Лекція 10. Цілочислове програмування

10.1. Постановка задач цілочислового програмування.

Серед прикладних задач оптимізації важливе місце займають задачі дискретного характеру. Розрізняють задачі комбінаторного типу, в яких область допустимих значень складається зі скінченної кількості точок, задачі цілочислового програмування, в яких змінні набувають цілочислових значень і комбіновані задачі (задачі частково дискретного програмування), що містять лише частину дискретних змінних.

Задача про рюкзак.

Мандрівник, збираючись у похід має упакувати з собою деякі з n предметів, що можуть йому знадобитись. Відома корисність c_j одного предмету j-го найменування. У поході можуть бути потрібними декілька однакових предметів. Рюкзак має m обмежень за своїми характеристиками (об‘єм, лінійні розміри, вага тощо). Нехай a_{ij} – i-та характеристика $\left(i=\overline{1,m}\right)$ предмету j-го найменування $\left(j=\overline{1,n}\right)$, b_i – максимальне значення i-ої характеристики рюкзака.

Визначити, які предмети та в якій кількості слід завантажити в рюкзак, щоб їх сумарна корисність була максимальною.

Складемо математичну модель задачі у вигляді ЗЛП. Нехай x_j - кількість предметів j-го найменування, що потрібно завантажити в рюкзак, тоді

$$Z(x)=\sum_{j=1}^{n}c_jx_j\to\max$$

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

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

Задача про оптимальні призначення.

Є n видів робіт та n кандидатів для їх виконання (виконавців). Вважається, що кожен виконавець $i=\overline{1,n}$ може виконувати будь-яку роботу $j=\overline{1,n}$. Відомі витрати c_{ij} при виконанні j-ої роботи i-им виконавцем. Необхідно розподілити кандидатів на виконання робіт таким чином, щоб кожен виконавець отримав єдине призначення, кожна роботи виконувалась єдиним виконавцем і сумарні витрати від призначень були мінімальними.

Складемо математичну модель задачі. Необхідно знайти матрицю $x=\left(x_{ij}\right)$ $i=\overline{1,n},\ j=\overline{1,n}$, що мінімізує цільову функцію

$$Z(x)=\sum_{i=1}^{m}\sum_{j=1}^{n}c_{ij}x_{ij}\to \min$$

за умов

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

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

$$x_{ij}\in\{0,1\};\ i,j=\overline{1,n}.$$

Це типова комбінаторна задача, розв‘язком якої є деяка перестановка чисел $1,...,n$. Усього таких перестановок $n!$, тому при великих n розв‘язати задачу шляхом прямого перебору неможливо.

Цілочисловою задачею лінійного програмування (ЦЗЛП) у канонічній формі будемо називати задачу вигляду

$$Z(x)=\sum_{j=1}^{n}c_jx_j\to \max \tag{10.1}$$

$$\sum_{j=1}^{n}a_{ij}x_j=b_i,\ i=\overline{1,m}$$
$$x_j\ge 0,\ j=\overline{1,n}, \tag{10.2}$$

$$x_j - \text{ціле},\ j=\overline{1,k}\,(k\le n). \tag{10.3}$$

Дискретною задачею лінійного програмування (ДЗЛП) у канонічній формі будемо називати задачу вигляду

$$Z(x)=\sum_{j=1}^{n}c_jx_j\to \max$$
$$\sum_{j=1}^{n}a_{ij}x_j=b_i,\ i=\overline{1,m}$$
$$x_j\ge 0,\ j=\overline{1,n}, \tag{10.4}$$
$$x_j\in\{x_j^1,...,x_j^k\},\ j=\overline{1,k}\,(k\le n).$$

В задачах дискретного програмування множина допустимих розв‘язків є неопуклою і незв‘язною. Тому знаходження розв‘язку значно ускладнено. Неможливо застосування стандартних прийомів, що дозволяють замінити дискретну задачу її неперервним аналогом і надалі застосувати округлення розв‘язку до найближчого цілочислового.

