Задача про призначення

Постановка задачі про призначення

Нехай для виконання n різних робіт виділяється n працівників різної кваліфікації. Відомі працевитрати c_{ij} (або продуктивність) кожного j-го працівника при виконанні кожної i-ї роботи. Необхідно розподілити робітників за видами роботи таким чином, щоб загальні витрати на виконання комплексу робіт були мінімальними (або загальна продуктивність максимальною) [2, 3, 10, 11, 14].

Виконаємо математичну постановку задачі про призначення. Введемо змінні x_{ij}; причому $x_{ij} = 1$, якщо i-та робота закріплена за j-м працівником, $x_{ij} = 0$ – у протилежному випадку.

Сформулюємо обмеження задачі:

– за кожною роботою може бути закріплений тільки один працівник:

$$\sum_{j}^{n} x_{ij} = 1 \quad (i = 1, 2, \ldots, n); \tag{8.1}$$

– кожний працівник може бути закріплений тільки за однією роботою:

$$\sum_{i}^{n} x_{ij} = 1 \quad (j = 1, 2, \ldots, n). \tag{8.2}$$

Цільова функція задачі про призначення (залежно від фізичної сутності c_{ij}):

$$C = \sum_{i}^{n} \sum_{j}^{n} c_{ij} x_{ij} \to \min(\max).$$

Ця задача в наведеній постановці є задачею лінійного цілочислового програмування. Задача про призначення є частковим випадком транспортної задачі та може бути розв'язана у транспортній таблиці методом потенціалів. Однак особливості задачі про призначення дозволяють розв'язати її більш простими методами [2, 3, 11, 14]. До таких особливостей можна віднести:

– транспортна таблиця є квадратною матрицею (кількість робіт дорівнює кількості працівників);

– сума перевезень по кожному рядку транспортної таблиці дорівнює одиниці (8.1), сума по кожному стовпчику таблиці також дорівнює одиниці (8.2).

– перевезення x_{ij} можуть набувати значення тільки нуль або одиниця (булівські змінні).

Розв'язання задачі про призначення здійснюється в табличному вигляді у матриці вартості.

Допустимим є такий розв'язок задачі, при якому в кожному з n стовпчиків і в кожному з рядків матриці вартості буде тільки один зайнятий елемент. *Оптимальний* розв'язок задачі про призначення – це один з допустимих розв'язків, при якому загальна вартість зайнятих елементів матриці буде мінімальною (максимальною).

Серед найбільш поширених методів розв'язання задачі про призначення слід назвати метод К. Мака та Угорський метод.

8.2. Розв'язання задачі про призначення методом Мака

Метод, що розроблений К. Маком, є найбільш простим та інтуїтивно спрямованим; розв'язання задачі про призначення – це логічний процес послідовного розподілу в матриці вартості мінімальних елементів кожного рядка між стовпчиками для досягнення мінімального значення загальної вартості.

Нижче наведено алгоритм Мака для розв'язання задачі про призначення [2, 11]. На початку необхідно виділити в кожному рядку матриці вартості по одному мінімальному елементу (у виділеній клітині матриці $x_{ij} = 1$, у всіх інших – $x_{ij} = 0$). Стовпчики, у яких немає виділених елементів, називаються вільними, а інші – зайнятими.

Крок 1. Якщо в матриці немає жодного вільного стовпчика, то отримане рішення є оптимальним, інакше необхідно перейти до *кроку 2*.

Крок 2. Позначити * (зірочкою) будь-який стовпчик, у якому більше одного виділеного елемента.

Крок 3. Для всіх *i*-х рядків, у яких є виділені елементи в позначених * стовпчиках, знайти різницю Δ_i між мінімальним елементом *i*-го рядка серед непозначених * стовпчиків – a_i та мінімальним елементом *i*-го рядка серед позначених * стовпчиків – b_i, тобто $\Delta_i = a_i - b_i$.

Крок 4. Серед знайдених на кроці 3 різниць знайти мінімальну $\Delta = \min\{\Delta_i\}$.

Крок 5. Перетворити поточну таблицю, додавши до всіх елементів усіх позначених * стовпчиків Δ. Непозначені стовпчики залишаються без змін.

Крок 6. Елемент a_i у непозначеному стовпчику, який використовувався для розрахунку мінімальної різниці Δ, виділити підкресленням як альтернативний (тепер він має ту саму вартість, що й елемент b_i у позначеному * стовпчику).

