Журнал обчислювальної та прикладної математики | 2022, № 1 | Journal of Numerical & Applied Mathematics

УДК 519.85
MSC 90B06, 90C08

TWO-STAGE TRANSPORTATION PROBLEM AND ITS TWO MODIFICATIONS

P. Stetsyuk, V. Stovba, O. Khomiak
V.M. Glushkov Institute of Cybernetics, the National Academy of Sciences of Ukraine, Kyiv, Ukraine, E-mail: {stetsyukp, vik.stovba, khomiak.olha}@gmail.com

ДВОЕТАПНА ТРАНСПОРТНА ЗАДАЧА ТА ДВІ ЇЇ МОДИФІКАЦІЇ

П. Стецюк, В. Стовба, О. Хом'як
Інститут кібернетики імені В.М. Глушкова, Національна академія наук України, Київ, Україна, E-mail: {stetsyukp, vik.stovba, khomiak.olha}@gmail.com

Abstract. In this paper, a mathematical model of an open two-stage transportation problem and its two modifications are considered. The first modification takes into account the upper bounds of transitional points capacities, the second takes into account the possibility of selection of the fixed number of transitional points, which is less than their total number. For all three cases the necessary and sufficient conditions of constraints feasibility are substantiated. The results of the computational experiments using gurobi and cplex solvers are presented.
Keywords: Two-stage transportation problem, linear programming problem, Boolean linear programming problem, AMPL, gurobi, cplex.

Анотація. У статті розглядається математична модель відкритої двоетапної транспортної задачі, а також двох її модифікацій. Перша модифікація враховує верхні границі на пропускні спроможності проміжних пунктів, друга враховує можливість вибору фіксованої кількості проміжних пунктів, меншої за їх загальну кількість. Для всіх трьох випадків обгрунтовано необхідні та достатні умови сумісності системи обмежень кожної задачі. Наведено результати обчислювальних експериментів з використанням солверів gurobi та cplex.
Ключові слова: Двоетапна транспортна задача, задача лінійного програмування, задача булевого лінійного програмування, AMPL, gurobi, cplex.

Робота виконана за підтримки CRDF Global (грант G-202102-68020).

Вступ

В класичній двоетапній транспортній задачі [1,2] вантаж (продукція) перевозиться від постачальників до споживачів тільки через проміжні пункти (див. рисунок 1). В якості проміжних пунктів можуть виступати посередницькі фірми та різного роду сховища (склади).

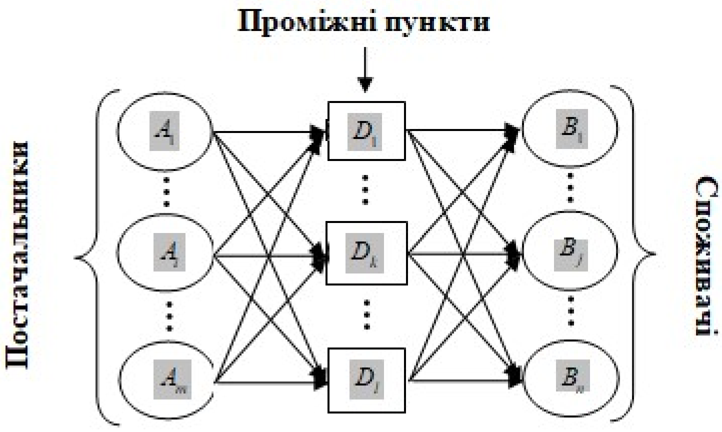

Рис. 1. Схема функціонування перевезень «постачальники — проміжні пункти — споживачі»

В статті розглядається відкрита двоетапна транспортна задача для знаходження найекономічнішого плану перевезення частини однорідної продукції від постачальників до споживачів та дві її модифікації [3]. Перша модифікація враховує обмеження на пропускні спроможності проміжних пунктів. Друга модифікація враховує як верхні межі на пропускні спроможності проміжних пунктів так і можливість вибору фіксованої кількості проміжних пунктів, яка є меншою за їх загальну кількість.

Математичні моделі двоетапної транспортної задачі та її першої модифікації представлені задачами лінійного програмування, а математична модель другої модифікації представлена у вигляді задачі булевого лінійного програмування. Розглядається два часткові випадки другої модифікації: випадок «одного споживача» та випадок «одного постачальника».

Послідовність викладення матеріалу статті є такою. В першому розділі наведено математичну модель відкритої двоетапної транспортної задачі та обґрунтувано необхідні та достатні умови сумісності системи обмежень цієї задачі. В другому розділі описано дві модифікації двоетапної транспортної задачі: перша враховує верхні границі на пропускні спроможності проміжних пунктів, друга — можливість вибору фіксованої кількості проміжних пунктів, меншої за їх загальну кількість. У третьому розділі наведено математичні моделі другої модифікації двоетапної транспортної задачі для двох часткових випадків: випадку «одного постачальника» та випадку «одного споживача». Четвертий розділ містить результати обчислювальних експериментів з тестування запропонованих модифікацій з використанням мови моделювання AMPL та солверів gurobi та cplex.

1. Двоетапна транспортна задача

Нехай в m пунктах постачання $A_1, \ldots, A_m$ є $a_1, \ldots, a_m$ одиниць продукції, частину якої потрібно перевезти до n споживачів $B_1, \ldots, B_n$, задовольнивши їх потреби $b_1, \ldots, b_n$. Для транспортування продукції від постачальників до споживачів можна задіяти l проміжних пунктів $D_1, \ldots, D_l$. Потрібно знайти оптимальний план транспортування продукції, де c_{ik} — витрати на перевезення одиниці продукції від постачальника A_i до проміжного пункту D_k, а c_{kj} — витрати на перевезення одиниці продукції від проміжного пункту D_k до споживача B_j.

