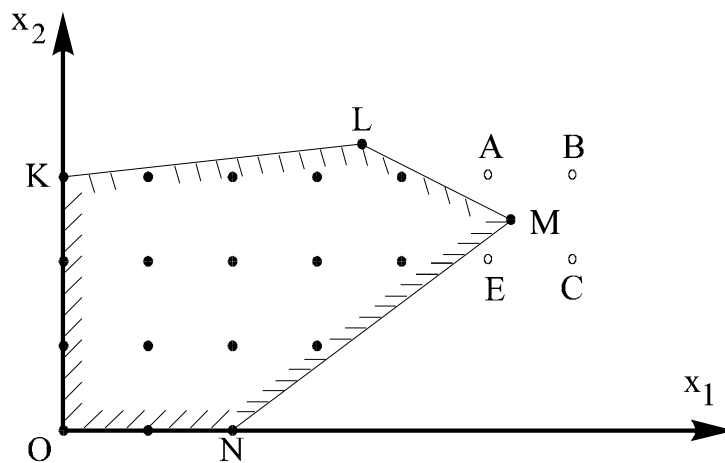


Рис. 8.1.

транспортні задачі з цілочисловими умовами. Іноді вдається знайти оптимальний план цілочислової задачі, заокругливши дробові значення компонент до найближчих цілих чисел; наприклад, якщо одиниця продукції становить малу частину

загального обсягу випуску цієї продукції. У загальному випадку заокруглення компонент нецілочислового розв'язку може не тільки привести до планів, далеких від оптимальних, а й вивести за межі допустимої області. Це пов'язано із дискретністю множини допустимих розв'язків цілочислових задач. Наприклад, нехай оптимальне значення задачі лінійного програмування досягається у точці M опуклого многокутника $OKLMN$ (рис. 8.1).

Заокруглення компонент розв'язку до найближчих цілих чисел приводить до точок A, B, C, E, жодна з яких не належить множині допустимих планів задачі.

Отже, умова цілочисловості суттєво ускладнює задачу математичного програмування і вимагає застосування спеціальних методів пошуку цілочислових розв'язків. Усі відомі методи розв'язування задач лінійного цілочислового програмування умовно можна поділити на три групи: методи відтинання; комбінаторні; наближені.

8.2. Метод відтинання.

Розглянемо повністю числову задачу для $\bar{x} \in R^n$:

$$F = \bar{c} \cdot \bar{x} \rightarrow \max; \quad (8.10)$$

$$A\bar{x} = \bar{b}; \quad (8.11)$$

$$\bar{x} \geq 0; \quad (8.12)$$

$$x_j - \text{ціле}, \quad j = \overline{1, n}, \quad (8.13)$$

де всі коефіцієнти a_{ij} , b_i , c_j – цілі числа.

Відомо, що задача лінійного програмування (8.10) – (8.12) має опуклу множину планів D . Якщо накласти вимогу цілочисловості (8.13), то множина допустимих розв'язків задачі (8.10) – (8.13). Всі крайні точки оболонки $D' \subset D$ цілочислові. Очевидно, що оптимальне значення цільової функції (8.10) на множині співпадає з оптимальним розв'язком цілочислової задачі (8.10) – (8.13). Це дає змогу використовувати методи розв'язання задач лінійного програмування для розв'язування задач лінійного цілочислового програмування.

Методи відтинання дають спосіб побудови оболонки D' шляхом послідовного відкидання деяких підмножин області D , які не містять цілочислових точок. Один із таких методів запропонував у 1958 році американський математик Р. Гоморі. Ідея методу Гоморі для повністю цілочислової задачі лінійного програмування полягає в наступному. Спочатку задача (8.10) – (8.12) розв'язується симплексним методом. Якщо одержаний оптимальний план є цілочисловим, то він буде розв'язком задачі (8.10) – (8.13). Якщо оптимальний розв'язок задачі (8.10) – (8.12) не задовільняє умову (8.13), то складається додаткове лінійне обмеження, яке відтинає від D частину області, в якій немає планів задачі (8.10) – (8.13), але є знайдений оптимальний план задачі (8.10) – (8.12). Одержана розширена задача лінійного програмування розв'язується двоїстим симплекс-методом. Процес введення додаткових обмежень і розв'язування розширених задач повторюють доти, поки не одержать цілочисловий розв'язок.

Розглянемо детальніше алгоритм Гоморі.

Нехай задача (8.10) – (8.12) розв'язана симплекс-методом. Для простоти будемо вважати, що оптимальний план $\bar{x}^*$ має вигляд $\bar{x}^* = (b_1^*, b_2^*, \dots, b_m^*, 0, \dots, 0)$, тобто на останній ітерації система рівнянь (8.11) приведена до такого базисного вигляду:

$$x_i = b_i^* - \sum_{j=m+1}^n a_{ij}^* x_j, \quad i = 1, 2, \dots, m. \quad (8.14)$$

Якщо всі b_i^* – цілі ($i = \overline{1, m}$), то знайдений оптимальний план є розв'язком задачі (8.10) – (8.13); задача розв'язана.

Цілою частиною числа a називається найбільше ціле число, яке не перевищує a ; позначається $[a]$.

$$\text{Наприклад, } \left[\frac{8}{3} \right] = 2, [4] = 4, [-2, 4] = -3.$$

Дробовою частиною числа a називають число $\langle a \rangle = a - [a]$.

$$\text{Наприклад, } \left\langle \frac{8}{3} \right\rangle = \frac{8}{3} - 2 = \frac{2}{3}; \langle 4 \rangle = 4 - [4] = 0; \langle -2, 4 \rangle = -2, 4 - (-3) = 0, 6.$$

Очевидно, що $0 \leq \langle a \rangle \leq 1$.

Нехай серед компонент оптимального плану $\bar{x}^*$ є дробові. Виберемо серед них число b_k^* з найменшою дробовою частиною; серед коефіцієнтів a_{kj}^* , $j = m+1, \dots, n$, також є дробові. Позначимо через $[a_{kj}^*]$ і $[b_k^*]$ цілі частини чисел a_{kj}^* і b_k^* , а $\langle a_{kj}^* \rangle$, $\langle b_k^* \rangle$ – їхні дробові частини. Враховуючи введені позначення, k -те рівняння системи (8.14) запишемо у вигляді:

$$x_k = [b_k^*] + \langle b_k^* \rangle - \sum_{j=m+1}^n ([a_{kj}^*] + \langle a_{kj}^* \rangle) x_j$$

або

$$x_k + \sum_{j=m+1}^n [a_{kj}^*] x_j - [b_k^*] = \langle b_k^* \rangle - \sum_{j=m+1}^n \langle a_{kj}^* \rangle x_j. \quad (8.15)$$

Якщо $\bar{x} = (x_1, \dots, x_n)$ – цілочисловий допустимий розв'язок задачі (8.10) – (8.13), то ліва частина рівняння (8.15) завжди буде цілочисловою. Тому і права частина цього рівняння повинна бути цілочисловою. Крім того, оскільки $\langle a_{kj}^* \rangle \geq 0$ і $x_j \geq 0$, то $\sum_{j=m+1}^n \langle a_{kj}^* \rangle x_j \geq 0$; звідки маємо нерівність:

$$\langle b_k^* \rangle - \sum_{j=m+1}^n \langle a_{kj}^* \rangle x_j \leq \langle b_k^* \rangle.$$

Враховуючи цілочисловість правої частини рівняння (8.15) і оцінку $0 \leq \langle b_k^* \rangle < 1$, приходимо до обмеження Гоморі:

$$\langle b_k^* \rangle - \sum_{j=m+1}^n \langle a_{kj}^* \rangle x_j \leq 0, \quad (8.16)$$

якому задовольняють усі допустимі розв'язки задачі (8.10) – (8.13).

Покажемо, що компоненти оптимального нецілочислового плану $\bar{x}^*$ задачі (8.10) – (8.12) не задовольняють обмеження (8.16). Дійсно, оскільки $\langle b_k^* \rangle > 0$, $x_j^* = 0$, $j = m+1, \dots, n$, то нерівність (8.16) не виконується.

Отже, лінійне обмеження (8.16) відтинає від множини допустимих розв'язків частину області, в якій немає жодного допустимого цілочислового плану.

В обмеження (8.16) введемо додаткову змінну $x_{n+1} \geq 0$ і запишемо відтинання у вигляді рівняння:

$$-\sum_{j=m+1}^n \langle a_{kj}^* \rangle x_j + x_{n+1} = -\langle b_k^* \rangle \quad (8.17)$$

з базисною змінною x_{n+1} .

Приєднаємо рівняння відтинання (8.17) до обмежень (8.14) попередньої задачі. Одержану розширену задачу розв'язуємо двоїтим симплекс-методом. Якщо розв'язок виявиться дробовим, то вводимо нове додаткове обмеження виду (8.17). Процес повторюємо доти, поки розв'язок не стане цілочисловим або не виявиться, що задача не має розв'язку.

Приклад 8.1. Керівництво малого підприємства планує створити нову лінію випуску продукції харчування. Підприємство спроможне виділити до 18 тис. грн. на придбання обладнання двох типів A та B і відвести до 40 кв.м площі під нову лінію. Відомо, що обладнання A коштує 3 тис. грн, потребує 5 кв. м площі і випускає 6 тис. од. продукції за зміну, а обладнання B коштує 2 тис. грн, потребує 6 кв. м площі і випускає 7 тис. од. продукції.

Визначити оптимальний варіант придбання обладнання, при якому буде максимальним обсяг випуску продукції підприємством.

Розв'язання. Позначимо через x_1 і x_2 кількість одиниць обладнання відповідно типу A та B . Математична модель задачі має вигляд:

$$F = 6x_1 + 7x_2 \rightarrow \max ;$$

$$\begin{cases} 3x_1 + 2x_2 \leq 18, \\ 5x_1 + 6x_2 \leq 40; \end{cases}$$

$$x_1 \geq 0, x_2 \geq 0; \quad x_1, x_2 - \text{цілі.}$$

Розглянемо канонічну форму задачі лінійного програмування без вимоги цілочисловості змінних:

$$F = 6x_1 + 7x_2 \rightarrow \max ;$$

$$\begin{cases} 3x_1 + 2x_2 + x_3 = 18, \\ 5x_1 + 6x_2 + x_4 = 40; \end{cases}$$

$$x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, x_4 \geq 0.$$

Розв'яжемо її симплексним методом. Результати обчислень наведені в табл.

8.1. Видно, що оптимальним розв'язком даної задачі є вектор $\bar{x}^* = \left(3\frac{1}{2}; 3\frac{3}{4}; 0; 9\right)$ і $F_{\max} = \frac{189}{4}$. Оскільки значення $x_1^* = b_1^* = 3\frac{1}{2}$, $x_2^* = b_2^* = 3\frac{3}{4}$ дробові, то знайдений план не є оптимальним планом цілочислової задачі.

Таблиця 8.1

$\bar{C}_{\text{баз}}$	$x_{\text{баз}}$	$\bar{b}$	6	7	0	0
			$\bar{A}_1$	$\bar{A}_2$	$\bar{A}_3$	$\bar{A}_4$
0	x_3	18	3	2	1	0
0	x_4	40	5	6	0	1
Δ		0	-6	-7	0	0
0	x_3	$14\frac{2}{3}$	$4\frac{2}{3}$	0	1	$-\frac{1}{3}$

7	x_2	$\frac{40}{6}$	$\frac{5}{6}$	1	0	$\frac{1}{6}$
Δ		$\frac{140}{3}$	$-\frac{1}{6}$	0	0	$\frac{7}{6}$
6	x_1	$\frac{7}{2}$	1	0	$\frac{3}{4}$	$-\frac{1}{4}$
7	x_2	$\frac{15}{4}$	0	1	$-\frac{5}{8}$	$\frac{3}{8}$
Δ		$\frac{189}{4}$	0	0	$\frac{1}{8}$	$\frac{9}{8}$

Згідно з алгоритмом Гоморі знаходимо:

$$\min\{\langle x_1^* \rangle; \langle x_2^* \rangle\} = \min\left\{\left\langle 3\frac{1}{2} \right\rangle; \left\langle 3\frac{3}{4} \right\rangle\right\} = \min\left\{\frac{1}{2}; \frac{3}{4}\right\} = \frac{1}{2} = \langle x_1^* \rangle.$$

Тому відтинання складаємо за змінною x_1 ; За першим рядком останньої ітерації табл. 8.1 маємо обмеження:

$$\left\langle \frac{3}{4} \right\rangle x_3 + \left\langle -\frac{1}{4} \right\rangle x_4 \geq \left\langle \frac{7}{2} \right\rangle, \text{ тобто } \frac{3}{4}x_3 + \frac{3}{4}x_4 \geq \frac{1}{2}.$$

Ввівши додаткову змінну $x_5 \geq 0$, запишемо обмеження у вигляді:

$$\left\langle \frac{3}{4} \right\rangle x_3 + \left\langle -\frac{1}{4} \right\rangle x_4 + x_5 = -\frac{1}{2}.$$

Таблиця 8.2

$\overline{C}_{\text{баз}}$	$x_{\text{баз}}$	$\overline{b}$	6	7	0	0	0
			$\overline{A}_1$	$\overline{A}_2$	$\overline{A}_3$	$\overline{A}_4$	$\overline{A}_5$
6	x_1	$\frac{7}{2}$	1	0	$\frac{3}{4}$	$-\frac{1}{4}$	0
7	x_2	$\frac{15}{4}$	0	1	$-\frac{5}{8}$	$\frac{3}{8}$	0
0	x_5	$-\frac{1}{2}$	0	0	$-\frac{3}{4}$	$-\frac{3}{4}$	1
Δ		$\frac{189}{4}$	0	0	$\frac{1}{8}$	$\frac{9}{8}$	0

Доповнимо цим обмеженням останню симплекс-таблицю; приходимо до табл. 8.2 з базисними змінними x_1, x_2, x_5 . Одержану розширену задачу розв'язуємо двоїстим симплекс-методом:

а) вибираємо рядок r , в якому $b_r^* < 0$;

б) проглядаємо від'ємні коефіцієнти a_{rl}^* , для якого: $\frac{\Delta_l}{a_{rl}^*} = \max\left\{\frac{\Delta_j}{a_{rj}^*}\right\}$;

в) у симплексній таблиці виконуємо перетворення Жордана-Гауса з розв'язувальним елементом a_{rl}^* .

