

**НОВИЙ МЕТОД РОЗВ'ЯЗАННЯ ЗАДАЧІ ПРО ПРИЗНАЧЕННЯ**

**Abstract.** A new method and proper to him solving algorithms of assignment problem is offered, witch in case of a big dimension is more effective comparatively to the Hungarian method algorithm.

**Key words:** multitude of cortej permutations, Euclidean combinatorial optimization, assignment problem, Hungarian method.

**Анотація.** Запропоновано новий метод та відповідний йому алгоритм розв'язання задачі про призначення, який при великій розмірності задачі є більш ефективним порівняно з алгоритмом угорського методу.

**Ключові слова:** множина переставлень кортежів, евклідова комбінаторна оптимізація, задача про призначення, угорський метод.

**Аннотация.** Предложен новый метод и соответствующий ему алгоритм решения задачи о назначениях, который при большой размерности задачи более эффективный по сравнению с алгоритмом венгерского метода.

**Ключевые слова:** множество перестановок кортежей, эвклидова комбинаторная оптимизация, задача о назначениях, венгерский метод.

**1. Вступ**

Метою даної роботи є розроблення нового методу та відповідного йому алгоритму розв'язання задачі про призначення, який полягає у зведенні її до задачі оптимізації на множині переставлень кортежів та застосуванні методу потенціалів для транспортної задачі. Враховуючи одержані результати чисельної реалізації алгоритму, вважаємо, що викладений алгоритм при великій розмірності задачі є більш ефективним порівняно з алгоритмом угорського методу.

**2. Постановка задачі**

Задача про призначення: є  $n$  різних робіт і  $n$  різних механізмів (виконавців) для їх здійснення, причому кожен механізм може виконувати будь-яку роботу. Нехай  $c_{ij}$  – продуктивність  $i$ -го механізму при виконанні  $j$ -ої роботи,  $i, j \in J_n$ ,  $J_n = \{1, 2, \dots, n\}$ . Задача полягає у визначенні розподілу механізмів за роботами, при якому сумарна ефективність була б максимальною.

Математична модель задачі:

$$\sum_{i,j=1}^n c_{ij} \cdot x_{ij} \rightarrow \max ,$$

$$\sum_{i=1}^n x_{ij} = 1, \sum_{j=1}^n x_{ij} = 1, x_{ij} \in \{0, 1\}.$$

Уперше задача про призначення була розглянута в геометричній формі Гаспаром Монжем у 1784 році. Проте на початку ХХ століття була встановлена некоректність розв'язку Монжа. Наступні кроки в розв'язанні задачі про призначення зроблені Кенігом і Егерварі в першій третині ХХ століття. Кеніг і Егерварі розглядали цю задачу як задачу пошуку досконалого паросполучення мінімальної ваги у зваженому дводольному графі. Їх роботи стали основою для угорського методу, розробленого Куном у 50-х роках минулого століття [1]. У 1947 році Данциг запропонував симплекс-метод для розв'язання загальної задачі лінійного програмування, до якої легко зводиться задача про призначення. Задача про призначення, поставлена Данцигом і Фалкерсоном, може також

розглядатися як задача про максимальний потік мінімальної вартості. У 1961 році Басакер і Гоуен опублікували алгоритм для її розв'язання. Цей алгоритм для загальної задачі, як і алгоритм симплекс-методу, має експоненціальну складність, а для задачі про призначення – поліноміальну. Поліноміальний алгоритм для задачі про максимальний потік мінімальної вартості, заснований на алгоритмі Клейна, що був опублікований у 1967 році, вперше запропонований у 1987 році Гольбергом і Тар'яном [2].

Теоретичний аналіз складності алгоритмів показує, що алгоритми Куна й Басакера-Гоуена мають однакову теоретичну складність, меншу, ніж алгоритм Гольдберга-Тар'яна. Проте відповідь на питання про визначення найкращого алгоритму вимагає емпіричних досліджень [2].

У [3] викладено результати проведених емпіричних досліджень для угорського методу алгоритмів Басакера-Гоуена і Гольдберга-Тар'яна. Для алгоритму угорського методу емпірична складність  $T(n) = cn^{2.2}$ , для алгоритму Басакера-Гоуена –  $T(n) = cn^{3.05}$ , а для алгоритму Гольдберга-Тар'яна –  $T(n) = cn^{5.4}$ . Аналізуючи отримані результати, можна зробити висновок: найбільш швидким алгоритмом для розв'язання задачі про призначення є угорський метод.

