

Прискорення роботи Gurobi та CPLEX при розв'язанні задач пошуку найкоротшого циклу чи шляху, що відвідує задану кількість вершин кластерів графа

Микола М. КОРАБЛЬОВ¹, Петро І. СТЕЦЬЮК²

¹) Київський академічний університет, Київ, Україна

E-mail: m.m.korablov@gmail.com

ORCID: <https://orcid.org/0009-0007-3965-2857>

²) Інститут кібернетики ім. В.М. Глушкова НАН України, Київ, Україна

E-mail: stetsyukp@gmail.com

ORCID: <https://orcid.org/0000-0003-4036-2543>

Отримано 20 квітня 2025, у фінальній формі 7 червня 2025

<https://doi.org/10.3842/confKAU.2025.math.01>

Анотація. Однією з найвідоміших та найбільш вивчених задач дослідження операцій є задача комівояжера. Не дивлячись на свою просту постановку, вона знайшла застосування у багатьох різних галузях і по сьогодні є темою активних досліджень пов'язаних як з розробкою нових методів та підходів до її розв'язання, так і зі створенням узагальнених постановок задач і нових математичних моделей для їх формалізації. Дана робота присвячена дослідженню впливу двох видів додаткових обмежень на швидкість роботи програм Gurobi та CPLEX при розв'язанні спеціального класу задач маршрутизації. Наведені постановки задач пошуку найкоротшого циклу і шляху, що відвідує задану кількість вершин кластерів графа. Описані математичні моделі змішаного цілочислового лінійного програмування для їх розв'язання на основі обмежень Міллера, Такера і Земліна та обмежень Гевіша і Грейвса. Наведені результати обчислювальних експериментів, які демонструють вплив додаткових обмежень на швидкість розв'язання задач даними моделями.

Ключові слова: узагальнені задачі маршрутизації; задача комівояжера; найкоротший цикл та шлях у графі; змішане цілочислове лінійне програмування; AMPL; Gurobi; CPLEX; NEOS Server.

1 Вступ

Задача комівояжера (Travelling Salesman Problem, TSP) є однією з найвідоміших задач дослідження операцій, у своїй найпростішій постановці вона звучить наступним чином: знайти такий найкоротший можливий маршрут відвідування заданого набору локацій, який відвідує кожна з них лише один раз та повертається до початкової локації. Ця задача знайшла велику кількість застосувань як у сферах пов'язаних з логістикою (планування транспортування, маршрутизація транспортних засобів, проектування мереж та ін.), так і у значно більш неочікуваних застосуваннях (картування геномів у генетичних дослідженнях, рентгівська кристалографія, виробництво електронних плат, планування робочих графіків для наземних і космічних телескопів, капітальний ремонт газотурбінних двигунів та ін.) [5, 10].

У зв'язку із застосовністю при розв'язанні багатьох прикладних задач, задача комівояжера є однією з найбільш досліджених задач даної галузі математики. Але, незважаючи на інтуїтивно зрозумілу постановку, через свою NP-повноту досі залишається однією з найскладніших для розв'язання. За останні декілька століть дослідниками були запропоновані різні математичні моделі її формалізації та підходи до її розв'язання, які можна розділити

на дві великі групи: моделі та методи для пошуку точного розв'язку задачі та евристичні і мета-евристичні методи для пошуку наближеного розв'язку [9].

У багатьох ситуаціях виникає потреба обов'язково враховувати спеціальну структуру задачі або ж деякі обмеження спеціального виду, продиктовані предметною областю, що досліджується. У таких випадках класичної задачі комівояжера недостатньо і для досягнення бажаних результатів необхідно розробляти узагальнені постановки задач та математичні моделі їхньої формалізації. Активний розвиток обчислювальної техніки протягом останніх десятиліть сприяв створенню спеціалізованого програмного забезпечення, яке дозволяє використовувати розроблені моделі у промислових сценаріях, зокрема і для розв'язання задач великих розмірів.

Дана робота присвячена емпіричному дослідженню впливу спеціальних надлишкових обмежень на швидкість роботи двох програм для отримання точних розв'язків задач змішаного цілочислового лінійного програмування при розв'язанні спеціального класу задач маршрутизації. У параграфі 2 наведено короткий огляд найвідоміших моделей для розв'язання звичайної задачі комівояжера та розглянуто деякі її узагальнення. Параграф 3 присвячений опису задач пошуку найкоротшого циклу та найкоротшого шляху, що відвідують задану кількість вершин кластерів графа, наведені математичні моделі змішаного цілочислового лінійного програмування для їх розв'язання. Спеціальні обмеження для прискорення методів гілок та меж, які використовуються програмами Gurobi та CPLEX, наведені у параграфі 4 разом з результатами обчислювальних експериментів.

2 Огляд літератури

Розглянемо математичну постановку класичної задачі комівояжера. Нехай матриця $D_{(n,n)}$ задає повний граф на n вершинах, а її елементи $d_{ij} \geq 0$ позначають довжину ребра між вершинами i та j . Будемо вимагати, щоб граф не містив петель — тобто щоб $d_{ii} = 0 \forall i \in V$, де $V = \{1, 2, \dots, n\}$ позначає множину всіх вершин графа. Якщо необхідно врахувати несиметричність відстаней між вершинами графу, то це можна зробити присвоєнням різних значень елементам d_{ij} та d_{ji} . Без втрати загальності оберемо вершину 1 початковою. Тоді задача комівояжера полягає у пошуку найкоротшого циклу, який розпочинається у вершині 1, відвідує кожную вершину графа рівно один раз та повертається до стартової вершини.

