**Київський національний університет
імені Тараса Шевченка**

В.І.Тюптя, В.І.Шевченко, В.К.Стрюк

Дискретне програмування

Методичні вказівки до проведення практичних
та самостійних занять з курсу **“Дослідження операцій”**
для студентів факультету кібернетики

**Київ
Електронна бібліотека факультету кібернетики
2003**

“Дискретне програмування”. Методичні вказівки до проведення практичних та самостійних занять з курсу “Дослідження операцій” для студентів факультету кібернетики / Упорядн. Володимир Іванович Тюптя, Віталій Іванович Шевченко, Віктор Кіндратович Стрюк. — К.: Електронне видання. Електронна бібліотека факультету кібернетики Київського національного університету імені Тараса Шевченка, 2003, — 35 с.

Рецензенти: С.І. Ляшко, д-р фіз.-мат. наук;
В.Ф. Кузенко, канд. фіз.-мат. наук

Затверджено вченою радою
факультету кібернетики
21 жовтня 2002 року

Ел.бібліотека факультету кібернетики КНУ, 2003

§ 1. Постановки задач

Ключовою проблемою сучасної теорії управління є оптимізація. Розв′язання цієї проблеми вимагає розробки та практичного застосування методів оптимізації, що базуються на використанні сучасних ЕОМ. Серед прикладних задач оптимізації важливе місце займають задачі дискретного програмування. При цьому розрізняють задачі комбінаторного типу, допустима множина яких має скінченну кількість точок, задачі цілочисельного програмування, де змінні приймають цілочисельні значення, та задачі частково дискретного програмування, в яких лише частина змінних приймає дискретні значення.

Розглянемо декілька прикладів.

1) Задача про оптимальні призначення.

Є n видів робіт та n кандидатів для їх виконання (виконавців). Вважається, що кожен з кандидатів $i=1,...,n$ може виконувати будь-яку роботу $j=1,...,n$, при цьому c_{ij} — витрати, пов'язані з призначенням i- го кандидата на j- й вид роботи. Необхідно так розподілити кандидатів на виконання робіт, щоб кожен з кандидатів одержав єдине призначення, кожна з робіт одержала єдиного виконавця і сумарні витрати, пов'язані з призначеннями, були мінімальними.

Це типова комбінаторна задача. Її розв'язком є деяке переставлення чисел $1,...,n.$ Число переставлень дорівнює $n!$, тому при великих n розв'язати цю задачу шляхом прямого перебирання усіх можливих переставлень неможливо. Однак сформульовану задачу можна записати у вигляді задачі лінійного програмування (*ЗЛП)* з цілочисельними змінними. Дійсно, нехай $x_{ij}=1$, якщо i- й виконавець призначається на j - у роботу, інакше $x_{ij}=0$. Тоді математична модель задачі про оптимальні призначення приймає таку форму:

знайти матрицю $\mathbf{X}=\left\|x_{ij}\right\|, i,j=1,...,n,$ що мінімізує цільову функцію

$$L(\mathbf{X})=\sum_{i=1}^{n}\sum_{j=1}^{n}c_{ij}x_{ij}$$

при виконанні обмежень

$$\sum_{i=1}^{n}x_{ij}=1,\ j=1,...,n,$$

$$\sum_{j=1}^{n}x_{ij}=1,\ i=1,...,n,$$

$$x_{ij}=0 \text{ або } 1,\ i,j=1,...,n.$$

2) Задача про рюкзак.

Мандрівник, збираючись у похід, має упакувати у рюкзак деякі з n предметів, що можуть бути йому потрібними. При цьому вважається, що відома корисність c_j одного предмета j - го найменування, а також, що у поході можуть бути потрібними декілька однакових предметів. Рюкзак має m обмежень за своїми характеристиками (об'єм, лінійні розміри, вага і т. п.). Нехай a_{ij} — i-а характеристика $(i=1,...,m)$ предмета j- го найменування $(j=1,...,n)$, b_i — максимальне значення i- ї характеристики рюкзака. Необхідно визначити, які предмети та в якій кількості слід завантажити у рюкзак, щоб їх сумарна корисність була максимальною.

Позначимо через x_j кількість предметів j- го найменування, що планується для завантаження у рюкзак, тоді математичне формулювання цієї задачі набирає вигляду:

$$L(\mathbf{x}) = \sum_{j=1}^{n} c_j x_j \to \max,$$

$$\sum_{j=1}^{n} a_{ij} x_j \le b_i, \, i = 1,...,m,$$

$$x_j \ge 0, x_j - \text{ціле}, \; j=1,...,n..$$

Розглянута задача являє собою повністю цілочисельну *ЗЛП*.

3) Задача вибору засобів доставки.

Нехай для перевезення p видів вантажів можна використовувати судна n типів, причому a_k — кількість вантажу k- го виду, g_j — кількість суден типу j, $j=1,...,n$, c_j — витрати, пов'язані з використанням одного судна j- го типу. Кожне судно має m ємностей (палуби, трюми і т. п.), d_{ij} — вантажопідйомність ємності i $(i=1,...,m)$ на судні типу j. Необхідно вибрати найбільш економний комплекс засобів доставки вантажів та план завантаження суден.

Позначимо через $x_j, j=1,...,n,$ кількість суден j - го типу, що планується для перевезень, y_{ik} — кількість вантажу k - го виду, що підлягає завантаженню у ємність i. Тоді, очевидно, задача формально зводиться до вибору змінних x_j, $j=1,...,n,$ $y_{ik}, i=1,...,m,$ $k=1,...,p,$ що мінімізують цільову функцію

$$L(\mathbf{x}) = \sum_{j=1}^{n} c_j x_j,$$

задовольняючи умови

$$\sum_{j=1}^{n} d_{ij} x_j - \sum_{k=1}^{p} y_{ik} \ge 0, \, i = 1,...,m,$$

$$\sum_{i=1}^{m} y_{ik} = a_k, \, k = 1,...,p,$$

$$0 \le x_j \le g_j, x_j - \text{ціле}, \, j=1,...,n,$$

$$y_{ik} \ge 0, i=1,...,m, k=1,...,p.$$

У цій задачі на змінні y_{ik} умова цілочисельності не накладається, тому це частково цілочисельна задача.

Наведемо тепер формальні постановки задач.

Цілочисельною ЗЛП (у стандартній формі) будемо називати задачу знаходження вектора $\mathbf{x} = (x_1,...,x_n) \in E^n$, *що мінімізує цільову функцію*

$$L(\mathbf{x}) = \sum_{j=1}^{n} c_j x_j, \tag{1}$$

задовольняючи умови

$$\sum_{j=1}^{n} a_{ij} x_j = b_i,\ i = 1,...,m, \tag{2}$$

$$x_j \geq 0,\ j = 1,...,n, \tag{3}$$

$$x_j - \text{ціле},\ j = 1,...,k \ \ (k \leq n). \tag{4}$$

Якщо $k = n,$ то (1)–(4) називається *повністю цілочисельною ЗЛП* (*ПЦЗЛП*), інакше — *частково цілочисельною ЗЛП* (*ЧЦЗЛП*).

Дискретною ЗЛП (у стандартній формі) називається така задача:

$$L(\mathbf{x}) = \sum_{j=1}^{n} c_j x_j \to \min, \tag{5}$$

$$\sum_{j=1}^{n} a_{ij} x_j = b_i,\ i = 1,...,m, \tag{6}$$

$$x_j \geq 0,\ j = 1,...,n, \tag{7}$$

$$x_j \in \left\{x_j^1, x_j^2, ..., x_j^{n_j}\right\},\ j = 1,...,k \ \ (k \leq n). \tag{8}$$

Якщо $k = n,$ то задачу (5)–(8) називають *повністю дискретною ЗЛП* (*ПДЗЛП*), інакше — *частково дискретною ЗЛП* (*ЧДЗЛП*).

Як бачимо, сформульовані задачі (1)–(4) та (5)–(8) відрізняються від звичайної *ЗЛП* наявністю умов (4) та (8). Це призводить до необхідності застосування спеціальних методів їх розв'язування.

Звичайно, у деяких випадках з практичної точки зору прийнятними є розв'язки, одержані тим чи іншим наближеним методом. Наприклад, якщо в *ПЦЗЛП* (1)–(4) $x_j,\ j = 1,..., n,$ відображає запланований випуск масової продукції, то можна розв'язати відповідну *ЗЛП* (1)–(3) і заокруглити одержаний розв'язок. Проте слід мати на увазі, що безпосереднє застосування цієї ідеї до розв'язування *ПЦЗЛП* може дати розв'язок далекий від оптимального, що має місце, наприклад, у такій задачі:

$$L(\mathbf{x}) = x_1 - 3x_2 + 3x_3 \to \max,$$

$$\begin{aligned} 2x_1 + x_2 - x_3 &\leq 4, \\ 4x_1 - 3x_2 &\leq 2, \\ -3x_1 + 2x_2 + x_3 &\leq 3, \\ x_j \geq 0,\ x_j - \text{ціле},\ j &= 1,2,3. \end{aligned}$$

Розв'язком цієї *ПЦЗЛП* є $\mathbf{x}^* = (2,2,5).$ Якщо ж розв'язати відповідну *ЗЛП* одним з відомих методів, то одержимо $\mathbf{x}^* = (0.5,0,4.5).$ Зауважимо, що ніякі варіанти заокруглення одержаного розв'язку не дають навіть допустимого розв'язку *ПЦЗЛП*, не кажучи вже про оптимальний.

Далі будуть розглянуті деякі методи розв'язування сформульованих вище задач.

§ 2. Методи відтинання. Перший метод Гоморі

Коротко викладемо основну ідею цих методів на прикладі *ПЦЗЛП*.

Нехай $D = \{\mathbf{x} \in E^n : \mathbf{Ax} = \mathbf{b}, \mathbf{x} \geq \mathbf{0}\}$. У стандартних позначеннях *ПЦЗЛП* зводиться до пошуку

$$\boldsymbol{x}^* = \underset{\boldsymbol{x} \in D,\ \boldsymbol{x}-\text{цілочисельний}}{\operatorname{argmin}} \boldsymbol{cx}. \qquad (9)$$

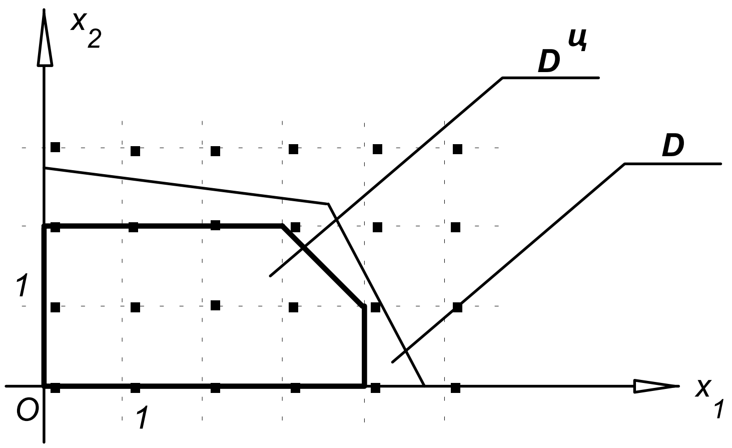

Рис. 1

Позначимо через $D^ц$ опуклу лінійну оболонку цілочисельних точок області D (див. рис. 1) і розглянемо допоміжну *ЗЛП*, що полягає у пошуку

$$\boldsymbol{x}^* = \underset{\boldsymbol{x} \in D^ц}{\operatorname{argmin}} \boldsymbol{cx}. \qquad (10)$$

Легко показати, що всі вершини області $D^ц$ є цілочисельними (тобто, всі базисні розв'язки *ЗЛП* (10) є цілочисельними). Отже, оптимальний розв'язок *ПЦЗЛП* (9) співпадає з оптимальним розв'язком *ЗЛП* (10). Тобто, існує можливість одержати розв'язок *ПЦЗЛП* шляхом розв'язування деякої спеціальним чином побудованої *ЗЛП*, використовуючи при цьому, зрозуміло, добре розроблені методи лінійного програмування.

На жаль, побудова множини $D^ц$, а, отже, і безпосередній перехід до *ЗЛП* (10), є досить складною задачею. Проте існує можливість розв'язати цю задачу у певному розумінні частково, послідовно відтинаючи від множини D деякі її частини за допомогою так званих *відтинаючих площин*.

Ідея *методів відтинання* полягає ось у чому. Розв'язується *ЗЛП*, одержана з *ПЦЗЛП* відкиданням умови цілочисельності змінних. Якщо її розв'язок є цілочисельним, то він же є і розв'язком *ПЦЗЛП*. Якщо ж *ЗЛП* розв'язку не має, то і *ПЦЗЛП* розв'язку не має. Якщо розв'язок *ЗЛП* не є цілочисельним, то від розв'язаної *ЗЛП* переходять до нової допоміжної *ЗЛП* шляхом приєднання лінійного обмеження, яке задовольняють усі цілочисельні розв'язки *ПЦЗЛП*, але не задовольняє одержаний нецілочисельний розв'язок початкової *ЗЛП*. Це додаткове лінійне обмеження визначає деяку *відтинаючу площину* і називається *правильним відтином*. Приєднання нових правильних відтинів до початкової допоміжної *ЗЛП* здійснюється доти, поки на деякому кроці не буде одержаний цілочисельний розв'язок допоміжної задачі, який, очевидно, буде оптимальним розв'язком вихідної *ПЦЗЛП*.