Крок 7. Якщо альтернативний елемент a_i, отриманий на кроці 6, розташований у зайнятому стовпчику, то позначити цей стовпчик * (зірочкою) і перейти до *кроку 3*, інакше – перейти до *кроку 8*.

Крок 8. Показати стрілочкою перенесення по рядку виділеного елементу у вільний стовпчик в клітину з альтернативним елементом.

Крок 9. Якщо стовпчик, з якого було перенесено виділений елемент, став вільним, необхідно перейти до *кроку 8*, інакше – перейти до *кроку 10*.

Крок 10. Перетворити матрицю, зробивши виділеними відповідні альтернативні елементи. Прибрати всі позначки * і підкреслення та перейти до *кроку 1*.

Умова задачі. Між пунктами *А* і *Б* курсують парні та непарні пасажирські поїзди. Кожен поїзд може обслуговуватися бригадою локомотивного депо *А* чи *Б*. Тривалість руху поїзда між пунктами 4 години. Мінімальний час перебування бригади в пункті обороту $T_{\min} = 2$ години. Потрібно таким чином закріпити локомотивні бригади за поїздами, щоб їх загальний простій у пунктах обороту був мінімальним. Розклад руху поїздів на ділянці *А–Б* задано в табл. 8.1.

Таблиця 8.1

Розклад руху поїздів між пунктами *А* та *Б*

Пункти	Час відправлення						
А	1	3	5	14	17	21	22
Б	1	8	10	12	19	20	23

Розв'язання. Побудуємо графік руху поїздів на ділянці (рис. 8.1).

Визначимо можливі витрати часу бригад локомотивного депо *А* в пункті обороту *Б*. Наприклад, поїзд № 1 прибуває в пункт *Б* о 05:00. Якщо бригада буде відправлена з поїздом № 2 о 01:00, то її простій складе 20 годин. Ця величина занесена в ліву верхню клітину табл. 8.2 (ліва частина). При визначенні часу простою необхідно враховувати обмеження за мінімальним часом перебування бригад у пунктах обороту. Так, якщо бригада, що прибула з поїздом № 3, буде відправлятися з поїздом № 4, то її простій у пункті *Б* складатиме не 1 годину, а 25 годин. Можливі простої бригад з *А* для всіх інших непарних поїздів заносяться в ліву частину табл. 8.2, яка заповнюється по рядках. Можливі витрати часу бригад локомотивного депо *Б* в пункті обороту *А* визначаються аналогічно і заносяться в праву частину табл. 8.2, яка заповнюється по стовпчиках.

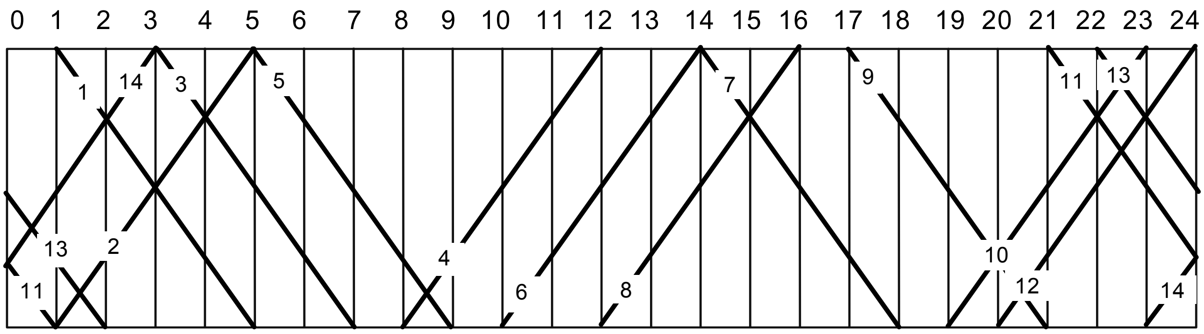

Рис. 8.1. Графік руху поїздів на ділянці *А–Б*

На основі порівняння значень простою бригад в пунктах обороту *А* та *Б* для однакових пар поїздів визначаються мінімальні простої бригад і заносяться в табл. 8.3. Пункт приписки бригади, для якої простій мінімальний, вказується у відповідній клітині табл. 8.3. Якщо простої бригад депо *А* та *Б* в пунктах обороту однакові, то у табл. 8.3 вказують «АБ».

На основі отриманої матриці можливих простоїв (табл. 8.3) з використанням алгоритму Мака необхідно знайти таке закріплення бригад за поїздами, яке забезпечує їх мінімальний простій у пунктах обороту. Приклад виконання розрахунків наведено на рис. 8.2.