Приміром, для задачі

$$Z(x) = x_1 - 3x_2 + 3x_3 \to \max$$

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

$$x_j \ge 0, x_j - \text{ціле}, \ j = 1,3$$

розв‘язок, знайдений симплекс-методом без урахування цілочисельності змінних $\tilde{x}^* = \left(\frac{1}{2};0;\frac{9}{2}\right)$, $Z\left(\tilde{x}^*\right) = 14$. А оптимальний розв‘язок ЦЗЛП $x^* = (2;\,2;\,5)$, $Z\left(x^*\right) = 11$.

Одним з методів розв‘язання ЦЗЛП є метод віток і меж. Цей метод належить до класу комбінаторних методів і зводиться до направленого перебору варіантів розв‘язку оптимізаційної задачі, коли розглядаються лише ті з них, які виявляються за певними ознаками перспективними, і відкидаються одразу цілі множини варіантів, що є безперспективними. Для задач лінійного програмування реалізацією методу віток і меж є метод Ленд-Дойг. Передбачається, що ОДР задачі обмежена.

Розглядається допоміжна (послаблена) ЗЛП (10.1), (10.2) яка одержана з вихідної ЦЗЛП (10.1) – (10.3) відкиданням умови цілочисельності змінних (10.3) (вітка 0;1). Якщо її розв‘язок $X^{0,1}$ цілочисловий, то він і є розв‘язком задачі $x^{0,1} = x^*$. Інакше величина $\xi^{0,1} = Z\left(x^{0,1}\right)$ дає верхню оцінку (межу) цільової функції на множині $G^{0,1} = G$, що визначається співвідношеннями (10.2), (10.3). Нехай деяка координата $x_j^{0,1}$ $j = \overline{1,n}$ розв‘язку $x^{0,1}$ не є цілочисловою. В цьому випадку здійснюється розгалуження множини $G^{0,1}$ на підмножини $G^{1,1}$ і $G^{1,2}$. Розгалуження розбиває ОДР на дві підмножини шляхом формування додаткових обмежень з метою виключення нецілочислової точки $x^{0,1}$ і знаходження розв‘язку принаймні однієї з нових задач цілочисловим за обраною координатою $x_j^{0,1}$.

Змінна $x_j^{0,1}$ вибирається виходячи з умов:

1) нецілочислова координата з найменшим індексом;

2) нецілочислова координата з найменшою (найбільшою) дробовою частиною;

3) нецілочислова координата, якій відповідає найбільший коефіцієнт в цільовій функції;

4) нецілочислова координата обирається на основі пріоритету, виходячи з фізичного змісту задачі.

Додаткові обмеження мають вигляд $x_j \le \left[x_j^{0,1}\right]$ і $x_j \ge \left[x_j^{0,1}\right]+1$ відповідно, де $[x]$ - ціла частина числа x. Далі розв‘язуються нові допоміжні ЗЛП з обмеженнями, які визначаються підмножинами $G^{1,1}$ і $G^{1,2}$:

$$Z(x)=\sum_{j=1}^{n} c_j x_j \to \max \qquad\qquad Z(x)=\sum_{j=1}^{n} c_j x_j \to \max$$

$$G^{1,1}:\begin{cases}\sum_{j=1}^{n} a_{ij}x_j = b_i,\ i=\overline{1,m}\\ x_j \le \left[x_j^{0,1}\right]\\ x_j \ge 0,\ j=\overline{1,n},\end{cases} \qquad G^{1,2}:\begin{cases}\sum_{j=1}^{n} a_{ij}x_j = b_i,\ i=\overline{1,m}\\ x_j \ge \left[x_j^{0,1}\right]+1\\ x_j \ge 0,\ j=\overline{1,n},\end{cases}$$

знаходять межі $\xi^{1,1}=Z\left(x^{1,1}\right)$ і $\xi^{1,2}=Z\left(x^{1,2}\right)$. Для подальшого розгалуження вибирається перспективна множина $G^{k,r}$ з найбільшою межею $\xi^{k,r}$. 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 $G^{k,r}$ з умов (10.1), (10.2) і додатковими обмеженнями, що виникають в процесі розгалуження (п.5). На 0-му кроці покладають $G^{0,1}=G$, де G задається умовами (10.1), (10.2).

2. Розв‘язують допоміжні задачі на множинах $G^{k,r}$. Нехай $x^{k,r}$ - оптимальні розв‘язки цих задач.