Перший метод Гоморі належить до розглянутого класу і полягає ось у чому. Розв'язується *ПЦЗЛП*

$$L(\mathbf{x}) = \sum_{j=1}^{n} c_j x_j \to \min,$$

$$\sum_{j=1}^{n} a_{ij} x_j = b_i,\ i = 1,\dots,m, \tag{11}$$

$$x_j \ge 0,\ j = 1,\dots,n,\quad x_j - \text{ціле},\ j = 1,\dots,n.$$

Розглядаємо допоміжну *ЗЛП*

$$L(\mathbf{x}) = \sum_{j=1}^{n} c_j x_j \to \min,$$

$$\sum_{j=1}^{n} a_{ij} x_j = b_i,\ i = 1,\dots,m, \tag{12}$$

$$x_j \ge 0,\ j = 1,\dots,n,$$

що одержується з (11) відкиданням умови цілочисельності змінних.

Нехай допоміжна *ЗЛП* (12) розв'язується симплекс-методом і на останній ітерації непрямі обмеження цієї задачі набули вигляду:

$$x_i + \sum_{j=m+1}^{n} \alpha_{ij} x_j = \beta_i,\ i = 1,\dots,m, \tag{13}$$

і, таким чином, розв'язком допоміжної *ЗЛП* є n - вимірний вектор

$$\mathbf{x} = (\beta_1,\dots,\beta_m,0,\dots,0).$$

Нехай існує номер l такий, що β_l – дробове число (інакше цілочисельний вектор $\mathbf{x}$ є розв'язком *ЦЗЛП* (11)) і, як завжди, $[z]$ та $\{z\}$ — відповідно ціла та дробова частини числа z.

Згідно загальної ідеї методів відтинання потрібно перейти до розв'язання допоміжної *ЗЛП*, у якій поряд з обмеженнями (13) розглядається додаткове обмеження, яке реалізує правильний відтин.

Теорема 1. *Лінійне обмеження*

$$\{\beta_l\} - \sum_{j=m+1}^{n} \{\alpha_{lj}\} x_j \le 0 \tag{14}$$

є правильним відтином для ПЦЗЛП (11).

Доведення. Покажемо спочатку, що обмеження (14) є відтином, тобто, що оптимальний нецілочисельний розв'язок $\mathbf{x} = (\beta_1,\dots,\beta_m,0,\dots,0)$ це обмеження не задовольняє. Дійсно, оскільки $\{\beta_l\} > 0$, то, підставивши координати вектора $\mathbf{x}$ у ліву частину (14), маємо

$$\{\beta_l\} - \sum_{j=m+1}^{n} \{\alpha_{lj}\} x_j = \{\beta_l\} > 0,$$

тобто точка $\mathbf{x}$ обмеження (14) не задовольняє.

Доведемо тепер, що (14) є правильним відтином, іншими словами, що будь-який допустимий розв'язок *ПЦЗЛП* (11) задовольняє обмеження (14). Дійсно, з (13) маємо

$$x_i + \sum_{j=m+1}^{n} [\alpha_{ij}] x_j - [\beta_i] = \{\beta_i\} - \sum_{j=m+1}^{n} \{\alpha_{ij}\} x_j.$$

Нехай $\mathbf{x} = (x_1,\dots, x_n)$ — цілочисельний допустимий розв'язок *ПЦЗЛП* (11). Тоді ліва частина останньої рівності є цілочисельною, тобто, цілочисельною є і величина

$$\{\beta_i\} - \sum_{j=m+1}^{n}\{\alpha_{ij}\}x_j.$$

Решту доведення проведемо від супротивного: нехай цілочисельний розв'язок задовольняє нерівність

$$\{\beta_l\} - \sum_{j=m+1}^{n}\{\alpha_{lj}\}x_j > 0, \qquad (15)$$

яка є протилежною (14). Оскільки $0 < \{\beta_l\} < 1, 0 \le \{\alpha_{lj}\} < 1, x_j \ge 0$, то

$$\{\beta_l\} - \sum_{j=m+1}^{n}\{\alpha_{lj}\}x_j < 1,$$

що разом з (15) суперечить цілочисельності $\{\beta_l\} - \sum_{j=m+1}^{n}\{\alpha_{lj}\}x_j$ і завершує доведення теореми.

Отже, якщо до обмежень *ЗЛП* (13) або, що рівносильно, до непрямих обмежень *ЗЛП* (12), додати обмеження (14), яке можна записати в еквівалентному вигляді

$$x_{n+1} - \sum_{j=m+1}^{n}\{\alpha_{lj}\}x_j = \{\beta_l\}, x_{n+1} \ge 0, \qquad (16)$$

де x_{n+1} — додаткова змінна, і розв'язати *ЗЛП* (12), (16), то одержимо розв'язок, відмінний від $(\beta_1,...,\beta_m,0,...,0)$. Новий розв'язок також може бути нецілочисельним, що приведе до необхідності додавання нового обмеження виду (16) і т. д. Зауважимо, що якщо l в обмеженні (16) є індекс першої нецілочисельної змінної, то можна гарантувати скінченність алгоритму першого методу Гоморі, що наводиться нижче.

Алгоритм першого методу Гоморі

1. Розв'язуємо допоміжну *ЗЛП* (12). Нехай $\mathbf{x}(0)$ — її оптимальний розв'язок. Якщо оптимальний розв'язок не існує, то вихідна *ПЦЗЛП* (11) також не має оптимального розв'язку.

2. Нехай на s-й ітерації розв'язана допоміжна *ЗЛП*, що має M обмежень та N змінних, $\mathbf{x}(s)$ — її оптимальний розв'язок. Будемо вважати, що канонічні обмеження останньої ітерації, що визначають $\mathbf{x}(s)$, мають вигляд:

$$x_i + \sum_{j=M+1}^{N}\alpha_{ij}x_j = \beta_i, i = 1,...,M.$$

Звідси N-вимірний вектор $\mathbf{x}(s) = (\beta_1,...,\beta_M,0,...,0)$.

3. Якщо $\beta_i, i=1,...,M,$ — цілі, то кінець: $\mathbf{x}(s)$ є оптимальним розв'язком задачі (11). Якщо існує хоча б одне i таке, що β_i — дріб, то переходимо до пункту 4.

4. Знаходимо $l = \min\{i\}$, де мінімум обчислюється по всіх i таких, що β_i — дріб, і будуємо додаткове обмеження

$$x_{N+1} - \sum_{j=M+1}^{N}\{\alpha_{lj}\}x_j = -\{\beta_l\}, x_{N+1} \ge 0,$$

де x_{N+1} — додаткова змінна.

5. Розширюємо симплекс-таблицю за рахунок $(M+1)$ - го рядка (додаткове обмеження) та $(N+1)$ - го стовпця, що відповідає додатковій змінній x_{N+1}.

6. Розв'язуємо розширену таким чином *ЗЛП* двоїстим симплекс-методом і переходимо до пункту 2, заміняючи s на $s+1$. Якщо при цьому на якій-небудь ітерації двоїстого симплекс-методу одна з додаткових змінних задачі (тобто, тих, що з'явилися при побудові правильних відтинів) повторно стає базисною, то виключаються з подальшого розгляду відповідні їй рядок та стовпець.

Приклад 1. *Розв'язати ЦЗЛП*

$$L(\mathbf{x}) = -2x_1 + x_2 \to \min,$$
$$-x_1 + x_2 \le 3,$$
$$6x_1 + 7x_2 \le 8,$$
$$2x_1 - 3x_2 \le 6,$$
$$x_j \ge 0, x_j - \text{ціле}, j=1,2.$$

Уводячи додаткові змінні $x_3, x_4, x_5 \ge 0,$, зводимо відповідну *ЗЛП* до канонічної *ЗЛП* (КЗЛП) і розв'язуємо її звичайним симплекс-методом (див. таблицю 1).

Таблиця 1

$x_{баз}$	x_1	x_2	x_3	x_4	x_5	β
x_3	*–1*	*1*	*1*	*0*	*0*	*3*
x_4	***6***	*7*	*0*	*1*	*0*	*8*
x_5	*2*	*–3*	*0*	*0*	*1*	*6*
Δ	*–2*	*1*	*0*	*0*	*0*	*0*
x_3	*0*	*13/6*	*1*	*1/6*	*0*	*26/6*
x_1	*1*	*7/6*	*0*	*1/6*	*0*	*8/6*
x_5	*0*	*–32/6*	*0*	*–2/6*	*1*	*20/6*
Δ	*0*	*20/6*	*0*	*2/6*	*0*	*16/6*

Як бачимо, її оптимальний розв'язок $\mathbf{x} = \left(\frac{8}{6}, 0, \frac{26}{6}, 0, \frac{20}{6}\right)$ не є цілочисельним. Тому згідно п. 2 алгоритму за другим непрямим обмеженням допоміжної *ЗЛП*

$$x_1 + \left(\frac{7}{6}\right)x_2 + \left(\frac{1}{6}\right)x_4 = \frac{8}{6},$$

будуємо правильний відтин, що визначається співвідношеннями (16):

$$-\left(\frac{1}{6}\right)x_2 - \left(\frac{1}{6}\right)x_4 + x_6 = -\frac{2}{6}, x_6 \ge 0,$$

де x_6— додаткова змінна. Розширюємо останню симплекс-таблицю на один рядок, що відповідає додатковому обмеженню та на один стовпець, що відповідає додатковій змінній. Одержану *ЗЛП* розв'язуємо двоїстим симплекс-методом (див. таблицю 2).

Таблиця 2

$x_{баз}$	x_1	x_2	x_3	x_4	x_5	x_6	β
x_3	*0*	*13/6*	*1*	*1/6*	*0*	*0*	*26/6*
x_1	*1*	*7/6*	*0*	*1/6*	*0*	*0*	*8/6*
x_5	*0*	*—32/6*	*0*	*–2/6*	*1*	*0*	*20/6*
x_6	*0*	*–1/6*	*0*	***–1/6***	*0*	*1*	*–2/6*
Δ	*0*	*20/6*	*0*	*2/6*	*0*	*0*	*16/6*
x_3	*0*	*2*	*1*	*0*	*0*	*1*	*4*
x_1	*1*	*1*	*0*	*0*	*0*	*1*	*1*
x_5	*0*	*–5*	*0*	*0*	*1*	*–2*	*4*
x_4	*0*	*1*	*0*	*1*	*0*	*–6*	*2*
Δ	*0*	*3*	*0*	*0*	*0*	*2*	*2*

На першому ж кроці двоїстого симплекс-методу одержуємо оптимальний цілочисельний розв'язок допоміжної *ЗЛП*. Отже, $\mathbf{x}^* = (1,0)$ є оптимальним розв'язком вихідної *ПЦЗЛП*. При цьому $L(\mathbf{x}^*) = -2$.

§ 3. Частково цілочисельні задачі лінійного програмування.

Другий метод Гоморі

Другий метод Гоморі призначається для розв'язування частково (зокрема, повністю) цілочисельних *ЗЛП* (*ЧЦЗЛП*):

$$L(\boldsymbol{x}) = \sum_{j=1}^{n} c_j x_j \to \min, \tag{17}$$

$$\sum_{j=1}^{n} a_{ij} x_j = b_i,\ i = 1,\dots,m, \tag{18}$$

$$x_j \ge 0,\ j = 1,\dots,n, \tag{19}$$

$$x_j - \text{ціле},\ j = 1,\dots,p\ \ (p \le n). \tag{20}$$

Метод розв'язування задачі (17)–(20) грунтується на тій же ідеї, що і метод розв'язування *ПЦЗЛП*. А саме: розв'язується допоміжна *ЗЛП* (17)–(19), що одержується з вихідної відкиданням умови цілочисельності (20). Якщо ця задача розв'язку не має, то, очевидно, і вихідна *ЧЦЗЛП* не має розв'язку. Якщо ж *ЗЛП* (17)–(19) має розв'язок, то він аналізується на допустимість для задачі (17)–(20). Якщо знайдений оптимальний розв'язок є цілочисельним (у розумінні умов (20)), то він одночасно є оптимальним і для задачі (17)–(20). Інакше від розв'язаної *ЗЛП* переходять до нової допоміжної ЗЛП додаванням лінійного обмеження, яке задовольняють цілочисельні розв'язки вихідної *ЧЦЗЛП*, але не задовольняє одержаний нецілочисельний розв'язок вихідної *ЗЛП*. Це додаткове обмеження визначає деяку відтинаючу площину і називається правильним відтином. Додавання нових правильних відтинів відбувається доти, поки на деякому кроці не буде одержано цілочисельний розв'язок допоміжної задачі, що є, очевидно, оптимальним розв'язком *ЧЦЗЛП*.

Нехай на останній ітерації симплекс-методу при розв'язуванні допоміжної *ЗЛП* її непрямі обмеження набули вигляду:

$$x_i + \sum_{j=m+1}^{n} \alpha_{ij} x_j = \beta_i,\ i = 1,\dots,m, \tag{21}$$

і, отже, її розв'язком є n-вимірний вектор $\mathbf{x} = (\beta_1,\dots,\beta_m,0,\dots,0)$.

Нехай існує номер $r\,(r \le p)$ такий, що β_r — дробове число. Тоді правильний відтин у другому методі Гоморі будується згідно наступної теореми, яку ми наводимо без доведення.