Згідно з правилами двоїстого симплекс-методу змінна x_3 вводиться до базису замість змінної x_5 ; $a_{33}^* = -\frac{3}{4}$ – розв'язувальний елемент. Одержимо табл. 8.3, звідки видно, що в оптимальному плані $\bar{x}^* = \left(3; \frac{25}{6}; \frac{2}{3}; 0; 0\right)$

значення $x_2^* = b_2^* = 25/6$ ще дробове. Змінна x_2 приводить до лінійного обмеження виду:

$$\left\langle -\frac{5}{6} \right\rangle x_5 \geq \left\langle \frac{25}{6} \right\rangle \text{ або } \frac{1}{6} x_5 \geq \frac{1}{6}.$$

Таблиця 8.3

$\overline{C}_{\text{баз}}$	$x_{\text{баз}}$	$\overline{b}$	6	7	0	0	0
			$\overline{A}_1$	$\overline{A}_2$	$\overline{A}_3$	$\overline{A}_4$	$\overline{A}_5$
6	x_1	3	1	0	0	-1	1
7	x_2	$25/6$	0	1	0	1	$-5/6$
0	x_3	$2/32$	0	0	1	1	$-4/3$
Δ		$283/6$	0	0	0	1	$1/6$

Після введення додаткової змінної $x_6 \geq 0$ відтинання матиме вигляд:

$$-x_5/6 + x_6 = -1/6.$$

Таблиця 8.4

$\overline{C}_{\text{баз}}$	$x_{\text{баз}}$	$\overline{b}$	6	7	0	0	0	0
			$\overline{A}_1$	$\overline{A}_2$	$\overline{A}_3$	$\overline{A}_4$	$\overline{A}_5$	$\overline{A}_6$
6	x_1	3	1	0	0	-1	1	0
7	x_2	$25/6$	0	1	0	1	$-5/6$	0
0	x_3	$2/32$	0	0	1	1	$-4/3$	0
0	x_6	$-1/6$	0	0	0	0	$-1/6$	1
Δ		$283/6$	0	0	0	1	$1/6$	0
6	x_1	2	1	0	0	-1	0	6
7	x_2	5	0	1	0	1	0	-5
0	x_3	2	0	0	1	1	0	-8
0	x_5	1	0	0	0	0	1	-6
Δ		47	0	0	0	1	0	1

Доповнимо цим рівнянням табл. 8.3, одержимо табл. 8.4. За допомогою двоїстого симплекс-методу вводимо до базису змінну x_5 замість змінної x_6 (табл. 8.4).

Із таблиці 8.4 видно, що $\overline{x}^* = (2; 5; 2; 0; 1; 0)$ – оптимальний план розширеної задачі лінійного програмування. Оскільки компоненти $x_1^* = 2$, $x_2^* = 5$ цього вектора цілі числа, то маємо оптимальний розв'язок вихідної цілочислової задачі: $x_1 = 2$, $x_2 = 5$, $F_{\max} = 47$ тис. од. продукції.

Дано геометричне тлумачення методу Гоморі. Многокутник $OABC$ розв'язків вихідної нецілочислової задачі лінійного програмування зображено на рис.8.2.

Видно, що максимум цільової функції досягається в точці

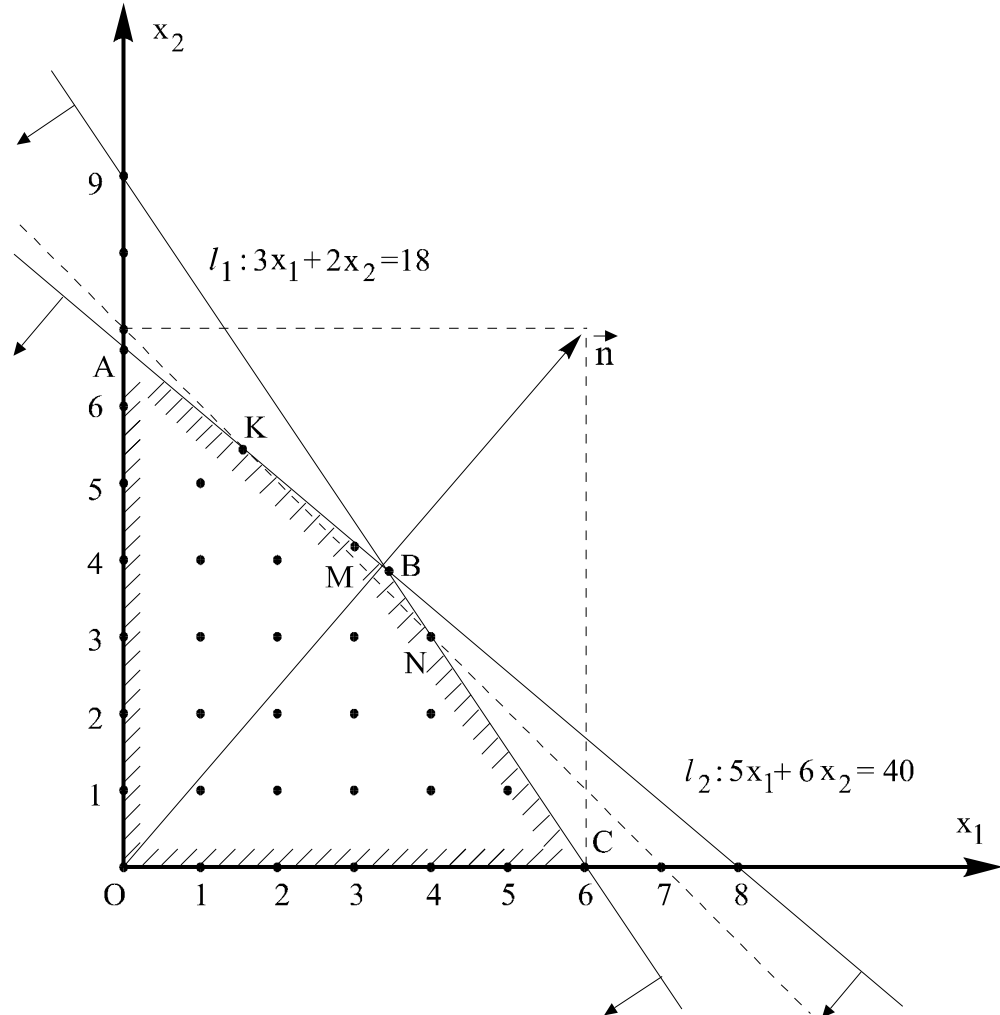


Рис. 8.2

$B\left(3\frac{1}{2}; 3\frac{3}{4}\right)$, координати якої не є цілочисловими. Для знаходження цілочислового розв'язку ввели два додаткові обмеження, які можна записати у вигляді:

$$x_5 = -\frac{1}{2} + \frac{3}{4}x_3 + \frac{3}{4}x_4 \quad \text{і} \quad x_6 = -\frac{1}{6} + \frac{x_5}{6}.$$

Виразимо змінні x_5 і x_6 через змінні x_1, x_2 :

$$x_5 = -\frac{1}{2} + \frac{3}{4}(18 - 3x_1 - 2x_2) + \frac{3}{4}(40 - 5x_1 - 6x_2) = 43 - 6x_1 - 6x_2;$$

$$x_6 = -\frac{1}{6} + \frac{1}{6}(43 - 5x_1 - 6x_2) = 7 - x_1 - x_2.$$

Оскільки $x_5 \geq 0$ і $x_6 \geq 0$, то відтинання будуть мати вигляд нерівностей

$$6x_1 + 6x_2 \leq 43 \quad \text{і} \quad x_1 + x_2 \leq 7.$$

Ці обмеження в сумі відтинають від многокутника розв'язків частину області – трикутник BKN разом із точками B та $M\left(3; 4\frac{1}{6}\right)$, які мають дробові координати. У вершині $K(2; 5)$ одержаного многокутника $OAKNC$ цільова функція досягає максимального значення.

8.3. Метод віток і меж.

Метод Гоморі не враховує специфічних особливостей конкретної задачі і має невисоку ефективність із-за великої кількості обчислень. Таких недоліків позбавлений метод віток і меж. Вперше цей метод запропонували Ленд і Дойг в 1960 р.

Метод віток і меж належить до групи методів, які враховують комбінаторний характер задачі, дискретність окремих змінних. Він ґрунтується на впорядкованому виявленні серед них перспективних за певними ознаками і відкиданні цілої множини безперспективних варіантів.

Розглянемо загальну ідею методу віток і меж загальної задачі дискретного програмування:

$$F(\bar{x}) \rightarrow \max; \quad (8.18)$$

$$\bar{x} \in G, \quad (8.19)$$

де $G \subset \mathbb{R}^n$ – скінченна множина.

1. Знаходимо верхню оцінку (межу значень цільової функції на множині G , тобто число $\varphi(G)$ таке, що $F(\bar{x}) \leq \varphi(G)$, для всіх $\bar{x} \in G$. Якщо при цьому знайдено такий план $\bar{x}^*$ задачі (8.18) – (8.19), для якого виконується рівність $F(\bar{x}^*) = \varphi(G)$, то $\bar{x}^*$ – оптимальний розв'язок задачі.

2. Якщо оптимальний план не знайдений, то деяким способом множину G розбиваємо (проводимо розгалуження) на скінченне число підмножин $G_1^1, G_2^1, \dots, G_r^1$ (рис. 8.3) таких, що:

$$G = G_1^1 \cup G_2^1 \cup \dots \cup G_r^1, \quad G_1^1 \cap G_2^1 \cap \dots \cap G_r^1 = \emptyset.$$

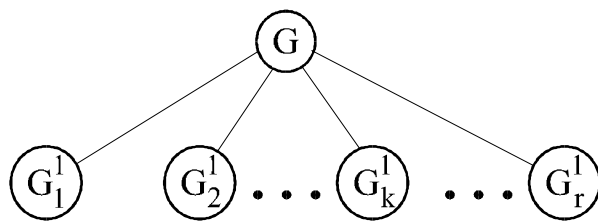


Рис. 8.3

Для кожної із цих підмножин обчислюємо верхню оцінку $\varphi(G_i^1)$ функції $F(\bar{x})$: $F(\bar{x}) \leq \varphi(G_i^1)$, $\bar{x} \in G_i^1$, $i = 1, \dots, r$. Якщо при цьому знаходимо такий план $\bar{x}^* \in G_k^1$, $1 \leq k \leq r$, для якого виконуються співвідношення $F(\bar{x}^*) = \varphi(G_k^1) \geq \varphi(G_i^1)$, $i = \overline{1, r}$, то $\bar{x}^*$ –

оптимальний план задачі (8.18) – (8.19).