Для формального опису цієї задачі при знаходженні її точних розв'язків часто використовують змішане цілочислове лінійне програмування (mixed integer linear programming, MILP). Задачі такого виду засновані на задачі про призначення і виглядають наступним чином:

$$d_{\text{tsp}}^* = \min_{x_{ij}} \left\{ d_{\text{tsp}}(x) = \sum_{i=1}^n \sum_{j=1, j \neq i}^n d_{ij} x_{ij} \right\}, \quad (1)$$

$$\sum_{j=1, j \neq i}^n x_{ij} = 1, \quad \sum_{j=1, j \neq i}^n x_{ji} = 1, \quad i \in V, \quad (2)$$

$$x_{ij} = 0 \vee 1, \quad i, j \in V, \quad i \neq j, \quad (3)$$

$$\text{обмеження, що забезпечують зв'язність циклу}, \quad (4)$$

$$\text{обмеження на додаткові змінні (якщо вони є)}. \quad (5)$$

У задачах даного типу використовуються булеві змінні x_{ij} (3) які набувають значення 1, якщо відповідне ребро, яке з'єднує вершини i та j , наявне у циклі, інакше — 0. Обмеження (2) забезпечують одноразовий вихід з та вхід у кожную з вершин графа, і разом зі змінними (3) утворюють задачу про призначення. Але просто задачі про призначення недостатньо для того, щоб забезпечити зв'язність отриманого циклу. Тому додатково вводять

обмеження виду (4) та (5), які разом з (2) та (3) дозволяють описати всі допустимі розв'язки. Розв'язанню задачі комівояжера ж відповідає знаходження такого розв'язку, який мінімізує загальну довжину маршруту (1).

Дослідники по-різному підходили до визначення обмежень для гарантування зв'язності отриманого циклу, серед перших буди Данциг, Фулкерсон і Джонсон які у роботі [6] запропонували обмеження наступного виду, які не потребують введення додаткових змінних:

$$\sum_{i,j \in S} x_{ij} \leq |S| - 1, \quad S \subset V, \quad 2 \leq |S| \leq n - 2, \quad (6)$$

де $S \subset V$, $2 \leq |S| \leq n - 2$ це всі можливі підмножини множини всіх вершин графа. Використання обмежень (6) на практиці є обчислювально складним, адже потребує повного перебору всіх можливих підмножин на кожній ітерації алгоритму, який розв'язує задачу комівояжера.

У роботі [11] Міллером, Такером і Земліном були запропоновані наступні обмеження (7), які використовують додаткові змінні u (8), що задають порядок відвідування вершин та можуть набувати як додатних цілих, так і невід'ємних дійсних значень (у даній роботі розглядається перший випадок):

$$u_i - u_j + nx_{ij} \leq n - 1, \quad i, j \in \{2, \dots, n\}, \quad i \neq j, \quad (7)$$

$$u_i \in \{1, \dots, n - 1\}, \quad i \in \{2, \dots, n\}. \quad (8)$$

Дані обмеження мають наступний зміст. Якщо $x_{ij} = 0$, тобто ребро, яке з'єднує вершини i та j , не входить до циклу — то відповідні змінні u_i та u_j можуть набувати довільних значень з множини $\{1, \dots, n - 1\}$. Якщо ж $x_{ij} = 1$, тобто ребро, яке з'єднує вершини i та j , входить до циклу — то відповідні змінні u_i та u_j повинні задовольняти нерівності $u_i \leq u_j - 1$. Так як в такому випадку вершина j слідує одразу за вершиною i і ці змінні задають порядок відвідування вершин, то повинна досягатися рівність $u_i = u_j - 1$, що накладає обмеження на допустимі значення u_i та u_j і задає порядок відвідування всіх вершин графа.

Інший підхід був запропонований Гевішом та Грейвсом у роботі [8], де зв'язність розв'язку забезпечується з допомогою обмежень (9)–(11) наступного виду, які засновані на дійсних невід'ємних змінних z (12) що задають потік деякого абстрактного товару по мережі:

$$z_{ij} - (n - 1)x_{ij} \leq 0, \quad i, j \in \{1, \dots, n\}, \quad i \neq j, \quad (9)$$

$$\sum_{j=2}^n z_{1j} = n - 1, \quad \sum_{j=2}^n z_{j1} = 0, \quad (10)$$

$$\sum_{j=1, j \neq i}^n z_{ij} - \sum_{j=1, j \neq i}^n z_{ji} = -1, \quad i \in \{2, \dots, n\}, \quad (11)$$

$$z_{ij} \geq 0, \quad i, j \in \{1, \dots, n\}, \quad i \neq j. \quad (12)$$

Дані обмеження мають наступний зміст. Обмеження (9) гарантують перевезення продукту між вершинами i та j тільки тоді, коли $x_{ij} = 1$, обмеження (10) означають, що із початкової вершини 1 потрібно виїхати із $n - 1$ одиницями продукту та повернутися в неї із 0 одиницями продукту, а обмеження (11) означають, що у вершині i потрібно залишати рівно одну одиницю продукту.

Окрім розробки нових математичних моделей для звичайної задачі комівояжера також триває і робота над формулюванням нових, узагальнених задач маршрутизації та математичних моделей для їх розв'язання. Наприклад, у роботах [2, 14] сформульовані задачі знаходження найкоротшого циклу або шляху в графі, які проходять через задану кількість

вершин графа за умови, що лише деякі з цих k вершин задані обов'язковими для відвідування. Для задання таких умов на маршрут використовуються булеві змінні $y_i = 0 \vee 1$, які набувають значення 1, якщо вершина i входить до маршруту, інакше — 0.