Теорема 2. *Лінійне обмеження*

$$\sum_{j=m+1}^{n} \gamma_{rj} x_j \ge \{\beta_r\},$$

або, що рівносильно,

$$x_{n+1} - \sum_{j=m+1}^{n} \gamma_{rj} x_j = -\{\beta_r\},\ x_{n+1} \ge 0, \tag{22}$$

де

$$\gamma_{rj} = \begin{cases} \{\alpha_{rj}\}, & \text{якщо} \quad j \le p,\ \{\alpha_{rj}\} \le \{\beta_r\}, \\ \{\beta_r\}(1-\{\alpha_{rj}\})/(1-\{\beta_r\}), & \text{якщо} \quad j \le p,\ \{\alpha_{rj}\} > \{\beta_r\}, \\ \alpha_{rj}, & \text{якщо} \quad j > p, \alpha_{rj} \ge 0, \\ \{\beta_r\}(-\alpha_{rj})/(1-\{\beta_r\}), & \text{якщо} \quad j > p, \alpha_{rj} < 0, \end{cases} \tag{23}$$

а x_{n+1} — *додаткова змінна, є правильним відтином для ЧЦЗЛП* (17)–(20).

Зауважимо, що якщо r в обмеженні (22) є індекс першої нецілочисельної змінної серед перших p змінних, то алгоритм другого методу Гоморі, що формулюється нижче, є скінченним.

Алгоритм другого методу Гоморі

1. Розв'язуємо допоміжну *ЗЛП* (17)–(19). Нехай $\mathbf{x}(0)$ — її оптимальний розв'язок. Якщо оптимального розв'язку не існує, то вихідна *ЧЦЗЛП* (17)–(20) також не має розв'язку.

2. Нехай на s-й ітерації розв'язана допоміжна *ЗЛП*, що має M обмежень та N змінних, $\mathbf{x}(s)$ — її оптимальний розв'язок. Будемо вважати, що канонічні обмеження останньої ітерації, що визначають $\mathbf{x}(s)$, мають вигляд:

$$x_i + \sum_{j=M+1}^{N} \alpha_{ij} x_j = \beta_i,\ i = 1,\dots,M.$$

Звідси N - вимірний вектор $\mathbf{x}(s) = (\beta_1,\dots,\beta_M,0,\dots,0)$.

3. Якщо $\beta_i, i=1,...,p,$ — цілі, то кінець: $\mathbf{x}(s)$ є оптимальним розв'язком вихідної *ЧЦЗЛП*. Якщо існує хоча б одне i таке, що β_i — дріб $(i=1,...,p)$, то переходимо до пункту 4.

4. Знаходимо $r=\min\{i\}$, де мінімум береться по всіх i $(i=1,...,p)$ таких, що β_i — дріб, і будуємо додаткове обмеження за формулами (22), (23) при $m=M$, $n=N$.

5. Розширюємо симплекс-таблицю за рахунок $(M+1)$ - го рядка (додаткове обмеження) та $(N+1)$ - го стовпця, що відповідає додатковій змінній x_{N+1}.

6. Розв'язуємо розширену таким чином *ЗЛП* двоїстим симплекс-методом і переходимо до пункту 2, заміняючи s на $s+1$. Якщо при цьому на деякій ітерації двоїстого симплекс-методу одна з додаткових змінних повторно стає базисною, то з подальшого розгляду виключаються відповідні їй рядок та стовпець.

Приклад 2. *Розв'язати ЦЗЛП*

$$L(\mathbf{x})=-x_1-8x_2\to\min,$$
$$3x_1+x_2\le 9,$$
$$0.16x_1+x_2\le 1.9,$$
$$x_j\ge 0, x_j - \text{ціле}, j=1,2.$$

Вводячи додаткові змінні x_3 та x_4, зводимо відповідну *ЗЛП* до канонічної форми і розв'язуємо її симплекс-методом (див. таблицю 3). Таблиця 3 містить

Таблиця 3

$x_{баз}$	x_1	x_2	x_3	x_4	β
x_3	*3.00*	*1.00*	*1.00*	*0.00*	*9.00*
x_4	*0.16*	***1.00***	*0.00*	*1.00*	*1.90*
Δ	*–1.00*	*–8.00*	*0.00*	*0.00*	*0.00*
x_3	*2.84*	*0.00*	*1.00*	*–1.00*	*7.10*
x_2	*0.16*	*1.00*	*0.00*	*1.00*	*1.90*
Δ	*0.28*	*0.00*	*0.00*	*8.00*	*15.20*

оптимальний розв'язок вихідної *ЗЛП*: $\mathbf{x}(0)=(0,1.9,7.1,0)$.

Оскільки умову цілочисельності цей розв'язок не задовольняє, то будуємо правильний відтин за другим непрямим обмеженням *ЗЛП* згідно співвідношень (22), (23). Обмеження

$$0.16x_1+x_4\ge 0.9,$$

або

$$-0.16x_1-x_4+x_5=-0.9, x_5\ge 0$$

є правильним відтином, де x_5 — додаткова змінна. Симплекс-таблиця розширюється за рахунок додаткового обмеження та додаткової змінної (див. таблицю 4). Таблиця 5 являє собою результати виконання двох ітерацій двоїстого симплекс-методу.

Таблиця 4

$x_{баз}$	x_1	x_2	x_3	x_4	x_5	β
x_3	*2.84*	*0.00*	*1.00*	*–1.00*	*0.00*	*7.10*
x_2	*0.16*	*1.00*	*0.00*	*1.00*	*0.00*	*1.90*
x_5	***–0.16***	*0.00*	*0.00*	*–1.00*	*1.00*	*–0.90*
Δ	*0.28*	*0.00*	*0.00*	*8.00*	*0.00*	*15.20*

Таблиця 5

$x_{баз}$	x_1	x_2	x_3	x_4	x_5	β
x_3	*0.00*	*0.00*	*1.00*	***–18.75***	*17.75*	*–8.88*
x_2	*0.00*	*1.00*	*0.00*	*0.00*	*1.00*	*1.00*
x_1	*1.00*	*0.00*	*0.00*	*6.25*	*–6.25*	*5.63*
Δ	*0.00*	*0.00*	*0.00*	*6.25*	*1.75*	*13.63*
x_4	*0.00*	*0.00*	*–0.05*	*1.00*	*–0.95*	*0.47*
x_2	*0.00*	*1.00*	*0.00*	*0.00*	*1.00*	*1.00*
x_1	*1.00*	*0.00*	*0.33*	*0.00*	*–0.33*	*2.67*
Δ	*0.00*	*0.00*	*0.33*	*0.00*	*7.67*	*10.67*

Оптимальний розв'язок $\mathbf{x}(1) = (2.67, 1.00, 0.00, 0.47, 0.00)$ розширеної *ЗЛП* умову цілочисельності не задовольняє, тому за третім непрямим обмеженням будується правильний відтин:

$$-0.33x_3 - 0.67x_5 + x_6 = -0.67, x_6 \geq 0,$$

де x_6 — додаткова змінна. Розширена симплекс-таблиця наведена у таблиці 6. Виконавши один крок двоїстого симплекс-методу, одержуємо оптимальний розв'язок допоміжної *ЗЛП* $\mathbf{x}(2) = (2.00, 1.00, 2.01, 0.58, 0.00, 0.00)$, який задовольняє умову цілочисельності. Тому $\mathbf{x}^* = (2,1)$ є оптимальним розв'язком вихідної задачі. При цьому $L(\mathbf{x}^*) = -10$.

Таблиця 6

$x_{баз}$	x_1	x_2	x_3	x_4	x_5	x_6	β
x_4	*0.00*	*0.00*	*–0.05*	*1.00*	*–0.95*	*0.00*	*0.47*
x_2	*0.00*	*1.00*	*0.00*	*0.00*	*1.00*	*0.00*	*1.00*
x_1	*1.00*	*0.00*	*0.33*	*0.00*	*–0.33*	*0.00*	*2.67*
x_6	*0.00*	*0.00*	***–0.33***	*0.00*	*–0.67*	*1.00*	*–0.67*
Δ	*0.00*	*0.00*	*0.33*	*0.00*	*7.67*	*0.00*	*10.67*
x_4	*0.00*	*0.00*	*0.00*	*1.00*	*–0.84*	*–0.16*	*0.58*
x_2	*0.00*	*1.00*	*0.00*	*0.00*	*1.00*	*0.00*	*1.00*
x_1	*1.00*	*0.00*	*0.00*	*0.00*	*–1.00*	*1.00*	*2.00*
x_3	*0.00*	*0.00*	*1.00*	*0.00*	*2.00*	*–3.00*	*2.01*
Δ	*0.00*	*0.00*	*0.00*	*0.00*	*7.00*	*1.00*	*10.00*

§ 4. Третій метод Гоморі

Розглядаємо *ПЦЗЛП*:

$$L(\boldsymbol{x}) = \sum_{j=1}^{n} c_j x_j \to \min, \quad (24)$$

$$\sum_{j=1}^{n} a_{ij} x_j = b_i,\ i = 1,...,m, \quad (25)$$

$$x_j \geq 0,\ j = 1,...,n, \quad (26)$$

$$x_j - \text{ціле},\ j = 1,...,n. \quad (27)$$

Нехай непрямі обмеження *ЗЛП* (24)–(26), наприклад, у базисі $\mathbf{A}_1,...,\mathbf{A}_m$ приведені до майже канонічного вигляду:

$$x_i + \sum_{j=m+1}^{n} \alpha_{ij} x_j = \beta_i,\ i = 1,...,m, \quad (28)$$

де $\alpha_{ij}, \beta_i, i = 1,...,m, j = m+1,...,n,$ — цілі і симплекс-різниці $\Delta_j \geq 0, j = 1,...,n.$ З непрямими обмеженнями (28) *ЗЛП* (24)–(26) називається майже канонічною (*МКЗЛП*). Якщо $\beta_i \geq 0, i = 1,...,m,$ то обмеження визначають оптимальний розв'язок задачі (24)–(27), інакше визначається деякий цілочисельний майже допустимий базисний розв'язок (*МДБР)* вихідної *ЗЛП*. Можна було б, звичайно, вибрати один з індексів i, для якого $\beta_i < 0,$ і виконати ітерацію двоїстого симплекс-методу. Проте у цьому випадку цілочисельність параметрів нової симплекс-таблиці була б, взагалі кажучи, порушена через необхідність ділення на ведучий елемент перетворення. Цілочисельність нової таблиці гарантується лише тоді, коли ведучий елемент дорівнює *–1*.

Виявляється, що можна побудувати додаткове обмеження, якому задовольняють всі цілочисельні розв'язки задачі (24)–(27) і яке разом з тим визначає ведучий рядок перетворення, що має ведучий елемент *–1*. Будується воно за l-м обмеженням системи (28), для якого $\beta_l < 0$:

$$x_l + \sum_{j=m+1}^{n} \alpha_{lj} x_j = \beta_l. \tag{29}$$

Якщо серед обмежень (28) є декілька з від'ємною правою частиною, то l вибирається, як правило, з умови

$$\beta_l = \min_{i:\beta_i<0} \beta_i.$$

Поділимо обидві частини (29) на довільне число $\alpha > 0$ і запишемо одержаний результат у вигляді:

$$\frac{x_l}{\alpha} + \sum_{j=m+1}^{n} \left\{\frac{\alpha_{lj}}{\alpha}\right\} x_j - \left\{\frac{\beta_l}{\alpha}\right\} = \left[\frac{\beta_l}{\alpha}\right] - \sum_{j=m+1}^{n} \left[\frac{\alpha_{lj}}{\alpha}\right] x_j. \tag{30}$$

Ліва частина (30) при $x_j \geq 0,\ j = 1,...,n,$ являє собою різницю двох величин

$$\frac{x_l}{\alpha} + \sum_{j=m+1}^{n} \left\{\frac{\alpha_{lj}}{\alpha}\right\} x_j \geq 0, \quad \left\{\frac{\beta_l}{\alpha}\right\} < 1,$$

тобто строго більша *–1*. Отже, права частина, будучи цілим числом при цілочисельних x_j, задовольняє умову

$$\left[\frac{\beta_l}{\alpha}\right] - \sum_{j=m+1}^{n} \left[\frac{\alpha_{lj}}{\alpha}\right] x_j \geq 0, \tag{31}$$

яка є вірною для будь-якого допустимого розв'язку задачі (24)–(27).

Уводячи додаткову змінну x_{n+1}, перепишемо (31) у вигляді

$$x_{n+1} + \sum_{j=m+1}^{n} \left[\frac{\alpha_{lj}}{\alpha}\right] x_j = \left[\frac{\beta_l}{\alpha}\right],\ x_{n+1} \geq 0. \tag{32}$$

Очевидно, що з від'ємності α_{lj} випливає від'ємність $\lfloor \alpha_{lj}/\alpha \rfloor$ і навпаки. Тому $[\beta_l/\alpha] < 0$ і серед чисел $\lfloor \alpha_{lj}/\alpha \rfloor$ є від'ємні, тобто рядок таблиці, що визначається новим обмеженням (32), може бути прийнятий за ведучий для наступного симплексного перетворення. Разом з тим при

$$\alpha = \max_{j=m+1,...,n} (-\alpha_{lj})$$

кожному від'ємному $\alpha_{lj},\ j = m+1,...,n,$ відповідає $\lfloor \alpha_{lj}/\alpha \rfloor = -1$, тобто ведучий елемент цього перетворення явно дорівнює *–1*.