3. Обчислюють оцінки (межі) $\xi^{k,r}=Z\left(x^{k,r}\right)$.

4. Якщо існують k і l такі, що $x^{k,l}$ - цілочисловий і для усіх віток r на k-му кроці виконується співвідношення $Z\left(x^{k,l}\right)=\xi^{k,l}\ge\xi^{k,r}$. Тоді $x^*=x^{k,l}$ оптимальний розв‘язок ЦЗЛП.

5. Розгалуження здійснюється за нецілочисловою компонентою $x_j^{k,r}$ (з мінімальним номером j) розв‘язку $x^{k,r}$, що відповідає перспективній вітці $(k;r)$ додаванням до множини $G^{k,r}$ одного з обмежень $x_j \le \left[x_j^{k,r}\right]$ або $x_j \ge \left[x_j^{k,r}\right]+1$.

Приклад 10.1. Розв‘язати ЦЗЛП

$$Z(x) = x_1 + 3x_2 \to \max$$

$$\begin{cases} x_1 + 4x_2 \le 14; \\ 2x_1 + 3x_2 \le 12; \end{cases}$$

$x_j \ge 0,\ x_j$ – цілі, $j = 1,2$.

0 крок. Розв‘язуємо допоміжну задачу без умови цілочисельності змінних (рис.10.1). $Z(x) = x_1 + 3x_2 \to \max$, $x \in G^{0,1}$. Розв‘язок $x^{0,1} = \left(\frac{6}{5}; \frac{16}{5}\right)$, $Z\left(x^{0,1}\right) = \frac{54}{5}$.

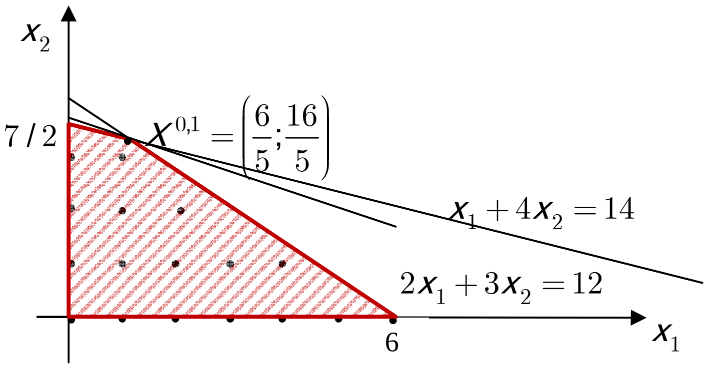

Рис.10.1

Обчислюємо межу $\xi^{0,1} = Z\left(x^{0,1}\right) = \frac{54}{5}$.

Проводимо розгалуження множини $G^{0,1}$:

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

$$G^{1,1} = \left\{x, x \in G^{0,1}, x_1 \le \left[\frac{6}{5}\right] = 1\right\},$$

$$G^{1,2} = \left\{x, x \in G^{0,1}, x_1 \ge \left[\frac{6}{5}\right] + 1 = 2\right\}$$

1 крок. Розв‘язуємо допоміжні ЗЛП (рис. 10.2)

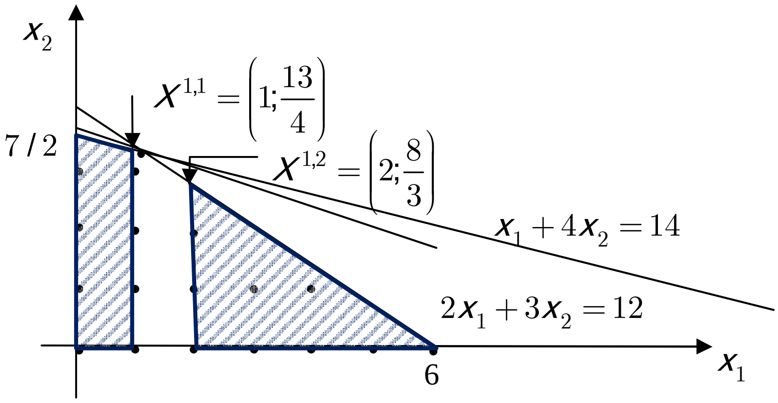

Рис.10.2

$Z\left(x\right)=x_1+3x_2 \to \max,\ x\in G^{1,1}$,

$Z\left(x\right)=x_1+3x_2 \to \max,\ x\in G^{1,2}$.