Нехай $x = \{x_{ik}\}_{i=1,\ldots,m}^{k=1,\ldots,l}$, де x_{ik} — кількість одиниць продукції, яка перевозиться від постачальника A_i до пункту D_k; $y = \{y_{kj}\}_{k=1,\ldots,l}^{j=1,\ldots,n}$, де y_{kj} — кількість продукції від пункту D_k до споживача B_j.

Класична двоетапна транспортна задача, яка визначає найекономічніший план перевезення продукції від постачальників до споживачів через проміжні пункти, має такий вигляд: знайти

$$f_{xy}^* = f_{xy}(x^*, y^*) = \min_{x,y} \left\{ f(x,y) = \sum_{i=1}^{m} \sum_{k=1}^{l} c_{ik} x_{ik} + \sum_{k=1}^{l} \sum_{j=1}^{n} c_{kj} y_{kj} \right\} \tag{1}$$

за таких обмежень

$$\sum_{k=1}^{l} x_{ik} \le a_i, \quad i = \overline{1,m}, \tag{2}$$

$$\sum_{k=1}^{l} y_{kj} = b_j, \quad j = \overline{1,n}, \tag{3}$$

$$\sum_{i=1}^{m} x_{ik} - \sum_{j=1}^{n} y_{kj} = 0, \quad k = \overline{1,l}, \tag{4}$$

$$x_{ik} \ge 0, \;\; y_{kj} \ge 0, \;\; i = \overline{1,m}, \;\; k = \overline{1,l}, \;\; j = \overline{1,n}. \tag{5}$$

Задача (1)–(5) є задачею лінійного програмування, яка містить $(n+m) \times l$ неперервних змінних x та y, $m+n+l$ лінійних обмежень. Цільова функція (1) задає сумарні витрати на транспортування продукції від постачальників до споживачів. Обмеження (2) означають транспортування частини $a_1, \ldots, a_m$ одиниць продукції із пунктів постачання до проміжних пунктів, а обмеження (3) — що споживачам потрібно доставити необхідні об'єми $b_1, \ldots, b_n$ одиниць продукції з проміжних пунктів. Обмеження (4) задають умови на те, щоб уся продукція, яка приходить від постачальників до кожного проміжного пункту, була обов'язково відправлена споживачам.

Задача (1)–(5) відноситься до відкритих задач транспортного типу, тобто частина продукції залишається у постачальників, а інша частина продукції повинна бути доставлена споживачам, не залишаючи нічого в проміжних пунктах. Це диктує умови на сумісність системи обмежень (2)–(5) — системи лінійних рівностей та лінійних нерівностей. Справедливим є наступне твердження.

Лема 1. *Система обмежень* (2)–(5) *є сумісною тоді й лише тоді, коли виконується умова*

$$\sum_{i=1}^{m} a_i \geq \sum_{j=1}^{n} b_j. \tag{6}$$

Доведення. Доведемо необхідність. Нехай існують невід'ємні

$$\overline{x} = \left\{\overline{x}_{ik}, i = \overline{1,m}, k = \overline{1,l}\right\} \quad \text{та} \quad \overline{y} = \left\{\overline{y}_{kj}, k = \overline{1,l}, j = \overline{1,n}\right\},$$

які задовольняють систему обмежень (2)–(4). Тобто для $\overline{x} \geq 0$ та $\overline{y} \geq 0$ виконуються такі рівності та нерівності:

$$\sum_{k=1}^{l} \overline{x}_{ik} \leq a_i, \quad i = \overline{1,m}, \tag{7}$$

$$\sum_{k=1}^{l} \overline{y}_{kj} = b_j, \quad j = \overline{1,n}, \tag{8}$$

$$\sum_{i=1}^{m} \overline{x}_{ik} - \sum_{j=1}^{n} \overline{y}_{kj} = 0, \quad k = \overline{1,l}. \tag{9}$$

Просумувавши нерівності (7) за індексом $i = \overline{1,m}$ та рівності (8)–(9) за індексами $j = \overline{1,n}$ та $k = \overline{1,l}$ відповідно, маємо:

$$\sum_{i=1}^{m} \sum_{k=1}^{l} \overline{x}_{ik} \leq \sum_{i=1}^{m} a_i, \tag{10}$$

$$\sum_{j=1}^{n} \sum_{k=1}^{l} \overline{y}_{kj} = \sum_{j=1}^{n} b_j, \tag{11}$$

$$\sum_{k=1}^{l} \sum_{i=1}^{m} \overline{x}_{ik} - \sum_{k=1}^{l} \sum_{j=1}^{n} \overline{y}_{kj} = 0. \tag{12}$$

Використовуючи нерівність (10) та рівності (11)–(12), і, можливо, змінюючи порядок в індексах, отримуємо такий ланцюжок нерівностей:

$$\sum_{i=1}^{m} a_i \geq \sum_{k=1}^{l} \sum_{i=1}^{m} \overline{x}_{ik} = \sum_{k=1}^{l} \sum_{j=1}^{n} \overline{y}_{kj} = \sum_{j=1}^{n} b_j. \tag{13}$$

Звідси випливає, що $\sum_{i=1}^{m} a_i \geq \sum_{j=1}^{n} b_j$.

