Міністерство освіти і науки України
Харківський національний університет імені В. Н. Каразіна
Факультет математики і інформатики
Кафедра прикладної математики

До захисту допущено
кафедрою прикладної математики, протокол №5 від12 червня 2026 р.

завідувач кафедри
прикладної математики
доктор фіз.-мат. наук, професор *Валерій КОРОБОВ*

КВАЛІФІКАЦІЙНА РОБОТА

здобувача першого (бакалаврського) рівня вищої освіти

«Математичні підходи до розв'язання задач багатокритеріальної оптимізації»

Спеціальність 113 Прикладна математика
Освітня програма Прикладна математика

Здобувач Анжеліка СИНЯКІНА

Науковий керівник кандидат фіз.-мат. наук, доцент
кафедри прикладної математики
Тетяна РЕВІНА

Харків – 2026

АНОТАЦІЯ

Кваліфікаційну роботу присвячено дослідженню методів багатокритеріальної оптимізації та їх практичному застосуванню для розв'язання задачі розподілу вагонів між вантажними пунктами.

Для розв'язку задачі багатокритеріальної оптимізації спочатку наведено огляд теорії Парето-ефективних розв'язків задачі. Потім досліджено різні методи вибору вагових коефіцієнтів або лінійної згортки (метод розподілу балів, метод ранжування критеріїв та метод попарних порівнянь). Також досліджено метод цільового програмування. Обидва підходи дозволяють звести багатокритеріальну задачу до однокритеріальної для подальшого застосування симплекс-методу.

Теоретичні методи були застосовані для розв'язку практичної задачі розподілу вагонів по вантажним фронтам. Розроблено програмний скрипт мовою Python для пошуку Парето-ефективних розв'язків.

Наукова новизна роботи полягає в аналітичному обґрунтуванні траєкторії переходу між двома цільовими стратегіями у багатокритеріальній задачі. Завдяки аналітичному визначенню граничних значень вагових коефіцієнтів доведено, що прямий перехід від стратегії «пріоритет залізниці» до «пріоритет клієнтів» є неможливим. Обґрунтовано, що цей процес відбувається виключно послідовно й обов'язково передбачає пошук компромісного рішення. У статті [9] до задачі розподілу вагонів було застосовано метод Парето та метод вагових коефіцієнтів (випадок рівної важливості критеріїв), тоді як у кваліфікаційній роботі застосовано метод вагових коефіцієнтів (випадки домінування одного з критеріїв) та метод цільового програмування як альтернативний підхід.

У результаті дослідження було знайдено оптимальні розв'язки для задачі розподілу вагонів на основі методу лінійної згортки критеріїв. Завдяки проведеному аналізу було знайдено критичні значення вагових коефіцієнтів та визначено інтервали дії кожної зі стратегій.

Ключові слова: багатокритеріальна оптимізація, множина Парето, лінійна згортка критеріїв, вагові коефіцієнти, симплекс-метод.

ABSTRACT

Syniakina Anzhelika. Mathematical Approaches to Solving Multi-Criteria Optimization Problems.

This Bachelor's Thesis investigates multi-objective optimization methods and their practical application in solving the wagon allocation problem among loading points.

First, the work reviews the theory of Pareto-optimal solutions. Then, it examines different weighting methods: point allocation, ranking and pairwise comparisons. The goal programming method is also analyzed. Both approaches transform the multi-criteria problem into a single-criterion form. This allows using the simplex method to solve it.

These theoretical methods were used to solve a practical problem of wagon allocation. A Python script was created to find the Pareto-optimal solutions.

The scientific novelty is that the work explains how to transition between two main strategies. First, the boundary values of the weight coefficients are determined analytically. Based on these results, a direct shift from "railway priority" to "customer priority" is proven to be impossible. This transition is shown to be strictly sequential. The process always involves finding a compromise solution. In the article [9], the Pareto method and the weighted coefficients method were used for the wagon distribution problem. The weighted coefficients method was applied in the case of equal importance of the criteria. In the qualification thesis, the weighted coefficients method was also used, but for cases where one criterion dominates the others. Goal programming was applied as an alternative approach.

As a result of the research, optimal solutions for the wagon allocation problem were found based on the linear scalarization method. Through the conducted analysis, the critical values of the weighting coefficients were identified, and the effective intervals for each strategy were determined.

Keywords: multi-objective optimization, Pareto set, linear scalarization, weighting coefficients, simplex method.

Зміст

ВСТУП

Задачі оптимізації виникають практично в усіх сферах життєдіяльності та охоплюють такі дисципліни, як наука, економіка, промисловість і технології. У сучасній прикладній математиці дедалі частіше доводиться працювати з оптимізацією відразу багатьох критеріїв. Специфіка багатокритеріальних задач полягає в тому, що задоволення інтересів одного суб'єкта зазвичай відбувається за рахунок погіршення показників іншого. Такі задачі незамінні у багатьох прикладних напрямах, зокрема в логістиці, операційному менеджменті та теорії прийняття рішень.

Становлення багатокритеріальної оптимізації як самостійного наукового напряму зазвичай пов'язують із початком 1950-х років. Фундаментальний внесок у формування цієї теорії у 1951 році незалежно один від одного зробили Koopmans [1], який увів у науковий обіг базове поняття домінування альтернатив, а також Kuhn and Tucker [2], які розробили математичні умови оптимальності для векторних задач. Попри важливість цих праць, тривалий час дослідження залишалися переважно в межах абстрактної математики.

Велике значення для розвитку цього напряму мала перша міжнародна конференція з багатокритеріальної оптимізації 1972 року в університеті Південної Кароліни в Америці. Відтоді інтерес до теми зріс і вона набула широкого застосування. Наразі існує велика кількість різноманітних методів для розв'язання задач багатокритеріальної оптимізації. Зокрема, такі підходи, як метод зваженого добутку, цільове програмування, метод компромісного програмування, метод аналізу ієрархій, лексикографічний метод, метод контрольних (опорних) точок тощо, були детально розглянуті у роботі [3].

В кваліфікаційній роботі досліджується задача розподілу вагонів по вантажним пунктам. Математичною задачею тут є суперечність між цілями залізничного перевізника, який прагне мінімізувати свої витрати (стратегія «пріоритет залізниці»), та інтересами вантажовласників, (стратегія «пріоритет клієнтів»). Саме така задача і лежить в основі практичної частини моєї роботи.

РОЗДІЛ 1. МАТЕМАТИЧНА МОДЕЛЬ ЗАДАЧ БАГАТОКРИТЕРІАЛЬНОЇ ОПТИМІЗАЦІЇ

1.1. Загальна характеристика задач багатокритеріальної оптимізації

Багатокритеріальна оптимізація (Multi-criteria Optimization) – це процес одночасної оптимізації двох або більше цільових функцій у межах однієї області допустимих розв’язків.

У загальному вигляді математична модель задачі багатокритеріальної оптимізації полягає у пошуку такого вектора змінних $x = (x_1, x_2, \dots, x_n)$, який би оптимізував (мінімізував або максимізував) вектор цільових функцій $F(x)$ при дотриманні заданих обмежень.

