

Ж. А. КРУТОВИЙ

С. В. ЛЮБАР

Н. В. МАНЖОС

ОПТИМІЗАЦІЯ ТЕХНОЛОГІЧНИХ ПРОЦЕСІВ

Частина 1

НАВЧАЛЬНИЙ ПОСІБНИК

ХАРКІВ 2011

МІНІСТЕРСТВО ОСВІТИ І НАУКИ,
МОЛОДІ ТА СПОРТУ УКРАЇНИ
ХАРКІВСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ХАРЧУВАННЯ ТА ТОРГІВЛІ

Ж. А. Крутовий

С. В. Любар

Н. В. Манжос

ОПТИМІЗАЦІЯ ТЕХНОЛОГІЧНИХ ПРОЦЕСІВ

Частина 1

МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ

НАВЧАЛЬНИЙ ПОСІБНИК

ХАРКІВ 2011

УДК 519.68(075.8)

ISBN

Крутовий Ж. А. та ін. Оптимізація технологічних процесів. Ч.1: Навч. посібник / Ж. А. Крутовий, **С. В. Любар**, Н. В. Манжос / Харк. держ. ун-т харч. та торгівлі. – Харків: ХДУХТ, 2011, – 300с.; 60 іл.; 61 табл.; 31 бібліогр. назв.

Рекомендовано Вченою Радою ХДУХТ (протокол №8 від 29.03.2011р.)

Перша частина посібника містить математичні моделі та методи розв'язування задач лінійного програмування, моделі теорії ігор та опуклого програмування.

Алгоритми розв'язування задач лінійного програмування, які найчастіше використовуються при дослідженні актуальних технологічних проблем, доведені до комп'ютерної реалізації з використанням пакета MathCAD.

З основних розділів посібника надано задачі для самостійного розв'язування в кількості, достатній для індивідуалізації навчального процесу.

Посібник призначено для студентів технологічних факультетів, аспірантів, наукових співробітників та викладачів вищих навчальних закладів III – IV рівнів акредитації. Може бути корисним економістам, які вивчають та застосовують економіко-математичні методи.

Рецензенти:

д-р фіз.-мат. наук, проф. Ольшанський В.П.
д-р техн. наук, проф. Янютін Є.Г.

ISBN

© Харківський державний університет харчування та торгівлі, 2011

© Крутовий Ж.А., **Любар С.В.**, Манжос Н.В., 2011.

ЗМІСТ

Передмова	6
Глава 1. Деякі попередні відомості	
з лінійної та векторної алгебри.....	8
1.1 Лінійні векторні простори	8
1.2 Розв'язування систем m лінійних рівнянь з n невідомими методом Жордана-Гаусса	14
1.3 Перетворення однократного заміщення. Базисні розв'язки ...	23
1.4 Опорні розв'язки системи лінійних рівнянь. Симплексні перетворення	25
1.5 Знаходження вихідного опорного розв'язку системи лінійних рівнянь	29
Питання для самоконтролю.....	32
Задачі до глави 1	32
Глава 2. Математичне моделювання	
задач лінійного програмування	35
2.1 Приклади математичних моделей задач лінійного програмування (ЛП)	35
2.2 Різні види моделей (форми запису) задач лінійного програмування	43
2.3 Геометричне тлумачення задачі ЛП. Графічний метод	51
Питання для самоконтролю.....	62
Задачі до глави 2	62
Глава 3. Симплексний метод розв'язування задач ЛП	65
3.1 Симплексний метод розв'язування не вироджених задач ЛП..	65
3.2 Альтернативний оптимум.....	77
3.3 Поняття про вироджені задачі лінійного програмування	79
3.4 Симплексний метод зі штучним базисом	81
3.5 Двоїстість (спряженість) у лінійному програмуванні	91
3.6 Основні теореми двоїстості	97

	Питання для самоконтролю.....	100
	Задачі до глави 3	101
Глава 4.	Математичне моделювання транспортних задач лінійного програмування.....	104
4.1	Загальні зауваження	104
4.2	Знаходження вихідного опорного розв'язку	107
4.3	Перетворення однократного заміщення. Цикл перерахунку ...	113
4.4	Алгоритм розв'язування ТЗ розподільним методом	119
4.5	Розв'язування транспортних задач методом потенціалів	121
4.6	Математичні моделі транспортних задач з неправильним балансом	133
	Питання для самоконтролю.....	140
	Задачі до глави 4	140
Глава 5.	Математичне моделювання матричних ігор та прийняття господарських рішень.....	148
5.1	Ввідні поняття та означення. Види ігор	148
5.2	Приклади математичного моделювання конфліктних ситуацій	154
5.3	Розв'язування матричних ігор в чистих стратегіях	159
5.4	Розв'язування матричних ігор в змішаних стратегіях. Задача Неймана.....	165
5.5	Розв'язування матричних ігор методами лінійного програмування	173
5.6	Математичне моделювання ігор та розв'язування господарських задач.....	180
	Питання для самоконтролю.....	192
	Задачі до глави 5	193
Глава 6.	Опукле програмування	201
6.1	Задачі нелінійного опуклого програмування	201

6.2	Основні методи розв’язування задач опуклого програмування	210
	Питання для самоконтролю.....	231
Глава 7.	Застосування пакета MathCAD	
	для розв’язування задач лінійного програмування	232
7.1	Основні відомості про систему MathCAD	232
7.2	Розв’язування задач графічним методом у системі MathCAD	235
7.3	Розв’язування задач симплексним методом у системі MathCAD	248
7.4	Розв’язування задачі ЛП зі штучним базисом у системі MathCAD	257
7.5	Розв’язування задач ЛП за допомогою вбудованих функцій MathCAD	262
7.6	Розв’язування транспортної задачі у системі MathCAD	271
7.7	Приклади розробки математичних моделей проектування оптимальних рецептур продуктів харчування, використання методів ЛП та комп’ютерних технологій	275
	Питання для самоконтролю.....	282
Додатки	284
Література	298

ПЕРЕДМОВА

Посібників, в яких викладені методи математичного програмування, досить багато. Проте чимало з них стали бібліографічною рідкістю, інші видані малим тиражем і практично недоступні студентам. Переважна більшість посібників видана російською мовою. Має місце певний дефіцит книг українською мовою.

В сучасних умовах в нашій країні стрімко розширюється сфера застосування методів математичного програмування, зокрема під час проведення наукових досліджень, виконання курсових та наукових проектів і т. ін. Виникає необхідність доведення алгоритмів розв'язання задач тим чи іншим методом до комп'ютерної реалізації.

Крім того, в умовах кредитно-модульної системи підготовки фахівців особливий акцент робиться на організацію ефективної самостійної роботи студентів. Ця обставина обумовлює доцільність доповнення теоретичних розділів задачами для самостійної роботи студентів.

Запропонований посібник написано на базі лекцій, практичних занять та лабораторних робіт, які протягом багатьох років здійснювались авторами з таких навчальних дисциплін: математичне програмування, економіко-математичні методи та моделі в торгівлі, математичні моделі на ЕОМ. Книга є апробованим, доповненим та допрацьованим варіантом посібника одного із авторів (Крутовой Ж.А. «Экономико-математические методы в торговле и общественном питании» часть 2, Харьков, 1991).

Автори ставили перед собою наступні цілі:

1) Об'єднати математичні методи, які на сьогоднішній день вивчаються в різних навчальних дисциплінах.

2) Поглибити інтеграційні міждисциплінарні зв'язки між математичними та спеціальними дисциплінами, зробити крок у напрямку створення інтегративної системи освіти на основі фундаментальної освіти.

3) Допомогти студенту не тільки оволодіти математичними методами, але й сформуванню певних практичних навичок розв'язання основних класів задач з використанням персональних комп'ютерів.

4) Створити дидактичні умови для самостійного оволодіння математичними методами та комп'ютерними технологіями.

5) Сприяти розвитку, по-перше, творчого мислення майбутніх фахівців, по-друге, процедурного мислення.

В книзі розглядаються методи лінійного програмування (ЛП), один із спеціальних класів задач ЛП – транспортні задачі та методи їх розв'язання, елементи теорії ігор та опуклого програмування.

Алгоритми розв'язання задач лінійного програмування доведені до комп'ютерної реалізації з використанням пакета MathCAD.

В кінці глав надано перелік питань для самоперевірки, а також (з основних розділів книги) задачі для самостійного розв'язування в кількості достатній для індивідуального навчального процесу.

Додатки до посібника містять детальний опис роботи в пакеті MathCAD: робота з графіками, перелік найбільш важливих операцій MathCAD і. т. ін.

В посібнику прийнято наступний спосіб нумерації. Номер параграфа складається з двох чисел: перше число відповідає номеру глави, друге – номеру параграфа в цій главі. Номери формул і задач складаються з трьох чисел, які означають: перше – номер глави, друге – номер параграфа, третє – номер формули або задачі в даному параграфі. Номери рисунків і таблиць складаються з двох чисел – номера глави та номера рисунка або таблиці в главі.

Автори щиро вдячні лауреату Державної премії України, доктору технічних наук, професору Синєкопу М.С. за здійснення загальної редакції посібника, підтримку та цінні зауваження, які сприяли підготовці книги до видання.

ГЛАВА 1.

ДЕЯКІ ПОПЕРЕДНІ ВІДОМОСТІ ІЗ ЛІНІЙНОЇ ТА ВЕКТОРНОЇ АЛГЕБРИ

Викладенню задач та методів лінійного програмування (ЛП) передують стислий розгляд деяких елементів лінійної та векторної алгебри, зокрема лінійних векторних просторів, метода Жордана-Гаусса, перетворення однократного заміщення, симплексних перетворень. Вказані питання є основою найбільш універсальних методів розв'язування задач лінійного програмування.

1.1 Лінійні векторні простори

Нехай вибрана декартова система координат відповідно на деякій прямій або на площині, або у тривимірному просторі. Точка O – початок координат. Тоді положення довільної точки M або, що те ж саме, радіусвектор \overrightarrow{OM} на вказаній прямій або на площині, або у тривимірному просторі визначається, як відомо із аналітичної геометрії, відповідно одним, двома, або трьома дійсними числами (координатами):

– на прямій $\overrightarrow{OM} = (x)$;

– на площині $\overrightarrow{OM} = (x, y)$;

– у тривимірному просторі $\overrightarrow{OM} = (x, y, z)$.

Кажуть: пряма – одновимірний, площина – двовимірний векторний простір. Будь-який вектор \vec{a} тривимірного простору визначається трьома координатами $\vec{a} = (x, y, z)$, або $\vec{a} = (x_1, x_2, x_3)$.

По аналогії з векторами в одновимірному, двовимірному та тривимірному просторах вводиться поняття вектора в n -вимірному просторі.

Означення. Сукупність n довільних дійсних чисел $\alpha_1, \alpha_2, \dots, \alpha_n$, записаних у певному порядку, називається n -вимірним вектором, який позначається так:

$$\vec{a} = (\alpha_1, \alpha_2, \dots, \alpha_n).$$

Числа α_i ($i = 1, 2, \dots, n$) називаються компонентами (складовими) або координатами вектора \vec{a} (або точки) у n -вимірному просторі.

Приклади n -вимірних векторів:

1. $\vec{X} = (x_1, x_2, \dots, x_n)$ – план випуску продукції підприємством (x_i – кількість продукції i -го виду);
2. $\vec{Y} = (y_1, y_2, \dots, y_m)$ – рецептура кондитерського виробу (y_j – кількість сировини j -го виду в рецептурі);
3. $\vec{Z} = (z_1, z_2, \dots, z_e)$ – план перевезення вантажу з деякого числа пунктів відправлення до деякого числа пунктів призначення.

Вектори $\vec{a} = (\alpha_1, \alpha_2, \dots, \alpha_n)$ і $\vec{b} = (\beta_1, \beta_2, \dots, \beta_n)$ називаються рівними, якщо рівні їх відповідні координати, тобто $\alpha_i = \beta_i$ при $i = 1, 2, \dots, n$.

Сумою (різницею) векторів $\vec{a} = (\alpha_1, \alpha_2, \dots, \alpha_n)$ і $\vec{b} = (\beta_1, \beta_2, \dots, \beta_n)$ називається вектор $\vec{a} + \vec{b} = (\alpha_1 + \beta_1, \alpha_2 + \beta_2, \dots, \alpha_n + \beta_n)$ або відповідно $\vec{a} - \vec{b} = (\alpha_1 - \beta_1, \alpha_2 - \beta_2, \dots, \alpha_n - \beta_n)$, координатами яких є суми (різниці) відповідних координат векторів \vec{a} і \vec{b} . Протилежним вектору $\vec{a} = (\alpha_1, \alpha_2, \dots, \alpha_n)$ називається вектор $-\vec{a} = (-\alpha_1, -\alpha_2, \dots, -\alpha_n)$. Нульовим називається вектор $\vec{0} = (0, 0, \dots, 0)$. Добутком вектора $\vec{a} = (\alpha_1, \alpha_2, \dots, \alpha_n)$ на дійсне число λ називається вектор $\lambda\vec{a} = (\lambda\alpha_1, \lambda\alpha_2, \dots, \lambda\alpha_n)$, координати якого дорівнюють відповідним координатам вектора \vec{a} , помноженим на число λ .

Перелічимо властивості лінійних операцій (додавання, віднімання та множення на дійсне число) над n -вимірними векторами:

1) комутативність

$$\vec{a} + \vec{b} = \vec{b} + \vec{a}, \quad \vec{\lambda a} = \vec{a\lambda};$$

2) асоціативність

$$\vec{a} + (\vec{b} + \vec{c}) = (\vec{a} + \vec{b}) + \vec{c};$$

3) дистрибутивність

$$\lambda(\vec{a} + \vec{b}) = \lambda\vec{a} + \lambda\vec{b}; \quad (\lambda + \mu)\vec{a} = \lambda\vec{a} + \mu\vec{a};$$

тут λ та μ – деякі дійсні числа.

Означення. Сукупність всіх n -вимірних векторів, над якими можуть бути виконані лінійні операції, утворюють лінійний n -вимірний простір.

Зауваження. Реальної геометричної інтерпретації n -вимірного вектору при $n > 3$ не існує.

Лінійна залежність n -вимірних векторів

Означення 1. Вектори $\vec{a}_1, \vec{a}_2, \dots, \vec{a}_m$ називаються лінійно залежними, якщо існують дійсні числа $\lambda_1, \lambda_2, \dots, \lambda_m$, серед яких хоч одне відмінне від нуля, такі що має місце рівність

$$\lambda_1 \vec{a}_1 + \lambda_2 \vec{a}_2 + \dots + \lambda_m \vec{a}_m = 0. \quad (1.1.1)$$

Означення 2. Вектори $\vec{a}_1, \vec{a}_2, \dots, \vec{a}_m$ називаються лінійно залежними, якщо хоч один з них є лінійною комбінацією всіх інших. Вектори $\vec{a}_1, \vec{a}_2, \dots, \vec{a}_m$, які не є лінійно залежними, називаються лінійно незалежними.

Означення 3. Вектори $\vec{a}_1, \vec{a}_2, \dots, \vec{a}_m$ називаються лінійно незалежними, якщо співвідношення (1.1.1) можливе тільки у випадку, коли всі дійсні числа $\lambda_1, \lambda_2, \dots, \lambda_m$ дорівнюють нулю.

Теорема 1. Необхідною та достатньою умовою лінійної залежності двох тривимірних векторів \vec{a} та \vec{b} є їх колінеарність, тобто вони лежать на одній або на паралельних прямих.

Висновок. Якщо \vec{a} та \vec{b} не колінеарні, то вони лінійно незалежні.

Теорема 2. Необхідною та достатньою умовою лінійної залежності трьох тривимірних векторів \vec{a}, \vec{b} і \vec{c} є їх компланарність, тобто вони лежать на одній або на паралельних площинах.

Висновок 1. Якщо вектори \vec{a}, \vec{b} і \vec{c} не компланарні, то вони лінійно незалежні.

Висновок 2. Нехай \vec{a} та \vec{b} деякі неколінеарні вектори. Тоді будь-який вектор \vec{c} , що лежить в одній площині з векторами \vec{a} та \vec{b} , можна зобразити у виді

$$\vec{c} = \lambda_1 \vec{a} + \lambda_2 \vec{b}, \quad (1.1.2)$$

де λ_1 та λ_2 – дійсні числа.

Теорема 3. Нехай \vec{a}, \vec{b} і \vec{c} – деякі некопланарні тривимірні вектори, тоді будь-який тривимірний вектор \vec{d} можна зобразити у виді

$$\vec{d} = \lambda_1 \vec{a} + \lambda_2 \vec{b} + \lambda_3 \vec{c}, \quad (1.1.3)$$

де $\lambda_1, \lambda_2, \lambda_3$ – дійсні числа.

Теорема 4. Якщо в системі векторів $\vec{a}_1, \vec{a}_2, \dots, \vec{a}_m$ є нульовий вектор, то ці вектори є лінійно залежними.

Теорема 5. Якщо серед m векторів $\vec{a}_1, \vec{a}_2, \dots, \vec{a}_m$ будь-які l векторів ($l < m$) є лінійно залежними, то і всі m векторів також лінійно залежні.

Теорема 6. Будь-яка сукупність $n + 1$ векторів n -вимірного простору є лінійно залежною.

Розглянемо систему одиничних векторів n -вимірного векторного простору:

$$\begin{aligned}
\vec{e}_1 &= (1, 0, 0, \dots, 0), \\
\vec{e}_2 &= (0, 1, 0, \dots, 0), \\
\vec{e}_3 &= (0, 0, 1, \dots, 0), \\
&\dots \\
\vec{e}_n &= (0, 0, 0, \dots, 1).
\end{aligned}
\tag{1.1.4}$$

Теорема 7. Одиничні n -вимірні вектори $\vec{e}_1, \vec{e}_2, \dots, \vec{e}_n$ є лінійно незалежними.

Доведення. Розглянемо суму добутків цих векторів на дійсні числа $\lambda_1, \lambda_2, \dots, \lambda_n$:

$$\lambda_1 \vec{e}_1 + \lambda_2 \vec{e}_2 + \dots + \lambda_n \vec{e}_n = (\lambda_1, \lambda_2, \dots, \lambda_n). \tag{1.1.5}$$

Неважко побачити, що лінійна комбінація (1.1.5) векторів $\vec{e}_1, \vec{e}_2, \dots, \vec{e}_n$ буде рівною нулю тільки тоді, коли всі числа λ_i ($i = 1, 2, \dots, n$) будуть рівними нулю. Отже, n одиничних, n -вимірних векторів (1.1.1) є лінійно незалежними.

Ранг та базис n -вимірного векторного простору

Означення 1. Рангом r системи векторів $\vec{a}_1, \vec{a}_2, \dots, \vec{a}_m$ називається найбільше число лінійно незалежних векторів цієї системи.

Теорема 1. Ранг n -вимірного векторного простору дорівнює n , тобто вимірності простору.

Означення 2. Базисом n -вимірного векторного простору називається сукупність будь-яких n лінійно незалежних n -вимірних векторів.

Теорема 2. Будь який вектор \vec{a} n -вимірного простору можна розкласти, причому тільки одним способом, за векторами будь-якого базису цього простору, тобто його можна зобразити у вигляді лінійної комбінації векторів базису

пунктирними лініями). З елемента a_{ik} , розташованого на головній діагоналі, віднімається добуток елементів побічної діагоналі, поділений на розв'язувальний елемент a_{qp} .

Перша ітерація завершується обчисленням за «правилом прямокутників» нових коефіцієнтів всіх рівнянь, за винятком розв'язувального.

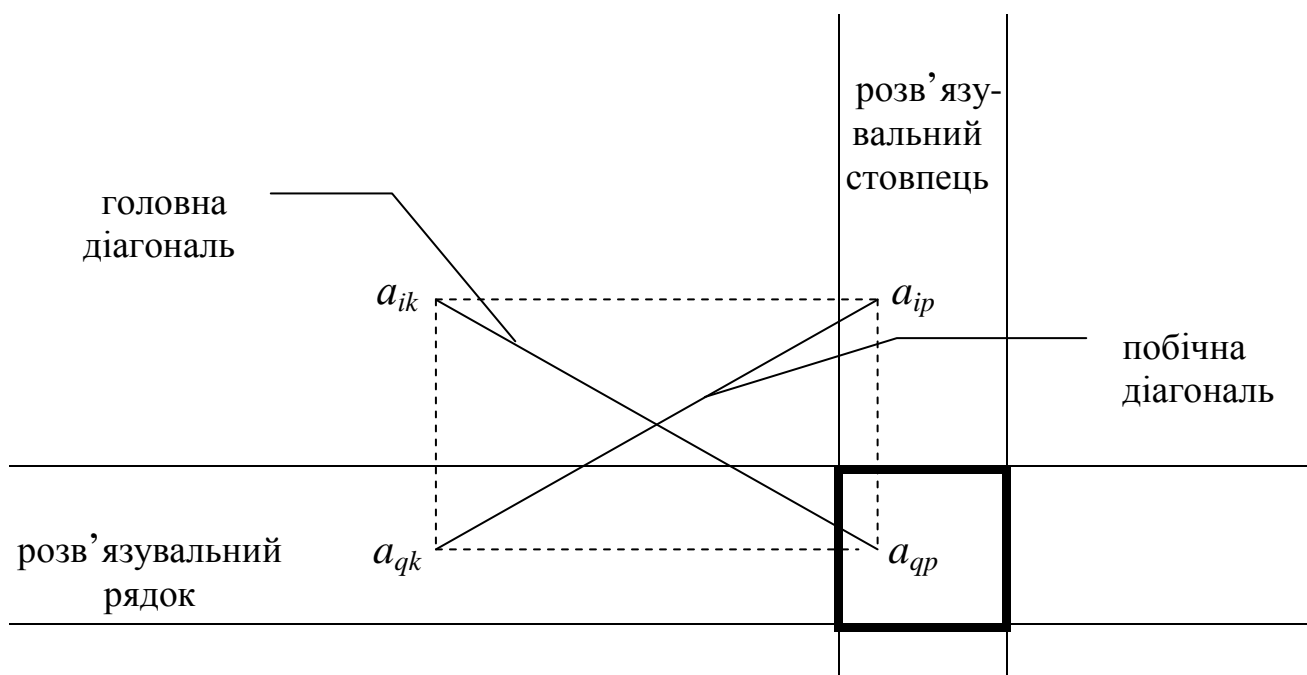


Рис. 1.1.

На другій ітерації вибирається новий розв'язувальний елемент у будь-якому рівнянні, крім q -го, деяка інша невідома (не x_p) виключається із усіх рівнянь, крім розв'язувального, і т. ін.

У процесі послідовного виключення невідомих можливі такі випадки:

1) Деяке рівняння «зникає», тобто приводиться до виду

$$0 \cdot x_1 + 0 \cdot x_2 + \dots + 0 \cdot x_n = 0. \quad (1.2.5)$$

Це означає, що вказане рівняння є лінійною комбінацією інших. Наприклад, у системі наведеній нижче, третє рівняння є сумою першого та другого рівнянь:

$$\begin{cases} x_1 + x_2 = 5; \\ 2x_1 - 3x_2 = 7; \\ 3x_1 - 2x_2 = 12. \end{cases} \quad (1.2.6)$$

Якщо цю систему розв'язувати методом Жордана-Гаусса, то третє рівняння може «зникнути».

2) Одержимо рівняння, в якому ліва частина дорівнює нулю, а права – відмінна від нуля

$$0 \cdot x_1 + 0 \cdot x_2 + \dots + 0 \cdot x_n = b_{i0} \quad (b_{i0} \neq 0). \quad (1.2.7)$$

Це означає, що система несумісна, тобто не має розв'язку.

У разі сумісності системи, число ітерацій дорівнює рангу системи, тобто максимальному числу лінійно незалежних рівнянь.

Система може мати єдиний розв'язок. Тоді вона має вид

$$\begin{cases} x_1 & = b_{10}; \\ x_2 & = b_{20}; \\ & \ddots \quad \dots \\ x_r & = b_{r0}. \end{cases} \quad (1.2.8)$$

де x_1, x_2, \dots, x_r – базисні невідомі.

Якщо система має і вільні невідомі, тоді базисні невідомі можна виразити через вільні коефіцієнти і вільні невідомі, тобто:

$$\begin{cases} x_1 = b_{10} - b_{1,r+1}x_{r+1} - \dots - b_{1,n}x_n, \\ x_2 = b_{20} - b_{2,r+1}x_{r+1} - \dots - b_{2,n}x_n, \\ \dots \\ x_r = b_{r0} - b_{r,r+1}x_{r+1} - \dots - b_{r,n}x_n. \end{cases} \quad (1.2.9)$$

Тоді така система має нескінченну кількість розв'язків.

В цьому випадку можна одержувати багато частинних розв'язків, надаючи довільні значення вільним невідомим.

Приклад 1. Розв'язати систему рівнянь методом Жордана-Гаусса або довести її несумісність:

$$\begin{cases} 2x_1 + 3x_2 + 4x_3 + x_4 = 2, \\ x_1 + x_2 + 7x_3 + x_4 = 6, \\ 3x_1 + 2x_2 + x_3 + 5x_4 = 8. \end{cases} \quad (1.2.10)$$

Розв'язання. Коефіцієнти при невідомих, а також вільні члени рівнянь запишемо у таблицю Гаусса (табл. 1.1). Зауважимо, що правий стовпець введено для контролю. Елементи цього стовпця для початкової системи одержано як алгебраїчну суму всіх інших елементів рядка.

Прямокутниками обведено розв'язувальні елементи, які вибираються на кожній ітерації.

У процесі виконання ітерацій елементи таблиць, в тому числі контрольного стовпця, обчислюються за формулами (1.2.3) та (1.2.4) або з використанням «правила прямокутників». Контроль правильності обчислень здійснюється після кожної ітерації. Якщо обчислення виконано правильно, то сума елементів кожного рядка, крім контрольного, повинна дорівнювати знайденому раніше контрольному елементу цього рядка.

Перша ітерація. Як розв'язувальний елемент виберемо $a_{21} = 1$ (його взяли в рамку). Невідому x_1 виключимо із всіх рівнянь, крім другого. Для цього елементи першого і третього рядка обчислимо, керуючись «правилом прямокутників». В результаті виконання першої ітерації друге рівняння розв'язано відносно x_1 .

Друга ітерація. Розв'язувальний елемент $a_{12} = 1$. Невідому x_2 виключимо із усіх рівнянь, крім першого.

Третя ітерація. Розв'язувальний елемент $a_{34} = 1$. Невідому x_4 виключимо із усіх рівнянь, крім третього.

Таблиця 1.1.

№ ітерації	x_1	x_2	x_3	x_4	a_{i0}	$\sum_j a_{ij}$
Початкова система	2	3	4	1	2	12
	1	1	7	1	6	16

	3	2	1	5	8	19
I	0	1	-10	-1	-10	-20
	1	1	7	1	6	16
	0	-1	-20	2	-10	-29
II	0	1	-10	-1	-10	-20
	1	0	17	2	16	36
	0	0	-30	1	-20	-49
III	0	1	-40	0	-30	-69
	1	0	77	0	56	134
	0	0	-30	1	-20	-49

В результаті здійснення трьох ітерацій методом Жордана-Гаусса система рівнянь (1.2.10) має вид

$$\begin{cases} x_2 - 40x_3 = -30; \\ x_1 + 77x_3 = 56; \\ -30x_3 + x_4 = -20. \end{cases} \quad (1.2.11)$$

або

$$\begin{cases} x_1 + 77x_3 = 56; \\ x_2 - 40x_3 = -30; \\ -30x_3 + x_4 = -20. \end{cases} \quad (1.2.11)'$$

або

$$\begin{cases} x_1 + 77x_3 = 56; \\ x_2 - 40x_3 = -30; \\ x_4 - 30x_3 = -20. \end{cases} \quad (1.2.11)''$$

тобто приведена до одиничного базису.

Загальний розв'язок:

$$\begin{aligned} x_1 &= 56 - 77x_3, \\ x_2 &= -30 + 40x_3, \\ x_4 &= -20 + 30x_3. \end{aligned} \quad (1.2.12)$$

Якщо візьмемо $x_3 = 0$, то одержимо такий частинний розв'язок $\bar{X}_1 = (56; -30; 0; -20)$. При $x_3 = 1$ частинний розв'язок має вид $\bar{X}_2 = (-21; 10; 1; 10)$.

Приклад 2. Розв'язати систему лінійних рівнянь

$$\begin{cases} 3x_1 - x_2 + 4x_3 + x_4 = 5, \\ x_1 + 2x_2 - 2x_3 - x_4 = 23, \\ 2x_1 + x_2 + 3x_3 - 3x_4 = 24, \\ x_1 - x_2 + 5x_3 - 2x_4 = 1. \end{cases} \quad (1.2.13)$$

Розв'язування наведено у таблиці 1.2.

Таблиця 1.2

№ ітерації	x_1	x_2	x_3	x_4	a_{i0}	$\sum_j a_{ij}$
Початкова система	3	-1	4	1	5	12
	1	2	-2	-1	23	23
	2	1	3	-3	24	27
	1	-1	5	-2	1	4
I	3	-1	4	1	5	12
	4	1	2	0	28	35
	11	-2	15	0	39	63
	7	-3	13	0	11	28
II	7	0	6	1	33	47
	4	1	2	0	28	35
	19	0	19	0	95	133
	19	0	19	0	95	133
III	7	0	6	1	33	47
	4	1	2	0	28	35
	19	0	19	0	95	133
	1	0	1	0	5	7
	0	0	-1	1	-2	-2
	0	1	-2	0	8	7
	0	0	0	0	0	0
1	0	1	0	5	7	

Розв'язувальні елементи вибирались у такій послідовності: на першій ітерації $a_{14} = 1$, на другій – $a_{22} = 1$, на третій – $a_{41} = 19$. В результаті виконання третьої ітерації всі коефіцієнти третього рівняння стали рівними нулю. Це означає, що третє рівняння початкової системи є лінійною комбінацією першого, другого та четвертого рівнянь. Розв'язок системи рівнянь, які залишились, буде також розв'язком і третього рівняння початкової системи.

Система має такий вид (приведена до одиничного базису)

$$\begin{cases} -x_3 + x_4 = -2; \\ x_2 - 2x_3 = 8; \\ x_1 + x_3 = 5, \end{cases} \quad (1.2.14)$$

або

$$\begin{cases} x_1 + x_3 = 5; \\ x_2 - 2x_3 = 8; \\ -x_3 + x_4 = -2; \end{cases} \quad (1.2.14)'$$

або

$$\begin{cases} x_1 + x_3 = 5, \\ x_2 - 2x_3 = 8, \\ x_4 - x_3 = 2. \end{cases} \quad (1.2.14)''$$

Знаходимо загальний розв'язок системи (1.2.13)

$$\begin{cases} x_1 = 5 - x_3, \\ x_2 = 8 + 2x_3, \\ x_4 = -2 + x_3. \end{cases} \quad (1.2.15)$$

Невідомі x_1, x_2, x_4 є базисними, x_3 – вільна невідома. Їй можна надавати будь-які довільні значення. Задаючись різними значеннями x_3 , неважко одержати частинні розв'язки (їх нескінченність) початкової системи рівнянь.

Наприклад, при $x_3 = 0$ маємо $x_1 = 5, x_2 = 8, x_4 = -2$, або у векторній формі $\bar{X}_1 = (5; 8; 0; -2)$. При $x_3 = 7$ одержимо $x_1 = -2, x_2 = 22, x_4 = 5$ або $\bar{X}_2 = (-2; 22; 7; 5)$.

Приклад 3. Розв'язати систему лінійних рівнянь

$$\begin{cases} x_1 + 2x_2 - x_3 + 3x_4 = 5, \\ -3x_1 + x_2 + 2x_3 + 4x_4 = 10, \\ -2x_1 + 3x_2 + x_3 + 7x_4 = 6. \end{cases} \quad (1.2.16)$$

Розв'язування наведено у таблиці 1.3

Таблиця 1.3

№	x_1	x_2	x_3	x_4	a_{i0}	$\sum a_{ij}$
---	-------	-------	-------	-------	----------	---------------

ітерації						
Початкова система	1	2	-1	3	5	10
	-3	1	2	4	10	14
	-2	3	1	7	6	15
I	1	2	-1	3	5	10
	0	7	-1	13	25	44
	0	7	-1	13	16	35
II	1	-5	0	-10	-20	-34
	0	-7	1	-13	-25	-44
	0	0	0	0	-9	-9

В результаті здійснення двох ітерацій одержали суперічність: $a_{31} = a_{32} = a_{33} = a_{34} = 0$, при цьому $a_{30} = -9$. Отже, початкова система лінійних рівнянь несумісна.

Відповідь на поставлені та інші питання легко знайти за допомогою перетворення однократного заміщення, яке базується на методі Жордана-Гаусса.

Суть методу однократного заміщення: із числа коефіцієнтів при вільних невідомих в системі, яка уже приведена до одиничного базису, вибирається коефіцієнт, наприклад a_{qp} , як розв'язувальний елемент. Далі виключається невідома x_p із всіх рівнянь, окрім q -го. Коефіцієнти q -го рівняння обчислюють за уже відомою формулою (1.2.3), а коефіцієнти інших рівнянь, окрім q -го, за формулою (1.2.4).

В результаті q -те рівняння стає розв'язаним відносно невідомої x_p , тобто невідома x_p стає базисною (до цього вона була вільною), а невідома x_q , яка була базисною, стає вільною. Інших замінь у складі базисних та вільних невідомих, окрім вказаного однократного заміщення, не відбулося. Отже, одержимо новий набір базисних і вільних невідомих.

Нові вільні невідомі дорівнюємо нулю. Тоді одержимо новий базисний розв'язок, у якому базисні невідомі дорівнюють правим частинам системи.

Приклад. Знайти два базисних розв'язки системи лінійних рівнянь

$$\begin{cases} x_1 & - x_4 + x_5 = 4; \\ x_2 & + 2x_4 - x_5 = 3; \\ x_3 & + x_4 + 2x_5 = 2. \end{cases} \quad (1.3.2)$$

Розв'язання. Ця система приведена до одиничного базису, тому, надаючи вільним невідомим значення нуль, можна записати початковий базисний розв'язок $\vec{X}_1 = (4; 3; 2; 0; 0)$.

Здійснивши перетворення однократного заміщення, знайдемо інший базисний розв'язок. Для цього скористаємось таблицею Гаусса, доповненою двома стовпцями: стовпцем базисних невідомих, відносно яких розв'язані відповідні рівняння системи, а також стовпцем, у якому в векторній формі записуються базисні розв'язки (табл. 1.4). Розв'язувальним елементом

вибрано $a_{15} = 1$. В результаті однократного заміщення система рівнянь приведена до нового одиничного базису: x_5, x_2, x_3 – базисні невідомі, x_1, x_4 – вільні.

Таблиця 1.4

№ ітерації	Базисні невідомі	x_1	x_2	x_3	x_4	x_5	a_{i0}	$\sum_j a_{ij}$	Базисні розв'язки
Початкова система	x_1	1	0	0	-1	1	4	5	$\vec{X}_1 = (4; 3; 2; 0; 0)$
	x_2	0	1	0	2	-1	3	5	
	x_3	0	0	1	1	2	2	6	
I	x_5	1	0	0	-1	1	4	5	$\vec{X}_2 = (0; 7; -6; 0; 4)$
	x_2	1	1	0	1	0	7	10	
	x_3	-2	0	1	3	0	-6	-4	

Новий базисний розв'язок має вид: $x_1 = 0, x_4 = 0, x_5 = 4, x_2 = 7, x_3 = -6$, або $\vec{X}_2 = (0; 7; -6; 0; 4)$.

Перебираючи різні набори базисних невідомих, можна одержати різні базисні розв'язки. При цьому треба слідкувати, щоб новий набір базисних невідомих не збігався з одним із тих, які вже були.

1.4 Опорні розв'язки системи лінійних рівнянь.

Симплексні перетворення

Означення 1. Допустимим розв'язком системи лінійних рівнянь називається розв'язок, в якому всі невідомі невід'ємні.

Означення 2. Опорним розв'язком називається допустимий базисний розв'язок, тобто розв'язок, у якому вільні невідомі дорівнюють нулю, а базисні – невід'ємні. Розглянемо систему m лінійних алгебраїчних рівнянь з n невідомими, приведену до одиничного базису

$$\left\{ \begin{array}{l} x_1 \quad \quad \quad + a_{1,r+1} x_{r+1} + \dots + a_{1,p} x_p + \dots + a_{1,n} x_n = a_{10}, \\ \quad x_2 \quad \quad \quad + a_{2,r+1} x_{r+1} + \dots + a_{2,p} x_p + \dots + a_{2,n} x_n = a_{20}, \\ \quad \quad \quad \dots \quad \quad \quad \dots \quad \quad \quad \dots \quad \quad \quad \dots \quad \quad \quad \dots \quad \quad \quad \dots \\ \quad \quad \quad x_q \quad \quad \quad + a_{q,r+1} x_{r+1} + \dots + a_{q,p} x_p + \dots + a_{q,n} x_n = a_{q0}, \\ \quad \quad \quad \dots \quad \quad \quad \dots \quad \quad \quad \dots \quad \quad \quad \dots \quad \quad \quad \dots \quad \quad \quad \dots \\ x_r + a_{r,r+1} x_{r+1} + \dots + a_{r,p} x_p + \dots + a_{r,n} x_n = a_{r0}. \end{array} \right. \quad (1.4.1)$$

де x_1, x_2, \dots, x_r – базисні невідомі, а $x_{r+1}, x_{r+2}, \dots, x_n$ – вільні невідомі. Припустимо, що праві частини всіх рівнянь – невід’ємні. Якщо вільні невідомі дорівняти нулю, то базисні невідомі будуть дорівнювати невід’ємним правим частинам, тобто знайдений базисний розв’язок буде опорним.

Інші опорні розв’язки цієї системи, якщо вони існують, можна знайти з допомогою так званих симплексних перетворень системи рівнянь.

Симплексні перетворення системи – це перетворення однократного заміщення, які доповнені такими правилами вибору розв’язувального елемента:

1) за розв’язувальний вибирається той стовпець коефіцієнтів при вільних невідомих, наприклад x_p , в якому є хоча б один додатній елемент ($a_{ip} > 0$);

2) розв’язувальний елемент a_{qp} у вибраному стовпці обов’язково повинен бути додатнім ($a_{qp} > 0$) і вибиратись із умови

$$\frac{a_{q0}}{a_{qp}} = \min_{a_{ip} > 0} \left(\frac{a_{i0}}{a_{ip}} \right) \quad (1.4.2)$$

Теорема. Нехай в системі лінійних алгебраїчних рівнянь, яка приведена до одиничного базису, праві частини невід’ємні. Тоді в результаті симплексних перетворень ці праві частини залишаються невід’ємними.

Доведення. Припустимо, що розв’язувальним елементом при симплексному перетворенні вибрано елемент $a_{qp} > 0$. Згідно з умовою теореми

всі праві частини a_{i0} невід'ємні, в тому числі $a_{q0} \geq 0$. Тоді нова права частина розв'язувального рівняння, що визначається за формулою

$$a_{q0}^* = \frac{a_{q0}}{a_{qp}}, \quad (1.4.3)$$

також невід'ємна, оскільки знаменник у правій частині (1.4.1) додатній, а чисельник невід'ємний. Обчислення правих частин інших рівнянь здійснюється за формулою

$$a_{i0}^* = a_{i0} - a_{ip} \cdot \frac{a_{q0}}{a_{qp}}. \quad (1.4.4)$$

Для визначення знаку a_{i0}^* розглянемо 3 випадки.

1) Нехай $a_{ip} < 0$. Тоді обидва доданки в правій частині (1.4.4) невід'ємні, оскільки $a_{q0} \geq 0$. Отже, $a_{i0}^* \geq 0$.

2) Нехай $a_{ip} > 0$. Перепишемо (1.4.4) так

$$a_{i0}^* = a_{ip} \left(\frac{a_{i0}}{a_{ip}} - \frac{a_{q0}}{a_{qp}} \right). \quad (1.4.5)$$

Кожен доданок виразу в дужках в (1.4.5) невід'ємний. Виходячи із умови (1.4.2) вибору розв'язувального елемента, запишемо

$$\frac{a_{q0}}{a_{qp}} \leq \frac{a_{i0}}{a_{ip}}.$$

Враховуючи сказане, а також припущення $a_{ip} > 0$, одержимо $a_{i0}^* \geq 0$.

3) Нехай $a_{ip} = 0$. Тоді $a_{i0}^* = a_{i0} \geq 0$.

Теорема доведена. Отже, доведено, що за допомогою симплексних перетворень системи рівнянь неважко здійснити перехід від одного опорного розв'язку до іншого.

Для одержання всіх опорних розв'язків необхідно процедуру симплексного перетворення системи повторити скінченне число раз. При цьому, вибираючи розв'язувальний елемент, необхідно слідкувати, як і при

отримуванні всіх базисних розв'язків, за складом базисних та вільних невідомих. Це необхідно для того, щоб розв'язок, який намагаємось одержати, не збігався з одним із розв'язків, що були отримані раніше.

Зауваження. Якщо умові (1.4.2) задовольняє декілька додатних елементів, то за розв'язувальний вибирається будь-який з них.

Приклад. Знайти два опорних розв'язки системи лінійних рівнянь

$$\begin{cases} x_1 + x_4 + 2x_5 = 2, \\ x_2 - x_4 + x_5 = 3, \\ x_3 + 3x_4 - x_5 = 4. \end{cases} \quad (1.4.6)$$

Розв'язання надано у таблиці Гаусса (табл. 1.5), що доповнена трьома стовпцями. У першому записані базисні невідомі, відносно яких розв'язано кожне з рівнянь системи, у другому додатковому стовпці містяться значення відношень $\frac{a_{i0}}{a_{ip}}$, у третьому – опорні розв'язки у векторній формі.

За розв'язувальний вибрано стовпець коефіцієнтів при невідомій x_5 . Розв'язувальний елемент $a_{15} = 2$ (його взято у рамку).

Знайдені опорні розв'язки мають вид:

– перший розв'язок $x_4 = 0, x_5 = 0, x_1 = 2, x_2 = 3, x_3 = 4$, або

$$\vec{X}_1 = (2; 3; 4; 0; 0);$$

– другий розв'язок $x_1 = 0, x_4 = 0, x_5 = 1, x_2 = 2, x_3 = 5$, або

$$\vec{X}_2 = (0; 2; 5; 0; 1);$$

Таблиця 1.5

№ ітерації	Базисні невідомі	x_1	x_2	x_3	x_4	x_5	a_{i0}	$\sum_j a_{ij}$	$\frac{a_{i0}}{a_{ip}}$	Опорні розв'язки
Початкова система	x_1	1	0	0	1	2	2	6	2/2	$\vec{X}_1 = (2; 3; 4; 0; 0)$
	x_2	0	1	0	-1	1	3	4	3/1	
	x_3	0	0	1	3	-1	4	7	–	
I	x_5	1/2	0	0	1/2	1	1	3	2	$\vec{X}_2 = (0; 2; 5; 0; 1)$
	x_2	-1/2	1	0	-3/2	0	2	1	–	
	x_3	1/2	0	1	7/2	0	5	10	10/7	

Якщо опорних розв'язків немає, це означає, що задача не має розв'язків в області допустимих значень.

1.5 Знаходження вихідного опорного розв'язку системи лінійних рівнянь

Вище було розглянуто, як систему лінійних алгебраїчних рівнянь, вже приведену до одиничного базису, причому рівнянь з невід'ємними правими частинами, можна привести до іншого одиничного базису, зберігши при цьому праві частини невід'ємними. Або коротко: розглянуто перетворення системи, яке дозволяє здійснювати перехід від одного опорного розв'язку до іншого. Виникає питання: а як одержати перший (вихідний) опорний розв'язок?

Вихідний опорний розв'язок системи m лінійних алгебраїчних рівнянь з n невідомими можна знайти у такий спосіб:

- 1) рівняння з від'ємними правими частинами помножують на -1 , з тим, щоб всі рівняння були з невід'ємними правими частинами;
- 2) знайдену систему приводять до одиничного базису за допомогою симплексних перетворень.

Зауваження. Якщо систему не можна привести до одиничного базису за допомогою симплексних перетворень, наприклад, з причини неможливості вибору розв'язувального елемента, то система не має розв'язку в області допустимих значень, тобто не має опорних розв'язків.

Приклад 1. Знайти вихідний опорний розв'язок системи лінійних алгебраїчних рівнянь

$$\begin{cases} x_1 - x_2 + x_3 - x_4 + x_5 = 6, \\ -2x_2 + x_3 + 3x_4 = -4, \\ 3x_1 + 2x_3 + x_4 = 2. \end{cases} \quad (1.5.1)$$

Вихідний опорний розв'язок зручно шукати з використанням таблиць Гаусса.

Розв'язання. Ліву та праву частини другого рівняння помножимо на -1 .

В результаті система (1.5.1) має вид

$$\begin{cases} x_1 - x_2 + x_3 - x_4 + x_5 = 6, \\ 2x_2 - x_3 - 3x_4 = 4, \\ 3x_1 + 2x_3 + x_4 = 2. \end{cases} \quad (1.5.2)$$

Приведення одержаної системи (1.5.2) до одиничного базису за допомогою симплексних перетворень надано у таблиці Гаусса (табл. 1.6).

Розв'язувальний елемент на кожній ітерації взято в рамку.

Таблиця 1.6

№ ітерації	x_1	x_2	x_3	x_4	x_5	a_{i0}	$\sum_j a_{ij}$	$\frac{a_{i0}}{a_{ip}}$	Опорний розв'язок
Початкова система	1	-1	1	-1	1	6	7	-	
	0	2	-1	-3	0	4	2	-	
	3	0	2	1	0	2	8	2/1	
I	4	-1	3	0	1	8	15		
	9	2	5	0	0	10	26		
	3	0	2	1	0	2	8		
II	4	-1	3	0	1	8	15	-	
	9/2	1	5/2	0	0	5	13	5/1	
	3	0	2	1	0	2	8	-	
	17/2	0	11/2	0	1	13	28		
	9/2	1	5/2	0	0	5	13		$\vec{X}_0 = (0; 5; 0; 2; 13)$
	3	0	2	1	0	2	8		

Приклад 2. Знайти вихідний опорний розв'язок системи рівнянь

$$\begin{cases} x_1 + x_2 - x_3 - 2x_4 = -2, \\ 2x_1 + x_2 - x_3 + x_4 = 4, \\ 3x_1 + 2x_2 + x_3 - 3x_4 = 1. \end{cases} \quad (1.5.3)$$

Розв'язання. Ліву та праву частини першого рівняння помножимо на -1 .

Реалізація алгоритму знаходження вихідного опорного розв'язку системи (1.5.3) наведена у таблиці 1.7.

Таблиця 1.7

--	--	--	--	--	--	--	--

№ ітерації	x_1	x_2	x_3	x_4	a_{i0}	$\sum_j a_{ij}$	$\frac{a_{i0}}{a_{ip}}$
Початкова система	-1	-1	1	2	2	3	2
	2	1	-1	1	4	7	-
	3	2	1	-3	1	4	1
I	-4	-3	0	5	1	-1	1/5
	5	3	0	-2	5	11	-
	3	2	1	-3	1	4	-
II	-4/5	-3/5	0	1	1/5	-1/5	
	5	3	0	-2	5	11	
	3	2	1	-3	1	4	
	-4/5	-3/5	0	1	1/5	-1/5	-
	17/5	9/5	0	0	27/5	53/5	3
3/5	1/5	1	0	8/5	17/5	8	
III	-4/5	-3/5	0	1	1/5	-1/5	
	17/9	1	0	0	3	53/9	
	3/5	1/5	1	0	8/5	17/5	
	1/3	0	0	1	2	10/3	
	17/9	1	0	0	3	53/9	
2/9	0	1	0	1	20/9		

Остаточно система рівнянь має вид

$$\begin{cases} \frac{1}{3} \cdot x_1 & + x_4 = 2, \\ \frac{17}{9} \cdot x_1 + x_2 & = 3, \\ \frac{2}{9} \cdot x_1 & + x_3 = 1. \end{cases} \quad (1.5.4)$$

Знаходимо загальний розв'язок системи

$$\begin{cases} x_2 = 3 - \frac{17}{9} \cdot x_1, \\ x_3 = 1 - \frac{2}{9} \cdot x_1, \\ x_4 = 2 - \frac{1}{3} \cdot x_1. \end{cases} \quad (1.5.5)$$

та вихідний опорний розв'язок

$$x_1 = 0, x_2 = 3, x_3 = 1, x_4 = 2,$$

або

$$\vec{X}_2 = (0; 3; 1; 2).$$

Питання для самоконтролю

1. Що таке вектор?

2. Що таке лінійні операції над векторами?
3. Що таке векторний простір?
4. За якої умови система векторів називається лінійно незалежною?
5. Що називається базисом n -вимірному векторного простору?
6. В чому сутність методу Жордана-Гаусса?
7. В чому сутність перетворення однократного заміщення?
8. Який розв'язок системи лінійних алгебраїчних рівнянь (СЛАР) називається базисним?
9. Як знайти всі базисні розв'язки СЛАР?
10. Який розв'язок називається опорним?
11. В чому сутність симплексного перетворення СЛАР?
12. Як знайти вихідний опорний розв'язок СЛАР?
13. Як знайти всі опорні розв'язки СЛАР?

Задачі до глави 1

1.1. Розв'язати методом Жордана-Гаусса або довести несумісність наступних СЛАР:

1.

$$\begin{cases} x_1 + x_2 - 4x_3 + 2x_4 = 5, \\ -x_1 + 2x_2 + 3x_3 - x_4 = 2, \\ 2x_1 - x_2 + 5x_3 + x_4 = 1, \\ 4x_1 + x_2 - 3x_3 + 5x_4 = 11. \end{cases}$$

2.

$$\begin{cases} 6x_1 + x_2 - x_3 + x_4 = 6, \\ 2x_1 - x_2 + 2x_3 + 2x_4 = -4, \\ 3x_1 + 2x_2 + x_3 + x_4 = 6, \\ 4x_1 + x_2 - x_3 - x_4 = 4. \end{cases}$$

3.

$$\begin{cases} 2x_1 + x_2 - x_3 + x_4 = 3, \\ x_1 - x_2 + x_3 + 2x_4 = 6, \\ -x_1 + x_2 + x_3 = 4, \\ 3x_1 \quad \quad x_3 + 3x_4 = 12. \end{cases}$$

4.

$$\begin{cases} 3x_1 - x_2 + x_3 - 2x_4 = 5, \\ x_1 + 2x_2 - x_3 + x_4 = 7, \\ -x_1 + x_2 - 4x_3 + x_4 = 2, \\ 3x_1 + 3x_2 \quad \quad -4x_4 = 10. \end{cases}$$

5.

6.

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

7.

$$\begin{cases} x_1 - 2x_2 + x_3 - x_4 = 3, \\ 2x_1 + x_2 - x_3 + x_4 = 2, \\ x_1 - x_2 + x_3 + 2x_4 = 3, \\ 3x_1 - x_2 \quad \quad \quad x = 6. \end{cases}$$

9.

$$\begin{cases} 3x_1 + x_2 - x_3 + x_4 = 9, \\ x_1 - x_2 + 2x_3 - x_4 = -2, \\ 2x_1 + 2x_2 - x_3 + x_4 = 8. \end{cases}$$

$$\begin{cases} 3x_1 - 4x_2 + x_3 - x_4 + x_5 = 0, \\ 2x_1 + 3x_2 - x_3 + x_4 + 2x_5 = 3, \\ -x_1 + x_2 + 2x_3 - 2x_4 - x_5 = -3, \\ 2x_1 - 4x_2 \quad x_3 - x_4 + 2x_5 = 1. \end{cases}$$

8.

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

10.

$$\begin{cases} x_1 + x_2 - x_3 - x_4 = 5, \\ 2x_1 - x_2 + 2x_3 + x_4 = 4, \\ 3x_1 \quad \quad + x_3 - 2x_4 = 6. \end{cases}$$

1.2. Знайти три базисних розв'язки наступних СЛАР:

1.

$$\begin{cases} x_1 + x_3 - x_5 = -4, \\ x_2 - x_3 + 3x_5 = 5, \\ x_1 \quad 2x_3 + x_4 + x_5 = 6. \end{cases}$$

2.

$$\begin{cases} x_1 + x_3 - x_5 = 1, \\ x_2 - x_3 + x_5 = 4, \\ \quad \quad 3x_3 + x_4 + 2x_5 = 6. \end{cases}$$

3.

$$\begin{cases} x_1 + x_2 \quad \quad \quad = -3, \\ \quad -x_2 + x_3 + \quad \quad \quad = 2, \\ \quad \quad x_2 \quad \quad + 3x_4 + x_5 = 6. \end{cases}$$

4.

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

5.

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

6.

$$\begin{cases} 2x_1 - x_2 + x_3 \quad \quad \quad + x_6 = 6, \\ -x_1 + 3x_2 - 2x_3 + x_4 \quad \quad = -3, \\ \quad x_1 + x_2 \quad \quad \quad + x_5 = 2. \end{cases}$$

7.

8.

$$\begin{cases} x_1 - x_4 + 3x_5 = -3; \\ x_2 + x_4 - x_5 = 5; \\ x_3 + 2x_4 + x_5 = 8. \end{cases}$$

9.

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

$$\begin{cases} x_1 - 2x_2 + x_3 = 5; \\ -x_1 + x_2 + x_4 = -7; \\ x_1 - x_2 + x_5 = 6. \end{cases}$$

10.

$$\begin{cases} x_1 - 3x_4 + x_5 = 8, \\ x_2 - x_4 + 2x_5 = 5, \\ x_3 + x_4 - x_5 = -2. \end{cases}$$

1.3. Знайти з використанням симплексних перетворень три опорних розв'язки наступних СЛАР:

1.

$$\begin{cases} 3x_1 + x_2 + x_4 = 4, \\ x_1 + x_3 - x_4 = 5, \\ x_1 + 2x_4 + x_5 = 2. \end{cases}$$

2.

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

3.

$$\begin{cases} 2x_1 - x_2 + x_4 = 4, \\ x_1 + x_2 + x_3 = 5, \\ 3x_1 + 2x_2 + x_5 = 9. \end{cases}$$

4.

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

5.

$$\begin{cases} x_1 + x_2 - x_4 = 5, \\ 3x_2 + x_3 + 2x_4 = 2, \\ -x_2 + 3x_4 + x_5 = 9. \end{cases}$$

6.

$$\begin{cases} x_1 + x_4 + x_5 = 3, \\ x_2 + 2x_4 + x_5 = 4, \\ x_3 - x_4 + 3x_5 = 6. \end{cases}$$

7.

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

8.

$$\begin{cases} 2x_1 + x_2 - x_4 = 2, \\ x_1 + x_3 + 2x_4 = 5, \\ -x_1 + 3x_4 + x_5 = 6. \end{cases}$$

9.

$$\begin{cases} x_1 + x_2 + 3x_5 = 6, \\ -2x_2 + x_3 + 2x_5 = 4, \\ 2x_2 + x_4 - x_5 = 2. \end{cases}$$

10.

$$\begin{cases} 3x_1 + x_2 + 2x_4 = 2, \\ x_1 + x_3 - x_4 = 5, \\ x_1 + x_4 + x_5 = 3. \end{cases}$$

ГЛАВА 2.

МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ЗАДАЧ ЛІНІЙНОГО ПРОГРАМУВАННЯ

Математичне моделювання – це опис процесу або явища, що розглядається, здійснений з використанням математичної символіки. В різних сферах діяльності людей виникають задачі оптимізації (максимізації або мінімізації) деяких лінійних функцій багатьох змінних на множинах, які визначаються системами лінійних рівнянь та нерівностей. Такі задачі називаються задачами лінійного програмування, а відповідний розділ теорії екстремальних задач – лінійним програмуванням (ЛП).

Вперше, в 1938 – 1939 рр., сформулював вказаний клас задач та запропонував методи їх розв’язування стосовно задач розкрою листів фанери, шкіри і т. ін. радянський математик Л. В. Канторович.

Пізніше лінійне програмування було перевідкрито Дж. Данцігом. У 1975 р. академіку Л. В. Канторовичу, авторитет якого визнано у всьому світі, сумісно з американським вченим Т. Купмансом присуджена Нобелівська премія за результати досліджень з оптимального використання ресурсів.

Розглянемо декілька прикладів задач лінійного програмування.

2.1 Приклади математичних моделей задач лінійного програмування (ЛП)

2.1.1 Задача вибору оптимального плану випуску

продукції підприємством

Розглядається робота деякого підприємства харчування або одного з його цехів, наприклад, кондитерського цеху. Виходимо з того, що цех має різні види сировини заданої кількості на деякий період планування (табл. 2.1). Цех може здійснювати випуск продукції, скажімо, трьох видів. Відомі витрати кожного виду сировини на випуск продукції №1, №2, №3. Крім того, відомий прибуток, який отримає цех від реалізації одиниці продукції кожного виду. Необхідно визначити план (програму) роботи цеху на плановий період (що виробляти і в якій кількості), що забезпечить найбільший (максимально можливий) прибуток цеху.

Сформульовану економічну задачу перекладемо на мову математики. З цією метою введемо позначення: x_1, x_2, x_3 – невідома кількість продукції відповідно №1, №2, №3. Іншими словами, план випуску продукції позначимо як вектор з невідомими координатами

$$\vec{X} = (x_1, x_2, x_3) \quad (2.1.1)$$

Таблиця 2.1

Сировина		Витрати сировини на одиницю продукції, кг		
Вид	Кількість, кг	№ 1	№ 2	№ 3
Борошно	1000	0,4	0,8	0,6
Цукор	500	0,2	0,5	0,4
Масло	300	0,15	0,4	0,3
Ізюм	100	0,05	–	0,1
Прибуток від продажу одиниці продукції, грн		0,8	0,4	0,5

Очевидно випуск продукції повинен здійснюватись з урахуванням обмежень по запасам сировини

$$\begin{aligned} 0,4x_1 + 0,8x_2 + 0,6x_3 &\leq 1000, \\ 0,2x_1 + 0,5x_2 + 0,4x_3 &\leq 500, \\ 0,15x_1 + 0,4x_2 + 0,3x_3 &\leq 300, \\ 0,05x_1 + 0,1x_3 &\leq 100. \end{aligned} \quad (2.1.2)$$

Зауважимо, що величини x_i ($i=1, 2, 3$) за своєю економічною суттю не можуть приймати від'ємні значення. Отже, повинні задовольняти таким умовам:

$$x_1 \geq 0, x_2 \geq 0, x_3 \geq 0. \quad (2.1.3)$$

При плані випуску продукції $\vec{X} = (x_1, x_2, x_3)$ цех отримує прибуток

$$Z = 0,8x_1 + 0,4x_2 + 0,5x_3, \quad (2.1.4)$$

де Z – цільова функція (критерій оптимальності).

Економіко-математична модель задачі. Необхідно знайти вектор (2.1.1), координати якого задовольняють обмеженням (2.1.2) і (2.1.3), і забезпечують максимум функції (2.1.4). Сформульована задача називається задачею лінійного програмування, оскільки нерівності (2.1.2), (2.1.3) є лінійними, цільова функція (2.1.4) – також лінійна. Термін «математичне програмування» походить від слів: програма (план) випуску продукції або роботи підприємства, яка визначається математичними засобами.

2.1.2 Задача про добовий раціон годування тварин

Для годування тварин можна використовувати різні види кормів, а саме: сіно, силос, концентрати тощо. Відомий вміст поживних речовин №1, №2, №3 і т. ін., зокрема білку, кальцію, фосфору і т. ін., в одиниці ваги кожного виду корму (табл. 2.2).

Добовий раціон годування однієї тварини повинен містити необхідну кількість поживних речовин (мінімальна добова потреба також вказана в таблиці 2.2). Крім того, відома ціна 1кг корму кожного виду.

Необхідно визначити найбільш дешевий раціон годування тварин, який би забезпечував добову потребу тварин в поживних речовинах.

Таблиця 2.2

Види кормів	Ціна за 1 кг, грн	Вміст поживних речовин у 1 кг корму, мг		
		№ 1	№ 2	№ 3
Сіно	0,5	9	5	7
Силос	0,7	4	12	3
Концентрати	1,0	2	8	15
Мінімальна добова потреба в поживних речовинах, мг		150	80	120

Позначимо x_1, x_2, x_3 – шукану кількість відповідно сіна, силосу, концентратів у добовому раціоні. Іншими словами, невідомий раціон позначимо як вектор

$$\vec{X} = (x_1, x_2, x_3). \quad (2.1.5)$$

Вимога про те, що добовий раціон повинен містити поживних речовин відповідно не менше ніж 150, 80, 120 мг, математично виражається такими нерівностями:

$$\begin{aligned} 9x_1 + 4x_2 + 2x_3 &\geq 150, \\ 5x_1 + 12x_2 + 8x_3 &\geq 80, \\ 7x_1 + 3x_2 + 15x_3 &\geq 120. \end{aligned} \quad (2.1.6)$$

Крім того, невідомі величини x_1, x_2, x_3 повинні задовольняти очевидній умові

$$x_1 \geq 0, x_2 \geq 0, x_3 \geq 0. \quad (2.1.7)$$

Вартість раціону становить

$$Z = 0,5x_1 + 0,7x_2 + 1,0x_3 \quad (2.1.8)$$

Економіко-математична модель задачі. Треба знайти вектор (2.1.5), координати якого задовольняють обмеженням (2.1.6) і (2.1.7), і забезпечують мінімум функції (2.1.8).

Розглянута задача є однією з найпростіших задач про суміші. До задач цього класу відноситься, наприклад, така задача. На підприємстві є нафтопродукти різного складу та різної вартості. Одержати бензин з певними

характеристиками: заданим октановим числом, ступенем очищення і т. ін., можна, змішуючи різні нафтопродукти.

Необхідно вибрати найбільш дешевий склад суміші, який дозволить одержувати бензин із заданими характеристиками.

2.1.3 Транспортна задача за критерієм вартості

На деяких пунктах відправлення (заводах, базах, складах і т. ін) знаходиться відома кількість однорідного вантажу. Вантаж треба доставити в деякі пункти призначення (заводи, будівельні майданчики, магазини і т. ін.) у заданій кількості (табл. 2.3).

Таблиця 2.3

Пункти відправлення \ Пункти призначення	170	180	150	100	\sum_j 600
100	x_{11} 2	x_{12} 6	x_{13} 2	x_{14} 5	
300	x_{21} 5	x_{22} 3	x_{23} 4	x_{24} 6	
200	x_{31} 4	x_{32} 7	x_{33} 5	x_{34} 3	
\sum_i 600					

Причому сумарна кількість вантажу, який відправляється, дорівнює кількості вантажу, який необхідно доставити в пункти призначення. Відомі тарифи c_{ij} доставки одиниці вантажу із i -го пункту відправлення до j -го пункту призначення ($i = 1, 2, 3; j = 1, 2, 3, 4$). Вони вказані у верхніх кутах клітинок. Необхідно скласти план перевезень так, щоб весь вантаж було вивезено з пунктів відправлення та доставлено до пунктів призначення в потрібній кількості. При цьому сумарні витрати повинні бути мінімальними. Для побудови математичної моделі задачі позначимо x_{ij} – шукану кількість вантажу, який доцільно перевезти із i -го пункту відправлення до j -го пункту призначення. Невідомі величини повинні задовольняти умовам

$$x_{11} + x_{12} + x_{13} + x_{14} = \sum_{j=1}^4 x_{1j} = 100,$$

$$x_{21} + x_{22} + x_{23} + x_{24} = \sum_{j=1}^4 x_{2j} = 300, \quad (2.1.9)$$

$$x_{31} + x_{32} + x_{33} + x_{34} = \sum_{j=1}^4 x_{3j} = 200;$$

$$x_{11} + x_{21} + x_{31} = \sum_{i=1}^3 x_{i1} = 170,$$

$$x_{12} + x_{22} + x_{32} = \sum_{i=1}^3 x_{i2} = 180, \quad (2.1.10)$$

$$x_{13} + x_{23} + x_{33} = \sum_{i=1}^3 x_{i3} = 150,$$

$$x_{14} + x_{24} + x_{34} = \sum_{i=1}^3 x_{i4} = 100.$$

Зміст умов (2.1.9) в тому, що весь вантаж вивезено з пунктів відправлення, а умов (2.1.10), – що вантаж завезено в пункти призначення у потрібній кількості. Крім того, величини x_{ij} повинні задовольняти очевидним умовам

$$x_{ij} \geq 0, \text{ для } i = 1, 2, 3; j = 1, 2, 3, 4. \quad (2.1.11)$$

Сумарна вартість доставки вантажу становить

$$Z = \sum_{i=1}^3 \sum_{j=1}^4 c_{ij} x_{ij}. \quad (2.1.12)$$

Економіко-математична модель задачі. Необхідно знайти вектор

$$\vec{X} = (x_{11}, x_{12}, x_{13}, \dots, x_{34}), \quad (2.1.13)$$

координати якого задовольняють системам рівнянь (2.1.9), (2.1.10), нерівностям (2.1.11) і забезпечують мінімум цільової функції (2.1.12).

Зауваження. Сформульована задача називається транспортною задачею (ТЗ) за критерієм вартості доставки (перевезень). Поряд із задачами такого типу існує велика кількість транспортних задач інших видів. В їх числі так звані ТЗ за критерієм часу, які формулюються таким чином: необхідно скласти план перевезень, який характеризується найменшими сумарними витратами часу на доставку вантажу. Задачі такого виду виникають при організації перевезень термінових вантажів або таких, що швидко псується.

$$Z = \sum_{i=1}^n c_i x_i \quad (2.2.15)$$

найбільше значення.

Слід зауважити, по-перше, що крім наведених, існують і деякі інші форми запису задачі ЛП; по-друге, за допомогою простих математичних перетворень будь-яку із форм задачі ЛП можна привести до будь-якої іншої форми, яка еквівалентна початковій.

Перехід здійснюється за допомогою таких перетворень.