Розглянемо порядок розв'язання детальніше. Виділимо в кожному рядку матриці вартості по одному мінімальному елементу (рис. 8.2, *ітерація 1*, *табл. 1*). Один із стовпчиків (наприклад, стовпчик 7), у якому кількість виділених елементів більше, ніж один, позначимо зірочкою (*). Для рядків 3 та 5, у яких є виділені елементи у поміченому (*) стовпчику 7, визначимо різницю між мінімальними елементами цих рядків та виділеними елементами: $\Delta_3 = 3 - 2 = 1$; $\Delta_5 = 3 - 2 = 1$ (рис. 8.2, *ітерація 1*, *табл. 1*). Серед отриманих різниць визначимо мінімальну $\Delta = 1$ (у нашому випадку, оскільки $\Delta_3 = \Delta_5 = 1$, то можна вибрати будь-який рядок, наприклад рядок 5).

Таблиця 8.2

Можливі витрати часу бригад у пунктах обороту

Витрати бригад з *А* в пункті обороту *Б*								Витрати бригад з *Б* в пункті обороту *А*							
№ поїзда	**2**	**4**	**6**	**8**	**10**	**12**	**14**	№ поїзда	**2**	**4**	**6**	**8**	**10**	**12**	**14**
1	20	3	5	7	14	15	18	**1**	20	13	11	9	2	25	22
3	18	25	3	5	12	13	16	**3**	22	15	13	11	4	3	24
5	16	23	25	3	10	11	14	**5**	24	17	15	13	6	5	2
7	7	14	16	18	25	2	5	**7**	9	2	24	22	15	14	11
9	4	11	13	15	22	23	2	**9**	12	5	3	25	18	17	14
11	24	7	9	11	18	19	22	**11**	16	9	7	5	22	21	18
13	23	6	8	10	17	18	21	**13**	17	10	8	6	23	22	19

Таблиця 8.3

Мінімальні витрати часу бригад у пунктах обороту

№ поїздів	2	4	6	8	10	12	14
1	20 АБ	3 А	5 А	7 А	2 Б	15 А	18 А
3	18 А	15 Б	3 А	5 А	4 Б	3 Б	16 А
5	16 А	17 Б	15 Б	3 А	6 Б	5 Б	2 Б
7	7 А	2 Б	16 А	18 А	15 Б	2 А	5 А
9	4 А	5 Б	3 Б	15 А	18 Б	17 Б	2 А
11	16 Б	7 А	7 Б	5 Б	18 А	19 А	18 Б
13	17 Б	6 А	8 АБ	6 Б	17 А	18 А	19 Б

Ітерація 1; таблиця 1

						*	
20	3	5	7	2	15	18	
18	15	3	5	4	3	16	
16	17	15	3	6	5	2	3-2=1
7	2	16	18	15	2	5	
4	5	3	15	18	17	2	3-2=1
16	7	7	5	18	19	18	
17	6	8	6	17	18	19	

Ітерація 1; таблиця 2

						*
20	3	5	7	2	15	19
18	15	3	5	4	3	17
16	17	15	3	6	5	3
7	2	16	18	15	2	6
4	5	3	15	18	17	3
16	7	7	5	18	19	19
17	6	8	6	17	18	20

Ітерація 2; таблиця 1

			*				
20	3	5	7	2	15	19	
18	15	3	5	4	3	17	
16	17	15	3	6	5	3	
7	2	16	18	15	2	6	
4	5	3	15	18	17	3	
16	7	7	5	18	19	19	7-5=2
17	6	8	6	17	18	20	6-6=0

Ітерація 2; таблиця 2

	*		*				
20	3	5	7	2	15	19	
18	15	3	5	4	3	17	
16	17	15	3	6	5	3	
7	2	16	18	15	2	6	2-2=0
4	5	3	15	18	17	3	
16	7	7	5	18	19	19	7-5=2
17	6	8	6	17	18	20	8-6=2

Ітерація 2; таблиця 3

	*		*		*		
20	3	5	7	2	15	19	
18	15	3	5	4	3	17	3-3=0
16	17	15	3	6	5	3	
7	2	16	18	15	2	6	6-2=4
4	5	3	15	18	17	3	
16	7	7	5	18	19	19	7-5=2
17	6	8	6	17	18	20	8-6=2