Оптимізувати:

$$F(x) = [F_1(x), \dots, F_m(x)] \to min, \qquad m \geq 2, \tag{1.1}$$

при обмеженнях:

$$g_k = \sum_{j=1}^{n} a_{kj} x_j \leq 0, \tag{1.2}$$

кожна цільова функція має вигляд:

$$F_i(x) = \sum_{j=1}^{n} c_{ij} x_j, \tag{1.3}$$

де:

x_j – шукані змінні, $j = 1, \dots, n$;

n – кількість змінних;

m – кількість цільових функцій;

a_{kj} – коефіцієнти при змінних у системі обмежень;

c_{ij} – вагові коефіцієнти (параметри) i-ї цільової функції.

Зауважимо, що багато методів багатокритеріальної оптимізації застосовуються у випадках, коли цільові функції і обмеження не обов’язково лінійні. В кваліфікаційній роботі буде розглянуто тільки лінійний випадок.

1.2. Методи розв’язання задач багатокритеріальної оптимізації

1.2.1. Метод Парето

Класичним підходом до розв’язання задач багатокритеріальної оптимізації прийнято вважати метод Парето. Поняття Парето-оптимальності у 1896 році вперше було сформульовано італійським економістом Вільфредом Парето, а строге математичне обґрунтування цього принципу наведено в роботі [4].

У задачах багатокритеріальної оптимізації з вектором цільових функцій (1.1) не існує єдиного рішення, яке було б оптимальним одразу для всіх критеріїв. В такому випадку використовується метод Парето, який виключає всі доміновані рішення і знаходить множину недомінованих (Парето-оптимальних) рішень. Цей принцип працює однаково як для лінійних, так і для нелінійних задач.

Область допустимих розв’язків-це множина.

$$D = \left\{ x \in \mathbb{R}^n \,\middle|\, \sum_{j=1}^{n} a_{kj} x_j \le 0 \right\} \tag{1.4}$$

Означення 1. Домінування (за Парето). Вектор розв’язку $x' \in D$ домінує вектор $x \in D$ (позначається як $x' \succ x$), якщо:

1) $F_i(x') \le F_i(x), \forall i = 1, \dots, m$
2) $\exists k \in \{1, \dots, m\}: F_k(x') < F_k(x)$

Тобто x не гірший за всіма критеріями і строго кращий хоча б за одним. Якщо ж для двох векторів $x', x \in D$ не виконується ні умова $x' \succ x$, ні $x \succ x'$ то такі розв'язки називаються **незрівняними**.

Означення 2. Парето-оптимальний розв’язок. Розв'язок x називається Парето-оптимальним (або ефективним), якщо в області допустимих розв'язків D не існує іншого вектора x', який би домінував x. Тобто, неможливо покращити значення жодного з критеріїв $F_i(x)$, не погіршивши при цьому значення хоча б одного іншого критерію.

Означення 3. Множина Парето-оптимальних розв’язків – це множина всіх розв’язків, які не домінуються жодним іншим допустимим розв’язком і визначається як:

$$X = \{x \in D \mid \nexists\, x' \in D : F(x') \succ F(x)\}$$

Означення 4. Фронт Парето - образ множини Парето-оптимальних розв'язків у просторі цільових функцій при відображенні F і визначається як:

$$XF = \{u = F(x) \mid x \in X\}$$

Отже, множину, що складається з Парето-оптимальних розв'язків у просторі цільових функцій, називають множиною Парето, а криву або поверхню, яку утворює ця множина - фронт Парето.

Метод Парето не знаходить єдиного рішення, отже після застосування цього методу зазвичай використовують метод вагових коефіцієнтів, де надають пріоритет одному з критеріїв, щоб отримати єдиний розв'язок з множини Парето-оптимальних розв'язків.

1.2.2. Метод вагових коефіцієнтів

Метод вагових коефіцієнтів (або лінійної згортки) формувався поступово завдяки працям багатьох дослідників. Математичне обґрунтування методу наведено у роботі [5].

Суть цього методу полягає у переході від векторної задачі (одночасної мінімізації двох і більше цільових функцій) до скалярної задачі шляхом формування єдиного критерію F:

$$F = w_1 \cdot F_1 + \dots + w_m \cdot F_m \to min, \quad m \geq 2, \tag{1.5}$$

де w_m - це вагові коефіцієнти, що відображають відносну важливість (пріоритетність) кожного з критеріїв, і для яких виконується умова нормалізації:

$$\sum w_i = 1 \tag{1.6}$$

Варто зазначити, що цей метод є універсальним: тип загальної задачі залежить тільки від того, якими є самі критерії. Тобто, якщо вихідні цільові функції F_i лінійні, то й загальна задача буде лінійною, і навпаки.

Існує багато підходів для вибору вагових коефіцієнтів. Теоретичні відомості основних наведені в наступному підрозділі.

1.2.3. Методи вибору вагових коефіцієнтів

Порівняльний аналіз підходів для визначення вагових коефіцієнтів наведені у роботі [6].

1. Метод розподілу балів (Point Allocation)

Суть методу полягає у розподілі фіксованої суми балів між критеріями пропорційно до їхньої суб'єктивної важливості.

Нехай S - це загальна сума балів, яку треба розподілити між усіма критеріями.

У методі "Point Allocation" традиційно приймається $S = 100$.

Нехай p_i - кількість балів, присвоєна i-му критерію, де $i = 1, \dots, m$.

Умова розподілу:

$$\sum_{i=1}^{m} p_i = S \tag{1.7}$$

Визначення ваги:

$$w_i = \frac{p_i}{S} \tag{1.8}$$

Якщо відсутня інформація щодо домінування одного з критеріїв, прийнято вважати їх рівнозначними. В такому випадку застосовується принцип недостатньої підстави, згідно з яким бали розподіляються порівну. Якщо маємо 2 критерія, то розподіл балів відбудеться наступним чином :

$$w_1 = w_2 = \frac{50}{100} = 0{,}5$$

2. Метод ранжування критеріїв (Ranking Method)

За допомогою методу, представленому в роботі [6], ми могли б не призначати ваги, а проранжувати критерії.

Формула Rank Sum (Сума рангів):

$$w_j = \frac{m - p_j + 1}{\sum_{k=1}^{m} m - p_k + 1}, \tag{1.9}$$

де p_j −це ранг j-го критерію, $j = 1,2, .., m$.

Головним недоліком методу ранжування є відсутність так званої «відстані» між рангами. На відміну від попереднього методу розподілу балів, тут ми лише вибудовуємо критерії в порядку їх важливості, але не вказуємо, наскільки один критерій важливіший за інший.

3. **Метод попарних порівнянь (Pairwise comparison)**

В основі цього методу лежить використання шкали відносної важливості Томаса Сааті, запропонованої в роботі [7] і наведеної в таблиці 1.1.

1	Рівна важливість
3	Помірна важливість
5	Сильна важливість
7	Дуже сильна важливість
9	Надзвичайна важливість
2, 4, 6, 8	Проміжні значення

Таблиця 1.1

Формула кількості порівнянь:

$$C = \frac{m(m-1)}{2}, \quad (1.10)$$

де m-кількість критеріїв.

Хоча метод попарних порівнянь і є найбільш точним серед розглянутих, він водночас має найбільшу математичну складність. Його головний недолік полягає у збільшенні кількості порівнянь при додаванні нових критеріїв, що суттєво ускладнює його використання. Наприклад для задачі з 10 критеріями доведеться виконати 45 порівнянь.

Метод попарних порівнянь складається з кількох етапів [7]:

1. Побудова матриці попарних порівнянь

   На першому етапі розробляється матриця, де всі критерії порівнюються між собою.