Отже, розширюємо наявну симплекс-таблицю за рахунок $(m+1)$-го рядка з елементами $\lfloor \alpha_{lj}/\alpha \rfloor$ (елементи, що відповідають базисним змінним, рівні нулю) та одиничного стовпця $\mathbf{A}_{n+1}$, що відповідає додатковій змінній x_{n+1}. Потім виконується симплекс-перетворення з $(m+1)$-м ведучим рядком і ведучим стовпцем, що вибирається за правилами двоїстого симплекс-методу. Тоді нова симплекс-таблиця буде повністю цілочисельною. Описана послідовність дій складає окрему ітерацію алгоритму третього методу Гоморі. Ітерації виконуються доти, поки не буде отримана симплекс-таблиця, в якій усі праві частини невід'ємні, або є рядок з від'ємною правою частиною і невід'ємними рештою елементів. У першому випадку *ПЦЗЛП* розв'язана, у другому — її обмеження є суперечливими. Якщо на будь-якій ітерації одна з додаткових змінних переходить з небазисних у базисні, то відповідні їй рядок і стовпець симплекс-таблиці викреслюються з подальшого розгляду.

Алгоритм третього методу Гоморі

1. Зводимо *ЗЛП* (24)–(26) до *МКЗЛП* з цілочисельними коефіцієнтами, що визначає цілочисельний *МДБР* $\mathbf{x}(0)$, для якого $\Delta_j \geq 0, j = 1,...,n.$

2. Нехай на s-й ітерації одержана повністю цілочисельна *МКЗЛП* з непрямими обмеженнями виду

$$x_i + \sum_{j=M+1}^{N} \alpha_{ij} x_j = \beta_i, \; i = 1,...,M,$$

що визначає цілочисельний N-вимірний *МДБР* $\mathbf{x}(s) = (\beta_1,...,\beta_M,0,...,0),$ для якого $\Delta_j \geq 0, j = 1,...,N.$

3. Якщо $\beta_i \geq 0, i = 1,...,M,$ то кінець: $\mathbf{x}(s)$ є оптимальним розв'язком *ПЦЗЛП*.

4. Якщо для деякого i, такого, що $\beta_i < 0, \alpha_{ij} \geq 0, j = 1,...,N,$ то кінець: *ПЦЗЛП* (24)–(27) не має допустимих розв'язків. Якщо таких i немає, то

5. Знаходимо індекс l з умови: $\beta_l = \min \beta_i$, де мінімум визначається на множині тільки тих i, для яких $\beta_i < 0$. Знаходимо $\alpha = \max(-\alpha_{lj})$, де максимум визначається на множині $j = M+1,...,N$, і будуємо додаткове обмеження (32) при $m = M$ та $n = N$.

6. Розширюємо симплекс-таблицю за рахунок $(M+1)$-го рядка (додаткове обмеження) та $(N+1)$ -го стовпця, що відповідає додатковій змінній x_{N+1}.

7. Знаходимо індекс k з умови: $\Delta_k = \min \Delta_j$, де мінімум визначається на множині тільки тих j, для яких $\alpha_{M+1,j} < 0.$ Переходимо до нового цілочисельного *МДБР* $\mathbf{x}(s+1)$, виконуючи симплекс-перетворення з ведучими рядком $(M+1)$ і стовпцем k. Якщо k — індекс однієї з додаткових змінних, то переходимо до пункту 8, інакше — до пункту 3, заміняючи s на $s+1$, M на $M+1$, N на $N+1$.

8. Виключаємо з подальшого розгляду (викреслюємо) k-й стовпець та $(M+1)$-й (останній) рядок симплекс-таблиці, перенумеровуємо решту додаткових змінних для збереження неперервної нумерації всіх змінних задачі і переходимо до пункту 3, заміняючи s на $s+1$.

Звертаючи увагу на пункт 1 алгоритму, зауважимо, що інколи побудова цілочисельної симплекс-таблиці з невід'ємними значеннями симплекс-різниць не вимагає обчислень. У загальному ж випадку цей етап зводиться до застосування цього ж методу до деякої допоміжної задачі, причому на одній з його ітерацій може виявитися неможливість побудови потрібної вихідної таблиці. Враховуючи це, навряд чи варто застосовувати цей метод до розв'язування задач, у яких зміст вихідної таблиці не є очевидним.

Приклад 3. *Розв'язати ЦЗЛП*

$$L(\mathbf{x}) = 6x_1 + 4x_2 \to \min,$$

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

$$x_1 - 2x_2 \le 2,$$
$$3x_1 + 2x_2 \ge 1,$$
$$x_j \ge 0,\ x_j - \text{ціле},\ j=1,2.$$

Ця задача введенням додаткових змінних x_3, x_4, x_5 легко зводиться до майже канонічного виду (див. таблицю 7), що визначає *МДБР* $\mathbf{x}(0) = (0,0,-3,2,-1)$.

Таблиця 7

$x_{баз}$	x_1	x_2	x_3	x_4	x_5	β
x_3	–2	–1	1	0	0	–3
x_4	1	–2	0	1	0	2
x_5	–3	–2	0	0	1	–1
Δ	6	4	0	0	0	0

Оскільки серед компонент вектора $\mathbf{x}(0)$ є від'ємні, то за першим обмеженням ($l=1$) будуємо правильний відтин при $\alpha = 2$:

$$[-3/2] - [-2/2]x_1 - [-1/2]x_2 \ge 0,$$

або $\quad -2 + x_1 + x_2 \ge 0.$

Уводимо додаткову змінну $x_6 \ge 0$ і записуємо останню нерівність у вигляді:

$$-x_1 - x_2 + x_6 = -2,\ x_6 \ge 0.$$

Таблиця 8

$x_{баз}$	x_1	x_2	x_3	x_4	x_5	x_6	β
x_3	–2	–1	1	0	0	0	–3
x_4	1	–2	0	1	0	0	2
x_5	–3	–2	0	0	1	0	–1
x_6	–1	**–1**	0	0	0	1	–2
Δ	6	4	0	0	0	0	0

` *Таблиця 9*

$x_{баз}$	x_1	x_2	x_3	x_4	x_5	x_6	β
x_3	–1	0	1	0	0	–1	–1
x_4	3	0	0	1	0	–2	6
x_5	–1	0	0	0	1	–2	3
x_2	1	1	0	0	0	–1	2
Δ	2	0	0	0	0	4	–8

Розширюємо симплекс-таблицю на один рядок (додаткове обмеження) і один стовпець (додаткова змінна x_6) (див. таблицю 8) і згідно двоїстого симплекс-методу виконуємо симплекс-перетворення з ведучими четвертим рядком та другим стовпцем (див. таблицю 9).

Новий *МДБР* $\mathbf{x}(1)=(0,2,-1,6,3,0)$ не є оптимальним. Тому виконується ще одна ітерація методу Гоморі-3 (див. таблицю 10, де наведені розширена задача та наступна ітерація).

Таблиця 10

$x_{баз}$	x_1	x_2	x_3	x_4	x_5	x_6	x_7	β
x_3	*–1*	*0*	*1*	*0*	*0*	*–1*	*0*	*–1*
x_4	*3*	*0*	*0*	*1*	*0*	*–2*	*0*	*6*
x_5	*–1*	*0*	*0*	*0*	*1*	*–2*	*0*	*3*
x_2	*1*	*1*	*0*	*0*	*0*	*–1*	*0*	*2*
x_7	***–1***	*0*	*0*	*0*	*0*	*–1*	*1*	*–1*
Δ	*2*	*0*	*0*	*0*	*0*	*4*	*0*	*–8*
x_3	*0*	*0*	*1*	*0*	*0*	*0*	*–1*	*0*
x_4	*0*	*0*	*0*	*1*	*0*	*–5*	*3*	*3*
x_5	*0*	*0*	*0*	*0*	*1*	*–1*	*–1*	*4*
x_2	*0*	*1*	*0*	*0*	*0*	*–2*	*1*	*1*
x_1	*1*	*0*	*0*	*0*	*0*	*1*	*–1*	*1*
Δ	*0*	*0*	*0*	*0*	*0*	*2*	*2*	*-10*

Як бачимо, остання симплекс-таблиця визначає оптимальний розв'язок задачі $\mathbf{x}^*=(1,1)$. При цьому $L(\mathbf{x}^*)=10$. Зауважимо, що побудова останнього додаткового обмеження була зайвою, якщо прийняти до уваги, що при виконанні симплекс-перетворення у таблиці 9 у відповідності з двоїстим симплекс-методом ведучий елемент був би рівним –*1*.

§ 5. Дискретні задачі лінійного програмування. Метод Дальтона-Ллевеліна

Другий алгоритм Гоморі може бути видозмінений для розв'язування частково (зокрема, повністю) дискретних *ЗЛП:*

$$L(\mathbf{x})=\sum_{j=1}^{n}c_j x_j \to \min, \tag{33}$$

$$\sum_{j=1}^{n}a_{ij}x_j=b_i,\ i=1,...,m, \tag{34}$$

$$x_j\ge 0,\ j=1,...,n, \tag{35}$$

$$x_j \in \{x_j^1, x_j^2, ..., x_j^{n_j}\}, \; j = 1, ..., k \;\; (k \le n). \quad (36)$$

Нехай $0 = x_j^1 < x_j^2 < .. < x_j^{n_j}$.

До умов (34), якщо це необхідно, приєднані нерівності $x_j \le x_j^{n_j}, \; j = 1, ..., p,$ так що довільний план $\mathbf{x}$ *ЗЛП* (33)-(35) явно задовольняє умови $0 \le x_j \le x_j^{n_j}, \; j = 1, ..., p.$

Слідуючи ідеї методів відтинання, розв'язується допоміжна *ЗЛП* (33)–(35). Нехай на останній ітерації симплекс-методу її непрямі обмеження набирають вигляду

$$x_i + \sum_{j=m+1}^{n} \alpha_{ij} x_j = \beta_i, \; i = 1, ..., m, \quad (37)$$

тобто її розв'язком є n-вимірний вектор $\mathbf{x} = (\beta_1, ..., \beta_m, 0, ..., 0)$, де для деякого номера $r \; (r \le p)$

$$x_r^{\nu} < \beta_r < x_r^{\nu+1}, \text{де } 1 \le \nu < n_r. \quad (38)$$

Тоді правильний відтин у методі Дальтона-Ллевеліна будується згідно наступної теореми.

Теорема 3. *Лінійне обмеження*

$$\sum_{j=m+1}^{n} \gamma_{rj} x_j \ge \gamma_r, \quad (39)$$

або, що рівносильно,

$$x_{n+1} - \sum_{j=m+1}^{n} \gamma_j x_j = -\gamma_r, \; x_{n+1} \ge 0, \quad (40)$$

де

$$\gamma_r = \beta_r - x_r^{\nu}, \quad (41)$$

$$\gamma_{rj} = \begin{cases} \alpha_{rj}, & \text{якщо} \quad \alpha_{rj} \ge 0, \\ \dfrac{\beta_r - x_r^{\nu}}{x_r^{\nu+1} - \beta_r}(-\alpha_{rj}), & \text{якщо} \quad \alpha_{rj} < 0, \end{cases} \quad (42)$$

а x_{n+1} *— додаткова змінна, є правильним відтином для дискретної ЗЛП* (33)–(36).

<u>Доведення</u>. Позначимо через NB множину індексів небазисних змінних, тобто $NB = \{m+1, ..., n\}$. Покажемо, що обмеження (39) є відтином.

Дійсно, з врахуванням (38) маємо

$$\sum_{j=m+1}^{n} \gamma_{rj} x_j = \sum_{j \in NB} \gamma_{rj} x_j = 0 < \gamma_r = \beta_r - x_r^{\nu},$$

тобто, точка $\mathbf{x}$ обмеження (39) не задовольняє.

Доведемо тепер, що (39) є правильним відтином. Будь-який допустимий розв'язок задачі (33)–(36) задовольняє одну з двох нерівностей:

$$1) x_r \ge x_r^{\nu+1}, \quad (43)$$

$$2) x_r \le x_r^{\nu}. \quad (44)$$

Запишемо r-е обмеження системи (37) у вигляді

$$x_r = \beta_r + \sum_{j \in NB} (-\alpha_{rj}) x_j. \quad (45)$$

Увівши позначення

$$NB^{+} = \{j : j \in NB, \alpha_{rj} < 0\},$$

$$NB^{-} = \{j : j \in NB, \alpha_{rj} \geq 0\},$$

$$S^{+} = \sum_{j \in NB^{+}} (-\alpha_{rj}) x_j,$$

$$S^{-} = \sum_{j \in NB^{-}} (-\alpha_{rj}) x_j,$$

перепишемо співвідношення (45) у вигляді:

$$x_r = \beta_r + S^{+} + S^{-}. \tag{46}$$

За означенням NB^{+} і NB^{-} та з невід'ємності x_j, маємо

$$S^{+} \geq 0, \tag{47}$$

$$S^{-} \leq 0. \tag{48}$$

Розглянемо перший випадок. Приймаючи до уваги (43) та (46), послідовно маємо:

$$\beta_r + S^{+} + S^{-} \geq x_r^{\nu+1},$$

$$S^{+} \geq (x_r^{\nu+1} - \beta_r) - S^{-},$$

звідки в силу (48) одержуємо

$$S^{+} \geq x_r^{\nu+1} - \beta_r,$$

або, що рівносильно,

$$\frac{\beta_r - x_r^{\nu}}{x_r^{\nu+1} - \beta_r} S^{+} \geq \beta_r - x_r^{\nu}.$$

Об'єднуючи цю нерівність з (48), маємо:

$$-S^{-} + \frac{\beta_r - x_r^{\nu}}{x_r^{\nu+1} - \beta_r} S^{+} \geq \beta_r - x_r^{\nu}. \tag{49}$$

Для випадку (44) з урахуванням (46) маємо:

$$\beta_r + S^{+} + S^{-} \leq x_r^{\nu},$$

$$-S^{-} \geq (\beta_r - x_r^{\nu}) + S^{+},$$

або, приймаючи до уваги (47),

$$-S^{-} \geq \beta_r - x_r^{\nu}.$$

Об'єднуючи останню нерівність з очевидною нерівністю

$$\frac{\beta_r - x_r^{\nu}}{x_r^{\nu+1} - \beta_r} S^{+} \geq 0,$$

одержуємо співвідношення

$$-S^{-} + \frac{\beta_r - x_r^{\nu}}{x_r^{\nu+1} - \beta_r} S^{+} \geq \beta_r - x_r^{\nu},$$

яке співпадає з (49). Отже, будь-який допустимий розв'язок задачі (33)–(36) задовольняє нерівність (49), яку, враховуючи уведені позначення, переписуємо у вигляді

$$\sum_{j \in NB^{+}} \frac{\beta_r - x_r^{\nu}}{x_r^{\nu+1} - \beta_r} (-\alpha_{rj}) x_j + \sum_{j \in NB^{-}} \alpha_{rj} x_j \geq \beta_r - x_r^{\nu}.$$

Легко бачити, що одержана нерівність співпадає з нерівністю (39), якщо увести позначення (41), (42). Отже, нерівність (39) є правильним відтином. Доведення завершено.