### 3. Обґрунтування нового методу розв'язання задачі про призначення

Нехай задано матрицю ефективності  $(c_{ij})_{n \times n}$ . Розглянемо кортежі  $z_1 = (1, 0, \dots, 0)$ ,  $z_2 = (0, 1, \dots, 0)$ , ...,  $z_n = (0, 0, \dots, 1)$ . Кожен кортеж  $z_i$ ,  $i \in J_n$  містить  $n$  елементів, на  $i$ -тому місці – число 1, а усі інші його елементи дорівнюють 0. Множину  $\{(z_{i_1}, z_{i_2}, \dots, z_{i_n})\}$ , де  $i_j \neq i_k$ , якщо  $j \neq k$ ,  $j, k \in J_n$  переставлень кортежів, занурену в евклідов арифметичний простір  $R^{n^2}$ , позначимо  $ET_n^n$  [4]. Якщо позначити

$$C = (c_{11}, c_{12}, \dots, c_{1n}, c_{21}, c_{22}, \dots, c_{2n}, \dots, c_{n1}, c_{n2}, \dots, c_{nn}),$$

то задача про призначення має вигляд

$$\begin{aligned} C \cdot X &\rightarrow \max, \\ X &\in ET_n^n. \end{aligned} \tag{1}$$

Для матриці  $(c_{ij})_{n \times n}$  і переставлення кортежів  $\pi = (z_{i_1}, z_{i_2}, \dots, z_{i_n})$  уведемо на множині кортежів бінарне відношення  $\prec (c_{ij})$ , яке далі позначатимемо  $\prec$ , завжди маючи на увазі відповідну матрицю  $(c_{ij})_{n \times n}$ :

$$(z_i^k \prec (c_{ij}) z_j^l) \Leftrightarrow ((c_{ki} + c_{lj}) - (c_{kj} + c_{li}) \geq 0).$$

У позначенні  $z_i^k, z_j^l$  верхній індекс указує на порядковий номер кортежів  $z_i$  та  $z_j$  у переставленні кортежів.

Переставлення кортежів, у якому всі кортежі попарно знаходяться у відношенні  $\prec$ , позначатимемо  $\pi_\prec$ . Значення цільової функції (1) на переставленні  $\pi_\prec$  позначимо  $F(\pi_\prec)$ .

Властивість відношення  $\prec$ .

Після виконання однієї транспозиції довільної пари кортежів у переставленні  $\pi_{\prec} = (z_{i_1}, z_{i_2}, \dots, z_{i_n})$  значення цільової функції (1) не збільшиться.

Дійсно, нехай маємо переставлення  $\pi_{\prec}$ . Тоді

$$F(\pi_{\prec}) = c_{1i_1} + c_{2i_2} + \dots + c_{ki_k} + \dots + c_{li_l} + \dots + c_{ni_n}.$$

Виконаємо транспозицію кортежів  $z_{i_k}^k, z_{i_l}^l$ . Одержане переставлення кортежів позначимо

$\pi_{\prec T(k,l)}$ . Маємо

$$F(\pi_{\prec T(k,l)}) = c_{1i_1} + c_{2i_2} + \dots + c_{ki_l} + \dots + c_{li_k} + \dots + c_{ni_n},$$

$$F(\pi_{\prec}) - F(\pi_{\prec T(k,l)}) = (c_{ki_k} + c_{li_l}) - (c_{ki_l} + c_{li_k}) \geq 0.$$

Для матриці  $(c_{ij})_{n \times n}$  переставлення кортежів  $\pi_{\prec}$  визначається у загальному випадку неоднозначно. Причому для різних переставлень кортежів типу  $\pi_{\prec}$  значення цільової функції (1), власне кажучи, різні.

Якщо ж у переставленні  $\pi$  не всі кортежі попарно знаходяться у відношенні  $\prec$ , то, виконуючи по чергові транспозиції кортежів, які не знаходяться у відношенні  $\prec$ , на кожному кроці одержуватимемо переставлення кортежів з більшим, порівняно з попереднім, значенням цільової функції (1). Це безпосередньо впливає із означення відношення  $\prec$ . Звідси маємо таке твердження: переставлення кортежів, на якому цільова функція (1) задачі про призначення досягає глобального максимуму, належить до переставлень кортежів типу  $\pi_{\prec}$ .

Для знаходження оптимального розв'язку задачі про призначення спочатку будемо одне із переставлень кортежів  $\pi_{\prec}$ . Одержуємо розв'язок задачі, при якому цільова функція досягає локального максимуму. Далі до одержаного розв'язку, як до транспортної задачі з побудованим опорним планом, застосуємо метод потенціалів.

#### 4. Алгоритм

Попередні зауваження до алгоритму.

Вхідними даними для алгоритму є елементи матриці  $(c_{ij})_{n \times n}$ . При виборі найбільшого елемента у довільному рядку матриці  $(c_{ij})_{n \times n}$  рядок продивляємося зліва направо. Якщо найбільших елементів у рядку декілька, то вибираємо із них перший по порядку. Стовпці з вибраними найбільшими у рядках елементами називаємо зайнятими. При виборі найбільшого елемента кожного наступного рядка елементи зайнятих стовпців не враховуємо.