Їх розв'язки:

$x^{1,1}=\left(1,\frac{13}{4}\right),\ Z\left(x^{1,1}\right)=\frac{43}{4},\ x^{1,2}=\left(2,\frac{8}{3}\right),\ Z\left(x^{1,2}\right)=10$.

Обчислюємо межі $\xi^{1,1}=Z\left(x^{1,1}\right)=\frac{43}{4},\ \xi^{1,2}=Z\left(x^{1,2}\right)=10$.

Продовжуємо розгалуження множини $G\left(1;1\right)$:

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

$G^{2,1}=\left\{x, x\in G^{1,1}, x_2\le\left[\frac{13}{4}\right]=3\right\}$,

$G^{2,2}=\left\{x, x\in G^{1,1}, x_1\ge\left[\frac{13}{4}\right]+1=4\right\}$.

Покладаємо $G^{2,3}=G^{1,2}$.

2 крок. Розв'язуємо допоміжні ЗЛП.

$Z\left(x\right)=x_1+3x_2 \to \max,\ x\in G^{2,1}$,

$Z\left(x\right)=x_1+3x_2 \to \max,\ x\in G^{2,2}$.

Розв'язок першої задачі має вигляд $x^{2,1}=\left(1,3\right),\ Z\left(x^{2,1}\right)=10$ (рис.10.3). Друга задача розв'язку немає, оскільки множина $G^{2,2}$ порожня.

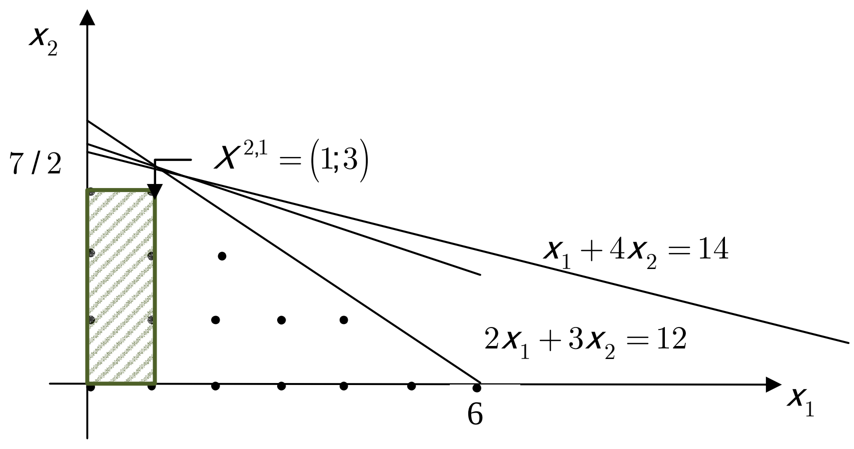

Рис.10.3

Аналізуючи розв'язки ЗЛП з областями $G^{2,1}, G^{2,2}, G^{2,3}$ та використовуючи критерій оптимальності, приходимо до висновку, що розв'язок ЦЗЛП $x^*=\left(1;3\right), Z\left(x^*\right)=10$. Схема розв'язання (рис.10.4):