   Нехай А - вихідна матриця попарних порівнянь розміром $m \times m$, де m - кількість критеріїв, а a_{ij}- елемент матриці.

$$A = \begin{bmatrix} a_{11} & \cdots & a_{1m} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mm} \end{bmatrix} \tag{1.11}$$

Для заповнення матриці використовується числова шкала, що представлена в таблиці 1.

Діагональні елементи матриці завжди дорівнюють 1. Матриця заповнюється таким чином, що якщо елемент (i, j) має певне значення, то елемент (j, i) має обернене значення.

2. Обчислення вагових коефіцієнтів
   - Обчислення суми кожного стовпця (S_j):

$$S_j = \sum_{i=1}^{m} a_{ij} \tag{1.12}$$

   - Побудова нормалізованої матриці (A_{norm}):

$$A_{norm} = \begin{bmatrix} a'_{11} & \cdots & a'_{1m} \\ \vdots & \ddots & \vdots \\ a'_{m1} & \cdots & a'_{mm} \end{bmatrix} \qquad a'_{ij} = \frac{a_{ij}}{S_j} = \frac{a_{ij}}{\sum_{k=1}^{m} a_{kj}} \tag{1.13}$$

   - Обчислення фінальних ваг (w_i):

$$w_i = \frac{\sum_{j=1}^{m} a'_{ij}}{m} \tag{1.14}$$

1.2.4. Метод цільового програмування

Метод цільового програмування (Goal Programming) є одним із найпопулярніших теоретичних підходів серед методів багатокритеріальної оптимізації. Цей метод створений на основі праць таких науковців, як Charnes, Cooper (1955, 1961) та Ijiri (1965). Зокрема, Charnes і Cooper першими обґрунтували концепцію «архімедового» зваженого програмування [8], де для кожної окремої цільової функції $F_i(x)$ встановлюються конкретні бажані цільові рівні - так звані рівні аспірації (aspiration levels) g_i.

Для врахування відхилень кожна i-та цільова функція подається у вигляді рівняння шляхом введення двох додаткових невід'ємних змінних:

- d_i^+ - змінна позитивного відхилення, яка відображає перевищення встановленої цілі g_i.
- d_i^- - змінна негативного відхилення, яка відображає недосягнення встановленої цілі g_i.

Повне відхилення записується у вигляді абсолютного значення, яке визначається як сума його позитивної та негативної складових:

$$|d_i| = d_i^+ + d_i^- \tag{1.15}$$

Також мають виконуватись умови невід'ємності та ортогональності:

$$d_i^+ \geq 0, \quad d_i^- \geq 0, \quad d_i^+ \cdot d_i^- = 0$$

Остання умова гарантує, що для одного критерію неможливе одночасне перевищення, і недосягнення цільового показника (тобто щонайменше одна зі змінних завжди дорівнює нулю).

У загальному вигляді математична модель зваженого цільового програмування формулюється як задача мінімізації суми небажаних відхилень від встановлених цілей:

$$min \sum_{i=1}^{m} (\alpha_i \, d_i^+ + \beta_i d_i^-) \tag{1.16}$$

при дотриманні системи цільових обмежень:

$$F_i(x) - d_i^+ + d_i^- = g_i, \quad i = 1, 2, \dots, m \tag{1.17}$$

Коефіцієнти α_i та β_i відіграють роль “штрафних” коефіцієнтів, які визначають вагомість перевищення або недосягнення відповідної i-ї цілі.

Варто зазначити, що наведена модель підходить виключно для лінійних задач. Для застосування методу в нелінійних задачах необхідно перейти до мінімізації суми квадратів відхилень.

РОЗДІЛ 2. ЗАДАЧА РОЗПОДІЛУ ВАГОНІВ

2.1. Постановка задачі розподілу вагонів по вантажним пунктам

У статті [9] було наведено приклад задачі розподілу вагонів між вантажними пунктами. До вантажної станції прибуває 50 вагонів, які необхідно вивантажити впродовж доби на трьох вантажних пунктах місткістю відповідно $m_1 = 25$ вагонів, $m_2 = 28$ вагонів, $m_3 = 20$ вагонів. Через різне технічне обладнання вантажних пунктів, витрати залізниці, пов'язані з вивантаженням вагонів на них, різні і становлять $C_1 = 20$ у.о./ваг, $C_2 = 30$ у.о./ваг, $C_3 = 35$ у.о./ваг. Плата клієнтів за вивантаження вантажів складає $\Pi_1 = 60$ у.о./ваг, $\Pi_2 = 55$ у.о./ваг, $\Pi_3 = 45$ у.о./ваг. Необхідно знайти ефективні розв'язки, які забезпечують мінімум витрат залізниці та клієнтів при вивантаженні вагонів.

При формалізації задачі в якості змінних обираємо кількість вагонів, які розвантажуються на вантажних пунктах: x_1 - на першому пункті, x_2 - на другому пункті, x_3 - на третьому пункті. Цільові функції матимуть такий вигляд:

$$C = 20x_1 + 30x_2 + 35x_3 \rightarrow min \qquad (2.1)$$

$$\Pi = 60x_1 + 55x_2 + 45x_3 \rightarrow min \qquad (2.2)$$

Задача має такі обмеження:

– по загальній кількості вагонів, які розвантажуються:

$$x_1 + x_2 + x_3 = 50 \qquad (2.3)$$

– по місткості вантажних пунктів:

$$x_1 \leq 25,\ x_2 \leq 28,\ x_3 \leq 20 \qquad (2.4)$$

Слід зазначити, що кількість вагонів не може бути від'ємним числом: $x_i \geq 0$.

Таким чином, задача розподілу вагонів по вантажним пунктам матиме такий вигляд:

$$\begin{cases} x_1 + x_2 + x_3 = 50; \\ x_1 \leq 25 \\ x_2 \leq 28 \\ x_3 \leq 20 \\ x_i \geq 0 \end{cases} \qquad (2.5)$$

$$C = 20x_1 + 30x_2 + 35x_3 \rightarrow min$$
$$\Pi = 60x_1 + 55x_2 + 45x_3 \rightarrow min$$

2.2. Застосування різних методів розв'язку

2.2.1. Метод Парето

У статті [9] було знайдено множину ефективних рішень за Парето. Результат був представлений на рисунку 1 та у таблиці 2 без алгоритму обчислення. Тому мною було розроблено програмну реалізацію мовою Python для знаходження множини ефективних рішень Парето. Код наведено в Додатках (с. 32) і в результаті його використання, мною була отримана таблиця з такими самими значеннями.

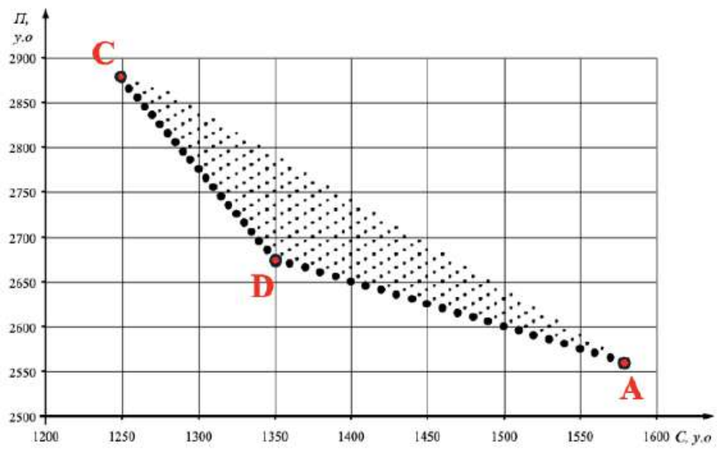

Рисунок 2.1

На рисунку 2.1 представлена множина усіх допустимих рішень у вигляді множини точок, яка містить 294 варіанти розв'язку. Точки, що виділені жирним чорним кольором, утворюють підмножину ефективних за Парето рішень, що представлені в таблиці 2.2. Кутові точки, виділені червоним кольором, відповідають оптимальним планам. Точка А відповідає рішенню, де витрати залізниці є мінімальними. Точка С навпаки, відповідає випадку, коли мінімальними є витрати клієнтів. Точка В, характерно своєму положенню посередині, відповідає рівній важливості мінімізації як витрат залізниці, так і витрат клієнтів.