Інша модифікація задачі комівояжера під назвою “Clustered Travelling Salesman Problem” (CTSP) представлена у роботі [4]. У ній розглядається ситуація, коли вершини графа розділені на кластери, і вершини кожного кластеру потрібно відвідати послідовно у деякому порядку перед тим, як розпочинати відвідування вершин деякого іншого кластеру. При пошуку оптимального маршруту за даних умов необхідно одночасно визначити і порядок відвідування кожного з кластерів, і порядок відвідування всіх вершин кожного кластеру.

У роботах [12,13] запропоновано інше узагальнення задачі комівояжера під назвою “Generalized Travelling Salesman Problem” (GTSP) та ефективні методи його розв'язання. У даній постановці задачі всі вершини графа також розбиваються на кластери які, в загальному випадку, можуть перетинатися — і задача полягає у знаходженні найкоротшого маршруту який відвідує рівно по одній вершині з кожного кластеру.

3 Постановки задач та математичні моделі, що розглядаються

Наведемо постановки узагальнених задач оптимальної маршрутизації, які були запропоновані та досліджені у роботах [1, 15, 16]. Перша з них описує задачу пошуку найкоротшого циклу при обмеженнях на кількість вершин з кластерів графа для відвідування, а друга — задачу пошуку найкоротшого шляху за схожих обмежень.

Для опису математичних моделей для розв'язання цих задач введемо наступні додаткові позначення для підмножин множини всіх вершин графа V : $V_{sf} = V \setminus \{s, f\}$, $V_s = V \setminus \{s\}$, $V_f = V \setminus \{f\}$, $V_i = V \setminus \{i\}$ та $V_j = V \setminus \{j\}$.

3.1 Задача пошуку найкоротшого циклу, що відвідує задану кількість вершин кластерів графа

Нехай матриця $D_{(n,n)}$ так само, як і в параграфі 2, задає повний граф на n вершинах. Оберемо одну вершину з множини V , яка буде початковою, та позначимо її s — а всі інші вершини графа розділимо на K кластерів $C_1, C_2, \dots, C_K$ таким чином, щоб $\sum_{k=1}^K |C_k| = n-1$ та $C_p \cap C_q = \emptyset$ для $p \neq q$, де $|C_k|$ позначає кількість елементів множини C_k , $\forall k \in \{1, \dots, K\}$. Введемо параметри m_k , $0 \leq m_k \leq |C_k|$, що будуть позначати кількість різних вершин, які необхідно відвідати з кожного кластера C_k .

При такій розширеній постановці можна сформулювати складнішу задачу пошуку оптимального циклу: знайти найкоротший можливий цикл у графі $D_{(n,n)}$ який розпочинається у вершині s , відвідує рівно m_k разів кожен кластер C_k без відвідування тієї ж вершини більше одного разу (тобто відвідуючи рівно $M = \sum_{k=1}^K m_k$ різних вершин з $n-1$ наявних у графі) та повертається у вершину s .

3.2 Модель на основі обмежень Міллера, Такера і Земліна

Перша модель для розв'язання задачі пошуку найкоротшого циклу, що відвідує задану кількість вершин кластерів графа, базується на обмеженнях Міллера, Такера і Земліна, версії яких для моделювання звичайної задачі комівояжера були описані вище.

Введемо такі змінні та їх позначення: булеві змінні $x_{ij} = 0 \vee 1$, $i, j \in V$, $i \neq j$, кожна з яких дорівнює 1, якщо відповідне ребро, яке з'єднує вершини i та j , наявне у циклі, інакше нулю; булеві змінні $y_i = 0 \vee 1$, $i \in V \setminus \{s\}$, кожна з яких дорівнює 1, якщо відповідна вершина i відвідується циклом, інакше нулю; цілочислові змінні $u_i \in \mathbb{Z}^+$, $1 \leq u_i \leq M$, $i \in V \setminus \{s\}$, що набувають значень від 1 до M та позначають крок, на якому відвідується відповідна

вершина i . MILP задача на основі обмежень Міллера, Такера і Земліна для розв'язання сформульованої задачі пошуку найкоротшого циклу має наступний вигляд:

$$d_1^* = \min_{x_{ij}, y_i, u_i} \left\{ d_1(x) = \sum_{i \in V} \sum_{j \in V_i} d_{ij} x_{ij} \right\}, \quad (13)$$

$$\sum_{j \in V_s} x_{sj} = 1, \quad \sum_{j \in V_s} x_{js} = 1, \quad (14)$$

$$\sum_{j \in V_i} x_{ij} = y_i, \quad \sum_{j \in V_i} x_{ji} = y_i, \quad i \in V_s, \quad (15)$$

$$\sum_{i \in C_k} y_i = m_k, \quad k \in \{1, \dots, K\}, \quad (16)$$

$$u_i - u_j + Mx_{ij} \leq M - 1, \quad i, j \in V_s, \quad i \neq j, \quad (17)$$

$$x_{ij} = 0 \vee 1, \quad i, j \in V, \quad i \neq j, \quad (18)$$

$$y_i = 0 \vee 1, \quad i \in V_s, \quad (19)$$

$$u_i \in \mathbb{Z}^+, \quad 1 \leq u_i \leq M, \quad i \in V_s. \quad (20)$$

У роботі [16] доведена теорема про необхідність та достатність обмежень (14)–(20) наведеної вище моделі (13)–(20) для розв'язання описаної задачі.