3. Якщо такий план не знайдений, то вибираємо підмножину G_k^1 з найбільшою оцінкою:

$$\varphi(G_k^1) = \max_{1 \leq i \leq r} \varphi(G_i^1).$$

Множину G_k^1 розбиваємо на декілька підмножин, які не перетинаються:

$$G_k^1 = G_{k1}^1 \cup G_{k2}^1 \cup \dots \cup G_{ks}^1.$$

Ці s підмножин разом із рештою $r - 1$ підмножинами перепозначимо через G_i^2 , $i = 1, 2, \dots, r + s - 1$ (рис. 8.4) і для кожної з них знайдемо оцінку $\varphi(G_i^2)^1$ (оцінки обчислюються тільки для новоутворених підмножин, оскільки для решти підмножин оцінки знайдено раніше). Якщо знаходимо такий план $\bar{x}^* \in G_l^2$, $1 \leq l \leq r + s - 1$, що $F(\bar{x}^*) = \varphi(G_l^2) \geq \varphi(G_i^2)$ для $i = 1, 2, \dots, r + s - 1$, то $\bar{x}^*$ – оптимальний розв'язок задачі (8.18) – (8.19).

4. Якщо оптимальний план не знайдений, то вибираємо підмножину з найбільшою оцінкою і виконуємо розгалуження. Процес продовжують до

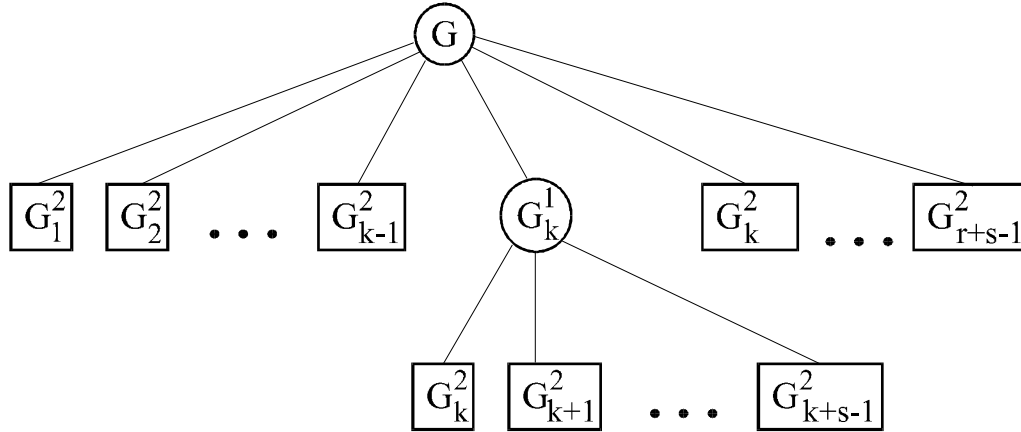


Рис. 8.4

тих пір, поки не одержать оптимальний план.

Правила розгалуження, способи обчислення оцінок і знаходження планів вибираються в залежності від специфіки конкретної задачі дискретного програмування. Зокрема, для задачі лінійного цілочислового програмування (8.1) – (8.3) за оцінку цільової функції $F = \bar{c} \cdot \bar{x}$ беруть її значення при оптимальному плані відповідної задачі лінійного програмування (8.1) – (8.2).

Позначимо через G множину планів задачі на максимум (8.1) – (8.3), а через $\bar{x}^* = (x_1^*, x_2^*, \dots, x_n^*)$ – оптимальний план задачі (8.1) – (8.2), тоді $\varphi(G) = F(\bar{x}^*)$. Якщо перші l компонент плану $\bar{x}^*$ є цілочисловими, то він буде оптимальним розв'язком задачі (8.1) – (8.3).

Нехай серед перших l компонент плану $\bar{x}^*$ є дробові, наприклад, значення $\bar{x}_k^*$ – нецілочислове, $1 \leq k \leq l$. В цьому випадку множину планів G задачі (8.1) – (8.3) розбивають на дві підмножини G_1^1 і G_2^1 , де

$$G_1^1 = \{\bar{x} | \bar{x} \in G, x_k \leq \lfloor x_k^* \rfloor\}, \quad G_2^1 = \{\bar{x} | \bar{x} \in G, x_k \geq \lfloor x_k^* \rfloor + 1\}.$$

Далі продовжують розв'язувати дві задачі (8.1) – (8.3), перша з яких доповнена умовою $x_k \leq \lfloor x_k^* \rfloor$, а друга – умовою $x_k \geq \lfloor x_k^* \rfloor + 1$.

Приклад.8.2. Методом віток і меж розв'язати задачу:

$$F = 3x_1 + 4x_2 \rightarrow \max;$$

$$\begin{cases} 3x_1 + 2x_2 \leq 7, \\ 9x_1 + 8x_2 \leq 45; \end{cases}$$

$$x_1 \geq 0, x_2 \geq 0, x_1, x_2 - \text{цілі.} \quad (8.20)$$

Розв'язання. Множину планів цієї задачі позначимо через G . Знайдемо спочатку розв'язок задачі (8.20) без вимоги цілочисловості. Ввівши дві додаткові змінні $x_3 \geq 0, x_4 \geq 0$, одержану канонічну задачу лінійного програмування розв'язуємо симплексним методом (табл. 8.5).

Із табл. 8.5 випливає, що $\bar{x}^* = (3, 4; 1, 8)$ – оптимальний план задачі і $F(\bar{x}^*) = 17,4$. Цей план містить дробові компоненти, тому він не може бути розв'язком цілочислової задачі (8.20).

Згідно методу віток і меж, Значення $F(\bar{x}^*) = 17,4$ беремо з верхню межу (оцінку) $\varphi(G)$ цільової функції задачі (8.20):

$$\varphi(G) = F(\bar{x}^*) = 17,4.$$

Таблиця 8.5

$\bar{C}_{\text{баз}}$	$x_{\text{баз}}$	$\bar{b}$	6	7	0	0
			$\bar{A}_1$	$\bar{A}_2$	$\bar{A}_3$	$\bar{A}_4$
0	x_3	7	1	<u>2</u>	1	0
0	x_4	45	9	8	0	1
Δ		0	-3	-4	0	0
4	x_2	3,5	0,5	1	0,5	0
0	x_4	17	<u>5</u>	0	-4	1
Δ		14	-1	0	2	0
4	x_2	1,8	0	1	0,9	-0,1
3	x_1	3,4	1	0	-0,8	0,2
Δ		17,4	0	0	1,2	0,2

Виберемо дробову компоненту плану $\bar{x}^*$, наприклад $x_1^* = 3,4$, і розіб'ємо множину G на дві підмножини G_1^1 і G_1^2 :

$$G_1^1 = \{\bar{x} | \bar{x} \in G, x_1 \leq [3,4]\} = \{\bar{x} | \bar{x} \in G, x_1 \leq 3\},$$

$$G_1^2 = \{\bar{x} | \bar{x} \in G, x_1 \geq [3,4] + 1\} = \{\bar{x} | \bar{x} \in G, x_1 \geq 4\}.$$

Знайдемо оцінки множин G_1^1 і G_1^2 . Для цього розв'яжемо такі дві задачі лінійного програмування:

$$F = 3x_1 + 4x_2 \rightarrow \max;$$

$$\begin{cases} 3x_1 + 2x_2 \leq 7, \\ 9x_1 + 8x_2 \leq 45, \\ x_1 \leq 3; \end{cases}$$

$$x_1 \geq 0, x_2 \geq 0; \quad (8.21)$$

$$F = 3x_1 + 4x_2 \rightarrow \max;$$

$$\begin{cases} 3x_1 + 2x_2 \leq 7, \\ 9x_1 + 8x_2 \leq 45, \\ x_1 \geq 4; \end{cases}$$

$$x_1 \geq 0, x_2 \geq 0; \quad (8.22)$$

Задачу (8.21) зведемо до канонічного виду і розв'яжемо симплекс-методом (табл. 8.6).

Таблиця 8.6

$\bar{C}_{\text{баз}}$	$x_{\text{баз}}$	$\bar{b}$	3	4	0	0	0
			$\bar{A}_1$	$\bar{A}_2$	$\bar{A}_3$	$\bar{A}_4$	$\bar{A}_5$
0	x_3	7	1	<u>2</u>	1	0	0
0	x_4	45	9	8	0	1	0
0	x_5	3	1	0	0	0	1
Δ		0	-3	-4	0	0	0
4	x_2	3,5	0,5	1	0,5	0	0

0	x_4	17	5	0	-4	1	0
0	x_5	3	1	0	0	0	1
Δ		14	-1	0	2	0	0
4	x_2	2	0	1	0,5	0	-0,5
0	x_4	2	0	0	-4	1	-5
3	x_1	3	1	0	0	0	1
Δ		17	0	0	2	0	1

Із табл. 8.6 видно, що $\bar{x}' = (3; 2)$ – оптимальний план задачі (8.21), тому $\phi(G_1^1) = F(\bar{x}') = 17$.

Задачу (8.22) запишемо у канонічній формі і розв'яжемо M -методом (табл. 8.7). Із табл. 8.7 випливає, що оптимальним розв'язком задачі (8.22) є вектор $\bar{x}'' = (4; \frac{9}{8})$; тому $\phi(G_2^1) = F(\bar{x}'') = 16,5$.

Для плану $\bar{x}' = (3; 2)$ задачі (8.21) виконується співвідношення $\phi(G_1^1) = 17 > \phi(G_2^1) = 16,5$. Оскільки цей план є цілочисловим, то $\bar{x}' = (3; 2)$ – оптимальний розв'язок задачі (8.20) і $F_{\max} = F(\bar{x}') = 17$.

Таблиця 8.7

$\bar{C}_{\text{баз}}$	$x_{\text{баз}}$	$\bar{b}$	3	4	0	0	0	$-M$
			$\bar{A}_1$	$\bar{A}_2$	$\bar{A}_3$	$\bar{A}_4$	$\bar{A}_5$	$\bar{A}_6$
0	x_3	7	1	2	1	0	0	0
0	x_4	45	9	8	0	1	0	0
$-M$	x_6	4	1	0	0	0	-1	1
Δ		$-4M$	$-M-3$	-4	0	1	M	0
0	x_3	3	0	2	1	0	1	-1
0	x_4	9	0	8	0	1	9	-9
3	x_1	4	1	0	0	0	-1	1
Δ		12	0	-4	0	0	-3	$M+3$
0	x_3	$\frac{3}{4}$	0	0	1	$-\frac{1}{4}$	$-\frac{5}{4}$	X
4	x_2	$\frac{9}{8}$	0	1	0	$\frac{1}{8}$	$\frac{9}{8}$	
3	x_1	4	1	0	0	0	-1	
Δ		16,5	0	0	0	0,5	1,5	

Завдання для самоконтролю

1. Навести приклади конкретних задач лінійного цілочислового програмування.

2. Дати коротку характеристику методів розв'язування задач цілочислового програмування.

3. У чому полягає метод Гоморі?

4. Як скласти додаткове обмеження, якщо оптимальний план має дробові компоненти?

5. Який геометричний зміст введеного геометричного обмеження?

6. Опишіть основу ідею методу гілок і меж.

7. Розв'язати методом Гоморі наступну ЗЛП

$$F = x_1 + x_2 \rightarrow \max;$$

$$\begin{cases} 8x_1 - 3x_2 \leq 24, \\ 3x_1 + 2x_2 \leq 13, \end{cases} \quad x_1 \geq 0, \quad x_2 \geq 0, \quad x_1, \quad x_2 - \text{цілі}.$$

(Відповідь. $X_1 = (1; 5)$, $X_2 = (0; 6)$, $F_{\max} = 6$).

8. Розв'язати методом Гоморі наступну ЗЛП

$$F = 2x_1 + 5x_2 \rightarrow \max;$$

$$\begin{cases} 3x_1 + 4x_2 \leq 25, \\ 2x_1 - x_2 \leq 1, \\ x_1 \leq 5. \end{cases} \quad x_1 \geq 0, \quad x_2 \geq 0, \quad x_1, \quad x_2 - \text{цілі}.$$

(Відповідь. $X_{\text{opt}} = (3; 4)$, $F_{\max} = 26$).

9. Розв'язати методом Гоморі наступну ЗЛП

$$F = x_1 - 7x_2 \rightarrow \max ;$$

$$\begin{cases} 7x_1 - x_2 \geq 5, \\ 2x_1 + 3x_2 \leq 11, \\ x_1 \leq 5. \end{cases} \quad x_1 \geq 0, x_2 \geq 0, \quad x_1, x_2 - \text{цілі.}$$

(Відповідь. $X_{opt1} = (5; 0)$, $F_{\max} = 5$).

10. Розв'язати методом Гоморі наступну ЗЛП

$$F = x_1 + 5x_2 \rightarrow \max ;$$

$$\begin{cases} x_1 + x_2 \leq 7, \\ -3x_1 + 2x_2 \leq 5, \\ x_1 \leq 6. \end{cases} \quad x_1 \geq 0, x_2 \geq 0, \quad x_1, x_2 - \text{цілі.}$$

(Відповідь. $X_{opt1} = (2; 5)$, $F_{\max} = 27$).