1. Якщо деяка невідома не підлягає умові невід'ємності ($x_i \geq 0$), то її можна зобразити у вигляді різниці двох нових невідомих, які обов'язково підлягають цій умові

$$x_i = x'_i - x''_i, \quad (2.2.16)$$

$$x'_i \geq 0, \quad x''_i \geq 0. \quad (2.2.17)$$

2. Задача мінімізації цільової функції Z може бути змінена на задачу максимізації нової цільової функції Z_1

$$Z = f(x_1, x_2, \dots, x_n) \rightarrow \min,$$

$$Z_1 = -f(x_1, x_2, \dots, x_n) \rightarrow \max.$$

Потім розв'язується задача ЛП із функцією Z_1 . Щоб одержати оптимум початкової задачі ЛП, необхідно взяти оптимум перетвореної задачі з протилежним знаком.

3. Для здійснення переходу від обмежень – нерівностей до обмежень – рівнянь розглянемо наступні міркування. Припустимо, що знайдено деякий допустимий розв'язок

$$\overline{X}^\partial = (x_1^\partial, x_2^\partial, \dots, x_n^\partial) \quad (2.2.18)$$

задачі ЛП з обмеженнями – нерівностями. Тоді після підстановки вказаного розв'язку кожне з обмежень – нерівностей перетворюється або в числову рівність виду

$$a_{i,1}x_1^\partial + a_{i,2}x_2^\partial + \dots + a_{i,n}x_n^\partial = a_{i0}, \quad (2.2.19)$$

або в нерівність

$$a_{i,1}x_1^\partial + a_{i,2}x_2^\partial + \dots + a_{i,n}x_n^\partial < a_{i0}. \quad (2.2.20)$$

Неважно бачити, що для кожного допустимого розв'язку і кожного обмеження – нерівності існує таке невід'ємне число, наприклад x_{n+1} , додавання якого до лівої частини початкової нерівності, легко перетворює останню у тотожність.

Іншими словами, якщо вектор (2.2.18) є розв'язком задачі ЛП, який задовольняє умові

Приклад 1. Привести до канонічної форми наступну задачу лінійного програмування*

$$\begin{cases} x_1 + x_2 - x_3 + x_4 \leq 5, \\ 3x_1 - x_2 + 4x_3 - 2x_4 \geq 1, \\ x_1 - x_2 + 5x_3 = 3, \\ x_1 \geq 0, x_3 \geq 0, x_4 \geq 0, \end{cases} \quad (2.2.30)$$

$$Z = 3x_1 + 4x_2 - x_3 - x_4 \rightarrow \min.$$

Розв'язання. Невідому x_2 , на яку не накладена умова невід'ємності, запишемо у виді

$$x_2 = x_5 - x_6, \text{ де } x_5 \geq 0, x_6 \geq 0. \quad (2.2.31)$$

Потім (2.2.31) підставимо у (2.2.30).

Задача (2.2.30) містить дві нерівності, які необхідно замінити рівностями. З цією метою введемо невід'ємні додаткові невідомі $x_7 \geq 0, x_8 \geq 0$.

Задачу мінімізації функції Z замінимо на задачу максимізації функції $Z_1 = -Z$. В результаті одержимо задачу ЛП у канонічній формі

$$\begin{cases} x_1 + (x_5 - x_6) - x_3 + x_4 + x_7 = 5, \\ 3x_1 - (x_5 - x_6) + 4x_3 - 2x_4 - x_8 = 1, \\ x_1 - (x_5 - x_6) + 5x_3 = 3, \end{cases} \quad (2.2.32)$$

$$x_i \geq 0, i = 1, 2, \dots, 8,$$

$$Z_1 = -Z = -3x_1 - 4(x_5 - x_6) + x_3 + x_4 \rightarrow \max.$$

Приклад 2. Привести до форми II наступну задачу ЛП:

$$\begin{cases} x_1 + x_4 - 3x_5 = 3, \\ x_2 - x_4 + x_5 = 1, \\ x_3 + 2x_4 - x_5 = 2, \end{cases} \quad (2.2.33)$$

$$x_i \geq 0, i = 1, 2, \dots, 5.$$

$$Z = x_1 + 2x_2 + x_3 - 5x_4 + 6x_5 \rightarrow \max.$$

Розв'язання. Тут система рівнянь приведена до одиничного базису. Виразимо базисні невідомі x_1, x_2, x_3 через вільні невідомі

$$\begin{cases} x_1 = 3 - x_4 + 3x_5, \\ x_2 = 1 + x_4 - x_5, \\ x_3 = 2 - 2x_4 + x_5. \end{cases} \quad (2.2.34)$$

Підставимо (2.2.34) у вираз для цільової функції

$$Z = (3 - x_4 + 3x_5) + 2(1 + x_4 - x_5) + (2 - 2x_4 + x_5) - 5x_4 + 6x_5 =$$

* Задача записана у скороченому вигляді. У подальшому будемо користуватись аналогічними скороченнями при записі задач ЛП в різних формах.

$$= -6x_4 + 8x_5 + 7 \rightarrow \max.$$

Опустимо невід'ємні базисні невідомі, одержимо наступну форму запису задачі ЛП:

$$\begin{cases} x_4 - 3x_5 \leq 3, \\ -x_4 + x_5 \leq 1, \\ 2x_4 - x_5 \leq 2, \\ x_4 > 0, x_5 \geq 0, \\ Z = -6x_4 + 8x_5 + 7 \rightarrow \max. \end{cases} \quad (2.2.35)$$

Отже, завжди можна здійснити еквівалентний перехід від однієї форми запису задачі ЛП до будь-якої іншої. У подальшому, виходячи із міркувань зручності, задачу ЛП будемо розглядати в тій або іншій формі.

2.3 Геометричне тлумачення задачі ЛП. Графічний метод

Задача лінійного програмування має геометричне тлумачення, причому воно найбільш наочне у випадку, коли задача містить не більше ніж дві невідомі у стандартній формі або не більше ніж дві вільні невідомі у канонічній формі (після приведення до одиничного базису). У цьому випадку розв'язок задачі ЛП можна одержати так званим графічним методом, який дозволяє сформулювати основні властивості задач ЛП, а також пояснити геометричний зміст деяких методів їх розв'язування і в загальному випадку.

Розглянемо задачу ЛП у стандартній формі

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 &\leq a_{10}, \\ a_{21}x_1 + a_{22}x_2 &\leq a_{20}, \\ &\dots\dots\dots \end{aligned} \quad (2.3.1)$$

$$\begin{aligned} a_{m1}x_1 + a_{m2}x_2 &\leq a_{m0}, \\ x_1 \geq 0, x_2 &\geq 0, \end{aligned} \quad (2.3.2)$$

$$Z = c_1x_1 + c_2x_2 \rightarrow \max. \quad (2.3.3)$$

Процедуру розв'язування задачі ЛП графічним методом умовно розіб'ємо на три етапи.

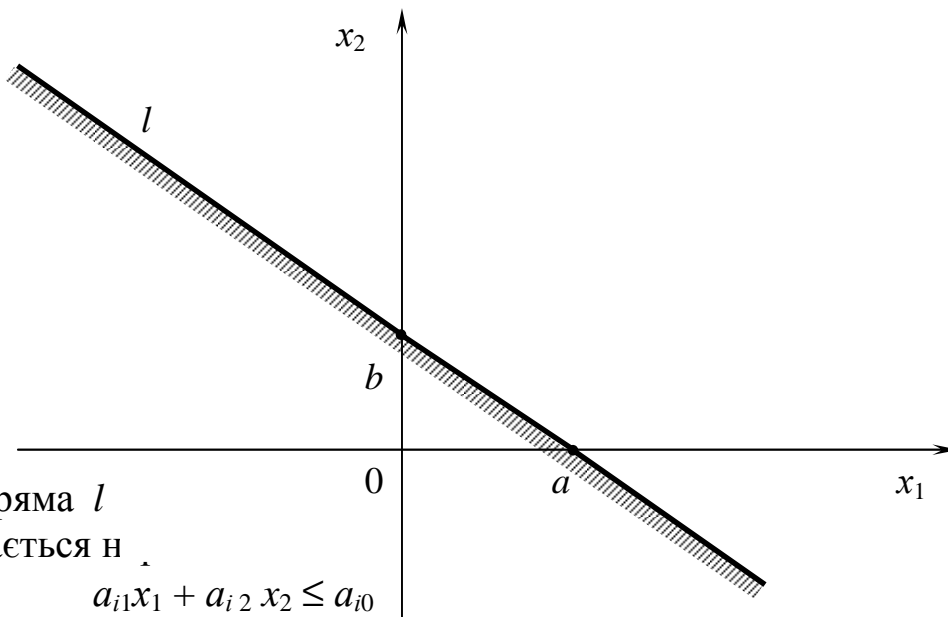
1. Перший етап: знаходження області допустимих розв'язків задачі (ОДР), тобто множини точок на площині, координати яких задовольняють всім нерівностям (2.3.1) та (2.3.2).

Для цього виберемо на площині декартову систему координат з осями Ox_1 та Ox_2 . Нагадаємо, що рівнянню

$$a_{i1}x_1 + a_{i2}x_2 = a_{i0} \quad (2.3.4)$$

відповідає пряма лінія на координатній площині x_1Ox_2 . Його зручно записувати у виді рівняння у відрізках на осях

$$\frac{x_1}{\frac{a_{i0}}{a_{i1}}} + \frac{x_2}{\frac{a_{i0}}{a_{i2}}} = 1, \left(\frac{x_1}{a} + \frac{x_2}{b} = 1 \right)$$



Пряма l
визначається н

$$a_{i1}x_1 + a_{i2}x_2 \leq a_{i0}$$

дна із яких

$$(2.3.5)$$

ї називаються допустимою. Допустима півплощина виділяється штриховкою, а її визначення здійснюється підстановкою координат деякої контрольної точки в нерівність. Якщо координати контрольної точки задовольняють нерівності, то півплощина з контрольною точкою є допустимою. Якщо ж координати контрольної точки не задовольняють нерівності, то інша півплощина є допустимою.

Зауваження. По-перше, контрольна точка не повинна лежати на прямій l . По-друге, дуже часто за контрольну точку зручно вибирати початок координат $O(0, 0)$.

Приклад. Знайти півплощину яка визначається нерівністю

$$x_1 - 2x_2 \leq 4. \quad (2.3.6)$$

Розв'язання. Побудуємо пряму (рис. 2.2)

$$x_1 - 2x_2 = 4. \quad (2.3.7)$$

За контрольну доцільно взяти точку $O(0, 0)$. Після підстановки координат вибраної точки у (2.3.6) маємо $0 < 4$, тобто півплощина, в якій знаходиться початок координат є допустимою (на рис. 2.2 вона заштрихована).

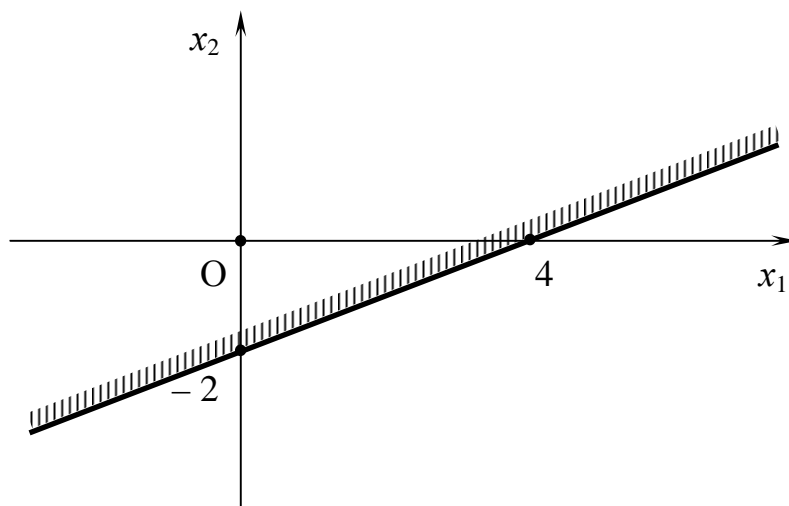


Рис. 2.2.

Отже, кожна з нерівностей розглядуваної задачі визначає деяку допустиму півплощину. Повторюємо: множина точок площини x_1Ox_2 , координати яких задовольняють всім нерівностям (2.3.1) та (2.3.2), утворює область допустимих розв'язків (ОДР). Іншими словами, ОДР – це частина площини x_1Ox_2 , яка належить всім допустимим півплощинам, тобто їх спільна частина («перетин» всіх півплощин).

Приклад 1. Знайти область допустимих розв'язків наступної задачі лінійного програмування:

$$\begin{aligned} -x_1 + 3x_2 &\leq 33, \\ 3x_1 + 8x_2 &\leq 139, \\ 11x_1 - 5x_2 &\leq 132, \end{aligned} \tag{2.3.8}$$

$$\begin{aligned} x_1 + x_2 &\geq 5, \\ x_1 \geq 0, \quad x_2 &\geq 0, \end{aligned} \tag{2.3.9}$$

$$Z = 2x_1 + x_2 \rightarrow \max. \tag{2.3.10}$$

ОДР цієї задачі зображена на рис. 2.3. Вона є опуклим багатокутником $ABEDNM$, сторони якого AB , BE , ED , DN , NM та MA визначаються відповідно рівняннями

$$\begin{aligned} -x_1 + 3x_2 = 33, \quad 3x_1 + 8x_2 = 139, \quad 11x_1 - 5x_2 = 132, \\ x_2 = 0, \quad x_1 + x_2 = 5, \quad x_1 = 0. \end{aligned} \tag{2.3.11}$$

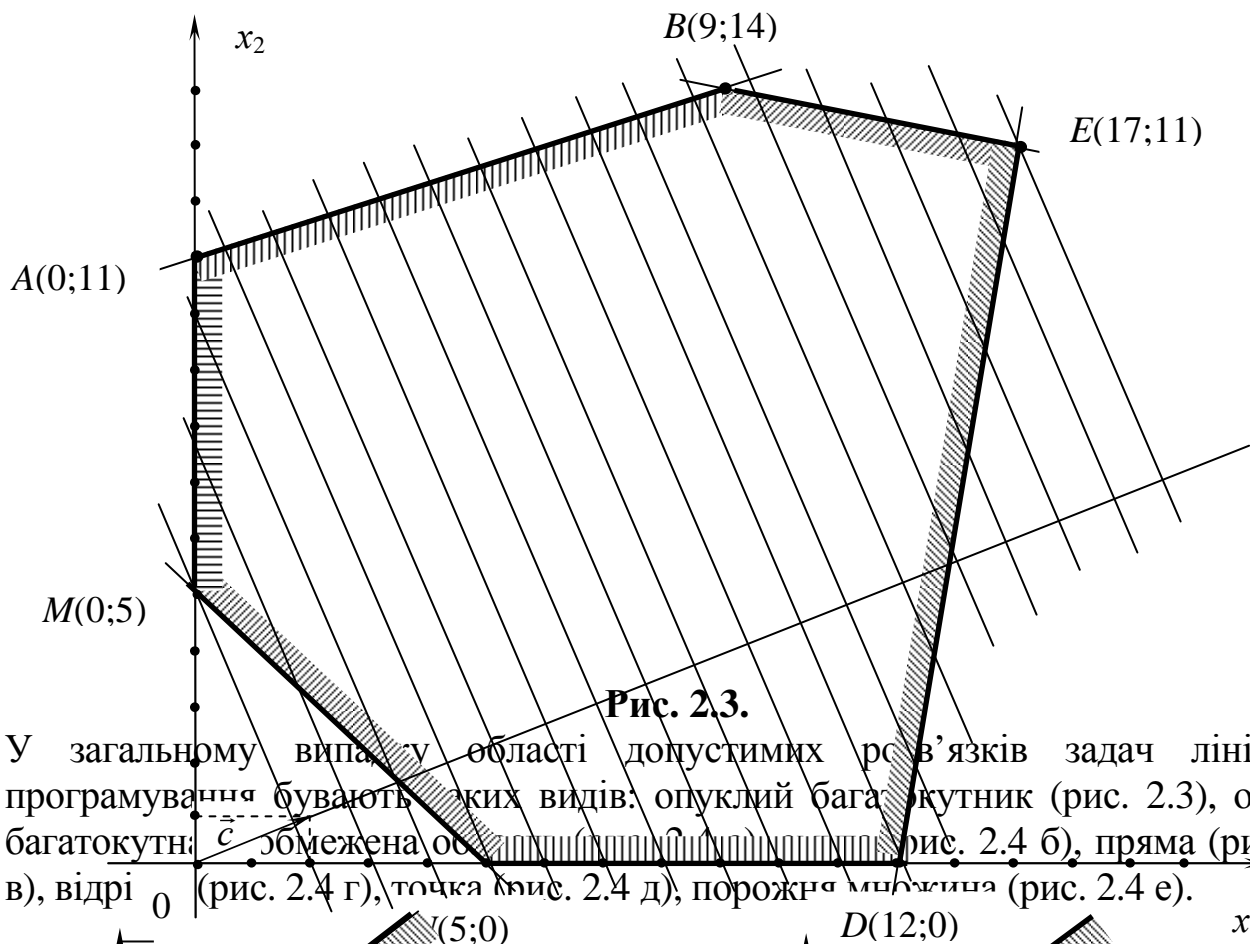
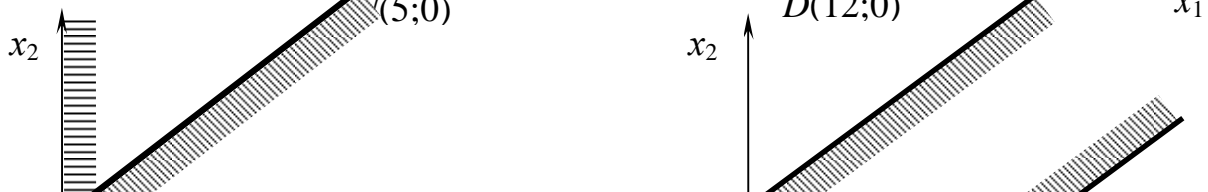


Рис. 2.3.

У загальному випадку області допустимих розв'язків задач лінійного програмування бувають таких видів: опуклий багатокутник (рис. 2.3), опукла багатокутна область (рис. 2.4 а), обмежена область (рис. 2.4 б), пряма (рис. 2.4 в), відрізок (рис. 2.4 г), точка (рис. 2.4 д), порожня множина (рис. 2.4 е).



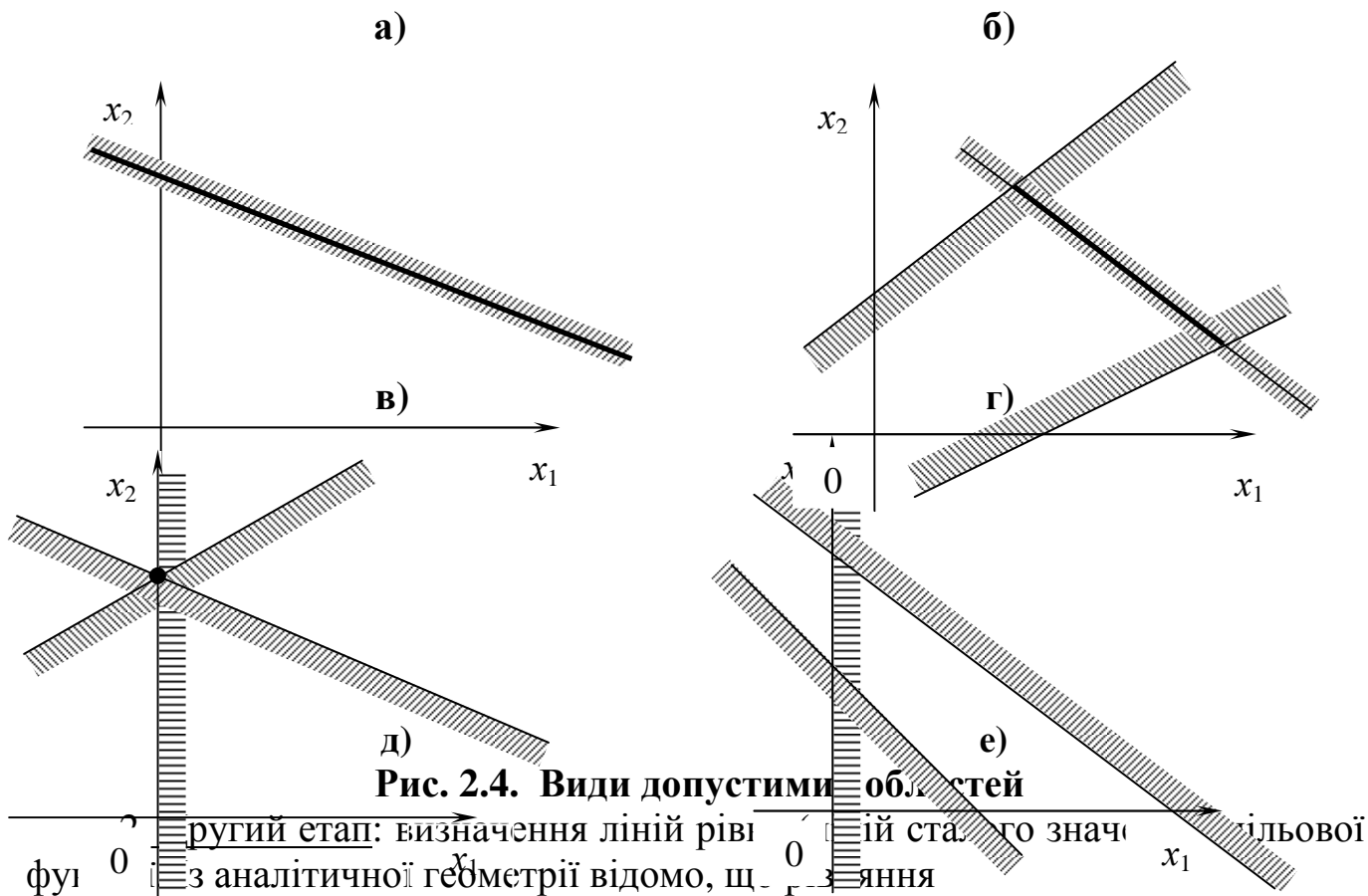


Рис. 2.4. Види допустимих областей

Другий етап: визначення ліній рівня цільової функції. З аналітичної геометрії відомо, що рівняння

$$c_1x_1 + c_2x_2 = a, \tag{2.3.12}$$

де a – деяке дійсне число, є рівнянням прямої лінії, перпендикулярної вектору $\vec{c} = (c_1, c_2)$. Функція $Z = c_1x_1 + c_2x_2$ має одне і теж значення $Z = a$ у всіх точках прямої лінії (2.3.12). Тоді, змінюючи параметри a , одержимо сімейство паралельних прямих, названих лініями рівня цільової функції $Z = c_1x_1 + c_2x_2$ (рис. 2.3).

Легко бачити, що збільшення параметра a (отже і значення цільової функції) рівнозначно зміщенню лінії рівняння у напрямку вектора її нормалі $\vec{c} = (c_1, c_2)$, зменшення цього параметра означає зміщення лінії рівняння у напрямку протилежному вектору \vec{c} .

3. Третій (завершальний) етап: визначення точки (інколи точок) на площині x_1Ox_2 , яка належить ОДР та лежить на лінії максимально або мінімально можливого рівня відповідно у випадку задачі максимізації або мінімізації.

Очевидно, цільова функція досягне найбільшого значення тоді, коли лінія рівня, яка зміщується у напрямку вектора \vec{c} , буде проходити через крайню, найбільш віддалену від початку координат, точку (точки) області

допустимих розв'язків (при подальшому переміщенні лінія рівня вже не буде перетинатися з ОДР).

У сформульованій задачі лінійного програмування (2.3.8) – (2.3.10) цільова функція досягне найбільшого значення у точці E (рис. 2.3). Координати точки E – точки перетину прямих, які визначаються рівняннями

$$3x_1 + 8x_2 = 139,$$

$$11x_1 - 5x_2 = 132,$$

дорівнюють $x_1^0 = 17$, $x_2^0 = 11$. Отже, єдиний оптимальний розв'язок задачі (2.3.8) – (2.3.10) має вид $x_1^0 = 17$, $x_2^0 = 11$, або $\bar{X}_{\text{опт}} = (17; 11)$. Максимальне значення цільової функції дорівнює

$$Z_{\text{max}} = 2 \cdot 17 + 11 = 45.$$

Приклад 2. Знайти найбільше значення функції

$$Z = 4x_1 + 9x_2 + x_3 + x_4, \quad (2.3.13)$$

за умов

$$\begin{aligned} -2x_1 + x_2 + x_3 &= 4, \\ 2x_1 + 7x_2 + x_4 &= 92, \\ 2x_1 - 3x_2 + x_5 &= 12, \\ x_i &\geq 0, \quad i = 1, 2, \dots, 5. \end{aligned} \quad (2.3.14)$$

Розв'язання. Виразимо базисні невідомі x_3, x_4, x_5 через вільні x_1 та x_2

$$\begin{aligned} x_3 &= 4 + 2x_1 - x_2, \\ x_4 &= 92 - 2x_1 - 7x_2, \\ x_5 &= 12 - 2x_1 + 3x_2. \end{aligned} \quad (2.3.15)$$

Підставимо (2.3.15) у вираз для цільової функції (2.3.13). В результаті одержимо

$$Z = 4x_1 + 9x_2 + (4 + 2x_1 - x_2) + (92 - 2x_1 - 7x_2) = 4x_1 + x_2 + 96. \quad (2.3.16)$$

Опустимо невід'ємні невідомі x_3, x_4, x_5 у (2.3.14), а також враховуючи (2.3.16), перейдемо до моделі задачі ЛП з обмеженнями-нерівностями

$$\begin{aligned} -2x_1 + x_2 &\leq 4, \\ 2x_1 + 7x_2 &\leq 92, \\ 2x_1 - 3x_2 &\leq 12, \\ x_1 &\geq 0, \quad x_2 \geq 0, \end{aligned} \quad (2.3.17)$$

$$Z = 4x_1 + x_2 + 96 \rightarrow \max.$$

В результаті одержимо задачу лінійного програмування, в якій обмеження містять дві невідомих x_1 та x_2 . Будуємо ОДР, вектор цільової функції (2.3.16) та лінії рівняння (рис. 2.5).



0

 x_1 **Рис. 2.5.**

Оптимальний розв'язок цієї задачі визначають координати точки M – точки перетину прямих

$$\begin{aligned} 2x_1 + 7x_2 &= 92, \\ 2x_1 - 3x_2 &= 12. \end{aligned} \tag{2.3.18}$$

Розв'язавши систему рівнянь (2.3.18), одержимо $x_1 = 18$, $x_2 = 8$. Підставимо одержані значення вільних невідомих x_1 та x_2 у співвідношення (2.3.15) та у вираз (2.3.16) для цільової функції Z , знайдемо значення базисних невідомих

$$x_3 = 4 + 2 \cdot 18 - 8 = 32,$$

$$x_4 = 92 - 2 \cdot 18 - 7 \cdot 8 = 0,$$

$$x_5 = 12 - 2 \cdot 18 + 3 \cdot 8 = 0,$$

а також максимальне значення цільової функції

$$Z_{\max} = 4 \cdot 18 + 8 + 96 = 176,$$

або

$$Z_{\max} = 4 \cdot 18 + 9 \cdot 8 + 32 + 0 = 176.$$

Отже, оптимальний розв'язок розглянутої задачі має вид

$$\bar{X}_{\text{опт}} = (18; 8; 32; 0; 0).$$

Максимальне значення цільової функції становить

$$Z_{\max} = 176.$$

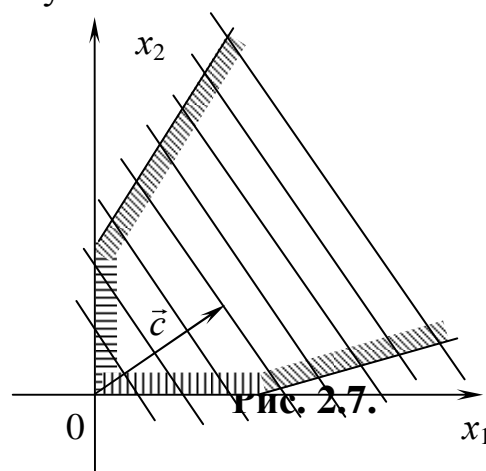
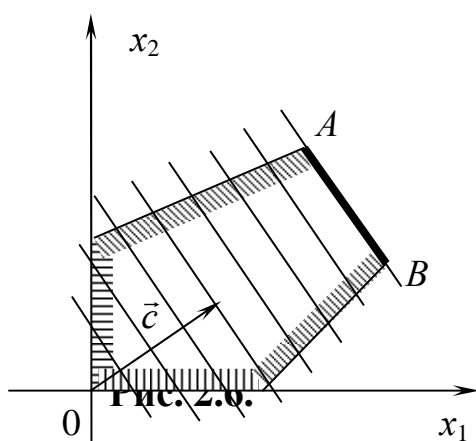
Таким чином, якщо задача лінійного програмування в канонічній формі містить дві вільні невідомі за будь-якого числа базисних (число лінійно незалежних обмежень – рівнянь на два менше за загальне число невідомих), то її можна розв'язати графічним методом.

Проаналізувавши розглянуті задачі ЛП, сформулюємо деякі загальні, геометрично очевидні твердження, які є властивостями розв'язків задач лінійного програмування.

1. Якщо існує оптимальний розв'язок задачі ЛП, то він обов'язково знаходиться на границі ОДР.

2. Задача лінійного програмування може мати не один оптимальний розв'язок, а нескінченну множину їх. Це випадок так званого альтернативного оптимуму. Він має місце тоді, коли лінії рівня паралельні тій стороні багатокутника допустимих розв'язків, на якій досягається максимальне (мінімальне) значення цільової функції (рис. 2.6).

3. Задача лінійного програмування може не мати оптимального розв'язку навіть у випадку, коли ОДР не є порожньою множиною. Це буває тоді, коли цільова функція не обмежена в ОДР зверху у випадку задачі на максимум (рис. 2.7) та відповідно знизу в задачі на мінімум.



4. Якщо оптимальний розв'язок задачі ЛП існує, то він досягається принаймні у одній з вершин багатокутника допустимих розв'язків (у випадку альтернативного оптимуму він досягається в двох вершинах ОДР).

5. Якщо в задачі лінійного програмування, заданій в канонічній формі, число вільних невідомих дорівнює двом, і оптимальний розв'язок задачі існує, то він досягається в точці (вершині), в якій принаймні дві невідомі дорівнюють нулю, а решта – невід'ємні. Отже, вершинам багатокутника допустимих розв'язків відповідають опорні розв'язки системи рівнянь задачі ЛП. Зокрема, у прикладі 2, в точці M шукані величини відповідно дорівнюють: $x_4 = 0$, $x_5 = 0$, $x_1 = 18$, $x_2 = 8$, $x_3 = 32$, тобто оптимальний розв'язок є опорним. Коротко: якщо оптимальний розв'язок задачі ЛП, заданій в канонічній формі, існує, то він збігається принаймні з одним із опорних розв'язків системи лінійних рівнянь.

6. Оптимальний розв'язок задачі ЛП можна знайти простим перебором скінченного числа опорних розв'язків (вершин ОДР), виділивши із них той, в якому цільова функція досягає максимуму (мінімуму).

Зауваження 1. Графічним методом можна, хоч і дещо обтяжливо, знайти розв'язок деяких задач ЛП тривимірного простору (в системі

обмежень – рівностей три вільних невідомих, наприклад, x_1, x_2, x_3 , решта невідомих – базисні. В цих випадках ОДР – це опуклий багатогранник, зокрема необмежений, розв’язків, тобто частина тривимірного простору, для якої виконуються умови $x_i \geq 0, (i = 1, 2, \dots, n)$. Значення цільової функції $Z = c_1x_1 + c_2x_2 + c_3x_3$ зростають у напрямку вектора $\vec{c} = (c_1, c_2, c_3)$, перпендикулярного до площин рівня. Оптимальний розв’язок, якщо він існує, збігається з однією (або кількома) вершинами багатогранника.

Задачу лінійного програмування чотирьохвимірного (та більше) простору зобразити графічно неможливого. Разом з тим буває зручним використання відповідної геометричної термінології.

Зауваження 2. Можна довести, що сформульовані твердження стосовно оптимальних розв’язків задач лінійного програмування мають місце і в загальному випадку n -вимірного простору ($n > 3$). Вони становляться інтуїтивно зрозумілими, якщо врахувати, що поняття та властивості n -вимірного векторного простору є результатом формалізації та узагальнення цих же понять та властивостей тривимірного простору.

Питання для самоконтролю

1. Наведіть приклади задач лінійного програмування у різних сферах людської діяльності.
2. Назвіть різні форми запису задач ЛП.
3. В чому полягає загальна, стандартна, канонічна форми запису задач лінійного програмування?
4. Назвіть етапи переходу:
 - а) до канонічної форми запису задачі ЛП;
 - б) від канонічної до стандартної форми запису задачі ЛП.
5. В чому полягає суть графічного методу розв'язування задач ЛП?
6. Назвіть етапи розв'язування задачі ЛП графічним методом.
7. Що таке ОДР задачі ЛП і як вона знаходиться?
8. Які види ОДР ви знаєте? Зобразіть їх на рисунку.
9. Що таке лінії рівня цільової функції та як вони знаходяться?
10. Сформулюйте загальні твердження стосовно властивостей розв'язків задач ЛП.
11. Що таке альтернативний оптимум задачі ЛП?

Задачі до глави 2

2.1. Привести до канонічної форми запису наступні задачі лінійного програмування:

$$1) \quad x_1 + 3x_2 - x_3 + 2x_4 \leq 10,$$

$$-x_1 + x_2 + 2x_3 - x_4 \geq 5,$$

$$3x_1 - x_2 + 5x_4 = 6,$$

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

$$Z = 5x_1 - x_2 + 6x_3 - x_4 \rightarrow \min.$$

$$3) \quad x_1 + 3x_2 - x_3 + x_5 \leq 9,$$

$$-x_1 - x_2 + x_3 - 2x_4 - x_5 \geq 2,$$

$$2x_1 - x_2 - 2x_3 + x_4 \geq 4,$$

$$x_i \geq 0, \quad i=3,4,5,$$

$$Z = 4x_1 - x_2 + x_3 - 5x_4 \rightarrow \min.$$

$$2) \quad 2x_1 - x_2 + x_3 - x_4 = 7,$$

$$-x_1 + x_2 - 2x_3 \geq 3,$$

$$x_1 - x_2 + 4x_3 - x_4 \geq 4,$$

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

$$Z = 2x_1 + x_2 - x_4 \rightarrow \max$$

$$4) \quad 3x_1 - x_2 + 2x_3 - x_4 \geq 4,$$

$$-x_1 + x_2 - x_3 + 2x_4 \leq 2,$$

$$4x_1 - x_2 + x_3 - x_4 = 7,$$

$$x_1 \geq 0, \quad x_2 \geq 0,$$

$$Z = 3x_1 + x_2 - x_3 + 6x_4 \rightarrow \min.$$

2.2. Привести до стандартної форми з обмеженнями – нерівностями наступні задачі лінійного програмування:

$$\begin{array}{l}
 1) \quad x_1 - x_4 + x_5 = 2, \\
 \quad x_2 + x_4 - 2x_5 = 4, \\
 \quad x_3 + 2x_4 + x_5 = 5, \\
 \quad x_i \geq 0, \quad i=1, 2, \dots, 5, \\
 Z = x_1 + x_2 - x_3 + 2x_4 - x_5 \rightarrow \max.
 \end{array}
 \quad
 \begin{array}{l}
 2) \quad x_1 + x_2 - x_3 + 2x_5 - 2x_6 = 9, \\
 \quad -x_1 + x_3 + x_4 - x_5 - x_6 = 4, \\
 \quad 2x_1 - x_3 + x_5 + x_6 = 3, \\
 \quad x_i \geq 0, \quad i=1, 2, \dots, 6, \\
 Z = 5x_1 - x_2 + 4x_3 + x_4 - x_6 \rightarrow \max.
 \end{array}$$

$$\begin{array}{l}
 3) \quad 2x_1 + x_2 - x_3 + 4x_4 = 1, \\
 \quad x_1 + 3x_3 - x_4 = 7, \\
 \quad -x_1 + x_3 - x_4 + x_5 = 6, \\
 \quad x_i \geq 0, \quad i=1, 2, \dots, 5, \\
 Z = x_1 + x_2 - 4x_3 + 2x_5 \rightarrow \max.
 \end{array}
 \quad
 \begin{array}{l}
 4) \quad x_1 - x_2 + 4x_3 + 2x_4 - x_5 = 3, \\
 \quad x_2 - x_3 + x_5 = 5, \\
 \quad 2x_2 + x_3 + x_4 + x_5 = 2, \\
 \quad x_i \geq 0, \quad i=1, 2, \dots, 5, \\
 Z = x_1 + x_2 - 4x_3 + x_4 - 6x_5 \rightarrow \max.
 \end{array}$$

2.3. Розв'язати графічним методом задачі лінійного програмування:

$$\begin{array}{l}
 1) \quad 2x_1 + x_2 \geq 4, \\
 \quad x_1 + 5x_2 \leq 33, \\
 \quad 4x_1 + 2x_2 \leq 24, \\
 \quad x_1 \geq 0, \quad x_2 \geq 0, \\
 Z = x_1 + 2x_2 \rightarrow \max.
 \end{array}
 \quad
 \begin{array}{l}
 2) \quad -x_1 + 2x_2 \leq 18, \\
 \quad 3x_1 + x_2 \leq 30, \\
 \quad 2x_1 + x_2 \geq 4, \\
 \quad x_1 \geq 0, \quad x_2 \geq 0, \\
 Z = x_1 + 2x_2 \rightarrow \max.
 \end{array}
 \quad
 \begin{array}{l}
 3) \quad 2x_1 + x_2 \leq 14, \\
 \quad x_1 + 3x_2 \leq 22, \\
 \quad 2x_1 + x_2 \geq 6, \\
 \quad x_1 \geq 0, \quad x_2 \geq 0, \\
 Z = 2x_1 + 3x_2 \rightarrow \max.
 \end{array}$$

$$\begin{array}{l}
 4) \quad 5x_1 + x_2 \leq 33, \\
 \quad x_1 + x_2 \geq 2, \\
 \quad 2x_1 + 4x_2 \leq 24, \\
 \quad x_1 \geq 0, \quad x_2 \geq 0, \\
 Z = 2x_1 + x_2 \rightarrow \max.
 \end{array}
 \quad
 \begin{array}{l}
 5) \quad x_1 + 2x_2 \geq 4, \\
 \quad 3x_1 + x_2 \leq 22, \\
 \quad x_1 + 2x_2 \leq 14, \\
 \quad x_1 \geq 0, \quad x_2 \geq 0, \\
 Z = 3x_1 + 2x_2 \rightarrow \max.
 \end{array}
 \quad
 \begin{array}{l}
 6) \quad x_1 + x_2 \geq 2, \\
 \quad x_1 + 2x_2 \leq 16, \\
 \quad 3x_1 - 5x_2 \leq 4, \\
 \quad x_1 \geq 0, \quad x_2 \geq 0, \\
 Z = 3x_1 + x_2 \rightarrow \max.
 \end{array}$$

$$\begin{array}{l}
 7) \quad x_1 + 2x_2 \leq 20, \\
 \quad 2x_1 + x_2 \geq 4, \\
 \quad 3x_1 - 5x_2 \leq 16, \\
 \quad x_1 \geq 0, \quad x_2 \geq 0, \\
 Z = 3x_1 + x_2 \rightarrow \max.
 \end{array}
 \quad
 \begin{array}{l}
 8) \quad -x_1 + 3x_2 \leq 16, \\
 \quad x_1 + 2x_2 \geq 5, \\
 \quad 2x_1 + x_2 \leq 17, \\
 \quad x_1 \geq 0, \quad x_2 \geq 0, \\
 Z = 3x_1 + 4x_2 \rightarrow \max.
 \end{array}
 \quad
 \begin{array}{l}
 9) \quad x_1 + 2x_2 \geq 4, \\
 \quad 4x_1 - 2x_2 \leq 28, \\
 \quad x_1 + 3x_2 \geq 21, \\
 \quad x_1 \geq 0, \quad x_2 \geq 0, \\
 Z = 2x_1 + x_2 \rightarrow \min.
 \end{array}$$

$$\begin{array}{lll}
10) \quad x_1 + 2x_2 \geq 4, & 11) \quad x_1 + 3x_2 \geq 3, & 12) \quad -2x_1 + x_2 \leq 28, \\
2x_1 - x_2 \leq 20, & 3x_1 + 2x_2 \leq 28, & 2x_1 + x_2 \geq 5, \\
x_1 + 3x_2 \leq 30, & x_1 + 6x_2 \leq 52, & 3x_1 + x_2 \leq 21, \\
x_1 \geq 0, \quad x_2 \geq 0, & x_1 \geq 0, \quad x_2 \geq 0, & x_1 \geq 0, \quad x_2 \geq 0, \\
Z = 2x_1 + x_2 \rightarrow \min. & Z = x_1 + 2x_2 \rightarrow \max. & Z = x_1 + 2x_2 \rightarrow \min.
\end{array}$$

$$\begin{array}{ll}
13) \quad 4x_1 + x_2 \leq 45, & 14) \quad 2x_1 + x_2 \leq 20, \\
x_1 + x_2 \geq 3, & x_1 + 3x_2 \geq 6, \\
-x_1 + 2x_2 \leq 12, & 5x_1 - 3x_2 \leq 17, \\
x_1 \geq 0, \quad x_2 \geq 0, & x_1 \geq 0, \quad x_2 \geq 0, \\
Z = 2x_1 + 3x_2 \rightarrow \max. & Z = x_1 + 3x_2 \rightarrow \max.
\end{array}$$

ГЛАВА 3.

СИМПЛЕКСНИЙ МЕТОД РОЗВ'ЯЗУВАННЯ ЗАДАЧ ЛП

3.1 Симплексний метод розв'язування не вироджених задач ЛП

Вище було сказано, що графічний метод розв'язування задач лінійного програмування застосовується в тих випадках, коли система обмежень – нерівностей містить не більше за дві невідомі величини або ж система обмежень – рівностей – не більше ніж дві вільні невідомі; при трьох вільних невідомих геометрична інтерпретація розв'язання ускладнена, хоча і можлива. В інших випадках ($n > 3$), для розв'язування задач ЛП застосовуються не геометричні, а обчислювальні методи. Найбільш поширеним методом, за допомогою якого можна розв'язати будь-яку задачу лінійного програмування, є симплексний метод.

Суть симплексного методу полягає в послідовному переході від одного опорного розв'язку до іншого, кращого з точки зору цільової функції. В результаті скінченного числа таких переходів одержують оптимальний (найкращий) розв'язок задачі ЛП, або ж встановлюють її нерозв'язуваність. Іншими словами, метод полягає в послідовному покращенні розв'язку (плану). Звідси походить друга назва цього методу – метод послідовного покращення плану.

Розглянемо задачу ЛП у канонічній формі. При цьому припустимо, що система обмежень – рівностей уже приведена до одиничного базису, невідомі x_1, x_2, \dots, x_r – базисні, $x_{r+1}, x_{r+2}, \dots, x_n$ – вільні

Вираз (3.1.7), завдяки введенню в правій частині (3.1.3) доданку $-c_0$, має вид ідентичний виразам (3.1.6). У зв'язку з цим є можливість записати одну загальну формулу для визначення всіх коефіцієнтів у (3.1.5).

$$a_{0k} = \sum_{i=1}^r c_i a_{ik} - c_k, \quad k = r + 1, r + 2, \dots, n, 0. \quad (3.1.8)$$

Коефіцієнти a_{0k} при вільних невідомих у виразі (3.1.5) для цільової функції будемо називати оцінками. Зміст терміну пояснимо нижче.

Припустимо, що праві частини всіх рівнянь (3.1.1) додатні (випадок рівності нулю буде розглянуто окремо). Тоді, виходячи із (3.1.1) та (3.1.5), неважко записати вихідний опорний розв'язок задачі ЛП, що розглядається,

$$\bar{X}_0 = (a_{10}, a_{20}, \dots, a_{r0}, 0, \dots, 0) \quad (3.1.9)$$

і відповідне значення цільової функції $Z(\bar{X}_0) = a_{00}$.

Із фундаментальної теореми лінійного програмування відомо: якщо оптимальний розв'язок задачі ЛП існує, то він збігається хоча б з одним із опорних розв'язків. Як з'ясувати, чи є знайдений опорний розв'язок найкращим? Як перейти до нового опорного розв'язку, який був би кращим знайденого з точки зору цільової функції?

Вище встановлено, що перехід від одного опорного розв'язку (системи лінійних алгебраїчних рівнянь) до іншого здійснюється за допомогою симплексного перетворення цієї системи.

Можна довести твердження: щоб новий опорний розв'язок був кращим (у загальному випадку не гіршим) за попередній з точки зору цільової функції, необхідно правило вибору розв'язувального елемента при симплексному перетворенні системи рівнянь доповнити наступною умовою вибору розв'язувального стовпця: за розв'язувальний повинен

визбиратись стовпець коефіцієнтів при тій вільній невідомій, наприклад x_p , для якої оцінка a_{0p} від'ємна.

З урахуванням сказаного перехід до нового, кращого, опорного розв'язку повинен здійснюватись за схемою:

1) За розв'язувальний вибирається стовпець елементів при невідомій x_p , для якої оцінка $a_{0p} < 0$, та у відповідному стовпці є хоча б один додатний елемент $a_{ip} > 0$.

2) Розв'язувальний елемент a_{qp} вибирається виходячи з умови

$$\frac{a_{q0}}{a_{qp}} = \min_{\text{для } a_{ip} > 0} \left\{ \frac{a_{i0}}{a_{ip}} \right\}. \quad (3.1.10)$$

3) Елементи розв'язувального (q -го) рядка визначаються за формулою

$$a_{qk}^* = \frac{a_{qk}}{a_{qp}}. \quad (3.1.11)$$

4) Елементи інших рядків, в тому числі і вираз (3.1.5), визначається за формулою («правило прямокутників»)

$$a_{ik}^* = a_{ik} - a_{ip} \frac{a_{qk}}{a_{qp}}. \quad (3.1.12)$$

Згідно до наведених формул приріст цільової функції становить

$$\Delta Z = Z(\bar{X}_1) - Z(\bar{X}_0) = a'_{00} - a_{00} = -a_{0p} \frac{a_{q0}}{a_{qp}}.$$

Оскільки $a_{q0} > 0$, $a_{qp} > 0$, то приріст буде додатним, якщо оцінка $a_{0p} > 0$. А ця умова була прийнята раніше. Таким чином, здійснена ітерація дала змогу одержати новий, кращий опорний розв'язок.

Перелічимо етапи розв'язування задач ЛП симплексним методом:

1) Задача ЛП приводиться до канонічної форми з додатними правими частинами.

2) За допомогою симплексних перетворень задача приводиться до

одиночного базису, або ж встановлюється, що задача не має допустимих розв'язків.

3) Складається симплексна таблиця, обчислюються оцінки вільних невідомих, записується початковий опорний розв'язок і значення цільової функції, яке йому відповідає.

4) Здійснюється перехід до нового кращого опорного розв'язку (у загальному випадку не гіршого).

Покажемо, що новий опорний розв'язок, одержаний в результаті ітерації, здійсненої викладеним способом, буде кращим ніж вихідний.

В результаті ітерації вільна невідома x_p стала базисною, а базисна x_q вилучена із числа базисних (стала вільною). Приріст невідомої x_p становить

$$\Delta x_p = a'_{q0} = \frac{a_{q0}}{a_{qp}}. \quad (3.1.13)$$

Приріст цільової функції, як було сказано вище, дорівнює

$$\Delta Z = Z(\overline{X}_1) - Z(\overline{X}_0) = a'_{00} - a_{00} = -a_{0p} \frac{a_{q0}}{a_{qp}}. \quad (3.1.14)$$

Нагадаємо, що $a_{q0} > 0$, $a_{qp} > 0$. Тоді із (3.1.14) видно, що приріст цільової функції буде додатним, повторимо, тільки, якщо оцінка a_{0p} від'ємна. Твердження доведено.

Отже, після здійснення однієї ітерації одержано новий, кращий опорний розв'язок. Оскільки множина опорних розв'язків скінченна, то після певного числа ітерацій буде знайдено найкращий опорний розв'язок, в тому числі і у випадку, коли задача ЛП не має оптимального розв'язку.

Правила знаходження оптимального розв'язку задачі ЛП симплексним методом формулюємо у вигляді теореми.

Теорема. Нехай після виконання деякої ітерації одержано наступне:

1. Є принаймні одна від'ємна оцінка, наприклад $a_{0p} < 0$, причому у відповідному стовпці коефіцієнтів (при невідомій x_p) є хоча б один додатний елемент $a_{ip} > 0$. Тоді знайдений опорний розв'язок можна покращити, здійснивши наступну ітерацію.

2. Є принаймні одна від'ємна оцінка $a_{0p} < 0$, але додатних коефіцієнтів при відповідній невідомій (невідомих) x_p немає. Тоді цільова функція необмежено зростає на області допустимих розв'язків. Задача лінійного програмування не має оптимального розв'язку.

3. Всі оцінки a_{0p} невід'ємні. Тоді знайдений опорний розв'язок є оптимальним.

Пункт 1 теореми доведено вище. У правильності пункту 2 теореми неважко переконатися. Нехай є одна від'ємна оцінка, наприклад $a_{0,r+1} < 0$, причому у відповідному стовпці коефіцієнтів при x_{r+1} немає жодного додатного. Тоді покращити знайдений опорний розв'язок не можна, оскільки, неможливо вибрати розв'язувальний елемент у відповідності з правилами симплексного перетворення. Разом з тим при необмеженому зростанні вільної невідомої $x_{r+1} \rightarrow +\infty$ має місце наступне. По-перше, базисні невідомі залишаються невід'ємними. Дійсно, якщо доданки $a_{i,r+1}x_{r+1}$ перенести в праві частини рівнянь виду (3.1.1), то легко бачити, що значення відповідних базисних невідомих у цьому випадку одержують невід'ємні прирости, оскільки коефіцієнти $a_{i,r+1} \leq 0$ (згідно з припущенням). По-друге, цільова функція при $x_{r+1} \rightarrow +\infty$ необмежено зростає завдяки тому, що доданок $-a_{0,r+1}x_{r+1}$ приймає як завгодно великі значення. Іншими словами, є нескінченна множина неопорних розв'язків, за яких цільова функція більша ніж при знайденому опорному розв'язку на будь-яке, як завгодно велике, значення.

Пункт 3 впливає з виразу (3.1.14): при всіх $a_{0p} \geq 0$ приріст цільової функції буде $\Delta Z \leq 0$. Отже, за допомогою наступної ітерації знайдений опорний розв'язок покращити неможливо. Він є оптимальним.

Таким чином, в результаті виконання скінченного числа ітерацій або знаходиться оптимальний опорний розв'язок, або ж робиться висновок про те, що задача лінійного програмування не має оптимального розв'язку, оскільки цільова функція необмежено зростає на ОДР.

Розв'язування задачі ЛП викладеним способом зручно здійснювати, користуючись симплексними таблицями типу 3.1. За допомогою цієї таблиці, складеної для задачі (3.1.1) – (3.1.3), легко обчислюються оцінки $a_{0,r+1}, a_{0,r+2}, \dots, a_{0,n}$ вільних невідомих (див. формулу 3.1.8). Наприклад, оцінка $a_{0,r+1}$ знаходиться так: від суми добутків елементів першого з лівого боку стовпця на відповідні коефіцієнти при x_{r+1} , віднімається верхній, записаний над x_{r+1} елемент c_{r+1} .

Інші оцінки обчислюються аналогічно. Елементи першого зліва стовпця та верхнього рядка використовуються після кожної ітерації для контрольного обчислення вказаним способом елементів рядка оцінок (індексного рядка). Якщо ці результати збігаються із одержаними раніше за формулою 3.1.12, то роблять висновок про правильність обчислень.

Варто зауважити, що у другому та третьому стовпцях (зліва) симплексної таблиці міститься інформація про опорний розв'язок задачі та значення цільової функції, яке йому відповідає.

Таблиця 3. 1

$C_{\text{баз}}$	$X_{\text{баз}}$	c_0	c_1	c_2	...	c_r	c_{r+1}	c_{r+2}	...	c_p	...	c_n	$\frac{a_{i0}}{a_{ip}}$
		a_{i0}	x_1	x_2	...	x_r	x_{r+1}	x_{r+2}	...	x_p	...	x_n	
c_1	x_1	a_{10}	1	0	...	0	$a_{1,r+1}$	$a_{1,r+2}$...	$a_{1,p}$...	$a_{1,n}$	

c_2	x_2	a_{20}	0	1	...	0	$a_{2,r+1}$	$a_{2,r+2}$...	$a_{2,p}$...	$a_{2,n}$	
...	
c_r	x_r	a_{r0}	0	0	...	1	$a_{r,r+1}$	$a_{r,r+2}$...	$a_{r,p}$...	$a_{r,n}$	
Z		a_{00}	0	0	...	0	$a_{0,r+1}$	$a_{0,r+2}$...	$a_{0,p}$...	$a_{0,n}$	

Зауваження. Якщо є декілька від'ємних оцінок, то для виконання наступної ітерації доцільно із них вибрати найбільшу за модулем. Такий вибір часто, хоча і не завжди, приводить до зменшення числа ітерацій і дозволяє при розв'язанні задач «вручну» скоріше знайти оптимальний розв'язок.

Приклад. Розв'язати за допомогою симплексного методу наступну задачу лінійного програмування

$$\begin{aligned} -2x_1 + x_2 + x_3 &= 4, \\ 2x_1 + 7x_2 + x_4 &= 92, \\ 2x_1 - 3x_2 + x_5 &= 12, \end{aligned}$$

$$x_i \geq 0, \quad i = 1, 2, \dots, 5,$$

$$Z = 4x_1 + 9x_2 + x_3 + x_4 \rightarrow \max.$$

Розв'язання. Заповнимо симплексну таблицю (табл. 3.2). Визначимо елементи рядка оцінок. В результаті одержимо вихідний опорний розв'язок (початковий опорний план) $\vec{X}_0 = (0; 0; 4; 92; 12)$, якому відповідає наступне значення цільової функції $Z(\vec{X}_0) = 96$.

Таблиця 3.2

$C_{\text{баз}}$	$X_{\text{баз}}$	0	4	9	1	1	0	a_{i0}
		a_{i0}	x_1	x_2	x_3	x_4	x_5	a_{ip}
1	x_3	4	-2	1	1	0	0	-

1	x_4	92	2	7	0	1	0	46
0	x_5	12	2	-3	0	0	1	6
Z		96	-4	-1	0	0	0	

Обидві оцінки від'ємні: $a_{01} = -4$, $a_{02} = -1$, причому у відповідних стовпцях є додатні елементи. Отже, вихідний опорний розв'язок можна покращити, здійснивши ітерацію симплексного перетворення.

Виберемо розв'язувальний елемент $a_{31} = 2$. Виконаємо першу ітерацію, визначивши всі елементи симплексної таблиці (табл. 3.3). Потім здійснимо (повторно) контрольне обчислення елементів рядка оцінок. Після першої ітерації маємо опорний розв'язок $\vec{X}_1 = (6; 0; 16; 80; 0)$ та значення цільової функції, яке йому відповідає $Z(\vec{X}_1) = 120$.

Таблиця 3.3

$C_{\text{баз}}$	$X_{\text{баз}}$	0	4	9	1	1	0	$\frac{a_{i0}}{a_{ip}}$
		a_{i0}	x_1	x_2	x_3	x_4	x_5	a_{ip}
1	x_3	16	0	-2	1	0	1	-
1	x_4	80	0	10	0	1	-1	8
4	x_1	6	1	$-\frac{3}{2}$	0	0	$\frac{1}{2}$	-
Z		120	0	-7	0	0	2	

Серед оцінок є одна від'ємна $a_{02} = -7$, у відповідному стовпці є додатний елемент. Розв'язок можна покращити, здійснивши наступну ітерацію. Виберемо розв'язувальний елемент $a_{22} = 10$. Виконуємо ітерацію (табл. 3.4) та контрольне обчислення елементів рядка оцінок.

Новий опорний розв'язок має вид $\vec{X}_2 = (18; 8; 32; 0; 0)$. Йому відповідає значення цільової функції $Z(\vec{X}_2) = 176$.

Таблиця 3.4

$C_{\text{баз}}$	$X_{\text{баз}}$	0	4	9	1	1	0	$\frac{a_{i0}}{a_{ip}}$
		a_{i0}	x_1	x_2	x_3	x_4	x_5	
1	x_3	32	0	0	1	$\frac{1}{5}$	$\frac{4}{5}$	
9	x_2	8	0	1	0	$\frac{1}{10}$	$-\frac{1}{10}$	
4	x_1	18	1	0	0	$\frac{3}{20}$	$\frac{7}{20}$	
Z		176	0	0	0	$\frac{7}{10}$	$\frac{13}{10}$	

Із таблиці 3.4 видно, що від'ємних оцінок немає. Отже, знайдений опорний розв'язок є оптимальним.

Дамо геометричне тлумачення розглянутій задачі. Її розв'язок, знайдений симплексним методом, порівняємо з розв'язком, знайденим графічним методом (рис. 2.5). Проаналізувавши симплексні таблиці (3.2) – (3.4) та рисунок 2.5, приходимо до висновку, що переходу від вихідного опорного розв'язку \vec{X}_0 до розв'язку \vec{X}_1 , а потім до \vec{X}_2 на графіку відповідає переміщення з вершини $O(0; 0)$ до вершини $N(6; 0)$, а потім до вершини $M(18; 8)$.

Сформулюємо детальний алгоритм розв'язування задач лінійного програмування симплексним методом.

1. Задача ЛП приводиться до канонічної форми.

2. Всі рівняння, які мають від'ємну праву частину, множаться на -1 . Потім за допомогою симплексних перетворень система рівнянь приводиться до одиничного базису або ж з'ясовується, що задача ЛП не має допустимих (невід'ємних) розв'язків.

3. Складається симплексна таблиця, обчислюються оцінки вільних невідомих, записується вихідний опорний розв'язок та значення цільової функції, яке йому відповідає.

4. За певним правилом, перевіряється: має задача ЛП оптимальний розв'язок чи ні.

5. Якщо серед оцінок є принаймні одна від'ємна і при цьому у відповідному стовпці є хоча б один додатний елемент, то здійснюється перехід до нового, кращого, опорного розв'язку і т. д.

6. Після виконання скінченного числа ітерацій процес закінчується, робиться висновок про те, що задача ЛП не має оптимального розв'язку (пункту 4), або ж – знайдено оптимальний розв'язок. При цьому ознакою оптимальності знайденого опорного розв'язку є невід'ємність всіх оцінок.

Зауваження 1. Величини a_{0p} називаються оцінками вільних невідомих. Пояснимо математичний зміст цього терміну.

Нехай $a_{0p} < 0$. Якщо вибрати розв'язувальний елемент, наприклад a_{qp} , здійснити симплексне перетворення, то, повторимо, вільна невідома x_p стане базисною, а базисна x_q буде вилучена із числа базисних. Невідома x_p отримає приріст

$$\Delta x_p = \frac{a_{q0}}{a_{qp}}.$$

Приріст цільової функції буде дорівнювати

$$\Delta Z = -a_{0p} \cdot \frac{a_{q0}}{a_{qp}},$$

або

$$\Delta Z = -a_{0p} \cdot \Delta x_p. \quad (3.1.15)$$

Математичний зміст виразу (3.1.15): значення a_{0p} дорівнює приросту цільової функції при зміні x_p на одиницю, тобто при $\Delta x_p = 1$. Іншими словами, величина a_{0p} дозволяє оцінити, на скільки збільшиться цільова функція, якщо вільну невідому x_p включити до числа базисних зі значенням рівним одиниці.

Зауваження 2. Нехай за умовою задачі необхідно мінімізувати цільову функцію $Z = \sum_{i=1}^n c_i x_i$.

У цьому випадку для розв'язання задачі ЛП можна скористуватись одним із підходів:

1) Задача мінімізації функції Z замінюється на задачу максимізації функції $Z_1 = -Z$, знаходиться оптимальний розв'язок (якщо він існує) та $Z_{1\max}$. Потім для знайденого розв'язку визначається Z_{\min} за формулою

$$Z_{\min} = -Z_{1\max}.$$

2) Змінюється умова вибору розв'язувального елемента (при розв'язуванні задачі симплексним методом) та критерій оптимальності розв'язку задачі ЛП з функцією Z на наступну: за розв'язувальний вибирається стовпець коефіцієнтів при невідомій x_p , для якої оцінка додатна ($a_{0p} > 0$) і т.ін; критерій оптимального розв'язку – всі оцінки недодатні.

3.2 Альтернативний оптимум

Вище (п. 2.3) йшла мова про так званий альтернативний оптимум задачі ЛП, який має місце у випадку, коли лінії рівня паралельні тій грані опуклого багатокутника (ОДР), на якій досягається найбільше значення цільової функції.

Альтернативний оптимум задач лінійного програмування з двома вільними невідомими характеризується наявністю двох оптимальних опорних розв'язків та нескінченного числа неопорних оптимальних розв'язків (геометрично їм відповідають координати точок на прямій лінії, яка з'єднує дві кутові точки).

Неважко зрозуміти, що задача ЛП з трьома вільними невідомими та площиною рівня, паралельною відповідній грані опуклого багатогранника (ОДР), може мати велике (проте визначене) число оптимальних опорних розв'язків, яким відповідають вершини вказаної грані ОДР. Оптимальним неопорним розв'язкам відповідають точки на цій же грані.

Природно, що у загальному випадку задача ЛП може мати будь-яке визначене число оптимальних опорних розв'язків $\bar{X}_1, \bar{X}_2, \dots, \bar{X}_l$. Можна довести, що кожний розв'язок, знайдений за формулою

$$\bar{X} = \sum_{i=1}^l t_i \bar{X}_i \quad (3.2.1)$$

де

$$\sum_{i=1}^l t_i = 1, \quad t_i \geq 0, \quad i = 1, 2, \dots, l, \quad (3.2.2)$$

також буде оптимальним. Геометричний образ розв'язку (3.2.1) – точка на відповідній грані ОДР.

Що є ознакою альтернативного оптимуму?

Розглянемо вираз для приросту цільової функції

$$\Delta Z = -a_{0p} \cdot \frac{a_{q0}}{a_{qp}} \quad (3.2.3)$$

Із (3.2.3) видно, що $\Delta Z = 0$, якщо оцінка a_{0p} дорівнює нулю.

Нехай в результаті здійснення деякого числа ітерацій симплексним методом знайдено опорний розв'язок задачі ЛП, причому оцінки вільних невідомих додатні, крім принаймні однієї, рівної нулю, наприклад $a_{0p} = 0$.

Згідно з основною теоремою симплексного методу знайдений опорний розв'язок є оптимальним. Разом з тим, якщо серед коефіцієнтів при невідомій x_p (у відповідному стовпці симплексної таблиці) є хоча б один додатний елемент, то, виконавши наступну ітерацію, можна перейти до нового опорного розв'язку, для якого цільова функція має те ж саме значення що і для раніше знайденого оптимального розв'язку, оскільки, $\Delta Z = 0$.

Отже, в цьому випадку новий опорний розв'язок також буде оптимальним.

Таким чином, ознакою існування альтернативного оптимуму є наявність в оптимальній симплексній таблиці принаймні однієї рівної нулю оцінки (решта оцінок – додатні) та у відповідному стовпці хоча б одного додатного елемента.

Для знаходження оптимального розв'язку за формулою (3.2.1) необхідно визначити всі оптимальні опорні розв'язки, для яких цільова функція приймає одне і те ж максимальне (мінімальне) значення. Пошук останніх повинен здійснюватись шляхом послідовного введення вільних невідомих, оцінки яких дорівнюють нулю, до числа базисних. При цьому,

природно, необхідно слідкувати за тим, щоб опорні розв'язки не повторювались.

Зауваження. Розглянуті задачі лінійного програмування є однокритеріальними. Критерій оптимальності – максимум (мінімум) цільової функції. У випадку альтернативного оптимуму за оптимальний можна вибрати розв'язок, знайдений за формулою (3.2.1), причому такий, що враховує поряд з основним деякий інший критерій. Іншими словами, в цьому випадку є принципова можливість при виборі оптимального розв'язку врахувати деякі інші, не відображені в початковій моделі задачі ЛП, вимоги, умови, критерії, обмеження тощо.

3.3 Поняття про вироджені задачі лінійного програмування

При розгляді симплексного методу виходили з припущення, що всі праві частини системи обмежень – рівностей, приведеної до одиничного базису, додатні. Розглянемо випадок, коли хоча б у одному з рівнянь права частина, наприклад a_{q0} , дорівнює нулю. Тоді у вихідному опорному розв'язку відповідна базисна невідома x_q дорівнює нулю.

Означення. Опорний розв'язок, у якому хоча б одна (або декілька) базисна невідома дорівнює нулю, називається виродженим, а відповідна задача лінійного програмування – виродженою.

Нехай деяка оцінка, наприклад a_{0p} , менша за нуль і у відповідному стовпці симплексної таблиці є принаймні один додатний елемент. Може трапитись так, що розв'язувальним елементом виявиться a_{qp} . Тоді здійснивши наступну ітерацію симплексним методом, одержимо новий опорний розв'язок, базисні невідомі якого відповідно дорівнюють

$$a'_{q0} = \frac{a_{q0}}{a_{qp}} = 0, \quad (3.3.1)$$

$$a'_{i0} = a_{i0} - a_{q0} \frac{a_{ip}}{a_{qp}} = a_{i0}, \quad i \neq q. \quad (3.3.2)$$

Із (3.3.1) та (3.3.2) видно, що нові значення базисних невідомих дорівнюють старим, у новому опорному розв'язку колишня вільна невідома x_p (рівна нулю) стала базисною, рівною нулю, а колишня базисна невідома x_q , рівна нулю, стала вільною, також рівною нулю. Значення інших базисних невідомих не змінились. Іншими словами, новий опорний розв'язок повністю збігається з попереднім, тобто ітерація здійснена, але кращого опорного розв'язку не одержано («холоста» ітерація). Правда, при цьому змінився базис, і як наслідок, склад базисних невідомих, змінились елементи симплексної таблиці, в тому числі в загальному випадку змінились оцінки вільних невідомих.