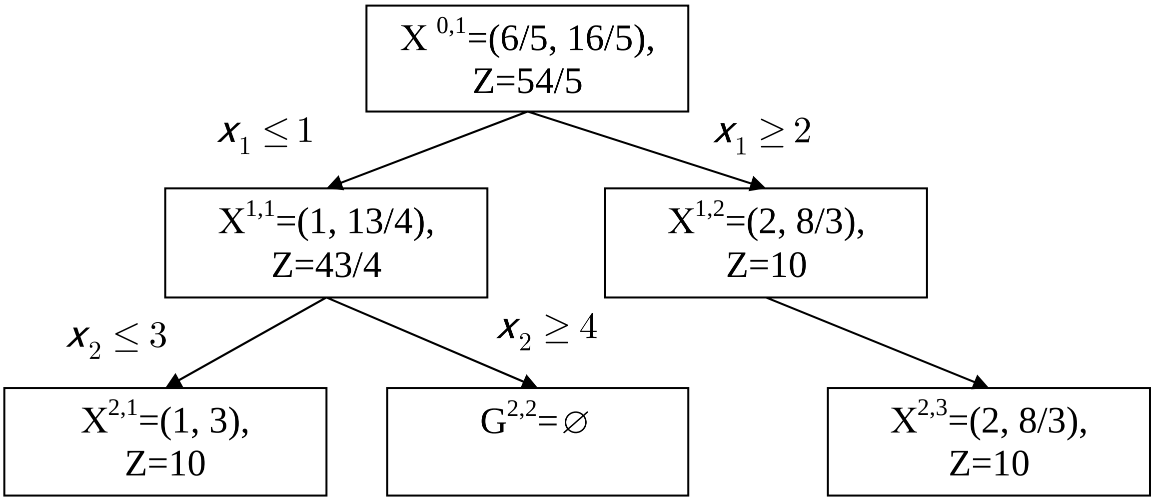

Рис. 10.4

10.2. Задача про оптимальні призначення. Угорський метод.

Угорський алгоритм розроблений і опублікований Харолдом Куном (1955), оснований на попередніх роботах угорський математиків Кьоніга і Егерварі і тому автором був названий угорським.

Математична модель задачі про оптимальні призначення має вигляд

$$Z(x) = \sum_{i \in I} \sum_{j \in J} c_{ij} x_{ij} \to \min \tag{10.5}$$

$$\sum_{i \in I} x_{ij} = 1,\ j \in J, \tag{10.6}$$

$$\sum_{j \in J} x_{ij} = 1,\ i \in I, \tag{10.7}$$

$$x_{ij} \in \{0,1\};\ i \in I, j \in J, I = \{1,\ldots,n\}, J = \{1,\ldots,n\} \tag{10.8}$$

де c_{ij} - витрати на призначення i-го виконавця на j-ту роботу.

ЗЛП, в яких на змінні накладаються умови (10.8) називаються задачами з булевими змінними. Тому, задача про призначення є задачею з булевими змінними. Якщо умови (10.8) замінити умовами $x_{ij} \ge 0;\ i \in I, j \in J$, то задачу про призначення можна розглядати як транспортну задачу з обсягами виробництва $a_i = 1, i \in I$ і обсягами споживання $b_j = 1, j \in J$. Отже, її можна розв‘язувати методом потенціалів. Проте, специфіка цієї задачі дозволяє її розв‘язати більш простими методами.

Матриці $C = (c_{ij})$ і $D = (d_{ij}),\quad i \in I, j \in J$ будемо називати *еквівалентними*, якщо елементи однієї з них можна одержати з елементів іншої додаванням деяких певних чисел до кожного рядка і кожного стовпця.

Тобто матриці C і D еквівалентні, якщо

$$d_{ij} = c_{ij} + u_i + v_j, \quad i \in I, j \in J. \tag{10.9}$$

Приміром, матриці

$$C = \begin{pmatrix} 1 & 2 & 5 \\ 0 & -1 & 4 \\ 3 & -5 & 0 \end{pmatrix} \quad \text{та} \quad D = \begin{pmatrix} 5 & 10 & 10 \\ 1 & 4 & 6 \\ 3 & -1 & 1 \end{pmatrix}$$

еквівалентні, оскільки матрицю D одержують з матриці C додаванням до рядків чисел 3, 0, -1, а до стовпців чисел 1, 5, 2.

Теорема 10.1. Оптимальні розв‘язки задач про оптимальні призначення з еквівалентними матрицями витрат співпадають.