3.3 Модель на основі обмежень Гевіша та Грейвса

Друга модель для розв'язання задачі пошуку найкоротшого циклу, що відвідує задану кількість вершин кластерів графа, базується на обмеженнях Гевіша та Грейвса, версії яких для моделювання звичайної задачі комівояжера були описані вище.

Введемо такі змінні та їх позначення: булеві змінні $x_{ij} = 0 \vee 1, i, j \in V, i \neq j$ та булеві змінні $y_i = 0 \vee 1, i \in V \setminus \{s\}$ такі, як і в моделі, описаній у параграфі 3.2; дійсні невід'ємні змінні $z_{ij} \geq 0, i, j \in V, i \neq j$, які задають величину потоку деякого абстрактного товару по мережі. MILP задача на основі обмежень Гевіша та Грейвса для розв'язання сформульованої задачі пошуку найкоротшого циклу має наступний вигляд:

$$d_2^* = \min_{x_{ij}, y_i, z_{ij}} \left\{ d_2(x) = \sum_{i \in V} \sum_{j \in V_i} d_{ij} x_{ij} \right\}, \quad (21)$$

$$\sum_{j \in V_s} x_{sj} = 1, \quad \sum_{j \in V_s} x_{js} = 1, \quad (22)$$

$$\sum_{j \in V_i} x_{ij} = y_i, \quad \sum_{j \in V_i} x_{ji} = y_i, \quad i \in V_s, \quad (23)$$

$$\sum_{i \in C_k} y_i = m_k, \quad k \in \{1, \dots, K\}, \quad (24)$$

$$z_{ij} - Mx_{ij} \leq 0, \quad i, j \in V, \quad i \neq j, \quad (25)$$

$$\sum_{j \in V_s} z_{sj} = M, \quad \sum_{j \in V_s} z_{js} = 0, \quad (26)$$

$$\sum_{j \in V_i} z_{ij} - \sum_{j \in V_i} z_{ji} = -y_i, \quad i \in V_s, \quad (27)$$

$$x_{ij} = 0 \vee 1, \quad i, j \in V, \quad i \neq j, \quad (28)$$

$$y_i = 0 \vee 1, \quad i \in V_s, \quad (29)$$

$$z_{ij} \geq 0, \quad i, j \in V, \quad i \neq j. \quad (30)$$

У роботі [16] доведена теорема про необхідність та достатність обмежень (22)–(30) наведеної вище моделі (21)–(30) для розв’язання описаної задачі. А у роботі [1] розглянуто комбіновану модель для розв’язання задачі, що розглядається. Така модель використовує одночасно обмеження (17) та обмеження (25)–(27) для забезпечення зв’язності отриманого розв’язку та відсутності підциклів у ньому.

3.4 Задача пошуку найкоротшого шляху, що відвідує задану кількість вершин кластерів графа

Нехай, як і в попередній постановці задачі у параграфі 3.1, матриця $D_{(n,n)}$ задає повний граф на n вершинах. Оберемо дві вершини з множини V , одна з яких буде початковою (позначимо її s), а інша — кінцевою (позначимо її f). Всі інші вершини графа розділимо на K кластерів $C_1, C_2, \dots, C_K$ таким чином, щоб $\sum_{k=1}^K |C_k| = n - 2$ та $C_p \cap C_q = \emptyset$ для $p \neq q$, де $|C_k|$ позначає кількість елементів множини C_k , $\forall k \in \{1, \dots, K\}$. Як і перед цим, введемо параметри m_k , $0 \leq m_k \leq |C_k|$, що будуть позначати кількість різних вершин, які необхідно відвідати з кожного кластера C_k .

При такій розширеній постановці можна сформулювати складнішу задачу пошуку оптимального шляху: знайти найкоротший можливий шлях у графі $D_{(n,n)}$, який розпочинається у вершині s , відвідує рівно m_k разів кожен кластер C_k без відвідування тієї ж вершини більше одного разу (тобто відвідуючи рівно $M = \sum_{k=1}^K m_k$ різних вершин з $n - 2$ наявних у графі) та закінчується у вершині f .

3.5 Модель на основі обмежень Міллера, Такера і Земліна

Перша модель для розв’язання задачі пошуку найкоротшого шляху, що відвідує задану кількість вершин кластерів графа, базується на обмеженнях Міллера, Такера і Земліна, версії яких для моделювання звичайної задачі комівояжера були описані вище.

Введемо такі змінні та їх позначення: булеві змінні $x_{ij} = 0 \vee 1$, $i, j \in V$, $i \neq j$ та булеві змінні $y_i = 0 \vee 1$, $i \in V_{sf}$ такі, як і в моделі, описаній у параграфі 3.2; цілочислові змінні $u_i \in \mathbb{Z}^+$, $1 \leq u_i \leq M$, $i \in V_{sf}$ такі, як і в моделі, описаній у параграфі 3.2. MILP задача на основі обмежень Міллера, Такера і Земліна для розв’язання сформульованої задачі пошуку найкоротшого шляху має наступний вигляд:

$$d_3^* = \min_{x_{ij}, y_i, u_i} \left\{ d_3(x) = \sum_{j \in V_s} d_{sj} x_{sj} + \sum_{i \in V_{sf}} \sum_{j \in V_{sf}} d_{ij} x_{ij} + \sum_{i \in V_f} d_{if} x_{if} \right\}, \quad (31)$$