У подальшому може трапитись, що в результаті наступної ітерації знову буде одержано той же опорний розв'язок, який відповідає новому базису.

Оскільки множина базисів у векторному просторі задачі ЛП є скінченною, то після деякого числа «холостих» ітерацій можлива одна з наступних ситуацій:

1. Знайдений опорний розв'язок, на якому «тупцювали», є оптимальним.
2. Задача лінійного програмування не має оптимального розв'язку, оскільки цільова функція не обмежена на ОДР.
3. Одержано новий опорний розв'язок.
4. Знайдений опорний розв'язок не змінився, більше того, він відповідає одному з тих базисів, який вже зустрічався на одній із попередніх ітерацій.

Ситуації 1, 2, 3 слід вважати благополучними. Ситуація 4 характеризується тим, що має місце та сама симплексна таблиця, яка вже була декілька ітерацій тому. Причому така ситуація може повторюватись як завгодно багато раз. У такому випадку кажуть про зациклювання процесу розв'язування задачі лінійного програмування.

Зауваження. 1. Зациклювання процесу розв'язування задачі ЛП виникає надзвичайно рідко. 2. Існують принципові нескладні уточнення до правила вибору розв'язувального елемента, які гарантують відсутність (неможливість) зациклювання. Проте врахування їх в програмі для ЕОМ обумовлює суттєве ускладнення її та значне збільшення обсягу обчислень.

3.4 Симплексний метод зі штучним базисом

Здійснювати пошук оптимального розв'язку задачі ЛП розглянутим вище методом зручно в тих випадках, коли система обмежень – рівностей вже приведена (або легко може бути приведена) до одиничного базису, якому відповідає вихідний опорний розв'язок. В тих випадках, коли система не приведена до одиничного базису, оптимальний розв'язок задачі ЛП можна знайти за допомогою симплексного методу із введенням так званого штучного базису.

Суть цього методу у наступному:

1. В обмеження задачі штучно вводиться декілька додаткових невідомих для того, щоб одержана нова система рівнянь містила одиничний базис і можна було б легко записати вихідний опорний розв'язок.

2. Замість початкової задачі ЛП симплексним методом розв'язується задача із вказаними обмеженнями та деякою, спеціально побудованою, цільовою функцією.

$$Y = (0, 0, \dots, 0, a_{10}, a_{20}, \dots, a_{m0}). \quad (3.4.9)$$

Невід'ємні невідомі $x_{n+1}, x_{n+2}, \dots, x_{n+m}$ штучно введені для того, щоб можна було одержати вказаний розв'язок. Система одиничних векторів, координатами яких є коефіцієнти при штучних невідомих, утворює штучний базис m -вимірному векторного простору.

Оптимальний розв'язок M -задачі можна знайти викладеним вище симплексним методом.

Можна довести наступне твердження. Якщо для всіх достатньо великих чисел M :

1. Розширена задача має оптимальний розв'язок, у якому штучні невідомі дорівнюють нулю, то, опустивши їх, неважко одержати оптимальний розв'язок початкової задачі ЛП.

2. Розширена задача має оптимальний розв'язок, у якому принаймні одна штучна невідома відмінна від нуля, то початкова задача ЛП не має допустимих розв'язків (система обмежень несумісна).

3. M -задача не має оптимального розв'язку, то початкова задача ЛП також не має оптимального розв'язку.

Сформульоване твердження встановлює зв'язок між оптимальними розв'язками початкової та розширеної задачі.

Таким чином, якщо знайдено такий оптимальний розв'язок розширеної задачі, у якому змінні $x_{n+1}, x_{n+2}, \dots, x_{n+m}$ дорівнюють нулю, то це означає, що по суті, знайдено оптимальний розв'язок початкової задачі ЛП.

Приступаючи до розв'язування M -задачі симплексним методом, цільову функцію (3.4.5) виразимо через вільні невідомі. Повторюючи міркування, аналогічні викладеним у п. 3.1, одержимо

$$W + a_{01}x_1 + a_{02}x_2 + \dots + a_{0n}x_n = a_{00}, \quad (3.4.10)$$

де

$$a_{00} = -c_0 - M \sum_{i=1}^m a_{i0}, \quad (3.4.11)$$

$$a_{0p} = -c_p - M \sum_{i=1}^m a_{ip}, \quad p = 1, 2, \dots, n. \quad (3.4.12)$$

Тоді значення цільової функції, яке відповідає вихідному опорному розв'язку

$$\bar{Y}_0 = (0, 0, \dots, 0, a_{10}, a_{20}, \dots, a_{m0})$$

розширеної задачі, становить

$$W(\bar{Y}_0) = a_{00}. \quad (3.4.13)$$

Пошук оптимального розв'язку M -задачі може здійснюватись симплексним методом, викладеним у п. 3.1, з використанням симплексних таблиць. При цьому попередньо повинно бути вибрано конкретне достатньо велике число M .

Разом з тим розширена задача може розв'язуватися при будь-якому достатньо великому числі M , тобто, коли конкретне значення M не вибирається. У цьому випадку зручно дещо змінити симплексні таблиці, які зустрічалися раніше.

Вирази для цільової функції, доданку a_{00} , оцінок a_{0p} перепишемо так:

$$W = Z - M \sum_{i=n+1}^{n+m} x_i, \quad (3.4.14)$$

$$a_{00} = a'_{00} + Ma''_{00}, \quad (3.4.15)$$

$$a_{0p} = a'_{0p} + Ma''_{0p}, \quad (3.4.16)$$

де

$$a'_{00} = -c_0, \quad a''_{00} = - \sum_{i=1}^m a_{i0}, \quad (3.4.17)$$

$$a'_{0p} = -c_p, \quad a''_{0p} = -\sum_{i=1}^m a_{ip}, \quad (3.4.18)$$

Із (3.4.14) – (3.4.16) видно, що і значення цільової функції M -задачі і значення a_{00} і оцінок a_{0p} складаються із двох незалежних частин.

Причому одна з них залежить від M , а друга – ні.

З урахуванням сказаного уявляється зручним, по-перше, індексний рядок симплексної таблиці розбити на два: з оцінками відповідно a'_{0p} та a''_{0p} (нижній рядок); по-друге, ввести вектори \bar{C}' та \bar{C}'' .

$$\bar{C}' = (c_1, c_2, \dots, c_n, \dots, \overbrace{0, 0, \dots, 0}^{m - \text{нулів}}), \quad (3.4.19)$$

$$\bar{C}'' = (\underbrace{0, 0, \dots, 0}_n, -M, -M, \dots, -M). \quad (3.4.20)$$

Проте, для забезпечення контролю розрахунків ввести стовпець та рядок з елементами рівними відповідно $\frac{1}{M}C''_{\text{баз}}$ та $\frac{1}{M}C''_i$. Підрядковим індексом «баз» позначаються координати вектора, які відповідають базисним невідомим.

Варто зазначити, що роль доданків із коефіцієнтом M у цільовій функції інтуїтивно зрозуміла і полягає у наступному: ці доданки суттєво зменшують значення цільової функції, яка максимізується. У оптимальному розв'язку (при великому числі M) ці доданки повинні дорівнювати нулю.

Отже, при здійсненні перетворень з використанням симплексних таблиць із числа базисних повинні виключатись в першу чергу штучні невідомі з максимальною за модулем від'ємною оцінкою a''_{0p} .

При цьому штучний вектор, виключений із базису, у подальшому (на наступних ітераціях) недоцільно вводити в базис. У відповідності з

цим перетворення елементів стовпців, які є коефіцієнтами при штучних невідомих, виключених із числа базисних, як правило, зайве. Їх можна вилучати із симплексних таблиць.

Зауважимо, що перерахування елементів симплексних таблиць при переході від одного опорного розв'язку M -задачі до іншого здійснюється за викладеними вище правилами симплексного методу.

Елементи верхнього та нижнього рядків оцінок визначаються з використанням коефіцієнтів відповідно $C'_{\text{баз}}$ та $\frac{1}{M}C''_{\text{баз}}$.

Ітераційний процес з використанням рядка оцінок a''_{0p} здійснюється до тих пір, поки або всі штучні невідомі будуть виключенні із числа базисних, або ж у вказаному рядку не буде від'ємних оцінок, хоча і не всі штучні вектори виключені із базису. У першому випадку пошук оптимального розв'язку задачі продовжується з використанням верхнього рядка оцінок (a'_{0p}).

Можна довести, що у другому випадку, якщо $a''_{00} < 0$, то задача (3.4.1) – (3.4.4) не має розв'язку, оскільки виконання вказаної нерівності означає, що принаймні одна із штучних базисних невідомих у одержаному опорному розв'язку відміна від нуля; якщо ж $a''_{00} = 0$, то знайдений опорний розв'язок початкової задачі є виродженим, оскільки праві частини рівнянь, розв'язаних відносно штучних невідомих, дорівнюють нулю. У цьому випадку вибір розв'язувального елемента при подальших ітераціях необхідно здійснювати у стовпці, для якого, по-перше, оцінка a''_{0p} дорівнює нулю, по-друге, оцінка a'_{0p} має найбільше за модулем від'ємне значення. Процес покращення розв'язку необхідно завершити, тоді, коли не буде жодної від'ємної оцінки a'_{0p} , яка стоїть над нульовою оцінкою a''_{0p} .

Перелічимо етапи розв'язування задачі лінійного програмування симплексним методом зі штучним базисом.

1. Складається (формулюється) M -задача.

2. Шукається вихідний опорний розв'язок M -задачі.

3. За допомогою симплексних перетворень виключаються штучні невідомі із числа базисних. В результаті або визначається опорний розв'язок початкової задачі, або встановлюється її несумісність.

4. Використовуючи знайдений опорний розв'язок, симплексним методом знаходиться оптимальний розв'язок початкової задачі або ж робиться висновок про його відсутність.

Приклад. Симплексним методом із штучним базисом розв'язати наступну задачу лінійного програмування:

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

$$x_i \geq 0, \quad i = 1, 2, \dots, 4,$$

$$Z = x_1 - x_2 - 5x_3 + x_4 + 4 \rightarrow \max.$$

Розв'язання. Система обмежень – рівнянь не приведена до одиничного базису. Додамо до кожного рівняння по одній невід'ємній штучній невідомій (відповідно $x_5 \geq 0$, $x_6 \geq 0$) та перейдемо до M -задачі

$$\begin{cases} x_1 + 3x_2 + 3x_3 + x_4 + x_5 = 3, \\ 2x_1 + 3x_2 - x_4 + x_6 = 4, \end{cases}$$

$$x_i \geq 0, \quad i = 1, 2, \dots, 6,$$

$$W = x_1 - x_2 - 5x_3 + x_4 + 4 - M(x_5 + x_6) \rightarrow \max.$$

Одержану задачу будемо розв'язувати симплексним методом, вважаючи, що M – деяке дуже велике додатне число. Побудуємо початкову симплексну таблицю (табл. 3.5). Нагадаємо, що у верхньому

рядку оцінок записуються величини a'_{0p} , незалежні від M , у нижньому a''_{0p} – обумовлені введенням штучного базису.

Таблиця 3. 5

$\frac{1}{M} C''_{\text{баз}}$	$C'_{\text{баз}}$	$\frac{1}{M} C''$	0	0	0	0	0	-1	-1	$\frac{a_{i0}}{a_{ip}}$
		C'	-4	1	-1	-5	1	0	0	
		$x_{\text{баз}}$	a_{i0}	x_1	x_2	x_3	x_4	x_5	x_6	
-1	0	x_5	3	1	3	3	1	1	0	3/3
-1	0	x_6	4	2	3	0	-1	0	1	4/3
W			4	-1	1	5	-1	0	0	
			-7	-3	-6	-3	0	0	0	

Зауважимо, що елементи нижнього рядка (a''_{0p}) таблиці 3.5 розраховані за формулою (3.4.18) з використанням елементів першого (зліва) стовпця, та елементів верхнього рядка. У подальших розрахунках ці елементи a''_{0p} будуть обчислюватись двічі:

- перший раз за «правилом прямокутника»,
- другий (для контролю) – з використанням елементів названих контрольних стовпця та рядка.

Вихідний опорний розв’язок M -задачі має вид

$$\bar{Y}_0 = (0, 0, 0, 0, 3, 4).$$

Йому відповідає значення цільової функції

$$W(\bar{Y}_0) = 4 - 7M.$$

Неважко бачити, що вказаний розв’язок M -задачі не є оптимальним, оскільки є від’ємні оцінки $a''_{01} = -3$, $a''_{02} = -6$, $a''_{03} = -3$. Найбільша за модулем оцінка $a''_{02} = -6$. Невідому x_2 вводимо в число базисних, а x_5 виключаємо із числа базисних. Побудуємо нову таблицю (табл. 3.6)

Таблиця 3.6

$\frac{1}{M}C''_{\text{баз}}$	$C'_{\text{баз}}$	$\frac{1}{M}C''$	0	0	0	0	0	-1	-1	$\frac{a_{i0}}{a_{ip}}$
		C'	-4	1	-1	-5	1	0	0	
		$x_{\text{баз}}$	a_{i0}	x_1	x_2	x_3	x_4	x_5	x_6	
0	-1	x_2	1	1/3	1	1	1/3	1/3	0	3
-1	0	x_6	1	1	0	-3	-2	-1	1	1
W			3	-4/3	0	4	-4/3	-1/3	0	
			-1	-1	0	3	2	2	0	

Знайдений опорний розв'язок $\bar{Y}_1 = (0, 1, 0, 0, 0, 1)$ M -задачі, якому відповідає значення цільової функції $W(\bar{Y}_1) = 3 - M$, також не є оптимальним. Продовжуємо процес поліпшення розв'язку. В число базисних вводимо невідому x_1 , а штучну невідому x_6 виключаємо із числа базисних.

В результаті здійснення другої ітерації (табл. 3.7) одержано опорний розв'язок M -задачі $\bar{Y}_2 = (1, 2/3, 0, 0, 0, 0)$ та відповідне значення цільової функції $W(\bar{Y}_2) = 13/3$.

Таблиця 3.7

$\frac{1}{M}C''_{\text{баз}}$	$C'_{\text{баз}}$	$\frac{1}{M}C''$	0	0	0	0	0	-1	-1	$\frac{a_{i0}}{a_{ip}}$
		C'	-4	1	-1	-5	1	0	0	
		$x_{\text{баз}}$	a_{i0}	x_1	x_2	x_3	x_4	x_5	x_6	
0	-1	x_2	2/3	0	1	2	1	2/3	-1/3	2/3
0	1	x_1	1	1	0	-3	-2	-1	1	-
W			13/3	0	0	0	-4	-5/3	4/3	-
			0	0	0	0	0	1	1	

В базис входять тільки основні вектори (коефіцієнтів при основних невідомих x_1, x_2) оскільки всі штучні невідомі виключені із числа базисних. Отже, згідно зі сказаним вище, стовпці, які відповідають штучним невідомим x_5 та x_6 , а також рядок оцінок a''_{0p} , стовпець $\frac{1}{M}C''_{\text{баз}}$, а також рядок $\frac{1}{M}C''$, опустимо.

У подальшому пошук оптимального розв'язку початкової задачі будемо здійснювати, використовуючи звичайну симплексну таблицю.

Із таблиці 3.7 видно, що є від'ємна оцінка $a'_{04} = -4$. Це означає, що знайдений опорний розв'язок $\bar{X}_2 = (1, 2/3, 0, 0)$, якому відповідає значення цільової функції $Z(\bar{X}_2) = 13/3$, не є оптимальним.

В число базисних невідомих введемо x_4 , виключимо x_2 . Одержимо симплексну таблицю з оптимальним розв'язком (табл. 3.8).

Таблиця 3. 8

$C'_{\text{баз}}$	C'	-4	1	-1	-5	1	$\frac{a_{i0}}$
	$x_{\text{баз}}$	a_{i0}	x_1	x_2	x_3	x_4	a_{ip}
1	x_4	2/3	0	1	2	1	
1	x_1	7/3	1	2	1	0	
Z		7	0	4	8	0	

Оптимальний розв'язок початкової задачі має вид

$$\bar{X}_{\text{опт}} = (7/3, 0, 0, 2/3).$$

Йому відповідає максимальне значення цільової функції $Z_{\text{max}} = 7$.

Отже, нами детально розглянута процедура одержування оптимального розв'язку задачі ЛП симплексним методом зі штучним базисом у тому випадку, коли вихідний опорний розв'язок задачі невідомий, або, іншими

словами, коли система обмежень – рівнянь не приведена до одиничного базису, якому відповідає допустимий, а отже, і опорний розв’язок.

3.5 Двоїстість (спряженість) у лінійному програмуванні

3.5.1. Поняття про двоїсті (спряжені) задачі

Розглянемо наступні дві економічні задачі. Підприємство має чотири види сировини *A, B, C, D* у кількості відповідно 40, 30, 10, 60 одиниць. Підприємство може здійснювати випуск трьох видів продукції. Витрати сировини на одиницю продукції, що випускається, надані у таблиці 3.9. Крім того, відомі ціни на продукцію.

Таблиця 3. 9

Сировина		Витрати сировини на одиницю продукції, яку виробляють			Ціна одиниці сировини, тис. грн
Вид	Кількість	I	II	III	
A	40	4	3	5	y_1
B	30	3	5	2	y_2
C	10	1	2	1	y_3
D	60	2	3	4	y_4
Ціна одиниці продукції, тис. грн		15	10	20	

Задача 1. Необхідно визначити план виробництва, який би забезпечив найбільшу вартість продукції, що випускається. Позначимо x_1, x_2, x_3 – кількість продукції, яка випускається, відповідно першого, другого та третього видів.

Математична модель задачі 1. Знайти вектор $\vec{X} = (x_1, x_2, x_3)$, координати якого задовольняють умовам

$$\begin{cases} 4x_1 + 3x_2 + 5x_3 \leq 40, \\ 3x_1 + 5x_2 + 2x_3 \leq 30, \\ x_1 + 2x_2 + x_3 \leq 10, \\ 2x_1 + 3x_2 + 4x_3 \leq 60, \end{cases} \quad (3.5.1)$$

$$x_i \geq 0, \quad i = 1, 2, 3 \quad (3.5.2)$$

та забезпечують максимум цільової функції

$$Z = 15x_1 + 10x_2 + 20x_3 \rightarrow \max \quad (3.5.3)$$

Задача 2. Розглянуте підприємство має можливість не здійснювати випуск продукції, а реалізувати сировину за договірними цінами y_1, y_2, y_3, y_4 , які встановлюються, виходячи із наступних міркувань: по-перше, підприємство-продавець прагне отримати за продаж сировини суму, не меншу ніж від реалізації продукції; виготовленої із вказаної сировини; по-друге, підприємство-покупець переслідує мету: мінімізувати сумарну вартість сировини.

Математична модель задачі 2. Знайти вектор $\vec{Y} = (y_1, y_2, y_3, y_4)$, який задовольняє умовам

$$\begin{cases} 4y_1 + 3y_2 + y_3 + 2y_4 \geq 15, \\ 3y_1 + 5y_2 + 2y_3 + 3y_4 \geq 10, \\ 5y_1 + 2y_2 + y_3 + 4y_4 \geq 20, \end{cases} \quad (3.5.4)$$

$$y_i \geq 0, \quad i = 1, 2, 3, 4 \quad (3.5.5)$$

та забезпечує мінімум цільової функції

$$W = 40y_1 + 30y_2 + 10y_3 + 60y_4 \rightarrow \min. \quad (3.5.6)$$

Аналіз математичних моделей двох задач, що розглядаються, дозволяє помітити наступні особливості їх:

1) число невідомих величин у першій задачі дорівнює числу основних обмежень другої задачі і навпаки;

2) матриця коефіцієнтів другої задачі одержана із матриці коефіцієнтів першої задачі транспонуванням;

4.1 Вихідна задача

$$Z = CX \rightarrow \min,$$

$$AX = A_0,$$

$$X \geq 0.$$

4.2 Двоїста (спряжена) задача

$$W = YA_0 \rightarrow \max,$$

$$YA \leq C.$$

Зауваження. У двоїстих несиметричних задачах лінійного програмування на невідомі величини не накладається умова їх невід'ємності.

У подальшому будемо розглядати лише спряжені симетричні задачі.

Сформулюємо правила складання задач, спряжених по відношенню до вихідних.

1. Всі обмеження – нерівності вихідної задачі приводяться до нерівностей однакового змісту: відповідно « \leq » у випадку задачі максимізації та « \geq » – у задачі мінімізації. З цією метою всі нерівності, в яких вказана вимога не виконується, множаться на -1 .

2. Записується матриця-рядок невідомих спряженої задачі $Y = (y_1, y_2, \dots, y_m)$, де m – число основних обмежень вихідної задачі.

3. Складається система обмежень спряженої задачі. З цією метою здійснюється наступне:

- а) матриця-рядок невідомих $Y = (y_1, y_2, \dots, y_m)$ множиться на матрицю A ;
- б) за вільні члени нерівностей вибираються коефіцієнти при невідомих у виразі для цільової функції вихідної задачі;
- в) записуються нерівності протилежного змісту у порівнянні з нерівностями вихідної задачі (після виконання п.1).

4. Складається вираз для цільової функції спряженої задачі. При цьому за коефіцієнти при невідомих вибираються вільні члени обмежень – нерівностей вихідної задачі (після виконання п.1).

5. Визначається вид екстремуму цільової функції спряженої задачі: мінімум (максимум), якщо у вихідній задачі необхідно знайти максимум (мінімум).

6. Записується умова невід'ємності невідомих спряженої задачі.

Приклад. Скласти задачу, спряжену до наступної:

$$\begin{aligned} Z &= 2x_1 + x_2 \rightarrow \max, \\ -x_1 + 3x_2 &\leq 33, \\ 3x_1 + 8x_2 &\leq 139, \\ 11x_1 - 5x_2 &\leq 132, \\ x_1 + x_2 &\geq 5, \\ x_1 \geq 0, \quad x_2 &\geq 0. \end{aligned}$$

У відповідності з п.1 правил складання спряженої задачі четверту нерівність помножимо на -1 . В результаті одержимо

$$\begin{aligned} -x_1 + 3x_2 &\leq 33, \\ 3x_1 + 8x_2 &\leq 139, \\ 11x_1 - 5x_2 &\leq 132, \\ -x_1 - x_2 &\leq -5. \end{aligned}$$

Матриця-рядок невідомих спряженої задачі має вид

$$Y = (y_1, y_2, y_3, y_4).$$

Помножимо цю матрицю на матрицю коефіцієнтів при невідомих вихідної задачі

$$(y_1, y_2, y_3, y_4) \times \begin{pmatrix} -1 & 3 \\ 3 & 8 \\ 11 & -5 \\ -1 & -1 \end{pmatrix} = \begin{pmatrix} -y_1 + 3y_2 + 11y_3 - y_4 \\ 3y_1 + 8y_2 - 5y_3 - y_4 \end{pmatrix}.$$

Запишемо систему обмежень

$$\begin{cases} -y_1 + 3y_2 + 11y_3 - y_4 \geq 2, \\ 3y_1 + 8y_2 - 5y_3 - y_4 \geq 1. \end{cases}$$

$y_{m+1}, y_{m+2}, \dots, y_{m+n}$ – відповідно основні та додаткові невідомі спряженої задачі.

Теорема 2 (без доведення). Оптимальні значення невідомих спряженої (вихідної) задачі лінійного програмування дорівнюють абсолютним значенням коефіцієнтів при відповідних невідомих цільової функції вихідної (спряженої) задачі, для якої знайдено не вироджений оптимальний розв'язок.

Із сформульованих теорем випливає, що між оптимальними розв'язками вихідної та спряженої задач існує відповідність. Знаючи оптимальний розв'язок однієї задачі, неважко записати оптимальний розв'язок другої задачі.

Варто зазначити, що розв'язок спряженої задачі може допомогти і дуже часто допомагає глибше розкрити економічний зміст розв'язку вихідної задачі.

Приклад. Розв'язати наступну задачу лінійного програмування:

$$Z = 8x_1 + 9x_2 \rightarrow \min,$$

$$\begin{cases} 2x_1 + x_2 \geq 3, \\ x_1 + 3x_2 \geq 2, \end{cases}$$

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

Сформулюємо задачу, спряжену (двоїсту) до вихідної

$$W = 3y_1 + 2y_2 \rightarrow \max,$$

$$\begin{cases} 2y_1 + y_2 \leq 8, \\ y_1 + 3y_2 \leq 9, \end{cases}$$

$$y_1 \geq 0, y_2 \geq 0.$$

Для розв'язання цієї задачі введемо дві додаткові невідомі

$$\begin{cases} 2y_1 + y_2 + y_3 = 8, \\ y_1 + 3y_2 + y_4 = 9. \end{cases}$$

Спряжену задачу розв'язуємо симплексним методом (табл. 3.10)

Таблиця 3. 10

$C_{\text{баз}}$	$Y_{\text{баз}}$	0	3	2	0	0	a_{i0}
		a_{i0}	y_1	y_2	y_3	y_4	a_{ip}
0	y_3	8	2	1	1	0	8/2
0	y_4	9	1	3	0	1	9/1
W		0	-3	-2	0	0	
3	y_1	4	1	1/2	1/2	0	8
0	y_4	5	0	5/2	-1/2	1	2
W		12	0	-1/2	3/2	0	
3	y_1	3	1	0	3/5	-1/5	
2	y_2	2	0	1	-1/5	2/5	
W		13	0	0	7/5	1/5	

Оптимальний розв'язок спряженої задачі

$$\vec{Y} = (3, 2, 0, 0), W_{\max} = 13.$$

Оптимальний розв'язок вихідної задачі знайдемо, використовуючи елементи рядка оцінок у симплексній таблиці 3.10 після виконання останньої ітерації, а також враховуючи наступну відповідність між невідомими вихідної та спряженої задач:

$$\begin{array}{cc} \overbrace{x_1, x_2,} & \overbrace{x_3, x_4,} \\ \updownarrow \quad \updownarrow, & \updownarrow \quad \updownarrow \\ y_3, y_4, & y_1, y_2. \end{array}$$

З урахуванням сказаного запишемо $x_3 = 0$, $x_4 = 0$, $x_1 = 7/5$, $x_2 = 1/5$, або $\vec{X} = (7/5, 1/5, 0, 0)$. При оптимальному розв'язку вихідної задачі цільова функція досягає найменшого значення $Z_{min} = 8 \cdot \frac{7}{5} + 9 \cdot \frac{1}{5} = 13$, тобто $Z_{min} = W_{max}$.

Зауваження. На теорії двоїстості (спряженості) базується так званий прямо двоїстий симплексний метод розв'язування задач ЛП, який дозволяє у разі великої кількості невідомих суттєво спростити процедуру та зменшити обсяг обчислень. Детальніше про це дивись, наприклад, у [23].

Питання для самоконтролю

1. В чому полягає симплексний метод розв'язування задач ЛП?
2. Яку геометричну інтерпретацію можна надати симплексному методу?
3. Що таке альтернативний оптимум? Що є ознакою його існування?
4. Що таке зациклювання? В яких задачах воно може відбутися?
5. Що таке розширена задача лінійного програмування?
6. В яких випадках для розв'язування задач ЛП застосовується симплексний метод зі штучним базисом?
7. Коли і в якому випадку, маючи оптимальний розв'язок M -задачі, неважко одержати оптимальний розв'язок вихідної задачі?
8. Як у процесі розв'язування M -задачі можна зробити висновок про несумісність вихідної задачі ЛП?
9. Назвіть етапи розв'язування задачі ЛП симплексним методом зі штучним базисом.
10. В чому полягає суть двоїстості (спряженості) у лінійному програмуванні?

11. Назвіть правила формулювання (складання) задачі, спряженої до симетричної вихідної задачі ЛП.
12. Сформулюйте основні теореми двоїстості.
13. Як знайти оптимальний розв'язок вихідної задачі ЛП, використовуючи дані симплексної таблиці з оптимальним розв'язком спряженої задачі?

Задачі до глави 3

3.1. Розв'язати симплексним методом наступні задачі лінійного програмування

$$\begin{aligned} 1) \quad & x_1 + 2x_2 + x_3 = 16, \\ & 3x_1 - 5x_2 + x_4 = 4, \\ & x_i \geq 0, \quad i = 1, 2, 3, 4, \end{aligned}$$

$$Z = 8x_2 + x_3 - x_4 - 10 \rightarrow \max.$$

$$\begin{aligned} 2) \quad & x_1 + x_3 + x_4 = 9, \\ & x_2 - 2x_3 + 2x_4 = 6, \\ & x_i \geq 0, \quad i = 1, 2, 3, 4, \end{aligned}$$

$$Z = x_1 + x_2 + x_3 + 8x_4 - 10 \rightarrow \max.$$

$$\begin{aligned} 3) \quad & -x_1 + 2x_2 + x_3 = 18, \\ & 3x_1 + x_2 + x_4 = 30, \\ & x_i \geq 0, \quad i = 1, 2, 3, 4, \end{aligned}$$

$$Z = -3x_1 - 5x_2 - x_3 - x_4 + 40 \rightarrow \min.$$

$$\begin{aligned} 4) \quad & x_1 + 2x_3 + x_4 = 13, \\ & x_2 + x_3 + 2x_4 = 11, \\ & x_i \geq 0, \quad i = 1, 2, 3, 4, \end{aligned}$$

$$Z = x_1 + x_2 + 7x_3 + 5x_4 - 20 \rightarrow \max.$$

$$\begin{aligned} 5) \quad & x_1 + x_3 - x_4 = 2, \\ & x_2 - x_3 + 4x_4 = 4, \\ & x_i \geq 0, \quad i = 1, 2, 3, 4, \end{aligned}$$

$$Z = 5 - x_1 - x_2 - 2x_3 - 4x_4 \rightarrow \min.$$

$$\begin{aligned} 6) \quad & x_1 + 2x_3 - x_4 = 7, \\ & x_2 + x_3 + 3x_4 = 14, \\ & x_i \geq 0, \quad i = 1, 2, 3, 4, \end{aligned}$$

$$Z = x_1 + x_2 + 5x_3 + 3x_4 - 20 \rightarrow \max.$$

$$\begin{aligned} 7) \quad & 4x_1 - 2x_2 + x_3 = 28, \\ & x_1 + 3x_2 + x_4 = 21, \\ & x_i \geq 0, \quad i = 1, 2, 3, 4, \end{aligned}$$

$$Z = 100 - 7x_1 - 2x_2 - x_3 - x_4 \rightarrow \min.$$

$$\begin{aligned} 8) \quad & 5x_1 + x_2 + x_4 = 33, \\ & 2x_1 + 4x_2 + x_3 = 24, \\ & x_i \geq 0, \quad i = 1, 2, 3, 4, \end{aligned}$$

$$Z = 9x_1 + 6x_2 + x_3 + x_4 - 50 \rightarrow \max.$$

3.2. Розв'язати симплексним методом зі штучним базисом задачі ЛП:

$$\begin{aligned}
1) \quad & 4x_1 + 2x_2 + x_3 = 24, \\
& 2x_1 + x_2 - x_4 = 4, \\
& x_1 + 5x_2 + x_5 = 33, \\
& x_i \geq 0, \quad i = 1, 2, \dots, 5,
\end{aligned}$$

$$Z = x_1 + 2x_2 + 30 \rightarrow \max.$$

$$\begin{aligned}
3) \quad & x_1 + 2x_2 - x_3 = 4, \\
& 3x_1 + x_2 + x_4 = 22, \\
& x_1 + 2x_2 + x_5 = 14, \\
& x_i \geq 0, \quad i = 1, 2, \dots, 5,
\end{aligned}$$

$$Z = 3x_1 + 2x_2 + 14 \rightarrow \max.$$

$$\begin{aligned}
5) \quad & 2x_1 + x_2 + x_3 = 20, \\
& 5x_1 - 3x_2 + x_4 = 17, \\
& x_1 + 3x_2 - x_5 = 6, \\
& x_i \geq 0, \quad i = 1, 2, \dots, 5,
\end{aligned}$$

$$Z = x_1 + 3x_2 + 35 \rightarrow \max.$$

$$\begin{aligned}
2) \quad & 2x_1 + x_2 = 6, \\
& x_1 - 3x_2 = 8, \\
& 3x_1 - 2x_2 = 10, \\
& x_1 \geq 0, \quad x_2 \geq 0,
\end{aligned}$$

$$Z = 2x_1 + x_2 \rightarrow \max.$$

$$\begin{aligned}
4) \quad & 3x_1 - x_2 = 7, \\
& x_1 + 2x_2 = 5, \\
& 4x_1 + x_2 = 10, \\
& x_1 \geq 0, \quad x_2 \geq 0,
\end{aligned}$$

$$Z = x_1 + 4x_2 - 3 \rightarrow \min.$$

$$\begin{aligned}
6) \quad & 5x_1 + 3x_2 = 10, \\
& 2x_1 - x_2 = 6, \\
& 3x_1 + 4x_2 = 2, \\
& x_1 \geq 0, \quad x_2 \geq 0,
\end{aligned}$$

$$Z = 4x_1 - 3x_2 + 8 \rightarrow \min.$$

3.3. Скласти двоїсті задачі до наступних вихідних та показати взаємну спряженість відповідних пар задач:

$$\begin{aligned}
1) \quad & 3x_1 + x_2 - x_3 \leq 6, \\
& x_1 - 2x_2 + 4x_3 \leq 10,
\end{aligned}$$

$$x_i \geq 0, \quad i = 1, 2, 3,$$

$$Z = 5x_1 - x_2 + 6x_3 \rightarrow \max.$$

$$\begin{aligned}
2) \quad & 4x_1 - x_2 + x_3 - x_4 \leq 8, \\
& 2x_1 - x_3 + 2x_4 \geq 4, \\
& x_1 + x_2 - x_3 \leq 9,
\end{aligned}$$

$$x_i \geq 0, \quad i = 1, 2, 3, 4,$$

$$Z = 3x_1 + x_2 - 4x_4 \rightarrow \max.$$

$$\begin{aligned}
3) \quad & -5x_1 - x_2 + x_3 \leq 7, \\
& 2x_1 - 4x_2 - x_3 \leq 5,
\end{aligned}$$

$$x_i \geq 0, \quad i = 1, 2, 3,$$

$$Z = 3x_1 + x_2 - x_3 \rightarrow \min.$$

$$\begin{aligned}
4) \quad & 7x_1 + x_2 - x_3 \geq 10, \\
& 4x_1 - x_2 + x_3 \leq 3, \\
& 3x_1 + 5x_2 - 2x_3 \geq 2,
\end{aligned}$$

$$x_i \geq 0, \quad i = 1, 2, 3,$$

$$Z = 4x_1 - x_2 - 2x_3 \rightarrow \min.$$

3.4. Розв'язати наступні задачі ЛП, використовуючи дані симплексних таблиць, складених при розв'язанні двоїстих задач

1) $-x_1 + 3x_2 + x_3 \geq 1,$
 $-3x_1 + 2x_2 + 6x_3 \geq 2$

$$x_i \geq 0, \quad i = 1, 2, 3,$$

$$Z = -3x_1 + 28x_2 + 52x_3 \rightarrow \min.$$

2) $x_1 + 2x_2 \leq 17,$

$$3x_1 - x_2 \leq 16,$$

$$2x_1 + x_2 \geq 5,$$

$$x_1 \geq 0, \quad x_2 \geq 0,$$

$$Z = 4x_1 + 3x_2 \rightarrow \max.$$

3) $-x_1 + 2x_2 + x_3 \geq 2,$
 $-2x_1 - x_2 + 3x_3 \geq 1,$

$$x_i \geq 0, \quad i = 1, 2, 3,$$

$$Z = -4x_1 + 20x_2 + 30x_3 \rightarrow \min.$$

4) $2x_1 + x_2 \leq 20,$

$$-5x_1 + 3x_2 \leq 16,$$

$$x_1 + 2x_2 \geq 4,$$

$$x_1 \geq 0, \quad x_2 \geq 0,$$

$$Z = x_1 + 3x_2 \rightarrow \max.$$

5) $4x_1 - x_2 - x_3 \geq 2,$
 $x_1 - x_2 + 2x_3 \geq 3,$

$$x_i \geq 0, \quad i = 1, 2, 3,$$

$$Z = 45x_1 - 3x_2 + 12x_3 \rightarrow \min.$$

6) $x_1 + 2x_2 \geq 6,$

$$3x_1 + x_2 \leq 22,$$

$$x_1 + 2x_2 \leq 14,$$

$$x_1 \geq 0, \quad x_2 \geq 0,$$

$$Z = 3x_1 + 2x_2 \rightarrow \max.$$

ГЛАВА 4.

МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ТРАНСПОРТНИХ ЗАДАЧ ЛІНІЙНОГО ПРОГРАМУВАННЯ

4.1 Загальні зауваження

Як за допомогою розглянутого вище симплексного методу, в тому числі зі штучним базисом, так і двоїстого симплексного методу можна розв'язувати будь-які задачі лінійного програмування. Разом з тим існують класи задач ЛП, для яких створенні більш прості методи, які враховують особливості цих задач. Один з таких класів складають так звані транспортні задачі (ТЗ) або задачі визначення оптимального плану перевезень вантажу із заданих пунктів відправлення до заданих пунктів призначення.

Зауважимо, що перша строга постановка ТЗ належить Хічкоку, а перший точний метод її розв'язування – радянським математикам Л.В. Канторовичу та М.К. Гавуріну.

Вище (§ 2.1) була сформульована транспортна задача за критерієм вартості. Розглянемо її детальніше, причому в загальній постановці.

Є деяка кількість, наприклад m , пунктів відправлення, в яких розміщено однорідний вантаж в кількості a_1, a_2, \dots, a_m одиниць. Вантаж необхідно доставити в n пунктів призначення в кількості відповідно b_1, b_2, \dots, b_n одиниць. Крім того, відомі тарифи c_{ij} ($i = 1, 2, \dots, m; j = 1, 2, \dots, n$) – вартість доставки одиниці товару із i -го пункту відправлення до j -го пункту призначення.

Припустимо, що сумарна кількість вантажу, який знаходиться в пунктах відправлення, дорівнює сумарній кількості, яку необхідно доставити споживачам

$$\sum_{i=1}^m a_i = \sum_{j=1}^n b_j. \quad (4.1.1)$$

Необхідно скласти план перевезень вантажу так, щоб споживачі одержали необхідну кількість вантажу і сумарна вартість всіх перевезень була мінімальною.

Позначимо через x_{ij} – кількість вантажу, який треба перевезти із i -го пункту відправлення до j -го пункту призначення.

Математичне формулювання задачі. Необхідно знайти вектор, $\vec{X} = \{x_{11}, x_{12}, \dots, x_{mn}\}$, який задовольняє умовам:

$$\left\{ \begin{array}{l} \sum_{j=1}^n x_{1j} = a_1, \\ \sum_{j=1}^n x_{2j} = a_2, \\ \dots\dots\dots \\ \sum_{j=1}^n x_{mj} = a_m, \end{array} \right. \quad (4.1.2) \quad \left\{ \begin{array}{l} \sum_{i=1}^m x_{i1} = b_1, \\ \sum_{i=1}^m x_{i2} = b_2, \\ \dots\dots\dots \\ \sum_{i=1}^m x_{in} = b_n, \end{array} \right. \quad (4.1.3)$$

$$x_{ij} \geq 0, \quad i = 1, 2, \dots, m, \quad j = 1, 2, \dots, n \quad (4.1.4)$$

та забезпечує мінімум функції

$$Z = \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} \rightarrow \min. \quad (4.1.5)$$

Умови (4.1.2) означають, що вантаж повинен повністю бути вивезеним із усіх пунктів відправлення, смисл рівнянь (4.1.3) полягає в тому, що замовлення всіх споживачів на доставку вантажу повинні бути виконаними повністю. Вказані умови називаються балансовими.

Модель транспортної задачі (ТЗ), для якої сумарна кількість вантажу, що знаходиться в пунктах відправлення, дорівнює сумарній кількості вантажу, який необхідно доставити споживачам, тобто математична модель, в якій виконується умова (4.1.1) називається

закритою моделлю (або моделлю з правильним балансом). Існує також відкрита модель, або модель з неправильним балансом, для якої не виконується умова (4.1.1), тобто

$$\sum_{i=1}^m a_i \neq \sum_{j=1}^n b_j.$$

Розглянута задача називається ТЗ за критерієм вартості, оскільки в ній шукається план перевезень, який мінімізує їх сумарну вартість. Задачі, в яких мінімізуються сумарні витрати часу на перевезення вантажу із пунктів відправлення до пунктів призначення, називається ТЗ за критерієм часу.

Поряд із задачами по перевезенню однорідного вантажу на практиці зустрічаються ТЗ по перевезенню неоднорідного вантажу, а також задачі, в яких виключаються (з різних причин) доставки із деяких пунктів відправлення до деяких пунктів призначення (ТЗ з блокуванням перевезень). Крім того, часто виникає необхідність сумісного планування виробництва та перевезень вантажу, так звані виробничо-транспортні задачі. В таких задачах визначається, де і скільки необхідно виробляти продукції, куди і в якій кількості доставляти її, причому необхідно, щоб сумарні витрати на виробництво та перевезення (не тільки на перевезення) були мінімальними. Також зустрічається низка інших транспортних задач (детальніше про це див. [7]).

Предметом вивчення є класична ТЗ, яка задається математичною моделлю (4.1.1) – (4.1.5). У подальшому будуть розглянуті також ТЗ з неправильним (порушеним) балансом.

Зауважимо, що із $m + n$ рівнянь (4.1.2) та (4.1.3), яким повинні задовольняти невідомі величини x_{ij} , лінійно незалежними є $m + n - 1$, оскільки умови (4.1.2) та (4.1.3) пов'язані однією лінійною залежністю. Справді, якщо додамо всі рівняння (4.1.2) і окремо всі рівняння (4.1.3), то

одержимо згідно з (4.1.1) одне те саме число. Отже, ранг системи рівнянь (4.1.2) та (4.1.3) або, що те ж саме, системи векторів, координатами яких є коефіцієнти при невідомих, дорівнює $r = m + n - 1$. Таким чином, кількість базисних невідомих становить $m + n - 1$.

У загальному випадку ТЗ з правильним балансом зобразимо у вигляді таблиці 4.1.

Таблиця 4.1

$a_i \backslash b_j$	b_1	b_2	...	b_n
a_1	c_{11}	c_{12}	...	c_{1n}
a_2	c_{21}	c_{22}	...	c_{2n}
...
a_m	c_{m1}	c_{m2}	...	c_{mn}

Записавши в такому вигляді дані, легко побудувати вихідний опорний розв'язок.

4.2 Знаходження вихідного опорного розв'язку

Означення 1. Сукупність значень x_{ij} ($i = 1, 2, \dots, m; j = 1, 2, \dots, n$) називається допустимим розв'язком ТЗ, якщо вона задовольняє балансовим умовам (4.1.2) та (4.1.3), а також умові (4.1.4).

Означення 2. Допустимий розв'язок ТЗ називається опорним, якщо в ньому відмінні від нуля не більше ніж $r = m + n - 1$ невідомих.

Означення 3. Оптимальним розв'язком ТЗ називається допустимий розв'язок, який забезпечує найменшу сумарну вартість всіх перевезень.

Розв'язування ТЗ, як і будь-якої іншої задачі лінійного програмування, починається з визначення вихідного опорного розв'язку. Існує декілька способів для його знаходження. Розглянемо деякі з них.

4.2.1 Метод «північно-західного» кута (діагональний метод)

Викладення цього методу здійснимо на прикладі ТЗ, наведеної у таблиці 4.2. Заповнення (завантаження) таблиці починається з клітини, розташованої у верхньому лівому куті – «північно-західному» куті (звідси походить назва методу). В ній записується найменше із чисел a_1, b_1 , тобто $x_{11} = \min\{a_1; b_1\}$.

Таблиця 4. 2

$a_i \backslash b_j$	70	120	180	80
100	70 4	30 5	×	×
150	×	90 6	60 3	×
200	×	×	120 5	80 4

У наведеному прикладі $x_{11} = \min\{100; 70\} = 70$. В результаті закривається (подумки викреслюється) перший стовець, якщо замовлення першого споживача виконано повністю, або перший рядок, якщо вантаж повністю вивезено із першого пункту відправлення. Подальший рух здійснюється по відкритому рядку або стовпцю, зокрема, якщо $b_1 < a_1$ – по відкритому рядку. Змінна x_{12} вибирається із умови $x_{12} = \min\{a_1 - b_1; b_2\}$ і т.д. При цьому на кожному кроці закривається або

рядок або стовпець (випадок, коли одночасно закривається і рядок і стовпець, буде розглянуто окремо). У прикладі, що розглядається $x_{12} = \min\{100 - 70; 120\} = 30$. Перший рядок закривається.

Рухаємося по другому стовпцю, оскільки друге замовлення виконано лише частково. Величина x_{22} вибирається так $x_{22} = \min\{150; 120 - 30\} = 90$. Другий стовпець закривається (замовлення другого споживача виконано повністю).

Рух продовжується по другому рядку. Змінна x_{23} дорівнює $x_{23} = \min\{150 - 90; 180\} = 60$. На цьому кроці закривається другий рядок.

Оскільки замовлення третього споживача виконано не повністю, то рух здійснюємо по третьому стовпцю $x_{33} = \min\{200; 180 - 60\} = 120$. Третій стовпець закривається.

Рухаємося по третьому рядку, $x_{34} = \min\{200 - 120; 80\} = 80$. Закривається третій рядок і четвертий стовпець.

Неважко бачити, що сукупність знайдених величин x_{ij} ($i = 1, 2, 3$; $j = 1, 2, 3, 4$) є опорним розв'язком. По-перше, виконуються балансові умови (4.1.2) та (4.1.3), по-друге, всі величини x_{ij} – невід'ємні, по-третє, число заповнених (завантажених) клітин дорівнює $m + n - 1 = 3 + 4 - 1 = 6$.

Значення цільової функції, яке відповідає знайденому опорному розв'язку (сумарна вартість перевезень), становить

$$Z = 70 \cdot 4 + 30 \cdot 5 + 90 \cdot 6 + 60 \cdot 3 + 120 \cdot 5 + 80 \cdot 4 = 2070 \text{ (грн)}.$$

Зауваження. Бувають випадки, коли на деякому, не останньому, кроці одночасно закриваються і рядок і стовпець. У кожному з таких випадків необхідно записати нуль (базисний нуль) в одну з наступних клітин по рядку або стовпцю, а саме: в ту, у якій тариф менший. Якщо ж тарифи в обох вказаних клітинах рівні, то базисний нуль слід записати у будь-яку з них. Такий запис нуля означає, що відповідна базисна невідома

покладається рівною нулю. Якщо ж нуль не записати, то число базисних невідомих (заповнених клітин) буде менше ніж $m + n - 1$, що унеможливить знаходження оптимального розв'язку методами, які будуть викладені нижче.

Транспортна задача, в якій хоч одна базисна невідома дорівнює нулю, називається «виродженою», а відповідний опорний розв'язок – «виродженим».

Зазначимо, що випадки «виродження» можуть мати місце не тільки при визначенні вихідного опорного розв'язку, але також і в процесі оптимізації.

Приклад «виродженого» вихідного опорного розв'язку ТЗ наведено у таблиці 4.3.

Таблиця 4.3

$a_i \backslash b_j$	60	40	200	100
100	60 2	40 6	0 1	× 4
170	× 5	× 3	170 4	× 5
130	× 3	× 2	30 5	100 3

4.2.2 Метод мінімального елемента (найменшого тарифу, мінімальної вартості)

Існує декілька різновидів цього методу, які відрізняються послідовністю заповнення (завантаження) клітин з найменшим тарифом. Один з них розглянемо на прикладі, наведеному у таблиці 4.4.

Таблиця 4. 4

$a_i \backslash b_j$	70	120	180	80
100	×	×	100	×
150	×	×	80	70
200	70	120	×	10

Першою заповнюється клітина з найменшим тарифом у таблиці (якщо таких клітин декілька, – то будь-яка з них). В результаті закривається рядок або стовець. Рух здійснюється по відкритому стовпцю (рядку). Потім знову заповнюється клітина з найменшим тарифом і т. д. У наведеній задачі спочатку заповнюється клітина (3;1): $x_{31} = \min\{a_3; b_1\} = \min\{200; 70\} = 70$. Перший стовець закривається. Рух здійснюється по третьому рядку. Заповнюється клітина (3;2) з найменшим тарифом із числа тих, що залишились у цьому рядку: $x_{32} = \min\{200 - 70; 120\} = 120$. Другий стовець закривається. Рух здійснюємо по третьому рядку. Заповнюється клітина (3;4): $x_{34} = 10$. Третій рядок закривається. Рух продовжуємо по четвертому стовпцю. Найменший тариф у клітині (2;4). Змінна x_{24} вибирається рівною $x_{24} = \min\{150; 80 - 10\} = 70$. Четвертий стовець закривається. У незакритому другому рядку найменший тариф показано в клітині (2;1). Але ж її заповнювати не можна, оскільки перший стовець вже закритий, тобто замовлення першого споживача виконано повністю.

У другому рядку заповнюється клітина (2;3). Величина x_{23} дорівнює $x_{23} = \min\{150 - 70; 180\} = 80$. Потім заповнюється клітина (1;3): $x_{13} = 100$.

Легко бачити, що розв'язок, зображений у таблиці 4.4, є опорним. Кількість заповнених (базисних) клітин дорівнює $r = m + n - 1 = 3 + 4 - 1 = 6$. Значення цільової функції, що відповідає знайденому вихідному опорному розв'язку, становить

$$Z = 70 \cdot 1 + 120 \cdot 3 + 10 \cdot 4 + 70 \cdot 3 + 80 \cdot 3 + 100 \cdot 2 = 1120.$$

Варто зазначити, що у загальному випадку вихідний опорний розв'язок не є оптимальним. У подальшому повинні бути здійснені наступні етапи розв'язування ТЗ, які дозволять знайти оптимальний розв'язок.

Зауваження 1. При визначенні вихідного опорного розв'язку методом мінімального елемента також виникають випадки, коли на неостанньому кроці одночасно закривається і рядок і стовпець. У кожному з таких випадків згідно з міркуваннями, аналогічними викладеним при розгляді методу «північно-західного кута», записується базисний нуль в клітину з найменшим тарифом у рядку або стовпцю, а саме: в ту, в яку здійснювався б рух, якби рядок (стовпець) не закрився. Приклад «виродженого» вихідного опорного розв'язку транспортної задачі, знайденого розглянутим методом, наведено у таблиці 4.5.

Таблиця 4. 5

$a_i \backslash b_j$	60	40	80	70
80	0 2	× 4	10 5	70 2
100	60 2	40 1	× 3	× 4
70	× 4	× 2	70 6	× 3

Зауваження 2. Крім розглянутих, існують інші методи визначення вихідного опорного розв'язку. Зокрема, метод мінімуму по рядку, мінімуму по стовпцю.

4.3 Перетворення однократного заміщення.

Цикл перерахунку

У таблиці 4.6 зображено вихідний опорний розв'язок ТЗ (план перевезень вантажу), знайдений методом «північно-західного кута», при якому сумарна вартість перевезень становить

$$Z_1 = 40 \cdot 8 + 25 \cdot 6 + 35 \cdot 4 + 50 \cdot 9 + 20 \cdot 5 + 40 \cdot 4 + 90 \cdot 5 = 1770 \text{ (грн)}.$$

Таблиця 4. 6

$a_i \backslash b_j$	40	25	85	60	90
100	40 – 8	25 6	35 + 4	4	5
70	+ 2	7	50 – 9	20 5	6
130	4	1	3	40 4	90 5

Спробуємо покращити цей план перевезень. Для цього перенесемо, наприклад 40 одиниць вантажу із клітини (1;1) до клітини (2;1). Тоді, щоб не порушити балансових умов (4.1.2) та (4.1.3) необхідно перенести ті ж 40 од. із клітини (2;3) до клітини (1;3). В результаті одержимо новий опорний розв'язок (табл. 4.7). Сумарна вартість перевезень дорівнює

$$Z_2 = 256 + 754 + 402 + 109 + 205 + 404 + 905 = 1330 \text{ (грн)}.$$

Таблиця 4. 7

$a_i \backslash b_j$	40	25	85	60	90
100		8	6	4	5
70	40	2	7	9	6
130		4	1	3	5
				4	90

Diagram illustrating a cycle in the transportation table. The cycle consists of cells (2;1), (2;3), (1;3), and (1;1). The flow adjustments are: +20 at (2;1), -10 at (2;3), +75 at (1;3), and -25 at (1;1). The total flow adjustment is 40 units.

Порівнявши вартості перевезень, які відповідають попередньому та одержаному планам, переконуємось, що друга (Z_2) на 440 грн менша за першу (Z_1).

Отже, сумарна вартість перевезень зменшилась завдяки переміщенню 40 одиниць вантажу (без порушення балансових умов) по замкнутому циклу, який утворюють клітини (2;1), (2;3), (1;3), (1;1). Стрілками показано напрям обходу циклу. Базисна невідома x_{11} стала вільною, а вільна x_{21} стала базисною. Причому вказане однократне заміщення зв'язано з переміщенням (зсуненням) деякого вантажу по циклу. У зв'язку з цим детальніше розглянемо набори (сукупність) клітин, які називаються циклами.

Означення. Циклом у транспортній таблиці називається набір клітин, із яких тільки дві сусідні лежать в одному рядку або стовпцю, причому остання клітина знаходиться у тому ж рядку чи стовпцю, що й перша. Зауважимо, що замкнута ламана лінія, яка з'єднує клітини циклу, в кожній клітині здійснює поворот на 90^0 . Приклади циклів, які найчастіше зустрічаються, зображені на рисунку 4.1.

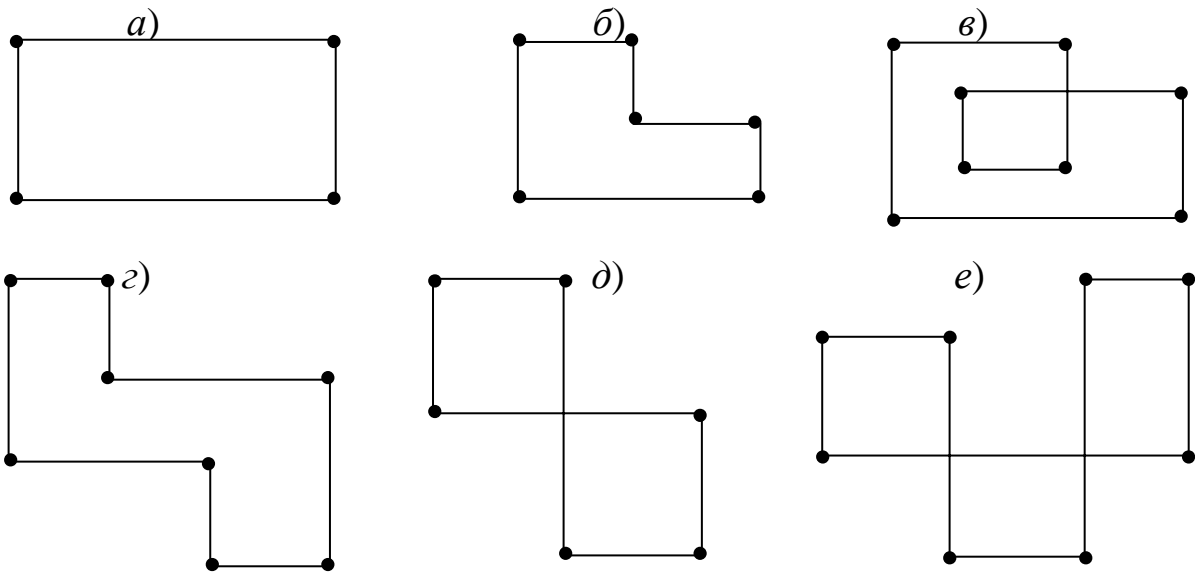


Рис. 4. 1.

Як правило, знаком «+» позначають ті вершини циклу, в яких збільшується обсяг перевезень, а знаком «-» – ті вершини, в яких вказаний обсяг зменшується.

Теорема 1 (без доведення). Неможливо утворити жодного циклу з вершинами тільки в базисних (завантажених) клітинах транспортної таблиці.

Теорема 2 (без доведення). Для кожної вільної (незавантаженої) клітини транспортної таблиці існує, причому єдиний, цикл, який містить цю клітину та деяке число базисних клітин.

Основні правила утворення циклів

1. До циклу повинна входити тільки одна вільна клітина, для якої він, власне, і утворюється, та деяка кількість базисних (завантажених) клітин.
2. Кожні дві сусідні клітини з'єднуються відрізком.
3. Цикл повинен зображатися ламаною лінією, відрізки якої перетинаються під прямим кутом.
4. У кожному стовпці або рядку, де є клітини циклу, повинні бути тільки дві вершини циклу.

Перетворення однократного заміщення в транспортній задачі, як і в будь-якій іншій задачі лінійного програмування, полягає в переведенні однієї із вільних невідомих до числа базисних та в вилученні відповідної невідомої із числа базисних. Розглянемо перетворення однократного заміщення у загальному випадку, який полягає в тому, що будь-яка вільна невідома, наприклад, x_{st} , транспортної задачі включається до числа базисних. Це означає, що клітина $(s; t)$ завантажується деяким вантажем в кількості λ одиниць.

Враховуючи, що при цьому балансові умови (4.1.2) та (4.1.3) повинні виконуватись, утворимо цикл з вихідною вершиною у вказаній клітині $(s; t)$. Клітини циклу у довільному напрямку пронумеруємо, присвоївши клітині $(s; t)$ перший номер. До чисел, записаних, у непарних клітинах циклу, додамо число λ , від чисел у парних клітинах віднімемо це ж число.

Залучення клітини $(s; t)$ до числа базисних (завантажених) повинно супроводжуватись вилученням однієї клітини із числа базисних (розвантаженням її). Очевидно, це можна зробити, вибравши число λ рівним найменшому із чисел, записаних у парних клітинах циклу. Не обмежуючи загальність міркувань, будемо вважати, що таким числом є значення базисної невідомої x_{kl} . Враховуючи сказане, вибір λ повинен здійснюватися у такий спосіб:

$$\lambda = \min_{\text{пар}} \{x_{ij}\} = x_{kl}. \quad (4.3.1)$$

В результаті переміщення числа λ по циклу вільна невідома x_{st} стане базисною ($x_{st} = \lambda$), а невідома x_{kl} вільною ($x_{kl} = 0$), і отже, одержимо новий опорний розв'язок транспортної задачі. Переміщення числа λ по циклу означає збільшення перевезень, вказаних у непарних вершинах циклу (вони позначені знаком «+»), на λ та зменшення перевезень, які стоять у

парних вершинах циклу (вони позначені знаком «—»), на ту ж кількість одиниць вантажу.

Компоненти нового опорного розв'язку, одержаного після зсуву розв'язку на величину λ , визначаються за формулою

$$x'_{ij} = \begin{cases} x_{ij} & - \text{ для базисних клітин, які не належать до циклу,} \\ x_{ij} + \lambda & - \text{ для парних клітин циклу,} \\ x_{ij} - \lambda & - \text{ для непарних клітин циклу.} \end{cases} \quad (4.3.2)$$

Зауваження 1. Якщо при однократному заміщенні одночасно розвантажуються декілька парних клітин циклу (в них записане одне й те ж число рівне розміру вантажу λ , що переміщується по циклу), то необхідно діяти так:

а) залишити вільною ту із вказаних клітин циклу, що розвантажуються, в якій тариф найбільший;

б) в решту клітин із числа розглянутих необхідно записати базисні нулі (це випадок виродженого розв'язку). Тоді число базисних клітин та відповідно – базисних невідомих залишиться рівним $m + n - 1$, у протилежному випадку – зменшиться.

Завантаження транспортної таблиці базисними нулями має на меті забезпечити можливість подальшого покращення розв'язку ТЗ, оскільки при числі базисних клітин меншому ніж $m + n - 1$, не для кожної вільної клітини можна побудувати цикл, і отже, не будь-яку вільну невідому можна перевести до числа базисних.

Зауваження 2. Якщо серед непарних клітин циклу є клітина, завантажена базисним нулем, то після зсуву розв'язку на число λ , в ній буде записане вказане число. Якщо ж клітина з базисним нулем є серед парних клітин циклу, то в цьому випадку по циклу переміщується нуль ($\lambda = 0$); приріст цільової функції при переході до нового опорного розв'язку,

як буде показано нижче, дорівнюватиме нулю, тобто покращення розв'язку не відбувається («холоста» ітерація).

Розглянемо, як зміниться значення цільової функції при здійсненні однократного заміщення в ТЗ. Неважко бачити, що в цьому випадку цільова функція збільшиться на суму добутків тарифів у непарних клітинах циклу на число λ (розмір зсуву) та зменшиться на суму добутків тарифів, записаних у парних клітинах циклу, на це ж число λ . Іншими словами, приріст цільової функції визначається виразом

$$\Delta Z = \sum_{\text{н}} c_{ij} \cdot \lambda - \sum_{\text{п}} c_{ij} \cdot \lambda \quad (4.3.3)$$

або

$$\Delta Z = \Delta_{st} \cdot \lambda, \quad (4.3.4)$$

де

$$\Delta_{st} = \sum_{\text{н}} c_{ij} - \sum_{\text{п}} c_{ij}. \quad (4.3.5)$$

Величину Δ_{st} будемо називати оцінкою вільної невідомої x_{st} або ціною циклу. Вона показує, на скільки зміниться цільова функція, якщо по циклу, складеному для вільної клітини $(s; t)$, перемістити одну одиницю вантажу.

Нагадаємо, що в транспортних задачах шукається розв'язок, який мінімізує цільову функцію. Отже, новий опорний розв'язок буде кращим за попередній, якщо $\Delta Z < 0$.

Оскільки $\lambda \geq 0$, то з (4.3.4) випливає, що необхідною умовою переходу до кращого опорного розв'язку з точки зору сумарної вартості перевезень є $\Delta_{st} < 0$.

Таким чином, якщо до числа базисних невідомих ввести вільну невідому, якій відповідає від'ємна оцінка, то новий опорний розв'язок, отриманий в результаті однократного заміщення, буде кращим за

попередній. На цій ідеї базуються різні методи визначення оптимальних розв'язків транспортних задач, зокрема так званий розподільний метод.

4.4 Алгоритм розв'язування ТЗ розподільним методом

1. Шукається вихідний опорний розв'язок транспортної задачі.

2. Для вільних клітин транспортної (розподільної) таблиці утворюються (будуються) цикли і за формулою (4.3.5) обчислюються значення оцінок відповідних вільних невідомих.

3. Якщо виявиться, що всі оцінки невід'ємні, то з цього випливає, що вихідний опорний розв'язок є оптимальним.

4. Якщо є хоча б одна від'ємна оцінка, наприклад $\Delta_{st} < 0$, то знаходиться розмір λ вантажу, який переміщується по циклу (здійснюється зміна перевезень по циклу), і як висновок, знаходиться новий, у загальному випадку кращий, опорний розв'язок ТЗ.

5. Здійснюється повернення до п.2. Процедура покращення (плану перевезень) продовжується до тих пір, поки буде знайдено оптимальний (найкращий) розв'язок. Ознака оптимальності – невід'ємність оцінок всіх вільних невідомих одержаного опорного розв'язку.

6. Визначається значення цільової функції, яке відповідає оптимальному розв'язку.

Здійснимо одну ітерацію однократного заміщення на прикладі ТЗ, зображеної у таблиці 4.7. Збудуємо цикл з вихідною вершиною в клітині (3; 2) (табл. 4.7).

Оцінка вільної невідомої x_{32} становить

$$\Delta_{32} = (1 + 5 + 4) - (6 + 9 + 4) = -9.$$

Розмір вантажу, який необхідно переміщувати по циклу, дорівнює

$$\lambda = \min\{25; 10; 40\} = 10.$$

Здійснивши переміщення λ по циклу, одержимо новий опорний розв'язок (табл. 4.8), який зменшує сумарну вартість перевезень на величину

$$\Delta Z = \Delta_{32} \cdot \lambda = -9 \cdot 10 = 90.$$

Таблиця 4. 8

$a_i \backslash b_j$	40	25	85	60	90
100	8	6	4	4	5
70	40	2	7	9	5
130	4	10	1	3	4

Цільова функція, що відповідає знайденому плану перевезень, становить

$$Z_0 = Z_2 + \Delta Z = 1330 - 90 = 1240(\text{грн}).$$

Варто вказати на суттєвий недолік розподільного методу, який полягає в необхідності утворення в одній таблиці великої кількості циклів, що відповідають різним вільним невідомим. Ця обставина створює суттєві незручності при розв'язуванні ТЗ цим методом. Вказаний недолік не притаманний методу потенціалів, який відрізняється від розподільного тільки способом обчислення оцінок вільних невідомих.

4.5 Розв'язування транспортних задач методом потенціалів

Із формули (4.3.5) видно, що оцінка вільних невідомих опорного розв'язку ТЗ обчислюється з використанням елементів матриці тарифів. Для розрахунку оцінок безпосередньо «в лоб» за цією формулою будуються цикли з вихідними вершинами у вільних клітинах транспортної таблиці, потім підсумовуються тарифи окремо у всіх непарних клітинах циклу і окремо – в парних, та із першої суми віднімається друга. Вказана процедура і незручна (багато циклів в одній таблиці) і досить трудомістка.

Підкреслимо одну дуже важливу особливість: у вираз (4.3.5) входить тільки один тариф вільної клітини – вихідної клітини циклу, решта тарифів відповідають базисним клітинам циклу.