Алгоритм Дальтона–Ллевеліна

1. Розв'язуємо допоміжну *ЗЛП* (33)–(35). Нехай $\mathbf{x}(0)$ — її оптимальний розв'язок. Якщо *ЗЛП* (33)–(35) розв'язку не має, то *ЧДЗЛП* (33)–(36) також не має розв'язку.

Нехай на s-й ітерації розв'язана допоміжна *ЗЛП*, що містить M обмежень та N змінних, $\mathbf{x}(s)$ — її оптимальний розв'язок. Нехай канонічні обмеження останньої ітерації, що визначають $\mathbf{x}(s)$, мають вигляд:

$$x_i + \sum_{j=M+1}^{N} \alpha_{ij} x_j = \beta_i,\ i = 1,...,M,$$

тобто $\mathbf{x} = (\beta_1,...,\beta_M,0,...,0)$.

2. Якщо $\beta_i, i = 1,...,p,$ задовольняють умови (36), то кінець: $\mathbf{x}(s)$ є оптимальним розв'язком вихідної *ЧДЗЛП*. Інакше

3. Знаходимо $r = \min\{i\}$, де мінімум береться по всіх $i\ (i = 1,...,p)$ таких, що для певного $\nu, 1 \le \nu \le n_i - 1,\ x_i^{\nu} < \beta_i < x_i^{\nu+1}$, і будуємо додаткове обмеження за формулами (40), (41), (42) при $m = M, n + N$.

4. Розширюємо симплекс-таблицю за рахунок $(M+1)$-го рядка (додаткове обмеження) та $(N+1)$-го стовпця, що відповідає додатковій змінній x_{N+1}.

5. Розв'язуємо розширену таким чином *ЗЛП* двоїстим симплекс-методом і переходимо до пункту 2, заміняючи s на $s+1$. Якщо при цьому на деякій ітерації двоїстого симплекс-методу одна з додаткових змінних повторно стає базисною, то з подальшого розгляду виключаються відповідні їй рядок та стовпець.

Зауважимо, що, очевидно, розглянутий алгоритм можна застосовувати також і для розв'язування повністю та частково цілочисельних *ЗЛП*, але, мабуть, з меншою ефективністю, ніж відповідні алгоритми Гоморі.

Приклад 4. *Розв'язати ПДЗЛП*

$$\begin{aligned}
&L(\mathbf{x}) = x_1 + x_2 \to \max,\\
&\quad -x_1 + 4x_2 \le 12,\\
&\quad 4x_1 - x_2 \le 12,\\
&\quad x_1 \ge 0, x_2 \ge 0,\\
&\quad x_1 \in \{0,1,4\},\\
&\quad x_2 \in \{0,1,3,5\}.
\end{aligned}$$

Відповідна *ЗЛП* введенням невід'ємних додаткових змінних x_3, x_4 зводиться до канонічного вигляду

$$\begin{aligned}
&L(\mathbf{x}) = -x_1 - x_2 \to \min,\\
&\quad -x_1 + 4x_2 + x_3 \quad\; = 12,\\
&\quad 4x_1 - x_2 \qquad + x_4 = 12,
\end{aligned}$$

$$x_j \geq 0,\ j = 1,...,4,$$

яка розв'язується симплекс-методом (див. таблицю 11).

Таблиця 11

$x_{баз}$	x_1	x_2	x_3	x_4	β
x_3	*–1*	*4*	*1*	*0*	*12*
x_4	***4***	*–1*	*0*	*1*	*12*
Δ	*–1*	*–1*	*0*	*0*	*0*
x_3	*0*	***15/4***	*1*	*1/4*	*15*
x_1	*1*	*–1/4*	*0*	*1/4*	*3*
Δ	*0*	*–5/4*	*0*	*1/4*	*3*
x_2	*0*	*1*	*4/15*	*1/15*	*4*
x_1	*1*	*0*	*1/15*	*4/15*	*4*
Δ	*0*	*0*	*1/3*	*1/3*	*8*

Остання симплекс-таблиця визначає оптимальний розв'язок *ЗЛП* $\mathbf{x} = (4,4)$. Оскільки змінна x_2 не задовольняє умову дискретності, то за першим непрямим обмеженням, де змінна x_2 є базисною, будуємо додаткове обмеження згідно формул (40)–(42):

$$-(4/15)x_3 - (1/15)x_4 + x_5 = -1,\ x_5 \geq 0.$$

Згідно алгоритму розширюємо симплекс-таблицю за рахунок правильного відтину і розв'язуємо одержану *ЗЛП* двоїстим симплекс-методом (див. таблицю 12).

Таблиця 12

$x_{баз}$	x_1	x_2	x_3	x_4	x_5	β
x_2	*0*	*1*	*4/15*	*1/15*	*0*	*4*
x_1	*1*	*0*	*1/15*	*4/15*	*0*	*4*
x_5	*0*	*0*	***–4/15***	*–1/15*	*1*	*–1*
Δ	*0*	*0*	*1/3*	*1/3*	*0*	*8*
x_2	*0*	*1*	*0*	*0*	*1*	*3*
x_1	*1*	*0*	*0*	*1/4*	*1/4*	*15/4*
x_3	*0*	*0*	*1*	*1/4*	*–15/4*	*15/4*
Δ	*0*	*0*	*0*	*1/4*	*5/4*	*27/4*

В оптимальному розв'язку цієї задачі $\mathbf{x} = (15/4, 3, 15/4, 0, 0)$ змінна x_1 умову дискретності не задовольняє, тому за другим непрямим обмеженням, де x_1 є базисною, будуємо правильний відтин згідно формул (40)–(42):

$-(1/4)x_4-(1/4)x_5+x_6=-11/4,\ x_6\geq 0.$

Розширена допоміжна *ЗЛП* розв'язується двоїстим симплекс-методом (див. таблицю 13).

Таблиця 13

$x_{баз}$	x_1	x_2	x_3	x_4	x_5	x_6	β
x_2	*0*	*1*	*0*	*0*	*1*	*0*	*3*
x_1	*1*	*0*	*0*	*1/4*	*1/4*	*0*	*15/4*
x_3	*0*	*0*	*1*	*1/4*	*–15/4*	*0*	*15/4*
x_6	*0*	*0*	*0*	***–1/4***	*–1/4*	*1*	*–11/4*
Δ	*0*	*0*	*0*	*1/4*	*5/4*	*0*	*27/4*
x_2	*0*	*1*	*0*	*0*	*1*	*0*	*3*
x_1	*1*	*0*	*0*	*0*	*0*	*1*	*1*
x_3	*0*	*0*	*1*	*0*	*–4*	*1*	*1*
x_4	*0*	*0*	*0*	*1*	*1*	*–4*	*11*
Δ	*0*	*0*	*0*	*0*	*1*	*1*	*4*

Її оптимальний розв'язок $\mathbf{x}=(1,3,1,11,0,0)$ умову дискретності задовольняє. Отже, $\mathbf{x}^*=(1,3)$ є оптимальним розв'язком вихідної задачі. При цьому $L(\mathbf{x}^*)=4$.

§ 6. Метод віток та границь. Алгоритм Ленд-Дойг

Для розв'язання задач дискретного (зокрема цілочисельного) лінійного програмування широко використовується метод віток та границь. Цей метод належить до класу комбінаторних методів і зводиться до направленого перебору варіантів розв'язків оптимізаційної задачі, коли розглядаються лише ті з них, які виявляються за певними ознаками перспективними, і відкидаються відразу цілі множини варіантів, що є безперспективними.

Розглянемо загальну схему методу на прикладі оптимізаційної задачі

$$f(\mathbf{x})\to\min,\mathbf{x}\in D, \tag{50}$$

де D — скінченна множина, $\mathbf{x}\in E^n$.

Основу методу складають такі процедури.

1.*Обчислення нижньої оцінки* (*границі*) для значень цільової функції $f(\mathbf{x})$ на допустимій множині $D=D^0$ (або на деякій її підмножині), тобто, знаходження числа $\xi(D^0)$, такого, що $f(\mathbf{x})\geq\xi(D^0)$ для всіх $\mathbf{x}\in D^0$. Питання про те, як знаходиться $\xi(D^0)$, вирішується окремо для кожної задачі.

2. ***Розбиття на підмножини*** (***розгалуження***)**.** Реалізація методу пов'язана з розгалуженням множини D (або деякої її підмножини) в дерево підмножин згідно такої схеми.

Нульовий (*початковий*) *крок.* Деяким чином (в залежності від задачі) множина D^0 розбивається на скінченне число підмножин $D^{1,1},...,D^{1,N_1}$, таких, що не перетинаються між собою, і

$$D^0 = \bigcup_{l=1}^{N_1} D^{1,l}$$

(див. рис.2).

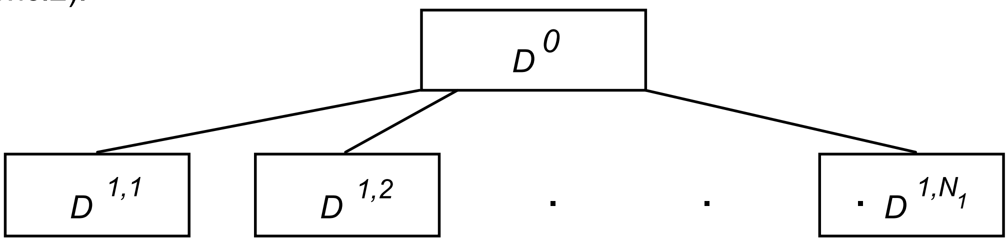

Рис.2

S-й крок (*s≥1*). Маємо множини $D^{s,1},...,D^{s,N_s}$, одержані на попередньому кроці. За певним правилом (що формулюється нижче) серед них вибирається множина $D^{s,u}$, яка вважається перспективною. Ця множина розбивається на скінченне число підмножин $D^{s,u,1},...,D^{s,u,l},...,D^{s,u,k_u}$ таких, що не перетинаються між собою, і

$$D^{s,u} = \bigcup_{l=1}^{k_u} D^{s,u,l}$$

(див. рис.3).

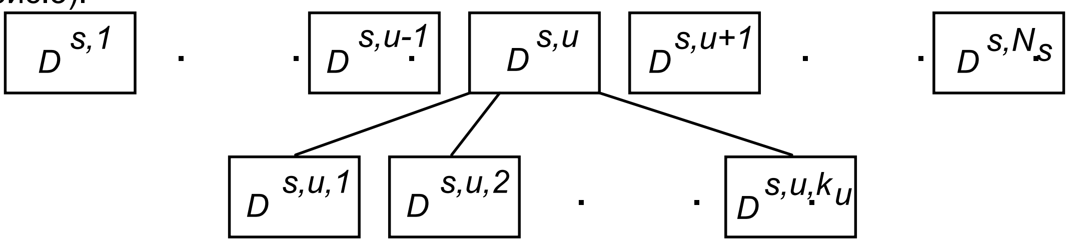

Рис. 3

Перепозначаємо множини $D^{s,1},...,D^{s,u-1},D^{s,u+1},...,D^{s,N_s}$, що не розгалужувалися, та множини $D^{s,u,1},...,D^{s,u,l},...,D^{s,u,k_u}$, одержані розгалуженням $D^{s,u}$, через $D^{s+1,1},...,D^{s+1,N_{s+1}}$.

3. ***Обчислення оцінок.*** На кожному кроці розгалуження знаходимо оцінки $\xi(D^{s,l})$, $l = 1,...,N_s$, такі, що $f(\mathbf{x}) \geq \xi(D^{s,l})$ для всіх $\mathbf{x} \in D^{s,l}$. В будь-якому випадку, якщо

$$D^{s,u} = \bigcup_{l=1}^{k_u} D^{s,u,l},$$

то, як легко бачити, $\xi(D^{s,u,l}) \geq \xi(D^{s,u}), l = 1,...,k_u$.