11. Розв'язати методом гілок і меж. Мале підприємство випускає дві моделі A та B книжкових полиць. Виробництво обмежене наявністю сировини та часом машинної обробки. Для кожного виробу моделі A потрібно 2 кв. м. дощок, а для виробу моделі B – 5 кв. м. дощок. Мале підприємство може одержувати від постачальників до 1600 кв. м. дощок на тиждень. Для виготовленні кожного виробу моделі A потрібно 10 хв. машинної обробки, а виробу B – 12 хв. Час машинної обробки в тиждень 100 год. Скільки виробів кожної моделі слід випускати малому підприємству за тиждень для одержання максимального прибутку якщо кожен виріб моделі A приносить 20 гр. од. прибутку, а кожен виріб моделі B – 40 гр. од. прибутку.

(Відповідь. $X_{opt1} = (415; 154)$, $F_{\max} = 14461$).

Лекція № 9. Нелінійне програмування

Загальна задача нелінійного програмування.

Класична задача нелінійного програмування.

Опуклі і вгнуті функції.

Задача опуклого програмування.

Задача квадратичного програмування.

Градiєнтні методи.

Методи нульового порядку.

Метод штрафних функцій.

9.1. Загальна задача нелінійного програмування.

Загальна задача нелінійного програмування має вигляд:

$$F(x) \rightarrow \max(\min); \quad (9.1)$$

$$\varphi_i(x) \leq 0, \quad i = 1, 2, \dots, m, \quad (9.2)$$

де або цільова функція $F(x)$, або хоча б одна із функцій $\varphi_i(x)$ – нелінійні; $x = (x_1, x_2, \dots, x_n) \in R^n$. В конкретних задачах деякі із обмежень (9.2) або всі можуть бути рівняннями. Крім того, ці обмеження можуть містити умови невід’ємності змінних.

Нелінійність функцій $F(x)$, $\varphi_i(x)$ викликає суттєві відмінності задачі (9.1) – (9.2) від задачі лінійного програмування. Зокрема, якщо в задачі лінійного програмування точка екстремуму цільової функції є крайньою точкою многогранника допустимих розв’язків, то в задачі нелінійного програмування вона може знаходитися як на межі області допустимих розв’язків, так і всередині неї. Нелінійна цільова функція може мати кілька точок екстремуму

Нелінійність обмежень (9.2) може спричинити неопуклість області допустимих розв’язків (рис.9.1, а) або її незв’язність (рис.9.1, б).

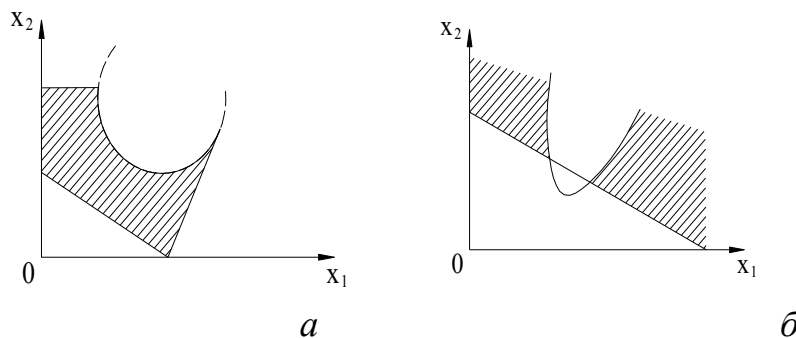


Рис.9.1

Наведені особливості задач нелінійного програмування суттєво ускладнюють їх розв’язання. Зокрема, на відміну від задач лінійного програмування задача (9.1) – (9.2) не має критерію оптимальності; не існує загальних методів, аналогічних симплекс-методу, які б дозволили розв’язати будь-яку задачу нелінійного програмування. Тому виділяються окремі спеціальні типи нелінійних задач із своїми методами розв’язування.

За допомогою більшості із цих методів можна знайти точку локального екстремуму. Щоб визначити глобальний екстремум, необхідно обчислити цільову функцію у всіх підозрілих на екстремум точках і порівняти одержані значення між собою.

В нелінійному програмуванні виділяють два основні типи задач – опуклого і неопуклого програмування. В задачах *опуклого програмування* на максимум цільова функція вгнута (в задачах на мінімум – опукла), а обмеження задають опуклу множину допустимих розв’язків. Всі інші типи нелінійних задач належать до задач *неопуклого програмування*.

В опуклому програмуванні в свою чергу спеціальний клас утворюють задачі *квадратичного програмування*. В них цільова функція є поліном другого степеня, а область допустимих розв’язків задається системою лінійних обмежень.

9.2.Класична задача нелінійного програмування

Розглянемо частинний випадок загальної задачі нелінійного програмування (9.1) – (9.2):

$$F(x) \rightarrow \max(\min); \quad (9.3)$$

$$\varphi_i(x) = 0, \quad i = \overline{1, m}, \quad (9.4)$$

тут система обмежень (9.4) складається лише із рівнянь. Якщо функції $F(x)$, $\varphi_i(x)$ неперервні і мають частинні похідні другого порядку, то в курсі математичного аналізу задачу (9.3) – (9.4) називають задачею на *умовний екстремум* або *класичною задачею оптимізації*. Як відомо, для розв’язування цієї задачі складається *функція Лагранжа*

$$L(\vec{x}, \vec{\lambda}) = F(\vec{x}) + \sum_{i=1}^m \lambda_i \varphi_i(\vec{x}), \quad (9.5)$$

де $\lambda_1, \lambda_2, \dots, \lambda_m$ – множники Лагранжа.

Необхідна умова екстремуму функції $L(x, \lambda)$ має вигляд системи $n+m$ рівнянь відносно змінних x_j і множників λ_i :

$$\begin{cases} \frac{\partial L}{\partial x_j} = \frac{\partial F}{\partial x_j} + \sum_{i=1}^m \lambda_i \frac{\partial \varphi_i}{\partial x_j} = 0, & j = \overline{1, n}; \\ \frac{\partial L}{\partial \lambda_i} = \varphi_i(x) = 0, & i = \overline{1, m}. \end{cases} \quad (9.6)$$

Із системи (9.6) знаходяться стаціонарні точки (x^0, λ^0) функції (9.5). Оскільки $\varphi_i(\vec{x}^0) = 0$, $i = \overline{1, m}$, то точка $x^0 = (x_1^0, x_2^0, \dots, x_n^0)$ буде точкою можливого екстремуму функції $F(x)$. Характер екстремуму в точці x^0 з’ясовується за допомогою достатніх умов. Функція $F(x)$ в стаціонарній точці x^0 має умовний максимум, якщо $d^2 L(x^0, \lambda^0) < 0$ і умовний мінімум, якщо $d^2 L(x^0, \lambda^0) > 0$.

Часто для визначення глобального екстремуму функції $F(x)$ достатньо обчислити значення цієї функції у відповідних стаціонарних точках.

Приклад 9.1. Знайти умовний екстремум функції $F = x_1 x_2 + x_2 x_3$, якщо

$$\begin{cases} x_1 + x_2 = 4, \\ x_2 + x_3 = 4. \end{cases}$$

Розв'язання. Складемо функцію Лагранжа

$$L(x, \lambda) = x_1 x_2 + x_2 x_3 + \lambda_1 (x_1 + x_2 - 4) + \lambda_2 (x_2 + x_3 - 4).$$

Знайдемо частинні похідні по змінних x_1, x_2, λ_1 і λ_2 :

$$\begin{aligned} \frac{\partial L}{\partial x_1} &= x_2 + \lambda_1; & \frac{\partial L}{\partial x_2} &= x_1 + x_3 + \lambda_1 + \lambda_2; & \frac{\partial L}{\partial x_3} &= x_2 + \lambda_2; \\ \frac{\partial L}{\partial \lambda_1} &= x_1 + x_2 - 4; & \frac{\partial L}{\partial \lambda_2} &= x_2 + x_3 - 4. \end{aligned}$$

Маємо систему рівнянь для визначення стаціонарних точок:

$$\begin{cases} x_2 + \lambda_1 = 0, \\ x_1 + x_3 + \lambda_1 + \lambda_2 = 0, \\ x_2 + \lambda_2 = 0, \\ x_1 + x_2 - 4 = 0, \\ x_2 + x_3 - 4 = 0. \end{cases}$$

Із першого та третього рівняння видно, що $\lambda_1 = \lambda_2 = -x_2$; тому вилучаємо λ_1, λ_2 :

$$\begin{cases} x_1 - 2x_2 + x_3 = 0, \\ x_1 + x_2 = 4, \\ x_2 + x_3 = 4; \end{cases} \Rightarrow \begin{cases} x_1 - 2x_2 + x_3 = 0, \\ x_1 + 2x_2 + x_3 = 8, \\ x_1 + x_2 = 4; \end{cases} \Rightarrow \begin{cases} 2x_1 + 2x_3 = 8, \\ 4x_2 = 8, \\ x_1 + x_2 = 4; \end{cases} \Rightarrow$$

$$x_2 = 2, \quad x_1 = 2, \quad x_3 = 2. \text{ Маємо стаціонарну точку } x^0 = (2; 2; 2).$$

Перевіримо достатню умову екстремуму в цій точці. Скориставшись обмеженнями, вилучимо змінні x_1, x_3 із функції $F(x)$:

$$F(x_2) = 2x_2(4 - x_2) = 8x_2 - 2x_2^2.$$

Оскільки $F'' = -4 < 0$, то x^0 – точка максимуму і $F_{\max} = F(x^0) = 8$ •

9.3. Опуклі і вгнуті функції

Функція $F(x)$, яка задана на опуклій множині X , називається *опуклою*, якщо для будь-яких двох точок $x^1, x^2 \in X$ і довільного $\lambda \in [0; 1]$ виконується нерівність

$$F(\lambda x^1 + (1 - \lambda)x^2) \leq \lambda F(x^1) + (1 - \lambda)F(x^2). \quad (9.7)$$

Функція $F(x)$ буде *строго опуклою*, якщо в наведеному означенні знак " $\leq$ " можна замінити знаком "<".

Аналогічно означаються поняття вгнутої і строго вгнутої функцій.

Функція $F(x)$, яка задана на опуклій множині X , називається *вгнутою*, якщо для будь-яких двох точок $x^1, x^2 \in X$ і довільного $\lambda \in [0; 1]$ виконується нерівність

$$F(\lambda x^1 + (1 - \lambda)x^2) \geq \lambda F(x^1) + (1 - \lambda)F(x^2). \quad (9.8)$$

Якщо $F(x)$ опукла функція, то $-F(x)$ – вгнута, і навпаки. Лінійна функція $F(x) = c \cdot x$ є прикладом одночасно опуклої і вгнутої функції, оскільки для будь-яких x^1, x^2 і довільного $\lambda \in [0; 1]$ виконується рівність

$$c \cdot (\lambda x^1 + (1-\lambda)x^2) = \lambda c x^1 + (1-\lambda)c \cdot x^2.$$

В багатьох випадках перевірка функції на опуклість та вгнутість є непростою проблемою. В конкретних випадках таку перевірку можна зробити простіше.

Нехай функція $F(x)$ двічі неперервно диференційовна по всіх змінних $x_1, x_2, \dots, x_n$. Складемо матрицю $H(x)$:

$$H_{n \times n} = (F''_{ij}(x)), \quad (9.9)$$

де $F''_{ij}(x) = \frac{\partial^2 F}{\partial x_i \partial x_j}$. Матриця $H(x)$ називається *матрицею Гессе* або *гессіаном функції $F(x)$* . Позначимо через M_j головні мінори j -того порядку ($j=1, 2, \dots, n$) матриці $H(x)$:

$$M_1 = F''_{11}, \quad M_2 = \begin{vmatrix} F''_{11} & F''_{12} \\ F''_{21} & F''_{22} \end{vmatrix}, \dots, \quad M_n = |H(x)|.$$

Нехай X – опукла множина і $x^0 \in X$. Для того, щоб функція $F(x)$ була строго опуклою в деякому околі точки x^0 , необхідно і достатньо, щоб в точці x^0 , виконувались умови

$$M_1(x^0) > 0, \quad M_2(x^0) > 0, \dots, \quad M_n(x^0) > 0. \quad (9.10)$$

Умови строгої вгнутості функції $F(x)$ в околі точки x^0 мають вигляд

$$M_1(x^0) < 0, \quad M_2(x^0) > 0, \quad M_3(x^0) < 0, \dots, \quad (-1)^n M_n(x^0) > 0. \quad (9.11)$$

Якщо наведені умови опуклості (вгнутості) виконуються для всіх $x \in X$, то неперервно диференційовна функція $F(x)$ буде строго опуклою (вгнутою) скрізь на множині X .