Метод потенціалів, як було сказано вище, відрізняється від розподільного лише суттєвим спрощенням процедури обчислення оцінок вільних невідомих, спрощенням, яке базується на так званому еквівалентному перетворенні матриці тарифів з використанням названої вище особливості формули (4.3.5).

Теорема (без доведення). Якщо до всіх елементів будь-якого рядка або стовпця матриці тарифів додати одне й те ж дійсне число, то значення оцінок вільних невідомих опорного розв'язку, визначені з використанням нової матриці, залишаться незмінними, тобто такими ж, якими вони були при обчисленні їх з використанням вихідної матриці тарифів.

Іншими словами, твердження теореми означає: якщо значення оцінки Δ_{st} вільної невідомої x_{st} обчислити двома способами:

- а) з використанням матриці C з елементами C_{ij} , $i = 1, 2, \dots, m$, $j = 1, 2, \dots, n$;
- б) з використанням матриці C^{Π} з елементами

$$C_{ij}^{\Pi} = C_{ij} + u_i + v_j \quad (4.5.1)$$

де u_i , v_j – будь-які дійсні числа, то одержимо одне й те саме значення.

Матрицю C^{Π} назвемо матрицею псевдо-тарифів. Враховуючи, що числа u_i та v_j можуть бути довільними, виникає питання: чи можна їх вибрати так, щоб суттєво спростились розрахунки оцінок за формулою (4.3.5)?

Відповідь на поставлене запитання позитивна і полягає вона у наступному: розрахунок значень Δ_{st} буде значно простішим, якщо величини u_i та v_j вибрати так, щоб псевдо-тарифи, які відповідають базисним клітинам, були рівними нулю:

$$C_{ij}^{\Pi} = C_{ij} + u_i + v_j = 0 \quad (4.5.2)$$

для всіх базисних клітин.

Зазвичай числа u_i та v_j називають потенціалами (звідси походить назва методу).

Спочатку покажемо, що вибрати потенціали, виходячи з умови (4.5.2), можна, потім, що такий вибір забезпечує значне спрощення процедури обчислення оцінок вільних невідомих.

Кількість базисних клітин дорівнює $m + n - 1$, отже, стільки ж рівнянь (4.5.2). Кількість потенціалів складає $m + n$. Отже, число невідомих потенціалів на одиницю більше числа рівнянь, із яких вони визначаються. Тоді, якщо один потенціал, наприклад u_1 , вибрати рівним будь-якому дійсному числу (найпростіше, рівним нулю), то решта $m + n - 1$ потенціалів можуть бути однозначно визначені в результаті розв'язання системи рівнянь (4.5.2).

На прикладі вихідного опорного розв'язку ТЗ, зображеного в таблиці 4.9, покажемо, як безпосереднім підбором потенціалів можна забезпечити виконання умов (4.5.2).

Таблиця 4. 9

$a_i \backslash b_j$	60	30	40	70
80	60 6	20 5	4	2
70	2	$10 - \lambda$ 3	40 5	$20 + \lambda$ 4
50	4	$+\lambda$ 1	3	$50 - \lambda$ 6

Матриця тарифів зображеної транспортної задачі має вид

$$C = \begin{matrix} & & & & u_i \\ \begin{pmatrix} \underline{6} & \underline{5} & 4 & 2 \\ 2 & \underline{3} & \underline{5} & \underline{4} \\ 4 & 1 & 3 & \underline{6} \end{pmatrix} & u_1 \\ & & & & u_2 \\ & & & & u_3 \\ v_j & v_1 & v_2 & v_3 & v_4 \end{matrix} \quad (4.5.3)$$

В C підкреслено тарифи, які відповідають базисним клітинам. Матриця доповнена стовпцем потенціалів u_i , $i = 1, 2, 3$ та рядком потенціалів v_j , $j = 1, 2, 3, 4$. Покладемо потенціал першого рядка рівним нулю, тобто $u_1 = 0$ (див. (4.5.4)). Шість потенціалів: u_2 , u_3 , v_1 , v_2 , v_3 , v_4 необхідно вибрати так, щоб шість псевдо-тарифів C_{11}^{Π} , C_{12}^{Π} , C_{22}^{Π} , C_{23}^{Π} , C_{24}^{Π} , C_{34}^{Π} в матриці, яку одержимо в результаті перетворення, були рівними нулю.

Запишемо $C_{11}^{\Pi} = C_{11} + u_1 + v_1 = 0$. Враховуючи, що $C_{11} = 6$, $u_1 = 0$, маємо $6 + 0 + v_1 = 0$. Отже, потенціал елементів першого стовця дорівнює $v_1 = -6$ (крайній ліворуч елемент рядка v_j у (4.5.4) дорівнює -6).

$$C = \begin{pmatrix} \underline{6} & \underline{5} & 4 & 2 \\ 2 & \underline{3} & \underline{5} & \underline{4} \\ 4 & 1 & 3 & \underline{6} \end{pmatrix} \begin{matrix} u_i \\ 0 \\ 2 \\ 0 \end{matrix} \quad (4.5.4)$$

$$v_j \quad -6 \quad -5 \quad -7 \quad -6$$

Аналогічно $C_{12}^{\Pi} = C_{12} + u_1 + v_2 = 0$. Враховуючи, що $C_{12} = 5$, $u_1 = 0$ одержимо $5 + 0 + v_2 = 0$ або $v_2 = -5$. У рядку v_j потенціал елементів другого стовпця v_2 запишемо рівним -5 . Таким чином, із системи (4.5.2) вже використано два рівняння, що відповідають підкресленим тарифам першого рядка матриці C .

У другому стовпці матриці C після вибору потенціалу v_2 залишився ще один підкреслений тариф. Запишемо $C_{22}^{\Pi} = C_{22} + u_2 + v_2 = 0$ або $3 + u_2 - 5 = 0$. Звідси маємо $u_2 = 2$, тобто потенціал елементів другого рядка дорівнює 2. У другому рядку є ще два підкреслених тарифи. Отже,

$$C_{23}^{\Pi} = C_{23} + u_2 + v_3 = 0,$$

$$C_{24}^{\Pi} = C_{24} + u_2 + v_4 = 0$$

або

$$5 + 2 + v_3 = 0,$$

$$4 + 2 + v_4 = 0.$$

Звідси одержимо потенціали елементів третього та четвертого стовпців: $v_3 = -7$, $v_4 = -6$. У третьому стовпці підкреслених тарифів більше немає, у четвертому – є ще один. Запишемо для нього умову (4.5.2)

$$C_{34}^{\Pi} = C_{34} + u_3 + v_4 = 0 \text{ або } 6 + u_3 - 6 = 0. \text{ Звідси } u_3 = 0.$$

Таким чином, всі потенціали знайдені. Матриця псевдо-тарифів, елементи якої визначаються за формулою (4.5.1), має вид

$$C_0^{\Pi} = \begin{pmatrix} \underline{0} & \underline{0} & -3 & -4 \\ -2 & \underline{0} & \underline{0} & \underline{0} \\ -2 & \boxed{-4} & -4 & \underline{0} \end{pmatrix}. \quad (4.5.5)$$

Покажемо, що з використанням матриці псевдо-тарифів суттєво спрощується обчислення оцінок вільних невідомих опорного розв'язку ТЗ. Дійсно, за формулою (4.3.5) маємо

$$\Delta_{st} = \sum_{\text{H}} c_{ij}^{\Pi} - \sum_{\text{П}} c_{ij}^{\Pi}$$

або (згідно зі сформульованою теоремою)

$$\Delta_{st} = \sum_{\text{H}} c_{ij} - \sum_{\text{П}} c_{ij}. \quad (4.5.6)$$

У відповідності з (4.5.6) значення оцінки вільної невідомої x_{st} обчислюється з тарифи базисних клітин дорівнюють нулю (згідно з відповідним вибором потенціалів), в тому числі базисних клітин, що належать до циклу з початком у клітині $(s;t)$. Отже, маємо

$$\Delta_{st} = C_{st}^{\Pi}. \quad (4.5.7)$$

Співвідношення (4.5.7) означає, що оцінка вільної невідомої x_{st} дорівнює псевдо-тарифу C_{st}^{Π} , який відповідає вільній клітині $(s; t)$.

Таким чином, в результаті викладеного перетворення із матриці тарифів одержимо матрицю псевдо-тарифів, яка по своїй суті є матрицею оцінок вільних невідомих (всі не підкреслені елементи матриці C^{Π} – оцінки). У подальшому її так і будемо називати. Причому для обчислення оцінки будь-якої вільної невідомої не треба утворювати цикл з початком у відповідній вільній клітині. Друга перевага методу потенціалів у порівнянні з розподільним методом полягає в тому, що оцінки всіх вільних невідомих визначаються в результаті здійснення однієї, а не великого числа процедур.

Алгоритм розв'язування транспортних задач методом потенціалів

1. Будь-яким із відомих методів знаходиться вихідний опорний розв'язок транспортної задачі.

2. Записується матриця тарифів, у якій підкреслюються тарифи, що відповідають базисним невідомим опорного розв'язку.

3. Безпосереднім підбором знаходяться потенціали рядків та стовпців із умови: псевдо-тарифи, які відповідають базисним невідомим, дорівнюють нулю $C_{ij}^{\Pi} = C_{ij} + u_i + v_j = 0$ (для всіх базисних невідомих). При цьому один із потенціалів, наприклад u_1 , можна вибрати довільно, зокрема, для спрощення процедури прийняти рівним нулю.

4. Обчислюються елементи матриці псевдо-тарифів, яка є матрицею оцінок, за формулою (4.5.1)

$$C_{ij}^{\Pi} = C_{ij} + u_i + v_j.$$

Якщо виявиться, що оцінки всіх вільних невідомих – невід'ємні, то це означає, що знайдений опорний розв'язок є оптимальним.

5. Якщо ж серед оцінок є хоча б одна від'ємна, наприклад $\Delta_{st} < 0$, то знаходиться новий, кращий опорний розв'язок транспортної задачі. З цією метою будується цикл перерахунку з початковою вершиною у вільній клітині $(s; t)$, визначається розмір перерозподілу вантажу λ , здійснюється зсув λ по циклу. В результаті знаходиться новий опорний розв'язок з меншими сумарними витратами на перевезення.

6. Здійснюється повернення до п.2, тобто знову записується матриця тарифів, у якій підкреслюються тарифи, що відповідають опорному розв'язку і т. д.

Процедура покращення розв'язку продовжується до тих пір поки буде знайдено оптимальний розв'язок ТЗ. Нагадаємо ознаку

оптимальності опорного розв'язку: оцінки всіх вільних невідомих – невід'ємні.

Зауваження. Якщо є декілька від'ємних оцінок, то доцільно з них вибрати ту, абсолютна величина якої найбільша.

Розглянемо приклад розв'язування транспортної задачі, зображеної в таблиці 4.9, методом потенціалів. Вихідний опорний розв'язок цієї задачі, знайдений методом «північно-західного кута», запишемо в матричній формі*

$$X_0 = \begin{pmatrix} 60 & 20 & \bullet & \bullet \\ \bullet & 10 & 40 & 20 \\ \bullet & \bullet & \bullet & 50 \end{pmatrix} \quad (4.5.8)$$

Йому відповідає наступне значення цільової функції:

$$Z(X_0) = 60 \cdot 6 + 20 \cdot 5 + 10 \cdot 3 + 40 \cdot 5 + 20 \cdot 4 + 50 \cdot 6 = 1070 \text{ (грн)}.$$

Матриця оцінок вільних невідомих цього розв'язку має вид (4.5.5). Є від'ємні оцінки. Отже, вихідний опорний розв'язок не є оптимальним. Для введення до числа базисних виберемо вільну невідому x_{32} , якій відповідає оцінка $\Delta_{32} = -4$ (її взято в рамку). На основі початкового опорного розв'язку, зображеного в таблиці (4.9), згідно з правилами, утворимо цикл з початком у клітині (3.2).

Таблиця 4. 10

$a_i \backslash b_j$	60	30	40	70
80	60 6	20 5	4	2
70	2	10 - λ 3	40 5	20 + λ 4
50	4	+ λ 1	3	50 - λ 6

* *) Для наочності нульові значення вільних невідомих, в матриці опорного розв'язку, позначимо знаком „•”.

Кількість вантажу, який необхідно перемістити по циклу, становить $\lambda = \min\{50; 10\} = 10$. Після переміщення по циклу $\lambda = 10$ одержимо новий опорний розв'язок

$$X_1 = \begin{pmatrix} 60 & 20 & \cdot & \cdot \\ \cdot & 10 - \lambda & 40 & 20 + \lambda \\ \cdot & +\lambda & \cdot & 50 - \lambda \end{pmatrix} = \begin{pmatrix} 60 & 20 & \cdot & \cdot \\ \cdot & \cdot & 40 & 30 \\ \cdot & 10 & \cdot & 40 \end{pmatrix}, \quad (4.5.9)$$

якому відповідає значення цільової функції

$$Z(X_1) = Z(X_0) + \lambda \cdot \Delta_{32} = 1070 + 10 \cdot (-4) = 1030 \text{ (грн)}.$$

Знайдемо потенціали та обчислимо елементи матриці оцінок вільних невідомих розв'язку X_1

$$C = \begin{matrix} & & & & u_i \\ \begin{pmatrix} \underline{6} & \underline{5} & 4 & 2 \\ 2 & 3 & \underline{5} & \underline{4} \\ 4 & \underline{1} & 3 & \underline{6} \end{pmatrix} & & & & \begin{matrix} 0 \\ 6 \\ 4 \end{matrix} \\ v_j & -6 & -5 & -11 & -10 \end{matrix} \quad (4.5.10)$$

Послідовність визначення потенціалів: $u_1 = 0$, $v_1 = -6$, $v_2 = -5$, $u_3 = 4$, $v_4 = -10$, $u_2 = 6$, $v_3 = -11$.

Після обчислення одержимо матрицю псевдо-тарифів (оцінок):

$$C_1^{\Pi} = \begin{pmatrix} \underline{0} & \underline{0} & -7 & \boxed{-8} \\ 2 & 4 & \underline{0} & \underline{0} \\ 2 & \underline{0} & -4 & \underline{0} \end{pmatrix}. \quad (4.5.11)$$

Аналіз елементів матриці (4.5.11) показує, що є три від'ємні оцінки: $\Delta_{13} = -7$, $\Delta_{14} = -8$, $\Delta_{33} = -4$. Отже, розв'язок можна покращити. За вільну невідому, яку доцільно ввести в базис, виберемо x_{14} . Утворимо цикл з початком у клітині (1, 4) (табл. 4.11).

Таблиця 4. 11

b_j	60	30	40	70
a_i				

80	60	6	$20 - \lambda$	5	4	$+$	λ	2
70		2		3	40	5	30	4
50		4	$10 + \lambda$	1		3	$40 - \lambda$	6

Величина λ дорівнює $\lambda = \min\{40; 20\} = 20$.

Після переміщення λ одиниць вантажу по циклу одержимо новий опорний розв'язок.

$$X_2 = \begin{pmatrix} 60 & 20 - \lambda & \cdot & + \lambda \\ \cdot & \cdot & 40 & 30 \\ \cdot & 10 + \lambda & \cdot & 40 - \lambda \end{pmatrix} = \begin{pmatrix} 60 & \cdot & 20 \\ \cdot & \cdot & 40 & 30 \\ \cdot & 30 & \cdot & 20 \end{pmatrix}. \quad (4.5.12)$$

Йому відповідає значення цільової функції

$$Z(X_2) = Z(X_1) + \lambda \cdot \Delta_{14} = 1030 - 20 \cdot (-8) = 870 \text{ (грн)}.$$

Перевіримо, чи є знайдений опорний розв'язок оптимальним. Для цього знову виберемо потенціали

$$C = \begin{pmatrix} \underline{6} & 5 & 4 & \underline{2} \\ 2 & 3 & \underline{5} & \underline{4} \\ 4 & \underline{1} & 3 & \underline{6} \end{pmatrix} \begin{matrix} u_i \\ 0 \\ -2 \\ -4 \end{matrix} \quad (4.5.13)$$

$$v_j \quad -6 \quad 3 \quad -3 \quad -2$$

Послідовність підбору потенціалів: $u_1 = 0$, $v_1 = -6$, $v_4 = -2$, $u_2 = -2$, $u_3 = -4$, $v_3 = -3$, $v_2 = 3$.

Обчислимо елементи матриці оцінок вільних невідомих розв'язку X_2

$$C_2^{\Pi} = \begin{pmatrix} \underline{0} & 8 & 1 & \underline{0} \\ \underline{-6} & 4 & \underline{0} & \underline{0} \\ -6 & \underline{0} & -4 & \underline{0} \end{pmatrix} \quad (4.5.14)$$

Оскільки є від'ємні оцінки, то розв'язок X_2 можна покращити. Виберемо оцінку $\Delta_{21} = -6$. Для вільної клітини (2, 1) утворимо цикл (табл. 4.12)

Таблиця 4. 12

$a_i \backslash b_j$	60	30	40	70
80	$60 - \lambda$ 6	5	4	$20 + \lambda$ 2
70	$+\lambda$ 2	3	40 5	$30 - \lambda$ 4
50	4	30 1	3	20 6

Визначимо $\lambda = \min\{30; 60\} = 30$. Здійснимо переміщення $\lambda = 30$ по циклу і одержимо новий опорний розв'язок T_3

$$X_3 = \begin{pmatrix} 60 - \lambda & \cdot & 20 + \lambda \\ +\lambda & \cdot & 40 & 30 - \lambda \\ \cdot & 30 & \cdot & 20 \end{pmatrix} = \begin{pmatrix} 30 & \cdot & \cdot & 50 \\ 30 & \cdot & 40 & \cdot \\ \cdot & 30 & \cdot & 20 \end{pmatrix}. \quad (4.5.15)$$

Цільова функція, що відповідає цьому розв'язку, дорівнює

$$Z(X_3) = Z(X_2) + \lambda \cdot \Delta_{21} = 870 + 30 \cdot (-6) = 690 \text{ (грн)}.$$

Знову перевіримо, чи є розв'язок X_3 оптимальним

$$C = \begin{pmatrix} \underline{6} & 5 & 4 & \underline{2} \\ \underline{2} & 3 & \underline{5} & 4 \\ 4 & \underline{1} & 3 & \underline{6} \end{pmatrix} \begin{matrix} u_i \\ 0 \\ 4 \\ -4 \end{matrix}, \quad (4.5.16)$$

$$v_j \quad -6 \quad 3 \quad -9 \quad -2$$

$$C_3^{\Pi} = \begin{pmatrix} \underline{0} & 8 & -5 & \underline{0} \\ \underline{0} & 10 & \underline{0} & 6 \\ -6 & \underline{0} & \boxed{-10} & \underline{0} \end{pmatrix}. \quad (4.5.17)$$

Розв'язок X_3 не є оптимальним. Здійснимо ітерацію для покращення знайденого розв'язку. Вільну невідому x_{33} введемо до числа базисних. Для цього збудуємо цикл (табл. 4.13).

Таблиця 4. 13

$a_i \backslash b_j$	60	30	40	70
80	$30 - \lambda$	30	$40 - \lambda$	$50 + \lambda$
70	$30 + \lambda$	30	$40 - \lambda$	$50 + \lambda$
50	$30 + \lambda$	30	$40 - \lambda$	$50 + \lambda$

Визначимо $\lambda = \min\{20; 30; 40\} = 20$. Після переміщення $\lambda = 20$ одиниць вантажу по циклу опорний розв'язок має вид

$$X_4 = \begin{pmatrix} 30 - \lambda & \cdot & \cdot & 50 + \lambda \\ 30 + \lambda & \cdot & 40 - \lambda & \cdot \\ \cdot & 30 & +\lambda & 20 - \lambda \end{pmatrix} = \begin{pmatrix} 10 & \cdot & \cdot & 70 \\ 50 & \cdot & 20 & \cdot \\ \cdot & 30 & 20 & \cdot \end{pmatrix}. \quad (4.5.18)$$

Значення цільової функції дорівнює

$$Z(X_4) = Z(X_3) + \lambda \cdot \Delta_{33} = 690 - 20 \cdot 10 = 490 \text{ (грн)}.$$

Перевірка на оптимальність розв'язку X_4 :

$$C = \begin{pmatrix} \underline{6} & 5 & 4 & \underline{2} \\ \underline{2} & 3 & \underline{5} & 4 \\ 4 & \underline{1} & \underline{3} & 6 \end{pmatrix} \begin{matrix} u_i \\ 0 \\ 4, \\ 6 \end{matrix}, \quad (4.5.19)$$

$$v_j \quad -6 \quad -7 \quad -9 \quad -2$$

$$C_4^{\Pi} = \begin{pmatrix} \underline{0} & -2 & \boxed{-5} & \underline{0} \\ \underline{0} & \underline{0} & \underline{0} & 6 \\ 4 & \underline{0} & \underline{0} & 10 \end{pmatrix}. \quad (4.5.20)$$

Розв'язок X_4 не є оптимальним. Здійснимо наступну ітерацію. Цикл перерахунку має вид (табл. 4.14)

Таблиця 4. 14

$a_i \backslash b_j$	60	30	40	70
80	$10 - \lambda$ 6	5 5	$+$ λ 4	70 2
70	$50 + \lambda$ 2	3 3	$20 - \lambda$ 5	4 4
50	4 4	30 1	20 3	6 6

Після переміщення $\lambda = \min\{20; 10\} = 10$ одиниць вантажу по циклу одержимо опорний розв'язок

$$X_5 = \begin{pmatrix} 10 - \lambda & \cdot & + \lambda & 70 \\ 50 + \lambda & \cdot & 20 - \lambda & \cdot \\ \cdot & 30 & 20 & \cdot \end{pmatrix} = \begin{pmatrix} \cdot & \cdot & 10 & 70 \\ 60 & \cdot & 10 & \cdot \\ \cdot & 30 & 20 & \cdot \end{pmatrix} \quad (4.5.21)$$

$$Z(X_5) = Z(X_4) - \lambda \cdot \Delta_{13} = 490 - 5 \cdot 10 = 440 \text{ (грн).}$$

Перевірка розв'язку X_5 на оптимальність:

$$C = \begin{pmatrix} 6 & 5 & \underline{4} & \underline{2} \\ \underline{2} & 3 & \underline{5} & 4 \\ 4 & \underline{1} & \underline{3} & 6 \end{pmatrix} \begin{matrix} u_i \\ 0 \\ -1 \\ 1, \end{matrix} \quad (4.5.22)$$

$$v_j \quad -1 \quad -2 \quad -4 \quad -2$$

$$C_5^{\Pi} = \begin{pmatrix} 5 & 3 & \underline{0} & \underline{0} \\ \underline{0} & 0 & \underline{0} & 1 \\ 4 & \underline{0} & \underline{0} & 5 \end{pmatrix}. \quad (4.5.23)$$

Із матриці (4.5.23) видно, що оцінки всіх вільних невідомих невід'ємні. Отже, опорний розв'язок X_5 є оптимальним. Найменша сумарна вартість доставки вантажу становить $Z_{\min} = 440$ грн.

Зауваження. Ознакою альтернативного оптимуму транспортної задачі є рівність нулю однієї або кількох оцінок вільних невідомих знайденого оптимального опорного розв'язку.

Із (4.5.23) видно, що оцінка вільної невідомої x_{22} в оптимальному опорному розв'язку дорівнює нулю. Отже, розглянута ТЗ має альтернативний оптимум.

4.6 Математичні моделі транспортних задач з неправильним балансом

Вище розглядалися, так звані, закриті моделі ТЗ або моделі з правильним балансом – моделі задач, в яких сумарна кількість вантажу в пунктах відправлення дорівнювала сумарній кількості вантажу, який необхідно доставити в пункти призначення або коротко: сума запасів дорівнювала сумі заявок (виконувалась умова (4.1.1)). Разом з тим зустрічаються задачі, в яких вказана умова не виконується. В цих випадках кажуть про відкриті моделі ТЗ або про ТЗ з неправильним (порушеним) балансом.

Якщо виконується умова

$$\sum_{i=1}^m a_i > \sum_{j=1}^n b_j, \quad (4.6.1)$$

то ТЗ називається задачею з надлишком запасів (пропозицій), якщо ж

$$\sum_{i=1}^m a_i < \sum_{j=1}^n b_j \quad (4.6.2)$$

– задачею з надлишком заявок (попиту).

Моделі ТЗ з надлишком запасів

В m пунктах відправлення знаходиться однорідний вантаж в кількості відповідно a_1, a_2, \dots, a_m одиниць. Вказаний вантаж необхідно доставити в n пунктів призначення в кількості b_1, b_2, \dots, b_n одиниць відповідно. Причому виконується умова (4.6.1). Необхідно скласти такий план перевезень (x_{ij}) , при якому б всі заявки були виконані і сумарні затрати були мінімальними (найменшими)

$$Z = \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} \rightarrow \min. \quad (4.6.3)$$

Величини x_{ij} задовольняють умовам

$$\sum_{j=1}^n x_{ij} \leq a_i, \quad i = 1, 2, \dots, m, \quad (4.6.4)$$

$$\sum_{i=1}^m x_{ij} = b_j, \quad j = 1, 2, \dots, n, \quad (4.6.5)$$

$$x_{ij} \geq 0, \quad i = 1, 2, \dots, m, \\ j = 1, 2, \dots, n. \quad (4.6.6)$$

Задача, що розглядається, легко може бути зведена до задачі з правильним балансом шляхом введення фіктивного споживача з заявкою

$$b_{\phi} = \sum_{i=1}^m a_i - \sum_{j=1}^n b_j. \quad (4.6.7)$$

При цьому вартість доставки одиниці вантажу з будь-якого пункту відправлення до фіктивного пункту призначення покладаємо рівною нулю. У подальшому ця задача розв'язується методами, викладеними вище для задач із правильним балансом.

В цій задачі планування доставки вантажу із деякого пункту

відправлення до фіктивного пункту призначення, по суті, означає, що відповідна кількість вантажу ($x_{i\phi}$) залишається у вказаному пункту відправлення.

Моделі ТЗ з надлишком заявок (попиту)

В пунктах відправлення знаходиться однорідний вантаж в кількості a_1, a_2, \dots, a_m , який необхідно доставити в пункти призначення в кількості b_1, b_2, \dots, b_n . При цьому виконується умова (4.6.2). Необхідно скласти такий план перевезень вантажу, за якого б весь вантаж повністю був вивезений із пунктів відправлення і сумарні транспортні витрати були мінімальними.

Задача, що розглядається, зводиться до ТЗ з правильним балансом шляхом введення фіктивного постачальника з кількістю вантажу, який дорівнює

$$a_{\phi} = \sum_{j=1}^n b_j - \sum_{i=1}^m a_i, \quad (4.6.8)$$

та нульовою вартістю перевезень із фіктивного пункту відправлення до всіх пунктів призначення.

Планування доставки деякого вантажу від фіктивного постачальника до того або іншого пункту призначення, по суті, означає, що вказаний вантаж не буде доставлено відповідному замовнику (заявка не буде виконана частково або навіть повністю).

Приклад. Розв'язати ТЗ з надлишком заявок, умови якої надано в таблиці 4.15.

Таблиця 4. 15

b_j	80	120	70	30
a_i				
90	5	2	3	6

60		2		3		4		5
100		6		4		2		1

Розв'язання. Визначимо сумарні запаси вантажу та сумарну заявку на його доставку

$$\sum_{i=1}^3 a_i = 250 \text{ од.}, \quad \sum_{j=1}^4 b_j = 300 \text{ од.},$$

різниця між заявками та запасами становить

$$\sum_{j=1}^4 b_j - \sum_{i=1}^3 a_i = 50 \text{ од.} \quad (4.6.9)$$

Введемо фіктивний пункт відправлення з кількістю вантажу рівному $a_\phi = 50$ од., та нульовим тарифом (табл. 4.16)

Таблиця 4.16

$a_i \backslash b_j$	80	120	70	30
90	5	70	20	6
60	2	3	4	5
100	6	4	2	1
50	0	0	0	0

Вихідний опорний розв'язок задачі, що розглядається, знайдено методом мінімального елемента і зображено в таблиці 4.16, тобто

$$X_0 = \begin{pmatrix} \bullet & 70 & 20 & \bullet \\ 60 & \bullet & \bullet & \bullet \\ 20 & \bullet & 50 & 30 \\ \bullet & 50 & \bullet & \bullet \end{pmatrix}.$$

Йому відповідає значення цільової функції $Z(X_0) = 570$ грн.

Послідовне покращення плану перевезень надано нижче

$$C = \begin{pmatrix} 5 & \underline{2} & \underline{3} & 6 \\ \underline{2} & 3 & 4 & 5 \\ \underline{6} & 4 & \underline{2} & \underline{1} \\ 0 & \underline{0} & 0 & 0 \end{pmatrix} \begin{matrix} u_i \\ 0 \\ 5 \\ 1 \\ 2 \end{matrix}, \quad C_0^{\Pi} = \begin{pmatrix} -2 & \underline{0} & \underline{0} & 6 \\ \underline{0} & 6 & 6 & 8 \\ \underline{0} & 3 & \underline{0} & \underline{0} \\ \boxed{-5} & \underline{0} & -1 & 0 \end{pmatrix}, \quad (4.6.10)$$

$$v_j = -7 \quad -2 \quad -3 \quad -2$$

Таблиця 4. 17

$a_i \backslash b_j$	80	120	70	30	
90		5	2	3	6
60	60	2	3	4	5
100	20 - λ	6	4	2	1
50	+ λ	0	0	0	0

Diagram details: A closed loop is formed by four cells: (row 1, col 2) with value 5, (row 1, col 3) with value 2, (row 3, col 3) with value 4, and (row 3, col 2) with value 6. The cells (row 1, col 3) and (row 3, col 3) contain dots. The cells (row 1, col 2) and (row 3, col 2) contain the expressions $70 + \lambda$ and $50 - \lambda$ respectively. The cell (row 3, col 3) also contains the expression $20 - \lambda$. The cell (row 3, col 2) contains the expression $50 + \lambda$. The cell (row 1, col 2) contains the expression $20 - \lambda$.

$$\lambda = \min\{20; 50\} = 20,$$

$$X_1 = \begin{pmatrix} \cdot & 70+\lambda & 20-\lambda & \cdot \\ 60 & \cdot & \cdot & \cdot \\ 20-\lambda & \cdot & 50+\lambda & 30 \\ +\lambda & 50-\lambda & \cdot & \cdot \end{pmatrix} = \begin{pmatrix} \cdot & 90 & 0 & \cdot \\ 60 & \cdot & \cdot & \cdot \\ \cdot & \cdot & 70 & 30 \\ 20 & 30 & \cdot & \cdot \end{pmatrix} \quad (4.6.11)$$

$$Z(X_1) = Z(X_0) - \lambda \cdot \Delta_{41} = 570 + 20 \cdot (-5) = 470 \text{ (грн)},$$

$$C = \begin{pmatrix} 5 & \underline{2} & \underline{3} & 6 \\ \underline{2} & 3 & 4 & 5 \\ 6 & 4 & \underline{2} & \underline{1} \\ \underline{0} & \underline{0} & 0 & 0 \end{pmatrix} \begin{matrix} u_i \\ 0 \\ 0 \\ -1 \\ 2 \end{matrix} \quad C_1^{\Pi} = \begin{pmatrix} 3 & \underline{0} & \underline{0} & 6 \\ \underline{0} & 1 & 1 & 5 \\ 3 & 1 & \underline{0} & \underline{0} \\ \underline{0} & \underline{0} & \boxed{-1} & 2 \end{pmatrix} \quad (4.6.12)$$

$$v_j \quad -2 \quad -2 \quad -3 \quad 0$$

Таблиця 4. 18

$a_i \backslash b_j$	80	120	70	30	
90		5	2	3	6
60	60	2	3	4	5
100		6	4	2	1
50	20	0	0	0	0

$90 + \lambda$ (above 5), $0 - \lambda$ (above 2), $30 - \lambda$ (above 0), $+\lambda$ (above 0)

$$\lambda = \min\{0; 30\} = 0,$$

$$X_2 = \begin{pmatrix} \cdot & 90+\lambda & 0-\lambda & \cdot \\ 60 & \cdot & \cdot & \cdot \\ \cdot & \cdot & 70 & 30 \\ 20 & 30-\lambda & +\lambda & \cdot \end{pmatrix} = \begin{pmatrix} \cdot & 90 & \cdot & \cdot \\ 60 & \cdot & \cdot & \cdot \\ \cdot & \cdot & 70 & 30 \\ 20 & 30 & 0 & \cdot \end{pmatrix}, \quad (4.6.13)$$

$$Z(X_2) = Z(X_1) - \lambda \cdot \Delta_{43} = 470 + 0 \cdot (-1) = 470 \text{ (грн)},$$

$$C = \begin{matrix} & & & & u_i \\ \begin{pmatrix} 5 & \underline{2} & 3 & 6 \\ \underline{2} & 3 & 4 & 5 \\ 6 & 4 & \underline{2} & \underline{1} \\ \underline{0} & \underline{0} & \underline{0} & \underline{0} \end{pmatrix} & \begin{matrix} 0 \\ 0 \\ 0 \\ 2 \end{matrix} & C_2^{\Pi} = & \begin{pmatrix} 3 & \underline{0} & 1 & 5 \\ \underline{0} & 1 & 2 & 4 \\ 4 & 2 & \underline{0} & \underline{0} \\ \underline{0} & \underline{0} & \underline{0} & 1 \end{pmatrix} & (4.6.14) \\ v_j & -2 & -2 & -2 & -1 \end{matrix}$$

Розв'язок (4.6.13) є оптимальним, оскільки оцінки всіх вільних невідомих цього розв'язку невід'ємні. У відповідності з цим планом всі 90 одиниць вантажу, що знаходиться в першому пункті відправлення, повинні бути перевезені до другого пункту призначення, а всі 60 одиниць вантажу, що знаходиться у другого постачальника, необхідно доставити першому споживачу; із третього пункту відправлення необхідно доставити 70 одиниць вантажу третьому споживачу та 30 одиниць – четвертому.

Заявка першого споживача не виконана на 20 одиниць вантажу, а другого на 30 одиниць. Мінімальні сумарні затрати на доставку вантажу становлять

$$Z_{\min} = 470 \text{ грн.}$$

Зауваження. У розглянутій задачі визначався найкращий план перевезень, виходячи з критерію мінімуму сумарних витрат на доставку вантажу, при цьому міркування «справедливості» або «несправедливості» виконання заявок не враховувались. Можна показати, що при інших постановках ТЗ, наприклад, з урахуванням вимоги про рівну частку виконання заявок або про розподіл вантажів з урахуванням важливості

кожного пункту доставки, задача також може бути зведена до ТЗ з правильним балансом.

Питання для самоконтролю

1. Сформулювати транспортну задачу лінійного програмування. Яка модель ТЗ називається закритою, а яка – відкритою?
2. Які існують способи визначення вихідного опорного розв'язку ТЗ?
3. В чому полягає сутність методу «північно-західного кута»?
4. В чому суть методу мінімального елемента?
5. Скільки базисних невідомих повинен мати опорний розв'язок ТЗ?
6. Як здійснюється однократне заміщення в транспортних задачах?
7. Що називається циклом транспортної задачі?
8. Що таке метод потенціалів розв'язування транспортних задач?
9. Що таке матриця псевдо-тарифів та як вона використовується при розв'язуванні ТЗ методом потенціалів?
10. За якої умови знайдений опорний план ТЗ буде оптимальним?
11. Що таке ТЗ з надлишком запасів та як вона зводиться до задачі з правильним балансом?
12. Що таке ТЗ з надлишком заявок (попиту) та як вона зводиться до задачі з правильним балансом?

Задачі до глави 4

4.1. Визначити вихідні опорні розв'язки:

- а) методом «північно-західного кута»;
- б) методом мінімального елемента,

а також обчислити значення цільової функції, що відповідають знайденим розв'язкам транспортних задач, умови яких зображені у наступних таблицях.

1)

$a_i \backslash b_j$	20	30	45	55
30	4	3	5	3
40	3	2	6	4
30	2	1	4	5
50	5	4	5	6

2)

$a_i \backslash b_j$	70	20	60	50
90	2	6	3	3
40	5	1	4	5
30	4	5	5	3
40	3	2	1	4

3)

$a_i \backslash b_j$	50	30	40	70
60	3	6	7	5
20	4	2	4	4
10	2	1	3	1
100	5	3	3	4

4)

$a_i \backslash b_j$	60	20	80	40
50	5	6	3	2
30	2	4	5	3
90	3	3	2	5
30	4	1	6	3

5)

$a_i \backslash b_j$	70	50	30	60
10	4	5	3	2
40	6	3	5	3
70	1	2	2	4
90	3	4	5	3

6)

$a_i \backslash b_j$	40	70	60	50
100	3	5	3	1
20	4	4	5	2
40	3	3	3	3
60	2	1	2	4

7)

$a_i \backslash b_j$	20	60	40	80
40	4	3	2	5
30	1	2	3	3
60	1	2	2	4
40	3	4	5	3
30	1	2	3	4

8)

$a_i \backslash b_j$	30	70	90	60
30	4	3	2	5
40	2	2	1	3
60	1	3	2	3
70	2	4	2	3
50	1	1	5	1

9)

b_j a_i	60	80	50	110
40	1	3	2	4
90	2	2	4	3
70	1	1	2	5
20	2	4	3	3
80	1	4	3	2

10)

b_j a_i	10	110	20	120
100	3	1	2	4
20	1	2	4	2
10	2	3	1	2
70	3	5	3	2
60	4	3	2	3

4.2. Розв'язати методом потенціалів транспортні задачі, умови яких зображені у наступних таблицях:

1)

b_j a_i	25	45	55	75
70	5	3	4	3
40	4	6	5	6
60	5	6	4	3
30	6	4	5	7

2)

b_j a_i	40	60	70	130
100	5	4	1	2
20	3	5	4	6
110	2	3	3	6
70	4	1	5	4

3)

$a_i \backslash b_j$	80	50	40	30
90	3	4	5	2
10	2	3	4	3
40	3	2	3	2
60	1	3	4	3

4)

$a_i \backslash b_j$	20	100	50	30
40	2	1	4	3
20	3	2	3	2
70	2	3	2	1
70	1	2	3	3

5)

$a_i \backslash b_j$	55	75	35	95
60	5	4	1	3
100	4	3	2	3
10	5	5	1	2
90	3	2	2	2

6)

$a_i \backslash b_j$	70	80	50	40
30	1	2	3	2
70	2	3	4	3
60	1	2	3	2
80	2	3	4	3

7)

$b_j \backslash a_i$	90	40	70	80
60	4	3	1	2
50	5	3	1	4
30	2	3	4	5
140	1	2	3	4

8)

$b_j \backslash a_i$	40	65	45	70
100	1	4	2	5
20	2	1	3	4
30	3	2	1	3
80	4	5	3	2

9)

$b_j \backslash a_i$	70	90	60	30
20	4	5	3	2
40	3	2	4	1
80	2	3	5	3
110	1	4	2	5

10)

$b_j \backslash a_i$	30	80	70	100
50	2	3	4	5
40	1	2	3	4
30	3	4	2	1
160	4	1	5	2

4.3. Розв'язати наступні транспортні задачі з неправильним балансом:

1)

$b_j \backslash a_i$	70	80	40	100
40	4	3	2	6
50	3	5	4	4
100	1	4	6	4
60	2	4	6	6

2)

$b_j \backslash a_i$	100	30	90	50
100	4	5	4	4
120	2	4	2	3
70	4	5	3	3
30	3	3	1	4

3)

$a_i \backslash b_j$	40	65	85	60
70	2	1	4	3
80	3	3	2	2
50	1	2	3	3
30	1	3	5	1

4)

$a_i \backslash b_j$	30	40	20	60
30	3	4	3	2
80	4	5	4	3
50	5	2	5	4
60	4	3	6	5

5)

$a_i \backslash b_j$	30	60	40	70
40	4	2	1	3
80	3	4	2	3
70	1	3	4	2
60	2	3	4	1

6)

$a_i \backslash b_j$	80	70	30	70
50	4	3	1	4
70	3	4	1	2
40	1	2	3	3
60	2	1	4	5

7)

$a_i \backslash b_j$	40	80	60	20
30	1	3	2	3
90	4	2	3	2
60	3	1	4	3
70	4	3	2	4

8)

$a_i \backslash b_j$	100	70	60	40
60	3	4	3	1
70	2	3	4	3
50	1	2	2	3
80	4	3	4	2

9)

$a_i \backslash b_j$	50	60	30	70
70	4	3	1	5
40	2	1	3	4
55	3	2	4	1
45	1	4	2	3

10)

$a_i \backslash b_j$	60	30	100	70
80	5	2	4	3
20	1	2	3	4
40	3	1	4	2
60	4	3	1	1

ГЛАВА 5.

МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ МАТРИЧНИХ ІГОР ТА ПРИЙНЯТТЯ ГОСПОДАРСЬКИХ РІШЕНЬ

5.1 Ввідні поняття та означення. Види ігор

В економіці, військовій справі та інших сферах людської діяльності часто виникають так звані конфліктні ситуації, в яких інтереси двох (або більше) сторін повністю або частково протилежні.

Предметом теорії ігор є аналіз конфліктних ситуацій математичними засобами. Мета – розробка рекомендацій з вибору раціональних дій учасників в умовах конфлікту, який багаторазово повторюється. При цьому під грою розуміємо математичну модель конфліктної ситуації. Гравцем будемо називати одного або групу учасників гри, які мають спільні інтереси, що не збігаються з інтересами інших груп учасників.

Приклади:

а) у грі приймають участь три людини, причому кожний грає тільки за себе, отже, у грі є три гравця;

б) у грі приймають участь шестеро людей, об'єднаних в групи по троє (гра три на три), в цьому випадку в математичній моделі гри розглядається два гравця;

в) при грі в футбол, хокей і т. ін., учасники гри розділені на дві команди, гравці кожної з команд мають одну мету, в моделі цієї гри слід розглядати двох гравців.

Варто відмітити, що противником одного з гравців (другим гравцем) може бути не реальний суб'єкт, який свідомо діє проти першого гравця, а фіктивний, який уособлює, наприклад, дії деяких стихійних сил або недостатню обізнаність першого гравця про умови, в яких він вимушений приймати свої рішення. В теорії ігор таким фіктивним гравцем прийнято

називати природу. В задачах з фіктивним гравцем, які мають місце в багатьох економічних, технічних та інших застосуваннях, теорію ігор можна розглядати як математичний апарат прийняття рішень в умовах так званої природної невизначеності. Це поняття у подальшому уточнимо.

Гра ведеться у відповідності з правилами, що визначають можливі дії кожної із сторін (гравців).

Коротка історична довідка. Перші дослідження деяких питань теорії ігор були здійснені французьким математиком Е. Борелем в двадцятих роках ХХ сторіччя. Проте виникнення теорії ігор як самостійної науки відносять до 1944 р. і пов'язують з виходом у світ монографії Дж. Неймана та О. Моргенштерна «Теорія ігор та економічна поведінка». У подальші роки теорія ігор бурхливо розвивалась і розвивається. Великий внесок у її розвиток внесли Д. Нейман, О. Моргенштерн, Д. Неш, А. Таккер, С. Шеплі, а також радянські вчені М. М. Воробйов та Ю. Б. Гермейер.

Стратегією в теорії ігор називається певний план дій (система поведінки) гравця, що показує, як повинен діяти гравець у всіх можливих випадках розвитку гри, у найпростішому випадку – це вибір однієї з альтернатив (можливих дій), що є у гравця.

Ходом (кроком) гравця називається вибір та здійснення їм однієї із передбачених правилами дій.

Правилами гри передбачаються різні виграші в залежності від наслідків гри. В теорії ігор досліджуються тільки ігри, в яких виграші виражаються кількістю, зокрема балами, очками, вартістю тощо. При цьому програш розглядається як від'ємний виграш.

Задача теорії ігор полягає в знаходженні розв'язку гри, тобто оптимальних стратегій для кожного з гравців та ціни гри.

Означення. Ціною гри називається виграш (програш), який відповідає оптимальним стратегіям гравців.

Велике розмаїття реальних конфліктних ситуацій обумовило створення теорії різних видів ігор. На сьогоднішній день ігри класифікуються за кількістю гравців, стратегій, ходів, характером взаємовідносин гравців, характером виграшів, видом функції виграшів і т. ін. Розглянемо деякі з них. В залежності від кількості гравців аналізуються ігри двох, трьох і т. д. гравців. В залежності від кількості стратегій ігри поділяються на скінченні та нескінченні.

Гра називається скінченною, якщо вона має скінченну кількість ходів, на кожному з яких здійснюється вибір із скінченної кількості стратегій (альтернатив). Інакше гра називається нескінченною, зокрема, якщо хоча б один з гравців має нескінченну кількість можливих стратегій. Що більше можливих стратегій, то важче знайти розв'язок гри.

В залежності від кількості ходів ігри діляться на одноходові та багатходові.

Гра називається одноходовою (однокроковою), якщо її результат досягається після того, як кожен з гравців здійснить по одному ходу. Прикладом багатходових ігор є ігри в шахи або шашки.

В залежності від характеру взаємовідносин ігри поділяються на безкоаліційні, кооперативні та коаліційні.

Означення. Гра називається безкоаліційною, якщо в ній гравці не можуть (не мають права) створювати коаліції, укладати угоди про спільні дії в процесі гри. Якщо гравці мають право створювати коаліції, укладати угоди про спільні дії, гра називається коаліційною. Кооперативною називається коаліційна гра з коаліціями, визначеними до початку гри.

В залежності від характеру виграшів ігри поділяються на ігри з нульовою та ненульовою сумою. Зокрема, у випадку двох гравців виграш

одного з них дорівнює програшу іншого. Саме такими є багато ігор, в які грають заради розваги. Крім того, багато економічних та воєнних конфліктних ситуацій моделюються іграми з нульовою сумою.

Означення. Гра з ненульовою сумою – це гра, в якій сума вигравів усіх гравців відмінна від нуля.

Приклад. Торгові взаємовідносини між сторонами, за яких всі країни, торгові партнери в результаті застосування своїх стратегій можуть мати додатний виграв.

В залежності від виду функції вигравів ігри діляться на матричні, біматричні, неперервні, опуклі і т. ін.

Означення. Матричною називається скінченна гра двох гравців з нульовою сумою, в якій виграв першого гравця задається в вигляді матриці, яка називається матрицею платежів. При цьому номери рядків матриці відповідають номерам стратегій першого гравця, номери стовпців – номерам стратегій другого гравця; елемент матриці, що знаходиться на перетині деякого рядка та стовпця показує виграв першого гравця у разі застосування ним відповідної стратегії; виграв другого гравця дорівнює програшу першого.

Приклад. Матриця платежів двох гравців А та В має такий вид (табл. 5.1):

Таблиця 5. 1

$A_i \backslash B_j$	B_1	B_2	B_3	B_4
A_1	a_{11}	a_{12}	a_{13}	a_{14}
A_2	a_{21}	a_{22}	a_{23}	a_{24}
A_3	a_{31}	a_{32}	a_{33}	a_{34}

У гравця А є три стратегії A_1, A_2, A_3 , у гравця В – чотири B_1, B_2, B_3, B_4 . Одноходова гра відбувається так: гравець А вибирає один з трьох рядків матриці платежів, наприклад i -тий, гравець В вибирає один з чотирьох стовпців, наприклад j -тий, тієї ж матриці, не знаючи результатів вибору гравця А. Тоді елемент a_{ij} матриці платежів визначає виграш гравця А, за умови, що він вибирає свою i -ту стратегію, а гравець В свою j -ту стратегію; значення a_{ij} дорівнює програшу гравця В.

Задача теорії ігор полягає в тому, щоб знайти в деякому розумінні найкращі способи, якщо вони існують, проведення гри для кожного із гравців.

Оскільки мета гравців у скінченній грі з нульовою сумою прямо протилежна, то таку гру часто називають антагоністичною.

Означення. Біматричною грою називається скінченна гра двох гравців з нульовою сумою, в якій виграш для кожного з гравців задається окремою матрицею, причому виграші в результаті застосування i -тої стратегії гравцем А та j -тої – гравцем В будуть вказані для гравця А в першій матриці (перетин i -го рядка та j -го стовпця), для гравця В відповідно у другій матриці.

Гра гравців називається неперервною, якщо стратегії гравців виражені числами із визначених відрізків, а функції виграшів гравців є неперервними в залежності від стратегії. Окремим випадком неперервних ігор є опуклі ігри, в яких функції виграшів є опуклими.

Означення. Диференціальною називається багатоходова гра, в якій ходи можуть здійснюватись неперервно, а на поведінку гравців накладаються певні умови, які записуються диференціальними рівняннями. Наприклад, гра типу погоні, в якій один гравець прагне досягти деякої області, другий не допустити його до вказаної області.

В залежності від інформації, яка є у гравців стосовно попередніх

ходів, ігри поділяються на ігри з повною та неповною інформацією. Прикладом гри з повною інформацією є гра в шахи або шашки.

В залежності від джерел невизначеності результати гри поділяються на комбінаторні, азартні, стратегічні.

Означення. Комбінаторною називається гра, непередбачуваність результату якої визначається головним чином комбінаторними причинами. Прикладом комбінаторної гри є шахи.

Означення. Азартною називається гра, результат (наслідок) якої є невизначеним тільки завдяки випадковим причинам. В умовах цих ігор бажаний результат не може гарантуватися ніякою майстерністю гравця. Приклади: гра в «орлянку» (герб – не герб), гра у гральні кості тощо.

Означення. Стратегічною називається гра, в якій визначеність наслідку обумовлюється стратегічними причинами – гравець не може знати, якої стратегії дотримується супротивник.

Варто зазначити, що існують інші види ігор, а також інші класифікації їх. Окремо виділимо ігри з природою.

Означення. Грою з природою називається гра з фіктивним гравцем. Іншими словами – це математична модель конфліктної ситуації, яка виникає при виборі рішень, наслідки яких залежать не тільки від того, хто здійснює вибір, але й від низки непередбачуваних випадкових обставин таких, як дощ, град, засуха, землетрус, зміна попиту, поява нових видів товарів та інших, зокрема, недостатньої обізнаності об'єкта, який приймає рішення.

У подальшому нами буде надана лише загальна уява про теорію ігор, детально буде розглянуто лише математичний апарат теорії математичних ігор двох гравців з нульовою сумою.

З урахуванням сказаного уточнимо означення гри.

Означення. Грою називається математична модель конфліктної ситуації, в якій визначені набори стратегій кожного гравця та матриця платежів. Необхідно визначити оптимальні стратегії гравців, вважаючи, що кожен з них діє найкращим для нього способом.

5.2 Приклади математичного моделювання конфліктних ситуацій

Розглянемо декілька прикладів математичного моделювання конфліктних ситуацій.

Приклад 1. Гра в монети («орлянку»). У ній беруть участь двоє гравців. Перший учасник накриває монету рукою. При цьому він може накрити монету, коли вона лежить гербом догори (перша стратегія) або номіналом (решкою) догори (друга стратегія). Ставити монету «на ребро» забороняється правилами. Другий гравець вгадує, яким боком монета повернута догори. Якщо він відгадав, то перший гравець сплачує йому, наприклад, одну копійку. Якщо ж не відгадав, то він сплачує першому гравцю 1 коп.

Очевидно, що інтереси обох учасників протилежні і полягають в тому, щоб одержати виграш за рахунок іншого. Отже, учасники є гравцями А і В у вказаному розумінні, а гра належить до класу матричних ігор двох гравців з нульовою сумою.

Другий гравець теж має дві стратегії: перша стратегія V_1 полягає в тому, що він називає «герб», тобто вважає, що накрита монета лежить гербом догори, друга стратегія V_2 – гравець називає «решка».

Якщо перший гравець застосує стратегію A_1 (накрив монету, яка лежить гербом догори), при цьому другий гравець також застосує свою першу стратегію V_1 , то в цьому випадку перший гравець програє 1 копійку (його виграш складе -1 копійку), якщо ж перший гравець застосує

стратегію A_1 , а другий – стратегію B_2 , то виграш гравця А складе 1 копійку і т. ін. Матриця виграшів гравця А (матриця платежів) має вид (табл. 5.2)

Таблиця 5. 2

$A_i \backslash B_j$	B_1 (герб)	B_2 (решка)
A_1 (герб)	-1	1
A_2 (решка)	1	-1

Приклад 2. Гра «два пальці». В грі приймають участь двоє гравців, які одночасно і незалежно показують один або два пальці. Виграш, наприклад в гривнях, дорівнює сумарній кількості пальців, показаних обома гравцями. Якщо сума парна, виграє перший учасник, а другий йому сплачує, якщо непарна – то навпаки.

Оскільки в цій конфліктній ситуації інтереси учасників протилежні, то кожного з них слід вважати гравцем, і отже математичною моделлю конфліктної ситуації буде матрична гра двох гравців з нульовою сумою. Стратегіями обох гравців, природно, є їх можливості показувати один або два пальця.

Матриця виграшів гравця А має вид (табл. 5.3)

Таблиця 5. 3

$A_i \backslash B_j$	B_1	B_2
A_1	2	-3

A_2	-3	4

Приклад 3. Оптимізація плану випуску продукції.

Підприємство харчування здійснює випуск двох видів швидкопсувних продуктів P_1 і P_2 . Якщо продукція реалізується в день її випуску, то підприємство одержує прибуток 0,5 грн, та 0,35 грн за одиницю продукції відповідно P_1 і P_2 . Якщо ж продукція не реалізується в день випуску, то її якість істотно погіршується і її можна реалізувати наступного дня тільки за цінами, які нижчі ніж в день випуску, більш того, – нижчі за собівартість продукції. В цьому випадку підприємство зазнає збитків 0,4 грн та 0,3 грн за одиницю продукції відповідно P_1 і P_2 .

Кількість продукції, що реалізується, залежить від стану погоди. Встановлено, що за гарної погоди в день випуску реалізується 1000 одиниць продукції P_1 та 8000 одиниць продукції P_2 ; у негоду – відповідно 6000 одиниць продукції P_1 та 1200 одиниць продукції P_2 . Необхідно скласти оптимальний план випуску продукції, виходячи з того, що прогнозу погоди немає.

Природно, що на підприємстві повинні планувати випуск продукції, прагнучи, по-перше, одержати найбільший прибуток, по-друге, врахувати найбільш несприятливий для нього стан погоди.

Сформульовану ситуацію можна моделювати як антагоністичну гру двох гравців: підприємства та природи. У підприємства (гравця А) є дві стратегії: здійснювати випуск продукції, розраховуючи на гарну погоду (стратегія A_1) та на погану погоду (стратегія A_2). Природа (гравець В)

також має дві стратегії: створити гарну (стратегія B_1) або погану (стратегія B_2) погоду.

Матриця прибутку підприємства має вид (табл. 5.4)

Таблиця 5.4

$A_i \backslash B_j$	B_1 (гарна погода)	B_2 (погана погода)
A_1	a_{11}	a_{12}
A_2	a_{21}	a_{22}

Елемент a_{ij} – це прибуток підприємства при i -тій стратегії підприємства та j -тій стратегії погоди.

Визначимо елементи матриці прибутку. Припустимо, що підприємство застосує свою першу стратегію, тобто, орієнтуючись на гарну погоду, випустить продукції Π_1 – 1000 одиниць, Π_2 – 8000 одиниць. Тоді, якщо природа також застосує свою першу стратегію, тобто погода буде гарною, то всю вироблену продукцію буде в той саме день реалізовано. Прибуток підприємства становитиме

$$a_{11} = 1000 \cdot 0,5 + 8000 \cdot 0,35 = 3300 \text{ (грн).}$$

Якщо ж підприємство вибере свою першу стратегію, а природа – другу, тобто погода буде поганою, тоді в день випуску буде реалізовано відповідно 1000 одиниць продукції Π_1 , та лише 1200 одиниць продукції Π_2 , а решта $8000 - 1200 = 6800$ одиниць цієї продукції реалізується на наступний день за цінами, при яких підприємство зазнає збитків. Сумарний прибуток підприємства в цьому випадку становитиме

$$a_{12} = 1000 \cdot 0,5 + 1200 \cdot 0,35 + (8000 - 1200) \cdot (-0,3) = -1120 \text{ (грн).}$$

Якщо підприємство застосує свою другу стратегію, тобто орієнтуючись на негоду, випустить продукції P_1 – 6000 одиниць, P_2 – 1200 одиниць, а природа вибере свою першу стратегію, тобто погода буде гарною, то в той же день підприємство реалізує тільки 1000 одиниць продукції P_1 із 6000 одиниць виготовлених та 1200 одиниць продукції P_2 . Залишок ($6000 - 1000 = 5000$) продукції P_1 буде реалізовано на другий день за зниженими цінами. Прибуток підприємства в цьому випадку дорівнюватиме

$$a_{21} = 1000 \cdot 0,5 + (6000 - 1000) \cdot (-0,4) + 1200 \cdot 0,35 = -1080 \text{ (грн)}.$$

Якщо підприємство застосує свою другу стратегію, а природа – також свою другу стратегію, то всю продукцію, виготовлену у розрахунку на погану погоду, буде реалізовано в перший день. Прибуток підприємства буде

$$a_{22} = 6000 \cdot 0,5 + 1200 \cdot 0,35 = 3420 \text{ (грн)}.$$

Матриця прибутку має вид (табл. 5.5)

Таблиця 5. 5

$A_i \backslash B_j$	B_1	B_2
A_1	3300	-1120
A_2	-1080	3420

Варто зауважити, що природа не є розумним противником, який має мету, протилежну меті підприємства, та максимально шкодить йому. Разом з тим ігровий підхід до розглянутої конфліктної ситуації дозволяє знайти розв'язок, виходячи з найбільш несприятливих для підприємства погодних умов. Якщо ж умови будуть не найгіршими для підприємства, то воно може збільшити прибуток шляхом покращення плану, знайденого ігровими методами.

Зауваження 1. Сформульована задача може бути розглянута без використання методів теорії ігор, а саме: шляхом врахування стану погоди в середньому, природно, за наявності відповідних статистичних даних.

Наведені приклади показують, як можна формалізувати конфліктні ситуації, створюючи для них математичні моделі матричних ігор. Нижче буде розглянуто методи розв'язування ігор вказаного класу.

Зауваження 2. В процесі моделювання реальних конфліктних ситуацій, особливо економічних, у вигляді гри виникає низка труднощів. Зокрема, з описом стратегій гри, а також з визначенням вигравів в залежності від стратегій, що застосовуються.

5.3 Розв'язування матричних ігор в чистих стратегіях

Деякі приклади розв'язування задач теорії ігор для двох гравців А та В з нульовою сумою розглянемо на наступному прикладі. Припустимо, що матриця платежів одноходової гри має вид (табл. 5.6)

Таблиця 5. 6

$A_i \backslash B_j$	B_1	B_2	B_3	B_4	min
A_1	2	5	4	6	2
A_2	6	7	5	7	5
A_3	1	1	2	6	1
A_4	1	4	3	2	1
max	6	7	5	7	

Вище було сказано, що мета теорії ігор полягає у визначенні найкращих (оптимальних) стратегій для кожного з гравців.

Питання: у якому розумінні стратегії оптимальні? Чи існують вони? Якщо так, то як їх знайти?

Означення. Оптимальною стратегією гравця в теорії ігор називається стратегія, застосування якої забезпечує йому максимально можливий гарантований виграш (або, що теж саме, мінімально можливий середній програш) за будь-яких можливих стратегій іншого гравця, виграш, який суперник ніяк не може зменшити.

В процесі пошуку оптимальних стратегій в теорії ігор керуються наступними припущеннями:

1. Кожний гравець має супротивника не менш досвідченого, ніж він сам. І такого, що має протилежну мету. Іншими словами не враховуються можливі прорахунки та похибки кожного з гравців.

2. Кожен гравець є обережним і прагне до середнього гарантованого результату (зокрема до виграшу, не меншому від деякої величини, або програшу, не більшому за щось), виключаються елементи азарту і ризику.

Означення. Чистою стратегією гравця називається стратегія, яка полягає у виборі того чи іншого рядка (стовпця) матриці платежів.

Якщо гравець А гри (табл. 5.6) вибирає свою першу стратегію, то мінімальний виграш його складе 2 одиниці, якщо другу стратегію, – 5 одиниць, при третій і четвертій стратегіях – 1 одиницю. Отже, гравцю А краще за все вибрати свою другу стратегію, для якої гарантований виграш складе 5 одиниць.

Якщо гравець В вибере свою першу стратегію, то його максимальний програш складе 6 одиниць, якщо другу стратегію, – 7 одиниць, при третій – 5 одиниць, при четвертій – 7 одиниць.

Таким чином, гравцю В вигідно вибрати свою третю стратегію, при якій він не програє більше за 5 одиниць. Висновок: гравцю А найвигідніше за все вибрати свою другу стратегію, гравцю В – свою третю стратегію.

Розглянемо загальний випадок. Нехай відома матриця платежів вимірністю $m \times n$ (табл. 5.7):

Таблиця 5.7

$A_i \backslash B_j$	B_1	B_2	...	B_n	α_i (мінімуми рядків)
A_1	a_{11}	a_{12}	...	a_{1n}	
A_2	a_{21}	a_{22}	...	a_{2n}	
...	
A_m	a_{m1}	a_{m2}	...	a_{mn}	
β_j (максимуми стовпців)					

Із матриці видно, що гравці А та В мають відповідно m та n чистих стратегій. При виборі пари $(i; j)$ стратегій (i -ої гравцем А та j -ої гравцем В) гравець В сплачує гравцю А в кінці гри a_{ij} одиниць.

Якщо гравець А вибере свою i -ту стратегію, то виграш його складе принаймні величину

$$\alpha_i = \min_j a_{ij} \tag{5.3.1}$$

Тут мінімум береться по всіх чистих стратегіях гравця В. Числа α_i записуються в крайній правий стовпець матриці платежів. Природно, що гравець А прагне вибрати свою стратегію так, щоб його, виграш $\min_j a_{ij}$ був, якомога більшим, тобто становив величину, не меншу ніж

$$\alpha = a_{i_0 j_0} = \max_i \min_j a_{ij}. \tag{5.3.2}$$

Тут максимум береться по всіх стратегіях гравця А.

Означення. Число α , що визначається за формулою (5.3.2), називається нижньою ціною гри або максиміном. Воно означає, що стратегія i_0 гравця А гарантує йому виграш, не менший ніж α за будь-якої стратегії гравця В.

Стратегія i_0 називається максимінною стратегією гравця А.

Гравець В прагне зробити якнайменшим виграш гравця А. Отже, він повинен проаналізувати всі свої чисті стратегії, виділивши в кожній з них максимальне значення виграшу гравця А

$$\beta_j = \max_i a_{ij}. \quad (5.3.3)$$

Потім знайти таку свою j -ту стратегію, за якої розмір його програшу буде найменшим серед усіх найбільших платежів, яких прагне досягти перший гравець, тобто гравець В повинен знайти

$$\beta = a_{i_1 j_1} = \min_j \max_i a_{ij}. \quad (5.3.4)$$

Числа β_j записуються в рядок, розміщений нижче матриці платежів.

Означення. Число β , що визначається за формулою (5.3.4), називається верхньою ціною гри або мінімаксом. Воно означає, що коли гравець В вибере свою мінімаксну стратегію j_1 , то її застосування дасть йому гарантію того, що виграш гравця А за будь-якої його стратегії буде, не більшим ніж β .

Підсумовуючи сказане, підкреслимо, що, вибираючи чисті стратегії у відповідності з принципом розумності, гравець А може забезпечити собі виграш, не менший ніж α , а гравець В може не допустити, щоб виграш гравця А був більшим ніж β .

Означення. Принцип, у відповідності з яким гравець А вибирає свою максимінну, а гравець В свою мінімаксну стратегію, називається

принципом мінімакса. Цей принцип впливає з припущення про розумність гравців в розумінні, про яке було сказано вище.

Означення. Гра, в якій нижня і верхня ціни гри збігаються ($\alpha = \beta$), називається грою з сідловою точкою в чистих стратегіях. Число

$$V = \alpha = \beta \quad (5.3.5)$$

називається чистою ціною гри.

Сідловою точкою називається пара чистих стратегій $(i_0; j_0)$, за яких виконується рівність $\alpha = \beta$. Зміст цієї точки у наступному: якщо один з гравців вибрав стратегію, що відповідає сідловій точці, то у іншого гравця немає кращого вибору ніж вибір своєї стратегії, яка відповідає цій же точці.

Для стратегій i_0, j_0 , що утворюють сідлову точку, має місце співвідношення

$$a_{ij_0} \leq a_{i_0j_0} \leq a_{i_0j} \quad (5.3.6)$$

де i, j – будь-які чисті стратегії гравців.

Із (5.3.6) видно, що елемент $a_{i_0j_0}$ є мінімальним в i_0 -му рядку, та максимальним в j_0 -му стовпці матриці платежів.

Сідлову точку шукають так: у матриці платежів послідовно в кожному рядку знаходять мінімальний елемент і при цьому перевіряють, чи є він одночасно максимальним у своєму стовпці. Якщо є, то знайдений елемент називають сідловим, а стратегії, що йому відповідають, визначають сідлову точку.

Означення. Чисті стратегії $i_0; j_0$, що визначають сідлову точку, називаються оптимальними чистими стратегіями. Оптимальні чисті стратегії, а також сідловий елемент $a_{i_0j_0}$ називаються розв'язком гри.

Зауваження. Гра може мати декілька сідлових точок.

Приклад. Гра задана платіжною матрицею (табл. 5.8):

Таблиця 5.8

$A_i \backslash B_j$	B_1	B_2	B_3	B_4	α_i
A_1	1	6	2	-4	-4
A_2	2	4	3	2	2
A_3	-2	8	-5	1	-5
β_j	2	8	3	2	

Необхідно знайти розв'язок гри.

Розглянемо послідовно рядки матриці та визначимо в них мінімальні елементи. У першому рядку мінімальне значення виграшу дорівнює $\alpha_1 = -4$, елемент $a_{14} = -4$ не є максимальним у своєму четвертому стовпці. Отже, він не може бути сідловим.