Ітерація 2; таблиця 4

	*	*	*		*		
20	3	5	7	2	15	19	
18	15	3	5	4	3	17	4-3=1
16	17	15	3	6	5	3	
7	2	16	18	15	2	6	6-2=4
4	5	3	15	18	17	3	3-3=0
16	7	7	5	18	19	19	16-5=11
17	6	8	6	17	18	20	17-6=11

Ітерація 2; таблиця 5

	*	*	*		*	*	
20	3	5	7	2	15	19	
18	15	3	5	4	3	17	4-3=1
16	17	15	3	6	5	3	5-3=2
7	2	16	18	15	2	6	6-2=4
4	5	3	15	18	17	3	4-3=1
16	7	7	5	18	19	19	16-5=11
17	6	8	6	17	18	20	17-6=11

Ітерація 2; таблиця 6

	*	*	*		*	*
20	4	6	8	2	16	20
18	16	4	6	4	4	18
16	18	16	4	6	6	4
7	3	17	19	15	3	7
4	6	4	16	18	18	4
16	8	8	6	18	20	20
17	7	9	7	17	19	21

Оптимальне рішення

20	4	6	8	2	16	20
18	16	4	6	4	4	18
16	18	16	4	6	6	4
7	3	17	19	15	3	7
4	6	4	16	18	18	4
16	8	8	6	18	20	20
17	7	9	7	17	19	21

Рис. 8.2. Приклад розв’язання задачі про призначення методом Мака

Перетворюємо матрицю, додавши до всіх елементів позначеного стовпця 7 мінімальну різницю $\Delta = 1$ (рис. 8.2, *ітерація 1; табл. 2*). У новій матриці підкреслимо як альтернативний елемент у клітині 5–3, який використано при визначенні мінімальної різниці. Тепер цей елемент має таке саме значення, як і виділений елемент 5–7 у рядку 5, тобто виділення елемента $\left(x_{ij} = 1\right)$ можна перенести (показано стрілкою) з клітини 5–7 в альтернативну клітину 5–3 (рис. 8.2, *ітерація 1; табл. 2*). Оскільки альтернативний елемент 5–3 знаходиться у вільному стовпчику, то в новій матриці цей елемент у рядку 5 стає виділеним замість елемента 5–7. Таким чином, отримуємо нову матрицю, у якій прибираємо всі позначки стовпчиків та підкреслення елементів (рис. 8.2, *ітерація 2; табл. 1*).

У новій матриці знову помічаємо (*) стовпець, у якому кількість виділених елементів більша за один (у нашому випадку – це стовпець 4). Далі діємо за алгоритмом Мака.

Зазначимо, що коли альтернативний елемент розташований у зайнятому стовпчику, то перенесення не відбувається, а цей стовпчик позначається (*), а потім виконується розрахунок різниць Δ_i з урахуванням нового позначеного (*) стовпчика. Так, для матриці, що наведена на рис. 8.2, *ітерація 2, табл. 2*, альтернативним є елемент у клітині 7–2, який розташований у зайнятому стовпчику 2; тому цей стовпчик помічається (*) і здійснюється новий розрахунок Δ_i .

У матриці, яка наведена на рис. 8.2, *ітерація 2, табл. 6*, після перенесення виділеного елемента з клітини 5–3 у клітину 5–1 стовпчик 3 стає вільним, однак у цьому стовпчику є альтернативний елемент у клітині 2–3, у який можна перенести виділений елемент з клітини 2–6. Оскільки при цьому стовпчик 6 стає вільним, то виділений елемент необхідно перенести в цей стовпчик у клітину 4–6 (там розташований альтернативний елемент) з клітини 4–2. У стовпчик 2, який стає вільним, виділений елемент переноситься в клітину 7–2 (альтернативний елемент) з клітини 7–4; при цьому в стовпчику 4 залишається один виділений елемент у клітині 6–4. Після перенесення виділених елементів отримуємо матрицю, у якій відсутні вільні стовпчики; при цьому в кожному рядку та в кожному стовпчику є тільки по одному виділеному елементу. Отже, ця матриця і є розв'язком задачі (рис. 8.2, *оптимальний розв'язок*).

Клітини, які є виділеними в матриці оптимального рішення, виділяємо також і в таблиці мінімально можливих витрат часу (табл. 8.4).