$\triangle$ Нехай матриця $C = (c_{ij})$ еквівалентна матриці $D = (d_{ij}), i \in I, j \in J$ і $(1, j_1), ..., (n, j_n)$ - оптимальні призначення у задачі з матрицею C, тобто i-ий виконавець призначається на виконання роботи j_i, $i \in I, j_i \in J$. Припустимо, що це призначення не є оптимальним у задачі з матрицею витрат D. Припустимо, що $(1, j_1'), ..., (n, j_n')$ - оптимальне призначення в задачі з матрицею D, причому:

$$\sum_{i \in I} d_{ij_i'} < \sum_{i \in I} d_{ij_i}.$$

Тоді, враховуючи (10.9) маємо

$$\sum_{i \in I} \left(c_{ij_i'} + u_i + v_{j_i'}\right) < \sum_{i \in I} \left(c_{ij} + u_i + v_{j_i}\right)$$

або

$$\sum_{i \in I} \left(c_{ij_i'} + v_{j_i'}\right) < \sum_{i \in I} \left(c_{ij} + v_{j_i}\right)$$

Але $\sum_{i \in I} v_{j_i'} = \sum_{i \in I} v_{j_i}$ оскільки в обох випадках це сума чисел, що додаються до стовпців матриці C. Отже, маємо

$$\sum_{i \in I} c_{ij_i'} < \sum_{i \in I} c_{ij},$$

що суперечить оптимальності призначення $(1, j_1), ..., (n, j_n)$ для задачі з матрицею C.

Оскільки властивість еквівалентності матриць є взаємною, то оптимальне призначення матриці витрат C є оптимальним і для матриці витрат D.▲

Задача про призначення полягає в знаходженні таких u_i і v_j, $i \in I, j \in J$, що

$$c_{ij} \geq u_i + v_j.$$

Це можна зробити за допомогою операції зведення для матриці витрат $C=\left(c_{ij}\right)$. Для цього:

1) В кожному рядку шукаємо мінімальний елемент і віднімаємо його від усіх елементів рядка. Одержуємо матрицю витрат, зведену по рядках.

2) В кожному стовпці віднімаємо мінімальний елемент від усіх елементів стовпця. Одержуємо зведену матрицю.

Оптимальне призначення для одержаної зведеної матриці зберігається.

Нехай $\left(1,j_1\right),...,\left(n,j_n\right)$ - деяке призначення. Позначимо

$$C_{j_1,j_2,...,j_n}=c_{1j_1}+...+c_{1j_n} \text{ -}$$

витрати на призначення по рядках.

Тоді для зведеної матриці

$$\tilde{C}_{j_1,j_2,...,j_n}=C_{j_1,j_2,...,j_n}-\sum_{i\in I}u_i-\sum_{j\in J}v_j .$$

Позначимо $S=\sum_{i\in I}u_i+\sum_{j\in J}v_j$.

Очевидно, якщо знайдуться такі $j_1,...,j_n$ і такі u_i, v_j, $i\in I, j\in J$, що

$$C_{j_1,j_2,...,j_n}=\sum_{i=1}^{n}u_i+\sum_{j=1}^{n}v_j ,$$

то витрати на призначення будуть мінімальними, а величина S - максимальною. При цьому, для зведеної матриці

$$\tilde{C}_{j_1,j_2,...,j_n}=C_{j_1,j_2,...,j_n}-\sum_{i=1}^{n}u_i-\sum_{j=1}^{n}v_j=0.$$

Остання рівність є критерієм оптимального призначення. Тобто, якщо для зведеної матриці можна зробити призначення з нульовою сумою витрат, то це призначення буде оптимальним.

Таким чином, задачі про призначення ставиться у відповідність двоїста задача:

$$\sum_{i\in I}u_i+\sum_{j\in j}v_j\rightarrow \max$$

$$u_i+v_j\le c_{ij}, i\in I, j\in J,$$

причому для оптимальних розв'язків $X^*=\left(x_{ij}^*\right)$ і $Y^*=\left(u_1^*,...,u_n^*,v_1^*,...,v_n^*\right)$ виконується умова

$$\sum_{i\in I}\sum_{j\in J}c_{ij}x_{ij}^*=\sum_{i\in I}u_i^*+\sum_{j\in J}v_j^* .$$

10.3. Алгоритм угорського методу.

1. Віднімаємо в кожному i-му рядку матриці витрат C мінімальний елемент цього рядка, $i \in I$. Одержуємо зведену по рядках матрицю.

2. Віднімаємо від кожного j-го стовпця одержаної матриці мінімальний елемент цього стовпця $j \in J$. Одержуємо зведену матрицю, кожний рядок і стовпець якої містить принаймні один нуль.