$$\sum_{j \in V_s} x_{sj} = 1, \quad \sum_{j \in V_s} x_{js} = 0, \quad (32)$$

$$\sum_{j \in V_f} x_{jf} = 1, \quad \sum_{j \in V_f} x_{fj} = 0, \quad (33)$$

$$\sum_{j \in V_i} x_{ij} = y_i, \quad \sum_{j \in V_i} x_{ji} = y_i, \quad i \in V_{sf}, \quad (34)$$

$$\sum_{i \in C_k} y_i = m_k, \quad k \in \{1, \dots, K\}, \quad (35)$$

$$u_i - u_j + M x_{ij} \leq M - 1, \quad i \in V_{sf}, \quad j \in V_{sf}, \quad i \neq j, \quad (36)$$

$$x_{ij} = 0 \vee 1, \quad i, j \in V, \quad i \neq j, \quad (37)$$

$$y_i = 0 \vee 1, \quad i \in V_{sf}, \quad (38)$$

$$u_i \in \mathbb{Z}^+, \quad 1 \leq u_i \leq M, \quad i \in V_{sf}. \quad (39)$$

3.6 Модель на основі обмежень Гевіша та Грейвса

Друга модель для розв'язання задачі пошуку найкоротшого шляху, що відвідує задану кількість вершин кластерів графа, базується на обмеженнях Гевіша та Грейвса, версії яких для моделювання звичайної задачі комівояжера були описані вище.

Введемо такі змінні та їх позначення: булеві змінні $x_{ij} = 0 \vee 1$, $i, j \in V$, $i \neq j$ та булеві змінні $y_i = 0 \vee 1$, $i \in V_{s,f}$ такі, як і в моделі, описаній у параграфі 3.2; дійсні невід'ємні змінні $z_{ij} \geq 0$, $i, j \in V$, $i \neq j$ такі, як і в моделі, описаній у параграфі 3.3. MILP задача на основі обмежень Гевіша та Грейвса для розв'язання сформульованої задачі пошуку найкоротшого шляху має наступний вигляд:

$$d_4^* = \min_{x_{ij}, y_i, z_{ij}} \left\{ d_4(x) = \sum_{j \in V_s} d_{sj} x_{sj} + \sum_{i \in V_{sf}} \sum_{j \in V_{sf}} d_{ij} x_{ij} + \sum_{i \in V_f} d_{if} x_{if} \right\}, \quad (40)$$

$$\sum_{j \in V_s} x_{sj} = 1, \quad \sum_{j \in V_s} x_{js} = 0, \quad (41)$$

$$\sum_{j \in V_f} x_{jf} = 1, \quad \sum_{j \in V_f} x_{fj} = 0, \quad (42)$$

$$\sum_{j \in V_i} x_{ij} = y_i, \quad \sum_{j \in V_i} x_{ji} = y_i, \quad i \in V_{sf}, \quad (43)$$

$$\sum_{i \in C_k} y_i = m_k, \quad k \in \{1, \dots, K\}, \quad (44)$$

$$z_{ij} - M x_{ij} \leq 0, \quad i, j \in V, \quad i \neq j, \quad (45)$$

$$\sum_{j \in V_s} z_{sj} = M, \quad \sum_{j \in V_f} z_{jf} = 0, \quad (46)$$

$$\sum_{j \in V_i} z_{ij} - \sum_{j \in V_i} z_{ji} = -y_i, \quad i \in V_{sf}, \quad (47)$$

$$x_{ij} = 0 \vee 1, \quad i, j \in V, \quad i \neq j, \quad (48)$$

$$y_i = 0 \vee 1, \quad i \in V_{sf}, \quad (49)$$

$$z_{ij} \geq 0, \quad i, j \in V, \quad i \neq j. \quad (50)$$

У роботі [15] розглянуто комбіновану модель для розв'язання задачі, що розглядається. Така модель використовує одночасно обмеження (36) та обмеження (45)–(47) для забезпечення зв'язності отриманого розв'язку та відсутності підциклів у ньому. Описані у параграфах 3.5 та 3.6 моделі змішаного цілочислового лінійного програмування були отримані із комбінованої моделі вилученням певного типу обмежень.

4 Додаткові обмеження та обчислювальні експерименти

У роботі [3] показано вплив двох типів додаткових обмежень у формі лінійних нерівностей, які при розв'язанні задачі комівояжера були додані до моделі на основі обмежень Гевіша та Грейвса з метою прискорення методів гілок та меж, які використовуються програмами Gurobi [18] та CPLEX [17].

Перший тип лінійних обмежень-нерівностей приймає вигляд:

$$x_{ij} + x_{ji} \leq 1, \quad i, j \in V, \quad i \neq j, \quad (51)$$

та задає умову, що по кожному ребру $i - j$ у графі можна пересуватися лише в одному довільному напрямку.

Другий тип обмежень для задач що розглядаються у даній роботі приймає вигляд:

$$\sum_{i=1}^n \sum_{j=1, j \neq i}^n x_{ij} \leq M + 1, \quad (52)$$

та задає умову, що отриманий цикл або шлях має містити не більше, ніж $M + 1$ дуг.

Метою обчислювальних експериментів є дослідження впливу додаткових обмежень (51) та (52) на швидкість роботи програм Gurobi та CPLEX, наявних на NEOS сервері [19], при розв’язанні тестових задач пошуку найкоротших циклів та шляхів, описаних у параграфі 3.