Проаналізуємо отриманий у табл. 8.4 результат. У рядку 1 виділеним є елемент «2Б» у стовпчику 10 – це означає, що бригада з пункту *Б* відправляється з поїздом № 10 та прямує в пункт *А*, де очікує 2 години та відправляється у пункт *Б* з поїздом № 1. Натомість, бригада з пункту *А* відправляється з поїздом № 3 (рядок 3, елемент «3А») та прямує у пункт *Б*, де після очікування тривалістю 3 години відправляється в пункт *А* з поїздом № 6.

Закріплення локомотивних бригад до поїздів, що виконане згідно з табл. 8.4, наведено на графіку руху поїздів (рис. 8.3).

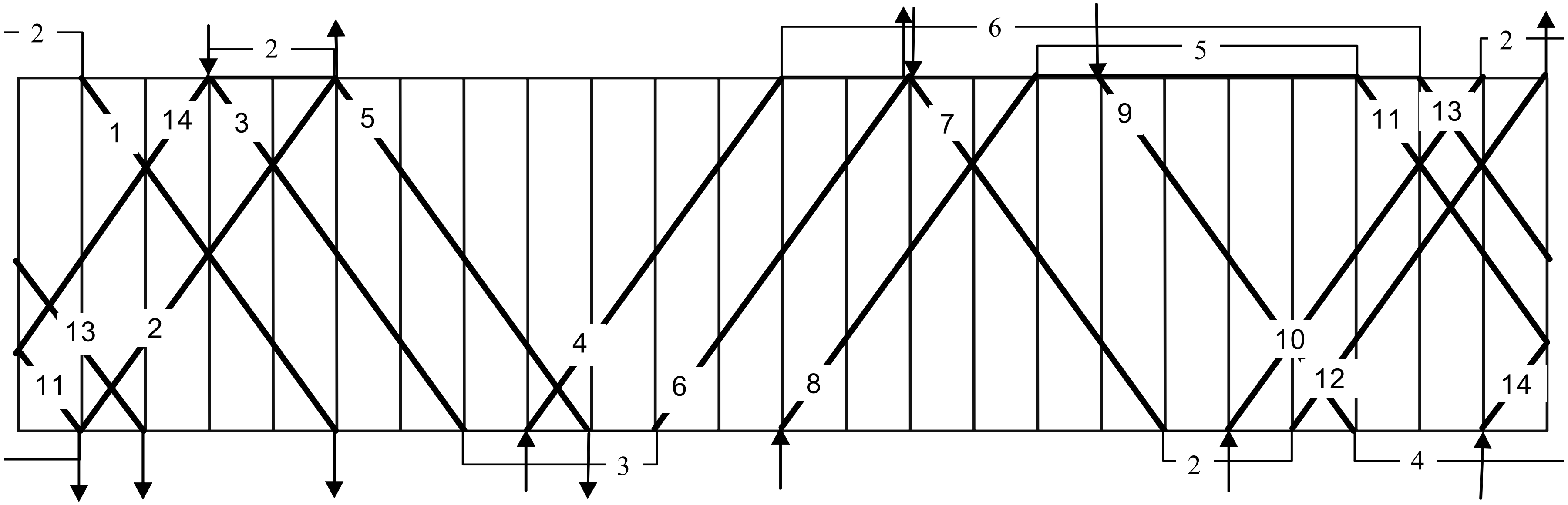

Рис. 8.3. Оптимальний графік закріплення локомотивних бригад до поїздів

Таблиця 8.4

Оптимальна підв'язка локомотивних бригад

№ поїздів	2	4	6	8	10	12	14
1	20 АБ	3 А	5 А	7 А	2 Б	15 А	18 А
3	18 А	15 Б	3 А	5 А	4 Б	3 Б	16 А
5	16 А	17 Б	15 Б	3 А	6 Б	5 Б	2 Б
7	7 А	2 Б	16 А	18 А	15 Б	2 А	5 А
9	4 А	5 Б	3 Б	15 А	18 Б	17 Б	2 А
11	16 Б	7 А	7 Б	5 Б	18 А	19 А	18 Б
13	17 Б	6 А	8 АБ	6 Б	17 А	18 А	19 Б

Таким чином, за допомогою розв'язання задачі про призначення методом Мака визначена оптимальна схема закріплення локомотивних бригад до пасажирських поїздів, що забезпечує мінімальний загальний час очікування бригадами в пунктах обороту, який становить: $C = 2 + 3 + 2 + 2 + 4 + 5 + 6 = 24$ години.