Множина ефективних за Парето рішень

№	x_1	x_2	x_3	C	$П$	№	x_1	x_2	x_3	C	$П$
1	25	5	20	1 350	2 675	23	23	7	20	1 370	2 665
2	25	6	19	1 345	2 685	24	22	8	20	1 380	2 660
3	25	7	18	1 340	2 695	25	21	9	20	1 390	2 655
4	25	8	17	1 335	2 705	26	20	10	20	1 400	2 650
5	25	9	16	1 330	2 715	27	19	11	20	1 410	2 645
6	25	10	15	1 325	2 725	28	18	12	20	1 420	2 640
7	25	11	14	1 320	2 735	29	17	13	20	1 430	2 635
8	25	12	13	1 315	2 745	30	16	14	20	1 440	2 630
9	25	13	12	1 310	2 755	31	15	15	20	1 450	2 625
10	25	14	11	1 305	2 765	32	14	16	20	1 460	2 620
11	25	15	10	1 300	2 775	33	13	17	20	1 470	2 615
12	25	16	9	1 295	2 785	34	12	18	20	1 480	2 610
13	25	17	8	1 290	2 795	35	11	19	20	1 490	2 605
14	25	18	7	1 285	2 805	36	10	20	20	1 500	2 600
15	25	19	6	1 280	2 815	37	9	21	20	1 510	2 595
16	25	20	5	1 275	2 825	38	8	22	20	1 520	2 590
17	25	21	4	1 270	2 835	39	7	23	20	1 530	2 585
18	25	22	3	1 265	2 845	40	6	24	20	1 540	2 580
19	25	23	2	1 260	2 855	41	5	25	20	1 550	2 575
20	25	24	1	1 255	2 865	42	4	26	20	1 560	2 570
21	25	25	0	1 250	2 875	43	3	27	20	1 570	2 565
22	24	6	20	1 360	2 670	44	2	28	20	1 580	2 560

Таблиця 2.2

2.2.2. Метод вагових коефіцієнтів

У статті [9] було обрано значення коефіцієнтів: $w_1 = w_2 = 0{,}5$. Обидва коефіцієнта є рівними, тобто інтереси залізниці (мінімізація С) є такими ж важливими як і інтереси клієнтів (мінімізація П), тобто це відповідає компромісному розв’язку.

Використовуючи критерій (1.5), зведемо дві цільові функції до однієї та запишемо загальну математичну модель задачі з урахуванням початкових обмежень по місткості вантажних пунктів:

$$\begin{cases} x_1 + x_2 + x_3 = 50; \\ x_1 \leq 25 \\ x_2 \leq 28 \\ x_3 \leq 20 \\ x_1, x_2, x_3 \geq 0 \end{cases} \tag{2.6}$$

$$F = 0{,}5(20x_1 + 30x_2 + 35x_3) + 0{,}5(60x_1 + 55x_2 + 45x_3) \rightarrow min$$

Або після згортки критеріїв:

$$F = 40x_1 + 42{,}5x_2 + 40x_3 \rightarrow min$$

Обчислимо результат за допомогою симплекс методу (код Python), що представлений у Додатках (с. 31). У результаті обчислень отримано такий оптимальний план розподілу вагонів: (x_1, x_2, x_3) = (25; 5; 20).

Витрати залізниці (С) : 1350.00 у.о.

Плата клієнтів (П) : 2675.00 у.о.

Отже, якщо ми хочемо порівно враховувати як інтереси залізниці, так і клієнтів, слід по максимуму використати 1 і 3 вантажний пункти. Тоді витрати залізниці складатимуть 1350 у.о., а плата клієнтів 2675 у.о.

Тепер дослідимо як зміниться результат, якщо надати перевагу одному з критеріїв, використовуючи методи, описані в підрозділі 1.2.3.

1. *Метод розподілу балів*

- "Пріоритет залізниці": Надамо цьому критерію більшу вагу.

Нехай w_1= 0.8, w_2= 0.2

Тоді нова цільова функція матиме такий вигляд:

$$F = 0{,}8 * C + 0{,}2 * \text{П} \rightarrow min$$

$$F = 0{,}8 * (20x_1 + 30x_2 + 35x_3) + 0{,}2 * (60x_1 + 55x_2 + 45x_3)$$
$$= 28x_1 + 35x_2 + 37x_3 \rightarrow min$$

Знаходимо розв'язки за допомогою симплекс методу (код Python):

Оптимальний план (x_1, x_2, x_3): (25; 25; 0).

Витрати залізниці (С) : 1250.00 у.о.

Плата клієнтів (П) : 2875.00 у.о.

Отже, щоб мінімізувати витрати залізниці, не використовуємо 3 вантажний пункт, натомість використовуємо 1(найдешевший) і 2(середній за вартістю). Витрати залізниці зменшуються до 1250 у.о., але в такому випадку клієнти змушені платити більше, а саме 2875 у.о.

- "Пріоритет клієнтів": Надамо цьому критерію більшу вагу.

Беремо w_1= 0.2, w_2= 0.8

Нова цільова функція:

$$F = 52x_1 + 50x_2 + 43x_3 \rightarrow min$$

Знаходимо розв'язки за допомогою симплекс методу(код Python):

Оптимальний план (x_1, x_2, x_3): (2; 28; 20).

Витрати залізниці (С) : 1580.00 у.о.

Плата клієнтів (П) : 2560.00 у.о.

Порівняння результатів в залежності від коефіцієнтів:

Варіант пріоритету	Вагові коефіцієнти w_1, w_2	Оптимальний план (x_1, x_2, x_3)	Витрати залізниці	Плата клієнтів
Однаковий пріоритет	0,5; 0,5 *розібрано в статті	(25; 5; 20)	1350 у.о.	2675 у.о.
Пріоритет залізниці	0,8; 0,2	(25; 25; 0)	1250 у.о.	2875 у.о.
Пріоритет клієнтів	0,2; 0,8	(2; 28; 20)	1580 у.о.	2560 у.о.

Таблиця 2.3

Для подальшого більш глибокого аналізу застосуємо інші методи визначення вагових коефіцієнтів (описані у підрозділі 1.2.3) та порівняємо їх вплив на кінцевий розв'язок задачі.