У другому рядку мінімальне значення виграшу гравця А становить $\alpha_2 = 2$, мінімальні елементи $a_{21} = 2$ та $a_{24} = 2$. Легко перевірити, що вони є максимальними відповідно у першому та четвертому стовпцях. Отже вказані елементи є сідловими і утворюють розв'язки:

- оптимальна чиста стратегія першого гравця А – друга; гравець В має дві оптимальні чисті стратегії: першу та четверту,
- ціна гри $V = 2$.

У третьому рядку мінімальне значення виграшу гравця А дорівнює -5 , але елемент $a_{33} = -5$ не є максимальним у третьому стовпці, і отже, він не є сідловим.

Таким чином, розглянута гра має дві сідлові точки у чистих стратегіях. Нижня та верхня ціни гри $\alpha = 2$ та $\beta = 2$ збігаються. Оптимальні чисті стратегії: (2; 1) та (2; 4).

Неважко бачити, що ігри, розглянуті у 5.2, не мають сідлових точок в чистих стратегіях.

5.4 Розв'язування матричних ігор в змішаних стратегіях.

Задача Неймана

Вище було сказано: якщо матриця платежів має сідлову точку, то розв'язування гри зводиться до визначення цієї точки. Проте в іграх, які мають практичне значення, найчастіше верхня та нижня ціни гри відрізняються. Виникає питання: як в цьому випадку знайти розв'язок гри?

Якщо гравець А вибере свою максимінну, а гравець В – свою мінімаксу стратегії, то в цьому випадку виграш гравця А буде не меншим за α одиниць, а програш гравця В – не більшим ніж β одиниць. Інтервал $[\alpha, \beta]$ є областю невизначеності, в межах якої гравець А може прагнути збільшити свій виграш, а гравець В – зменшити свій програш.

Якщо розглядати одноходову гру, то у гравця А немає можливості збільшити гарантований виграш. Проте при багаторазовому повторенні гри можна підвищити середній гарантований виграш гравця А за рахунок чергування чистих стратегій.

Означення. Стратегії, що полягають у випадковому чергуванні чистих стратегій, називаються змішаними.

Змішані стратегії відповідно гравців А та В у матричній грі $m \times n$ позначимо так:

$$\bar{X} = (x_1, x_2, \dots, x_m), \tag{5.4.1}$$

$$\bar{Y} = (y_1, y_2, \dots, y_n),$$

де x_i ($i = 1, 2, \dots, m$), y_j ($j = 1, 2, \dots, n$) – ймовірності застосування гравцями своїх чистих стратегій відповідно A_i та B_j .

Величини x_i, y_j задовольняють очевидним умовам

$$x_i \geq 0, y_j \geq 0, \tag{5.4.2}$$

$$\sum_{i=1}^m x_i = 1, \quad \sum_{j=1}^n y_j = 1.$$

Неважко бачити, що будь-яка чиста стратегія є окремим випадком змішаної стратегії, в якій всі ймовірності, крім однієї, дорівнюють нулю, а ймовірність, що залишилась, дорівнює одиниці. Уточнимо поняття розв'язку гри.

Означення. Розв'язком матричної гри двох гравців з нульовою сумою називається пара оптимальних стратегій (в загальному випадку змішаних), що характеризуються наступною властивістю: якщо один з гравців дотримується своєї оптимальної стратегії, то другому гравцю не вигідно відхилитись від своєї оптимальної стратегії.

Означення. Середнім виграшем гравця А називається математичне сподівання його виграшів, яке визначається виразом

$$\sum_{i=1}^m \sum_{j=1}^n a_{ij} x_i y_j, \tag{5.4.3}$$

де a_{ij} – елементи матриці платежів у грі.

Із (5.4.3) видно, що середній виграш першого гравця є функцією від x_i та y_j .

Означення. Середній виграш гравця А (середній програш гравця В), який відповідає оптимальному розв'язку, називається ціною гри V .

Сформулюємо основну теорему теорії ігор.

Теорема Неймана (без доведення). Кожна матрична гра двох гравців з нульовою сумою має хоча б один розв'язок.

Зауважимо, що в оптимальну змішану стратегію гравця можуть входити не всі його чисті стратегії.

Означення. Активними називаються чисті стратегії гравця, що входять до його оптимальної стратегії з ймовірностями відмінними від нуля.

При розв'язуванні ігор часто користуються наступною теоремою.

Теорема про активні стратегії (без доведення). Нехай один із гравців дотримується своєї оптимальної змішаної стратегії. При цьому дії іншого гравця знаходяться в межах його активних стратегій (він вибирає будь-яку з них або змішує їх у будь-яких пропорціях). Тоді середній виграш першого гравця залишається незмінним і рівним ціні гри.

Розглянемо гру 2×2 , задану матрицею виграшів гравця А (табл.5.9):

Таблиця 5. 9

B_j	B_1	B_2
A_i		
A_1	a_{11}	a_{12}
A_2	a_{21}	a_{22}

Припустимо, що вказана гра не має сідлової точки в чистих стратегіях (інакше розв'язок очевидний).

Оскільки згідно з теоремою Неймана ця гра має розв'язок, то сідлову точку необхідно шукати в області змішаних стратегій

$$\bar{X} = (x_1, x_2), \bar{Y} = (y_1, y_2).$$

Зауважимо, що в грі, яка розглядається, обидві стратегії кожного з гравців обов'язково активні (інакше гра мала б сідлову точку в чистих

стратегіях, що суперечить припущенню). Знайдемо розв'язок гри. Згідно з теоремою про активні стратегії середній виграш гравця А залишається незмінним за будь-яких стратегій гравця В (в тому числі за будь-якої чистої стратегії) та рівним ціні гри V . Тоді мають місце співвідношення

$$\begin{aligned} a_{11}x_1 + a_{21}x_2 &= V, \\ a_{12}x_1 + a_{22}x_2 &= V. \end{aligned} \tag{5.4.4}$$

Перше співвідношення – це середній виграш гравця А за використання ним своєї оптимальної стратегії та – першої чистої стратегії гравцем В, друге співвідношення – те ж саме за другої чистої стратегії гравця В.

Із (5.4.4) з урахуванням рівності

$$x_1 + x_2 = 1 \tag{5.4.5}$$

маємо

$$\begin{aligned} x_1 &= \frac{a_{22} - a_{21}}{a_{11} + a_{22} - a_{12} - a_{21}}, \\ x_2 = 1 - x_1 &= \frac{a_{11} - a_{12}}{a_{11} + a_{22} - a_{12} - a_{21}}. \end{aligned} \tag{5.4.6}$$

Після підстановки виразів (5.4.6) у (5.4.4) одержимо формулу для визначення ціни гри

$$V = \frac{a_{22}a_{11} - a_{12}a_{21}}{a_{11} + a_{22} - a_{12} - a_{21}}. \tag{5.4.7}$$

Аналогічно знайдемо оптимальну стратегію гравця В. Із рівностей

$$\begin{aligned} a_{11}y_1 + a_{12}y_2 &= V, \\ a_{21}y_1 + a_{22}y_2 &= V. \end{aligned} \tag{5.4.8}$$

з урахуванням

$$y_1 + y_2 = 1 \tag{5.4.9}$$

одержимо

$$y_1 = \frac{a_{22} - a_{12}}{a_{11} + a_{22} - a_{12} - a_{21}},$$

$$y_2 = 1 - y_1 = \frac{a_{11} - a_{21}}{a_{11} + a_{22} - a_{12} - a_{21}}.$$
(5.4.10)

Приклад. Знайти розв'язок гри «Оптимальний план випуску продукції» (табл. 5.5)

Неважко бачити, що ця гра не має сідлових точок в чистих стратегіях. А тому розв'язок знайдемо в області змішаних стратегій $\bar{X} = (x_1, x_2)$, $\bar{Y} = (y_1, y_2)$. Значення шуканих ймовірностей знайдемо за формулами (5.4.6) та (5.4.10). В результаті одержимо

$$x_1 = 0,504, x_2 = 0,496, y_1 = 0,509, y_2 = 0,491.$$

Ціна гри, визначена за формулою (5.4.6), становить $V = 1130$ грн.

У відповідності з одержаним розв'язком, гравцю А (підприємству) рекомендується у 50,4% всіх випадків (ймовірність 0,504) застосовувати свою першу чисту стратегію, тобто здійснювати випуск продукції, орієнтуючись на гарну погоду, і в 49,6% випадків (ймовірність 0,496) застосовувати свою другу чисту стратегію, орієнтуючись на погану погоду. При цьому середній щоденний прибуток підприємства становитиме 1130 грн. Природа максимально нашкодить підприємству, якщо у 50,9% випадків буде дотримуватись своєї першої чистої стратегії (погода буде гарною) і в 49,1% випадків буде вибирати свою другу чисту стратегію (погода буде поганою).

Нагадуємо, що перша чиста стратегія підприємства полягає в випуску 1000 одиниць продукції P_1 та 8000 одиниць продукції P_2 , друга стратегія – у випуску відповідно 6000 одиниць продукції P_1 та 1200 одиниць продукції P_2 . Оптимальний розв'язок гри з природою рекомендує підприємству випадково (з ймовірністю $\approx 0,5$) змінювати свої чисті стратегії.

Практична реалізація такої рекомендації може бути спряженою з різними труднощами в роботі підприємства зокрема пов'язаними з умовами доставки, зберігання та переробки сировини, що швидко псується. Враховуючи сказане, знайдений оптимальний розв'язок (змішану стратегію) краще використовувати так: щоденно планувати виробництво продукції в кількості:

а) продукції P_1

$$1000x_1 + 6000x_2 = 1000 \cdot 0,504 + 6000 \cdot 0,496 = 3480 \text{ одиниць};$$

б) продукції P_2

$$8000x_1 + 1200x_2 = 8000 \cdot 0,504 + 1200 \cdot 0,496 = 4627 \text{ одиниць}.$$

Тоді щоденний прибуток підприємства незалежно від погоди буде 1130 грн. Перевіримо це.

Якщо погода буде гарною, то прибуток буде дорівнювати

$$1000 \cdot 0,5 + (3480 - 1000) \cdot (-0,4) + 4627 \cdot 0,35 = 1130 \text{ грн.}$$

За поганої погоди прибуток підприємства харчування становитиме

$$3480 \cdot 0,5 + 1200 \cdot 0,35 + (4627 - 1200) \cdot (-0,3) = 1130 \text{ грн.}$$

Зауваження 1. Результати моделювання розглянутої конфліктної ситуації у вигляді гри підприємства з природою можна використовувати на практиці лише в тому випадку, якщо кількість виробів P_1 та P_2 , які щоденно реалізуються буде несуттєво відрізнятися від вказаних в умові задачі відповідно як для гарної, так і для поганої погоди.

Зауваження 2. Якщо в місті, в якому функціонує розглянуте підприємство харчування, є бюро прогнозу погоди, причому його прогнози з високою ймовірністю виправдовуються, то в цьому разі ситуація, що аналізується, втрачає ігровий зміст. Задача може бути розв'язана імовірнісними методами.

Отже нами розглянуто простий спосіб розв'язування ігор 2×2 в області змішаних стратегій. Як знайти розв'язок у загальному випадку гри $m \times n$, особливо при $m > 3$, $n > 3$? Методи розв'язування таких ігор будуть розглянуті нижче. Тут лише зауважимо, що інколи вимірність гри можна зменшити, в тому числі до 2×2 або $2 \times n$, або $m \times 2$, шляхом викреслювання стратегій, що дублюються, або явно не вигідних стратегій (редукція гри), а потім використовувати викладений спосіб розв'язування ігор.

Вказану процедуру пояснимо на прикладі гри, матриця платежів якої має вид (табл. 5.10)

Таблиця 5. 10

$A_i \backslash B_j$	B_1	B_2	B_3	B_4
A_1	2	0	-5	1
A_2	-1	3	2	1
A_3	3	4	-1	2
A_4	-1	3	2	1

Неважко бачити, що стратегії A_2 та A_4 збігаються. Отже, одну з них, наприклад четверту, доцільно викреслити. Крім того, елементи третього рядка більше відповідних елементів першого рядка. Враховуючи, що елементи матриці – це виграші гравця А, скажімо підприємства в грі з природою, приходимо до висновку про явну не вигідність стратегії A_1 в порівнянні з A_3 . Після викреслювання першого рядка одержимо (табл. 5.11).

Таблиця 5. 11

$A_i \backslash B_j$	B_1	B_2	B_3	B_4
A_2	-1	3	2	1
A_3	3	4	-1	2
A_4	-1	3	2	1

$A_i \backslash B_j$	B_1	B_2	B_3	B_4
A_2	-1	3	2	1
A_3	3	4	-1	2

Продовжуючи аналіз матриці платежів, відмітимо, що для гравця В, зацікавленого у зменшенні свого програшу, стратегія B_2 явно гірше стратегії B_1 (кожний елемент другого стовпця не менше ніж відповідний елемент першого стовпця). Отже, стратегія B_2 не може бути в числі оптимальних. Зважаючи на сказане, при знаходженні розв'язку гри стратегію B_2 можна не розглядати. Викреслимо стратегію B_2 . Остаточо одержимо таку матрицю платежів (табл. 5.12):

Таблиця 5.12

$A_i \backslash B_j$	B_1	B_3	B_4
A_2	-1	2	1
A_3	3	-1	2

Виявлення явно невігідних стратегій, а також тих, що дублюють інші стратегії, дозволяє спростити гру та полегшити пошук її розв'язку.

Зауваження. Теорія ігор, як і будь-який інший математичний апарат, має свої обмеження. Найважливіше з них полягає у наступному: виграш зводиться тільки до одного числа, інакше кажучи, ефективність тієї чи іншої стратегії характеризується лише одним параметром. Разом з тим вибір найкращих стратегій в реальних конфліктних ситуаціях може

вимагати кількох показників ефективності. І все ж таки ігровими методами можуть знаходитися якщо не оптимальні (з точки зору кількох показників ефективності), то принаймні близькі до них раціональні стратегії.

5.5 Розв'язування матричних ігор методами лінійного програмування

Один із методів розв'язування ігор великої вимірності базується на такому факті: будь-якій матричній грі можна зіставити пару спряжених (двоїстих) задач лінійного програмування (ЛП), розв'язки яких визначають оптимальні стратегії гравців, і навпаки, для кожної пари спряжених задач ЛП, що мають розв'язки, можна побудувати гру так, що оптимальні стратегії гравців дадуть розв'язки вказаних задач.

Покажемо, як здійснюється перехід від гри до задач лінійного програмування. Розглянемо скінченну гру двох гравців А та В з нульовою сумою. Припустимо, що гравець А має чисті стратегії A_1, A_2, \dots, A_m , а гравець В – відповідно B_1, B_2, \dots, B_n , відома матриця платежів з елементами a_{ij} . Необхідно знайти оптимальні змішані стратегії гравців:

$$\bar{X} = (x_1, x_2, \dots, x_m), \quad \bar{Y} = (y_1, y_2, \dots, y_n), \quad (5.5.1)$$

де x_i, y_j – ймовірності застосування гравцями своїх чистих стратегій, які задовольняють умовам

$$\sum_{i=1}^m x_i = 1, \quad \sum_{j=1}^n y_j = 1, \quad (5.5.2)$$
$$x_i \geq 0, \quad y_j \geq 0.$$

Крім того, припустимо також, що всі елементи матриці платежів – невід’ємні ($a_{ij} \geq 0$). Якщо вказана умова не виконується, то неважко зробити так, щоб всі елементи стали невід’ємними. Для цього необхідно до них додати достатньо велике додатне число M . Правда, в цьому випадку ціна гри збільшиться на M , проте оптимальні стратегії не зміняться.

Нагадаємо: із означення розв’язку гри випливає, що оптимальна стратегія гравця A забезпечує йому виграш, не менший ніж ціна гри V , за будь-якої (у тому числі за будь-якої чистої) стратегії гравця B .

Якщо гравець A застосує свою оптимальну (поки що невідому) стратегію, а гравець B – свою чисту стратегію B_j , то середній виграш гравця A буде

$$\sum_{i=1}^m a_{ij} x_i \geq V, \quad j = 1, 2, \dots, n. \quad (5.5.3)$$

Розділимо обидві частини нерівності (5.2.3) на величину V , яка обов’язково додатна, оскільки всі елементи матриці платежів невід’ємні. В результаті одержимо

$$\sum_{i=1}^m a_{ij} x'_i \geq V, \quad j = 1, 2, \dots, n. \quad (5.5.4)$$

де

$$x'_i = \frac{x_i}{V}. \quad (5.5.5)$$

Умова $\sum_{i=1}^m x_i = 1$ після переходу до нових змінних набуде вигляду

$$\sum_{i=1}^m x'_i = \frac{1}{V}. \quad (5.5.6)$$

Гравець A прагне зробити свій гарантований виграш максимальним, і отже, величину $\frac{1}{V}$ – мінімальною.

Таким чином, задача розв'язання гри зводиться до такої задачі: необхідно знайти значення змінних x'_i ($i=1,2,\dots,m$), які мінімізують лінійну функцію

$$Z = \sum_{i=1}^m x'_i \quad (5.5.7)$$

за обмежень

$$\begin{aligned} \sum_{i=1}^m a_{ij} x'_i &\geq 1, \quad j=1,2,\dots,n, \\ x'_i &\geq 0, \quad i=1,2,\dots,m \end{aligned} \quad (5.5.8)$$

Нами сформульована задача лінійного програмування, розв'язавши яку знайдемо величини x'_i . Далі визначимо ціну гри

$$V = \frac{1}{\sum_{i=1}^m x'_i} = \frac{1}{Z_{min}} \quad (5.5.9)$$

та ймовірності

$$x_i = x'_i \cdot V \quad (5.5.10)$$

оптимальної стратегії гравця А.

Визначення оптимальної стратегії гравця В зводиться до подібної задачі, але не на мінімум, а на максимум, оскільки гравець В прагне мінімізувати величину V (виграш гравця А), і отже максимізувати $\frac{1}{V}$.

Коротко повторимо міркування, аналогічні викладеним вище. Оптимальна стратегія гравця В гарантує йому середній програш не більший, ніж величина V за будь-якої (у тому числі будь-якої чистої) стратегії гравця А.

За оптимальної стратегії гравця В та стратегії A_i гравця А програш гравця В буде

$$\sum_{j=1}^n a_{ij} y_j \leq V, \quad i = 1, 2, \dots, m. \quad (5.5.11)$$

Після переходу до нових змінних y'_j одержимо наступну задачу лінійного програмування. Необхідно знайти значення змінних y'_j ($j = 1, 2, \dots, n$), які максимізують лінійну функцію

$$W = \sum_{j=1}^n y'_j \quad (5.5.12)$$

за умов

$$\sum_{j=1}^n a_{ij} y'_j \leq 1, \quad i = 1, 2, \dots, m, \quad (5.5.13)$$

$$y'_j \geq 0, \quad j = 1, 2, \dots, n. \quad (5.5.14)$$

Друга сформульована задача ЛП є спряженою (двоїстою) до першої задачі. В результаті її розв'язання відомими методами лінійного програмування визначаються величини y'_j , потім ціна гри

$$V = \frac{1}{\sum_{j=1}^n y'_j} = \frac{1}{W_{max}} \quad (5.5.15)$$

та ймовірності

$$y_j = y'_j \cdot W, \quad j = 1, 2, \dots, n. \quad (5.5.16)$$

Приклад. Знайти одним із методів лінійного програмування розв'язок гри, яка задана матрицею платежів (табл. 5.13):

Таблиця 5. 13

$A_i \backslash B_j$	B_1	B_2
A_1	$\frac{2}{3}$	$\frac{1}{11}$

A_2	$\frac{1}{6}$	$\frac{5}{11}$
-------	---------------	----------------

Спочатку знайдемо ймовірності x_1, x_2 – застосування чистих стратегій гравцем А. Згідно з викладеним, задача лінійного програмування має вид

$$\begin{cases} \frac{2}{3}x'_1 + \frac{1}{6}x'_2 \geq 1, \\ \frac{1}{11}x'_1 + \frac{5}{11}x'_2 \geq 1, \\ x'_1 \geq 0, x'_2 \geq 0, \end{cases} \quad (5.5.17)$$

$$Z = x'_1 + x'_2 \rightarrow \min. \quad (5.5.18)$$

Розв'язок знайдемо графічним методом. На рисунку 5.1 зображена область допустимих значень змінних x'_1, x'_2 та вектор \vec{c} , що визначає напрям зростання цільової функції

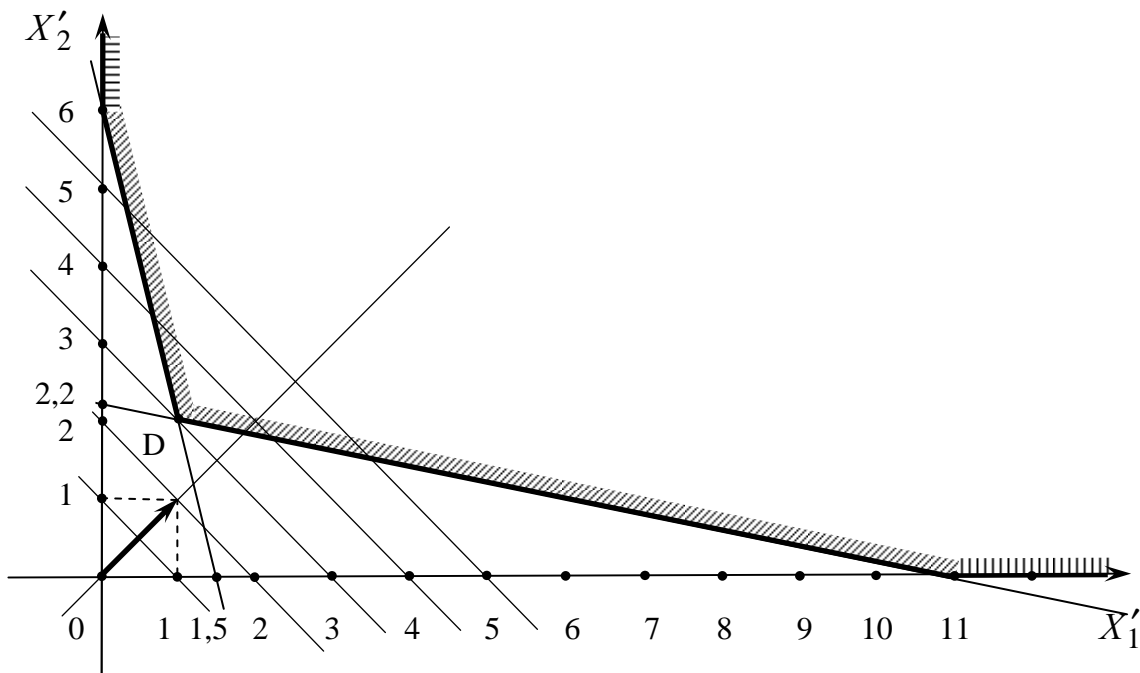


Рис. 5. 1.

Неважко бачити, що цільова функція набуває найменшого значення у точці D, координати якої, знайдені в результаті розв'язання системи рівнянь

$$\begin{cases} \frac{2}{3}x'_1 + \frac{1}{6}x'_2 = 1, \\ \frac{1}{11}x'_1 + \frac{5}{11}x'_2 = 1, \end{cases} \quad (5.5.19)$$

дорівнюють $x'_1 = 1$, $x'_2 = 2$. Тоді мінімум цільової функції $Z_{min} = 1 + 2 = 3$.

Ціна гри дорівнює

$$V = \frac{1}{Z_{min}} = \frac{1}{3}.$$

Ймовірності x_1 та x_2 визначаються за формулою (5.5.10)

$$x_1 = 1 \cdot \frac{1}{3} = \frac{1}{3}, \quad x_2 = 2 \cdot \frac{1}{3} = \frac{2}{3}.$$

Отже оптимальна змішана стратегія гравця А має вид $\bar{X} = \left(\frac{1}{3}, \frac{2}{3} \right)$.

Це означає, що чисті стратегії A_1 та A_2 застосовуються гравцем А з ймовірностями рівними відповідно $\frac{1}{3}$ та $\frac{2}{3}$.

Оптимальна стратегія гравця В знаходиться аналогічно. Запишемо задачу ЛП для визначення y'_1 та y'_2 .

$$W = y'_1 + y'_2 \rightarrow \max, \quad (5.5.20)$$

$$\begin{cases} \frac{2}{3}y'_1 + \frac{1}{11}y'_2 \leq 1, \\ \frac{1}{6}y'_1 + \frac{5}{11}y'_2 \leq 1, \\ y'_1 \geq 0, \quad y'_2 \geq 0. \end{cases} \quad (5.5.21)$$

Область допустимих значень змінних y'_1 та y'_2 , а також вектор цільової функції зображені на рис 5.2.

Розв'язок задачі ЛП, знайдений графічним методом, має вид

$$y'_1 = \frac{24}{19}, \quad y'_2 = \frac{33}{19}, \quad W_{max} = 3.$$

Тоді одержимо

$$V = \frac{1}{W_{max}} = \frac{1}{3}, \quad y_1 = \frac{8}{19}, \quad y_2 = \frac{11}{19}.$$

Оптимальна змішана стратегія гравця В полягає в тому, що він з ймовірністю $\frac{8}{19}$ вибирає свою першу чисту стратегію та з ймовірністю $\frac{11}{19}$ – другу чисту стратегію.

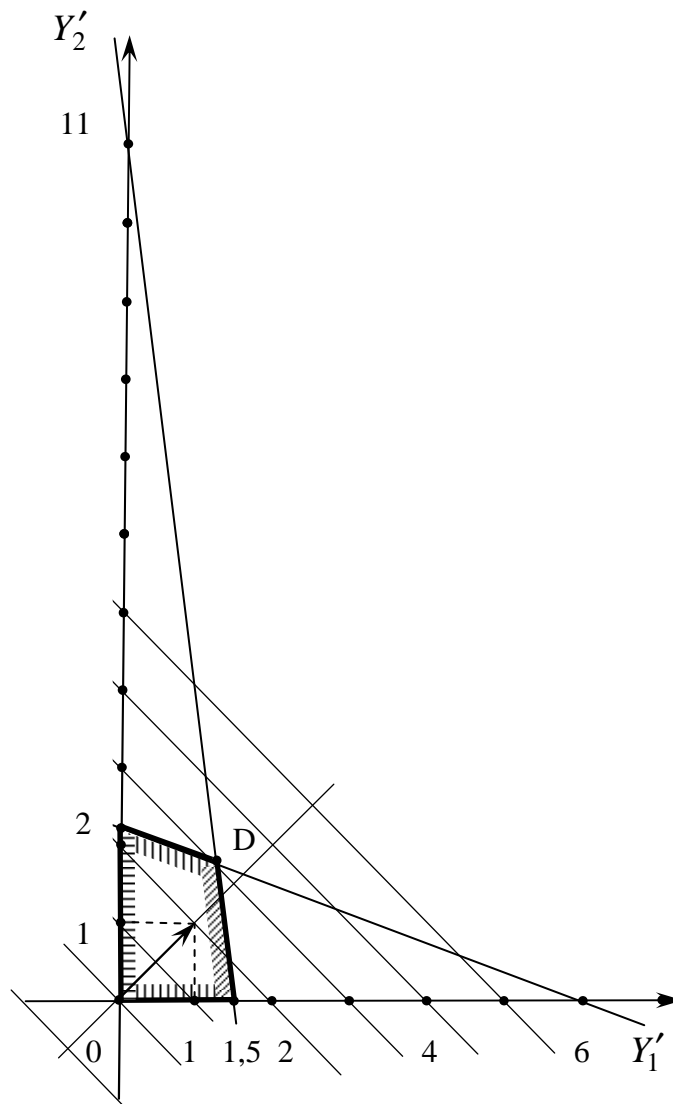


Рис. 5. 2.

Зауваження. Крім викладених, для розв'язування ігор широко застосовується так званий метод ітерацій, який дозволяє знаходити наближені розв'язки, які забезпечують середній виграш, достатньо близький до ціни гри.

5.6 Математичне моделювання ігор та розв'язування господарських задач

Задачі прийняття рішень виникають в різних сферах цілеспрямованої людської діяльності, в тому числі в торгівлі та громадському харчуванні, і виникають тоді, коли визначена мета, яку необхідно досягти, відомі різні способи її досягнення (стратегії), та існують різні обмеження, які накладаються на вибір стратегій.

Проблемам прийняття оптимальних рішень стосовно об'єктів управління різної природи і в різних умовах присвячена велика кількість наукових дисциплін.

В залежності від випадкових та невизначених факторів, які впливають на результат операції, задачі прийняття рішень (ЗПР) можна розділити на такі класи:

1. Прийняття рішень у випадку визначеності (детерміновані ЗПР);
2. Прийняття рішень за наявності ризику (стохастичні ЗПР);
3. Прийняття рішень в умовах невизначеності.

Перший клас утворюють найбільш прості ЗПР, в яких все визначено: відомі конкретні результати застосування кожної стратегії, критерії оптимальності та обмеження, які залежать тільки від стратегій, що застосовуються, та повністю відомих факторів. Для розв'язування задач цього типу застосовуються різні методи математичного програмування, зокрема добре розроблені методи лінійного програмування та інші.

У разі стохастичних ЗПР застосування тієї чи іншої стратегії приводить до одного з множини результатів, ймовірності яких заздалегідь відомі (тому, хто приймає рішення).

В задачах прийняття рішень в умовах невизначеності критерій оптимальності залежить не тільки від стратегії, яка застосовується, деяких

фіксованих факторів, але також, що дуже важливо, – від невизначених факторів, які обумовлюють різні результати кожної стратегії, ймовірності яких або невідомі, або відомі з недостатньою точністю, або ж зовсім не мають значення.

Детальніше, розглянемо ці два класи задач, особливо останній, його зв'язок з теорією ігор. Зауважимо, що цей клас задач найчастіше зустрічається в економічних дослідженнях, і є найбільш важким.

В стохастичних ЗПР відомі набори можливих стратегій A_1, A_2, \dots, A_m , операції та ймовірності можливих результатів за різних стратегій, а також значення деякого показника ефективності операції за кожної стратегії та різних результатів. Для розв'язування застосовується спосіб «оптимізація в середньому», який полягає в наступному: спочатку шукають математичне сподівання показника ефективності для кожної із стратегій. За оптимальну вибирається стратегія, яка максимізує вказану величину математичного сподівання.

Цей спосіб оптимізації зводить стохастичну задачу прийняття рішень до детермінованої.

ЗПР в умовах невизначеності – це задачі вибору оптимальної стратегії в операції, результат якої залежить, по-перше, від стратегій сторони, яка оперує; по-друге, від так званих випадкових факторів, про які є повна стохастична інформація, достатня для визначення ймовірностей різних результатів операції, по-третє, від деяких невизначених факторів, невідомих тому, хто приймає рішення, і некерованих ним.

Найбільший інтерес викликають невизначені фактори наступних видів: 1) фактори, обумовлені участю в операції деяких розумних, активно діючих учасників, які переслідують різні цілі (випадок стратегічної невизначеності); 2) фактори обумовлені тим, що вся сукупність обставин,

за яких доводиться приймати рішення, недостатньо вивчена, досліджена (випадок природної невизначеності або невідомого стану природи).

Для розв'язування ЗПР за умов стратегічної невизначеності застосовується спеціальний математичний апарат – теорія ігор (інакше – теорія стратегічних ігор) та теорія мінімакса. Математичним апаратом розв'язування ЗПР за умов природної невизначеності є теорія ігор з природою (теорія стохастичних рішень).

Принципова різниця ЗПР за умов стратегічної та природної невизначеності полягає в наступному. По-перше, в ЗПР першого типу кожній стороні, яка оперує, відома мета, набір можливих стратегій учасників операції; по-друге, учасники є розумними, активно діють і прагнуть до максимально можливого досягнення власної мети. В ЗПР другого типу природу не можна розглядати як розумного та активно діючого учасника, який прагне до свідомо поставленої мети. Отже, в цьому випадку невизначеність виступає в більш «тяжкій» формі, оскільки ускладнюється формулювання принципу оптимальності.

Перш, ніж приступати до детального розгляду ЗПР за умов природної невизначеності, зробимо декілька зауважень з приводу попереднього аналізу інформації, що міститься в матриці виграшів.

В іграх з природою доцільно матрицю виграшів за можливістю спростити, відкинувши дублюючі та явно невігідні (для сторони, що оперує) стратегії. Словом, зробити так, як в стратегічних іграх, з тією лише різницею, що явно невігідні стратегії варто відкинути тільки для гравця А, який приймає рішення, оскільки природа свідомо не вибирає свої стратегії, щоб якомога більше йому нашкодити. Крім того, має сенс перейти від матриці виграшів до так званої матриці ризиків.

Означення. Ризиком r_{ij} гравця А, який вибрав свою стратегію A_i , у випадку, коли природа знаходиться в стані Π_j ($j = 1, 2, \dots, n$), називається

різниця між максимальним виграшем, який міг би бути, якби гравець А знав, що природа знаходиться в стані Π_j , та виграшем, який відповідає стратегії A_i та стану Π_j . Ризик визначається за формулою

$$r_{ij} = \beta_j - \alpha_{ij}, \quad (5.6.1)$$

де

$$\beta_j = \max_i a_{ij}, \quad i = 1, 2, \dots, m. \quad (5.6.2)$$

Через β_j позначено максимально можливий виграш гравця А за станом природи Π_j .

Приклад. Нижче наведені матриця виграшів (табл. 5.14) та відповідна їй матриця ризиків(табл. 5.15) гравця А.

Таблиця 5. 14

$A_i \backslash \Pi_j$	Π_1	Π_2	Π_3	Π_4
A_1	5	4	7	3
A_2	1	3	12	5
A_3	4	3	4	6

Таблиця 5. 15

$A_i \backslash \Pi_j$	Π_1	Π_2	Π_3	Π_4
A_1	0	0	5	3
A_2	4	1	0	1
A_3	1	1	8	0

Елементи матриці ризиків одержимо так: у кожному стовпці матриці виграшів знайдено максимальний елемент (у першому стовпці $\beta_1 = 5$, у другому – $\beta_2 = 4$, у третьому – $\beta_3 = 12$, у четвертому – $\beta_4 = 6$), потім кожний елемент матриці виграшів віднімемо від максимального у даному стовпці значення (див. формулу (5.6.1)).

Зміст матриці ризиків полягає в тому, що вона дає менш перекручене уявлення про відносну вигідність тієї чи іншої стратегії гравця А. Пояснимо сказане. Нехай, наприклад, $a_{ij} > a_{kl}$. При цьому може трапитись, що виграш a_{ij} більше за a_{kl} не тому, що стратегія A_i краща за стратегію A_k , а через те, що стан природи Π_j більш сприятливий ніж стан Π_l .

Проаналізуємо дві матриці, наведені в прикладі. В матриці виграшів елементи a_{31} та a_{33} дорівнюють один одному $a_{31} = a_{33} = 4$. Проте ці виграші не можна вважати рівноцінними з точки зору успішності вибору стратегії гравцем А. Справді, в першому випадку за стану природи Π_1 виграш $a_{31} = 4$ одиниці гарний, оскільки найбільший можливий виграш для цього стану природи становить 5 одиниць, в той саме час точно такий за розміром виграш $a_{33} = 4$ одиниці для стану природи Π_3 є поганим. І, отже, вибір стратегії A_3 , виходячи з одного й того ж значення виграшу, в одному випадку – гарний, в другому – поганий. Вказану обставину віддзеркалює матриця ризиків – $r_{31} = 1$, $r_{33} = 8$.

Розглянемо різні підходи до ігор з природою. Виділимо наступні випадки невизначеності:

- 1) відомі апріорні недостовірні значення ймовірностей різних станів природи;
- 2) апріорна інформація про стани природи відсутня, проте є суб'єктивні міркування про їх ймовірності;
- 3) немає жодної інформації про ймовірності можливих станів природи.

Пошук розв'язків гри з природою у разі невизначеності першого виду здійснюється за аналогією зі стохастичними ЗПР, тобто застосовується спосіб оптимізації в середньому. В результаті вибирається стратегія, яка максимізує середній виграш гравця А.

Чим же відрізняються стохастичні задачі прийняття рішень від ЗПР у разі природної невизначеності з апіорно відомими станами природи? Різниця в наступному: в стохастичних ЗПР є достовірна інформація про ймовірності достовірних результатів (наслідків) операції, в другому випадку – апіорна недостовірна інформація про ті ж ймовірності (невизначеність обумовлена саме недостатністю вказаної інформації). Досліднику, який приймає рішення, не залишається нічого іншого, як застосувати той же спосіб оптимізації в середньому, хоча якість вихідної (початкової) інформації про ймовірності станів природи гірша.

Розглянемо ситуацію: сторона, що приймає рішення, має стратегії A_1, A_2, \dots, A_m , природа може знаходитись в стані Π_1 , або Π_2, \dots , або Π_n , апіорні ймовірності яких відповідно дорівнюють q_1, q_2, \dots, q_n . Матриця виграшів гравця А має вид (табл. 5.16)

Таблиця 5. 16

$A_i \backslash \Pi_j$	Π_1	Π_2	...	Π_n
A_1	a_{11}	a_{12}	...	a_{1n}
A_2	a_{21}	a_{22}	...	a_{2n}
...
A_m	a_{m1}	a_{m2}	...	a_{mn}

Необхідно вибрати оптимальну стратегію гравця А.

Як було сказано вище, в задачах такого типу за показник ефективності приймається математичне сподівання виграшу гравця А.

Оптимальна стратегія вибирається за умови максимізації цього показника. Визначимо середнє значення виграшу гравця А для i -тої стратегії

$$a_i = a_{i1} \cdot q_1 + a_{i2} \cdot q_2 + \dots + a_{in} \cdot q_n = \sum_{j=1}^n a_{ij} \cdot q_j. \quad (5.6.3)$$

Найкращою вибирається стратегія, за якої величина a_i набуває найбільшого значення. Оптимальна стратегія при багатократному повторенні ситуації в однакових умовах приводить до найкращого в середньому виграшу.

Можна довести, що в задачах класу, що розглядається, найкращий розв'язок полягає в виборі однієї з чистих стратегій гравця А, застосування змішаних стратегій з будь-якими ймовірностями x_1, x_2, \dots, x_m не може бути вигіднішим для нього, ніж застосування чистої стратегії, вибраної вказаним способом. Іншими словами, в цьому випадку можна обмежитись лише чистими стратегіями, не застосовуючи змішаних.

Приклад. Підприємство харчування при визначенні щоденного плану роботи може вибрати одну із своїх чистих стратегій. Кожна стратегія – це план, складений заздалегідь, орієнтуючись на певний стан природи, під яким розуміють погодні умови, попит, ту чи іншу ситуацію з поставками сировини і т. ін.

Різні стратегії забезпечують підприємству різний прибуток (в тис. грн) в залежності від стану природи, який набуває значень $\Pi_1, \Pi_2, \Pi_3, \Pi_4$. Матриця прибутку підприємства наведена нижче (табл. 5.17). Крім того, апріорно відомі ймовірності різного стану природи:

$$q_1 = 0,1, q_2 = 0,2, q_3 = 0,4, q_4 = 0,3.$$

Необхідно вибрати план роботи підприємства, який би забезпечував максимальний прибуток.

Таблиця 5. 17

--	--	--	--	--	--

$A_i \backslash P_j$	P_1	P_2	P_3	P_4	a_i
A_1	4	8	5	3	4,9
A_2	10	6	3	4	4,6
A_3	3	4	9	5	6,2
A_4	2	5	6	8	6,0
q_i	0,1	0,2	0,4	0,3	

Середній очікуваний прибуток підприємства за різних стратегій, обчислений за формулою (5.6.3), надано в крайньому правому стовпці таблиці; в нижньому рядку таблиці записані ймовірності стану природи.

Із наведених результатів видно, що оптимальною стратегією є стратегія A_3 , за якої середній очікуваний прибуток становить 6,2 тис. грн.

Зауваження. В ЗПР з природною невизначеністю розглянутого виду, поряд з викладеним, часто застосовують інший спосіб вибору оптимального розв'язку, який полягає у наступному: будується матриця ризику, потім визначається середній ризик для кожної із стратегій; найкращою вибирається стратегія, яка мінімізує середній ризик. Доведено, що оптимальні розв'язки, знайдені обома способами (оптимізації в середньому та мінімізації середнього ризику) збігаються.

Розглянемо другий із названих випадків природної невизначеності, який полягає в тому, що сторона, яка приймає рішення, не має апріорної інформації про ймовірності стану природи, проте є деякі інтуїтивні міркування з цього питання. Зокрема, якщо сторона, яка приймає рішення, вважає рівно можливими різні стани природи, то їх ймовірності приймаються рівними

$$q_1 = q_2 = \dots = q_n = \frac{1}{n}. \quad (5.6.4)$$

Часто з цією метою використовуються методи експертних оцінок. Вони полягають у визначенні середніх значень ймовірностей різного стану природи, призначених незалежно один від одного групою кваліфікованих експертів. Потім для вибору оптимальної стратегії застосовується метод оптимізації в середньому, або мінімізації середнього ризику.

Найскладнішим випадком природної невизначеності є випадок відсутності будь-якої інформації про ймовірність різного стану природи. Методи, які застосовуються для дослідження таких конфліктних ситуацій, зводяться до забезпечення деякого гарантованого результату.

Розглянемо декілька критеріїв вибору оптимальних розв'язків, які застосовуються в теорії ігор з природою.

Максимінний критерій Вальда (критерій найбільшого песимізму).

Сутність його у наступному: за оптимальну вибирається та стратегія гравця А, за якої мінімальний виграш найбільший. Інакше кажучи, за якої гарантований виграш не менший ніж «максимін»

$$W = \max_i \min_j a_{ij}, \quad j = 1, 2, \dots, n, \quad i = 1, 2, \dots, m. \quad (5.6.5)$$

По суті, за такого підходу природа розглядається не як зацікавлена сторона, а як активний зловмисний супротивник. Це, звичайно ж, песимістичний погляд на природу. Він орієнтує людину, яка приймає рішення, на найгірші умови. У відповідності з цим критерієм оптимальною вибирається стратегія, для якої в найгірших умовах виграш максимальний.

Критерій мінімаксного ризику Севіджа

Сутність критерію: за оптимальну рекомендується вибрати ту стратегію, за якої величина ризику набуває найменшого значення в найбільш несприятливій ситуації (коли ризик максимальний) тобто стратегію, яка гарантує мінімум максимального ризику

$$S = \min_i \max_j r_{ij}, \quad j = 1, 2, \dots, n, \quad i = 1, 2, \dots, m. \quad (5.6.6)$$

Іншими словами, критерій, який розглядається, рекомендує особі, яка приймає рішення, уникати великого ризику в процесі прийняття рішень.

Цей критерій також є критерієм найбільшого песимізму, який проявляється в наступному: найгіршими умовами вважаються умови максимального ризику (найбільша втрата виграшу в порівнянні з виграшем, який міг би бути за даних умов).

Критерій Гурвиця (критерій песимізму – оптимізму)

У відповідності з цим критерієм вибору стратегії за умов невизначеності рекомендується уникати крайностей: крайнього песимізму та легковажного оптимізму.

Критерій Гурвиця має вид

$$H = \max_i \left\{ \chi \cdot \min_j a_{ij} + (1 - \chi) \max_j a_{ij} \right\}, \quad (5.6.7)$$

де χ – коефіцієнт, який вибирається довільно (особою, яка приймає рішення) між нулем та одиницею.

Із (5.6.7) видно, що при $\chi = 1$ критерій Гурвиця стає песимістичним критерієм Вальда, при $\chi = 0$ – критерієм найбільшого оптимізму, оскільки в цьому випадку він рекомендує вибрати стратегію, для якої виграш максимальний в найгірших умовах.

Якщо $0 < \chi < 1$, то критерій рекомендує орієнтуватись на щось середнє. Коефіцієнт χ є начебто мірою песимізму того, хто приймає рішення за умов невизначеності. Що небезпечніша ситуація, то більше прагнення «підстрахуватись», то ближче до одиниці вибирається коефіцієнт χ .

Зауваження. В іграх з природою за умов невизначеності доцільно вибрати оптимальну стратегію, керуючись всіма трьома критеріями одночасно. Збіг трьох розв'язків є вагомою підставою для вибору тієї стратегії, яка ними рекомендується.

У разі не збігання розв'язків, знайдених з використання різних критеріїв, рекомендується проаналізувати їх переваги та недоліки і вибрати один з них.

Приклад. Застосування різних критеріїв оптимальності розглянемо на прикладі вибору оптимального плану роботи підприємства харчування. Матриця прибутку підприємства за різних планів його роботи (стратегій) та різних станах природи наведена нижче (табл. 5.). Ймовірності різних станів природи вважаємо невідомими. Необхідно знайти оптимальну стратегію (план), використовуючи критерії Вальда, Севіджа, Гурвиця при $\chi = 0,5$.

Розв'язання 1 (за допомогою критерію Вальда).

Таблиця 5.18

$A_i \backslash P_j$	P_1	P_2	P_3	P_4	$\min_j a_{ij}$
A_1	4	8	5	3	3
A_2	5	6	4	5	4
A_3	10	6	3	4	3
A_4	3	4	9	5	3

A ₅	2	5	6	8	2
----------------	---	---	---	---	---

$$W = \max_{1 \leq i \leq 5} \min_{1 \leq j \leq 4} a_{ij} = 4.$$

Оптимальна стратегія підприємства – A₂.

Розв'язання 2 (з використанням критерію Севіджа).

Матриця ризиків має вид (табл. 5.19)

Таблиця 5. 19

A _i \ П _j	П ₁	П ₂	П ₃	П ₄	max _j r _{ij}
A ₁	6	0	4	5	6
A ₂	5	2	5	3	5
A ₃	0	2	6	4	6
A ₄	7	4	0	3	7
A ₅	8	3	3	0	8

$$S = \min_{1 \leq i \leq 5} \max_{1 \leq j \leq 4} a_{ij} = 5.$$

Оптимальна стратегія підприємства A₂.

Розв'язання 3 (з використанням критерію Гурвиця при $\chi = 0,5$).

Таблиця 5. 20

A _i \ П _j	П ₁	П ₂	П ₃	П ₄	h _i = min _j a _{ij}	r _i = max _j a _{ij}	H _i = 0,5h _i + 0,5r _i
A ₁	4	8	5	3	3	8	5,5
A ₂	5	6	4	5	4	6	5
A ₃	10	6	3	4	3	10	6,5
A ₄	3	4	9	5	3	9	6
A ₅	2	5	6	8	2	8	5

$$H = \max_{1 \leq i \leq 5} \left\{ 0,5 \min_{1 \leq j \leq 4} a_{ij} + 0,5 \max_{1 \leq j \leq 4} a_{ij} \right\} = 6,5.$$

Оптимальна стратегія підприємства – A₃.

Таким чином, критерії Вальда та Севіджа рекомендують вибирати стратегію A_2 , а критерій Гурвиця при $\chi = 0,5$ – стратегію A_3 . Якщо у керівника, який приймає рішення, є більше підстав для песимізму ніж для оптимізму, то він може за оптимальну прийняти стратегію A_2 .

Питання для самоконтролю

1. Що називається грою? Який зв'язок між конфліктною ситуацією та грою?
2. Сформулюйте предмет та задачу теорії ігор.
3. Яка гра називається скінченною?
4. Яка гра називається матричною?
5. Наведіть приклад моделювання конфліктної ситуації.
6. Які стратегії називаються: а) чистими; б) змішаними?
7. Що таке нижня та верхня ціни гри?
8. Що називається сідловою точкою?
9. В чому сутність розв'язування гри в змішаних стратегіях?
10. Сформулюйте основну теорему теорії ігор.
11. Як змінюється перехід від гри до задач лінійного програмування?
12. Які існують етапи розв'язування гри графічним методом.
13. Що називається ризиком гравця? Як він визначається?
14. Сформулюйте критерії вибору оптимальних розв'язків, які застосовуються в теорії ігор з природою.
15. В чому зміст критеріїв:
 - а) Вальда?
 - б) Севіджа?
 - в) Гурвиця?

Задачі до глави 5

5.1. Гра « n сірників». В грі приймають участь дві особи, які одночасно та незалежно викладають один або два, або три і т. д., до n штук сірників. Виграш, наприклад в умовних одиницях, дорівнює сумарній кількості сірників, викладених обома гравцями. Якщо вона парна, то виграє перший гравець; якщо ж – непарна, то – другий. Необхідно побудувати матрицю платежів.

Номер задачі	Значення n	Номер задачі	Значення n	Номер задачі	Значення n	Номер задачі	Значення n	Номер задачі	Значення n
5.1.1	8	5.1.6	13	5.1.11	6	5.1.16	12	5.1.21	15
5.1.2	14	5.1.7	9	5.1.12	15	5.1.17	9	5.1.22	5
5.1.3	11	5.1.8	12	5.1.13	10	5.1.18	4	5.1.23	10
5.1.4	4	5.1.9	5	5.1.14	7	5.1.19	11	5.1.24	6
5.1.5	8	5.1.10	9	5.1.15	13	5.1.20	14	5.1.25	12

5.2. Нижче наведено матриці платежів для задач 5.2.1 – 5.2.24. Необхідно знайти нижню та верхню ціни гри, а також розв’язок гри в чистих стратегіях, якщо він існує.

5.2.1

$A_i \backslash B_j$	B_1	B_2	B_3	B_4
A_1	6	-3	4	5
A_2	5	2	-5	3
A_3	1	2	6	4
A_4	7	-4	0	3

5.2.2

$A_i \backslash B_j$	B_1	B_2	B_3	B_4
A_1	9	5	7	6
A_2	4	11	-3	8
A_3	13	15	7	8
A_4	9	6	10	5

5.2.3

$A_i \backslash B_j$	B_1	B_2	B_3	B_4
A_1	-2	-3	4	0
A_2	7	6	5	8
A_3	-3	-2	1	-7
A_4	-1	5	0	8

5.2.4

$A_i \backslash B_j$	B_1	B_2	B_3	B_4
A_1	7	3	2	5
A_2	5	1	5	-2
A_3	-4	2	3	1
A_4	8	3	4	0

5.2.7

$A_i \backslash B_j$	B_1	B_2	B_3	B_4
A_1	2	3	-7	4
A_2	1	-2	4	5
A_3	6	1	-6	2
A_4	3	5	6	1

5.2.10

$A_i \backslash B_j$	B_1	B_2	B_3	B_4
A_1	5	4	-3	6
A_2	4	3	6	2
A_3	2	0	-5	3
A_4	8	3	2	5

5.2.13

$A_i \backslash B_j$	B_1	B_2	B_3	B_4
A_1	4	-5	2	3
A_2	3	0	3	1
A_3	-2	1	4	2
A_4	5	2	-2	1

5.2.16

$A_i \backslash B_j$	B_1	B_2	B_3	B_4
A_1	3	-3	1	2
A_2	3	0	2	-1
A_3	5	7	4	6
A_4	4	1	2	5

5.2.5

$A_i \backslash B_j$	B_1	B_2	B_3	B_4
A_1	6	10	8	5
A_2	2	7	4	0
A_3	8	5	-6	3
A_4	0	-3	6	5

5.2.8

$A_i \backslash B_j$	B_1	B_2	B_3	B_4
A_1	-1	0	2	1
A_2	9	8	2	6
A_3	5	-6	4	8
A_4	0	3	6	5

5.2.11

$A_i \backslash B_j$	B_1	B_2	B_3	B_4
A_1	2	4	3	5
A_2	10	7	9	7
A_3	6	-6	4	3
A_4	-5	1	3	5

5.2.14

$A_i \backslash B_j$	B_1	B_2	B_3	B_4
A_1	7	-3	5	6
A_2	8	3	7	10
A_3	3	4	9	3
A_4	9	-1	5	4

5.2.17

$A_i \backslash B_j$	B_1	B_2	B_3	B_4
A_1	5	3	4	9
A_2	-1	4	-7	8
A_3	5	2	7	9
A_4	3	4	6	12

5.2.6

$A_i \backslash B_j$	B_1	B_2	B_3	B_4
A_1	1	4	1	3
A_2	13	7	-6	1
A_3	4	7	4	5
A_4	-1	3	2	5

5.2.9

$A_i \backslash B_j$	B_1	B_2	B_3	B_4
A_1	0	2	-1	3
A_2	2	5	3	6
A_3	2	7	4	0
A_4	-4	6	7	8

5.2.12

$A_i \backslash B_j$	B_1	B_2	B_3	B_4
A_1	9	10	6	9
A_2	12	7	4	8
A_3	4	3	5	10
A_4	7	8	7	6

5.2.15

$A_i \backslash B_j$	B_1	B_2	B_3	B_4
A_1	6	4	1	4
A_2	5	3	7	5
A_3	9	7	5	-3
A_4	6	3	7	-6

5.2.18

$A_i \backslash B_j$	B_1	B_2	B_3	B_4
A_1	4	7	6	10
A_2	8	-3	6	4
A_3	4	7	8	-5
A_4	10	5	3	4

5.2.19

$A_i \backslash B_j$	B_1	B_2	B_3	B_4
A_1	8	3	0	4
A_2	1	-3	2	5
A_3	3	0	5	2
A_4	3	1	4	-2

5.2.20

$A_i \backslash B_j$	B_1	B_2	B_3	B_4
A_1	3	0	2	4
A_2	2	7	-4	0
A_3	6	10	5	0
A_4	-9	3	5	8

5.2.21

$A_i \backslash B_j$	B_1	B_2	B_3	B_4
A_1	3	5	7	10
A_2	0	-3	6	5
A_3	2	5	3	-6
A_4	8	5	6	2

5.2.22

$A_i \backslash B_j$	B_1	B_2	B_3	B_4
A_1	0	-4	2	4
A_2	7	3	1	5
A_3	1	5	4	3
A_4	5	2	-2	6

5.2.23

$A_i \backslash B_j$	B_1	B_2	B_3	B_4
A_1	6	0	8	9
A_2	7	6	4	-1
A_3	2	5	3	6
A_4	4	-3	3	2

5.2.24

$A_i \backslash B_j$	B_1	B_2	B_3	B_4
A_1	1	2	0	-1
A_2	2	3	-3	4
A_3	4	3	11	8
A_4	2	5	3	6

5.3. Розглядається скінченна гра двох гравців з нульовою сумою. В таблиці надано елементи a_{ij} матриці вигравів гравця А. Необхідно знайти розв'язок гри в змішаних стратегіях.

Номер задачі	Елементи матриці вигравів			
	a_{11}	a_{12}	a_{21}	a_{22}
1	2	3	4	5
5.3.1	20	10	9	21
5.3.2	29	7	8	30
5.3.3	56	23	24	55
5.3.4	38	12	13	39
5.3.5	48	15	16	50
5.3.6	28	8	9	30
5.3.7	45	20	19	43
5.3.8	40	18	16	41
5.3.9	19	7	6	20
5.3.10	35	14	13	34
5.3.11	46	25	24	45
5.3.12	61	41	40	60
5.3.13	51	9	8	50
5.3.14	28	7	6	29
5.3.15	34	18	16	35
5.3.16	28	5	4	30
5.3.17	51	30	28	50

1	2	3	4	5
5.3.18	47	25	24	46
5.3.19	18	4	3	19
5.3.20	30	7	8	31
5.3.21	54	26	25	52
5.3.22	27	8	9	26
5.3.23	16	4	3	15
5.3.24	18	9	8	20
5.3.25	39	19	17	40

5.4. В таблиці надано елементи a_{ij} матриці платежів скінченної гри двох гравців А і В з нульовою сумою. Необхідно знайти розв'язок гри одним з методів лінійного програмування.

Номер задачі	Елементи матриці платежів			
	a_{11}	a_{12}	a_{21}	a_{22}
5.3.1	2	2	1	3
5.3.2	3	2	1	4
5.3.3	2	1	2	3
5.3.4	2	1	3	5
5.3.5	4	2	3	5
5.3.6	7	6	5	8
5.3.7	6	6	5	7
5.3.8	9	7	8	10
5.3.9	8	6	7	9
5.3.10	4	3	4	5
5.3.11	6	5	4	7
5.3.12	3	2	4	6
5.3.13	7	5	6	8
5.3.14	3	2	3	4
5.3.15	8	7	6	9
5.3.16	6	4	5	7
5.3.17	5	4	6	8
5.3.18	7	6	7	8
5.3.19	5	4	3	6
5.3.20	5	3	4	6
5.3.21	7	6	8	9
5.3.22	5	4	5	6
5.3.23	4	3	2	5
5.3.24	6	5	7	9
5.3.25	4	4	3	5

5.5. Кафе при визначенні щоденного плану роботи може вибрати один з п'яти планів A_1, A_2, \dots, A_5 , заздалегідь складених виходячи з орієнтації на певні умови роботи: погодні умови, наявність сировини і т. ін. (стан природи $\Pi_1, \Pi_2, \Pi_3, \Pi_4$). Матриця прибутку підприємства (в тис. грн.) за різних планів його роботи та стану природи надана нижче. Крім того, в таблиці наведені (для задач 5.5.1 – 5.5.25) апріорні значення ймовірностей q_i стану природи. Необхідно вибрати план роботи підприємства, який би забезпечував максимальний середній прибуток.

$A_i \backslash \Pi_j$	Π_1	Π_2	Π_3	Π_4
A_1	3,5	2,1	1	2,8
A_2	1,6	2,5	4	2
A_3	2	3,6	1,4	1,7
A_4	1,8	1,2	2,3	3,6
A_5	1,5	2,6	2,5	2

Номер задачі	Ймовірності стану природи			
	q_1	q_2	q_3	q_4
1	2	3	4	5
5.5.1	0,2	0,2	0,3	0,3
5.5.2	0,1	0,5	0,3	0,1
5.5.3	0,2	0,1	0,4	0,3
5.5.4	0,3	0,2	0,3	0,2
5.5.5	0,4	0,3	0,1	0,2
5.5.6	0,5	0,2	0,2	0,1
5.5.7	0,3	0,4	0,1	0,2
5.5.8	0,4	0,3	0,1	0,2
5.5.9	0,2	0,1	0,4	0,3
5.5.10	0,1	0,2	0,3	0,4
5.5.11	0,2	0,3	0,3	0,2
5.5.12	0,3	0,4	0,2	0,1
5.5.13	0,4	0,1	0,3	0,2
5.5.14	0,5	0,1	0,1	0,3
5.5.15	0,3	0,3	0,1	0,3
5.5.16	0,4	0,2	0,3	0,1

	1	2	3	4	5
5.5.17		0,2	0,2	0,5	0,1
5.5.18		0,1	0,3	0,5	0,1
5.5.19		0,2	0,3	0,2	0,3
5.5.20		0,5	0,1	0,2	0,2
5.5.21		0,3	0,1	0,3	0,3
5.5.22		0,4	0,2	0,1	0,3
5.5.23		0,2	0,3	0,2	0,3
5.5.24		0,1	0,3	0,4	0,2
5.5.25		0,2	0,2	0,3	0,3

5.6. Планується будівництво комбінату харчування. Запропоновано чотири різні проекти комбінату A_1, A_2, A_3, A_4 , які відрізняються складом, потужністю цехів, обладнанням, яке планується використовувати, і т. ін.

Нижче (для задач 5.6.1 – 5.6.25) надано таблиці очікуваного різного прибутку a_{ij} комбінату (в тис. євро) в залежності від різних умов його роботи $\Pi_1, \Pi_2, \Pi_3, \Pi_4$ (наявності сировини, попиту, ефективності використання обладнання і т. ін.). Необхідно вибрати найкращий проект комбінату, застосувавши критерії Вальда, Севіджа та Гурвиця при $\chi = 0,3; 0,4; 0,5; 0,6; 0,7$.

5.6.1

$\Pi_j \backslash A_i$	Π_1	Π_2	Π_3	Π_4
A_1	200	130	300	150
A_2	40	210	70	280
A_3	140	230	150	200
A_4	260	100	90	170

5.6.2

$\Pi_j \backslash A_i$	Π_1	Π_2	Π_3	Π_4
A_1	150	190	60	230
A_2	80	60	200	100
A_3	210	310	250	80
A_4	400	170	90	60

5.6.3

$\Pi_j \backslash A_i$	Π_1	Π_2	Π_3	Π_4
A_1	310	160	100	70
A_2	150	200	280	100
A_3	80	300	250	120
A_4	270	130	300	160

5.6.4

$\Pi_j \backslash A_i$	Π_1	Π_2	Π_3	Π_4
A_1	250	80	110	200
A_2	130	200	170	150
A_3	320	160	130	100
A_4	170	340	150	70

5.6.5

$\Pi_j \backslash A_i$	Π_1	Π_2	Π_3	Π_4
A_1	60	190	300	230
A_2	180	150	170	150
A_3	250	270	120	180
A_4	340	80	100	90

5.6.6

$\Pi_j \backslash A_i$	Π_1	Π_2	Π_3	Π_4
A_1	130	280	70	250
A_2	360	200	230	180
A_3	150	130	300	100
A_4	180	90	310	320

5.6.7

$A_i \backslash \Pi_j$	Π_1	Π_2	Π_3	Π_4
A_1	120	380	250	150
A_2	200	130	270	340
A_3	400	170	250	300
A_4	180	400	230	180

5.6.8

$A_i \backslash \Pi_j$	Π_1	Π_2	Π_3	Π_4
A_1	250	130	310	150
A_2	160	370	280	310
A_3	180	90	400	190
A_4	310	150	270	220

5.6.9

$A_i \backslash \Pi_j$	Π_1	Π_2	Π_3	Π_4
A_1	380	180	270	250
A_2	90	200	220	210
A_3	150	250	300	140
A_4	210	370	60	400

5.6.10

$A_i \backslash \Pi_j$	Π_1	Π_2	Π_3	Π_4
A_1	300	200	150	210
A_2	120	330	300	270
A_3	190	250	240	200
A_4	400	210	250	320

5.6.11

$A_i \backslash \Pi_j$	Π_1	Π_2	Π_3	Π_4
A_1	200	280	340	380
A_2	100	320	350	200
A_3	150	310	240	280
A_4	350	130	100	150

5.6.12

$A_i \backslash \Pi_j$	Π_1	Π_2	Π_3	Π_4
A_1	250	140	260	270
A_2	130	280	200	210
A_3	400	320	150	260
A_4	210	270	100	300

5.6.13

$A_i \backslash \Pi_j$	Π_1	Π_2	Π_3	Π_4
A_1	400	90	210	260
A_2	370	300	250	280
A_3	130	180	310	300
A_4	100	320	210	200

5.6.14

$A_i \backslash \Pi_j$	Π_1	Π_2	Π_3	Π_4
A_1	100	180	250	300
A_2	180	200	260	310
A_3	150	300	140	270
A_4	380	240	320	230

5.6.15

$A_i \backslash \Pi_j$	Π_1	Π_2	Π_3	Π_4
A_1	180	340	290	380
A_2	210	170	300	200
A_3	250	280	150	200
A_4	280	100	300	150

5.6.16

$A_i \backslash \Pi_j$	Π_1	Π_2	Π_3	Π_4
A_1	400	250	300	210
A_2	350	310	240	150
A_3	250	280	150	300
A_4	210	320	200	270

5.6.17

$A_i \backslash \Pi_j$	Π_1	Π_2	Π_3	Π_4
A_1	100	200	140	230
A_2	250	260	140	320
A_3	270	320	280	140
A_4	240	300	200	180

5.6.18

$A_i \backslash \Pi_j$	Π_1	Π_2	Π_3	Π_4
A_1	210	300	250	400
A_2	380	350	310	120
A_3	170	200	300	280
A_4	190	150	400	250

5.6.19

$A_i \backslash \Pi_j$	Π_1	Π_2	Π_3	Π_4
A_1	140	210	300	280
A_2	210	310	250	90
A_3	230	150	180	80
A_4	70	100	150	200

5.6.20

$A_i \backslash \Pi_j$	Π_1	Π_2	Π_3	Π_4
A_1	200	170	160	170
A_2	250	200	130	70
A_3	130	200	300	320
A_4	310	200	250	160

5.6.21

$A_i \backslash \Pi_j$	Π_1	Π_2	Π_3	Π_4
A_1	150	60	250	70
A_2	70	280	300	270
A_3	200	170	160	170
A_4	80	200	250	100

5.6.22

$A_i \backslash \Pi_j$	Π_1	Π_2	Π_3	Π_4
A_1	250	150	120	180
A_2	150	200	300	180
A_3	150	200	300	210
A_4	180	170	230	300

5.6.23

$A_i \backslash \Pi_j$	Π_1	Π_2	Π_3	Π_4
A_1	340	270	100	180
A_2	400	310	90	80
A_3	250	150	120	150
A_4	200	210	300	280

5.6.24

$A_i \backslash \Pi_j$	Π_1	Π_2	Π_3	Π_4
A_1	150	160	280	70
A_2	150	200	300	180
A_3	120	280	300	150
A_4	60	250	170	210

5.6.25

$A_i \backslash \Pi_j$	Π_1	Π_2	Π_3	Π_4
A_1	190	170	270	100
A_2	60	150	250	80
A_3	150	160	170	80
A_4	200	170	100	150

ГЛАВА 6.

ОПУКЛЕ ПРОГРАМУВАННЯ

6.1 Задачі нелінійного опуклого програмування

Вище розглянуто методи розв'язування широкого класу задач на знаходження екстремумів різних функцій, задач, в яких як цільові функції, так і обмеження є лінійними. Звідси, нагадаємо, і назва – задачі лінійного програмування. При створенні математичних моделей цих задач основними були припущення стосовно пропорційності та лінійності. Наприклад, в задачі про вибір оптимальної програми роботи підприємства припускали, що прибуток пропорційний кількості продукції різних видів, що випускається підприємством протягом певного інтервалу часу.