8.3. Розв'язання задачі про призначення Угорським методом

Алгоритм Угорського методу. Так званий Угорський метод [3, 5] часто дозволяє отримати рішення задачі про призначення за меншу кількість ітерацій порівняно з алгоритмом Мака. Цей алгоритм був розроблений і опублікований Гарольдом Куном у 1955 р. Сам Кун дав алгоритму назву «Угорський», тому що він значною мірою базувався на більш ранніх роботах двох угорських математиків – Денеша Кеніга і Ейгена Егервари. У 1957 р. професор Джеймс Манкрес показав, що цей алгоритм працює за поліноміальний час, тому в літературі він відомий не тільки як «Угорський», але і як «алгоритм Куна–Манкреса» або «алгоритм Манкреса». Втім, у 2006 р. з'ясувалося, що точно такий же алгоритм був винайдений за століття до Куна німецьким математиком Карлом Густавом Якобі. Справа в тому, що його робота «Про дослідження порядку системи звичайних довільних диференціальних рівнянь», надрукована посмертно у 1890 р., що містила, крім інших результатів, і поліноміальний алгоритм розв'язання задачі про призначення, була написана на латині, і її публікація пройшла непоміченою серед математиків.

Алгоритм Угорського методу складається з трьох етапів [3, 5, 14]:

Етап І:

Крок 1.1. Виконати редукцію рядків матриці вартості. Для цього потрібно знайти найменший елемент у кожному рядку матриці й відняти його від усіх елементів відповідного рядка.

Крок 1.2. Виконати редукцію стовпців матриці аналогічним чином.

Етап ІІ:

Крок 2.1. Знайти рядок з **одним** нульовим елементом і відмітити (виділити) цей елемент, інакше – перейти до *кроку 2.4*.

Крок 2.2. Викреслити всі нульові елементи стовпця, у якому розташований відмічений елемент.

Крок 2.3. Виконувати *кроки 2.1* та *2.2* до тих пір, поки продовження цієї процедури виявиться неможливим. Якщо в кожному рядку та стовпці буде по одному відміченому елементу, то отриманий розв'язок є оптимальним, інакше – перейти до *кроку 2.4*.

Крок 2.4. Знайти стовпець з **одним** нульовим елементом і відмітити (виділити) цей елемент, інакше – перейти до *кроку 2.7*.

Крок 2.5. Викреслити всі нульові елементи рядка, у якому розташований відмічений елемент.

Крок 2.6. Виконувати *кроки 2.4* та *2.5* до тих пір, поки продовження цієї процедури виявиться неможливим. Якщо в матриці всі нульові елементі виявилися відміченими або викресленими, а отриманий розв'язок є неприпустимим, то перейти до **етапу ІІІ**, інакше – перейти до *кроку 2.1*. Якщо розв'язок є припустимим, то воно є оптимальним.

Крок 2.7. Знайти рядок з мінімальною кількістю нульових елементів та повідмітити лише один з них, а інші – викреслити. Також викреслити всі нульові елементи стовпця, у якому розташований відмічений елемент, і перейти до *кроку 2.1*.

Етап ІІІ:

Крок 3.1. Визначити кількість нульових елементів у кожному рядку та стовпці, які не викреслені.

Крок 3.2. Викреслити рядок або стовпець з максимальною кількістю нульових елементів. Якщо в матриці викреслені усі нульові елементи, то перейти до *кроку 3.3*, інакше – до *кроку 3.1*.

Крок 3.3. Від усіх невикреслених елементів відняти мінімальний невикреслений елемент та додати його до кожного елемента, розташованого на перетині двох ліній. Перейти до **етапу II**.

База	Споживач			
	1	2	3	4
А	68	72	75	83
Б	56	60	58	63
В	38	40	35	45
Г	47	42	40	45

Рис. 8.4. Матриця відстаней між базами та споживачами

Умова задачі. Компанія має 4 збутові бази та 4 замовлення, які необхідно доставити різним споживачам. Ємність складського приміщення кожної бази цілком достатня для того, що б вмістити одне з цих замовлень. Відстань між базами та споживачами наведена у вигляді матриці вартості (рис. 8.4). Потрібно розподілити замовлення між базами таким чином, щоб загальна відстань перевезень була мінімальною.

Розв'язання. Розв'язання задачі про призначення Угорським методом починається з редукції (зменшення) рядків та стовпців матриці вартості. З цією метою виконаємо пошук мінімального значення в кожному рядку матриці (рис. 8.5, *а*). Віднімаючи отримані значення від усіх елементів відповідних рядків, отримаємо матрицю, що зображена на рис. 8.5, *б*.