4. *Знаходження розв'язків.* Для конкретних задач можна вказати різні способи (що, звичайно, визначаються специфікою задачі) знаходження допустимих розв'язків на послідовно розгалужуваних підмножинах.

5. *Критерій оптимальності.* Нехай

$$D^0 = \bigcup_{l=1}^{N_s} D^{s,l}$$

і $\mathbf{x}^{s,u} \in D^{s,u}$. Якщо при цьому $f(\mathbf{x}^{s,u}) = \xi(D^{s,u}) \le \xi(D^{s,l})$, $l = 1,...,N_s$, то $\mathbf{x}^{s,u}$ є оптимальним розв'язком задачі (50).

Всі описані вище дії дозволяють сформулювати

Алгоритм методу віток та границь

1. Обчислюється оцінка $\xi(D) = \xi(D^0)$. Якщо при цьому знаходиться такий план $\mathbf{x}^*$, що $f(\mathbf{x}^*) = \xi(D^0)$, то кінець: $\mathbf{x}^*$ є оптимальний розв'язок задачі (50). Інакше множина D^0 розбивається на скінченне число підмножин $D^{1,1},...,D^{1,N_1}$, причому

$$D^0 = \bigcup_{l=1}^{N_1} D^{1,l}.$$

Обчислюємо оцінки $\xi(D^{1,l})$, $l = 1,...,N_1$. Якщо для деякого l ($l = 1,...,N_1$) $D^{1,l}$ є порожньою множиною, то покладаємо $\xi(D^{1,l}) = \infty$.

2. Нехай на *s*-у кроці маємо множини $D^{s,1},...,D^{s,N_s}$, для яких $\xi(D^{s,1}),\dots,\xi(D^{s,N_s})$ є, відповідно, нижніми оцінками для значень цільової функції. Якщо вдається знайти такий план $\mathbf{x}^*$, що $\mathbf{x}^* \in D^{s,i}$ при деякому $i = 1,...,N_s$, і $f(\mathbf{x}^*) = \xi(D^{s,i}) \le \xi(D^{s,l})$ при $i = 1,...,N_s$, то кінець: $\mathbf{x}^*$ — оптимальний план. Інакше

3. Знаходимо u з умови $\xi\left(D^{s,u}\right) = \min\limits_{l=1,...,N_S} \xi\left(D^{s,l}\right)$. Якщо таких індексів декілька, то можна вибрати будь-який з них, або всі відразу. Розбиваємо множину $D^{s,u}$ на скінченне число підмножин $D^{s,u,1},...,D^{s,u,k_u}$ таких, що не перетинаються між собою, і

$$D^{s,u} = \bigcup_{l=1}^{k_u} D^{s,u,l}.$$

Обчислюємо оцінки $\xi(D^{s,u,l}), l = 1,...,k_u$, перепозначаємо множини $D^{s,1},...,D^{s,u-1},D^{s,u+1},...,D^{s,N_s}$ та $D^{s,u,l}, l = 1,...,k_u$, через $D^{s+1,1},...,D^{s+1,N_{s+1}}$ і переходимо до пункту 2 алгоритму, заміняючи s на $s+1$.

Зрозуміло, що розглянутий метод через скінченність допустимої множини D є скінченним. Однак у загальному випадку неможливо аналітично оцінити кількість ітерацій, необхідних для розв'язання конкретної оптимізаційної задачі. Практика свідчить, що для задач з великим числом змінних кількість ітерацій може виявитися неприпустимо великою навіть при розв'язанні цієї задачі на *ЕОМ*. Проте відомі приклади, коли і задачі з порівняно невеликою кількістю змінних не могли бути розв'язані методом віток та границь за прийнятний час через те, що направлений перебір незначно відрізнявся від повного.

Ще раз підкреслимо, що при реалізації вищеописаної загальної схеми методу віток та границь для окремих задач дискретного програмування необхідно розроблювати, виходячи із специфіки, правила розгалуження, способи обчислення оцінок та знаходження розв'язків.

Далі наводиться *алгоритм методу Ленд-Дойг*, який являє собою реалізацію *методу віток та границь* для задачі цілочисельного лінійного програмування:

$$L(\boldsymbol{x})=\sum_{j=1}^{n} c_j x_j \to \min, \qquad (51)$$

$$\sum_{j=1}^{n} a_{ij} x_j R_i b_i,\ i=1,...,m, \qquad (52)$$

$$0 \le x_j \le d_j,\ j=1,...,n, \qquad (53)$$

$$x_j - \text{ціле},\ j=1,...,n, \qquad (54)$$

де R_i $(i=1,...,m)$ — будь-які з відношень $\le$, $\ge$, $=$.

Деякі з чисел d_j в (53) можуть дорівнювати $+\infty$. Передбачається, що многогранна множина, яка визначена співвідношеннями (52), (53), є обмеженою.

Виклад методу Ленд-Дойг

Розв'язується допоміжна *ЗЛП* (51)–(53), яка отримана з вихідної *ЦЗЛП* (51)–(54) відкиданням умови цілочисельності змінних (54) (вітка ***0;1***). Якщо її розв'язок $\mathbf{x}(\mathbf{0};\mathbf{1})$ — цілочисельний, то він же є і розв'язком вихідної *ЦЗЛП*. Інакше величина $\xi(\mathbf{0;1}) = L(\mathbf{x}(\mathbf{0;1}))$ дає нижню оцінку (*границю*) цільової функції *ЦЗЛП* на множині $D(\mathbf{0};\mathbf{1}) = D$, що визначається співвідношеннями (52)–(53).

Нехай деяка координата $x_j(\mathbf{0};\mathbf{1})$ $(j=1,...,n)$ розв'язку $\mathbf{x}(\mathbf{0};\mathbf{1})$ не є цілочисельною. В цьому випадку здійснюється *розгалуження* множини $D(\mathbf{0};\mathbf{1})$ на дві підмножини $D(\mathbf{1};\mathbf{1})$ і $D(\mathbf{1};\mathbf{2})$ додаванням до обмежень, що задають $D(\mathbf{0};\mathbf{1})$, обмежень $x_j \le [x_j(\mathbf{0};\mathbf{1})]$ та $x_j \ge [x_j(\mathbf{0};\mathbf{1})]+1$ відповідно, де $[z]$ — ціла частина числа z. Далі розв'язуються нові допоміжні *ЗЛП* з обмеженнями, які визначаються підмножинами $D(\mathbf{1};\mathbf{1})$ та $D(\mathbf{1};\mathbf{2})$, знаходяться границі $\xi(\mathbf{1};\mathbf{1})$ та $\xi(\mathbf{1};\mathbf{2})$ і т. д.

Для подальшого розгалуження обирається *перспективна множина* $D(\mathbf{k};\mathbf{r})$ з найменшою границею $\xi(\mathbf{k};\mathbf{r})$. Процес продовжується доти, поки не буде отримано розв'язок, який задовольняє умову цілочисельності і для якого виконується ознака оптимальності (див. п. 4 алгоритму). Внаслідок обмеженості допустимої множини *ЗЛП* (скінченності допустимої множини *ЦЗЛП*) метод Ленд-Дойг скінченний.

Алгоритм методу Ленд-Дойг

1. Визначаються множини $D(\mathbf{k};\mathbf{r})$ умовами (52), (53) і додатковими обмеженнями, які виникають в процесі розгалуження (див. пункт 5). На ***0***-у кроці покладаємо $D(\mathbf{0};\mathbf{1}) = D$, де D задається умовами (52), (53).

2. Розв'язуються допоміжні *ЗЛП* на множинах $D(\mathbf{k};\mathbf{r})$. Нехай $\mathbf{x}(\mathbf{k};\mathbf{r})$ — оптимальні розв'язки вказаних *ЗЛП*.

3. Обчислюються границі на множинах $D(\mathbf{k};\mathbf{r})$ за формулою $\xi(\mathbf{k};\mathbf{r}) = L(\mathbf{x}(\mathbf{k};\mathbf{r}))$.

4. Якщо існують k,l такі, що $\mathbf{x}(\mathbf{k};\mathbf{l})$ — цілочисельний розв'язок та для всіх віток $\mathbf{r}$ на $\mathbf{k}$ - у кроці виконуються співвідношення

$$L(\mathbf{x}(\mathbf{k};\mathbf{l})) = \xi(\mathbf{k};\mathbf{l}) \le \xi(\mathbf{k};\mathbf{r}),$$

то $\mathbf{x}^* = \mathbf{x}(\mathbf{k};\mathbf{l})$ — оптимальний розв'язок *ЦЗЛП*.

5. Розгалуження здійснюється по нецілочисельній компоненті $\mathbf{x}_j(\mathbf{k};\mathbf{r})$ (з мінімальним j) розв'язку $\mathbf{x}(\mathbf{k};\mathbf{r})$, що відповідає перспективній вітці $\mathbf{k};\mathbf{r}$ (якщо таких віток декілька, то вибирається вітка з мінімальним номером $\mathbf{r}$), додаванням до $D(\mathbf{k};\mathbf{r})$ однієї з підмножин $\mathbf{x}_j \le \lfloor \mathbf{x}_j(\mathbf{k};\mathbf{r}) \rfloor$ або $\mathbf{x}_j \ge \lfloor \mathbf{x}_j(\mathbf{k};\mathbf{r}) \rfloor + 1$.

Приклад 5. *Розв'язати задачу*

$$L(\mathbf{x}) = -x_1 - 3x_2 \to \min,$$
$$x_1 + 4x_2 \le 14,$$
$$2x_1 + 3x_2 \le 12,$$
$$x_j \ge 0, x_j - \text{ціле}, j = 1,2.$$

Розв'язок, що наводиться нижче, ілюструється схемою, яка представлена на рис. 7.

Будемо позначати через D з належними індексами допустимі області допоміжних *ЗЛП*, які отримуються з допустимих множин відповідних *ЦЗЛП* відкиданням умови цілочисельності.

0-й крок. Розв'язуємо допоміжну *ЗЛП*

$$L(\mathbf{x}) = -x_1 - 3x_2 \to \min, \mathbf{x} \in D^0.$$

Її розв'язок має вигляд (див. рис. 4): $\mathbf{x}^0 = (6/5, 16/5), L(\mathbf{x}^0) = -54/5.$

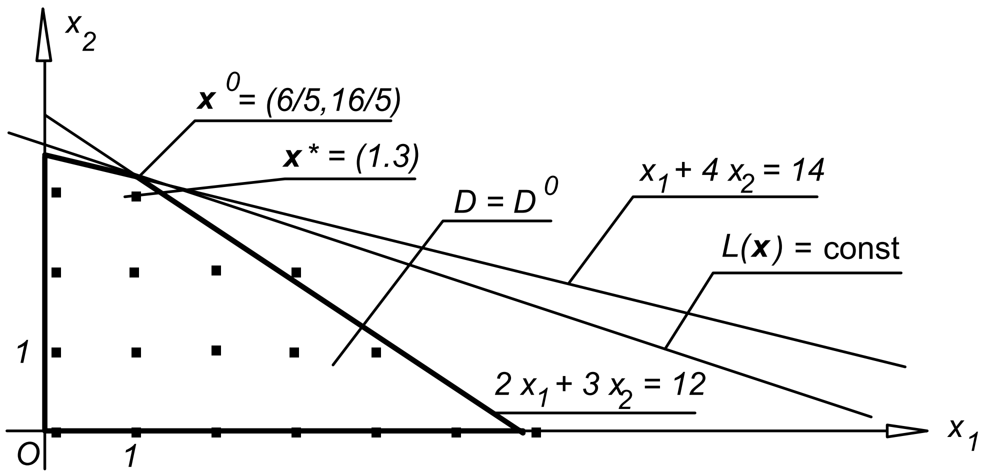

Рис. 4

Обчислюємо границю $\xi(D^0) = L(\mathbf{x}^0) = -54/5$.

Проводимо розгалуження множини D^0:

$D^0 = D^{1,1} \cup D^{1,2}$,

$D^{1,1} = \{\mathbf{x} : \mathbf{x} \in D^0, x_1 \le [6/5] = 1\}$,

$D^{1,2} = \{\mathbf{x} : \mathbf{x} \in D^0, x_1 \ge [6/5] + 1 = 2\}$.

1-й крок. Розв'язуємо допоміжні *ЗЛП*

$L(\mathbf{x}) = -x_1 - 3x_2 \to \min, \mathbf{x} \in D^{1,1}$.

$L(\mathbf{x}) = -x_1 - 3x_2 \to \min, \mathbf{x} \in D^{1,2}$.