Доведемо достатність. Нехай виконується нерівність $\sum_{i=1}^{m} a_i \geq \sum_{j=1}^{n} b_j$. Розглянемо величини $\tilde{x}_{ik}$ та $\tilde{y}_{kj}$, $i = \overline{1,m}$, $k = \overline{1,l}$, $j = \overline{1,n}$, які визначаються так:

$$\tilde{x}_{ik} = \begin{cases} a_i(1-t), & \text{якщо } k=1 \\ 0, & \text{інакше} \end{cases} \quad \text{та} \quad \tilde{y}_{kj} = \begin{cases} b_j, & \text{якщо } k=1 \\ 0, & \text{інакше} \end{cases}, \tag{14}$$

де $t = 1 - \sum_{j=1}^{n} b_j / \sum_{i=1}^{m} a_i$, причому $0 < 1-t \leq 1$. Покажемо, що ці величини задовольняють обмеження задачі (1)–(5). Дійсно,

1. $\displaystyle\sum_{k=1}^{l} \tilde{x}_{ik} = \tilde{x}_{i1} = a_i(1-t) \leq a_i, \quad i = \overline{1,m};$
2. $\displaystyle\sum_{k=1}^{l} \tilde{y}_{kj} = \tilde{y}_{1j} = b_j, \quad j = \overline{1,n};$
3. $\displaystyle\sum_{i=1}^{m} \tilde{x}_{ik} - \sum_{j=1}^{n} \tilde{y}_{kj} = \sum_{i=1}^{m} \tilde{x}_{i1} - \sum_{j=1}^{n} \tilde{y}_{1j} = \sum_{i=1}^{m} a_i(1-t) - \sum_{j=1}^{n} b_j =$
   $\displaystyle = (1-t)\sum_{i=1}^{m} a_i - \sum_{j=1}^{n} b_j = \frac{\sum_{j=1}^{n} b_j}{\sum_{i=1}^{m} a_i} \sum_{i=1}^{m} a_i - \sum_{j=1}^{n} b_j = 0;$
4. $\tilde{x}_{ik} \geq 0$, $\tilde{y}_{kj} \geq 0$, оскільки $a_i > 0$, $b_j > 0$, $1-t > 0$.

Отже, система обмежень (2)–(5) має допустиму точку (14). □

Якщо $\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$, то задача (1)–(5) відноситься до закритих задач транспортного типу [1, 2, 4], тобто вся продукція постачальників повинна бути доставлена споживачам, не залишаючи нічого в проміжних пунктах. Для двоетапної транспортної задачі закритого типу нерівність (2) перетворюється на рівність

$$\sum_{k=1}^{l} x_{ik} = a_i, \quad i = \overline{1,m}, \tag{15}$$

і має місце такий наслідок.

Наслідок 1. *Система обмежень* (15), (3)–(5) *є сумісною тоді й лише тоді, коли виконується умова*

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j. \tag{16}$$

Умова (16) означає, що обмеження (15), (3)–(5) є лінійно залежними і одне з рівнянь може бути вилучено, причому довільне. Для задачі відкритого типу в цьому необхідності немає, оскільки додавання нової невід'ємної змінної до обмеження (2) означає лінійну незалежність обмежень (2)–(5).

2. Дві модифікації задачі (1)–(5)

Перша модифікація враховує верхні границі на пропускні спроможності проміжних пунктів, де $d_1^{up}, \ldots, d_l^{up}$ — максимальні пропускні спроможності пунктів $D_1, \ldots, D_l$. Двоетапна транспортна задача з обмеженнями на пропускні спроможності проміжних пунктів має такий вигляд: знайти

$$f_{xy}^* = f_{xy}(x^*, y^*) = \min_{x,y}\left\{ f(x,y) = \sum_{i=1}^{m}\sum_{k=1}^{l} c_{ik}x_{ik} + \sum_{k=1}^{l}\sum_{j=1}^{n} c_{kj}y_{kj} \right\} \tag{17}$$

за таких обмежень

$$\sum_{k=1}^{l} x_{ik} \le a_i, \quad i = \overline{1,m}, \tag{18}$$

$$\sum_{k=1}^{l} y_{kj} = b_j, \quad j = \overline{1,n}, \tag{19}$$

$$\sum_{i=1}^{m} x_{ik} - \sum_{j=1}^{n} y_{kj} = 0, \quad k = \overline{1,l}, \tag{20}$$

$$\sum_{i=1}^{m} x_{ik} \le d_k^{up}, \quad k = \overline{1,l}, \tag{21}$$

$$x_{ik} \ge 0, \;\; y_{kj} \ge 0, \;\; i = \overline{1,m}, \;\; k = \overline{1,l}, \;\; j = \overline{1,n}. \tag{22}$$

Задача (17)–(22) є задачею лінійного програмування та містить $(m+n)\times l$ змінних x, y, $m+n+2l$ лінійних обмежень. Тут цільова функція (17) та обмеження (18)–(20), (22) мають такий же зміст, як цільова функція та обмеження задачі (1)–(5). Обмеження (21) задають верхні межі на пропускні спроможності проміжних пунктів. Їх також можна записати так:

$$\sum_{j=1}^{n} y_{kj} \le d_k^{up}, \quad k = \overline{1,l}. \tag{23}$$

Задача (17)–(22) відноситься до відкритих задач транспортного типу, для якої справедливим є наступне твердження.

Лема 2. *Система обмежень* (18)–(22) *є сумісною тоді й лише тоді, коли виконуються умови*

$$\sum_{i=1}^{m} a_i \ge \sum_{j=1}^{n} b_j, \quad \sum_{j=1}^{n} b_j \le \sum_{k=1}^{l} d_k^{up}. \tag{24}$$

Доведення. Доведемо необхідність. Справедливість першої умови теореми випливає з необхідності леми 1. Покажемо справедливість другої умови. Для цього просумуємо рівність (19) за індексом $j = \overline{1,n}$, рівність (20) за індексом $k = \overline{1,l}$ та нерівність (21) за індексом $k = \overline{1,l}$. Користуючись цими рівностями та нерівностями, маємо такий ланцюжок нерівностей:

$$\sum_{k=1}^{l}\sum_{i=1}^{m} \overline{x}_{ik} = \sum_{k=1}^{l}\sum_{j=1}^{n} \overline{y}_{kj} = \sum_{j=1}^{n} b_j \le \sum_{k=1}^{l} d_k^{up}, \tag{25}$$

звідки випливає справедливість другої умови леми.