Вказані припущення зазвичай виконуються при складанні планів на невеликі проміжки часу. Більш точними є нелінійні залежності між показниками та факторами, які впливають на критерій оптимальності. Більш повний та точний облік норм споживання тих чи інших ресурсів обумовлює, як правило, побудову нелінійних обмежень, отже, виникає необхідність модернізації моделей лінійного програмування.

Задача нелінійного програмування (НП) в загальному випадку формулюється так. Необхідно знайти точку $X = (x_1, x_2, \dots, x_n)$, n -вимірному евклідового простору, яка мінімізує (максимізує) функцію

$$F(X) = F(x_1, x_2, \dots, x_n) \quad (6.1.1)$$

за наявності обмежень

$$q_i(X) = q_i(x_1, x_2, \dots, x_n) \geq 0, \quad (6.1.2)$$

$$x_i \geq 0, \quad i = 1, 2, \dots, n, \quad (6.1.3)$$

де $F(X)$ та $q_i(X)$ – нелінійні функції.

Функція $F(X)$ називається цільовою, а $q_i(X)$ – функціями обмежень.

Означення. Точка X n -вимірного простору, координати якої задовольняють обмеженням (6.1.2), (6.1.3) називається допустимою.

Множина всіх допустимих точок X утворює допустиму область D .

Означення. Оптимальним розв'язком задачі (6.1.1) – (6.1.3) називається допустима точка X , яка мінімізує (максимізує) цільову функцію $F(X)$.

Розв'язання задачі нелінійного програмування зводиться до знаходження такої точки на допустимій області, в якій досягається поверхня найменшого (найбільшого) рівня. З урахуванням сказаного уточнимо означення задач нелінійного програмування: це екстремальні задачі, в яких або цільова функція, або функції обмежень, або і ті і другі є нелінійними.

Цільову функцію можна уявити як деяку місцевість з узгір'ями, ущелинами, ярами тощо, обмеження – огорожі на цій місцевості, лінії рівня – лінії перетину площин рівня з рельєфом місцевості. Тоді «висота» точок, які належать відгородженій ділянці і мають найнижчий рівень, буде характеризувати мінімальне значення цільової функції.

Приклад. Знайти точку $X = (x_1, x_2)$, в якій цільова функція $Z = (x_1 - 3)^2 + (x_2 - 4)^2$ приймає мінімальне значення за обмежень

$$\begin{cases} x_1 + 2x_2 \geq 1, \\ 2x_1 + x_2 \leq 6, \\ x_1 \geq 0, x_2 \geq 0. \end{cases}$$

Розв'язок знайдемо графічним методом.

На рисунку 6.1 зображена область допустимих розв'язків: чотирикутник $ABMN$. Лінії рівня одержимо, поклавши $Z = C$ ($C > 0$).

Легко бачити, що $(x_1 - 3)^2 + (x_2 - 4)^2 = C$ є рівнянням кола з центром у точці $O_1(3; 4)$ та радіусом $R = \sqrt{C}$.

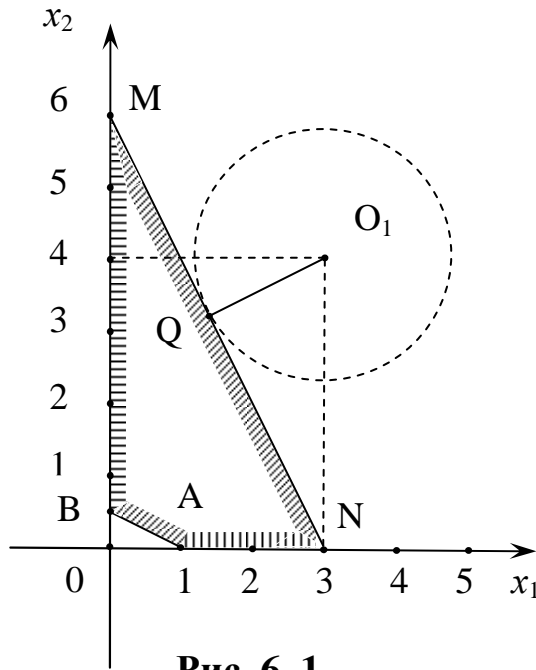


Рис. 6. 1.

Що менший радіус, то меншим є значення цільової функції. Цільова функція, що розглядається, приймає найменше значення в точці Q, в якій коло дотикається прямої MN.

Координати точки Q визначаються в результаті розв'язання системи двох рівнянь, що відповідають прямим MN та O_1Q

$$\begin{cases} 2x_1 + x_2 = 6 & (MN), \\ -x_1 + 2x_2 = 5 & (O_1Q), \end{cases}$$

$Q(1,4; 3,2)$.

Мінімальне значення цільової функції дорівнює відстані від точки O_1 до точки Q або до прямої MN

$$Z_{min} = \frac{|2 \cdot 3 + 4 - 6|}{\sqrt{2^2 + 1^2}} = \frac{4}{\sqrt{5}}.$$

Природно, що клас задач нелінійного програмування ширший за клас задач лінійного програмування.

Розглянемо основні відмінності задач нелінійного програмування від задач лінійного програмування.

По-перше, в задачах лінійного програмування область допустимих розв'язків обов'язково опукла, оптимальний розв'язок шукається на скінченній множині крайніх точок; в задачах нелінійного програмування вказана область може бути не опуклою і мати нескінченно велику кількість крайніх точок.

По-друге, нелінійна цільова функція може приймати екстремальні значення як на границі, так і в середині допустимої області, причому може мати декілька локальних та глобальних екстремумів. Разом з тим в задачах лінійного програмування локальний максимум або мінімум обов'язково є і глобальним.

Означення 1. Цільова функція має в деякій точці допустимої області глобальний (абсолютний) мінімум (максимум), якщо в жодній іншій точці вказаної області вона не приймає меншого (більшого) значення.

Означення 2. Цільова функція має локальний мінімум (максимум) в деякій точці X^* , якщо в жодній із точок достатньо близьких до вказаної, вона не приймає меншого (більшого) значення. Іншими словами, X^* є точкою локального мінімуму функції $F(X)$, якщо

$$F(X^*) \leq F(X)$$

для всіх точок X , які задовольняють умові

$$|X - X^*| < \varepsilon,$$

де ε – як завгодно мале додатне число.

Очевидно, в задачах нелінійного програмування глобальний мінімум (максимум) збігається з одним із локальних, обернене твердження в загальному випадку неправильне. В задачах класу, що розглядається, необхідно перевіряти на екстремум як граничні, так і внутрішні точки

допустимої області і, крім того, перевіряти, чи є глобальним знайдений екстремум.

На сьогоднішній день не розроблені загальні методи, які дозволяють розв'язувати будь-які задачі нелінійного програмування, існує низка окремих методів, призначених для розв'язування задач якогось одного типу.

В нелінійному програмуванні особливе місце належить опуклому програмуванню (ОП).

Означення 1. Задача нелінійного програмування, в якій цільова функція є опуклою донизу (вгору), а допустима область D – опуклою множиною, називається задачею опуклого програмування.

Означення 2. Опуклим програмуванням називається розділ нелінійного програмування, предметом якого є вивчення методів пошуку мінімуму (максимуму) опуклої донизу (вгору) функції на опуклій множині.

Означення 3. Множина W , називається опуклою, якщо для будь-яких $X' \in W$, $X'' \in W$ та $\lambda \in [0, 1]$ випливає, що

$$(1 - \lambda) \cdot X' + \lambda \cdot X'' \in W.$$

Геометрично це означає: якщо дві точки X' та X'' належать деякій множині W , то їй же належить будь-яка точка відрізка, який з'єднує вказані точки X' та X'' .

Приклади опуклих множин: відрізок, опуклі багатогранники, n -вимірна куля. Зауважимо, що n -вимірна сфера $(|X - X^0| = r)$ не є опуклою множиною.

Як приклад на рисунку 6.2 зображена допустима область, яка визначається обмеженнями

$$(x_1 - 1)^2 + (x_2 - 1)^2 \leq 1,$$

$$x_1 \cdot x_2 \leq 1.$$

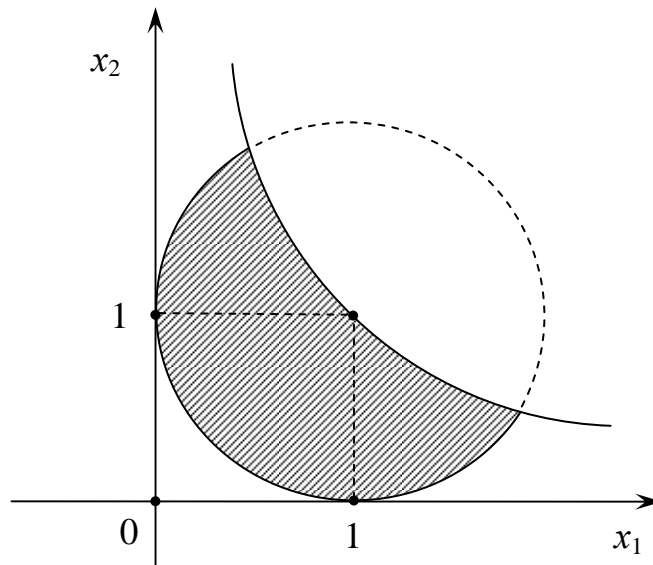


Рис. 6. 2.

Із зображеного рисунка видно, що допустима область (вона заштрихована) не є опуклою, оскільки відрізок, що з'єднує будь-які дві точки на гіперболі, не належить цій області.

Зауважимо також, що всі точки, які лежать на дузі кола, яке обмежує допустиму область, є крайніми. Отже, будь-яка задача нелінійного програмування з допустимою областю, що розглядається, має нескінченну множину крайніх точок.

Означення. Функція $F(X)$, яка задана на опуклій множині W , називається опуклою донизу (або просто опуклою), якщо для будь-яких двох точок X' та X'' цієї множини та $0 \leq \lambda \leq 1$ виконується нерівність

$$F(X) \leq \lambda F(X'') + (1 - \lambda) \cdot F(X'), \quad (6.1.4)$$

де

$$X = \lambda X'' + (1 - \lambda) \cdot X', \quad (6.1.5)$$

та угнутою, якщо

$$F(X) \geq \lambda F(X'') + (1 - \lambda) \cdot F(X'), \quad (6.1.6)$$

Означення. Функція називається строго опуклою і відповідно строго угнутою, якщо для неї виконується строга нерівність виду (6.1.4) або(6.1.6).

Геометрична інтерпретація: поверхня $Z = F(X)$ – опукла донизу, якщо відрізок, що з'єднує будь-які дві її точки, лежить на ній або вище від неї.

Рисунок 6.3 ілюструє виконання нерівності (6.1.4) для будь-якої точки $x \in [x_1, x_2]$ функції однієї змінної $y = f(x)$.

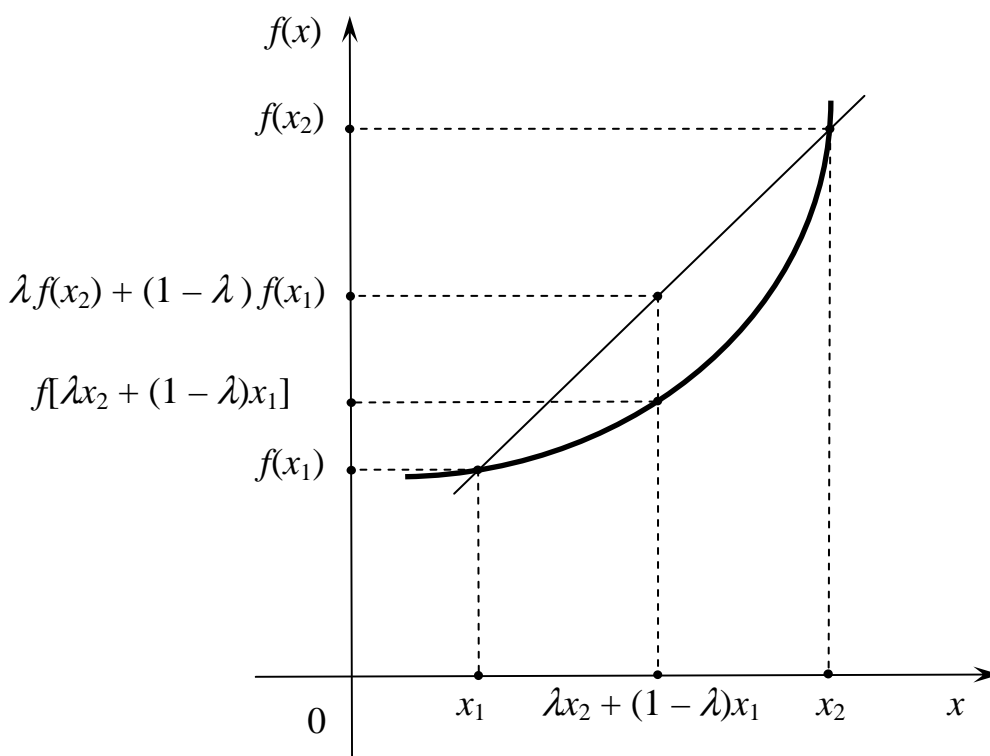


Рис. 6. 3.

За допомогою наведених означень нелегко з'ясувати опукла чи угнута та чи інша функція. В зв'язку з цим використовуються також і інші критерії, зокрема наступний: двічі диференційована функція $F(X)$ є строго угнутою в околі точки X^0 , якщо всі непарні визначники із визначників типу (6.1.7) і т. ін. від'ємні, при цьому визначники парного порядку є додатними, і строго опуклою, – якщо вказані вище визначники (і парного і

непарного порядків) додатні. Тут $F_{ij}(X^0)$ – частинні похідні другого порядку в точці X^0 .

$$F_{11}(X^0), \begin{vmatrix} F_{11}(X^0) & F_{12}(X^0) \\ F_{21}(X^0) & F_{22}(X^0) \end{vmatrix}, \begin{vmatrix} F_{11}(X^0) & F_{12}(X^0) & F_{13}(X^0) \\ F_{21}(X^0) & F_{22}(X^0) & F_{23}(X^0) \\ F_{31}(X^0) & F_{32}(X^0) & F_{33}(X^0) \end{vmatrix} \quad (6.1.7)$$

Якщо $F(X)$ опукла (угнута) для всіх $X \in W$, то кажуть, що вона опукла (угнута) всюди на множині W .

Приклад. Дослідити функцію

$$Z = x_1^2 - x_1x_2 + 5x_2^2 \quad (6.1.8)$$

на опуклість (угнутість) в околі точки $X^0 = (0;1)$.

Розв'язання. Складемо та обчислимо визначники:

$$F_{11}(X^0) = 2 > 0, \begin{vmatrix} F_{11}(X^0) & F_{12}(X^0) \\ F_{21}(X^0) & F_{22}(X^0) \end{vmatrix} = \begin{vmatrix} 2 & -1 \\ -1 & 10 \end{vmatrix} = 22 > 0.$$

Легко бачити, що функція (6.1.8) є строго опуклою, причому не тільки в точці X^0 , але й для всіх значень змінних x_1 та x_2 .

Зауваження. Суми опуклих (угнутих) на множині W функцій є опуклою (угнутою) функцією.

Теорема (без доведення). В задачі опуклого програмування локальний екстремум обов'язково є і глобальним.

Іншими словами, оскільки цільова функція опукла (угнута), в середині допустимої опуклої області може існувати тільки один мінімум (максимум).

Завдяки вказаній особливості задач опуклого програмування в них суттєво спрощується процедура визначення глобального екстремуму у порівнянні із загальним випадком нелінійного програмування.

Квадратичне програмування

Задачами квадратичного програмування називаються ті задачі нелінійного математичного програмування, в яких функції обмежень є лінійними, а цільова функція є сумою лінійної та квадратичної функцій

$$F(x_1, x_2, \dots, x_n) = \sum_{j=1}^n c_j x_j + \sum_{i=1}^n \sum_{j=1}^n d_{ij} x_i x_j. \quad (6.1.9)$$

До найбільш простого класу задач квадратичного програмування відносяться задачі опуклого квадратичного програмування. Можна довести, що у випадку, коли всі коефіцієнти d_{ij} невід'ємні (не додатні) або, що теж саме, матриця $D = \|d_{ij}\|$, $i, j = 1, 2, \dots, n$ невід'ємна (не додатна), то цільова функція $F(X)$ опукла (угнута) і задача квадратичного програмування є задачею опуклого (угнутого) квадратичного програмування.

Підсумовуючи сказане, перелічимо засоби, за допомогою яких можна дослідити функцію $F(X)$ на опуклість (угнутість):

- 1) безпосередньо з використанням означень опуклої (угнутої) функції, заданої на множині W ;
- 2) з використанням визначників, складених із частинних похідних другого порядку $F_{ij}(X^0)$, знайдених у точках області, яка аналізується;
- 3) з використанням (у випадку квадратичного програмування) матриці D коефіцієнтів квадратичної форми.

Теорема Куна Танкера.

Ця теорема посідає центральне місце в теорії нелінійного програмування.

Розглянемо задачу нелінійного програмування (6.1.1) – (6.1.3). Складемо функцію Лагранжа у такий спосіб:

$$L(X, \Lambda) = F(X) + \sum_{i=1}^m \lambda_i q_i(X). \quad (6.1.10)$$

Теорема (без доведення). Точка X^0 є оптимальним розв'язком задачі нелінійного програмування тоді і тільки тоді, коли існує така точка Λ^0 , що для будь-яких $X \geq 0$ і $\Lambda \geq 0$ виконується нерівність

$$L(X, \Lambda^0) \leq L(X^0, \Lambda^0) \leq L(X^0, \Lambda). \quad (6.1.11)$$

Точка (X^0, Λ^0) називається сідловою точкою.

Теорема, що розглядається, дозволяє звести задачу опуклого програмування до задачі пошуку сідлової точки функції Лагранжа, яка може бути розв'язана відомими графічними методами.

6.2 Основні методи розв'язування задач опуклого програмування

6.2.1 Класичний метод визначення екстремуму

Для розв'язання деяких найпростіших задач опуклого програмування не виникає потреби в застосуванні методів математичного програмування; вони можуть бути розв'язані методами математичного аналізу.

Наведемо приклад. Знайти найбільше значення функції

$$y = -x^2 + 2x + 5 \text{ за умови, що } x \in [0; 3].$$

Із математичного аналізу відомо, що найбільше значення диференційованої функції може бути або в точці, в якій похідна функції дорівнює нулю (легко бачити, що такою точкою у наведеному прикладі є точка $x = 1$), або ж на кінцях інтервалу, тобто при $x = 0$ або $x = 3$.

Знайдемо значення функції в указаних трьох точках $y(1) = 6$, $y(0) = 5$, $y(3) = 2$. Отже, найбільше значення функція має при $x = 1$. Задача розв'язана.

Причини надзвичайно легкого розв'язання полягають в тому, що функція залежить лише від однієї змінної, крім того, вона диференційована на всьому інтервалі $[0; 3]$ та задана надзвичайно простим аналітичним виразом.

Для розв'язування реальних практичних задач, в яких цільові функції залежать від великого числа змінних, класичний математичний аналіз, як правило, не придатний.

6.2.2 Метод множників Лагранжа

Розглянемо задачу математичного програмування: необхідно знайти точку $X = (x_1, x_2, \dots, x_n)$, яка мінімізує (максимізує) цільову функцію

$$Z = F(x_1, x_2, \dots, x_n) \quad (6.2.1)$$

за обмежень

$$q_i(x_1, x_2, \dots, x_n) = 0, \quad i = 1, 2, \dots, m. \quad (6.2.2)$$

Припустимо, що функції $F(X)$ та $q_i(X)$ є неперервними разом зі своїми частинними похідними першого порядку.

Особливість задачі, яка розглядається, в тому, що обмеження задані рівняннями. Метод множників Лагранжа є класичним методом знаходження умовного екстремуму функції кількох змінних. Сутність його в тому, що задача знаходження умовного екстремуму функції (6.2.1) зводиться до задачі знаходження безумовного екстремуму функції Лагранжа

$$L(x_1, x_2, \dots, x_n, \lambda_1, \lambda_2, \dots, \lambda_m) = F(x_1, x_2, \dots, x_n) + \sum_{i=1}^m \lambda_i q_i(x_1, x_2, \dots, x_n), \quad (6.2.3)$$

де λ_i – невизначені множники Лагранжа.

У відповідності з цим методом знаходять похідні

$$\frac{\partial L}{\partial x_j} \quad (j = 1, 2, \dots, n) \text{ та } \frac{\partial L}{\partial \lambda_i} \quad (i = 1, 2, \dots, m),$$

потім їх дорівнюють нулю

$$\begin{cases} \frac{\partial L}{\partial x_j} = \frac{\partial F(X)}{\partial x_j} + \sum_{i=1}^m \lambda_i \frac{\partial q_i(X)}{\partial x_j} = 0 & j = 1, 2, \dots, n, \\ \frac{\partial L}{\partial \lambda_i} = q_i(x_1, x_2, \dots, x_n) = 0, & i = 1, 2, \dots, m. \end{cases} \quad (6.2.4)$$

Доведено, що у випадку, коли функція $F(X)$ має екстремум у точці $X^0 = (x_1^0, x_2^0, \dots, x_n^0)$, то існує таке значення $\Lambda^0 = (\lambda_1^0, \lambda_2^0, \dots, \lambda_m^0)$, що точка (X^0, Λ^0) є розв'язком системи рівнянь (6.2.4).

В результаті розв'язання системи рівнянь (6.2.4) в загальному випадку знаходиться множина точок, в яких цільова функція може мати екстремальні значення. Потім обчислюються значення функції $F(x_1, x_2, \dots, x_n)$ в кожній із знайдених точок. В задачах опуклого програмування система (6.2.4) може мати лише один розв'язок, оскільки строго опукла (угнута) на всій допустимій області функція $F(X)$ має тільки один локальний мінімум (максимум), який є і глобальним.

Приклад. Знайти точку умовного екстремуму функції $z = x_1^2 + x_2^2$ за обмеженням $x_1 + x_2 = 2$.

Розв'язання. Складемо функцію Лагранжа

$$L(x_1, x_2, \lambda) = x_1^2 + x_2^2 + \lambda(x_1 + x_2 - 2).$$

Знайдемо похідні по змінним x_1, x_2, λ та дорівняємо їх нулю. В результаті одержимо таку систему рівнянь:

$$\begin{cases} \frac{\partial L}{\partial x_1} = 2x_1 + \lambda = 0, \\ \frac{\partial L}{\partial x_2} = 2x_2 + \lambda = 0, \\ \frac{\partial L}{\partial \lambda} = x_1 + x_2 - 2 = 0. \end{cases}$$

Розв'язок системи рівнянь має вид $\lambda = -2$, $x_1^0 = 1$, $x_2^0 = 1$. Цільова функція дорівнює $Z(x_1^0, x_2^0) = 2$.

Оскільки коефіцієнти при x_1^0 та x_2^0 у виразі для функції Z додатні ($d_{11} = 1 > 0$, $d_{22} = 1 > 0$), то цільова функція опукла. Отже, знайдений екстремум є мінімумом

$$X_{min}^0 = (1; 1), Z_{min}(1; 1) = 2.$$

Зауваження 1. Метод множників Лагранжа може використовуватись також і в тих випадках, коли деякі обмеження мають вид нерівностей. У вказаних випадках застосуванню метода передуює перетворення обмежень-нерівностей в рівняння. Це здійснюється шляхом введення додаткових змінних, на які накладається умова невід'ємності.

Зауваження 2. Пошук розв'язку методом множників Лагранжа задачі нелінійного програмування загального виду значно складніший, ніж в частинному випадку задачі опуклого програмування, дуже часто спряжений з великим обсягом обчислень. Більше того, інколи трапляються випадки, коли екстремальні точки явно існують, але система рівнянь (6.2.4) розв'язків не має.

6.2.3 Метод наближеного розв'язання задач із сепарабельними функціями

Означення. Функція n змінних $F(x_1, x_2, \dots, x_n)$ називається сепарабельною, якщо вона є сумою n функцій, кожна з яких залежить тільки від однієї змінної

$$F(x_1, x_2, \dots, x_n) = F_1(x_1) + F_2(x_2) + \dots + F_n(x_n). \quad (6.2.5)$$

Розглянемо наступну задачу опуклого програмування. Необхідно знайти точку $X = (x_1, x_2, \dots, x_n)$, координати якої мінімізують цільову функцію

$$F(x_1, x_2, \dots, x_n) = \sum_{j=1}^n F_j(x_j) \quad (6.2.6)$$

за умов

$$\sum_{j=1}^n q_{ij}(x_j) \begin{cases} \geq \\ = \\ \leq \end{cases} b_i, \quad i = 1, 2, \dots, m. \quad (6.2.7)$$

При цьому припускається, що цільова функція опукла, допустима множина точок X також опукла.

Розв'язування задач виду, що розглядається, базується на кусково-лінійній апроксимації функцій однієї змінної $F_j(x_j)$, $q_{ij}(x_j)$ та наступному зведенні задачі опуклого програмування до задачі лінійного програмування, яка може бути розв'язана симплексним методом.

Процедуру апроксимації пояснимо на прикладі деякої функції однієї змінної $y(x)$, неперервної на інтервалі $[a, c]$. Вказаний інтервал розіб'ємо на r підінтервалів (рис. 6.4) так, щоб $x_0 = a \leq x_1 \leq x_2 \leq \dots \leq x_r = c$. Точці x_k відповідає значення функції $y(x_k) = y_k$.

Точки (x_0, y_0) , (x_1, y_1) , \dots , (x_k, y_k) , (x_{k+1}, y_{k+1}) , \dots , (x_r, y_r) з'єднаємо відрізками прямих, одержимо ламану лінію. Кусково-лінійна функція $\hat{y}(x)$ апроксимує функцію $y(x)$ на інтервалі $[a, c]$. Очевидно, що збільшуючи

число підінтервалів розбиття, можна забезпечити будь-яку, як завгодно, високу, точність апроксимації.

Знайдемо аналітичний вираз для функції $\hat{y}(x)$. Неважко бачити, що ця функція на інтервалі $[x_k, x_{k+1}]$ має вид

$$\hat{y}(x) = y_k + \frac{y_{k+1} - y_k}{x_{k+1} - x_k} \cdot (x - x_k). \quad (6.2.8)$$

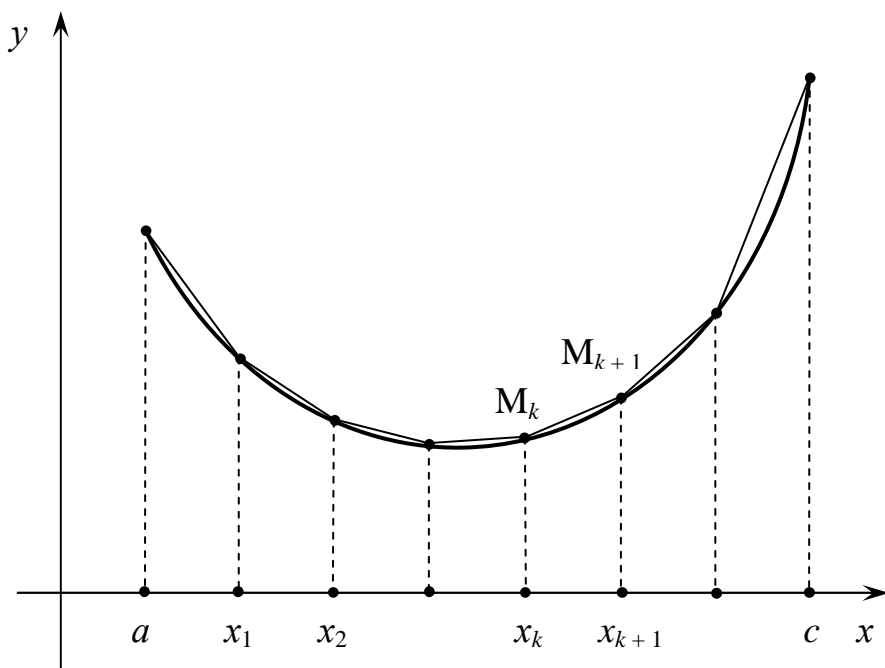


Рис. 6. 4.

Рівняння (6.2.8) є рівнянням прямої, яка проходить через дві точки $M_k(x_k, y_k)$ та $M_{k+1}(x_{k+1}, y_{k+1})$. На цій прямій виберемо довільну точку $M(x, y)$, розміщену між точками M_k та M_{k+1} . позначимо через λ відношення

$$\frac{M_k M}{M_k M_{k+1}} = \lambda, \quad (0 \leq \lambda \leq 1).$$

Запишемо очевидну рівність

$$\frac{x - x_k}{x_{k+1} - x_k} = \lambda. \quad (6.2.9)$$

Звідси маємо

$$x = \lambda x_{k+1} + (1 - \lambda)x_k. \quad (6.2.10)$$

Вираз (6.2.9) перепишемо так:

$$x - x_k = \lambda(x_{k+1} - x_k). \quad (6.2.11)$$

Підставивши вираз (6.2.11) у вираз (6.2.8), одержимо

$$\hat{y}(x) = \lambda y_{k+1} + (1 - \lambda)y_k. \quad (6.2.12)$$

Позначимо

$$\lambda = \lambda_{k+1}, \quad 1 - \lambda = \lambda_k. \quad (6.2.13)$$

З урахуванням прийнятих позначень вирази (6.2.10) та (6.2.12) набудуть виду

$$x = \lambda_k x_k + \lambda_{k+1} x_{k+1}. \quad (6.2.14)$$

$$\hat{y}(x) = \lambda_k y_k + \lambda_{k+1} y_{k+1}, \quad (6.2.15)$$

де

$$\lambda_k + \lambda_{k+1} = 1, \quad \lambda_k \geq 0, \quad \lambda_{k+1} \geq 0. \quad (6.2.16)$$

Узагальнивши одержані результати на весь інтервал $[a, c]$, запишемо

$$x = \sum_{k=0}^r \lambda_k x_k, \quad (6.2.17)$$

$$\hat{y}(x) = \sum_{k=0}^r \lambda_k y_k, \quad (6.2.18)$$

де

$$\sum_{k=0}^r \lambda_k = 1, \quad \lambda_k \geq 0, \quad k = 0, 1, \dots, r. \quad (6.2.19)$$

При цьому припускаємо, що відмінними від нуля можуть бути не більше ніж два сусідні коефіцієнти λ_k . Тоді (6.2.18) є аналітичним виразом кусково-лінійної функції.

Повернемося до задачі лінійного програмування, що розглядається, з великим числом функцій однієї змінної x_j ($j = 0, 1, \dots, n$). Припускаємо, що інтервали значень змінних x_1, x_2, \dots, x_n відомі, функції $F_j(x_j), q_{ij}(x_j)$ неперервні на відповідних інтервалах.

Інтервали значень кожної змінної x_j розіб'ємо на r_j підінтервалів виду $[x_{kj}, x_{k+1, j}]$. Скористаємося викладеним способом апроксимації функцій однієї змінної. Тоді кусково-лінійні апроксимації $\hat{F}_j(x_j)$, $\hat{q}_{ij}(x_j)$ функцій $F_j(x_j)$, $q_{ij}(x_j)$ за аналогією з (6.2.17) – (6.2.19) можна записати у вигляді

$$\hat{F}_j(x_j) = \sum_{k=0}^{r_j} \lambda_{kj} F_{kj}, \quad (6.2.20)$$

$$\hat{q}_{ij}(x_j) = \sum_{k=0}^{r_j} \lambda_{kj} q_{kij}, \quad (6.2.21)$$

$$x_j = \sum_{k=0}^{r_j} \lambda_{kj} x_{kj}, \quad (6.2.22)$$

$$\sum_{k=0}^{r_j} \lambda_{kj} = 1, \quad \lambda_{kj} \geq 0 \text{ для всіх } k, j; \quad j = 1, 2, \dots, n, \quad k = 1, 2, \dots, m. \quad (6.2.23)$$

Тут

$$F_{kj} = F_j(x_{kj}), \quad q_{kij} = q_{ij}(x_{kj}). \quad (6.2.24)$$

Замість задачі математичного програмування розглянемо наближену задачу

$$\min \hat{F} = \min \sum_{j=1}^n \hat{F}_j(x_j), \quad (6.2.25)$$

за умов

$$\hat{q}_{ij}(x_j) = \sum_{j=1}^n q_{ij}(x_j) \begin{cases} \geq \\ = \\ \leq \end{cases} b_i, \quad i = 1, 2, \dots, m, \quad j = 1, 2, \dots, n; \quad (6.2.26)$$

$$x_j \geq 0. \quad (6.2.27)$$

Після підстановки виразів (6.2.20), (6.2.21) у (6.2.25) та (6.2.26) одержимо наближену задачу, яка відрізняється від початкової складом невідомих величин (замість x_j використовуються змінні λ_{kj}):

$$\min \hat{F} = \min \sum_{j=1}^n \sum_{k=0}^{r_j} F_{kj} \lambda_{kj}, \quad (6.2.28)$$

$$\sum_{j=1}^n \sum_{k=0}^{r_j} q_{kij} \lambda_{kj} \begin{cases} \leq \\ = \\ \geq \end{cases} b_i, \quad i = 1, 2, \dots, m, \quad (6.2.29)$$

$$\sum_{k=0}^{r_j} \lambda_{kj} = 1, \quad (6.2.30)$$

$$\lambda_{kj} \geq 0 \text{ для всіх } k, j; \quad k = 1, 2, \dots, m, \quad j = 1, 2, \dots, n. \quad (6.2.31)$$

Тут $\lambda_{kj} \geq 0$, F_{kj}, q_{kij} – деякі числа.

Зазначимо, що для даного j можуть бути додатними не більше за два сусідні значення λ_{kj} . Вказана величина повинна враховуватись при виборі базису.

Отже, задача опуклого програмування з сепарабельними функціями зведена до задачі лінійного програмування, яку можна розв'язати симплексним методом.

Оскільки мінімум опуклої функції на опуклій множині може бути тільки один, то локальний мінімум збігається з глобальним.

Задача лінійного програмування, що одержана в результаті викладеної заміни, має точний розв'язок, який є наближеним розв'язком задачі опуклого програмування. При цьому можна забезпечити яку завгодно високу точність цього розв'язку шляхом відповідного розбиття інтервалів зміни змінних x_1, x_2, \dots, x_n . Більш детально ознайомитись з цим методом можна, наприклад у [21].

Зауваження 1. Функції початкової задачі нелінійного програмування можуть бути не сепарабельними. Проте часто задачу можна звести до виду (6.2.6), (6.2.7). Для цього існує низка практичних прийомів, що базується на переході до нових змінних. Наприклад, нехай до цільової функції входить добуток $x_i \cdot x_j$ (функція не сепарабельна). Якщо перейти до нових змінних

$$y_i = \frac{1}{2}(x_i + x_j), \quad y_j = \frac{1}{2}(x_i - x_j),$$

то одержимо $x_i \cdot x_j = y_i^2 - y_j^2$.

Отже, цільова функція стала сепарабельною відносно нових змінних y_i та y_j .

Зауваження 2. В результаті апроксимації різних сепарабельних функцій задача виду (6.2.6), (6.2.7) навіть невеликої вимірності може звестись до задачі лінійного програмування (6.2.28) – (6.2.31) дуже великої вимірності. У зв'язку з цим часто має сенс спочатку розв'язати задачу при грубому розбитті інтервалів зміни змінних x_i (одержати «грубий» розв'язок), потім у його околі здійснити розбиття на маленькі інтервали та уточнити розв'язок.

6.2.4 Градієнтні методи

Найбільш поширеними методами розв'язування задач нелінійного програмування, в тому числі опуклого, є градієнтні методи, яких створено багато. Спочатку розглянемо найпростіший з них, який застосовується для розв'язування задач на безумовний екстремум.

Розв'язування задач на безумовний екстремум

Нехай задано диференційовану функцію $F(X)$. Необхідно знайти безумовний максимум (мінімум) цієї функції, тобто екстремум без обмежень на X .

Для розв'язання цієї задачі виберемо деяку точку X^0 . Від неї будемо рухатись в напрямку зростання функції $F(X)$. Напрямок руху визначимо вектором $\vec{l} = (l_1, l_2, \dots, l_n)$.

Вважаємо, що в цьому напрямку просунулись на величину кроку h_0 . Тоді координати нової точки X' мають вид

$$x'_j = x_j^0 + h_0 \cdot l_j, \quad j = 1, 2, \dots, n. \quad (6.2.32)$$

У відповідності з формулою Тейлора значення функції, що розглядається, у точці X' наближено дорівнює

$$F(X') \approx F(X) + h(\overline{\nabla F} \cdot \vec{l})^* \quad (6.2.33)$$

де $\overline{\nabla F}$ – градієнт функції $F(X)$ у точці X^0 , який визначається виразом

$$\overline{\nabla F} = \left(\frac{\partial F}{\partial x_1}, \frac{\partial F}{\partial x_2}, \dots, \frac{\partial F}{\partial x_n} \right); \quad (6.2.34)$$

$(\overline{\nabla F} \cdot \vec{l})$ – скалярний добуток векторів $\overline{\nabla F}$ та \vec{l} .

Зауважимо, що градієнт функції $F(X)$ у точці X^0 може позначатися також так:

$$\overrightarrow{\text{grad}F(X^0)}, \text{ або } \frac{\partial F}{\partial X}(X^0).$$

Враховуючи, що

$$(\overline{\nabla F} \cdot \vec{l}) = |\overline{\nabla F}| \cdot |\vec{l}| \cdot \cos \varphi,$$

* Символ ∇ читається „набла”(гр. pabla - арфа) і означає – оператор або оператор Гамільтона

де φ – кут між векторами $\overline{\nabla F}$ та \vec{l} , приходимо до висновку про те, що приріст функції $F(X)$ приймає найбільше значення при $\varphi = 0$, тобто, якщо за напрямок руху вибраний напрямок градієнта функції

$$\vec{l} = \overline{\nabla F}(X^0). \quad (6.2.35)$$

З урахуванням сказаного вираз (6.2.32) набуде вигляду

$$x'_j = x_j^0 + h_0 \cdot \overline{\nabla F}_j(X^0), \quad j = 1, 2, \dots, n. \quad (6.2.36)$$

Використовуючи знайдене значення X' , зробимо ще один крок в напрямку градієнта функції $F(X)$ у точці X' . В результаті одержимо

$$x''_j = x'_j + h_1 \cdot \overline{\nabla F}_j(X'), \quad j = 1, 2, \dots, n. \quad (6.2.37)$$

де h_1 – розмір кроку на другій ітерації. Продовжуючи процедуру на $(k + 1)$ -ій ітерації маємо

$$x_j^{(k+1)} = x_j^{(k)} + h_k \cdot \overline{\nabla F}_j(X^{(k)}) \quad (6.2.38)$$

або

$$x_j^{(k+1)} = x_j^{(k)} + h_k \frac{\partial F}{\partial x_j}(X^{(k)}), \quad j = 1, 2, \dots, n, \quad (6.2.39)$$

де h_k – розмір кроку на k -ій ітерації; $X^{(k)}$ – положення точки, що рухається, після здійснення k -ої ітерації.

Ітераційний процес продовжується до тих пір, поки похідні $\frac{\partial F}{\partial x_j}$,

$j = 1, 2, \dots, n$ стануть достатньо малими, оскільки необхідною умовою наявності екстремуму диференційованої функції є рівність нулю вказаних похідних.

Критерій завершення пошуку екстремуму може мати вид

$$\sum_{j=1}^n \left[\frac{\partial F}{\partial x_j}(X^{(k)}) \right]^2 < \delta, \quad (6.2.40)$$

де δ – як завгодно мале додатне число, яке визначає точність розв’язання екстремальної задачі, що розглядається.

Інколи пошук екстремуму припиняється, якщо зміна цільової функції не перевищує деякого наперед заданого значення.

Відмітимо, що розмір кроку може вибиратися незмінним (сталим). Якщо крок малий, то збіжність градієнтного методу буде повільною. Найчастіше крок вибирається так:

$$h_k = \frac{h_0}{k}.$$

Природно, що у випадку мінімізації функції рух необхідно здійснювати в напрямку протилежному напрямку градієнта (кажуть, що використовується антиградієнт).

$$x_j^{(k+1)} = x_j^{(k)} - h_k \cdot \overline{\nabla F}_j(X^{(k)}), \quad j = 1, 2, \dots, n. \quad (6.2.41)$$

Метод найшвидшого підйому (спуску)

або повноходовий метод

Метод, що розглядається, є одним із градієнтних і полягає у наступному. Вибирається початкова точка, яка є деяким допустимим розв’язком задачі. В ній визначається градієнт цільової функції $F(X)$, потім здійснюється крок у напрямку знайденого градієнта (або в напрямку антиградієнта у випадку мінімізації). Якщо значення функції $F(X)$ при цьому зросло (або зменшилось у випадку задачі мінімізації), то здійснюється наступний крок в цьому ж напрямку і т. д. до тих пір, поки у вибраному напрямку буде знайдено максимум (або відповідно мінімум), точніше, поки цільова функція досягне максимуму (мінімуму) на промені, який виходить із початкової точки. У знайденій точці визначається новий градієнт, у напрямку якого знову-таки здійснюється деяке число кроків

тощо. Координати точки, що рухається, при цьому визначаються за формулою

$$x_j^{(k+1)} = x_j^{(k)} - h \cdot \frac{\partial F}{\partial x_j}(\mathbf{X}^p), \quad j = 1, 2, \dots, n, \quad (6.2.42)$$

де h – розмір кроку, \mathbf{X}^p – точка, в якій здійснювався «останній» розрахунок градієнта цільової функції.

Різниця між методом, що розглядається, та раніше викладеним методом для випадку безумовної оптимізації полягає в тому, що в методі найшвидшого підйому градієнт обчислюється не в кожній точці, не на кожному кроці (в загальному випадку декілька кроків здійснюється з використанням одного й того ж градієнта).

Для завершення процесу пошуку оптимуму часто, окрім вказаного вище (6.2.40), використовується наступний критерій:

$$\sqrt{\sum_{j=1}^n (x_j^M - x_j^N)^2} < \delta, \quad (6.2.43)$$

де x_j^M , x_j^N – j -ті координати відповідно початкової та кінцевої точок останнього здійсненого етапу підйому (спуску);

δ – деяке наперед задане, достатньо мале додатне число, яке характеризує точність розв'язання екстремальної задачі.

Розв'язування задач з обмеженнями

Розглянемо задачу нелінійного математичного програмування у такій постановці. Необхідно знайти точку $\mathbf{X} = (x_1, x_2, \dots, x_n)$, яка максимізує (мінімізує) цільову функцію

$$Z = F(\mathbf{X}) \quad (6.2.44)$$

за умов

$$q_i(\mathbf{X}) \geq 0, \quad i = 1, 2, \dots, m, \quad (6.2.45)$$

$$X \geq 0. \tag{6.2.46}$$

Припускається, що функція $F(X)$ є угнутою (опуклою).

Відмітимо, що у випадку, коли точка екстремуму знаходиться всередині допустимої області, яка визначається обмеженнями, процес пошуку розв'язку по суті такий саме, як і в задачах на безумовний екстремум. Вказаний процес суттєво ускладнюється, якщо оптимум знаходиться за межами допустимої області або на її границі. Один з підходів до розв'язування задач, що розглядаються, полягає у сполученні методів так званих штрафних функцій з градієнтними методами.

Методи штрафних функцій – це клас методів розв'язування задач нелінійного програмування, в тому числі опуклого; за допомогою яких задача оптимізації функції $F(X)$ за наявності обмежень зводиться до задачі на безумовний екстремум деякої допоміжної функції

$$R(X) = F(X) + H(X), \tag{6.2.47}$$

де $H(X)$ – функція, що є своєрідний «штраф» за порушення обмежень задачі (штрафна функція).

Якщо розв'язання задачі, що розглядається, інтерпретувати як пошук найвищої (найменшої) точки «місцевості за огорожею», то введення штрафної функції можна уявити як появу сил, які «притягнуть» точку до огорожі, якщо вона, рухаючись у пошуку оптимуму, опинилася за межами допустимої області.

Припустимо, що $H(X)$ є угнутою (опуклою) функцією, яка задовольняє такій умові: вона дорівнює нулю, якщо точка X , що рухається, знаходиться всередині допустимої області або на її границі, та відмінна від нуля, якщо X – за межами вказаної області, причому $H(X) < 0$ в задачах максимізації та $H(X) > 0$ в задачах мінімізації цільової функції.

Існують різні варіанти вибору штрафної функції. Зокрема, доведено, що її можна вибрати у такому вигляді:

$$H(X) = \sum_{i=1}^m \alpha_i(X) q_i(X), \quad (6.2.48)$$

де

$$\alpha_i(X) = \begin{cases} 0, & \text{якщо } q_i(X) \geq 0; \\ \alpha, & \text{якщо } q_i(X) < 0, \end{cases} \quad i = 1, 2, \dots, m \quad (6.2.49)$$

де α – деяке додатне число, воно вибирається так, щоб за межами допустимої області градієнт функції $R(X)$ був напрямлений в бік цієї області.

Викладений метод дозволяє звести задачу умовної оптимізації функції $F(X)$ до задачі безумовної оптимізації допоміжної угнутої (опуклої) функції $R(X)$, для розв'язування якої можуть застосовуватися розроблені градієнтні методи, зокрема, метод, викладений вище.

Розглянемо алгоритм двох градієнтних методів: спочатку прямого градієнтного методу, потім на конкретному прикладі алгоритм методу Ерроу-Гурвиця.

Алгоритм прямого градієнтного методу має вид

$$x_j^{(k+1)} = \max \left\{ 0; x_j^{(k)} + h \left[\frac{\partial F}{\partial x_j} (X^{(k)}) + \sum_{i=1}^m \alpha_i (X^{(k)}) \frac{\partial q_i}{\partial x_j} (X^{(k)}) \right] \right\}, \quad (6.2.50)$$

де $X^{(k)}$ – точка, що знайдена на k -ій ітерації.

Легко бачити, що поки точка $X^{(k)}$, що рухається, знаходиться в межах допустимої області, приріст координат визначається градієнтом цільової функції (другий доданок в квадратних дужках виразу (6.2.50) дорівнює нулю). Іншими словами, процес пошуку оптимуму відбувається так само, як і у випадку відсутності обмежень. Порушення хоча б одного з обмежень, наприклад i -го на k -му кроці обумовлює появу відмінного від нуля коефіцієнта $\alpha_i(X^{(k)})$. Отже, в цьому випадку приріст координат

точки X буде визначатись з урахуванням і другого доданку в квадратних дужках алгоритму (6.2.50), який, як було сказано вище, спрямовує рух точки $X^{(k)}$ в допустиму область. Словом, штрафна функція спрямовує пошук оптимуму у вказану область.

Пошук розв'язку задачі рекомендується починати з відносно невеликих значень коефіцієнта α . Потім, знайшовши грубе значення $X^{\text{опт}}$, доцільно збільшити α та уточнити розв'язок.

Для завершення пошуку розв'язку застосовуються ті ж критерії, що і у випадку безумовної оптимізації.

Приклад. Необхідно знайти точку $X = (x_1, x_2)$, яка максимізує функцію

$$F(X) = 2x_1 + x_2 - 0,4x_1^2 - x_2^2 \quad (6.2.51)$$

за наявності обмежень

$$\begin{cases} -4x_1 - 2x_2 + 4 \geq 0, \\ -3x_1 - x_2 + 7 \geq 0, \end{cases} \quad (6.2.52)$$

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

Розв'язання. Частинні похідні цільової функції та функцій обмежень мають вид

$$\begin{aligned} \frac{\partial F}{\partial x_1} &= 2 - 0,8x_1, & \frac{\partial F}{\partial x_2} &= 1 - 2x_2, \\ \frac{\partial q_1}{\partial x_1} &= -4, & \frac{\partial q_1}{\partial x_2} &= -2, \\ \frac{\partial q_2}{\partial x_1} &= -3, & \frac{\partial q_2}{\partial x_2} &= -1. \end{aligned} \quad (6.2.53)$$

За початкову вибирається будь-яка точка, яка є допустимим розв'язком задачі, наприклад $X^0 = (0; 0)$. Цільова функція у початковій

точці дорівнює $F(X^0) = 0$. Розмір кроку h та число α штрафної функції приймемо відповідно рівними $h = 0,2$, $\alpha = 1$.

З урахуванням вибраних значень, а також одержаних виразів для похідних, алгоритм градієнтного методу приймає вид

$$\begin{aligned} x_1^{(k+1)} &= \max\left\{0; x_1^{(k)} + 0,2\left[\left(2 - 0,8x_1^{(k)}\right) + \alpha_1(X^{(k)}) \cdot (-4) + \alpha_2(X^{(k)}) \cdot (-3)\right]\right\} \\ x_2^{(k+1)} &= \max\left\{0; x_2^{(k)} + 0,2\left[\left(1 - 2x_2^{(k)}\right) + \alpha_1(X^{(k)}) \cdot (-2) + \alpha_2(X^{(k)}) \cdot (-1)\right]\right\} \end{aligned} \quad (6.2.54)$$

Враховуючи, що в точці X^0 обидва обмеження виконуються, у відповідності з формулою (6.2.49) маємо $\alpha_1(X^0) = 0$, $\alpha_2(X^0) = 0$.

Перший крок. Визначимо координати точки $X^{(1)}$ за формулами (6.2.54)

$$\begin{aligned} x_1^{(1)} &= \max\{0; 0 + 0,2[(2 - 0) + 0 \cdot (-4) + 0 \cdot (-3)]\} = 0,4, \\ x_2^{(1)} &= \max\{0; 0 + 0,2[(1 - 0) + 0 \cdot (-2) + 0 \cdot (-1)]\} = 0,2. \end{aligned} \quad (6.2.55)$$

Перевіримо, чи виконуються в точці $X^{(1)}(0,4; 0,2)$ обмеження задачі

$$-4 \cdot 0,4 - 2 \cdot 0,2 + 4 = 2 > 0,$$

$$-3 \cdot 0,4 - 0,2 + 7 = 5,6 > 0.$$

Видно, що обидва обмеження виконуються. Отже, за формулами (6.2.49) маємо $\alpha_1(X^{(1)}) = 0$, $\alpha_2(X^{(1)}) = 0$. Знаходимо значення функції $F(X)$ в точці $X^{(1)}$

$$F(X^{(1)}) = 2 \cdot 0,4 + 0,2 - 0,4^3 - 0,2^2 = 0,896.$$

Другий крок.

$$\begin{aligned} x_1^{(2)} &= \max\{0; 0,4 + 0,2[(2 - 0,8 \cdot 0,4) + 0]\} = 0,736, \\ x_2^{(2)} &= \max\{0; 0,2 + 0,2[(1 - 2 \cdot 0,2) + 0]\} = 0,32. \end{aligned} \quad (6.2.56)$$

Перевірка показує, що обидва обмеження задачі в точці $X^{(2)}(0,736; 0,32)$ виконуються. З цього випливає

$$\alpha_1(X^{(2)}) = 0, \alpha_2(X^{(2)}) = 0, F(X^{(2)}) \approx 1,47.$$

Третій крок.

$$\begin{aligned}x_1^{(3)} &= \max\{0; 0,736 + 0,2[(2 - 0,8 \cdot 0,736) + 0]\} = 1,018, \\x_2^{(3)} &= \max\{0; 0,32 + 0,2[(1 - 2 \cdot 0,32) + 0]\} = 0,392.\end{aligned}\tag{6.2.57}$$

Перевірка показує, що друге обмеження виконується, а перше ні. Отже,

$$\alpha_1(X^{(3)}) = \alpha = 1, \quad \alpha_2(X^{(3)}) = 0.$$

Оскільки точка $X^{(3)}(1,018; 0,392)$ не належить допустимій області, то цільову функцію в цій точці обчислювати не треба.

Четвертий крок.

$$\begin{aligned}x_1^{(4)} &= \max\{0; 1,018 + 0,2[(2 - 0,8 \cdot 1,018) + 1 \cdot (-4)]\} = 0,455, \\x_2^{(4)} &= \max\{0; 0,392 + 0,2[(1 - 2 \cdot 0,392) + 1 \cdot (-2)]\} = 0,035.\end{aligned}\tag{6.2.58}$$

В результаті здійснення четвертого кроку точка, що рухається, відкинута назад в допустиму область, причому, що надзвичайно небажано, далеко від границі цієї області. Неважко бачити, що віддалення від границі допустимої області залежить від вибору розміру кроку h та коефіцієнта α штрафної функції. За менших значень h та α віддалення від границі буде меншим. Наприклад, якщо значення h залишити без зміни, а α прийняти рівним 0,50 замість 1, то точка $X^{(4)}$ набуде виду

$$\begin{aligned}x_1^{(4)} &= \max\{0; 1,018 + 0,2(-0,814)\} = 0,865, \\x_2^{(4)} &= \max\{0; 0,392 + 0,2(-0,784)\} = 0,235.\end{aligned}\tag{6.2.59}$$

Аналогічно здійснюється подальший пошук екстремуму. Для завершення пошуку можна використати вказаний вище критерій (6.2.40). Викладений метод називається прямим градієнтним методом. Його основний недолік полягає в тому, що рухома точка здійснює значні коливання відносно границі. Вказана особливість називається нестійкістю поблизу границі. Один із способів зменшення вказаних коливань полягає в виборі коефіцієнтів α_i штрафної функції в залежності від того, як

«сильно» порушується те, чи інше обмеження. На цій ідеї базується метод Ерроу-Гурвиця, який відрізняється від прямого градієнтного методу лише способом вибору розмірів штрафів. В ньому коефіцієнти штрафної функції визначаються так:

$$j = 1, 2, \dots, n \quad (6.2.60)$$

де через $\alpha_i^{(k)}$ позначені коефіцієнти α_i на k -му кроці.

Відмітимо, що за початкові значення коефіцієнтів α_i^0 вибираються довільні невід'ємні числа. Прийmemo $\alpha_1^0 = \alpha_2^0 = 0$.

Застосуємо метод Ерроу-Гурвиця для розв'язування задачі, яка розглядається. Легко бачити, що перші три кроки, виконані обома методами, збігаються. Ця обставина пояснюється тим, що рухома точка $X^{(k)}$ знаходиться в допустимій області i , отже, $\alpha_i^{(k)} = 0$ для $k = 1, 2, i = 1, 2$.

В результаті виконання третього кроку одержана точка $X^{(3)} = (1,018; 0,392)$.

Четвертий крок.

Визначимо величини

$$\begin{aligned} q_1(X^{(3)}) &= -4 \cdot 1,018 - 2 \cdot 0,392 + 4 = -0,856 < 0, \\ q_2(X^{(3)}) &= -3 \cdot 1,018 - 0,392 + 7 = 3,554 > 0. \end{aligned} \quad (6.2.61)$$

Тоді за формулами (6.2.60) одержимо

$$\begin{aligned} \alpha_1^{(3)} &= \max\{0; 0 - 0,2(-0,856)\} = 0,171, \\ \alpha_2^{(3)} &= \max\{0; 0 - 0,2 \cdot 3,554\} = 0. \end{aligned} \quad (6.2.62)$$

Коефіцієнт $\alpha_1^{(3)}$ відмінний від нуля, оскільки порушено перше обмеження в результаті виконання третьої ітерації.

Знайдемо координати точки $X^{(4)}$ за формулами (6.2.54)

$$\begin{aligned} x_1^{(4)} &= \max\{0; 1,018 + 0,2[(2 - 0,8 \cdot 1,018) + 0,171 \cdot (-4) + 0]\} = 1,118, \\ x_2^{(4)} &= \max\{0; 0,392 + 0,2[(1 - 2 \cdot 0,392) + 0,171 \cdot (-2) + 0]\} = 0,367. \end{aligned}$$

П'ятий крок.

$$q_1(X^{(4)}) = -4 \cdot 1,118 - 2 \cdot 0,367 + 4 = -1,206 < 0,$$

$$q_2(X^{(4)}) = -3 \cdot 1,118 - 0,367 + 7 = 3,28 > 0;$$

$$\alpha_1^{(4)} = \max\{0; 0,171 - 0,2(-1,206)\} = 0,412,$$

$$\alpha_2^{(4)} = \max\{0; 0 - 0,2 \cdot 3,28\} = 0.$$

Із наведених результатів видно, що із недопустимої точки $X^{(3)}$ перемістились в недопустиму точку $X^{(4)}$. При цьому перше обмеження порушено. Отже зростає значення коефіцієнта $\alpha_1^{(4)}$, від якого залежить «розмір штрафу» за порушення вказаного обмеження.

Визначимо координати точки $X^{(5)}$.

$$x_1^{(5)} = \max\{0; 1,118 + 0,2[(2 - 0,8 \cdot 1,118) + 0,412 \cdot (-4) + 0]\} = 1,010,$$

$$x_2^{(5)} = \max\{0; 0,367 + 0,2[(1 - 2 \cdot 0,367) + 0,412 \cdot (-2) + 0]\} = 0,256.$$

Шостий крок.

$$q_1(X^{(5)}) = -0,552 < 0,$$

$$q_2(X^{(5)}) = 3,714 > 0,$$

$$\alpha_1(X^{(5)}) = \max\{0; 0,412 - 0,2 \cdot (-0,552)\} = 0,522,$$

$$\alpha_2(X^{(5)}) = \max\{0; 0 - 0,2 \cdot 3,714\} = 0;$$

$$x_1^{(6)} = \max\{0; 1,010 + 0,2[(2 - 0,8 \cdot 1,010) + 0,522 \cdot (-4) + 0]\} = 0,831,$$

$$x_2^{(6)} = \max\{0; 0,256 + 0,2[(1 - 2 \cdot 0,256) + 0,522 \cdot (-2) + 0]\} = 0,145.$$

Перевіримо виконання обмежень

$$q_1(X^{(6)}) = -4 \cdot 0,831 - 2 \cdot 0,145 + 4 = 0,386 > 0,$$

$$q_2(X^{(6)}) = -3 \cdot 0,831 - 0,145 + 7 = 5,362 > 0.$$

Отже, точка $X^{(6)}$ належить допустимій області. Обчислимо значення цільової функції в цій точці

$$F(X^{(6)}) = 1,600.$$

Із наведених результатів видно, що в процесі пошуку екстремуму рухома точка зміщується до оптимальної вздовж границі допустимої області, причому далеко за її межі не рухається та сильно не віддаляється всередину цієї області. Подальший пошук розв'язку задачі здійснюється аналогічно.

Зауваження 1. Крім викладених, є велика кількість різних інших методів та алгоритмів, створених для розв'язування задач опуклого квадратичного програмування. Зокрема, метод спряжених градієнтів, алгоритм Гілдрета та інші. Детальніше про це дивись в [2].

Зауваження 2. Поряд з великою кількістю так званих детермінованих методів, в яких траєкторія руху до екстремальної точки є визначеною, існують методи випадкового пошуку екстремуму. В них випадковим чином шукається той напрямок переміщення рухомої точки, в якому функція не зменшується (у випадку задачі максимізації). У знайденому напрямку здійснюються рівні кроки до тих пір, поки приріст цільової функції не змінить знак. У подальшому процес повторюється. Перевага цих методів полягає в їх простоті, а також у відсутності обмежень на цільову функцію. Детально про це викладено, наприклад, в [2].

Питання для самоконтролю

1. Які задачі відносяться до задач опуклого програмування?
2. Які задачі відносяться:
 1. а) до задач квадратичного програмування?
 2. б) до задач опуклого квадратичного програмування?
3. Які задачі опуклого програмування можна розв'язати за допомогою класичного методу множників Лагранжа?

4. В чому полягає сутність методу множників Лагранжа?
5. Зміст теореми Куна-Таккера.
6. В чому полягає метод наближеного розв'язання задач з сепарабельними функціями?
7. В чому сутність градієнтних методів розв'язування задач опуклого програмування?
8. Назвіть етапи розв'язування задач на безумовний екстремум градієнтним методом.
9. В чому сутність методу найшвидшого підйому (спуску)?
10. Що таке штрафна функція та її роль?
11. За якими правилами будуються штрафні функції?
12. В чому полягає основний недолік прямого градієнтного методу?
13. В чому сутність методу Ерроу-Гурвиця?

ГЛАВА 7.

ЗАСТОСУВАННЯ ПАКЕТА MATHCAD ДЛЯ РОЗВ'ЯЗУВАННЯ ЗАДАЧ ЛІНІЙНОГО ПРОГРАМУВАННЯ

Нижче в стислій формі надано основні відомості про роботу в пакеті MathCAD, а також, – і це головне, – комп'ютерна реалізація найголовніших алгоритмів розв'язування задач лінійного програмування:

- розв'язування задач графічним методом;
- алгоритми розв'язування задач симплексним методом;
- алгоритми розв'язування транспортних задач;
- алгоритми розв'язування задач ЛП симплексним методом із штучним базисом.

7.1 Основні відомості про систему MathCAD

На сьогоднішній день розроблено та існує багато пакетів для розв'язування різних математичних задач: Maple, Matlab, Mathematica і т. ін. Кожний з них має свої переваги та недоліки, а також свої сфери застосування.

Пакет MathCAD створений для розв'язування математичних, фізичних задач, інженерних розрахунків, а також для виконання символічних операцій. MathCAD – універсальна програма, вона значною мірою може справлятися з задачами з усіх областей математики: алгебра і математичний аналіз, диференціальні рівняння і ряди, теорія ймовірностей і статистика. Однією з переваг системи MathCAD є також те, що вона може взаємодіяти з іншими системами: Excel, Matlab, AutoCad, Visual Basic тощо. MathCAD має дуже зручний інтерфейс з вільною формою запису, як на класній дошці; в разі помилки, програма повідомляє про помилку з поміткою формули, в якій вона знаходиться.

Запуск MathCAD

Для того, щоб увійти в MathCAD, знайдіть на робочому столі відповідну піктограму і натисніть два рази на ліву кнопку мишки. Якщо ж відповідної піктограми немає на робочому столі, то зайдіть в «Пуск» – «Програми» – «MathSoft Apps» – «MathCAD [Version] Professional», натисніть Enter. На екрані з'явиться заставка MathCAD Plus, а потім саме вікно MathCAD.

Зауваження стосовно роботи в системі MathCAD

Пакет MathCAD не вимагає від користувача спеціальної підготовки програміста. Відображення інформації, з якою доводиться працювати, зокрема рівнянь, нерівностей, матриць тощо, майже не відрізняється від тієї, що міститься у навчальних посібниках. Проте існують розбіжності у візуальному наданні матеріалу у друкованих виданнях і на екрані монітора. Це зумовлено тим, що деякі символи містять декілька призначень. Наприклад знак рівності «=» застосовується як знак присвоювання $f(x) = \sin x$, як знак обчислення $2 \cdot 5 = 10$, як знак рівності за умовою в рівняннях і т. ін. Звісно, що в програмних продуктах такої універсальності досягти неможливо. Тому для надання різного змісту застосовують різні символи. Наприклад для присвоювання застосовується символ «:=», для обчислення стандартний знак рівності «=», для позначення знака рівності за умовою символ « \Rightarrow » і т. ін. Існують ще розбіжності, до яких досить швидко можна звикнути за невеликий проміжок часу роботи з пакетом MathCAD.

MathCAD має 9 основних панелей: «Calculator» («Арифметика»), «Graph» («Графіки»), «Greek» («Грецький алфавіт»), «Matrix» («Матриці»), «Calculus» («Матаналіз»), «Evaluation» («Обчислення»), «Boolean» («Булево»), «Symbolic» («Символи»), «Programming»

(«Програмування»), див. рис. 7.1.

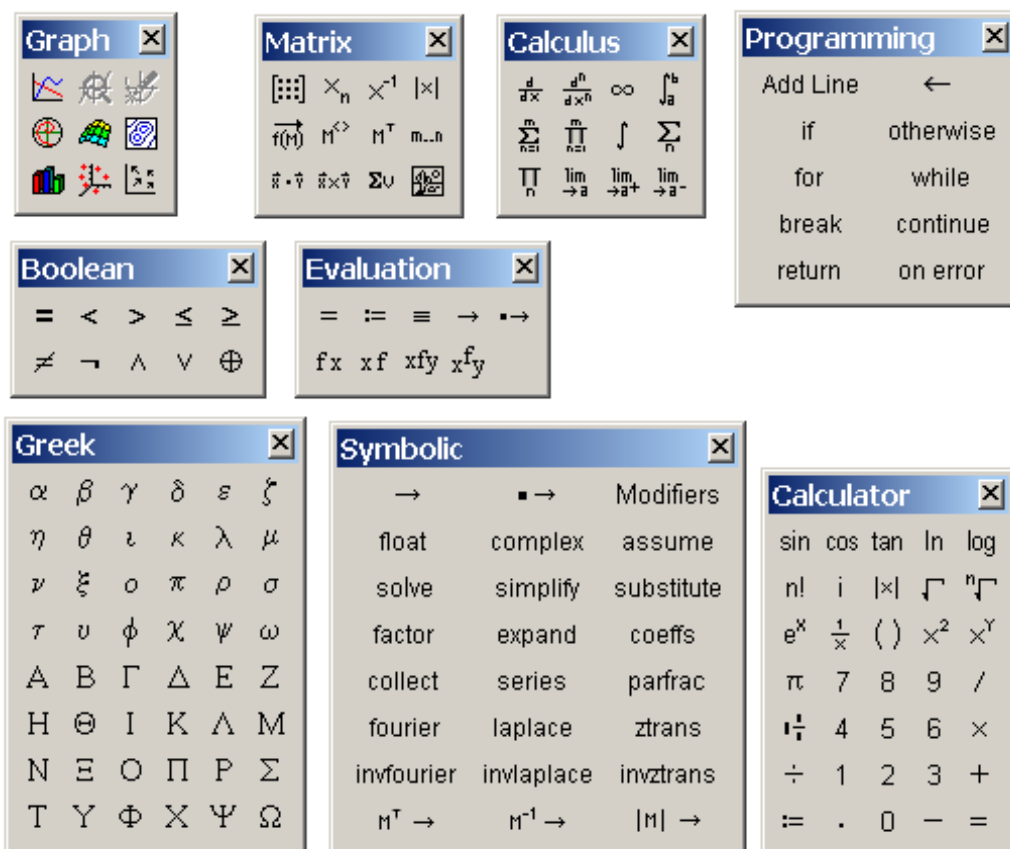


Рис. 7.1. Панелі інструментів

Якщо до знаку на панелі підвести курсор, то з'явиться підказка про призначення цього знаку та як його набрати з клавіатури.