2. *Метод ранжування*

- "Пріоритет залізниці"

Ранг 1 (p_1 = 1): Витрати С

Ранг 2 (p_2 = 2): Плата П

Використовуючи формулу суми рангів (1.9), обчислимо вагові коефіцієнти:

$$w_1 = \frac{2-1+1}{(2-1+1)+(2-2+1)} = \frac{2}{3} \approx 0{,}67$$

$$w_2 = \frac{2-2+1}{(2-1+1)+(2-2+1)} = \frac{1}{3} \approx 0{,}33$$

- "Пріоритет клієнтів"

Ранг 1 (p_1 = 1): Плата П

Ранг 2 (p_2 = 2): Витрати С

$$w_1 = \frac{2-2+1}{(2-1+1)+(2-2+1)} = \frac{1}{3} \approx 0{,}33$$

$$w_2 = \frac{2-1+1}{(2-1+1)+(2-2+1)} = \frac{2}{3} \approx 0{,}67$$

Таким чином, замість 0,8 та 0,2, ми могли б використати 0,67 і 0,33.

3. Метод попарних порівнянь

Заповнюємо матрицю (1.11) використовуючи числову шкалу (Saaty), представлену в таблиці 1.1. Припустимо, що обрано помірну перевагу критерію мінімізації витрат залізниці (С) над критерієм плати клієнтів (П), що відповідає оцінці 3. Тоді матриця попарних порівнянь набуде вигляду:

$$A = \begin{bmatrix} 1 & 3 \\ {}^{1}/_{3} & 1 \end{bmatrix}$$

Обчислюємо суми елементів для кожного стовпця за формулою (1.12):

$$S_1 = 1 + \frac{1}{3} = \frac{4}{3}$$

$$S_2 = 3 + 1 = 4$$

Будуємо нормалізовану матрицю (1.13):

$$A_{norm} = \begin{bmatrix} 1/(4/3) & 3/4 \\ (1/3)/(4/3) & 1/4 \end{bmatrix} = \begin{bmatrix} 0{,}75 & 0{,}75 \\ 0{,}25 & 0{,}25 \end{bmatrix}$$

Обчислюємо фінальні вагові коефіцієнти за формулою (1.14):

$$w_1 = \frac{0{,}75 + 0{,}75}{2} = 0{,}75$$

$$w_2 = \frac{0{,}25 + 0{,}25}{2} = 0{,}25$$

Узагальнимо отримані на попередніх етапах розв'язки у вигляді порівняльної таблиці 2.4.

Метод	Випадок	Вагові коефіцієнти w_1, w_2	Оптимальний план (x_1, x_2, x_3)	Витрати залізниці, у.о.	Плата клієнтів, у.о.
“Розподіл балів”	1	1; 0	(25; 25; 0)	1250	2875
	2	0; 1	(2; 28; 20)	1580	2560
	3	0,5; 0,5	(25; 5; 20)	1350	2675
	4	0,8; 0,2	(25; 25; 0)	1250	2875
	5	0,2; 0,8	(2; 28; 20)	1580	2560
“Метод ранжування”	6	0,67; 0,33	(25; 25; 0)	1250	2875
	7	0,33; 0,67	(2; 28; 20)	1580	2560
“Попарного порівняння”	8	0,75; 0,25	(25; 25; 0)	1250	2875
	9	0,25; 0,75	(2; 28; 20)	1580	2560

Таблиця 2.4

За результатами розрахунків, наведених у табл. 2.4, можна зробити висновок, що для стратегії «Пріоритет залізниці» (випадки 1, 4, 6, 8) та стратегії «Пріоритет клієнтів» (випадки 2, 5, 7, 9) оптимальні плани є однаковими, незалежно від обраного методу визначення вагових коефіцієнтів.

2.2.3. Метод цільового програмування

Крок 1. Визначення «Ідеальних цілей» (Targets)

Спочатку необхідно знайти абсолютний мінімум для кожної функції окремо.

1) Найкращий план для залізниці ($C \to min$):

   Необхідно розвантажити 50 вагонів на трьох вантажних пунктах, тож обираємо найвигідніший для залізниці план:

   - Заповнюємо найдешевший 1-й пункт на повну: $x_1 = 25$.
   - Повністю з заповнюємо другий за вартістю пункт: $x_2 = 25$.

Мінімальні витрати: $C = 20 \cdot (25) + 30 \cdot (25) + 30 \cdot (0) = 1250$у.о.

2) Найкращий план для клієнтів ($П \to min$):

   Обираємо найдешевші пункти для клієнтів для розвантаження 50 вагонів:

- Заповнюємо найдешевший 3-й пункт на повну: $x_3 = 20$.
- Повністю з заповнюємо другий за вартістю пункт: $x_2 = 28$.
- Залишок вагонів відправляємо на останній пункт: $x_1 = 2$.

Мінімальні витрати: $П = 45 \cdot (20) + 55 \cdot (28) + 60 \cdot (2) = 2560$ у.о.

Знайдені мінімальні значення також збігаються з попередніми обчисленнями методом вагових коефіцієнтів.

Встановлення цілей: Зазвичай під час переговорів сторони йдуть на незначні поступки. Нехай залізниця фіксує свій цільовий бюджет як $T_C = 1300$ у.о., а клієнти як $T_П = 2600$ у.о.

Крок 2. Формування математичної моделі

На цьому кроці вводимо змінні відхилень. Необхідно мінімізувати суму надвитрат (позитивних відхилень d_C^+ та $d_П^+$) від встановлених бюджетів.

Цільова функція (штрафи за надвитрату):

$$F_{\text{штраф}} = d_C^+ + d_П^+ \to min$$

Нова система з урахуванням цілей:

$$\begin{cases} 20x_1 + 30x_2 + 35x_3 - d_C^+ + d_C^- = 1300 \\ 60x_1 + 55x_2 + 45x_3 - d_П^+ + d_П^- = 2600 \end{cases}$$

Початкові обмеження:

$$\begin{cases} x_1 + x_2 + x_3 = 50; \\ 0 \le x_1 \le 25 \\ 0 \le x_2 \le 28 \\ 0 \le x_3 \le 20 \\ d_C^+, d_C^-, d_П^+, d_П^- \ge 0 \end{cases}$$

Крок 3. Аналітичний розв'язок

Зменшуємо розмірність, виразивши x_3 через x_1 та x_2 :

$$x_3 = 50 - x_1 - x_2$$

І застосуємо обмеження:

1) $x_3 \ge 0$:

$$50 - x_1 - x_2 \ge 0 \Rightarrow x_1 + x_2 \le 50$$

2) $x_3 \le 20$:

$$50 - x_1 - x_2 \le 20 \Rightarrow x_1 + x_2 \ge 30$$

Таким чином, область допустимих розв'язків у двовимірному просторі змінних є опуклим обмеженим багатогранником, обмеженим системою нерівностей:

$$\begin{cases} x_1 + x_2 \leq 50 \\ x_1 + x_2 \geq 30 \\ \quad x_1 \leq 25 \\ \quad x_2 \leq 28 \\ \quad x_{1,2} \geq 0 \end{cases}$$

Теорема 1. У задачі лінійного програмування, якщо область допустимих рішень є опуклим обмеженим багатогранником, то оптимальне значення цільової функції досягається принаймні в одній із кутових точок (вершин) цієї області [10, с. 154–155].

Знайдемо координати всіх вершин багатогранника шляхом попарного розв'язання систем лінійних рівнянь на його границях:

Вершина A: Перетин прямих $x_2 = 28$ та $x_1 + x_2 = 30$.

$$\begin{cases} x_2 = 28 \\ x_1 + x_2 = 30 \end{cases} \Rightarrow x_1 = 2, x_2 = 28$$

Тоді $x_3 = 50 - 2 - 28 = 20$.