Доведемо достатність. Нехай виконуються нерівності (24). Визначимо індекс k^* таким чином:

$$\sum_{k=1}^{k^*} d_k^{up} \le \sum_{j=1}^{n} b_j, \quad \sum_{k=1}^{k^*+1} d_k^{up} > \sum_{j=1}^{n} b_j. \tag{26}$$

Розглянемо величини $\tilde{x}_{ik}$ та $\tilde{y}_{kj}$, $i = \overline{1,m}$, $k = \overline{1,l}$, $j = \overline{1,n}$, визначені так:

$$\tilde{x}_{ik} = \begin{cases} a_i t_1 t_2, & k \le k^* \\ a_i t_1 t^*, & k = k^* + 1 \\ 0, & \text{інакше} \end{cases}, \quad \tilde{y}_{kj} = \begin{cases} b_j t_3, & k \le k^* \\ b_j t^*, & k = k^* + 1 \\ 0, & \text{інакше} \end{cases}, \tag{27}$$

де $t_1 = \dfrac{\sum_{j=1}^n b_j}{\sum_{i=1}^m a_i}$, $t_2 = \dfrac{d_k^{up}}{\sum_{j=1}^n b_j}$, $t_3 = \dfrac{d_k^{up}}{\sum_{j=1}^n b_j}$, $t^* = \dfrac{\sum_{j=1}^n b_j - \sum_{k=1}^{k^*} d_k^{up}}{\sum_{j=1}^n b_j}$.

Перевіримо, що величини $\tilde{x}_{ik}$ та $\tilde{y}_{kj}$ задовольняють обмеження (18)–(22).

1. $\displaystyle\sum_{k=1}^{l} \tilde{x}_{ik} = \sum_{k=1}^{k^*} a_i t_1 t_2 + a_i t_1 t^* = a_i t_1 \left(\frac{\sum_{k=1}^{k^*} d_k^{up}}{\sum_{j=1}^n b_j} + \frac{\sum_{j=1}^n b_j - \sum_{k=1}^{k^*} d_k^{up}}{\sum_{j=1}^n b_j} \right) =$
   $= a_i t_1 \le a_i,\ i = \overline{1,m}$;
2. $\displaystyle\sum_{k=1}^{l} \tilde{y}_{kj} = \sum_{k=1}^{k^*} b_j t_3 + b_j t^* = b_j \left(\frac{\sum_{k=1}^{k^*} d_k^{up}}{\sum_{j=1}^n b_j} + \frac{\sum_{j=1}^n b_j - \sum_{k=1}^{k^*} d_k^{up}}{\sum_{j=1}^n b_j} \right) =$
   $= b_j, j = \overline{1,n}$;
3. При $k \le k^*$ маємо
   $\displaystyle\sum_{i=1}^{m} \tilde{x}_{ik} - \sum_{j=1}^{n} \tilde{y}_{kj} = \sum_{i=1}^{m} a_i t_1 t_2 - \sum_{j=1}^{n} b_j t_3 = \frac{\sum_{j=1}^n b_j}{\sum_{i=1}^m a_i} \frac{d_k^{up}}{\sum_{j=1}^n b_j} \sum_{i=1}^{m} a_i -$
   $\displaystyle- \frac{d_k^{up}}{\sum_{j=1}^n b_j} \sum_{j=1}^{n} b_j = 0,$
   а при $k = k^* + 1$
   $\displaystyle\sum_{i=1}^{m} \tilde{x}_{i,k^*+1} - \sum_{j=1}^{n} \tilde{y}_{k^*+1,j} = \sum_{i=1}^{m} a_i t_1 t^* - \sum_{j=1}^{n} b_j t^* =$
   $\displaystyle= t^* \left(\frac{\sum_{j=1}^n b_j}{\sum_{i=1}^m a_i} \sum_{i=1}^{m} a_i - \sum_{j=1}^{n} b_j \right) = 0;$
4. При $k \le k^*$ маємо
   $\displaystyle\sum_{i=1}^{m} \tilde{x}_{ik} = \sum_{i=1}^{m} \frac{a_i t_1 d_k^{up}}{\sum_{i=1}^m a_i t_1} = d_k^{up},$
   а при $k = k^* + 1$
   $\displaystyle\sum_{i=1}^{m} \tilde{x}_{i,k^*+1} = \sum_{i=1}^{m} \frac{a_i t_1 \left(\sum_{j=1}^n b_j - \sum_{k=1}^{k^*} d_k^{up} \right)}{\sum_{i=1}^m a_i t_1} = \sum_{j=1}^{n} b_j - \sum_{k=1}^{k^*} d_k^{up} \le d_{k^*+1}^{up};$
5. Невід'ємність величин $\tilde{x}_{ik}$ та $\tilde{y}_{kj}$ випливає з їх побудови.

Отже система обмежень (18)–(22) має допустиму точку (27). □

Якщо $\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$, то задача (17)–(22) відноситься до закритих задач транспортного типу, при цьому нерівність (18) замінюється рівністю (15), і має місце такий наслідок.