3. Проглядаємо послідовно рядки матриці витрат починаючи з першого. Якщо рядок має один нуль, то помічаємо його позначкою * і закреслюємо (позначкою $^\wedge$) усі нулі в цьому ж стовпці. Нуль вважають поміченим, якщо він має позначку *. Повторюємо дії, поки кожен рядок не буде мати непозначених нулів, або їх буде принаймні два.

4. Повторюємо дії пункту 3 для усіх стовпців матриці витрат. Усі позначені в п. 3, 4 нулі лінійно незалежні.

5. Дії пунктів 3,4 повторюємо послідовно, поки не отримаємо один з можливих випадків:

1) кожен рядок має призначення;

2) є принаймні два непомічених нулі в деяких рядках і деяких стовпцях матриці витрат;

3) немає непомічених нулів, але повне призначення не отримано (число позначених * нулів менше n).

6. У випадку 1) задача розв‘язана, $x_{ij}^* = 1$ і відповідають 0^*, решта елементів дорівнюють 0. Кінець алгоритму.

У випадку 2) помічаємо довільно один з непозначених нулів позначкою *, закреслюємо решту нулів у тому ж рядку і стовпці і повертаємось до пункту 3.

Якщо маємо випадок 3) переходимо до пункту 7.

7. Помічаємо позначкою # рядки, для яких не одержано призначення (немає 0^*). Такі рядки вважаємо поміченими, решту – непоміченими. Таку ж термінологію використовуємо і для стовпців матриці витрат.

8. Помічаємо позначкою # ще не помічені стовпці, які маю закреслений нуль ($0^\wedge$) у помічених рядках.

9. Помічаємо позначкою # ще не помічені рядки, які мають призначення (тобто 0^*) у помічених стовпцях.

10. Повторюємо дії пунктів 8 та 9 поки це можливо (більше не можна помітити рядки і стовпці).

11. Закреслюємо позначкою & непомічені рядки і помічені стовпці матриці витрат.

Дії 7-10 виконують для того, щоб закреслити усі нулі найменшою можливою кількістю рядків і стовпців.

12. Знаходимо мінімальний елемент незакресленої підматриці витрат, віднімаємо його від усіх елементів незакреслених рядків та додаємо до елементів усіх закреслених стовпців і переходимо до пункту 3. Усі позначки (* і ^) втрачають силу.

Нехай $C_1 = \left(c_{ij} - u_i - v_j\right)_{i\in I, j\in J}$ - зведена матриця призначень на першій ітерації, $\tilde{Z}\left(X^1\right) = \sum_{i\in I}\sum_{j\in J}\left(c_{ij} - u_i - v_j\right)x_{ij}$ - значення цільової функції для цієї матриці. Позначимо індексною множиною I_r закреслені рядки, а J_c - закреслені стовпці зведеної матриці призначень, h - мінімальний елемент незакресленої підматриці. Згідно з п.12 алгоритму число h віднімають від елементів рядків $I \setminus I_r$ і додають до елементів стовпців J_c.

Тоді нове значення цільової функції

$$\tilde{Z}\left(x^1\right) = \sum_{i\in I\setminus I_r}\sum_{j\in J_c}\left(c_{ij} - u_i - \cancel{d} - v_j + \cancel{d}\right)x_{ij} + \sum_{i\in I\setminus I_r}\sum_{j\in J\setminus J_c}\left(c_{ij} - u_i - d - v_j\right)x_{ij} +$$

$$+\sum_{i\in I_r}\sum_{j\in J_c}\left(c_{ij} - u_i - v_j + d\right)x_{ij} + \sum_{i\in I_r}\sum_{j\in J\setminus J_c}\left(c_{ij} - u_i - v_j\right)x_{ij} =$$

$$= \sum_{i\in I}\sum_{j\in J}\left(c_{ij} - u_i - v_j\right)x_{ij} - \sum_{i\in I\setminus I_r}\sum_{j\in J\setminus J_c} dx_{ij} + \sum_{i\in I_r}\sum_{j\in J_c} dx_{ij} =$$

$$= \tilde{Z}\left(x^2\right) - \sum_{i\in I\setminus I_r}\sum_{j\in J\setminus J_c} dx_{ij} + \sum_{i\in I_r}\sum_{j\in J_c} dx_{ij}.$$