Координати вершини: A(2, 28, 20)

Вершина B: Перетин прямих $x_2 = 28$ та $x_1 + x_2 = 50$.

$$\begin{cases} x_2 = 28 \\ x_1 + x_2 = 50 \end{cases} \Rightarrow x_1 = 22, x_2 = 28$$

Тоді $x_3 = 50 - 22 - 28 = 0$.

Координати вершини: B(22, 28, 0)

Вершина C: Перетин прямих $x_1 = 25$ та $x_1 + x_2 = 50$.

$$\begin{cases} x_1 = 25 \\ x_1 + x_2 = 50 \end{cases} \Rightarrow x_1 = 25, x_2 = 25$$

Тоді $x_3 = 50 - 25 - 25 = 0$.

Координати вершини: C(25, 25, 0)

Вершина D: Перетин прямих $x_1 = 25$ та $x_1 + x_2 = 30$.

$$\begin{cases} x_1 = 25 \\ x_1 + x_2 = 30 \end{cases} \Rightarrow x_1 = 25, x_2 = 5$$

Тоді $x_3 = 50 - 25 - 5 = 20$.

Координати вершини: D(25, 5, 20)

Крок 4. Розрахунок функції штрафів у вершинах багатогранника

Здійснимо обчислення значень критеріїв C та П, величин перевищення встановлених бюджетів (d_C^+, d_Π^+) та значення цільової функції $F_{\text{штраф}}$ для кожної з чотирьох знайдених вершин.

Змінні позитивного відхилення розраховуються за формулами:

$$d_C^+ = \max(0, C(x) - 1300)$$

$$d_\Pi^+ = \max(0, \Pi(x) - 2600)$$

1. Обчислення значень у вершині A(2, 28, 20):

   $C(A) = 20 \cdot 2 + 30 \cdot 28 + 35 \cdot 20 = 40 + 840 + 700 = 1580$ у.о.

   $\Pi(A) = 60 \cdot 2 + 55 \cdot 28 + 45 \cdot 20 = 120 + 1540 + 900 = 2560$ у.о.

   Відхилення:

   $d_C^+ = \max(0, 1580 - 1300) = 280$; $d_\Pi^+ = \max(0, 2560 - 2600) = 0$

   $F_{\text{штраф}}(A) = 280 + 0 = 280$

2. Обчислення значень у вершині B(22, 28, 0):

   $C(B) = 20 \cdot 22 + 30 \cdot 28 + 35 \cdot 0 = 440 + 840 = 1280$ у.о.

   $\Pi(B) = 60 \cdot 22 + 55 \cdot 28 + 45 \cdot 0 = 1320 + 1540 = 2860$ у.о.

   Відхилення:

   $d_C^+ = \max(0, 1280 - 1300) = 0$; $d_\Pi^+ = \max(0, 2860 - 2600) = 260$

   $F_{\text{штраф}}(B) = 0 + 260 = 260$

3. Обчислення значень у вершині C(25, 25, 0):

   $C(C) = 20 \cdot 25 + 30 \cdot 25 + 35 \cdot 0 = 500 + 750 = 1250$ у.о.

   $\Pi(C) = 60 \cdot 25 + 55 \cdot 25 + 45 \cdot 0 = 1500 + 1375 = 2875$ у.о.

   Відхилення:

   $d_C^+ = \max(0, 1250 - 1300) = 0$; $d_\Pi^+ = \max(0, 2875 - 2600) = 275$

   $F_{\text{штраф}}(C) = 0 + 275 = 275$

4. Обчислення значень у вершині D(25, 5, 20):

   $C(D) = 20 \cdot 25 + 30 \cdot 5 + 35 \cdot 20 = 500 + 150 + 700 = 1350$ у.о.

$\Pi(D) = 60 \cdot 25 + 55 \cdot 5 + 45 \cdot 20 = 1500 + 275 + 900 = 2675$ у.о.

Відхилення:

$d_C^+ = \max(0, 1350 - 1300) = 50$; $d_\Pi^+ = \max(0, 2675 - 2600) = 75$

$F_{\text{штраф}}(D) = 50 + 75 = 125$

Порівняємо отримані значення штрафних відхилень:

$$F_{\text{штраф}}(D) = 125 < F_{\text{штраф}}(B) = 260 < F_{\text{штраф}}(C) = 275 < F_{\text{штраф}}(A) = 280$$

Ми аналітично довели, що мінімум цільової функції відхилень досягається у вершині D. Таким чином, оптимальний компромісний план розподілу вагонів за методом цільового програмування: (25; 5; 20). Отриманий за методом цільового програмування оптимальний план повністю збігається з розв'язком, знайденим у підрозділі 2.2 за допомогою методу лінійної згортки критеріїв для випадку, коли інтереси залізниці та клієнтів враховувалися як рівнозначні.

РОЗДІЛ 3. АНАЛІЗ РЕЗУЛЬТАТІВ ОПТИМІЗАЦІЇ

3.1. Дослідження зміни оптимальних планів

Доцільно дослідити, за яких саме критичних значень вагових коефіцієнтів відбувається зміна одного оптимального плану на інший.

Для доведення зобразимо область допустимих розв’язків у вигляді опуклого обмеженого багатогранника графічно.

Враховуючи отримані з попереднього підрозділу нерівності та початкові обмеження, побудуємо область допустимих розв'язків на координатній площині (рис. 2.2).

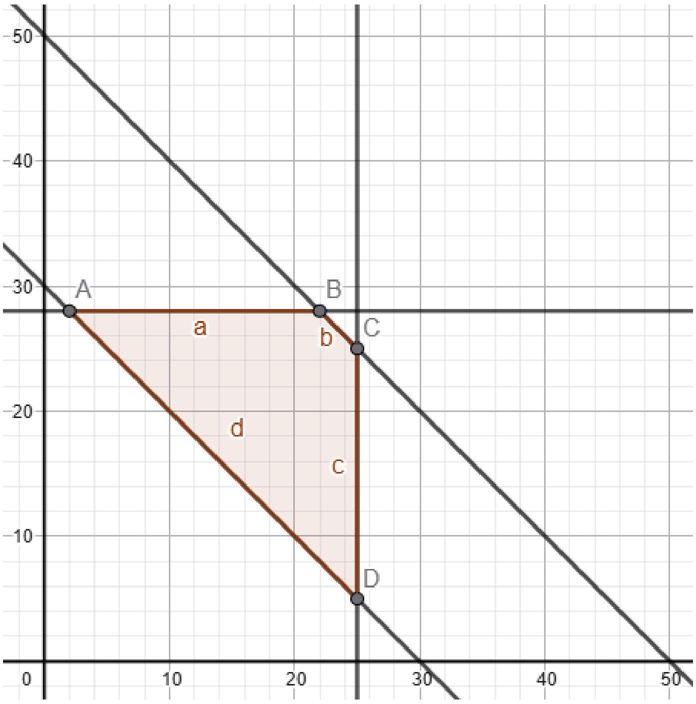

Рисунок 2.2

Координати вершин A(2, 28), C(25, 25), D(25, 5) збігаються з з оптимальними планами, що були знайдені раніше: (2, 28, 20), (25, 25, 0) та (25, 5, 20).

Натомість оптимальний план (22, 28, 0), якому відповідає точка B(22, 28) відсутній в попередніх обчисленнях, тобто в таблиці 2.4, де наведені всі результати.