Наслідок 2. *Система обмежень* (15), (18)–(22) *є сумісною тоді й лише тоді, коли виконується умова*

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j \leq \sum_{k=1}^{l} d_k^{up}.$$

Друга модифікація двоетапної транспортної задачі враховує можливість вибору d — фіксованої кількості проміжних пунктів [5], яка є меншою за їх загальну кількість l. Нехай $z = \{z_k\}_{k=1,\ldots,l}$ — булеві змінні, де z_k рівна одиниці, якщо проміжний пункт використовується, та нулю в протилежному випадку. Двоетапна транспортна задача з заданою кількістю проміжних пунктів d $(1 < d < l)$ формулюється так: знайти

$$f_{xyz}^* = f_{xyz}\left(x^*, y^*\right) = \min_{x,y,z} \left\{ f(x,y) = \sum_{i=1}^{m} \sum_{k=1}^{l} c_{ik} x_{ik} + \sum_{k=1}^{l} \sum_{j=1}^{n} c_{kj} y_{kj} \right\} \tag{28}$$

за таких обмежень

$$\sum_{k=1}^{l} x_{ik} \leq a_i, \quad i = \overline{1,m}, \tag{29}$$

$$\sum_{k=1}^{l} y_{kj} = b_j, \quad j = \overline{1,n}, \tag{30}$$

$$\sum_{i=1}^{m} x_{ik} - \sum_{j=1}^{n} y_{kj} = 0, \quad k = \overline{1,l}, \tag{31}$$

$$\sum_{i=1}^{m} x_{ik} \leq d_k^{up} z_k, \quad k = \overline{1,l}, \tag{32}$$

$$\sum_{k=1}^{l} z_k = d, \tag{33}$$

$$x_{ik} \geq 0,\ y_{kj} \geq 0,\ z_k = 0 \vee 1, \quad i = \overline{1,m},\ k = \overline{1,l},\ j = \overline{1,n}. \tag{34}$$

Задача (28)–(34) є задачею булевого лінійного програмування: $(m+n) \times l$ неперервних змінних x, y та l булевих змінних z. Задача містить $m+n+2l+1$ обмежень. Обмеження (33) означає, що задіяно рівно d проміжних пунктів, а обмеження (32) задають для них верхні межі на пропускні спроможності. Зауважимо, що обмеження (32) можна записати у такому вигляді:

$$\sum_{j=1}^{n} y_{kj} \leq d_k^{up} z_k, \quad k = \overline{1,l}. \tag{35}$$

Задача (28)–(34) відноситься до відкритих задач транспортного типу, для якої справедливим є наступне твердження.

Лема 3. *Система обмежень* (29)–(34) *є сумісною тоді й лише тоді, коли виконуються умови*

$$\sum_{i=1}^{m} a_i \geq \sum_{j=1}^{n} b_j, \ \sum_{j=1}^{n} b_j \leq \sum_{k=1}^{d} d_k^{up}, \tag{36}$$

де $d_1^{up} \geq d_2^{up} \geq \cdots \geq d_l^{up}$.

Доведення леми 3 аналогічно доведенню леми 2 для $l = d$. Також якщо $\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$, то задача (28)–(34) відноситься до закритих задач транспортного типу, при цьому нерівність (29) перетворюється на рівність (15), і має місце такий наслідок.

Наслідок 3. *Система обмежень* (15), (30)–(34) *є сумісною тоді й лише тоді, коли виконуються умови*

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j \leq \sum_{k=1}^{d} d_k^{up}.$$

3. Часткові випадки задачі (28)–(34)

Нижче наведемо математичні моделі другої модифікації двоетапної транспортної задачі для двох часткових випадків: випадок «одного споживача» та випадок «одного постачальника» (див. рисунок 2).

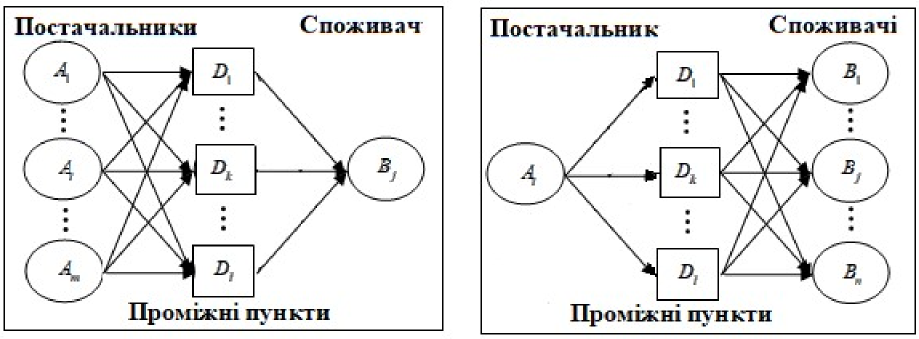

Рис. 2. Схеми функціонування перевезень «постачальники – проміжні пункти – споживач» та «постачальник – проміжні пункти – споживачі»

Для часткового випадку «одного споживача» друга модифікація двоетапної транспортної задачі значно спрощується і відповідна їй задача булевого лінійного програмування буде мати вигляд: знайти

$$f_{xyz}^{*} = f_{xyz}(x^*, y^*) = \min_{x,y,z} \left\{ f(x,y) = \sum_{i=1}^{m} \sum_{k=1}^{l} c_{ik} x_{ik} + \sum_{k=1}^{l} c_k y_k \right\} \tag{37}$$

за таких обмежень

$$\sum_{k=1}^{l} x_{ik} \le a_i, \quad i = \overline{1,m}, \tag{38}$$

$$\sum_{k=1}^{l} y_k = b, \tag{39}$$

$$\sum_{i=1}^{m} x_{ik} - y_k = 0, \quad k = \overline{1,l}, \tag{40}$$

$$y_k \le d_k^{up} z_k, \quad k = \overline{1,l}, \tag{41}$$

$$\sum_{k=1}^{l} z_k = d, \tag{42}$$

$$x_{ik} \ge 0, \quad z_k = 0 \vee 1, \quad i = \overline{1,m}, \quad k = \overline{1,l}. \tag{43}$$