а

База	Споживач				min
	1	2	3	4	
А	68	72	75	83	68
Б	56	60	58	63	56
В	38	40	35	45	35
Г	47	42	40	45	40

б

База	Споживач			
	1	2	3	4
А	0	4	7	15
Б	0	4	2	7
В	3	5	0	10
Г	7	2	0	5

Рис. 8.5. Редукція рядків матриці вартості

Редукція стовпців виконується так само, як і редукція рядків матриці, але всі дії виконуються лише із стовпцями матриці. Так, після пошуку мінімального значення в кожному стовпці матриці (див. рис. 8.6, *а*), отримані значення віднімемо від всіх елементів відповідних стовпців. У результаті виконаних дій отримаємо зменшену матрицю, яка зображена на рис. 8.6, *б*.

У матриці (рис. 8.6, *б*) виконаємо пошук рядка з **одним** нульовим елементом. Нехай це буде рядок, що відповідає базі «А». У цьому рядку відмічаємо нульовий елемент, який розташований у стовпці, що відповідає споживачу «1» (див.

рис. 8.7, *табл. 1*). Це означає, що замовлення закріплено між базою збуту «А» та клієнтом «1». У стовпці 1 викреслюємо всі нульові елементи (клітина А–1). У матриці є ще один рядок з **одним** нульовим елементом (рядок 3, клітина В–3). Відмічаємо елемент В–3, а в стовпці 3 викреслюємо всі нульові елементи (див. рис. 8.7, *табл. 2*).

а

База	Споживач			
	1	2	3	4
А	0	4	7	15
Б	0	4	2	7
В	3	5	0	10
Г	7	2	0	5
min	0	2	0	5

б

База	Споживач			
	1	2	3	4
А	0	2	7	10
Б	0	2	2	2
В	3	3	0	5
Г	7	0	0	0

Рис. 8.6. Редукція стовпців матриці

Таблиця 1

База	Споживач			
	1	2	3	4
А	**0**	2	7	10
Б	~~0~~	2	2	2
В	3	3	0	5
Г	7	0	0	0

Таблиця 2

База	Споживач			
	1	2	3	4
А	**0**	2	7	10
Б	~~0~~	2	2	2
В	3	3	**0**	5
Г	7	0	~~0~~	0

Таблиця 3

База	Споживач			
	1	2	3	4
А	**0**	2	7	10
Б	~~0~~	2	2	2
В	3	3	**0**	5
Г	7	**0**	~~0~~	~~0~~

Рис. 8.7. Пошук розв’язку (ітерація 1)

Більше рядків з одним нульовим елементом в матриці немає, але є стовпці – це стовпці 2 та 4. Відмічаємо один з елементів, наприклад елемент Г–2, при цьому в рядку Г викреслюємо всі нульові елементи (див. рис. 8.7, *табл. 3*). У результаті виконаних дій отримано розв’язок, який є недопустимим, оскільки не за всіма базами та споживачами закріплено по одному замовленню.

Згідно з алгоритмом Угорського методу у випадку отримання недопустимого розв’язку (етап III) потрібно виконати перетворення матриці. З цією метою послідовно викреслюють рядки або стовпці матриці, які мають найбільшу кількість нульових елементів. Вказану процедуру виконують до тих пір, поки в матриці не будуть викреслені всі нульові елементи. Так, рядок Г має 3 нульових елементи, а стовпці 1 та 3 – по 2 елементи. Отже, викреслюємо рядок Г (див. рис. 8.8, *табл. 1*). Далі послідовно викреслюємо стовпці 1 та 3 (див. рис. 8.8, *табл. 2*).

Серед усіх елементів, через які не проходять лінії викреслювання, знаходимо мінімальне значення, яке складає $v = 2$. Значення елементів, через які не проходять лінії, зменшуємо на величину v. Для елементів, які розташовані на перетині ліній викреслювання (клітини Г–1 та Г–3), значення збільшуємо на величину v. Значення інших елементів матриці залишаємо без змін. У результаті отримаємо нову матрицю, яка зображена на рис. 8.8, *табл. 3*.

Таблиця 1

База	Споживач			
	1	2	3	4
А	**0**	2	7	10
Б	0	2	2	2
В	3	3	**0**	5
Г	7	**0**	0	0

Таблиця 2

База	Споживач			
	1	2	3	4
А	**0**	2	7	10
Б	0	2	2	2
В	3	3	**0**	5
Г	7	**0**	0	0

Таблиця 3

База	Споживач			
	1	2	3	4
А	0	0	7	8
Б	0	0	2	0
В	3	1	0	3
Г	9	0	2	0

Рис. 8.8. Перетворення матриці

Пошук розв'язку з новою (перетвореною) матрицею виконується відповідно до алгоритму Угорського методу (етапи І та ІІ). Послідовність пошуку розв'язку наведена на рис. 8.9.