Їх розв'язки мають вигляд (див. рис. 5):

$\mathbf{x}^{1,1} = (1,13/4),\ L(\mathbf{x}^{1,1}) = -43/4, \qquad \mathbf{x}^{1,2} = (2,8/3),\ L(\mathbf{x}^{1,2}) = -10.$

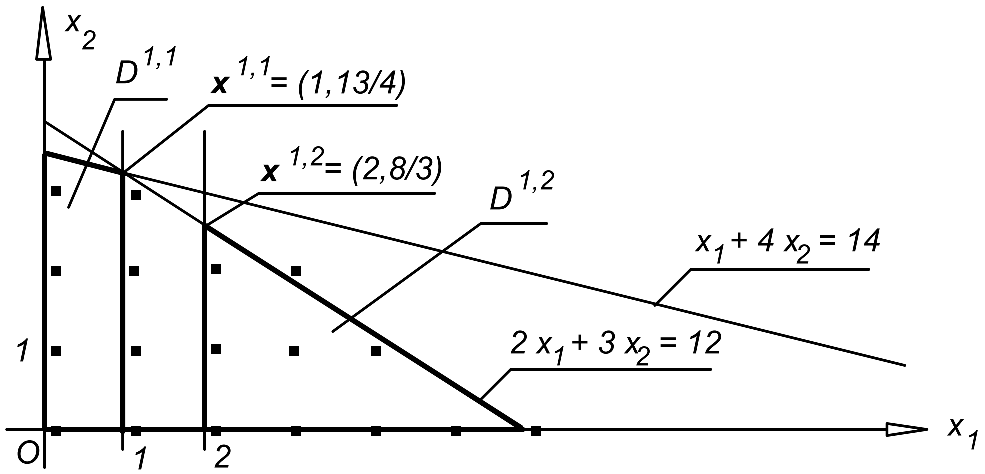

Рис. 5

Обчислюємо границі $\xi(D^{1,1}) = L(\mathbf{x}^{1,1}) = -43/4,\ \xi(D^{1,2}) = L(\mathbf{x}^{1,2}) = -10.$

Проводимо розгалуження множини $D^{1,1}$:

$D^{1,1} = D^{2,1} \cup D^{2,2}$,

$D^{2,1} = \{\mathbf{x}\colon \mathbf{x} \in D^{1,1}, x_2 \le [13/4] = 3\}$,

$D^{2,2} = \{\mathbf{x}\colon \mathbf{x} \in D^{1,1}, x_2 \ge [13/4] + 1 = 4\}$.

Покладаємо $D^{2,3} = D^{1,2}$.

2-й крок. Розв'язуємо допоміжні *ЗЛП*

$$L(\mathbf{x}) = -x_1 - 3x_2 \to \min,\ \mathbf{x} \in D^{2,1}.$$
$$L(\mathbf{x}) = -x_1 - 3x_2 \to \min,\ \mathbf{x} \in D^{2,2}.$$

Розв'язок першої задачі має вигляд (див. рис. 6):

$$\mathbf{x}^{2,1} = (1,3),\ \ L(\mathbf{x}^{2,1}) = -10,$$

друга задача розв'язку не має, оскільки множина $D^{2,2}$ порожня.

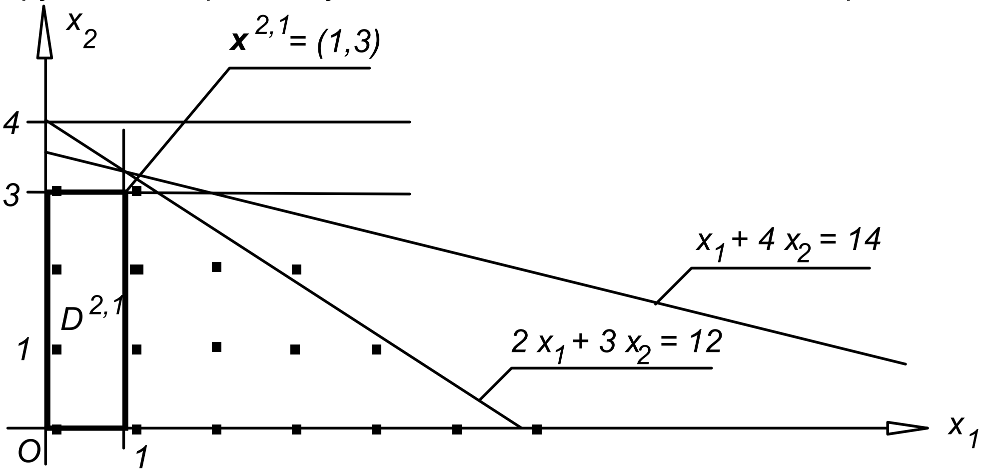

Рис. 6

Обчислюємо границю $\xi(D^{2,1}) = L(\mathbf{x}^{2,1}) = -10$ та покладаємо $\xi(D^{2,2}) = \infty$.

Аналізуючи розв'язки *ЗЛП*, пов'язаних з областями $D^{2,1}$, $D^{2,2}$, $D^{2,3}$, та використовуючи критерій оптимальності, приходимо до висновку, що розв'язок вихідної *ЦЗЛП* має вигляд: $\mathbf{x}^* = (1,3),\ \ L(\mathbf{x}^*) = -10.$

Зауважимо ще раз, що процес розв'язування наведеного прикладу ілюструється схемою, яка представлена на рис. 7.

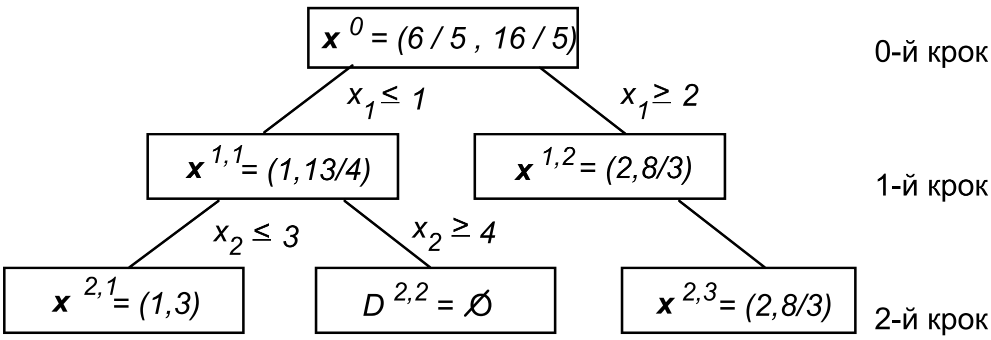

Рис. 7

§ 7. Задача про оптимальні призначення. Угорський метод. Метод Мака

Математичну постановку задачі про оптимальні призначення наведено у § 1:

знайти матрицю $\mathbf{X} = \left\| x_{i,j} \right\|,\ i, j = 1,...,n,$ *що мінімізує цільову функцію*

$$L(\boldsymbol{X}) = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} \qquad (55)$$

при виконанні обмежень

$$\sum_{i=1}^{n} x_{ij} = 1, \; j = 1,...,n, \tag{56}$$

$$\sum_{j=1}^{n} x_{ij} = 1, \; i = 1,...,n, \tag{57}$$

$$x_{ij} = 1 \text{ або } 0, \; i, j = 1,...,n. \tag{58}$$

де c_{ij} — витрати, пов'язані з використанням i-го виконавця для виконання j-ї роботи. *Елементи c_{ij} утворюють матрицю витрат* $\mathbf{C}$.

У відповідності з постановкою цієї задачі, розв'язати її — значить, іншими словами, вибрати у матриці $\mathbf{C}$ n елементів, по одному з кожного рядка (рядки відповідають виконавцям) і кожного стовпця (стовпці відповідають роботам) так, щоб сума вибраних елементів, яка дорівнює сумарним витратам, пов'язаним з призначеннями, була найменшою порівняно з її значеннями при всіх інших таких призначеннях.

Зауважимо, що задачі математичного програмування, в яких на змінні накладаються умови (58), називаються *задачами з булевими змінними.* Отже, *задача про оптимальні призначення є задачею лінійного програмування (ЗЛП) з булевими змінними*. Крім того, цю задачу можна розглядати як частинний випадок транспортної задачі лінійного програмування. Дійсно, якщо умову (58) замінити умовою невід'ємності змінних, то (55)–(58) перетворюється у звичайну транспортну задачу, в якої об'єми виробництва та об'єми споживання є цілочисельними і рівними одиниці. Якщо розв'язати цю задачу методом потенціалів, або іншим методом, що забезпечує цілочисельний оптимальний розв'язок при цілочисельних об'ємах виробництва та споживання, то одержаний розв'язок буде автоматично задовольняти не враховане обмеження (58). Проте специфіка цієї задачі дозволяє розв'язати її більш простими методами, ніж метод потенціалів. Такими методами є, наприклад, угорський метод та метод Мака.

Квадратні матриці $\mathbf{C} = \|c_{ij}\|$ та $\mathbf{D} = \|d_{ij}\|$ $(i, j = 1,...,n)$ будемо називати *еквівалентними*, якщо елементи однієї з них одержуються з елементів другої шляхом додавання певних чисел до кожного рядка і кожного стовпця. Ці числа можуть бути різними для різних рядків та стовпців. Отже, матриці $\mathbf{C}$ та $\mathbf{D}$ еквівалентні, якщо, наприклад,

$$d_{ij} = c_{ij} + \alpha_i + \beta_j, \; i, j = 1,...,n. \tag{59}$$

Зрозуміло, що ця властивість є взаємною.

Приклад 6. Матриці

$$\mathbf{C} = \begin{Vmatrix} 4 & 0 & -1 & 5 \\ 2 & 3 & 0 & -1 \\ 1 & 1 & 2 & 3 \\ -2 & -2 & -3 & 0 \end{Vmatrix} \quad \text{та} \quad \mathbf{D} = \begin{Vmatrix} 6 & 3 & 3 & 6 \\ 4 & 6 & 4 & 0 \\ 4 & 5 & 7 & 5 \\ -1 & 0 & 0 & 0 \end{Vmatrix}$$

є еквівалентними, бо матриця $\mathbf{D}$ одержується з матриці $\mathbf{C}$ додаванням до рядків чисел *0, 0, 1, –1* і додаванням до стовпців чисел *2, 3, 4, 1,* відповідно.

Теорема 4. *Оптимальні розв'язки задач про оптимальні призначення з еквівалентними матрицями витрат співпадають.*

Доведення. Нехай матриця $\mathbf{C}$ еквівалентна матриці $\mathbf{D}$ і $(1, j_1)$, $(2, j_2), \ldots, (n, j_n)$ — оптимальні призначення у задачі з матрицею витрат $\mathbf{C}$, тобто $i-$й виконавець призначається на виконання роботи $j_i, i = 1,...,n.$ Припустимо, що це призначення не є оптимальним у задачі з матрицею витрат $\mathbf{D}$.

Нехай $(1, j_1'),\ (2, j_2'), ..., (n, j_n')$ — оптимальні призначення у задачі з матрицею витрат $\mathbf{D}$, причому

$$\sum_{i=1}^{n} d_{ij_i'} < \sum_{i=1}^{n} d_{ij_i}.$$

В силу (59) звідси маємо

$$\sum_{i=1}^{n} (c_{ij_i'} + \alpha_i + \beta_{j_i'}) < \sum_{i=1}^{n} (c_{ij_i} + \alpha_i + \beta_{j_i})$$

або

$$\sum_{i=1}^{n} c_{ij_i'} + \sum_{i=1}^{n} \beta_{j_i'} < \sum_{i=1}^{n} c_{ij_i} + \sum_{i=1}^{n} \beta_{j_i}. \tag{60}$$

Проте $\sum_{i=1}^{n} \beta_{j_i'} = \sum_{i=1}^{n} \beta_{j_i}$, бо в обох випадках це є сума чисел, які додаються до стовпців матриці $\mathbf{C}$. Отже, з (60) маємо, що $\sum_{i=1}^{n} c_{ij_i'} < \sum_{i=1}^{n} c_{ij_i}$, що протирічить тому, що для задачі з матрицею витрат $\mathbf{C}$ призначення $(i, j_i),\ i = 1,...,n,$ є оптимальним.

Оскільки властивість еквівалентності матриць є взаємною, то оптимальні призначення у задачі з матрицею витрат $\mathbf{C}$ співпадають з оптимальними призначеннями у задачі з матрицею витрат $\mathbf{D}$. Доведення завершене.

Доведена теорема дозволяє, якщо це необхідно, переходити від даної задачі про оптимальні призначення до задачі з еквівалентною матрицею витрат. Тому вихідну задачу завжди можна звести до задачі про оптимальні призначення з матрицею витрат, яка має лише невід'ємні елементи. Оскільки найменше можливе значення суми n елементів такої матриці, очевидно, дорівнює нулю, то задача зводиться до вибору у матриці витрат, або еквівалентній їй, n нульових елементів, по одному в кожному рядку і кожному стовпці. В цьому, власне, полягає неформальний зміст алгоритму угорського методу та методу Мака, що викладаються нижче, де матриці, еквівалентні вихідній матриці витрат $\mathbf{C}$, називаються просто матрицями витрат.

Алгоритм угорського методу

1. Віднімаємо у матриці $\mathbf{C}$ від кожного елемента i-го рядка мінімальний елемент цього рядка, $i = 1,...,n.$

2. Віднімаємо від кожного елемента j-го стовпця перетвореної матриці витрат його мінімальний елемент, $j = 1,...,n.$

В результаті виконання двох пунктів кожний рядок і кожний стовпець матриці витрат мають принаймні один *0*.

3. Проглядаємо послідовно рядки матриці витрат, починаючи з першого. Якщо рядок має лише один непозначений *0*, позначаємо його символом ***** і закреслюємо (за допомогою символа **^**) всі нулі у цьому ж стовпці. *0* вважається позначеним, якщо він позначений символом *. Повторюємо ці дії, поки кожен рядок не буде мати непозначених нулів, або буде мати їх принаймні два.

4. Дії пункту 3 повторюємо для всіх стовпців матриці витрат.

5. Дії пунктів 3 та 4 повторюємо послідовно (якщо необхідно), поки не одержимо один з трьох можливих випадків:

i) кожний рядок має призначення (має *0* з позначкою *****);

ii) є принаймні два непозначених нулі в деяких рядках і деяких стовпцях матриці витрат;

iii) немає непозначених нулів і повне призначення ще не отримане (число нулів з позначкою * менше n).