Тестові задачі для експериментів було сформовано на графі *st70.tsp*, який містить 70 вершин, з бібліотеки TSPLIB [20]. Кожна вершина задана порядковим номером i та набором координат виду (x_i, y_i) . Відстані між усіма вершинами в графі вважаються симетричними та обчислюються за формулою:

$$d_{ij} = d_{ji} = \text{round}\left(\sqrt{(x_j - x_i)^2 + (y_j - y_i)^2}\right) \quad \forall i, j = 1, \dots, 70.$$

Для формування тестової задачі пошуку оптимального циклу даний граф був розділений на 5 неперетинних кластерів $C_1, \dots, C_4, C_{5.1}$ і стартову вершину s , а для тестової задачі пошуку оптимального шляху — на 5 неперетинних кластерів $C_1, \dots, C_4, C_{5.2}$, стартову вершину s і кінцеву вершину f . Ці розбиття графу наведені у таблиці 1 разом із значеннями m_k кількості вершин з кожного кластеру для відвідування.

Стартова вершина $s = 53$, кінцева вершина $f = 17$		
Кластери C_k	Вершини у кластерах	m_k
C_1	11, 33, 48, 51, 54, 56, 62, 64, 65, 67	3
C_2	1, 13, 22, 23, 29, 31, 35, 36, 38, 59, 63, 66, 69, 70	4
C_3	2, 3, 7, 8, 19, 26, 28, 32, 49, 55	3
C_4	9, 25, 27, 39, 40, 45, 46, 61	3
$C_{5.1}$	$\{1, \dots, 70\} \setminus \{s \cup C_1 \cup C_2 \cup C_3 \cup C_4\}$	8
$C_{5.2}$	$\{1, \dots, 70\} \setminus \{s \cup f \cup C_1 \cup C_2 \cup C_3 \cup C_4\}$	8

Табл. 1. Розбиття графу *st70.tsp* на кластери та постановки тестових задач.

Досліджувався вплив додаткових обмежень при пошуку оптимального циклу з використанням моделі на основі обмежень Міллера, Такера і Земліна (13)–(20), моделі на основі обмежень Гевіша та Грейвса (21)–(30) та комбінованої моделі, яка використовує обидва типи обмежень (13)–(30), та вплив цих додаткових обмежень при пошуку оптимального шляху з використанням, відповідно, моделей (31)–(39), (40)–(50) та комбінованої моделі (31)–(50). Кожне поєднання виду “модель + додаткове обмеження” було використане десять разів для розв’язання тестових задач кожною з двох програм, отриманий час розв’язання був усереднений. Отримані оптимальні розв’язки тестових задач пошуку оптимального циклу та шляху наведені на рисунках 1 та 2 у додатку А, відповідно. Детальні результати обчислювальних експериментів наведені в додатку Б у таблицях 2 та 3, відповідно.

Як бачимо з наведених у таблицях результатів, для більшості MILP задач використання розглянутих додаткових обмежень дозволяє прискорити роботу програм Gurobi і CPLEX при розв’язанні тестових задач.

А саме, для задачі (13)–(20) додавання обмежень першого типу прискорює отримання розв’язку програмою Gurobi в середньому приблизно у 1,6 разів (28432 секунд проти

45509 секунд), а от програма CPLEX так і не змогла розв'язати цю тестову задачу за максимальний допустимий час. Для задачі (21)–(30) додаткові обмеження впливають на час роботи обох програм, Gurobi завдяки додаванню обох типів обмежень розв'язує задачу в середньому приблизно у 1,36 разів швидше (118 секунд проти 160 секунд), а CPLEX завдяки обмеженням другого типу — в середньому у 2 рази швидше (388 секунд проти 776 секунд). А от при розв'язанні комбінованої задачі (13)–(30) додавання цих обмежень не мало бажаного впливу, в середньому Gurobi та CPLEX найшвидше справляються без них (283 секунди та 764 секунди, відповідно).

Для задачі (31)–(39) додавання обмежень також не мало бажаного результату. Найшвидше в середньому програма Gurobi розв'язувала задачу без них (31523 секунд), а програма CPLEX не спромоглася розв'язати задачу за максимальний допустимий час ані з додатковими обмеженнями, ані без них. Для задачі (40)–(50) додавання обмежень першого типу дозволило в середньому прискорити роботу програми Gurobi приблизно у 1,1 рази (183 секунд проти 203 секунд), а от на програму CPLEX ці обмеження знову не мали бажаного впливу — найшвидше в середньому розв'язувалася задача без них (828 секунд). Для задачі (31)–(50) додаткові обмеження також не мали впливу на програму CPLEX (2232 секунди без них), а от програму Gurobi вдалося в середньому прискорити приблизно у 4 рази завдяки використанню обох типів обмежень (114 секунд проти 457 секунд).

5 Висновки

У даній роботі було досліджено, як використання двох типів додаткових обмежень впливає на швидкість розв'язання спеціального класу задач оптимальної маршрутизації при використанні програм Gurobi та CPLEX. Було проведено обчислювальні експерименти з розв'язання однакових тестових задач, сформованих на графі *st70.tsp*, з допомогою обох програм, використовуючи для цього різні моделі змішаного цілочислового лінійного програмування, без та з додатковими обмеженнями.

Для отримання усередненого часу роботи програм кожне поєднання виду “модель + додаткове обмеження” запускалося десять разів для розв'язання кожною з двох програм. Результати свідчать про те, що при розв'язанні розглянутих тестових задач роботу програм Gurobi та CPLEX можна прискорити з допомогою додаткових обмежень у 1,1–4 рази.