Щоб дізнатися, чи є цей план ефективним за Парето, перевіримо його за таблицею 2.2. Для значення $x_1 = 22$ у таблиці присутній лише розв'язок № 24 з координатами (22; 8; 20), що не відповідає шуканому плану за двома іншими координатами. Оскільки план (22; 28; 0) у множині ефективних розв'язків відсутній, точка B є домінованою (неефективною за Парето) і виключається з подальшого аналізу.

Розглянемо лінійну згортку наших функцій:

$$F = w_1 * C + w_2 * \Pi$$

Враховуючи умову нормалізації $w_1 + w_2 = 1$, можемо виразити все через один параметр w_1:

$F = w_1(20x_1 + 30x_2 + 35x_3) + (1 - w_1)(60x_1 + 55x_2 + 45x_3) = (60 - 40\ w_1)\ x_1 + (55 - 25\ w_1)x_2 + (45 - 10\ w_1)x_3$

Згрупуємо коефіцієнти при змінних x_i і замість x_i підставляємо значення координат точок, тобто наших оптимальних планів.

$F(A) = (60 - 40\ w_1)\ 2 + (55 - 25\ w_1)28 + (45 - 10\ w_1)20 = 2560 - 980w_1$

$F(C) = (60 - 40\ w_1)\ 25 + (55 - 25\ w_1)25 + (45 - 10\ w_1)0 = 2875 - 1625w_1$

$F(D) = (60 - 40\ w_1)\ 25 + (55 - 25\ w_1)5 + (45 - 10\ w_1)20 = 2675 - 1325w_1$

Тепер введемо функції різниці значень цільових функцій, щоб дізнатися критичне значення вагового коефіцієнта при якому відбувається зміна оптимального плану і відповідно плата.

1. Введемо функцію різниці значень цільових функцій для планів, що відповідають точкам A та D:

   $\Delta_{A-D}(w_1) = F(A) - F(D) = 2560 - 980w_1 - (2675 - 1325w_1) = 345w_1 - 115$

Знаходимо критичне значення (нуль функції): $\Delta_{A-D}(w_1) = 0$

$$345w_1 - 115 = 0 \Rightarrow\ w_1 = \frac{1}{3}$$

Застосування методу інтервалів:

Оскільки ваговий коефіцієнт w_1 може набувати значень лише на відрізку [0; 1], знайдена критична точка $w_1 = \frac{1}{3}$ розбиває цю область на два інтервали. Застосуємо метод інтервалів та визначимо знак функції Δ_{A-D} на кожному з них.

1) Інтервал $w_1 \in \left[0,\ \frac{1}{3}\right]$: $w_1 = 0,\ \Delta_{A-D}(0) = -115 < 0$

2) Інтервал $w_1 \in \left[\frac{1}{3}, 1\right]$: $w_1 = 1,\ \Delta_{A-D}(1) = 230 > 0$

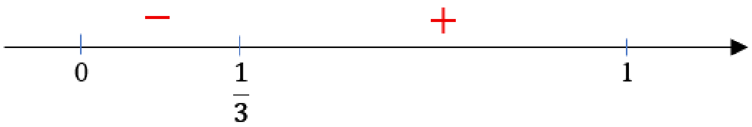

Рисунок 2.3

Знак функції Δ_{A-D} дозволяє визначити оптимальний план:

- Якщо $\Delta_{A-D}(w_1) < 0$, то $F(A) < F(D) \Longrightarrow$ оптимальним є план, що відповідає точці A
- Якщо $\Delta_{A-D}(w_1) > 0$, то $F(A) > F(D) \Longrightarrow$ оптимальним є план, що відповідає точці D

2. Введемо функцію різниці для планів, що відповідають точкам D та C: $\Delta_{D-C}(w_1) = \mathrm{F(D)} - \mathrm{F(C)} = 2675 - 1325w_1 - (2875 - 1625w_1) = 300w_1 - 200$

Знайдемо критичне значення вагового коефіцієнта:

$$300w_1 - 200 = 0 \Longrightarrow\ w_1 = \frac{2}{3}$$

Оскільки для $w_1 < \frac{1}{3}$ оптимальним було визнано план A, застосуємо метод інтервалів для відрізка $\left[\frac{1}{3}, 1\right]$:

1) Інтервал $w_1 \in \left[\frac{1}{3},\ \frac{2}{3}\right]$: $w_1 = \frac{1}{2},\ \Delta_{D-C}\left(\frac{1}{2}\right) = -50 < 0 \Longrightarrow$оптимальним є план, що відповідає точці D

2) Інтервал $w_1 \in \left[\frac{2}{3}, 1\right]$: $w_1 = 1,\ \Delta_{D-C}(1) = 100 > 0 \Longrightarrow$оптимальним є план, що відповідає точці C.

Рисунок 2.4

3. Дослідимо можливість прямої зміни оптимального плану C на план A:

$$\Delta_{C-A}(w_1) = \mathrm{F(C)} - \mathrm{F(A)} = 2875 - 1625w_1 - (2560 - 980w_1) = 315 - 645w_1$$

$$315 - 645w_1 = 0 \Rightarrow \; w_1 = \frac{21}{43}$$

При спробі перейти напряму від оптимального плану, що відповідає точці C до оптимального плану, що відповідає точці A ми отримали значення $w_1 = \frac{21}{43}$.

Ця точка належить інтервалу $\left[\frac{1}{3}, \frac{2}{3}\right]$, який в свою чергу відповідає оптимальному плану точки D. Отже, перейти напряму від оптимального плану A до C неможливо. Перехід між оптимальними планами можливий лише послідовно $A \longrightarrow D \longrightarrow C$.

Перевіримо, до яких інтервалів належать критичні точки.

$w_1 = \frac{1}{3}$:

$$F(A) = 2560 - 980 * \frac{1}{3} = \frac{6700}{3}$$

$$F(D) = 2675 - 1325 * \frac{1}{3} = \frac{6700}{3}$$

Обидві цільові функції мають однаковий результат при підставленні критичної точки, тобто в цій точці ми можемо обрати як оптимальний план A, так і оптимальний план D. Аналогічно і з іншими критичними точками.

(2, 28, 20): $0 \leq w_1 \leq \frac{1}{3}$

(25, 5, 20): $\frac{1}{3} \leq w_1 \leq \frac{2}{3}$

(25, 25, 0): $\frac{2}{3} \leq w_1 \leq 1$

Для перевірки також використовуємо код симплекс-методу, представлений в додатку (с. 30).

ВИСНОВКИ

На основі методу лінійної згортки було знайдено розв´язки для задачі розподілу вагонів по вантажним пунктам. Застосований підхід дав змогу перетворити двокритеріальну модель, що враховувала мінімізацію операційних витрат перевізника та фінансових витрат вантажовласників, у однокритеріальну з єдиною цільовою функцією. Спочатку з усіх можливих розв’язків виділяються Парето-оптимальні. До цих розв’язків застосовується метод лінійної згортки критеріїв, а саме три різні підходи до розрахунку вагових коефіцієнтів: метод розподілу балів, ранжування та попарні порівняння. Крім того, для безпосереднього пошуку компромісного рішення було розглянуто альтернативний підхід - метод цільового програмування.

Обчислення оптимальних планів виконано за допомогою симплекс-методу з використанням стандартних бібліотек середовища Python. Крім цього, створено програмний алгоритм, призначений для визначення області допустимих значень та фронту Парето.