6. У випадку i) задача про оптимальні призначення розв'язана: x_{ij}^*, що відповідають 0^*, дорівнюють *1*, решта — *0*, кінець алгоритму. У випадку ii) довільно вибираємо один з непозначених нулів, позначаємо його символом *****, закреслюємо *решту* нулів у тому ж рядку і у тому ж стовпці і повертаємося до пункту 3. Якщо має місце випадок iii), то переходимо до пункту 7.

7. Позначаємо символом **#** рядки, для яких не одержано призначення (в яких немає 0^*). Такі рядки вважаємо позначеними, решту — непозначеними. Таку ж термінологію будемо використовувати і для стовпців матриці витрат.

8. Позначаємо символом **#** ще непозначені стовпці, які мають закреслений *0* (позначений символом **^**) у позначених рядках.

9. Позначаємо символом **#** ще непозначені рядки, які мають призначення (тобто 0^*) у позначених стовпцях.

10. Повторюємо дії пунктів 8 та 9 доти, поки більше не можна позначити рядків і стовпців матриці витрат.

11. Закреслюємо (за допомогою позначки &) непозначені рядки і позначені стовпці матриці витрат.

12. Знаходимо мінімальний незакреслений елемент матриці витрат, віднімаємо його від елементів кожного з незакреслених рядків, додаємо до елементів усіх закреслених стовпців і переходимо до пункту 3. При цьому позначки елементів матриці витрат (* та ^) втрачають свою силу.

Зауважимо, що якщо в задачі про оптимальні призначення (55)-(58) цільову функцію (55) потрібно максимізувати (у цьому випадку c_{ij} - ефективність, пов'язана з призначенням i - го виконавця на j - й вид роботи), то для її розв'язання можна застосувати угорський метод, замінивши матрицю $\mathbf{C}$ на $-\mathbf{C}$.

Приклад 7. *Розв'язати задачу про оптимальні призначення з матрицею витрат* $\mathbf{C}$.

$$\boldsymbol{C} = \begin{Vmatrix} 3 & 10 & 5 & 9 & 16 \\ 6 & 8 & 11 & 8 & 18 \\ 7 & 13 & 10 & 3 & 4 \\ 5 & 9 & 6 & 21 & 12 \\ 5 & 4 & 11 & 6 & 13 \end{Vmatrix} \qquad \boldsymbol{C}_1 = \begin{array}{cc} \begin{array}{ccccc} \# & & & & \end{array} & \\ \begin{Vmatrix} 0^* & 7 & 1 & 6 & 12 \\ 0^\wedge & 2 & 4 & 2 & 11 \\ 4 & 10 & 6 & 0^* & 0^\wedge \\ 0^\wedge & 4 & 0^* & 16 & 6 \\ 1 & 0^* & 6 & 2 & 8 \end{Vmatrix} & \begin{array}{c} \# \\ \# \\ \\ \\ \\ \end{array} \end{array}$$

Виконуючи дії пунктів 1, 2 алгоритму, які інколи називають попередніми перетвореннями, одержуємо еквівалентну матрицю $\mathbf{C}_1$. Позначаємо нулі матриці $\mathbf{C}_1$ у відповідності з пунктами 3, 4, 5 алгоритму. Оскільки кількість нулів, позначених *, менша $n = 5$, то переходимо до пункту 7. Виконуючи дії пунктів 7–10, позначаємо символом **#** спочатку другий рядок, потім перший стовпець та перший рядок матриці $\mathbf{C}_1$. Після виконання пункту 11 алгоритму одержуємо матрицю $\mathbf{C}_2$.

$$\boldsymbol{C}_2 = \begin{array}{cc} \begin{array}{ccccc} \& & & & & \end{array} & \\ \left\| \begin{array}{ccccc} 0^* & 7 & 1 & 6 & 12 \\ 0^\wedge & 2 & 4 & 2 & 11 \\ 4 & 10 & 6 & 0^* & 0^\wedge \\ 0^\wedge & 4 & 0^* & 16 & 6 \\ 1 & 0^* & 6 & 2 & 8 \end{array} \right\| & \begin{array}{c} \\ \\ \& \\ \& \\ \& \end{array} \end{array} \qquad \boldsymbol{C}_3 = \begin{array}{cc} \begin{array}{ccccc} \# & & \# & & \end{array} & \\ \left\| \begin{array}{ccccc} 0^\wedge & 6 & 0^\wedge & 5 & 11 \\ 0^* & 1 & 3 & 1 & 10 \\ 5 & 10 & 6 & 0^* & 0^\wedge \\ 1 & 4 & 0^* & 16 & 6 \\ 2 & 0^* & 6 & 2 & 8 \end{array} \right\| & \begin{array}{c} \# \\ \# \\ \\ \# \\ \\ \end{array} \end{array}$$

Мінімальний незакреслений елемент матриці $\mathbf{C}_2$ дорівнює *1*. Після виконання пункту 12 алгоритму одержуємо матрицю витрат $\mathbf{C}_3$, до якої також внесені позначки у відповідності з діями пунктів 3–10 алгоритму. Виконання пунктів 11, 12 приводить, відповідно, до матриць $\mathbf{C}_4$ та $\mathbf{C}_5$.

$$\boldsymbol{C}_4 = \begin{array}{cc} \begin{array}{ccccc} \& & & \& & & \end{array} & \\ \left\| \begin{array}{ccccc} 0^\wedge & 6 & 0^\wedge & 5 & 11 \\ 0^* & 1 & 3 & 1 & 10 \\ 5 & 10 & 6 & 0^* & 0^\wedge \\ 1 & 4 & 0^* & 16 & 6 \\ 2 & 0^* & 6 & 2 & 8 \end{array} \right\| & \begin{array}{c} \\ \\ \& \\ \\ \& \end{array} \end{array} \qquad \boldsymbol{C}_5 = \left\| \begin{array}{ccccc} 0^* & 5 & 0^\wedge & 4 & 10 \\ 0^\wedge & 0^\wedge & 3 & 0^* & 9 \\ 6 & 10 & 7 & 0^\wedge & 0^* \\ 1 & 3 & 0^* & 15 & 5 \\ 3 & 0^* & 7 & 2 & 8 \end{array} \right\|$$

Матриця $\mathbf{C}_5$ містить також позначки, що відображають дії пунктів 3–5 алгоритму на наступній ітерації. Так як кількість *0** дорівнює *5*, то одержано оптимальний розв'язок $\mathbf{x}^*$:

$$\boldsymbol{x}^* = \left\| \begin{array}{ccccc} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{array} \right\|,$$

тобто перший виконавець призначається на виконання першої роботи, другий — четвертої, третій — п'ятої, четвертий — третьої, п'ятий — другої. При цьому $L(\mathbf{x}^*) = 25$.

Алгоритм методу Мака

1. Позначаємо мінімальний елемент кожного рядка матриці витрат позначкою *. Якщо таких елементів декілька, позначаємо будь який з них.

2. Для кожного рядка, що має інший мінімальний елемент, проглядаємо стовпець, до якого цей елемент належить. Можливі випадки:

i) стовпець не має позначених елементів;

ii) стовпець має принаймні один позначений елемент.

3. У випадку i) позначаємо інший мінімальний елемент рядка позначкою *. Всі інші позначки в цьому рядку ліквідовуються. У випадку ii) позначаємо інший мінімальний елемент рядка позначкою **^**, якщо елемент цього рядка з позначкою * не є єдиним позначеним елементом у своєму стовпці.

4. Дії пунктів 2 та 3 повторюємо послідовно для всіх рядків, що мають більше одного мінімального елемента.

5. Якщо кожний стовпець матриці витрат має елемент з позначкою *, то кінець алгоритму: ці елементи визначають оптимальні призначення. Інакше переходимо до наступного пункту.

6. Позначаємо (символом **&**) стовпці, що мають більше одного позначеного елемента. Вони утворюють множину $\mathbf{B}$, інші стовпці матриці витрат утворюють множину $\mathbf{A}$.

7. Для кожного рядка матриці витрат, в якому елемент з позначкою * належить множині $\mathbf{B}$, знаходиться мінімальна різниця між елементами множини $\mathbf{A}$ і елементом з позначкою *.

8. Знаходимо найменшу з вказаних різниць, додаємо її до кожного елемента множини $\mathbf{B}$ і повертаємось до пункту 2.

Приклад 8. *Розв'язати задачу про оптимальні призначення з матрицею витрат* $\mathbf{C}$ (*співпадає з умовою прикладу* 7).

$$C = \begin{array}{ccccc} \& & & & & \\ \hline \end{array}\! \begin{array}{c} \begin{matrix} \& & & & & \end{matrix} \\ \begin{Vmatrix} 3^* & 10 & 5 & 9 & 16 \\ 6^* & 8 & 11 & 8 & 18 \\ 7 & 13 & 10 & 3^* & 4 \\ 5^* & 9 & 6 & 21 & 12 \\ 5 & 4^* & 11 & 6 & 13 \end{Vmatrix} \end{array} \qquad C_1 = \begin{array}{c} \begin{matrix} \& & & & & \end{matrix} \\ \begin{Vmatrix} 4^* & 10 & 5 & 9 & 16 \\ 7^* & 8 & 11 & 8 & 18 \\ 8 & 13 & 10 & 3^* & 4 \\ 6 & 9 & 6^* & 21 & 12 \\ 6 & 4^* & 11 & 6 & 13 \end{Vmatrix} \end{array}$$

В матриці $\mathbf{C}$ позначкою * позначені елементи згідно пункту 1 алгоритму. Інших мінімальних елементів рядки матриці витрат не мають, тому, виконуючи пункт 6, позначаємо перший стовпець символом **&**. Множина $\mathbf{B}$ містить перший стовпець, решта стовпців належать до множини $\mathbf{A}$. Мінімальні різниці обчислюються для першого, другого та четвертого рядків. Вони дорівнюють, відповідно, *2*, *2* та *1*. Мінімальна з цих різниць (*1*) додається до елементів першого стовпця. Оскільки у четвертому рядку з'явився інший мінімальний елемент (*6*), то, виконуючи пункти 3 та 4 алгоритму, позначаємо його позначкою *. Перший стовпець формує множину $\mathbf{B}$ (дивись матрицю $\mathbf{C}_1$). Мінімальна різниця дорівнює *1*.

$$C_2 = \begin{array}{c} \begin{matrix} \& & \& & \& & \& & \end{matrix} \\ \begin{Vmatrix} 5^* & 10 & 5^\wedge & 9 & 16 \\ 8^* & 8^\wedge & 11 & 8^\wedge & 18 \\ 9 & 13 & 10 & 3^* & 4 \\ 7 & 9 & 6^* & 21 & 12 \\ 7 & 4^* & 11 & 6 & 13 \end{Vmatrix} \end{array} \qquad C_3 = \begin{Vmatrix} 6^* & 11 & 6^\wedge & 10 & 16 \\ 9^* & 9^\wedge & 12 & 9^\wedge & 18 \\ 10 & 14 & 11 & 4^* & 4 \\ 8 & 10 & 7^* & 22 & 12 \\ 8 & 5^* & 12 & 7 & 13 \end{Vmatrix}$$

Дії наступного кроку наведені в матриці $\mathbf{C}_2$. Мінімальна різниця дорівнює *1*. Збільшуючи елементи множини $\mathbf{B}$ на *1*, одержуємо матрицю $\mathbf{C}_3$. Позначаючи символом * інший мінімальний елемент (*4*) у третьому рядку, позначаємо символом * інший мінімальний елемент другого рядка (*9*), що знаходиться у четвертому стовпці.

$$C_4 = \begin{Vmatrix} 6^* & 11 & 6 & 10 & 16 \\ 9 & 9 & 12 & 9^* & 18 \\ 10 & 14 & 11 & 4 & 4^* \\ 8 & 10 & 7^* & 22 & 12 \\ 8 & 5^* & 12 & 7 & 13 \end{Vmatrix} \qquad x^* = \begin{Vmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{Vmatrix}$$

Після цього кожен стовпець матриці витрат має елемент, позначений позначкою * (див. матрицю $\mathbf{C}_4$). Отже, оптимальні призначення визначаються матрицею $\mathbf{x}^*$, що наводиться вище. При цьому $L(\mathbf{x}^*) = 25$.

Література

1. Ермольев Ю.М., Ляшко И.И., Михалевич В.С., Тюптя В.И. Математические методы исследования операций. К.: Вища школа, 1979.
2. Зайченко Ю.П., Шумилова С.А. Исследование операций. Сборник задач. К.: Вища школа, 1990.
3. Корбут А.А.,Финкельштейн Ю.Ю. Дискретное программирование. М.: Наука, 1969.
4. Попов Ю.Д. Линейное и нелинейное программирование. К.: УМК ВО, 1988.
5. Попов Ю.Д., Тюптя В.І., Шевченко В.І. Методи оптимізації. Навчальний посібник для студентів спеціальностей «Прикладна математика», «Інформатика», «Соціальна інформатика». К.: Абрис, 1999.

Зміст

Навчальне електронне видання

ТЮПТЯ Володимир Іванович
ШЕВЧЕНКО Віталій Іванович
СТРЮК Віктор Кіндратович

“Дискретне програмування”.
Методичні вказівки
до проведення практичних та самостійних занять
з курсу “Дослідження операцій”
для студентів факультету кібернетики