Фінансування

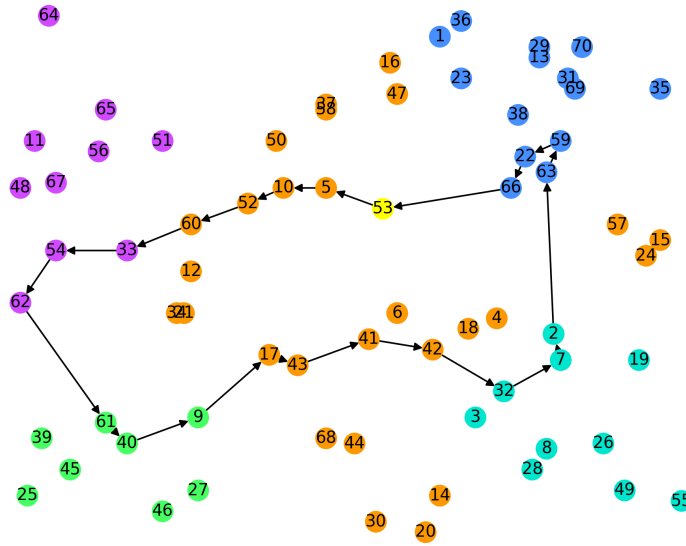
Робота підтримана НДР “Розробити методи негладкої оптимізації для спеціальних задач штучного інтелекту” (0124U001061).

Список використаних джерел

- [1] Стецюк П.І., Корабльов М.М., Стоян О.О., Губернатор О.А., Михайленко О.В., [Комбінована модель знаходження найкоротшого циклу проходження заданої кількості вершин кластерів графа: приклад застосування для пішохідного туризму](#), *Міжнародний науково-технічний журнал “Проблеми керування та інформатики”* 69 (2024), по. 4, 5–27.
- [2] Стецюк П.І., Соломон Д.І., Григорак М.Ю., [Задачі про найкоротші \$k\$ -вершинні цикли та шляхи](#), *Кібернетика та комп'ютерні технології* (2021), по. 3. 15–33.
- [3] Стецюк П.І., Супрун А.А., Прискорення Gurobi та CPLEX для задачі комівояжера, Міжнародний сателітний симпозіум “Інтелектуальні рішення – 2021-С”, 29 вересня 2021, Київ, Україна.
- [4] Chisman J.A., [The clustered traveling salesman problem](#), *Comput. Oper. Res.* **2** (1975), 115–119.
- [5] Cook W.J., [In pursuit of the traveling salesman: mathematics at the limits of computation](#), Princeton University Press, 2012.

- [6] Dantzig G.B., Fulkerson D.R., Johnson S.M., [Solution of a large-scale travelling-salesman problem](#), *J. Oper. Res. Soc. America* **2** (1954), 393–410.
- [7] Fourer R., Gay D.M., Kernighan B.W. AMPL: A modeling language for mathematical programming, Duxbury Press, Pacific Grove, 2002.
- [8] Gavish B., Graves S.C. The travelling salesman problem and related problems, *Working Paper OR-078-78*, Operations Research Center, MIT, Cambridge, MA. 1978, p. 33.
- [9] Laporte G., [The travelling salesman problem: An overview of exact and approximate algorithms](#), *European J. Oper. Res.* **59** (1992), 231–247.
- [10] Matai R., Singh S.P., Mittal M.L., Travelling salesman problem: An overview of applications, formulations, and solution approaches, in D. Davendra (Ed.), *Traveling Salesman Problem, Theory and Applications*, InTech, 2010, pp. 1–24.
- [11] Miller C.E., Tucker A.W., Zemlin R.A., [Integer programming formulation of travelling salesman problem](#), *J. ACM.* **3** (1960), 326–329.
- [12] Noon C.E., The generalized travelling salesman problem, Ph.D. thesis, The University of Michigan, Department of Industrial and Operations Engineering, 1988.
- [13] Noon C.E., Bean J.C., [A lagrangian based approach for the asymmetric generalized travelling salesman problem](#), *Oper. Res.* **39** (1991), 623–632.
- [14] Stetsyuk P.I., [Problem statements for \$k\$ -node shortest path and \$k\$ -node shortest cycle in a complete graph](#), *Cybern. Syst. Anal.* **52** (2016), no. 1, 71–75.
- [15] Stetsyuk P.I., Korablov M.M., A combined mathematical model for finding the shortest path to visit a given number of vertices from the graph clusters with special constraints International Scientific Symposium “Intelligent Solutions” (IntSol-2025), Uzhhorod, Ukraine, May 1–5, 2025 (accepted for publication).
- [16] Stetsyuk P.I., Korablov M.M., Stoian O.O., Hubernator O.A., Mathematical models for finding the shortest cycle to visit a given number of vertices from the graph clusters, *3rd International Conference on Problems of Logistics, Management and Operation in the East-West Transport Corridor (PLMO 2024)*, May 15–17, 2024, Baku, Azerbaijan.
- [17] IBM, CPLEX User's Manual, Version 12 Release 8, available at <https://www.ibm.com/docs/en/icos/22.1.2?topic=optimizers-users-manual-cplex>.
- [18] Gurobi Optimization Inc., Gurobi Optimizer Reference Manual, available at <https://docs.gurobi.com/current/>.
- [19] NEOS Server, State-of-the-Art Solvers for Numerical Optimization, available at <https://neos-server.org/neos/>.
- [20] A Traveling Salesman Problem Library, available at <http://comopt.ifl.uni-heidelberg.de/software/TSPLIB95/>