Отриманий розв'язок (рис. 8.9, *табл. 4*) є оптимальним, тому відмічені елементи необхідно перенести на початкову матрицю відстаней (рис. 8.10).

Таким чином, загальна відстань перевезень між базами та споживачами буде мінімальною і складає $L = 56 + 72 + 35 + 45 = 208$ км.

Задачу про призначення також можна розв'язати з використанням електронних таблиць MS Excel (див. дод. А).

Таблиця 1

База	Споживач			
	1	2	3	4
А	~~0~~	**0**	7	8
Б	0	~~0~~	2	0
В	3	1	0	3
Г	9	~~0~~	2	0

Таблиця 2

База	Споживач			
	1	2	3	4
А	~~0~~	**0**	7	8
Б	0	~~0~~	2	0
В	3	1	**0**	3
Г	9	~~0~~	2	0

Таблиця 3

База	Споживач			
	1	2	3	4
А	~~0~~	**0**	7	8
Б	**0**	~~0~~	2	~~0~~
В	3	1	**0**	3
Г	9	~~0~~	2	0

Таблиця 4

База	Споживач			
	1	2	3	4
А	~~0~~	**0**	7	8
Б	**0**	~~0~~	2	~~0~~
В	3	1	**0**	3
Г	9	~~0~~	2	**0**

Рис. 8.9. Пошук розв’язку (ітерація 2)

База	Споживач			
	1	2	3	4
А	68	**72**	75	83
Б	**56**	60	58	63
В	38	40	**35**	45
Г	47	42	40	**45**

Рис. 8.10. Оптимальний розв’язок

Контрольні запитання та завдання

1. Дати математичну постановку задачі про призначення.
2. Навести приклади задачі про призначення.
3. Сформулювати обмеження задачі про призначення.
4. Які особливості задачі про призначення на відміну від основної задачі лінійного програмування?
5. Який розв’язок задачі про призначення є допустимим?
6. Який розв’язок задачі про призначення є оптимальним?
7. Скільки базисних змінних повинно бути при розв’язанні задачі про призначення? Яких значень можуть набувати базисні змінні?

8. Скільки зайнятих клітин (виділених) повинно бути в кожному рядку таблиці при розв'язанні задачі про призначення методом Мака?
9. Які стовпчики в таблиці називають «зайнятими», а які вільними при розв'язанні задачі про призначення методом Мака?
10. Скільки вільних стовпчиків має бути в таблиці, що відповідає оптимальному розв'язку, при розв'язанні задачі про призначення методом Мака?
11. Серед яких рядків визначається мінімальна різниця при розв'язанні задачі про призначення методом Мака?
12. На скільки змінюються значення відмічених та невідмічених стовпчиків після перетворення таблиці при розв'язанні задачі про призначення методом Мака?
13. Що розуміють під «альтернативним елементом» при розв'язанні задачі про призначення методом Мака?
14. Які дії передбачає алгоритм Мака, якщо альтернативний елемент розташований у зайнятому стовпчику? У вільному стовпчику?
15. У скільки етапів розв'язується задача про призначення Угорським методом?
16. Що таке редукція матриці вартості в Угорському методі?
17. Які елементи матриці відмічаються на етапі ІІ Угорського методу?
18. Як здійснюється перетворення матриці вартості на етапі ІІІ Угорського методу?
19. У якому випадку отриманий за допомогою Угорського методу розв'язок задачі про призначення є оптимальним?
20. На рис. 8.11 задана матриця вартостей до задачі про призначення. Привести цю задачу до основної задачі лінійного програмування (записати відповідні обмеження і цільову функцію) та отримати її числовий розв'язок.

5	8
4	5

Рис. 8.11

21. Між пунктами *А* і *Б* курсують парні та непарні пасажирські поїзди. Кожен поїзд може обслуговуватися бригадою локомотивного депо *А* чи *Б*. Тривалість руху поїзда між пунктами 4 години. Мінімальний час перебування бригади в пункті обороту $T_{\min} = 1$ година. Потрібно таким чином закріпити локомотивні бригади за поїздами, щоб їх загальний простій у пунктах