Для часткового випадку «одного постачальника» друга модифікація двоетапної транспортної задачі формулюється за допомогою задачі булевого лінійного програмування: знайти

$$f_{xyz}^* = f_{xyz}(x^*, y^*) = \min_{x,y,z} \left\{ f(x,y) = \sum_{k=1}^{l} c_k x_k + \sum_{k=1}^{l} \sum_{j=1}^{n} c_{kj} y_{kj} \right\} \tag{44}$$

за таких обмежень

$$\sum_{k=1}^{l} x_k \le a, \tag{45}$$

$$\sum_{k=1}^{l} y_{kj} = b_j, \quad j = \overline{1,n}, \tag{46}$$

$$x_i - \sum_{j=1}^{n} y_{kj} = 0, \quad k = \overline{1,l}, \tag{47}$$

$$x_k \le d_k^{up} z_k, \quad k = \overline{1,l}, \tag{48}$$

$$\sum_{k=1}^{l} z_k = d, \tag{49}$$

$$y_{kj} \ge 0, \quad z_k = 0 \vee 1, \qquad k = \overline{1,l}, \quad j = \overline{1,n}. \tag{50}$$

Задачі (37)–(43) та (44)–(50) можна адаптувати під відомі методи кластеризації, такі як k-means [6], k-means++ [7], k-medians [8]. Для цього необхідно додати обмеження булевості на змінні x_{ik} та y_{kj}, встановити одиничними об'єми постачальників та споживачів, визначити коефіцієнти c_{ik}

та c_{kj} як квадрати евклідових відстаней між точками (для k-means) та евклідові відстані (для k-medians) між об'єктами, які ставляться у відповідність виробникам та споживачам. Відкритість задач (37)–(43) та (44)–(50) дозволяє розширити вказані методи кластеризації на випадок, де необхідно не враховувати певну кількість точок.

Обчислювальні експерименти

Для тестування модифікації двоетапної транспортної задачі з обмеженням на кількість проміжних пунктів було використано AMPL-коди для відповідних задач булевого програмування, орієнтованих на велику кількість постачальників, проміжних пунктів та споживачів. Для цього було розроблено AMPL-коди для двох тестових прикладів: тест I — з випадковими вартостями (див. додаток 1), тест II — з випадковими координатами (див. додаток 2). За їх допомогою досліджено ефективність сучасного програмного забезпечення з NEOS-сервера (відомих програм **gurobi** та **cplex**) для розв'язання тестових задач середніх розмірів (розглядаються декілька сотень постачальників, проміжних пунктів та споживачів).

У таблиці 1 наведено результати розрахунку для тестової задачі з такими розмірами: 100 постачальників, 250 споживачів, 100 проміжних пунктів, 10 — верхня границя на кількість проміжних пунктів [9]. Таблиця також містить порівняння часу (в секундах), затраченого програмами **gurobi** та **cplex** на розв'язання задач булевого лінійного програмування для різних значень $d = \overline{2,10}$. Тут також наведено максимальну q_{max} та мінімальну q_{min} долі завантаження проміжних пунктів при реалізації оптимального плану транспортування продукції.

Таблиця 1. Затрати програм **gurobi** та **cplex** для $m = 100$, $l = 100$, $n = 250$

d	f^*_{xyz}	**time gurobi (sec)**	q_{max}	q_{min}	**time cplex (sec)**
2	719902	736.731	0.57	0.43	271.351
3	559784	495.499	0.36	0.29	322.735
4	483311	428.549	0.28	0.22	321.833
5	432791	478.035	0.23	0.14	324.89
6	388650	446.238	0.20	0.13	216.514
7	355522	341.772	0.18	0.12	150.851
8	331430	308.822	0.15	0.10	101.462
9	314464	438.228	0.14	0.07	106.149
10	298488	94.3449	0.13	0.06	86.6779

З таблиці 1 видно, що для булевої двоетапної транспортної задачі (35 000 змінних, 550 обмежень) час розв'язання за допомогою програми **cplex** складає не більше десяти хвилин, що можна використати для розміщення до 10 накопичувачів, які можуть запасати електричну енергію від 100 постачальників та можуть її передавати 250 споживачам [9–11]. Аналогічні затрати по часу для програм **gurobi** та **cplex** будуть мати місце, якщо кількість постачальників буде дорівнювати 250, а кількість споживачів буде дорівнювати 100. Зауважимо, що відповідні ЛП-задачі (1)–(5) та

(17)–(22) при $m = 100$, $l = 100$, $n = 250$ розв'язуються за 1–2 секунди з використанням **gurobi** та **cplex**.

Висновок

Запропоновано дві модифікації класичної двоетапної транспортної задачі за умови, що кількість проміжних пунктів є заданою та їх пропускні спроможності є обмеженими зверху. В їх основу покладена відкрита двоетапна транспортна задача. Перша модифікація враховує верхні границі на пропускні спроможності проміжних пунктів, друга враховує можливість вибору фіксованої кількості проміжних пунктів, меншої за їх загальну кількість. Результати обчислювальних експериментів показують, що розв'язання запропонованих модифікацій задач з використанням солверів gurobi та cplex займає не більше 10 хвилин, а їх відповідні ЛП-задачі — 1-2 секунди.

Першу модифікацію (17)–(22) можна застосовувати при розподіленні та доставці вирощеної продукції для продажу або переробки на власних потужностях агропідприємств. Друга модифікація (28)–(34) є актуальною для пошуку раціонального розташування заданої кількості складів з урахуванням визначеного положення постачальників та отримувачів матеріально-технічних засобів [12]. Вона може бути використана для визначення відповідних місць розташування в ОЕС України накопичувачів електричної енергії та їх енергоємностей [10].

Розглянуті модифікації та їх часткові випадки належать до класу відомих виробничо-транспортних задач. Численні застосування субградієнтних алгоритмів негладкої оптимізації для таких задач описано в [13–16]. На їхній основі розроблено програмне забезпечення для низки моделей оптимального планування, проектування і керування. Наприклад, виробничо-транспортні задачі великої розмірності розв'язувалися для розподілу суден по лініях річкових перевезень, задачі особливо великої розмірності виникали при плануванні завантаження прокатних станів.