А Оптимальні розв'язки тестових задач



Б Детальні результати обчислювальних експериментів

У таблицях 2 та 3 наведені детальні результати обчислювальних експериментів, описаних у параграфі 4. У колонках $t_{\min}$, t_{avg} і $t_{\max}$ наведено відповідно мінімальний, середній та максимальний час розв’язання після 10 запусків. Тут і далі символом “—” позначено ситуацію, коли програма припиняла свою роботу та виводила повідомлення “ERROR: Your job was terminated because it exceeded the maximum allotted time for a job”, яке означає вичерпання ліміту допустимого часу на розв’язання задачі на NEOS Server, який становить 8 годин. Також в усіх таблицях жирним шрифтом виділено середній час роботи тієї комбінації “модель + додаткове обмеження”, якою кожна з програм розв’язувала тестову задачу найшвидше.

Модель	Gurobi 12.0.1			CPLEX 22.1.2		
	$t_{\min}$	t_{avg}	$t_{\max}$	$t_{\min}$	t_{avg}	$t_{\max}$
(13)–(20)	36714,000	45508,520	76622,900	—	—	—
(13)–(20), (51)	24848,100	28432,050	32241,200	—	—	—
(13)–(20), (52)	44529,000	52863,420	62180,200	—	—	—
(13)–(20), (51), (52)	27006,800	34136,520	71416,300	—	—	—
(21)–(30)	144,326	160,132	210,679	600,321	776,098	1043,700
(21)–(30), (51)	176,235	233,689	259,229	569,341	977,008	1645,030
(21)–(30), (52)	141,233	175,539	211,757	348,027	388,053	480,994
(21)–(30), (51), (52)	104,387	117,544	160,607	774,781	896,732	1089,890
(13)–(30)	204,784	283,309	340,106	608,103	763,597	843,597
(13)–(30), (51)	281,132	324,811	483,967	929,905	1006,915	1223,420
(13)–(30), (52)	229,715	307,178	395,912	981,682	1630,475	3033,460
(13)–(30), (51), (52)	297,990	388,469	495,191	742,829	1076,475	1654,560

Табл. 2. Результати обчислювальних експериментів з розв’язання тестової задачі пошуку оптимального циклу у графі *st70.tsp*.

Модель	Gurobi 12.0.1			CPLEX 22.1.2		
	$t_{\min}$	t_{avg}	$t_{\max}$	$t_{\min}$	t_{avg}	$t_{\max}$
(31)–(39)	26171,900	31523,410	39327,200	—	—	—
(31)–(39), (51)	65778,500	72314,280	84258,000	—	—	—
(31)–(39), (52)	29915,000	42678,690	82472,400	—	—	—
(31)–(39), (51), (52)	72925,900	85064,420	99255,600	—	—	—
(40)–(50)	187,116	203,304	227,641	621,096	827,530	1022,040
(40)–(50), (51)	177,911	183,262	201,259	1259,040	1538,684	3450,820
(40)–(50), (52)	235,282	260,869	366,143	4675,030	5952,134	7391,950
(40)–(50), (51), (52)	497,726	910,329	1082,310	2322,110	3962,063	5768,910
(31)–(50)	377,827	457,844	608,641	1878,790	2232,082	3238,490
(31)–(50), (51)	369,387	433,503	620,789	1968,610	3106,824	4384,070
(31)–(50), (52)	481,356	575,300	635,934	3528,070	4031,410	4445,410
(31)–(50), (51), (52)	102,646	114,360	126,952	1628,440	3041,426	3884,350

Табл. 3. Результати обчислювальних експериментів з розв’язання тестової задачі пошуку оптимального шляху у графі *st70.tsp*.

Acceleration of Gurobi and CPLEX when solving problems of finding the shortest cycle or path that visits a given number of vertices from the graph clusters

Mykola M. KORABLOV¹, Petro I. STETSYUK²

¹⁾ *Kyiv Academic University, Kyiv, Ukraine*

E-mail: m.m.korablov@gmail.com

ORCID: <https://orcid.org/0009-0007-3965-2857>

²⁾ *V.M. Glushkov Institute of Cybernetics of the NAS of Ukraine, Kyiv, Ukraine*

E-mail: stetsyukp@gmail.com

ORCID: <https://orcid.org/0000-0003-4036-2543>

Received April 20, 2025, in the final form June 7, 2025

<https://doi.org/10.3842/confKAU.2025.math.01>

Abstract. One of the most famous and most studied problems of operations research is the traveling salesman problem. Despite its simple formulation, it has found application in many different fields and to this day is the subject of active research, both related to the development of new methods and approaches to its solution and the creation of generalized problem statements and new mathematical models for their formalization. This paper is devoted to the study of the influence of two types of additional constraints on the speed of the Gurobi and CPLEX programs when solving a special class of routing problems. Formulations of problems of finding the shortest cycle and path that visits a given number of vertices from the graph clusters are presented. Mathematical models of mixed integer linear programming for their solution based on Miller, Tucker and Zemlin constraints and on Gavish and Graves constraints are described. The results of computational experiments are presented, which demonstrate the impact of additional constraints on the speed of solving problems using these models.

Keywords: generalized routing problems; travelling salesman problem; shortest cycle and path in a graph; mixed integer linear programming; AMPL; Gurobi; CPLEX; NEOS Server.