Часто при перевірці функції на опуклість користуються властивістю: якщо функції $f_i(x), i=1, k$ опуклі (вгнуті) на множині X , то функція $F(x) = \sum_{i=1}^k f_i(x)$ також опукла (вгнута) на X .

Приклад 9.2. Перевірити опуклість або вгнутість функції

$$F(x) = 3x_1^2 - 3x_1x_2 + 2x_2^2 + 5x_1 - 7 \text{ в околі точки } x^0 = (1; 2).$$

Розв'язання. Знаходимо частинні похідні другого порядку:

$$\begin{aligned} \frac{\partial F}{\partial x_1} &= 6x_1 - 3x_2 + 5, & \frac{\partial F}{\partial x_2} &= -3x_1 + 4x_2, \\ \frac{\partial^2 F}{\partial x_1^2} &= 6, & \frac{\partial^2 F}{\partial x_1 \partial x_2} &= \frac{\partial^2 F}{\partial x_2 \partial x_1} = -3, & \frac{\partial^2 F}{\partial x_2^2} &= 4 \end{aligned}$$

і записуємо матрицю Гессе (14.9):

$$H(x) = \begin{pmatrix} 6 & -3 \\ -3 & 4 \end{pmatrix}$$

Оскільки елементи цієї матриці є числа, то головні мінори не залежать від x . Обчислюємо визначники

$$M_1 = 6 > 0, \quad M_2 = \begin{vmatrix} 6 & -3 \\ -3 & 4 \end{vmatrix} = 15 > 0.$$

Згідно умов (14.10) функція $F(x)$ строго опукла при всіх значеннях $x = (x_1, x_2)$ •

Розглянемо деякі властивості опуклих і вгнутих функцій.

Теорема 9.1. Локальний мінімум (максимум) опуклої (вгнутої) функції $F(x)$ заданої на опуклій множині X , співпадає з її глобальним мінімумом (максимумом) на цій множині.

Доведення: Нехай функція $F(x)$ має локальний мінімум в точці $x^0 \in X$. Це означає, що в деякому околі цієї точки виконується нерівність $F(x^0) \leq F(x)$.

Припустимо, що глобальний мінімум досягається в точці $x^* \in X$: $F(x^*) < F(x^0)$.

Розглянемо відрізок, що з'єднує точки x^0 і x^* . В силу опуклості множини X цей відрізок належить X . Візьмемо на ньому довільну точку $\lambda x^0 + (1 - \lambda)x^*$, де $\lambda \in (0; 1)$. Оскільки функція $F(x)$ опукла, то виконується співвідношення $F(\lambda x^0 + (1 - \lambda)x^*) \leq \lambda F(x^0) + (1 - \lambda)F(x^*)$.

Дану нерівність можна підсилити, якщо замість $F(x^*)$ підставити справа $F(x^0) > F(x^*)$; маємо нерівність $F(\lambda x^0 + (1 - \lambda)x^*) < F(x^0)$

Значення λ можна вибрати так, щоб точка $\lambda x^0 + (1 - \lambda)x^*$ знаходилась як завгодно близько до x^0 . Тоді в околі точки x^0 існує точка x , в якій $F(x) < F(x^0)$. Це суперечить тому, що x^0 – точка локального мінімуму.

Одержане протиріччя доводить теорему •

Наслідок 1. Якщо глобальний екстремум досягається у двох різних точках, то він досягається і в будь-якій точці відрізка, що сполучає дані точки.

Наслідок 2. Якщо $F(x)$ строго опукла (вгнута) на множині X функція, то вона має глобальний мінімум (максимум) в єдиній точці

Нехай функція $F(x)$ опукла (вгнута) на множині X і неперервно диференційовна на цій множині.

Теорема 9.2. Опукла (вгнута) функція $F(x)$ досягає свого глобального мінімуму (максимуму) на множині X в кожній точці x^0 , в якій $\text{grad } F(x) = 0$.

Наведені властивості опуклих (вгнутих) функцій спрощують розв'язування задач опуклого програмування.

9.4. Задача опуклого програмування.

Розглянемо задачу нелінійного програмування:

$$F(x) \rightarrow \min; \quad (9.12)$$

$$\varphi_i(x) \leq 0, \quad i = 1, 2, \dots, m; \quad x \geq 0. \quad (9.13)$$

Якщо функція $F(x)$ опукла, а обмеження (9.13) задають опуклу множину допустимих розв'язків D , то маємо задачу опуклого програмування. Для такої задачі, на відміну від загальної задачі нелінійного програмування, існує критерій оптимальності розв'язку – умови Куна-Таккера. Вони узагальнюють метод множників Лагранжа на задачі, обмеження яких містять нерівності і умови невід'ємності змінних.

Для задачі (9.12) – (9.13) складемо функцію Лагранжа $L(x, \lambda)$ виду (9.5).

Кажуть, що область допустимих розв'язків D задачі (9.12) – (9.13) задовольняє умові регулярності, якщо існує точка $x \in D$, що $\varphi_i(x) < 0$ для всіх $i = \overline{1, m}$.

Точка (x^0, λ^0) називається *сідловою точкою* функції $L(x, \lambda)$, якщо виконується нерівність

$$L(x^0, \lambda) \leq L(x^0, \lambda^0) \leq L(x, \lambda^0) \quad (9.14)$$

при всіх $x \geq 0, \lambda \geq 0$.

Із нерівності (9.14) видно, що сідловою буде така точка (x^0, λ^0) , в якій функція $L(x, \lambda)$ має мінімум по змінних x і максимум по змінних λ (рис.9.2).

Припустимо, що $F(x)$, $\varphi_i(x)$ – диференційовні функції. Якщо немає обмежень на знак змінних x_j, λ_i , то похідні $\frac{\partial L}{\partial x_j}, \frac{\partial L}{\partial \lambda_i}$ в сідловій точці (x^0, λ^0) повинні дорівнювати нулю. При наявності обмежень $x_j \geq 0, \lambda_i \geq 0$ точка екстремуму може бути на межі $x_j = 0$ або $\lambda_i = 0$. При цьому похідні $\frac{\partial L}{\partial x_j}, \frac{\partial L}{\partial \lambda_i}$ або дорівнюють нулю, або $\frac{\partial L(x^0, \lambda^0)}{\partial x_j} > 0$ і $\frac{\partial L(x^0, \lambda^0)}{\partial \lambda_i} < 0$. Отже, умови того, що в точці (x^0, λ^0) завжди або похідні, або змінні дорівнюють нулю, набувають вигляд:

$$\begin{cases} x_j^0 \frac{\partial L(x^0, \lambda^0)}{\partial x_j} = 0, & j = \overline{1, n}; \\ \lambda_i^0 \frac{\partial L(x^0, \lambda^0)}{\partial \lambda_i} = 0, & i = \overline{1, m}; \end{cases} \quad (9.15)$$

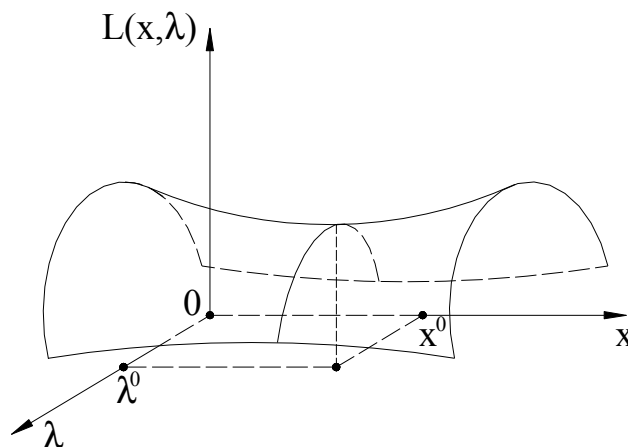


Рис. 9.2.

Співвідношення (9.15) є необхідними умовами того, що точка (x^0, λ^0) буде сідловою точкою функції Лагранжа для задачі (9.12) – (9.13), і називаються *умовами Куна-Таккера*.

Теорема 9.3 (теорема Куна-Таккера). Нехай область допустимих розв'язків задачі (9.12) – (9.13) є регулярною. Тоді для оптимальності вектора x^0 в

задачі (9.12) – (9.13) необхідно і достатньо, щоб існував вектор $\lambda^0 \geq 0$, при якому пара (x^0, λ^0) буде сідловою точкою функції Лагранжа (9.5).

Таким чином, задача опуклого програмування (9.12) – (9.13) еквівалентна задачі про сідлову точку відповідної функції Лагранжа і зводиться до розв'язування системи рівнянь (9.15). Проте розв'язування цієї системи часто вимагає виконання значного обсягу обчислень.

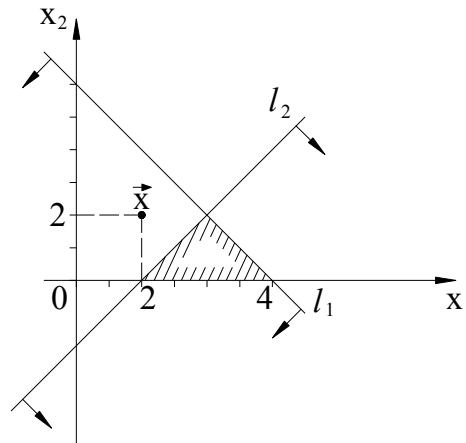
Приклад 9.3. Розв'язати задачу

$$F(x) = 3(x_1 - 2)^2 + (x_2 - 2)^2 - 16 \rightarrow \min;$$

$$\begin{cases} x_1 + x_2 \leq 6, \\ x_1 - x_2 \geq 2; \end{cases}$$

$$x_1 \geq 0, \quad x_2 \geq 0$$

Розв'язання. Область допустимих розв'язків D задачі представлена на рис.9.3.



Систему обмежень приведемо до вигляду (9.13) і складемо функцію Лагранжа:

$$L(x, \lambda) = 3(x_1 - 2)^2 + (x_2 - 2)^2 - 16 + \lambda_1(x_1 + x_2 - 6) + \lambda_2(2 - x_1 + x_2).$$

Знаходимо частинні похідні

$$\frac{\partial L}{\partial x_1} = 6x_1 - 12 + \lambda_1 - \lambda_2, \quad \frac{\partial L}{\partial x_2} = 2x_2 - 4 + \lambda_1 + \lambda_2,$$

$$\frac{\partial L}{\partial \lambda_1} = x_1 + x_2 - 6, \quad \frac{\partial L}{\partial \lambda_2} = 2 - x_1 + x_2$$

і виписуємо систему рівнянь (9.15):

$$\begin{cases} x_1(6x_1 - 12 + \lambda_1 - \lambda_2) = 0, \\ x_2(2x_2 - 4 + \lambda_1 + \lambda_2) = 0, \\ \lambda_1(x_1 + x_2 - 6) = 0, \\ \lambda_2(2 - x_1 + x_2) = 0. \end{cases}$$

При розв'язанні цієї системи потрібно розглянути різні можливі випадки значень невідомих: $x_1 > 0, x_2 > 0$; $x_1 > 0, x_2 = 0$; $x_1 = 0, x_2 > 0$; $x_1 = x_2 = 0$. Якщо $x_1 > 0, x_2 > 0$, то умови Куна-Таккера набувають вигляду

зв'ідки знаходимо:

- Оскільки за теоремою Куна-Таккера вектор $\lambda = (\lambda_1, \lambda_2) \geq 0$, а $x = (2; 2) \notin D$, то маємо сідлову точку (x^0, λ^0) , де $x^0 = (2, 5; 0, 5)$, $\lambda^0 = (0; 3)$.

Аналогічно, у випадку $x_1 > 0, x_2 = 0$ знаходимо ще одну сідлову точку (x', λ') , де $x' = (2; 0), \lambda' = 0$.

Шляхом обчислення значення цільової функції $F(x)$ в точках x^0, x' переконуємось, що $F_{\min} = F(x^0) = -13$ •

Частинним випадком задачі опуклого програмування є задача квадратичного програмування. В ній цільова функція містить квадратичну форму, а система обмежень має лінійний вигляд.

Квадратичною формою відносно змінних $x_1, x_2, \dots, x_n$ називається числова функція, яка має вигляд

(9.16)

Квадратичну форму можна записати так

Видно, якщо $i \neq j$, то коефіцієнт при $x_i x_j$ дорівнює $q_{ij} + q_{ji}$. Звідси випливає, що матрицю $Q_{n \times n} = (q_{ij})$ коефіцієнтів квадратичної форми можна вважати симетричною. Дійсно, вважаючи новими коефіцієнтами

Скільки $q'_{ij} + q'_{ji} = q_{ij} + q_{ji}$, то при такому переході до нових коефіцієнтів значення виразу $Q(x)$ не змінюється. Тому будемо вважати, що матриця Q коефіцієнтів квадратичної форми завжди симетрична. Неважко переконатись, що в цьому випадку вона буде пропорційна матриці Гессе із коефіцієнтом пропорційності 2.