Близькі математичні моделі двоетапної транспортної задачі розглядаються в роботах [17, 18], де досліджуються методи розв'язання двоетапної неперервно-дискретної задачі оптимального розбиття-розподілу з заданими положеннями центрів підмножин [17] та з пошуком оптимальних координат розташування центрів підмножин [18]. Ці задачі характеризуються наявністю двох етапів і полягають у визначенні зон збору неперервно розподіленого ресурсу (сировини) підприємствами першого етапу і обсягів перевезень переробленого продукту від підприємств першого етапу до споживачів (пунктів другого етапу) з метою мінімізації сумарних витрат на транспортування ресурсу від постачальників через пункти переробки (збору, зберігання) до споживачів.

Додаток 1. AMPL-код для тесту I (випадкові вартості)

```
param m = 100; # Кількість постачальників (пункти A)
param n = 25;  # Кількість споживачів (пункти B)
param l = 50;  # Всього проміжних пунктів (пункти D)
param d = 5;   # Кількість пунктів D, які обираються (1<d<=l)

# Вартості перевезення одиниці продукції від A до B
param ciki in 1..m, k in 1..l = Uniform(0.1,10);
param ckjk in 1..l, j in 1..n = Uniform(0.1,10);

# Продукція в A та Потреби в B
param x0i in 1..m, j in 1..n = Uniform(0.1,2);
param ai in 1..m = sumj in 1..n x0[i,j]; # Продукція в A
param bj in 1..n = sumi in 1..m x0[i,j]; # Потреби в B
# check: sum{i in 1..m} a[i] = sumj in 1..n b[j];

# Межі на пропускні спроможності пропукних пунктів
param dlowk in 1..l = 0; # нижня межа
param dupk in 1..l = sumi in 1..m a[i]; # верхня межа

# Невідомі (продукція, яку потрібно перевезти)
var xi in 1..m, k in 1..l >= 0; # від A до D
var yk in 1..l, j in 1..n >= 0; # від D до B
var zk in 1..l binary; # 1-задіяно D, 0-незадіяно D

minimize f_opt: # Мінімізувати витрати на перевезення продукції
sumi in 1..m, k in 1..l cik[i,k] * x[i,k] + # від A до D
sumk in 1..l, j in 1..n ckj[k,j] * y[k,j]; # від D до B
subject to # за обмежень
# на перевезення продукції з A до D
con2 i in 1..m: sumk in 1..lx[i,k] = a[i];
# на перевезення продукції з D до B
con3 j in 1..n: sumk in 1..ly[k,j] = b[j];
# не залишати продукцію в проміжних пунктах D
con4 k in 1..l: sumi in 1..mx[i,k] - sumj in 1..ny[k,j] = 0;
# включити/виключити проміжний пункт
con5a k in 1..l: sumi in 1..mx[i,k] <= z[k] * dup[k];
con5b k in 1..l: sumi in 1..mx[i,k] >= z[k] * dlow[k];
# вибрати рівно d проміжних пунктів
con6: sumk in 1..lz[k] = d;

problem dka: f_opt,x,y,z,con2,con3,con4,con5a,con5b,con6;

# option solver gurobi;
solve dka; display f_opt, _solve_time;
```

Додаток 2. AMPL-код для тесту II (випадкові координати)

```
param m = 100; # Кількість постачальників (пункти A)
param n = 25;  # Кількість споживачів (пункти B)
param l = 100; # Всього проміжних пунктів (пункти D) #10000;
param d = 10;  # Кількість пунктів D, які вибираються (1<d<=l)

# випадкові координати (постачальників, споживачів, пр.пунктів)
param  xai in 1..m >= 0;
param  yai in 1..m >= 0;
param  xbj in 1..n >= 0;
param  ybj in 1..n >= 0;
param  xdk in 1..l >= 0;
param  ydk in 1..l >= 0;

# Вартості перевезення одиниці продукції від A до B
param ciki in 1..m, k in 1..l >= 0; # = Uniform(0.1,10);
param ckjk in 1..l, j in 1..n >= 0; # = Uniform(0.1,10);

# Продукція в A та Потреби в B
param x0i in 1..m, j in 1..n = Uniform(0.1,2);
param ai in 1..m = sumj in 1..n x0[i,j]; # Продукція в A
param bj in 1..n = sumi in 1..m x0[i,j]; # Потреби в B
# check: sumi in 1..m a[i] = sumj in 1..n b[j];

# Межі на пропускні спроможності пропукних пунктів
param dlowk in 1..l = 0; # нижня межа
param dupk in 1..l = sumi in 1..m a[i]; # верхня межа

# Невідомі (продукція, яку потрібно перевезти)
var xi in 1..m, k in 1..l >= 0; # від A до D
var yk in 1..l, j in 1..n >= 0; # від D до B
var zk in 1..l binary; # 1-задіяно D, 0-незадіяно D

minimize f_opt: # Мінімізувати витрати на перевезення продукції
sumi in 1..m, k in 1..l cik[i,k] * x[i,k] + # від A до D
sumk in 1..l, j in 1..n ckj[k,j] * y[k,j]; # від D до B
subject to # за обмежень
# на перевезення продукції з A до D
con2 i in 1..m: sumk in 1..lx[i,k] = a[i];
# на перевезення продукції з D до B
con3 j in 1..n: sumk in 1..ly[k,j] = b[j];
# не залишати продукцію в проміжних пунктах D
con4 k in 1..l: sumi in 1..mx[i,k] - sumj in 1..ny[k,j] = 0;
# включити/виключити проміжний пункт
con5a k in 1..l: sumi in 1..mx[i,k] <= z[k] * dup[k];
con5b k in 1..l: sumi in 1..mx[i,k] >= z[k] * dlow[k];
# вибрати рівно d проміжних пунктів
con6: sumk in 1..lz[k] = d;

problem dka: f_opt,x,y,z,con2,con3,con4,con5a,con5b,con6;

for {i in 1..m}{
   let xa[i] := 100 * Uniform(0,1);
   let ya[i] := 100 * Uniform(0,1);
}

for {j in 1..n}{
   let xb[j] := 100 * Uniform(0,1);
   let yb[j] := 100 * Uniform(0,1);
```

```
    }

    for {k in 1..l}{
       let xd[k] := 100 * Uniform(0,1);
       let yd[k] := 100 * Uniform(0,1);
    }
    # display xa,ya;

    for {i in 1..m}{
       for {k in 1..l}{
          let cik[i,k] := round(sqrt((xa[i] - xd[k]) * (xa[i] - xd[k]) +
          (ya[i] - yd[k]) * (ya[i] - yd[k])));
            }
    }

    for {k in 1..l}{
       for {j in 1..n}{
          let ckj[k,j] := round(sqrt((xd[k] - xb[j]) * (xd[k] - xb[j]) +
          (yd[k] - yb[j]) * (yd[k] - yb[j])));
            }
    }

    # option solver gurobi;
    solve dka; display f_opt, _solve_time;
```