Збереження робочого документу

Щоб зберегти робочий документ, необхідно вибрати пункт «Save» з меню «File» або натиснути кнопку з дискетою на панелі інструментів. Якщо файл до цього часу не зберігався, то з'явиться діалогове вікно «Save as». Введіть ім'я файлу, вказавши повний опис шляху. Інакше MathCAD збереже файл в каталозі, в якому він встановлений, або з якого зчитаний останній робочий документ.

Вихід з MathCAD

Для того щоб вийти з системи MathCAD по закінченню роботи, необхідно вибрати пункт «Exit» з меню «File» або натиснути Alt+F4. MathCAD закриє всі

відкриті ним вікна і вийде в диспетчер програм. Якщо в робочому документі були якісь зміни, то MathCAD виведе діалогове вікно з запитанням про збереження цих змін.

7.2 Розв'язування задач графічним методом у системі MathCAD

Розглянемо задачу ЛП з двома змінними у стандартній формі


$$\begin{aligned} a_{11} x_1 + a_{12} x_2 &\leq a_{10}, \\ a_{21} x_1 + a_{22} x_2 &\leq a_{20}, \\ &\dots\dots\dots \end{aligned} \tag{7.2.1}$$

$$\begin{aligned} a_{m1} x_1 + a_{m2} x_2 &\leq a_{m0}, \\ x_1 \geq 0, x_2 &\geq 0, \end{aligned} \tag{7.2.2}$$

$$Z = c_1 x_1 + c_2 x_2 \rightarrow \max. \tag{7.2.3}$$

Етап 1. Завдання цільової функції Z (7.2.3).

В MathCAD можна задавати будь-які функції користувача. Для цього потрібно вказати ім'я функції (довільна комбінація букв, цифр та знаку «_», що починається з букви), її аргументи, знак присвоювання «:=» та вираз функції. Наприклад, $Z(x):=5x$, $my_f1(x, y):=x+y^2$. Знак присвоювання можна визвати, натиснувши мишею відповідну кнопку на панелі «Calculator» або «Evaluation» (рис. 7.1), або набравши на клавіатурі клавішу «:=».

Функція обов'язково повинна містити аргумент. Якщо змінних декілька, то необхідно вказати їх всі через кому. Аргументом може бути і вектор. За замовчанням в MathCAD перший елемент масиву (вектору) має індекс 0,0 (0). Для того, щоб його змінити на більш звичний індекс 1,1, потрібно набрати команду $ORIGIN:=1$. Індекс набирають, використовуючи або кнопку  з панелі «Matrix» або з клавіатури «[».

В MathCAD цільову функцію $Z = c_1 x_1 + c_2 x_2$ можна задати кількома способами (рис.7.2).

Спосіб I. Цільова функція задається як функція двох змінних

$$Z(x_1, x_2) := c_1 \cdot x_1 + c_2 \cdot x_2. \tag{7.2.4}$$

Зауваження 1. Всі сталі коефіцієнти необхідно задавати вище або ліворуч від місця, де вони перший раз зустрічаються (приклад 1 на рис. 7.2), або замість коефіцієнтів записувати їх значення (приклад 1а на рис. 7.2)

Зауваження 2. x_1 – це координата вектора x . Таке ім'я можна надати змінній, використовуючи команду «нижній регістр» (з клавіатури

x , «.» та 1). Машина розрізняє індекс та нижній регістр, але на екрані вони відображаються однаково: x_1 . Тому з метою уникнення плутанини доцільно змінній надати ім'я x або x_1 .

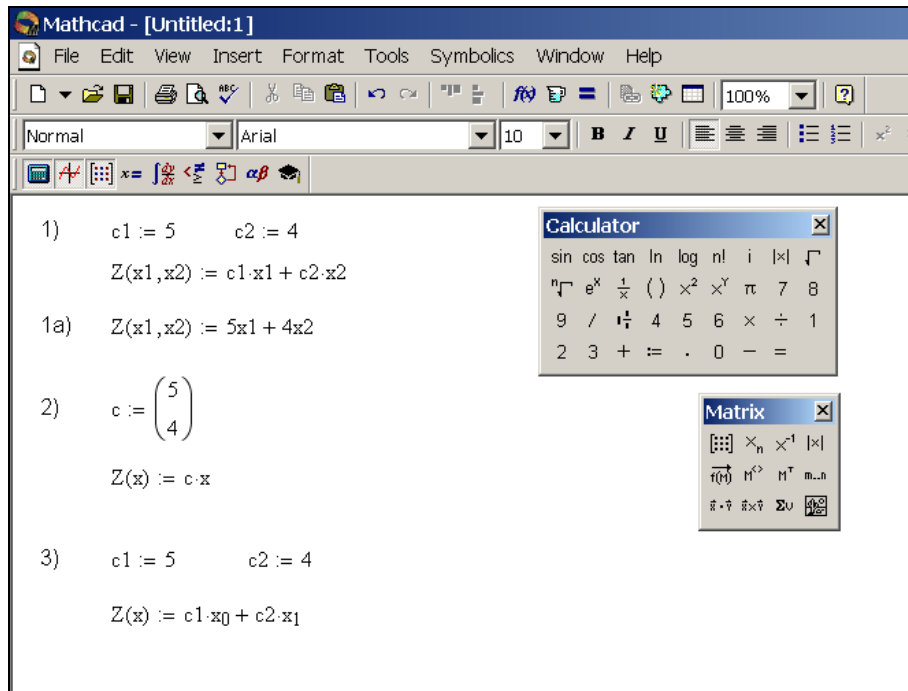



Рис. 7.2. Різні способи завдання цільової функції

Спосіб II. Цільова функція задається у векторному виді (приклад 2 на рис. 7.2)

$$Z(x) := c \cdot x \quad (7.2.5)$$

Для того щоб задати вектор, потрібно набрати «c», «:» та кнопку  (створити матрицю) панелі «Matrix». З'явиться діалогове вікно (рис. 7.3), в якому потрібно вказати кількість рядків (Rows) та стовпців (Columns), після чого ввести координати вектора. Зауважимо, що в MathCAD вектор задається матрицею з одним стовпцем.

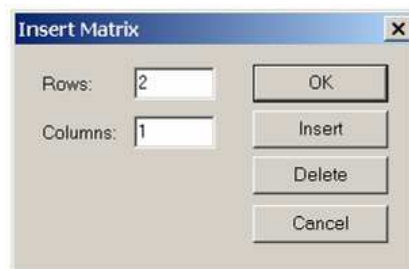


Рис. 7.3. Вставка матриці

Спосіб III. Цільова функція задається у векторному виді з явно вказаною правою частиною

$$Z(x) := c1 \cdot x_0 + c2 \cdot x_1, \quad (7.2.6)$$

де x_0, x_1 – координати вектора x .

Якщо додати команду `ORIGIN:=1`, то функція запишеться так:

$$Z(x) := c1 \cdot x_1 + c2 \cdot x_2. \quad (7.2.6)'$$

Етап 2. Знаходження області допустимих розв'язків задачі (ОДР).

Побудуємо границю першої півплощини, що відповідає нерівності $a_{11}x_1 + a_{12}x_2 \leq a_{10}$. Для цього рівняння $a_{11}x_1 + a_{12}x_2 = a_{10}$ потрібно розв'язати відносно x_2


$$x_2 = \frac{a_{10} - a_{11}x_1}{a_{12}}. \quad (7.2.7)$$

Функцію границі області потрібно задати в явному виді як функцію від x_1 (x_1 – позначимо просто x). Назвемо цю функцію y_1 . В MathCAD вона запишеться так:

$$y_1(x) := \frac{a_{10} - a_{11}x}{a_{12}}. \quad (7.2.8)$$

Аналогічно задаються границі інших півплощин. Зауважимо, що кожній функції необхідно надати своє ім'я.

Побудова області допустимих значень.

Натиснувши на кнопку  (або клавішу «@») на панелі «Graph» (рис. 7.1), викличемо шаблон для побудови графіка в декартових координатах (рис. 7.4).

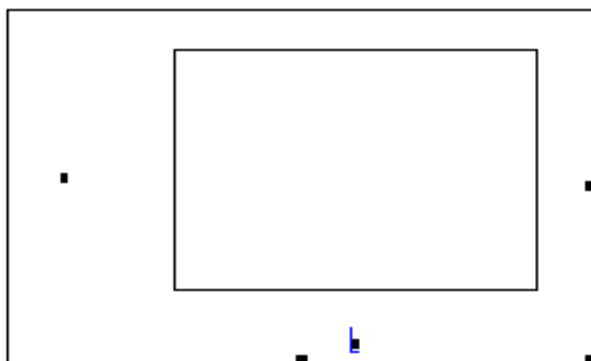


Рис. 7.4.

Знизу в середньому квадратику необхідно вказати аргументи функції (в нашому випадку x). Зліва та праворуч від аргументу з'являться ще дві позначки, для введення початку та кінця інтервалу, на якому змінюється аргумент (в лівий квадратик доцільно записати 0, оскільки за умовою $x_1 \geq 0$ (7.2.2), правий – можна не заповнювати). Якщо залишити ці поля пустими, то в MathCAD вони заповняться автоматично значеннями $(-10, 10)$. Якщо є умови виду $x \geq a_i$, $x \leq b_j$ ($i = 1, 2, \dots, k$, $j = 1, 2, \dots, l$), то, записавши в квадратик початку інтервалу максимальне значення серед a_i , а в квадратик кінця – мінімальне серед b_j , на графіку будуть відображені тільки точки, що задовольняють цим умовам. По осі Oy в середньому квадратику необхідно вказати ім'я функції, що будується, та її аргумент. Якщо на одному полі потрібно побудувати графіки кількох функцій, то імена функцій необхідно записати через кому. Зверху та знизу можна також

вказати границі змін значення функції (має сенс поставити нижню границю 0, оскільки є умова $x_2 \geq 0$, або у випадку наявності умов $y \geq c_i$, $y \leq d_j$ ($i = 1, 2, \dots, k1$, $j = 1, 2, \dots, l1$) записати в нижній квадратик максимальне значення серед c_i , у верхній – мінімальне значення серед d_j) (рис. 7.5).

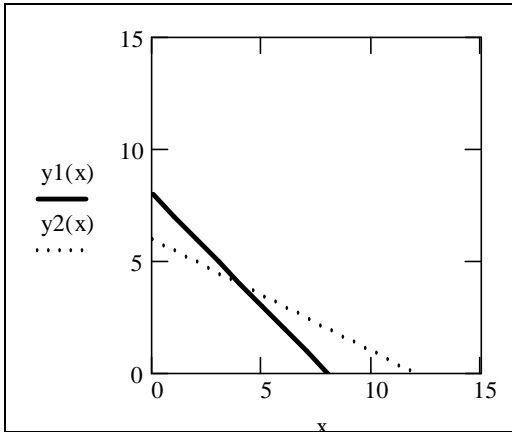


Рис. 7. 5.

Зобразимо на графіку осі координат. Для цього потрібно двічі клацнути лівою кнопкою миші по графіку, після чого з'являється меню (рис. 7.6)

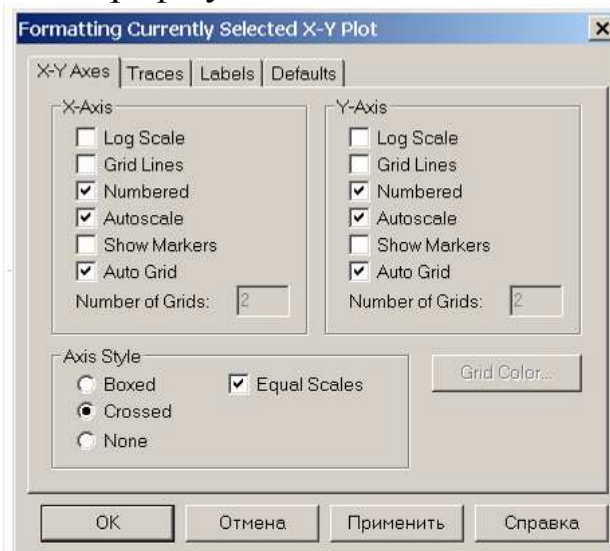


Рис. 7.6.

Необхідно вибрати «Axis Style Crossed» (осі перетинаються), та за бажанням «Equal Scales» (однакові масштаби по осям Ox та Oy). Натискаючи клавішу «Применить», можна одразу бачити зміни на графіку, не закриваючи діалогове вікно (рис. 7.7).

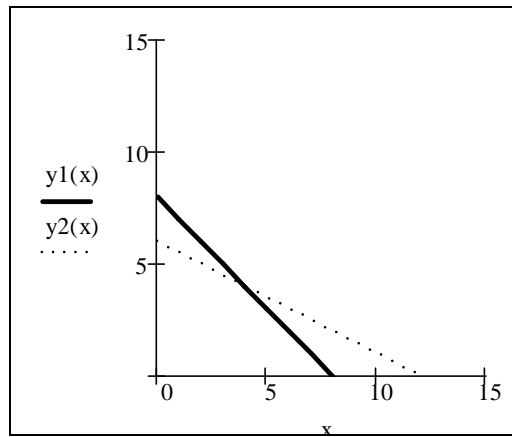


Рис. 7.7.

В MathCAD немає можливості на графіку безпосередньо будувати півплощини, що відповідають нерівностям, та знаходити перетин областей. Але можна задати додаткові функції і за їх допомогою зобразити на графіку штриховку для ілюстрації відповідних півплощин.

Введемо параметр **a**, що змінюється від 0 до 1. Для цього набираємо

a, команду присвоєння «:=», команду  з панелі «Matrix» (рис. 7.1)

(або набираємо з клавіатури «;»), з'являються два квадратики $a := \square \dots \square$, в які вводимо початкове та кінцеве значення змінної.

Задамо функцію $y_{a1}(x, a)$, графік якої буде відображати штриховку до першої лінії $y_1(x)$, відмічаючи потрібну півплощину.

Оскільки перша нерівність має вид $a_{11} x_1 + a_{12} x_2 \leq a_{10}$, то в допустимій області потрапляють точки, що лежать, наприклад, на лінії $a_{11} x_1 + a_{12} x_2 = a_{10} - a$. Тому функцію $y_{a1}(x, a)$ можна задати так

$$y_{a1}(x, a) = \frac{a_{10} - a - a_{11} x}{a_{12}}. \quad (7.2.9)$$

Подібним чином задаються інші функції.

Зауваження. У випадку нерівності $a_{11} x_1 + a_{12} x_2 \geq a_{10}$ потрібно розглядати функцію $y_{a1}(x, a) = \frac{a_{10} + a - a_{11} x}{a_{12}}$.

Зобразимо нові функції на графіку (рис. 7.8). Оскільки останні функції залежать від двох змінних (змінна x , та параметр **a**), то перед побудовою графіка потрібно задати границі змін аргументу x ($x:=0..15$). Густоту штриховки можна регулювати, змінюючи крок для змінної x . Задавши $x:=0,0.1..15$, отримаємо густі «зигзаги», при $x:=0,2..15$ – більш рідкі.

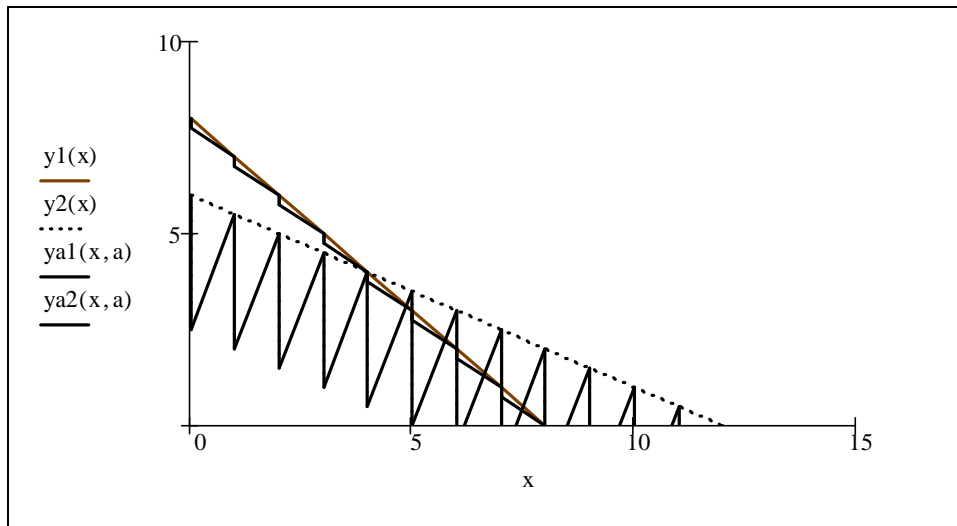


Рис. 7. 8.

Якщо зображена штриховка до лінії зовсім маленька або навпаки дуже велика, то доцільно взяти параметр **a** зі збільшеним або зменшеним коефіцієнтом, наприклад,

$$ya1(x,a) = \frac{a_{10} - 10a - a_{11} x}{a_{12}} \quad \text{або}$$

$$ya1(x,a) = \frac{a_{10} - 0.1a - a_{11} x}{a_{12}}.$$

Якщо вибрати вкладку «Traces» (рис. 7.9), то можна змінити вид ліній. Для цього потрібно вибрати лінію (вибрана лінія виділяється кольором). Столпчик «Legend Label» дозволяє надати лінії будь-яке ім'я. Його просто набирають в нижньому полі. Столпчик «Line» дає змогу вибрати тип лінії: безперервну (solid), пунктирну (dash), зображену точками (dot), тощо. Всі можливі варіанти розкриваються у списку, якщо натиснути кнопку з трикутником знизу стовпця. Столпець «Color» дозволяє змінювати колір лінії, стовпець «Weight» – товщину.

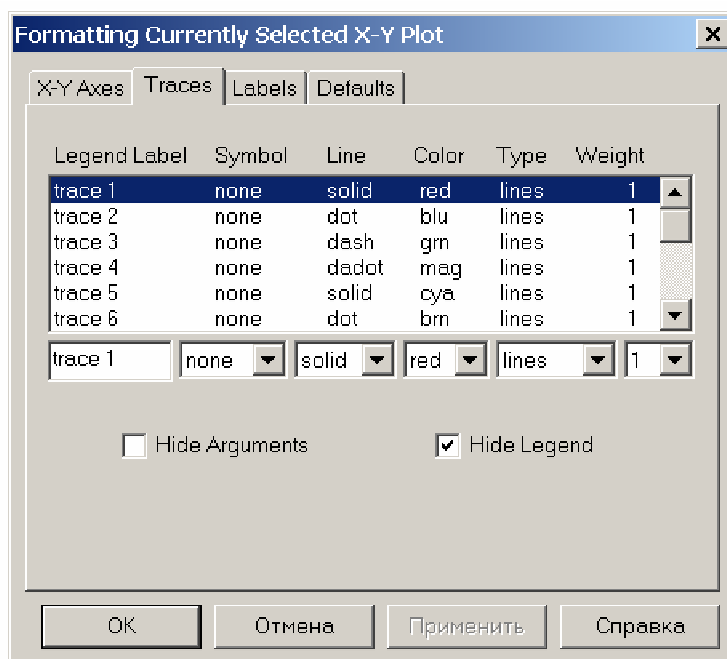


Рис. 7. 9.

Можна всі лінії зробити неперервними (solid), основні – більш товстими (Weight 2), а колір штриховки вибрати таким саме що й колір відповідних ліній (рис. 7.10).

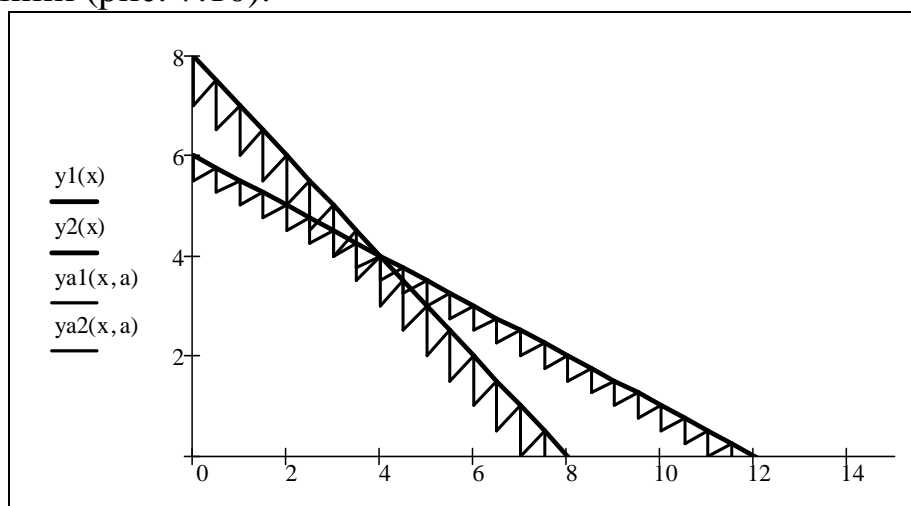


Рис. 7. 10.

Залишилось знайти область допустимих значень. В програмі цю область (перетин півплощин) виділити автоматично неможливо. Тому можна поступити наступним чином:

а) взяти параметр **a** дуже великим, ОДР –це область, де перетинаються штриховки (на рис. 7.11 параметр **a** дорівнює 100);

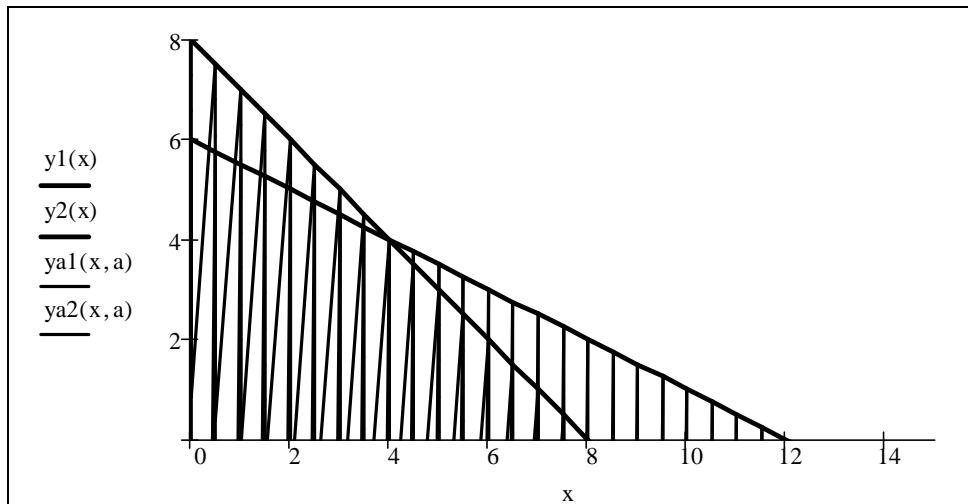


Рис. 7. 11.

б) знайти область «вручну»: це точки, які лежать вище горизонтальної осі Ox (умова $y \geq 0$), праворуч від вертикальної осі Oy (умова $x \geq 0$) (інші точки, завдяки завданню границь інтервалів, на графіку не зображаються), та належать допустимим півплощинам (виділені штриховкою).

Зауваження. Можна виділити область, що задовольняє умовам $y \geq 0$, $x \geq 0$. Відповідні граничні лінії ми відобразили на графіку, вибравши команду «Axis Style» «Crossed» – зобразити осі. Для того щоб відобразити штриховку до осі Ox , необхідно задати функцію $ox(x,a) := a$ (можна помножити a на будь-який коефіцієнт, щоб збільшити або зменшити висоту штриховки). Трохи складніше виділити область $y \geq 0$. Для цього потрібно задати інтервальну змінну t – вона буде визначати густоту та ширину штриховки, її значення залежать від масштабу по осі Ox (на рис. 7.12 $t := 0,0.3..0.8$) та функцію $oy(x,a) := n \cdot a$ (значення n визначає висоту штриховки, в нашому випадку $n = 8$). Після виділення півплощин (з використанням функцій $oy(t,a)$, $ox(x,a)$, що відповідають нерівностям $y \geq 0$, $x \geq 0$.) одержимо область допустимих розв’язків задачі лінійного програмування (див. рис. 7.12).

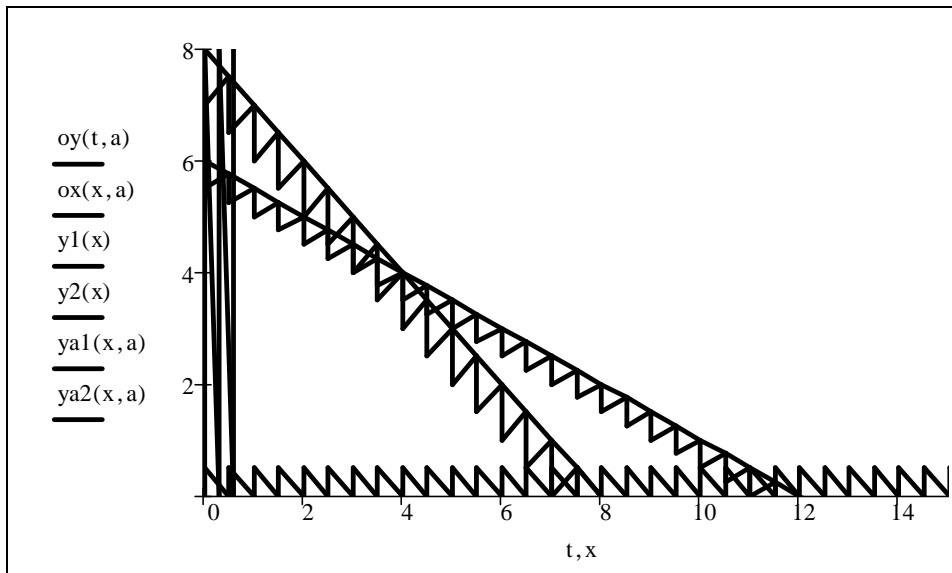


Рис. 7. 12.

Етап 3. Визначення ліній рівня (ліній сталого значення) цільової функції та знаходження точки максимуму.

Задамо лінії рівня $Z = \text{const}$ ($c_1x_1 + c_2x_2 = C$):

$$y_c(x, C) := \frac{C - c_1x_1}{c_2}. \quad (7.2.10)$$

На графіку зобразимо дві лінії рівня $y_c(x, C)$, взявши, наприклад, значення $C=20$, и $C=45$, для того щоб визначити напрямок зростання цільової функції (рис. 7.13).

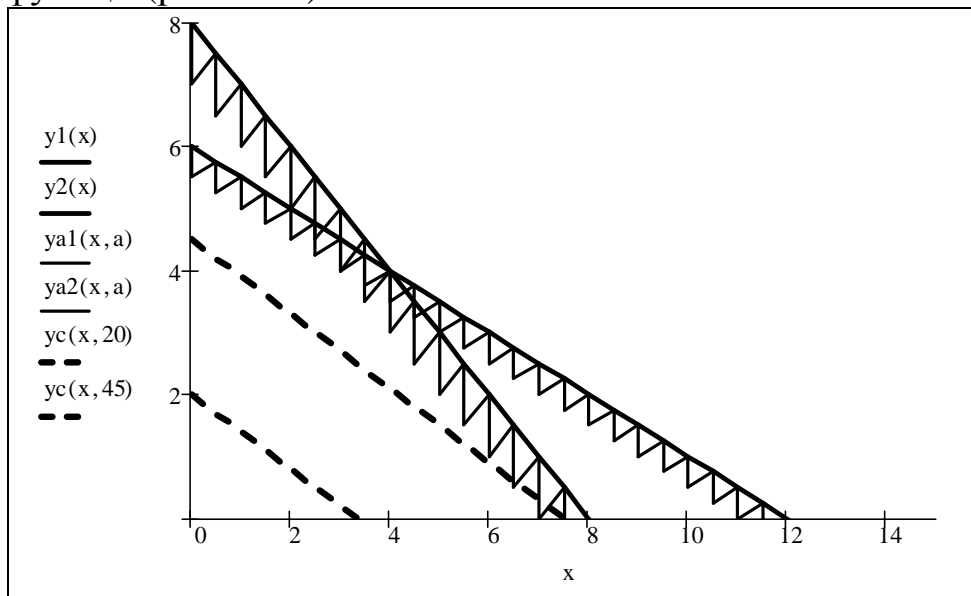


Рис. 7. 13.

Будемо збільшувати значення C , слідкуючи за тим, щоб лінія рівня перетинала ОДЗ. Із графіка видно, що точка максимуму – це точка перетину прямих $y_1(x)$ і $y_2(x)$ (рис. 7.14).

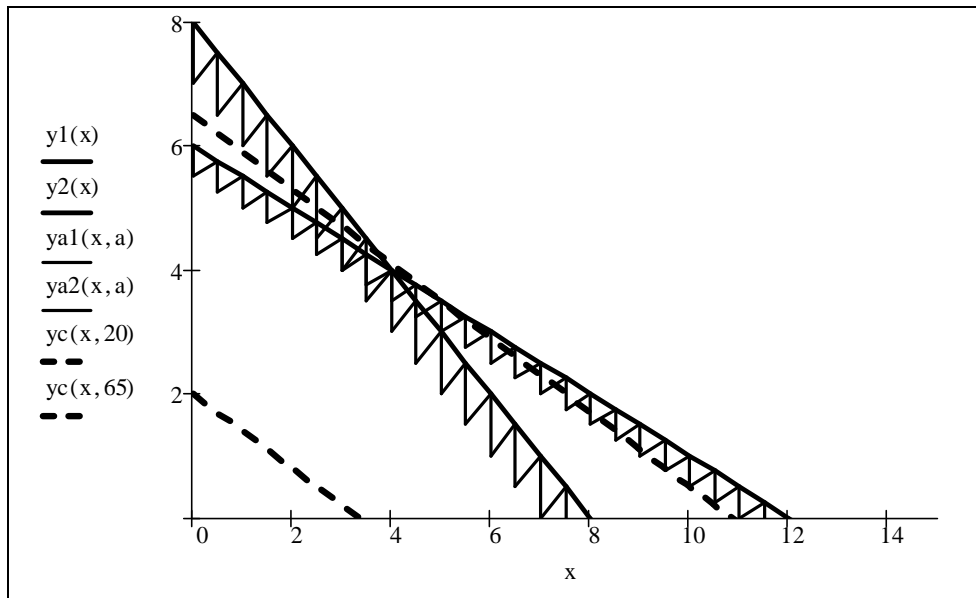


Рис. 7. 14.

Знайдемо координати точки перетину, розв'язавши систему

$$\begin{cases} a_{11} x_1 + a_{12} x_2 = a_{10}, \\ a_{21} x_1 + a_{22} x_2 = a_{20}. \end{cases}$$

Для розв'язання системи в MathCAD можна використати спеціальний обчислювальний блок, що складається з чотирьох послідовних елементів:

- ✓ **початкові значення** для невідомих змінних x_1, \dots, x_m ;
- ✓ **Given** – ключове слово;
- ✓ **система**, записана логічними операторами у вигляді рівностей (або нерівностей);
- ✓ **Find(x_1, \dots, x_m)** – вбудована функція, яка знаходить якісь значення змінних x_1, \dots, x_m , що задовольняють системі.

Логічні оператори задають за допомогою панелі «Boolean» (рис. 7.1). Блок «Given» – «Find» використовує для пошуку розв'язку ітераційні методи, тому вище ключового слова «Given» потрібно задати початкові значення для всіх невідомих x_1, \dots, x_m . Значенням функції Find є вектор, елементи якого є розв'язком системи. Маємо

$$x_1 := 0 \quad x_2 := 0$$

Given

$$a_{11} x_1 + a_{12} x_2 = a_{10}$$

$$a_{21} x_1 + a_{22} x_2 = a_{20}$$

$$X := \text{Find}(x_1, x_2)$$

Систему рівнянь можна записати у такому виді:

Given

$$x_2 = y_1(x_1)$$

$$x_2 = y_2(x_1)$$

Щоб знайти максимум цільової функції, треба підставити координати вектора X в цільову функцію $Z(X_0, X_1)$.

Зауваження. Якщо із графіка важко з'ясувати в якій точці буде оптимум, то потрібно за наведеним вище правилом перевірити всі «підозрілі» точки. Якщо значення цільової функції для двох вершин збігається, то має місце альтернативний оптимум.

Розв'яжемо приклад 1 із §2.3 за допомогою MathCAD.

$$-x_1 + 3x_2 \leq 33,$$

$$3x_1 + 8x_2 \leq 139,$$

$$11x_1 - 5x_2 \leq 132,$$

$$x_1 + x_2 \geq 5,$$

$$x_1 \geq 0, \quad x_2 \geq 0,$$

$$Z = 2x_1 + x_2 \rightarrow \max.$$

Позначимо невідому x_1 через x , а x_2 через y . Задамо в явному виді границі областей $y_1(x)$, $y_2(x)$, $y_3(x)$, $y_4(x)$; штриховку до них $ya_1(x)$, $ya_2(x)$, $ya_3(x)$, $ya_4(x)$; штриховку до осей $oy(t,a)$, $ox(x,a)$; цільову функцію $Z(x, y)$; лінію рівня $us(x,C)$. (рис. 7.15)

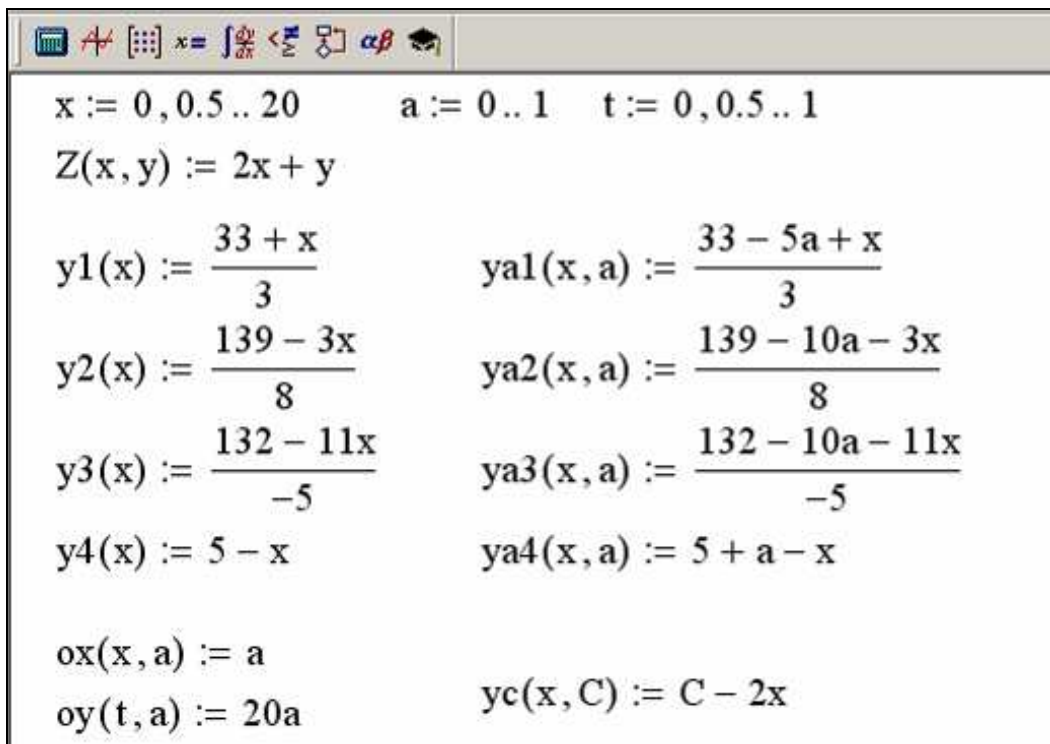


Рис. 7. 15.

Нижні границі для x та y задамо рівними нулю, зобразимо осі. Тоді отримаємо такий графік (рис. 7.16)

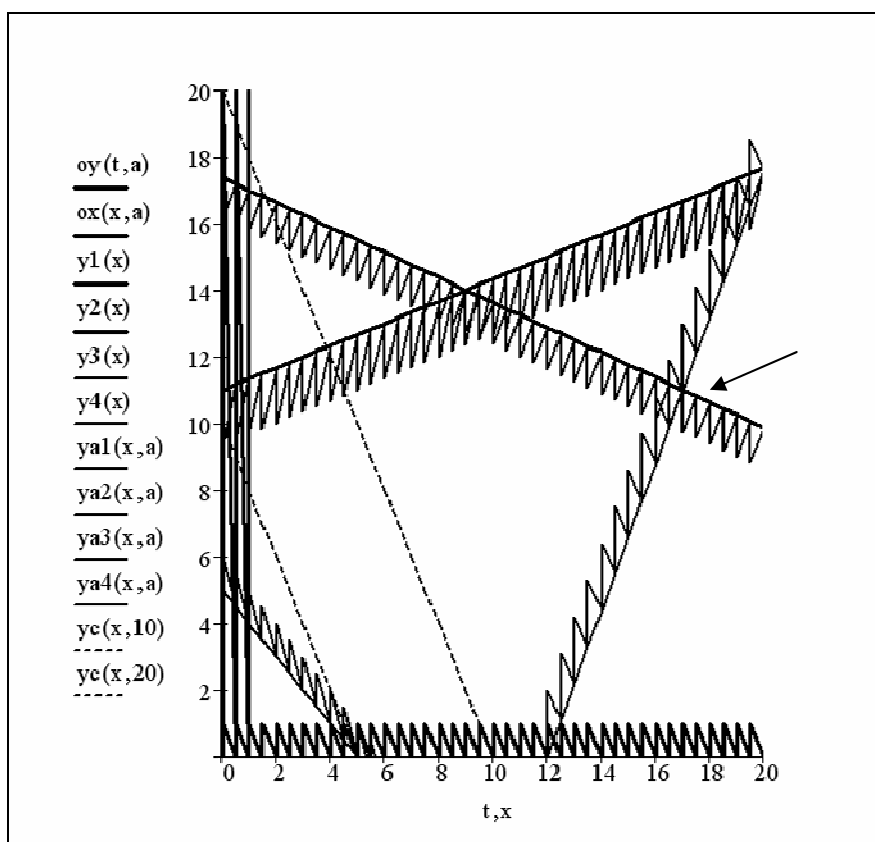



Рис. 7. 16.

Очевидно, що максимум знаходиться у точці перетину прямих $y2(x)$ та $y3(x)$ (на рисунку 7.16 точка вказана стрілкою). Знайдемо координати

набрати «A», «:» та кнопку  (створити матрицю) панелі «Matrix». В діалоговому вікні потрібно вказати кількість рядків (Rows) та стовпців (Columns), після чого ввести потрібні координати.

$$A := \begin{pmatrix} a_{10} & 1 & 0 & \dots & 0 & \dots & 0 & a_{1,m+1} & \dots & a_{1n} \\ a_{20} & 0 & 1 & \dots & 0 & \dots & 0 & a_{2,m+1} & \dots & a_{2n} \\ \dots & & & & & & & & & \\ a_{q0} & 0 & 0 & \dots & 1 & \dots & 0 & a_{q,m+1} & \dots & a_{qn} \\ \dots & & & & & & & & & \\ a_{m0} & 0 & 0 & \dots & 0 & \dots & 1 & a_{m,m+1} & \dots & a_{mn} \end{pmatrix}.$$

Задамо числа m – кількість рівнянь системи та n – що на 1 більше за кількість невідомих. Ці значення можна знайти використовуючи вбудовані функції MathCAD.

$$m := \text{rows}(A) \quad n := \text{cols}(A)$$

Задамо рядок C , що містить коефіцієнти цільової функції (7.3.3)

$$C := (-c_0 \quad c_1 \quad c_2 \quad \dots \quad c_p \quad \dots \quad c_m \quad c_{m+1} \quad \dots \quad c_n).$$

Знайдемо оцінки змінних. Приєднаємо рядок $-C$ до матриці A знизу. Це можна зробити за допомогою вбудованої функції `stack`

$$A := \text{stack}(A, -C).$$

Отримаємо

$$A = \begin{pmatrix} a_{10} & 1 & 0 & \dots & 0 & \dots & 0 & a_{1,m+1} & \dots & a_{1n} \\ a_{20} & 0 & 1 & \dots & 0 & \dots & 0 & a_{2,m+1} & \dots & a_{2n} \\ \dots & & & & & & & & & \\ a_{q0} & 0 & 0 & \dots & 1 & \dots & 0 & a_{q,m+1} & \dots & a_{qn} \\ \dots & & & & & & & & & \\ a_{m0} & 0 & 0 & \dots & 0 & \dots & 1 & a_{m,m+1} & \dots & a_{mn} \\ c_0 & -c_1 & -c_2 & \dots & -c_p & \dots & -c_m & -c_{m+1} & \dots & -c_n \end{pmatrix}.$$


Із 3.1 відомо наступне: якщо за допомогою методу Жордана-Гаусса отримати нулі в останньому рядку коефіцієнтів при базисних змінних, то для вільних змінних коефіцієнти нижнього рядка перетворяться на оцінки.

Нагадаємо, що елементи a_{ik}^* при перетворенні Жордана-Гаусса (a_{qp} – розв’язувальний елемент) знаходяться за формулами

$$a_{ik}^* = \frac{a_{ik}}{a_{qp}}, \quad \text{якщо } i = q,$$

$$a_{ik}^* = a_{ik} - a_{ip} \frac{a_{qk}}{a_{qp}} \quad \text{в інших випадках.} \quad (7.3.4)$$

Алгоритм 7.3.4 можна реалізувати з використанням елементів програмування.

Вкажемо, як змінюються індекси матриці. Покладемо `ORIGIN:=1`. Тоді індекс `i` приймає цілі значення від 1 до `m+1` («пробігає» всі рядки). Щоб задати `i`, набираємо `i`, команду присвоювання «:=», команду  з панелі «Matrix» (або набираємо з клавіатури «;»), з'являються два квадратики `i := ■ .. ■`, в які вводимо початкове та кінцеве значення змінної. Аналогічно задаємо `k` («пробігає» всі стовпці).

`i := 1..m + 1` `k := 1..n`.



Ітерація методом Жордана-Гаусса. Задаємо розв'язувальний елемент. Для цього достатньо вказати його індекси:

`r` – це номер стовпця, в якому стоїть базисна змінна, а відповідний елемент нижнього рядка не дорівнює нулю;

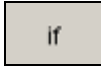
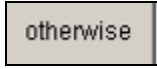
`q` – це номер рядка, в якому стоїть базисна одиниця.

Введемо додаткову матрицю `a`, з використанням якої знайдемо нові коефіцієнти:

`a := A`.

Індекс елемента матриці набирають, використовуючи або кнопку  з панелі «Matrix» або з клавіатури «[». Перший індекс матриці відокремлюється від другого комою. Елемент матриці позначається такою ж літерою, що й матриця. Наберемо «A», «[», «i,j» «:», на панелі інструментів «Programming» (рис. 7.1) виберемо команду  (добавити лінію). Ця команда задає програмний модуль. З'явиться

$$A_{i,k} := \begin{array}{|l} \blacksquare \\ \blacksquare \end{array}$$

Поставимо курсор в верхній квадратик, з панелі інструментів «Programming» вибираємо команду  (якщо), в нижньому квадратіку вибираємо команду  (інакше).

$$A_{i,k} := \begin{array}{|l} \blacksquare \text{ if } \blacksquare \\ \blacksquare \text{ otherwise} \end{array}$$

Для оператора «if» в першу позначку вводиться дія, яку потрібно виконати за умови, яка вказується у другій позначці (з використанням елементів панелі «Boolean»). У позначці оператора «Otherwise» вказується дія, яку потрібно виконати, якщо умова «if» не виконується.

Запишемо алгоритм Жордана-Гаусса (рис. 7.18).

$$A_{i,k} := \begin{array}{|l} \frac{a_{i,k}}{a_{q,p}} \quad \text{if } i = q \\ a_{i,k} - a_{i,p} \cdot \frac{a_{q,k}}{a_{q,p}} \quad \text{otherwise} \end{array}$$

Рис. 7. 18.

Повторюємо цей крок для всіх базисних змінних, при яких коефіцієнт нижнього рядка не дорівнює нулю.

Отримаємо нову матрицю A, в нижньому рядку якої стоять оцінки змінних та значення цільової функції Z в опорному розв'язку

$$A = \begin{pmatrix} a_{10} & 1 & 0 & \dots & 0 & \dots & 0 & a_{1,m+1} & \dots & a_{1n} \\ a_{20} & 0 & 1 & \dots & 0 & \dots & 0 & a_{2,m+1} & \dots & a_{2n} \\ & & & & & & & \dots & & \\ a_{q0} & 0 & 0 & \dots & 1 & \dots & 0 & a_{q,m+1} & \dots & a_{qn} \\ & & & & & & & \dots & & \\ a_{m0} & 0 & 0 & \dots & 0 & \dots & 1 & a_{m,m+1} & \dots & a_{mn} \\ Z & 0 & 0 & \dots & 0 & \dots & 0 & a_{0,m+1} & \dots & a_{0,n} \end{pmatrix}.$$

Етап 2. Симплексне перетворення.

Якщо серед оцінок $a_{0,m+1} \dots a_{0,n}$ є від'ємні, то такий розв'язок можна покращити за правилом:

1) За розв'язувальний вибирається стовпець елементів при невідомій x_p , для якої оцінка $a_{0p} < 0$, та у відповідному стовпці є хоча б один додатний елемент $a_{ip} > 0$.

2) Розв'язувальний елемент a_{qp} вибирається виходячи з умови

$$\frac{a_{q0}}{a_{qp}} = \min_{\text{для } a_{jp} > 0} \left\{ \frac{a_{j0}}{a_{jp}} \right\}. \quad (7.3.5)$$

3) Для елемента a_{qp} здійснюємо крок Жордана-Гаусса.

Реалізуємо правило 7.3.5 в MathCAD. Введемо стовпець T , елементами якого будуть відношення $\frac{a_{j0}}{a_{jp}}$ за умови $a_{jp} > 0$. Якщо $a_{jp} \leq 0$, замість відношення поставимо знак « ∞ ». MathCAD це число відображає як 1×10^{307} . Індекс j змінюється від 1 до m (в останньому рядку стоять оцінки, які ми не можемо вибрати за розв'язувальний елемент).

$$j := 1..m$$

$$a := A$$

$$T_j := \begin{cases} \frac{a_{j,1}}{a_{j,p}} & \text{if } a_{j,p} > 0 \\ \infty & \text{otherwise} \end{cases}.$$

Задамо індекс q – це номер рядка, в якому стоїть мінімальний елемент. Здійснимо ітерацію методом Жордана-Гаусса (див. рис. 7.18). Повторюємо етап 2, доки всі оцінки не стануть додатними.

Зауваження. Межі змін індексів i, j, k можна вказати один раз на початку документа.

Розв'яжемо задачу ЛП, наведену у §3.3 за допомогою MathCAD.

$$\begin{aligned} -2x_1 + x_2 + x_3 &= 4, \\ 2x_1 + 7x_2 + x_4 &= 92, \\ 2x_1 - 3x_2 + x_5 &= 12, \end{aligned} \tag{7.3.6}$$

$$x_i \geq 0, \quad i = 1, 2, \dots, 5, \tag{7.3.7}$$

$$Z = 4x_1 + 9x_2 + x_3 + x_4 \rightarrow \max. \tag{7.3.8}$$

Покладемо початковий індекс масиву рівним 1.

Запишемо A – розширену матрицю системи 7.3.6, C – коефіцієнти цільової функції (7.3.8) (першим стоїть 0, оскільки доданку c_0 немає), задамо m – кількість рядків матриці A , n – кількість стовпців матриці A , та вкажемо межі зміни індексів i, j, k . Приєднаємо рядок $-C$ до матриці A знизу (рис. 7.19).

```

ORIGIN := 1
A := ( 4  -2  1  1  0  0
      92  2  7  0  1  0
      12  2 -3  0  0  1 )
m := rows(A)      n := cols(A)
i := 1..m+1      k := 1..n      j := 1..m
C := ( 0  4  9  1  1  0 )
A := stack(A, -C)
A = ( 4  -2  1  1  0  0
     92  2  7  0  1  0
     12  2 -3  0  0  1
     0 -4 -9 -1 -1  0 )

```

Рис. 7.19.

Обчислимо оцінки. Для базисної змінної x_3 розв’язувальний елемент має індекс 1,4 ($q=1, r=4$), здійснимо ітерацію методом Жордана-Гаусса (рис. 7.20) Для базисної змінної x_4 розв’язувальний елемент має індекс 2,5. Задамо відповідні змінні та виконаємо ще один раз

перетворення Жордана-Гаусса. Для базисної змінної x_5 в останньому рядку вже стоїть 0, тому ще одну ітерацію робити не потрібно.

$a := A \quad q := 1 \quad p := 4$ $A_{i,k} := \begin{cases} \frac{a_{i,k}}{a_{q,p}} & \text{if } i = q \\ a_{i,k} - a_{i,p} \cdot \frac{a_{q,k}}{a_{q,p}} & \text{otherwise} \end{cases}$	$A = \begin{pmatrix} 4 & -2 & 1 & 1 & 0 & 0 \\ 92 & 2 & 7 & 0 & 1 & 0 \\ 12 & 2 & -3 & 0 & 0 & 1 \\ 4 & -6 & -8 & 0 & -1 & 0 \end{pmatrix}$
$a := A \quad q := 2 \quad p := 5$ $A_{i,k} := \begin{cases} \frac{a_{i,k}}{a_{q,p}} & \text{if } i = q \\ a_{i,k} - a_{i,p} \cdot \frac{a_{q,k}}{a_{q,p}} & \text{otherwise} \end{cases}$	$A = \begin{pmatrix} 4 & -2 & 1 & 1 & 0 & 0 \\ 92 & 2 & 7 & 0 & 1 & 0 \\ 12 & 2 & -3 & 0 & 0 & 1 \\ 96 & -4 & -1 & 0 & 0 & 0 \end{pmatrix}$

Рис. 7. 20.

Отримали рядок оцінок. Обидві оцінки вільних змінних від'ємні, причому у відповідних стовпцях є додатні елементи. Отже, вихідний опорний розв'язок можна покращити, здійснивши ітерацію симплексного перетворення. За розв'язувальний стовпець виберемо другий ($p = 2$).

З'ясуємо, який рядок потрібно вибрати за розв'язувальний (рис. 7.21).

$a := A$ $p := 2 \quad T_j := \begin{cases} \frac{a_{j,1}}{a_{j,p}} & \text{if } a_{j,p} > 0 \\ \infty & \text{otherwise} \end{cases}$	$T = \begin{pmatrix} 1 \times 10^{307} \\ 46 \\ 6 \end{pmatrix}$
--	--

Рис. 7. 21.

Мінімальний елемент стоїть у третьому рядку, тому $q = 3$. Здійснимо для елемента (3,2) ітерацію методом Жордана-Гаусса (рис. 7.22).

$$\begin{array}{l}
 q := 3 \\
 A_{i,k} := \begin{cases} \frac{a_{i,k}}{a_{q,p}} & \text{if } i = q \\ a_{i,k} - a_{i,p} \cdot \frac{a_{q,k}}{a_{q,p}} & \text{otherwise} \end{cases}
 \end{array}
 \quad
 A = \begin{pmatrix} 16 & 0 & -2 & 1 & 0 & 1 \\ 80 & 0 & 10 & 0 & 1 & -1 \\ 6 & 1 & -1.5 & 0 & 0 & 0.5 \\ 120 & 0 & -7 & 0 & 0 & 2 \end{pmatrix}$$

Рис. 7. 22.

Для змінної x_2 оцінка від'ємна ($p = 3$), і в цьому стовпці є додатний елемент ($q = 2$). Оскільки додатний елемент один, крок для визначення розв'язувального рядка (знаходження вектора T) можна опустити. Здійснимо ще одне симплексне перетворення (рис. 7.23).

$$\begin{array}{l}
 a := A \\
 p := 3 \\
 T_j := \begin{cases} \frac{a_{j,1}}{a_{j,p}} & \text{if } a_{j,p} > 0 \\ \infty & \text{otherwise} \end{cases}
 \end{array}
 \quad
 T = \begin{pmatrix} 1 \times 10^{307} \\ 8 \\ 1 \times 10^{307} \end{pmatrix}$$

$$\begin{array}{l}
 q := 2 \\
 A_{i,k} := \begin{cases} \frac{a_{i,k}}{a_{q,p}} & \text{if } i = q \\ a_{i,k} - a_{i,p} \cdot \frac{a_{q,k}}{a_{q,p}} & \text{otherwise} \end{cases}
 \end{array}
 \quad
 A = \begin{pmatrix} 32 & 0 & 0 & 1 & 0.2 & 0.8 \\ 8 & 0 & 1 & 0 & 0.1 & -0.1 \\ 18 & 1 & 0 & 0 & 0.15 & 0.35 \\ 176 & 0 & 0 & 0 & 0.7 & 1.3 \end{pmatrix}$$

Рис. 7. 23.

В отриманій матриці від'ємних оцінок немає. Отже, знайдений опорний розв'язок є оптимальним. $\vec{X}_{max} = (18; 8; 32; 0; 0)$. $Z(\vec{X}_{max}) = 176$.

Зауваження. В MathCAD результат обчислень можна відобразити у вигляді звичайного дроби. Для цього потрібно двічі клацнути лівою кнопкою миші по об'єкту, який хочете відобразити в іншому форматі або виділити об'єкт, вибрати на верхній панелі «Format», «Result». З'явиться діалогове вікно (рис. 7.24)

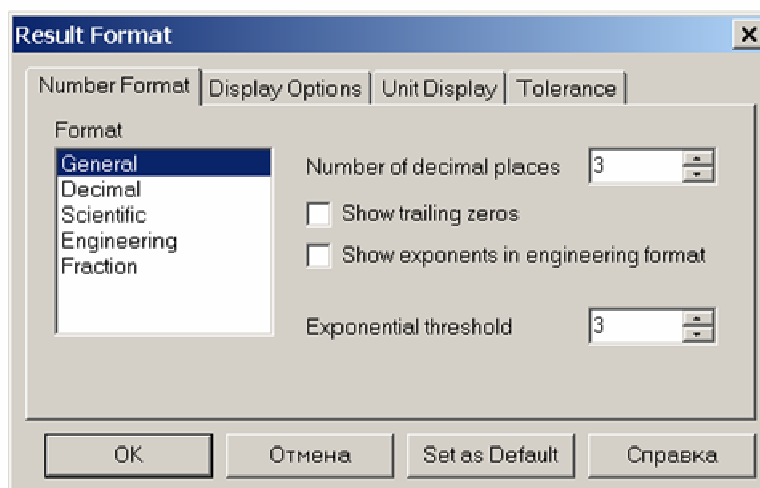


Рис. 7. 24.

Потім вибрати «Fraction» (дріб) і отримати

$$A = \begin{pmatrix} 32 & 0 & 0 & 1 & \frac{1}{5} & \frac{4}{5} \\ 8 & 0 & 1 & 0 & \frac{1}{10} & -\frac{1}{10} \\ 18 & 1 & 0 & 0 & \frac{3}{20} & \frac{7}{20} \\ 176 & 0 & 0 & 0 & \frac{7}{10} & \frac{13}{10} \end{pmatrix} .$$

7.4 Розв'язування задачі ЛП зі штучним базисом у системі MathCAD

У MathCAD М-задача розв'язується за таким же алгоритмом, що і задачі, які не містять штучного базису. Потрібно тільки задати будь-яке значення для числа М (зазвичай його беруть на кілька порядків більшим за коефіцієнти цільової функції) і слідкувати за складом штучних та основних змінних.

Розв'яжемо задачу ЛП із §3.4 за допомогою MathCAD

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

$$x_i \geq 0, \quad i = 1, 2, \dots, 4,$$

$$Z = x_1 - x_2 - 5x_3 + x_4 + 4 \rightarrow \max .$$

Їй відповідає M -задача

$$\begin{cases} x_1 + 3x_2 + 3x_3 + x_4 + x_5 = 3, \\ 2x_1 + 3x_2 - x_4 + x_6 = 4, \end{cases} \quad (7.4.1)$$

$$x_i \geq 0, \quad i = 1, 2, \dots, 6,$$

$$W = x_1 - x_2 - 5x_3 + x_4 + 4 - M(x_5 + x_6) \rightarrow \max, \quad (7.4.2)$$

де змінні x_5, x_6 – штучні.

Запишемо дані задачі.

Покладемо початковий індекс масиву рівним 1. Сталу величину M покладемо рівною 100. Запишемо A – розширену матрицю системи 7.4.1, C – коефіцієнти цільової функції (7.4.2), задамо m та n – відповідно кількість рядків та стовпців матриці A , вкажемо межі зміни індексів i, j, k . Приєднаємо рядок $-C$ до матриці A знизу (рис. 7.25).

```

ORIGIN := 1
M := 100

A := (3 1 3 3 1 1 0)
      (4 2 3 0 -1 0 1)

C := (-4 1 -1 -5 1 -M -M)

m := rows(A)
n := cols(A)
i := 1..m+1
k := 1..n
j := 1..m

A := stack(A, -C)

A = (3 1 3 3 1 1 0)
     (4 2 3 0 -1 0 1)
     (4 -1 1 5 -1 100 100)

```

Рис. 7. 25.

Обчислимо оцінки. Для цього одержимо нулі в шостому та сьомому стовпцях (рис. 7.26).

$$\begin{array}{l}
 a := A \quad q := 1 \quad p := 6 \\
 A_{i,k} := \begin{cases} \frac{a_{i,k}}{a_{q,p}} & \text{if } i = q \\ a_{i,k} - a_{i,p} \cdot \frac{a_{q,k}}{a_{q,p}} & \text{otherwise} \end{cases} \quad A = \begin{pmatrix} 3 & 1 & 3 & 3 & 1 & 1 & 0 \\ 4 & 2 & 3 & 0 & -1 & 0 & 1 \\ -296 & -101 & -299 & -295 & -101 & 0 & 100 \end{pmatrix} \\
 \\
 a := A \quad q := 2 \quad p := 7 \\
 A_{i,k} := \begin{cases} \frac{a_{i,k}}{a_{q,p}} & \text{if } i = q \\ a_{i,k} - a_{i,p} \cdot \frac{a_{q,k}}{a_{q,p}} & \text{otherwise} \end{cases} \quad A = \begin{pmatrix} 3 & 1 & 3 & 3 & 1 & 1 & 0 \\ 4 & 2 & 3 & 0 & -1 & 0 & 1 \\ -696 & -301 & -599 & -295 & -1 & 0 & 0 \end{pmatrix}
 \end{array}$$

Рис. 7. 26.

Отримали рядок оцінок. Всі оцінки вільних змінних від’ємні, причому у відповідних стовпцях є додатні елементи. Отже, вихідний опорний розв’язок можна покращити, здійснивши ітерацію симплексного перетворення. За розв’язувальний стовпець виберемо третій ($p = 3$), оскільки відповідна оцінка найменша.

З’ясуємо, який рядок потрібно вибрати за розв’язувальний, і здійсимо для нього симплекс-крок (рис. 7.27).

$$\begin{array}{l}
 a := A \\
 p := 3 \quad T_j := \begin{cases} \frac{a_{j,1}}{a_{j,p}} & \text{if } a_{j,p} > 0 \\ \infty & \text{otherwise} \end{cases} \quad T = \begin{pmatrix} 1 \\ 1.333 \end{pmatrix} \\
 \\
 q := 1 \quad A_{i,k} := \begin{cases} \frac{a_{i,k}}{a_{q,p}} & \text{if } i = q \\ a_{i,k} - a_{i,p} \cdot \frac{a_{q,k}}{a_{q,p}} & \text{otherwise} \end{cases} \quad A = \begin{pmatrix} 1 & 0.333 & 1 & 1 & 0.333 & 0.333 & 0 \\ 1 & 1 & 0 & -3 & -2 & -1 & 1 \\ -97 & -101.333 & 0 & 304 & 198.667 & 199.667 & 0 \end{pmatrix}
 \end{array}$$

Рис. 7. 27.

Штучна змінна x_5 – стала вільною, тому відповідний стовпець можна видалити. Залишиться тільки одна штучна змінна x_6 . Змінних стане на одну менше, тому потрібно змінити межі змін індексу k (рис. 7.28).

$$\begin{aligned}
 & A := \text{augment}(\text{submatrix}(A, 1, 3, 1, 5), A^{(7)}) \\
 & A = \begin{pmatrix} 1 & 0.333 & 1 & 1 & 0.333 & 0 \\ 1 & 1 & 0 & -3 & -2 & 1 \\ -97 & -101.333 & 0 & 304 & 198.667 & 0 \end{pmatrix} \\
 & k := 1..n-1
 \end{aligned}$$

Рис. 7. 28.

Є одна від’ємна оцінка, тому виберемо стовпець $p = 2$ за розв’язувальний (рис.7.29).

$$\begin{aligned}
 & a := A \\
 & p := 2 \quad T_j := \begin{cases} \frac{a_{j,1}}{a_{j,p}} & \text{if } a_{j,p} > 0 \\ \infty & \text{otherwise} \end{cases} \quad T = \begin{pmatrix} 3 \\ 1 \end{pmatrix} \\
 & q := 2 \quad A_{i,k} := \begin{cases} \frac{a_{i,k}}{a_{q,p}} & \text{if } i = q \\ a_{i,k} - a_{i,p} \cdot \frac{a_{q,k}}{a_{q,p}} & \text{otherwise} \end{cases} \quad A = \begin{pmatrix} 0.667 & 0 & 1 & 2 & 1 & -0.333 \\ 1 & 1 & 0 & -3 & -2 & 1 \\ 4.333 & 0 & 0 & 0 & -4 & 101.333 \end{pmatrix}
 \end{aligned}$$

Рис. 7. 29.

Остання штучна змінна x_6 – стала вільною, тому останній стовпець можна видалити. Змінних стане ще на одну менше, тому заново задамо межі змін індексу k (рис. 7.30).

$$\begin{aligned}
 & A := \text{submatrix}(A, 1, 3, 1, 5) \\
 & A = \begin{pmatrix} 0.667 & 0 & 1 & 2 & 1 \\ 1 & 1 & 0 & -3 & -2 \\ 4.333 & 0 & 0 & 0 & -4 \end{pmatrix} \\
 & k := 1..n-2
 \end{aligned}$$

Рис. 7. 30.

Знову є від’ємна оцінка a_{04} , виберемо відповідний стовпець ($p = 5$) за розв’язувальний (рис. 7.31).

$$\begin{array}{l}
 a := A \\
 p := 5 \quad T_j := \begin{cases} \frac{a_{j,1}}{a_{j,p}} & \text{if } a_{j,p} > 0 \\ \infty & \text{otherwise} \end{cases} \quad T = \begin{pmatrix} 0.667 \\ 1 \times 10^{307} \end{pmatrix} \\
 q := 1 \quad A_{i,k} := \begin{cases} \frac{a_{i,k}}{a_{q,p}} & \text{if } i = q \\ a_{i,k} - a_{i,p} \cdot \frac{a_{q,k}}{a_{q,p}} & \text{otherwise} \end{cases} \quad A = \begin{pmatrix} 0.667 & 0 & 1 & 2 & 1 \\ 2.333 & 1 & 2 & 1 & 0 \\ 7 & 0 & 4 & 8 & 0 \end{pmatrix}
 \end{array}$$

Рис. 7. 31.

Запишемо елементи матриці A у виді звичайного дробу:

$$A = \begin{pmatrix} \frac{2}{3} & 0 & 1 & 2 & 1 \\ \frac{7}{3} & 1 & 2 & 1 & 0 \\ 7 & 0 & 4 & 8 & 0 \end{pmatrix}.$$

В отриманій матриці від’ємних оцінок немає. Всі змінні основні. Отже, знайдений опорний розв’язок є оптимальним. $\vec{X}_{max} = (7/3, 0, 0, 2/3)$, максимальне значення цільової функції $Z(\vec{X}_{max}) = 7$.

7.5 Розв’язування задач ЛП за допомогою вбудованих функцій MathCAD

Для розв’язання будь-яких задач лінійного програмування в системі MathCAD можна використати спеціальні вбудовані функції Maximize – для знаходження максимуму функції та Minimize – для знаходження мінімуму функції.

Для знаходження оптимуму функції задача повинна бути записана в такій послідовності:

- ✓ функція $f(x_1, \dots, x_m)$, для якої знаходиться максимум (мінімум);
- ✓ початкові значення для невідомих змінних x_1, \dots, x_m ;

✓ **Given** – ключове слово;

✓ **система**, записана логічними операторами у вигляді рівностей (або нерівностей);

✓ **Maximize(f,x₁,...,x_m) (Minimize(f,x₁,...,x_m))** – вбудована функція для знаходження максимуму (мінімуму) відносно змінних x₁, ... ,x_m.

Згідно з наведеною послідовністю спочатку потрібно задати цільову функцію Z та початкові значення змінних, їх можна покласти, наприклад, рівними нулю. Після ключового слова **Given** записують систему обмежень (умови набирають з використанням панелі «Boolean»). Якщо в задачі потрібно знайти максимум, вибирають функцію **Maximize**. Перший аргумент функції **Maximize** – цільова функція, вказується тільки її ім'я без аргументів, далі перелічуються змінні, за якими потрібно знайти максимум. Змінні можна записувати в довільному порядку. Результат відображається у вигляді вектора, елементи якого – це змінні x₁, ... ,x_m, які стоять в тому ж порядку, в якому вони були записані у функції **Maximize**. Значення цільової функції потрібно знайти окремо.

Розв'яжемо приклад 1 §2.3 з використанням вбудованих функцій MathCAD.