Квадратична форма $Q(x)$ називається *додатно (від'ємно) визначеною*, якщо $Q(x) > 0 (Q(x) < 0)$ для всіх значень $x \in R^n$, при $x \neq 0$.

Квадратична форма $Q(x)$ називається *невід'ємно (недодатньо)* визначеною, якщо $Q(x) \geq 0$ ($Q(x) \leq 0$) для всіх значень x і існує $x \neq 0$, при якому $Q(x) = 0$.

Квадратична форма $Q(x)$ називається *невизначеною*, якщо вона додатна для одних значень x і від'ємна для інших.

Справедливе твердження: невід'ємно (додатно) визначена квадратична форма $Q(x)$ є опуклою (вгнутою) функцією. Це дає можливість випуклість квадратичної функції перевірити за допомогою *критерію Сільвестра*: квадратична форма $Q(x)$ буде строго опуклою, якщо всі головні мінори матриці її коефіцієнтів додатні; якщо всі непарні головні мінори матриці коефіцієнтів квадратичної форми від'ємні, а парні додатні, то вона буде строго вгнутою.

Розглянемо загальну задачу квадратичного програмування:

$$F(x) = c \cdot x + Q(x) \rightarrow \min; \quad (9.17)$$

$$Ax \leq b; \quad (9.18)$$

$$x \geq 0, \quad (9.19)$$

де $Q(x)$ – невід'ємно визначена квадратична форма, x , $c \in R^n$, $b \in R^m$, $A_{m \times n} = (a_{ij})$. Для такої задачі функція Лагранжа має вигляд

$$L(\vec{x}, \vec{\lambda}) = F(\vec{x}) + \sum_{i=1}^m \lambda_i \left(\sum_{j=1}^n a_{ij} x_j - b_i \right) \quad (9.20)$$

Якщо функція $L(x, \lambda)$ має сідлову точку (x^0, λ^0) , то в ній виконуються умови (9.15), які перепишемо в іншому вигляді:

$$\left\{ \begin{array}{l} \frac{\partial L(x^0, \lambda^0)}{\partial x_j} \geq 0, \\ x_j^0 \frac{\partial L(x^0, \lambda^0)}{\partial x_j} = 0, \\ x_j^0 \geq 0, \quad j = \overline{1, n}, \\ \frac{\partial L(x^0, \lambda^0)}{\partial \lambda_i} \leq 0, \\ x_i^0 \frac{\partial L(x^0, \lambda^0)}{\partial \lambda_i} = 0, \\ x_i^0 \geq 0, \quad i = \overline{1, m}. \end{array} \right. \quad (9.21)$$

Введемо балансові змінні v_j і u_i у відповідні нерівності $\frac{\partial L(x^0, \lambda^0)}{\partial x_j} \geq 0$ та $\frac{\partial L(x^0, \lambda^0)}{\partial \lambda_i} \leq 0$, $j = \overline{1, n}$, $i = \overline{1, m}$ системи (9.21). Тоді умови Куна-Таккера (9.21) існування сідлової точки для функції Лагранжа можна записати так:

$$\begin{cases} \frac{\partial L(\vec{x}^0, \lambda^0)}{\partial x_j} - v_j = 0, & j = \overline{1, n}, \\ \frac{\partial L(\vec{x}^0, \lambda^0)}{\partial \lambda_i} + u_i = 0, & i = \overline{1, m}; \end{cases} \quad (9.22)$$

$$\begin{cases} x_j^0 v_j = 0, & j = \overline{1, n}, \\ \lambda_i^0 u_i = 0, & i = \overline{1, m}; \end{cases} \quad (9.23)$$

$$x_j^0 \geq 0, v_j \geq 0, \lambda_i^0 \geq 0, u_i \geq 0 \quad (j = \overline{1, n}, i = \overline{1, m}). \quad (9.24)$$

Отже, щоб знайти розв'язок x^0 задачі квадратичного програмування (9.17) – (9.19), потрібно визначити невід'ємний розв'язок системи лінійних рівнянь (9.22), який задовольняє умовам (9.23). Такий розв'язок можна знайти за допомогою модифікації методу штучного базису. Для цього в систему рівнянь (9.22) вводяться штучні змінні $z_k \geq 0, k = \overline{1, s}$ ($s \leq n + m$). Знайти початковий опорний план, симплексним методом мінімізується вираз $Z = M \sum_{k=1}^s Z_k$ при додаткових умовах (9.23).

Ці умови змінюють правило вибору вільної змінної, що вводиться до базисних: якщо $x_j > 0$, то забороняється вводити до базисних змінну v_j , і навпаки (аналогічно для змінних λ_i та u_i). Такі симплексні перетворення виконують до тих пір, поки не приходять до базисного розв'язку, при якому виконується рівність $z=0$.

Отже, за скінчене число ітерацій симплекс-методу знаходять оптимальний план задачі квадратичного програмування (9.17) – (9.19).

9.6. Градієнтні методи.

Розглянемо задачу мінімізації нелінійної диференційованої функції $F(x)$:

$$F(x) \rightarrow \min; \quad x \in R^n.$$

Позначимо через $\nabla F(\vec{x}) = \left(\frac{\partial F}{\partial x_1}, \frac{\partial F}{\partial x_2}, \dots, \frac{\partial F}{\partial x_n} \right)$ градієнт функції $F(x)$.

Відомо, що напрямок найшвидшого зростання функції $F(x)$ в точці x співпадає із напрямком градієнта $\nabla F(x)$, а напрямок найшвидшого спадання – з напрямком антиградієнта $-\nabla F(x)$. Ця властивість градієнта лежить в основі градієнтних методів. У кожному з них процес починається в деякій точці x^0 початкового (нульового) наближення.

Нехай точка x^0 обрана. Тоді *метод градієнта* мінімізує функцію $F(x)$ шляхом побудови послідовності $\{x^k\}$, елементи якої визначаються за рекурентною формулою

$$x^{k+1} = x^k - h_k \nabla F(x^k), \quad k = 1, 2, \dots, \quad (9.25)$$

де $h_k > 0$ – крок. Якщо $\nabla F(x^k) \neq 0$, то завжди можна вибрати таку довжину кроку h_k переходу в точку x^{k+1} , у якій $F(x^{k+1}) < F(x^k)$. Якщо $\nabla F(x^k) = 0$, то у випадку опуклості функції $F(x)$ точка x^k є точкою її мінімуму; в інших випадках необхідно перевірити наявність екстремуму в точці x^k .

Гradientні методи розрізняються способами вибору довжини кроку h_k .

Найпростіший із них полягає в тому, що на кожній ітерації приймається $h_k = h$, якщо при цьому виконується умова монотонності

$$F(x^{k+1}) < F(x^k), \quad k = 1, 2, \dots \quad (9.26)$$

При порушенні монотонності число h зменшують, поки монотонність не відновиться. На практиці застосовується і більш гнучка тактика «розгону і гальмування», при якій величина кроку збільшується або зменшується в залежності від поведінки функції в ході пошуку.

Для підвищення швидкодії методу доцільно кожен крок виконувати з максимально можливим кроком. Метод, в якому реалізована ця ідея, називається методом *якнайшвидшого спуску*. Нехай у початковій точці x^0 обчислено градієнт $\nabla F(x^0)$. Будемо рухатись у напрямку антиградієнта доти, поки буде зменшуватись функція $F(x)$. Точка x^1 , в якій функція перестав зменшуватись, є точкою її мінімуму в напрямку $-\nabla F(x^0)$. В точці x^1 обчислюється новий градієнт і в напрямку $-\nabla F(x^1)$, робимо максимально можливий крок h_1 , який приводить у точку x^2 , і т.д.

Таким чином, чергова точка x^{k+1} у (15.16) є точкою мінімуму функції $F(x)$ на промені $x = x^k - h\nabla F(x^k)$, $h \geq 0$, спрямованому по антиградієнту. Звідси довжина кроку h_k визначається із умови мінімуму функції $F(h)$ однієї змінної:

$$F_k(h_k) = \min_{h \geq 0} F(x^k - h\nabla F(x^k)).$$

Ітераційний процес побудови послідовності $\{x^k\}$ припиняється при виконанні деякого критерію закінчення пошуку мінімуму функції однієї змінної. Цей критерій визначає близькість точки x^k до шуканої точки мінімуму. Як один із таких критеріїв можна використати оцінку близькості або двох сусідніх точок послідовності, або значень функцій в таких точках, або модуль градієнта, тобто припинити обчислення при виконанні однієї із умов:

$$|\bar{x}^k - \bar{x}^{k+1}| \leq \delta, \text{ або } |F(\bar{x}^k) - F(\bar{x}^{k+1})| \leq \varepsilon, \text{ або } |\nabla F(\bar{x}^k)| \leq \varepsilon_1,$$

де $\delta, \varepsilon, \varepsilon_1$ – задані числа.

9.7. Методи нульового порядку.

Методи, в яких використовуються похідні першого або другого порядку, за певних умов забезпечують гарну збіжність. Проте вони чутливі до вибору початкової точки пошуку: для їхньої збіжності вона повинна бути близькою до околу точки мінімуму. Крім того, не всі функції $F(x)$ мають перші і тим більше другі похідні. В методах нульового порядку використовуються лише значення функції. Незважаючи на повільну збіжність таких методів, вони мають широке використання.

Спільним для більшості методів нульового порядку є те, що спуск із чергової точки робиться за напрямком однієї із координатних осей. Послідовність вибору цих осей устанавлюється різними способами.

До методів, що використовують лише значення функції, відносяться методи *випадкового пошуку*, у яких переміщення до точки екстремуму здійснюється випадковим чином. Один із найпростіших методів випадкового пошуку полягає в наступному. Вибирають величину кроку h і з початкової точки x^0 роблять у випадковому напрямку переміщення на відстань h . Якщо значення функції $F(x)$ у новій точці не зменшилось, то повертаються в точку x^0 і здійснюють у випадковому напрямку новий крок на відстань h . У випадку невдачі знову повертаються в точку x^0 і процес повторюють доти, поки чергове переміщення не приведе до зменшення цільової функції. Після цього в напрямку останнього переміщення роблять кроки величиною h доти, поки приріст функції $F(x)$ не змінить знак. Потім процес повторюють. При цьому, якщо із чергової точки x^k певне число випадкових переміщень на відстань h не привело до зменшення функції, то точку x^k вважають точкою мінімуму.

Головна перевага методів випадкового пошуку – їхня простота; ці методи не накладають ніяких обмежень на функцію $F(x)$. Основний їхній недолік – повільна збіжність.

9.8. Метод штрафних функцій.

Розглянемо задачу опуклого програмування (9.12). Якщо мінімум цільової функції знаходиться всередині області допустимих значень, то наявність обмежень не впливає на процес пошуку розв'язку. Мінімум шукається практично так само, як і у випадку без обмежень. Якщо точка мінімуму перебуває за межами допустимої області, то відшукування найменшого значення цільової функції, при наявності обмежень на змінні, істотно ускладнюється.

Одним із найефективніших методів розв'язування задач з обмеженнями є *метод штрафних функцій*. Ідея цього методу така. Замість того, щоб безпосередньо розв'язувати задачу (15.1) з обмеженнями, переходять до послідовності задач мінімізації допоміжної функції $f(x) = F(x) + H(x)$, у якій на x не накладається ніяких обмежень. Доданок $H(x)$ до функції $F(x)$ можна трактувати як спеціальний штраф за неточне виконання обмежень $\varphi_i(x) \leq 0$, $i = \overline{1, m}$. Тому функцію $H(x)$ називають *функцією штрафу*. Метод штрафних функцій дає можливість зводити задачі на умовний екстремум до задач на безумовний екстремум. Функції штрафу повинні дорівнювати нулю для тих значень x , при яких обмеження виконуються, і нескінченно великі для тих x , при яких обмеження порушуються.

Виходячи з цих міркувань, варто вибирати функцію штрафу $H_k(x)$ так, щоб вона мала такі властивості

$$\lim_{k \rightarrow \infty} H_k(x) = \begin{cases} 0, & \text{при } \varphi_i(x) \leq 0; \\ \infty, & \text{при } \varphi_i(x) > 0. \end{cases}$$

Тоді точка x , що мінімізує функцію $f(x)$, буде точкою мінімуму і функції $F(x)$. Можна утворити багато функцій, що мають зазначені властивості. Наприклад, візьмемо функцію

$$H_k(x) = C_k H(x), \quad H(x) = \sum_{i=1}^m [\max(\varphi_i(\vec{x}), 0)]^\alpha,$$

де $C_k > 0$, $k = 1, 2, \dots$, $\lim_{k \rightarrow \infty} C_k = \infty$, $\alpha \geq 1$ – фіксоване число.