Література

1. Карагодова О. О., Кігель В. Р., Рожок В. Д. Дослідження операцій: Навч. посіб. К.: Центр учбової літератури. 2007. 256 с.
2. Наконечний С. І., Савіна С. С. Математичне програмування: Навч. посіб. К.: КНЕУ. 2003. 452 с.
3. Стецюк П. І., Хом'як О. М. Двоетапна транспортна задача та її модифікації. Тези доповідей XIX міжнародної науково-практичної конференції «Математичне та програмне забезпечення інтелектуальних систем (MPZIS-2021)», м. Дніпро, 17-19 листопада 2021 р. Дніпро: ДНУ. 2021. С. 179–186.
4. Стецюк П. І., Ляшко В. І., Мазютинець Г. В. Двоетапна транспортна задача та її AMPL-реалізація. *Наукові записки НаУКМА. Комп'ютерні науки*. 2018. Т. 1. С. 14–20.
5. Стецюк П. І., Бисага О. П., Трегубенко С. С. Двоетапна транспортна задача з обмеженням на кількість проміжних пунктів. *Компьютерная математика*. 2018. №2. С. 119–128.
6. Lloyd S. Least squares quantization in PCM. *IEEE Trans. Inf. Theory*. 1982. 28(2). P. 129–137.
7. Arthur D., Vassilvitskii S. Kmeans++: the advantage of careful seeding. In: Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms. SODA '07, SIAM, Philadelphia. 2007. P. 1027–1035.
8. Jain A. K., Dubes R. C. Algorithms for Clustering Data. Prentice-Hall. 1988. 320 p.
9. Стецюк П. І., Горбачук В. М., Хом'як О. М. Розроблення оптимізаційних процедур для задач розташування накопичувачів електроенергії в ОЕС України в сучасних умовах технологічних змін. Етап 1. Розроблення математичних

моделей, методів та програмного забезпечення для спеціальних класів двоетапних транспортних задач. Заключний звіт про науково-дослідну роботу № держ. реєстрації 0119U001641. К.: Ін-т кібернетики імені В.М. Глушкова НАН України. 2019. 81 с.
10. Буткевич О. Ф., Юнєєва Н. Т., Гурєєва Т. М., Стецюк П. І. Задача розташування накопичувачів електроенергії в ОЕС України з урахуванням його впливу на потоки потужності контрольованими перетинами. *Технічна електродинаміка.* 2020. № 4. С. 46–50.
11. Стецюк П. І., Буткевич О. Ф., Хом’як О. М., Стовба В. О. Методи оптимізації зі зменшенням ризиків для розміщення об’єктів у виробництві відновлюваної енергії. Заключний звіт про науково-дослідну роботу № держ. реєстрації 0121U113859. К.: Ін-т кібернетики імені В.М. Глушкова НАН України. 2021. 60 с.
12. Романченко І. С., Хазанович О. І., Трегубенко С. С. Моделювання системи матеріально-технічного забезпечення. Центр НДІ Збройн. Сил України. Львів: Нац. акад. сухопут. військ. 2015. 155 с.
13. Шор Н. З. Методы минимизации недифференцируемых функций и их приложения. Киев: Наукова думка, 1979. 200 с.
14. Михалевич В. С., Трубин В. А., Шор Н. З. Оптимизационные задачи производственно-транспортного планирования: модели, методы, алгоритмы. М.: Наука. Гл. ред. физ.-мат. лит. 1986. 264 с.
15. Соломон Д. И. Дробное программирование и недифференцируемая оптимизация. Кишинэу, Эврика. 2010. 554 с.
16. Сергієнко І. В. Методи оптимізації та системного аналізу для задач трансобчислювальної складності. К.: Академперіодика, 2010. 296 с.
17. Киселева Е. М., Притоманова О. М., Ус С. А. Решение двухэтапной непрерывно-дискретной задачи оптимального разбиения-распределения с заданным положением центров подмножеств. *Кибернетика и системный анализ.* 2020. № 1. С. 3–15.
18. Kiseleva E., Prytomanova O., Hart L. Solving a Two-stage Continuous-discrete Problem of Optimal Partitioning-Allocation with the Subsets Centers Placement. Open Computer Science. De Gruyter. 2020. Vol. 10. P. 124–136.

Надійшла: 06.06.2022 / Прийнята: 27.06.2022