Знайти найбільше значення функції

$$Z = 2x_1 + x_2$$

за умов

$$-x_1 + 3x_2 \leq 33,$$

$$3x_1 + 8x_2 \leq 139,$$

$$11x_1 - 5x_2 \leq 132,$$

$$x_1 + x_2 \geq 5,$$

$$x_1 \geq 0, \quad x_2 \geq 0,$$

Позначимо x₁ через x, x₂ через y. В нашому випадку функція **Maximize** має три аргументи: перший Z, другий і третій – змінні, за якими потрібно знайти максимум – x та y. Розв'язання наведено на рис. 7.32.

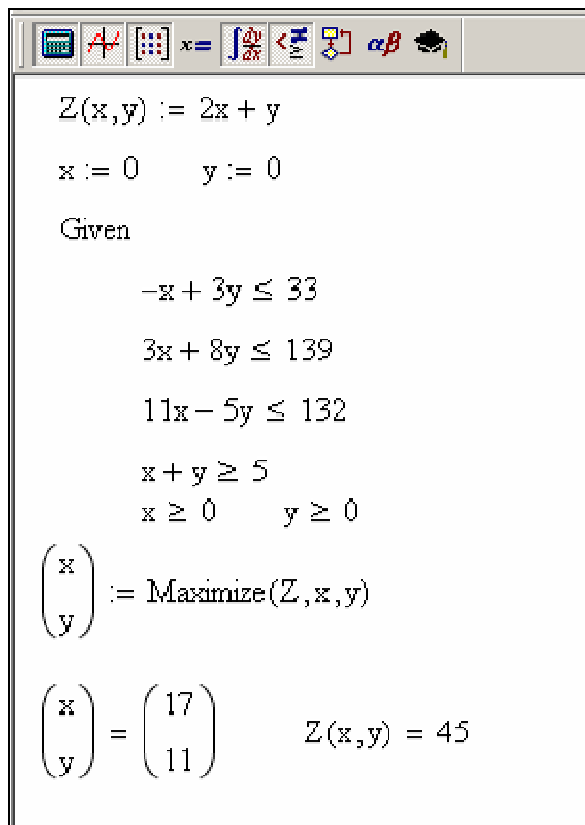


Рис. 7. 32.

Таким чином, максимум цільової функції дорівнює 45 коли $x_1 = 17, \quad x_2 = 11$.

Розв'яжемо приклад із §3.3 за допомогою MathCAD.

Знайти найбільше значення функції

$$Z = 4x_1 + 9x_2 + x_3 + x_4$$

за умов

$$-2x_1 + x_2 + x_3 = 4,$$

$$2x_1 + 7x_2 + x_4 = 92,$$

$$2x_1 - 3x_2 + x_5 = 12,$$

$$x_i \geq 0, \quad i = 1, 2, \dots, 5.$$

Будемо розглядати змінні x_1, x_2, x_3, x_4, x_5 як координати вектора x . Початкові значення змінних можна задати кількома способами, наприклад:

1) задати значення кожної змінної – $x_1 := 0, x_2 := 0$, і т. ін.

2) Оскільки x_1, x_2, x_3, x_4, x_5 – координати вектора x , то можна задати

початкове значення для вектора: $x := \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$, або у більш компактному

виді $x := (0 \ 0 \ 0 \ 0 \ 0)^T$, де T – операція транспонування.

Розв'язання задачі наведено на рис. 7.33.

```

ORIGIN := 1

Z(x) := 4x1 + 9x2 + x3 + x4

x := (0 0 0 0 0)T

Given

    x ≥ 0
    -2x1 + x2 + x3 = 4
    2x1 + 7x2 + x4 = 92
    2x1 - 3x2 + x5 = 12

X := Maximize(Z, x1, x2, x3, x4, x5)

XT = (18 8 32 0 0)    Z(X) = 176
    
```

Рис. 7. 33.

Зауваження. Аргументи функції `Maximize` можна записати більш компактно: `Maximize(Z,x)`.

Таким чином, максимум функції Z досягається при $x_1=18$, $x_2=8$, $x_3=32$, $x_4=0$, $x_5=0$ і дорівнює 176.

Розв'яжемо приклад із §3.4 за допомогою MathCAD

Розв'язати наступну задачу лінійного програмування:

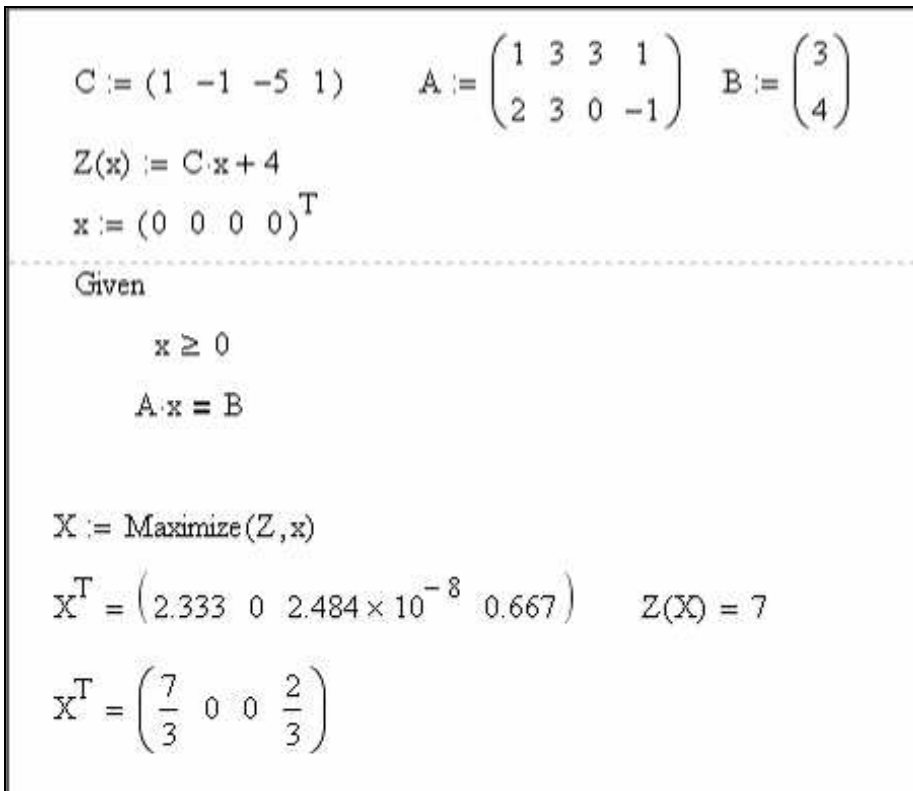
$$\begin{cases} x_1 + 3x_2 + 3x_3 + x_4 = 3, \\ 2x_1 + 3x_2 - x_4 = 4, \end{cases} \quad (7.5.1)$$

$$x_i \geq 0, \quad i = 1, 2, \dots, 4, \quad (7.5.2)$$

$$Z = x_1 - x_2 - 5x_3 + x_4 + 4 \rightarrow \max. \quad (7.5.3)$$

Якщо задача записана у симетричному виді, систему обмежень можна записати більш компактно у матричному виді. Введемо рядок C – коефіцієнти при невідомих цільової функції, матриця A – коефіцієнти при невідомих в системі обмежень, вектор B – права частина системи обмежень.

Тоді (7.5.1) запишеться у вигляді $Ax=B$, (7.5.2) як $x \geq 0$, цільова функція (7.5.3) – $Z=Cx+4$, де $x=(x_1, x_2, x_3, x_4, x_5)$. На рис. 7.34 наведено результати розв'язання задачі.



$C := (1 \ -1 \ -5 \ 1)$ $A := \begin{pmatrix} 1 & 3 & 3 & 1 \\ 2 & 3 & 0 & -1 \end{pmatrix}$ $B := \begin{pmatrix} 3 \\ 4 \end{pmatrix}$
 $Z(x) := C \cdot x + 4$
 $x := (0 \ 0 \ 0 \ 0)^T$

Given

$x \geq 0$

$A \cdot x = B$

$X := \text{Maximize}(Z, x)$

$X^T = (2.333 \ 0 \ 2.484 \times 10^{-8} \ 0.667)$ $Z(X) = 7$

$X^T = \left(\frac{7}{3} \ 0 \ 0 \ \frac{2}{3} \right)$

Рис. 7. 34.

Значення, одержані у вигляді звичайних дробів, збігаються з результатом, одержаним при розв'язанні задачі симплекс-методом. Якщо розв'язок записаний у виді десяткового дробу – третя координата дуже маленька, але відрізняється від нуля. Цього можна уникнути, якщо задати більш високу точність (STOL). Для цього можна визвати діалогове вікно «Tools» (інструментарій) «Worksheet Option» (опції робочого аркуша) і

змінити значення CTOL (рис.7.35) або задати $CTOL:=10^{-8}$ (рис. 7.36).

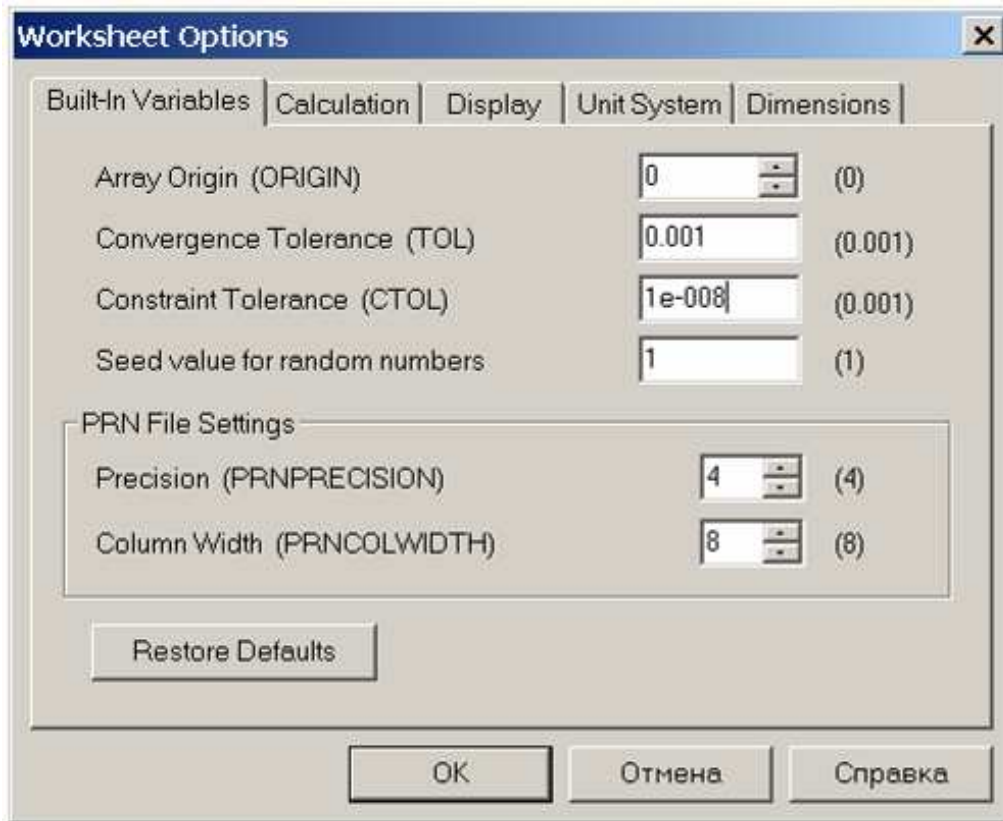


Рис. 7. 35.

$$\begin{aligned} X^T &= (2.333 \ 0 \ 0 \ 0.667) & Z(X) &= 7 \\ X^T &= \left(\frac{7}{3} \ 0 \ 0 \ \frac{2}{3} \right) \end{aligned}$$

Рис. 7. 36.

Функції Maximize (Minimize) призначені для знаходження максимуму / мінімуму як лінійних так і нелінійних функцій кількох змінних. Через таку універсальність у випадках розв'язання задач ЛП програма не завжди видає вичерпну відповідь. Розглянемо кілька таких випадків.

Приклад 1.

$$\begin{cases} x + y \geq 3, \\ x \geq 0, y \geq 0, \end{cases}$$

$$Z = x + y \rightarrow \max.$$

Очевидно, що область припустимих розв'язків не обмежена і цільова функція не має на ній максимуму. При розв'язанні задачі в MathCAD, програма видає повідомлення «Could not find a maximum/minimum» (Максимум/мінімум не знайдено) – тобто задача розв'язків не має (рис.7.37).

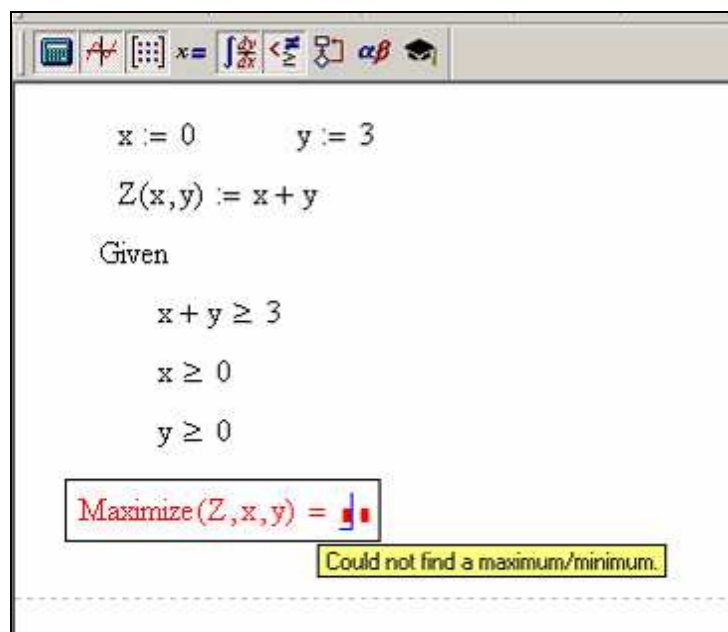


Рис. 7. 37.

Приклад 2.

$$\begin{cases} x + y \leq 3, \\ x \geq 0, y \geq 0, \end{cases}$$

$$Z = x + y \rightarrow \max.$$

Очевидно, що в усіх точках, що лежать на прямій $x + y = 3$, та задовольняють умовам $x \geq 0, y \geq 0$, функція Z має максимум. Тобто ця задача має альтернативний оптимум. Розв'яжемо її у MathCAD (рис. 7.38)

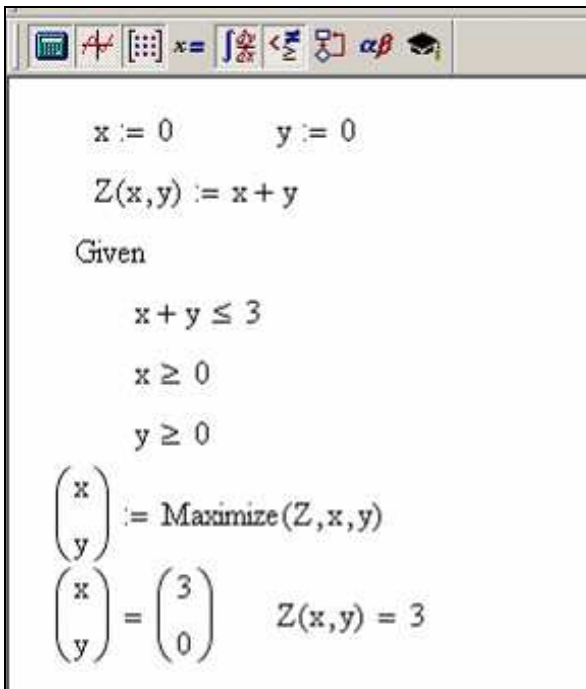


Рис. 7. 38.

Тобто машина знаходить **тільки** один з розв'язків. Є кілька шляхів перевірити, чи має задача кілька розв'язків:

1) змінити початкові значення змінних. Але цей спосіб не завжди дає результат. Так навіть при завданні початкових $x = 0$ та $y = 3$ (інший розв'язок) машина видає такий же результат $(3, 0)$.

2) змінити умову для змінних. Так, якщо задати умову $y \geq 1$ (зменшимо область), машина знаходить інший розв'язок $(2, 1)$ з таким же максимальним значенням (рис. 7.39). Але не завжди легко зрозуміти, яку або які змінні змінювати та на скільки.

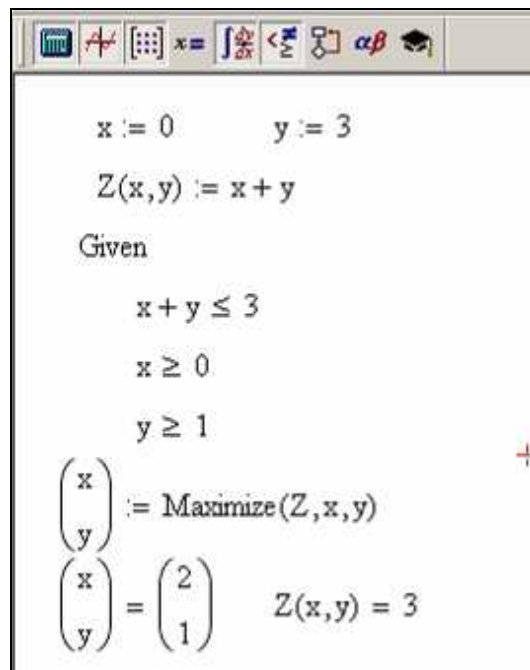


Рис. 7. 39.

Приклад 3.

$$\begin{cases} x + y \leq 3, \\ x + y \geq 5, \\ x \geq 0, y \geq 0, \end{cases}$$

$$Z = 2x + y \rightarrow \max.$$

Очевидно, що така задача не має розв'язку. Машина виділить функцію червоним і видасть підказку «No solution was found. Try changing the guess value or value of TOL or CTOL» (Розв'язок не знайдено. Спробуйте змінити початкові значення або величини TOL або CTOL) (рис. 7.40).

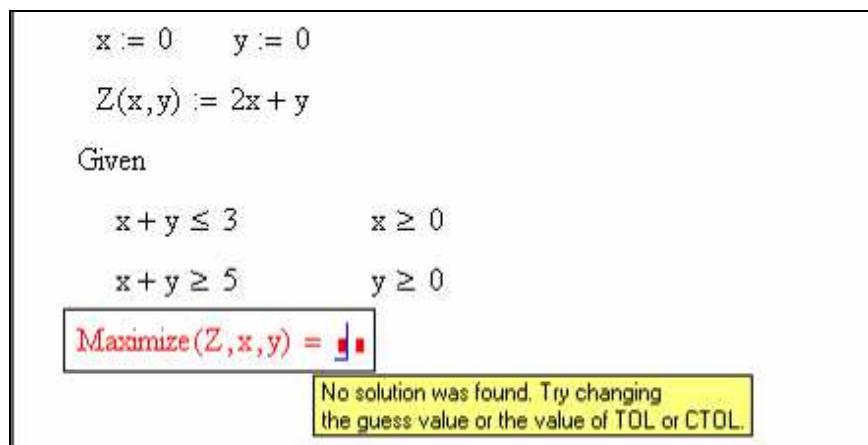


Рис. 7. 40.

Тобто такий саме коментар ми отримаємо і в випадку, якщо ітераційний процес повільно збігається, або невдало вибрані початкові значення. Тому можна:

- 1) змінити початкові значення змінних;
- 2) зменшити точність обчислень (збільшити значення для TOL або STOL);
- 3) перевірити, чи не є область пустою. Це можна зробити за допомогою функції Find (знаходить значення змінних, що задовольняють системі) або Minerr (знаходить приблизний розв'язок системи). Minerr використовують в разі повільного збігання ітераційного процесу, або в випадку великої кількості обмежень.

7.6 Розв'язування транспортної задачі у системі MathCAD

Для розв'язування транспортної задачі можна використовувати вбудовані функції Maximize (Minimize).

Відмітимо, що для розв'язування ТЗ з використанням таких функцій немає потреби знаходити вихідний опорний розв'язок. Також непотрібно відкрити модель транспортної задачі зводити до закритої.


Розв'яжемо ТЗ із §4.5 за допомогою вбудованих функцій.

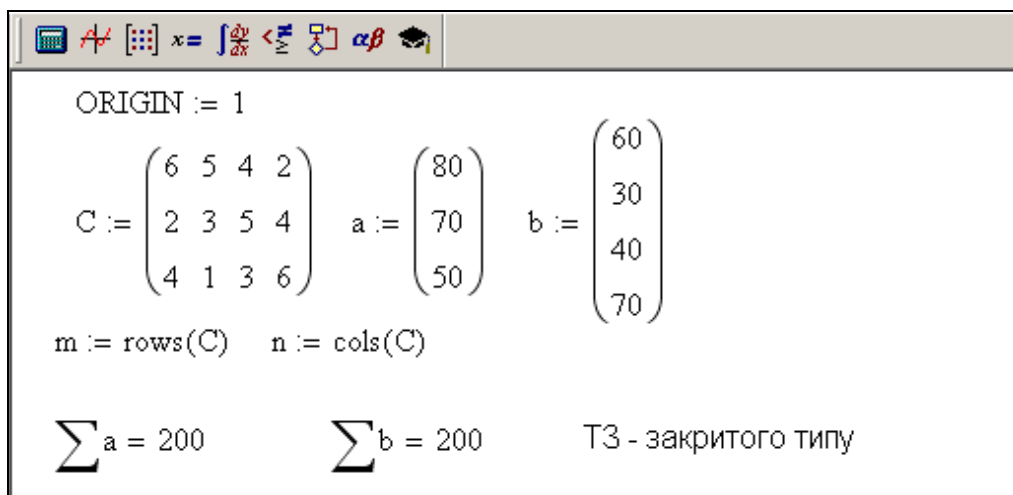
Є 3 пункти відправлення, в яких розміщено однорідний вантаж в кількості 80, 70, 50 одиниць. Цей вантаж необхідно доставити в 4 пункти призначення в кількості відповідно 60, 30, 40, 70 одиниць. Крім того, відомі тарифи c_{ij} ($i = 1, 2, 3; j = 1, 2, 3, 4$) – вартість доставки одиниці товару із i -го пункту відправлення до j -го пункту призначення (в грн). ТЗ задана таблицею 7.1.

Необхідно скласти план перевезень вантажу таким чином, щоб споживачі одержали необхідну кількість вантажу і сумарна вартість всіх перевезень була мінімальною.

Таблиця 7.1

$a_i \backslash b_j$	60	30	40	70
80	6	5	4	2
70	2	3	5	4
50	4	1	3	6

Запишемо дані задачі в матричному вигляді: C – матриця тарифів, \mathbf{a} – вектор кількості вантажу в пунктах відправлення, \mathbf{b} – вектор кількості вантажу в пунктах призначення, \mathbf{m} – кількість пунктів відправлення, \mathbf{n} – кількість пунктів призначення. Перевіримо, що задача відноситься до закритого типу. Для цього можна на панелі «Matrix» натиснути кнопку , вказати ім'я вектора, суму елементів якого потрібно знайти (\mathbf{a} чи \mathbf{b}), та натиснути «= \Rightarrow » (рис. 7.41).



ORIGIN := 1
 $C := \begin{pmatrix} 6 & 5 & 4 & 2 \\ 2 & 3 & 5 & 4 \\ 4 & 1 & 3 & 6 \end{pmatrix}$ $\mathbf{a} := \begin{pmatrix} 80 \\ 70 \\ 50 \end{pmatrix}$ $\mathbf{b} := \begin{pmatrix} 60 \\ 30 \\ 40 \\ 70 \end{pmatrix}$
 $m := \text{rows}(C)$ $n := \text{cols}(C)$
 $\sum \mathbf{a} = 200$ $\sum \mathbf{b} = 200$ TЗ - закритого типу

Рис. 7. 41.

Задамо вихідний розв'язок матрицю \mathbf{x} – її елементи можуть приймати будь-які значення, наприклад, можна покласти їх рівними нулю.

Тепер задамо цільову функцію і після ключового слова Given запишемо систему обмежень. І обмеження і цільову функцію можна записати в загальному вигляді 4.1.2 – 4.1.5. Розв'язок задачі наведений на

рис.7.42.

$$\begin{aligned}
 \mathbf{x} &:= \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \\
 Z(\mathbf{x}) &:= \sum_{i=1}^m \sum_{j=1}^n C_{i,j} x_{i,j} \\
 \text{Given} \\
 \mathbf{x} &\geq 0 \\
 \sum_{i=1}^m x_{i,1} &= b_1 \quad \sum_{i=1}^m x_{i,2} = b_2 \quad \sum_{i=1}^m x_{i,3} = b_3 \quad \sum_{i=1}^m x_{i,4} = b_4 \\
 \sum_{j=1}^n x_{1,j} &= a_1 \quad \sum_{j=1}^n x_{2,j} = a_2 \quad \sum_{j=1}^n x_{3,j} = a_3 \\
 X &:= \text{Minimize}(Z, \mathbf{x}) \\
 X &= \begin{pmatrix} 0 & 0 & 10 & 70 \\ 60 & 0 & 10 & 0 \\ 0 & 30 & 20 & 0 \end{pmatrix} \quad Z(X) = 440
 \end{aligned}$$

Рис. 7. 42.

Таким чином, з першого пункту відправлення a_1 потрібно вивезти 10 одиниць товару до 3-го та 70 од. до 4-го пункту призначення, з a_2 – 60 од. до 1-го та 10 од. до 3-го, з a_3 – 30 од. до 2-го та 20 од. до 3-го. Вартість перевезення при цьому буде найменшою і становитиме 440 грн.

Розглянемо задачу відкритого типу, задану таблицею 7.2.

Таблиця 7.2

$a_i \backslash b_j$	20	80
50	12	16
70	8	10

Аналогічно до попереднього прикладу запишемо дані, та перевіримо

7.7 Приклади розробки математичних моделей проектування оптимальних рецептур продуктів харчування, використання методів ЛП та комп'ютерних технологій

В різних сферах людської діяльності виникає багато задач математичного програмування, і зокрема ЛП, які можуть бути розв'язані викладеними вище методами з використанням пакета MathCAD. Сказане ілюструємо, розглянувши задачі із галузі виробництва продукції харчування.

Один із підходів до розробки проекту нової рецептури, як відомо, полягає у раціональній (обґрунтованій) корекції рецептури-аналога. Вказаний підхід часто реалізується на практиці. Разом з тим задача розробки проекту нової рецептури може бути сформульована інакше, а саме, як задача розробки рецептури такого виробу, який би задовольняв тій або іншій сукупності наперед заданих вимог та умов.

Велика кількість видів сировини, що використовується у рецептурі, різноманітність їх хімічного складу, показників, яким повинен задовольняти майбутній новий виріб – все це нерідко робить задачу створення проекту рецептури досить складною. Пошук раціональних розв'язків її неможливо здійснити без застосування математичного апарату та комп'ютерних технологій, що вимагає послідовного виконання низки робіт технологічного, математичного, інформаційного, обчислювального і знову технологічного характеру. Сформулюємо найбільш важливі етапи цієї роботи:

1. Технологічне формулювання задачі розробки нової рецептури.
2. Розробка математичної моделі задачі вибору оптимальної рецептури.
3. Підготовка необхідних початкових даних.
4. Підготовка відповідного алгоритму та програмного забезпечення на ПК.

5. Здійснення розрахунків на ПК.
6. Технологічний аналіз одержаних результатів та вибір оптимальної рецептури виробу, який задовольняє тим або іншим вимогам та умовам.

Задачі вибору нових рецептур відносяться до класу так званих екстремальних задач, тобто таких, у яких забезпечується максимум або мінімум цільової функції.

Нижче для ілюстрації розглянемо одну із актуальних задач розробки рецептури страви лікувального призначення [34].

Розробка рецептури страви з максимально можливим вмістом кальцію

Харчування – один з найважливіших факторів, що визначає стан здоров'я населення. Особливу роль у харчуванні відіграє кальцій. Нестача його у порівнянні з іншими елементами спостерігається найчастіше. Без кальцію не можливе утворення в організмі кісткової тканини, зубів. Він необхідний для підтримки нервово-м'язової збуджуваності, приймає участь у процесі згортання крові, впливає на проникність кліткових оболонок, зменшує прояву алергічних реакцій організму, тощо. Нестача кальцію може призвести у дитячому віці до розвитку рахіту, у підлітків – до затримки росту, у жінок – до ускладнень вагітності та періоду менопаузи, у літніх людей – до остеопорозів тощо. Кальцій є природнім антагоністом стронцію, що робить його незамінним для профілактики та при лікуванні променевиx захворювань.

Згідно до фізіологічних норм харчування, визначених МОЗ України, потреба організму в кальції становить: для дітей – 1200, для дорослих – 1100 мг на добу. Для забезпечення фізіологічної потреби організму випускаються фармацевтичні препарати, що вміщують кальцій, але з фізіологічної точки зору вважається доцільним задовольняти потреби

організму не за рахунок лікарських препаратів, а шляхом надходження необхідних компонентів з харчовими продуктами.

Науковцями ХДУХТ запропоновано як джерело кальцію – напівфабрикат кістковий харчовий (НКХ). Цей продукт вміщує біоорганічний кальцій у легкозасвоюваній формі. Вміст кальцію в ньому – 15000 мг у 100г продукту, для порівняння вміст у молоці – 121, у твердому сирі до 1040.

Засвоювання кальцію організмом суттєво залежить від взаємодії його з іншими нутрієнтами, що входять до складу страв. Розглянемо головні з них.

Співвідношення між вмістом кальцію та фосфору

Для утворення легкорозчинних і легко всмоктуваних двоосновних фосфорнокислих солей оптимальним є співвідношення Са:Р = 1:1. За даними інших дослідників Са:Р = 2:1. Збільшення вмісту фосфору погіршує кальцієвий баланс шляхом створення у шлунково-кишковому тракті (ШКТ) нерозчинного фосфорнокислого кальцію.

Слід зазначити, що у м'ясі вміст кальцію у 10-20 разів менший, ніж фосфору. Тому доцільно для засвоювання кальцію додатково вводити у м'ясну продукцію добавки, що містять кальцій.

Співвідношення між вмістом кальцію та жиру

На метаболізм кальцію значно впливає кількість жиру у страві. Оптимальним вважається співвідношення 10-15 мг кальцію на 1 г жиру (тобто приблизно 1:100÷1:67). Вважають, що зміщення цього співвідношення у будь-яку із сторін негативно впливає на процес всмоктування кальцію із ШКТ.

Взаємодія кальцію з білком

Щодо взаємодії кальцію з білком існує думка, що підвищений вміст білку призводить до втрат кальцію. Вважають, що на кожні 50 г білку в їжі з сечею виводиться додатково 60 мг кальцію, що призводить до

негативного кальцієвого балансу. Новітні дослідження не підтверджують цю думку, визначивши, що додаткове виведення кальцію спостерігається лише нетривалий час.

Задача. Створити рецептуру виробу з м'яса (м'ясних котлет), збагачених на кальцій за рахунок додаткового введення в рецептуру кальційвміщуючого напівфабрикату (НКХ), з урахуванням вимог щодо співвідношення нутрієнтів, що впливають на засвоєння кальцію організмом та органолептичні показники виробу:

- співвідношення між вмістом кальцію та фосфору;
- співвідношення між вмістом кальцію та жиру;
- обмеження на вміст білку у страві;
- співвідношення між вмістом м'яса та хліба;
- співвідношення між вмістом НКХ та м'яса;
- обмеження на вміст інших інгредієнтів у рецептурі.

Позначимо:

x_1 – м'ясо яловичини (г);

x_2 – кістковий харчовий напівфабрикат (паста КХН, г);

x_3 – хліб пшеничний (г);

x_4 – вода (г);

x_5 – сухарі (г);

x_6 – жир тваринний (г);

x_7 – сіль (г);

x_8 – сало (г);

$c_i - i = (\overline{1,8})$ вміст кальцію в 1 г сировини i -го виду;

$f_i - i = (\overline{1,8})$ вміст фосфору в 1 г сировини i -го виду;

$\beta_i - i = (\overline{1,8})$ вміст білку в 1 г сировини i -го виду (визначений за фізіологічною нормою);

$g_i - i = (\overline{1,8})$ вміст жиру в 1 г сировини i -го виду;

B', B'' – відповідно мінімальний та максимальний вміст білку.

Математичне формулювання задачі

Знайти вектор $\bar{X} = (x_1, x_2, \dots, x_8)$, координати якого максимізують

цільову функцію $Z = \sum_{i=1}^8 c_i x_i \rightarrow \max$ та задовольняють таким умовам:

1) співвідношенню між вмістом кальцію та фосфору

$$1 \leq \frac{\sum_{i=1}^8 c_i x_i}{\sum_{i=1}^8 f_i x_i} \leq 2; \quad (7.7.1)$$

2) співвідношенню між вмістом кальцію та жиру

$$67 \leq \frac{\sum_{i=1}^8 g_i x_i}{\sum_{i=1}^8 c_i x_i} \leq 100; \quad (7.7.2)$$

3) обмеженню на мінімальний та максимальний вміст білку

$$10 \leq \sum_{i=1}^8 \beta_i x_i \leq 39; \quad (7.7.3)$$

4) обмеженню на мінімальний вміст м'яса

$$x_1 \geq 50; \quad (7.7.4)$$

5) обмеженню на мінімальний вміст інших інгредієнтів

$$1 \leq x_7 \leq 1,2; \quad (7.7.5)$$
$$x_i \geq 0, \quad i = 2, 3, 4, 5, 6, 8;$$

6) умові стосовно сумарної ваги набору сировини

$$\sum_{i=1}^8 x_i = 100; \quad (7.7.6)$$

7) співвідношенню між вмістом хліба та м'яса

$$0,22x_1 \leq x_3 \leq 0,25x_1; \quad (7.7.7)$$

8) співвідношенню між вмістом НКХ та м'яса

$$0,05x_1 \leq x_2 \leq 0,1x_1. \quad (7.7.8)$$

Аналіз розробленої математичної моделі показує, що це модель лінійного програмування.

В таблиці 7.3 надані вихідні дані:

Таблиця 7.3

№ з/п	Назва сировини	Вміст у 1г сировини, г			
		Білок, β_i	Жир, g_i	Кальцій, c_i	Фосфор, f_i
1.	Яловичина	$18,9 \cdot 10^{-2}$	$16,6 \cdot 10^{-2}$	$9 \cdot 10^{-5}$	$198 \cdot 10^{-5}$
2.	Паста КХН	$12,2 \cdot 10^{-2}$	$11 \cdot 10^{-2}$	$15 \cdot 10^{-2}$	$4 \cdot 10^{-2}$
3.	Хліб пшеничний	$7,6 \cdot 10^{-2}$	$0,9 \cdot 10^{-2}$	$26 \cdot 10^{-5}$	$83 \cdot 10^{-5}$
4.	Вода	–	–	$4,1 \cdot 10^{-5}$	–
5.	Сухарі	$11,2 \cdot 10^{-2}$	$1,4 \cdot 10^{-2}$	$37,0 \cdot 10^{-5}$	$124 \cdot 10^{-5}$
6.	Жир тваринний	–	$99,7 \cdot 10^{-2}$	–	–
7.	Сіль	–	–	$4,85 \cdot 10^{-3}$	–
8.	Шпиг свинячий	$1,4 \cdot 10^{-2}$	$92,8 \cdot 10^{-2}$	–	–

Запишемо початкові дані задачі за допомогою векторів В – вміст білку у продуктах, G – жиру, С – кальцію, F – фосфору. Математичну модель проектування рецептури виробу запишемо у векторному виді і розв’яжемо з використанням вбудованої функції Maximize (рис 7.45).

ORIGIN := 1

$$B := \begin{pmatrix} 0.189 \\ 0.122 \\ 0.076 \\ 0 \\ 0.112 \\ 0 \\ 0 \\ 0.014 \end{pmatrix} \quad G := \begin{pmatrix} 0.166 \\ 0.11 \\ 0.009 \\ 0 \\ 0.014 \\ 0.997 \\ 0 \\ 0.928 \end{pmatrix} \quad C := \begin{pmatrix} 0.00009 \\ 0.15 \\ 0.00026 \\ 0.000041 \\ 0.00037 \\ 0 \\ 0.00485 \\ 0 \end{pmatrix} \quad F := \begin{pmatrix} 0.00198 \\ 0.04 \\ 0.00083 \\ 0 \\ 0.00124 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$f(x) := C \cdot x \quad x := (0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0)^T$$

Given

$$F \cdot x \leq C \cdot x \leq 2 \cdot F \cdot x$$

Співвідношення між вмістом кальцію та фосфору (7.7.1)

$$67 \cdot C \cdot x \leq G \cdot x \leq 100 C \cdot x$$

Співвідношення між вмістом кальцію та жиру (7.7.2)

$$10 \leq B \cdot x \leq 39$$

Обмеження на мінімально та максимально можливий вміст білку (7.7.3)

$$x_1 \geq 50$$

Обмеження на мінімальний вміст м'яса (7.7.4)

$$1 \leq x_7 \leq 1.2$$

$$x_2 \geq 0 \quad x_5 \geq 0$$

$$x_3 \geq 0 \quad x_6 \geq 0$$

$$x_4 \geq 0 \quad x_8 \geq 0$$

Обмеження на інші інгредієнти (7.7.5)

$$\sum x = 100$$

Умова стосовно сумарної ваги набору сировини (7.7.6)

$$0.22 x_1 \leq x_3 \leq 0.25 x_1$$

Співвідношення між вмістом хліба та м'яса (7.7.7)

$$0.05 \cdot x_1 \leq x_2 \leq 0.1 x_1$$

Співвідношення між вмістом НКХ та м'яса (7.7.8)

$$X := \text{Maximize}(f, x)$$

$$X^T = (57.94 \ 3.39 \ 12.75 \ 0 \ 0 \ 24.92 \ 1 \ 0)$$

$$Z := f(X)$$

$$Z = 0.522$$

Рис. 7. 45.

Згідно з розрахунками, оптимальна рецептура котлети містить м'яса яловичини (x_1) - 57, 9 г, напівфабрикату кісткового харчового (x_2), – 3, 4 г, хліба пшеничного(x_3) - 12, 8 г, жиру тваринного (x_6) – 24, 9 г, солі кухонної (x_7) – 1 г і забезпечує 522 мг кальцію.

Таким чином, в результаті застосування математичних методів та комп'ютерних технологій спроектовано оптимальну рецептуру м'ясного виробу з максимально можливим вмістом кальцію у сполученні з іншими нутрієнтами, що сприяють його найкращому засвоєнню.

Запропонований підхід має перспективу визначення оптимального вмісту та додержання умов взаємодії нутрієнтів не тільки у окремому виробі, а й в раціоні харчування. Це, в свою чергу, відкриває можливості створення продукції нового покоління з чітко заданими властивостями, оскільки майбутнє за індивідуалізацією потреби людини у харчових речовинах та енергії.

Питання для самоконтролю

14. Як записуються числа в MathCAD?
15. В чому полягає різниця між вектором та матрицею в MathCAD?
16. Як можна задати матрицю в MathCAD?
17. За допомогою якої панелі інструментів вводяться оператори основних арифметичних дій?
18. За допомогою якої панелі інструментів вводяться знаки « \leq », « \geq »?
19. Як задати границю ОДР задачі ЛП при розв'язуванні її графічним методом в MathCAD?
20. Як знайти ОДР задачі ЛП при розв'язуванні її графічним методом в MathCAD?
21. Як зобразити лінії рівня цільової функції в MathCAD?
22. Як можна знайти розв'язок системи лінійних рівнянь?
23. Назвіть, які обов'язкові елементи повинна містити функція Find?

24. Як реалізувати перетворення Жордана-Гаусса в MathCAD?
25. Як реалізувати симплексне перетворення в MathCAD?
26. Перелічить етапи розв'язання задачі ЛП з використанням вбудованої функції Maximize/Minimize?
27. Як розв'язати транспортну задачу в MathCAD? Назвіть етапи розв'язування ТЗ.

ДОДАТОК А

Приклади простих обчислень та робота з текстом

Весь аркуш MathCAD призначений для розрахунків. Місце, в якому буде вводиться формула позначається червоним хрестиком. Після початку вводу формули хрестик змінюється на блакитний кут, нижня лінія якого показує до якої частини виразу відноситься дія. Змінити довжину лінії можна за допомогою клавіші пробіл.

Приклад. Обчислити $\frac{1+2}{5}$ та $1+\frac{2}{5}$. В першому випадку потрібно набрати 1+2 і виділити обидві цифри 1+2 після чого набрати «/» з клавіатури або з панелі «Calculator». В другому випадку виділити потрібно тільки останню цифру 2.

Числа задаються за допомогою арабських цифр, десяткової **точки** (а не коми) та знака «-» (мінус). Наприклад, дві цілі одна десята – 2.1. Дуже великі або маленькі числа можна записувати за допомогою мантиси та порядку. Наприклад, $12.3 \cdot 10^{25}$. Знак множення * при введенні числа на екран змінюється на крапку, а операція піднесення до степені (символ «^» з клавіатури) відображається у вигляді надрядкового елемента. Десятинні числа мають основу 10. Діапазон їх можливих значень лежить в межах від 10^{-15} до 10^{307} (це машинний нуль та машинна нескінченність).

Імена змінних та функцій можуть складатися з великих та маленьких літер, грецьких символів, знаку підкреслювання (_), символів штрих, відсоток, нескінченність та нижніх символів (потрібно відрізнити індекс (вводиться як «[» з клавіатури) – посилання на елемент матриці або вектора та нижній регістр (вводиться як «.» з клавіатури) – варіант написання імені), але починатись повинні з букви.

Потрібно пам'ятати, що MathCAD **відрізняє** букви, які виглядають *однаково*, але набрані в різних мовах, наприклад, англійська A та російська або українська А, а також великі та маленькі літери.

Для того, щоб визначити змінну x необхідно ввести символ x та « := » або « ≡ » з панелі «Evaluation» («Вычисления») (з клавіатури знак « : » або « ~ » відповідно). Перший знак означає локальне, другий глобальне присвоювання. Локально визначену змінну x можна використовувати в обчисленнях всюди **нижче і праворуч** рівності, де вона визначена, глобальне присвоювання діє по всьому документу, але використати її в іншому глобальному присвоюванні можна тільки нижче та праворуч рівності. Щоб подивитись, чому дорівнює значення змінної, потрібно набрати $x =$.

Для того щоб обчислити будь-який вираз в діапазоні значень x , необхідно вибрати на панелі «Matrix» («Матрицы») команду « m..n » – «задати як дискретну змінну» (з клавіатури « ; »). Потрібно вказати перше значення, що приймає x , набрати кому та наступне значення, кінцеве значення. Наприклад, $x := 1,1.2..2$ – x приймає значення 1, 1.2, 1.4, 1.6, 1.8, 2. Якщо наступне значення не вказується, за замовчанням MathCAD бере крок, що дорівнює 1. Наприклад, $x := 1..5$, тоді x приймає значення 1, 2, 3, 4, 5.

MathCAD містить велику кількість вбудованих функцій, але за потреби можна задавати і будь-які функції користувача. Для цього потрібно вказати ім'я функції (вимоги такі як і для імені змінної), її аргументи, знак присвоювання «:=» та вираз функції. Наприклад, $Z(x):=5x$, $my_function(x, y):=x+y^2$. Знак присвоювання можна визвати, натиснувши мишею відповідну кнопку на панелі «Calculator» або «Evaluation» або набравши на клавіатурі клавішу «:».

Функція обов'язково повинна містити аргумент. Якщо змінних декілька, то їх необхідно вказати всі через кому.

Якщо потрібно:

1) знайти значення функції в будь-якій точці, то достатньо підставити її значення замість аргументу. Наприклад, знайти $Z(x)$ при $x=2$. В MathCAD набираємо $Z(2)=$ і програма видає результат 10, або можна зробити так: $x:=2$ $Z(x)=$ – машина видасть той саме результат;

2) обчислити функцію для кожного значення a із деякого діапазону, то потрібно задати діапазон значень a , і підставити a замість аргументу. Наприклад, знайти значення функції $Z(x)$ для всіх x з інтервалу (20, 26) з кроком 2. В MathCAD набираємо $x:=20,22..26$ $Z(x)=$ і машина видасть результат у вигляді таблиці.

$Z(x)=$

100
110
120
130

Для того, щоб ввести текст, необхідно виконати пункт «Text Region» («Создать текстовую область») з меню «Insert» («Вставка»), або натиснути клавішу з подвійними лапками (“). Після того, як виконаєте одну з цих операцій, MathCAD замінить хрестик на вертикальну лінію маркер вводу. Цей маркер буде обрамлений чорною рамкою і позначатиме текстову область. Текст буде вводитися після маркеру, текстова область буде розширюватися до розмірів тексту. Також текст можна вставляти з інших текстових редакторів, наприклад, Word. Це можна зробити двома способами: або безпосередньо вставити в MathCAD–документ скопійований текст (Ctrl + V або Shift + Insert), або

вставити його в попередньо відкриту текстову область. В першому випадку текст буде вставлений в вигляді об'єкта OLE, і для його редагування буде викликатися та програма, в якій він був створений, в другому випадку редагування можливе і в MathCAD. Перший спосіб зручніше, коли текст повинен бути добре відформатований – наприклад, в MathCAD немає системи автоматичних переносів.

ДОДАТОК Б

Робота з матрицями та векторами

Вектор (матрицю) можна задати як дискретну величину, якщо відома формула для його елементів (наприклад, вектор $n=(0,1,2,3)$; набираємо $i:=0..3$, та $n_i:=i$) або за допомогою команди «Matrix or Vector» («Вставити матрицю») з панелі «Matrix» («Матриць»): з'являється форма, в якій потрібно вказати кількість рядків та стовпців, а потім ввести елементи вектора (матриці).

Вектором в MathCAD називається матриця, що має один стовпець. Потрібно відрізнити поняття матриці та вектора, тому що деякі операції та деякі функції можливо застосувати тільки до вектора. Наприклад, операція векторний добуток $\vec{X} \times \vec{Y}$, або функція $\text{length}(x)$ – знаходить кількість елементів вектора. Якщо вектор має велику кількість елементів, відображати на екрані його зручніше у вигляді рядка за допомогою функції транспонування (T).

Потрібно пам'ятати, що перший елемент матриці має індекс 0. Для того щоб змінити початковий індекс на одиницю, використовують функцію ORIGIN.

Оператори для векторів та матриць		
Назва оператора	набір з клавіатури	позначення на панелі «Матрицы»
Векторний добуток	Ctrl + 8	$\vec{X} \times \vec{Y}$
Визначник		$ X $
Скалярний добуток	*	$\vec{X} \cdot \vec{Y}$
Обернена матриця	\wedge_{-1}	A^{-1}
Множення матриць	*	$A \cdot B$
Нижній індекс вектор(матриця)	[X_n
Верхній індекс (виводить вказаний стовпчик)	Ctrl + 6	$M^{<n>}$
Транспонування	Ctrl + 1	A^T
Оператор векторизації*	Ctrl + -	$\overline{f(M)}$
Утворити матрицю	Ctrl + M	$\begin{bmatrix} \vdots \\ \vdots \\ \vdots \end{bmatrix}$
Сума елементів вектору	Ctrl + 4	$\sum v$

* Операція векторизації працює так: вказана операція виконується для відповідних елементів векторів (матриць). Наприклад $\overline{A + BC}$, виконується $a_i + b_i c_i$ для всіх i . Всі матриці повинні мати однакову розмірність.

Нижче приведені результати деяких операцій.

ORIGIN := 1

$A := \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$

$B := \begin{pmatrix} 2 \\ 4 \\ -1 \end{pmatrix}$

$C := \begin{pmatrix} 1 & 5 & 1 \\ 2 & 5 & 2 \\ 4 & 1 & 3 \end{pmatrix}$

Елементи $A_1 = 1$ $C_{1,1} = 1$

Скалярний добуток $A \cdot B = 7$

Векторний добуток $A \times B = \begin{pmatrix} -14 \\ 7 \\ 0 \end{pmatrix}$

Операція векторизації $\overrightarrow{(A \cdot B)} = \begin{pmatrix} 2 \\ 8 \\ -3 \end{pmatrix}$

Визначник C $|C| = 5$

Обернена матриця C $C^{-1} = \begin{pmatrix} 2.6 & -2.8 & 1 \\ 0.4 & -0.2 & 0 \\ -3.6 & 3.8 & -1 \end{pmatrix}$

Транспонований вектор A $A^T = (1 \ 2 \ 3)$

Матрицы ✖

$\begin{bmatrix} \vdots \\ \vdots \\ \vdots \end{bmatrix}$ \times_n \times^{-1} $|x|$

$\overrightarrow{f(x)}$ $M^{<n>}$ M^T $m..n$

\otimes \cdot \cdot \otimes \otimes Σ \int $\frac{d}{dx}$

ДОДАТОК В

Основні матричні функції.

Функція	Повертає
cols(A)	число стовпців матриці A
rows (A)	число рядків матриці A
last(v)	індекс останнього елементу вектора v
length(v)	число елементів вектора v
max(A)	найбільший елемент матриці A
min(A)	найменший елемент матриці A
csort(A, n)	сортує стовпці за зростанням елементів у рядку n
rsort(A, n)	сортує рядки за зростанням елементів у стовпці n
sort (v)	сортує за зростанням елементи вектора v
reverse(A)	вектор (матрицю), елементи (рядки) якого записані у оберненому порядку
diag (v)	діагональна матриця, що містить на діагоналі елементи вектора v
identify(n)	одичну матрицю розміру $n \times n$ (n – натуральне)
matrix (m, n, f)	створює матрицю розміру $m \times n$, у якій (i, j) -ий елемент має значення $f(i, j)$, де $i = 0..m-1$, $j = 0..n-1$.
augment(A, B)	до матриці A справа приєднується матриця B (матриці повинні мати однакову кількість рядків)
stack (A, B)	до матриці A знизу приєднується матриця B (матриці повинні мати однакову кількість стовпців)

Функція	Повертає
submatrix(A,a,b,c,d)	матриця, яка складається з елементів A, що знаходяться на перетині рядків з a по b та стовпців з c по d
lsolve(M,v)	вектор x, який є розв'язком рівняння Mx=v
rank(A)	ранг матриці A
lookup(z, A, B)	в матриці A шукає значення $a_{i,j} = z$, повертає елементи B, які мають такі саме індекси

Приклади застосування деяких функцій наведено нижче.

ORIGIN := 1

i := 1..3

$$A_i := \sin(2 \cdot i) + 1 \quad A = \begin{pmatrix} 1.909 \\ 0.243 \\ 0.721 \end{pmatrix} \quad B := \begin{pmatrix} 1 & 5 \\ 5 & 2 \\ 3 & 1 \end{pmatrix}$$

Елемент вектора A

$$A_1 = 1.909$$

Елемент матриці B

$$B_{1,1} = 1$$

Число стовпців B

$$\text{cols}(B) = 2$$

Число рядків B

$$\text{rows}(B) = 3$$

Індекс останнього елемента A

$$\text{last}(A) = 3$$

Сортування B по першому стовпцю

$$\text{csort}(B, 1) = \begin{pmatrix} 1 & 5 \\ 3 & 1 \\ 5 & 2 \end{pmatrix}$$

Сортування B по другому стовпцю

$$\text{csort}(B, 2) = \begin{pmatrix} 3 & 1 \\ 5 & 2 \\ 1 & 5 \end{pmatrix}$$

Сортування A

$$\text{sort}(A) = \begin{pmatrix} 0.243 \\ 0.721 \\ 1.909 \end{pmatrix}$$

Виділення мінору матриці B, що стоїть на перехресті другого-третього рядка та першого-другого стовпця

$$\text{submatrix}(B, 2, 3, 1, 2) = \begin{pmatrix} 5 & 2 \\ 3 & 1 \end{pmatrix}$$

Приєднання до A матриці B справа

$$\text{augment}(A, B) = \begin{pmatrix} 1.909 & 1 & 5 \\ 0.243 & 5 & 2 \\ 0.721 & 3 & 1 \end{pmatrix}$$

Приєднання до A^T транспонованої матриці B знизу

$$\text{stack}(A^T, B^T) = \begin{pmatrix} 1.909 & 0.243 & 0.721 \\ 1 & 5 & 3 \\ 5 & 2 & 1 \end{pmatrix}$$

Виділення другого стовпця B

$$B^{<2>} = \begin{pmatrix} 5 \\ 2 \\ 1 \end{pmatrix}$$

Виділення другого рядка B

$$(B^T)^{<2>T} = (5 \ 2) \quad \text{або}$$

$$\text{submatrix}(B, 2, 2, 1, 2) = (5 \ 2)$$

ДОДАТОК Г

Побудова графіків у MathCAD

Пакет MathCAD має великі графічні можливості. В MathCAD є такі види графіків: декартовий, полярний, поверхні, карта ліній рівня, векторне поле, тривимірний точковий, тривимірна стовпчаста діаграма. Всі графіки є стандартними об'єктами MathCAD: їх можна редагувати, а при перерахунку вихідних даних вони автоматично перерисовуються.

Декартовий графік будується, як правило, в три кроки:

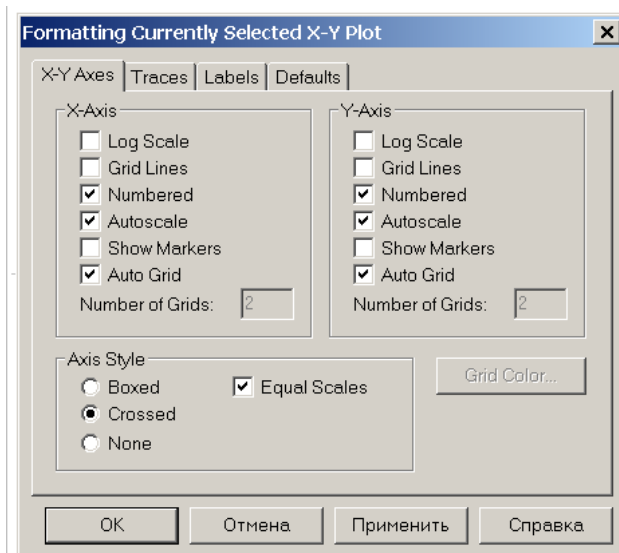
- крок 1: формування вектора значень аргументу;
- крок 2: завдання виду функції однієї змінної;
- крок 3: побудова графіку.

Третій крок в свою чергу можна поділити ще на три кроки:

- крок 3.1: виклик на екран дисплею шаблону графіка – прямокутника з чорними квадратами ліворуч та знизу; шаблон графіка з'являється у відміченому курсором місці після того, як користувач натисне одну з кнопок панелі «Graph» («Графіки»);

- крок 3.2: заповнення користувачем двох чорних квадратів шаблону графіка ім'ям функції та ім'ям аргументу. Якщо функцій більше за одну, то їх імена вводяться через кому. В заготовці є й інші чорні квадрати, які можна не заповнювати. Середовище MathCAD заповнить їх автоматично. Графік з'являється на дисплеї після виводу курсору з зони графіка. Параметри графіка задаються стандартами за замовчанням;

- крок 3.3 необхідний, якщо параметри графіка, встановлені за замовчанням не задовольняють користувача і він хоче їх змінити. Клацнувши двічі по графіку з'являється панель «Formatting Currently Selected X-Y Plot» («Форматування вибраного графіка»).



За допомогою прапорців можна змінити:


- **Grid lines** (Лінії сітки) — показати лінії сітки;
- **Numbered** (Нумерація) — показати нумерацію шкали. Якщо прибрати цей прапорець, то числа, що розмічають шкалу, пропадуть;
- **Autoscale** (Автоматичний масштаб) — вибір діапазону осі проводиться автоматично процесором Mathcad;
- **Show markers** (Показати маркери) — виділення одного або двох додаткових значень на осях пунктирними лініями;
- **Autogrid** (Автоматична шкала) — розбиття шкали проводиться автоматично процесором Mathcad. Якщо цей прапорець знятий, в полі введення поряд з ним слід вказати бажану кількість міток шкали;
- **Equal scales** (Однаковий масштаб) — осі x і y зображаються в однаковому масштабі;
- **Axis Style** (Вид осі) — можна вибрати один з трьох видів системи координат:
 - **Boxed** (Прямокутник) – графік зображається в прямокутній області;

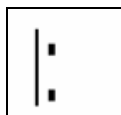
- Crossed (Перетин) – координатні осі у вигляді двох прямих, що перетинаються;
- None (Немає) – координатні осі не показуються на графіку.

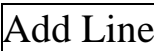
ДОДАТОК Д


Елементи програмування

Розглянемо основні команди панелі «Programming» («Программирование»). Програмні оператори можна визвати, або натиснувши відповідну кнопку панелі «Programming», або набравши з клавіатури поєднання спеціальних клавіш.

 – викликає шаблон для створення блоків виразів (його також можна викликати натиском клавіші «] ».). Шаблон має вид:




Якщо програма містить більше рядків, їх можна додати повторно натиснувши клавішу .

 – знак присвоювання змінній конкретного значення. Його можна викликати натиском клавіші « { ».

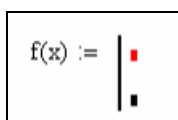


В першу позначку вводиться ім'я змінної, в другу – її значення.

 – викликає шаблон для введення умовного оператора if (якщо) в програму. Його можна викликати також натиском клавіші « } »). В другу позначку вводиться умова, а в першу – результат виконаної дії.

Приклад. Задати функцію, що дорівнює 1 для додатних значень аргументу, 0 – для всіх інших.

Наберіть « $f(x):=$ » і в позначку введіть оператор Add Line:



У всі позначки введемо умовний оператор if, тобто натиснемо «if» на панелі управління. На екрані будемо мати

$$f(x) := \begin{cases} \blacksquare \text{ if } \blacksquare \\ \blacksquare \text{ if } \blacksquare \end{cases}$$

Задамо відповідні умови та вирази для обчислення функції. Для введення символів «>», «□» використаємо панель «Boolean» («Булевы»). Маємо

$$f(x) := \begin{cases} 1 \text{ if } x > 0 \\ 0 \text{ if } x \leq 0 \end{cases}$$

otherwise – дозволяє перетворити неповну альтернативу в повну (його можна викликати натиском клавіш Ctrl + }).

Приклад.

Задати функцію двох аргументів, що дорівнює 1, якщо обидва аргументи додатні, та 0 в інших випадках.

Функція задається аналогічно до попередньої, тільки в нижню позначку замість команди «if» вводимо «otherwise».

$$f(x, y) := \begin{cases} 1 \text{ if } x > 0 \wedge y > 0 \\ 0 \text{ otherwise} \end{cases}$$

while – викликає шаблон оператора while для створення оператора циклу з передумовою. Шаблон має вигляд

```
while ■
  ■
```

В верхній позначці вводиться логічний (булевий) вираз, в нижній – оператор циклу. Перевірка правильності логічного виразу відбувається перед виконанням нової ітерації. Оператор можна викликати натиском

клавіш Ctrl +].

Приклад. Задати функцію, що знаходить факторіал.

Програма обчислення матиме вигляд:

```
fact(n) := | f ← 1
           | k ← 1
           | while k ≤ n
           |   | f ← f·k
           |   | k ← k + 1
           | f
```

for – кнопка для вводу в програми циклу з параметром. Введення оператора for можна також здійснити комбінацією клавіш Ctrl + ".

```
for ■ ∈ ■
  ■
```

В першу позначку вводиться ім'я параметру циклу, в другій задається початкове, кінцеве значення параметру, крок, на який змінюється параметр циклу. В нижню позначку записується тіло циклу.

Приклад. Задати функцію, що знаходить факторіал.

```
fact1(n) := | f ← 1
            | for k ∈ 1..n
            |   f ← f·k
            | f
```

ДОДАТОК Е

Список деяких вбудованих функцій

У наведеній таблиці x, y – дійсні числа; z – дійсне або комплексне число, n – натуральне число. Всі кути вимірюються у радіанах. Імена функцій слід писати так само як вони наведені нижче:

Функція	Повертає
$\text{acos}(z)$	арккосинус
$\text{asin}(z)$	арксинус
$\text{atan}(z)$	арктангенс
$\text{cos}(z)$	косинус
$\text{cot}(z)$	котангенс
$\text{sin}(z)$	синус
$\text{tan}(z)$	тангенс
$\text{angle}(x, y)$	кут між віссю Ox та точкою (x, y)
$\text{exp}(z)$	експонента e^z
$\text{ceil}(x)$	найменше ціле, що більше або дорівнює x ($\text{ceil}(1.1)=2$)
$\text{floor}(x)$	найбільше ціле, що менше або дорівнює x ($\text{floor}(1.1)=1$)
$\text{round}(x, n)$	x округлює до n знаків після точки ($\text{round}(1.235, 2)=1.24$)
$\text{ln}(z)$	натуральний логарифм z
$\text{log}(z, b)$	логарифм z за основою b , якщо b опустити, основа дорівнює 10
$\delta(m, n)$	символ Кронекера = $\begin{cases} 1, & \text{якщо } m = n \\ 0, & \text{якщо } m \neq n \end{cases}$
$\Phi(x)$	сходинова функція Хевісайда = $\begin{cases} 1, & \text{якщо } x \geq 0 \\ 0, & \text{якщо } x < 0 \end{cases}$

ДОДАТОК Є

Змінні з їх значеннями за замовчанням

Змінна	Значення
$\pi = 3,14159\dots$	Число π . У чисельних розрахунках MathCAD використовує 15 значущих цифр числа π . Щоб надрукувати π натисніть Ctrl+Shift+p
$e = 2,71828\dots$	Основа натурального логарифма
$\infty = 10^{307}$	Нескінченність. Щоб надрукувати ∞ натисніть Ctrl+Shift+z
ORIGIN = 0	Початок масиву. Означає індекс першого елемента масиву

ЛІТЕРАТУРА

1. Балашевич В.А. Математические методы в управлении производством. – Минск: Высшая школа, 1976. –336с.
2. Бейко И.В. и др. Методы и алгоритмы решения задач оптимизации / И.В. Бейко, Б.Н. Бублик, П.Н. Зинько. – Киев: Вища школа, 1983. – 512с.
3. Вентцель Е.С. Исследование операций. – М.: Высшая школа, 2007. – 208с.
4. Вентцель Е.С. Исследование операций: задачи, принципы, методология. – М.: Кнорус, 2010.–192с.
5. Вітлінський В.В., Наконечний С.І., Терещенко Т.О. Математичне програмування: Навч.-метод. Посібник для самостійного вивчення дисципліни. – К: КНЕУ, 2006. –248с.
6. Гетманцев В.Д. Лінійна алгебра і лінійне програмування: Навч. посіб. - К.: Либідь, 2001. – 256с.
7. Гольштейн Е.Г., Юдин Д.Б. Задачи и методы линейного программирования. Задачи транспортного типа. – М.: Либроком, 2010.– 184с.
8. Деордица Ю.С., Савченко В.Т. Компьютерные технологии в экономике и менеджменте: Учеб. пособие. -Луганск: ВУГУ, 1999.–212с.
9. Дж. фон Нейман, О. Моргенштерн. Теория игр и экономическое поведение. – М.: Наука, 1970.–708с.
10. Ермольев Ю.М., Ляшко И.И., Михалевич В.С., Тюття В.И. Математические методы исследования операций – Киев: Вища школа, 1979. –312с.
11. Зайченко Ю.П. Исследование операций. – Киев: Слово, 2003. – 688с.
12. Испирян Г.П., Рожок В.Д., Романюк Т.П. Математические методы и модели в планировании и управлении в легкой промышленности. – Киев: Вища школа, 1978. – 279с.
13. Калихман И.Л. Линейная алгебра и программирование. – М.: Высшая школа, 1967. – 427с.
14. Карпелевич Ф.И., Садовский Л.Е. Элементы линейной алгебры и линейного программирования. – М.: Наука, 1967. – 312с.
15. Кирьянов Д.В. Самоучитель Mathcad 11. – СПб: БХВ-Петербург, 2003. –520с.
16. Корбут А.А., Финкельштейн Ю.Ю. Дискретное программирование. – М.: Наука, 1969. – 368с.
17. Крутовой Ж.А. Экономико-математические методы в торговле и общественном питании: Учеб. пособие/Ч. 2 – Харьков – 1991. – 265с.
18. Крутовой Ж.А., Черкашина Л.В. Прибыль можно увеличить, если изменить производственную программу // Общественное питание. – 1979. – № 8.
19. Крушевский А.В. Теория игр. – Киев: Вища школа, 1977. – 216с.

20. Кузнецов Ю.Н. и др. Математическое программирование / Ю.Н. Кузнецов, В.И. Кузубов, А.Б. Волощенко. – М.: Высшая школа, 1980. – 300с.
21. Ляшенко И.Н, Карагодова Е.А, Черникова Н.В., Шор Н.З. Линейное и нелинейное программирование / Под ред. И.Н. Ляшенко. – Киев: Вища школа, 1975. – 372с.
22. Мазаракі А.А., Толбатов Ю.А. Математичне програмування в Excel: Навч. посіб. - К.: Четверта хвиля, 1998. –208с.
23. Матряшин Н.П., Макеева В.К. Математическое программирование. – Харьков: Вища школа, 1978. –160с.
24. Охорзин В.А. Оптимизация экономических систем. Примеры и алгоритмы в среде Mathcad: Учеб. пособие. – М.: Финансы и статистика, 2005. – 144с.
25. Пападимитриу Х., Стайглиц К. Комбинаторная оптимизация: Алгоритмы и сложность. – М.: Мир, 1985. – 512с.
26. Пенина Г.Г. Экономико-математические методы в торговле. – Киев: Вища школа, 1984.
27. Спирин А.А., Фомин Г.П. Экономико-математические методы и модели в торговле. – М.: Экономика, 1988. –148с.
28. Таха Хемди А. Введение в исследование операций М.: Вильямс, 2005. – 912с.
29. Черевко О.І., Крутовий Ж.А., та ін. Розробка рецептур страв для добового раціону харчування з максимально можливим вмістом кальцію // Прогресивні техніка та технології харчових виробництв ресторанного господарства і торгівлі: Зб. наук. праць. – Вип.2(4). – Х.: ХДУХТ, 2006.
30. Щедрин Н.И., Кархов А.Н. Математические методы программирования в экономике. – М.: Статистика, 1974.– 143с.
31. Щедрин Н.И., Кархов А.Н. Экономико-математические методы в торговле. – М.: Экономика, 1980. –176с.