Таким чином, вихідну задачу розв'язують в результаті послідовної мінімізації функції $f(x)$ із зростаючим штрафом за порушення обмежень. При цьому точка мінімуму, яка одержана на попередньому кроці k , буде початковою точкою пошуку на наступному кроці $k + 1$.

Завдання для самоконтролю.

1. Які задачі відносяться до задач нелінійного програмування?
2. Вкажіть на відмінність між задачами нелінійного та лінійного програмування.
3. Назвіть типи задач нелінійного програмування. Чим вони характеризуються?
4. Який тип задач нелінійного програмування розв'язується методом множників Лагранжа? В чому суть цього методу?
5. Дайте означення опуклої (вгнутої) функції.
6. Який вигляд має матриця Гессе і яке її застосування?
7. Сформулюйте необхідні умови Куна-Таккера для задачі мінімізації.
8. Який вигляд матимуть умови Куна-Таккера для задачі на максимум?
9. При розв'язуванні яких задач ефективно використовується симплекс-метод лінійного програмування?
10. На чому ґрунтуються градієнтні методи? Охарактеризуйте недоліки і переваги цих методів.
11. Яка основна ідея методів випадкового пошуку? Вкажіть переваги і недоліки цих методів.
12. Що таке функція штрафу і яке її призначення?
13. Розв'язати графічно задачу нелінійного програмування:

$$F = x_1^2 + x_2^2 - 6x_1 - 4x_2 + 13 \rightarrow \max(\min);$$

$$\begin{cases} x_1 + x_2 \leq 7, \\ 2x_1 - x_2 \leq 8; \\ x_1 \geq 0, \quad x_2 \geq 0. \end{cases}$$
(Відповідь, $F_{\max} = F(0, 7) = 34$, $F_{\min} = F(3, 2) = 0$.)
14. Розв'язати графічно задачу нелінійного програмування

$$F = (x_1 - 1)^2 + (x_2 - 5)^2 \rightarrow \max(\min);$$

$$\begin{cases} -x_1 + 2x_2 \leq 6, \\ 3x_1 - 2x_2 \leq 12; \\ x_1 \geq 0, \quad x_2 \geq 0. \end{cases}$$

(Відповідь $F_{\max} = F(9; 7, 5) = 70,25$. $F_{\min} = 1,8$)

15. Знайти умовні екстремуми функції

$F = 8 - 3x_1 - 4x_2$, якщо $x_1^2 + x_2^2 = 4$. (Відповідь $F_{\min} = 8 - 3,6 - 6,4 = -2$.

$F_{\max} = 8 + 3,6 + 6,4 = 18$.)

16. Розв'язати задачу квадратичного програмування:

$$F = x_1^2 + x_2^2 + 2x_1 - 8x_2 \rightarrow \min;$$

$$\begin{cases} x_1 + 2x_2 \leq 12, \\ -x_1 + x_2 \geq -8; \end{cases}$$

$$x_1 \geq 0, x_2 \geq 0.$$

(Відповідь: точка $(x^0, \lambda^0) = (0; 4; 0; 0)$ буде сідловою для функції Лагранжа, $x^0 = (0; 4)$ – оптимальний розв'язок задачі і $F_{\min} = F(x^0) = -16$)

Лекція № 10. Динамічне програмування

*Загальна характеристика задач динамічного програмування.
Економічна інтерпретація задачі динамічного програмування
Обчислювальні методи динамічного програмування.*

10.1. Загальна характеристика задач динамічного програмування.

Будемо розглядати задачі математичного програмування, в яких процес прийняття рішень розбивається на кілька кроків або етапів, які в основному залежать від часу. Такі задачі називаються *багатокроковими* або *багатоетапними*. Сукупність рішень, що приймаються на кожному етапі з метою вплинути на процес, називається *керуванням*. Наприклад, в економічних процесах керування полягає в розподілі і перерозподілі ресурсів на кожному етапі.

Математичні методи розв'язування багатокрокових задач оптимального керування називають *динамічним програмуванням*. Ці методи ґрунтуються на рекурентних співвідношеннях. Отже, термін «динамічне програмування» визначає не стільки особливий тип задач, скільки характеризує методи розв'язування окремих класів оптимізаційних задач, що можуть відноситися до задач як лінійного, так і нелінійного програмування.

Сформулюємо загальну задачу динамічного програмування. Нехай деяка фізична керована система S характеризується певним набором параметрів. В цій системі відбуваються якісь процеси (економічні, виробничі, технологічні тощо), які можна уявити як багатокрокові. На кожному кроці процесам у системі відповідають певні значення параметрів, які описують *стан системи*. Задані умови, які дозволяють визначити або початковий S_0 або кінцевий S_n стани системи, або обидва ці стани. Іноді додаються області початкових і кінцевих станів. Оскільки керування системою здійснюється для досягнення конкретної мети, то наведено *критерій ефективності* F . Він називається цільовою функцією і чисельно виражає ефект або «виграш», одержаний при тому чи іншому керуванні із множини допустимих керувань U . Внаслідок керування зазначена система переходить із початкового стану S_0 у кінцевий стан S_n . Задача динамічного програмування полягає в тому, щоб із множини допустимих керувань U вибрати таке керування u^* , при якому функція $F(U)$ набуває оптимальне (максимальне або мінімальне) значення $F(u^*)$. При цьому керування u^* називають *оптимальним*.

10.2. Економічна інтерпретація задачі динамічного програмування

Розглянемо кілька прикладів багатокрокових економічних задач.

Задача 10.1. Планується діяльність деякої системи S промислових підприємств $\Pi_1, \Pi_2, \dots, \Pi_m$ на період n господарських років. Для цього на розвиток підприємств виділені кошти в розмірі Q млн.грн, які на початку кожного господарського року розподіляються між підприємствами.

Одночасно з цим між ними розподіляється прибуток, одержаний за минулий рік. Як потрібно розподілити виділені кошти між підприємствами $\Pi_j (j = \overline{1, m})$ і одержаний ними прибуток, щоб загальний прибуток від всієї системи підприємств був максимальним?

Сформулювати задачу в термінах загальної задачі динамічного програмування.

Розв'язання. Нехай t_i – господарський рік діяльності підприємств ($i = \overline{1, 2, \dots, n}$). Позначимо через X_{ij} кошти, виділені на початку року t_i підприємству $\Pi_j (j = \overline{1, m}, i = \overline{1, n})$. Будемо розглядати такий розподіл коштів як реалізацію деякого керування u_i . Таким чином, керування u_i полягає в тому, що на i -му кроці підприємству Π_1 виділяється x_{i1} млн.грн., підприємству Π_2 – x_{i2} млн.грн. і т.д., тобто сукупність чисел $x_{i1}, x_{i2}, \dots, x_{im}$ визначає керування u_i як m -вимірний вектор. Керування U всією групою підприємств за n років складається із системи керувань на кожному кроці: $U = (u_1, u_2, \dots, u_n)$.

За критерій оцінки ефективності вибраного розподілу коштів обрано загальний прибуток за n років, який залежить від всієї сукупності керувань: $F = F(u_1, u_2, \dots, u_n)$.

Отже, задача полягає у виборі таких керувань u_i^* розподілу коштів, при яких функція F набуде максимального значення•

Видно, що наведена задача є багатокроковою. Ця багатокроковість визначається її умовою, якою передбачається прийняття певних рішень на початку кожного року відповідного періоду часу $t_i, i = \overline{1, n}$. Разом з тим в інших задачах динамічного програмування така багатоетапність безпосередньо не спостерігається в їхніх умовах. Проте, для відшукування оптимального розв'язку, такі задачі доцільно розглядати як багатокрокові.

Задача 10.2. Група підприємств $\Pi_1, \Pi_2, \dots, \Pi_n$ випускає продукцію, яка користується попитом. Для збільшення обсягів випуску продукції підприємствам виділені капіталовкладення обсягом Q тис.грн. Вкладені кошти x_i тис.грн. із виділеної суми в підприємство Π_i забезпечують приріст випуску продукції, що визначається значенням нелінійної функції $\varphi_i(x_i)$, $i = \overline{1, n}$.

Знайти розподіл капіталовкладень між підприємствами, що забезпечує максимальний приріст випуску продукції.

Розв'язання. Маємо задачу нелінійного програмування: знайти максимальне значення функції

$$F = \sum_{i=1}^n \varphi_i(x_i) \quad (10.1)$$

за умов

$$\sum_{i=1}^n x_i = Q \quad (10.2)$$

$$x_i \geq 0, \quad i = \overline{1, n}. \quad (10.3)$$

Якщо $\varphi_i(x_i)$ – опуклі (вгнуті) функції, то задачу (10.1) – (10.3) можна розв’язати, наприклад, методом множників Лагранжа. Якщо ж функції $\varphi_i(x_i)$ не є такими, то неможливо визначити глобальний максимум функції (10.1) відомими методами нелінійного програмування. Тоді розв’язок задачі (10.1) – (10.3) можна знайти за допомогою методу динамічного програмування. Для цього дану задачу потрібно розглядати як багатоетапну. Замість того, щоб розглядати допустимі варіанти розподілу капіталовкладень між n підприємствами і оцінювати ефективність цих варіантів, будемо досліджувати ефективність вкладення коштів на одному підприємстві, на двох підприємствах і т.д., нарешті, на n підприємствах. Таким чином, одержимо n етапів, на кожному з яких стан системи (обсяг капіталовкладень Q) описується обсягом коштів, виділених k підприємствам ($k = \overline{1, n}$). Рішення про обсяги капіталовкладень в підприємства Π_k ($k = \overline{1, n}$) будуть керуваннями. Задача полягає у виборі таких керувань, при яких функція (10.1) набуває найбільше значення•

10.3. Обчислювальні методи динамічного програмування.

Розглянемо обчислювальний алгоритм розв’язування загальної задачі динамічного програмування. Для цього введемо позначення і зробимо деякі припущення.

Нехай процес керування системою S складається із n кроків. На i -тому кроці керування u_i переводить систему із стану X_{i-1} у новий стан $X_i = (x_{i1}, x_{i2}, \dots, x_{in})$. Компоненти вектора X_i одержано в результаті реалізації керування u_i . При цьому будемо вважати, що стан X_i , до якого перейшла система S , залежить від попереднього стану X_{i-1} (умова відсутності післядії).

Позначимо через $W_i(X_{i-1}, u_i)$ ефект або виграш, одержаний внаслідок реалізації i -того кроку. Надалі приймаємо *принцип адитивності*: загальний ефект або виграш за n кроків становить

$$F = \sum_{i=1}^n W_i(X_{i-1}, u_i) \quad (10.4)$$

В основі методів динамічного програмування лежить *принцип оптимальності*, сформульований Р.Беллманом: *яким би не був стан системи перед черговим кроком, треба вибрати керування на цьому кроці так, щоб виграш на даному кроці плюс оптимальний виграш на всіх наступних кроках був максимальним.*

Звідси випливає, що оптимальну стратегію керування можна одержати, якщо знайти спочатку оптимальне керування на останньому n -ому кроці. Плануючи цей крок, потрібно зробити різні припущення про те, як закінчився попередній $(n-1)$ -й крок; для кожного із цих припущень знайти *умовно-оптимальне керування* і *умовно-оптимальний ефект* на n -ому кроці. Після цього оптимізуємо керування на $(n-1)$ -му кроці. Знову зробимо всі можливі припущення про те, як закінчився попередній $(n-2)$ -й крок, і для кожного із цих припущень знайдемо таке керування на $(n-1)$ -му

кроці, при якому ефект за останні два кроки (із яких n -й уже оптимізовано) буде максимальним. Так знайдемо для кожного наслідку $(n-2)$ -го кроку умовно-оптимальне керування на $(n-1)$ -му кроці і умовно-оптимальний ефект на двох останніх кроках. Потім, рухаючись назад, оптимізуємо керування на $(n-2)$ -му кроці і т.д., поки не дійдемо до першого.

Запишемо принцип оптимальності в математичній формі. Позначимо через $F_n(X_0)$ максимальний ефект, одержаний за n кроків при переході системи S із початкового стану X^0 в кінцевий стан X_n внаслідок реалізації оптимального управління $U = (u_1, u_2, \dots, u_n)$, а через $F_{n-i}(X_i)$ – максимальний виграш, одержаний при переході із стану X_i в кінцевий стан X_n внаслідок оптимальної стратегії керування на решті $n-i$ кроках. Тоді маємо співвідношення

$$F_n(X_0) = \max_{u_{i+1}} [W_1(X_0, u_1) + \dots + W_n(X_{n-1}, u_n)]; \quad (10.5)$$

$$F_{n-i}(X_i) = \max_{u_{i+1}} [W_{i+1}(X_i, u_{i+1}) + F_{n-i-1}(X_{i+1})], i = 0, n-1. \quad (10.6)$$

Рекурентне співвідношення (10.6) називається *основним функціональним рівнянням* динамічного програмування.