Алгоритм.

Крок 1. Знаходимо найбільший елемент у першому рядку. Ставимо на перше місце кортеж, номер якого збігається з номером стовпця, що містить вибраний елемент.

Крок 2. Знаходимо найбільший елемент у наступному рядку. Кортеж, номер якого збігається з номером стовпця і містить найбільший елемент рядка, що розглядається на даному кроці, ставимо на наступне місце у переставленні кортежів.

Крок 3. Перевіряємо по чергово, починаючи з кортежу, що стоїть на першому місці, чи знаходяться кортежі, які стоять на попередніх місцях у відношенні  $\prec$  з кортежем, який стоїть на останньому місці. Якщо так і число кортежів, упорядкованих за відношенням  $\prec$  а) менше  $n$ , то переходимо до кроку 2; б) рівне  $n$ , то переходимо до кроку 6. Якщо ні, переходимо до кроку 4.

Крок 4. Виконуємо транспозицію кортежу, що стоїть на останньому місці, з першим із кортежів, для якого не виконується відношення  $\prec$ . Переходимо до кроку 5.

Крок 5. Перевіряємо по чергово, чи знаходяться кортежі у відношенні  $\prec$  кожен з кожним. Якщо так і число кортежів, упорядкованих за відношенням  $\prec$  а) менше  $n$ , то переходимо до кроку 2; б) рівне  $n$ , то переходимо до кроку 6. Якщо ні, виконуємо транспозицію перших виявлених кортежів, для яких не виконується відношення  $\prec$ , і переходимо до кроку 5.

Крок 6. До одержаного на попередньому кроці розв'язку, як до опорного плану транспортної задачі, застосовуємо алгоритм методу потенціалів.

Крок 7. Завершення роботи алгоритму.

## 6. Результати чисельних експериментів

При проведенні чисельних експериментів генерувалися по 100 матриць  $(c_{ij})_{n \times n}$  з рівномірним розподілом елементів і така ж кількість матриць тієї ж розмірності з нормальним розподілом для кожного із значень  $n = 10, 20, \dots, 100, 150, 200, 250, \dots, 400, 500, 600, \dots, 900$ .

Отримані такі результати: на розмірності задач, де  $n < 200$ , запропонований алгоритм працює швидше за угорський метод для 40%–50% випадків; для  $200 < n < 600$  наведений алгоритм працює швидше за алгоритм угорського методу для різних значень  $n$  у 85–90% випадків; для  $n > 600$  кількість таких випадків досягає 95%–100%. Відношення часу роботи запропонованого алгоритму до часу роботи алгоритму угорського методу в задачах з однаковими вхідними даними для зазначених випадків у середньому дорівнює 1:3.

## 7. Висновки

Для розв'язання задачі про призначення, за умови великої її розмірності, доцільно використовувати викладений у даній роботі алгоритм, оскільки в такому випадку алгоритм викладеного методу її розв'язання має певну перевагу над угорським методом. Хоча обидва алгоритми мають однакові теоретичні оцінки складності –  $O(n^3)$ , але, як засвідчують чисельні експерименти, запропонований алгоритм є більш ефективним на задачах великої вимірності. До того ж зі зростанням розмірності задачі перевага запропонованого методу над угорським значно зростає. Цей факт можна пояснити тим, що в більшості випадків ще до кроку 6 алгоритму вже буде отримано оптимальний результат. Якщо ж результат після перших 5 кроків не є оптимальним, то в будь-якому разі він буде локальним оптимумом, і крок 6 дуже швидко призведе до пошуку оптимального розв'язку. Запропонований

алгоритм доцільно використовувати для реалізації на ЕОМ як складову частину програмних комплексів, що мають серед своїх завдань розв'язування задач про призначення, зокрема, задач великої розмірності.

#### **СПИСОК ЛІТЕРАТУРИ**

1. Kuhn H.W. On the Origin of the Hungarian Method / H.W. Kuhn // History of Mathematical Programming. – CWI-North-Holland, 1991. – P. 77 – 81.
2. Ху Т. Целочисленное программирование и потоки в сетях / Ху Т. – М.: Мир, 1974. – 520 с.
3. Шухман А.Е. Эмпирический анализ методов решения задачи о назначениях / А.Е. Шухман, Е.В. Шухман // Вызовы XXI века и образование: материалы всеросс. науч.-практ. конф. – Оренбург: ОГУ, 2006. – С. 227 – 230.
4. Гребенник И.В. Комбинаторное множество перестановок кортежей и его свойства / И.В. Гребенник // Радиоелектроніка. Інформатика. Управління. – 2005. – № 1. – С. 92 – 98.

*Стаття надійшла до редакції 02.07.2009*