Наступним кроком стало обчислення значень вагових коефіцієнтів, за яких відбувається перемикання між оптимальними планами. Завдяки дослідженню функції різниці методом інтервалів вдалося розрахувати критичні точки та визначити інтервали дії кожного плану. Аналітично доведено неможливість зміни полярних стратегій («пріоритет залізниці» та «пріоритет клієнтів») напряму. Доведено, що перехід завжди має послідовний характер і потребує проходження через проміжний компромісний розв'язок, тобто випадок, коли приорітети враховуються порівну.

Основні результати даної роботи були оприлюднені на науковій конференції [13]. Також кваліфікаційна робота приймала участь у всеукраїнському конкурсі студентських наукових робіт з галузей знань і спеціальностей у 2025-2026 навчальному році (Спеціальність “Прикладна математика”), де здобула ІІ місце.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Koopmans T. C. An analysis of production as an efficient combination of activities. *Activity analysis of production and allocation,* 1951. URL: https://archive.org/details/in.ernet.dli.2015.39951/page/n85/mode/2up (last accessed: 22.05.2026).
2. Kuhn H. W., Tucker A. W. Nonlinear programming. *Traces and emergence of nonlinear programming*. 2013. P. 247–258. URL: https://doi.org/10.1007/978-3-0348-0439-4_11 (last accessed: 25.05.2026).
3. Odu G. O., Charles-Owaba O. E. Review of Multi-criteria Optimization Methods – Theory and Applications. *IOSR Journal of Engineering*. 2013. Vol. 3, No. 10. P. 1–14. URL: https://doi.org/10.9790/3021-031020114 (last accessed: 15.05.2026).
4. Giagkiozis I., Fleming P. J. Methods for multi-objective optimization: An analysis. *Information Sciences*. 2015. Vol. 293. P. 338–350. URL: https://doi.org/10.1016/j.ins.2014.08.071 (last accessed: 22.05.2026).
5. Triantaphyllou E., Shu B., Sanchez S. N., Ray T. Multi-criteria Decision Making: An Operations Research Approach. *Encyclopedia of Electrical and Electronics Engineering* / ed. by J. G. Webster. New York: John Wiley & Sons, 1998. Vol. 15. P. 175–186. URL: https://onlinelibrary.wiley.com/doi/10.1002/047134608X.W3338 (last accessed: 15.05.2026).
6. Roszkowska E. Rank ordering criteria weighting methods – a comparative overview. *Optimum. Studia Ekonomiczne*. 2013. No. 5 (65). P. 14–33. URL: https://doi.org/10.15290/ose.2013.05.65.02 (last accessed: 15.05.2026).
7. Saaty T. L. A scaling method for priorities in hierarchical structures. *Journal of Mathematical Psychology*. 1977. Vol. 15, No. 3. P. 234–281. URL: https://doi.org/10.1016/0022-2496(77)90033-5 (last accessed: 15.05.2026).

8. Charnes A., Cooper W. W. Goal programming and multiple objective optimization; Part 1. *European Journal of Operational Research,* 1977*,* Vol. 1. Pg. 39-54. URL: https://doi.org/10.1016/S0377-2217(77)81007-2 (last accessed: 22.05.2026).
9. Чибісов Ю. В., Шульга Ю. С. Застосування методів багатокритеріальної оптимізації для вирішення задачі розподілу вагонів по вантажним фронтам. *Транспортні системи та технології перевезень*: зб. наук. праць ДНУЗТ ім. акад. В. Лазаряна. 2014. Вип. 7. С. 65–72. URL: https://tstt.ust.edu.ua/article/view/35994 (дата звернення: 28.05.2026).
10. Dantzig G. B. Linear Programming and Extensions. Princeton: Princeton University Press, 2016. 209 p. URL: https://doi.org/10.1515/9781400884179 (last accessed: 29.03.2026).
11. Odu G. O. Weighting Methods for Multi-Criteria Decision Making Technique. *Journal of Applied Sciences and Environmental Management.* 2019. Vol. 23, No. 8. P. 1449–1457. URL: https://doi.org/10.4314/jasem.v23i8.7 (last accessed: 15.05.2026).
12. Schultes J. Multiobjective optimization of shapes using scalarization techniques. Doctoral dissertation, Wuppertal, Bergische Universität Wuppertal, 2024. URL: https://elekpub.bib.uni-wuppertal.de/doi/10.25926/BUW/0-445 (last accessed: 28.05.2026).
13. Syniakina A.V. MATHEMATICAL APPROACHES TO SOLVING MULTI-CRITERIA OPTIMIZATION PROBLEMS. *Сучасні проблеми математики та її застосування в природничих науках та інформаційних технологіях* : тези доповідей XX Міжнародної науково-практичної конференції студентів та молодих вчених (5 - 6 травня 2026 р. м. Харків, Україна) – Харків : ХНУ імені В.Н. Каразіна, 2026. С. 78-81.

ДОДАТКИ

Програмна реалізація симплекс-методу (Python)

```python
import numpy as np
from scipy.optimize import linprog

# параметр зважування
lam = 0.667

# коефіцієнти критеріїв
C = np.array([20, 30, 35])
Pi = np.array([60, 55, 45])

# зважена цільова функція
c = lam * C + (1 - lam) * Pi

A_eq = [[1, 1, 1]]
b_eq = [50]

bounds = [(0, 25), (0, 28), (0, 20)]

result = linprog(
    c,
    A_eq=A_eq,
    b_eq=b_eq,
    bounds=bounds,
    method='highs'
)

if result.success:
    x1, x2, x3 = result.x
    print("Оптимальний план:", result.x)
    C_val = 20*x1 + 30*x2 + 35*x3
    Pi_val = 60*x1 + 55*x2 + 45*x3

    print(f"Витрати залізниці C = {C_val:.2f}")
    print(f"Плата клієнтів П = {Pi_val:.2f}")
else:
    print("Розв'язок не знайдено")
```

Програмна реалізація множини ефективних рішень Парето (Python)

```
import pandas as pd
feasible_solutions = []

for x1 in range(26):
    for x2 in range(29):
        x3 = 50 - x1 - x2
        # Перевірка обмежень для x3
        if 0 <= x3 <= 20:
            # Розрахунок цільових функцій
            C = 20*x1 + 30*x2 + 35*x3
            Pi = 60*x1 + 55*x2 + 45*x3

            feasible_solutions.append({
                'x1': x1,
                'x2': x2,
                'x3': x3,
                'C (Витрати)': C,
                'Pi (Плата)': Pi
            })
df = pd.DataFrame(feasible_solutions)
# 2. Пошук Парето-ефективних рішень
# Сортуємо за першим критерієм (Витрати C - від меншого до більшого)
df_sorted = df.sort_values(by=['C (Витрати)', 'Pi (Плата)'])
pareto_set = []
min_Pi_so_far = float('inf') # Поточне найкраще значення Плати

for index, row in df_sorted.iterrows():
    # Якщо поточне рішення має меншу Плату, ніж усі попередні (дешевші по C)
варіанти,
    # то воно є Парето-ефективним.
    if row['Pi (Плата)'] < min_Pi_so_far:
        pareto_set.append(row)
        min_Pi_so_far = row['Pi (Плата)']
# Створюємо фінальну таблицю
pareto_df = pd.DataFrame(pareto_set)
pareto_df = pareto_df.astype(int)
print("Таблиця Парето-ефективних рішень:")
print(pareto_df.to_string(index=False))
```