При $i=n-1$ рівняння (10.6) набуває вигляду:

$$F_1(X_{n-1}) = \max_{u_n} [W_n(X_{n-1}, u_n) + F_0(X_n)], \quad (10.7)$$

де $F_0(X_n)$ вважаємо відомими. Використовуючи тепер рівняння (10.7) і розглядаючи множину всіх можливих станів системи на $(n-1)$ -му кроці, знаходимо умовні оптимальні керування і відповідні значення функції (10.7).

Розглядаємо тепер функціональне рівняння (10.6) при $i=n-2$:

$$F_2(X_{n-1}) = \max_{u_{n-1}} [W_{n-1}(X_{n-2}, u_{n-1}) + F_1(X_{n-1})]. \quad (10.8)$$

Оскільки значення $F_1(X_{n-1})$ уже визначені, то потрібно обчислити значення функції $W_{n-1}(X_{n-2}, u_{n-1})$ при всіх можливих станах X_{n-2} і відповідних керуваннях u_{n-1} . Це дозволить визначити умовно-оптимальне керування u_{n-1} для кожного допустимого стану X_{n-2} . Кожне з таких керувань разом із обраним уже керуванням на останньому кроці забезпечує максимальний ефект на двох останніх кроках.

Послідовно продовжуючи такий ітераційний процес, дійдемо до першого кроку. Оскільки на цьому кроці відомий початковий стан системи, тому не потрібно робити припущень щодо можливого стану, а вибираємо оптимальне керування з урахуванням умовно-оптимальних керувань, які уже прийняті на всіх наступних кроках.

Отже, в результаті проходження всіх етапів від кінця до початку визначаємо максимальне значення виграшу за n кроків і для кожного з них знаходимо умовно-оптимальне керування.

Щоб визначити оптимальне керування, проведемо *безумовну оптимізацію*: проаналізуємо виконані розрахунки, проходячи всю послідовність кроків від початку до кінця. На першому кроці за оптимальне

керування u_1^* візьмемо умовно-оптимальне керування u_1^0 . На другому кроці знайдемо стан X_1^* , у який переходить система внаслідок керування u_1^* . Цей стан визначає знайдене умовно оптимальне керування u_2^0 , яке тепер будемо вважати оптимальним: $u_2^* = u_2^0$. Знаючи u_2^* , знаходимо X_2^* і т.д. до кінця. Отже, знаходимо оптимальне керування $U^* = (u_1^*, u_2^*, \dots, u_n^*)$ і відповідне максимальне значення цільової функції.

Запишемо функціональне рівняння для задачі (10.2). Нехай X – обсяг капіталовкладень, $F_k(X)$ – максимальний приріст випуску продукції при розподілі X тис.грн між k підприємствами ($k = \overline{1, n-1}$). Очевидно, що

$$F_1(X) = \max_{0 \leq X_1 \leq X} \{\varphi_1(x_1)\}. \quad (10.9)$$

Якщо процес двоетапний і на другому етапі розподіляється X_2 тис.грн., то на першому етапі буде $X - X_2$ і

$$F_2(X) = \max_{0 \leq X_2 \leq X} \{\varphi_2(X_2) + F_1(X - X_2)\}.$$

Очевидно, що для k -етапного процесу розподілу (на k -тому етапі розподіляється X_k тис.грн.) маємо рівняння

$$F_k(X) = \max_{0 \leq X_k \leq X} \{\varphi_k(X_k) + F_{k-1}(X - X_k)\}, \quad k = 2, 3, \dots, n-1 \quad (10.10)$$

Завдання для самоконтролю.

1. Сформулюйте загальну задачу динамічного програмування.
2. Які задачі можна розв'язувати методами динамічного програмування?
3. Дайте економічну інтерпретацію задачі динамічного програмування.
4. Сформулюйте принцип оптимальності, на якому ґрунтуються методи розв'язування задач динамічного програмування.
5. В чому суть методу функціональних рівнянь?

6. На реконструкцію і модернізацію виробництва підприємств $\Pi_1, \Pi_2, \dots, \Pi_n$ ($n=3$) галузі виділено кошти $Q=700$ тис.грн. Для кожного із n підприємств відомий можливий приріст $\varphi_i(x), i = \overline{1, n}$ випуску продукції в залежності від величини виділеної йому суми $X \in [0; Q]$ (табл.10.1). Потрібно розподілити кошти Q між підприємствами так, щоб загальний приріст випуску продукції на всіх n підприємствах був максимальним.

Таблиця 10.1.

Виділені кошти X_i , тис.грн	Приріст випуску продукції $\varphi_i(X_i)$ на підприємствах Π_i		
	Π_1	Π_2	Π_3
0	0	0	0
100	30	50	40
200	50	80	50
300	90	90	110

400	110	150	120
500	170	190	180
600	180	210	220
700	210	220	240

Розв'язання. Для розв'язування даної задачі динамічного програмування скористаємось функціональним рівнянням (9.26). Використовуючи ці рекурентні співвідношення і вихідні дані із табл.10.1, визначимо умовно-оптимальний розподіл коштів. Для цього за формулою (9.25) знаходимо обсяг коштів X , виділених підприємству Π_1 і значення функції $F_1(X)$ для кожного можливого значення $X \in \{0;100;200;300;400;500;600;700\}$:

$$X = 0 \Rightarrow F_1(0) = 0, X_1^0 = 0;$$

$$X = 100 \Rightarrow F_1(100) = \max\{0; 30\} = 30, X_1^0 = 100;$$

$$X = 200 \Rightarrow F_1(200) = \max\{0; 30; 50\} = 50, X_1^0 = 200;$$

$$X = 300 \Rightarrow F_1(300) = \max\{0; 30; 50; 90\} = 90, X_1^0 = 300;$$

$$X = 400 \Rightarrow F_1(400) = \max\{0; 30; 50; 90; 110\} = 110, X_1^0 = 400;$$

$$X = 500 \Rightarrow F_1(500) = \max\{0; 30; 50; 90; 110; 170\} = 170, X_1^0 = 500;$$

$$X = 600 \Rightarrow F_1(600) = \max\{0; 30; 50; 90; 110; 170; 180\} = 180, X_1^0 = 600;$$

$$X = 700 \Rightarrow F_1(700) = \max\{0; 30; 50; 90; 110; 170; 180; 210\} = 210, X_1^0 = 700.$$

Результати обчислень запишемо в табл.10.2.

Таблиця 10.2.

Обсяг коштів X , (тис.грн.) , виділених Π_1	Максимальний приріст $F_1(X)$ випуску продукції	Умовно- оптимальний обсяг коштів X_1^0 , виділених Π_1
0	0	0
100	30	100
200	50	200
300	90	300
400	110	400
500	170	500
600	180	600
700	210	700

Використовуючи тепер дані із табл. 10.1, 10.2, за формулою (9.26) при $k=2$ визначимо умовно-оптимальні обсяги капіталовкладень у підприємство Π_2 при кожному допустимому значенні X :

$$X = 0 \Rightarrow F_2(0) = 0, X_2^0 = 0;$$

$$X = 100 \Rightarrow F_2(100) = \max\{0 + 30; \underline{50} + 0\} = 50, X_2^0 = 100;$$

$$X = 200 \Rightarrow F_2(200) = \max\{0 + 50; \underline{50} + 30; 80 + 0\} = 80, X_2^0 = 100;$$

$$X = 300 \Rightarrow F_2(300) = \max\{0 + 90; 50 + 50; \underline{80 + 30}; 90 + 0\} = 110, X_2^0 = 200;$$

$$X = 400 \Rightarrow F_2(400) = \max\{0 + 110; 50 + 90; 80 + 50; 90 + 30; \underline{150 + 0}\} = 150,$$

$$X_2^0 = 400;$$

$$X = 500 \Rightarrow F_2(500) = \max\left\{\begin{array}{l} (0 + 170; 50 + 110; \underline{80 + 90}); \\ (90 + 50; \underline{130 + 30}; 190 + 0) \end{array}\right\} = 190, X_2^0 = 500;$$

$$X = 600 \Rightarrow F_2(600) = \max\left\{\begin{array}{l} (0 + 180; 50 + \underline{170}; 80 + 110; 90 + 90); \\ (150 + 50; 190 + 30; \underline{210 + 0}) \end{array}\right\} =$$

$$= 220, X_2^0 = 100;$$

$$X = 700 \Rightarrow F_2(700) = \max\left\{\begin{array}{l} (0 + 210; 50 + 180; \underline{80 + 170}); \\ (90 + 110; \underline{130 + 90}); \\ (190 + 50; \underline{210 + 30}; 220 + 0) \end{array}\right\} = 250, X_2^0 = 200.$$

Отримані результати записуємо в табл.10.3.

Таблиця 10.3.

Обсяг коштів X , виділені х двом підприємствам	Максимальн ий приріст $F_2(X)$ випуску продукції	Умовно- оптимальний обсяг коштів X_2^0 , виділених Π_2
0	0	0
100	50	100
200	80	100
300	110	200
400	150	400
500	190	500
600	220	100
700	250	200

Використовуючи дані табл.10.1 і 10.3 за формулою (9.26) при $k = 3$ обчислюємо

$$F_3(X) = \max_{0 \leq X_3 \leq X} \{\varphi_3(x_3) + F_2(X - X_3)\}..$$

Оскільки в даному випадку число підприємств дорівнює 3, то обчислення виконуємо лише для одного значення $X=700$:

$$F_3(700) = \max \left\{ \begin{array}{l} 0+250 \\ 40+220 \\ 50+190 \\ 110+150 \\ 120+210 \\ 180+80 \\ \underline{220+50} \\ 240+0 \end{array} \right\} = 270, \quad x_3^0 = 600.$$

Таким чином, максимальний приріст випуску продукції становить 270 тис.грн. Це буде тоді, коли підприємству Π_3 виділяється 600 тис.грн., а Π_1 і Π_2 – 100 тис.грн. Із табл.10.3 видно, що Π_2 потрібно виділити 100 тис.грн. Отже, одержали оптимальний план розподілу капіталовкладень між підприємствами галузі: $x_1^* = 0, x_2^* = 100, x_3^* = 600$ ●

Література.

1. *И.Л. Акулин*. Математическое программирование в примерах и задачах. – М.: Высш. шк., 1986. – 319 с.
2. *В.А. Балашевич*. Основы математического программирования. – Мн.: Высш. шк., 1985. – 173 с.
3. *Э.Х. Блох, Л.И. Лошинский, В.Я. Турин*. Основы линейной алгебры и некоторые ее приложения. – М.: Высш. шк., 1971.
4. *Дж. Данциг*. Линейное программирование, его обобщения и применение. – М.: Прогресс, 1966.
5. *С.И. Зуховицкий, Л.И. Авдеева*. Линейное и выпуклое программирование. – М.: Наука, 1967. – 460 с.
6. *В.А. Ильин, Э.Г. Позняк*. Линейная алгебра. – М.: Наука, 1978. – 304 с.
7. *И.Л. Калихман*. Сборник задач по математическому программированию. – М.: Высш. шк., 1975. – 270 с.
8. *В.Ф. Капустин*. Практические занятия по курсу математического программирования. – Л.: Изд-во ЛГУ, 1976. – 192 с.
9. *В.Г. Карманов*. Математическое программирование. – М.: Наука, 1975. – 151 с.
10. *А.В. Кузнецов, В.А. Сакович, Н.И. Холод и др.* Высшая математика: Математическое программирование. – Мн.: Высш. шк., 1994.
11. *Ю.Н. Кузнецов, В.И. Козубов, А.Б. Волощенко*. Математическое программирование. М.: Высш. шк., 1976. – 352 с.
12. Линейное и нелинейное программирование /Под ред. *И.Н. Ляшенко*. – К.: Высш. шк., 1975. – 371 с.
13. *М.А. Мартиненко, О.М. Нецадим, В.М. Сазонов*. Математичне програмування. - К.: Четверта хвиля, 2002. – 220 с.
14. Новые области применения математики /Под ред. *Дж. Лайтхилла*. – Мн.: Высш. шк., 1981. – 494 с.
15. Сборник задач и упражнений по высшей математике /Под ред. *А.В. Кузнецова*. – Мн.: Высш. шк., 1995. – 382 с.
16. *Дж. Хедли*. Нелинейное и динамическое программирование. – М.: Мир, 1967. – 506 с.
17. *Г.Г. Цегелик*. Лінійне програмування. – Львів: Світ, 1995. – 216 с.